Swift Compatible Applications

Alliance Access Integration – Automated File Transfer

Technical Qualification Test 2026

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

Version 1

February 2026

Table of Contents

1 Preface 3
1.1 Purpose 3
1.2 Audience 3
1.3 Swift Compatible Application Programme 3
1.4 Related Documents 3

2 Introduction 4
2.1 Swift Compatible Application Validation 4

3 Alliance Access configuration 5
3.1 Alliance Access 5
3.2 Message Partner configuration 5
3.3 Exchanging messages using Message Partner 5

4 AFT Test Preparation 7
4.1 Reconciliation of Delivery Notification 7
4.2 Transmission Notification 7

5 Swift Compatible AFT testing 8
5.1 Test Evidence 8

6 Annexure 9
6.1 Input MT Message sample in RJE format 9
6.2 Input MT Message sample in XML v2 format 10
6.3 ACK Message with Original Message 10
6.4 Delivery Notification Message 11
6.5 Output MT Message sample in RJE format 11
6.6 NAK Message with Original Message 12
6.7 Input message in XML V2 for FileAct 13
6.8 Ack message in XML V2 for FileAct 14
6.9 OUT message in XML V2 for FileAct 15
6.10 Delivery Notification message in XML V2 for FileAct 18

1 Preface

1.1 Purpose

This document describes the test scenarios to qualify software integrating with Swift Alliance 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 partners considering the compliance validation of a product, and Swift Users that look after an overview of the Swift Compatible Application 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 Swift Compatible Application Programme

The Swift Compatible Application programme covers the entire financial application chain, from Trade, Treasury and Payment, to Corporate and Securities segments.

Each Swift Compatible Application 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

The following documents on Alliance Access 7.7 (or higher) can be found in the knowledge centre on swift.com:

- Alliance Access 7.7 release overview
- Alliance Access - Administration Guide
- Alliance Access - Configuration 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. AFT functionality is aligned with the way all other Message Partner works in Alliance Access.

2.1 Swift Compatible Application Validation

Support of AFT is one of the qualification criteria for granting the Swift Compatible Application label to a financial application.

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

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 Alliance Access

To prepare for the tests, the Alliance Access system must be installed and configured at Partner premises.

Alliance Access must be upgraded to release 7.7 or higher version. It is necessary to get acquainted with the Alliance Access - Administration Guide and Alliance 7.7 or higher release letters 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 the 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 partner 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 Admistration Guide – Alliance Access for additional information for configuring and managing Message Partner Profiles.

3.3 Exchanging messages using Message Partner

- AFT supports both MT and MX messages.MT test messages can be exchanged using RJE format and XML v2 format, while MX test messages can only be sent in XMLv2 format.
- 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 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 – 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 - Administration Guide – Message Formats Used in AI". A sample of the RJE and XMLv2 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 Swift compliance validation, at least one of the following file formats will be tested:

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

4.1 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 6.4

4.2 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 partner 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 6.3.

5 Swift Compatible 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.

1. The partner application provider prepares the number of MT / MX test messages generated by the application as required for the Swift Compatible 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.

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 access can read is XML v2 format.

4. For individual label requirements, the partner application must exchange application supported Swift Messages. The partner 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. 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 reconciled in the application. Evidences of reconciliation mechanism (screen dump, event log, dataset extract,) will be handled back to Swift Vaidation service provider.

5.1 Test Evidence

The Partner should 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
- Message Partner Configuration details
- 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.

6 Annexure

6.1 Input MT Message sample in RJE format

{1:F01SPXAINJJAXXX0001000455}{2:I569SPXAINJJXXXXN2}{3:{108:25-569-SR-P001}}{4:
:001/LAST
:22C::SEME//25-569-ACK-P026
:23G:CANC
:98C::PREP//22140222100000
:22F::STBA//EOSP
:22H::REPR//PROV
:16R:COLLPRTY
:95Q::CLPA//PARTY NAME CLT
:97A::SAFE//123456
:16S:COLLPRTY
:16R:LINK
:22C::PREV//569001
:16S:LINK
:16S:GENL
:16R:SUMM
:19A::COVA//USD10000000,
:19A::TEXA//USD9000000,
:19A::MARG//USD270000,
:19A::TCOR//USD11000000,
:92A::MARG//3,0
:25D::GCST//DEFI
:98A::VALN//22140222
:16S:SUMM
:16R:SUME
:22F::COLA//OPTN
:19A::COVA//USD10000000,
:19A::TEXA//USD9000000,
:19A::TCOR//USD11000000,
:92A::MARG//3,0
:25D::GETS//DEFI
:16R:SUMC
:13B::ELIG//1234PROF145
:95P::PTYB//SWHQBEBB
:19A::COVA//USD10000000,
:19A::TEXA//USD9000000,
:19A::TCOR//USD11000000,
:19A::MARG//USD270000,
:92A::MARG//3,0
:25D::GCOS//FLAT
:16R:TRANSDET
:22C::CLTR//CLN00102AB235
:98A::TERM//22222222
:98A::EXRQ//22222222
:19A::COVA//USD10000000,
:19A::TEXA//USD9000000,
:19A::TRAA//USD9000000,
:19A::TCOR//USD11000000,
:22F::MICO//A013
:92A::MARG//5,1
:16R:VALDET
:17B::COLL//Y
:17B::SECU//Y
:98A::SETT//22222222
:19A::MVPF//USD10000000,
:19A::LICO//USD9000000,
:19A::MVBF//USD9000000,

```
:92B::EXCH//USD/GBP/2,0
:92A::VAFC//2,
:22F::MICO//A013
:16R:SECDET
:35B:ISIN US0000123456
:36B::SECV//UNIT/10000,
:95L::ALTE//LEGALENTIDENTIFIER45
:95P::ACOW//SWHQBEBB
:97B::SAFE//DVPA/234-87-65767
:25D::SETT//PEND
:11A::DENO//USD
:90A::MRKT//PRCT/2,0
:94B::RATS//VEND
:70C::RATS//AAA+
:16S:SECDET
:16S:VALDET
:16S:TRANSDET
:16S:SUMC
:16S:SUME
-}
```

6.2 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>MT101-b64-test2-feb10</SenderReference>
      <MessageIdentifier>fin.101</MessageIdentifier>
      <Format>MT</Format>
      <Sender>
        <BIC12>SPXAINJJAA05</BIC12>
      </Sender>
      <Receiver>
        <BIC12>SPXAINJJXXXX</BIC12>
      </Receiver>
      <InterfaceInfo>
        <UserReference>101-ACK-N-XXX-01</UserReference>
      </InterfaceInfo>
      <NetworkInfo>
        <Service>Swift.fin!p</Service>
      </NetworkInfo>
      <SecurityInfo>
        <SWIFTNetSecurityInfo/>
      </SecurityInfo>
    </Message>
  </Header>

<Body>DQo6MjA6MTAzLUFDSy1OLVhYWC0wMw0KOjIzQjpDUkVEDQo6MzJBOjExMDIxMlVTRDEw
MDAsMDANCjozM0I6VVNEMTAwMCwwMA0KOjUwQTovMTIzNDU2DQpTV0hRQkVCQg0KOjU3QTp
TV0hRQkVCQg0KOjU5QTpTV0hRQkVCQg0KOjcxQTpTSEENCi19</Body>
</DataPDU>
```

6.3 ACK Message with Original Message

```
{1:F21SPXAINJJAXXX0062122997}{4:{177:1107130722}{451:0}{108:25-540-SR-
P001}}{1:F01SPXAINJJAXXX0062122997}{2:I540SPXAINJJXXXXN2}{3:{108:25-540-SR-P001}}{4:
:16R:GENL
:22C::SEME//25-540-SR-2222
```

:23G:NEWM
:16R:LINK
:22C::POOL//REFERENCE
:36B::TURN//FAMT/1000000,
:16S:LINK
:16S:GENL
:16R:TRADDET
:98A::SETT//22222122
:35B:ISIN US0000123456
ABCD
:16R:FIA
:94B::PLIS/XNYS/EXCH/NEWYORK STOCC EXCHANGE
:22F::MICO//A012
:98A::MATU//22222122
:16S:FIA
:22F::RPOR//DEFR
:16S:TRADDET
:16R:FIAC
:36B::SETT//FAMT/100000,
:97A::SAFE//123456
:16S:FIAC
:16R:SETDET
:22F::STCO//NPAR
:22F::SETR//TURN
:16R:SETPRTY
:95P::BUYR//SWHQBEBB
:16S:SETPRTY
:16R:SETPRTY
:95P::SELL//SWHQBEBB
:97A::SAFE//123456
:16S:SETPRTY
:16R:SETPRTY
:95P::DEAG//SWHQBEBB
:16S:SETPRTY
:16R:SETPRTY
:95C::PSET//US
:16S:SETPRTY
:16S:SETDET
-}{5:{MAC:00000000}{CHK:D37613267275}}

6.4 Delivery Notification Message

{1:F21SPXAINJJAXXX0062240262}{4:{177:1107131151}{451:0}}{1:F01SPXAINJJAXXX0062240262}{2:O0110621110713DYLRXXXXHXXX000041880111071307 21S}{4:{175:0722}{106:110713SPXAINJJAXXX0062122997}{108:25-540-SR-P001}{175:0722}{107:110713SPXAINJJAXXX0062240261}}{5:{CHK:70D184C5EACD}{SYS:}}{S:{COP:P}}

6.5 Output MT Message sample in RJE format

{1:F21SPXAINJJAXXX0062240261}{4:{177:1107131150}{451:0}}{1:F01SPXAINJJAXXX0062240261}{2:O5400722110713SPXAINJJAXXX0062122997 1107130722N}{3:{108:25-540-SR-P001}}{4:
:16R:GENL
:22C::SEME//25-540-SR-2222

:23G:NEWM
:16R:LINK
:22C::POOL//REFERENCE
:36B::TURN//FAMT/1000000,
:16S:LINK
:16S:GENL
:16R:TRADDET
:98A::SETT//22222122
:35B:ISIN US0000123456
ABCD
:16R:FIA
:94B::PLIS/XNYS/EXCH/NEWYORK STOCC EXCHANGE
:22F::MICO//A012
:98A::MATU//22222122
:16S:FIA
:22F::RPOR//DEFR
:16S:TRADDET
:16R:FIAC
:36B::SETT//FAMT/100000,
:97A::SAFE//123456
:16S:FIAC
:16R:SETDET
:22F::STCO//NPAR
:22F::SETR//TURN
:16R:SETPRTY
:95P::BUYR//SWHQBEBB
:16S:SETPRTY
:16R:SETPRTY
:95P::SELL//SWHQBEBB
:97A::SAFE//123456
:16S:SETPRTY
:16R:SETPRTY
:95P::DEAG//SWHQBEBB
:16S:SETPRTY
:16R:SETPRTY
:95C::PSET//US
:16S:SETPRTY
:16S:SETDET
-}{5:{MAC:00000000}{CHK:D37613267275}}{S:{SAC:}{COP:P}}

6.6 NAK Message with Original Message

{1:F21SPXAINJJAA050011096785}{4:{177:1102101229}{451:1}{405:E46005}{108:25-101-NAK-01}}{1:F01SPXAINJJAA050011096785}{2:I101SPXAINJJXA05N}{3:{108:25-101-NAK-01}}{4:
:22:10-101-NAK-01
:13C:/RNCTIME/1240+0100
:23B:CRTS
:23E:TELI/Additional Information
:23E:TELI
:32A:091123USD15000,00
: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//ADDITONAL INFO
//CONTINUATION OF ADDITIONAL INFO
-}{5:{MAC:00000000}{CHK:F815F08BA4D7}}

6.7 Input message in XML V2 for FileAct

_001154_____<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<Saa:DataPDU xmlns:Saa="urn:Swift:saa:xsd:saa.2.0" xmlns:Sw="urn:Swift:snl:ns.Sw"
xmlns:SwGbl="urn:Swift:snl:ns.SwGbl" xmlns:SwInt="urn:Swift:snl:ns.SwInt"
xmlns:SwSec="urn:Swift:snl:ns.SwSec">

<Saa:Revision>2.0.7</Saa:Revision>

<Saa:Header>

<Saa:Message>

<Saa:SenderReference>AFT_FA_test1</Saa:SenderReference>

<Saa:MessageIdentifier>camt.007.002.02</Saa:MessageIdentifier>

<Saa:Format>File</Saa:Format>

<Saa:Sender>

<Saa:DN>o=spxainjj,o=Swift</Saa:DN>

<Saa:FullName>

<Saa:X1>SPXAINJJXXX</Saa:X1>

</Saa:FullName>

</Saa:Sender>

<Saa:Receiver>

<Saa:DN>o=spxainjj,o=Swift</Saa:DN>

<Saa:FullName>

<Saa:X1>SPXAINJJXXX</Saa:X1>

</Saa:FullName>

```
    </Saa:Receiver>
    <Saa:InterfaceInfo>
      <Saa:UserReference>AFT_FA_test1</Saa:UserReference>
    </Saa:InterfaceInfo>
    <Saa:NetworkInfo>
      <Saa:Service>Swift.generic.fast!x</Saa:Service>
    </Saa:NetworkInfo>
   </Saa:Message>
  </Saa:Header>
  <Saa:Body>AFT_FA.JPG</Saa:Body>
</Saa:DataPDU>
```

6.8 Ack message in XML V2 for FileAct

```
_002298_____<?xml version="1.0" encoding="UTF-8" ?>
<Saa:DataPDU xmlns:Saa="urn:Swift:saa:xsd:saa.2.0" xmlns:Sw="urn:Swift:snl:ns.Sw"
xmlns:SwInt="urn:Swift:snl:ns.SwInt" xmlns:SwGbl="urn:Swift:snl:ns.SwGbl"
xmlns:SwSec="urn:Swift:snl:ns.SwSec">
<Saa:Revision>2.0.7</Saa:Revision>
<Saa:Header>
<Saa:TransmissionReport>
<Saa:SenderReference>AFT_FA_test1</Saa:SenderReference>
<Saa:ReconciliationInfo>SNL02213D11330601714009891C</Saa:ReconciliationInfo>
<Saa:NetworkDeliveryStatus>NetworkAcked</Saa:NetworkDeliveryStatus>
<Saa:OriginalInstanceAddressee>
<Saa:X1>SPXAINJJXXX</Saa:X1>
</Saa:OriginalInstanceAddressee>
<Saa:ReportingApplication>SWIFTNetInterface</Saa:ReportingApplication>
<Saa:NetworkInfo>
<Saa:Priority>Normal</Saa:Priority>
<Saa:IsPossibleDuplicate>false</Saa:IsPossibleDuplicate>
<Saa:Service>Swift.generic.fast!x</Saa:Service>
<Saa:Network>SWIFTNet</Saa:Network>
<Saa:SessionNr>000083</Saa:SessionNr>
<Saa:SeqNr>000000004</Saa:SeqNr>
<Saa:SWIFTNetNetworkInfo>
<Saa:RequestType>camt.007.002.02</Saa:RequestType>
<Saa:SNLRef>SNL02216-2222-01-01T12:08:36.8976.000007Z</Saa:SNLRef>
```

<Saa:Reference>5d8c33e0-5ae4-4509-b6dc-813a2eafa1ef</Saa:Reference>

<Saa:SnFInputTime>0140:2222-01-01T12:10:10</Saa:SnFInputTime>

<Saa:TransferRef>SNL02213D11330601714009891C</Saa:TransferRef>

<Saa:FileStartTime>22222101173833</Saa:FileStartTime>

<Saa:FileEndTime>22222101173858</Saa:FileEndTime>

</Saa:SWIFTNetNetworkInfo>

</Saa:NetworkInfo>

<Saa:Interventions>

<Saa:Intervention>

<Saa:IntvCategory>TransmissionReport</Saa:IntvCategory>

<Saa:CreationTime>22222101173833</Saa:CreationTime>

<Saa:OperatorOrigin>SYSTEM</Saa:OperatorOrigin>

<Saa:Contents>

<AckNack>

<PseudoAckNack>{1:F21SPXAINJJAXXX000083000000004}{4:{177:1221011738}{451:0}{311:ACK}{108:AFT_FA_test1}}</PseudoAckNack>

</AckNack>

</Saa:Contents>

</Saa:Intervention>

</Saa:Interventions>

<Saa:IsRelatedInstanceOriginal>true</Saa:IsRelatedInstanceOriginal>

<Saa:MessageCreator>ApplicationInterface</Saa:MessageCreator>

<Saa:IsMessageModified>false</Saa:IsMessageModified>

<Saa:MessageFields>NoOriginal</Saa:MessageFields>

</Saa:TransmissionReport>

</Saa:Header>

</Saa:DataPDU>

6.9 OUT message in XML V2 for FileAct

_004279_____<?xml version="1.0" encoding="UTF-8" ?>

<Saa:DataPDU xmlns:Saa="urn:Swift:saa:xsd:saa.2.0" xmlns:Sw="urn:Swift:snl:ns.Sw" xmlns:SwInt="urn:Swift:snl:ns.SwInt" xmlns:SwGbl="urn:Swift:snl:ns.SwGbl" xmlns:SwSec="urn:Swift:snl:ns.SwSec">

<Saa:Revision>2.0.7</Saa:Revision>

<Saa:Header>

<Saa:Message>

<Saa:SenderReference>OSPXAINJJXXX007AFT_FA_test1$122101</Saa:SenderReference>

<Saa:MessageIdentifier>camt.007.002.02</Saa:MessageIdentifier>
<Saa:Format>File</Saa:Format>
<Saa:SubFormat>Output</Saa:SubFormat>
<Saa:Sender>
<Saa:DN>o=spxainjj,o=Swift</Saa:DN>
<Saa:FullName>
<Saa:X1>SPXAINJJXXX</Saa:X1>
</Saa:FullName>
</Saa:Sender>
<Saa:Receiver>
<Saa:DN>o=spxainjj,o=Swift</Saa:DN>
<Saa:FullName>
<Saa:X1>SPXAINJJXXX</Saa:X1>
</Saa:FullName>
</Saa:Receiver>
<Saa:InterfaceInfo>
<Saa:UserReference>AFT_FA_test1</Saa:UserReference>
<Saa:MessageCreator>SWIFTNetInterface</Saa:MessageCreator>
<Saa:MessageContext>Original</Saa:MessageContext>
<Saa:MessageNature>Financial</Saa:MessageNature>
</Saa:InterfaceInfo>
<Saa:NetworkInfo>
<Saa:Priority>Normal</Saa:Priority>
<Saa:IsPossibleDuplicate>false</Saa:IsPossibleDuplicate>
<Saa:Service>Swift.generic.fast!x</Saa:Service>
<Saa:Network>SWIFTNet</Saa:Network>
<Saa:SessionNr>006100</Saa:SessionNr>
<Saa:SeqNr>000000116</Saa:SeqNr>
<Saa:SWIFTNetNetworkInfo>
<Saa:RequestType>camt.007.002.02</Saa:RequestType>
<Saa:SNLRef>SNL02222-2222-01-01T12:08:36.8976.000007Z</Saa:SNLRef>
<Saa:Reference>5d8c33e0-5ae4-4509-b6dc-813a2eafa1ef</Saa:Reference>
<Saa:SnFQueueName>spxainjj_file!x</Saa:SnFQueueName>
<Saa:SnFInputTime>0140:2222-01-01T12:10:10</Saa:SnFInputTime>
<Saa:SnFDeliveryTime>2222-01-01T12:10:10Z</Saa:SnFDeliveryTime>
<Saa:TransferRef>SNL02213D11330601738010042C</Saa:TransferRef>
<Saa:StoredTransferRef>SNL02213D11330601714009891S</Saa:StoredTransferRef>

<Saa:FileStartTime>22222101173858</Saa:FileStartTime>

<Saa:FileEndTime>22222101173926</Saa:FileEndTime>

</Saa:SWIFTNetNetworkInfo>

</Saa:NetworkInfo>

<Saa:SecurityInfo>

<Saa:SWIFTNetSecurityInfo>

<Saa:SignerDN>cn=fin1spxainjj,o=spxainjj,o=Swift</Saa:SignerDN>

<Saa:NRType>SvcOpt</Saa:NRType>

<Saa:SignatureResult>Success</Saa:SignatureResult>

<Saa:SignatureValue>

<SwInt:RequestPayload type="Swift.fileact.secsecureddata">

<Sw:FileRequestHeader>

<SwInt:Requestor>o=spxainjj,o=Swift</SwInt:Requestor>

<SwInt:Responder>o=spxainjj,o=Swift</SwInt:Responder>

<SwInt:Service>Swift.generic.fast!x</SwInt:Service>

<SwInt:RequestType>camt.007.002.02</SwInt:RequestType>

<SwInt:Priority>Normal</SwInt:Priority>

<SwInt:RequestRef>AFT_FA_test1</SwInt:RequestRef>

</Sw:FileRequestHeader>

<Sw:TransferRef>SNL02213D11330601714009891</Sw:TransferRef>

<Sw:Digest>

<Sw:DigestAlgorithm>SHA-1</Sw:DigestAlgorithm>

<Sw:DigestValue>5iSXh3GDOdNT4cUGaOFeMhO6/og=</Sw:DigestValue>

</Sw:Digest>

</SwInt:RequestPayload>

<SwSec:Crypto>

<SwSec:CryptoInternal>

<SwSec:CipherKey>UEVNRkBQcm9jLVR5cGU6IDQsTUlDLU9OTFkNCkNvbnRlbnQtRG9tYWluOiBSRkM4MjINCkVudHJ1c3RGaWxlLVZlcnNpb246IDIuMA0KT3JpZ2luYXRvci1ETjogY249ZmluMXNweGFpbmpqLG89c3B4YWluamosbz1zd2lmdA0KT3JpZy1TTjogMTMwNzEzNjk2Mw0KTUlDLUluZm86IFNIQTI1NiwgUlNBLA0KIFpqbmZHMWJLR3RIS25YcnI6RHBaQ0dySCtNc0pXbDNvWjNjVHRxWWdZUUljWXEwMVp5YmY2ZGhPSW8xSE1SSkcNCiBON3JicW10ckdYU01idFBGajJjZytLU2FrUFNxNytjUjhvV2J0SW00WjBrTGx5UVkwM2FNMVVPbHBwNEdJd3hUDQogamF4RmtTNWZvT0R4KzJjczNJcGE3QVBkL2tsTy9pcTNDSGYzeFNFdEp4REZLNGJ4WHNuRXFNT0tiVG12VzJOZw0KIDd5bnlyN3R2Wk0wazBsTzQ5aUZrUVlLUk1wTGVyS3FaYzZtYktoMDBBRENiMDU5TDVYV0I4VUFwREhlazJOaFkNCiBWaDNTeXF1NFR4UFU5bVNjckpaRGdhYS9BaTVMS2F6QzM0aEpjRmplYjlpVm9leE9BM25EWEkxQyt5OGlVWHpYDQogalZVZFp3a2tpc2NhWWcrYnhaaHV2dz09DQo=</SwSec:CipherKey>

<SwSec:CryptoProtocol>4.0:3.0</SwSec:CryptoProtocol>

</SwSec:CryptoInternal>

<SwSec:CryptoInfo>
<SwSec:MemberRef>RequestPayload</SwSec:MemberRef>
<SwSec:SignDN>cn=fin1spxainjj,o=spxainjj,o=Swift</SwSec:SignDN>
</SwSec:CryptoInfo>
</SwSec:Crypto>
</Saa:SignatureValue>
<Saa:FileDigestAlgorithm>SHA-1</Saa:FileDigestAlgorithm>
<Saa:FileDigestValue>5iSXh3GDOdNT4cUGaOFeMhO6/og=</Saa:FileDigestValue>
</Saa:SWIFTNetSecurityInfo>
</Saa:SecurityInfo>
<Saa:FileLogicalName>AFT_FA.JPG</Saa:FileLogicalName>
</Saa:Message>
</Saa:Header>
<Saa:Body>AFT_FA.JPG.SNL02213D11330601738010042C.ath</Saa:Body>
</Saa:DataPDU>

6.10 Delivery Notification message in XML V2 for FileAct

_004279_____<?xml version="1.0" encoding="UTF-8" ?>
<Saa:DataPDU xmlns:Saa="urn:Swift:saa:xsd:saa.2.0" xmlns:Sw="urn:Swift:snl:ns.Sw" xmlns:SwInt="urn:Swift:snl:ns.SwInt" xmlns:SwGbl="urn:Swift:snl:ns.SwGbl" xmlns:SwSec="urn:Swift:snl:ns.SwSec">
<Saa:Revision>2.0.7</Saa:Revision>
<Saa:Header>
<Saa:DeliveryNotification>
<Saa:ReconciliationInfo>SNL02213D11330601714009891C</Saa:ReconciliationInfo>
<Saa:ReceiverDeliveryStatus>RcvDelivered</Saa:ReceiverDeliveryStatus>
<Saa:MessageIdentifier>Delivery Notification</Saa:MessageIdentifier>
<Saa:InterfaceInfo>
<Saa:MessageCreator>SWIFTNetInterface</Saa:MessageCreator>
<Saa:MessageContext>Original</Saa:MessageContext>
<Saa:MessageNature>Network</Saa:MessageNature>
</Saa:InterfaceInfo>
<Saa:NetworkInfo>
<Saa:Priority>Normal</Saa:Priority>
<Saa:IsPossibleDuplicate>false</Saa:IsPossibleDuplicate>
<Saa:Network>SWIFTNet</Saa:Network>
<Saa:SessionNr>006079</Saa:SessionNr>
<Saa:SeqNr>000000557</Saa:SeqNr>

```
</Saa:NetworkInfo>
</Saa:DeliveryNotification>
</Saa:Header>
<Saa:Body>
<Sw:NotifySnFRequestHandle>
<Sw:SnFRef>SNL02213D113306017140098 91C</Sw:SnFRef>
<Sw:SnFRefType>FileAct</Sw:SnFRefType>
<Sw:AcceptStatus>Accepted</Sw:AcceptStatus>
<Sw:AckSwiftTime>2222-01-01T12:10:44Z</Sw:AckSwiftTime>
</Sw:NotifySnFRequestHandle>
</Saa:Body>
</Saa:DataPDU>
```

Legal Notices

Copyright
Swift ©2026. All rights reserved.

Restricted Distribution
Do not distribute this publication outside your organisation unless your subscription or order expressly grants you that right, in which case ensure you comply with any other applicable conditions.

Disclaimer
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 and binding version.

Trademarks
Swift is the trade name of S.W.I.F.T. SC. The following are registered trademarks of Swift: Swift, the Swift logo, 3SKey, Innotribe, MyStandards, Sibos, SWIFTNet, Swift Institute, the Standards Forum logo, Swift gpi with logo, the Swift gpi logo, and UETR. Other product, service, or company names in this publication are trade names, trademarks, or registered trademarks of their respective owners.

***** End of Document *****