Paper # 2013

Integrating RF Clients with HP 9000 Business Servers

Mark Mitchener
Jeff Dunn

Dofasco Inc.
Hamilton, Ontario
Canada
L8N 3J5
(905) 544-3761

Introduction

Dofasco, one of Canada's largest integrated steel mills, produces over 3 million tons of sheet metal products annually for the automotive, packaging and construction industries. Dofasco's main manufacturing facility is spread over several hundred acres and steel products in various stages of processing are transported by a large array of mobile equipment.

In order to meet customer's ever increasing demands of quality control, cost reduction and delivery performance it became necessary to develop a real time transactional interface between mobile equipment and the various on-line product tracking, scheduling and quality systems.

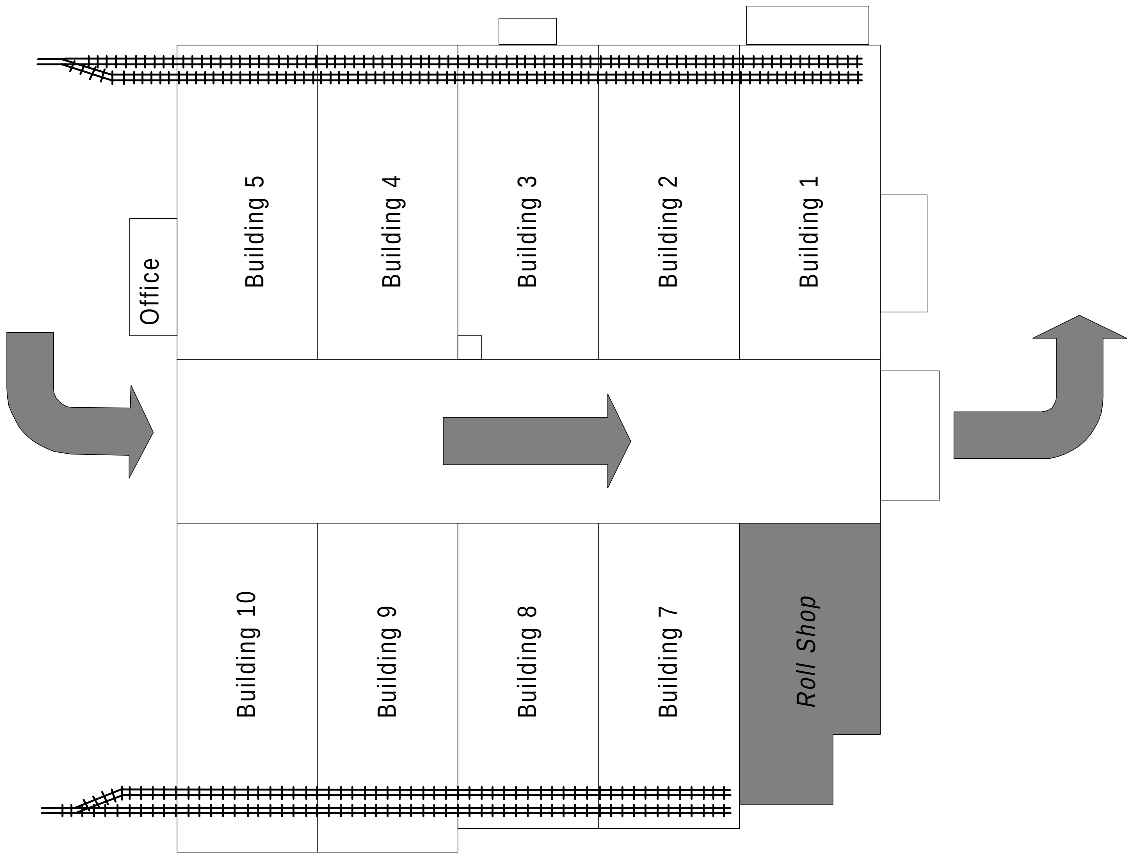
The need: wireless mobile clients (complete with an easy to use GUI and bar code scanners) that could access vertical applications running on Dofasco's HP business servers.

The plan: integrate HP-UX 9000-800 systems with Telxon Spread Spectrum Radio LAN technology to form a client server solution.

This paper provides a case study of how Dofasco integrated Telxon Radio LAN technology and HP systems to implement a mobile client server solution. The focus will be on the design of the server applications and the unique problems encountered with wireless mobile clients.

The Central Shipping Pilot Project

Dofasco's Central Shipping complex is a nine acre climate controlled facility capable of storing 200,000+ tons of finished product inventory**.** The complex utilizes rack storage which allows for approximately 32,000 physical locations. The warehouse is subdivided into nine areas which are referred to as buildings. Each building has a captive stacker crane which is capable of manipulating material both in the adjoining as well as it's own building.



The complex is an active facility servicing well over 300 customer trucks and 150 mill transporters per 24 hour period. Although this encompasses the bulk of the activity in the complex, Central Shipping also accommodates rail and container / van shipments.

The Manual Process:

Once the customer's order is taken, the product is produced according to specification and the acknowledged delivery date.

After the material has gone through it's final operation it is moved into Central Shipping via large transporters. Accompanying the load is a paper copy of the product identification tags and a Product Transfer Report (internal bill of lading).

Prior to unloading the transporter deck, the Central Shipping crane operator would use the product id. tags to determine what the product is and where it should be located within the building.

As each piece was placed into storage the operator would manually write the location code on each of the tags. Once all of the material on the load had been located the product id. tags (and product transfer report) would be placed in a collection box.

These collection boxes were emptied periodically throughout the shift and subsequently entered into Dofasco's Order Handling System via a hardwired terminals.

With the material in a shippable location, loads can be / are assembled and the appropriate carrier is booked.

The carrier will arrive according to schedule and report to the bill dispensation office to obtain the appropriate bill(s) of lading.

The carrier will then prepare his truck for loading and wait to be marshaled into the complex by the warehouse coordinator. The carrier is guaranteed to be serviced within a two hour window or demurrage charges will result.

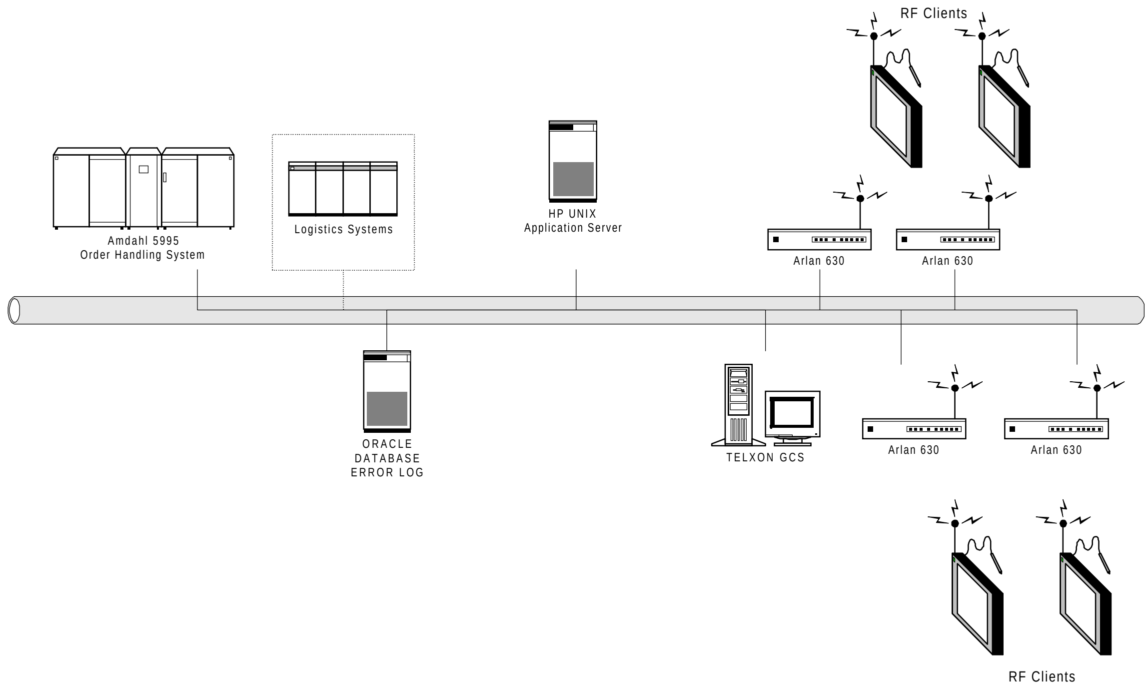
Once the carrier has entered the facility he/she will hand the bill of lading over to the crane operator. The operator will then utilize this document to locate the corresponding product and load the carrier.

The Objective

The goal is to evolve Central Shipping into a paperless warehouse. To achieve this objective will require a migration in technology as well as business practices within the business unit. In order to effectively manage this cultural change Dofasco chose to approach the challenge in phases.

The Architecture

The first step was to determine the system architecture. A few methods of integrating radio frequency technology into Dofasco's infrastructure were available. Considering the legacy mainframe environment that was in place, the simplest alternative would have been to implement a terminal emulation configuration. Standard screens could be developed on the host and the radio frequency PC would merely behave like any other hardwired terminal. However, this configuration was considered to be too limiting. It offered the inflexibility of being directly tied to the mainframe and would be restricted to a character based user interface. Also playing a key role in the consideration of the design was Dofasco's ongoing migration from a centralized Order Handling System (Amdahl 5995) to a distributed client-server architecture. The magnitude of this migration dictates that it be conducted in stages and as a result the Central Shipping Inventory System had to be designed with the flexibility to communicate with multiple hosts as Dofasco's Logistics functionality and information is progressively displaced off of the Amdahl. With this in mind the following client-server design was developed.



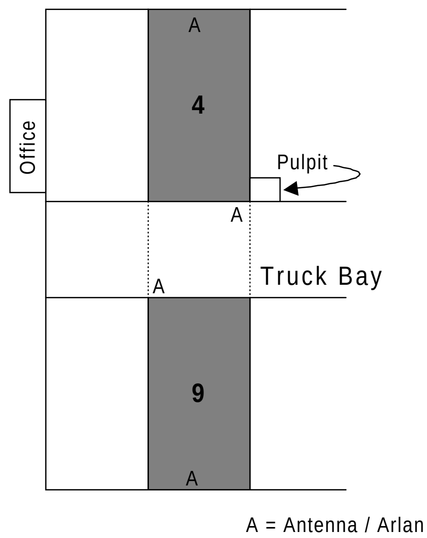
The premise of the design is that the radio frequency PC's would be used to their full potential as clients. These RF Clients would communicate with server applications on a HP-UX 9000 system via TCP/IP stream sockets. The server applications would interpret the incoming packets and based on the request would then communicate (again via TCP/IP sockets) with the appropriate host (initially the Amdahl 5995) to request & receive the desired information. The information would then be returned to the requesting RF client for presentation. A similar chain of events would be followed for update transactions with the additional requirement that store and forward capability be implemented at each level to guarantee delivery of every update transaction.

With the system architecture in hand the implementation phases were determined.

Phase I: A Proof of Concept

The purpose of this pilot phase was to quantifiably demonstrate the performance of the client server design and equipment durability in the warehouse environment.

To accomplish this a radio frequency network was installed in buildings 4 and 9 of the complex.



These “internal” buildings were specifically selected because it was felt that redundant RF coverage within the interior of the complex would be more difficult since there were no walls to reflect and propagate the signal back into the facility. Also, the proximity to existing hardwired network drops within this area of the complex made it somewhat easier to install the pilot network.

Simple applications with core functionality were developed for the RF client, the HP application server and the mainframe Order Handling System. A simple logging mechanism was incorporated into the application so that the performance of the system could be quantifiably assessed.

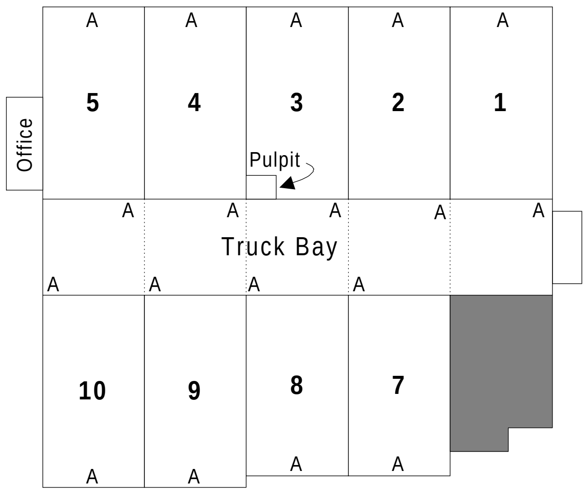
The pilot phase was conducted for a three month period. During that time frame statistics on every transaction for the two RF clients were logged and the results reported. In addition to operating the equipment in conjunction with the job functions in the warehouse, stress testing was also conducted. In an effort to overload the radio frequency network, seven RF PC clients were utilized to progressively walk through the RF coverage cells performing a high level of transactions. In addition, the hardwired network in the complex was isolated and an abnormally high traffic level was created (network utilization was raised to 50% - 60%).

The result of the pilot phase was that response times averaged between one and three seconds and the equipment performed without fault in the warehouse environment.

Phase II: Introduction

The purpose of this phase was to introduce the new technology into the job procedures through the automation of piece status queries and the receive / relocate functions within Central Shipping. Before this introduction could occur the supporting infrastructure had to be installed.

From a hardware perspective this required the permanent installation of the RF network throughout the shipping facility. Eighteen antenna / access points were required to provide redundant coverage for the complex. As illustrated below, the radio access points were staggered throughout the truck bay providing a high level of redundancy in the loading / unloading area as well as coverage into the interior of the building. Overlapping coverage within the rack storage of the buildings was achieved by adding additional antenna / access points at the north or south walls of each of the buildings.



A = Antenna / Arlan

From a software perspective additional business functionality had to be supplied as well as the implementation of a fault tolerant network communication layer on all platforms.

Design Criteria

The goal of the Central Shipping Inventory (CSI) system is to provide accurate real time data to high level production planning and scheduling systems. The main issues driving the design are reliability, flexibility, centralized management and performance.

Reliability

With a three (3) tier client-server architecture there are several points of failure. The software must guarantee the delivery of messages at each level by being capable of detecting communication errors and providing a recovery strategy.

Flexibility

Currently all coil information is maintained in CICS/VSAM files on an Amdahl mainframe. Dofasco is currently in the process of downsizing its corporate mainframe, eventually all the data will be migrated to an HP/Oracle database. The CSI system must be capable of accessing a variety of data sources.

Centralized Management

A centralized error logging facility is required to report network and system errors to operations personnel.

Business data and logic errors are to be logged and easily accessible to warehouse management.

With PTC units in remote and often poorly accessible locations, the CSI system needs to provide a method of distributing software updates from a central server.

Performance

Response time must be under two (2) seconds in order not to interfere with the stacker crane loading operations.

Hardware Components

Radio Frequency (RF)

- **Aironet Arlan 630 -** The Arlan 630 is an RF to Ethernet access point that uses patented spread spectrum radio technology for fast, secure and reliable communications. The Arlan unit is equipped with a 900 MHz radio and can be placed anywhere along an Ethernet LAN. The 630 allows wireless stations in its coverage area to transparently access the corporate network. Aironet's MicroCellular™ architecture permits several Arlan 630 units to form a vast (indoor or outdoor) wireless network thereby enabling the mobile user to seamlessly roam throughout the coverage area.

- **Telxon Gateway Connectivity System -** The GCS is used to provide connectivity between a host computer and Telxon's family of portable radio frequency microcomputers. The intelligent communications controller can manage up to 64 client sessions. The system is also equipped with built in error detection and recovery mechanisms that eliminate the risk of data loss due to RF interference.

- **Telxon PTC-870IM -** The 870IM is a ruggedized DOS based 486 SLC 50 MHz PC with an electro-luminescent display. This RF Client utilizes a graphical, pen/touch based user interface to facilitate easier user interaction and superior presentation capabilities. By using the event driven application development tool Penright!™ Dofasco was able to effectively incorporate store and forward capabilities into the single threaded operating environment

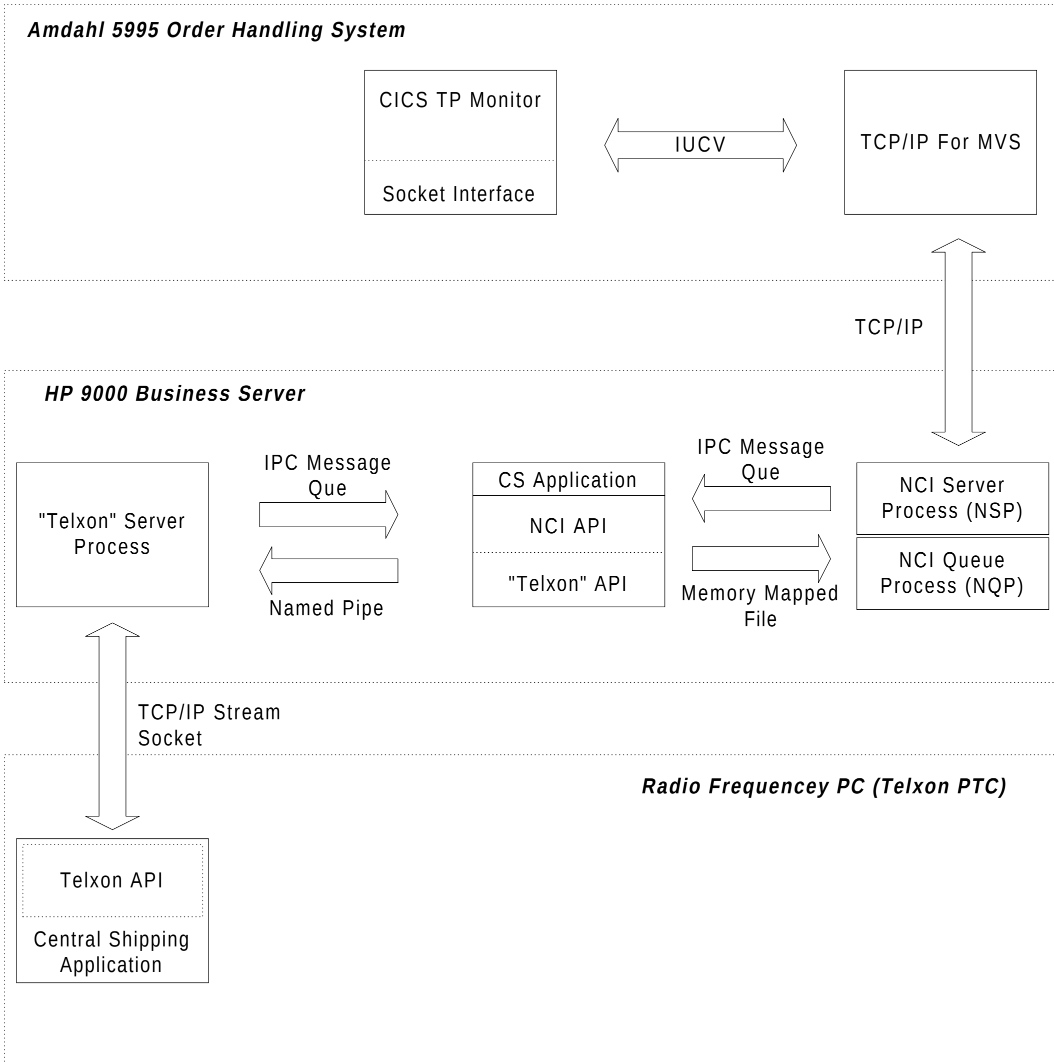
HP Application Server

- An HP 9000 I50 Business Server with 256 MB memory, running HP-UX 9.04 is used as the application server. All of Dofasco's HP 9000 systems are on a FDDI network. Bridges are used to link the FDDI ring to the Ethernet attached Arlan 630's and the mainframe.

Mainframe

- Dofasco's corporate mainframe is an Amdahl 5995-1100A running MVS/ESA 4.2 , CICS/XA 2.1.2 and TCP/IP 2.1.2. The mainframe is connected to the network with a BusTech channel attached Ethernet controller.

Software Components



RF Client (PTC)

- **NCI/DOS Socket Library -** The NCI/DOS Socket Library provides an API interface for RF client applications to communicate with a Telxon Server Process running on an HP-UX system. The API hides the complexity of TCP/IP sockets (provided by the Telxon Socket Library), communications protocols and store and forward functionality by providing four (4) functions to the user application (NET_CONNECT, NET_SEND, NET_RECEIVE and NET_CLOSE).

HP 9000 Business Server

- **Telxon Server Process -** The Telxon Server Process handles all the communication between the PTC and user applications. Communications to the PTC's are handled through TCP/IP stream sockets. The Telxon server transfers messages to user applications using IPC Message Queues and receives transaction requests through Named Pipes (FIFO's).

- **Data Access Programs (NINO) -** The NINO programs receive Coil Status and Update requests from RF clients. These programs currently use the NCI API to access data on CICS/MVS and embedded SQL to access ORACLE databases.
- **Network Communications Interface (NCI) -** The Network Communications Interface on HP-UX provides a reliable store a forward messaging system to CICS/MVS, OS/2 and DEC/VMS systems.

MVS/ESA

- **NCI Socket Library for CICS -** The NCI Socket Library for CICS allows CICS/Cobol transactions to send/receive message from HP-UX Business Servers. The NCI Socket Library implements the NCI application protocol on top of IBM's Socket API.
- **CICS Listener Transaction (CSKL) -** The CICS Listener (provided with IBM's TCP software) allows remote systems to connect to a CICS region (via TCP/IP) and initiate transactions.

Supporting Architecture

Reliability

To be reliable the CSI system must be capable of handling a variety of network and system errors. The most common type of problems are:

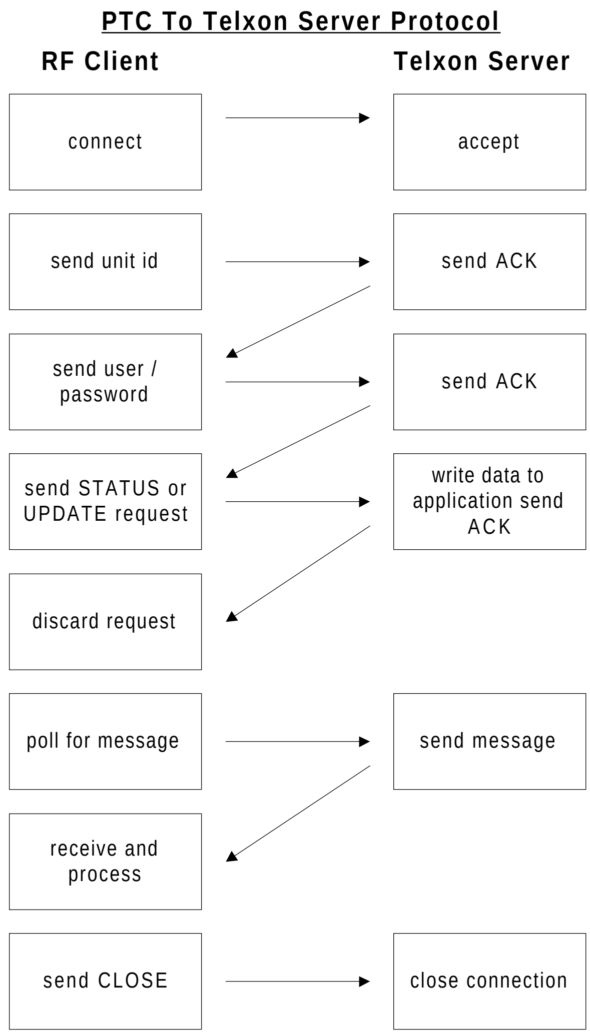
- CICS/VSAM or Oracle files are off line for maintenance
- the PTC unit is out of range or the RF network is down
- the HP-UX server or applications are unavailable
- the Amdahl mainframe is unreachable
- Network transmission errors (both RF and Ethernet)

To handle these errors the CSI system uses: Stream (TCP) sockets as a reliable delivery mechanism, an application protocol to confirm the delivery of messages between components, a store and forward queuing system to buffer messages between levels and non-volatile storage to preserve data in the event of a system failure.

The Protocol

In order to guarantee that updates would be committed to the target database the CSI system implements an application level protocol to provide handshaking between the client process and the server process. This server process would ACK the client only when the data was correctly received and committed to non-volatile storage. Upon receipt of the ACK, the client would discard the data knowing that it had been successfully committed on the server. If a NCK was received or no response was returned within the designated time-out period the client software would log a message to the central error logging facility and then attempt to re-transmit the data.

PTC To Telxon Server



Connecting

The PTC connects to the Telxon server process. If the connect request is accepted the PTC broadcasts it's unit identification. The server process checks if it has the PTC already registered in it's Active Connection Database. If the unit is already registered the existing connection is closed, the Active Connection Database is updated to reflect the new connection and any messages previously queued for the unit are discarded. An ACK is sent to the client

The PTC then transmits a (User Name/Password) string for security validation. If the server validates the user an ACK is sent to the client, otherwise a NAK is sent and the connection is closed.

Sending Messages

The client sends a message to the server. The server verifies that the message has been correctly received. The server then writes the data to the designated IPC Message Queue and then returns an ACK indicating that the data has been committed. The client is now free to resume processing.

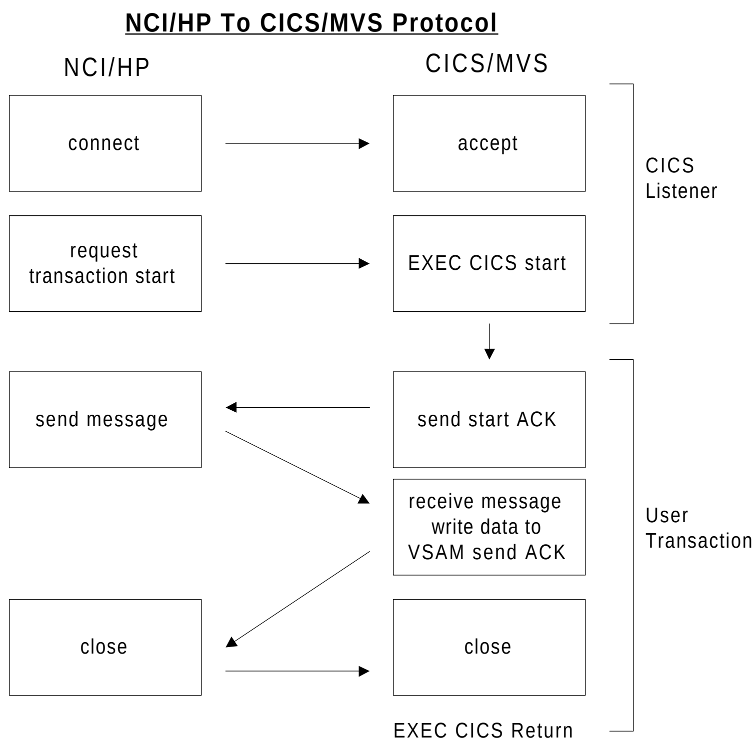
Receiving Messages

Since the PTC is DOS based (and hence not multiprocessing) it is not possible to send data to the client asynchronously. The PTC must poll the server for any pending data. If the server has data for this client unit the data is transmitted otherwise a NOMSG is returned. Since any data sent to the PTC is for display purposes only, no confirmation is expected from the PTC and the Telxon Server does not hold data for transmission retries.

Closing

The PTC sends a CLOSE message to the server and then closes the communications socket. Upon receiving the CLOSE the server closes the communications socket and deletes the unit registration information from its database.

NCI/HP to CICS/MVS



Connecting

The NCI Queue Processor (NQP) is responsible for all outbound message transfers. NQP connects to the CICS region and sends a transaction start request to the CICS listener program. The CICS listener starts the requested transaction and passes relevant connection information through the CICS/COMM area.

Upon successful initialization the requested transaction sends an ACK to the NQP indicating that it successfully initialized and is now ready to accept data.

Sending Messages

The NQP sends a message to the awaiting CICS transaction. The CICS transaction validates the message, writes it to a VSAM holding file and sends an ACK to NQP indicating the message was successfully received. The CICS transaction may now process the message or transfer control to another program.

Receiving Messages

Since both HP-UX and CICS/MVS are multiprocessing and multitasking operating systems data is received asynchronously. The NCI Server Processor (NSP) is responsible for all inbound messages.

The CICS transaction connects to the NSP and then transmits the message. The NSP verifies the message was received correctly and writes the message to the specified IPC Message Queue. If the queue write is successful the NSP returns an ACK to the CICS transaction.

Store and Forward

In addition to unexpected system and network errors, it is not uncommon to have components of the CSI system unavailable at regularly scheduled intervals:

- CICS regions are taken down daily for backups and file maintenance.
- Mainframe Batch jobs take CICS files off-line for updates.
- The Oracle database is off-line for cold backups.

To handle these component downtimes without impacting operations the CSI system implements store and forward queuing on the PTC and HP 9000 level. When the operator performs an update operation a transaction record specifying the destination, post time and user message is written to non-volatile storage.

The NCI software then attempts to transmit the transaction to the remote system. If the message transfer is successful (as confirmed by the application protocol) the transaction record is discarded. If the remote software component does not respond and the transaction is flagged as RETRY (Coil Updates are flagged as RETRY, Coil Status are discarded) the user message is maintained in the non-volatile storage. If the remote system does not respond NCI will either retry at the next transaction request or at timed intervals.

Note: that any holding transactions are processed before the new request. This is done to preserve the chronological order of transactions.

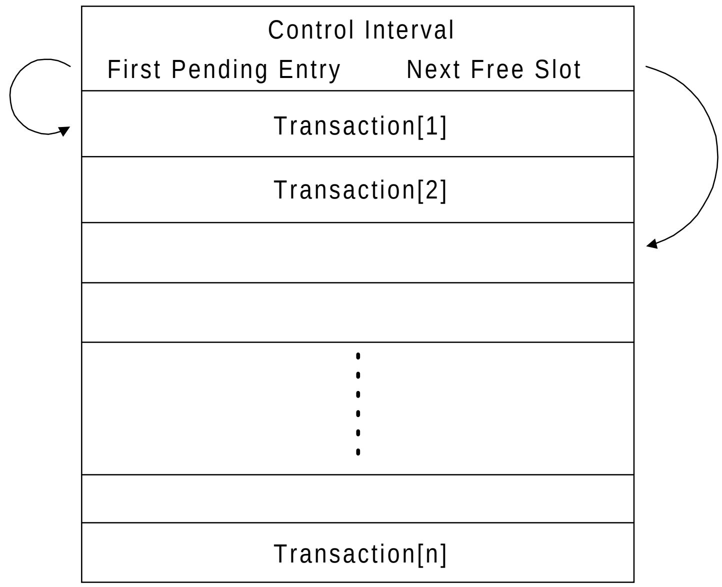
PTC to HP Telxon Server

Since the PTC is running on DOS and connects only to a single node (the Telxon server), the store and forward mechanism is implemented in a standard DOS stream file. The file is allocated with the following structure:

```
struct CONTROL_INTERVAL
{
        int     FirstPendingEntry;
        int     NextFreeSlot;
        int     ActiveEntries;
}

struct TRANSACTION_RECORD
{
        int     JobNumber;
        int     PostTime;
        .
        .
        int     UserMessageSize
        char    UserMessage[512]
}
```

```
struct HOLDING_FILE
{
        struct CONTROL_INTERVAL CI;
        struct TRASACTION_RECORD[MAX_ENTRIES];
}
```



The control interval contains pointers (record numbers) to the first pending transaction record, next available slot and the number of pending entries. The file is accessed in a circular fashion with the FirstPendingEntry and NextFreeSlot pointers wrapping around when they reach MAX_ENTRIES.

As new transaction requests are made they are written in the next free slot and the NextFreeSlot pointer is incremented. As pending transactions are successfully processed the FirstPendingEntry pointer is incremented moving it to the next transaction. When NextFreeSlot = FirstPendingEntry and:

ActiveEntries = 0 there are no pending transactions.

ActiveEntries = MAX_ENTRIES the holding file is full and no more transactions can be held. An error is return to the application.

HP to CICS/MVS

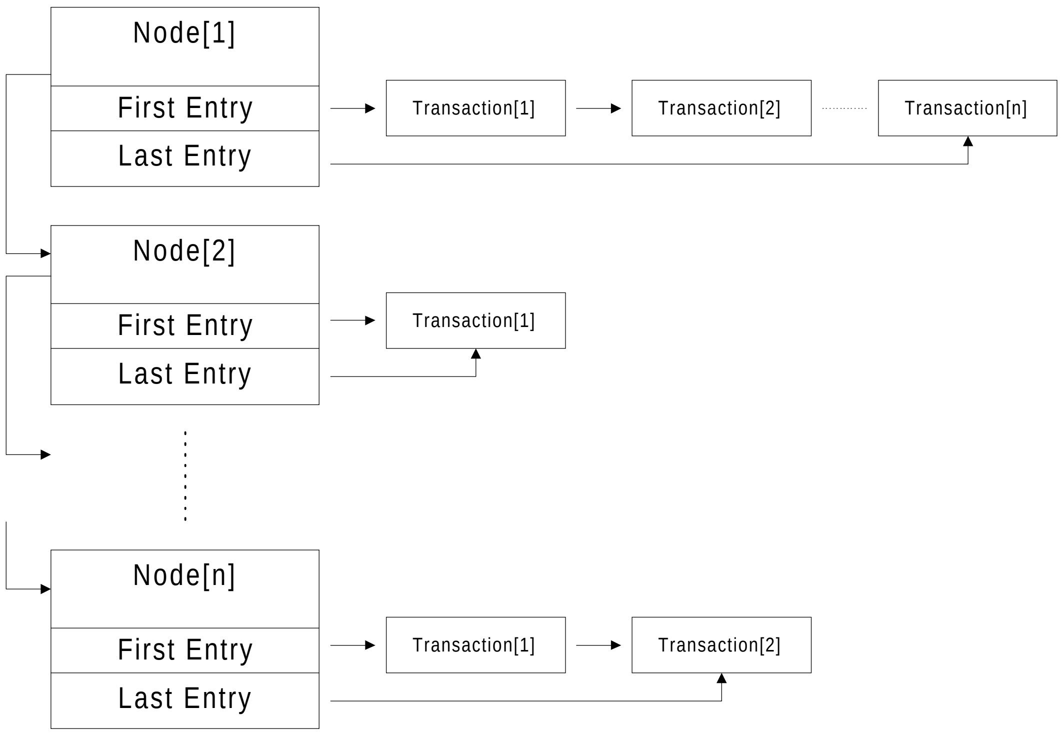
Since the HP version of NCI is required to handle communications between many network nodes and many applications it's store and forward system is a little more sophisticated.

The transaction queuing system is implemented in a memory mapped file. The use of a memory mapped file allows the NQP to implement a flexible link list structure that can be flushed to disk at frequent intervals to ensure the data survives a system crash.

```
struct QHEAD
{
        char    TargetNode[];
        int     NumPendingEntries;
        .
        .
        struct TRANSACTION_RECORD *FirstPendingEntry;
}
```

```
struct TRANSACTION_RECORD
{
        struct TRANSACTION_RECORD *NextPendingEntry;
        int         InUse;
        int         JobId;
        .
        .
        int         UserMessageLength;
        char        UserMessage[2048];
}

struct HOLDING_QUEUE
{
        struct QHEAD NodePendingList[MAX_NODES];
        struct TRANSACTION_RECORD PendingEntry[MAX_ENTRIES];
}
```



As user applications request transactions, the NCI API inserts it at the end of the linked list for the node and notifies NQP through semaphores. The NQP attempts to transmit the message and upon successful completion the entry is unlinked from the queue and the slot is marked free. If the transmit fails the entry remains on the linked list and is retried at the next iteration.

Flexibility

Having the data access programs on HP-UX allows the CSI system to evolve as infrastructure changes occur at Dofasco. The NINO programs are capable of accessing ORACLE and SYBASE databases anywhere on the network as well as communicating with applications on any of the NCI supported remote systems.

Centralized Management

Central Error Reporting

Dofasco currently uses HP's Operation Center (OPC) as a central event manager. The OPC console is monitored 24x7 by operations staff. Serious Problems encountered by the Telxon server or NCI communications software are logged to the OPC system.

The RF network hardware (Arlan and GCS) support SNMP. Although we have not yet done so, it is possible to have OPC poll the units and warn operations staff of usual error conditions.

Business Logic Errors

Business logic errors detected at the CSI application level or at the CICS transaction level are logged to an Oracle database. Management personnel at Central Shipping can view the error log through an in-house agent or with Excel and Microsoft Query.

Software Distribution

With PTC units mounted in mobile equipment it would be very difficult and time consuming to have to update each individual unit with new releases of software. The CSI system stores binary images of the DOS applications on the HP-UX server. When an operator signs on, the PTC Front End application connects to the Telxon Server and requests the version number of the current CSI application. If required the Front End application can initiate a download of the current version from the Telxon Server.

To minimize download time the binary image is stored on the HP in a compressed format (PKZIP). The front end program will then decompress the file and load the program for execution.

Performance

The RF network has a Maximum Transmission Unit (MTU) of 1024 bytes and a sustained speed of approximately 1.5 KB/sec. In order meet performance requirements the CSI system must minimize the amount of RF network traffic by:

- implementing as much business logic and data access as possible on the HP and CICS systems.
- discard out of date messages destined to the PTC.
- using data compression.

Phase III: Building Upon The Foundation

The purpose of this phase will be to reap the benefits of the developed infrastructure and target additional business functionality and operating units.

- Implement SNMP for the RF Network.
- Rollout piece location functionality to WIP areas throughout Dofasco.
- Intelligent inventory management and shipping functionality for Central Shipping.