JRTechnologies

Business Requirements Document (Level 2)

Airline Sales File Converter – VSTS 83112

01/02/2021
Version 0.6

Revision Control

Name	Date	Changes summary	Version
Kalliopi Kalantzaki	12/11/2020	Initial version	0.1
Kalliopi Kalantzaki	14/12/2020	Remove parts for captruring using non-ifg sales functionality	0.2
Kolvalsky Dmitry	15/12/2020	Review 0.2	0.2
Kalliopi Kalantzaki	16/12/2020	Reply to comments	0.3
Kolvalsky Dmitry	09/01/2021	Review 0.3	0.3
Kalliopi Kalantzaki	14/01/2021	Reply to comments	0.4
Kalliopi Kalantzaki	29/01/2021	Addition of SLA details	0.5
Kalliopi Kalantzaki	01/02/2021	Addition of CCGR fop	0.6

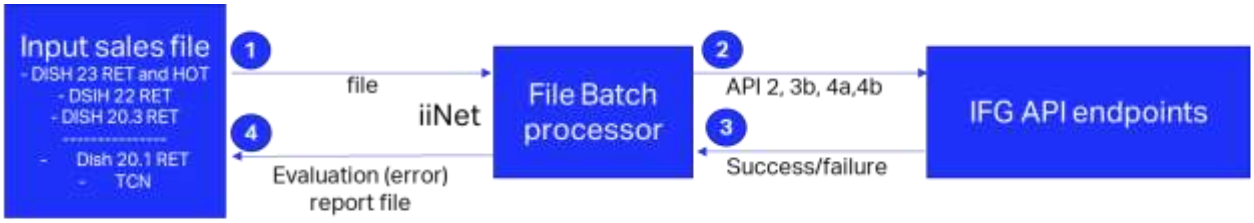
Contents

Introduction

Overview

The document describes details about new functionality that is goint to be implemented in IFG regarding the process of loading airline sales files into IFG processing. Currently IFG Customers can report payment transactions to IFG using IFG API. In some specific cases Airline cannot implement the API but is able to provide the sales data in post-sales reporting files (RET / HOT / TCN) which would be possible to be loaded to IFG for processing. This document will describe the converter which will process the input files and call regular IFG API input to sales to IFG using existing “non-IFG sales” flow.

The general flow of the new score is the following:



The new batch file processor added in IFG will allow the conversion of sales from airlines received in the form of a file into the IFG APIs. It will operate at the entry of IFG before the API endpoints, and transactions sent through it will be processed like all other IFG transactions.

Reporting to IFG will be done as “non-IFG Sales”, so no API1 will be needed. 1, wheras the input file accepted at the entry of this bacth file processor will be DISH RET 20.1, 20.3, 22.0 and 23.0, DISH HOT 23, TCN.

In this analysis the IFG API mapping from DISH RET 20.3 to DISH RET 22.0 and DISH RET 23.0 will be addressed.

Important note: This CR lies to P2 SLA rules. More specifically, all input files sent from IATA through iiNET must be successfully been parsed and reported to IFG within 8 hours from the time JRT has received files. Related steps that lie within SLA restrictions are described in SLA

Intended Audience and Reading Suggestions

This document is intended for:

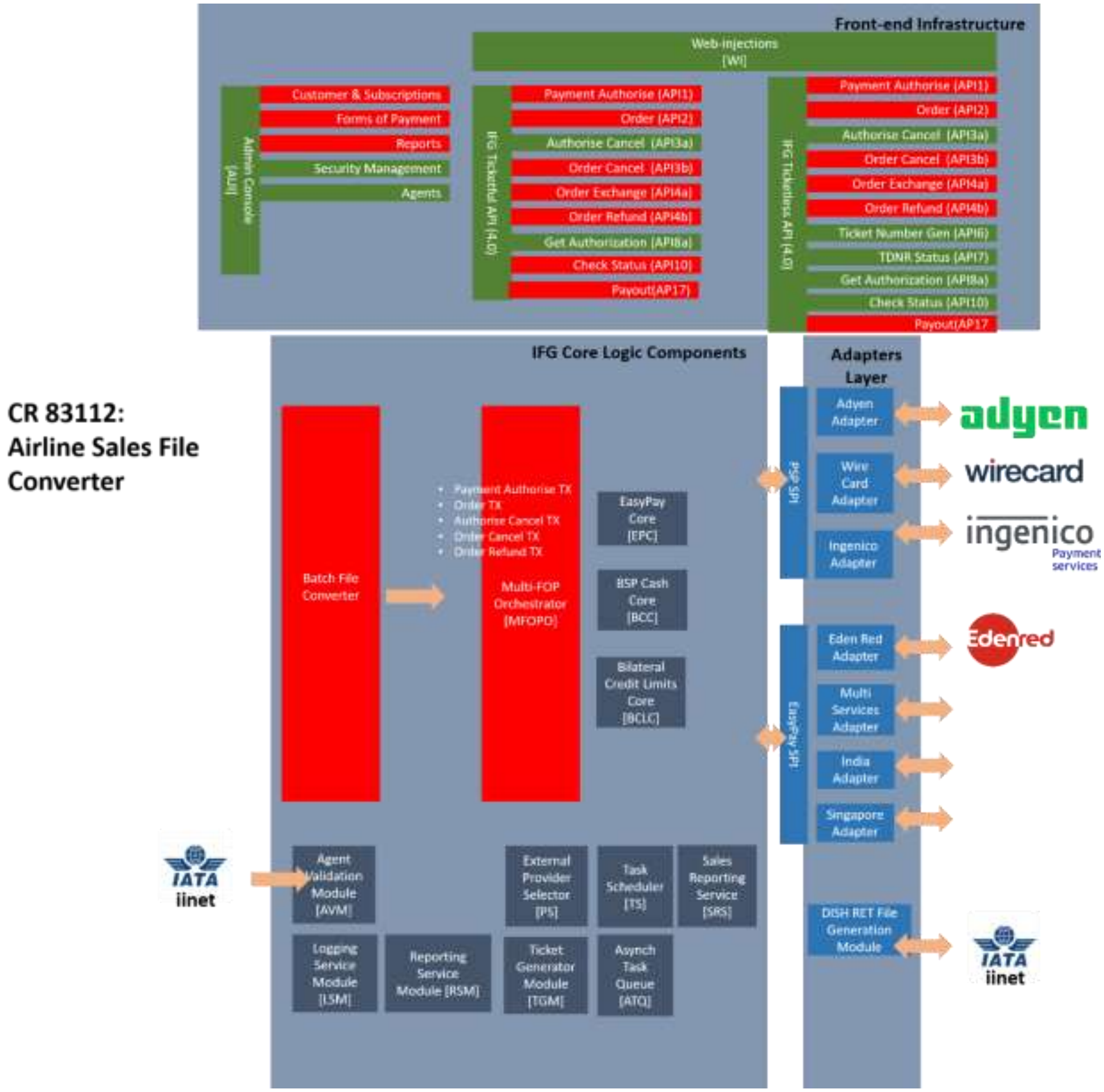
1. *Developer* – this document provides detailed information to developers tasked with implementing this initiative
2. *Technical Architect* – this document defines overall non-functional constraints that need to be addressed in the core design of the proposed system
3. *Business Analysts* – this document describes the functional requirements of each process within the scope of the system
4. *Business stakeholders of the project* – this document describes details about features of the required functionality

References

1. *VSTS item: 83112*
2. *IATA BRD Level 1: IFG processing for airline sales files reports.docx*

IFG Functional Architecture

The functional components within the scope of the deliverable are as illustrated below. The components marked in red are the ones impacted by the scope of the proposed system:



CR 83112: Airline Sales File Converter

Figure 1 - Impact of Airline Sales Batch IFG Converter to Functional Architecture

Business Entities

The following illustrates which IFG business entities are impacted by the proposed changes:

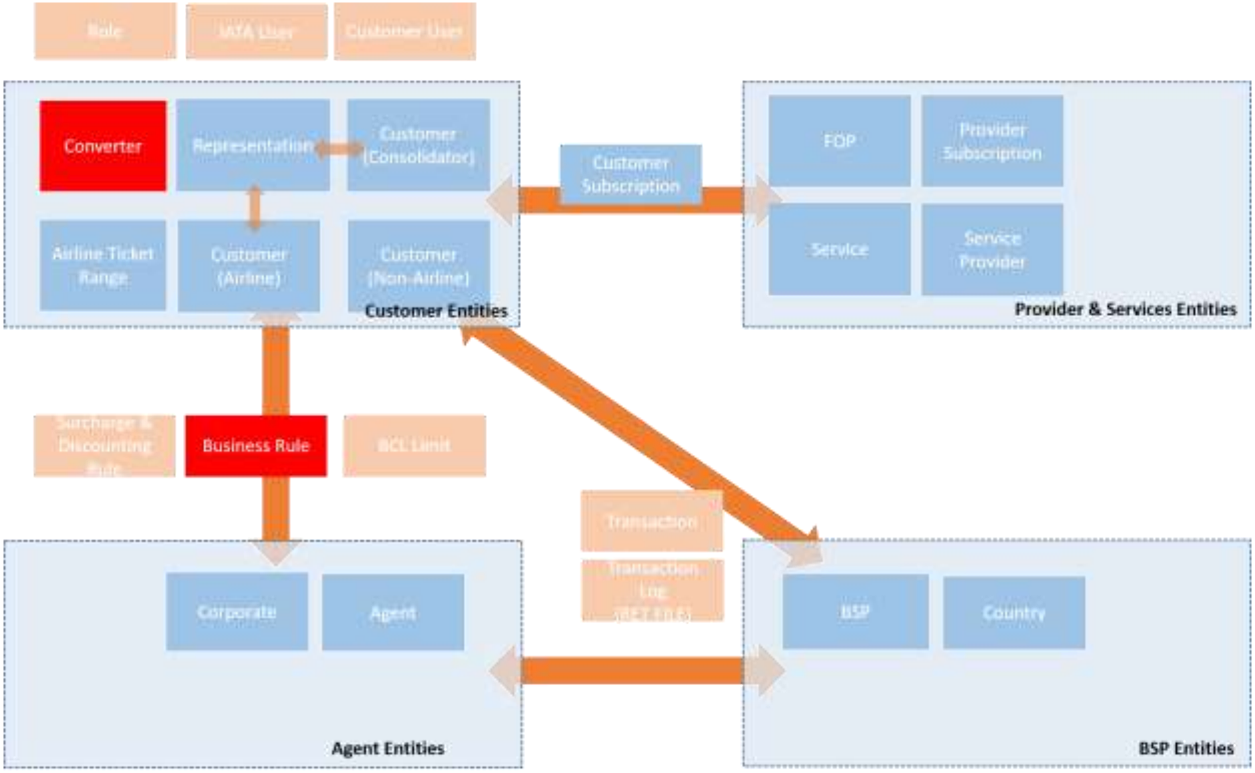


Figure 2 - Impact of CR 83112 on key IFG entities

CR 83112-Airline Sales Converter Processes

Processes List

The list of user and system processes covered in this document are as follows:

ID	Process Name	Process Description
1.	*AUI.ADD_CUSTOMER*	Add customer
2.	*AUI.VIEW_SALES_CONVERSION_RULE*	IATA user views a sales conversion rule
3.	*AUI.EDIT_SALES_CONVERSION_RULE*	IATA user edits a sales conversion rule
4.	*AUI.ADD_SALES_CONVERSION_RULE*	IATA user adds a sales conversion rule
5.	*AUI.REVIEW_SALES_CONVERSION_RULE*	IATA user reviews a sales conversion rule
6.	*AUI.ADD_BUSINESS_RULE*	IATA user adds a business rule
7.	*AUI.EDIT_BUSINESS_RULE*	IATA user edits a business rule
8.	*AUI.REVIEW_BUSINESS_RULE*	IATA user reviews a business rule
9.	*AUI.ADD_ADDITIONAL_SERVICE_BUSINESS_RULE*	Add business rule of an additional service
10.	*AUI. VIEW_ADMINISTRATIVE_FILES*	View administrative reports for files
11.	*AUI. VIEW_PAYMENT_TXs_VIA API*	List Payment TX by API
12.	*AUI.VIEW_PAYMENT_TXs*	List all transactions for tickets
13.	*AUI.VIEW_PAYMENT_TX_DAY_SUMMARY*	View daily transactions reports
14.	*API16.REQUEST_3DS*	IBE or Airline System Requests a 3DS verification for a payment
15.	*API1.AUTHORISE_PAYMENT*	Payment authorization via API
16.	*API2.CONFIRM_ORDER*	Order via API
17.	*API3B.CANCEL_ORDER*	Order cancellation via API
18.	*API4A.EXCHANGE_ORDER*	Order exchange via API
19.	*API4B.REFUND_ORDER*	Order refund via API
20.	*API15.GET_PAYEMENT_DETAILS*	Get payment details per form of payment
21.	*API17.PAYOUT*	Payout API
22.	*DRGM.PARSE_INPUT_FILE*	Parse ailine sales input file (RET, HOT,TC)
23.	*S.SALES_EVALUATION_REPORT*	Summary generation reports of related files after being parsed within IFG
24.	*S. GENERATE_MACHINE_READABLE_SUMMARY_REPORT*	Machine readable summary report of related files after being parsed within IFG
25.	*DRGM.GENERATE_CUSTOMER_API_CREDENTIALS*	Generate API credentials for a new customer user

AUI.ADD_CUSTOMER

<table>
<tr><td><b>Internal Reference</b></td><td>AUI.ADD_CUSTOMER</td></tr>
<tr><td><b>Description</b></td><td>This is the process through which an IATA User can create a Customer</td></tr>
<tr><td><b>Actors</b></td><td>An IATA User administers information across all IFG Customers</td></tr>
<tr><td><b>Required Inputs</b></td><td>No external inputs – user input collected within process</td></tr>
<tr><td><b>Expected Initial State</b></td><td>No initial state expected</td></tr>
<tr><td><b>Pre-start conditions</b></td><td>User has to have these permissions:
• ADD_CUSTOMER
• VIEW_CUSTOMER</td></tr>
<tr><td><b>Steps</b></td><td>Current Functionality:
1. The user navigates to “Customers & Subscriptions” -> “Customers”
2. The user clicks the “Add Customer” button
3. A popup opens for the user to enter the required fields
4. The user clicks the “Save” button to save Subscription

New Functionality:
3. For airline customers, there will be also a new field as multi- selection list where user can set all acceptable file types for sales file conversion. These are: { RET DISH 20.1, RET DISH 20.3, DISH HOT 23, TCN}</td></tr>
<tr><td><b>Alternative flows</b></td><td>Current Functionality:
1. The user can add a logo to the Customer by clicking the “Upload Logo” button.
• A new popup “Upload Logo” opens
• The user clicks on “Choose a file” button to select a file
• The user clicks “Upload” button to save the logo
• The user is presented with “File uploaded successfully” message
• The user clicks “Close” button
2. The user clicks the “Cancel” button
• The popup closes and the user returns at the previous page</td></tr>
<tr><td><b>Error Conditions</b></td><td><table>
<tr><th>Errors</th><th>Description</th></tr>
<tr><td>Missing required fields</td><td>“This input is required”</td></tr>
<tr><td>Invalid agent data</td><td>Agent’s schema file is missing.</td></tr>
</table></td></tr>
<tr><td><b>Expected Final State in case of successful completion</b></td><td>Customer.Status=Pending and all values of the customer will remain unchanged (until such changes are approved via a separate process)</td></tr>
<tr><td><b>Log Entries</b></td><td>Admin Console : Log entry to record this activity at the Admin Console level as this might be a sensitive operation</td></tr>
</table>

AUI.VIEW_SALES_CONVERSION_RULE

Internal Reference	AUI.VIEW_SALES_CONVERSION_RULE
Description	This is the process through which an IATA User or a Customer User can view list of airline's rules regarding sales conversion matching
Actors	An IATA User administers information across all IFG Customers. A Customer User administers only information specific to that Customer.
Required Inputs	No external inputs – user input collected within process
Expected Initial State	There can be zero, one or more Sales Conversion Rules
Pre-start conditions	User has to have these permissions: • VIEW_SALES_CONVERSION_RULE
Steps	1. The user navigates to "Customers & Subscriptions"->"Sales File Conversion Rules" 2. The user is presented with a list of Customers 3. The user expands a Customer 4. The user is presented with a list of all rules for sales conversion • File Version • Input Field Name • Input Field Value • Target Field Name • Target Filed Value • Status: {ACTIVE, INACTIVE} • .
Alternative flows	None
Error Conditions	None
Expected Final State in case of successful completion	No state changes
Functional Module Logging	None

AUI.ADD_SALES_CONVERSION_RULE

<table>
<tr><td><b>Internal Reference</b></td><td>AUI. ADD_SALES_CONVERSION_RULE</td></tr>
<tr><td><b>Description</b></td><td>This is the process through which an IATA User or a Customer User can create a new customer rule regarding sales conversion matching</td></tr>
<tr><td><b>Actors</b></td><td>An IATA User administers information across all IFG Customers. A Customer User administers only information specific to that Customer.</td></tr>
<tr><td><b>Inputs</b></td><td>No external inputs – user input collected within process</td></tr>
<tr><td><b>Expected Initial State</b></td><td>No initial state expected</td></tr>
<tr><td><b>Pre-start conditions</b></td><td>User has to have these permissions:
• ADD_ SALES_CONVERSION_RULE
• VIEW_SALES_CONVERSION_RULE</td></tr>
<tr><td><b>Steps</b></td><td>1. The user navigates to “Customers & Subscriptions”->“Sales Conversion Rules”
2. The user is presented with a list of Customers
3. The user expands a Customer
4. The user is presented with a list of current added customer sales conversion rules
5. The user clicks “Add Rule”
6. A popup opens displaying the “Add Rule” form with the following mandatory fields:
a. File Version : List of {RET DISH 20.1, RET DISH 20.3, DISH HOT 23, TCN}. Required
b. Source Field Name: Text Input field. Required – It represents the RET/DISH/TNC field that needs to be matched to a different value.
c. Field Value: Text Input field. If not provided means any value .
The value of RET/DISH/TNC field that needs to be changed in initial input file.
d. Target Field Value: Required Input text – The value to which the field value will be matched on related IFG API
e. Status: {ACTIVE, INACTIVE} – Default ACTIVE.
f. Rule Description: Free text
~~7. It can be more complex mapping rules which cannot be configured this way. Such rules will be coded as part of the Airline transition to this convertor. Fir such rules – we add a rule type which in step d. displays the read-only description of the conversion coded (not here, in the Edit screen).~~
8. The user fills the required inputs and user clicks the “Save” button.</td></tr>
<tr><td><b>Alternative flows</b></td><td>The user clicks the “Cancel” button
• The popup closes and the user returns at the previous page</td></tr>
<tr><td><b>Error Conditions</b></td><td><table><tr><th>Errors</th><th>Description</th></tr><tr><td>Missing required fields</td><td>“This input is required”</td></tr></table></td></tr>
</table>

Expected Final State in case of successful completion	Sales.Converter.Rule.Status=Pending
Functional Module Logging	Admin Console : Log entry to record this activity at the Admin Console level as this might be a sensitive operation

AUI.EDIT_SALES_CONVERSION_RULE

Internal Reference	AUI.EDIT_SALES_CONVERSION_RULE
Description	This is the process through which an IATA User or a Customer User modifies an existing customer rule regarding sales conversion matching
Actors	An IATA User administers information across all IFG Customers. A Customer User administers only information specific to that Customer.
Inputs	No external inputs – user input collected within process
Expected Initial State	Subscription being edited has to have *Sales.Conversion.Rule.Status = Approved*
Pre-start conditions	User has to have these permissions: - EDIT_SALES_CONVERSION_RULE - VIEW_SALES_CONVERSION_RULE
Steps	1. The user navigates to the “Customers & Subscriptions” -> “Sales Conversion Rules” The user is presented with a list of Customers 2. The user expands a Customer 3. The user is presented with a list of Sales Conversion Rules 4. The user clicks the blue pencil icon next to a sale conversion rule. 5. A popup opens displaying the “Edit Sale Conversion Rule” form - The user modifies the entry with the same fields and constraints as those specified in AUI: ADD_SALES_CONVERSION_RULE 6. If any mandatory fields have not been filled, the “Save” button remains disabled 7. The user clicks the “Save” button
Alternative flows	The user clicks the “Cancel” button. - The popup closes and the user returns at the previous page
Error Conditions	Same as AUI.ADD_ SALES_CONVERSION_RULE
Expected Final State in case of successful completion	Sales Conversion.Rule.Status=Pending. All values of the entry will remain unchanged (until such changes are approved via a separate process)
Functional Module Logging	Admin Console : Log entry to record this activity at the Admin Console level as this might be a sensitive operation

AUI.REVIEW_SALES_CONVERSION_RULE

Internal Reference	AUI. REVIEW _SALES_CONVERSION_RULE
Description	This is the process through which an IATA User or a Customer User reviews an existing customer rule regarding sales conversion matching
Actors	An IATA User administers information across all IFG Customers. A Customer User administers only information specific to that Customer.
Inputs	No external inputs – user input collected within process
Expected Initial State	Subscription being edited has to have *Sales.Conversion.Rule.Status = Approved*
Pre-start conditions	User has to have these permissions: • VIEW_SALES_CONVERSION_RULE • REVIEW _SALES_CONVERSION_RULE
Steps	1. The user navigates to the “Customers & Subscriptions” -> “Sales Conversion Rules” The user is presented with a list of Customers 2. The user expands a Customer 3. The user is presented with a list of Sales Conversion Rules 4. The user clicks on the red exclamation mark next to the entry with Subscription.Status=Pending 5. A popup opens displaying the entry values. The user clicks the “APPROVE” button.
Alternative flows	5a. The user clicks the “Reject” button ○ The popup closes and the user returns at the previous page. If the entry pending approval was new, is removed from the list. Otherwise, it returns in the previous state. 5a. The user clicks the “Cancel” button • The popup closes and the user returns at the previous page
Error Conditions	Same as AUI.ADD_SALES_CONVERSION_RULE
Expected Final State in case of successful completion	Sales Conversion.Rule.Status=Pending. All values of the entry will remain unchanged (until such changes are approved via a separate process)
Functional Module Logging	Admin Console : Log entry to record this activity at the Admin Console level as this might be a sensitive operation

AUI.ADD_BUSINESS_RULE

Internal Reference	AUI.ADD_BUSINESS_RULE
Description	This is the process through which an IATA User or a Customer User can add a Customer Business Rule
Actors	An IATA User administers information across all IFG Customers. A Customer User administers only information specific to that Customer.
Inputs	No external inputs – user input collected within process
Expected Initial State	No initial state required
Pre-start conditions	User has to have these permissions: - CREATE_RULE - VIEW_RULES
Steps	Current Functionality: 1. The user navigates to the “Forms of Payment” 2. The list of supported Forms of Payment, is shown in the list 3. User will click on “Business Rules” 4. The IATA User selects a Customer. The Customer User can see only his own Business Rules. 5. The user clicks on the “Add Rule” button 6. A new page opens displaying the Add Business rule form. Different forms can be rendered according to the Form of Payment 7. User clicks the “Save” button New Functionality: 6. For all forms of payments, add as option the NDC value in the Channel Type selection list.
Alternative flows	7. The user clicks the “Cancel” button - The user returns at the previous page.
Error Conditions	None
Expected Final State in case of successful completion	Rule.Status=Pending
Functional Module Logging	Current Functionality: - Admin Console : Log entry to record this activity at the Admin Console level as this might be a sensitive operation New Functionality: - None

AUI.EDIT_BUSINESS_RULE

Internal Reference	AUI.EDIT_BUSINESS_RULE
Description	This is the process through which an IATA User or a Customer User can edit a Customer Business Rule
Actors	An IATA User administers information across all IFG Customers. A Customer User administers only information specific to that Customer.
Required Inputs	No external inputs – user input collected within process
Expected Initial State	Customer Business Rule.Status=Approved
Pre-start conditions	User has to have these permissions: • EDIT_RULE • VIEW_RULES
Steps	Current Functionality 1. The user navigates to the “Forms of Payment” -> *Form of Payment* -> “Business Rules” 2. The IATA User selects the Customer. The Customer User can see only his own Business Rules 3. The user clicks on the ‘pencil’ button next to a Business Rule 4. A popup opens displaying the rule form 5. The user clicks the “Edit” button 6. A new page opens for the user to enter the required fields 7. The user clicks the “Save” button to save Business Rule New Functionality: 6. User will be able to edit all fields as described in AUI.ADD_BUSINESS_RULE process.
Alternative flows	The user clicks the “Cancel” button • The user returns at the previous page
Error Conditions	Same as AUI.ADD_BUSINESS_RULE
Expected Final State in case of successful completion	Business Rule.Status=Pending and all values of the Customer Business Rule will remain unchanged (until such changes are approved via a separate process)
Functional Module Logging	Current Functionality: • Admin Console : Log entry to record this activity at the Admin Console level as this might be a sensitive operation New Functionality: • Create a new audit entry to be used in the “Reports” -> “Administrative Reports” -> “Administrative Changes” report

AUI.REVIEW_BUSINESS_RULE

Internal Reference	AUI.REVIEW_BUSINESS_RULE
Description	This is the process through which an IATA User or a Customer User can review (approves/rejects) a new or an updated AFOP Business Rule
Actors	An IATA User administers information across all IFG Customers. A Customer User administers only information specific to that Customer.
Required Inputs	No external inputs – user input collected within process
Expected Initial State	*Business Rule.Status = Pending*
Pre-start conditions	User has to have these permissions: • REVIEW_RULE • VIEW_RULES
Steps	Current Functionality: 1. The user navigates to the "Forms of Payment". The list of supported Forms OF Payment, is shown in the list 2. User will click on "Business Rules" 3. The IATA User selects a Customer. The Customer User can see only his own Business Rules. 4. The user clicks the red "!" button next to a Business Rule pending for approval 5. A new page opens displaying the Business Rule form 6. The user reviews the fields 7. The user clicks the "Approve Rule" button to approve the Business Rule New Functionality: 1. Review functionality will be the same and will include any change for new fields as described in AUI.ADD_BUSINESS_RULE process. 2. The rest of the functionality remains unchanged.
Alternative flows	1. The user clicks the "Cancel" button • The popup closes and the user returns at the previous page 2. The user clicks the "Reject Rule" button • The changes are rejected and the rule is deleted. If the subscription pending approval was new, is removed from the list. Otherwise, it returns in the previous state. 3. Only in case the Rule is edited, the user has a third button to click the "View Active Rule" • The user by pressing this button is able to see the previous active status of the rule before the changes • The user clicks "Back to Pending" to return in the previous page • The user clicks "Cancel" to exit the process of approval
Error Conditions	None
Expected Final State in case of successful completion	Business Rule.Status=Active

Functional Module Logging	Current Functionality: • Admin Console : Log entry to record this activity at the Admin Console level as this might be a sensitive operation New Functionality: • None

AUI.ADD_ADDITIONAL_SERVICE_BUSINESS_RULE

Internal Reference	AUI.ADD_ADDITIONAL_SERVICE_BUSINESS_RULE
Description	This is the process through which an IATA user or an IATA customer can add a Business Rule of an additional service.
Actors	An IATA user administers information across all IFG Customers henceforth called “the IATA user” Or Customer user for the own customer organization
Required Inputs	No external inputs – user input collected within process.
Expected Initial State	• A Credit Card Subscription exists for the user’s airline for case of 3DS.
Pre-start conditions	User has to have these permissions: • CREATE_ADDITIONAL_SERVICES_RULE • VIEW_ ADDITIONAL_SERVICES_RULE • VIEW_ ADDITIONAL_SERVICES
Steps	Current Functionality: 1. The user navigates to the Additional services -> 3D Secure -> Business Rules from the navigation bar 2. The user expands the airline for which he needs to update or create a new rule and presses ADD RULE button. 3. The user fills all related 3DS rule fields. 4. The user clicks the “Submit for Approval“ (in case of existing rule) or the “Save” button in case of a new rule. • The system checks for the user selected countries if there is an active subscription to FOPs. Otherwise show related error. New Functionality: 3. Both for TASF and 3DSecure, add as option the NDC value in the Channel Type selection list. 4. User clicks the “Submit for Approval” button.
Alternative flows	5a. The user clicks the “Cancel” button • The popup closes and nothing happens
Error Conditions	New Functionality: None
Expected Final State in case of successful completion	• 3DS is configured in the rule. • Unless the rule is approved no change is made. • Rule. Status=Pending
Log Entries	Old requirements • Admin Console : Audit log entry to record this activity at the Admin Console level as this might be a sensitive operation .

	New Requirements: • None

AUI. VIEW_ADMINISTRATIVE_FILES

Internal Reference	AUI. VIEW_ADMINISTRATIVE_FILES
Description	This is the process through which an IATA user can view files that were created or received from administrative reports section.
Actors	IATA User or a Customer User henceforth called "the user"
Required Inputs	No external inputs – user input collected within process
Expected Initial State	No initial state expected
Pre-start conditions	User has to have these permissions: • VIEW_ ADMINISTRATIVE_CHANGES_REPORTS
Steps	Current Functionality: • The user navigates to the "Reports" -> "Administrative Reports" -> "File Interface" • The user fills the mandatory filters • The user clicks the "Search" button • The search results are presented at the bottom of the page New Functionality: • New field for sales conversion files will be added as *Sales Conversion Files*. • The user clicks the "Search" button • The search results are presented at the bottom of the page ○ If user selects *Received*, then each line in the results will be showing all filenames received and processed/rejected from file input processor converter. The File Type column will be showing the file version (DISH, HOT, TCN), if available in related input file. ○ If user selects *Sent,* each line in the results will be showing all filenames that were created after processing the input file. A new column will be also added in order user to be able to download the generated report.
Alternative flows	1. The user clicks "Export" button under the search box a. A popup window is presented asking the user "Are you sure you want to export data to excel file? b. The user clicks "Ok" button and an excel file including the search results is downloaded.
Error Conditions	None
Expected Final State in case of successful completion	No state changes
Functional Module Logging	Current Functionality: • Admin Console : Audit log entry to record this activity at the Admin Console level as this might be a sensitive operation (related to export of data) New Functionality • None

AUI. VIEW_PAYMENT_TXs_VIA API

Internal Reference	AUI.VIEW_PAYMENT_TXs_VIA_API
Description	This is the process through which an IFG user can view payment transactions made via API This corresponds to API Request Report
Actors	IATA User or a Customer User henceforth called "the user"
Required Inputs	No external inputs – user input collected within process
Expected Initial State	No initial state expected
Pre-start conditions	User has to have these permissions: • VIEW_API_REQUESTS_REPORTS
Steps	Current Functionality: 1. The user navigates to the "Reports" -> "Transaction Reports" -> "API Request Report" 2. The user fills the mandatory filters 3. The user clicks the "Search" button 4. The search results are presented at the bottom of the page New Functionality: • In "*Channel*" field NDC value will also be available. Related results will isolate all requests with NDC channel type
Alternative flows	The user clicks "Export" button under the search box a. A popup window is presented asking the user "Are you sure you want to export data to excel file? b. The user clicks "Ok" button and an excel file including the search results is downloaded
Error Conditions	None
Expected Final State in case of successful completion	No state changes
Functional Module Logging	Current Functionality: • Admin Console : Audit log entry to record this activity at the Admin Console level as this might be a sensitive operation (related to export of data) New Functionality: • None

AUI.VIEW_PAYMENT_TXs

Internal Reference	AUI.VIEW_PAYMENT_TXs
Description	This is the process through which an IFG user can view payment transactions aggregated by day. This corresponds to Daily Summary Reports.
Actors	IATA User or a Customer User henceforth called “the user”
Required Inputs	No external inputs – user input collected within process
Expected Initial State	No initial state expected
Pre-start conditions	User has to have these permissions: • VIEW_DAILY_REPORTS
Steps	Current Functionality: 1. The user navigates to the “Reports” -> “Transaction Reports” -> “Transaction Report” 2. The user fills the mandatory filters 3. The user clicks the “Search” button 4. The search results are presented at the bottom of the page New Functionality: a. In “*Channel*” field NDC value will also be available. Related results will isolate all requests with NDC channel type
Alternative flows	1. The user clicks “Export” button under the search box a. A popup window is presented asking the user “Are you sure you want to export data to excel file? b. The user clicks “Ok” button and an excel file including the search results is downloaded
Error Conditions	None
Expected Final State in case of successful completion	None
Functional Module Logging	Current Functionality: • Admin Console : Audit log entry to record this activity at the Admin Console level as this might be a sensitive operation (related to export of data) New Functionality: • None

AUI. VIEW_PAYMENT_TX_DAY_SUMMARY

Internal Reference	AUI.VIEW_PAYMENT_TX_DAY_SUMMARY
Description	This is the process through which an IFG user can view payment transactions aggregated by day. This corresponds to Daily Summary Reports.
Actors	IATA User or a Customer User henceforth called “the user”
Required Inputs	No external inputs – user input collected within process
Expected Initial State	No initial state expected
Pre-start conditions	User has to have these permissions: • VIEW_DAILY_REPORTS
Steps	Current Functionality: 1. The user navigates to the “Reports” -> “Transaction Reports” -> “Daily Summary Reports” 2. The user fills the mandatory filters 3. The user clicks the “Search” button 4. The search results are presented at the bottom of the page New Functionality: a. In “*Channel*” field NDC value will also be available. Related results will isolate all requests with NDC channel type
Alternative flows	1. The user clicks “Export” button under the search box a. A popup window is presented asking the user “Are you sure you want to export data to excel file? b. The user clicks “Ok” button and an excel file including the search results is downloaded
Error Conditions	None
Expected Final State in case of successful completion	None
Functional Module Logging	Current Functionality: • Admin Console : Audit log entry to record this activity at the Admin Console level as this might be a sensitive operation (related to export of data) New Functionality: • None

DRGM.PARSE_INPUT_FILE

Internal Reference	DRGM.PARSE_INPUT_FILE
Description	This process deals with parsing of RET and/or HOT input files. The process analyzed the content of the files and sends batch requests to IFG APIs as non-ifg sales. Each file will refer to one airline client and may contain data from one or many different countries.
Actors	IATA system interacting with IFG parsing system. The latter interacts with IFG APIs.
Required Inputs	DISH RET 20.1,20.3, 22,23, TCN and HOT v.X Note: • Input files size may vary. Average size around 1-3 mb, but we saw instance of 21mb per file. ~~Upper bound of transactions per file 86.000 transactions~~ The upper bound of expected transactions per file is the current number of GOL live transactions per day x10. • Expected input file frequency expected: Per day, 1 global file for all BSPs per airline or one file per country per BSP per airline. Typically, per airine the system may receive per day close to 60 files that refer to single BSP files. • The number of airlines expected initially to be integrated are close to 10 airlines. • Current analysis focuses on RET 20.3, 22 and 23 input files
Expected Initial State	Input files will be sent to JRT thought IATA iiNET from different account compared to the one currently used for HOT/RET files.
Steps	See Process Steps
Alternative flows	None
Error Conditions	See Response Codes
Expected Final State in case of successful completion	An airline sales input file has been parsed successfully and related sales transactions have been made to IFG APIs in order to be further reported in RET.
Functional Module Logging	Log entry to record this activity at the process level as this is a sensitive operation

Process Steps

Below are the high-level steps for RET, HOT, TCN file parsing.

	Steps	Description
1.	File name checks	Check file input name to be as XXddZZZLINK_YYYYMMDD_AAAC_sequence_ISOC, where - XX: ISO Country Code - dd: file descriptor {ew: single country DISH RET file , eg: multi country DISH RET file} - ZZZ: can be NDC of WEB (legacy NDCLink and Weblink files). - YYYYMMDD: File generation date - AAAC: Airline 3D code and its' check digit (Mod7). - Sequence: 000 if it is the first file uploaded for this Airline in that date. An incremental number for next files sent for the same Airline at the same date. - ISOC: Optional, if provided all transactions in the file should be related to the BSP country of ISOC and rejected in other case. If file name is not correct, exit with PRS002. If file type is not supported by customer settings, then reject with PRS005 For the case when file is rejected - An email notification must be sent to support-ifg@jrtechnologies.com, with following fields: ○ Subject: IFG file parsing rejection ○ Content: File <%s File Name> was rejected with error code PRS002:<description> at <%s Server DateTime>. - Log file name rejection and datetime of rejection. Also the list of rejected files should be available via IFG Reports / Admin reports / File Interface. Same place should be used to show success / failures for processed files and responses.
2.	Integrity validation	Based on the input file type (DISH 22, 23 or 20.3) check the file integrity. The expected IT elements we focus on are included in related Appendix with DISH and API matching. For this purpose, check if all mandatory IT elements exist in the file and if the structure of these is correct. - For header and trailer IT records (IT01 and IT0Z) also make related format and required fields validation. If there is any error, then consider there is integrity violation error. - If header elements indicate a file type different than the one of RET 20.3, 22, 23, TCN or HOT, then also the file will be rejected (this will gradually change once adding next versions of input files). If there is any file integrity error validation, then reject the file altogether and exit with PRS003. In this case – still to generate the validation report, stating 0 records processed in the 1st section / file structure issue.
3.	Transaction validation	For all transactions included in file and in IT records apart from header and trailer, check if there is any integrity error per transaction. If there was any error as described below, then do not reject the file, only related transaction and continue with next transaction in sequence. Related error will be included in Error Report. To this end: - Check if all mandatory fields per IT record are present and if maximum length per line is correct. Otherwise, reject transaction. - Check that the available transaction codes (IT02 TRNC) are one of these: {TKTT, RFND, CANX, EMDA, EMDS}. Otherwise, reject transaction. - In case there are transactions with conjunction tickets and there is any error in the set of the conjunction, then reject the conjunction set altogether.

		Additionally, if ISOC is provided in file name, then also check that each transaction is on the same country code as the one of the ISOC. Otherwise, consider this as file violation error and reject **file altogether with** PRS004. Log file name rejection and datetime of rejection.
4.	Custom fields mapping	Each file is per airline. Based on related ifgCode, check if there is any active rule for custom field mapping. For all active custom fields, update all transactions as read from file with related values as set in the rule. • Upon making related updates based on custom fields, check that formOfPayment (FPTP) is any of following and convert as. ○ CCAANNNN to CCAA, where AA defines the card type. For the case when CCAA does not exist in IFG fop.type supported list, then additionally this fop.type will have to be added in the list of supported fop.types only on endpoints level. These fops will be reported as CCAANNNN, where the NNNN will be the first 4 characters of accountNumber as given in related FPAC element ○ CCGRXXXX or CCGR. Simillarly to CCAANNNN, this fop.type will be reported as CCGR along with NNNN that may exist on related FPAC element. ○ CCEPXX, CCEPXX, CCEPEX to EP • Otherwise if FPTP does not contain any of the following, reject transaction Also , SASI field should be added to related API calls, if provided by the customer to be able to include it in the output RET from IFG to BSP Note: Many rules conversion will be applied in this step per customer needs – there won't be any configuration for these additional set of rules.
5.	Transactions Sequence	Transactions present in the files will have to be sent to APIs with following sequence: 1. All targeted for API2 (including conjunction sets) (Sales) 2. All API4A (including conjunction sets) (Exchanges) 3. All API3B (Cancellations) 4. All API4B (Refunds)
6.	IFG API Execution	Based on sorted transaction set, per transaction proceed to following: a. Construct IFG API RQ based on the API field matching as set in Appendix. a. In related mapping use NDC in channel, if ZZZ is NDC and NONGDS if ZZZ=WEB in file name. b. All related API calls will be conducted using non-ifg sales flow, providing none trackID in related IFG API request. b. Make related IFG API call to related endpoint. c. System should use logic of Unique transactionID to re-execute calls when error / timeout received for transactions, at least 5 retries. d. Based on the successful/failed responses, keep track of the number of successful and failed transactions. If there is any failed response, then keep this info to further report it in related summary report. e. System should make delay between API calls in x- milliseconds, parameter defined in the IFG configuration (to prevent massive number of calls at the same time when processing the file)
7.	Summary Report	After having completed all calls towards IFG, procced to create related summary report.
8.		

9.	Logging	If there was any error that would result to reject entire file, then related file name, error code and datetime should be logged.
10.	Persistence	In terms of persistence, file inputs and file input data won't be persisted. Input files will remain within IFG for 3 months. Related API calls towards IFG and related IFG responses will be persisted under IFG as developed so far.

Table 1- Process sub-steps for DRGM.PARSE_INPUT_FILE

Response Codes

Code		Description
PRS001	Success	File %s was parsed successfully
PRS002	Error	The format of file name %s is not correct.
PRS003	Error	There was an integrity error in %s record and file <%s filename>
PRS004	Error	Invalid BSP country code in %s record and file <%s filename>
PRS005	Error	This file type is not supported by customer settings

S.RET_ CONVERTOR_EVALUATION_REPORT

Internal Reference	S.RET_CONVERTOR_EVALUATION_REPORT
Description	This process is responsible for generating summary reports of related files after being parsed within IFG, for cases when initial file contained transactions that were rejected, when file contained none errorous transaction and when file was totally rejected – so in all cases after the file processing
Actors	IFG system
Required Inputs	N/A
Expected Initial State	Process DRGM.PARSE_INPUT_FILE must have been proceeded.
Steps	Create a file with following name as: XXerAAAC_YYYYMMDD_sequence_original-file-descriptor where: • XX: ISO country code • Er: File descriptor for error report • AAAC: It is the Airline 3D code who previously sent the file plus check digit (Mod7). • YYYYMMDD: File generation date • Sequence: 000, if it is the first file uploaded for this user in that date. An incremental number for next files sent for the same airline at the same date. • Original-file descriptor: File descriptor of the original file i.e. ew, eg, et or ep The content of generated file are as: IFG NDCLINK SUMMARY REPORT (for ZZZ=NDC of original file) or IFG WEBLINK SUMMARY REPORT (for ZZZ=WEB of original file) RPSI: WEBL or EDIS (as per RPSI) AIRLINE CODE: <ifg_code> NAME: <airline name> DATE: 2007-03-07 TIME: 10:47:03 SUMMARY FOR: <original file name parsed> --- Total of Read Transactions: <total transactions read from initial file) Total of Accepted Transactions: <number of succesfull IFG API calls> Total of Rejected Transactions: <number of rejected IFG API calls + invalid transactions parsed> Percentage of Accepted Transactions: <percentage of successful transactions> -- In case there are errors, then all parsing errors will be depicted in section DATA MAPPING ERRORS as: Line: <file line> TRNN: <line value> Record: <record name> Element: # <element sequence> <element code> <element name>

	Error: error description --- In case there are errors from IFG level this will be shown in section DATA SUBMISSSION ERROORS as: Document: <TDNR> <API NAME> <API code>, HTTP Status: <HTTP status>, trackID: <trackID>, description:<API code description> Generated file will be sent to iINET using a different account compared to those used for RET files – iiNet account will be provided by IATA when development will start (for UAT, SNB and PROD environments) Generated file will be logged with the direction Sent along with the filename and timestamp Attached sample structure in Appendix: Attached Documents
Alternative flows	None
Error Conditions	N/A
Expected Final State in case of successful completion	Airline files have been successfully parsed by IFG system and IFG calls have been sent to IFG APIs. The summary report will be generated and will describe related summary of these processes.
Functional Module Logging	Generated file will be logged with the direction Sent along with the filename and timestamp

API16.REQUEST_3DS

1st Step – Initiation

Internal Reference	API16.REQUEST_3DS
Description	This is the 1st step of 3DS process through which an IBE or Airline System requests 3D Secure verification for a payment. This is a Web API call to initiate 3DS process for a CC payment as a new REST endpoint [**POST Endpoint**: /v4/3ds]
Actors	IBE or Airline System henceforth called "the client"
Inputs	See Inputs *as described in CR53720 -3DS BRD Level 2.docx*
Expected Initial State	N/A
Pre-start conditions	• Airline has 3DS subscriptions in the current country.
Steps	See Process Steps *as described in CR53720 -3DS BRD Level 2.docx* New Requirements: basicInfrormation. channel field will also accept value "NDC"
Alternative flows	N/A
Error Conditions	The codes (see Response Codes - 3DS) are described in *as described in CR53720 -3DS BRD Level 2.docx.*
Expected Final State in case of successful completion	The IBE or Airline System has received an request for 3DS validation. The system initiates the process of 3DS by providing to client a unique identifier to track the transaction for 3DS. •
Log Entries	Audit log entry to record this activity at the API level as this is a sensitive operation.

Process Steps

Below are the high-level steps for a 3DS API Request.

	Steps	**Description**	**Error**
1.	Validate consolidator & clientID	None change	TDS011 TDS012 TDS013
2.	Field validation	None change	TDS004, TDS005, TDS020

3.	Credit card PAN substitution	None change	N/A
4.	Subscription check	None change.	TDS003
5.	Agent validation	If channel =B2C, omit agent Validation call. Otherwise, call as usual agent validation service. In case of error, exit with TDS007. New Requirements: If channel is NDC, call agent validation service as normally as if provided channel was "NONGDS"	TDS007
6.	Business Rules check	New Requirements: If channel is NDC, then run business rules taking into account provided channel type.	TDS010 TDS006
7.	Apply surcharges & discounts	None change	TDS015
8.	Process 3DS request	None change	TDS009, TDS008
9.	Persist 3DS data	None change	TDS015
10.	Log RQ & RS	New Requirements: NDC channel will be logged in order to make related queries in reports.	TDS015

Table 3- Process sub-steps for API16.REQUEST_3DS

API1.AUTHORISE_PAYMENT

Internal Reference	API1.AUTHORISE_PAYMENT
Description	API1 (ticketing & ticketless) deals with the payment authorization of an IFG transaction across the various FOPs supported by IFG
Actors	Airline system interacting with IFG via API
Required Inputs	The signature of API 1 changes as: basicInfrormation. channel field will also accept value “NDC” formOfPayment.type: CCGR will be also added in the list of supported fops, only at endpoints level there will be none configuration regarding this fop (subscriptions etc)
Expected Initial State	No initial state Required
Steps	Current Functionality : The current functionality is described in BRD Level2 API.PAYMENT_AUTHORIZE – v4.docx in section *1.4 Process Steps* New Functionality: See Process Steps
Alternative flows	None
Error Conditions	Current Functionality : The current Error Codes are described in BRD Level2 API.PAYMENT_AUTHORIZE – v4.docx in section *1.7 Response Codes* New requirements: None
Expected Final State in case of successful completion	An API1 payment authorization has been submitted successfully to authorize a payment.
Functional Module Logging	Current Functionality: Log entry to record this activity at the API level as this is a sensitive operation New Functionality: None

Process Steps

Below are the high-level steps for a payment authorization. The orange steps indicate new implementation or changes to the sub-processes.

	Steps	Description
1.	Check Idempodency	None change
2.	Credit card PAN substitution	None change
3.	Field validation	None change
4.	Subscription check	New requirements: If CCGR fop.type is provided, system should exit with FOP004
5.	Fop Currency check	None change
6.	Agent validation	New requirements: If channel is NDC, call agent validation service as normally as if provided channel was "NONGDS"
7.	Payer fop validation	New requirements: Run payer validation service in spite if there is an active subscription to fopvalidation service or not. If there is any error from this process, then exit with FOP040 or FOP041.
8.	Business Rules check	New Requirements: If channel is NDC, then run business rules taking into account provided channel type.
9.	Apply surcharges & discounts	None change
10.	Fraud detection	None change
11.	Process form of payment	None change
12.	Log RQ &RS	New Requirements: NDC channel will be logged in order to make related queries in reports.

Table 4- Process sub-steps for API1:AUTHORISE_PAYMENT

API2:CONFIRM_ORDER

Internal Reference	API2.CONFIRM_ORDER
Description	API2 (ticketing & ticketless) deals with reporting the issued travel document, following the early authorized IFG payment across the various FOPs supported by IFG
Actors	Airline System interacting with IFG via API
Required Inputs	The signature of API2 changes as. basicInfrormation. channel field will also accept value "NDC" basicInformation.systemProviderIdentifier: String 3AN. A code to identify System Provider that was used to make the reservation either on its own or on behalf of another airline. If provided, it will be reported in place of SASI RET field. monetaryAmounts. taxOnCommissionAmount:1-11N. The amount of tax levied on the commission amount due an agent. monetaryAmounts. taxOnCommissionType: 1-6AN. A code to indicate the type of tax to be levied against the agent's commission according to the applicable tax rules of the country where the BSP is located. formOfPayment.type: CCGR will be also added in the list of supported fops, only at endpoints level there will be none configuration regarding this fop (subscriptions etc)
Expected Initial State	An *API1:AUTHORISE_PAYMENT* call has been previously executed.
Steps	Current Functionality : The current functionality is described in BRD Level2 API.ORDER – v4.docx in section *1.5 Process Steps & Flow Diagram.* New Functionality: See Process Steps
Alternative flows	None
Error Conditions	For the existing error codes you can see the section *1.9. Error Codes* of BRD Level2 API.ORDER – v4.docx.
Expected Final State in case of successful completion	An API2 request has been successfully submitted. For the existing success codes you can see the section *1.8. Success Codes* of BRD Level2 API.ORDER – v4.docx
Log Entries	Current Functionality: Log entry to record this activity at the API level as this is a sensitive operation New Functionality: None

Process Steps

Below diagram describes the general processes that take place in Order process. The orange steps indicate modifications to the sub-processes. The sections that follow describe in more details the changes.

	Steps	Description
1.	Validate basicInformation and clientID	None change
2.	Validate trackID	None change
3.	Idempotency check	None change
4.	Request merging	None change
5.	PAN extraction	None change
6.	Request field validation (merged for IFG, original for external sales)	None change
7.	Subscription check	None change
8.	Agent validation	New requirements: If channel is NDC, call agent validation service as normally as if provided channel was “NONGDS”
9.	Payer fop validation	New requirements: Run payer validation service in spite if there is an active subscription to fopvalidation service or not. If there is any error from this process, then exit with RET050 or RET051.
10.	Business Rules	New Requirements: If channel is NDC, then run business rules taking into account provided channel type
11.	Ticket number generation	None change
12.	1st Dish Validation	None change
13.	Perform Capture	None change
14.	Fraud detection	None change
15.	2nd Dish validation	None change
16.	Report to BSP	New requirements:

		If channel is NDC, follow the same logic for reporting as the NONGDS case and by providing NDC value field, if needed. If basicInformation.systemProviderIdentifier is provided, then SASI field will be filled based in this identifier. Also, fields monetaryAmounts. taxOnCommissionAmount and monetaryAmounts. taxCommissionType upon provided, will be reported to fields TCTA and TCTP respectively. For the case of non-ifg sales, CCGR should be also be reported upon being provided in related API request. Reporting will be conducted as instructed on DISH 22, 23 FPTP field as CCGRNNNN or CCGR
17.	Log RQ &RS	New Requirements: NDC channel will be logged in order to make related queries in reports.
18.	Save RS	None change

Table 1:Process sub-steps for API2:CONFIRM_ORDER

API4A.EXCHANGE_ORDER

Internal Reference	API4A.EXCHANGE_ORDER
Description	API4a deals with the exchange of an existing ticket with a new ticket. For an exchange request, if an additional amount needs to be paid, it is necessary first to perform an authorization. The process remains the same as the *API2: CONFIRM_ORDER*
Actors	Airline system interacting with IFG via API
Required Inputs	The signature of API4A changes as: basicInfrormation. channel field will also accept value “NDC” basicInformation.systemProviderIdentifier: String 3AN. A code to identify System Provider that was used to make the reservation either on its own or on behalf of another airline. If provided, it will be reported in place of SASI RET field. monetaryAmounts. taxOnCommissionAmount:1-11N. The amount of tax levied on the commission amount due an agent. monetaryAmounts. taxOnCommissionType: 1-6AN. A code to indicate the type of tax to be levied against the agent's commission according to the applicable tax rules of the country where the BSP is located. formOfPayment.type: CCGR will be also added in the list of supported fops, only at endpoints level there will be none configuration regarding this fop (subscriptions etc)
Expected Initial State	1. For the new ticket an *API1: PAYMENT_AUTH* call has been executed successfully, when additional collection was required. 2. For the old ticket an *API2: ORDER* has been executed successfully.
Steps	Current Functionality : The current functionality is described in BRD Level2 API.ORDER_EXCHANGE – v4.docx in section *1.5 Process Steps & Flow Diagram* New Functionality: See Process Steps
Alternative flows	None
Error Conditions	For the existing error codes you can see the section *1.9. Error Codes* of BRD Level2 API.ORDER_EXCHANGE – v4.docx
Expected Final State in case of successful completion	- An Order Exchange request has been successfully submitted. An exchange transaction has been successfully submitted. - For the existing error codes you can see the section *1.8. Success Codes* of BRD Level2 API.ORDER_EXCHANGE – v4.docx

Log Entries	Current Functionality: Log entry to record this activity at the API level as this is a sensitive operation New Functionality None

Process Steps

Below diagram describes the general processes that take place in exchange process. The orange steps indicate modifications to the sub-processes. The sections that follow describe in more details the changes.

	Steps	**Description**
1.	Validate basicInformation and clientID	None change
2.	Validate trackID	None change
3.	Idempotency check	None change
4.	Request merging	None change
5.	PAN extraction	None change
6.	Request field validation (merged for IFG, original for external sales)	None change
7.	FOP subscription check	None change
8.	Agent validation	New requirements: If channel is NDC, call agent validation service as normally as if provided channel was "NONGDS"
9.	Payer fop validation	New requirements: Run payer validation service in spite if there is an active subscription to fopvalidation service or not. If there is any error from this process, then exit with EXCH054 or EXCH055.
10.	Business Rules	New Requirements: If channel is NDC, then run business rules taking into account provided channel type
11.	Ticket number generation	None change
12.	1st Dish Validation	None change
13.	Exchange execution	None change
14.	Fraud detection	None change

15.	2nd Dish validation	None change
16.	Report to BSP	New requirements: If channel is NDC, follow the same logic for reporting as the NONGDS case and by providing NDC value field, if needed. If basicInformation.systemProviderIdentifier is provided, then SASI field will be filled based in this identifier. Also, fields monetaryAmounts. taxOnCommissionAmount and monetaryAmounts. taxCommissionType upon provided, will be reported to fields TCTA and TCTP respectively. For the case of non-ifg sales, CCGR should be also be reported upon being provided in related API request. Reporting will be conducted as instructed on DISH 22, 23 FPTP field as CCGRNNNN or CCGR
17.	Log Rq & RS	New Requirements: NDC channel will be logged in order to make related queries in reports.

Table 3 - Process sub-steps for API4A:EXCHANGE_ORDER

API3B.CANCEL_ORDER

Internal Reference	API3B.CANCEL_ORDER
Description	API3b deals with the order cancellation of an IFG order transaction.
Actors	Airline system interacting with IFG via API.
Required Inputs	The signature of API3b changes as. basicInfrormation. channel field will also accept value “NDC”
Expected Initial State	1. *API2: ORDER* call has been previously executed. 2. *API4a: ORDER_EXCHANGE* or *API4b: ORDER_REFUND* for the ticket has not been executed yet. 3. In case of Non-IFG sales, no *API3b: ORDER_CANCEL* request is executed.
Steps	Current Functionality: The current functionality is described in BRD Level2 API.ORDER_CANCEL – v4.docx in section *1.5 Process Steps & Flow Diagram* New Functionality: See Process Steps
Alternative flows	None
Error Conditions	No new error codes for Order Cancel process. You can see the existing error codes in the section *1.9. Error Codes* of BRD Level2 API.ORDER_CANCEL – v4.docx.
Expected Final State in case of successful completion	No new success codes for Order Cancel process. You can see the existing success codes in the section *1.8. Success Codes* of BRD Level2 API.ORDER_CANCEL – v4.docx.
Log Entries	Current Functionality: Log entry to record this activity at the API level as this is a sensitive operation New Functionality: None

Process Steps

Below are the high-level steps required for an Order Cancel. The orange steps indicate new implementation or changes to the sub-processes for Credit Limits. The sections that follow describe in more details the changes.

Steps	Description
1. Validate basicInformation.	None change
2. Format validation	None change
3. Validate trackID.	None change
4. Track cancellation requests.	None change
5. Track order request.	None change
6. Validate clientID.	None change
7. Validate duplicate request.	None change
8. Subscription check.	None change
9. TDNR check.	None change
10. TicketDocumentNumber check.	None change
11. Cancellation check via trackID	None change
12. Exchange/Refund check via trackID	None change
13. Capture check.	None change.
14. Check BSP cut-off time.	None change
15. Validate agent.	New requirements: If channel is NDC, call agent validation service as normally as if provided channel was "NONGDS"
16. Save Request.	None change
17. Perform cancellation	None change
18. 2nd Dish validation.	None change
19. BSP subscription check	New requirements: If channel is NDC, follow the same logic for reporting as the NONGDS case and by providing NDC value field, if needed.
20. Log Rq & RS	New Requirements: NDC channel will be logged in order to make related queries in reports.

API4B.REFUND_ORDER

Internal Reference	API4B.REFUND_ORDER
Description	API4b deals with the Order Refund of an IFG order transaction. In order for a payment to be refunded, it must have been captured first, as is the case with refunds in general. If ticket hasn't been reported via API 2 Order or Exchange yet - a simple authorization cancel will suffice.
Actors	Airline system interacting with IFG via API.
Required Inputs	The signature of API4b changes as: basicInfrormation. channel field will also accept value "NDC" In related formOfPayment and reportedFormOfPayment elements following fields will be available as well: formOfPayment. customerFileReference: String 1-27AN reportedFormOfPayment. customerFileReference: String 1-27AN formOfPayment.type: CCGR will be also added in the list of supported fops, only at endpoints level there will be none configuration regarding this fop (subscriptions etc)
Expected Initial State	1. *API2: ORDER* call has been previously executed with payment status "Captured" 2. *API3b: ORDER_CANCEL* or *API4a: ORDER_EXCHANGE or API4b: ORDER_REFUND* has not been executed yet.
Steps	Current Functionality : The current functionality is described in BRD Level2 API.ORDER_REFUND – v4.docx in section *1.4 Process Steps* New Functionality: See also Process Steps
Alternative flows	None
Error Conditions	You can see the existing error codes in the section *1.7. Response Codes* of BRD Level2 API.ORDER_REFUND – v4.docx.
Expected Final State in case of successful completion	You can see the existing error codes in the section *1.7. Response Codes* of BRD Level2 API.ORDER_REFUND – v4.docx.
Log Entries	Current Functionality: Log entry to record this activity at the API level as this is a sensitive operation New Functionality: None

Process Steps

Below are the high-level steps for an Order Refund. The orange steps indicate modifications to the sub-processes. The sections that follow describe in more details the changes.

	Steps	Description
1.	Populate refundDocumentNumber if missing	None change
2.	Validate basicInformation and clientID	None change
3.	Generate or fill relatedTDNRS	None change
4.	Fields validation	None change
5.	Idempotency check	None change
6.	Payment status check	None change
7.	Validate FOPs	None change
8.	Validate TDNRs	None change
9.	FOP subscription check	None change
10.	Agent validation	New requirements: If channel is NDC, call agent validation service as normally as if provided channel was "NONGDS"
11.	Payer Validation	New requirements: Run payer validation service in spite if there is an active subscription to fopvalidation service or not. If there is any error from this process, then exit with RFND060 or RND061.
12.	Business rules	New Requirements: If channel is NDC, then run business rules taking into account provided channel type
13.	1st Dish Validation	None change
14.	Credit limit checks	None change
15.	Refund execution	None change
16.	2nd Dish validation	None change
17.	Report to BSP	New requirements:

		If channel is NDC, follow the same logic for reporting as the NONGDS case and by providing NDC value field, if needed. Also, related field customerFileReference will be added in related RET record, if provided. For the case of non-ifg sales, CCGR should be also be reported upon being provided in related API request. Reporting will be conducted as instructed on DISH 22, 23 FPTP field as CCGRNNNN or CCGR
18.	Log Rq & RS	New Requirements: NDC channel will be logged in order to make related queries in reports.

Table 4 - Process sub-steps for API4B:REFUND_ORDER

API15.GET_PAYMENT_DETAILS

Internal Reference	API15.GET_PAYEMENT_DETAILS
Description	API15 (ticketing & ticketless) deals with getting details about the supported fops of a client. In case, user wants to get details about a specific IFG FOP product, then the system will provide these details for all FOPs as described in formOfPayment.details element. Currently, only AFOPs will be supported.
Actors	Airline system interacting with IFG via API
Required Inputs	API15 signature changes as: basicInfrormation. channel field will also accept value "NDC"
Expected Initial State	No initial state required
Steps	See Process Steps
Alternative flows	None
Error Conditions	See related section in *AFOP-Adyen Checkout Level 2 BRD.docx*
Expected Final State in case of successful completion	An API15 get payment details request has been submitted successfully. The user is presented with all available FOPs for the customer. If request provides formOfPayment.details[], then in the response it is enclosed info about banks.
Functional Module Logging	Log entry to record this activity at the API level as this is a sensitive operation

Process Steps

Below are the high-level steps for the new endpoint. The orange steps indicate modifications to the sub-processes. The sections that follow describe in more details the changes

	Steps	**Description**
1	Check authentication	None change
2	Field validation	None change.
3	Subscription check	None change

4	Agent validation	New requirements: If channel is NDC, call agent validation service as normally as if provided channel was "NONGDS"
5	Business Rules check	New Requirements: If channel is NDC, then run business rules taking into account provided channel type.
6	Get all FOP details	None change
7	Log RQ &RS	Log RQ & RS

API17.PAYOUT

Internal Reference	API17.PAYOUT POST: payment/payout
Description	API17 is responsible for making payouts. The actor will be sending funds to a recipient. Currently, the recipient will be an agent and this functionality will be available only for AP FOP.
Actors	Airline system interacting with IFG via API.
Required Inputs	The signature of API17 changes as: basicInfrormation. channel field will also accept value "NDC"
Expected Initial State	None
Steps	See section Process Steps Related SAP service will be charged as one call per entry (as done in other APIs for AP fop).
Alternative flows	None
Error Conditions	See error codes in section Response Codes
Expected Final State in case of successful completion	No new success codes for Capture process. You can see success codes in section Response Codes API Response: - status: String. one of the codes as described in section Response Codes - description: String . Available in case of error - httpstatus: Enum {200 - Success , 201 – Created, 400 – Bad Request, 401- Unauthorized, Status: 403 – Forbidden, Status: 404 - Not Found} - trackID: The related trackId of the request. - formOfPayment: Related trackID formOfPayment.type. Available for success codes. - paymentStatus: the payment status of trackID. Available for success codes.
Log Entries	Log entry to record this activity at the API level as this is a sensitive operation

Process Steps

Below are the high level steps required to make a payout. The orange steps indicate modifications to the sub-processes. The sections that follow describe in more details the changes.

	Steps	Description

1.	basicInfomation & clientID checks	None change
2.	Check Idempodency	None change
3.	Field validation	None change
4.	Subscription check	None change
5.	Fop Currency check	None change
6.	Agent valiadtion	New requirements: If channel is NDC, call agent validation service as normally as if provided channel was "NONGDS"
7.	Business Rules check	New Requirements: If channel is NDC, then run business rules taking into account provided channel type.
8.	Process payout	None change
9.	Log RQ &RS	New Requirements: NDC channel will be logged in order to make related queries in reports.

DRGM.GENERATE_CUSTOMER_API_CREDENTIALS

Internal Reference	DRGM.GENERATE_CUSTOMER_API_CREDENTIALS
Description	This is the process through which Api credentials are created on behalf of an IFG customer. The IFG customer provides this information in all API calls.
Actors	IATA admin
Required Inputs	• Username • Password • IfgCode
Expected Initial State	IATA admin has already created an IFG customer with related IFG code. A new customer user linked to this IFG customer will be assigned specific API credentials and have the responsibility to make API calls.
Pre-start conditions	IATA admin has already created an IFG customer with related IFG code.
Steps	Current functionality: 1. IATA admin imitates a process and informs JRT IT team to create a new account for a customer user that will be linked to an existing IFG customer. 2. IATA admin provides confidentially related info of ifgCode of the customer to whom new customer user will be linked, the username of the customer user, the password of the customer user. 3. JRT Database Admin will manually insert this information in IFG system in order to link related customer user with related API credentials. 4. Related password is saved in hashed code text New requirements: 4. Related password has to be also saved in non-hashed text as this info has to be loaded and attached to related API calls as described below in all related Agency portal processes (see all APUI.XXXXX processes) 5. In addition, related information should contain the ifgCode and the customer user information that will use related credentials in IFG API calls.
Alternative flows	N/A
Error Conditions	N/A
Expected Final State in case of successful completion	A process is initiated by IATA admin to generate API credentials for a specific customer user that is linked to an IFG customer. JRT Database Admin successfully inserts this information in IFG system.
Log Entries	N/A

Appendix: Attached Documents

Additional Files

DISH-23-API-mapping_DISH22 DISH20.

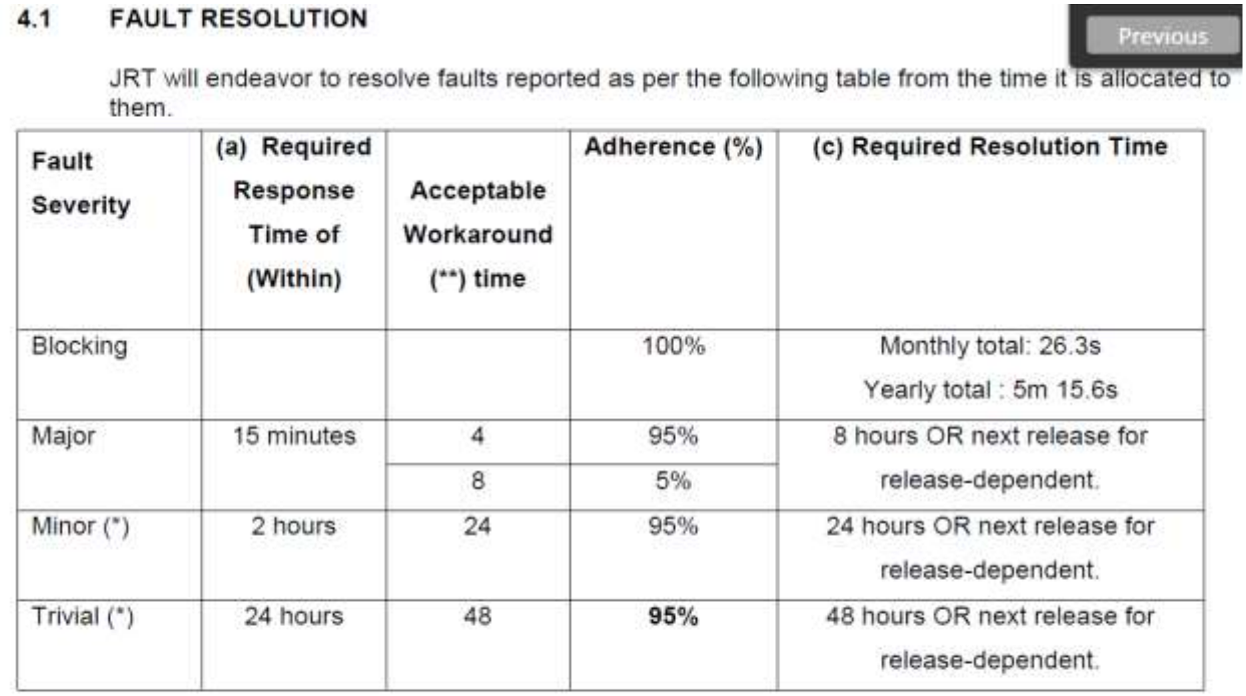
API Converter evaluation report v1

IATA BRD Level 1

IFG processing for airline sales files rep

SLA

In terms of SLA restrictions, JRT system will have to ensure that from the time it receives a related file for parsing will have to parse and send related API calls of enclosed transactions within 8 hours as indicated in table below in section Major. In case enclosed transactions cannot be reported to IFG system due to configuration settings or because related file is invalid or because the file was not delivered to JRT system from beginning, JRT will not bare any responsibility as indicated in SLA restrictions below.



4.1 FAULT RESOLUTION

JRT will endeavor to resolve faults reported as per the following table from the time it is allocated to them.

Fault Severity	(a) Required Response Time of (Within)	Acceptable Workaround (**) time	Adherence (%)	(c) Required Resolution Time
Blocking			100%	Monthly total: 26.3s Yearly total : 5m 15.6s
Major	15 minutes	4	95%	8 hours OR next release for release-dependent.
		8	5%	
Minor (*)	2 hours	24	95%	24 hours OR next release for release-dependent.
Trivial (*)	24 hours	48	**95%**	48 hours OR next release for release-dependent.

Document Sign-Off

Name	Role	Date
Dmitry Kovalskiy	Product Development and Planning manager (IATA)	
Pantelis Bousios	Senior Project Manager (JRT)	
Kalliopi Kalantzaki	Business Analyst (JRT)	