

SWIFT CERTIFICATION FOR NCS PRODUCT

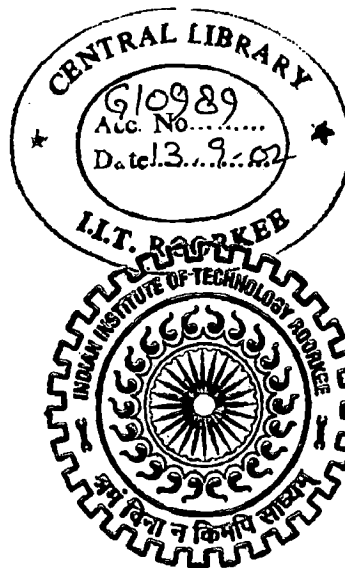
A DISSERTATION

*Submitted in partial fulfilment of the
requirements for the award of the degree
of*

MASTER OF COMPUTER APPLICATIONS

By

RAHUL



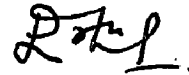
DEPARTMENT OF MATHEMATICS
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE-247 667 (INDIA)

MAY, 2002

CANDIDATE'S DECLARATION

I hereby declare that the project work entitled "Swift Certification for NCS Product" done at Tata Consultancy Services, Gurgaon, submitted in Department of Mathematics, IIT Roorkee, is an authentic work carried out by me during the period from 15 January 2002 to 25 May 2002, under the guidance of **Prof. R.C.Mittal** (Departmental Guide) and **Ms. Richa Goel** (Organisational Guide) for the partial fulfillment of the degree of "MASTER OF COMPUTER APPLICATIONS".

This has not been submitted anywhere else for the award of any other degree/diploma.



(Rahul)


Date: 31 May 2002

This is to certify that the above statement made by the candidate is correct to the best of our knowledge and belief.

Dated: 31 May 2002



(Dr. R.C. Mittal)
Professor
Deptt. Of Mathematics
Indian Institute of Technology
Roorkee- 247667
(INDIA)


Professor & Head
Department of Mathematics
I.I.T. Roorkee-247 687
7.6.02

DEPT. OF LABOR
BUREAU OF LABOR RELATIONS
WASHINGTON, D. C. 20340



TO WHOMSOEVER IT MAY CONCERN

This is to certify that the project titled "*SWIFT Certification for NCS Product*" being submitted by **Mr. Rahul**, student of **Indian Institute of Technology, Roorkee**, in partial fulfillment of the requirements for **Master of Computer Applications**, has been completed under the supervision of **Ms. Richa Goel** at **Tata Consultancy Services, Udyog Vihar, Phase – V, Gurgaon**, during the period from **January 16, 2002 to May 31, 2002**.

This work has not been submitted to any other Institution or University for the award of any degree to the best of my knowledge.

A handwritten signature in cursive script that reads 'Richa'.

Richa Goel

**Project Prime,
Tata Consultancy Services,
Plot No. 447, Udyog Vihar,
Phase – V, Gurgaon,
Haryana.**

Date: June 05, 2002.

TATA CONSULTANCY SERVICES

A Division of Tata Sons Limited

Plot No 447 Phase V Sector 19 Udyog Vihar Gurgaon Haryana 122016 India
Tel +91 0124 6342941 542 547 548 940 Fax +91 0124 6342543 6342544 6342550
E-mail: richa@tcs.com or richa.goel@tcs.com or website www.tcs.com
Registered Office: Bombay House 28 Feroz Road Mumbai 400 001

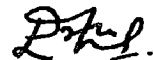
Acknowledgement

My gratitude goes to Dr. R.C. Mittal, my guide at IIT Roorkee whose enthusiasm is always a source of inspiration. Without his kind support, this project would not have had the direction that an academic dissertation, which interfaces with industry, should have.

Working at TCS, one marvel at the human machinery that is so gigantic yet works so nimbly. And it feels good to be a part of that workforce. This work would not have seen light of the day without the encouraging support of people at NCS-Fidelity.

I am deeply indebted to Ms. Seema Mehra (Project Manager) who gave me an opportunity to work on this project. I am also thankful to Ms. Richa Goel who not only assigned me a responsible role in the project but also showed confidence in me in spite of my being totally new to this kind of a work. The help provided by Ms. Rachna Ghai in acclimatizing to the work environment and her patient explanations to our queries was also immense. Under the able guidance of Sandeep and Suneet Dhar, I was able to meet the goals of the project. Thanks to the entire team of NCS-Fidelity for providing necessary details and tirelessly facing our queries.

Thanks are also due to the IDM staff at TCS Gurgaon who extended their kind support throughout the duration of our working.



(RAHUL)

Abstract

Custodial banking is a critical financial service. With millions of transactions occurring globally, online transaction processing plays a critical role. SWIFT messages are the communication standard for conducting such transactions. To enable global standardization, ISO 15022 standard for messaging was designed. SWIFT certification ensures that a custodial banking product is compliant to the ISO standard and is fully interoperable within the SWIFT network.

TCS has created NCS product for custodial banking services. For a product, which handles a critical service like custodial banking, being compliant with industry standards becomes a necessity.

This report reflects the work done towards preparing NCS product for SWIFT certification.

This work was done in the framework provided by the NCS-Fidelity project. A considerable portion of work involved testing the existing NCS-CA module for compliance to standard. This gave a fair idea of the present state of compliance. SWIFT standard was studied and a requirement document was prepared which specified the gaps.

To bridge these gaps, processes were suggested for taking certification process closer to the programmer level and implementation work of various modules was distributed among programmers.

Notice of proprietary information

All information contained in or disclosed in this document, herein after referred as 'Confidential Information', is proprietary to Tata Consultancy Services.

By accepting this material, the recipient agrees that this Confidential Information will be held in confidence, and will not be reproduced or used either in whole or in part, without prior permission from Tata Consultancy Services.

Table of Content

1	INTRODUCTION	8
1.1	FINANCIAL SERVICES.....	9
1.2	TCS.....	12
1.3	NCS PRODUCT.....	12
1.4	SWIFT.....	13
2	NCS PRODUCT	14
2.1	NCS MODULES.....	16
2.1.1	<i>Custody Management</i>	16
2.1.2	<i>Corporate Action</i>	16
2.1.3	<i>Trade Settlement</i>	16
2.2	TECHNICAL ARCHITECTURE.....	18
2.2.1	<i>System Components</i>	18
2.3	NCS - CA CYCLE PROCESS MODEL.....	20
3	SWIFT MESSAGES	21
3.1	MESSAGES.....	21
3.1.1	<i>Message Categories</i>	21
3.2	NCS AND SWIFT.....	24
3.2.1	<i>Logical Architecture</i>	25
4	SWIFT STANDARD	27
4.1	ISO 15022 STANDARD.....	27
4.1.1	<i>Basis for New standard:</i>	27
4.2	SWIFTRADY LABELS.....	31
4.2.1	<i>The labeling process</i>	32
4.3	CRITERIA.....	33
4.3.1	<i>Business Areas Requirement</i>	33
4.3.2	<i>Message Types Requirement:</i>	35
4.3.3	<i>Market Practices Requirement</i>	37
4.3.4	<i>SWIFT Alliance Access integration</i>	38
5	REQUIREMENT DOCUMENT	39
5.1	REQUIREMENT DEFINITION.....	39
5.2	DETAILED REQUIREMENT DESCRIPTION.....	39
5.2.1	<i>Swift Message Requirement</i>	39
5.2.2	<i>Requirement For MQSA</i>	40
5.2.3	<i>Requirement FOR Using BICPLUS data in NCS</i>	41
6	IMPLEMENTATION	42

6.1	MT564.....	42
6.2	IDENTIFYING THE BASIC PROBLEM.....	43
6.3	GAP SPECIFICATION.....	43
6.4	DESIGN	48
6.5	BEHAVIORAL MODELING	52
6.6	STRUCTURAL MODELING	54
6.7	DETAILED DESIGN.....	56
6.8	CODING	58
6.8.1	<i>Semantics of SWIFT messages</i>	58
6.8.2	<i>Fields</i>	58
6.8.3	<i>Field Structure Rules</i>	59
6.8.4	<i>Sequences/subsequences</i>	60
6.8.5	<i>Coding of Message bitmaps and headers</i>	62
7	TESTING.....	63
8	OVERVIEW.....	67
8.1	DATA DICTIONARY	67
8.2	IMPLEMENTATION OF BIC+ DATA GOLD.....	67
8.3	IMPLEMENTATION OF MQSA.....	70
8.4	WORK PROFILE.....	72
9	CONCLUSION.....	74
10	APPENDICES	75
A.	GLOSSARY.....	76
B.	DEVELOPMENT PROCESS	77
C.	TESTS AND REVIEWS.....	78
D.	NCS TECHNICAL SPECIFICATIONS.....	80
E.	BIBLIOGRAPHY	81

1 Introduction

"If debugging is the art of removing bugs, then programming must be the art of inserting them. Documentation is expressing plain regret."

-Anonymous

A software developer has an interesting process of evolution. He is born as an academic programmer – when he is *learning* to program, typically in a degree course. Then he goes to grind his nose against project deadlines in a real software development firm. Here he learns how to *deal* with problems- programs & bugs, handed down to him. This turns him into a professional programmer. With a few years of roasting, he realizes how to *create* problems, programs and bugs for others. Now he is a software developer.

This work reflects my transitions from being an academic programmer, to a probable professional programmer.

During the course of study, we had gained theoretical knowledge of software development process. This work provided a real world experience to apply that knowledge.

This report is broadly divided into three parts. In the first part, the information domains are summarized. This is necessary to understand all that happens later. In the second part understanding that was gained is further enhanced by analyzing the problem domain and recommending solutions. The final part deals with implementation of one part of the requirement.

1.1 Financial Services

The domain profile

A *security* is any financial instrument, which can be traded as scrip - like stock, bonds, equity etc. The securities market is a financial system where new capital is raised (the primary market) and where trading in existing securities takes place (the secondary market). Examples are stock exchanges around the world, as well as international capital markets.

Securities are offered in the market by *corporate issuer*. This offer is picked up by *customers* who may be individual investors or financial institutions like banks. The issuer causes a security to undergo a change in value or state by announcing an event - like declaring dividend or stock split. Such an announcement is called a *Corporate Action (CA)*.

The CA information is sent as a *message* between an account servicer (AS) and an account owner (AO). Such a message may

- Convey information regarding forthcoming CA events
- Enable AO to provide AS with the necessary details and instructions in order to carry out transaction relating to a CA event
- Confirm to the AO that AS has carried out the required transaction with the appropriate adjustments (credit/debit) to the account.
- Enable AO and AS to exchange narrative details, whether outlining complex instructions pertaining to a CA event or information regarding AGM or proxy voting actions/requirement.
- Enable AS to provide status information of an ongoing CA event or instruction sent by

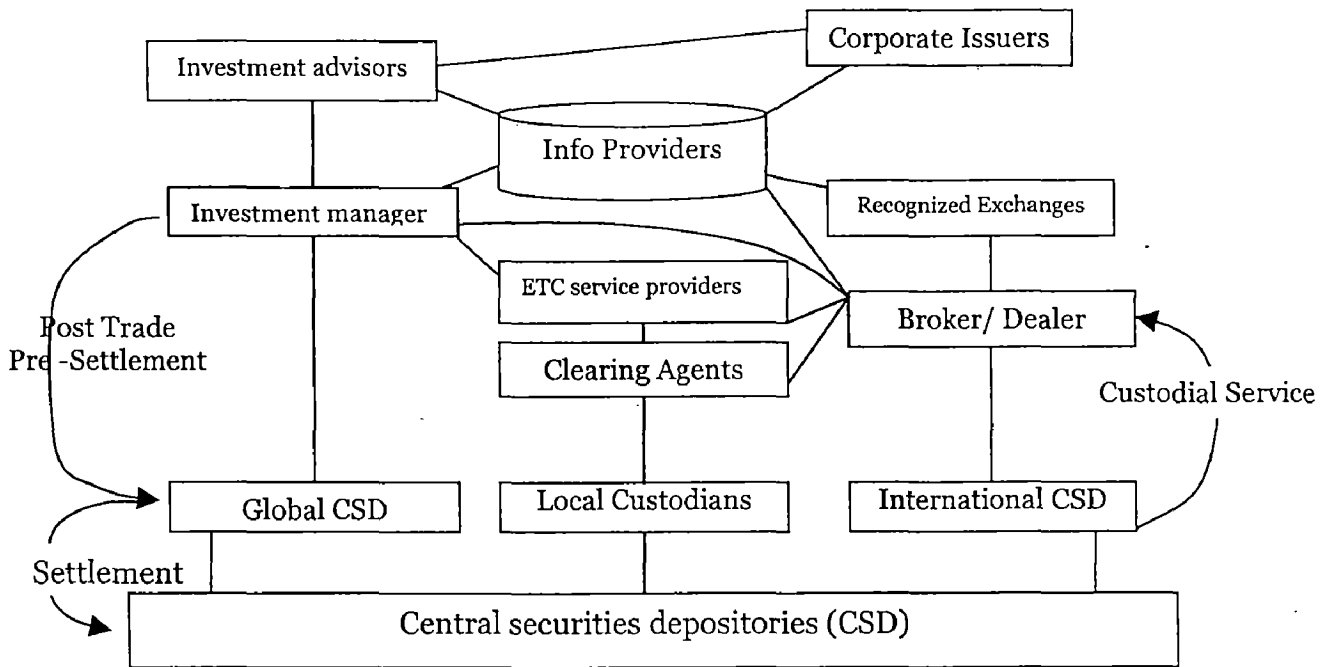


Fig. 1.1 Security Market

These CA messages initiate a chain of activities, which is responsible for the movement of securities in the market. There are four segments of activities in security market:

Pretrade Or trade	Post trade or pre-settlement	Clearing and settlement	Custody services
investment research, price discovery, new issue, trade order	allocation, confirmation	Settlement instruction and confirmation, status reports.	asset administration income collection corporate event processing

TABLE 1.1 ACTIVITIES

These activities occur among the market players as shown here:

A brief description of each market player is given below:

Central securities depositories: It is an institution, which holds, immobilized or dematerialized securities. Its primary functions are:

- Determining eligibility
- Book entry movements
- End of day cash settlement

-Collecting and distributing income and dividends

Local custodians: Is a bank that provides custody for securities that are settled in the country in which the custodian is located.

Global custodians: Provides its customers with worldwide custody service, usually by acting through a network of sub-custodians. It re-processes each transaction that settles in markets around the world in order to normalize the accounting and creating a single consolidated statement.

Clearing agents: Is an intermediary, a bank or a broker, that works between buyer and seller to settle securities transaction, including from other brokers.

International CSDs: It clears and settles international securities or cross border transaction in domestic securities, either directly or indirectly through local CSDs.

Investment managers: Institutions engaged primarily in the provision of investment management services and authorized by national or regulatory authorities for this purpose.

Brokers/dealers: Institutions engaged in dealing in securities related instruments. They must be recognized by at least one regulatory authority and/or be authorized to trade on a recognized exchange.

Recognized exchanges: Institutions providing exchange facilities and establishing rules followed by their members. They must maintain public list of securities and be recognized by at least one regulatory authority.

ETC service providers: An organization that provides a securities ETC processing service that includes matching and affirmation of trades between brokers, or brokers and investment managers; the providing of settlement instructions to a counterpart's settlement agent and status reporting of the match or affirm

The activities between international CSDs and Broker/Dealer are termed Custodial Services. The *Custodians* keep securities in safe-custody for their customers. When a particular CA occurs, they act upon security on the basis of standing instructions provided by a customer.

It is for this market that TCS has created its product NCS.

1.2 TCS

Tata Consultancy Services (TCS) is Asia's largest and global top 20 provider of software and software services. TCS predominantly undertakes assignments in the areas of system integration, systems development, and product implementation as well as application migration. TCS is a global company with over 19,000 consultants working on engagements for over 800 customers in 51 countries. TCS operates from 151 offices in 29 countries, including 50 in the US and Canada, and 45 in India. TCS' consultants are located either at TCS' offices or at customer sites all over the world

Financial Services are one of the core competency areas for TCS, accounting for approximately 40% of its revenue. The overall revenue is slated to touch a figure of USD 1 billion in the current year. In the financial service area, TCS has built-up considerable expertise and has carried out a number of projects for distinguished banks, financial institutions, stock exchanges, clearing organizations and depositories worldwide.

1.3 NCS Product

NCS is abbreviation for *Network Custody and Clearing System*. On TCS website, NCS product comes under the category *Industries > Financial Service> Products>NCS* and is briefly described as:

“NCS is a next generation custodial services system with a modular architecture and supports multi-currency environments. The product functions are governed by industry standards such as SWIFT messages for information exchange on one hand, and complex algorithms for valuation of new instruments on the other.”

It is a transaction-based system, which can achieve economies of scale by handling large transaction volumes. 'Rule based engine' lets custodians set customized and imaginative charge packages for their clients.

NCS has been implemented across Europe, Asia and Africa at multiple client locations.

1.4 SWIFT

SWIFT is *Society for Worldwide Interbank Financial Telecommunication*. [1]

Financial institutions rely upon the SWIFT messaging system to operate across borders. SWIFT is an industry owned cooperative supplying secure messaging services and interface software to over 7,000 financial institutions in 196 countries. It carried 1.5 billion messages in 2001. The average daily value of payment messages on SWIFT is estimated to be above USD 6 trillion. It provides messaging services to banks, broker-dealers and investment managers, as it'll as to market infrastructures in payments, treasury, derivatives, securities and trade services. These services help its customers reduce costs, improve automation and manage risk.

It maintains an open dialogue with regulatory authorities, while working across the company to guarantee the highest levels of system performance. While maintaining its full range of existing services, SWIFT is rolling out a highly secure, extremely reliable IP network to support our new generation of products and services.

Promoting end-to-end automated communications is at the heart of what it does and SWIFT standards are the accepted norm for financial messaging worldwide. SWIFT is also the messaging hub for a growing number of high-value payment systems, securities infrastructures and foreign exchange settlement systems. These market infrastructures need a trusted third party to provide secure, reliable and proven messaging solutions to their diverse users. Banks and regulators are turning to SWIFT to fulfil that role. Trust is also essential when trading over the Internet. SWIFT's standardized products enable financial institutions to offer trust services and e-payments to corporate customers active in B2B e-commerce. [2]

SWIFT's trusted third-party status extends beyond market infrastructures and the Internet. The international nature of its shareholders and its industry-owned co-operative structure have helped it to shape best practice in the financial industry and act as a forum to advance critical dialogue on industry-level issues and opportunities.

2 NCS Product

NCS is meant to support the custody related business requirements in both scripless as well as scrip-based environments. It facilitates a wide range of custodian activities. A few of the functional areas covered by the system are:

Custody management	Corporate action administration
Trade settlement	Rejection and claim monitoring
Registration	Cash planning
Charges and billing	Securities lending

Table 2.1 Function Areas

In addition, the system interfaces with several external systems such as:

SWIFT network for processing SWIFT messages	The Bank's cash and general ledger systems
Rate Distribution System for foreign exchange rates	Price Feed System to obtain security prices

Table 2.2 External Systems

A pictorial overview of the various sub-systems, the external entities with which NCS will interact and interface is given in the figure shown in the next page.

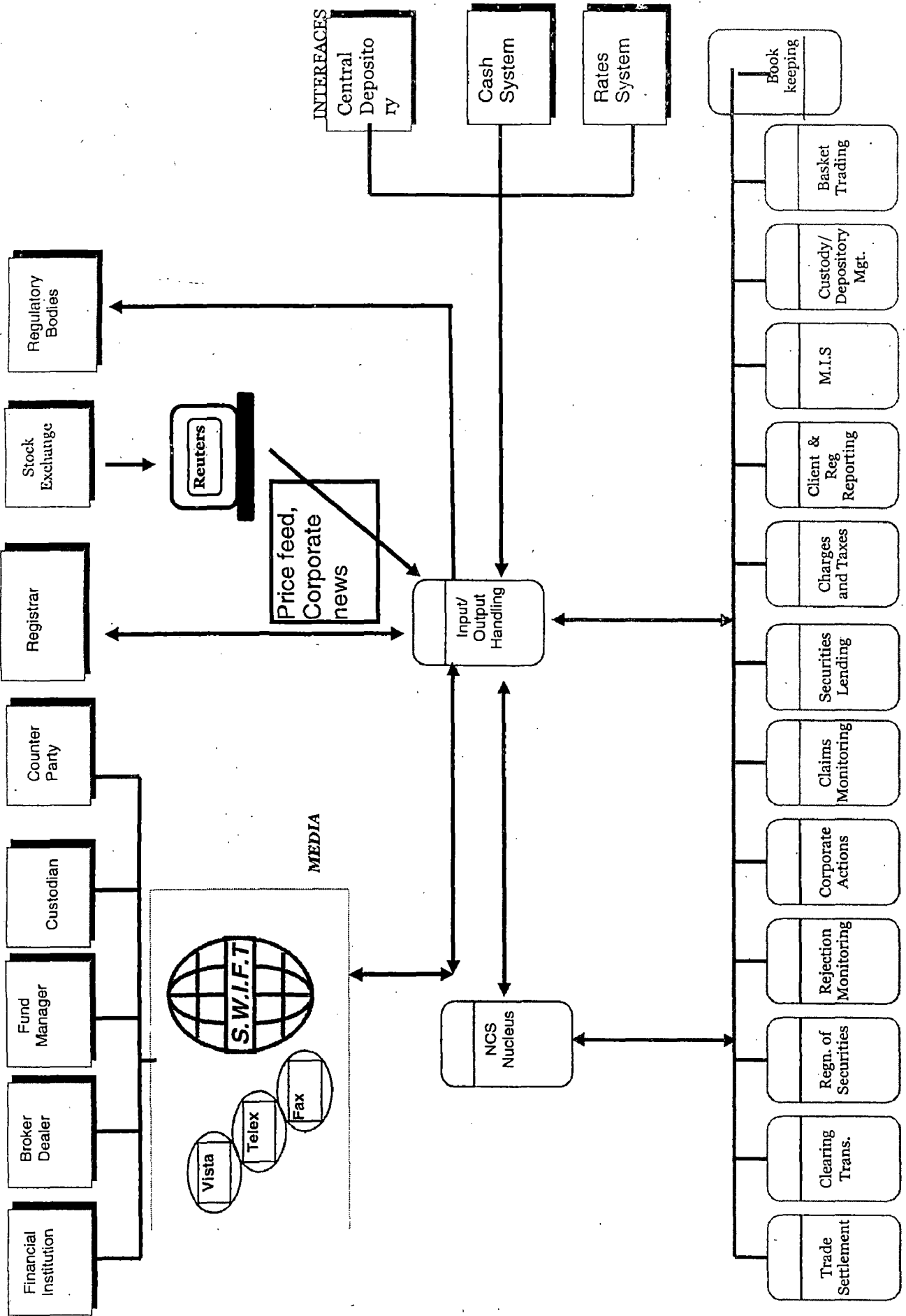


Figure 2.1 Network Custody & Clearing System

2.1 NCS Modules

NCS is a very large system of modules. It is not possible to summarize the functioning of each one of them in few pages. Hence only the modules relevant for the present study are described below. Appendix A gives a glossary of terms used.

2.1.1 Custody Management

It provides for management of physical documents such as share and bond certificates. The physical movement of securities into and out of safekeeping vaults is tracked by means of maintaining lodgment and withdrawal details. This function also enables maintenance of custody position at three levels for scrip based securities –

- the overall custody position as a quantity,
- the breakdown of this quantity in terms of the various denominations and
- by maintaining individual certificate numbers.

Other capabilities include catering for market specific requirements such as maintaining letters of guarantee details in lieu of share certificates (as in Manila) and legend status (as in Hong Kong) etc.

2.1.2 Corporate Action

A corporate action is defined in the system as any action on a security which can result in a cash or security earning or can even otherwise result in a change in the underlying security holding. The various corporate actions related requirements are covered under the following.

- Dividends (scrip and cash)
- Bonus issues
- Rights issues
- Conversions (e.g. of debt instruments/ loan stocks to equity)
- Stock splits/Construction/Sub-divisions
- New Issues/Initial Public offerings
- Redemption of securities
- Immobilization of securities (in markets undergoing transition to scripless environments)

It offers following services..

- Corporate action status information to client at all stages
- Request client instruction on corporate action options - cash, scrip and currency options
- Processing of client instruction and standing instructions on the options
- Computation of client entitlements
- Reconciliation of entitlements with actuals received from registrar
- Reminder to client who has not responded to a CA (e.g. in case of rights).

2.1.3 Trade Settlement

The trade settlement module facilitates manual order creation, affirmation and settlement for orders (buy, sell, receive and delivery) received from client. The orders are validated, accepted if the validations are successful or rejected if any errors are found.

- The accepted buy / sell orders can be placed on one or more brokers and tracked through to execution and settlement.
- The receive / delivery orders can be affirmed or failed by the user depending on whether or not the counter party confirms the order.

- On settlement date, the user can settle the orders (cash, stock or both), which will generate the necessary accounting entries to the cash system. Depository interface, where applicable, is available for trade settlement.

2.2 Technical Architecture

The System Architecture is based on the Client/Server model. The Client component interacts with the end-users and provides the User Interface functions. The Server component provides various business-related services to the Client and to other external systems like SWIFT. The Client and the Server components communicate between each other over a Network through requests and responses.

2.2.1 System Components

The various components can be grouped into the Client layer and the Server layer based on their expected functions. A layer wise representation is shown below:

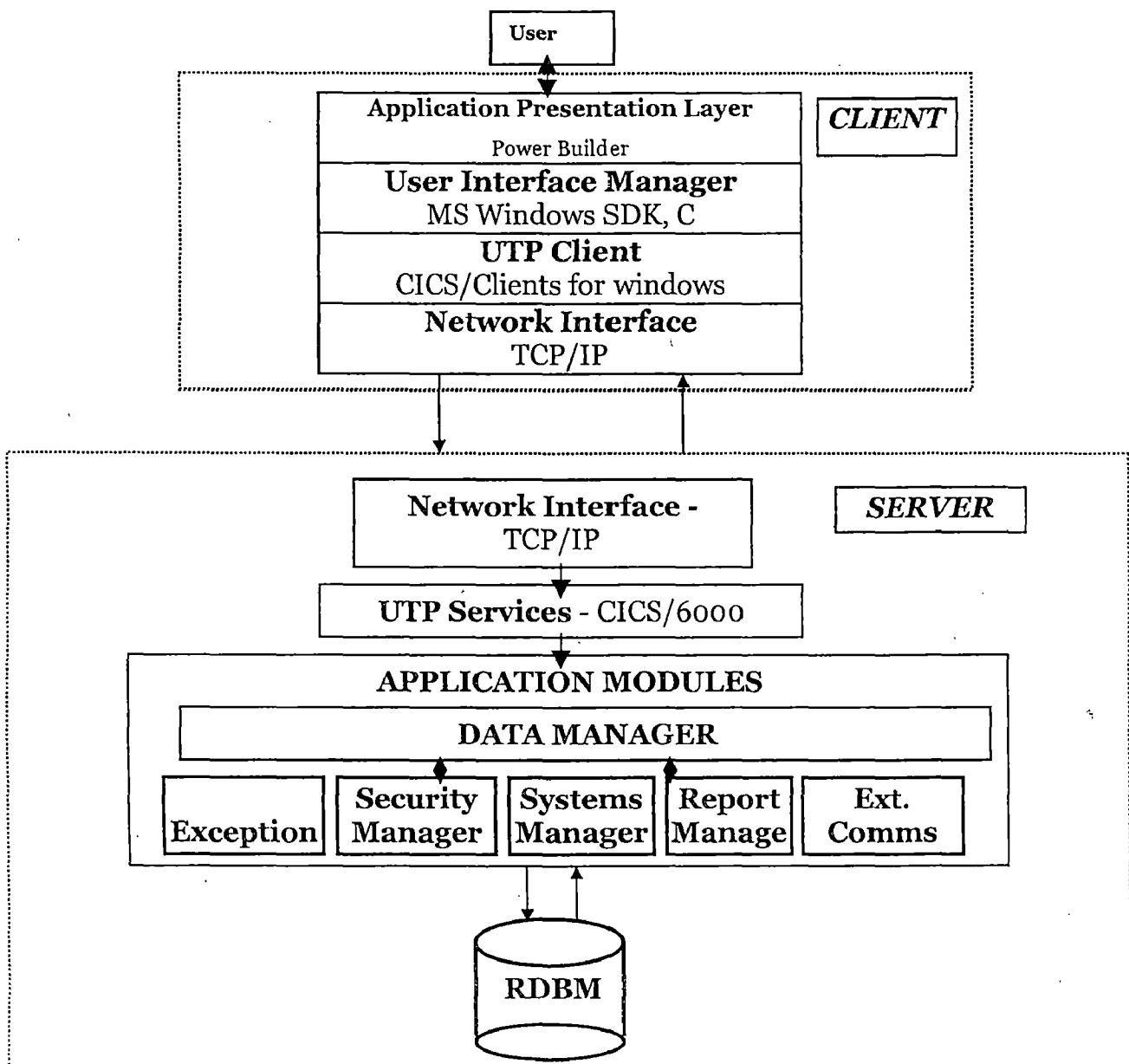


Figure 2.2 NCS Client/Server architecture layers

2.2.1.1 Application presentation layer

The application presentation layer of the Client provides the front end to the users for interaction with the application. The Presentation layer interacts with the end-user to get inputs and present the response. It also interacts with the Universal Transaction Processor on the Client to pass the user inputs and to get the responses.

2.2.1.2 User interface

This is window based GUI with which the user interacts.

2.2.1.3 Universal transaction processor

The Universal Transaction Processor implements the communication between the Client and the Server and the transaction management thus isolating other components of the system from the above. It comprises of two components: UTP Client and UTP Server.

- **Client UTP:** The UTP on the Client workstation provides a set of APIs for the application Presentation layer to initiate communication with the Server. The functions provided by the UTP Client include:
 - Initiate a transaction and get response for the transaction from the Server
 - Check and receive any Notification/broadcast messages from the Server for the Client
 - Pass the responses to the Presentation layer for subsequent processing
 - Authenticate the communicating party.
- **Server UTP:** The functions provided by the Server UTP include:
 - Receive the service request from the Clients
 - Interface with the Application security handler to validate the user and the service requested by the user
 - Invoke a service identified by the Client request
 - Communicate with the Data Access Module for any database related requirements
 - Send responses to the Clients for the services requested
 - Generate and send broadcast messages for the Client(s), automatically when necessary or upon request by the System Administrator
 - Provide a mechanism for sending of notification messages generated by the application modules
 - Maintain the control information of the users logged on to the system
 - Provide distributed transaction initiation/processing facilities for enabling inter-branch transactions.

2.2.1.4 Data Manager

The Data manager provides database access and update functions for different application modules. These functions will be called directly from the services on the server. It will thus encapsulate all the database-specific code used in the application. This will enhance reusability and will enable changes to the database without any change to the main application code as only the DM functions need to be touched in most cases. This will also enhance the portability of the system across different RDBMS platforms, if required, because only the Data Manager layer will have to be modified or rewritten in order to port the software to any new database. Though it is possible to access application database directly, to maintain consistent database access and modularity, all database access operations should be carried out through Data Access Module.

Appendix D provides Detailed technical specifications of NCS.

2.3 NCS - CA cycle process model

For this particular work we are concerned only with the NCS CA module. Processing a corporate action follows a well-defined process summarized below. It involves the users in the role of analyst, first manager (FM) and second manager. Each of them has a well defined task and access rights to process the message.

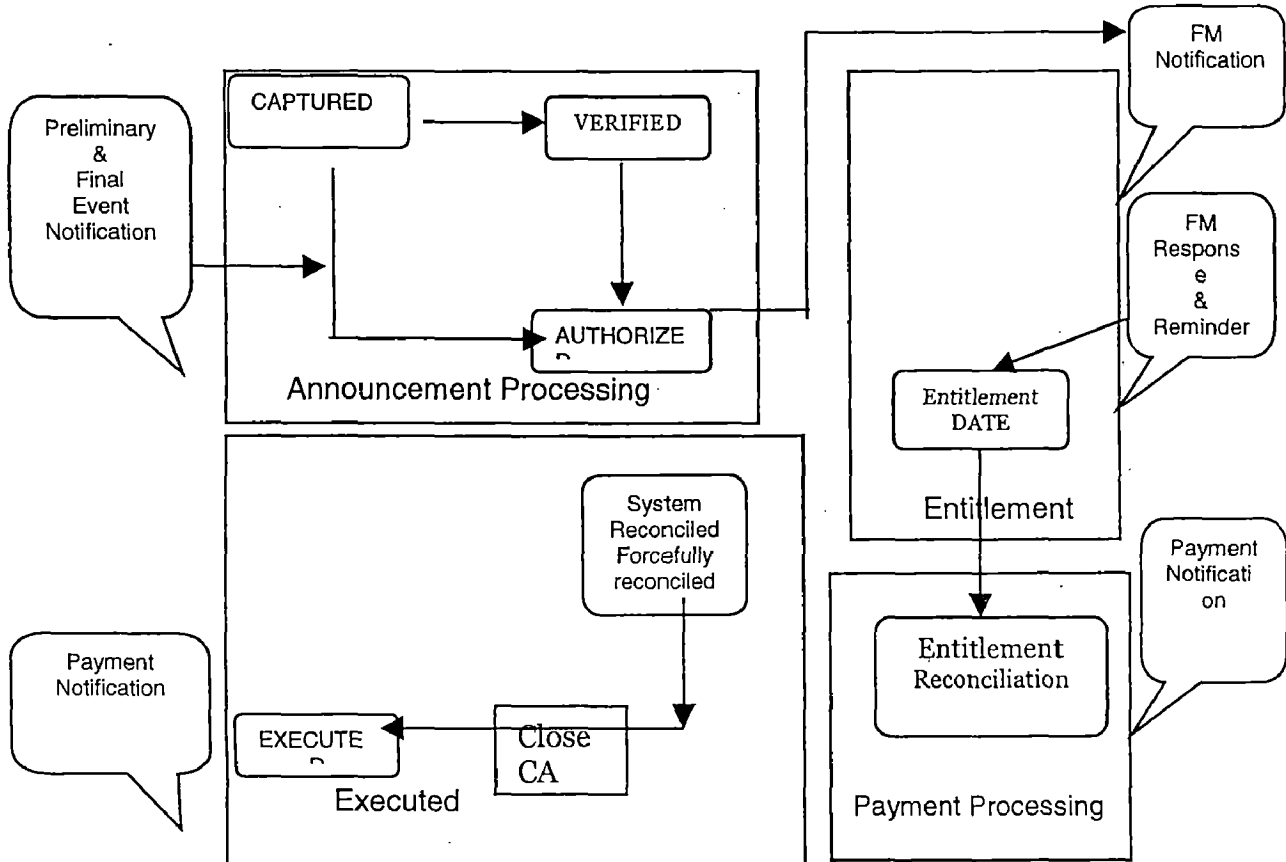


Figure 2.3 CA Cycle

3 Swift Messages

3.1 Messages

Financial Institutions around the world use Swift to exchange confidential financial data and messages, which enable them to conduct transactions in an efficient manner. They expect the messages to be delivered securely, accurately, and rapidly. Hence SWIFT has been designed to be a secure system providing:

- Confidentiality- info can only be disclosed to authorized user
- Integrity- receiver of the message must be confident that the data is genuine and unchanged
- Availability- the system and all info must be available when required
- Accountability – responsibility must be clearly defined in order to be able to set liability in case of problem

3.1.1 Message Categories

The entire collection of messages can be broadly categorized into three categories based on their functionality. [7]

- User to User messages

These are sent from one Financial Institution to another in order to conduct a transaction. These are subdivided into 9 categories

Financial Institution → Financial Institution
←

- System messages

These are sent from the Swift system to the user or vice versa. These are category 0 messages

SWIFT → Financial Institution
←

- Services messages

These messages, also called control messages, relate either to system commands like LOGIN SELECT QUIT or to acknowledgements.

FI → SWIFT

Each category contains a Message Type (MT) given by a 3 digit number like MT567. Here follows a brief description of each of the categories of messages.

i. system messages

These are sent from swift system to the user or vice a versa.

examples are

MT 020 Retrieval request (text and history)

MT 046 Undelivered message report request

MT 094 broadcast

ii. Customer transfers and cheques

Deal with payment information of payments in which ordering customer or the beneficiary customer or both are not FIs.

MT 101 Request for transfer

MT 103 Single customer credit transfer

1 MT 10 advice of cheques

iii. Financial Institutions transfer

Deal with payment or information of payments in which ordering customer and the beneficiary customer both are FIs.

MT 200 FI transfer for its own account

MT 202 General FI transfer

MT 210 Notice to receive

iv. Treasury markets –Foreign exchange, money markets, derivative

Messages exchanged b/w FIs on behalf of themselves providing information about confirmation and settlement of various money market operations. Confirmation messages are confirmation of information already known to both the FIs. They handle only the contract part of the business and are not used for transfer of funds.

MT 300 foreign exchange confirmation

MT 320 fixed loan /deposit confirmation

MT 340 forward rate agreement confirmation

v. Collections and cash letters

Exchanged between banks in the handling of documentary and clean collections as well as cash letters

MT 400 advice of payment

MT 420 tracer

MT 430 amendment of instructions

vi. Securities

Exchanged between FIs involved in security transactions

MT 521 receive against payment

MT 530 confirmation of receipt free

MT 553 instruction to a custodian

vii. Precious metals and syndications

Includes two sets of message types

-Exchanged between FIs involved in precious metal transactions

Syndication messages deal with drawdown/renewal of facility, setting of interest exchange rate of a facility and payment of principal , interest and /or fees due on a syndication

MT 600 precious metal trade confirmation

MT 605 precious metal option to receive
MT 609 statement of a metal account

viii. Documentary credits/ guaranties

Exchanged between banks involved in credits and guarantee business

700 issue of a documentary credit

707 amendment to a documentary credit

730 acknowledgement

ix. Travelers cheques

Exchanged between issuers and remitting agents, selling agents and refund agents in handling of travelers cheques for sales settlement refunds and inventory administration

800 t/c sale and settlement advice

810 refund request

812 refund authorization

x. cash management and customer status

Exchanged between FIs either on behalf of themselves, other fis or customers for

-balance reporting messages

-Netting messages

-Status messages

900 confirmation of debit

910 confirmation of credit

950statement message

Classes of Messages

The swift messages have been classified into ten different categories of which category 5 pertain to the securities market. The messages in this category can be grouped in three classes:

Trade Initiation and Confirmation

These include messages like buy/sell etc. For eg the message MT502 is used to give instruction, buy bonds. Similarly messages like MT515 is used for confirmation (confirmation of purchase in this case).

Settlement and Reconciliation

The settlement messages pertain to a trade of securities and involve transfer of securities and/or cash. Like MT541 message is for receive against payment, i.e. purchasing securities against cash. For reconciliation with bank, custodian etc, messages like MT535, which is a statement of holding, is sent.

Corporate Action

These are messages informing a particular event announced by a company like MT564, or are messages exchanged between the account owner and the account servicer like MT565 and MT567.

Next we move to see how SWIFT and NCS are related by design.

3.2 NCS and SWIFT

In NCS, message processing is summarized in the following diagram. Here following steps take place for an incoming message:

1. NCS interface handler/message controller receives an incoming message feed
2. A CICS Background Task (CBT) runs periodically to pick this message.
3. This batched message is deformatted
3. Information is parsed and populated in backend database tables with Data manager.

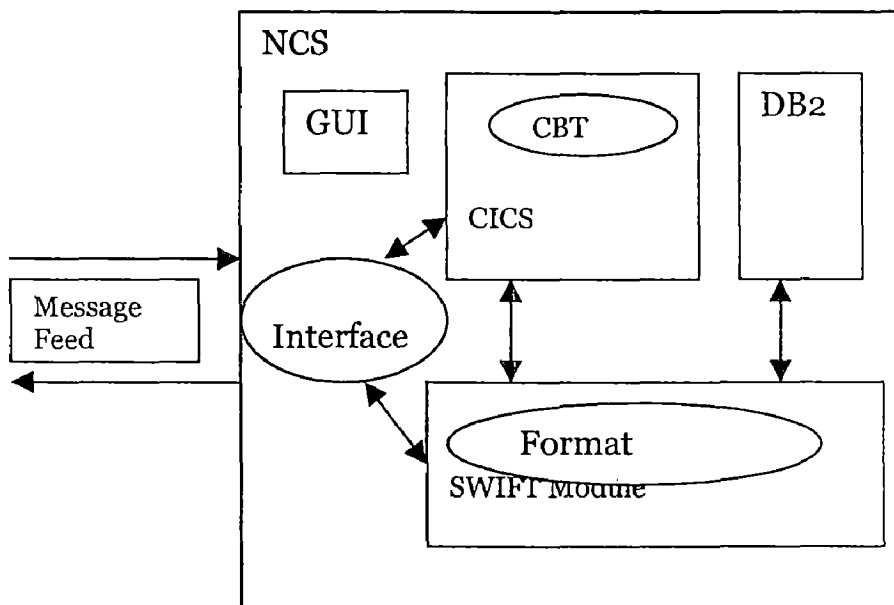


Figure 3.1 NCS SWIFT

In order to have a regular batch interface with SWIFT terminal devices and other systems, NCS has the Message Control Module. It provides the support of the messages in SWIFT format to receive orders from/ and send responses to the customers connected to NCS through SWIFT network. The key features of this are:

- Message acceptance from SWIFT network
- Message repair and rejection for incomplete / incorrect instructions
- Enabling straight through processing by automatic order creation for MT52X, MT54X, MT50x and MT592 messages
- Auto-release mechanism for system generated messages
- Ability for user to specify Sender to Receiver information
- Facility for user to initiate free format messages
- User controlled release of certain types of messages
- Routing of messages to other sites.

3.2.1 Logical Architecture

The various modules have been logically arranged as follows:

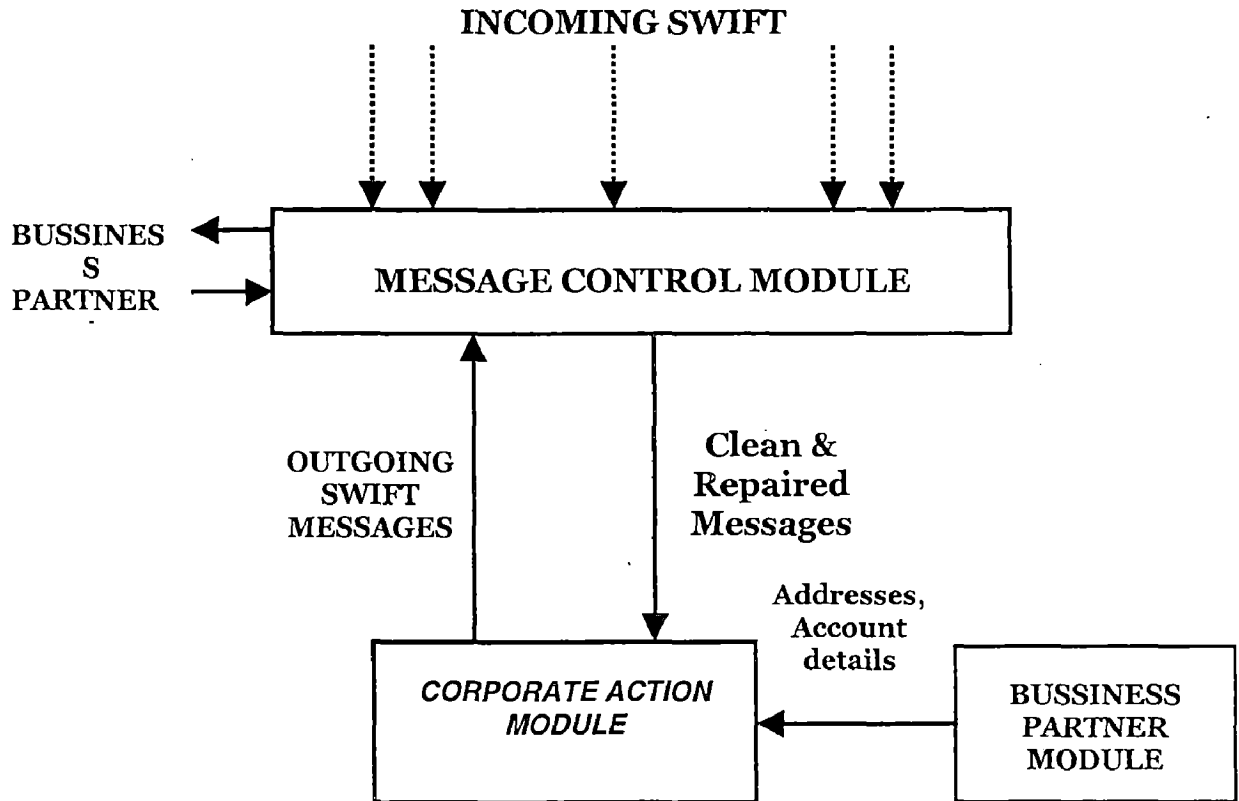


Figure 3.2 Logical Architecture, SWIFT

The Corporate Action module is central to this system and receives and sends swift messages through the Message Control module. It collects the information regarding the funds and business partners from the Business Partner module. The auto-receipt of messages is followed by a Straight through processing (STP) where the application of business logic auto-translates the message details into functional data without manual intervention.

In a typical CA lifecycle in the NCS Application, the message undergoes various stages of processing. This is illustrated through the following diagram

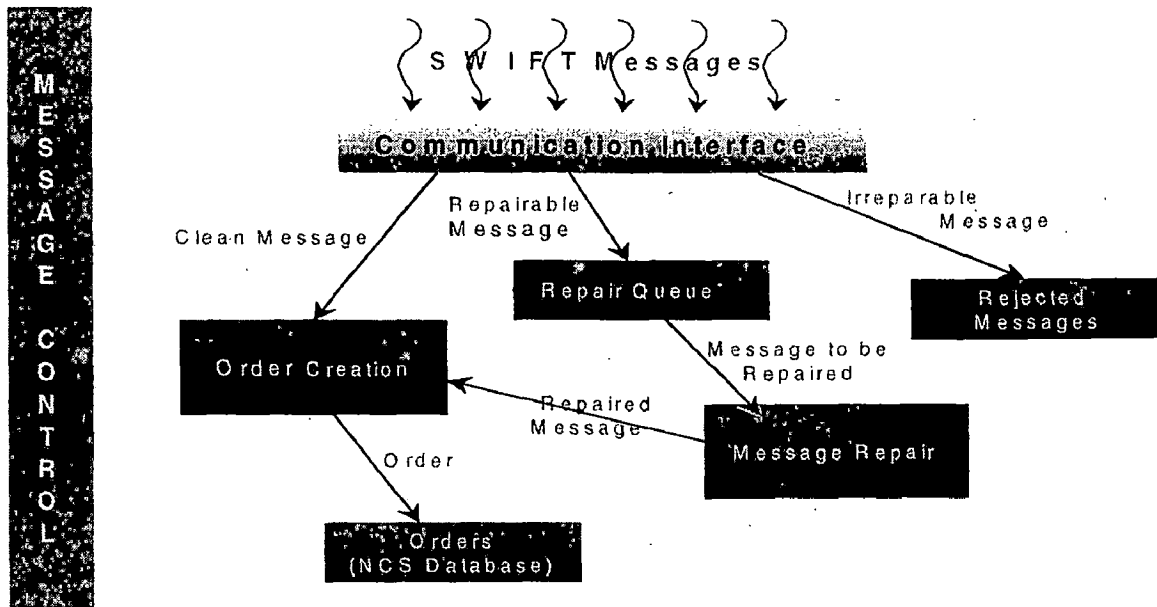


Figure 3.3 Message States

All the message processing is handled by the GUI at user level. It provides a comprehensive interface to the entire cycle for the various CA types. Discussion about GUI is out of the scope of the current work.

It should be noted that ideally all messages should undergo Straight Through Processing (STP), where manual intervention is not required to process the message unless there is an irrecoverable error. The processing of the messages is generally subjected to the Market Practices local to the place of processing.

4 SWIFT standard

In October 1998, SWIFT launched a new set of ISO 15022 compliant messages, 38 in number, which were destined to replace the existing ISO 7775 generation of messages. The design of ISO 15022 guarantees that efforts and investments will not be lost in a future evolution ex. Towards XML based standards. It gives Straight Through Processing and Market Practices there due place.

The old ISO 7775 messages, which have been used on the SWIFT network for the last 15 years, are technically not able to provide a correct answer for the needs of today:

- they are unable to match current market requirements, both in terms of complex transactions or new business developments
- they have reached their limit in terms of STP ability
- cant provide shorting the settlement cycle to T+1
- cant offer better risk management

So a new and better way to address communication for security transaction is needed. Hence ISO 15022 was introduced.

4.1 ISO 15022 standard

The basis of new standard is *business elements* defined for all transactions and grouped in data dictionary. There is only one element per specific business meaning, which remains the same for use in all-relevant messages. Business elements are grouped logically according to their business meaning. When more data is needed, new elements are defined and added to the dictionary. So all messages can easily be processed through a single application. The 'wrapping' of one business element in one syntax or another becomes a separate process that can be performed by a communication layer (often middle ware) without impacting business applications. ISO 15022 allows this migration to a syntax independent system. [8]

This also enables the community to define the messages they need when they need them. The actual messages are kept out of the ISO standard, so that the individual messages are not subject to the lengthy ISO approval process. [16]

4.1.1 Basis for New standard:

The new swift criteria are based upon the realization that message implementation should be isolated from business process or the way messages are transmitted. It models the standard by recognizing three distinct layers in which information can be mapped - Business, Logical, and Physical. It recognizes there are different standards layers and puts back to those layer what belongs to them. This enables the layers to evolve without being a burden on each other. This standard has been designed for separating business and representation by recognising the three layers :

- Business layer has data field dictionary for defining-Which business elements? What do they mean?
- Logical layer provides design standards for using data fields together by providing message design rules.
- Physical layer tells how to code them and how to transmit them -the language syntax and carriers.

Business Layer: it involves the Domain Information Model containing:

- Business Process
- Business Domain
- Actors

It is focussed on business not implementation and describes:

Business Context Of Solution

how the market works

what are the rules

who are the players

how they interact with each other

Activities that occur in this layer are

- Identify Business Domain
- Define Structure/Dynamics of Business Context
- Capture Business Information within Business

Analogy: “Architectural Designs for Building a House Based on Client Requirements”

Logical Layer : it involves the Message Information Model containing:

- Message Structure
- System Interaction

It specify the Exchange of Structured Business Data (Messages) and describes:

Abstract, Technology Neutral Solution

how to translate the components of the flows into specific data elements

what is their meaning

how to define them precisely

how these elements interact with each other

Activities that occur in this layer are:

- Message Interaction - Collaboration Diagrams
- Dynamics of Solution - Activity Diagrams
- Business Scenarios - Sequence Diagrams
- Message Structure/Content - Class Diagrams

Analogy: “Detailed Plans for Wiring a Building”

Physical Layer: It involves the low-level Message Formats containing Format Transformation Rules (E.g. DTDs, Schemas). It defines the Mapping Specifications from the Logical Layer to Target Implementation (XML, Java etc.). It specifies:

-Which syntax, networks and protocol to use:

Store and forward

Interactive

File transfer

In this layer Design Rules for each syntax are created. C13,14

Analogy: “Work Plan For Actual Building Construction”

It is important to keep these layers independent from each other. Each layer can then evolve without impacting the others, especially when it is not necessary. As an example, moving from T+3 to T+2 should not impact market practice, communication or technical standards. You must be able to change technology (technical standards) without having to change the content. [9]

Benefits of implementing ISO 15022 are as follows:

- ISO 15022 was built to support the entire transaction chain, enabling consistent transaction handling from the very beginning to the end of the securities transaction chain.
- Maximize the STP potential of each individual player in particular and of entire chain in general.
- To benefit from new technology
- Interactive communication
- Ip network
- Use of central database and matching system
- To allow you to handle more complex transactions

More importantly, a data dictionary implementation enables the customers to be fully syntax independent. Since the business elements and the business meaning will remain the same in FIN and XML, XML can then be treated for what it is; a technical syntax. To start using XML based standards in the future, no additional major work will be necessary, since all the work done to identify the elements will have already been undertaken as part of the ISO 15022 deployment initiative. It will allow you to easily continue using FIN messages with counter parties not having fully migrated to XML, for as long as necessary. [10, 11]

For example corporate action messages based on ISO 15022 offer the following advantages:

- Less operational risks
- Function of the message clearly defined (New, replace, reminder,...)
- Clarity on event type and one MT for all events
- Clear Indication of Mandatory/Voluntary
- Better tracking of the status of the CA (Preliminary, un/confirmed, complete)

Table 4.1 CA Transition and Swift Messages

CA Transition	Function of Message	Incoming/Outgoing	ISO7775	ISO15022
Announcement Processing	STP after Auto-receipt of announcement information	Incoming	MT550, MT551, MT552, MT556, MT560	MT564, MT568
Announcement Processing	Auto generation of Pre – advice	Outgoing	MT550, MT551, MT552, MT556, MT560	MT564
Entitlement/Reconciliation Processing	STP after Auto-receipt of entitlement information	Incoming	MT554, MT555, MT563	MT564, MT566
Entitlement/Reconciliation Processing	Auto generation of Pre – advice	Outgoing	MT550, MT551, MT552, MT556	MT564
Voluntary Response Processing	Instruction to Custodian	Outgoing	MT553	MT565
Voluntary Response Processing	Acknowledgement from Custodian	Incoming	MT562	MT567
Payment Processing	STP after Auto-receipt of payment information	Incoming	MT554, MT555, MT563	MT566
Payment Processing	Auto generation of Payment Advice	Outgoing	MT554, MT555, MT563	MT566

4.2 SWIFTReady labels

In today's world marketplace, a SWIFT interface must be broader and more flexible than traditional systems to be considered a viable international solution. To ensure this SWIFT provides a SwiftReady certification process for labeling products as SwiftReady Silver and Gold. It is based on ISO 15022 standard.



	
<p>SWIFTReady SILVER label indicates that the product supports SILVER messages in at least 3 User/Business functions SILVER Straight Through Processing/Operational Information Services, and ALLIANCE requirements</p>	<p>SWIFTReady GOLD label indicates that the product supports SILVER & GOLD messages in at least 3 User/Business functions GOLD STP/OIS, and ALLIANCE requirements</p>

Table 4.1 SWIFT Ready Labels

Vendor and release -specific label

The label is purely related to the product and not to the vendor who is marketing this product. Furthermore, the label is associated with a specific release of the product and does not apply for older releases/versions of the product. Typically, the labeled release is the latest release available to customers. The SWIFTReady label confirms that the labeled product satisfies the criteria as specified for the indicated market and for the Gold/Silver level.

Yearly review

The criteria for Silver and Gold qualification are reviewed every year. The requirements evolve to ensure they remain in line with the evolution and actual customer usage of SWIFT's products and services and the changes in the financial industry. As such, the labels are only valid for a period of maximum one year.

Every year, all labeled products have to be submitted for requalification to receive renewal of the label for the next year, based on the latest applicable criteria. SWIFT communicates changes to the criteria to the vendors at least six months in advance.

Components of the label

The label has several components, each with a specific meaning. The year indicates the version of the criteria that the labeled application complies with. The criteria are renewed every year to stay in line with the evolution of SWIFT's products and services. The market indicates the type of criteria the labeled application complies with. There are very specific and distinctive criteria per market. The qualifier can be Silver or Gold and indicates the level of compliance of the labeled application.

4.2.1 The labeling process

The process that vendors have to follow to obtain a label for one of their products is as follows:

1. Vendor completes a detailed questionnaire on the product, and sends it to SWIFT Partner Solutions
2. Partner Solutions analyses the completed questionnaire and determines whether the product on paper satisfies the Silver or Gold level criteria
3. Partner Solutions informs the vendor of the paper validation result (either the result is OK to proceed with the labeling or clarification is required on missing functionality) and schedules a follow-up meeting to discuss and/or prepare the technical/functional validation. This includes disclosure to the vendor of the testing scenarios that will be used during the technical validation
4. In parallel or after the technical validation, Partner Solutions needs to verify the compliance of the product versus the Silver or Gold criteria. This is called the functional validation and is done at SWIFT's or vendor's premises.
5. After positive completion of the functional and technical validation, contact is made with selective users of the product. Part of the validation process also focuses on the customer satisfaction with the quality and support of the product
6. Once all validations have been successfully completed, SWIFT awards the SWIFTReady Silver/Gold label for the product for the rest of the current calendar year, so for a maximum one year. The vendor signs an agreement (Letter of Agreement – Use of SWIFT Label) for the use of the SWIFTReady label in his publications and presentations (only for the labeled product), including compliance with the design guidelines on how to use the label.

4.3 Criteria

These criteria can broadly be categorized into following:

- Business Areas Requirement
- Message Types Requirement
- Market Practices Requirement
- SWIFTReady Access integration Requirement

4.3.1 Business Areas Requirement

The new criteria is based on the role application is playing in the securities industry. SWIFT has identified four business spaces and four types of financial institutions that are active in various securities business scenarios.

These spaces are

- Pre-trade and trade
- Post-trade and pre-settlement
- Clearing and settlement
- Custody

The four types of financial institutions are

- Broker/Dealer
- Investment Managers
- Custodians
- Universal Banks

Currently SWIFT does not offer standard messages in the first space (pre-trade/trade) but is present in the three other spaces. SWIFT is as well present in the four types of financial institutions.

Criteria 1

SWIFT require the support of all the messages in a specific business scenario related to the financial institution and the relevant three spaces where SWIFT is present.

SWIFTReady							
	Space 1	Space 2		Space 3		Space 4	
	Pre-trade / Trade	Post-trade / pre-settlement		Clearing and Settlement		Custody	
Messages		Retail	Wholesale	Settlement	Reporting	Corporate Actions	Collateral Management
SWIFTReady							
(IMI/BD)	Mandatory		Mandatory	Mandatory		(upgrade to Gold)	(upgrade to Gold)
(Cust.)	none			Mandatory		Mandatory	(upgrade to Gold)
SWIFTReady							
(U.B.)	Mandatory	retail or Wholesale		Mandatory		C.A. or collateral Mngt	
Silver	none	BIC		BIC & BIC+		BIC & BIC+	BIC and BIC+
GOLD (silver+)	none	none		Market Practices		Market Practices	none
Silver	none	AFT or MQSA					
GOLD (silver+)	none	AFT and MQSA					

Table 4.2 Business Areas

4.3.2 Message Types Requirement:

There is no difference anymore between the types of messages an application has to support to get a Gold or Silver label, all the messages related to a specific business scenario have to be supported for application to be eligible for a SWIFTReady label. To comply with the Label requirements you need to support all changes to existing message formats before the release date on the live SWIFT Network to leave enough time for your customer to test those modifications

Criteria 2

All the ISO 15022 based messages are included in criteria for 2002 standard.

All the ISO 7775 based messages will be removed from the criteria in 2002. These messages are all replaced by their ISO 15022-based equivalent.

Space 1		Space 2		Space 3		Space 4	
Pre-trade / Trade		Post-trade / pre-settlement		Clearing and Settlement		Custody	
Messages		Retail	Wholesale	Settlement	Reporting	Corporate Actions	Collateral Management
		Not applicable					
Send			513, 515, 509, 576, 518	540, 541, 542, 543, 548, 524	549	565, 567, 568	503, 504, 505, 507, 527, 558, 569
Receive	502		514, 509, 517, 549, 518	544, 545, 546, 547, 548, 578, 508	535, 536, 537, 538	564, 566, 567, 568	503, 504, 505, 506, 507, 527, 558, 569
		Not applicable					
Send	502		514, 509, 517, 549	540, 541, 542, 543, 548, 524	549	565, 567, 568	503, 504, 505, 507, 527, 558, 569
Receive			513, 515, 509, 576	544, 545, 546, 547, 548, 578, 508	535, 536, 537, 538	564, 566, 567, 568	503, 504, 505, 506, 507, 527, 558, 569
		Not applicable	Not applicable				
Send				540, 541, 542, 543, 544, 545, 546, 547, 548, 578, 524, 508	549, 535, 536, 537, 586, 538	564, 565, 566, 567, 568	503, 504, 505, 506, 507, 527, 558, 569
Receive				540, 541, 542, 543, 544, 545, 546, 547, 548, 578, 524, 508	549, 535, 536, 537, 586, 538	564, 565, 566, 567, 568	503, 504, 505, 506, 507, 527, 558, 569
Send	502	509, 549	514, 509, 517, 549	540, 541, 542, 543, 544, 545, 546, 547, 548, 578, 524, 508	549, 535, 536, 537, 538	564, 565, 566, 567, 568	503, 504, 505, 506, 507, 527, 558, 569
Receive		509, 515, 576	513, 515, 509, 576	540, 541, 542, 543, 544, 545, 546, 547, 548, 578, 524, 508	549, 535, 536, 537, 538	564, 565, 566, 567, 568	503, 504, 505, 506, 507, 527, 558, 569

Table 4.3 Message Types

4.3.3 Market Practices Requirement

Market Practice Rules for existing securities messaging standards, (e.g., FIX, ISITC, ISO 7775) have been historically defined after industry participants have implemented the standards. To further complicate industry information flows each industry participant has defined these rules separately and differently. This has resulted in an inefficient exchange of information whereby standards and their associated market practice rules have been interpreted and implemented differently by each industry participant in each geographic market. This inefficient exchange of information within and across different markets has limited Straight-Through-Processing (STP) in the securities industry. [3]

In July 1998, S.W.I.F.T. sponsored the organization of the Securities Market Practice Group (SMPG). The formation of the SMPG resulted in the establishment of National Market Practice Groups (NMPG) in 22 geographic markets comprised of broker/dealers, investment managers, custodian banks, central securities depositories and regulators.

The SMPG is a tactical initiative focused on enhancing the current securities industry infrastructure. However, this group also realizes the further benefit of industry utilities in dictating conformance to standards and market practice. As such, there is active dialogue between the SMPG and other industry initiatives (i.e., FIX, GSTPA, ISITC-IOA, SIA-T+1 (US), etc.), in order to ensure that the tactical work of the SMPG provides a first step toward the restructuring of the securities industry. [4]

In October 1999, the SMPG published market practices for Trade Settlement with input from 16 NMPGs. This document, divided into two sections titled "Common Elements" and "Country Specifics", details those business elements that are common across all 16 countries as well as additional requirements specific to each country.

Criteria 3

To qualify for a Gold label in 2002, it is a requirement (in the OIS/STP criteria) to support market practices. Validation of the presence of business elements when releasing an outgoing message is mandatory for Settlement and Corporate Action messages.

For Silver label support for BIC is required

For Gold label support for BIC+ is required.

4.3.4 SWIFT Alliance Access integration

In order to ensure interoperability among distinct systems in the SWIFT network , this requirement sets down the type of access/interface with which systems work in the swift alliance. [5]

Criteria 4

Silver label applications require automated File Transfer or interactive connectivity.

Gold Label applications always require interactive connection to the SWIFT Alliance Interface and automated File Transfer.

Interactive connectivity to SWIFT Alliance should be done using MQSeries Interface for Alliance (MQSA).

IBM MQSeries messaging software enables business applications to exchange information across different operating system platforms in a way that is straightforward and easy for programmers to implement. Moreover, MQSeries will be used as communication messaging middleware for SWIFTNet services and as such a real investment for the coming years.

5 Requirement Document

Purpose of the Document

The purpose of this document is to give a description of requirement identified in NCS to fulfill SWIFT READY requirements and to serve as the basis for preparing the further detailed documents. This document will identify the broad requirements for the NCS. The details of changes to current NCS business operations are also not a part of this document.

Background

NCS supports the processing of SWIFT messages. It is proposed to obtain SWIFTReady GOLD label for NCS product in October 2002. This requirement document is based on the SWIFT certification criteria 2002.

5.1 Requirement Definition

S. No	New Requirements	Remarks
1	Support for all ISO 15022 standard messages with data dictionary	
2	Support for communication using MQ Series for Swift Alliance (MQSA)	
3	Support for using BIC+ data	

5.2 Detailed requirement description

5.2.1 Swift Message Requirement

1. All the ISO 15022 based messages in a specific business scenario related to the financial institution and the relevant three spaces where S.W.I.F.T. is present should be implement for NCS to obtain SWIFTReady Gold label.

For present study we consider only the CA –Custodian part of NCS.

S. No	Requirements	Remarks
1	NCS System needs to process MT564 as announcement Message	
2	NCS System needs to generate MT565 as instruction to bank with responses of PM	
3	NCS System needs to process MT566 message for payment reconciliation.	

4	NCS System needs to process MT567 as acknowledgment to response of bank.	
5	NCS System needs to generate and process MT568 for inquiry and response of bank.	

2. Data dictionary of business elements should be used.

Swift now provides a set of business elements organized in data dictionary. This is aimed at isolating the implementation of processing logic from message definition or business process. This would make the system compatible with future evolution of SWIFT standard using XML etc. [6]

Description

The swift messages for CA- custodian is to be handled by NCS. It should support incoming & outgoing message functionality required by SWIFT.

Reference:

2002 Label Criteria	Clearing and Settlement - Settlement	Clearing and Settlement - Reporting	Custody - Corporate Actions	Custody - Collateral
S.W.I.F.T.Read y GOLD (User/Business functions)	Send : 540,541,542,543,54 4,545,546,547,548,5 78,524,508	Send : 549,535,536,537,586, 538	Send: 564,565,566,567, 568	Send: 503,504,50 5,506,507, 527,558,569
	Receive : 540,541,542,543,54 4,545,546, 547,548,578,524,50 8	Receive : 549,535,536,537,586, 538	Receive : 564,565,566,567, 568	Receive : 503,504,50 5,506,507, 527,558,569

5.2.2 Requirement For MQSA

SWIFT MQSA

S. No	Requirement	Remarks
1.	Real time handling of SWIFT incoming file.	All incoming messages will be transmitted through MQ and NCS would read the message from MQ.
2.	Real time handling of SWIFT outgoing file.	All outgoing messages will be transmitted through MQ and NCS would write the message to MQ.
3.	Real time handling of SWIFT acknowledgement.	NCS will process the incoming standard SWIFT Acknowledgement received from

		Mercator.
--	--	-----------

Description

It is required that SWIFT messages should be sent and received using MQ Series for Alliance. However NCS will not be exchanging the messages directly with SWIFT alliance terminal but with an intermediate communication engine (Mercator) in between.

5.2.3 Requirement FOR Using BICPLUS data in NCS

The NCS should be able to use the SWIFT address specified in the BIC+ database and should no longer accept swift address from manual input. SWIFT provides the address information about the institutions registered with it in one of its directory services component called BIC+ database. Few times in a year SWIFT releases this address information in the form of MS Access database.

The system should load BICPLUS data into NCS and maintain links between NCS address information and the BICPLUS address information.

Assumptions

During the initial setup of the system, the SWIFT address already present in the system will be synchronized with the data in BICPLUS. So the SWIFT address already present in the system is assumed to be correct.

S. No.	Requirement	Remarks
1	Load BICPLUS data for the BIC+ Keys already installed in NCS and for the rest of BIC+ keys.	
2	Initial setup of populating the BIC+ Keys for all the addresses that have swift address specified.	

6 Implementation

Once the requirements of certification were clear, each of them was suitably distributed for further analysis and coding among programmers. From the MT564 message for analysis and implementation. The next section describes this work.

6.1 MT564

This message is sent by an account servicer (account servicing institution) to an account owner or its designated agent. The account servicer may be a local agent (sub-custodian) acting on behalf of their global custodian customer, or a custodian acting on behalf of an investment management institution or a broker/dealer.

This message is used to provide an account owner with the details of a corporate action event along with the possible elections or choices available to the account owner. It can be initially sent as a preliminary advice and subsequently replaced by another MT 564 with complete or confirmed information.

This message will also be used to provide the account owner with details of the impact a corporate action event will have on a safekeeping or cash account, e.g., entitlement calculation.

This message may also be used to:

- Request the cancellation of a previously sent corporate action notification
- Re-send a corporate action notification previously sent by other means, eg, fax, telex
- Provide a third party with a copy of the message.

6.2 Identifying the Basic Problem

Fidelity Investments has two external accounting systems with which NCS needs to communicate. These External accounting systems were until now using incoming feeds from various sources about a corporate action. In order to couple NCS with the preinstalled accounting systems Fidelity Investments requires that the notification of a corporate action be received from NCS only. This will eliminate the need to send the feed to two sources i.e. first to NCS and then to the two accounting systems.

6.3 Gap Specification

- Currently Fidelity is using different incoming feeds for their fund accounting systems. As Fidelity is using NCS as a corporate Action system, CA feed will be sent from NCS only. There are two accounting systems. These two systems will use SWIFT feed in the form of message MT564 for CA announcement. For sending CA, CA status should be authorized and stage should not be in captured or verified stage. No late announcement for FAS will be handled by NCS. Ex: If user creates an event which has today's Inform accounting system date or inform accounting system date as past date, this CA will not be announced to FAS and user will create record manually in FAS.
- To create outgoing corporate action feed MT564 to be send to the Fund Accounting System.
- Although the security identification is different for both accounting systems, the same message file will be created for both systems by including multiple numbering schemes on each record. The SWIFT security identifier tag will contain following security numbering schemes:
FMR CUSIP
SEDOL
CUSIP
Canadian dual listed line
- With these numbering schemes both the current accounting systems will be able to identify the security id in their systems.
- After generating the MT564 message for a corporate action, an indicator in NCS will be updated that message has been sent. And this indicator will appear in CA List for each record.
- On the "beginning of day" of inform accounting system date, the message file containing MT564 will be created. This file will contain the swift message.

Recommended Solution

There is a date in base record named "Inform Accounting system date" which will be used for sending the message to accounting system. On the "beginning of day" of inform accounting system date, the message file containing MT564 will be created. This file will contain the swift message. Except for BLOCK-4 the header and trailer details of the SWIFT message will be as follows and these will be static:

Header:

```
{1:F01FMTCUS30AXXX0000123456}{2:O5641201040302FIDQUS30AXXX00001234560403021201N}{3:{108:XYZ1234567890123}}
```

Trailer:

```
{5:{MAC:123456FA}{CHK:123456789ABC}}
```

Apart from the swift message header and trailer there will be one more header and trailer that will be for the file containing messages. Format of header records and trailer records formatted as follows:

The following formats detail the "IMAHDR" (I'm a header) and "IMATRL" (I'm a trailer) record formats.

These records provide for audit and validation purposes. It communicates to the receiving application that the file is the correct file that is expected by providing the accepting application a means to ensure that the file is complete, and that the record match the number of records sent.

Field Name	Size	Description
Record Identifier	X(6)	Identifies Record Type. Values include 'IMAHDR' for header records and 'IMATRL' for trailer records.
Filler	X(1)	" (One blank space)
Source Node	X(8)	Identifies sending machine. For both records this will be set to 'NCS' with trailing spaces.
Filler	X(1)	" (One blank space)
File Identifier	X(8)	FMR file identifier. This will be set to 'SWIFT564'. This parameter will be site dependent.
Filler	X(1)	" (One blank space)
Business Date	X(8)	File Business Date in CCYYMMDD format.
Filler	X(1)	" (One blank space)
Creation Date	X(8)	File Creation Date in CCYYMMDD format.
Filler	X(1)	" (One blank space)
Create Time	X(6)	File Creation Time in HHMMSS format.
Filler	X(1)	" (One blank space)
Record Count	9(15)	On the trailer record, this field will be populated with the number of Detail records in the file not including header and trailer. The field will be zero filled on the header record.

Table 6.1 Header Trailer Definitions

No late announcement for FAS will be handled by NCS. If user creates an event which has today's Inform accounting system date or Inform accounting system date as past date, this CA will not be announced to FAS and user will create record manually in FAS.

For sending CA, CA status should be authorized and stage should not be in captured or verified stage.

Although the security identification is different for both accounting systems, the same message file will be created for both systems by including multiple numbering schemes on each record. The SWIFT security identifier tag will contain following security numbering schemes:

- FMR CUSIP
- SEDOL
- CUSIP
- Canadian dual listed line

With these numbering schemes both the current accounting systems will be able to identify the security id in their systems.

There will be a screen for each NCS instance for user to decide the CA types, which need to be sent to accounting system(s).

After generating the MT564 message for a corporate action, an indicator in NCS will be updated that message has been sent. And this indicator will appear in CA List for each record.

In order to generate MT564 for fund accounting system it is imperative to map mandatory and optional sequences in accordance to business rules and clients requirements. Following table presents a detailed mapping of individual tags in MT564 such that it can be used to send Bank notification C)21.

M/O	TAG	Field name	Qualifiers Available	Processing Logic
Mandatory Sequence A General Information				
M	16R	Start of Block	GENL	This is start of block indicator, not need to be processed

→

M	20C	Reference	CORP and SEME Qualifiers will be received in Message	NCS will use both the qualifier. CORP: NCS will generate with NCS CA Id SEME: a unique transaction reference number will be used.
---	-----	-----------	------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------

--

M	23G	Function of the Message	NEWM: Announcement message sent for the first time REPC: Replaces a previously sent notification that was designated as complete REPL : Replaces a previously sent notification that was designated as preliminary	To announce the feed to accounting system , only NEWM will be used.
---	-----	-------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------

→

M	22F	Indicator	CAEV CAMV	For: Bonus/Spin Off/Stock Split/Stock Dividend If base security and disbursed security are different SWIFT CODE WORD: SOFF If base security and disbursed security are different If (disbursed quantity/base quantity) >= 0.20 SWIFT CODE WORD: SPLF If (disbursed quantity/base quantity) < 0.20 SWIFT CODE WORD: DVSE Valid Values are CHOS – Mandatory CA with choice – will not be populated by NCS. MAND – Mandatory CA VOLU – Voluntary It would be used to identify if corporate action is Mandatory or Voluntary.
---	-----	-----------	------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

--|

O	98a	Date/Time	Recommend option A: PREP//<date>	NCS Will Populate date on which message was generated. NCS can support both options A and C. Currently NCS is populating option A.
M	25D	Status Code	PROC//PREU PROC//PREC PROC//COMP	NCS is generating MT564 on Info accounting date in NCS and on that date the CA supposed to be complete. So COMP will be used.

→ **Repetitive Optional Subsequence A1 Linkages**
(This is a optional link sequence and will not be populated)

M	16R	Start of Block	LINK	NCS will not populate this field.
O	22F	Indicator	LINK//AFTE LINK//BEFO LINK//WITH LINK//INFO	NCS will not populate this field.
O	13A	Number Identification	LINK//<Swift ID>	NCS will not populate this field.
M	20C	Reference	CORP, PREV, RELA	NCS will not populate this field.
M	16S	End of Block	LINK	NCS will not populate this field.

--| End of Subsequence A1 Linkages

M	16S	End of Block	GENL	This is end of block indicator.
---	-----	--------------	------	---------------------------------

End of Sequence A General Information

Table 6.2 Snapshot Mapping of SWIFT Message MT564 to NCS

6.4 DESIGN

Conceptualization

The purpose of the conceptualization stage is to formulate the idea for a system and an outline of its external requirements and success factors.. The product of the conceptualization stage will be a Use-Case Model. A Use-Case Model consists of a UML Use-Case Diagram, coupled with descriptions for each use case identified on the diagram. Use-case descriptions can be documented in text, or through a use-case template, or with activity diagrams, or using some combination of these methods. Use

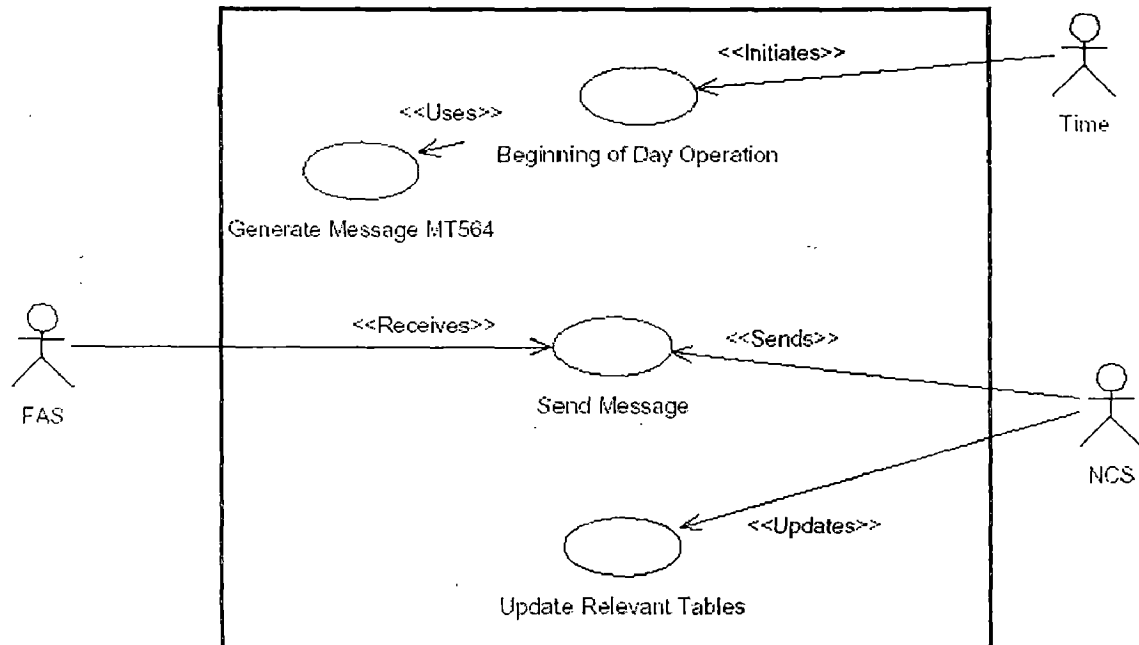


Figure 6.1 Use Case

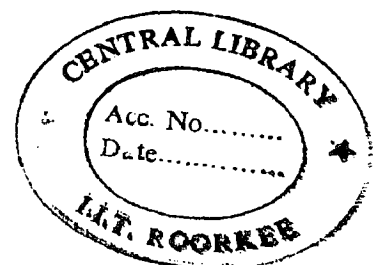
Use Case Name and Number	Beginning Of day operation 1.0
Summary and Goals	This use case finds the eligible corporate actions for which a fund accounting notification needs to be generated.
Author	Rahul.
Status	The use case has been checked with scenario walk-throughs, and tested against implementation.
Initiating Actor	Time
Involved Actors	NCS-Fidelity
Preconditions	It should be the time set for running the Beginning of day operation (generally at 2.00 am but can be set)
Triggering Event	-
Description of Main Success Path	The beginning of day operation carries out the following activities. It iterates through the corporate action list to find out the corporate actions eligible for which the notification would be sent as MT564 message. To find the eligibility it checks whether the CA is in authorized state , if it is then it checks whether the date on which notification is to be sent to FAS is the same as today and finally it ensures that for that particular CA type the indicator that FAS notification is to be sent is 'on'.
Postconditions Upon Success	For each CA found eligible an MT564 is generated
Postconditions Upon Failure	If any of the three checks for eligibility fail beginning of day operation moves to the next CA record.

Table 6.3 Use Case Description

Use Case Name and Number	Generate MT564 2.0
Summary and Goals	This use case enables generation of MT564
Author	Rahul
Status	The use case has been checked with scenario walk-throughs, and tested against implementation.
Initiating Actor	-
Involved Actors	NCS-Fidelity
Preconditions	Eligibility for a particular corporate Action has been determined
Triggering Event	Beginning of day operation
Description of Main Success Path	<p>MT564 is essentially a message, which informs FAS about a corporate action. The message has a series of mandatory and optional subsequences, which need to be populated. The data that is required to be put in each of the tags is picked up from the tables in the database on basis of the corporate action id, which is unique for a particular corporate action. After populating the mandatory and optional sequences of the message it is required that the message be in conformance to the SWIFT guidelines. In order to ensure this the formatter is called which performs the following:</p> <ol style="list-style-type: none"> 1. Field formatting as removing trailing or leading spaces from strings and removing leading zeroes in a numeric field or Carrying out date format conversions. 2. The formatter also takes as input a set of rules for a particular message which defines the format which each field will take, the formatter converts the message structure it receives to this form.
Postconditions Upon Success	A SWIFT standards compliant MT564
Postconditions Upon Failure	Error control is called

Table 6.4 – Use Case Description

510989.



Use Case Name and Number	Send MT564 3.0
Summary and Goals	This use case sends the generated MT564
Author	Rahul
Status	The use case has been checked with scenario walk-throughs, and tested against implementation.
Initiating Actor	-
Involved Actors	NCS-Fidelity, FAS
Preconditions	The formatter should have generated the complete message
Triggering Event	Generation of MT564
Description of Main Success Path	The Message which has been generated as per guidelines is to be sent to the fund accounting system. Only a part of this task is carried out in the system whereby it is saved in the FAS_SOM and FAS_SOT tables.
Postconditions Upon Success	Message successfully written at the specified tables
Postconditions Upon Failure	FAS_SOM,FAS_SOT not populated

Table 6.5 – Use Case Description

Use Case Name and Number	Updation of Tables 4.0
Summary and Goals	This use case enables table updation
Author	Rahul.
Status	The use case has been checked with scenario walk-throughs, and tested against implementation.
Initiating Actor	-
Involved Actors	NCS-Fidelity
Preconditions	The System should have populated FAS_SOM,FAS_SOT tables
Triggering Event	-
Description of Main Success Path	This use case enables the updation of the CAA_CA table whereby it sets the status of a corporate action that a notification has been sent to the fund accounting system for that specific corporate action..
Postconditions Upon Success	Successful updation
Postconditions Upon Failure	Error control routines.

Table 6.6 Use Case Description

6.5 Behavioral Modeling

The purpose of behavioral modeling is to understand, develop, and communicate system requirements, focusing on the dynamic relationships among the objects in the information domain. Particular emphasis is placed on relating use cases to the behavior of the system components.

27

27

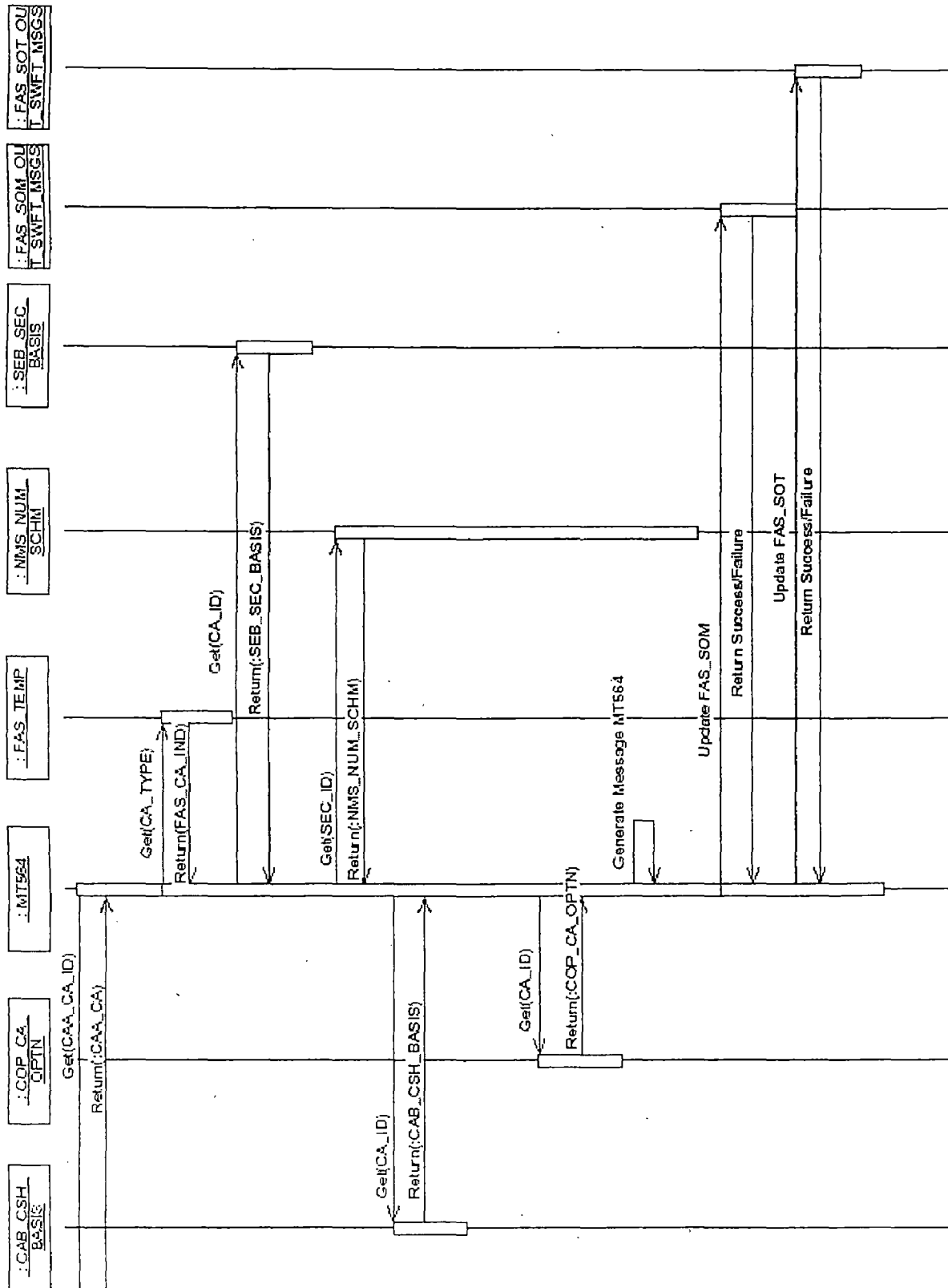


Figure 6.2 Sequence Diagram to GenerateMT564



6.6 Structural Modeling

The purpose of structural modeling is to understand, develop, and communicate system requirements, focusing on the components (that is, classes and objects) of the problem domain and the static relationships between those components. The product of the structural modeling stage is a UML Class Diagram that depicts the classes within the system being designed and the relationships between those classes.

Figure 6.3 shows the Class Diagram for the system under consideration.

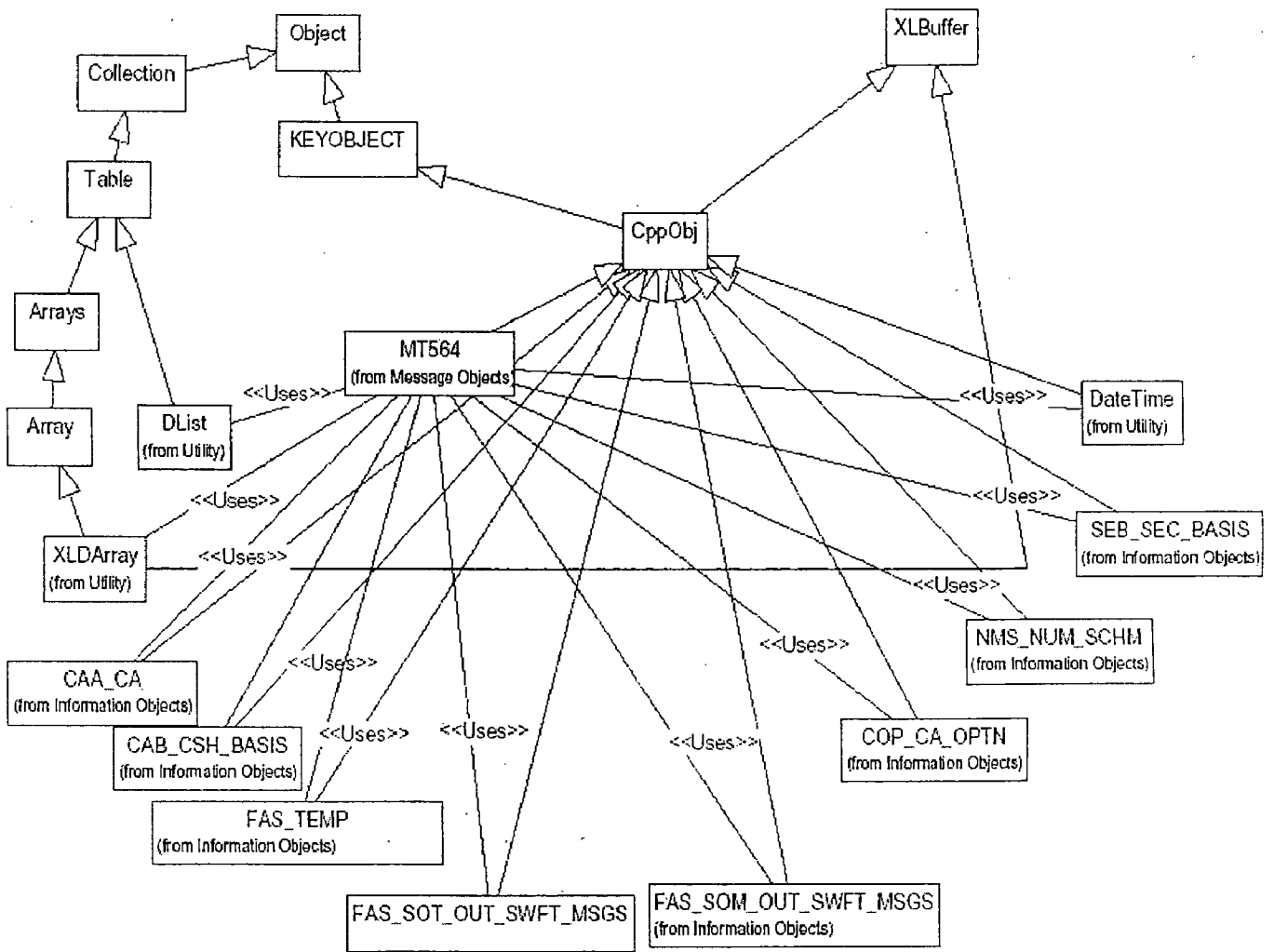


Figure 6-3 Class Diagram

6.7 Detailed Design

The purpose of detailed design is to understand, develop, and communicate the details that define how to implement the requirements, as documented in the previous stages of the design process. The major product is a refinement of the Class Diagram to document (class description) the operations and parameters for each class and to specify the visibility of each operation within the class; where applicable the visibility of the classes themselves can also be defined.

A brief Class Description follows.

Class Name	Object
Parent Class	None
Base Class	None
<p>This class is an abstract class intended to be used as the base for all objects in the associations. It would not contain any members, and serves only as a common "peg" to hang other classes from. It defines an abstract function to provide for equality of two classes. It also contains a Type Function to determine the type of the object. This function must be overridden by every derived class, which wants to be known as an individual class in its own right. Also equality functions can use this class to determine whether they are being compared with an object of the same class.</p>	
Constructor	Object () { }
Destructor	Virtual ~ Object () { }
Equality Operator	Virtual int operator == (const Object &) = 0; .
Type Enquiry Function	Virtual char *Type () const { return "Object"; }
Memory Management	<p><i>Memory Management operators new and delete overloaded.</i> Static void * operator new(size_t); Static void operator delete(void *);</p>

Table 6.7 Detailed Design

Class Name	Collection
Parent Class	Object
Base Class	Object
A generic collection class, which models other collection classes. The functionalities provided are for inserting, searching and removing elements of type Object. It is an object itself, as tables are but special objects. The difference between Table and Collection class is that – collection provides the method common in table and the hashtable (not shown in the hierarchy). But the table class gives the remaining methods needed for other data structures mentioned above.	
Constructor	Collection (enum TblType t = CP_PERS);
Destructor	-
Insertion	Appends an object to the collection Virtual int Insert (Object *) = 0;
Removal Functions	Delete an object from an collection Virtual int Delete (Object *) = 0; Virtual int Delete () = 0;
Search Function	Searches for an object in the collection Virtual Object * Search (Object *) = 0;
Enquiry Functions	<i>Return the number of elements in the collection and first and the last element of the collection.</i> Inline int nElem () ; Virtual Object *First () = 0; Virtual Object *Last () = 0;
Equality Operator	Virtual int operator ==(const Object &o) ;
Type Enquiry Function	Virtual char *Type () const ;
Class Name	KeyObject
Parent Class	Object
Base Class	Object
This class inherits from Object and is also an abstract class, but the only difference is that all key objects are also forced to provide a function to return a hash value of the object.	
Constructor	KeyObject () { }
Destructor	Virtual ~KeyObject () { }
Hashing Function	Virtual unsigned HashValue () const = 0 ;
Type Enquiry Function	Virtual char *Type () const { return "KeyObject"; }

Table 6.8 Detailed Design

6.8 Coding

For SWIFT message coding the Standard Release Guide plays an important role. Coding the processing logic involves creating following coding units.

Message files: these include the structure, bitmap and header file for a specific message

PSO: Period specific operation is a program that runs at periodic intervals for processing the batched messages

Service: if the message handling/generation is being done by GUI then services are written.

Function: it is responsible for generation the actual message

DM: Data manager are required for fetching the data from the data base tables for processing by function.

Service: If the generation of message is initiated from gui them a service needs to be coded for this purpose

MO: Message Objects are used while fetching data from several tables using an operation like join etc.

6.8.1 Semantics of SWIFT messages

SRG identifies each message with following attributes:

Status	Tag	Qualifier	Generic Field Name	Detailed Field Name	Content/Options	No.
--------	-----	-----------	--------------------	---------------------	-----------------	-----

- Identification number, like MT 564
- scope which defines its impact, parties involved and usages
- Format specification
- It is a table giving details about a specific message under the following columns
- network validated rules
- usage rules
- field specifications – format, presence(optional/mandatory), definition ,qualifier, code

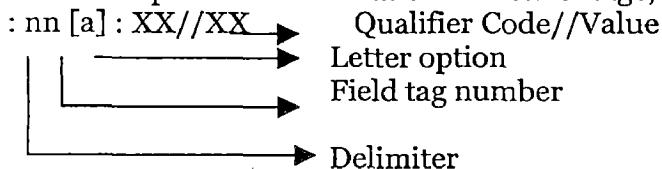
A SWIFT message consists of fields grouped in sequences/subsequences.

6.8.2 Fields

A typical SWIFT message is a flat file consisting of a sequence of fields. A small snippet is shown below:

```
: 16R: BENODET
: 95p:: OWND//BNPAFRPPXX
: 70E:: OWND//ADDITIONAL SHARES
: 16S: BENODET
```

Each field provides information in form of tags, structured as shown:



Qualifiers

Each generic class of information has its own field, plus a descriptive list of codes or qualifiers, which identify the specific type of information within the field. Qualifiers allow for identification of the type of data, which is same across all new security messages

:98A::COUP will always be next coupon date

6.8.3 Field Structure Rules

There are several rules which must be followed when structuring fields.

1. Each field is identified by a tag which consists of two digits, or two digits followed by a letter option.

2. Field structure consists of a colon ':', followed by a tag, followed by a colon ':' and the field content.

7. Field content may be composed of one or several subfields.

When subfields appear on separate lines, the Carriage Return, Line Feed (CrLf), which is not included in the number of characters for the length of the subfield, serves as the subfield separator.

subfields may themselves be of fixed or variable length

the order of subfields is fixed

when necessary, subfields are separated by special symbols, eg, '/' or '/'/'

9. Restrictions on the length of field or subfield content:

nn Maximum length n Digits only

nn! Fixed length n Alphabetic letters, upper case only

nn-nn Minimum and maximum length

c Alphabetic letters (upper case) and digits only

nn*nn Maximum number of lines times maximum line length

h Hexadecimal letters A through F (upper case) and digits only

x Any character of the permitted character set upper and lower case

y Any character of the EDIFACT level A character set as defined in ISO 9735

z Any character as defined by the Information Service

e Blank space

Examples

2n = up to 2 digits

3!a = always 3 letters

4*35x = up to 4 lines of up to 35 characters each

16-64h = at least 16 and up to 64 hexadecimal digits

10. In some messages, the field specifications may indicate specific characters, or sets of characters, for inclusion in the text of the field. These take the following forms:

a) codes, e.g., AMEND, TRF or o8

b) slash '/' or double slash '//'

c) slash or double slash followed by a code, e.g., //CH or /FIXED

d) slash followed by a code and another slash, e.g., /REC/

Note: All codes must be in upper-case alphabetic characters. When codes contain a mix of alpha and numeric characters, the alpha character must also be in uppercase.

6.8.4 Sequences/subsequences

Such fields are grouped in structures known as sequences. Each sequence is made up of a start block field one or more subsequences or fields an end of block field. A sequence is named like A, B, C etc. A sequence may further consist of subsequences numbered A1, B2 etc. a subsequence has exactly same structure as sequence only that it is nested and may be repetitive

Each sequence will have an option: mandatory or optional.

A general information (mandatory)

Includes message reference

Function of the message

Preparation date and time of instruction

A1 linkages

References a related instruction or provides a collective reference identifying a set of instruction

B trade details (mandatory)

Identification of the financial instruments

Date and time of the settlement

Price and currency of the deal

B1 financial instrument attributes

Provides additional info about financial instrument

C financial instrument /account (mandatory/repetitive)

Specifies the safekeeping account from which the security needs to be debited. Includes quantity of security and amount to be settled

D repo details (optional)

When the instruction is part of the repo transaction, this sequence is used to identify the details. Includes the interest rate and repurchase amount

E settlement details (mandatory)

Gives type of settlement, cash, parties involved and amounts

E1 settlement parties (mandatory)

Identifies parties involved in receiving and giving the financial instrument

E2 cash parties (optional)

Identifies parties of the cash side of the trade

E3 amounts (mandatory)

Includes the cash amount and the settlement amount relevant to the details

F other parties

Identifies relevant parties not directly involved in the trade or settlement such as trade regulators

6.8.5 Coding of Message bitmaps and headers

Generation of SWIFT message structures using message bitmaps and message headers involves coding following files.

Header file:

this gives the declaration of structures associated with the message in C – like syntax as shown

```
typedef struct
{
    STRUCT_MT569_SEQA seqa; //STRUCT_MT569_SEQA
    STRUCT_MT569_SEQB seqb; //STRUCT_MT569_SEQB
    STRUCT_MT569_SEQC seqc; //STRUCT_MT569_SEQC
}STRUCT_MT569;
.....
.....
};
```

Rest of the structures must be declared earlier just as in C. This determines the type of individual tag types – struct, char etc. This definition is followed through in other two files.

Structure file

shows the tag declarations as given below

```
static struct_field FldsOf_MT569_SC[] =
{
    { "A", "STRUCT_MT569_SEQA", "Start Of Sequence A", 0, 1, 0},
    { "16R", "char", "Block Start A", 5, 0, 0},
    { "28E", "char", "Page Number", 6, 0, 0},
    { "28E", "char", "Continuation Indicator", 5, 0, 0},
    { "20C", "FldsOf_20C_SC", "Link start 20C", 0, 0, 1},
    ....
    ....
};
```

It is used as an aid for displaying the message list in the gui interface

Map file

consists numeric coded declarations as shown below:

```
static STRUCT_FIELDS FldsOf_MT569[] =
{ "16R", 5, {0,0,0,0,0,0,0,0,0,0,13}, 0, 0},
{ "28E", 6, {0,0,0,0,0,0,0,0,0,0,0}, 0, 0},
{ "28E", 5, {1,0,2,0,0,0,0,0,0,0}, 0, 0},
...
...
};
```

it is sort of rule book that consists of all the possible value that a qualifier option can take. Each of these files is hand coded using the PDF. Once these are ready they are tested using a de-formatted message instance.

7 Testing

The general testing paradigm employed at TCS follows a step by step methodology wherein the first step that's is carried out is at the individual level wherein unit test specifications are prepared where the person developing the module states the test conditions and the expected results. Thereafter unit testing is carried out in accordance with the specifications. Further on a system test plan is prepared which enables integration testing of the individual units developed after integration with the GUI. On successful completion of the entire system test the next step carried out is Pre acceptance testing wherein the client provides test cases with sample data and functional testing is carried out. A functional expert carries out this phase of testing. The next phase of testing is the User acceptance testing wherein the actual user is asked to perform the testing of the system and results documented. After unit testing each stage entails carrying out of performance and regression testing.

The following is the unit test specification that has been prepared by me for carrying out unit testing for generation of SWIFT MT 564.

SERIAL NUMBER	TEST CONDITIONS	ACTUAL RESULT	REASON IF NOT TESTED
1	Check if a unique Transaction Reference Number is generated by NCS in Tag 20C. Store transaction no into 20C	A unique Transaction Reference Number is populated in Tag20C/SEME in MT564	
2.	Check if correct Ca Id is populated in the field of 20C in SEQ A. Store ca id into 20C	Correct Ca Id is populated in the field of 20C/CORP	
3.	If a new corporate action is announced 23G is populated with NEWM	23G is populated with NEWM	
4.	Store CAA_CA_TYP into the Field 22F CAEV of Sequence A.	22F is populated with CAEV and Ca Type	
5.	If CA is optional populate VOLU in CAMV field of 22F	22F is populated with CAMV and optional indicator VOLU	
	If CA is mandatory populate MAND in CAMV field of 22F	22F is populated with CAMV and mandatory	

		indicator MAND	
6.	Check if 98a is populated with tag PREP and the date can be identified By the system	98a is populated with PREP and the date is in the specified format	
7.	Check if 25D is populated with PROC .	25D is populated with PROC	
8.	If complete CA event details is obtained Set 25D with COMP .	25D is populated with PROC and COMP	
9.	If CA details are incomplete but Ca event is confirmed from second source set 25D PREC	25D is populated with PROC and PREC	
10.	If CA details are incomplete and Ca event is not confirmed from second source when mesg is sent set 25D PREU	25D is populated with PROC and PREU	
11.	Use standard dm to get CAA_SEC_ID from the caa table Store NMS_NMBRNG_SCHM into the Field 35B of Seq B	35B is populated with Identification and description of security	
12.	Populate SAFE in the tag 97a.In case of option C populate tag with GENR	93a is populated with SAFE and in case of option C with GENR also	
13.	Populate 98a with EX DATE if CAA_ENTTLMNT_DT is present	98a is populated with Ex Date	
14.	Populate 98a with RECIEPT DATE if CAA_REC_DT is specified	98a is populated with RDTE	
15.	Populate 98a with PAY DATE if CAA_PAY_DT is specified	98a is populated with PAY DATE	
16.	Populate 92a with ADEX if CA option is cash & stock or stock and no debit security basis is defined.	92a is populated with ADEX if condition holds	
17.	Populate 92a with GRSS if CA option is cash & stock or cash and	92a is populated with GRSS if condition holds	

18.	Populate EXER/ACTU in the tag 90a. In case of option B currency code and exercise price per base security is populated	90a is populated with EXER/ACTU	
19.	13A will be populated with option sequence number. It is populated with CAON to identify option number	13A will be populated with CAON and Number Id	
20.	11A will be populated with OPTN and valid Currency Code	11A is populated with OPTN	
21.	17B is populated with DFLT and if CAON is 1 then indicator is Y else N	17B is populated with CAON and Y/N	
22.	Use standard dm to get CAA_SEC_ID from the caa table Store NMS_NMBRNG_SCHM into the Field 35B of Seq B	35B is populated with Identification and description of security	
23.	Populate 92a with ADEX if CA option is cash & stock or stock and no debit security basis is defined.	92a is populated with ADEX if condition holds	
24.	Populate EXER/ACTU in the tag 90a. In case of option B currency code and exercise price per base security is populated	90a is populated with EXER/ACTU	
25.	If 22H is populated with CRDB then indicator has to be specified ie DEBIT or CREDIT	22H is populated with CRDB and CRED/DEBET	
26.	Use standard dm to get CAA_SEC_ID from the caa table Store NMS_NMBRNG_SCHM into the Field 35B of Seq B	35B is populated with Identification and description of security	
	Populate 92a with ADEX if CA option is cash & stock or stock	92a is populated with ADEX if	

27	and no debit security basis is defined.	condition holds	
28	Populate 92a with GRSS if CA option is cash & stock or cash and	92a is populated with GRSS if condition holds	

Table 7.1 Testing

The entire client specification, which required the generation of notification of a corporate action to fund accounting system, is under testing stage. The specific requirement, which was entrusted to me viz. the generation of notification via SWIFT MT564, has been developed. The code is running successfully and a sample message generated by the system is appended as annexure. Unit test plan for the same have been formulated. Further test cycle needs to be carried out. (Refer Appendix C)

8 Overview

In this section a brief idea is given regarding the implementation of rest of the requirements of SWIFT certification. This work was done by other programmers but is provided for sake of providing a complete picture of the work.

8.1 Data Dictionary

The communication manager component of NCS provides the required interfaces with Clients/Custodians and external systems with which the System interfaces and exchanges data. This component allows the system to receive instructions and sends output messages from/to the external entities via SWIFT and also allows the isolation of business logic from external format mapping.

The Communication Manager component of NCS is a directory driven component that supports external messages. All message structures have been defined in the Communication Manager Directory and are mapped to internal application formats for processing. ISO 15022 as a directory driven messaging format has been incorporated in the Communication Manager module of NCS.

The data dictionary has been created in-house as part of the product.

8.2 Implementation of BIC+ Data GOLD

BICPLUS tables integrated in NCS DB with support for Country codes and Currency codes and a User Interface to BICPLUS data

Features:

Address Information

Complete bicplus data.

Mapping with NCS.

Search Capability.

View Capabilities.

Updates

Upload of Updated BICPLUS data.

Simultaneous status mapping between NCS & BICPLUS for following status changes

Deleted

Modified

Moved

Application of Validation and Verification to BICPLUS data

Database Design

BICPLUS Table Structures

BICPLUS has five Tables. The Structure of these tables is given below.

This table contains address information.

S.No	Field Name	Type	Length	Primary Key
	Tag	Text	2	
	Flag	Text	1	
	BIC	Text	11	
	Institution	Text	105	

Table 8.1 : Bicplus

This table contains Country Information.

S.No	Field Name	Type	Length
	Tag	Text	2
	Flag	Text	1
	Country Code	Text	2
	Country Name	Text	70
	Filler1	Text	255

Table 8.2 : CT

This table contains Currency Information

S.No	Field Name	Type	Length
	Tag	Text	2
	Flag	Text	1
	Currency Code	Text	3
	Currency Name	Text	70
	Fractional Digit	Text	1
	Country Code	Text	2
	Country Name	Text	70
	Filler1	Text	255
	Filler2	Text	255
	Filler3	Text	255
	Filler4	Text	255
	Filler5	Text	255

Table 8.3 : CU

This table contains header information

S.No	Field Name	Type	Length
	Tag	Text	2
	Rebuild Date	Text	12
	Yyyymmdd	Text	8
	Version	Text	8

	Nnn	Text	3
	Filler1	Text	255
	Filler2	Text	255
	Filler3	Text	255
	Filler4	Text	255
	Filler5	Text	255

Table 8.4 : HD

This table contains release and version information

S.No	Field Name	Type	Length
	Bic+Data	Text	10
	Release Number	Text	10
6	Date	Date/Time	
	Version	Text	4

Table 8.5 : Release

8.3 Implementation of MQSA

NCS supports interface to SWIFT Alliance through MQSA, with a Mercator communication engine in between for security purpose.

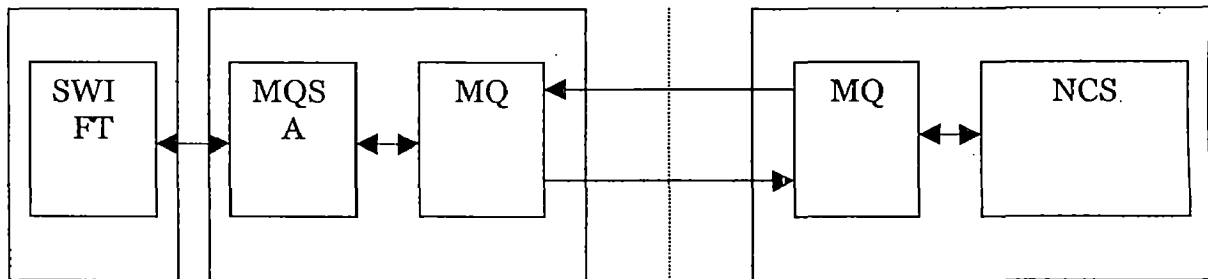


Figure 8.1 MQSA Interface Environment

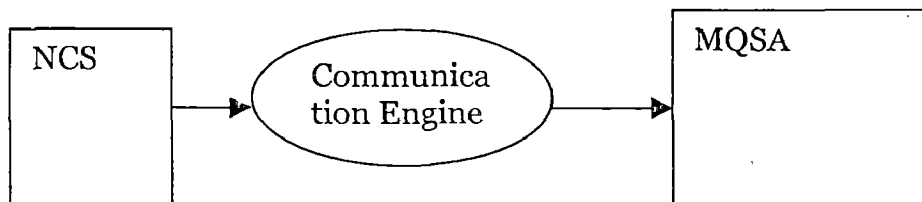


Figure 8.2 Modified implementation of MQSA in NCS

Features

- For each Local Queue defined in MQSA to receive messages from NCS , a remote queue is defined in NCS.
- For each Remote Queue defined in MQSA to send messages to NCS , a local queue is defined in NCS.
- Each Local queue is associated with a process with unique "Application ID" which determines from which queue messages have to be processed.
- Each stream of messaging is identified in NCS by an unique "Application ID".
- A mapping between "Application ID" and the application queue is maintained in NCS
- Every swift message that is sent to MQSA , is sent with "Request for Reply"
- Reply Reports Requested include
 - Positive Acknowledgment (MQRO_PAN)
 - Negative Acknowledgment (MQRO_NAN)
- Following negative acknowledgment are processed
 - Message Failed Validation in MQSA
 - Routing Failed
 - Message not Stored in Swift Alliance

- In NCS , every swift message is identified by a 16-digit random number called “Transaction Reference Number”.
- Message Id field set to “Transaction Reference Number” of the message descriptor of the message sent to MQSA.
- Application data of Transmission Report reconciliation message is looked for field 451 for positive or negative acknowledgment by S.W.I.F.T.
- All Queues defined under MQ manager on NCS side are configured to be persistent so that messages are not lost at restart or failure of MQ Manager.
- MQSeries will participate in the distributed environment as a resource managed by CICS which acts as external resource manager. This implies that all MQ queue operations are transactional.

Additional features:
File transfer

Summary

NCS now supports all the messages for the SWIFTReady Gold Custodian space. The refinement of the Business processing for these messages (e.g. some of the messages listed above) according to the Client specific requirements is easily customizable for the messages currently not being used by existing NCS clients. The message handler for NCS is a directory driven component, which supports external messages for different formats/vendors . The specific message structure is defined in the Communication Manager Directory and subsequently translated to the internal application formats or external formats for business processing purposes.

8.4 Work profile

This dissertation involved three domains of information – the financial services, NCS product and SWIFT standard. The initial period was spent in gaining the working knowledge of NCS functionality vis-à-vis the financial service it provides i.e. custodial banking. Here domain was restricted to NCS-CA module being customized for Fidelity Investments USA. Major activities carried out were - running the test cycles based on TCS internal test procedures, debugging front-end errors to back-end coding etc. A major effort was concentrated over SWIFT communication interface. Initially it handled a subset of SWIFT standard messages. Since NCS is aiming for SWIFTReady GOLD label, a gap analysis was done to enumerate the requirements.

Each of these requirements were then distributed over to programmers for implementation. Here I handled the MT564 message.

An effort was made to use UML in the analysis and design part. Tools like Rational Rose were useful not only in creating the design but also in reverse engineering a part of design from existing code. Otherwise most of the development process was as specified in TCS internal development manual.

It was noticed that much of the effort in coding the SWIFT messages is based on text manipulation using the Standard Release Guide and the existing code. Here it was suggested that a part of the process be automated using PERL scripts.

An important part of the project was to collaborate with message developers to determine the status of each message implementation as required by the SWIFT standard.

The following chart shows the work profile vis-à-vis the software development life cycle. This also shows the work done beyond the title of the project but necessary to gain a part of understanding of the work process. Appendix B gives the development process followed.

Abbreviations:

SS : system specification

UTS : unit test specification

CMR : change management request

STS : System test specification

CA : corporate action

UTM : user training manual

	Contribution	Module
Analysis	Modifying Impact document	<ol style="list-style-type: none"> 1. Requirements for SWIFT certification 2. Document management system
Design	SS, UTS, CMR, Workplan, STS, DB request	<ol style="list-style-type: none"> 1. Swift certification 2. Document management system 3. Develop spec to test messages for compliance
Coding & Debugging	Swift message header, structure, bitmaps Message control module, Document Management System	<ol style="list-style-type: none"> 1. Coding of new message bitmaps and headers according to SRG2002 For Swift messages MT564, MT565, MT566, MT568 2. Upgrading the code of existing message bitmaps and headers from previous SRG to SRG2002 3. Coding of sample SWIFT message for testing message bitmaps and message headers
Testing	Test cycles	<ol style="list-style-type: none"> 1. Testing CA cycles for Cash Dividend, Optional Merger, Spin off, Open offer, Doc Mgmt GUI , 2. Swift Message control module 3. Testing of messages generated by NCS-SWIFT module 4. Testing services (QPP/PPP) and Data Managers (ProC) 5. Testing of message structures using provided parsers 6. use case testing with scenario walk-throughs, and against implementation
Maintenance	Help Docs, UTM, Review document	<ol style="list-style-type: none"> 1. Swift messages 2. Document Management System

9 Conclusion

SWIFT is emerging as the message communication standard in the financial sector. Getting a hands-on experience is a crucial advantage. The main thrust of the work was to design a process of development and documentation that makes an existing product comply with an international standard. This report is the first step in creating an internal process for managing the development of NCS product such that it fulfills the certification requirements.

An important requirement was to document the SWIFT certification requirement so that programmers working on the project may be able to understand the significance and state of their module towards compliance. This report provides the initial mapping of the requirements to the NCS implementation.

Initially each programmer had idea of only a part of the picture. This report assembles them into a big picture. Collaborating with all individual developers was crucial part of the process.

9.1 Scope of further improvement

- Mapping of requirements to internal NCS implementation should be more detailed.
- No effort estimation or metrics calculation was done for SWIFT certification process.
- A centralized track record of the modules and the message should be kept versus their implementation status for easy visualization of status.
- UML can be used more liberally throughout the life cycle.
- More detailed analysis and design of the message processing in terms of DFDs etc. should be documented.

9/12

References

The literature survey for this project was mainly done using the web, referring to SWIFT standard reference guide and TCS internal documentation.

1. <http://www.swift.com> : The SWIFT official website
2. http://www.swift.com/resources/quickstart/qs_intro.zip
3. <http://www.swift.com/temp/40480/7560/Securities%20Market%20PracticeCAv4.pdf>
4. http://www.swift.com/temp/41752/7457/13057_SWIFT_AR_BR_thb.pdf
5. http://www.swift.com/temp/41788/7774/snna_glsnu_200201.pdf
6. http://www.swift.com/temp/5009/7407/Report_DSS_ISO15022.pdf
7. http://www.swift.com/resources/quickstart/qs_securities.zip
8. http://www.swift.com/temp/41796/7801/ISO15022_Warsaw_DH_v2.ppt
9. http://www.swift.com/temp/5809/4485/ISO15022_Q&A_Sep_v1.doc
10. <http://www.swift.com/temp/3493/1793/Solutions7.pdf>
11. <http://www.swift.com/temp/2860/1028/ISO15022thierry.pdf>
12. http://www.swift.com/temp/5009/7407/Report_DSS_ISO15022.pdf
13. http://www.swift.com/temp/3789/2062/ISO15022_readiness.pdf
14. http://www.swift.com/temp/6611/4175/FAQ_ISOXML_final.pdf
15. http://www.swift.com/temp/5829/7856/SE_Factsheets_20020507.pdf
16. <http://www.iso15022.org> : The ISO 15022 standard site
17. <http://www.iso.ch>: The official ISO website
18. Swift standard release guide 2001 and 2002 is available for download from the SWIFT website in PDF form.

TCS internal documentation has restricted distribution and is not available for general perusal.

10 Appendices

A. Glossary

Corporate Action	A corporate action is defined as any action on a security, which can result in a change in cash or security position in the system.
Counter Party	A trade is transacted by a representative Broker for an individual, or by an institution, termed as a Party. The opposite member in the trade is termed as the Counterparty or Seller/Buyer.
Custodian	Custodian is the entity that is responsible for the keeping of financial instruments.
CA Record Date or B/C Date (Book-Closure)	The published date on which a snapshot is taken of the Client Holdings, per Client Portfolio, at the end of the business day. These snapshots of the Client Holdings include settled positions, and pending positions for purchase made, CUM the entitlements associated with the CA Record Date. These positions are the basis on which entitlements are calculated for specific CAs.
CA Response Deadline Date or Client Deadline Date	The Response Deadline is the date by which all responses from Investors, relating to the CA, must be received by the Issuer / Transfer Secretary.
Eligibility	Eligibility is defined as the cash and/or security due to an individual fund as a result of a CA. e.g. If fund X has 100 shares of XYZ and the entitlement is 1 share of XYZ per 10 held, then the eligibility will be 10 shares.
Entitlement	Entitlement is the basis that is used to calculate the disbursed security and/or cash, which arises out of a CA. e.g. 1 share of XYZ is given per 10 shares held.
Ex Date	Date on or after which any trade done in the security undergoing the Corporate Action is not eligible for Corporate Action entitlement. The Ex-date determines whether trades are done on an Ex or Cum basis
Fund	A fund is collection financial instruments such as stocks, bonds, cash etc.
Paying Agent	Paying Agent is the entity responsible for disbursing/receiving the proceeds of a corporate action.
Registrar	Registrar is the entity responsible for maintaining the ownership details of financial instruments.
RVP Orders	Receive Vesus Payment
Safe Keeping Account	A safe keeping account is the NCS internal representation of a Fund.

Straight Through Processing

The system on receipt of an incoming message deformats the message and tries to create an order against it. If the deformatter does not encounter any errors the system is said to have Straight Through Processed the Message.

Trade Settlement

When a person buys or sells some securities, he is entitled to receive payments, against sales and make payments, against purchases of securities. The sequence of events that take place as a result of buying and selling of securities is collectively known as trade settlement. A trade is said to be settled if the buyer has received possession of what he has bought and the seller has received consideration for what he has sold.

B. Development process

The following Software Development Life Cycle is followed at NCS- Fidelity customization.:

- Prepare change impact analysis document
- Discussion is done on impact analysis document and a review report is prepared.
- Submit the impact analysis document to IAT.
- Identify all work items and prepare Work plan and give it to PL.
- Prepare System Specifications
- Prepare Unit test specifications
- Make required changes on items.
- Prepare folder and submit it to IAT.
- Close all the defects raised during folder review.
- Conduct unit testing, ML will verify the unit testing.
- Check in sources after approval from ML
- At IAT the IAT person does the check-out/check-in.
- Compile the items and create executable.
- Testing is conducted by IAT and Defect is raised for any error encountered.
- TM has to take appropriate action and redeliver the item to IAT.
- Carry out system testing after all customization is completed
- After the testing is over some external person does the final inspection.
- Finally items are delivered to on site.

C. Tests and Reviews

This is an overview of the test and review process

Phase	Items to be reviewed	E Estimated Effort	Type and no. of reviewers
Start-up	Incoming Items review (Gap Analysis, Interface Specifications)	3 PD	Internal - 3
	Estimate	1 PD	Internal - 1
Project Start-up	Project Plan	2 PD	Internal - 1 External - 1
	Induction manual	2 PD	Internal - 2
	Contract	3 PD	Internal - 2 External 1)
Construction	CMR Document	2 PD per CMR	Internal -2
	System Specification	0.25 PD/SS	Internal -1
	Unit Test Plan	1 PD	Internal - 1
	Unit Test Specification	0.25 PD/UTS	Internal - 1
	Traceability Matrix	1 PD	Internal - 1
	Program Code	0.25 PD /Work Item	Internal - 1
	Unit Test Result	0.25PD / UTS	Internal
	UPP (If updated)	1 PD	External - 1
System Testing	Estimate	1 PD	Internal -- 1
	System Test Plan	2 PD	Internal
	System Test specifications	10 PD	Internal
	Pre - Acceptance Test Cases from Client	5PD	Internal
	System Test Results	2 PD	Internal
	Code (if changed)	0.25 PD	Internal
	UPP (if updated)	1 PD	External
	Estimate	1 PD	Internal - 1
	Traceability Matrix(if changed)	1 PD	Internal - 1
	Installation Manual	1 PD	Internal - 1
Acceptance Test Support		2 PD	Internal -1, External - 1
	Code, Document	0.25 PD/problem report	Internal
	Test Results	0.5 PD/Problem Report	Internal
	Operations Manual	2 PD	Internal - 1, External - 1
Project	User cum Training Manual	2 PD	Internal - 1, External - 1
	Revised UPP	2 PD	Internal -1, External - 1

Wind-up			
---------	--	--	--

Table 10.2 Reviews

Phase	Items to be tested	Coverage	Type of tests
Construction & Unit Testing	Code	100% code	Unit Testing
System Testing	Code	100% functionality	System Testing

Table 10.1 Tests

The System test plan for the project will be prepared separately. This will cover, at a minimum, the following:

Types of tests

For each type of test

Testing objectives (features to be covered, features not to be covered, coverage, error detection)

Assumptions

Testing Approach

Testing Environment

Suspension / Resumption criteria

Resource requirements

Schedule for testing

Tools to be used, if any

Testing Organization, roles & responsibilities

D. NCS Technical specifications

The client workstation will be a PC-486/Pentium and the server will be a RISC-6000 machine

Operating system:

Operating system	Version
Server :AIX	4.3.3
Client Machine : Windows NT Windows	4.0 Windows 2000

System software:

System software	Version
UDB for AIX 4.3 (including XA Library)	7.2
IBM Transaction server/client for AIX 4.3(CICS/6000)	TXSeries 4.3
IBM Directory and Security Server for AIX 4.3 (DCE)	DCE 3.1
Encina Required for 2000 Web Version of NCS	4.3
C-Set++ for AIX 4.3	3.6.6
CICS Client for Windows	3.1
SQL Anywhere	6.0

Dev environment

Programming Languages:

Programming language	Version
C++	5.0.2
Pro*C	12
SQL	12

Special software:

Special software	Version
Power Builder	7.0
MS Office	Office 2000