

Messaging

SWIFTNet 6.3

Release Overview

Version 2.0

This document provides customers with an overview of the key features that SWIFT will introduce with SWIFTNet release 6.3.

30 September 2008



Legal notices

Copyright

Copyright © S.W.I.F.T. SCRL ("SWIFT"), avenue Adèle 1, B-1310 La Hulpe, Belgium, or its licensors, 2008. All rights reserved.

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

Confidentiality

This publication may contain SWIFT or third-party confidential information. Do not disclose this publication outside your organisation without the prior written consent of SWIFT.

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, S.W.I.F.T., the SWIFT logo, Sibos, SWIFTNet, SWIFTAlliance, SWIFTStandards, SWIFTReady, and Accord are trademarks of S.W.I.F.T. SCRL. Other SWIFT-derived service and product names, including SWIFTSolutions, SWIFTWatch, and SWIFTSupport, are tradenames of S.W.I.F.T. SCRL.

SWIFT is the trading name of S.W.I.F.T. SCRL.

Other product or company names in this publication are tradenames, trademarks, or registered trademarks of their respective owners.

Table of Contents

1	Introduction	5
2	Executive Summary	6
3	Key Release Dates	7
4	Expected Customer Impact	8
4.1	Customers.....	8
4.2	Interface and Application Providers	8
5	Detailed Release Contents and Expected Customer Impact	9
5.1	Changes for distributed architecture	9
5.1.1	Background.....	9
5.1.2	Changes introduced with this Release	9
5.1.3	Expected Impact	9
5.2	Increased HSM capacity and resilience.....	10
5.2.1	Background.....	10
5.2.2	Changes introduced with this release.....	10
5.2.3	Expected Impact	10
5.3	Traffic distribution to multiple SNLs	11
5.3.1	Background.....	11
5.3.2	Changes introduced with this release.....	11
5.3.3	Expected Impact	11
5.4	Enhanced access control for reroute function	12
5.4.1	Background.....	12
5.4.2	Changes introduced with this release.....	12
5.4.3	Expected Impact	12
5.5	Enhanced validation of FileAct header	13
5.5.1	Background.....	13
5.5.2	Changes introduced with this release.....	13
5.5.3	Expected Impact	13
5.6	Long term signature re-verification	14
5.6.1	Background.....	14
5.6.2	Changes introduced with this release.....	14
5.6.3	Expected Impact	14
5.7	Enhanced error text.....	15
5.7.1	Background.....	15
5.7.2	Changes introduced with this release.....	15
5.7.3	Expected Impact	15
5.8	Increased SNL server throughput	16
5.8.1	Background.....	16
5.8.2	Changes introduced with this release.....	16
5.8.3	Expected Impact	16

5.9	Enhanced service feature definition.....	17
5.9.1	Background.....	17
5.9.2	Changes introduced with this release.....	17
5.9.3	Expected Impact.....	17
5.10	Faster detection of session abort.....	18
5.10.1	Background.....	18
5.10.2	Changes introduced with this release.....	18
5.10.3	Expected Impact.....	18
5.11	Queue status report.....	19
5.11.1	Background.....	19
5.11.2	Changes introduced with this release.....	19
5.11.3	Expected Impact.....	19
5.12	Non-delivery warning.....	20
5.12.1	Background.....	20
5.12.2	Changes introduced with this release.....	20
5.12.3	Expected Impact.....	20
5.13	Increased receiver throughput.....	21
5.13.1	Background.....	21
5.13.2	Changes introduced with this release.....	21
5.13.3	Expected Impact.....	21
5.14	Increased flexibility for file delivery.....	22
5.14.1	Background.....	22
5.14.2	Changes introduced with this release.....	22
5.14.3	Expected Impact.....	22
5.15	Additional Browse user interface.....	23
5.15.1	Background.....	23
5.15.2	Changes introduced.....	23
5.15.3	Expected Impact.....	23
5.16	Updated usage rules for Browse.....	24
5.16.1	Background.....	24
5.16.2	Changes introduced.....	24
5.16.3	Expected Impact.....	24
5.17	Identifying a security officer with multiple certificates.....	25
5.17.1	Background.....	25
5.17.2	Changes introduced.....	25
5.17.3	Expected Impact.....	25

1 Introduction

This document provides customers with an overview of the key features that SWIFT will introduce with SWIFTNet 6.3.

Version 2.0 of this document provided the following key updates:

- Section 2 "Executive Summary" now contains additional entries for "Updated usage rules for Browse" and "Identifying a security officer with multiple certificates".
- Sections 5.16 "Updated usage rules for Browse" and 5.17 "Identifying a security officer with multiple certificates" have been added.
- All sections: The prefix "SWIFTNet" has been removed from "SWIFTNet InterAct", "SWIFTNet FileAct" and "SWIFTNet Browse" in line with the latest SWIFT naming and branding.

2 Executive Summary

SWIFTNet 6.3 is an optional minor SWIFTNet release. Table 1 provides an overview of the new messaging features or changes that are part of this release, which SWIFT expects to make available for live operations by the end of 1Q 2009.

Table 1: Summary of changes introduced with SWIFTNet 6.3

Description of changes	Messaging service			
	FIN	InterAct	FileAct	Browse
1. Changes for distributed architecture	✓	✓	✓	
2. Increased HSM capacity	✓	✓	✓	
3. Traffic distribution to multiple SNLs		✓	✓	
4. Enhanced access control for reroute function		✓	✓	
5. Enhanced validation of FileAct header			✓	
6. Long term signature verification		✓	✓	
7. Enhanced error text		✓	✓	
8. Increased SNL server throughput		✓	✓	
9. Enhanced service feature definition		✓	✓	
10. Faster detection of session abort		✓	✓	
11. Queue status report		✓	✓	
12. Non-delivery warning		✓	✓	
13. Increased receiver throughput		✓	✓	
14. Increased flexibility for file delivery			✓	
15. Additional Browse GUI framework				✓
16. Updated usage rules for Browse				✓
17. Identifying a security officer with multiple certificates		✓	✓	

As a precautionary measure, SWIFT recommends that developers and service providers conduct appropriate regression tests with SWIFTNet 6.3 before the release becomes available for live operations.

Customers who do not want to take advantage of the new optional features introduced by this release do not need to implement the corresponding interface release(s).

Note – Implementing SWIFTNet 6.3 is one of the options that customers have to acquire the changes for distributed architecture. For more information and an overview of the alternatives to implement the changes for distributed architecture, see knowledge base tip [#2160676](#) on www.swift.com.

Alliance customers wanting to use the new features will need release 6.3 of the Alliance Portfolio. For more information about the Alliance releases, see the *Alliance Release Overview*, available on www.swift.com.

For more information about the overall SWIFTNet release policy, see the *SWIFTNet and Alliance Release Policy*, which is available on www.swift.com.

3 Key Release Dates

Table 2 provides the key target release dates for SWIFTNet 6.3.

Table 2: Key target dates for SWIFTNet 6.3

Event	Target date / period	Description
Release Overview Vendors specifications	End of June 2008	Availability of the preliminary versions of the release overview and the developer's documentation
Final Release Overview Final Vendors specifications	End of September 2008	Availability of final release overview and the developers' documentation
Availability for live operations	End of March 2009	Full availability of SWIFTNet 6.3 on ITB for developer testing and on Production environment for live operations. Availability of the final version of the SWIFTNet communication software (SWIFTNet Link and Alliance Gateway software).

4 Expected Customer Impact

SWIFTNet release 6.3 has been designed and tested to ensure backward compatibility with customer systems working with SWIFTNet release 6.0 or 6.1.

SWIFTNet 6.3 may impact SWIFT users, service providers, and interfaces and application providers that decide to migrate to this optional release.

Note that customers who use FileAct and who do not yet comply with the rules for request type format, may be impacted by the item "Enhanced header validation". See section 5.5 for more details.

4.1 Customers

The potential customer impact of SWIFTNet release 6.3 should be considered separately for:

- Customers not wishing to take advantage of the new optional features introduced by this new release.

These customers can continue to use their current interface releases, as long as they remain supported by SWIFT. Note that these customers need to select one of the other options to acquire the changes for distributed architecture (see the note on page 9).

As a precautionary measure, SWIFT recommends that service administrators, vendors, and developers of in-house systems that support such customers conduct appropriate regression tests with release 6.3 of SWIFTNet on the developer's test environment, known as the Integration Testbed (ITB). SWIFT recommends that these tests involve representative traffic volumes and that they are performed as early as possible after the introduction of SWIFTNet release 6.3. This allows for increased reaction time in the unlikely event that problems are encountered.

- Customers wishing to take advantage of the new optional features introduced by this new release.

In addition to the precautionary measures above, service administrators, vendors, and developers of in-house systems that support such customers will need to adapt their existing systems to the new functionality and, where appropriate, conduct the appropriate integration testing.

Note that most of the new messaging features associated with SWIFTNet 6.3 are only available when the customer uses an application or interface that was upgraded for SWIFTNet 6.3.

For more information about the SWIFT interface releases, see the *Alliance Release Overview*, available on www.swift.com.

4.2 Interface and Application Providers

SWIFT also encourages interface and application providers to support the new, optional features that SWIFT is introducing with SWIFTNet 6.3.

5 Detailed Release Contents and Expected Customer Impact

This section outlines the changes that SWIFT will introduce with SWIFTNet 6.3.

5.1 Changes for distributed architecture

5.1.1 Background

As announced on swift.com, the SWIFT Board of Directors has approved the implementation of a distributed architecture for SWIFT's messaging services. Distributed architecture will partition messaging into two zones, the European messaging zone and the Trans-Atlantic messaging zone, with pairs of Operating Centres that store the traffic for each zone. The first deliverables include an enhanced core messaging platform to support multiple processing zones, a new Operating Centre (OPC) to be located in Switzerland, and a command and control capability in Hong Kong.

The new distributed architecture will improve resilience, add capacity, control long-term average message costs, and alleviate European data protection concerns.

5.1.2 Changes introduced with this Release

The SWIFTNet Link version that comes with SWIFTNet 6.3 includes the changes required for this distributed architecture. These changes relate to connection paths between the SWIFTNet Link and the SWIFT Operating Centres and are transparent to users of the SWIFTNet Link.

Customers who implement this SWIFTNet Link release will not need to install anything else related to distributed architecture.

Note

Customers who do not need the new functionality of SWIFTNet 6.3, and only want to install the changes for distributed architecture, can choose to just install a dedicated SWIFTNet Link patch instead. For more information and an overview of the alternatives to implement the changes for distributed architecture, see knowledge base tip [#2160676](#) on www.swift.com.

5.1.3 Expected Impact

Type of impact	Traffic will be stored and routed based on the principles of the distributed architecture. SWIFT does not expect users or services to be impacted.
Customers impacted	Customers that install the SWIFTNet Link version that comes with SWIFTNet 6.3.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible SWIFTNet Link version, and after SWIFT has deployed SWIFTNet 6.3 and centrally activated the routing changes for distributed architecture.

5.2 Increased HSM capacity and resilience

5.2.1 Background

Since SWIFTNet 6.0, SWIFTNet provides the capability to store PKI certificates on a Hardware Security Module (HSM). SWIFT supports several models, depending on the throughput and certificate capacity needs of the customers. These models include the HSM Box, the HSM Token and the HSM Card. The HSM Box currently can store up to 250 certificates. While each HSM Token (or HSM Card) contains a single certificate, the SWIFTNet Link currently allows connecting up to 4 HSM Tokens (or 4 HSM Cards) concurrently.

5.2.2 Changes introduced with this release

SWIFT introduces the following HSM enhancements as of SWIFTNet 6.3:

- Increased HSM certificate capacity. This is especially useful for customers with a large number of certificates, as all certificates used for live InterAct or FileAct traffic will need to be stored on HSM by end October 2009. This includes:
 - Increased certificate capacity for HSM Box. Customers who select this new option will be able to increase their HSM Box capacity from 250 certificates to at least 1,000 certificates. To implement this feature, customers do not need to replace the HSM Box itself. They can configure their HSM Box for increased capacity through a software upgrade, which preserves the certificates that the HSM Box already contains.
 - Increased limit of HSM Tokens or HSM Cards per SWIFTNet Link. Customers will be able to connect up to 10 HSM Tokens or 10 HSM Cards simultaneously per SWIFTNet Link. This change does not require any configuration.
- Network connection resilience on the HSM Box. The HSM Box will support dual network interfaces in active/standby mode. In case of network connection failure, the HSM Box will switch from the active network interface to the standby. To implement this feature, customers will need to configure their HSM Box through a software upgrade.

5.2.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of these features. Customers that want to take advantage of these features will need to install the new SWIFTNet Link version and, for the HSM Box changes, install the HSM Box software and configure it to support these new features. They also may need to adapt or configure their business applications or interfaces.
Customers impacted	Customers that use FIN, InterAct or FileAct and want to use these new features.
Start of impact	When the customer has installed the SWIFTNet Link version that comes with SWIFTNet 6.3, and the compatible interface software.

5.3 Traffic distribution to multiple SNLs

5.3.1 Background

Today, customers can define on which SWIFTNet Link (SNL) they receive real-time traffic. They do this by defining routing rules that are maintained on SWIFT's central systems. Each routing rule relates to a specific traffic flow (for example, traffic for one service) and defines a SWIFTNet Link endpoint on which SWIFT delivers that traffic.

Optionally, customers can define alternative SWIFTNet Link endpoints for the same rule. Customers can switch the current routing to such an alternate endpoint by executing the "reroute" command.

5.3.2 Changes introduced with this release

SWIFTNet 6.3 introduces the option to receive real-time traffic for the same traffic flow on several SWIFTNet Links in parallel. This is particularly useful for customers who have several systems that receive traffic and are operational at the same time, as such a setup provides enhanced resilience as well as increased throughput.

When customers use this option for a given traffic flow, SWIFT will distribute that traffic in a random manner to the defined SWIFTNet Links that are active at that moment. In this way, the traffic is (roughly) equally distributed amongst these SWIFTNet Links. Note that for FileAct traffic, a single file transfer (once started) is not split across different SNLs.

If one of the SWIFTNet Links becomes unavailable, SWIFT will notice this and will no longer distribute traffic to that SWIFTNet Link. SWIFT will continue to deliver traffic to the remaining SWIFTNet Links (if any) for that traffic flow. When the failed SWIFTNet Link becomes available again, SWIFT will include that SWIFTNet Link again in the traffic distribution. This happens automatically without the need for any customer or SWIFT intervention.

Customers can define their routing rules in a new way, to allow such traffic distribution. As of SWIFTNet 6.3, routing rules can share the same key fields (service, sender, receiver, request type) and have different SWIFTNet Link endpoints per rule. SWIFT's central routing function will automatically distribute traffic if there are two or more rules that have the same key fields.

Customers can now also enable or disable such routing rules themselves on-line. This enables customers to manage which SWIFTNet Links can participate in such a scheme at which point in time. Customers can use this, for example, to temporarily take out a system easily (for maintenance or upgrades), or to gradually introduce this new option.

This feature applies to traffic received in real-time only.

5.3.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces, for example to group the traffic together again towards the backoffice application(s).
Customers impacted	Customers that use InterAct or FileAct and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has made the feature available and the customer has updated their routing rules.

5.4 Enhanced access control for reroute function

5.4.1 Background

Today, SWIFTNet provides the ability to reroute real-time traffic between two or more SWIFTNet Links for all traffic, or for a specific traffic flow (for example, the traffic of one service). This allows a customer to switch traffic delivery from one system to another without the need to contact SWIFT. Similarly, SWIFT provides a function to query the routing rules and to display for each routing rule to which SWIFTNet Link SWIFT will deliver real-time traffic.

Customers can access these functions, for live or pilot traffic, if they have the role *SiteManager*.

5.4.2 Changes introduced with this release

SWIFTNet 6.3 introduces the capability to optionally restrict these functions to only live or only pilot traffic. Customers can use this option to avoid that teams who perform application testing and only need to access the pilot services, would inadvertently reroute traffic in the live environment, for example.

This enhanced access control is available through two new RBAC roles, *LiveSiteManager* and *PilotSiteManager*. Customers who want to use this finer access control can delegate these roles instead of the *SiteManager* role.

The use of these new roles is optional. Customers who prefer to continue to use the current *SiteManager* role can do so.

5.4.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature will need to delegate the new role(s) (and ungrant the existing <i>SiteManager</i> role, if applicable).
Customers impacted	Customers that use InterAct or FileAct and want to use this new feature.
Start of impact	When SWIFT has deployed SWIFTNet 6.3.

5.5 Enhanced validation of FileAct header

5.5.1 Background

With SWIFTNet 6.0, SWIFT introduced standardisation of the format of the request type field for FileAct. In addition, with SWIFTNet 6.1, services can use a new FileAct header field (HeaderInfo) to contain key summary information related to the file.

SWIFT has also provided rules on how to format these fields. Currently SWIFT centrally validates adherence to these rules using minimal validation checks.

5.5.2 Changes introduced with this release

As of SWIFTNet 6.3 release date, SWIFT will perform stronger central validation on the request type and HeaderInfo fields, for FileAct. In particular:

- Request Type

SWIFT will now perform full validation. This means, SWIFT will check that the business area (the first four characters) is a valid business area (for a list of all business areas, see the Annex A of the SWIFTNet Messaging Operations Guide). Also, the request type value must be in lowercase, use allowed characters, and the usage of the dot-separator must follow the rules as outlined in the Operations Guide. SWIFT will reject files that are not compliant.
- HeaderInfo

SWIFT will perform central validation of the HeaderInfo contents, both in terms of syntax and semantic validation. SWIFT will reject files with HeaderInfo contents that do not pass this validation, or that do not use the HeaderInfo field according to the rules defined for the service.

Customers should note that there is no change in rules and that therefore, customers who adhere to the rules already should not be impacted.

As of October 2008, SWIFT plans to proactively monitor the compliance of existing FileAct users with these new rules, and contact them if needed. All customers must adhere to the rules at the latest by the 2009 standards release (21 Nov 2009), after that date SWIFT will reject traffic that is not compliant.

Customers do not need to install the new SWIFTNet Link release to benefit from this enhancement, because it is managed at SWIFT's central systems. Therefore, it is applicable to all FileAct users, regardless of the version of their SWIFTNet Link.

5.5.3 Expected Impact

Type of impact	Files that do not use a valid Request Type or that do not pass the HeaderInfo validation, will be rejected by SWIFT.
Customers impacted	All customers using FileAct.
Start of impact	When SWIFT has deployed SWIFTNet 6.3 and has activated the enhanced validation.

5.6 Long term signature re-verification

5.6.1 Background

SWIFTNet currently provides a non-repudiation service to customers that have exchanged InterAct or FileAct traffic using the non-repudiation option. When this option is used, SWIFT stores the necessary data related to that traffic to enable after-the-fact signature re-verification. SWIFT can also provide additional related information, such as timestamps. Customers using the non-repudiation option can request this re-verification service from SWIFT during 124 days after the initial transaction took place.

Customers can use this service, for example, to obtain evidence in support of a dispute.

5.6.2 Changes introduced with this release

SWIFTNet 6.3 introduces a new option to allow after-the-fact signature re-verification during a longer period.

In this case, SWIFT does not store any data related to the traffic. Customers who want to re-verify a signature, need to provide to SWIFT both the traffic data and the signature-related information. SWIFT will retrieve certificate data from its systems, to check if the certificate was valid at the time of the transaction, and to re-verify the signature.

SWIFT provides this service up to 13 years after the transaction took place. It is available to all users that have exchanged signed traffic. Note however, that SWIFT does not store any data related to the transaction, and can thus not provide any related information, such as, the date and time of the transaction.

This new option will be available in addition to the current non-repudiation offering.

5.6.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces, so that the required information is safestored, and can be extracted when it needs to be presented to SWIFT.
Customers impacted	Customers that exchange InterAct or FileAct traffic and want to use this new feature.
Start of impact	After SWIFT has deployed SWIFTNet 6.3.

5.7 Enhanced error text

5.7.1 Background

Today, SWIFTNet provides different types of information in case of an error. This includes a severity rating (fatal, transient, etc) and a textual explanation about the error. When possible, SWIFT also provides advice on how to potentially solve the problem.

In some cases however, SWIFT does not report the severity precisely, or uses an error text that is not sufficiently clear. Sometimes, the error text is repeated several times with the same or a similar text.

5.7.2 Changes introduced with this release

For SWIFTNet 6.3, SWIFT has evaluated the most common errors. Where necessary, SWIFT has enhanced (and simplified) the error text or severity.

To ensure backwards compatibility, SWIFT does not provide the new error text by default. Therefore, application developers need to explicitly select the new error reporting to benefit from this enhancement. SWIFT expects that in a future release, this new capability will become the default mode.

Customers will see the new, simplified error text when they use applications that select the new error reporting mode and that show the SWIFTNet error text to customers.

5.7.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature will need to adapt their business applications or interfaces.
Customers impacted	Customers that use InterAct or FileAct and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has deployed SWIFTNet 6.3.

5.8 Increased SNL server throughput

5.8.1 Background

The SWIFTNet Link of a large user or service provider can potentially receive many incoming messages or files in real-time at (or around) the same time. The ability to successfully treat a large number of incoming messages or files depends on the time that the application needs to respond to the incoming request, and the number of outstanding requests that SWIFTNet Link can handle. In some cases, the latter was reported as being a throughput limitation.

5.8.2 Changes introduced with this release

The SWIFTNet Link version that comes with SWIFTNet 6.3 has been enhanced to be able to process a higher number of incoming requests concurrently. The number of concurrent outstanding requests (incoming messages/file awaiting an answer from the application) can now be up to 400. Of course, the application must be able to respond quickly to these requests to benefit fully from this enhancement.

5.8.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces.
Customers impacted	Customers that use InterAct and FileAct and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has deployed SWIFTNet 6.3.

5.9 Enhanced service feature definition

5.9.1 Background

Today, service administrators define the characteristics of their service through the Service Profile Form. They use this form to provide all the information that SWIFT requires in order to define and setup (or modify) their service(s). This can include the definition of optional features, such as the use of the non-repudiation option or the copy option. Today, such features can be "mandatory", "optional" or "not supported" for the service; this definition applies to all traffic exchanged in the service.

5.9.2 Changes introduced with this release

SWIFTNet 6.3 introduces the ability to define service features at the level of the Request Type.

This allows service administrators to define the list of Request Types for which the feature is mandatory, for which the feature is optional and for which the feature does not apply. Some of these lists can be empty.

A service administrator can decide, for example, that the non-repudiation option is mandatory for orders and instructions, while leaving it optional for all other types of traffic.

5.9.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature.
Customers impacted	SWIFT does not expect any impact for users or services that do not take advantage of this feature.
Start of impact	After SWIFT has deployed SWIFTNet 6.3.

5.10 Faster detection of session abort

5.10.1 Background

When customers use services that operate in store-and-forward mode, SWIFT will attempt to deliver traffic to the intended recipient. This happens in the context of a delivery session. In some cases, SWIFT may abort a delivery session (for example if there are too many errors occurring). In case of a session abort, the receiver's SWIFTNet Link becomes aware of this status after around 10 minutes on average. Each receiver's application can interrogate the status of the delivery session at any time by calling the appropriate API.

5.10.2 Changes introduced with this release

SWIFTNet 6.3 enables the receiver's application to detect session aborts faster. This is achieved through:

- a more frequent update of the SWIFTNet Link session status
- the availability of a new SWIFTNet Link function that can inform the application of a change in session state. When the application has subscribed to this new function, it will be informed as soon as the SWIFTNet Link status has changed.

These features allow the development of applications that become aware of session abort events more quickly, and react accordingly (for example, notify an operator and/or automatically re-open the queue).

5.10.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this new API function. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces.
Customers impacted	Customers that use InterAct or FileAct in store-and-forward mode and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has deployed SWIFTNet 6.3.

5.11 Queue status report

5.11.1 Background

For customers that send traffic using store-and-forward services, SWIFTNet already provides a report that shows an overview of traffic that remains undelivered at a certain time. However, SWIFT does not yet provide receivers with an overview of traffic that is pending in their queues waiting to be delivered.

5.11.2 Changes introduced with this release

SWIFTNet 6.3 introduces the Queue Status Report. This new feature allows a user to send a request to SWIFT to get a report of traffic that is pending in their queue(s). SWIFT will process this request, retrieve the necessary information and respond by sending the queue status report. To ensure that the report correctly reflects the queue status at the time of the request, it is possible to request that SWIFT delivers this report before any other pending traffic.

These exchanges are in the form of system messages. SWIFT describes the technical details in the SWIFTNet Messaging Services - Interfaces Vendor Specifications for InterAct and FileAct.

5.11.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces.
Customers impacted	Customers that use InterAct or FileAct in store-and-forward mode and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has deployed SWIFTNet 6.3.

5.12 Non-delivery warning

5.12.1 Background

For customers that send traffic using store-and-forward services, SWIFTNet already provides a number of delivery monitoring options. Customers can request a report that shows an overview of traffic that remains undelivered at a certain time. In addition, they can optionally flag each file or message sent with the delivery notification option, in which case SWIFT will send them an explicit notification when the file or message has been delivered.

5.12.2 Changes introduced with this release

SWIFTNet 6.3 introduces an additional, optional delivery monitoring feature, the non-delivery warning. When a customer selects this option when sending a message or file, they also indicate the period after which they require notification if the traffic is not yet delivered. SWIFT will then try to deliver the message or file. If the delivery happens within the indicated timeframe, SWIFT will not notify the sender. However, if the delivery has not yet happened by that time, SWIFT will send a non-delivery warning to the sender. In that case, SWIFT will still (try to) deliver the message or file.

This allows the sender to know that the receiver has not received the message or file in the expected timeframe, and the sender can decide, for example, to contact their correspondent to take appropriate actions.

The "delivery notification" option and the "non-delivery warning" can be used independently and in any combination. They are not linked to the priority used (unlike FIN).

The non-delivery warning is delivered in a system message. SWIFT describes the technical details in the SWIFTNet Messaging Services - Interfaces Vendor Specifications for InterAct and FileAct.

5.12.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces.
Customers impacted	Customers that use InterAct or FileAct in store-and-forward mode and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has deployed SWIFTNet 6.3.

5.13 Increased receiver throughput

5.13.1 Background

SWIFT uses a "window size" when delivering store-and-forward traffic during an output session. This "window size" defines the number of messages or files that are outstanding, this means, that are in the process of being delivered but are not yet acknowledged to SWIFT. SWIFT defines a default window size for each new queue. This default is currently set at 10.

5.13.2 Changes introduced with this release

SWIFTNet 6.3 introduces the ability to increase the window size for customers that need to achieve higher throughput. Customers can request to SWIFT a window size limit that is higher than the default. When their interface opens the queue and asks for a window size that is higher than the default, it will be granted this higher window size up to the provided limit.

SWIFT expects that most customers will not need to request a higher window size. However, customers with high traffic volumes and a performant system may need higher throughput and can request to SWIFT a window size increase. SWIFT will make available a dedicated form to request this.

5.13.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces.
Customers impacted	Customers that use InterAct or FileAct in store-and-forward mode and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has deployed SWIFTNet 6.3.

5.14 Increased flexibility for file delivery

5.14.1 Background

Customers who participate to services that use FileAct in store-and-forward mode, need to fetch files when they receive the file notification from their queue. Usually, the FileAct interface handles this automatically. If the interface does not fetch the files, then the delivery window will be blocked because an acknowledgement can only be sent to SWIFT once the file has been fully delivered.

5.14.2 Changes introduced with this release

SWIFTNet 6.3 introduces the option to receive FileAct file notifications without having to fetch the corresponding file immediately. Applications that support this optional mode, can thus receive all file notifications that are on the queue without fetching the corresponding files. In that case, the delivery window will not be blocked by undelivered file transfers. This allows receivers to fetch the files at a later stage, or in a different order than the arrival order of the file notifications.

Note that these applications are expected to fetch the file at some point, otherwise the files will expire and be considered undelivered (and the sender will receive a non-delivery notification).

5.14.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users or services that do not take advantage of this feature. Customers that want to take advantage of this feature may need to adapt or configure their business applications or interfaces.
Customers impacted	Customers that use FileAct and want to use this new feature.
Start of impact	When the customer has installed the SWIFTNet 6.3 compatible interface software, and after SWIFT has deployed SWIFTNet 6.3.

5.15 Additional Browse user interface

5.15.1 Background

Customers currently need the Alliance WebStation to access a service provider's Browse application over SWIFTNet. The Alliance WebStation provides the user interface functionality for Browse. This is the ability to browse the web site of the service provider over the SWIFT network as well as to perform InterAct and/or FileAct exchanges.

5.15.2 Changes introduced

As an alternative to the Alliance WebStation, SWIFT plans to introduce as of May 2009 a new lightweight implementation of the user interface to access Browse-based services on SWIFTNet. SWIFT will offer this new user interface as a part of the Alliance Web Platform, a component in the Alliance family designed to provide the users with a unified and easy-deployable user interface solution.

As of SWIFTNet 6.3, customers will be able to access Browse services using either the current Alliance WebStation or using the Browse capability on the Alliance Web Platform.

Service providers of Browse services will have to update their application, to support the access to their Browse service using Alliance Web Platform. These changes are fully backwards compatible and transparent to Alliance WebStation users of the Browse service(s).

Once the Alliance Web Platform with the Browse user interface is made available in 2009, the participants of the Browse services may request their service providers to enable their service(s) for the new user interface. Customers will be able to gradually use this new Browse user interface as their service provider(s) declare to be ready to support it.

SWIFT plans to mandate that service providers support the Browse GUI of the Alliance Web Platform by SWIFTNet 7.0.

5.15.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users of Browse services . Service providers will need to adapt their application.
Customers impacted	Customers/service providers that use/provide Browse and want to use this new Browse GUI.
Start of impact	When the service provider has installed these changes and declares the service to be ready.

5.16 Updated usage rules for Browse

5.16.1 Background

Currently SWIFT mandates the usage of the `swlogon` InterAct message to start a Browse session. This message allows to authenticate the end-user to the Browse application. It is usually transparent to users of the Browse application.

SWIFT also currently mandates that Browse service providers configure their Browse service to require mutual authentication. This configuration is also known as two-way Secure Sockets Layer (SSL). This implies that each user installs a web browser certificate and the browse service provider installs a web server certificate.

5.16.2 Changes introduced

SWIFT now supports that service providers use other authentication methods than the ones SWIFT has prescribed so far. This implies the following:

- The `swlogon` InterAct message is now optional

Service providers who have an alternative authentication mechanism can use that mechanism without the need to implement in addition the `swlogon` message. For example, a service provider who already has a web application that authenticates its users with a username and password, may decide to continue using that mechanism for Browse.

The `swlogon` InterAct message remains available and SWIFT recommends this solution to service providers who do not have another authentication mechanism.

Note that it remains the responsibility of the service provider to ensure a strong link between their authentication method and the security for the rest of the Browse session.

- The use of two-way SSL is now optional.

SWIFT no longer requires that service providers configure their Browse service for client authentication using web browser certificates. This will also simplify the configuration of the workstations that customers use to access Browse, once all their service providers have updated their configuration. Therefore SWIFT recommends that service providers configure their Browse applications not to require web browser certificates. SWIFT may decide to remove support for web browser certificates at some point in the future, with sufficient advance notice. Note that the use of SSL with web server certificates remains.

SWIFT has analysed possible security implications related to these enhancements. SWIFT has concluded that these changes can be introduced without affecting the level of security for using Browse on SWIFTNet.

These changes are in effect as of September 2008. The related documentation will be updated later, in line with the update cycle for SWIFTNet 6.3.

5.16.3 Expected Impact

Type of impact	Service providers may design their application to not use <code>swlogon</code> for authentication purpose. Service providers may configure their web server to not require web browser certificates. Customers may in future no longer need to install web browser certificates, if all the Browse service providers they use no longer require two-way SSL.
Customers impacted	Customers and providers of Browse services.
Start of impact	When the service provider has installed and communicated these changes.

5.17 Identifying a security officer with multiple certificates

5.17.1 Background

To identify himself or herself, a Security Officer can have multiple certificates that are typically stored in different HSM boxes for resilience purposes. In this case, SWIFT recommends Security Officers to use equivalent DNs. Distinguished Name (DN) equivalence is a naming scheme used to differentiate multiple DNs that identify the same entity. Equivalent DNs only differ by a numbered common name (%n) in the lowest segment. For example, the following three DNs are all equivalent:

```
cn=SO1,o=<BIC8>,o=swift;
cn=%1,cn=SO1,o=<BIC8>,o=swift;
cn=%2,cn=SO1,o=<BIC8>,o=swift.
```

For Security Officers using dual-authorisation, SWIFTNet ensures that a 4eyes request cannot be approved by a Security Officer whose certificate is equivalent to the certificate of the Security Officer who issued the request.

Currently, SWIFTNet:

- only allows Shared Security Officers with a DN that has maximum three segments (such as `cn=SO1,o=<BIC8>,o=swift`).
- determines the scope of authority of a Security Officer as including its parent node and all nodes below, Therefore Security Officers with an equivalent DN such as `cn=%1,cn=SO1,o=<BIC8>,o=swift` have their scope of authority limited to `cn=SO1,o=<BIC8>,o=swift`.

5.17.2 Changes introduced

SWIFTNet 6.3 better supports the use of equivalent Distinguished Names (DN) to identify a Security Officer with multiple certificates.

This includes the following enhancements:

- SWIFTNet 6.3 will allow to define a Shared Security Officer with a DN that has maximum four segments provided the DN is an equivalent DN such as `cn=%<n>,cn=SO1,o=<BIC8>,o=swift`. Otherwise, the maximum three segments rule is still applicable.
- SWIFTNet 6.3 will determine the scope of authority of a Security Officer with an equivalent DN from its parent node as starting point. For example, the following three equivalent DNs have the same scope of authority that includes everything under `o=<BIC8>,o=swift`:

```
cn=SO1,o=<BIC8>,o=swift;
cn=%1,cn=SO1,o=<BIC8>,o=swift;
cn=%2,cn=SO1,o=<BIC8>,o=swift.
```

5.17.3 Expected Impact

Type of impact	SWIFT does not expect any impact for users that do not take advantage of this feature.
Customers impacted	All Security Officers with multiple certificates based on equivalent DNs.
Start of impact	When SWIFT has deployed SWIFTNet 6.3. Customers may also need to upgrade their Security Officers' interface before they can take full advantages of these changes.