



SWIFT Partners

Alliance Access Integration – Automated File Transfer

Technical Qualification Test 2011

This document lists the tests for application providers that integrate their middleware or back-office application with Alliance Access using Automated File Transfer (AFT) Adaptor.

Version 1

January 2011

Legal Notices

Copyright

SWIFT © 2011. 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, SWIFT, SWIFTReady, and Accord. Other product, service, or company names in this publication are trade names, trademarks, or registered trademarks of their respective owners.

Table of Contents

1	Preface	4
1.1	Purpose.....	4
1.2	Audience	4
1.3	SWIFTReady Programme	4
1.4	Related Documents	4
2	Introduction	5
2.1	SWIFTReady Validation	5
3	Alliance Access configuration	6
3.1	Upgrade to Release 7.0.....	6
3.2	Message Partner configuration.....	6
3.3	Exchanging messages using Message Partner	6
4	Wipro Testing Service	7
5	AFT Test Preparation	8
5.1	Reconciliation of Delivery Notification.....	8
5.2	Transmission Notification.....	8
6	SWIFTReady AFT testing	9
6.1	Test Scenarios	9
6.2	Test Evidence	9
7	Annex	10
7.1	Sample Message Partner Configuration.....	10
7.2	Input MT Message sample in RJE format	13
7.3	Input MT Message sample in XML v2 format	13
7.4	ACK Message with Original Message	13
7.5	Delivery Notification Message	14
7.6	Output MT Message sample in RJE format.....	14
7.7	NAK Message with Original Message	14

1 Preface

1.1 Purpose

This document describes the test scenarios to qualify software integrating with SWIFTAlliance Access using AFT.

The purpose of the AFT qualification is to validate the generation, transmission and reconciliation of files transferred from the partner application to Alliance Access.

1.2 Audience

The target audience for this document is both vendors considering the certification of a product, and SWIFT Users that look after an overview of the SWIFTReady Label contents. This document guide the application providers that integrate their middleware or back-office application with Alliance Access using AFT. The audience should be familiar with SWIFT world from both a technical and a business perspective.

1.3 SWIFTReady Programme

The SWIFTReady label programme covers the entire financial application chain, from Trade, Treasury and Payment, to Corporate and Securities segments.

Each SWIFTReady label defines a set of criteria, which are reviewed every year to ensure that the software remains aligned with the financial market evolution and with customer needs.

These criteria are designed to reflect the capability of a financial application to provide message processing automation in a SWIFT context, and to support straight through processing (STP) in order to increase customer value, limit customisation needs and cost, and reduce time to market.

1.4 Related Documents

- [Alliance Access 7.0 - System Management Guide](#)
- [SWIFTAlliance Access header samples.pdf](#)
- [Wipro Testing Services – User Guide](#)

2 Introduction

Alliance Access provides different integration adapters for back-office applications and middleware products. One of the most common integration adapters to Alliance Access is the Automated File Transfer (AFT).

AFT is using the standard mechanism of Message Partner definition available in Alliance Access, including routing rules and profile definition. Routing rules must also be defined to integrate each Message Partner inside the Alliance Access routing scheme.

2.1 SWIFTReady Validation

Support of AFT is one of the qualification criteria for granting the SWIFTReady label to a financial application and EAI.

The Partners who deployed the SWIFT Alliance connectivity to SWIFT ITB can perform these tests independently. Test environment configuration is described in **section 3**.

As an alternative, Wipro Testing Services offers a test service for connecting to SWIFT ITB. Using this service, the Partner can connect to SWIFT ITB. Currently, the testing service allows testing application connectivity with Alliance Access AFT and MQHA. Please go to **section 4**, if you wish to use this service.

3 Alliance Access configuration

This set-up is applicable for Partners that deployed an ITB environment and connect to Alliance Access **from their premises**.

3.1 Upgrade to Release 7.0

To prepare for the tests, the Alliance Access system must be installed and configured at Partner premises. The following upgrade must be performed before attempting to communicate through AFT.

Alliance Access must be upgraded to release 7.0. It is necessary to get acquainted with the Alliance Access 7.0 System Management Guide to further configure the interface for test purpose. The Application Interface module of Alliance Access provides all the functions necessary to manage Message Partner profiles. Using Application Interface application, the Partner needs to set up the connection profiles that are used by Alliance Access to connect with external message partners. The Application Interface allows exchanging messages with external back-office systems or "Message Partners".

3.2 Message Partner configuration

The AFT communication session is set up and controlled with a dedicated Message Partner configuration in Alliance Access. Using the procedure described in the Alliance Access System Management Guide – File Transfer Connection Method, create a Message Partner for the file transfer.

- Specify the connection method as File Transfer
- Specify the direction of message transfer (from Message Partner and To Message Partner)
- Configure the other parameters

The partner must ensure that the Alliance Access server must be running and the Message Partner is enabled. The vendor application must have access to the input directory configured in the "From" Message Partner and provide access for Alliance Access to transfer messages in the output directory configured in the "To" Message Partner.

Please refer to System Management Guide – Alliance Access 7.0 for additional information for configuring and managing Message Partner Profiles. A screenshot of sample Message Partner is provided in [section 7](#).

3.3 Exchanging messages using Message Partner

- AFT supports both MT and MX messages. MT test messages can be exchanged using RJE or the XML v2 format, while MX test messages can only be sent in XMLv2 format. The application provider will prepare and route the messages in to the AFT folders for Alliance Access to process.
- The application provider will prepare a batch file containing application generated MT or MX messages and copy it in the input directory for Alliance Access to process
- The messages being sent must be the message types supported by the partner application.
- The file name used for automatic input must not contain any space or any other special character.
- The successfully processed test messages will be stored by Alliance Access in the _SI_to_SWIFT Queue [MT test messages) or _SI_to_SWIFTNet (MX test messages) Queue.
- SWIFT Network returns Notification messages for technical reconciliation and response messages, since the test messages were used for "self transfers", meaning, the sender and receiver BIC are the same
- The application must download the Network notifications and messages sent in "Output from SWIFT" direction

4 Wipro Testing Service

- You need to liaise with Wipro to enrol into Wipro Testing Services and test the exchange of messages through this service
- The vendor will be identified as the branch of Wipro and will be provided with a Wipro Branch PIC
- This PIC must be used in the sender and receiver block for exchanging messages over ITB
- The service is only available for self transfers, meaning that the sender and receiver of the test messages and test files must bear the same PIC and DN as provided by Wipro
- The connectivity is very similar when you connect from within your environment using any standard Secured File Transfer Protocol (SFTP) software, except for the reason that this connectivity is established outside your internal network environment
- Wipro will provide user credential once the vendor agrees on the terms to avail the testing service
- For a SFTP transfer, the vendor will be provided with a root folder with input and output sub-folders at Wipro, with appropriate access rights
- The SFTP uses port 22 to connect and exchange files over internet and hence necessary permission must be obtained upfront from your IT Security team
- The application provider will connect to the Wipro and transfer the application generated MT or MX messages in the input directory for Alliance Access to process
- The messages being sent must be the message types supported by the partner application.
- The file name used for automatic input must not contain any space or any other special character
- Once the connectivity is established, the outgoing messages will be picked automatically by Alliance Access
- SWIFT Network returns Notification messages for technical reconciliation and response messages, since the test messages were used for “self transfers”, meaning, the sender and receiver BIC are the same
- The application must download the Network notifications and messages sent in “Output from SWIFT” direction

5 AFT Test Preparation

AFT supports the exchange of messages in the following data formats:

- **RJE Format (MT messages only)**

Currently, RJE (Remote Job Entry) and XML v2 formats are supported for the exchange of test messages. The RJE format is only applicable for MT messages, while XML v2 format is applicable for both MT and MX messages.

In RJE formatted messages, all fields in Block 4 (Message Text Block) have a Carriage Return and Line Feed (CrLf) at the beginning of every field. For fields containing multiple lines, CrLf must be present at the end of every line. RJE format allows multiple messages sent in a batch file. The messages are separated with "\$" as delimiter and there must be no message separator at the end of the last message in the batch file. For more details about the RJE format, please refer to Alliance Access 6.3 – System Management guide.

- **XML version 2 (MX and MT messages, and files for FileAct service)**

A XML v2 message consists of an Alliance Access Header, an optional Application Header and a Message details block. In case MT messages are exchanged using XML v2 format, then the Message Block must be encrypted using Base 64.

For more information about these formats, see the Alliance Access System Management Guide – Message Formats Used in AI". A sample of the [RJE](#) and [XML v2](#) message format is provided in **Annexure**. Please refer to sam_aaccess_headers.pdf document for messages samples exchanged with Alliance Access, in scope of the SWIFT solution.

For SWIFTReady Label validation, at least one of the following file formats will be tested:

- RJE for MT message
- XML v2 format for MT or MX messages

5.1 Reconciliation of Delivery Notification

When the messages are sent to Alliance Access, the application can optionally request for a delivery notification. This will result in Alliance Access receiving a message about the message delivery, which can be reconciled with the original message.

The Partner must demonstrate the capability of their application to process the Delivery Notification Message and to reconcile it with the original message.

A sample Delivery Notification message is provided in [section 7](#).

5.2 Transmission Notification

A transmission notification is a message representing the result of transmission to SWIFT network. SWIFT performs full syntax and semantic checks before it returns an acknowledgement ([ACK](#)). Other checks, such as validity of the sender and the receiver, are also performed. These checks can cause a message to be rejected and a negative acknowledgement ([NAK](#)) is returned in response.

The vendor must demonstrate their application capability to process the Transmission Notification Message and reconcile with the original message.

A sample Transmission Notification message is provided in [section 7](#).

6 SWIFTRReady AFT testing

To be qualified as AFT compliant, Partners need to send MT and/or MX messages (depending on the label category), and act upon received MT/MX messages including ACK / NAK and delivery notifications.

6.1 Test Scenarios

1. The partner application vendor prepares the number of MT / MX test messages generated by the application as required for the SWIFTRReady label at stake. When these messages are placed on the input directory, the "From" session is initiated automatically and the test messages are processed by Alliance Access
2. Partners having their own ITB environment must use their PIC in the sender and receiver fields of the message. The Partner using Wipro Testing Services for connecting to ITB must use the PIC provided by Wipro in the sender and receiver field of the message.
3. If the partner application supports only MT messages, the test messages can be sent in either RJE format or in XML v2 format. If the business application supports SWIFT Solutions, then the only format Alliance can read is XML v2 format
4. For individual label requirements, the vendor application must exchange application supported SWIFT Messages. Vendor should refer to label specific technical validation guide for the in-scope SWIFT MT and/or MX Messages.
5. The partner application must include delivery notification instructions while generating the test messages. AFT will transmit the delivery notification either through system message. The back office application must receive the delivery notification information and reconcile with the original message sent to Alliance Access
6. The acknowledgement of transmission is sent through transmission notification message. The partner application must capture the ACK and NAK received back from SWIFT and reconcile in the application. Evidences of reconciliation mechanism (screen dump, event log, dataset extract,) will be handled back to SWIFT Qualification service provider.

6.2 Test Evidence

The Partner will extract the following evidences covering the testing period and send them via email to the Validation Service provider for Technical Validation of AFT connectivity test.

Partners with own ITB connectivity

- Alliance Access Event Journal report
- Message File report
- Samples of ASCII (MT) and XMLv2 files (MT and MX)
- Screenshots / Log File / Dataset extract / Reports generated from the Partner application evidencing the test execution through Partner application and the reconciliation mechanism against delivery notification and transmission notification for ACK and NAK

Partners testing through Wipro Testing Service

- Alliance Access Event Journal report and Message File report will be generated by Wipro
- The Partner to provide samples of ASCII (MT) and XMLv2 files (MT and MX)
- Screenshots / Log File / Dataset extract / Reports generated from the Partner application evidencing the test execution through Partner application and the reconciliation mechanism against delivery notification and transmission notification for ACK and NAK

7 Annex

7.1 Sample Message Partner Configuration

Direction: From Message Partner (to Alliance Access)

Format: RJE

The screenshot shows the 'Reception' tab of the configuration window. The 'Message partner' is 'MPA05FTMTIN' and the 'Status' is 'Enabled'. The 'Description' is 'FileTransfer of MT Messages for Branch A05'. The 'Allowed direction' is 'From Message Partner' and the 'Connection method' is 'File Transfer'. In the 'Details' section, 'Session initiation' is 'Automatic', 'Parameter file' is 'Not Required', and 'Format recognition' is 'Forced'. The 'Data format' is 'RJE'. The 'Input path name' is 'D:\BRANCHES\A05\50_0817162310_MT_PUT*'.

The screenshot shows the 'Parameters' and 'Routing' sections of the configuration window. In the 'Parameters' section, 'Validation level' is 'No Validation', 'Profile name' is 'R6.3_MsgPartner', 'Message modification allowed' is 'Allowed', 'Unit to be assigned' is 'None', and 'Batch file validation' is 'Continue on Rejection'. The 'Build unique file transfer reference' checkbox is unchecked. In the 'Routing' section, 'Disposition' is 'Dispose' and 'message in' is 'Ready-to-Send'.

Direction: To Message Partner (From Alliance Access)

Format: RJE

The screenshot shows the 'Application Interface - Message Partner MPA05FTMTOUT' window with the 'Session' tab selected. The configuration includes:

- Message partner: MPA05FTMTOUT
- Status: Enabled
- Description: To Receive out Msg(MT) for Branch A05
- Allowed direction: To Message Partner
- Connection method: File Transfer
- Transfer PKI Signatures:
- Always transfer MAC/PAC:
- Increment Sequence Number across Sessions:
- Details:
 - Session initiation: Automatic
 - Data format: RJE
 - Parameter file: Not Required
 - Output file extension: out
 - Output path name: D:\BRANCHES\A05\51_0817162310_MT_GET*
 - Local transfer command: (empty)
 - Parameters: (empty)
- Run output session:
 - Number of messages = 1
 - Or at (hh.mm): (empty)

The screenshot shows the 'Application Interface - Message Partner MPA05FTMTOUT' window with the 'Emission' tab selected. The configuration includes:

- Parameters:
 - Exit Points:
 - Available: EPMAHESLOCAL, FileActAcks, FileActReceived, FileActReject, FileDeliveryNotifAck, FileDeliveryNotifNak, LocalSwiftNaks, MXDeliveryNotifAcks, MXDeliveryNotifNaks
 - Selected: EPA05FTMTOUT
 - Routing code transmitted:
 - Notification emission:
 - Send original message: Always

Direction: From Message Partner (to Alliance Access)

Format: XML v2

Application Interface - Message Partner MPA05FTMXIN

Profile | Session | Authentication | Reception

Message partner: Status:

Description:

Allowed direction: Connection method:

Details:

Session initiation:

Parameter file:

Format recognition:

Input path name:

Input attachment directory:

Application Interface - Message Partner MPA05FTMXIN

Profile | Session | Authentication | Reception

Parameters:

Validation level:

Profile name:

Message modification allowed:

Unit to be assigned:

Batch file validation:

Build unique file transfer reference

Routing:

Disposition: message in

7.2 Input MT Message sample in RJE format

```
{1:F01SPXAINJAA050001000001}{2:I103SPXAINJXXXXN}{3:{108:103-ACK-N-XXX03}}{4:
:20:103-ACK-N-XXX-03
:23B:CRED
:32A:110212USD1000,00
:33B:USD1000,00
:50A:/123456
SWHQBEBB
:57A:SWHQBEBB
:59A:SWHQBEBB
:71A:SHA
-}
```

7.3 Input MT Message sample in XML v2 format

```
_001000_ <?xml version="1.0" encoding="UTF-8"?>
<DataPDU xmlns="urn:swift:saa:xsd:saa.2.0">
  <Header>
    <Message>
      <SenderReference>MT103-b64-test2-feb10</SenderReference>
      <MessageIdentifier>fin.103</MessageIdentifier>
      <Format>MT</Format>
      <Sender>
        <BIC12>SPXAINJAA05</BIC12>
      </Sender>
      <Receiver>
        <BIC12>SPXAINJXXXX</BIC12>
      </Receiver>
      <InterfaceInfo>
        <UserReference>103-ACK-N-XXX-03</UserReference>
      </InterfaceInfo>
      <NetworkInfo>
        <Service>swift.fin!p</Service>
      </NetworkInfo>
      <SecurityInfo>
        <SWIFTNetSecurityInfo/>
      </SecurityInfo>
    </Message>
  </Header>

  <Body>DQo6MjA6MTAzLUFDSy1OLVhYWw0Mw0KOjIzQjpdUkVEDQo6MzJBOjExMDIxM1VTRDEwM
  DAsMDANCjozM0I6VVNEMTAwMCwwMA0KOjUwQTovMTIzNDU2DQpTV0hRQkVCQg0KOjU3QTpTV0hR
  QkVCQg0KOjU5QTpTV0hRQkVCQg0KOjcxQTpTSEENCi19</Body>
</DataPDU>
```

7.4 ACK Message with Original Message

```
{1:F21SPXAINJAA050031096790}{4:{177:1102101234}{451:0}}{108:103-ACK-N-
XXX03}}{1:F01SPXAINJAA050031096790}{2:I103SPXAINJXXXXN}{3:{108:103-ACK-N-
XXX03}}{4:
:20:103-ACK-N-XXX-03
:23B:CRED
:32A:110212USD1000,00
:33B:USD1000,00
:50A:/123456
SWHQBEBB
:57A:SWHQBEBB
:59A:SWHQBEBB
:71A:SHA
-}{5:{MAC:00000000}{CHK:60A595E74B4F}}
```

7.5 Delivery Notification Message

```
{1:F01SPXAINJJAA050031027068}{2:O0111104110210DYLXXXXX00003608811102101204S}{4:
{175:1204}{106:110210SPXAINJJAA050031096784}{108:10-103-ACK-
N2}{175:1204}{107:110210SPXAINJJAA050031027067}}{5:{CHK:AEB36522FC63}{SYS:}}{S:{COP
:P}}
```

7.6 Output MT Message sample in RJE format

```
{1:F01SPXAINJJAA050031027067}{2:O1031204110210SPXAINJJAA0500310967841102101204N}{3:
{108:10-103-ACK-N2}}{4:
:20:10-103-ACK-N2-01
:13C:/CLSTIME/0945+0100
:23B:CRED
:23E:CHQB
:26T:K90
:32A:091120USD15000,00
:33B:USD15000,00
:50A:/123456
SWHQBEBB
:52A:SWHQBEBB
:53B:/C/23456789
NEW YORK BRANCH
:54D:/87654321
RECEIVERS CORRESPONDENT
NEW YORK BRANCH
USA
:55D:/456789
THIRD REIMBURSEMENT INSTITUTION
:56A:/C/654321
SWHQBEBB
:57D:ACCOUNT WITH INSTITUTION
NEW YORK
USA
584214
:59A:SWHQBEBB
:70:/ROC/REF12365
:71A:OUR
:71G:USD60,00
:77B:/ORDERRES/US
-}{5:{MAC:00000000}{CHK:F8400FA850E8}}{S:{SAC:}{COP:P}}
```

7.7 NAK Message with Original Message

```
{1:F21SPXAINJJAA050031096785}{4:{177:1102101209}{451:1}{405:E46005}{108:10-103-NAK-
01}}{1:F01SPXAINJJAA050031096785}{2:I103SPXAINJXA05N}{3:{108:10-103-NAK-01}}{4:
:20:10-103-NAK-01
:13C:/RNCTIME/1240+0100
:23B:CRTS
:23E:TELI/Additional Information
:23E:TELI
:32A:091120USD15000,00
:33B:USD15100,00
:50K:/12345678
ORDERING CUSTOMER NAME
:52A:SWHQBEBB
:53D:/C/23456789
SENDERS CORRESPONDENT
HONG KONG BRANCH
HONG KONG
:54B:/C/987654321
HONG KONG BRANCH
:56A:/C/654321
SWHQBEBB
:57D:/654321
```

ACCOUNT WITH INSTITUTION
:59:BENEFICIARY CUSTOMER NAME
HONG KONG
HONG KONG
:70:/INV/070717, 12345, INVOICE
NUMBERS 24578, 54789,
354789, 545884, 578822
987664, 3211458, 56412636
:71A:BEN
:71F:USD100,00
:72:/ACC/INSTRUCTIONS ARE FOR
//ACCOUNT WITH INSTITUTION
//TO EXECUTE
:77B:/ORDERRES/US//ADDITIONAL INFO
//CONTINUATION OF ADDITIONAL INFO
-}{5:{MAC:00000000}{CHK:F815F08BA4D7}}

End of document