

Sabre Offer and Order APIs

Overview

Sabre Offer and Order APIs provide developers with the ability to integrate dynamic air content into applications from airlines implementing the IATA New Distribution Capability (NDC) standards.

The Sabre Offer and Order APIs expose NDC content in a consistent manner irrespective of the NDC schema version being used by the airline or the NDC technology partner providing the NDC solution to the airline. Some nuances do exist between the airline implementations, these are highlighted in this document.

This user guide provides descriptions of the APIs along with examples of requests and explanations of responses from various airlines. These examples are presented in a logical flow to assist developers as you expand capabilities within your applications. Most servicing functions are optional and apply to specific use cases and in some circumstances may be applied in any order, for example a seat assignment may be added before or after payment and fulfillment has occurred, but a re-shop and ticket exchange may only occur after fulfillment.

Table of Contents

Revision History	6
About the Sabre Offer and Order APIs	7
Getting Started.....	7
Offer and Order Workflows	8
Shopping, Pricing and Creating an Order	8
Servicing NDC Orders	8
Shopping using the Bargain Finder Max (BFM) API	9
The “offers/shop” (Bargain Finder Max) endpoint.....	9
Specifying Content Sources.....	10
Preferencing Duplicate Offers	11
Requesting Multiple Brands for the Same Itinerary.....	12
Maximum Number of Upsells	12
Requesting Offers with an Account Code.....	13
Shopping with a Passenger’s Frequent Flyer Number	14
Show One Ways/Offer by Bound.....	15
Existing BFM Options that may be used to Shop for NDC Offers.....	16
Time Window	16
Specifying the Preferred Cabin.....	17
Specifying a Preferred Cabin by Leg.....	18
Corporate ID (SNAP Code).....	19
Overriding the Default Currency	20
Specifying the Number of Stops and Connections.....	21
Requesting Offers that Include Free Checked Baggage	22
Reading a BFM Response with NDC Offers.....	23
Sections of the BFM Response	23
Using the “offers/price” API to Obtain Detailed Information a Specific Offer	31
The “offers/price” endpoint	31
Examples of the “offers/price” Request.....	32
Reading the “offers/price” Response	35
Sections of the “offers/price” response.....	35

Using the “orders/create” API to Create an Order	46
The “orders/create” endpoint	46
State Management.....	47
Data Elements Required to Create an Order:.....	50
Example of an “orders/create” Request.....	51
Passenger Identity Documents	54
Reading the “orders/create” Response	56
Sections of the “orders/create” Response	56
A PNR with an NDC Order	70
Adding a Fiscal Code in the “orders/create” Request.....	71
Displaying an Order using the “orders/view” API.....	72
The “orders/view” endpoint	72
Examples of the “orders/view” Request.....	73
Canceling an Order Prior to Fulfillment	74
The “orders/cancel” endpoint.....	74
Using the “orders/cancel” API to Cancel All Air Segments	75
The “orders/change” endpoint	77
Using the “orders/change” API to Cancel an NDC Order	78
Updating Passenger Information in an Order.....	79
Add Secure Flight Passenger Data (SFPD) to an Order	79
Add a Frequent Flyer Number to an Order	80
Add Passport Information to an Order	81
Add a Known Traveler Number (KTN) to an Order	82
Delete a Known Traveler Number (KTN) from an Order	83
Correcting a Passenger’s Name.....	84
Repricing an Order.....	86
The “offers/repriceOrder” endpoint	86
Using the “offers/repriceOrder” API to Reprice an Unfulfilled Order.....	87
Accepting a Repriced Offer and Updating the Order	96
Fulfilling an Order	97
Airline and Neutral Stock.....	98
Forms of Payment.....	98

Three Domain Security v2 (3DSv2)	98
Document Types and Codes.....	99
Coupon Status	99
Examples of the “orders/change” Request to Fulfill an Order	100
Cancelling a Fulfilled Order	109
The “offers/reshop/cancelOrder” endpoint.....	109
Checking if Documents are in the Void or Refund Period	110
Cancellation of a Fulfilled Order	114
Pre-reserved Seat Assignments.....	116
Displaying a Seat Map	116
The “offers/getseats” endpoint.....	116
Request a Seat Map with an Offer ID.....	117
Request a Seat Map with an Order ID.....	118
Reading the Seat Map Response	119
Booking and Cancelling Pre-Reserved Seats	127
Booking a Pre-Reserved Seat	127
Cancelling a Pre-Reserved Seat	128
Rebooking a Pre-Reserved Seat	129
Fulfilling a Pre-Reserved Seat.....	133
Examples of the “orders/change” Requests to Fulfill a Pre-Reserved Seat.....	133
Cancelling an Order Item for a Fulfilled Pre-Reserved Seat	138
Ancillary Services	140
Displaying a List of Ancillary Services.....	140
The “offers/getAncillaries” endpoint	140
Request a List of Ancillary Services with an Order ID	141
Reading the “getAncillaries” Response	142
Booking and Cancelling Ancillary Services.....	144
Booking Ancillary Services.....	144
Cancelling Ancillary Services.....	146
Voluntary Changes	147
Re-shopping for New Air Offers.....	147
The “offers/reshop/shop” endpoint.....	147

Examples of the “offers/reshop/shop” Request.....	148
Reading a Re-shop Response with NDC Offers.....	150
Sections of the “offers/reshop/shop” response.....	150
The Second Reshop Request.....	165
The “offers/reshop/price” endpoint.....	165
Example of the “offers/reshop/price” Request.....	166
Rebooking a Passenger.....	167
The “orders/change/exchange” endpoint.....	167
Exchanging a Ticket.....	170
Additional Servicing Capabilities.....	172
Unused Ticket Credits.....	173
Cancelling an Order and Retaining a Ticket for Future Use.....	173
Pricing an Offer with an Unused Ticket.....	176
Voiding Tickets.....	178
Voiding a Ticket and Retaining the Itinerary.....	178
Order Synchronization.....	179
The “orders/sync” endpoint.....	179
Synchronizing a Sabre Order with an Airline.....	180
Glossary of Terms.....	182
Abbreviations and Acronyms.....	182

Revision History

Date:	Version:	Updated by:	Details:
14 Jun 2023	1.0	NDC Product Manager (NH)	Draft of structure, sections and headings
14 Jul 2023	1.1	NDC Product Manager (NH)	1 st release with shopping section
09 Aug 2023	1.2	NDC Product Manager (NH)	Draft of second release with pricing and all headers
09 Aug 2023	1.3	NDC Product Manager (NH)	2 nd release with shopping and pricing sections
19 Oct 2023	1.4	NDC Product Manager (NH)	Order Management, Fulfilment, Cancellations and Seats
24 Oct 2023	1.51	NDC Product Manager (NH)	3 rd release
08 Feb 2024	1.6	NDC Product Manager (NH/KP)	4 th release

About the Sabre Offer and Order APIs

Getting Started

The Sabre Offer and Order APIs are implemented as RESTful JSON services, to gain access to these services you must request an access token using valid Sabre credentials. There are two types of tokens, ATK which is stateless and ATH which is stateful.

Important Note: If your application has a requirement to **work with an open session** or if your **application works in different points of sale** or **uses multiple Pseudo City Codes**, it may be more efficient to use an ATH token as it supports the “Change of Context” capability. If your application does not have any of these requirements, then an ATK token may be used.

To obtain credentials make a request through your Sabre Account Manager, for more information on how to obtain a token, refer to the documentation in Dev Studio.

This link provides information for the ATK stateless token.

[OAuth Token Create REST API | Dev Studio \(sabre.com\)](#)

This link provides information for the ATH stateful (session-based) token.

[Create Session | Dev Studio \(sabre.com\)](#)

In addition to an access token, some APIs require an Application Identifier. This identifier can be obtained from your Sabre Account Manager and its usage is described in Dev Studio.

[Sabre API Customer Application IDs | Dev Studio](#)

APIs that require an application ID:

- [“offers/getseats”](#) – display the seat map with pricing, if an application ID is not included you will only receive informational offers.
- [“orders/change”](#) – to book a pre-reserved seat.
- [“orders/change”](#) – to cancel or modify a pre-reserved seat.

Offer and Order Workflows

Shopping, Pricing and Creating an Order

There are three steps to create an order:

1. Shop for offers - the shopping request allows the agent to specify criteria such as the number and types of passengers, preferred cabin, airline account codes, frequent flyer numbers and other options that help to ensure the offers in the response are the most desirable for the passenger. Offers in the response combines schedule and pricing information in a single response.
2. Obtain detailed pricing information for a specific offer – select an offer and request more detailed information.
3. Create an order based on a priced offer – add the passenger data with the priced offer to confirm the booking with the airline and create an order and PNR in the Sabre system.

Servicing NDC Orders

The servicing of an order consists of any action performed after an order has been created, this may be any of the following:

- Adding or modifying passenger data in an order.
- Cancellation before an order has been fulfilled.
- Fulfillment of an order.
- Cancellation after an order has been fulfilled with a ticket in the void or refund period.
- Cancellation after an order has been fulfilled and retaining the ticket for future use.
- Displaying the seat map for the flights within an order.
- Adding or cancelling a pre-reserved seat assignment.
- Changing a pre-reserved seat assignment.
- Displaying a list of ancillary services.
- Adding an ancillary service.
- Fulfillment of a pre-reserved seat and/or ancillary service.
- Re-shopping an itinerary.
- Modifying an itinerary after an order has been fulfilled.
- Exchanging a ticket after an itinerary has been modified.

Shopping using the Bargain Finder Max (BFM) API

Bargain Finder Max (BFM) is a proprietary Sabre air shopping API, it has been enhanced to include NDC offers along with existing options to shop for fares filed in ATPCo and content from Low-Cost- Carriers. Many of the existing BFM shopping qualifiers have been enabled for NDC shopping requests and some new qualifiers, specific to NDC, have been added. This user guide details these qualifiers to help the developer determine the optimal and most appropriate options for your application.

The “offers/shop” (Bargain Finder Max) endpoint

BFM queries use the “post” method, the current version of BFM is v5.

Certification environment:

<https://api.cert.platform.sabre.com/v5/offers/shop>

Production environment:

<https://api.platform.sabre.com/v5/offers/shop>

Specifying Content Sources

To specify which content sources should be queried, the “DataSources” options are set to “Enable” or “Disable” in the the BFM request:

```
... "TravelPreferences": {  
  "VendorPref": [  
    {  
      "Code": "QF"  
    }  
  ],  
  "TPA_Extensions": {  
    "NumTrips": {  
      "Number": 50  
    },  
  },  
  "DataSources": {  
    "NDC": "Enable",  
    "ATPCO": "Disable",  
    "LCC": "Disable" ...  
  }  
}
```

If the “ATPCO” source is specified, BFM will return results from the Sabre system based on fares filed by airlines and distributed by ATPCo. If NDC is specified, the system will make external NDC shopping requests based on the following rules:

1. The airline operates the route in the BFM request.
2. The agent’s PCC is authorized for the airline(s) that operate the route.

Note: Multiple sources may be specified in a single BFM request.

Preferencing Duplicate Offers

It is possible to specify a single or multiple content sources, though this means that duplicate offers may be returned in the response. Offers are determined to be duplicates when the itinerary (flights) and the total price are the same. This can happen when carriers file fares in ATPCo and make those same fares available through their NDC connections. To specify the preferred content source when duplicates are returned the following options are available within "TravelPreferences".

- Preferencing NDC orders in the case of a tie:

```
}
"PreferNDCSourceOnTie": {
  "Value": true
}
```

In this case BFM will only return NDC offers when there is a duplicate.

- Preferencing ATPCo fares in the case of a tie:

```
}
"PreferNDCSourceOnTie": {
  "Value": false
}
```

Setting this value to false means BFM will only return ATPCo fares if there are duplicates.

- Receiving duplicate offers in a BFM response may be desirable if you have your own logic for determining how duplicate offers are managed or if you wish to present duplicates to the agent.

Set the "MultipleSourcePerItinerary" value as follows:

```
"TPA_Extensions": {
  "IntelliSellTransaction": {
    "RequestType": {
      "Name": "200ITINS"
    },
    "MultipleSourcePerItinerary": {
      "Value": true
    }
  }
}
```

In this case duplicate offers may be returned.

Note that when an agency requests access to NDC content they specify the default preference which is applied if no preference is included in the BFM request. The agency may modify this default at any time.

Requesting Multiple Brands for the Same Itinerary

If you want to receive multiple offers for the same itinerary, you can set the “MultipleBrandedFares” value to “true”, if this is set to “false” or not specified you will only receive the lowest fare for each itinerary. This option is set within a new section of the BFM request called “NDCIndicators” and is a TPA extension:

```
"TPA_Extensions": {
  "NumTrips": {
    "Number": 50
  },
  "DataSources": {
    "NDC": "Enable",
    "ATPCO": "Disable",
    "LCC": "Disable"
  },
  "PreferNDCSourceOnTie": {
    "Value": true
  },
  "NDCIndicators": {
    "MultipleBrandedFares": {
      "Value": true
    }
  }
}
```

Because the use of this parameter increases the number of offers returned, the BFM response may take longer and have a larger payload.

Maximum Number of Upsells

Upsells are additional price points for an itinerary that may have fewer restrictions for changes and refunds, they may also include entitlements such as additional checked bags and complimentary seat assignments, upsells may also include offers in higher cabins.

The “MaxNumberOfUpsells” value is implemented under “NDCIndicators” as per the example:

```
"NDCIndicators": {
  "MultipleBrandedFares": {
    "Value": true
  },
  "MaxNumberOfUpsells": {
    "Value": 6
  },
}
```

The number of offers returned is the lowest priced offer plus the maximum number of upsells, for example, if the maximum number of upsells is set to “6”, you may receive “7” offers. However, if the airline does not offer that many fares for a specific itinerary then the number of offers will be fewer.

Requesting Offers with an Account Code

If agencies have been provided with account codes by airlines these may be used to obtain private fares or corporate bundles.

The option to send account code information is by airline, it is implemented under “NDCIndicators” and “CarrierSpecificQualifiers” as an array:

```
{
  "CarrierSpecificQualifiers": [
    {
      "CarrierCode": "QF",
      "AccountCode": {
        "Code": "ABC50"
      }
    }
  ]
}
```

Shopping with a Passenger's Frequent Flyer Number

NDC introduces the concept of personalized offers, in the past frequent flyer tier level pricing only applied to ancillary services where, for example, premium seats were made available or offered for no charge to frequent flyers with certain tier level status. With NDC, airlines may customize any type of offer for frequent flyers, this could be in the form of discounts, reduced rate upgrades and/or additional entitlements such as lounge access and early boarding. To get access to these frequent flyer offers the frequent flyer account details and the passenger's name are included in the shopping request.

The Frequent Flyer information is implemented under "NDCIndicators", the values required are the airline designator and the Account Number (frequent flyer number). The passenger reference ID must be included and the same reference must be in the "TravellerInfoSummary".

Multiple frequent flyer accounts may be specified for different airlines.

```
"NDCIndicators": {
  "MultipleBrandedFares": {
    "Value": true
  },
  "MaxNumberOfUpsells": {
    "Value": 6
  },
  "FrequentFlyer": [
    {
      "AccountNumber": "7531246",
      "AirlineDesignator": "QF",
      "PassengerId": "Passenger1"
    }
  ]
},
"TravelerInfoSummary": {
  "AirTravelerAvail": [
    {
      "PassengerTypeQuantity": [
        {
          "Code": "ADT",
          "Quantity": 1,
          "PassengerId": "Passenger1",
          "PersonName": {
            "GivenName": "SMITH",
            "Surname": "JOHN"
          }
        }
      ]
    }
  ]
}
```

There are currently no indicators in the BFM response that a frequent flyer discount has been applied by an airline.

Show One Ways/Offers by Bound

This option enables one way offers to be returned for round trip journeys (sometimes referred to as sector fares, offer slicing or offers by bound) that can be used to create a single priced offer and a single order, this capability has been implemented in two different ways and it is up to the developer to determine which method to use.

The two methods are as follows:

- Include the “ShowOneWays” parameter in the BFM request which will return offers for each leg of the journey and include two different Offer IDs in the pricing request.
- Shop for one way offers and specify two or more Offer IDs from different BFM responses in the pricing request.

This example describes the first method, the second method is described in the [pricing workflow](#).

The “ShowOneWays” option is implemented under “NDCIndicators”:

```
"NDCIndicators": {  
  "MultipleBrandedFares": {  
    "Value": true  
  },  
  "MaxNumberOfUpsells": {  
    "Value": 6  
  },  
  "ShowOneWays": {  
    "Value": true  
  }  
}
```

Not all airlines support offers by bound, therefore, if this qualifier is used in the BFM request, the “itineraryType” will be set to “oneWay” if an offer is for a single bound.

```
"itineraryGroups": [  
  {  
    "groupDescription": {  
      "itineraryType": "oneWay",  
    }  
  }  
]
```

Existing BFM Options that may be used to Shop for NDC Offers

BFM is a mature API and has a lot of existing qualifiers, only a subset of these options are available in NDC shopping, this is due to limitations in the NDC schema or because they have not been implemented by the airlines or their NDC technology providers. To ensure the highest level of capabilities within BFM, Sabre has enabled many of these options by developing to the schema when they are available or by the application of internal logic to provide the same user experience.

Time Window

The time window options are used to find flights that depart and/or arrive within a specified time range. These options may be applied per leg in the BFM request. This capability brings efficiency as BFM will return a reduced set of offers that meet the search criteria.

```
"OriginDestinationInformation": [
  {
    "RPH": "1",
    "DepartureDateTime": "2023-06-09T00:00:00",
    "DepartureWindow": "09001000",
    "ArrivalWindow": "10301200",
    "OriginLocation": {
      "LocationCode": "SYD"
    },
    "DestinationLocation": {
      "LocationCode": "MEL"
    }
  }
]
```

Although this is a very useful capability which reduces the number of offers returned, if it is used without an understanding of the schedules then these qualifiers could result in less desirable options being returned, for example, if a flight takes one hour and the departure and arrival windows are set too far apart you may only receive connecting flights in the BFM response.

Specifying the Preferred Cabin

Cabin preference is commonly used in point-of-sale applications, if no cabin is specified the offers returned are likely to be the lowest priced and may not include premium cabins.

If a cabin is specified, higher cabins may also be returned, for example, if you request economy, you may also receive offers in business class and first class.

The preferred cabin option is implemented under “TravelPreferences\CabinPref”:

```
"TravelPreferences": {
  "VendorPref": [
    {
      "Code": "QF"
    }
  ],
  "CabinPref": [
    {
      "Cabin": "S",
      "PreferLevel": "Only"
    }
  ]
}
```

The cabin codes are:

- P - Premium First
- F - First
- J - Premium Business
- C - Business
- S - Premium Economy
- Y - Economy

Along with the preferred cabin is an option to enforce only the preferred cabin, this is called “KeepSameCabin”, this is a boolean data type implemented under “TPA_Extensions”:

```
"TPA_Extensions": {
  "NumTrips": {
    "Number": 50
  },
  "KeepSameCabin": {
    "Enabled": true
  }
}
```

If you set the “KeepSameCabin” value to “true” you may not receive offers if the requested cabin is not available for the entire journey. Most carriers do not offer all cabins on domestic and/or short haul flights as they use smaller aircraft with simpler seating configurations.

Specifying a Preferred Cabin by Leg

To search for offers with cabins by leg, add the “CabinPref” as a TPA extension for each origin and destination.

```
"OriginDestinationInformation": [  
  {  
    "RPH": "1",  
    "DepartureDateTime": "2024-04-13T00:00:00",  
    "DepartureWindow": "15551605",  
    "OriginLocation": {  
      "LocationCode": "MEL"  
    },  
    "DestinationLocation": {  
      "LocationCode": "SYD"  
    },  
    "TPA_Extensions": {  
      "CabinPref": {  
        "Cabin": "C"  
      }  
    }  
  },  
  {  
    "RPH": "2",  
    "DepartureDateTime": "2024-04-18T00:00:00",  
    "DepartureWindow": "13251335",  
    "OriginLocation": {  
      "LocationCode": "SYD"  
    },  
    "DestinationLocation": {  
      "LocationCode": "MEL"  
    },  
    "TPA_Extensions": {  
      "CabinPref": {  
        "Cabin": "Y"  
      }  
    }  
  }  
],
```

Note that if you shop with cabin by leg with the “ShowOneWays” option you may receive offers that are not combinable.

If the shop by cabin by leg option is used without the Keep Same Cabin option, you may also receive upsell options, e.g., on a route that operates economy class and business class, if you request economy, you will receive offers for both cabins.

Corporate ID (SNAP Code)

The Corporate ID (SNAP) code is an internal code and is used to look up airline account codes. This shopping qualifier supports both ATPCo and NDC fare sources, for NDC shopping requests, the system performs a “look-up” and based on the carriers that service the route, the system will send the appropriate account codes to each airline.

The Corporate ID is implemented as an array under “NegotiatedFareCode” in the “PriceRequestInformation” node:

```
... "PriceRequestInformation": {  
  "NegotiatedFareCode": [  
    {  
      "Code": "NDC23"  
    }  
  ]  
}...
```

Overriding the Default Currency

The currency override option allows pricing information to be returned in an alternative currency than the agent's point of sale. Note that once an order has been created it can only be fulfilled in the currency that was specified during the shopping step and the point of sale of the agent processing the fulfillment must support settlement in that currency.

Currency override is implemented under "PriceRequestInformation" under "TravelerInfoSummary":

```
"TravelerInfoSummary": {
  "AirTravelerAvail": [
    {
      "PassengerTypeQuantity": [
        {
          "Code": "ADT",
          "Quantity": 1
        }
      ]
    }
  ],
  "PriceRequestInformation": {
    "CurrencyCode": "GBP"
  }
},
```

Note: The most common use case for currency override is when the local currency fluctuates to the extent that secondary currencies are used for financial transactions.

Specifying the Number of Stops and Connections

Destinations that have limited service may have lots of connecting flights, but these may not be desirable as the total journey time may be inconvenient for the passenger. This capability provides the ability to specify the maximum number of stops and the maximum number of connections. These options are implemented under “TravelPreferences”.

```
"TravelPreferences": {
  "VendorPref": [
    {
      "Code": "QF"
    }
  ],
  "MaxStopsQuantity": 1,
  "FlightTypePref": {
    "MaxConnections": "1"
  }
}
```

In most cases, the number of stops and the number of connections are the same, except for direct flights that have a stop and continue with the same flight number. An example of this is the Qantas flight from SYD to LHR that has a stop in SIN with the same flight number for the entire journey, this will be returned if you set the “MaxStopsQuantity” to “1” and the “MaxConnections” to “0”.

Requesting Offers that Include Free Checked Baggage

To see offers that include free baggage allowance (FBA), use the “Baggage” option with type “RequestType” “A” for allowance and the “FreePieceRequired” boolean value to specify a free piece is required, this reduces the result set to offers that include at least one free checked bag. These options are implemented under “TravelPreferences”:

```
"TravelPreferences": {  
  "VendorPref": [  
    {  
      "Code": "AA"  
    }  
  ],  
  "Baggage": {  
    "RequestType": "A",  
    "FreePieceRequired": true  
  }  
}
```

Sections of the BFM Response

The BFM response utilizes a format called “Grouped Itinerary Response” (GIR), this method allows for the grouping of data elements with assigned reference numbers. These reference numbers are then used throughout the response instead of repeating large amounts of data, resulting in smaller payload sizes and faster response times.

The BFM response consists of the following sections:

- Version
- Messages
- Statistics
- Schedule Descriptions (scheduleDescs)
- Fare Component Descriptions (fareComponentDescs)
- Baggage Allowance Descriptions (baggageAllowanceDescs)
- Leg Descriptions (legDescs)
- Passenger Descriptions (passengerDescs)
- Price Class Descriptions (priceClassDescriptions)
- Itinerary Groups (itineraryGroups)

Note that there may be some additional sections in the BFM response if “ATPCO” is also enabled as a data source in the request.

Version

The version of the message response.

```
version": "6 6 4"
```

Messages

Informational messages from the airlines or from the service, this may include diagnostic data.

```
"messages": [  
  {  
    "severity": "Warning",  
    "type": "SUPPLIERPROFILE",  
    "code": "PROCESS",  
    "text": "Airlines not operating on this route: UA, SQ, SN, AV, LH, AA, QR, LX, OS"  
  },  
  {  
    "severity": "Warning",  
    "type": "NDCCONNECTOR",
```

```
    "code": "PROCESS",
    "text": "Supplier MH: No Availability"
  },
```

Statistics

This includes the number of itineraries in the response.

```
"statistics": {
  "itineraryCount": 21
},
```

Schedule Descriptions

As this is a Grouped Itinerary Response (GIR), the schedules will have IDs that are referenced in the “Leg Descriptions”, this allows any number of schedules to exist in an offer without being duplicated. Within the schedules are: the origin and destination airport codes, number of stops, departure and arrival date and times (with GMT offset), elapsed travel time, marketing carrier designator and flight number, operating carrier designator and flight number and the equipment code.

Notes:

1. The stop count will only be greater than zero for flights that have a stop but continue with the same flight number.
2. Additional information may be provided for schedules that apply to other fare sources such as ATPCo.

```
"scheduleDescs": [
  {
    "id": 1,
    "stopCount": 0,
    "eTicketable": true,
    "elapsedTime": 201,
    "departure": {
      "airport": "MIA",
      "time": "21:16:00-04:00"
    },
    "arrival": {
      "airport": "AUS",
      "time": "23:37:00-05:00"
    },
    "carrier": {
      "marketing": "AA",
      "marketingFlightNumber": 1077,
      "equipment": {
        "code": "738"
      }
    }
  }
],
```


Fare Component Descriptions

The fare component descriptions provide fare information for legs within the journey, as with all descriptions it has an ID that is referenced in the offers. Other data included is the price, currency code, the fare basis code (FBC), applicable passenger type code (PTC), description text of the fare type, fare type code (from the PADIS code set composite\data element 9910), Q Surcharge information, brand code, brand name and the reference ID for the price class description text.

```
"fareComponentDescs": [
  {
    "id":1,
    "fareAmount":274 42,
    "fareCurrency":"USD",
    "fareBasisCode":"NVAIZRK1",
    "farePassengerType":"ADT",
    "negotiatedFare":true,
    "fareTypeBitmap":"04",
    "fareDescription":"Negotiated fare",
    "fareCode":"749",
    "segments": [
      {
        "segment": {
          "surcharges": [
            {
              "amount":32 56,
              "currency":"USD",
              "type": "Q"
            }
          ]
        }
      }
    ],
    "brand": {
      "code":"AADOM-MAINSL",
      "brandName":"Main Select",
      "priceClassDescriptionRef":2
    }
  },

```

The fareTypeBitmap is calculated based on the following hexadecimal values:

- 0x00 - normal fare
- 0x01 - discounted fare
- 0x02 - fare by rule
- 0x04 - negotiated fare
- 0x08 - private fare

Notes:

1. These values may be combined.

Baggage Allowance Descriptions

The Baggage Allowance Descriptions provide a reference ID and a Piece Count, the reference ID is used in the offer under the “baggageInformation” section and allows an application to display the number of checked pieces applicable to each offer.

```
"baggageAllowanceDescs": [  
  {  
    "id": 1,  
    "pieceCount": 2  
  },  
]
```

Leg Descriptions

The Leg Descriptions have a reference ID, elapsed time value (in minutes) and a mapping to the schedule ID. The leg values are components of the offers, this mapping of schedules to legs is used to reduce repetition. The schedules are flight specific, whereas the legs are single values and may reference one or more schedules if connections are returned.

```
"legDescs": [  
  {  
    "id":1,  
    "elapsedTime": 378,  
    "schedules": [  
      {  
        "ref": 18  
      },  
      {  
        "ref": 20  
      }  
    ]  
  },  
]
```

Passenger Descriptions

The Passenger Descriptions have a reference ID, and passenger ID, additional optional elements that are based on the information entered in the query including passenger name(s) are also returned. The passenger reference IDs are returned in the offers section of the response and are used to identify pricing information at the passenger level.

```
"passengerDescs": [  
  {  
    "id": 1,  
    "passengerId": "Passenger1",  
    "givenName": "SMITH",  
    "surname": "JOHN"  
  },  
]
```

Price Class Descriptions

The Price Class Descriptions are free text and have a reference ID, that is used in the offers. The data that is returned in the description text comes directly from the airline's response.

```
"priceClassDescriptions": [
  {
    "id": 1,
    "descriptions": [
      {
        "text": "Our largest, most comfortable seats"
      },
      {
        "text": "2 free checked bags*"
      },
      {
        "text": "Priority security at participating airports"
      },
      {
        "text": "Priority Boarding"
      },
      {
        "text": "Same-day flight change and standby when available on American flights"
      },
      {
        "text": "If you are a AAdvantage Executive Platinum, or on a plane with 3 cabins, you
get 1 extra checked bag "
      }
    ]
  },
]
```

Notes:

1. United Airlines (UA) do not return Price Class Descriptions

Itinerary Groups

The "Itinerary Groups" contains the offers, starting with a recap of the itinerary and the reference of the leg, followed by the pricing information in an array. There are likely to be several offers for each leg if the multiple branded fares option is sent as "true".

```
"itineraryGroups": [
  {
    "groupDescription":
      {
        "legDescriptions": [
          {
            "departureDate": "2023-04-12",
            "departureLocation": "SAT",
            "arrivalLocation": "DFW"
          }
        ]
      },
    "itineraries": [

```

```

{
  "id":1,
  "pricingSource": "ADVJR1",
  "legs": [
    {
      "ref":24
    }
  ],
  "pricingInformation": [...

```

NDC offers returned in BFM responses are written to the Sabre offer store to allow for subsequent requests to reference Offer IDs. All Offer IDs in the BFM response are generated by the Sabre system and follow a consistent format irrespective of the airline or the airline’s NDC technology provider.

There are 3 sections of each offer:

- pricingSubSource
 - offer
 - fare
- The “pricingSubSource” is the system that is responsible for sourcing the offer, for NDC offers this value is “NDC_CONNECTOR”
 - The “offer” section contains the “offerId” from the Sabre offer store, the “timeToLive” which is the validity of the offer in seconds and the source, which is set to “NDC” for NDC offers.

Note that the “timeToLive” is how long the offer is valid, it does not guarantee inventory and a subsequent pricing request must be made before the offer expires. Most offers are valid for 20 minutes (1200 seconds). The “source” is the name of the data source and will state “NDC” for NDC offers, “ATPCO” for filed fares that are in the Sabre system and “LCC” for fares from “Low-Cost Carriers” that are returned from proprietary airline API responses.

- The “fare” section details the pricing information about each offer at the passenger level, this includes the base amount, total tax and the reference to the baggage allowance. Also within the fare element is the “offerItemID” that is used in the pricing request. References in this section map back to the fare components, baggage allowance and passenger information.

```

"pricingInformation": [
  {
    "pricingSubsource": "NDC_CONNECTOR",
    "offer": {
      "offerId": "ce059b9b846bg5pznlewx8x0-1",
      "timeToLive": 1200,
      "source": "NDC"
    },
    "fare": {
      "validatingCarrierCode": "AA",
      "eTicketable": true,
      "passengerInfoList": [

```

```

{
  "passengerInfo": {
    "offerItemId": "ce059b9b846bg5pznlewzxc8x0-1-1",
    "mandatoryInd": true,
    "serviceId": "ce059b9b846bg5pznlewzxc8x0-1-1-1",
    "passengerType": "ADT",
    "passengerNumber": 1,
    "fareComponents": [
      {
        "ref": 12,
        "beginAirport": "SAT",
        "endAirport": "DFW",
        "segments": [
          {
            "segment": {
              "bookingCode": "N",
              "cabinCode": "Y"
            }
          }
        ]
      }
    ],
    "passengerTotalFare": {
      "totalFare": 205 90,
      "totalTaxAmount": 28 23,
      "currency": "USD",
      "baseFareAmount": 177 67,
      "baseFareCurrency": "USD",
      "equivalentAmount": 177 67,
      "equivalentCurrency": "USD"
    },
    "baggageInformation": [
      {
        "provisionType": "A",
        "airlineCode": "AA",
        "segments": [
          {
            "id": 0
          }
        ],
        "allowance": {
          "ref": 2
        }
      }
    ],
    "passengers": [
      {
        "ref": 1
      }
    ]
  }
}

```

```
        }
      ]
    }
  ],
  "totalFare": {
    "totalPrice": 205 90,
    "totalTaxAmount": 28 23,
    "currency": "USD",
    "baseFareAmount": 177 67,
    "baseFareCurrency": "USD",
    "equivalentAmount": 177 67,
    "equivalentCurrency": "USD"
  }
},
```

Using the “offers/price” API to Obtain Detailed Information a Specific Offer

To obtain detailed information of an offer use the “offers/price” API with an “Offer Item ID” from the BFM response. The request may also contain credit card information, this allows the carrier to determine applicable credit card (OB) fees. Although this is not mandatory, some carriers will fail the fulfillment request if an alternative form of payment is specified.

The pricing request may also contain more than one “Offer Item ID” if you are requesting details of offers for multiple PTCs. This is also how the request should be made for carriers that support “Offers by Bound”. It is possible to include more than one “Offer Item ID” from two or more individual BFM responses, this allows for different types of journeys to be constructed, such as, round trips, circle trips and open jaws.

The “offers/price” endpoint

The “offers/price” API uses the “post” method, the current version of the “offers/price” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/offers/price>

Production environment:

<https://api.platform.sabre.com/v1/offers/price>

Examples of the “offers/price” Request

The basic “offers/price” request includes an “Offer Item ID” from the BFM response.

```
{
  "query": [
    {
      "offerItemId": [
        "ce574f65f78f1z5mglg2riuev0-20-1"
      ]
    }
  ]
}
```

The “Bank Information Number” (BIN), along with the card vendor code and card type (debit or credit) may be included in the pricing request, this ensures accurate OB fee (credit card fees) are applied.

```
{
  "query": [
    {
      "offerItemId": [
        "ce574f65f78f1z5mglg2riuev0-20-1"
      ]
    }
  ],
  "params": {
    "formOfPayment": [
      {
        "binNumber": "374299",
        "subCode": "FCA",
        "cardType": "AX"
      }
    ]
  }
}
```


If you shop for more than one passenger type, e.g., an adult (ADT) and a child (CNN), you will get back “Offer Item IDs” for each passenger type in the BFM response. When you call the “offers/price” API you will need to include the “Offer Item IDs” for each passenger from the BFM response.

```
{
  "query": [
    {
      "offerItemId": [
        "ce8559598499x9nsj1kk7pkb10-1-1",
        "ce8559598499x9nsj1kk7pkb10-1-2"
      ]
    }
  ]
}
```

If you shop for one way offers you will need to specify “Offer Item IDs” per bound as they are returned as separate Offers.

```
{
  "query": [
    {
      "offerItemId": [
        "ce8559598499b5kn41kuaaq4w0-1-1",
        "ce8559598499b5kn41kuaaq4w0-33-1"
      ]
    }
  ]
}
```

If you shop for multiple PTCs e.g., an ADT and CNN **and** specify one way offers, you will need to include all “Offer Item IDs” in the “offers/price” request for each PTC per bound.

```
{
  "query": [
    {
      "offerItemId": [
        "ce8559598499b5kn41kvc49gt0-1-1",
        "ce8559598499b5kn41kvc49gt0-1-2",
        "ce8559598499b5kn41kvc49gt0-161-1",
        "ce8559598499b5kn41kvc49gt0-161-2"
      ]
    }
  ]
}
```

The second method of pricing one-way offers is to use two “Offer Item IDs” from separate BFM responses, note the response ID component is different indicating that the “Offer Item IDs” are from different BFM responses.

```
{
  "query": [
    {
      "offerItemId": [
        "ce8559598499x9nsj1kvqcyaw0-51-1",
        "ce8559598499b5kn41kvqfrzx0-44-1"
      ]
    }
  ]
}
```

Reading the “offers/price” Response

Sections of the “offers/price” response

The “offers/price” response includes additional information relating to the offer and contains the following sections:

- Messages
- Id
- Version
- Response

Messages

Informational messages from the airlines or from the service, this may include diagnostic data.

```
"messages": [  
  {  
    "type": "WARNING",  
    "message": "Qantas Frequent Flyers could earn Qantas Points and Status Credits for this  
booking.  
<a href=\"https://www.qantas.com/fflyer/dyn/program/terms\" target=\"_blank\">Terms and condition  
s apply.</a>",  
    "service": "NDC_CONNECTOR"  
  }  
],
```

Id

The response ID forms the basis of “Offer IDs” and “Offer Item IDs”.

```
"id": "ce059d8ccbbc4rldmljr9urx35",
```

Version

The version of the message response.

```
"version": "v1 0 0",
```

Response

The response section contains the detailed information of the offer with the following sections:

Offers

- id - The Sabre “Offer ID” consisting of the response ID and the sequential number of the offer.

```
"id": "ce059d8ccbbc4rldmljr9urx35-1",
```

- ttl – The amount of time the offer is valid (time-to-live), inventory is not guaranteed, and the order must be created before the offer expires. Most offers are valid for 20 minutes (1200 seconds).

```
"ttl": 1200,
```

- source - The name of the data source this will state “NDC” for NDC offers.

```
"source": "NDC",
```

- offerExpirationDateTime - The time in GMT/Zulu time when the offer will expire.

```
"offerExpirationDateTime": "2023-07-06T15:17:56Z",
```

- paymentTimeLimitText - A text-based representation of when fulfillment is required, historically this has been called the “ticket time limit”, this field is only returned as a text field if the airline does not include a time zone in the response.

```
"paymentTimeLimitText": "2023-07-07T23:59:00",
```

- journeys - References to the flight segments for the offer and the “priceClassRefId” which is the brand code for that segment. “Journeys” and “segmentRefIds” are both implemented as arrays as there may be more than one segment for connections and each set of segments could have different brand codes.

```
"journeys": [  
  {  
    "segmentRefIds": [  
      "Isgm01008eaa6d66b"  
    ],  
    "priceClassRefId": "MainCabin"  
  }  
],
```

- offerItems - Contains the following sections and is implemented as an array.

```
"offerItems": [  
  {...
```

- id – The “Offer Item ID”, a unique identifier, multiple “Offer Item IDs” will be in the response if multiple PTCs were priced.

```
"id": "ce8559598499x9nsj1kvqy5f76-1-1",
```

- mandatoryInd – States whether “Offer Items” can be booked individually or if it is mandatory to include all Offer Items in the “orders/create” request.

```
"mandatoryInd": true,
```

- Passengers – the passengers array contains the following elements:

```
"passengers": [  
  {
```

- id – a unique ID used to reference a specific passenger.

```
"id": "Passenger1",
```

- ptc – the PTC that the airline used to price the offer.

```
"ptc": "ADT",
```

- requestedPtc – the PTC that was requested (this may vary from the PTC that the airlines used to price the offer if there are no fares applicable to the requested PTC).

```
"requestedPtc": "ADT",
```

- baggage – an array of information relating to free baggage allowance, this may include checked and carry-on baggage, the number of pieces (quantity) and the segment references to which the allowance applies. Note that sometimes when there is no free baggage allowance the airline may still return the details, but the quantity will be “0”.

```
"baggage": [  
  {  
    "type": "CarryOnBag",  
    "applicablePartyText": "Traveler",
```

```

    "segments": [
      "Isgm5200349cc2a79"
    ],
    "details": [
      {
        "quantity": 2
      },
      {
        "quantity": 1,
        "applicableBagText": "CARRY ON UP TO 45 LI 115 LCM"
      },
      {
        "quantity": 1,
        "applicableBagText": "PERSONAL ITEM UP TO 40 LI"
      }
    ]
  },
  {
    "type": "CheckedBag",
    "applicablePartyText": "Traveler",
    "segments": [
      "Isgm5200349cc2a79"
    ],
    "details": [
      {
        "quantity": 0,
        "constrains": [
          {
            "max": 23,
            "unit": "KG"
          },
          {
            "max": 50,
            "unit": "LBS"
          }
        ]
      },
      {
        "quantity": 0,
        "applicableBagText": "UPTO50LB 23KG AND62LI 158LCM"
      }
    ]
  }
],

```

- price – the total amount, base amount (fare), total taxes and the tax breakdown with the tax code and a text description.

```

"price": {
  "totalAmount": {
    "amount": "275.90",
    "curCode": "USD"
  },
  "baseAmount": {
    "amount": "242.79",
    "curCode": "USD"
  },
  "taxes": {
    "total": {
      "amount": "33.11",
      "curCode": "USD"
    },
    "breakdown": [
      {
        "amount": {
          "amount": "18.21",
          "curCode": "USD"
        },
        "nation": "US",
        "taxCode": "US",
        "description": "U.S.A Transportation Tax"
      },
      {
        "amount": {
          "amount": "4.80",
          "curCode": "USD"
        },
        "nation": "US",
        "taxCode": "ZP",
        "description": "United States Flight Segment Tax Domestic"
      },
      {
        "amount": {
          "amount": "5.60",
          "curCode": "USD"
        },
        "nation": "US",
        "taxCode": "AY",
        "description": "United States Passenger Civil Aviation Security Service Fee"
      },
      {
        "amount": {
          "amount": "4.50",
          "curCode": "USD"
        }
      }
    ]
  }
}

```

```

    },
    "nation": "US",
    "taxCode": "XF",
    "description": "US Passenger Facility Charge"
  }
]
}
}

```

- fareComponents – contains: the pricing information of a portion of an itinerary between two fare construction points, the fare basis code, the fare type code, reservations booking designator (RBD), cabin code and name, fare rules, ticketing endorsements, flight segment information, the brand code, name and description text.

```

"fareComponents": [
  {
    "price": {
      "baseAmount": {
        "amount": "242.79",
        "curCode": "USD"
      },
      "taxes": {
        "total": {
          "amount": "33.11",
          "curCode": "USD"
        },
        "breakdown": [
          {
            "amount": {
              "amount": "18.21",
              "curCode": "USD"
            },
            "nation": "US",
            "taxCode": "US",
            "description": "US Transportation Tax"
          },
          {
            "amount": {
              "amount": "4.50",
              "curCode": "USD"
            },
            "nation": "US",
            "taxCode": "XF",
            "description": "US Passenger Facility Charge"
          }
        ]
      }
    }
  }
]

```



```

        "amount": {
            "amount": "4.80",
            "curCode": "USD"
        },
        "nation": "US",
        "taxCode": "ZP",
        "description": "United States Flight Segment Tax Domestic"
    },
    {
        "amount": {
            "amount": "5.60",
            "curCode": "USD"
        },
        "nation": "US",
        "taxCode": "AY",
        "description": "United States Passenger Civil Aviation Security Service Fee"
    }
]
}
},
"fareBasis": {
    "fareBasisCode": "VUAIZRUZ",
    "fareDescription": "Published fare",
    "fareCode": "70J",
    "fareBasisCityPair": "DEWORDAA",
    "rbd": "V",
    "cabinType": {
        "cabinTypeCode": "Y",
        "cabinTypeName": "ECONOMY"
    },
    "sabreCabinType": {
        "cabinTypeCode": "Y",
        "cabinTypeName": "Economy"
    }
},
"fareRules": {
    "penalty": {
        "cancelFeeInd": false,
        "changeFeeInd": false,
        "refundableInd": true,
        "details": [
            {
                "penaltyType": "Cancel",
                "metadata": [
                    {
                        "ruleId": "Cancel",
                        "values": [

```

```

                "instruction": "Allowed"
            }
        ]
    }
}
],
{
    "penaltyType": "Change",
    "metadata": [
        {
            "ruleId": "Change",
            "values": [
                {
                    "instruction": "Allowed"
                }
            ]
        }
    ]
}
]
}
},
"segments": [
{
    "id": "Isgm5200349cc2a79",
    "rbd": "V",
    "flightNumber": "1559",
    "marketingCarrier": "AA",
    "cabinType": {
        "cabinTypeCode": "Y",
        "cabinTypeName": "ECONOMY"
    },
    "departure": {
        "airport": "DFW",
        "date": "2023-09-17T14:55:00",
        "terminal": "0"
    },
    "arrival": {
        "airport": "ORD",
        "date": "2023-09-17T17:23:00"
    },
    "duration": "PT2H28M",
    "equipment": "738"
}
],
"brand": {
    "code": "AADOM-MAINFL",
    "brandName": "Main Cabin Flexible",

```

```

    "descriptions": [
      {
        "text": "Carry-on & personal item (access to overhead bin)"
      },
      {
        "text": "Choose your seat (fee may apply)"
      },
      {
        "text": "Eligible for upgrades on American Flights"
      },
      {
        "text": "Flight changes allowed (fee may apply)"
      },
      {
        "text": "General Boarding"
      }
    ]
  }
]

```

- price – the total price of this offer item, this is separated as the pricing in the “passengers” section is per passenger for a specific PTC.

```

"price": {
  "totalAmount": {
    "amount": "275.90",
    "curCode": "USD"
  }
}

```

- totalPrice - The total price of all Offer Items

```

"totalPrice": {
  "totalAmount": {
    "amount": "275.90",
    "curCode": "USD"
  }
}

```

Note that offer items are specific to a PTC, if you shop for multiple PTCs you will need to send unique offer item IDs in the pricing request and you will receive an offer item ID for each PTC in the response. If you have multiple passengers with the same PTC then you will receive a single offer item ID for all passengers.

- “obFees” - this is a conditional element and will only be displayed if a vendor code and BIN was used in the request and if the carrier files credit card surcharges. OB fees may not apply to all

passenger type codes, there is a passenger reference within this section to allow you to determine which passenger(s) the OB fee applies to.

```
"obFees": [
  {
    "cardType": "FC5",
    "surcharge": {
      "amount": {
        "amount": "3.50",
        "curCode": "AUD"
      },
      "baseAmount": {
        "amount": "3.18",
        "curCode": "AUD"
      },
      "taxes": {
        "total": {
          "amount": "0.32",
          "curCode": "AUD"
        },
        "breakdown": [
          {
            "amount": {
              "amount": "0.32",
              "curCode": "AUD"
            },
            "nation": "AU",
            "taxCode": "UO",
            "description": "GOODS AND SERVICES TAX GST"
          }
        ]
      },
      "paxRefs": [
        "Passenger1"
      ]
    }
  ]
}
```

Note that the “offers/price” response may also contain “Filing Information” and “Surcharges” if they are applicable and returned by the airline:

```
"filingInformation": {
  "baseAmount": {
    "amount": "875 00",
    "curCode": "SGD"
  }
}
```

```
}  
}
```

```
"surcharges": [  
  {  
    "total": {  
      "amount": "33 46",  
      "curCode": "AUD"  
    },  
    "fees": [  
      {  
        "amount": {  
          "amount": "33 46",  
          "curCode": "AUD"  
        },  
        "localAmount": {  
          "amount": "30 00",  
          "curCode": "SGD"  
        },  
        "designator": "K"  
      }  
    ]  
  }  
],
```

Using the “orders/create” API to Create an Order

Orders are created using the “orders/create” API, this process instructs the airline to confirm the flights and complete the reservation in their system. Once the booking has been confirmed, the airline replies with details of the order, upon receipt of the successful order creation response an order is created in the Sabre Order Management System (OMS) and a PNR is created in the Sabre system.

The “orders/create” endpoint

The “orders/create” API uses the “post” method, the current version of the “orders/create” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/orders/create>

Production environment:

<https://api.platform.sabre.com/v1/orders/create>

State Management

One of the first things to consider when creating an order is whether the order should be created as “stateful” or “stateless”. If the agent has an open session with an ATH token, the application can add NDC air content into that session. This may be an existing PNR or a PNR that has not yet been committed (End Transaction).

Under “transactionOptions” you can specify the “requestType” as “stateful”, in this case the three boolean options under “transactionOptions” are processed:

```
{
  "transactionOptions": {
    "requestType": "STATEFUL",
    "commitTransaction": true,
    "initialIgnore": false,
    "movePassengerDetails": true
  },
}
```

- commitTransaction
- initialIgnore
- movePassengerDetails

Note that if the “requestType” is specified as “STATELESS” these options are not processed.

There are several considerations when interacting with the agent’s session and the use of the transaction options:

If the “commitTransaction” value is set to “true”, the system will attempt to “End” the PNR.

If the “initialIgnore” value is set to “true”, the system will ignore any data in the agent’s session and will create a new PNR.

If the “initialIgnore” value is set to “false” then details of the order are added to the PNR (partial or complete) in the agent’s session.

If the “initialIgnore” value is set to “false” then the system will check to ensure the details in the order are compatible with the data in the PNR.

If the “movePassengerDetails” value is set to “true” then any passenger data included in the “orders/create” request is automatically added to the PNR, if this value is set to “false” this data is not synchronized back to the PNR. This option avoids having duplicate data in the PNR that may be required by the airline in the order.

Data elements that are synchronized:

- Passenger Name (will not be added if the passenger name matches)
- Passenger Contact Details
- Email Address
- Loyalty\Frequent Flyer information
- APIS data (DOCS)

If the name in the PNR does not match the name in the “orders/create” request and the “commitTransaction” value is set to “true” the following error is received:

```
{
  "errors": [
    {
      "code": "060160",
      "message": "Order passenger data not in sync with session passenger data"
    }
  ]
}
```

If the number of passengers in the PNR is different to the number of passengers in the order and the “commitTransaction” value is set to “true” the following error is received:

```
{
  "errors": [
    {
      "code": "500700",
      "message": "ET PROCESSING ERROR NUMBER OF NAMES NOT EQUAL TO RESERVATIONS"
    }
  ]
}
```

If the number of passengers in the PNR is different to the number of passengers in the order and the “commitTransaction” value is set to “false” details of the order are added to the active session, but the agent will receive an error at “end transaction” and will need to correct this before ending the PNR.

Example PNR:

```
*A<<
1.1ALPHA/ALLEN
1 QF 612E 16MAY J MELBNE HK1 155P 405P /NDQF*PCUK72/QF0816E9D1C92
2 QF 545E 16MAY J BNESYD HK1 510P 645P /NDQF*PCUK72/QF0816E9D1C92
3 VA 842Y 02JUN T SYDMEL SS2 100P 235P /DCVA /E
TKT/TIME LIMIT
1.TL1159P/16APR-THU
PHONES
1.DFW1231231
PASSENGER EMAIL DATA EXISTS *PE TO DISPLAY ALL
CUSTOMER NUMBER - 123456
```



```
RECEIVED FROM - ORDER EXISTS - SABRE ORDER ID 1SXXXCJRCTXJ5  
G7RE.G7RE*AWW 1451/16APR20
```

E<<

NUMBER OF NAMES NOT EQUAL TO RESERVATIONS

If an agent creates an order and attempts to add the order to a PNR that already contains an NDC order the following error message is returned.

```
{  
  "errors": [  
    {  
      "code": "060159",  
      "message": "This is not a valid OrderCreateRequest as reservation data already exists in  
database."  
    }  
  ]  
}
```

A future enhancement will allow for two NDC orders to exist in the same PNR.

If the “commitTransaction” value is set to false and the agent adds a received from field, this will override the received from field created by the OMS and the Sabre Order ID will not be saved in the PNR.

If the “commitTransaction” value is set to false and the agent ignores the PNR, then the order remains in the airline’s system and in the Sabre OMS, however, it is not possible to merge that order back into a PNR. The Sabre system will cancel this order after a short period of time.

Data Elements Required to Create an Order:

The “orders/create” API has mandatory and optional data elements:

Mandatory elements:

- Transaction Options
- An Offer ID from an “offers/price” response
- The Offer Item ID from the Offer ID
- A Telephone Contact
- Passenger Given Name
- Passenger Surname

Optional elements:

- Passenger Middle Name
- Passenger Title
- Passenger Suffix
- Passenger Date of Birth (this is a component of Secure Flight Passenger Data and may only be added at time of order creation for some carriers)
- Passenger Gender (this is a component of Secure Flight Passenger Data and may only be added at time of order creation for some carriers)
- Passenger Frequent Flyer account information
- Passenger Email Address
- Passenger Passport information
- Passenger Visa information
- Passenger Redress number
- Passenger Known Traveler information (TSA Precheck, Global Entry, NEXUS and SENTRI)
- Customer Number (not passed to airlines)
- Remarks (not passed to airlines)

Note: Some airlines require “Secure Flight Passenger Data” (SFPD) to issue tickets, primarily this is for flights to, from or within the United States of America.

Note: Passenger data is considered personal identifiable information and it is the responsibility of the developer to ensure the application is compliant will all applicable data privacy laws regarding the collection, storage and transmission of this information.

Example of an “orders/create” Request

This example is a single passenger “orders/create” request, it contains the mandatory data elements and some optional elements including the passenger’s date of birth and gender. These two elements are components of Secure Passenger Flight Data (SPFD) and are mandatory for fulfillment for airlines that operate to, from or within the U.S.

```
{
  "transactionOptions": {
    "requestType": "STATEFUL",
    "commitTransaction": true,
    "movePassengerDetails": true,
    "initialIgnore": true
  },
  "createOrders": [
    {
      "offerId": "ce8559598499x9nsjlllobppt91-1",
      "selectedOfferItems": [
        {
          "id": "ce8559598499x9nsjlllobppt91-1-1"
        }
      ]
    }
  ],
  "contactInfos": [
    {
      "id": "CI-1",
      "phones": [
        {
          "number": "5551231234"
        }
      ]
    }
  ],
  "passengers": [
    {
      "id": "Passenger1",
      "typeCode": "ADT",
      "contactInfoRefId": "CI-1",
      "birthdate": "1976-09-02",
      "genderCode": "M",
      "givenName": "BRIAN",
      "surname": "BRAVO",
      "loyaltyProgramAccounts": [
        {
          "id": "LY-1",

```

```

        "accountNumber": "ABCD1234",
        "carrier": {
            "carrierCode": "AA"
        }
    },
    ],
    "remarks": [
        {
            "id": "RM-1",
            "text": "TEST REMARK"
        }
    ],
    "identityDocuments": [
        {
            "id": "ID-1",
            "documentNumber": "157999443",
            "documentTypeCode": "K",
            "issuingCountryCode": "US"
        }
    ]
}
],
"customerNumber": {
    "number": "2345678"
}
}

```

Note: The “Offer ID” and “Offer Item ID(s)” are from the “offers/price” response, an order cannot be created with “Offer IDs” and “Offer Item IDs” from a BFM response.

Note: The “orders/create” request must be sent before the “time-to-live” for the offer has expired.

Note: Some airlines validate that the agency Pseudo City Code (PCC) that is requesting the creation of the order is the same as the PCC that was used in the pricing request.

Note: It is not possible to include payment information in the “orders/create” request, the instant payment process may be enabled in a future release.

Note: Only offers for airfares are supported in the “orders/create” request, ancillary services such as seats, bags and meals can only be added after an order has been created, this limitation may be addressed in a future release.

If you are creating an order for multiple passengers from a priced offer with different passenger type codes, you must include all Offer Item IDs from the pricing response:

```
... "createOrders": [  
  {  
    "offerId": "ce0d47db58874whqx1nlv6m3d7-1",  
    "selectedOfferItems": [  
      {  
        "id": "ce0d47db58874whqx1nlv6m3d7-1-1"  
      },  
      {  
        "id": "ce0d47db58874whqx1nlv6m3d7-1-2"  
      }  
    ]  
  }  
], ...
```

Passenger Identity Documents

There are various types of “passenger identity documents” and airlines require this information prior to passengers travelling.

Valid identity document codes that may be used in the “orders/create” API are:

- PT – Passport

```
...
"identityDocuments": [
  {
    "id": "ID-1",
    "documentNumber": "0123456789",
    "documentTypeCode": "PT",
    "issuingCountryCode": "US",
    "placeOfIssue": "GB",
    "citizenshipCountryCode": "US",
    "residenceCountryCode": "US",
    "titleName": "Mr",
    "givenName": "BRIAN",
    "middleName": "GEORGE",
    "surname": "BRAVO",
    "suffixName": "Jr.",
    "birthdate": "1975-09-01",
    "genderCode": "M",
    "issueDate": "2011-12-20",
    "expiryDate": "2025-12-30",
    "hostCountryCode": "US"
  }
]
...
```

- V – Visa

```
...
"identityDocuments": [
  {
    "documentNumber": "V1234567",
    "documentTypeCode": "V",
    "issuingCountryCode": "PL",
    "residenceCountryCode": "PL",
    "issueDate": "2022-04-11",
    "expiryDate": "2025-05-28",
    "birthdate": "1982-07-11",
  }
]
```

```
    "genderCode": "F",
    "givenName": "CHLOE",
    "surname": "CHARLIE",
    "birthplace": "LONDON",
    "visaHostCountryCode": "US",
    "stayDuration": "5",
    "entryQuantity": "2"
  }
]
...
```

- **R – Redress Number**

```
...
"identityDocuments": [
  {
    "id": "ID-1",
    "documentNumber": "157999443",
    "documentTypeCode": "R",
    "issuingCountryCode": "US"
  }
]
...
```

- **K – Known Traveler Number**

```
...
"identityDocuments": [
  {
    "id": "ID-1",
    "documentNumber": "157999777",
    "documentTypeCode": "K",
    "issuingCountryCode": "US"
  }
]
...
```

Reading the “orders/create” Response

Sections of the “orders/create” Response

After an order has been created using the “orders/create” API, or if an order is updated or modified using the “orders/change” API, a full display of the order is returned.

The top elements are “order” and “warnings”, the top level of the order contains summary information including the Sabre Order ID, the Sabre record locator and the date the order was created.

The hierarchical structure of an order is defined as:

- Order
 - Order Items
 - Service items

Multiple order items may exist within an order and multiple service items may exist within an order item.

```
{  
  "order": {  
    "id": "1SXXXCKXV5IES",  
    "type": "ORDER",  
    "pnrLocator": "EXOFLJ",  
    "pnrCreateDate": "2023-08-23",  
    "offerVendor": "AA",  
    "orderOwner": "1S",  
    "partition": "AA",  
    "primeHost": "1S",  
    "countryCode": "US",  
    "orderItems": [  
    ...  
  ]  
}
```


The “orderItems” array contains detailed information of the order starting with the internal and external IDs and the creation date and time of the Order Item.

```
...
"orderItems": [
  {
    "id": "6F4A9D9E-2A12-4FB7-89A6-9C14911540A4",
    "externalId": "PoIP4D059364-B45D-41DE-A78E-1-1",
    "externalOrderRefId": "C7AF9CE1-6986-47A3-B35D-32AA18EC9B89",
    "origin": "N",
    "creationDateTime": "2023-08-23T22:54:50Z",
    "fareDetails": [
...

```

The “fareDetails” array contains pricing data per passenger, below the fare details are the price and fare components.

The “price” is the full price of the order item for a specific passenger and includes the base amount and a full breakdown of all applicable taxes.

```
...
"fareDetails": [
  {
    "fareIndicatorCode": "0",
    "paxRefIds": [
      "Passenger1"
    ],
    "price": {
      "baseAmount": {
        "amount": "64.19",
        "code": "USD"
      },
      "totalAmount": {
        "amount": "83.90",
        "code": "USD"
      },
      "totalTaxAmount": {
        "amount": "19.71",
        "code": "USD"
      },
      "taxBreakdowns": [
        {
          "amount": {
            "amount": "4.81",
            "code": "USD"
          }
        }
      ]
    }
  }
]

```

```

        "taxCode": "US",
        "description": "U.S.A Transportation Tax"
    },
    {
        "amount": {
            "amount": "4.80",
            "code": "USD"
        },
        "taxCode": "ZP",
        "description": "United States Flight Segment Tax Domestic"
    },
    {
        "amount": {
            "amount": "5.60",
            "code": "USD"
        },
        "taxCode": "AY",
        "description": "United States Passenger Civil Aviation Security Service Fee"
    },
    {
        "amount": {
            "amount": "4.50",
            "code": "USD"
        },
        "taxCode": "XF",
        "description": "US Passenger Facility Charge"
    }
    ]
},
...

```

The “fareComponents” section provides pricing information at the “leg” level, this may include one or more flight segments including connections and/or stopovers that are not considered as a fare break point. There may be one or more fare components within an order item. The following information is included within the fare component:

- Fare Basis Code.
- Fare Description - based on the fare description returned by the airline or per the coded definition.
- Fare Code - per the definition in the PADIS code set 9910, not returned by all airlines.
- Fare Type - Public or Private.
- Fare Basis City Pair – the start and end airport codes of the fare component and the airline code.
- The Cabin Code.
- The Cabin Name.
- Pricing data for the fare component, including base amount and tax breakdown.
- The Price Class Reference – this reference provides a link to the “brand” information.
- The Segment Reference array – a link to the flight segments that comprise this fare component.

```

"fareComponents": [
  {
    "fareBasisCode": "QVAIXSBX",
    "fareDescription": "Published fare",
    "fareCode": "70J",
    "fareRecognizedType": "Public",
    "fareBasisCityPair": "LAXPHXAA",
    "cabinTypeCode": "Y",
    "cabinTypeName": "Economy",
    "price": {
      "baseAmount": {
        "amount": "64.19",
        "code": "USD"
      },
      "totalAmount": {
        "amount": "83.90",
        "code": "USD"
      },
      "totalTaxAmount": {
        "amount": "19.71",
        "code": "USD"
      },
      "taxBreakdowns": [
        {
          "amount": {
            "amount": "4.81",
            "code": "USD"
          },
          "taxCode": "US",
          "description": "US Transportation Tax"
        },
        {
          "amount": {
            "amount": "4.50",
            "code": "USD"
          },
          "taxCode": "XF",
          "description": "US Passenger Facility Charge"
        },
        {
          "amount": {
            "amount": "4.80",
            "code": "USD"
          },
          "taxCode": "ZP",
          "description": "United States Flight Segment Tax Domestic"
        }
      ]
    }
  }
]

```

```

    },
    {
      "amount": {
        "amount": "5.60",
        "code": "USD"
      },
      "taxCode": "AY",
      "description": "United States Passenger Civil Aviation Security Service Fee"
    }
  ]
},
"priceClassRef": "0D8A9AD0-6079-4113-BC3A-6C2B08BCDC4B",
"segmentRefIds": [
  "Isgm3200c75e68bf7"
]
}
]

```

The “price” information within the order item, is the total price for that order item and it may include one or more passengers.

```

"price": {
  "baseAmount": {
    "amount": "64.19",
    "code": "USD"
  },
  "totalAmount": {
    "amount": "83.90",
    "code": "USD"
  },
  "totalTaxAmount": {
    "amount": "19.71",
    "code": "USD"
  }
},

```

The “services” array of the order displays a list of references, with the following:

- id – the Sabre service ID.
- externalId – the reference of the service from the airline.
- passengerRefId - the passenger reference this service applies to.
- segmentRefId – the reference of the segment for this service.
- serviceDefinitionRefId – the reference to the definition of this service.
- externalOrderItemId – the airline’s Order Item ID.

```

"services": [
  {
    "id": "C4DE036E-DABC-4E99-8935-B37E3B75A99E",
    "externalId": "BAGGAGESRV1-T1",
    "passengerRefId": "Passenger1",
    "segmentRefId": "Isgm3200c75e68bf7",
    "serviceDefinitionRefId": "260BC162-D89C-4DA1-A6AD-3745BA939606",
    "externalOrderItemId": "PoIP4D059364-B45D-41DE-A78E-1-1"
  },
  {
    "id": "B379780F-F768-424A-AC8B-AF1ACA4626CD",
    "externalId": "BAGGAGESRV2-T1",
    "passengerRefId": "Passenger1",
    "segmentRefId": "Isgm3200c75e68bf7",
    "serviceDefinitionRefId": "5558E4BF-BDEE-4CE7-881A-99992DD315F8",
    "externalOrderItemId": "PoIP4D059364-B45D-41DE-A78E-1-1"
  },
  {
    "id": "1806D5B5-57B7-42B8-8836-FAFABCD02A81",
    "externalId": "SRV1-T1-S2",
    "statusCode": "HK",
    "passengerRefId": "Passenger1",
    "segmentRefId": "Isgm3200c75e68bf7",
    "externalOrderItemId": "PoIP4D059364-B45D-41DE-A78E-1-1"
  }
],

```

Additional elements of the order item are returned:

- offerItemId – the Sabre Offer Item ID for this order item.
- externalOfferItemId – the airline’s Offer Item ID for this order item.
- feeWarnings – information messages returned by the airline relating to this order item.
- paymentTimeLimitText – the payment time limit for this order item returned by the airline, this is returned as a text string if the airline does not include a time zone for this data element.

```

"offerItemId": "ce8559598499x9nsjlllobppt91-1-1",
"externalOfferItemId": "PoIP4D059364-B45D-41DE-A78E-1-1",
"feeWarnings": [
  {
    "text": "Departure Time changed for AA1270 2023-09-21 LAXPHX",
    "owner": "AA"
  },
  {
    "text": "Arrival Time changed for AA1270 2023-09-21 LAXPHX",
    "owner": "AA"
  }
]

```

```
],  
"paymentTimeLimitText": "2023-08-24T23:59:00",  
"itemOrigin": "SELL"
```

This is the end of the order item(s) section of the order, the following elements apply to the entire order.

- **contactInfos** – contact information for the agency and/or passenger, this can include telephone numbers and email addresses, the ID is included in the passenger information to link one or more passenger to the contact.

```
"contactInfos": [  
  {  
    "id": "B9B2A11A-FEE5-4A1D-9F94-14A98A069E44",  
    "phones": [  
      {  
        "id": "C2F24E71-0F15-4EFB-9C2C-473E8FE655B5",  
        "number": "5551231234"  
      }  
    ]  
  }  
],
```

- **products** – a summary of the products within the order including the status.

```
"products": [  
  {  
    "id": "4f513d3d",  
    "airSegment": {  
      "marketingCarrier": {  
        "airlineCode": "AA",  
        "flightNumber": "1270"  
      },  
      "departureDateTime": "2023-09-21T13:05:00",  
      "arrivalDateTime": "2023-09-21T14:35:00",  
      "departureAirport": "LAX",  
      "arrivalAirport": "PHX",  
      "actionCode": "HK"  
    }  
  }  
],
```

- passengers – information relating to each passenger in the order along with the unique passenger reference ID.

This example includes how Secure Flight Passenger Data (SFPD) is returned at the passenger level, there is no identity document type for SFPD, the required information in addition to the passenger name is data of birth and gender code. Valid codes for gender are:

- F – Female
- M – Male
- X – Unspecified

Also within this order is an identity document type “K” for known traveler number and a frequent flyer number.

```
"passengers": [
  {
    "id": "Passenger1",
    "externalId": "T1",
    "typeCode": "ADT",
    "contactInfoRefIds": [
      "B9B2A11A-FEE5-4A1D-9F94-14A98A069E44"
    ],
    "contactInfoRefId": "B9B2A11A-FEE5-4A1D-9F94-14A98A069E44",
    "birthdate": "1976-09-02",
    "givenName": "BRIAN",
    "surname": "BRAVO",
    "genderCode": "M",
    "remarks": [
      {
        "id": "RM-1",
        "text": "TEST REMARK"
      }
    ],
    "identityDocuments": [
      {
        "id": "0B10281C-E2F0-4CCF-8497-1BDD255655B5",
        "documentNumber": "157999443",
        "documentTypeCode": "K",
        "issuingCountryCode": "US"
      }
    ],
    "loyaltyProgramAccounts": [
      {
        "id": "A00BDCD4-2E02-4E4C-8CA2-4A2DC07ACEAF",
        "accountNumber": "ABCD1234-1.1",
        "carrier": {
```

```

        "carrierCode": "AA"
      }
    }
  ]
}
],

```

- journeys – journey information is a leg within an itinerary that maps to a fare component. The example only has a single segment reference ID, but this could be more than one.

```

"journeys": [
  {
    "id": "FGTIDXIWHPGUPTZN1IFON2E0EHHRNVJT22B40BYFLHLLOLVGEO1DTLQMLE",
    "segmentRefIds": [
      "Isgm3200c75e68bf7"
    ]
  }
]
],

```

- segments – Flight segments within the order, note that operating carrier is only included if the marketing carrier and operating carrier differ. This section provides detailed information including arrival and departure terminals (when returned by the airline), equipment (aircraft type) and flight duration.

```

"segments": [
  {
    "id": "Isgm3200c75e68bf7",
    "cabinTypeCode": "B",
    "departure": {
      "locationCode": "LAX",
      "stationName": "Los Angeles International Apt, US",
      "terminalName": "0",
      "scheduledDateTime": "2023-09-21T13:05:00"
    },
    "arrival": {
      "locationCode": "PHX",
      "stationName": "Phoenix Sky Harbor Intl Apt, US",
      "terminalName": "4",
      "scheduledDateTime": "2023-09-21T14:35:00"
    },
    "marketingCarrier": {
      "carrierCode": "AA",
      "carrierName": "American Airlines",
      "flightNumber": 1270,
      "classOfService": "B"
    },
    "datedOperatingLegs": [
      {

```



```

    "departure": {
      "locationCode": "LAX",
      "stationName": "Los Angeles International Apt, US",
      "terminalName": "0",
      "scheduledDateTime": "2023-09-21T12:45:00"
    },
    "arrival": {
      "locationCode": "PHX",
      "stationName": "Phoenix Sky Harbor Intl Apt, US",
      "scheduledDateTime": "2023-09-21T14:12:00"
    },
    "carrierAircraftType": {
      "code": "738",
      "name": "Boeing 737-800 Passenger"
    }
  }
],
"flightDuration": "PT1H30M"
}
],

```

- priceClasses – the price class information is implemented as an array and provides the brand code, brand name and descriptive text about the brand, this information is returned by the airline and as a result will vary based on the carrier implementation. As brands may vary by bound, multiple price classes may be listed.

```

"priceClasses": [
  {
    "id": "0D8A9AD0-6079-4113-BC3A-6C2B08BCDC4B",
    "code": "AADOM-BASIC",
    "name": "Basic Economy",
    "descriptions": [
      {
        "text": "No changes allowed*"
      },
      {
        "text": "Pay to choose any seat on American flights (or we'll assign seats at no charge during check-in)**"
      },
      {
        "text": "Board in last group***"
      },
      {
        "text": "No Elite Qualifying Credits (EQMs, EQSs or EQDs)"
      },
      {
        "text": "* For flights booked on or after April 1, 2021."
      }
    ]
  }
]

```

```

    },
    {
      "text": "*** Complimentary for AAdvantage® elite members based on status."
    },
    {
      "text": "*** Doesn't apply to AAdvantage® elite status members or eligible AAdvantage®
credit cardmembers."
    },
    {
      "text": "No changes allowed*"
    },
    {
      "text": "Pay to choose any seat on American flights (or we'll assign seats at no
charge during check-in)**"
    },
    {
      "text": "Board in last group***"
    },
    {
      "text": "No Elite Qualifying Credits (EQMs, EQSs or EQDs)"
    },
    {
      "text": "* For flights booked on or after April 1, 2021."
    },
    {
      "text": "*** Complimentary for AAdvantage® elite members based on status."
    },
    {
      "text": "*** Doesn't apply to AAdvantage® elite status members or eligible
AAdvantage® credit cardmembers."
    }
  ]
}
],

```

- **customerNumber** – The customer number is an internal number used by agencies to identify a corporation, the data in this field is not sent to the airline, this is represented in the PNR as the DK number and is included in the Interface User Record (IUR) message that is sent to the agent's back-office system.

```

"customerNumber": {
  "number": "2345678"
},

```

- **paymentTimeLimitText** – the payment time limit for the airfare

```
"paymentTimeLimitText": "2023-08-24T23:59:00",
```

- externalOrders – external references including the airline’s Order ID and Record Locator

```
"externalOrders": [  
  {  
    "id": "C7AF9CE1-6986-47A3-B35D-32AA18EC9B89",  
    "systemId": "AA",  
    "externalOrderId": "AA001HL85Q4A2",  
    "bookingReferences": [  
      {  
        "id": "LL85Q4",  
        "carrierCode": "F1"  
      },  
      {  
        "id": "HDZZDX",  
        "carrierCode": "AA"  
      }  
    ]  
  }  
],
```

- totalPrice – the total price at the order level, for all order items.

```
"totalPrice": {  
  "totalAmount": {  
    "amount": "83.90",  
    "code": "USD"  
  }  
},
```

- remarks – agency remarks, informational comments, these are not sent to the airline.

```
"remarks": [  
  {  
    "id": "4d773d3d",  
    "text": "ORDER CREATED",  
    "type": "21"  
  },  
  {  
    "id": "4e773d3d",  
    "text": "TEST REMARK",  
    "type": "1"  
  }  
],
```

- **serviceDefinitions** – additional services that are included in the price, typically this is free baggage allowance (FBA).

```

"serviceDefinitions": [
  {
    "id": "5558E4BF-BDEE-4CE7-881A-99992DD315F8",
    "externalId": "BASRV42CF2B46-CO",
    "baggageAllowanceRefId": "COD68521-16E1-4845-B8F0-7F7FF7208DEB",
    "name": "Carry On Bag Allowance"
  },
  {
    "id": "260BC162-D89C-4DA1-A6AD-3745BA939606",
    "externalId": "BASRVD85D65A7",
    "baggageAllowanceRefId": "CB90B171-D382-430B-A717-A13D0886E015",
    "name": "Checked Bag Allowance"
  }
],

```

- **baggageAllowances** – detailed information for carry on and checked baggage allowances including the permitted quantities.

```

"baggageAllowances": [
  {
    "id": "COD68521-16E1-4845-B8F0-7F7FF7208DEB",
    "externalId": "BA6D08155B-CO",
    "typeCode": "CarryOn",
    "baggageDeterminingCarrier": {
      "carrierCode": "AA"
    },
    "applicableParty": "Traveler",
    "pieceAllowances": [
      {
        "applicableParty": "Traveler",
        "totalQuantity": 1
      },
      {
        "applicableParty": "Traveler",
        "totalQuantity": 1
      }
    ]
  },
  {
    "id": "CB90B171-D382-430B-A717-A13D0886E015",
    "externalId": "BA45FDC9FC",
    "typeCode": "Checked",
    "baggageDeterminingCarrier": {
      "carrierCode": "AA"
    }
  }
],

```

```

    },
    "applicableParty": "Traveler",
    "pieceAllowances": [
      {
        "applicableParty": "Traveler"
      },
      {
        "applicableParty": "Traveler"
      }
    ]
  }
]

```

At the order level, the last piece of information is “warnings”, these can be from external or internal system processing and are mainly informational.

```

"warnings": [
  {
    "code": "059000",
    "message": "External system return warning: \n code:325 \n owner:AA \n message:Invalid
Frequent Traveler Account Number - AA ABCD1234"
  },
  {
    "code": "",
    "message": "Flight times have been updated"
  },
  {
    "code": "060548",
    "message": "Payment Time Limit does not contain timezone information"
  }
]

```

A PNR with an NDC Order

When an order is created the system will create a new PNR or add the segments to an existing PNR based on the instruction in the “orders/create” request, this enables agency workflows that are dependent on a PNR. Other content may be added to the PNR (flights, rental cars and hotels) and the PNR functions in the same way as other PNRs, however, NDC segments cannot be modified by use of command line entries, all modifications to an NDC order must be processed through the Offer and Order APIs.

```
*-BRAVO<<
1.1BRAVO/BRIAN
1 AA1270B 21SEP Q LAXPHX HK1 1305 1435 /NDAA*HDZZDX/AA001HL85Q4A2
TKT/TIME LIMIT
1.TL2359 /24AUG-THU
PHONES
1.DFW5551231234
CUSTOMER NUMBER - 2345678
REMARKS
1.TEST REMARK
RECEIVED FROM - ORDER EXISTS - SABRE ORDER ID 1SXXXCKXV5IES
Z9LK.Z9LK*ABD 1754/23AUG23 EXOFLJ H
```

The NDC segments in a PNR have the following attributes:

- The connectivity indicator “ND”, referencing the “New Distribution Capability”
- The airline’s record locator
- The airline’s order ID

Order ID’s follow a format defined by IATA, consisting of the following elements:

- Two-character industry designator (airline or system code), e.g. 1S for the Sabre system and SQ for Singapore Airlines
- Three-numeric airline accounting code or “XXX” for aggregators
- Eight-character unique identifier

Note that not all airlines have an Order Management System (OMS) and as a result they use the record locator from their legacy system in place of an Order ID.

It is not currently possible to add multiple NDC orders to a PNR, this restriction will be addressed in a future release.

Adding a Fiscal Code in the “orders/create” Request

Fiscal tax codes are used in Latin America for reporting and tax deduction purposes, the fiscal ID and Code are both required, the codes value by country are:

- Ecuador and Peru - RUC
- Argentina - CUIT/CUIL
- Bolivia - NIT

```
"passengers": [  
  {  
    "id": "Passenger1",  
    "typeCode": "ADT",  
    "contactInfoRefId": "CI-1",  
    "givenName": "KASIA",  
    "surname": "KILO",  
    "birthdate": "1982-05-25",  
    "genderCode": "F",  
    "identityDocuments": [  
      {  
        "id": "ID-1",  
        "documentNumber": "235581234",  
        "documentTypeCode": "FISCALID",  
        "documentSubTypeCode": "NIT"  
      }  
    ]  
  }  
]
```

Fiscal tax codes are optional; however, some airlines have a requirement that if one passenger has a fiscal tax code, then a code must be supplied for all passengers in the order, this should be the fiscal tax code of the person that is making the payment, i.e., the credit card holder.

Airlines that have this requirement:

- American Airlines (AA)
- Aero Mexico (AM)
- Lufthansa Group (LH, LX, OS and SN)
- United Airlines (UA)
- Qantas (QF)

Displaying an Order using the “orders/view” API

An order is displayed by calling the “orders/view” API, you can use the Sabre Order ID or the Sabre Record Locator. The response is the same as when an order is created or when any modification is made by the “orders/change” API.

The “orders/view” endpoint

The “orders/view” API uses the “post” method, the current version of the “orders/view” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/orders/view>

Production environment:

<https://api.platform.sabre.com/v1/orders/view>

Examples of the “orders/view” Request

To display an order by the Order ID:

```
{  
  "id": "1SXXXABCD1234"  
}
```

To display an order by the record locator:

```
{  
  "id": "JHZHZD"  
}
```

It is recommended that developers use the Order ID, as this is the key reference for the order.

Note that access to orders is restricted to the agency PCC that created the order and PCCs that have branch access to the creating PCC.

Canceling an Order Prior to Fulfillment

There are two methods to cancel an order prior to fulfillment, “orders/cancel” and “orders/change”. The “orders/cancel” method cancels all air content in the PNR (NDC and legacy air segments). The “orders/change” method allows for the airline’s NDC Order ID to be specified and cancels only the NDC segments in the PNR.

The “orders/cancel” endpoint

The “orders/cancel” API uses the “post” method, the current version of the “orders/cancel” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/orders/cancel>

Production environment:

<https://api.platform.sabre.com/v1/orders/cancel>

Using the “orders/cancel” API to Cancel All Air Segments

The “orders/cancel” API is used to cancel an order prior and post fulfillment (payment and ticketing), this example demonstrates how to use this request prior to fulfillment.

```
{
  "id": "1SXXXCX9EH7NS"
}
```

After the airline responds with confirmation of the cancellation, the order items are removed from the Sabre order and a display of the order is returned without any order items, the PNR is updated and all flight segments are removed.

```
{
  "order": {
    "id": "1SXXXCX9EH7NS",
    "type": "ORDER",
    "pnrLocator": "JHZHZD",
    "pnrCreateDate": "2023-10-10",
    "offerVendor": "AA",
    "orderOwner": "1S",
    "partition": "AA",
    "primeHost": "1S",
    "countryCode": "US",
    "contactInfos": [
      {
        "id": "C6A11D12-7562-40D6-BD0B-BAF22955B044",
        "phones": [
          {
            "id": "CF20AA2E-0C6E-49E9-95F7-568BEB98AB09",
            "number": "5551231231"
          }
        ],
        "emailAddresses": [
          {
            "id": "3F1DF19C-0FBB-4D43-A415-938B367B6C6B",
            "address": "DAVE@SABRE.COM"
          }
        ]
      }
    ],
    "passengers": [
      {
        "id": "Passenger1",
        "externalId": "T1",

```

```

    "typeCode": "ADT",
    "contactInfoRefIds": [
      "C6A11D12-7562-40D6-BD0B-BAF22955B044"
    ],
    "contactInfoRefId": "C6A11D12-7562-40D6-BD0B-BAF22955B044",
    "birthdate": "1975-01-01",
    "givenName": "DARRON",
    "surname": "DELTA",
    "genderCode": "M",
    "loyaltyProgramAccounts": [
      {
        "id": "D797E6C2-8AD9-45D5-BDF9-123E71BFBDD1",
        "accountNumber": "6H020M0-1.1",
        "carrier": {
          "carrierCode": "AA"
        }
      }
    ]
  },
  "customerNumber": {
    "number": "123456"
  }
}

```

Note: To cancel a fulfilled order a re-shop request is required to determine if the ticket(s) are in the void or refund period, the subsequent request to cancel the order requires an Offer ID from the re-shop response, therefore, if the request is sent per this example, it will receive an error. For more information on cancelling fulfilled orders refer to [“Cancelling a Fulfilled Order”](#).

The “orders/change” endpoint

The “orders/change” API uses the “post” method, the current version of the “orders/change” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/orders/change>

Production environment:

<https://api.platform.sabre.com/v1/orders/change>

Note that the “orders/change” API has many uses in addition to cancelling an NDC order, these are described later in this document.

Using the “orders/change” API to Cancel an NDC Order

Use the “orders/change” API to cancel the order in the airline’s system, the Sabre order and delete the NDC segments from the PNR:

```
{
  "id": "1SXXXABCD4321",
  "orderItemDeletes": [
    {
      "externalOrderId": "QF081T3R4E5W6"
    }
  ]
}
```

Updating Passenger Information in an Order

Passenger data may be updated using the “orders/change” API, the “orders/change” endpoint is described [here](#).

The following airlines do not support updating passenger data in an order, therefore, these types of updates must be made directly with the airline:

- Avianca/TACA (AV/TA)
- Finnair (AY)
- Qatar Airways (QR)
- Singapore Airlines (SQ)
- United Airlines (UA)
- Scandinavian Airlines (SK)
- LOT Polish (LO)
- Malaysia Airlines (MH)

Add Secure Flight Passenger Data (SFPD) to an Order

Use the “passengerUpdates” array to add Secure Flight Passenger Data (SFPD) to an order:

```
{
  "id": "1SXXXCL3PH50J",
  "passengerUpdates": [
    {
      "id": "Passenger1",
      "typeCode": "ADT",
      "givenName": "ALICE",
      "surname": "ALPHA",
      "birthdate": "1982-03-01",
      "genderCode": "F"
    }
  ]
}
```

Add a Frequent Flyer Number to an Order

Use the “loyaltyProgramAccountAdds” array to add a passenger’s frequent flyer number to an order:

```
{
  "id": "1SXXXCEKEHT9V",
  "loyaltyProgramAccountAdds": [
    {
      "accountNumber": "1295036",
      "carrier": {
        "carrierCode": "QF"
      },
      "nameReference": "Passenger1"
    }
  ]
}
```

Note that some airlines verify the frequent flyer number against the account details in their loyalty system and may reject this update if the frequent flyer number is invalid.

Add Passport Information to an Order

Use the “identityDocumentAdds” array to add a passenger passport information to an order:

```
{
  "id": "1SXXXCEK5HT9V",
  "identityDocumentAdds": [
    {
      "id": "P1",
      "nameReference": "Passenger1",
      "documentNumber": "0123456789",
      "documentTypeCode": "PT",
      "issuingCountryCode": "US",
      "placeOfIssue": "GB",
      "citizenshipCountryCode": "US",
      "residenceCountryCode": "US",
      "birthdate": "1976-04-01",
      "givenName": "FREDDIE",
      "surname": "FOXTROT",
      "genderCode": "M",
      "issueDate": "2011-12-20",
      "expiryDate": "2025-12-30",
      "hostCountryCode": "US"
    }
  ]
}
```

Add a Known Traveler Number (KTN) to an Order

Use the “identityDocumentAdds” array to add a Known Traveler Number (KTN) to an order:

```
{
  "id": "1SXXXCG6Q2JGY",
  "identityDocumentAdds": [
    {
      "documentNumber": "157999443",
      "documentTypeCode": "K",
      "issuingCountryCode": "US",
      "nameReference": "Passenger1"
    }
  ]
}
```

Once an order has been updated with a KTN it is assigned an ID in the “identityDocuments” array.

```
"identityDocuments": [
  {
    "id": "0B10281C-E2F0-4CCF-8497-1BDD255655B5",
    "documentNumber": "157999443",
    "documentTypeCode": "K",
    "issuingCountryCode": "US"
    "nameReference": "Passenger1"
  }
],
```

Delete a Known Traveler Number (KTN) from an Order

To delete a KTN from an order, populate the “identityDocumentDeletes” array with the details of the KTN and the reference ID:

```
{
  "id": "1SXXXCG6Q2JGY",
  "identityDocumentDeletes": [
    {
      "id": "0B10281C-E2F0-4CCF-8497-1BDD255655B5",
      "documentNumber": "157999443",
      "documentTypeCode": "K",
      "issuingCountryCode": "US",
      "nameReference": "Passenger1"
    }
  ]
}
```

The same process is used to add and delete Redress numbers but with the “documentTypeCode” value of “R”.

Correcting a Passenger's Name

If a passenger's name has been entered incorrectly, it can be corrected using the "orders/change" API, at this time this modification is only possible prior to payment and ticketing. To correct a passenger's name, specify the following in the request:

- The Sabre Order ID
- Under the "passengerUpdates" node:
 - The passenger ID
 - The passenger type code
 - The passenger's given name
 - The passenger's surname
 - The passenger's date of birth (from the original order)
 - The passenger's gender code (from the original order)

```
{
  "id": "1SXXXC6J0IQV5",
  "passengerUpdates": [
    {
      "id": "Passenger1",
      "typeCode": "ADT",
      "givenName": "JENNIE",
      "surname": "JULIETTE",
      "birthdate": "1987-09-12",
      "genderCode": "F"
    }
  ]
}
```

The response to this request is a view of the order with the modified passenger data, if the change is not permitted the response from the airline is returned. The passenger's name in the PNR is updated to reflect the modification.

If the airline does not allow name corrections the following error is returned:

```
{
  "errors": [
    {
      "code": "137",
      "message": "Name change not allowed, please contact airline directly: (Name change is not supported for this airline)"
    }
  ]
}
```

If the order has been fulfilled, the following error is returned:

```
{
  "errors": [
    {
      "code": "137",
      "message": "Name change not allowed, please contact airline directly: (Order contains
ticket with open status. Name change is not supported for this airline at this point.)"
    }
  ]
}
```

Notes

An airline's policy may restrict name changes or only allow a certain number of characters in a name to be changed.

The "givenName" and "surname" may be modified in a single request, if you need to correct either the gender or date of birth a separate request should be made.

If an agent uses this procedure to completely change a passenger, the airline may issue an Agency Debit Memo (ADM).

Changes to a passenger name may cause the stored frequent flyer number to be removed or cancellation of ancillary services/pre-reserved seats as these items are specific to a passenger.

Repricing an Order

To update the price of an order the “offers/repriceOrder” API is used, this can be useful for airlines that have a “price guarantee time limit” as it allows the agent to reprice the order and obtain a new price guarantee if the original one has expired. This can also be used to verify if any price changes have occurred prior to fulfilment.

Note that if the price guarantee time limit has expired, fulfilment of the order is restricted.

The “offers/repriceOrder” endpoint

The “offers/repriceOrder” API uses the “post” method, the current version of the “offers/repriceOrder” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/offers/repriceOrder>

Production environment:

<https://api.platform.sabre.com/v1/offers/repriceOrder>

Using the “offers/repriceOrder” API to Reprice an Unfulfilled Order

The “offers/repriceOrder” API is used to request repricing of an unfulfilled order, the order may be within the price guarantee time limit or outside, as long as the payment time limit has not expired.

The only parameter required in the request is the Sabre Order ID.

```
{
  "request": {
    "orderId": "1SXXXCRDOERU8"
  }
}
```

If the airline provides sufficient information in the response, then the system will perform a price comparison and return the results. If the prices are the same and the order is still within the price guarantee time limit no further action is required:

```
{
  "payloadAttributes": {
    "timestamp": "2024-02-02T20:29:36.431222616Z"
  },
  "warnings": [
    {
      "category": "WARNING",
      "type": "PROCESSING_WARNING",
      "description": "QF: Card Payment fees ranging between AUD 2.00 and 6.50 may apply depending on card type. An IIN/BIN can be sent in the request to get a specific fee or a full list of fees can be displayed via the Service List.",
      "service": "NDCC_ORDER_REPRICE",
      "code": 33
    }
  ],
  "response": {
    "hasSamePrice": true
  }
}
```

If the airline does not return full pricing data in the response, a comparison cannot be completed and a warning is returned. In this instance, a new offer is returned which may be accepted, once this step is completed the order will have a new price guarantee time limit and payment time limit:

```
"warnings": [
  {
    "category": "WARNING",
    "type": "PROCESSING_WARNING",
    "description": "Comparison was not possible due to insufficient information from
supplier.",
    "service": "REPRICE_NDC_ORDER_TN"
  }
],
```

If the payment guarantee has expired a new offer is returned with the “hasSamePrice” indicator (which may be true or false), if the agent accepts the offer a new price guarantee time limit is applied and the order may be fulfilled:

```
{
  "payloadAttributes": {
    "timestamp": "2024-02-07T20:26:13.399996852Z"
  },
  "warnings": [
    {
      "category": "WARNING",
      "type": "PROCESSING_WARNING",
      "description": "QF: Card Payment fees ranging between AUD 3.70 and 12.20 may apply
depending on card type. An IIN/BIN can be sent in the request to get a specific fee or a full
list of fees can be displayed via the Service List.",
      "service": "NDCC_ORDER_REPRICE",
      "code": 33
    },
    {
      "category": "WARNING",
      "type": "PROCESSING_WARNING",
      "description": "QF: Qantas Frequent Flyers could earn Qantas Points and Status Credits
for this booking. <a href=\"https://www.qantas.com/fflyer/dyn/program/terms\"
target=\"_blank\">Terms and conditions apply.</a>",
      "service": "NDCC_ORDER_REPRICE"
    }
  ],
  "response": {
    "datalists": {
      "segments": [
        {
          "segmentId": "S2",
```



```

    "departure": {
      "airportCode": "SYD",
      "departureDate": "2024-02-16",
      "departureTime": "17:35"
    },
    "arrival": {
      "airportCode": "BNE",
      "arrivalDate": "2024-02-16",
      "arrivalTime": "18:05"
    },
    "marketingAirline": {
      "code": "QF",
      "flightNumber": "542"
    },
    "datedOperatingLegs": [
      {
        "departure": {
          "airportCode": "SYD",
          "departureDate": "2024-02-16",
          "departureTime": "17:35"
        },
        "arrival": {
          "airportCode": "BNE",
          "arrivalDate": "2024-02-16",
          "arrivalTime": "18:05"
        }
      }
    ]
  },
  {
    "segmentId": "S1",
    "departure": {
      "airportCode": "BNE",
      "departureDate": "2024-02-10",
      "departureTime": "15:10"
    },
    "arrival": {
      "airportCode": "SYD",
      "arrivalDate": "2024-02-10",
      "arrivalTime": "17:45"
    },
    "marketingAirline": {
      "code": "QF",
      "flightNumber": "541"
    },
    "datedOperatingLegs": [
      {
        "departure": {

```

```

        "airportCode": "BNE",
        "departureDate": "2024-02-10",
        "departureTime": "15:10"
    },
    "arrival": {
        "airportCode": "SYD",
        "arrivalDate": "2024-02-10",
        "arrivalTime": "17:45"
    }
}
],
"passengers": [
    {
        "passengerId": "Passenger1",
        "typeCode": "ADT"
    }
],
"priceClasses": [
    {
        "priceClassId": "QFEconomyFlex-1",
        "code": "FLEX",
        "name": "Economy Flex",
        "descriptions": [
            {
                "description": "1. Changes:"
            },
            {
                "description": "- Permitted with a change fee of AUD99.00, inclusive of GST"
            },
            {
                "description": "2. Cancellations:"
            },
            {
                "description": "- Permitted with a cancellation fee of AUD99.00, inclusive
of GST"
            },
            {
                "description": "- No show results in loss of fare"
            }
        ]
    }
],
"voluntaryChangeInformation": [
    {
        "changeId": "66d9babe-6758-42ac-9093-91456d447dd6",
        "type": "Cancel",

```

```

        "isAllowed": true,
        "hasFee": true,
        "feeAmount": "99.00",
        "feeCurrencyCode": "AUD",
        "applicabilityList": [
            "Before Departure"
        ]
    },
    {
        "changeId": "b4c44584-7a29-4b2f-b26e-6295965ed654",
        "type": "Change",
        "isAllowed": true,
        "hasFee": false,
        "feeAmount": "0.00",
        "feeCurrencyCode": "AUD",
        "applicabilityList": [
            "Before Departure"
        ]
    },
    {
        "changeId": "02096159-45d9-48bd-abe3-6c3ad3dfc077",
        "type": "Change",
        "isAllowed": true,
        "hasFee": false,
        "feeAmount": "0.00",
        "feeCurrencyCode": "AUD",
        "applicabilityList": [
            "After Departure"
        ]
    },
    {
        "changeId": "61ff2cfc-b93f-4603-8397-5d350b1fa976",
        "type": "Refund",
        "isAllowed": true
    }
]
},
"offers": [
    {
        "offerId": "ce5b4fd69b46x4qh9lsc117kw1-1",
        "offerExpirationDateTime": "2024-02-07T20:46:13Z",
        "ownerCode": "QF",
        "offerItems": [
            {
                "offerItemId": "ce5b4fd69b46x4qh9lsc117kw1-1-1",
                "isMandatory": true,
                "passengers": [
                    {

```

```

"passengerRef": "Passenger1",
"pricedTypeCode": "ADT",
"price": {
  "totalAmount": {
    "amount": "1207.24",
    "currencyCode": "AUD"
  },
  "baseAmount": {
    "amount": "1032.74",
    "currencyCode": "AUD"
  },
  "taxSummary": {
    "totalTaxAmount": {
      "amount": "174.50",
      "currencyCode": "AUD"
    },
    "breakdown": [
      {
        "taxAmount": {
          "amount": "22.15",
          "currencyCode": "AUD"
        },
        "taxCode": "QR",
        "nationCode": "AU",
        "description": "PASSENGER SERVICE CHARGE DOMESTIC DEPARTURE"
      },
      {
        "taxAmount": {
          "amount": "22.15",
          "currencyCode": "AUD"
        },
        "taxCode": "QR",
        "nationCode": "AU",
        "description": "PASSENGER SERVICE CHARGE DOMESTIC ARRIVAL"
      },
      {
        "taxAmount": {
          "amount": "103.28",
          "currencyCode": "AUD"
        },
        "taxCode": "UO",
        "nationCode": "AU",
        "description": "GOODS AND SERVICES TAX GST"
      },
      {
        "taxAmount": {
          "amount": "6.48",
          "currencyCode": "AUD"
        }
      }
    ]
  }
}

```

```

    },
    "taxCode": "UO",
    "nationCode": "AU",
    "description": "GOODS AND SERVICES TAX GST"
  },
  {
    "taxAmount": {
      "amount": "20.44",
      "currencyCode": "AUD"
    },
    "taxCode": "WG",
    "nationCode": "AU",
    "description": "SAFETY AND SECURITY CHARGE DEPARTURE"
  }
]
}
},
"fareComponents": [
  {
    "fareBasis": {
      "code": "KFQA",
      "fareDescription": "Published fare",
      "fareCode": "70J",
      "bookingClassCodes": [
        "K"
      ],
      "cabins": [
        {
          "code": "Y",
          "name": "Economy",
          "carrierCabinCode": "Y",
          "carrierCabinName": "Economy"
        }
      ]
    },
    "segmentRefs": [
      "S1"
    ],
    "priceClassRef": "QFEconomyFlex-1",
    "voluntaryChangeInformationRefs": [
      "66d9babe-6758-42ac-9093-91456d447dd6",
      "b4c44584-7a29-4b2f-b26e-6295965ed654",
      "02096159-45d9-48bd-abe3-6c3ad3dfc077",
      "61ff2cfc-b93f-4603-8397-5d350b1fa976"
    ]
  },
  {
    "fareBasis": {

```

```

        "code": "KFQA",
        "fareDescription": "Published fare",
        "fareCode": "70J",
        "bookingClassCodes": [
            "K"
        ],
        "cabins": [
            {
                "code": "Y",
                "name": "Economy",
                "carrierCabinCode": "Y",
                "carrierCabinName": "Economy"
            }
        ]
    },
    "segmentRefs": [
        "S2"
    ],
    "priceClassRef": "QFEconomyFlex-1",
    "voluntaryChangeInformationRefs": [
        "66d9babe-6758-42ac-9093-91456d447dd6",
        "b4c44584-7a29-4b2f-b26e-6295965ed654",
        "02096159-45d9-48bd-abe3-6c3ad3dfc077",
        "61ff2cfc-b93f-4603-8397-5d350b1fa976"
    ]
}
]
}
],
"price": {
    "totalAmount": {
        "amount": "1207.24",
        "currencyCode": "AUD"
    },
    "baseAmount": {
        "amount": "1032.74",
        "currencyCode": "AUD"
    },
    "taxSummary": {
        "totalTaxAmount": {
            "amount": "174.50",
            "currencyCode": "AUD"
        }
    }
},
"originalOrderItem": {
    "orderItemId": "77249B85-438A-4265-9413-1AF6C7E3F805",
    "price": {

```

```

        "totalAmount": {
            "amount": "1207.04",
            "currencyCode": "AUD"
        },
        "baseAmount": {
            "amount": "1032.74",
            "currencyCode": "AUD"
        },
        "totalTaxAmount": {
            "amount": "174.30",
            "currencyCode": "AUD"
        }
    },
    "hasSamePrice": false
}
]
"hasSamePrice": false
}
}

```

If an order has been fulfilled the system will respond with the following error:

```

{
  "payloadAttributes": {
    "timestamp": "2024-01-18T14:40:28.061969206Z"
  },
  "errors": [
    {
      "category": "INVALID_DATA",
      "type": "INVALID_ORDER",
      "description": "Invalid order - Order has already been fulfilled, reprice is not permitted.",
      "service": "REPRICE_NDC_ORDER_TN"
    }
  ]
}

```

Accepting a Repriced Offer and Updating the Order

To accept a repriced offer the “orders/change” API is used. The input parameters are:

- The Sabre Order ID.
- The Offer ID from the reshop response.
- The Offer Item ID from the reshop response.
- The Passenger Reference(s).

```
{
  "id": "1SXXXC43RUPWF",
  "orderItemUpdates": [
    {
      "acceptOffers": [
        {
          "offerId": "cd0757cdbf75sxmns1bqi8k343-1",
          "selectedOfferItems": [
            {
              "id": "cd0757cdbf75sxmns1bqi8k343-1-1",
              "passengerRefIds": [
                "Passenger1"
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

The response is a full display of the order, the pricing information, payment time limit and, if applicable, the price guarantee time limit are updated.

Note that not all airlines utilize the price guarantee time limit.

Fulfilling an Order

Order fulfillment is the process of authorizing credit cards, dispensing document (ticket and EMD) numbers writing the records to the airlines' databases. Reporting to settlement systems (ARC and IATA BSP), enabling payments to the airlines, debiting of the agency accounts and the payment of up-front commissions.

To request fulfilment of an order the "orders/change" API is used, the "orders/change" endpoint is defined [here](#).

Airline and Neutral Stock

Airlines may choose which type of stock (document ranges) they use, the two options are:

- Airline stock – Numbers allocated to the airline for their own use – optional to report to settlement systems.
- Neutral stock – Numbers allocated to Ticket System Providers (TSPs), typically GDS's - mandatory to report to settlement systems.

Forms of Payment

Supported forms of payment:

- Cash – the agency collects payment and is billed a net amount by the airline.
- Credit Card – the airline collects payment and credits the agency for any commission.

Some countries do not have settlement systems, as a result airlines must use their own airline stock and typically only support credit card as a form of payment.

Note that at this time, only a single form of payment is supported per fulfillment request, although order items may be fulfilled separately with different forms of payment.

Three Domain Security v2 (3DSv2)

Some airlines have implemented 3DSv2, a secure payment protocol that enables a two-factor authentication process with the cardholder. When 3DSv2 is employed, the agent's point of sale application is responsible for initiating a query to the card issuer, which determines if a card is enrolled in 3DSv2. If it is, the point-of-sale application makes a secondary call to initiate authentication of the transaction by the cardholder, once this is obtained, authentication is returned to the point-of-sale application which is then passed to the airline. This process reduces fraudulent card transactions as well as reducing false positives. It also moves liability to charge backs from the agency to the airline. Credit card transactions may be reported as Mail Order/Telephone Order (MOTO) or eCommerce (EC), offline agents typically use the MOTO values and do not process 3DSv2, online agents are more likely to use the 3DSv2 process and send the EC values. The Sabre system does not store the 3DSv2 values, only the Payment Transaction Channel Code (MO, TO or EC), is saved in the order.

Document Types and Codes

Tickets and EMDs are generally termed as “documents”, and within the document section of an NDC order the document type is defined based on the PADIS code set directory composite 1001, the most common values are:

- 700 – All other document types (may be used for MCO’s in the U.S. point of sale)
- 702 – Ticket (electronic)
- J – EMD-A
- Y – EMD-S
- T – Ticket
- INF – Infant Ticket

Coupon Status

All documents have coupons, this is a legacy concept that was applicable to paper tickets as many airline operational and accounting functions (check-in, boarding and revenue accounting) rely on coupon status, this concept remains in place for electronic documents. Coupon status codes are defined in the PADIS code set directory under composite 4405, the most common coupon status codes are:

- I – Open for use, first issuance
- V – Voided (final state)
- RF – Refunded (final state)
- OPE – Open for use
- E – Exchanged (final state)
- AL – Airport Control
- B – Flown\Used (final state)

When a coupon is in a final state it cannot be transacted (Refunded, Voided, Exchanged and Flown\Used).

Examples of the “orders/change” Request to Fulfill an Order

Cash form of payment is available to agents that are accredited by IATA or ARC, when cash is used the agent is billed by the settlement system, this can be the full amount or a net amount minus any commission.

Data elements required in the fulfillment request:

- Sabre Order ID
- Total Amount
- Currency Code
- Payment Method
- Order Item Reference IDs

Fulfilling an order with cash as the form of payment:

```
{
  "id": "1SXXXCELULU9V",
  "actions": [
    {
      "fulfillOrder": {
        "paymentInfo": {
          "amount": {
            "amount": "213.67",
            "code": "AUD"
          },
          "paymentMethod": {
            "paymentCash": {}
          },
          "orderItemRefIds": [
            "743D38EA-304A-4FBE-8A5F-B842B00AC449"
          ]
        }
      }
    }
  ]
}
```

Fulfilling an order with credit card as the form of payment using the “MO” payment transaction code:

When fulfilling an order with a credit card form of payment it is required that the secured payment transaction channel code is specified, for point-of-sale applications that do not process credit card transactions with 3DSv2 validations the “MO” (Mail Order) or “TO” (Telephone Order) value is used.

```
{
  "id": "1SXXXCEKEHT9V",
  "actions": [
    {
      "fulfillOrder": {
        "paymentInfo": {
          "amount": {
            "amount": "213.67",
            "code": "AUD"
          },
          "paymentMethod": {
            "paymentCard": {
              "cardNumber": "3456XXXXXXXXXX",
              "contactInfoRefId": "CI-1",
              "cvv": "4111",
              "expirationDate": "0139",
              "vendorCode": "AX",
              "securePaymentVersion2": {
                "paymentTrxChannelCode": "MO"
              }
            }
          },
          "orderItemIds": [
            "743D38EA-304A-4FBE-8A5F-B842B00AC449"
          ]
        }
      }
    }
  ]
}
```

Notes:

1. The card number is masked in the example
2. The CVV is optional

Fulfilling an order with credit card as the form of payment using the "EC" payment transaction code:

If the point-of-sale application does process credit card transactions using the 3DSv2 protocol, then the fulfillment request is populated with the additional values and the payment transaction channel code is set to "EC" (eCommerce).

Example 1 - All of the supported values for 3DS2 eCommerce transactions

```
{
  "id": "1SXXXADRA5CAL",
  "actions": [
    {
      "fulfillOrder": {
        "paymentInfo": {
          "amount": {
            "amount": "213.67",
            "code": "AUD"
          },
        },
        "paymentMethod": {
          "paymentCard": {
            "cardNumber": "4929XXXXXXXXXXXX",
            "contactInfoRefId": "CI-1",
            "cvv": "1234",
            "expirationDate": "0129",
            "vendorCode": "VI",
            "securePaymentVersion2": {
              "authenticationMerchantName": "Merchant ABC",
              "authenticationValueText": "eJy1mdmzokizwN/5Kzr6e3TOsAtM",
              "authenticationTokenValue": "WTJGeVpHbHVZV3hqYjIxdaTCVYPkchVyAgIAo%3D",
              "cardNumberCollectionCode": "K",
              "directoryServerTrxID": "a28b-4473-a130-3fa5e1c2eef0",
              "electronicCommerceIndText": "vbv",
              "payerAuthenticationExemptionCode": "LV",
              "payerAuthenticationFailureToCompleteCode": "NC",
              "paymentTrxChannelCode": "EC",
              "programProtocolText": "2.1.0",
              "trxStatusReasonText": "17",
              "trxStatusText": "Y"
            }
          }
        },
      },
      "orderItemIds": [
        "743D38EA-304A-4FBE-8A5F-B842B00AC449"
      ]
    }
  ]
}
```

```
}  
]  
}
```

Example 2 - The mandatory subset of values for the “securePaymentVersion2” node.

```
"securePaymentVersion2": {  
  "authenticationTokenValue": "WTJGeVpHbHVZV3hqYjIxdFpYSmpaV0YxZEdnPQ==<",  
  "directoryServerTrxID": "M2Q1MGY2ZTUtZGY0My00M2M2LWF1YWYtZDY5OTE4NTg0ZjRl",  
  "authenticationValueText": " eJylmdmzokizwN/5Kzr6e3TOsAtM"  
  "electronicCommerceInd": "05",  
  "paymentTrxChannelCode": "EC",  
  "programProtocolText": "2.1.0",  
  "trxStatusText": "Y"  
}
```

Fulfilling an order with two order items (airfare and paid seat):

To fulfill more than one order item at the same time, populate the “orderItemRefIds” array with multiple Order Item IDs, a common use case is when an order contains an airfare and a paid seat.

```
{
  "id": "1SXXXSHAR1GY",
  "actions": [
    {
      "fulfillOrder": {
        "paymentInfo": {
          "amount": {
            "amount": "400.15",
            "code": "AUD"
          },
          "paymentMethod": {
            "paymentCash": {}
          },
          "orderItemRefIds": [
            "5FC41258-1317-4D0F-8A46-6E18DA64587A",
            "5201D54E-10AA-4756-B969-E57B435BD902"
          ]
        }
      }
    }
  ]
}
```


Fulfilling an order with a referenced form of payment:

A referenced form of payment may be used to fulfill an order, this can occur if the PNR contains more than one form of payment or if the credit card is masked, the system will detokenize the credit card before sending it to the airline.

```
*FOP
FORM OF PAYMENT
  1. PAYMENT CARD TYPE
    *VI4XXXXXXXXXXXX3453/0229
  2. PAYMENT CARD TYPE
    *AX3XXXXXXXXXXXX0007/0139
```

If the referenced form of payment is a card, it is also necessary to specify the payment transaction channel code to satisfy the secured payment (3DSv2) requirement.

```
{
  "id": "1SXXCG6QABBY",
  "actions": [
    {
      "fulfillOrder": {
        "paymentInfo": {
          "amount": {
            "amount": "410.19",
            "code": "AUD"
          },
          },
        "paymentMethod": {
          "paymentReferenceFormOfPayment": {
            "referenceId": "2",
            "securePaymentVersion2": {
              "paymentTrxChannelCode": "MO"
            }
          }
        },
        "orderItemIds": [
          "A6BE12C0-8777-4097-86BF-A5ACAAE87F9B"
        ]
      }
    }
  ]
}
```

Ticket information in an order

After an order has been fulfilled, the order will be updated with the “ticketDocumentInfo” section with individual document data elements. The following description describes some of those elements:

- Document Number – 13 numeric, formed of: airline accounting code (3), form code (4), serial number (6).
- Document Type – values are defined in the [fulfillment introduction](#).
- Issue Date and Time – the date and time the document was issued in Zulu time.
- Number of Booklets – the number of documents issued in conjunction (maximum 4).
- Coupons – Contains values at the coupon level.
 - Coupon Number – 1-4, some coupons may be omitted for tickets when there is a surface sector.
 - Coupon Status - values are defined in the [fulfillment introduction](#).
 - Service ID – a reference to the service this coupon was issued for.
 - Order Item Reference ID – The Order Item this coupon was issued for.
 - Segment Information – Segment Information this coupon was issued for.
- Pricing Information – base amount, total taxes and total.
- Passenger Reference – a reference to the passenger this document was issued for.

```
... "ticketingDocumentInfo": [  
  {  
    "document": {  
      "number": "0015750646616",  
      "type": "702",  
      "issueDateTime": "2023-10-12T11:58:00Z",  
      "ticketingLocation": "DFW",  
      "numberOfBooklets": 1,  
      "reportingType": "ARC",  
      "coupons": [  
        {  
          "number": 1,  
          "status": "I",  
          "serviceRefId": "9E56E41E-3282-42B5-95BE-A103797FC036",  
          "orderItemRefId": "5E39589C-45B4-4757-BBAA-52DEF37275EC",  
          "segmentInfo": {  
            "departureAirport": "DFW",  
            "arrivalAirport": "TPA",  
            "marketingAirlineCode": "AA",  
            "departureTime": "2023-10-31T13:13:00",  
            "arrivalTime": "2023-10-31T16:37:00",  
            "flightNumber": 2504,  
            "classOfService": "B"  
          }  
        }  
      ]  
    },  
    "price": {
```

```
    "baseAmount": {
      "amount": "68.84",
      "code": "USD"
    },
    "totalAmount": {
      "amount": "88.90",
      "code": "USD"
    },
    "totalTaxAmount": {
      "amount": "20.06",
      "code": "USD"
    }
  },
  "paxRefId": "Passenger1"
},
```

Updates to the PNR

After an Order Item has been fulfilled, the PNR is updated with the following information:

- Ticket number with the “TO” indicator (Ticket number for an Order item)

```
*T
TKT/TIME LIMIT
1.T-21OCT-27M7*ABC
2.TO 1342111057710-XX OSCAR/O 27M7*ABC 0835/21OCT*
```

- Accounting line for invoicing

```
*PAC
ACCOUNTING DATA
1. AV¥2111057710/ 0.00/AUD 490.00/ 274.40/ONE/CA 1
.1OSCAR OLLIE/1/F/E
```

Canceling a Fulfilled Order

After an order has been fulfilled, the order may be cancelled, however, accountable documents (tickets and EMDs) have to be taken into consideration. Depending on the airline's implementation and the settlement system there is a period of time after documents are issued when the sale can be cancelled, this is known as the "void period". If a document is in the void period and the agent cancels the order then the sale is not reported, however, if the void period has passed, then the document is outside of the void period and in the "refund period" and the refund process applies. Refund processing takes into account any penalty fees which are deducted from the amount due back to the passenger. Refunds may also take place after travel has commenced, in this instance the fare rules will determine how much the passenger may be due and the calculation may also deduct taxes that were applicable to the flown portion of the ticket. It should also be noted that credit card surcharges (OB fees) are not refundable even if the ticket is fully refundable. There is also an option available for some airlines to cancel the order and retain the ticket as a credit towards future travel.

The first step in the process to cancel a fulfilled order is to determine if the order is in the void or refund period, to make this determination use the "offers/reshop/cancelOrder" API.

The "offers/reshop/cancelOrder" endpoint

The "offers/reshop/cancelOrder" API uses the "post" method, the current version of the "offers/reshop/cancelOrder" API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/offers/reshop/cancelOrder>

Production environment:

<https://api.platform.sabre.com/v1/offers/reshop/cancelOrder>

Checking if Documents are in the Void or Refund Period

Void periods vary by settlement system and airline implementation, for airlines reporting to ARC and using neutral stock, the voiding period is midnight of the day after the ticket was issued, for IATA BSP it is midnight of the day the ticket was issued. Airlines that use their own stock determine their own voiding period.

Use the “offers/reshop/cancelOrder” API to determine if the documents are in the Void or Refund window.

The following elements are required in the request:

- The Sabre Order ID.
- The Order Item IDs that you want to check.

```
{
  "request": {
    "orderId": "1SXXXR0CKYC4T",
    "orderItemIds": [
      "03A12937-D2E8-45E4-B0A9-FDF87C2C08E2"
    ]
  }
}
```

Note that this request does not affect any change to the order, it is used solely to determine the options in case the passenger wishes to cancel the order.

An Offer for Cancelling an Order with a Ticket in the Void Period

The response to the “offers/reshop/cancelOrder” contains an Offer Item ID that you will need to use in the “orders/cancel” request if you want to proceed with cancelling the order and voiding the ticket. The “offerType” value states “VOID” and there is no penalty fee information.

```
{
  "response": {
    "warnings": [
      {
        "message": "Ancillary Service value e.g. Seats, may not be included in the total amount.",
        "service": "SPR"
      }
    ],
    "reshopOffers": [
      {
        "offerId": "ce0d47db58878w5xrlnn101mq4",
        "ownerCode": "AA",

```

```
"offerExpirationDateTime": "2023-10-12T23:05:52.41795858Z",
"offerType": "VOID",
"offerItems": [
  {
    "offerItemId": "ce0d47db58878w5xrlnnl01mq4-1",
    "mandatoryInd": true,
    "originalOrderItemDifferential": {
      "amount": {
        "amount": "124.65",
        "curCode": "USD"
      },
      "taxSummary": {
        "totalTaxAmount": {
          "amount": "24.25",
          "curCode": "USD"
        }
      }
    },
    "differentialAmountDue": {
      "amount": {
        "amount": "0.00",
        "curCode": "USD"
      }
    }
  }
]
```

An Offer for Cancelling an Order with a Ticket in the Refund Period

The response to the “offers/reshop/cancelOrder” contains an Offer Item ID that you will need to use in the “orders/cancel” request if you want to proceed with cancelling the order and refunding the ticket. The “offerType” value states “REFUND” and the “penaltyDifferential” is used to advise the passenger how much is owed to the airline.

```
{
  "response": {
    "warnings": [
      {
        "message": "Ancillary Service value e.g. Seats, may not be refundable.",
        "service": "SPR"
      }
    ],
    "reshopOffers": [
      {
        "offerId": "ce646d964597vlxh8lnop539i9",
        "ownerCode": "AA",
        "offerExpirationDateTime": "2023-10-13T15:29:16.262325137Z",
        "offerType": "REFUND",
        "offerItems": [
          {
            "offerItemId": "ce646d964597vlxh8lnop539i9-1",
            "mandatoryInd": true,
            "originalOrderItemDifferential": {
              "amount": {
                "amount": "133.95",
                "curCode": "USD"
              },
              "taxSummary": {
                "totalTaxAmount": {
                  "amount": "24.95",
                  "curCode": "USD"
                }
              }
            }
          }
        ],
        "differentialAmountDue": {
          "amount": {
            "amount": "0.00",
            "curCode": "USD"
          },
          "taxSummary": {
            "totalTaxAmount": {
              "amount": "0.00",
              "curCode": "USD"
            }
          }
        }
      }
    ]
  }
}
```



```
    }
  }
},
  "penaltyDifferential": {
    "amount": {
      "amount": "133.95",
      "curCode": "USD"
    }
  }
}
]
```

Cancellation of a Fulfilled Order

Cancelling an Order and Voiding or Refunding the Ticket

Having determined whether a fulfilled order is in the void or refund period, the order is cancelled by calling the “orders/cancel” API, include the Sabre Order ID and the Offer Item ID from the “offers/reshop/cancelOrder” response.

```
{
  "id" : "1SXXXR0CKYC4T",
  "reshopOfferItemId" : " ce0d47db58878w5xrlnn101mq4-1"
}
```

The response is a view of the order, all order items are deleted and the coupon status in the “ticketingDocumentInfo” is set to either void (V) or refund (RF).

```
... "ticketingDocumentInfo": [
  {
    "document": {
      "number": "0015750650186",
      "type": "702",
      "issueDateTime": "2023-10-13T09:24:00Z",
      "ticketingLocation": "DFW",
      "numberOfBooklets": 1,
      "reportingType": "ARC",
      "coupons": [
        {
          "number": 1,
          "status": "V",
          "segmentInfo": {
            "departureAirport": "DFW",
            "arrivalAirport": "ORD",
            "marketingAirlineCode": "AA",
            "departureTime": "2023-10-20T05:00:00",
            "arrivalTime": "2023-10-20T07:23:00",
            "flightNumber": 423,
            "classOfService": "B"
          }
        }
      ]
    }
  }
]
```

Note that a ticket can only be voided if all coupons are in an open status and no part of the ticket is flown.

Updates to the PNR

After a fulfilled Order has been cancelled, the PNR is updated with the following information:

- Ticket number with the “OV” or “OR” indicator (Order Item Voided or Order Item Refunded)

```
*T
TKT/TIME LIMIT
1.T-21OCT-27M7*ABC
2.TO 1342111057710-XX OSCAR/O 27M7*ABC 0835/21OCT*
3.OR 1342111057710-XX OSCAR/O 27M7*ABC 0713/24OCT
```

Canceling an Order and Retaining the Ticket for Future use

If the ticket has no refund value or a high penalty fee, it may be beneficial to cancel the order and retain the ticket for future use. This procedure does not require a “reshop” to determine if the ticket is in the void or refund period, although it may be considered best practice as it is preferable to void the ticket rather than have the funds held as a credit.

To cancel an order and retain a ticket for future use, the “orders/change” API is used:

```
{
  "id": "1SXXXCAAQHSD8",
  "cancelWithRetain": "true"
}
```

The response is a view of the order, all order items are deleted and the coupon status in the “ticketingDocumentInfo” remains in an open status (I).

Notes:

1. A future development will provide an API to allow agents to obtain a list of retained tickets.
2. Not all airlines support the “cancel and retain” process, the following airlines are enabled:
 - American Airlines (AA)
 - Qantas Airways (QF)

Pre-reserved Seat Assignments

The process for booking and paying for a pre-reserved seat assignment consists of three steps, display the seat map, book a seat and pay for a seat.

Displaying a Seat Map

To display a seat map the “offers/getseats” API is used, booking and fulfilling paid seats is enabled through the “orders/change” API.

The “offers/getseats” endpoint

The “offers/getseats” API uses the “post” method, the current version of the “offers/getseats” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/offers/getseats>

Production environment:

<https://api.platform.sabre.com/v1/offers/getseats>

Note: The “offers/getseats” API requires an application ID to obtain bookable offers, if this is not included in the header then the offers returned will be informational only. Refer to the [Getting Started](#) section to find out more about Application IDs.

Request a Seat Map with an Offer ID

There are two methods to display a seat map, you can use an Offer ID from a pricing response, or an Order ID from a confirmed Order. If you use an Offer ID, the seat map is informational only and you will not be able to reserve a seat with offers from the response, the Offer IDs in the response will be “displayOnlyMap-n”.

In the “offers/getseats” API request specify the “requestType” as “offerId” and insert the Offer ID from the pricing (offers/price) response.

```
{
  "requestType": "offerId",
  "request": {
    "offer": {
      "offerId": "ce646d964597v1xh81noyfb6v0-1-1"
    }
  },
  "pointOfSale": {
    "agentDutyCode": "*",
    "location": {
      "countryCode": "US",
      "cityCode": "SFO"
    }
  }
}
```

Note that seat pricing is based on the fare brand from the priced offer.

Request a Seat Map with an Order ID

To request a seat map with bookable seats, use the option with the Sabre Order ID and include a valid application ID in the header.

In the “offers/getseats” API request specify the “requestType” as “orderId” and send the Sabre Order ID from the confirmed order.

```
{
  "requestType": "orderId",
  "request": {
    "order": {
      "orderId": "1SXXXCQU1NH3D"
    }
  },
  "pointOfSale": {
    "agentDutyCode": "*",
    "location": {
      "countryCode": "US",
      "cityCode": "SFO"
    }
  }
}
```

Note that seat pricing is based on the fare brand and the passenger’s frequent flyer tier level, if there is a frequent flyer number in the order.

If an application ID is not included in the header, you will receive the following error:

```
{
  "errors": [
    {
      "code": "060090",
      "message": "Application ID was not provided in session or request header: [Not provided application id]"
    }
  ]
}
```

Reading the Seat Map Response

The first section details the offers, there may be several offers and some may be zero value (free seats). Offers have an Offer Item ID with pricing information, the service definition, passenger reference and segment reference.

```
{
  "response": {
    "aLaCarteOffer": {
      "offerId": "cd646d964597vlxh8lnp3e4td0",
      "ownerCode": "AA",
      "offerExpirationDateTime": "2023-10-13T21:29:56Z",
      "aLaCarteOfferItems": [
        {
          "offerItemID": "cd646d964597vlxh8lnp3e4td0-1",
          "source": "NDC",
          "unitPrice": {
            "totalAmount": {
              "amount": 34.56,
              "currencyCode": "USD"
            },
            "baseAmount": {
              "amount": 32.15,
              "currencyCode": "USD"
            },
            "taxSummary": {
              "totalTaxAmount": {
                "amount": 2.41,
                "currencyCode": "USD"
              },
              "taxes": [
                {
                  "amount": {
                    "amount": 2.41,
                    "currencyCode": "USD"
                  },
                  "taxCode": "US"
                }
              ]
            }
          }
        }
      ],
      "service": {
        "serviceId": "W4B62076F-72AC-4A05-AFC5-1-1-1",
        "serviceDefinitionRefID": "DS20231013T205952436497564800515930"
      },
      "eligibility": {
```

```
"paxRefIDs": [  
  "Passenger1"  
],  
"flightAssociations": {  
  "paxSegmentRefID": [  
    "Isgm5a00d3429a7a6"  
  ]  
}  
}  
},...
```

Note that offers have an expiration date and time, this is typically 20 minutes from the time of the request, seat inventory is not guaranteed.

The next section is “dataLists”, this contains origin & destination, segment, journey, passenger and service definition information, the service definitions have a mapping from the offer and contain the name of the seat product and descriptive text:

```
... "dataLists": {
  "originDests": [
    {
      "originDestID": "O2",
      "originCode": "DFW",
      "destCode": "PHX",
      "paxJourneyRefIDs": [
        "F2"
      ]
    }
  ],
  "paxSegments": [
    {
      "paxSegmentID": "Isgm5a00d3429a7a6",
      "departure": {
        "iataLocationCode": "DFW",
        "aircraftScheduledDateTime": "2023-11-07T16:40:00"
      },
      "arrival": {
        "iataLocationCode": "PHX"
      },
      "marketingCarrierInfo": {
        "carrierDesignatorCode": "AA",
        "carrierName": "American Airlines",
        "marketingCarrierFlightNumberText": "1155",
        "rbdCode": "B"
      },
      "datedOperatingLegs": [
        {
          "datedOperatingLegID": "SABRE_NDCC_LEG_ID_1",
          "carrierAircraftType": {
            "carrierAircraftTypeCode": "321",
            "carrierAircraftTypeName": "Airbus A321"
          },
          "departure": {
            "iataLocationCode": "DFW",
            "aircraftScheduledDateTime": "2023-11-07T16:40:00"
          },
          "arrival": {
            "iataLocationCode": "PHX"
          }
        }
      ]
    }
  ]
}
```

```

    }
  ],
  "paxJourneys": [
    {
      "paxJourneyID": "F2",
      "paxSegmentRefIDs": [
        "S2"
      ]
    }
  ],
  "paxs": [
    {
      "paxID": "Passenger1",
      "ptc": "ADT",
      "loyaltyProgramAccount": [
        {
          "airline": "AA",
          "accountNumber": "6H020M0-1.1"
        }
      ]
    }
  ],
  "serviceDefinitions": [
    {
      "serviceDefinitionID": "DS20231013T205952436497564800515930",
      "ownerCode": "AA",
      "name": "Main Cabin Extra",
      "validatingCarrierCode": "AA",
      "descriptions": [
        {
          "descriptionID": "SABRE_NDCC_ID_1",
          "descriptionText": "Extra legroom (Up to 6 inches)"
        },
        {
          "descriptionID": "SABRE_NDCC_ID_2",
          "descriptionText": "Complimentary beer, wine & spirits"
        },
        {
          "descriptionID": "SABRE_NDCC_ID_3",
          "descriptionText": "Surcharge"
        }
      ],
      "bookingInstructions": {
        "ssrCodes": [
          "SEAT"
        ],
        "method": "SSR",
        "text": [

```

```
        "DD"  
      ],  
      "equipment": "30A|321|321 OASIS(C20/Y170)"  
    },  
    "detail": {  
      "serviceCoupon": {  
        "instantPurchase": "TT",  
        "couponType": "EA"  
      }  
    }  
  }, ...  
}
```

The next section is the “seatMaps” array, the first part of this section describes the layout of the cabin and includes:

- First Row – the first row in the cabin for the booked class of service
- Last Row – the last row in the cabin for the booked class of service
- The column names and positions – column names typically progress from A alphabetically depending on the number of seats in a row, the position values are:
 - W – Window
 - C – Centre
 - A – Aisle
- First and Last rows
- Wing Position
- Exit Row Position

```
... "seatMaps": [  
  {  
    "paxSegmentRefID": "Isgm5a00d3429a7a6",  
    "cabinCompartments": [  
      {  
        "firstRow": 8,  
        "lastRow": 36,  
        "cabinType": {  
          "cabinTypeCode": "Y"  
        },  
        "cabinLayout": {  
          "columns": [  
            {  
              "id": "A",  
              "position": "W"  
            },  
            {  
              "id": "B",  
              "position": "C"  
            },  
            {  
              "id": "C",  
              "position": "A"  
            },  
            {  
              "id": "D",  
              "position": "A"  
            },  
            {  
              "id": "E",  
              "position": "C"  
            },  
            {
```

```

        "id": "F",
        "position": "W"
    }
],
"rows": {
    "firstRow": 8,
    "lastRow": 36
},
"wingRowPosition": [
    {
        "firstRow": 14,
        "lastRow": 22
    }
],
"exitRowPosition": [
    {
        "firstRow": 11,
        "lastRow": 11
    },
    {
        "firstRow": 24,
        "lastRow": 24
    }
]
],
},

```

The “seatRows” section includes details of each seat in the cabin with the following information:

- Row & Column
- Occupancy Status (PADIS composite 9865)
- Characteristics (PADIS composite 9825)
- Offer Item reference ID

```

... "seatRows": [
    {
        "row": 8,
        "seats": [
            {
                "column": "A",
                "occupationStatusCode": "F",
                "characteristics": [
                    {
                        "code": "L",
                        "description": "LegSpaceSeat"
                    },
                    {
                        "code": "CH",

```

```
        "description": "ChargeableSeat"
    },
    {
        "code": "K",
        "description": "BulkheadSeat"
    },
    {
        "code": "LS",
        "description": "LeftSideOfAircraft"
    },
    {
        "code": "W",
        "description": "Window"
    }
],
"offerItemRefIDs": [
    "cd646d964597v1xh81np3e4td0-2"
]
},
```

Booking a Pre-Reserved Seat

To book a pre-reserved seat (paid or free), the “orders/change” API is called, a valid application ID is required in the header. The “seatAdds” array allows for an application to specify multiple seats for the same or different passengers and/or segments.

The following parameters are required in the “orders/change” request:

- Sabre Order ID
- “seatAdds” array with the following values:
 - Offer Item ID
 - Passenger reference
 - Segment reference ID
 - Row number
 - Column

```
{
  "id": "1SXXXCLP17CHD",
  "seatAdds": [
    {
      "offerItemId": "cd646d964597n4xmflnt21y6c0-2",
      "passengerRefs": "Passenger1",
      "segmentRefId": "AA6C62AC-CF60-4B1F-AE3E-4FF1D3B6FCF4",
      "row": "8",
      "column": "A"
    }
  ]
}
```

The response returns a view of the order with the pre-reserved seat as an order item, the pre-reserved will have one of two status codes:

- HK – Confirmed, no EMD required (free seat)
- HD – Confirmed, EMD required (paid seat)

Cancelling a Pre-Reserved Seat

To cancel a pre-reserved seat (paid or free), the “orders/change” API is called, a valid application ID is required in the header.

The following parameters are required in the “orders/change” request:

- Sabre Order ID
- “seatDeletes” array with the Order Item ID(s).

```
{
  "id": "1SXXXCLP17CHD",
  "seatDeletes": [
    {
      "orderItemId": "B09E5DB8-6F28-4B78-966D-1EE1B1491AFF"
    }
  ]
}
```

Note that if the pre-reserved seat has been fulfilled, the airline will void or refund the EMD depending on whether the EMD is in the void or refund period.

Rebooking a Pre-Reserved Seat

To rebook a pre-reserved seat (paid or free), the “orders/change” API is called, a valid application ID is required in the header.

The following parameters are required in the “orders/change” request:

- Sabre Order ID
- “seatDeletes” array with the following values:
 - Order Item ID
- “seatAdds” array with the following values:
 - Offer Item ID
 - Passenger reference
 - Segment reference ID
 - Row number
 - Column

```
{
  "id": "1SXXXCLP17CHD",
  "seatDeletes": [
    {
      "orderItemId": "33CE41C7-DE37-44B3-9383-A49FC29C168B"
    }
  ],
  "seatAdds": [
    {
      "offerItemId": "cd646d964597vlxh8lntehiek0-2",
      "passengerRefs": "Passenger1",
      "segmentRefId": "Isgm5a00d3429a7a6",
      "row": "8",
      "column": "C"
    }
  ]
}
```

After a pre-reserved seat has been added or modified the order is displayed, the pre-reserved seat has its own Order Item with one of the following status codes:

- HK – Confirmed, no EMD required (free seat)
- HD – Confirmed, EMD required (paid seat)

The order item for the pre-reserved seat contains the following:

- Order Item ID
- External ID
- External Order reference ID
- Origin (value used for internal processing)
- Creation Date and Time
- Price – contains base, tax and total amounts
- Services – detailed information of the seat
 - ID number for the service
 - External ID of the service
 - Status code
 - Passenger ID
 - Service definition reference that contains additional information of the seat product
 - Seat selection (row and column)
 - Segment reference the seat is booked on
 - Seat characteristics (per the PADIS code set 9825)
 - External order item ID
- Offer Item ID
- External Offer Item ID
- Item Origin, how the order item was added to the order

```
...{
  "id": "B8285542-D464-4751-BC2F-C69B811E7D24",
  "externalId": "W32164868-422D-4E53-811F-1-2",
  "externalOrderRefId": "2A23942F-13FC-4E93-95BD-B10C10595EDD",
  "origin": "N",
  "creationDateTime": "2023-10-16T21:34:21Z",
  "price": {
    "baseAmount": {
      "amount": "33.61",
      "code": "USD"
    },
    "totalAmount": {
      "amount": "36.13",
      "code": "USD"
    },
    "totalTaxAmount": {
      "amount": "2.52",
      "code": "USD"
    }
  }
}
```

```

    }
  },
  "services": [
    {
      "id": "6F0EC41E-8A7C-4FB4-B653-EE0337AC69ED",
      "externalId": "SRVW32164868-422D-4E53-811F-1-2",
      "statusCode": "HD",
      "passengerRefId": "Passenger1",
      "serviceDefinitionRefId": "B0F81C55-D7B8-49CA-B5C5-46AFB47C0FB9",
      "selectedSeat": {
        "row": "8",
        "column": "C",
        "segmentRefId": "Isgm5a00d3429a7a6",
        "characteristics": [
          "L",
          "CH",
          "K",
          "LS",
          "A"
        ]
      }
    },
    {
      "externalOrderItemId": "W32164868-422D-4E53-811F-1-2"
    }
  ]
},
"offerItemId": "cd646d964597vlxh8lntehiek0-2",
"externalOfferItemId": "W32164868-422D-4E53-811F-1-2",
"itemOrigin": "SELL"
}...

```

The order item contains a service definition reference that provides additional information relating to the service, included are the seat product name and a text-based description.

```
"serviceDefinitions": [  
  {  
    "id": "58480666-C404-4032-9828-74036A445A9B",  
    "externalId": "A-SA-0B5-SA-EXIT",  
    "name": "EXTRA LEGROOM",  
    "serviceCode": "SA",  
    "reasonForIssuanceCode": "A",  
    "reasonForIssuanceSubCode": "0B5",  
    "ownerCode": "QF",  
    "descriptions": [  
      {  
        "text": "EXTRA LEGROOM"  
      }  
    ],  
    "settlementMethodCode": "EA"  
  },  
]
```

Fulfilling a Pre-Reserved Seat

When a pre-reserved paid seat is fulfilled, the airline issues an EMD-A, this type of EMD has coupon level associate to the passenger's ticket, this means there is a dependency on the ticket being issued before payment of the pre-reserved seat. However, it is possible to send a single fulfilment request for an airfare and a paid seat, in this case the airline will issue the ticket and EMD sequentially. An example of this request is in the ["Order Fulfilment Section"](#).

Examples of the "orders/change" Requests to Fulfill a Pre-Reserved Seat

To fulfil a pre-reserved seat assignment the "orders/change" API is called.

Data elements required in the fulfillment request:

- Sabre Order ID
- Total Amount
- Currency Code
- Payment Method
- Order Item Reference IDs of the Pre-Reserved Seat

```
{
  "id": "1SXXXC6UQ1Q84",
  "actions": [
    {
      "fulfillOrder": {
        "paymentInfo": {
          "amount": {
            "amount": "30.00",
            "code": "AUD"
          },
          "paymentMethod": {
            "paymentCard": {
              "cardNumber": "3456XXXXXXXXXX",
              "contactInfoRefId": "CI-1",
              "expirationDate": "0139",
              "vendorCode": "AX",
              "securePaymentVersion2": {
                "paymentTrxChannelCode": "MO"
              }
            }
          }
        },
        "orderItemRefIds": [
          "4EE479FF-77CB-4ED0-A29F-D3802F93ABDF"
        ]
      }
    }
  ]
}
```

```
}
  }
}
]
```

Notes:

1. American Airlines only accepts credit cards as payment for pre-reserved seats.
2. The card number is masked in the example.

EMD information in an order

After an order has been fulfilled, the order will be updated with the “ticketDocumentInfo” section with individual document data elements. The following description describes some of those elements:

- Document Number – thirteen numeric, formed of: airline accounting code (3), form code (4), serial number (6).
- Document Type – values are defined in the [fulfillment introduction](#).
- Issue Date and Time – the date and time the document was issued in Zulu time.
- Number of Booklets – the number of documents issued in conjunction (maximum 4).
- In Connection Document Number – the ticket number this EMD is associated to.
- Reason for Issuance Code – A single alpha character defined by IATA:
 - A - Air Transportation – to be used for all air transportation related sub-codes which include, but not limited to Charter, Involuntary Downgrading and Upgrading.
 - B - Surface Transportation/Non-Air Services- to be used for all surface transportation/non air services related sub-codes which include, but not limited to, Bus and Car Hire.
 - C - Baggage – to be used for all baggage related sub-codes which include, but not limited to, Angling Equipment and Biking Equipment
 - D - Financial Impact – to be used for all financial impact related sub-codes which include, but not limited to, Reservation Change Fee and Refundable Balances
 - E - Airport Services – to be used for all airport services related sub-codes which include, but not limited to, Check In and Lounge Access
 - F - Merchandise – to be used for all merchandise related sub-codes which include, but not limited to T-shirt
 - G - Inflight Services – to be used for all inflight services related sub-codes which include, but not limited to, Sleeper Berth and Beverage
 - H - Reserved for future ATPCO use
 - I - Reserved for individual airline use
- Reason for Issuance Sub-Code – three-character sub-code, defined by ATPCo or the airline.
- Coupons – Contains values at the coupon level.
 - Coupon Number – 1-4, coupons are **not** omitted when there is a surface sector.
 - Coupon Status - values are defined in the [fulfillment introduction](#).
 - Service ID – a reference to the service this coupon was issued for.

- Order Item Reference ID – The Order Item this coupon was issued for.
- Segment Information – Segment Information this coupon was issued for.
- Pricing Information – base amount, total taxes and total.
- Passenger Reference – a reference to the passenger this document was issued for.

```

...{
  "document": {
    "number": "0014576866296",
    "type": "J",
    "issueDateTime": "2023-10-12T11:59:00Z",
    "ticketingLocation": "DFW",
    "numberOfBooklets": 1,
    "routingCode": "D",
    "reportingType": "ARC",
    "inConnectionDocumentNumber": "0015750646616",
    "coupons": [
      {
        "number": 1,
        "status": "I",
        "serviceRefId": "E92034CB-7072-4316-B07B-7205B8F5E05F",
        "orderItemRefId": "0EC20EF4-B44B-4BD2-9CC1-F3A49DDB44AD",
        "inConnectionDocumentNumber": "0015750646616",
        "inConnectionCouponNumber": "1",
        "reasonForIssuanceCode": "A",
        "reasonForIssuanceSubCode": "0B5",
        "segmentInfo": {
          "departureAirport": "DFW",
          "arrivalAirport": "TPA",
          "marketingAirlineCode": "AA",
          "departureTime": "2023-10-31T13:13:00",
          "arrivalTime": "2023-10-31T16:37:00",
          "flightNumber": 2504,
          "classOfService": "B"
        }
      }
    ]
  },
  "price": {
    "baseAmount": {
      "amount": "24.76",
      "code": "USD"
    },
    "totalAmount": {
      "amount": "26.62",
      "code": "USD"
    },
    "totalTaxAmount": {

```

```
        "amount": "1.86",
        "code": "USD"
    },
    "paxRefId": "Passenger1"
},
```

Notes:

1. An EMD-A (type J) issued for ancillary services will be in an Open state (I) when they are issued, whereas an EMD-S (type Y) issued for miscellaneous fees and charges may be considered “consumed at issuance” and be set to Flow\Used (B). The status of the Order Item will be set to “HI”.

Updates to the PNR

After an Order Item for a pre-reserved seat has been fulfilled, an “MO” (Miscellaneous Order Item) ticket line is added to the PNR:

*T

TKT/TIME LIMIT

1. T-24OCT-27M7*ABC
2. TO 0815750722221-XX QUEBE/Q 27M7*ABC 0745/24OCT*
3. MO 0811818014437-XX QUEBE/Q 27M7*ABC 0745/24OCT*A

- Accounting lines for invoicing

*PAC

ACCOUNTING DATA

- | | | | | |
|----|--------------------------|----------|---------|----------------|
| 1. | QF¥5750722221/ | 0.00/AUD | 537.71/ | 77.63/ONE/CA 1 |
| | .1QUEBEC QUENTIN/1/D/E | | | |
| 2. | QF¥1818014437/ | 0.00/AUD | 9.09/ | 0.91/ONE/CA 1 |
| | .1QUEBEC QUENTIN/1/D/EMD | | | |

Canceling an Order Item for a Fulfilled Pre-Reserved Seat

Ancillary services are fulfilled using Electronic Miscellaneous Documents (EMDs), these are reported to the settlement systems the same as tickets, they are also subject to the same void and refund windows.

To cancel a fulfilled pre-reserved seat use the [“orders/change”](#) API to cancel the seat order item, if you use the [“offers/reshop/cancelOrder”](#) API and specify the Order Item for the pre-reserved seat the offer returned will be for cancellation of whole order (air segments and pre-reserved seats).

After a fulfilled Order Item for a pre-reserved seat has been deleted the order is displayed and the EMD will have either a status of either “V” (void) or “RF” (refund).

```
{
  "document": {
    "number": "0811818014995",
    "type": "J",
    "issueDateTime": "2023-10-25T03:08:00Z",
    "ticketingLocation": "BNE",
    "numberOfBooklets": 1,
    "reportingType": "BSP",
    "inConnectionDocumentNumber": "08157507246453",
    "coupons": [
      {
        "number": 1,
        "status": "V",
        "inConnectionDocumentNumber": "0815750724645",
        "inConnectionCouponNumber": "1",
        "reasonForIssuanceCode": "A",
        "reasonForIssuanceSubCode": "0B5",
        "segmentInfo": {
          "departureAirport": "MEL",
          "arrivalAirport": "CBR",
          "marketingAirlineCode": "QF",
          "departureTime": "2023-11-28T19:55:00",
          "arrivalTime": "2023-11-28T21:00:00",
          "flightNumber": 1530,
          "classOfService": "E"
        }
      }
    ]
  },
  "price": {
    "baseAmount": {
      "amount": "13.64",
      "code": "AUD"
    }
  }
}
```

```
    },
    "totalAmount": {
      "amount": "15.00",
      "code": "AUD"
    },
    "totalTaxAmount": {
      "amount": "1.36",
      "code": "AUD"
    }
  },
  "paxRefId": "Passenger1"
}
```

Note that as the EMD is issued after the ticket, it is possible that the ticket is in the refund period and the EMD is in the void period.

Updates to the PNR

After an Order Item for a pre-reserved seat has been deleted, an "OV" (Order Item voided) ticket line is added to the PNR:

```
*T
TKT/TIME LIMIT
1.T-24OCT-27M7*ABC
2.TO 0815750722221-XX QUEBE/Q 27M7*ABC 0745/24OCT*
3.MO 0811818014437-XX QUEBE/Q 27M7*ABC 0745/24OCT*A
```

Ancillary Services

Displaying a List of Ancillary Services

Ancillary services are services offered to passengers in addition to the air fare, this may include:

- Checked bag fees
- Meals
- Mobility Assistance (wheelchairs)
- Upgrades
- Carbon Off Sets

These services may be free or charged depending on the type of service and the itinerary (some meals are complimentary on long haul flights).

The “offers/getAncillaries” endpoint

The “offers/getAncillaries” API uses the “post” method, the current version of the “offers/getAncillaries” API is V2

Certification environment:

<https://api.cert.platform.sabre.com/v2/offers/getAncillaries>

Production environment:

<https://api.platform.sabre.com/v2/offers/getAncillaries>

Note: The “offers/getAncillaries” API requires an application ID to obtain bookable offers, if this is not included in the header then the offers returned will be informational only. Refer to the [Getting Started](#) section to find out more about Application IDs.

Request a List of Ancillary Services with an Order ID

To display a list of available ancillary services the “getAncillaries” API is used, this API provides a list of free and chargeable services that may be requested.

The input parameters are:

- The request type (orderId)
- The Sabre Order ID

```
{  
  "requestType": "orderId",  
  "request": {  
    "orderId": "1SXXXITLMWT5A"  
  }  
}
```

Note that unlike the “getSeats” API, Offer IDs are not supported as an input parameter.

Reading the “getAncillaries” Response

The “getAncillaries” response contains free and paid services, the example shown is a free service, it is likely that multiple services will be returned.

```
{
  "ancillaries": {
    "segments": [
      {
        "id": "Isgm5300c6e78e9b5",
        "bookingAirlineCode": "AA",
        "bookingFlightNumber": 2510,
        "departureAirportCode": "TPA",
        "arrivalAirportCode": "DFW",
        "departureDate": "2024-01-24",
        "bookingClassCode": "O"
      }
    ],
    "passengers": [
      {
        "passengerId": "Passenger1",
        "passengerTypeCode": "ADT",
        "givenName": "CHICKY",
        "surname": "CHARLIE"
      }
    ],
    "offer": {
      "offerId": "ce64ccdd878cx259slq2yqld10",
      "otherServices": [
        {
          "offerItemId": "ce64ccdd878cx259slq2yqld10-1",
          "serviceDefinitionRef": "ancillary_definition_20",
          "segmentRefs": [
            "Isgm5300c6e78e9b5"
          ],
          "passengerRefs": [
            "Passenger1"
          ]
        }
      ]
    },
    "serviceDefinitions": [
      {
        "id": "ancillary_definition_20",
        "serviceCode": "VGML",
        "airlineCode": "AA",

```

```
    "commercialName": "Vegetarian meal",
    "groupCode": "ML",
    "additionalInputRequirements": {
      "pattern": "%FREETEXT%",
      "variables": [
        {
          "key": "FREETEXT",
          "value": "^$"
        }
      ]
    }
  }
}
```

Notes

Not all services are available on all flights, and in some cases, services may be inventory controlled.

Some services require the agent to add more information, the pattern is defined with two possible values:

- "value": "^\$" – free text is not permitted.
- "value": "[A-Z0-9\\- /]*" – free text is mandatory, valid characters are A-Z, 0-9, "/" (forward slash), "-" (dash) and " " (space).

Booking and Cancelling Ancillary Services

Ancillary services are returned as offers with offer IDs from the “offers/getAncillaries” these offer IDs are required to book the services. Once booked, an ancillary service is added to an order with an order Item ID, the order Item ID is used to cancel the ancillary service.

Booking Ancillary Services

To book an ancillary service the “orders/change” API is used, a valid application ID is required in the header.

Supported parameters in the request are:

- The Sabre Order ID
- The “serviceAdds” array with the following elements:
 - The offer Item ID for the service you want to reserve.
 - The passenger reference.
 - Additional text (when required).

```
{
  "id": "1SXXX7HYW8DGF",
  "serviceAdds": [
    {
      "offerItemId": " ce64ccdd878cx259slq2yqldl0-1",
      "passengerRefId": "Passenger1",
      "productText": ""
    }
  ]
}
```

If an ancillary service requires additional text this is specified in the “offers/getAncillaries” response along with the pattern, the booking method is per this example:

```
{
  "id": "1SXXX7HYW8DGF",
  "serviceAdds": [
    {
      "offerItemId": " ce5b4fd69b46x4qh9lsce3y2v0-9",
      "passengerRefId": "Passenger1",
      "productText": "PASSENGER REQUIRES ASSISTANCE TO SEAT"
    }
  ]
}
```

Note

If an agent attempts to add a duplicate service for a passenger the following error is returned:

```
{
  "errors": [
    {
      "code": "060265",
      "message": "This order already has a WCHR for passenger Passenger1 on segment
Isgm2d00c869d786d"
    }
  ]
}
```

The response to the ancillary service request is a display of the full order, the ancillary service is an order item within the order:

```
{
  "id": "3A5DA875-E3EA-4329-8B0C-F4C2E66EDCDC",
  "externalId": "T1-S3-QF-DPNA",
  "externalOrderRefId": "71F42106-D0C8-4B51-936B-948798E509C9",
  "statusCode": "Fulfill",
  "origin": "N",
  "creationDateTime": "2024-02-07T23:10:24Z",
  "services": [
    {
      "id": "F7CE1B63-0C4B-489B-ABC1-B2815738EAE8",
      "externalId": "T1-S3-QF-DPNA",
      "statusCode": "PN",
      "passengerRefId": "Passenger1",
      "segmentRefId": "Isgmd201c08b112a7",
      "serviceDefinitionRefId": "B3273E09-AA68-461D-AA12-09FB18EB10A8"
    }
  ],
  "offerItemId": "ce5b4fd69b46x4qh9lsce3y2v0-9",
  "itemOrigin": "SELL"
}
```

Further information is detailed in the service definition referenced by the “serviceDefinitionRefId”:

```
"serviceDefinitions": [
  {
    "id": "B3273E09-AA68-461D-AA12-09FB18EB10A8",
    "externalId": "T1-S3-QF-DPNA",
    "name": "Disabled Passenger Needing Assistance",
    "serviceCode": "DPNA",
    "ownerCode": "QF",
```

```
    "bookingInstructions": {
      "specialServices": [
        {
          "specialServiceCode": "DPNA",
          "freeText": "PASSENGER REQUIRES ASSISTANCE TO SEAT"
        }
      ]
    }
  },
```

Canceling Ancillary Services

To cancel an ancillary service, the “orders/change” API is used, and the “orderItemDeletes” array is used to specify the order items that are no longer required:

```
{
  "id": "1SXXXC3K7EWN1",
  "orderItemDeletes": [
    {
      "id": "3A5DA875-E3EA-4329-8B0C-F4C2E66EDCDC"
    }
  ]
}
```

The response is a full display of the order, the deleted order items are no longer displayed.

Voluntary Changes

A voluntary change is a change to an order initiated by the passenger and is accomplished by a three-step process consisting of: “re-shopping” for new offers, rebooking the itinerary and updating the passenger’s ticket with the new flights and price information.

Re-shopping for New Air Offers

To initiate the “re-shop” request the “offers/reshop/shop” API is used, the Sabre Order ID is included along with new shopping criteria and instructions on which services should be retained, e.g., for a round trip it may be that the passenger only wants to change the return flight, in which case the outbound flight is retained. It is also possible to re-shop an order that has one or more flight segments that have already been flown.

The “offers/reshop/shop” endpoint

The “offers/reshop/shop” API uses the “post” method, the current version of the “offers/reshop/shop” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/offers/reshop/shop>

Production environment:

<https://api.platform.sabre.com/v1/offers/reshop/shop>

Examples of the “offers/reshop/shop” Request

The “offers/reshop/shop” API allows for all or part of an itinerary to be “re-shopped”, supported input parameters are:

- The Sabre Order ID
- One or More new Origin, Destination and Departure Date(s)
- Service IDs to be Retained (optional)

The Origins and Destinations array allows for an agent to specify departure points, arrival points and dates of travel:

```
{
  "request": {
    "orderId": "1SXXXXC70F9D8H",
    "originDestinations": [
      {
        "newOriginDestination": {
          "departureCode": "SYD",
          "departureDate": "2023-11-22T18:35:42",
          "arrivalCode": "MEL"
        }
      },
      {
        "newOriginDestination": {
          "departureCode": "MEL",
          "departureDate": "2023-11-24T07:35:42",
          "arrivalCode": "SYD"
        }
      }
    ]
  }
}
```

To re-shop just one bound or if the outbound flight has flown, the origin, destination and date of travel are specified along with the Service ID of any segments that are not being changed.

```
{
  "request": {
    "orderId": "1SXXXC70F9D8H",
    "originDestinations": [
      {
        "newOriginDestination": {
          "departureCode": "MEL",
          "departureDate": "2023-11-24T07:35:42",
          "arrivalCode": "SYD"
        }
      },
      {
        "retain": {
          "serviceId": "918793AD-ED0B-4399-9DE0-56C222F4F39A"
        }
      }
    ]
  }
}
```

Notes:

No other shopping qualifiers are currently supported.

Sections of the “offers/reshop/shop” response

The “offers/reshop/shop” response contains flight schedules and offers based on the new itinerary and the exchange value of the ticket.

The response contains the following sections:

- Messages
- Response
 - DataList
 - segments
 - passengers
 - baggages
 - priceClasses
 - voluntaryChangeInformation
 - Offers
 - offerId
 - source
 - isSellable
 - paymentType
 - offerExpirationDateTime
 - offerExpirationInSeconds
 - paymentTimeLimitText
 - ownerCode
 - validatingAirlineCode
 - offerItems

Messages

Informational messages from the airlines or the service, this may include diagnostic data.

```
"messages": [  
  {  
    "category": "WARNING",  
    "type": "PROCESSING_WARNING",  
    "description": "QF: Qantas Frequent Flyers could earn Qantas Points and Status Credits for  
this booking. <a href=https://www.qantas.com/fflyer/dyn/program/terms target=_blank>Terms  
and conditions apply.</a>",  
    "service": "NDCC_ORDER_RESHOP"  
  }  
],
```

Response

The “Response” section contains two sections, “datalist” and “offers”, the “datalist” is repeatable data for flight segments, passengers, baggage and price classes, each of the elements within this section has a reference that is used in the offers.

datalist

The following sections are contained in “datalist”:

- Segments - An array with the flight schedule options that may be booked with a flight segment reference.

```
"segments": [
  {
    "segmentId": "S1",
    "departure": {
      "airportCode": "SYD",
      "airportName": "Sydney Kingsford Smith Apt, AU",
      "terminal": "3",
      "departureDateTime": "2023-11-21T20:00:00"
    },
    "arrival": {
      "airportCode": "MEL",
      "airportName": "Melbourne Airport, AU",
      "arrivalDateTime": "2023-11-21T21:35:00"
    },
    "marketingAirline": {
      "code": "QF",
      "name": "Qantas Airways",
      "flightNumber": "493"
    },
    "datedOperatingLegs": [
      {
        "departure": {
          "airportCode": "SYD",
          "airportName": "Sydney Kingsford Smith Apt, AU",
          "terminal": "3",
          "departureDateTime": "2023-11-21T20:00:00"
        },
        "arrival": {
          "airportCode": "MEL",
          "airportName": "Melbourne Airport, AU",
          "arrivalDateTime": "2023-11-21T21:35:00"
        },
        "carrierAircraftType": {
```

```

        "code": "73H",
        "name": "Boeing 737-800 -Winglets- Passenger"
    }
}
]
},

```

Note that if the re-shop request contains a “serviceld” to be retained, then the only segment(s) for that leg will be the flight(s) that the agent wants to keep from the current order.

- Passengers - An array with passenger references and the priced passenger type code.

```

"passengers": [
  {
    "passengerId": "Passenger1",
    "typeCode": "ADT"
  }
],

```

- Baggages – An array with information relating to checked and carry-on baggage.

```

"baggages": [
  {
    "baggageId": "BADABA69C9-CO",
    "typeCode": "CarryOnBag",
    "descriptions": [
      "CARRYON HAND BAGGAGE ALLOWANCE"
    ],
    "baggageAllowance": [
      {
        "allowanceCategory": "Piece",
        "maximumPieces": 2
      }
    ],
    "allowanceQuantity": 2
  },
  {
    "baggageId": "BAC435E125",
    "typeCode": "CheckedBag",
    "descriptions": [
      "CHECKED ALLOWANCE"
    ],
    "baggageAllowance": [
      {
        "allowanceCategory": "Piece and Weight",
        "maximumPieces": 1,

```



```

        "maximumBagWeightInPounds": 50,
        "maximumBagWeightInKilograms": 23
    }
],
    "allowanceQuantity": 1
}
],

```

- priceClasses – the price class information is implemented as an array and provides the brand code, brand name and descriptive text about the brand, this information is returned by the airline and as a result will vary based on each airline’s implementation.

```

"priceClasses": [
  {
    "priceClassId": "QFFlex-1",
    "code": "FLEX",
    "name": "Flex",
    "descriptions": [
      {
        "description": "1. Changes:"
      },
      {
        "description": "- Changes are permitted"
      },
      {
        "description": "- New fare must be of equal or higher value"
      },
      {
        "description": "2. Cancellations:"
      },
      {
        "description": "- Permitted with a cancellation fee of AUD99.00, inclusive of GST"
      },
      {
        "description": "- Ticket value can be held in credit"
      },
      {
        "description": "3. Inclusions on Qantas operated flights:"
      },
      {
        "description": "- Included bags"
      },
      {
        "description": "- Carry On Baggage Allowance - All classes, one piece up to 10kgs and up to 7kgs Dash 8 services"
      },
      {

```

```

        "description": "4.Qantas Frequent Flyer benefits:"
    },
    {
        "description": "- Earn category is flexible economy"
    },
    {
        "description": "Upgrade eligibility:"
    },
    {
        "description": "- Eligible for upgrade"
    },
    {
        "description": "- Subject to availability"
    },
    {
        "description": "Lounge access:"
    }
]
},

```

- **voluntaryChangeInformation** – this array contains fare rule information, the “changeId” value is included at the offer level. Whether changes are permitted, if there is a fee, the fee amount and the applicability.

```

"voluntaryChangeInformation": [
  {
    "changeId": "8c0e77c0-0207-4ca4-b1af-350630f05e7c",
    "type": "Cancel",
    "isAllowed": true,
    "hasFee": true,
    "feeAmount": "99.00",
    "feeCurrencyCode": "AUD",
    "applicabilityList": [
      "Before Departure"
    ]
  }
],

```

Note that these values are only populated if they are included in the response from the airline.

The next section of the “Response” is “Offers”; this contains the re-shop offers including references to the elements in the datalist.

Offers

The following sections are contained in the “offers” array:

- offerId - The Sabre “Offer ID” consisting of the response ID and the sequential number of the offer.

```
"offers": [  
  {  
    "offerId": "ce098bb8c8fc19cx6lph8lqiq9-1",
```

- source - The name of the data source, this will state “NDC” for NDC offers.

```
"source": "NDC",
```

- isSellable - This value states whether the offers in the re-shop response are valid for booking using the “orders/change” request or whether this is an informational response. Some airlines require a secondary pricing step, if this value will be set to “false” the secondary [“offers/reshop/price”](#) request is required.

```
"isSellable": true,
```

- paymentType - This value will be set to either “Deferred” or “Instant”, if this is instant, then the payment information must be included in the request to book this service.

```
"paymentType": "deferred",
```

- offerExpirationDateTime - The time in GMT/Zulu time when the offer will expire.

```
"offerExpirationDateTime": "2023-10-30T17:42:15Z",
```

- offerExpirationInSeconds – The amount of time in seconds that the offer is valid.

```
"offerExpirationInSeconds": 1198,
```

- paymentTimeLimitText – A text-based representation of when fulfillment is required, historically this has been called the “ticket time limit”, this field is only returned as a text field if the airline does not include a time zone in the response.

```
"paymentTimeLimitText": "2023-11-03T23:59:00",
```

- ownerCode – the airline that has returned the offer.

```
"ownerCode": "QF",
```

- validatingAirlineCode – the airline that will be the validating carrier on the ticket.

```
"validatingAirlineCode": "QF",
```

- offerItems – Contains the following sections and is implemented as an array.

```
"offerItems": [  
  {
```

- offerItemId - The “Offer Item ID”, a unique identifier, multiple “Offer Item IDs” will be in the response if multiple PTCs were priced.

```
"offerItemId": "ce098bb8c8fc19cx6lph8lqiq9-1-1",
```

- isMandatory – States whether “Offer Items” can be booked individually or if it is mandatory to include all Offer Items in the rebooking request.

```
"isMandatory": true,
```

- orderItemRefIds – The external order item references.

```
"orderItemRefIds": [  
  "P4C8EF73-6C80-473C-9B8Eko8obq1qf7w3n-1-1"  
],
```

- journeys – References to the segments and price classes for this offer from the definitions within “dataLists”.

```
"journeys": [  
  {  
    "segmentRefIds": [  
      "S6"  
    ],  
    "priceClassRefId": "QFFlex-1"  
  }  
],
```

- passengers – the passengers array contains the following elements:

```
"passengers": [  
  {
```

- passengerRefId – A unique ID used to reference a specific passenger.

```
"passengerRefId": "Passenger1",
```

- pricedTypeCode - The PTC that the airline used to price the offer.

```
"pricedTypeCode": "ADT",
```

- price - The total amount, base amount (fare), total taxes and the tax breakdown with the tax code, a text description and the “taxPadisCode”, this last element is defined in the PADIS Code Set directory under composite/element 5305 and is used to specify if a tax is new, paid, refunded or forfeited.

```
"price": {  
  "totalAmount": {  
    "amount": "638.37",  
    "currencyCode": "AUD"  
  },  
  "baseAmount": {  
    "amount": "553.84",  
    "currencyCode": "AUD"  
  },  
  "taxSummary": {  
    "totalTaxAmount": {  
      "amount": "84.53",  
      "currencyCode": "AUD"  
    },  
    "breakdown": [  
      {  
        "taxAmount": {  
          "amount": "10.29",  
          "currencyCode": "AUD"  
        },  
        "taxCode": "QR",  
        "nationCode": "AU",  
        "description": "PASSENGER SERVICE CHARGE DOMESTIC DEPARTURE",  
        "taxPadisCode": "701"  
      },  
      {  
        "taxAmount": {  
          "amount": "5.99",  
          "currencyCode": "AUD"  
        }  
      }  
    ]  
  }  
}
```

```

    },
    "taxCode": "QR",
    "nationCode": "AU",
    "description": "PASSENGER SERVICE CHARGE DOMESTIC ARRIVAL",
    "taxPadisCode": "701"
  },
  {
    "taxAmount": {
      "amount": "55.38",
      "currencyCode": "AUD"
    },
    "taxCode": "UO",
    "nationCode": "AU",
    "description": "GOODS AND SERVICES TAX GST",
    "taxPadisCode": "701"
  },
  {
    "taxAmount": {
      "amount": "2.65",
      "currencyCode": "AUD"
    },
    "taxCode": "UO",
    "nationCode": "AU",
    "description": "GOODS AND SERVICES TAX GST",
    "taxPadisCode": "701"
  },
  {
    "taxAmount": {
      "amount": "10.22",
      "currencyCode": "AUD"
    },
    "taxCode": "WG",
    "nationCode": "AU",
    "description": "SAFETY AND SECURITY CHARGE DEPARTURE",
    "taxPadisCode": "701"
  }
]
}
},

```

- fareComponents – Contains: the pricing information of a portion of an itinerary between two fare construction points, the fare basis code, the fare type code, reservations booking designator (RBD), cabin code and name, fare rules, ticketing endorsements, flight segment references, the brand code, name & description text and the references for voluntary changes.

```
"fareComponents": [
```

```

{
  "fareBasis": {
    "code": "KFQA",
    "fareDescription": "Published fare",
    "fareCode": "70J",
    "reservationBookingDesignators": [
      "K"
    ],
    "cabinTypes": [
      {
        "code": "Y",
        "name": "Economy",
        "carrierCabinCode": "Y",
        "carrierCabinName": "Economy"
      }
    ]
  },
  "segmentRefIds": [
    "S6"
  ],
  "priceClassRef": "QFFlex-1",
  "voluntaryChangeInformationRefs": [
    "8c0e77c0-0207-4ca4-b1af-350630f05e7c",
    "a7fdcb4f-de83-4b0e-9853-0a232e5161a4",
    "4899e551-c664-42fc-9ad9-eff30853d87f",
    "d35c13c5-e266-4c95-b32b-d7a0c3ff0d5e"
  ]
}
],

```

- baggages – The baggage and segment references.

```

"baggages": [
  {
    "baggageRefId": "BA7977FBDA",
    "segmentRefIds": [
      "S6"
    ]
  },
  {
    "baggageRefId": "BA734F3707-CO",
    "segmentRefIds": [
      "S6"
    ]
  }
]

```

```
],
```

- **price** – The total price of this offer item, this is separated as the pricing in the “passengers” section is per passenger and for a specific PTC.

```
"price": {  
  "totalAmount": {  
    "amount": "638.37",  
    "currencyCode": "AUD"  
  },  
  "baseAmount": {  
    "amount": "553.84",  
    "currencyCode": "AUD"  
  },  
  "taxSummary": {  
    "totalTaxAmount": {  
      "amount": "84.53",  
      "currencyCode": "AUD"  
    }  
  }  
},
```

- **originalOrderItemDifferential** – A recap of the original order item.

```
"originalOrderItemDifferential": {  
  "differentialAmount": {  
    "amount": "553.84",  
    "currencyCode": "AUD",  
    "type": "Base"  
  },  
  "taxSummary": {  
    "totalTaxAmount": {  
      "amount": "84.53",  
      "currencyCode": "AUD"  
    }  
  },  
  "breakdown": [  
    {  
      "taxAmount": {  
        "amount": "10.29",  
        "currencyCode": "AUD"  
      },  
      "taxCode": "QR",  
      "nationCode": "AU",  
      "description": "PASSENGER SERVICE CHARGE DOMESTIC DEPARTURE"  
    },  
    {
```



```

    "taxAmount": {
      "amount": "5.99",
      "currencyCode": "AUD"
    },
    "taxCode": "QR",
    "nationCode": "AU",
    "description": "PASSENGER SERVICE CHARGE DOMESTIC ARRIVAL"
  },
  {
    "taxAmount": {
      "amount": "55.38",
      "currencyCode": "AUD"
    },
    "taxCode": "UO",
    "nationCode": "AU",
    "description": "GOODS AND SERVICES TAX GST"
  },
  {
    "taxAmount": {
      "amount": "2.65",
      "currencyCode": "AUD"
    },
    "taxCode": "UO",
    "nationCode": "AU",
    "description": "GOODS AND SERVICES TAX GST"
  },
  {
    "taxAmount": {
      "amount": "10.22",
      "currencyCode": "AUD"
    },
    "taxCode": "WG",
    "nationCode": "AU",
    "description": "SAFETY AND SECURITY CHARGE DEPARTURE"
  }
]
}
},

```

- **newOfferItemDifferential** – Summary of the new offer showing the base amount and total taxes.

```

"newOfferItemDifferential": {
  "differentialAmount": {
    "amount": "553.84",
    "currencyCode": "AUD",
    "type": "Base"
  }
}

```

```

    },
    "taxSummary": {
      "totalTaxAmount": {
        "amount": "84.53",
        "currencyCode": "AUD"
      }
    }
  },
},

```

- differentialAmountDue – There are two sections:
 - differentialAmount - The amount due from the passenger if the new offer is accepted.
 - balanceDifferentialAmount - The amount due to the passenger, this can be zero or a negative amount and may include refundable taxes.

Example 1. – Even exchange – the price of the new ticket is the same as the original ticket, there is no differential due from the passenger and there is nothing due to the passenger from the airline.

```

"differentialAmountDue": {
  "differentialAmount": {
    "amount": "0.00",
    "currencyCode": "AUD"
  },
  "balanceDifferentialAmount": {
    "amount": "0.00",
    "currencyCode": "AUD"
  }
}

```

Example 2. – Additional Collection – the price of the new ticket is higher than the original ticket, there is an amount due from the passenger which includes a tax amount.

```

"differentialAmountDue": {
  "differentialAmount": {
    "amount": "374.71",
    "currencyCode": "AUD"
  },
  "balanceDifferentialAmount": {
    "amount": "0.00",
    "currencyCode": "AUD"
  },
  "taxSummary": {
    "totalTaxAmount": {
      "amount": "34.07",
      "currencyCode": "AUD"
    }
  }
},

```

```

    "breakdown": [
      {
        "taxAmount": {
          "amount": "34.07",
          "currencyCode": "AUD"
        },
        "taxCode": "UO",
        "nationCode": "AU",
        "description": "GOODS AND SERVICES TAX GST"
      }
    ]
  }
}

```

Example 3. - Residual Value – The price of the new ticket is lower than the price of the original ticket and there is an amount due to the passenger as a refund including a tax.

```

"differentialAmountDue": {
  "differentialAmount": {
    "amount": "0.00",
    "currencyCode": "USD"
  },
  "balanceDifferentialAmount": {
    "amount": "-49.00",
    "currencyCode": "USD"
  },
  "taxSummary": {
    "totalTaxAmount": {
      "amount": "3.42",
      "currencyCode": "USD"
    },
    "breakdown": [
      {
        "taxAmount": {
          "amount": "3.42",
          "currencyCode": "USD"
        },
        "taxCode": "US",
        "nationCode": "US",
        "description": "U.S.A Transportation Tax"
      }
    ]
  }
}

```

- **totalPrice** - This is a summation of the offer items within the offer, note that there may be multiple offer items if there is more than one passenger type code.

```
"totalPrice": {
  "totalAmount": {
    "amount": "638.37",
    "currencyCode": "AUD"
  }
},
"differentialAmountDue": {
  "differentialAmount": {
    "amount": "0.00",
    "currencyCode": "AUD"
  },
  "balanceDifferentialAmount": {
    "amount": "0.00",
    "currencyCode": "AUD"
  }
}
```

The Second Reshop Request

If the initial reshop response returns the “isSellable” value as “false” a secondary pricing step is required, this is due to a variance in the exchange shopping workflow used by some airlines. To complete this step the “offers/reshop/price” API is used.

The “offers/reshop/price” endpoint

The “offers/reshop/price” API uses the “post” method, the current version of the “offers/reshop/price” API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/offers/reshop/price>

Production environment:

<https://api.platform.sabre.com/v1/offers/reshop/price>

Example of the “offers/reshop/price” Request

The “offers/reshop/price” request is used as a secondary request for airlines that return informational offers in the primary shopping response, the only supported input parameter is an Offer ID from the first reshop response.

```
{
  "request": {
    "offerId": "ce5b4fd69b46x4qh91sdbnhod2-1"
  }
}
```

The response to the “offers/reshop/price” request has the same data structure as the [“offers/reshop/shop”](#) response, though it will contain only a single offer.

Rebooking a Passenger

To rebook a passenger on new flights the “orders/change/exchange”, API is used, upon receipt of the request the airline cancels the existing flight segments and books the new flights, flights that were specified to be retained in the reshop requests are not changed.

The “orders/change/exchange” endpoint

The “orders/change/exchange” API uses the “post” method, the current version of the “orders/change/exchange” API is V1.

Certification environment:

<https://api.cert.platform.sabre.com/v1/orders/change/exchange>

Production environment:

<https://api.platform.sabre.com/v1/orders/change/exchange>

The request contains the Sabre Order ID, the "orderItemUpdates" array with details of the offers to be accepted in the "acceptOffers" array. This includes the Offer ID and the Offer Item IDs with the passenger reference ID.

```
{
  "orderId": "1SXXXCDQD0956",
  "orderItemUpdates": [
    {
      "acceptOffers": [
        {
          "offerId": "cd6db9897bc5kmzzq19ooz4188-1",
          "selectedOfferItems": [
            {
              "id": "cd6db9897bc5kmzzq19ooz4188-1-1",
              "passengerRefIds": [
                "Passenger1"
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

After the passenger has been rebooked the system responds with a view of the order, in some circumstances the airline will revalidate the ticket to reflect the new itinerary, this only occurs if the route and price are the same and the airline supports the automatic revalidation capability. If the ticket is not updated by the airline a flag is set at the document level and the agent must process a fulfilment request to exchange the ticket.

This example is from the view of the order after the passenger has been rebooked and shows the out of sync indicator "outOfSyncWithOrderItem": true:

```
"ticketingDocumentInfo": [
  {
    "document": {
      "number": "0815751711020",
      "type": "702",
      "issueDateTime": "2024-01-03T06:17:00Z",
      "ticketingLocation": "BNE",
      "numberOfBooklets": 1,
      "reportingType": "BSP",
      "coupons": [
        {
          "number": 1,
          "status": "I",
          "segmentInfo": {
```



```
        "departureAirport": "SYD",
        "arrivalAirport": "MEL",
        "marketingAirlineCode": "QF",
        "departureTime": "2024-02-28T22:00:00",
        "arrivalTime": "2024-02-28T23:35:00",
        "flightNumber": 499,
        "classOfService": "I"
    }
},
{
    "number": 2,
    "status": "I",
    "segmentInfo": {
        "departureAirport": "MEL",
        "arrivalAirport": "SYD",
        "marketingAirlineCode": "QF",
        "departureTime": "2024-03-04T10:40:00",
        "arrivalTime": "2024-03-04T11:50:00",
        "flightNumber": 436,
        "classOfService": "I"
    }
}
],
"outOfSyncWithOrderItem": true
},
```

Exchanging a Ticket

Once a passenger has been rebooked the next step is to exchange the ticket to match the new itinerary, as this is fundamentally fulfilment of an order the “orders/change” API is used. Refer to [“Fulfilling an Order”](#) for more information.

In the case of an even exchange when nothing is due to the airline, the form of payment should be specified as cash.

The response to the fulfilment request is a view of the order, this contains both the original ticket information with the coupon status set to “E” (Exchanged) and the new ticket with coupon status as either “I” or “OPE” (Open for use) depending on the airline.

This extract from a view of an order shows the original and new ticket information:

```
"ticketingDocumentInfo": [
  {
    "document": {
      "number": "6182445846069",
      "type": "T",
      "issueDateTime": "2024-01-04T00:00:00Z",
      "numberOfBooklets": 1,
      "reportingType": "BSP",
      "coupons": [
        {
          "number": 1,
          "status": "E"
        }
      ]
    },
    "price": {
      "baseAmount": {
        "amount": "260.00",
        "code": "AUD"
      },
      "totalAmount": {
        "amount": "329.10",
        "code": "AUD"
      },
      "totalTaxAmount": {
        "amount": "69.10",
        "code": "AUD"
      }
    },
    "paxRefId": "Passenger1"
  }
]
```

```

},
{
  "document": {
    "number": "6182445846077",
    "type": "T",
    "issueDateTime": "2024-01-04T00:00:00Z",
    "numberOfBooklets": 1,
    "reportingType": "BSP",
    "coupons": [
      {
        "number": 1,
        "status": "I",
        "serviceRefId": "232948EE-50BC-4CA5-AD1E-4D9AFF9C600A",
        "orderItemRefId": "5159F0E4-7233-49D5-84F5-DC9DD10058C8",
        "segmentInfo": {
          "departureAirport": "SIN",
          "arrivalAirport": "HKG",
          "marketingAirlineCode": "SQ",
          "departureTime": "2024-03-22T13:05:00",
          "arrivalTime": "2024-03-22T17:05:00",
          "flightNumber": 894,
          "classOfService": "E"
        }
      }
    ]
  },
  "price": {
    "baseAmount": {
      "amount": "633.00",
      "code": "AUD"
    },
    "totalAmount": {
      "amount": "373.00",
      "code": "AUD"
    },
    "totalTaxAmount": {
      "amount": "69.10",
      "code": "AUD"
    }
  },
  "originalIssueInfo": {
    "issueDate": "2024-01-04",
    "ticketNumber": "6182445846069"
  },
  "paxRefId": "Passenger1"
}

```

Additional Servicing Capabilities

In addition to primary servicing capabilities, a set of functions have been enabled to support the following use cases:

- Unused Ticket Credits – Cancelling an order and retaining a ticket for future use and the subsequent application of that credit towards a new order.
- Voiding Tickets – Voiding a ticket in case the passenger wishes to use a different form of payment or if the passenger may not travel.
- Order Synchronization – Synchronizing an Order between the airline and the Sabre Order Management System (OMS).

Unused Ticket Credits

In certain circumstances when an order is cancelled it may be preferable to retain the ticket for future use, this is a common practice when a ticket is non-refundable but has exchange value. This process is sometimes referred to as “credit on hold” or “future flight credit” and may be applied to completely unused or partially used tickets.

Canceling an Order and Retaining a Ticket for Future Use

To cancel an order and retain the ticket for future use, the “orders/change” API is used, the required parameters are:

- The Sabre Order ID.
- The “cancelWithRetain” value set to “true”.

```
{
  "id": "1SXXXC50FL6CP",
  "cancelWithRetain": "true"
}
```

The response is a view of the order, the order items are deleted, the “ticketingDocumentInfo” shows the ticket(s) with coupons in an open state. The NDC flight segments are removed from the PNR.

```
{
  "order": {
    "id": "1SXXXC50FL6CP",
    "type": "ORDER",
    "pnrLocator": "PNWTPR",
    "pnrCreateDate": "2024-01-08",
    "offerVendor": "AA",
    "orderOwner": "1S",
    "partition": "AA",
    "primeHost": "1S",
    "countryCode": "AU",
    "contactInfos": [
      {
        "id": "4CEA2EE7-AF90-407C-B0B5-0F9F8B262434",
        "phones": [
          {
            "id": "FB7D914F-FF31-4D9E-9877-538DDBC3B2D4",
            "number": "656675765"
          }
        ]
      }
    ]
  }
}
```

```

],
"passengers": [
  {
    "id": "Passenger1",
    "externalId": "T1",
    "typeCode": "ADT",
    "contactInfoRefIds": [
      "4CEA2EE7-AF90-407C-B0B5-0F9F8B262434"
    ],
    "birthdate": "1987-09-20",
    "givenName": "ZUBA",
    "surname": "ZULU",
    "genderCode": "M"
  }
],
"customerNumber": {},
"ticketingDocumentInfo": [
  {
    "document": {
      "number": "0015751726280",
      "type": "702",
      "issueDateTime": "2024-01-09T01:55:00Z",
      "ticketingLocation": "BNE",
      "numberOfBooklets": 1,
      "reportingType": "BSP",
      "coupons": [
        {
          "number": 1,
          "status": "I",
          "segmentInfo": {
            "departureAirport": "DFW",
            "arrivalAirport": "TPA",
            "marketingAirlineCode": "AA",
            "departureTime": "2024-03-20T18:30:00",
            "arrivalTime": "2024-03-20T21:48:00",
            "flightNumber": 2401,
            "classOfService": "N"
          }
        }
      ]
    }
  },
  {
    "price": {
      "baseAmount": {
        "amount": "297.00",
        "code": "AUD"
      },
      "totalAmount": {
        "amount": "341.90",

```

```

        "code": "AUD"
      },
      "totalTaxAmount": {
        "amount": "44.90",
        "code": "AUD"
      }
    },
    "paxRefId": "Passenger1"
  }
],
"remarks": [
  {
    "id": "4e413d3d",
    "text": "ORDER CREATED",
    "type": "21"
  }
],
"paymentInformation": [
  {
    "id": "44bcb212-de72-4b01-b19e-a0083d9ce96b",
    "amount": {
      "amount": "341.90",
      "code": "AUD"
    },
    "paymentMethod": {
      "paymentTypeCode": "CASH"
    },
    "orderItemRefIds": []
  }
]
}

```

Notes

The “cancel and retain” capability is not available for all airlines.

For multi-passenger orders it is not possible to retain a subset of tickets, all tickets are retained for future use.

There is not currently an NDC industry method to refund retained tickets, dependent upon the airline and the settlement system the agent may be able to request a refund through offline applications.

Pricing an Offer with an Unused Ticket

After a ticket has been retained for future use, the [“offers/price”](#) API is used to request pricing based on the exchange value of the ticket. This message is used after the shopping request and prior to order creation, the following parameters are required:

- offerItemId – the Offer Item ID of the offer from the shopping response.
- passengers – an array with the following items:
 - id – an ID assigned to a passenger.
 - type – the passenger type code returned in the shopping response.
 - unusedTicketNumber – a ticket number with exchange value (13 numbers consisting of the airlines account code, the form code and serial number).

```
{
  "query": [
    {
      "offerItemId": [
        "ce0669d959bd2pcmnlmpgs26u0-173-1"
      ]
    }
  ],
  "passengers": [
    {
      "id": "Passenger1",
      "type": "ADT",
      "unusedTicketNumber": "0015750242925"
    }
  ]
}
```

The “offers/price” response is similar to the prime flow, with some additional information:

- Within the price section, some taxes have a taxPadisCode value of “PD”, meaning this tax was on the original ticket and is considered “paid”.

```
"taxes": {
  "total": {
    "amount": "47.90",
    "curCode": "AUD"
  },
  "breakdown": [
    {
      "amount": {
        "amount": "6.70",
        "curCode": "AUD"
      }
    }
  ]
}
```



```
    },
    "nation": "US",
    "taxCode": "XF",
    "taxPadisCode": "PD",
    "description": "US Passenger Facility Charge"
```

- An indicator at the end of the response verifies that the value of the unused ticket was used in the calculation of the new offer.

```
"wasTicketValueUsed": true
```

If the airline determines the unused ticket has no value or cannot be used as a part payment then the following warning is returned:

```
{
  "type": "WARNING",
  "message": "Ticket passed for exchange is either not exchange eligible or has no value. If you
decide to proceed a new ticket will be issued.",
  "service": "NDC_CONNECTOR"
}
```

Notes

As this process uses the “offers/price” API, the carrier is not able to validate the name on the unused ticket is the same as the passenger’s name in the new order. If the passenger names are different, the airline could issue an Agency Debit Memo (ADM).

Voiding Tickets

Voiding a ticket is the process of cancelling a sale before it is reported to a settlement system (ARC or IATA BSP). The most common use cases are when the passenger wishes to use a different form of payment than was first used to issue the ticket, or if there is some doubt as to whether the passenger will travel. Voiding may only occur before the ticket has been reported, this is typically midnight local time on the day of issue in countries that report to IATA BSP and midnight of the next business day for ARC.

Voiding a Ticket and Retaining the Itinerary

To void a ticket and retain the order the “orders/change” API is used, the input parameters are:

- The Sabre Order ID.
- The instruction to cancel the document and retain the order.

```
{  
  "id": "1SXXXAFCRTOTL",  
  "cancelDocumentAndRetainOrder": "true"  
}
```

After a ticket has been voided the coupon status is set to “V” (Void), the original time limit applies, and the order must be fulfilled before this expires to avoid cancellation by the airline.

Notes:

Not all airlines support the void ticket request.

For airlines that use their own stock, the voiding period may vary from the standards applied by the IATA BSP and ARC settlement systems.

Order Synchronization

The order synchronization API provides the ability to update the order in the Sabre Order Management system to reflect the order in the airline's system. Orders can become "out of sync" due to various reasons, including:

- Modification of an order made directly with an airline.
- Addition of ancillary services on the airline's web site.
- Changes to an order due to an "irregular operation".

If an order is changed by an airline, the airline is responsible for notifying the booking system of the change, however, not all changes generate the notification message, this varies by airline, and although airlines are enhancing their systems to support additional use cases, there are times when the orders may become "out of synch".

The "orders/sync" endpoint

The "orders/sync" API uses the "post" method, the current version of the "orders/sync" API is V1

Certification environment:

<https://api.cert.platform.sabre.com/v1/orders/sync>

Production environment:

<https://api.platform.sabre.com/v1/orders/sync>

Synchronizing a Sabre Order with an Airline

The order synchronization request requires the Sabre Order ID, the system requests a view of the order from the airline and updates the Sabre Order with specific changes which includes:

- Changes to the flight segments.
- Changes to tickets and EMDs.
- Changes to pre-reserved seats and ancillary services.
- Addition of SSR remarks.
- Changes to the payment (ticket) time limit.

```
{
  "id": "1SXXX3EK7DL9M"
}
```

The response to a synchronization request is a time stamp confirming the time the synchronization request was initiated.

```
{
  "responseDateTime": "2024-01-06T15:35:27.951750947Z"
}
```

Synchronization requests are only permitted once an hour for any single order, further synchronization requests of the same order within the same 60 minute period will receive the following error response:

```
{
  "errors": [
    {
      "code": "060252",
      "message": "Order [1SXXXCKLKVILZ] was just updated, please try again later"
    }
  ]
}
```

After the order has been updates the following items are updated in the PNR:

- NDC Flight Segments.
- Ticket Time Limit.

Notes

Some airlines return blanked dates for the payment time limit if it has expired, in this scenario the PNR is updated as follows:

TKT/TIME LIMIT

1.T-NDC SYNC

If the synchronization process results in an update to a PNR then a “received from field” is added to the PNR and may be seen in the PNR history:

R- SYNC/QF/OTHER/16/2024FEB07 2311Z

Glossary of Terms

Abbreviations and Acronyms

API – Application Programming Interface, an interface for software developers to code against to communicate with a computer system.

APIS – Advanced Passenger Information System, a series of data elements that airlines collect and pass to governments for security checks, the most common implementation is passenger passport information.

ARC – Airline Reporting Corporation, a U.S. company that provides settlement services between airlines and travel agencies in the U.S., U.S. Virgin Islands, Puerto Rico and Guam.

ATPCo – Air Tariff Publishing Company, an organization that distributes airline fares to Global Distribution Systems.

BFM – Bargain Finder Max, this is Sabre's shopping API and is used to request NDC offers.

BSP – Billing Settlement Plan, a global settlement system governed by IATA that provides settlement services between airlines and travel agencies.

EMD – Electronic Miscellaneous Document – an electronic document used by airlines to collect fees for ancillary services and miscellaneous fees.

Fare Basis Code (FBC) – An alpha numeric field used in pricing and ticketing that has values describing a fare (1-15 characters).

GIR – Grouped Itinerary Response, a data structure that allows for the grouping of common data elements and uses a reference to avoid repetition and improve efficiency.

IATA – International Air Transport Association, the governing body of the airline industry responsible for defining industry standards.

JSON – Javascript Object Notation, an open standard file format for data interchange.

KTN - Known Traveler Number – a number assigned to a passenger by a security division of a government includes: TSA Precheck, Global Entry, NEXUS and SENTRI, a KTN allows passengers to pass through airport security checks with reduced screening.

NDC – New Distribution Capability, a series of XML messages defined by IATA to enable airline retailing for aggregators and agencies.

Offer ID – An identification number assigned to an offer, (displayed as “offerId”).

Offer Item ID - An identification number assigned to an item within an offer, (displayed as “offerItemId”).

Order ID - An identification number assigned to an order.

Order Item ID - An identification number assigned to an item within an order.

PADIS – Passenger and Airport Data Interchange Standards, the messaging standards defined by IATA.

PNR - Passenger Name Record a completed reservation with flight and passenger information.

PTC – Passenger Type Code a 3-character alpha numeric value that is used in shopping and pricing to differentiate passengers that may be eligible for different fares, e.g. ADT- Adult, GOV – Government, C06 – Child aged six years.

PSS – Passenger Sales and Service System, the system used by airlines for all reservations, ticketing and passenger service functions.

Redress Number – A Redress Number is used to facilitate security screenings for passengers whose names may be on watchlists although they are a different individual to the person that is restricted from travel.

REST - A software architecture that imposes conditions on how an API should work, REST was initially created as a guideline to manage communication on a complex network like the internet.

SFPD – Secure Flight Passenger Data, information that is required by some government security services to verify passenger identification, typically the name of the passenger on their travel documents (which may vary to the name in their reservation), the passenger’s gender and date of birth.

SNAP – Sabre Negotiated Air Pricing, a system that enables mapping of airline Account Codes to Corporate ID’s

SOAP – Simple Object Access Protocol, an official protocol maintained by the World Wide Web Consortium (W3C).

Ticket - An electronic document used by airlines to collect payment for air fares and track usage.

XML - Extensible Markup Language Lets you define and store data in a shareable manner XML supports information exchange between computer systems such as websites, databases, and third-party applications.