COMPAQ

Software Product Description

PRODUCT NAME: NonStop Payments Factory **SPD 70.61.00**

PAYplus RTGS™

DESCRIPTION

RTGS is a client/server funds transfer payment system that supports payment processing, risk management and regulatory compliance for clearing systems in countries that have established a Real Time Gross Settlement (RTGS) payment network. In many countries, these RTGS networks have been built out of S.W.I.F.T. communication infrastructures and have adopted communications protocols, technologies and message type formats originally designed for the S.W.I.F.T. network. Because PAYplus RTGS is fully compatible with S.W.I.F.T. network requirements, its message formats and network interface are fully compatible with RTGS network requirements.

In addition to standard payment processing activities, PAYplus RTGS enables users to manage their financial institution's settlement account at the central bank by providing continuously updated account status, and end-of-day forecasted balance. PAYplus RTGS expedites daily settlement activities by automating the reconciliation of transactions performed by customers of the financial institution directly with the central bank. Finally PAYplus RTGS provides the information and the means to help the operations staff manage non-sufficient-funds, operational risk, settlement risk and credit risks.

- Windows-based graphical user interface

  The graphic design is easy to learn, presents essential information for rapid decision making, and provides users with a customizable workflow enabling them to maximize payment processing efficiencies.

- Computer-generated memo posting accounting entries

  PAYplus RTGS automatically creates accounting entries for all payments, including fee entries, by combining customer-mandated accounting rules with specialized entries appropriate to a particular message type and subtype. PAYplus RTGS automatically isolates account numbers and names embedded within incoming wires and matches them to existing customer account records in the customer data base to speed the process of crediting the appropriate beneficiary.

- Comprehensive security

  The architectural design of PAYplus RTGS, incorporating Windows NT, DES and RSA standards and including Smartcard and Testkey functionality. PAYplus RTGS controls and ensures a message's integrity throughout its life cycle.

- Funds management and control

  PAYplus RTGS enhances the treasury function by ensuring that sufficient funds are available to cover outgoing payments in the sender's account, the sending bank's correspondent bank account, and in the account at the Central Bank.

- Automated advice

  Advices are automatically sent to branches and customers by facsimile or letter.

- Automated fee processing

  The automated funds transfer fee calculation and collection enables financial institutions to benefit from fees that are often erroneously waived by manually intensive funds transfer systems.

- Multi-currency payments

  PAYplus RTGS tracks the level of each currency, provides two levels of conversion, and provides automatic notification when threshold levels are exceeded.

- S.W.I.F.T. Interface

  PAYplus RTGS connects to the S.W.I.F.T. payment network through the S.W.I.F.T. Alliance, connecting its clients to thousands of banks worldwide

FaxWire™

DESCRIPTION

FaxWire is a client/server system that automates the receipt and testing of funds transfer instructions transmittedby fax. Facsimile transmissions have become widely accepted as a means of transmitting payment instructions from customers to financial institutions, but the format of a facsimile message is not accessible to standard payment processing systems. Invariably, an operator must review, authenticate and re-key faxed payment order instructions. FaxWire automates these activities and creates a seamless electronic interface between incoming faxed messages and the payment processing system.

FaxWire uses optical character recognition capabilities, enhanced by a dictionary and pattern-matching algorithm, to recognize and process a faxed payment order. FaxWire maintains comprehensive security throughout the entire process by conducting an automated signature comparison and validating any PIN numbers or Test Keys attached to the fax. When FaxWire has completed its processing, payment instructions have become payment messages that are ready for immediate delivery into FEDplu$, PAY$tar, PAYplus RTGS or other payment processing system for final verification, balance checking and release.

FaxWire's automation of this manual process helps financial institutions to realize immediate cost savings and results in a significant improvement in wire processing throughput and service quality by eliminating entry errors. FaxWire enhances the security of faxed instructions through automatic verification of test keys and sight verification of signatures. FaxWire also automatically returns an advice of receipt of the fax, assuring senders that their request was received and that a response is forthcoming.

Features

- Automated Capture and conversion of incoming fax instructions
- Automated optical disk storage

  FaxWire is fully compatible with WireHouse and other optical storage systems for rapid retrieval and compliance with regulatory mandated record retention and reporting requirements. (Optional attachment)

- Advanced Artificial Intelligence algorithms that enhance optical character recognition

  In addition to a fully functional Optical Character Recognition [OCR] program, FaxWire 's special algorithms ensure that even non-standard source data are captured and converted at an exceptionally high level of recognition.

- Computerized verification of accompanying PIN Numbers and Test Keys

  FaxWire's automated PIN Number and Test Key authentication algorithms provide additional validation of the faxed instruction's integrity.

- Visual Verification of the image of each field

  FaxWire automatically separates sections of the incoming fax into discreet images and displays them next to their corresponding converted fields to assist users as they review newly processed faxed payment instructions.

- Automatic search of an electronic signature gallery

  FaxWire enables users to store digitized copies of the signatures of authorized customers. Signatures on all subsequent faxed payment instructions delivered from that customer can be automatically compared to the signature on file, and exceptions brought to the attention of the Wire Room staff.

- Immediate automated fax advices sent to branches or customers to acknowledge the fax's receipt.

Benefits

Immediate Cost Savings Realized From

- Elimination of the redundant key entry of faxed wire instructions
- Elimination of manual Test Key and PIN number verification activities
- Elimination of the need to prepare advices of receipt manually
- Elimination of the need to maintain a manual log of incoming faxed wire instructions

Increased Capacity during Peak Hours by

- Automates the creation of outgoing funds transfer messages from instructions transmitted by fax
- Automates incoming fax storage activities

Errors Reduced by automating

- The translation of faxed funds transfer instructions from original customer documents
- The comparison of message details against lookup tables
- The "field-by-field" sight verification of source image versus funds transfer instructions

Service Quality Enhanced by

- Higher throughput during peak hours
- Expedited delivery of advice of receipt
- Fast retrieval from computerized storage

WireHouse™

DESCRIPTION

WireHouse is an automated system that archives wire transaction information and provides comprehensive historical on-line research capabilities. WireHouse is based on the W.O.R.M. ("Write Once Read Many") optical storage medium, and is easily integrated into the daily operations of FEDplu$, PAYplu$ RTGS, PAY$tar, and other wire payment processing systems. WireHouse provides financial institutions with accessible but comprehensive on-line research capabilities for investigations into historical payments transaction records and supporting documents.

With WireHouse, financial institutions can keep years of payment history records on high-capacity optical storage media; there is no longer any need to sacrifice space to store bulky paper-based transaction records. WireHouse's graphical query generator provides users with access to their entire historical database using a powerful but easy-to-use research tool. Users at all skill levels can quickly and easily pinpoint transaction data. Because it reduces the time and effort spent on past payment investigations, WireHouse significantly lowers operating costs and minimizes response time to customer inquiries.

Key Features

- Automated Storage

  WireHouse automatically records incoming and outgoing payment transactions as part of a daily down load of data from FEDplu$, PAY$tar or other transaction processing system.

- Paperless filing on Optical Media

  WireHouse stores and maintains all transactions on optical media.

- Versatile Research and Investigation

  WireHouse combines a graphical query generator that combines ease-of-use with the sophisticated search tools of a relational database. This flexible tool facilitates investigations into transaction records, allowing users to target their research more efficiently and develop individualized queries.

- Comprehensive Reporting

  WireHouse provides a wide variety of standard report templates, and its flexible report generator allows users to customize their reports to adapt to new and changing reporting requirements.

- Export of Data and supporting documents to other systems

  Because WireHouse is built on an open architecture, it supports electronic data exports through ODBC interfaces. All data records and document images from archived messages can be exported to other systems for further processing, facilitating exchange of information when responding to investigations and inquiries from customers and regulators.

- Electronic Feeds

  WireHouse supports feeds from all payment networks including FedWire, S.W.I.F.T., and CHIPS.

Benefits

Cost Savings from

- Automated record keeping
- On-line research and investigation
- Space-efficient paperless filing

Improved Customer Service

- Faster response
- Enhanced information

Improved Regulatory and Audit Compliance

- Reporting capabilities designed to comply with the Federal Reserve and other Central Bank regulatory requirements

- All transactions are available for automated analysis and reporting

Recon$tar™

DESCRIPTION

Recon$tar is a client/server product that provides financial institutions with streamlined, high volume reconciliation of electronic transactions. It features a user-friendly graphical user interface and a SQL Server relational database running under Microsoft's Windows NT™.

Recon$tar is a client/server system based on Windows NT and SQL Server software that enables financial institutions to reconcile automatically incoming and outgoing wire transfer transactions. This flexible system covers a wide range of reconciliation requirements, and can be tailored to meet specific processing, accounting and reporting needs. It maintains a comprehensive audit trail of all reconciliation activities. Reconciliation is a time consuming and often error prone process. With Recon$tar, Fundtech brings to the marketplace a comprehensive and versatile automated reconciliation solution. Powerful communications capabilities provide automated linkages to internal and external systems to minimize operator intervention. Fundtech's sophisticated queue-based transaction processing provides a crisp visual display of the information most relevant to the user.

Extensive reporting capabilities allow the user to generate statistics and management reports.

Key Features

- Versatile Reconciliation

  Recon$tar reconciles various transaction types, including: Accounts Payable, Accounts Receivable, Incoming/Outgoing Wire Transfers, Settlements, EDI, Due To/Due From, Correspondent Banking and Securities. Recon$tar features automated communication linkages to internal and external systems for transaction capture.

- Scheduled Processing

  Users may schedule automated processing so that business critical information is available when needed. Automatic reconciliation process can be set to execute at pre-established times—24 hours a day, 7 days a week.

- User-Defined Transaction Matching Criteria

  Recon$tar performs matches using a variety of data fields including: Date, Amount, Position, Account and Serial Number. Reconciliation rules and write-off parameters may be defined for each account.

- Data Validation

  As transactions are loaded into the system, all data is validated and erroneous transactions are rejected.

- Duplicate Identification

  Transactions that are suspected to be duplicated are identified and routed to the Duplicate Queue for manual processing.

Benefits

- Automated Fee Assessment

  Fees can be assessed based upon predefined criteria and calculations. All fees may be assessed, held or waived by users with the appropriate privileges.

- Automated Accounting Entries

  Recon$tar can generate accounting entries for all reconciled transactions to be uploaded into corporate financial/treasury applications.

- Innovative User Interface

  Recon$tar employs an exciting Graphical User Interface to improve workflow and simplify processing of transactions.

  All data may be filtered and viewed to the user’s preference.

- Extensive Reporting

  Numerous pre-defined reports and an interface with Crystal Reports allow users to get the information required in the format preferred.

IBM® MQSeries Integrator, NEON™ MQ Integrator, NEONet, NEONaccess, Neon

DESCRIPTION

IBM MQSeries Integrator is an enterprise application integration (EAI) engine that allows applications running on widely differing hardware and software systems to exchange information, even if the applications have no knowledge of each other. IBM MQSeries Integrator was announced by IBM on 1/26/99. It is jointly owned by IBM and NEON and incorporates IBM MQSeries’ for messaging and queuing and NEON Rules and NEON Formatter for rules-based routing and message format transformation.

MQSeries Integrator:
The new product from IBM MQSeries Integrator, the new product from IBM, is a toolkit that provides the components you need to implement enterprise-wide application integration. MQSeries Integrator includes MQSeries and Rules and Formatter. IBM MQSeries Integrator delivers unrivaled performance, reliability, functionality, and flexibility. The patented breakthrough algorithms of the rules and formatter components enable high-volume, high-performance, highly scalable data connectivity across heterogeneous networks and systems, including the Internet.

Rules for MQSeries Integrator:

- The Backbone of Enterprise Application Integration Supports the creation/dispatch of multiple messages to multiple destinations from a single input message and allows different formats for each.

- Gives your MIS staff the flexibility necessary in rapidly changing environments.

- Allows you to adapt without re-engineering or reprogramming as business processes change.

- Permits new receiver applications to enter the network without opening or re-testing the sender application. Reduces messaging overhead.

Rules for MQSeries Integrator:

Intelligent Rules-Based Message Routing
Across your business computing enterprise, MQSeries Integrator allows you to map your business processing onto Rules for MQSeries Integrator (Rules for short), and as your business grows MQSeries Integrator grows with it. MQSeries Integrator lets you develop message-driven processing rules for managing data flows between message senders and receivers. You can use the Rules as part of MQSeries Integrator in combination with Formatter and MQSeries or you can use the Rules API alone (and write the code necessary, so it works with any formatter and message queuing programs). Best of all, with MQSeries Integrator, you can create dynamic routing and data interchange flows.

Sophisticated Rules functionality reduces messaging overhead with content-based message recognition and routing without architectural limits. Since the analysis of fields is rules-based and table-driven, you can add, modify, or examine new message formats and fields without changing the user's program. Rules supports the creation and dispatch of multiple messages to multiple destinations from a single input message, while allowing different formats for each.

The dynamic nature of Rules means updates can be effective immediately, staged over time, or delayed, which gives your MIS staff the flexibility necessary to control modifications in a rapidly changing environment.

Rules helps you manage data across applications, databases, platforms, and networks. No matter which types of rules are most important to your business - accounting rules, MRP rules, any kind of business or processing rules - MQSeries Integrator makes Rules the backbone of your business enterprise.

MQSeries Integrator Intelligent Routing Architecture
MQIntegrator's intelligent routing takes the responsibility for data dissemination away from individual applications and consolidates it into the manageable and scaleable configuration domain of Rules. Freeing the sending application from a commitment to support the formats expected by receivers provides flexibility to modify or add applications without changing any code. Eliminating the need to change code also removes the need to re-test applications, so the effort and time required to handle changes is greatly reduced.

MQSeries Integrator Rules: Major Features:

- Scales well over very large rules bases, offering superior throughput.
- Scales well across large, disparate applications, databases, platforms and networks.
- Performs at high transaction-processing volumes.
- Provides a GUI-based Rules configuration utility for easy maintenance.

Benefits Summary:

- Delivers rules-based, Publish-Subscribe processing at very high speeds. Protects your business investment by integrating your computing enterprise (without re-engineering).

- Delivers portability in disparate environments.
- Reduces messaging overhead with content-based message recognition and routing.
- Leverages RDBMS performance.
- Shortens learning curve and reduces training costs.
- Simplifies message exchange in complex computing environments.
- Gives MIS flexibility to manage change - rules can be effective immediately, staged over time, or delayed.

The Company’s NEONet software provides organizations with a structured software platform for the rapid and efficient development and ongoing management of application integration among disparate applications across the enterprise. The principal modules of NEONet are as follows:

Messaging and Queuing - The foundation of the NEONet architecture is the NEONet Messaging and Queing module, which provides the Company’s basic asynchronous transport vehicle. The Messaging and Queuing module sends transactions from one application to another, consisting of instructions or data between applications and databases, and provides for guaranteed delivery of each message once and only once and in the same order the messages are sent. The system uses a message queue for the sending system to ensure that the sending system can distribute a high volume of messages in real time without the need to wait for confirmation of receipt, as well as a message queue for the recipient system to ensure that the recipient system can download messages when available.

Formatter - The NEONet Formatter module can be added to the Messaging and Queuing module to provide for dynamic reformatting of data messages in real time so they may be accepted and read by multiple receiving applications in heterogeneous environments. The NEONet Formatter parses and reformats messages by translating messages among different protocols, programming languages, applications, and hardware platforms. The sending applications can issue a message in a single format, and the dynamic formatter reformats the message into the new format required by each receiving application, a function that is critical to supporting database replication and application integration.

Rules Engine - The NEONet Rules Engine module provides for the publishing of a sending application’s single message to multiple recipient applications and databases, in each case in the proper format for the designated recipient, based upon a set of user-defined business rules. Each receiving application registers or subscribes to the data generated by multiple applications which, by specifying the values of data in the transactions that are of interest, enables each subscribing application to receive only the data it requires. The Rules Engine module is based on a proprietary, high-speed, scaleable architecture that can support high numbers of business rules while continuing to provide real-time data access and distribution. This enables an organization to prescribe sophisticated rules that determine which data needs delivery to specific applications and databases and under which circumstances. These rules can be readily modified or updated as business requirements change.

The NEONadapter For S.W.I.F.T. (Society for Worldwide Interbank Financial Telecommunication) is a Financial industry product bundle that facilitates the exchange of information between a company's trading systems and the S.W.I.F.T. network. The bundle includes the NEONaccess, MQSeries Integrator and the S.W.I.F.T. Format Integration libraries.

These components can be purchased together as the NEONadapter For S.W.I.F.T. bundle, or can be individually ordered if you don't require all of them.

NEONaccess provides support for S.W.I.F.T.'s strategic CAS protocol and MQSeries Integrator, allowing S.W.I.F.T. or Telex messages to move between a queue and SWIFTAlliance Access.

The SWIFT Integration Libraries provide format definitions for S.W.I.F.T. messages. There are eleven format libraries, each supporting a particular category of messages in both the inbound and outbound directions. Each message library consists of definitions of the SWIFT message, as required by the NEONFormatter. Purchasing the library makes it unnecessary for the customer to go through the time-consuming and error-prone process of describing the SWIFT messages to the Formatter. In addition, new formats and changes to existing formats are covered under maintenance.

The format library transforms S.W.I.F.T. messages into NEOSWIFT format (or vice versa), providing improved field-level granularity.

The customer uses the NEONFormatter GUIs to describe the proprietary formats in use, and then maps fields of these formats to the NEON-provided S.W.I.F.T. formats. In most cases, the format transformations are accomplished without the need for programming.

Our "S.W.I.F.T. Gold-Ready label" status assures you that we meet all of S.W.I.F.T.'s requirements for the implementation of S.W.I.F.T.-based Straight Through Processing.

Compaq NPF QGateway

DESCRIPTION

NPF provides an enhanced queuing API through its QGateway product, for payments which adds value to NEON Messaging and Queuing (NNMQ). One of these values is increased security through the implementation of a digital signature on each message, to discourage message forgery. Another value-add is better visibility into the message queues. This is achieved by exposing key fields of each payment message, such as the value amount of the payment, as a message attribute. This value-add increases the manageability of the system, by enabling the user to answer the question, "where is my money?" This NPF API is layered over the NNMQ API, allowing greater portability of the NPF application software to different messaging and queuing middleware.

NPF includes a QGateway Service, which is implemented as a native Windows NT system service. This QGateway Service was developed as solution to the following requirements:

- The need to interface enhanced (digitally signed) NPF queues to the queues utilized by NEONet SWIFT Alliance interfaces: NEONentry and NEONaccess.
- The need to load and unload test messages into the NPF queues.

- The need to interface NPF queues to foreign queuing systems, including MQ Series and TIBCO.

While developing the functional designs for the initial NPF customers, Compaq discovered that some customized processing of messages was required beyond the message reformatting and routing capabilities of NEON. One of these requirements is the interfacing of disparate queuing systems, as performed by the Queue Gateway Service. It was determined that the QGateway Service was a convenient place to address some of these other required capabilitiesas well, through the implementation of DLLs which plug into the QGateway Service. Some of these capabilities not provided through standard NEON software are:

- Database lookup for translation of fields. The NEON formatter only performs transformations based upon imbedded tables.

- Stripping and restoration of fields. Fields which are critical to originating applications must in some cases be stripped before the RTGS system will accept them. These fields must be restored when the response is delivered to the application. This capability is delivered through the use of a SQL Server table, keyed by the message ID, holding the stripped fields.

- Matching and filtering of redundant transactions. In one RTGS implemenation, applications deliver separate payment instructions and settlement instructions. Such messages from host applications must be paired up before delivery to PAYplus. [Note that in the architecture of this particular customer, PAYplus is virtualizing the RTGS interface to the bank's host applications. PAYplus provides a common RTGS gateway to these applications, enabling an integrated view of these payments.] Such messages are held in a separate SQL Server database until they are paired.

- A user interface to allow manual release of message exceptions (such as unpaired payment instruction or settlement instruction).

All of the above requirements have been addressed through the custom development of plug-in DLLs to the QGateway Service and associated user tools for specific banks. These requirements are very likely to be typical of the needs of future customers. These components can be reused both as source-code examples and parameterizable tools.

Another way to address some of these missing capabilities in NEON (such as table lookups) is through user exits from the NEONet rules engine. This technique has been heavily used at one NPF customer site. Conversely, at the customer site where the greatest use of custom QGateway Resource DLLs has been made, there is no significant use of NEON rules and formats.

Either the NEON rules engine or the QGateway Service can be viewed as a container for developing custom message processing. Choice of which to use will be determined by which has the more suitable toolbox at hand. The NEON formatter can be called from either. The NEON rules engine is the more suitable container where foreign queue interfaces are dealt with separately, and reformatting and routing can be determined by the content of each message individually.

Online memo posts from PAYplus to the host applications are also routed through NEON. FundTech has established a standard flat (fixed-length) format for these messages. These messages can be reformatted for various host accounting applications, using the NEON formatter. NEON input and output formats are being defined for the General Ledger applications at initial customer sites. These formats can be reused and adapted at other banks.

PAYplus generates a feed to the WireHouse application, in order to support permanent archival of payment messages. NPF also supports the archival of non-payment messages, such as securities transactions, which bypass PAYplus. NPF software generates this second feed to WireHouse directly from NEON queues. All applicable messages (that bypass PAYplus) are cloned into a separate NEON queue, based upon customizable NEON rules delivered with NPF. NPF includes a separate NEON Archiver program which drains this queue and converts the SWIFT messages into the format accepted by WireHouse.

Compaq has developed a utility, called QDump, for emptying specified queues. Messages are written out in a format which can be read by the user, and which can be read by QDump in order to reload messages to those queues. This utility can be used for housekeeping purposes, to empty error message queues. For these purposes, this utility is used in certain daily scripts provided with NPF. The utility can also be used to perform tests, prime demos or capture errors which require later analysis.

Compaq NPF Pursue

DESCRIPTION

This product is designed to cover the message tracking and auditing requirements of the NPF clients. QGateway plug-ins must now conform to certain processing requirements for message tracking these are outlined here. For the detailed plug-in descriptions see the relevant QGateway and Queuing and Routing Design documents.

Key Functional Requirements

- Open to support different types of queuing systems.
- Open framework to integrate events from other systems.

Features

In summary, the major areas of functionality addressed by Pursue are:

1. Migrate database viewer into a web browser
2. User Event Tracking
3. Provide performance and usage statistics
4. Provide ability to access NEON archive queue
5. Provide queue viewer to replace NEON M&Q and other viewers
6. Pull events from NEON Archiver
7. Pull events from PAYplus

8. Text search through message, especially for the beneficiary
9. Ability to view the archive tables.
10. Support a Journal flag, which maps to a journaling capability in the underlying queuing system.

FIRCO-57™, FIRCO-59™, FIRCO-5E™

DESCRIPTION

FIRCO 5X/57/59/5E
Perform Field-Level Repairs and Increase your STP Rate

Increasing STP with FIRCO
The FIRCO 5X/57/59/5E family of products has been designed to repair data fields in SWIFT payment messages and thus increase STP rates.

Common Product Architecture
FIRCO takes as input a free-text data field and returns, in milliseconds, a coded format. This repair transformation relies mainly on clearing or branch code files. Inside FIRCO natural language processing techniques are used to parse the free-text field and extract relevant information (bank name, branch name, address, city, country, etc.). This information is then used to query the reference database constructed from clearing or branch code files (BIC+, CIF, etc.). A successful query returns one or more solutions that are then filtered by a neural network that retains the correct solution.

FIRCO-5X
FIRCO-5X accepts as input any SWIFT message 5X field, D- or B-tagged, and returns the corresponding clearing code. The code is found in the reference database constructed from bank clearing codes files (BIC, BLZ, CIF, ...).

FIRCO-57
FIRCO-57 accepts as input SWIFT message 57 field, D- or B-tagged, and returns the corresponding branch name code. The branch name is found in the reference database constructed from the bank's branch name list.

FIRCO-59
FIRCO-59 accepts as input SWIFT message field 59 and returns, if the information exists, a qualified account number.

The detection of account numbers is based on simple yet powerful grammar rules that are bank specific.

FIRCO-5E
FIRCO-5E, as the E in its name conveys, is specially designed to repair outbound message fields. FIRCO-5E is primarily applied to D- or B-tagged 57 fields but can be used on any general SWIFT field. The reference database is constructed from the BIC+ file. FIRCO-5E can be used in manual mode to propose a set of solutions where the operator needs to choose from. In manual mode FIRCO-5E needs to be interfaced with the message entry system.

Product Characteristics

The general product characteristics are:

- Simple to interface Client/Server architecture;
- Advanced software techniques (AI) for free-text analysis and repair;
- Testing environment;
- Product documentation;
- Low operating and maintenance costs;
- Repair statistics for monitoring performance;
- Support and maintenance program;
- Y2k compliance.

More Detail:

FIRCO-57 is specifically designed for the production of the transit number identifying the branch of the receiving bank. This identification is based on the detection of the branch name, its address and city (and eventually by any other discriminating element), for example by analyzing a SWIFT field 57 MT 1XX or MT2XX. FIRCO-57 is dedicated to branch code identification for a given bank given a D/B tag field, generally the 57 field.

FIRCO-57 relies on a reference database in which a clear description of the branch (branch name, address, city) is associated with its identification code.

The first two examples are taken from FIRCO-57 in operation at a French bank:

- The first example shows a classical case where, in using the address (106 boulevard de Strasbourg) and the city (Le Havre) FIRCO-57 gives the code of the Crédit Lyonnais branch. Before looking for the branch in the reference database FIRCO-57 checks that Crédit Lyonnais is the bank concerned in the field.
- In the second example, also related to Crédit Lyonnais, the branch code (704) is explicitly in the field. The identification is possible using the branch location as well as using the code if it is present in the message (notice that this is not obvious since a number could be a number in a street, a part of an account number, etc.). Finally, in this example FIRCO-57 assumes the field concerns the target bank Crédit Lyonnais, even if its name is not explicitly there. If another bank name was detected, via grammatical analysis, FIRCO-57 would return a control code indicating that the field does not concern the target bank.

The three other examples are extracted from FIRCO-57 used by a Canadian bank:

- We see that in examples 3 and 5 the same branch code is given with more or less complete information. For example, if there is no ambiguity in the reference database, the sole address is sufficient (in case of ambiguity FIRCO-57 would give a specific control code indicating that more than one branch code is possible).

- In example 4 all the information is in the message: the name of the branch (Orchard Plaza Branch), the address (1840 Cooper Road) and the city name (Kelowna).

Reference database

The reference database consists of one unique file in which a clear description of the branches (name, address, city and country) are associated with the codes. Of course, function of the organization of each bank, the format of the reference file and the associated decision rules may be examined case by case.

Product Characteristics

The general product characteristics are:

- Simple to interface Client/Server architecture;
- Advanced software techniques (AI);
- Low operating and maintenance costs;
- Support and maintenance program;
- Y2k compliance.

FIRCO-59 is specifically designed to detect and validate client account numbers in payment message.

Product Purpose

FIRCO-59 was specifically designed to allow automatic processing of free text in messages which contains bank account number: detection then qualification and validation; typically, FIRCO-59 is very useful to process field 59 of incoming or outgoing MT100 messages. According to S.W.I.F.T standards, field 59 designates the beneficiary account.

This process relies heavily on

- a reference database that ties client’s names to account numbers;
- a tuning phase based on the content of the reference database and a large sample of S.W.I.F.T fields (generally several thousands).

The step-by-step processing of a 59 field is as follows:

1. parsing The free-text string is parsed to obtain :
   - the transit code
   - the account number

   If the parsing step fails, processing stops and FIRCO-59 returns a code indicating this impossibility.

In case of success, FIRCO-59 has now identified the account number and the transit code that will be returned to the requester during one of the later steps. The aim of these steps is to validate the solution against the bank's database. Various levels of validation are provided.

2. Reference database search. The transit code and the account number are used to query the database. This query provides the name of the client of the account and the city where this client is located.

   If the database does not contain the account number, then FIRCO-59 returns a code indicating the failure of the query. The transit code and the account number are returned as well; this is the lower level of validation.

3. Matching. In this step FIRCO-59 checks whether the client's name and the city are present in the incoming field and returns the solution together with a validation code that indicates which of them is present (if any).

   Together with FircoSoft's advice, the bank has to decide which of the return codes provide a validation level that allows secure straight-through processing.

   The reference database can be of two types:

   - A database in which the branch code is associated with all the account numbers inside it. The 59 field must contain, eventually among other information, the account number.
   - A database, much like that in FIRCO-5X and FIRCO-57, in which a clear description of the clients (name, address, etc.) is associated with their account numbers.

Product Characteristics

The general product characteristics are:

- Simple to interface Client/Server architecture;
- Advanced software techniques (AI);
- Low operating and maintenance costs;
- Support and maintenance program;
- Y2k compliance.

FIRCO-5E where E stands for emission is specifically designed to transform free text to bank code in a context of outbound messages, payments or securities.

In a SWIFT context, banks that emit messages with D tags are often penalized by the receiver's repair fee. To avoid being charged repair fees, banks can apply FIRCO-5E , at a centralized level, to repair fields 57D on the basis, for example, of the BIC+ file. From an integration point of view, there are two ways of using FIRCO-5E:

1. In automatic decision mode, FIRCO-5E repairs and out the message goes;

2. In consultation mode, FIRCO-5E looks up all possible repair alternatives and proposes them to the user who makes the final decision.

Product Characteristics

The general product characteristics are:

- Simple to interface Client/Server architecture
- Advanced software techniques (AI)
- Low operating and maintenance costs
- Support and maintenance program
- Y2k compliance

CONFORMANCE TO STANDARDS

- Optional W.O.R.M. storage device establishes permanent records of all payment activities as required by regulatory authorities and S.W.I.F.T. standards
- Improved audit acceptance
- Reporting capabilities designed to comply with Central Bank regulatory requirements

INSTALLATION

Compaq requires that a customer’s first purchase of this software product include Compaq Installation Services. These services provide for installation of the software product by an experienced Compaq Software Specialist.

For subsequent purchases of this product only experienced customers should attempt installation. Compaq recommends that all other customers purchase Compaq’s Installation Services.

Before installation of the software, the customer must:

- Previously have installed all prerequisite software and hardware including terminals.
- Make available for a reasonable period of time, as mutually agreed by Compaq and the customer, all hardware, communications facilities and terminals that are to be used during installation.
- For verification of installation and connectivity in a multi-node network, designate and provide access to host that has previously been installed by Compaq.

HARDWARE AND SOFTWARE REQUIREMENTS

NonStop Payments Factory is Windows NT based only.

The H/W and S/W requirements for this and all components of the NonStop Payments Factory (NPF) product, due to its complex nature, are tightly controlled and spelled out in the configurations document (Bill of Materials) available at the NPF product web site: http://nsis.ogo.dec.com/fsip/wholesale/NPF/part_numbers.htm

GROWTH CONSIDERATIONS

The minimum hardware and software requirements for any future version of this product may be different from the requirements for the current version.

DISTRIBUTION MEDIA

After installation, integration and customization activities are completed, the NonStop Payments Factory software will be distributed via a CD–ROM to the end-customer.

YEAR 2000 READINESS STATEMENT

This product is Year 2000 Ready.

Year 2000 Ready is defined: "Year 2000 Ready" products are defined by Compaq as products capable of accurately processing, providing, and/or receiving date data from, into and between the twentieth and the twenty-first centuries, and the years 1999 and 2000, including leap year calculations, when used in accordance with the associated product documentation and provided that all hardware, firmware and software used in combination with such products properly exchange accurate date data with the products.

For additional information visit the DIGITAL Brand area on Compaq's Year 2000 Ready web site located at http://ww1.digital.com/year2000/warranty.asp.

The NonStop Payments Factory Year 2000 Readiness Statement can be accessed on our web site: http://nsis.ogo.dec.com/fsip/wholesale/NPF.htm

SOFTWARE WARRANTY

This software is provided by Compaq with a 90 day conformance warranty in accordance with the Compaq warranty terms applicable to the license purchase.

ORDERING INFORMATION

NonStop Payments Factory Real Time Gross Settlement Base

QL-68SA9-3B	100 TRANS LIC PayPlus 100 Payments RTGS Engine
QL-68SA9-5B	100 TRANS UPD PayPlus 100 Payments RTGS Engine

NonStop Payments Factory Real Time Gross Settlement Upgrade

QL-6CZA9-3B	250 TRANS LIC PayPlus 250 Payments RTGS Engine
QL-6CZA9-5B	250 TRANS UPD PayPlus 250 Payments RTGS Engine
QL-6CZA9-3C	500 TRANS LIC PayPlus 500 Payments RTGS Engine
QL-6CZA9-5C	500 TRANS UPD PayPlus 500 Payments RTGS Engine
QL-6CZA9-3D	750 TRANS LIC PayPlus 750 Payments RTGS Engine
QL-6CZA9-5D	750 TRANS UPD PayPlus 750 Payments RTGS Engine
QL-6CZA9-3E	1K TRANS LIC PayPlus 1K Payments RTGS Engine
QL-6CZA9-5E	1K TRANS UPD PayPlus 1K Payments RTGS Engine
QL-6CZA9-3F	1500 TRANS LIC PayPlus 1500 Payments RTGS Engine
QL-6CZA9-5F	1500 TRANS UPD PayPlus 1500 Payments RTGS Engine
QL-6CZA9-3G	2000 TRANS LIC PayPlus 2000 Payments RTGS Engine
QL-6CZA9-5G	2000 TRANS UPD PayPlus 2000 Payments RTGS Engine
QL-6CZA9-3H	2500 TRANS LIC PayPlus 2500 Payments RTGS Engine
QL-6CZA9-5H	2500 TRANS UPD PayPlus 2500 Payments RTGS Engine
QL-6CZA9-3J	3K TRANS LIC PayPlus 3K Payments RTGS Engine
QL-6CZA9-5J	3K TRANS UPD PayPlus 3K Payments RTGS Engine
QL-6CZA9-3K	4K TRANS LIC PayPlus 4K Payments RTGS Engine
QL-6CZA9-5K	4K TRANS UPD PayPlus 4K Payments RTGS Engine
QL-6CZA9-3L	5K TRANS LIC PayPlus 5K Payments RTGS Engine
QL-6CZA9-5L	5K TRANS UPD PayPlus 5K Payments RTGS Engine
QL-6CZA9-3M	6K TRANS LIC PayPlus 6K Payments RTGS Engine
QL-6CZA9-5M	6K TRANS UPD PayPlus 6K Payments RTGS Engine
QL-6CZA9-3N	7K TRANS LIC PayPlus 7K Payments RTGS Engine
QL-6CZA9-5N	7K TRANS UPD PayPlus 7K Payments RTGS Engine
QL-6CZA9-3P	8K TRANS LIC PayPlus 8K Payments RTGS Engine
QL-6CZA9-5P	8K TRANS UPD PayPlus 8K Payments RTGS Engine
QL-6CZA9-3Q	9K TRANS LIC PayPlus 9K Payments RTGS Engine
QL-6CZA9-5Q	9K TRANS UPD PayPlus 9K Payments RTGS Engine
QL-6CZA9-3R	10K TRANS LIC PayPlus 10K Payments RTGS Engine
QL-6CZA9-5R	10K TRANS UPD PayPlus 10K Payments RTGS Engine

NonStop Payments Factory Payplus Development and D/R

QL-68TAY-AA	TRAD LIC PayPlus Dev. and D/R License
QL-68TAY-RA	TRAD UPD PayPlus Dev. and D/R License

NonStop Payments Factory FaxWire Feature

QL-68UAY-AA	TRAD LIC FaxWire
QL-68UAY-RA	TRAD UPD FaxWire

NonStop Payments Factory FaxWire Notification Feature

QL-68VAY-AA	TRAD LIC FaxWire Notification

QL-68VAY-RA	TRAD UPD FaxWire Notification
NonStop Payments Factory Wirehouse	
QL-6D5AY-AA	TRADE LIC WireHouse
QL-6D5AY-RA	TRADE UPD WireHouse
NonStop Payments Factory ReconStar Base	
QL-6D6A9-3B	2500 MATCHES Recon$tar 2500 Matches License
QL-6D6A9-5B	2500 MATCHES Recon$tar 2500 Matches Update
QL-6D6A9-3C	UNLT MATCHES Recon$tar Unlmited Matches License
QL-6D6A9-5C	UNLT MATCHES Recon$tar Unlmited Matches Update
NonStop Payments Factory ReconStar Upgrade	
QL-6D7A9-3B	5000 MATCHES Recon$tar 5000 Matches License
QL-6D7A9-5B	5000 MATCHES Recon$tar 5000 Matches Update
QL-6D7A9-3C	7500 MATCHES Recon$tar 7500 Matches License
QL-6D7A9-5C	7500 MATCHES Recon$tar 7500 Matches Update
QL-6D7A9-3D	10K MATCHES Recon$tar 10K Matches License
QL-6D7A9-5D	10K MATCHES Recon$tar 10K Matches Update
NonStop Payments Factory Messaging and Queing Server - Neon	
QL-68YAX-AA	SM SV TRAD LIC NEONet (Rules, Formatter, and on NEONet M&Q license) for Small NT System Class
QL-68YAX-RA	SM SV TRAD LIC NEONet (Rules, Formatter, and on NEONet M&Q update) for Small NT System Class
QL-68YAX-AX	MD SV TRAD LIC NEONet (Rules, Formatter, and on NEONet M&Q license) for Medium Unix System Class
QL-68YAX-RX	MD SV TRAD LIC NEONet (Rules, Formatter, and on NEONet M&Q update) for Medium Unix System Class
QL-68YAX-AY	LG SV TRAD LIC NEONet (Rules, Formatter, and on NEONet M&Q license) for Large Unix System Class
QL-68YAX-RY	LG SV TRAD LIC NEONet (Rules, Formatter, and on NEONet M&Q update) for Large Unix System Class
NonStop Payments Factory Messaging and Queing Server - MQ	
QL-6D1AX-AA	SM SV TRAD LIC MQ Integrator (Rules, Formatter, and one IBM MQSeries license) for Small NT System Class
QL-6D1AX-RA	SM SV TRAD LIC MQ Integrator (Rules, Formatter, and one IBM MQSeries update) for Small NT System Class
QL-6D1AX-AX	MD SV TRAD LIC MQ Integrator (Rules, Formatter, and one IBM MQSeries license) for Medium Unix System Class
QL-6D1AX-RX	MD SV TRAD LIC MQ Integrator (Rules, Formatter, and one IBM MQSeries update) for Medium Unix System Class
QL-6D1AX-AY	LG SV TRAD LIC MQ Integrator (Rules, Formatter, and one IBM MQSeries license) for Large Unix System Class
QL-6D1AX-RY	LG SV TRAD LIC MQ Integrator (Rules, Formatter, and one IBM MQSeries update) for Large Unix System Class
NonStop Payments Factory Messaging and Queing Server plus D/R	
QL-68XAX-AA	SM TRAD LC Disaster Recovery License for Small NT System Class
QL-68XAX-RA	SM TRAD LC Disaster Recovery Update for Small NT System Class

QL-68XAX-AX	MD TRAD LC Disaster Recovery License for Medium Unix System Class
QL-68XAX-RX	MD TRAD LC Disaster Recovery Update for Medium Unix System Class
QL-68XAX-AY	LG TRAD LC Disaster Recovery License for Large Unix System Class
QL-68XAX-RY	LG TRAD LC Disaster Recovery Update for Large Unix System Class

NonStop Payments Factory for Real Time SWIFT Messaging

QL-68WAY-AA	NTI TRAD LIC NEONaccess
QL-68WAY-RA	NTI TRAD UPD NEONaccess

NonStop Payments Factory for Batch SWIFT Messaging

QL-6D0AY-AA	NTI TRAD LIC NEONentry
QL-6D0AY-RA	NTI TRAD UPD NEONentry

NonStop Payments Factory Pursue

QL-6DAAY-AA	TRAD LIC NonStop Payments Factory Pursue
QL-6DAAY-RA	TRAD UPD NonStop Payments Factory Pursue

NonStop Payments Factory Qgateway

QL-6D8AY-AA	TRAD LIC NonStop Payments Factory QGateway
QL-6D8AY-RA	TRAD UPD NonStop Payments Factory QGateway

NonStop Payments Factory Firco 5E

QL-68ZA9-3B	1000 TRAN LIC FIRCO-5E 1000 Transaction License
QL-68ZA9-5B	1000 TRAN UPD FIRCO-5E 1000 Transaction License
QL-68ZA9-3C	2000 TRAN LIC FIRCO-5E 2000 Transaction License
QL-68ZA9-5C	2000 TRAN UPD FIRCO-5E 2000 Transaction License
QL-68ZA9-3D	UNLD TRAN LIC FIRCO-5E Unlimited Transaction License
QL-68ZA9-5D	UNLD TRAN UPD FIRCO-5E Unlimited Transaction License

NonStop Payments Factory Firco 57

QL-682A9-3B	1000 TRAN LIC FIRCO-57 1000 Transaction License
QL-682A9-5B	1000 TRAN UPD FIRCO-57 1000 Transaction License
QL-682A9-3C	2000 TRAN LIC FIRCO-57 2000 Transaction License
QL-682A9-5C	2000 TRAN UPD FIRCO-57 2000 Transaction License
QL-682A9-3D	UNLD TRAN LIC FIRCO-57 Unlimited Transaction License
QL-682A9-5D	UNLD TRAN UPD FIRCO-57 Unlimited Transaction License

NonStop Payments Factory Firco 59

QL-6D3A9-3B	1000 TRAN LIC FIRCO-59 1000 Transaction License
QL-6D3A9-5B	1000 TRAN UPD FIRCO-59 1000 Transaction License
QL-6D3A9-3C	2000 TRAN LIC FIRCO-59 2000 Transaction License
QL-6D3A9-5C	2000 TRAN UPD FIRCO-59 2000 Transaction License
QL-6D3A9-3D	UNLD TRAN LIC FIRCO-59 Unlimited Transaction License
QL-6D3A9-5D	UNLD TRAN UPD FIRCO-59 Unlimited Transaction License

QP-04PAA-AA	Financial Application Integration (FAI) Package: This includes NEONet (Rules, Formatter, and one NEONet M&Q license), QGateway and Pursue.
QP-04MAA-AA	RTGS Base package which includes the FAI package plus PayPlus.
QP-04NAA-AA	PBB Base package which includes the FAI package plus PBB.

The above information is valid at time of release. Please contact your local Compaq office for the most up to date information.

SOFTWARE PRODUCT SERVICES

Information about these services is detailed under NPF Bid and Proposal at the following web-site: http://nsis.ogo.dec.com/fsip/wholesale/NPF/npf_sales.htm

A variety of service options are available from Compaq. For more information, contact your local Compaq office.

™ Compaq and the Compaq Logo are registered in the United States Patent and Trademark Office.

™ FIRCO-57, FIRCO-59, FIRCO-5E are trademarks of FircoSoft

™ NEON is a trademark of NEON Inc.

™ PAYplus RTGS, FaxWire, WireHouse, Recon$tar are trademarks of Fundtech Ltd.

™ Windows NT is a trademark of Microsoft Corporation.

All other trademarks and registered trademarks are the property of their respective holders.