



SWIFT Partners

SWIFTReady Exceptions and Investigations

Label criteria 2010

Version 2

January 2010

Legal notices

Copyright

SWIFT © 2010. All rights reserved.

You may copy this publication within your organisation. Any such copy must include these legal notices.

Disclaimer

SWIFT supplies this publication for information purposes only. The information in this publication may change from time to time. You must always refer to the latest available version.

Translations

The English version of SWIFT documentation is the only official version.

Trademarks

SWIFT is the trade name of S.W.I.F.T. SCRL. The following are registered trademarks of SWIFT: SWIFT, the SWIFT logo, Sibos, SWIFTNet, SWIFTRReady, and Accord. Other product, service, or company names in this publication are trade names, trademarks, or registered trademarks of their respective owners.

Table of content

1	SWIFT FOR EXCEPTIONS AND INVESTIGATIONS: HIGH LEVEL INTRODUCTION.....	5
2	THE SWIFTREADY EXCEPTIONS AND INVESTIGATIONS LABEL	5
3	SWIFTREADY SOLUTION CRITERIA 2010.....	6
3.1	NEW CRITERIA FOR 2010.....	6
3.2	INSTALLED CUSTOMER BASE.....	6
3.3	CONNECTIVITY	6
3.4	MESSAGING	7
3.5	STANDARDS	8
3.6	MESSAGE RECONCILIATION	9
3.7	MESSAGE VALIDATION	9
3.8	BUSINESS WORKFLOW.....	10
3.9	USER INTERFACE	11
3.10	USER PROFILE MANAGEMENT	12
4	MARKETING AND SALES.....	12
5	RELATED DOCUMENTATION	12

Preface

Purpose of this document

This document explains the criteria needed to obtain the SWIFTReady Exceptions and Investigations 2010 label for your business application. The intended audience are Application Product Managers and Developers. Please refer to the following set of documents for further info on the SWIFTReady label programme:

Related documents

- [SWIFTReady applications guide](#)

Explains the 'Why and How' on joining the SWIFT Partner Management framework and its related SWIFTReady Accreditation programmes.

- [SWIFTReady criteria portfolio](#)

Explains the 'What' in a generic yet detailed manner on the criteria of your SWIFTReady Application.

- [SWIFTReady technical validation guide](#)

Explains the 'How' in a detailed manner on how your application will be validated to become SWIFTReady.

1 SWIFT for Exceptions and Investigations: high level introduction

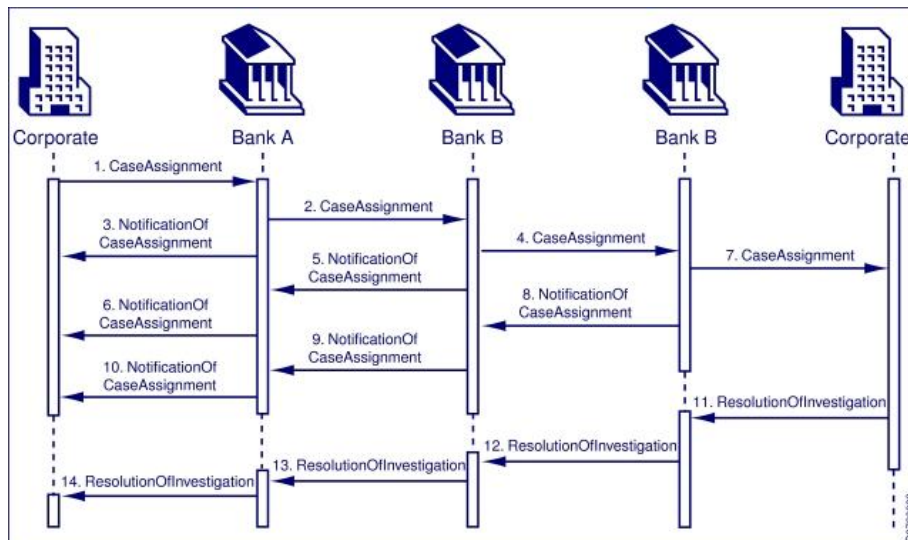
SWIFT for Exceptions and Investigations aims at supporting banks and their customers in their effort to streamline their payment related enquiries management processes. This is achieved by automating enquiries which have the potential to be automated, hereby increasing also the efficiency in handling enquiries that still require some manual intervention.

The following release is currently available for Exceptions and Investigations:

- Exceptions and Investigations 1.2 relating to the bank to bank environment as well as corporate to bank environment

Exceptions and Investigations combines the use of:

- 4 case assignment and 12 case management XML messages to be used in a bank-to-bank and corporate-to-bank environment. *Please note that the business content of the MXs used in the E&I Bank-to-Bank space is the same as in the E&I Corporate-to-Bank space. However, there is a slight difference in the header of the messages, hence the reference to two releases and two sets of documentation.*
- InterAct in Store and Forward mode. *Please refer to the relevant Service Description and the Integration Guide for a complete description of features and functions.*
- A Rulebook setting out the rules and describing best practice guidelines applicable to all Exceptions and Investigations users. This includes specific workflows supporting each exception or investigation process, i.e. Request for Cancellation, Request for Modification, Unable to Apply and Claim Non Receipt. The following workflow diagram describes the generic flow of messages for an enquiry from creation to closing.



2 The SWIFTReady Exceptions and Investigations label

To integrate SWIFT for Exceptions and Investigations into existing applications or databases and to fully automate the information flows, service users must enhance their existing application infrastructure or obtain new applications. They can develop applications in-house or outsource this to a third-party software vendor. The integrated solution links a SWIFT interface to back-office applications or to a Financial Enterprise Application Integration (EAI) solution.

The purpose of this label is to ensure that third party applications are interoperable with each other, supporting the Exceptions and Investigations standards and rules as described in the SWIFT User Handbook. Therefore the SWIFTReady Exceptions and Investigations label criteria focus on the ability of an application's support for:

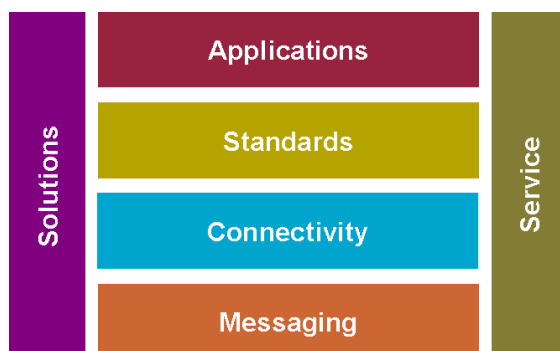
- Messaging: the InterAct messaging flows defining the request-response communication between sender/receiver and the store and forward service.
- Standards and Rulebook: all Exceptions and Investigations MT/MX standards and usage rules as outlined in the SWIFTStandards MX Message Reference Guide.

- Business workflows: a workflow defines the series of messages to be exchanged and the sequence they have to be sent in.

The SWIFTReady label criteria require automation based on local application intelligence to support the communication workflows (for example: the application should automatically generate and return a NotificationOfCaseAssignment when forwarding a case to the next party in the payment chain).

Integration is bank specific and will have to be included in the bank's requirements.

3 SWIFTReady Solution criteria 2010



3.1 New criteria for 2010

No new business criteria were added for 2010, compared to 2009, with the exception of an alternative connectivity option to SWIFT interfaces through the SOAP Host Adapter.

The SWIFTReady Exceptions and Investigations 2010 label requires the candidate application to support both the bank-to-bank and the corporate-to-bank functionality.

3.2 Installed Customer Base

To be eligible for the 2010 label the Exceptions and Investigations functionality of your solution must be implemented and used by at least 1 customer* using SWIFT for Exceptions and Investigations. The customer must be live or must at least have tested all Exceptions and Investigations messages.

**By 'customer' we mean separate financial institutions using the product to generate/receive messages transported over SWIFTNet.*

SWIFT reserves the right to contact the relevant customer to validate the functionality of the application submitted for SWIFTReady certification. A questionnaire will be used as the basis for the customer validation which can be in the form of a telephone interview, an e-mail or a discussion at the customer site. The information provided by the customer will be treated as confidential and will not be disclosed, unless explicitly expressed by the customer.

Important note: *If the partner has successfully passed all certification steps to obtain the 2010 label for his application, but fails to have a customer, by which the application is used as described above, his application will receive a **conformance statement**. This statement will automatically be transferred into the SWIFT Ready label once the partner has a customer that uses the E&I solution to send and receive live or at least test traffic. For more details, please refer to the [SWIFT Partner Programme Terms and Conditions](#).*

3.3 Connectivity

The application must be able to connect to Alliance Access or Alliance Gateway either directly, via one of the adapters mentioned below, or indirectly, preferably via one of the SWIFTReady Financial Enterprise Application Integration applications (EAI).

An application which does not support a link to an Alliance interface, even though it may support a link to a 3rd party SWIFT interface, will not be considered for a SWIFTReady label.

3.3.1 Option 1: Direct connectivity

In preference your application should connect to Alliance Access or, alternatively to Alliance Gateway.

Alliance Access provides a resilient Store and Forward local messaging hub and manages the SWIFT protocol (security, network and Store and Forward queues management), making it much simpler to integrate with SWIFT than Alliance Gateway.

The business application to Alliance Access connection can be achieved using the Alliance Access adapters, the MQ Series Alliance Access Adapter (MQSA), MQHA the new WebSphere MQ Adapter for Alliance Access (available as of release 6.2), Automated File Transfer (AFT) or SOAP Host Adapter. When connecting directly to Alliance Access, the business application should:

- Add the specific Alliance Access messaging interface header to the business payload (business payload consists of application header + Exceptions and Investigations business message).
- support the XML v2 data format which will replace XML v1 and will be the only Alliance Access data format as of Alliance Access release 7.0 (planned for distribution during the fourth quarter of 2010).

There are various options to connect applications to **Alliance Gateway**, for example MQ Series Host Adapter or Remote Access Host Adapter.

When connecting your application directly to Alliance Gateway, your application is required to handle the messaging protocol with SWIFT (for example the InterAct Store and Forward protocol used in the Exceptions and Investigations solution).

Important note: *Integrating with Alliance Gateway using these adapters is a complex enterprise requiring SWIFTNet protocol expertise and should be restricted to situations where Alliance Access cannot be used. As of SWIFTNet R7.0, any application that interfaces directly with Alliance Gateway using RAHA or MQHA, will be subject to qualification as SWIFT Interface provider.*

3.3.2 Option 2: Indirect connectivity

Alternatively, you can prove your application's InterAct support by providing SWIFT with evidence of an indirect connectivity solution consisting of your business application and a middleware (Enterprise Application Integration) solution. This is preferably one of the **SWIFTReady Financial Enterprise Application Integration** applications. The complete and up-to-date list of SWIFTReady applications can be found on www.swift.com/Partners

Indirect connectivity can be accepted only for business applications applying for a SWIFTReady Exceptions and Investigations label. Furthermore, its combination with an Enterprise Application Integration application needs to be proven through Integration Test Bed (ITB) testing.

When connecting your application indirectly to Alliance Gateway, the application or the Financial Enterprise Application Integration solution will be required to handle the connectivity to SWIFT (i.e. the InterAct Store and Forward protocol used in SWIFT for Exceptions and Investigations).

Please note that when connecting indirectly to Alliance Access, the business application or the Financial EAI solution is required to add the specific Alliance Access messaging interface header to the business payload (business payload consists of application header + Exceptions & Investigations business message).

Only on an exceptional basis will SWIFT Partner Management accept proof of your compliance with the connectivity criteria using the live or test and training connection of an Exceptions and Investigations test or production customer. In this case SWIFT Partner Management reserves the right to contact the relevant financial institution for further information.

3.4 Messaging

The application (optionally through a Financial EAI application) must support InterAct and its mandatory associated features as listed in the Service Description and Integration Guide for Exceptions and Investigations 1.2.

3.5 Standards

The application must support all case assignment messages and all case management messages following the rules as described in the SWIFT Standards MX Message Reference Guide 1.2 as well as the related FIN messages.

List of Message Types required for SWIFTReady Exceptions and Investigations

Request Type	Request Name	Incoming/ Outgoing
camt.007.002.02	RequestToModifyPaymentV02	✓
camt.008.002.02	RequestToCancelPaymentV02	✓
camt.026.001.02	UnableToApplyV02	✓
camt.027.001.02	ClaimNonReceiptV02	✓
camt.028.001.02	AdditionalPaymentInformationV02	✓
camt.029.001.02	ResolutionOfInvestigationV02	✓
camt.030.001.02	NotificationOfCaseAssignmentV02	✓
camt.031.001.02	RejectCaseAssignmentV02	✓
camt.032.001.01	CancelCaseAssignmentV01	✓
camt.033.001.02	RequestForDuplicateV02	✓
camt.034.001.02	DuplicateV02	✓
camt.035.001.01	ProprietaryFormatInvestigationV01	✓
camt.036.001.01	DebitAuthorisationResponseV01	✓
camt.037.001.02	DebitAuthorisationRequestV02	✓
camt.038.001.01	CaseStatusReportRequestV01	✓
camt.039.001.02	CaseStatusReportRequestV01	✓
	and	
MT192/MT292	Request for cancellation	✓
MT195/MT295	Queries	✓
MT196/MT296	Answers	✓
MT199/MT299	Free Format Message	✓

All changes to the messages must be supported by the application before the live release date on the SWIFT network. When new messages are introduced or those existing are significantly modified, we expect the application provider to provide adequate testing time to his customers prior to these messages going live.

3.5.1 Library of XML message templates

To minimise the need for manual entry, the application must provide a library of all standard Exceptions and Investigations XML messages which an automated process or operator can select from.

3.5.2 Correct payload structure

The application must be able to demonstrate the correct payload structure of the 16 XML schemas.

3.5.3 FIN/XML co-existence

The application must be capable to send/receive the Exceptions and Investigations XML and FIN (n92, n95, n96, n99) messages.

The combination of directory functionality within the application, containing the Exceptions and Investigations counterparties and the parallel support of FIN and XML investigation messages should enable the application to automatically create the next message to the following party in the payments chain in the appropriate syntax (FIN or XML).

The application is required to incorporate the Unique Case ID in the generated MT or MX message.

The SWIFTNet Services Directory on www.swift.com lists all test and live users that participate in Exceptions and Investigations. The application must enable a user to upload and manually update the available information from the Directory.

Note: To allow for consistency and a single standard support going forward, SWIFT plans to remove all FIN payment-related enquiry messages at the end of 2012, provided specific interim milestones are achieved.

3.5.4 Bank-to-bank and corporate-to-bank message flows

The Exceptions and Investigations solution is set up in two distinct SWIFTNet services to support the corporate-to-bank or the bank-to-bank message flows.

The application should support both environments.

Based on the information downloaded from the SWIFTNet Services Directory on www.swift.com, the application should be able to decide in which Exceptions and Investigations service the message has to be sent, i.e. the Exceptions and Investigations corporate-to-bank service or the Exceptions and Investigations bank-to-bank service, in the test & training or the live environment.

The application should provide the messaging interface with the above information to enable the appropriate InterAct header to be generated. The InterAct header will either be generated by Alliance Access (when integrating the application with Alliance Access) or by the business application or an EAI (when integrating the application with Alliance Gateway).

Please note that the service name for Exceptions and Investigations for corporate-to-bank (swift.corp.eni) in the header of the InterAct message differs from the one used in the bank-to-bank environment (swift.eni). For example:

- a corporate sending an investigation to a bank uses swift.corp.eni (live environment) or swift.corp.eni!p (test and training environment)
- a bank forwarding this investigation to another bank uses swift.eni (live environment) or swift.eni!p (test and training environment)
- a bank sending a notification to its corporate uses swift.corp.eni (live environment) or swift.corp.eni!p (test and training environment)

3.6 Message Reconciliation

The application must be able to reconcile technical messages such as SWIFTNet acknowledgements and delivery notifications. If the application connects to Alliance Access, the reconciliation of the local Alliance Access message status is also required.

3.7 Message Validation

All MX messages should be validated against the relevant XML schemas and against Extended Validation Rules that are provided in the Rulebook and SWIFT Standards MX Message Reference Guide 1.2

The application must provide validation on field and message level.

- On field level, the data structure such as length and structure of currency, BIC/BEI, date format and field length must be validated. The user must be prompted to correct the information if this is not according to the specified rules.

- On field content level, the user should be stopped if his action is not in line with the User Handbook or Exceptions and Investigations Rulebook (e.g. a RequestToModifyPayment must never be sent to request the modification of the currency of the original payment instruction).
- On message level, the application must provide the correct mapping including business information to the right XML tag.

It is important to clarify the date/time usage in messages. An application is often using local date/time but without indication, so that the receiver does not know what is the sender's local time. It is therefore strongly recommended to provide a non-ambiguous way of specifying date/time by either referring to GMT time or by giving the time offset as required by the ISO date time used in the schema.

3.8 Business Workflow

3.8.1 Automating the case assignment messages

The application must have the ability to provide a minimum level of straight through processing (ST) for the case assignment messages:

- when appropriate, the application must automatically generate a NotificationOfCaseAssignment,
- before closing a case the application must automatically generate a ResolutionOfInvestigation,
- when receiving a CaseStatusReportRequest, the application must be able to automatically generate a CaseStatusReport,
- for all other messages (if not fully automated) the operator should be offered a dropdown list of appropriate (not all) messages to select from. The application should also provide automation such as pre-filling/copying fields, from the underlying payment message into the appropriate Exceptions and Investigations message and from an incoming Exceptions and Investigations message into an outgoing Exceptions and Investigations message.

3.8.2 Support of the Unique Case Identifier

The application must be capable of generating a unique case identifier. This unique case identifier should not be changeable by the operator.

In order to make the case identifier unique for all the parties involved in a workflow, it is composed of two distinct parts: the case creator identification (usually a BIC or BEI) and a unique number (Casecreator reference).

If a sequential number is used for the Case creator reference, the range of numbers should be large enough to avoid ambiguity when restarting the sequence. Alternatively, date and time can be used followed by a sequential number.

3.8.3 Re-use of the Unique Case Identifier (Case ID)

The case assignee must be able to re-use the unique case identifier (the combination of the case creator and case creator reference) in all its communications with both its case assigner and possible further case assignees during the case life cycle.

3.8.4 Implementation of the 'Re-open' flag

Closed cases can be re-opened and re-assigned. When this happens the same unique case identifier is to be used as the one in the initial case, with a flag indicating that this is a re-opened case.

The application must allow for a case to be re-opened and the same case identifier (Case ID) to be re-used. The case can be re-opened as a new type by the initiating or the final party (i.e. an UnableToApply can be re-opened as a RequestToCancelPayment)

The application has to ensure that a message received with a re-open flag triggers the re-open flag to be passed on in all subsequent messages.

3.8.5 Relating the underlying payment instruction identification to Unique Case Identifier

When a case is assigned to a case assignee, the case assignee must first check the validity of the assignment. If the assignment is valid, the case assignee must check that there is no other case open on this underlying instruction.

- If there is no other case open for the same payment instruction, the case assignee must accept the case - acceptance is implicit.
- If there is an open case for the same payment instruction, the case assignee will request the closure of one of the open cases. This is achieved by sending a ResolutionOfInvestigation message indicating that the case is a duplicate of another case (with the reference to the case). The receipt of the ResolutionOfInvestigation with the DuplicateOf filled in will close the current case.

The application should guide the investigator also on how to handle concurrent workflows by prompting the investigator to select the assignment, which has priority, as stated in the table below, and close other assignment with an informative message.

When assignee has	Unable to Apply	ClaimNonReceipt
Request to modify payment	Continue with Request to Modify Payment	Continue with Request to Modify Payment
Request to cancel payment	Continue with Request to Cancel Payment	Continue with Request to Cancel Payment
Unable to Apply	N/A	Continue with Unable to Apply

Some examples:

- When a RequestToCancelPayment is received as well as an UnableToApply for the same payment, the application should be able to identify these investigation messages as relating to the same underlying payment and should update the case. The application should warn the investigator of the concurrent enquiry requests and prompt the investigator to take the appropriate action i.e. send a ResolutionOfInvestigation to the assigner of the UnableToApply indicating that cancellation will follow and process the RequestToCancelPayment
- When the same investigation has been received twice for the same underlying payment, the application should warn the investigator that a case is already open and prompt him to send a ResolutionOfInvestigation confirming duplication.

3.8.6 Support of the No By-Pass rule

The 'no by-pass' rule specifies that no party involved in the original payment transaction can be by-passed in the exceptions and investigations workflow. The application must prevent operators from violating this rule for example by pre-populating the party field with information from the underlying payment instruction.

3.8.7 Message grouping by case

The application must be able to allocate a message to the corresponding "case-ID" and to present all messages that have been exchanged in relation to a specific case (ID). The application should group incoming and outgoing messages using date/time, Case ID, Assigner, Assignee, etc.

The application must also be able to store the incoming and outgoing messages in the right order using date/time.

3.9 User Interface

The application should allow for manual entry/display capability and repair for XML and FIN (n92, n95, n96, n99) exceptions and investigations messages. The application should allow the creation of an XML/FIN message and manual repair before re-inserting the message into the output queue. Whenever relevant, SWIFT expects the application to offer a Graphical User Interface

- Allowing a user to manually input or modify any message
- Offering normalised fields for input (independent from underlying syntax and business meaning)
- Validating data input at field level – any invalid entry must be flagged, and the user prompted to correct the input
- Providing the user with an intuitive method to follow the status of a particular case

3.10 User Profile Management

The application must provide a user profile management tool (user-ID + password) to ensure that only authorised users can perform a specific task. You should demonstrate how profile management is assured by demonstrating create / update / delete user profiles and how different users log on and access is denied if a certain task is performed that the user is not entitled to.

The application must also be able to support the "four eyes principle". You should demonstrate how different activities require a second person to validate them before release.

4 Marketing and Sales

The partner will work together with SWIFT developing and generating more business. To optimize the collaboration, the partner is expected to provide SWIFT with administrative and marketing information. In particular the partner should provide SWIFT, under non-disclosure agreement, following customer related information:

- A list of all customers active in the finance sector. The list should provide institution names, locations, and an overview of the integration scope (domain, features, and sites) for the present and previous year.
- A product roadmap for 2010 and 2011 containing the plans for further developments, Solutions support and new releases.
- A complete set of documentation, including features overview, where appropriate SWIFT adapters, workflow engine capability and user manuals.
- A dedicated web page on Partner web site for the SWIFTReady label.

5 Related Documentation

The following information is available on www.swift.com for Exceptions and Investigations 1.2:

- Solutions Implementation Service Overview
- Service Description
- Integration Guide
- Standards MX Message Reference Guide
- Standards MX Schemas Standards MX Samples
- Samples with InterAct Headers
- Samples with Alliance Access Headers

-end of document-