Interfacing Customer Support System with E-Mail and Symix

PROJECT REPORT

Dissertation Submitted in partial fulfilment of the requirements for the Degree of

MASTER OF COMPUTER APPLICATIONS

of the Bharathiar University

By

TESSY TERESA GEORGE

Reg. No. 9438MO207

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Kumaraguru College of Technology

COIMBATORE - 641 006.

Certificate

This is to certify that this project work entitled

"Interfacing Customer Support System With E-Mail And Symix"

Submitted to

KUMARAGURU COLLEGE OF TECHNOLOGY
(Affiliated to Bharathiar University)

in partial fulfilment of the requirements for the award of Degree of
MASTER OF COMPUTER APPLICATIONS
is record of original work done by

TESSY TERESA GEORGE
(Reg.No. 9438MO207)

during his period of study in the Department of Computer Science and Engineering, Kumaraguru College of Technology, Coimbatore-641006, under my supervision and guidance and this project work has not formed the basis for the award of any Degree/Diploma/Associateship/Fellowship or similar title to any candidate of any university.

Professor and Head

Staff-in-Charge

Submitted for university Examination held on 03/OCT/1997

Internal Examiner

3/6/97
External Examiner

Declaration

I hereby declare that the project work entitled

INTERFACING CUSTOMER SUPPORT SYSTEM WITH E-MAIL AND SYMIX
at
WIPRO INFOTECH, SUPPORT DIVISION,
BANGALORE

Submitted in Partial fulfilment of the requirements for the award of the Degree of
MASTER OF COMPUTER APPLICATIONS
is a report of original work done by me during my period of study in

KUMARAGURU COLLEGE OF TECHNOLOGY
(Affiliated to Bharathiar University)
Coimbatore-641 006.

Under the Supervision of
Prof.P.SHANMUGHAM, M.Sc., (Engg.), M.S.(Hawai),

Name of the Candidate
TESSY TERESA GEORGE

Register Number
9438MO207

Signature of the Candidate
Tessy Tg

Date : 3/6/97
Place : Coimbatore

WIPRO
INFOTECH GROUP
SUPPORT DIVISION

Akshaya Commercial Complex
26 Victoria Road Bangalore-560 047 India
Tel : 91-80-5569991 / 2 / 3 / 5566420
5576712 / 5543503 / 5546523
Fax : 91-80-5560144

9-5-97

To Whom so ever it may Concern

This is to certify that Miss Tessy Teresa George of III MCA , Kumaraguru College of Technology has successfully completed the Customer Support System - E-Mail and Symix interface for **Wipro Infotech Group, Support Division** as a part of her Project work of her Curriculam during the period Jan 97 to May 97.

She has completed the project to the fullest satisifaction of our requirements. We wish her all the best.

Yours Sincerely,
For Wipro Infotech Group - Support Division,

P.Venkatachalam
Manager - Information Systems

Dedicated to

my

Beloved Parents

Acknowledgement

An endeavour over a long period can be successful only with the advice and support of many well wishers. I take this oppurtunity to express my gratitude and appreciation to all of them.

I am bound to express my gratitude to ***Dr.S.Subrahmaniam****, Principal, Kumaraguru College of Technology,(Affiliated to Bharathiar University),Coimbatore, Tamil Nadu,for his constant encouragement throughout my course.*

*I admit my hearfelt thanks to my Internal Project Guide,****Prof.P.Shanmugham****, H.O.D., Department of Computer Science and Engineering, Kumaraguru College of Technology,(Affiliated to Bharathiar University),Coimbatore,for constantly encouraging me to pursue new goals and ideas throughtout the tenure of my project.*

I owe much to ***Mr.N.Vasudeva Murthy,****HRD Manager,Wipro Infotech Group,Support Division,Bangalore, for prmitting me to do the project in this organization.*

I express my gratitude to ***Mr.M.BalaGiridhar****, General Manager and* ***Mr.P.Venkatachalam****,Project Manager, Wipro Infotech Group,Support Division for having provided me an oppurtunity to take up this Project and for their inspiring advice, immense help,whole-hearted support and constant encouragement throughout the tenure of this project work at this esteemed organisation.*

My Parents always provide love and support, and I appreciate them greatly.

Tessy Teresa George

April 1997
Bangalore

SYNOPSIS

Synopsis

This project entitled **"Interfacing Customer Support System to E-Mail and SYMIX"** is developed for **Wipro Infotech Group Support Division.**

Customer Support System(CSS) is an integrated system for monitoring Customer Calls and Annual Maintenance Contract Revenue. The Call Monitoring System enables the Wipro Customer Support Department to provide a high degree of Customer Satisfaction by providing various related operations. This module **'Interfacing Customer Support System to E-Mail'** constitutes maintaining the Norms based on the various criteria and Escalating the Calls to higher levels for further action, when the call exceeds the Norms. This Escalation is done using E-Mail which is a reliable Electronic Transmission Medium

SYMIX is an **Enterprise Resource Planning(ERP)** Package. This is being customized by Wipro Infotech to suit their requirement. The Wipro Infotech Support Division, having huge inventory of Spares to service the Customer Calls, has implemented SYMIX at their Central stores. The Transfer of Spare Indent Details raised by the Field Engineers from CSS to SYMIX is required to be on-line, to reduce the Turn-around Time in servicing the spares to the indenting locations. The module **'Interfacing CSS to SYMIX'** constitutes updating SYMIX database residing in the Unix server from CSS front-end. This is done by establishing a connectivity between them by creating sockets.

CONTENTS

INTRODUCTION

About the Organization

Wipro is a diversified, integrated Corporation in Services, Technology Products and Consumer Products. Each business : Information Technology, Financial Services. Health Care Systems, Consumer Products and Lighting & Fluid Power - is headed by a President. The diversity of the Corporation is integrated through a strong, focussed drive for Leadership in each of the Business, Shared Values and Business Practices.

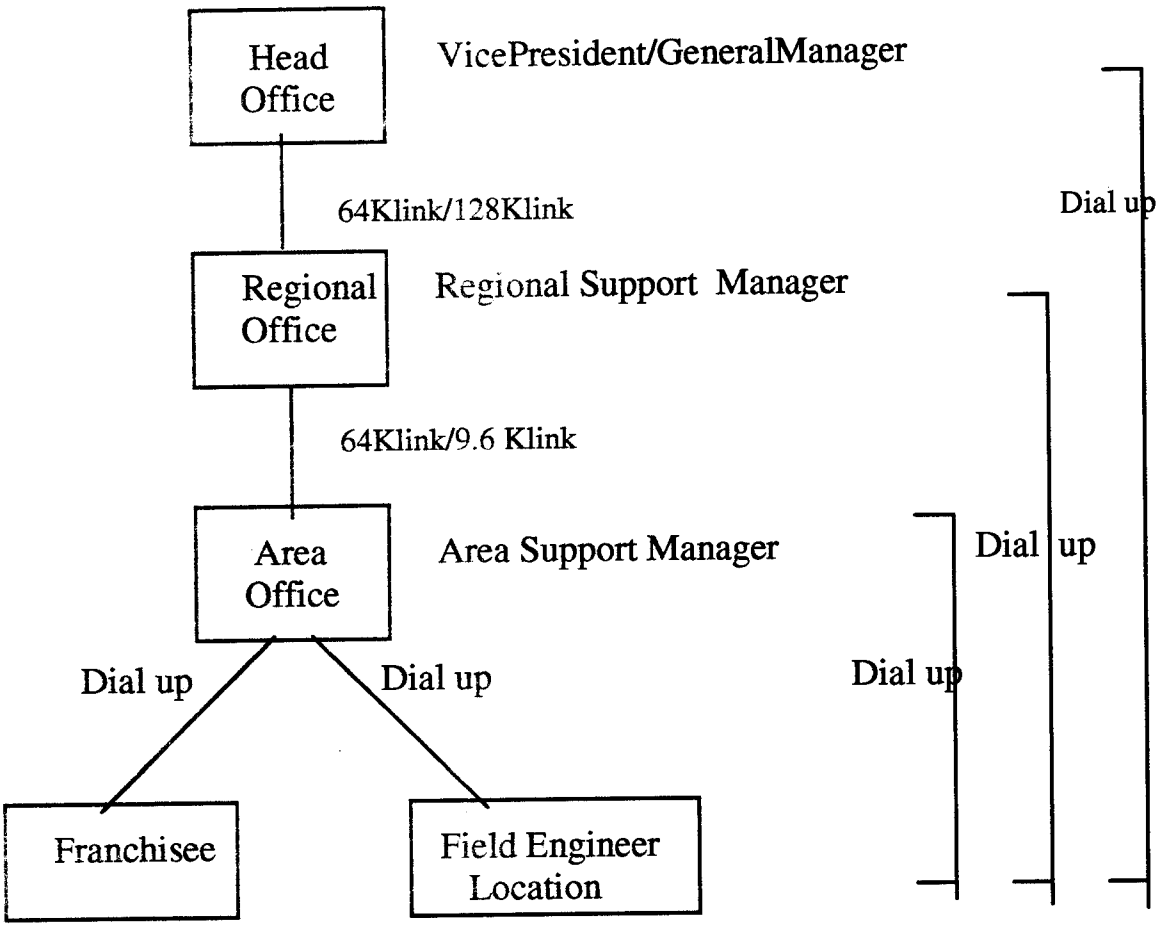
Wipro's Information Technology has a 14 year track record of continuous growth in Sales, Profitability and Customer Satisfaction which is unique in India. Wipro Infotech Group includes 2 joint venture Companies, Wipro Acer Ltd and Wipro BT Ltd.

The Support Division is part of the Infotech Group since 1981. Over the last 15 years, Support Division has grown to be the Maintenance Organization in the Country. Today the Organization supports a Large Client base totaling to more than 30,000 having a Machine base of 71,000 Micros and 4500 Mini Large Systems. This is in addition to the base of Network and Peripheral Clients. They are the core Assets of the Division. In the Global Operations, the Client list includes Sun Soft, Adaptec and UB Networks.

This Division operates on two strategic layers. One layer in the field comprising of the Regional Office, Area Offices and Engineer Locations. The other layer operates with 5 functions related to Technical Support, Business Development, Human Resource Development and Finance. The US operations are managed out of Wipro's Office at Cupertino, USA.

The Organization of the Division was done in such a way to be closest to the Customers. This was why they went into the leverage model where they offer their support through Core & Franchisees. The need was felt because of the increase in numbers of their Customers. Thereby the Core looks after the high-end segment and the Franchisees at the low-end segment

Organization Chart



Customer Support System

Hardware Configuration

Multi-user Version
Server :

- Acer Altos - Server
- P5/166 MHz Processor
- 64 MB RAM
- 1 GB X 2 SCSI Drives
- 525 MB Catridge Tape Drives
- SVGA Mono Monitor
- 16-bit Ethernet Card / 32 bit PCI Ethernet

Node :

- Genius 486 DX-2
- 8 MB RAM
- SVGA Mono Monitor
- 630 MB IDE Drive
- 16 Bit Ethernet Card

Single User Version :

- Genius 486 DX-2
- 16 MB RAM
- SVGA mono monitor
- 630 MB IDE drive
- 16 Bit Ethernet card

Software Configuration

MultiUser Version:
Server :
- SCO UNIX DNS Run Time
- 6 DDS - Development Version
- ORACLE 7.2

Node :
- Windows '95
- SQL*Net
- Powerbuilder Deployment Kit
- Microsoft Exchange for Windows 95 version 4

SingleUser Version:
- Windows 95
- Powerbuilder Deployment kit
- Microsoft Exchange for Windows 95 Version 4

SYMIX

Hardware Configuration

Acer Altos - Server
P5/166 MHz Processor
64 MB RAM
1 GB X 2 SCSI Drives
525 MB Catridge Tape Drives
SVGA Mono Monitor
16-bit Ethernet Card / 32 bit PCI Ethernet

Software Configuration

SCO UNIX Open Server
Progress 7.3

Wipronet Details

WIPRONET consists of various Wipro locations across the country interconnected for the purpose of sharing Data, Voice and FAX. The locations are webbed using either leased lines, VSATs or radiolinks whichever is feasible/desired.

The network is aimed to serve the following purposes :

1. Symix implementations across all the Wipro locations.
2. Remote access of UNIX,Windows and Novell servers across the networks.
3. FTP, Telnet, NFS operatios across TCP/IP hosts.
4. Mail transfer.
5. Voice and FAX access.
6. Internet and Intranet access.
7. Network Management of the entire network from a central site.

Interconnectivity is achieved by one of the following methods.

a) 128Kbps, 64Kbps and 9.6Kbps leased lines

b) VSAT networks from Wipro BT

c) Waveguides for line of site locations with a throughtput of 2Mbps

Features of PowerBuilder 5.0

New features :

- User Interface changes
- Code generation
- Distributed PowerBuilder
- Inbound OLE automation
- PowerBuilder Component Gallery
- PowerBuilder Foundation Class Library
- Powersoft ObjectCycle Overview
- DataWindow Enhancements
- Database Changes
- Object-oriented enhancements
- Internationalization
- PowerScript enhancements

User Interface Changes:

PowerBuilder 5.0 has enhanced the development user interface to incorporate a consistent presentation of information and a common look and feel throughout the application.

Code Generation:

In PowerBuilder 5.0, there is an option to compile applications in machine code for increased performance. The code generation feature:

Is available in 32-Bit environments (Windows 95 and Windows NT)

Uses the Watcom C++ compiler technology

Generates ANSI C code

Converts PBDs into DLLs

Distributed PowerBuilder:

PowerBuilder 5.0 gives the ability to build applications that run in a distributed computing environment. Distributed computing allows us to get the most out of our investment in the client/server architecture. By building distributed applications, we can:

Centralize business logic on servers

Partition application functions between the client and the server, thereby reducing the client workload

Make our applications scalable and easy to maintain

Distributed computing offers a natural way to separate user interface components from the business logic required by an application. In a distributed application, a client can invoke services provided by remote objects. A client can invoke methods (functions and events) that are associated with a remote object as if they were defined on a local object. Distributed PowerBuilder provides the following communication drivers:

Winsock

NamedPipes

OpenClientServer

Local

Inbound OLE Automation:

In PowerBuilder 5.0 non-visual objects may be driven through OLE (Object Linking and Embedding) automation. Driven means that any client program capable of OLE automation may create an instance of that type and read or write properties or invoke its methods.

Component Gallery:

PowerBuilder 5.0 includes a number of OLE custom controls (OCX) for use with the application development environment. The Component Gallery provides a collection of custom controls that encapsulate functionality that can be assembled into a larger application architecture. Other components that provide database access, network access, message handling, telecommunications and multi-media functions are now widely available. The properties, events and methods of the OLE custom controls become an extention of the standard PowerBuilder objects.

Object Cycle: ObjectCycle is a powerful tool for managing software components in a team setting. ObjectCycle can be used to quickly and easily set up development projects.The familiar Windows Explorer metaphor allows us to specify which objects to put under ObjectCycle's control.

ObjectCycle works with any type of object - including objects stored in PowerBuilder libraries such as windows, menus, DataWindows, OCX's, bitmaps, icons, DLLs (dynamic link libraries), and even Word Documents.

ObjectCycle Server resides on a network server, providing centralized version control services to multiple clients. ObjectCycle stores all objects in a relational database that resides on top of a multi-threaded, RPC-enabled server. The initial release supports Windows NT as a Win32 service process. Windows 95 is also supported as a shared peer-to-peer server.ObjectCycle is comprised of the following components:

- ObjectCycle Client - a library of software containing a thin layer RPC client API
- ObjectCycle Server - ObjectCycle server software and ObjectCycle Database
- ObjectCycle Client Utilities - Includes the PowerBuilder interface to ObjectCycle and optional software components used to connect to the ObjectCycle Server.

Datawindow Enhancements:

PowerBuilder 5.0 includes the following DataWindow enhancements

- New presentation styles
- DataWindow Painter enhancements
- Data manipulation enhancements

Database Changes:

PowerBuilder 5.0 has made following changes to Powersoft database interfaces:

ODBC support

Oracle support

Microsoft SQL Server support

OOE:

PowerBuilder 5.0 provides the following to its object oriented programming enhancements

Function overloading

Events using parameters

Function posting

Internationalization:

The following features have been added to PowerBuilder's Internationalization support:

International characters

Double-byte characters

Right-to-left orientation for cursor movement and alignment

MAPI Interface in PowerBuilder 5.0

PowerBuilder supports access to any e-mail system like MS-Mail that uses the mail standard called MAPI. MAPI stands for Message Application Program Interface and is a Programming Interface to Mail Systems. To build e-mail access into the application, some appropriate scripts are to be written using a few special mail functions that Powerbuilder supports. These mail functions will have effect on Windows 3.1 , Windows 95 and Windows NT platforms and they have no effect on Macintosh or Unix.

PROGRESS RDBMS

An Introduction

PROGRESS is a Relational Database Management System which supports Client/Server Architecture. It is a 4GL. It can be used on different platforms like DOS, OS/2, UNIX.etc., and allows remote access to databases which is an important feature useful in a distributed database system.

PROGRESS has compatible utilities with which non-PROGRESS routines can be executed. This feature helps in calling a C routine, which uses BSD Socket functions for Inter-Process Communication, from PROGRESS.

PROGRESS programming language is a very powerful language which supports all the facilities that are needed for full-fledged application development.

SYSTEM STUDY & ANALYSIS

CSS Overview

Customers,who have Service Contracts in terms of Warranty or Annual Maintenance (AMC) for their Machines,are provided with Service by Wipro Infotech. When there is a problem with the Machine or Unit of a Machine, customers communicate the same to Wipro and a Call is registered against the Customer and the Machine or Unit. Even if the Customer/Machine details are not found in the existing database,a call can be registered as a Not Found Call (NF). NF Calls can be either No Customer(NC) or No Machine(NM) calls. There is a facility to register calls for customers who do not own Machines but only certain Units like Terminals etc.Calls can be registered by employees also , even if the Customer doesn't call up, for Preventive Maintenance calls,Courtesy etc.

Calls can be of two types. Normal call and Project call. Normal calls include installation calls Preventive calls, Courtesy calls,Service calls,Inspection calls,Global Warranty Calls etc. The System facilitates scheduling of Preventive Maintenance calls based on Preventive Maintenace Norms. Calls are manually allocated to Field Engineers(FE) based on their skills and availability. FE attends to the Call and enters details of the service into the system through Field Engineering Service Report (FESR). The System has Provision for Call Allocation and FESR Entry . Calls can be pending because of FE, because of non-availability of spares or customer. If any spares are required to service the call. Spare Indents are raised with Stores. These Spare indent details are fed into the Inventory

Module in SYMIX to get the Spares serviced to the indenting location. System also facilitates monitoring of standbys.

Registered Calls can be Cancelled, in case, there is no need to attend to them. Monitoring the calls in terms of response time,machine downtine,time spent by FE,type of service provided are built into the system,and various reports and queries are provided for measuring Field Engineering Group's (FEG) performance and productivity.The system facilitates monitoring of High Call Machines in field.

There are 2 versions of CSS:

Multi-user version:

Client-server oracle 7.2 RDBMS

Powerbuilder 5.0 front end

This addresses the requirements of the Head office,Regional office and large AO. The database resides in a central server to which is connected several nodes accessing data. The server & nodes are networked.

Single-User Version:

This is used by small Wipro Area offices, single FE locations and franchisees. It is based on WATCOM database.The Database and CSS reside on the same machine.

SYMIX Overview

Support Division of M/s Wipro Limited has implemented the SYMIX Inventory module at their 'inter stores' in the field, thereby automating the entire stores activities.

Some of the functional areas of the Stores are not fully addressed by the SYMIX Inventory module and hence a few enhancements/modifications (Customization) are made to SYMIX to meet the specific requirements of the users.

The following are the various modules in SYMIX:

1. Order Processing
    - Customer Orders
2. Purchase Orders
3. Material Management
    - Inventory Control
    - Engineering Change Notice (ECN)
    - Material Resourse Planning (MRP)
4. Shop Floor Control
    - Job Order Processing
5. Capacity Management
6. Financials
    - General Ledger
    - Accounts Receivable (AR)
    - Accounts Payable (AP)
    - Fixed Assets
7. Employee Relations
    - Payroll
    - Human Resources
8. Data Collection

Need for E-Mail Interface to CSS

In CSS, the Customers are grouped under various categories. Norms are maintained for various categories of customers and various types of products. If the age of the customer call regarding problems, exceeds the time period specified in the norms, the call has to be escalated to the next higher level for further action. At the end of the day all pending calls in a location are to be transferred to their reporting location in the next higher level.

Previously, this Escalation Process was done manually. The manual process involves comparing each pending call against the time period specified in the norms, considering the various criteria and then the level to which a call is to be escalated was determined. The pending calls had to be monitored continuously to escalate it immediately when it exceeds the time period specified in the norms. The Problem was that the norms were not followed as it is and there was delay in escalation. Ultimately, this resulted in customer dissatisfaction.

Also the follow up of calls was a personal follow up by the person who is responsible for escalation rather than any specific system. Lot of interactions were done using telephones. The major disadvantage of using telephones is that :

i)We have to wait for the operator to connect and keep following her losing concentration on the job.

ii) And when connected , the person may not be there in his seat.

iii) The cost in explaining a problem over the phone is very high.

To overcome these problems , it is decided to use e-mail which is reliable now. The advantages of using e-mail are :

i) Usage of mail is cost-effective.

ii) The problem can be explained clearly.

iii) Individuals at either end need not be on-line.

If the escalation is made Automatic, then it will no longer be a personal follow up and will be escalated immediately when the norms are exceeded and the actions are forced by repeating the Escalation till the Problem is solved. This results in Customer Satisfaction.

The day-end process involves other than the house-keeping works, sending the pending calls in a location to its next higher level as a text file. This will be received at the other end and downloaded to the database for further follow-up.

Need for SYMIX Interface

CSS has the following modules :

COMMON - which maintains all master tables.

CMIF - which maintains Customer & Machines-in-field details.

CMON - which processes Customer calls regarding problems , projects and complaints.

REVENUE - which takes care of all the financial operations.

SYMIX takes care of all the Central stores activities related to Inventory of Spares.

These two systems require some common data to be passed on to each other. So, it was decided to link CSS and SYMIX so that when data is entered in one system, it should update the database in the other system also. The passing of data between CSS and SYMIX needs to be Automatic and transparent to the user. The first step to this link is to transfer the spare requirement details from CSS to SYMIX. There is no tool available for having Progress as back-end for the CSS front-end Powerbuilder so that it can update Progress database also. Moreover, this updation need to be on-line and reliable. And so it was decided to use the Socket facility.

At the **SYMIX end**, which runs on Unix, the Socket program can be done by **UNIX C** which has all the BSD compatible socket functions. This can be called from Progress, to update the tables. At the **CSS end**, Client Program can be done using the **Watcom C++ class builder** . The BSD socket functions are available in **WINSOCK.DLL** in Windows '95 and these functions can be called using the Class Builder.

Proposed System

The Proposed system "**Interfacing E-mail to CSS**" involves Automation of Escalation Process. This involves the following:

Master Maintenance:

This involves maintaining the employee details at various levels along with the mail system alias name and the e-mail address. This information is taken directly from the mail system address list so that no further validation need to be done. There is a provision for addition, deletion, modification and viewing the details.

Maintenance of Norms:

Norms are maintained for various categories of customers and for the various types of machines which includes the following information:

1) Category or Type of Machine

2) Duration

3) Escalation Level when the age of the call exceeds the norm

Fixing the Timer:

A Timer is fixed , at the elapse of which the system has to automatically check for Pending calls and invoke the timer event. There is Provision for varying the fixed time. User Permissions are considered in fixing the time.

Timer Event:

Timer event does the following:

1) Get all the calls which are pending and the calls which are closed but previously escalated.

2) Determine the Level of Escalation based on the various criteria and getting the e-mail addresses and alias names from the master table.

3) Forming the Text file which contains the problem related details and sending the mail.

4) And Updating the tables.

Uploading Data:

This involves Forming the Text file of all the Pending calls and sending it as an Attachment files through mail to the next higher location.

Downloading Data:

This involves Reading the Attachment file from the mail and Updating the table.

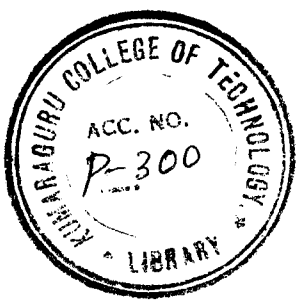
The System "**Interfacing CSS to SYMIX**" involves Spare Demands Requests to flow automatically from CSS to SYMIX Inventory Control System , to service the required spares to indending locations and Acknowledgement of Receipt of Demands Request to be sent from SYMIX to CSS. This requires: Server Process at the SYMIX side and a Client Process at the CSS side.

Client Process:

Whenever Spare Requirements are entered in CSS, the data is sent through the Socket using any port that is available at that time. An acknowledgement is received from the Server Process. If the acknowledgement is not received by the client , the data is resent.

Server Process:

The Server listens at some known port to service client requests. When data is arrived at this port, the Server Process forks to the Child Process with all the required information and the child will do the requested process and sends an acknowledgement to the client and kills itself. This is done to facilitate Multi-threading so that the Server process can listen to all the clients requesting for service at the same time.

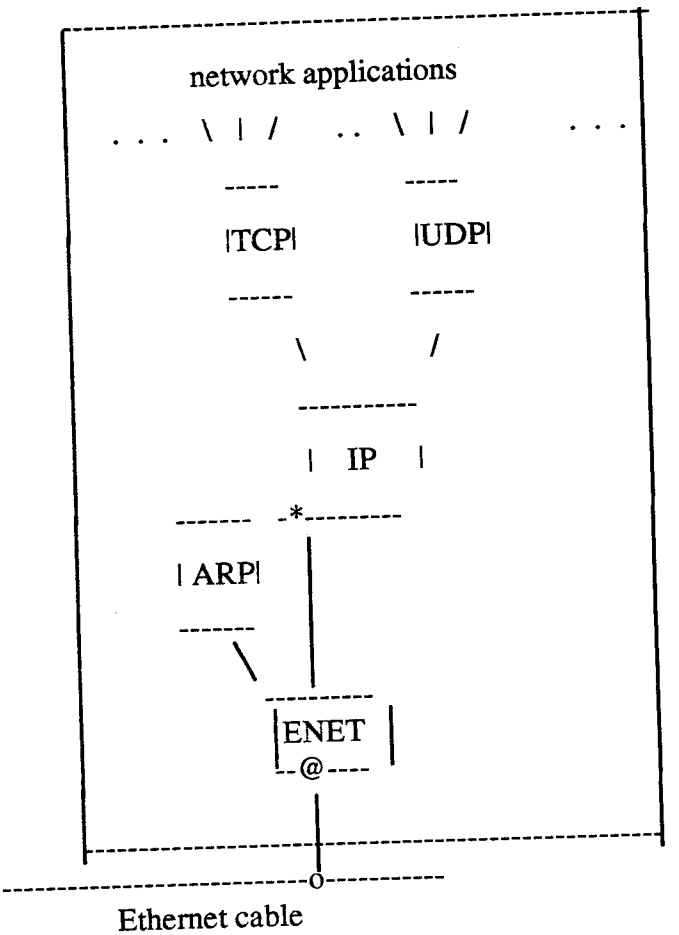


TCP/IP Overview

The generic term 'TCP/IP' usually means anything and everything related to the specific protocols of TCP and IP. It can include other protocols, applications, and even the network medium. A sample of these protocols are : UDP, ARP and ICMP. A sample of these applications are :TELNET, FTP, and RCP.A more accurate term is 'internet technology'. A netwok that uses internet technology is called an 'internet'

Basic Structure

To understand this technology we must first understand the following logical structure:



This is the logical structure of the layered protocols inside a computer on an internet. Each computer that can communicate using internet technology has such a logical structure. It is this logical structure that determines the behaviour of the computer on the internet. The boxes represent processing of the data as it passes through the computer, and the lines connecting boxes show the path of data. The horizontal line at the bottom represents the Ethernet cable; the 'o' is the transceiver. The " * " is the IP address and the "@" is the Ethernet address. Understanding this logical structure is essential to understanding internet technology.

For an application that uses TCP (Transmission Control Protocol), data passes between the application and the TCP module. FTP (File Transfer Protocol) is a typical application that uses TCP.

The computer has a 4-byte IP address.This address is located at the lower interface to the IP module. The IP address must be unique for an internet. The network manager assigns IP addresses to the computer according to the IP network to which the computer is attached. One part of a 4-byte IP address is the IP network number, the other part is the IP computer number (or host number).

People refer to computers by names, not numbers. A computer called alpha might have the IP address of 223.1.2.1. For small networks, this name-to-address translation data is often kept on each computer in the 'hosts' file.

Direct Routing:

Alpha is sending an IP packet to beta. The IP packet is in alpha's IP module and the destination IP address is beta or 223.1.2.2. IP extracts the network portion of this IP address and scans the first column of the table looking for a match. With this network a match is found on the first entry.

The other information in this entry indicates that computers on this network can be reached directly through interface number 1. An ARP table

translation is done on beta's IP address then the Ethernet frame is sent directly to beta via interface number 1.

If an application tries to send data to an IP adddress that is not on the development network, IP will be unable to find a match in the route table. IP then discards the IP packet. Some computers provide a 'Network not reachable' error message.

Tranmission Control Protocol

TCP offers a connection-oriented byte stream , instead of connectionless datagram delivery service. TCP guarantees delivery. TCP is used by network applications that require guaranteed delivery and cannot be bothered with doing time-outs and retransmissions. Two most typical network applications that use TCP are File Transfer Protocol (FTP) and the TELNET. Other popular TCP network applications include X-Window System, rcp(remote copy), and the r-series commands. TCP's greater capability is not without cost: it requires more CPU and network bandwidth. The internals of the TCP module are much more complicated than those in UDP module.

Network applications connect to TCP ports. Well-defined port numbers are dedicated to specific applications. For instance, the TELNET server uses port number 23. The TELNET client can find the server simply by connecting to port 23 of TCP on the specified computer.

When the application first starts using TCP, the TCP module on the client's computer and the TCP module on the server's computer start communicating with each other. These two end-point TCP modules contain state information that defines a virtual circuit. This virtual circuit consumes resources in both TCP end-points. The virtual circuit is full duplex; data can go in both directions simultaneously. The application writes data to the TCP port, the data traverses the network and is read by the application at the far end.

TCP packetizes the byte stream at will; it does not retain the boundaries between writes. For example, if an application does 5 writes to the TCP port, the application at the far end might do 10 reads to get all the data. Or it might get all the data with a single read. There is no correlation between the number and size of writes at one end to the number and size of reads at the other end.

TCP is a sliding window protocol with time-out and retransmits.Outgoing data must be acknowledged by the far-end TCP.Acknowledgements can be piggybacked on data. Both receiving ends can flow control the far end, thus preventing a buffer overrun.

As with all sliding window protocols, the protocol has a window size. The window size determines the amount of data that can be transmitted before an acknowledgement is required. For TCP, this amount is not a number of TCP segments but a number of bytes.

If we need a reliable stream delivery service and efficiency over long-haul circuits, TCP might be best.

Basic Concepts of Socket Programming

A network application links two computers that want to communicate, so they need a connection over which to exchange data. In TCP/IP, this connection is called a Socket. A Socket is an end-point of communication through which hosts can communicate.It is an application layer concept of networking which hides the internal working of other layers.This is possible because of TCP/IP. TCP sockets are Connection-Oriented and hence suitable for On-line communication.

Each Socket contains

The numeric IP address of the local computer.

The local port number.

The port number of the TCP/IP service on the remote computer.

To do Socket programming,

* The Operating systems must support TCP/IP.
* The Workstations must be in the Network(i.e., it must have an IP address).
* The Socket programs must be written at both Server and Client ends.
* The Operating systems must have the functions required for creating sockets,opening & closing connections,sending & receiving data.

Socket Programming is possible for communication between CSS & SYMIX because:

* Unix communicates using TCP/IP. And Windows'95 , in addition to Novell's SPX/IPX supports TCP/IP.

* Unix has all the required functions for creating sockets,opening & closing connections,etc., This extensive programming library was originally developed as part of the development of Unix & TCP/IP itself.

Windows'95 has WINSOCK.DLL to support socket programming. This Windows Socket API is baed on the Berkeley Unix Programming library.

The Client/Server System

At the server side , the server waits listening to the clients at some port. Once the client opens a socket connection (TCP socket) the IP-Address of the client, the port number is passed along with the data (request). All this information is passed as the TCP/IP Header block. The server accepts the connection and will fork a child process which gets the request and processes it. Meanwhile the server will go back to the listening state. This is because the server should be able to get the requests from various clients. So every client has got a child process at the server-end running, processing the request. Once the request result is gathered, the child will send back the result thru the same socket and kills itself closing the socket.

The client when opening the socket to the server should know to which port to send the request and the child process at the server-end which processes the request knows to which client the result should be sent. For example, if the server is listening at port the 2121 the client sends the request to the 2121 port through some port which is available at that time say 3010. The server gets the request, forks a child process to process the request and will send the result to port 3010 of the client machine thru some port which is available at that time.All this will be taken care by TCP/IP protocol.

SYSTEM DESIGN

Design Considerations

Methodology:

Client/Server technology is used as development and implementation environment for CSS.The client platform is GUI based front end - Power Builder Ver 5.0,supporting object oriented software development.The Server platform is ORACLE 7.2.

Object Oriented design methodologies emphasise on abstraction of the problem using real world concepts,rather than computer concepts.Also, implementation aspects are deferred to the later stages of design to preserve the flexibility of the design.It prefers reusability code and encapsulation of application objects.Considering the features and design objectives of CSS and the inherent advantages in Object Oriented design methodologies,it was decided to adopt the same for designing CSS.

Various object oriented design methodologies were evaluated and Object Modelling Technique by James Rumbaugh was chosen for designing CSS.

Object Modelling Technique enables the designer to model objects from the real world.Using these models, language-independent design can be built. This design can then be implemented on any platform as required.

The fundamental concept in OMT is an object. An object encapsulates both data structure and behaviour in a single entity.Similar objects are grouped into classes and sub-classes based on their common attributes and behaviour. Sub-classes inherit all the properties of the super-class and in addition have attributes/behaviour that are specific to the sub-class.

OMT covers the following stages in which the design is to be carried out:

Analysis: This phase is used to arrive at a concise and precise abstraction of the system requirements. It enables the analyst to clearly define what the system should do but not how to do it.

System Design: In this stage, the designer makes high-level descisions on the break up of the system into sub-systems,the overall architecture,performance criteria,etc.

Object design: Design model is built based on the analysis model. Implementation details and any decisions taken during the system design phase are added to arrive at the complete object design.

In analysis stage, OMT uses three models to describe the system. These models get further refined during the system design and object design stages.

The object model is the heart and soul of OMT. It describes the objects in the system, their classes and sub-classes and their relationships with other objects.

The dynamic model describes the sequence of interactions between the objects in the system. It describes the objects of the system that change over time. Event Trace diagrams and state diagrams are the outcome of dynamic modelling.

The Functional model describes the internal data value transformations of objects within the system by the way of data flow diagrams.

These three models are then combined to arrive at the object design. Attributes of each class are identified in object model. The operation pertaining to events and state transition are identified using Dynamic model. The other operations are derived from the Functional model . Each operation is associated with the object that it is acting upon. If there are multiple objects for the operation , the most central object is chosen to associate the operation. Thus , all attributes and operations of a class along with the

relationships with other classes forms the object design for the system. Class diagrams are used to depict the object design.

Operations can be identified and designed optimally using the following guidelines:

Each class must be responsible for performing operations.

Interfaces are to be defined at the highest possible level of abstraction.

The object model for CSS is implemented using Relational database design priciples. The operations for the objects will be implemented using PowerBuilder and Oracle development tools.

The OMT methodology has played an important role in building graphical user interface for CSS. The object model is the most important input to build the prototype of user interface.The real world objects and their associations with other objects provide vital input for building intuitive navigation in the user interface. Dynamic model has helped in building control structure of user interface.

Design Issues:

The design of CSS is done to make best use of Client/Server Architecture. This is ensured by using the desktop computing for presentation of data and implementation of business logic. The powerful database server will do all the database intensive porcessing.The entire user interface is redesigned to use the features and flexibility provided by the Graphical User Interface tools.Interface of CSS is designed to be intuitive and conductive to work flow of the users for the function.

Database Design

Customer Support System

Table Name: Call_hdr

Column Name	Data Type	Width	Dec	Null	Default
call_no	char	12		No	
call_date	timestamp			No	
mc_sl_no	char	14		Yes	
cust_code	char	9		Yes	
account_code	char	5		Yes	
account_name	varchar	40		No	
cust_addr1	varchar	40		No	
cust_addr2	varchar	40		Yes	
cust_addr3	varchar	40		Yes	
contract_type	numeric	6	0	Yes	
territory	char	2		Yes	
mc_type	numeric	6	0	Yes	
excl_onsite	char	1		No	
stat_code	numeric	6	0	No	
call_source	char	1		No	
comp_proj_no_ref	char	12		Yes	
to_be_att_date	timestamp			Yes	
chargeable	char	1		No	
qualified	numeric	6	0	No	
esc_flag	char	1		Yes	
esc_date	date			Yes	
hw_sw	char	1		No	
call_type	numeric	6	0	No	
first_att_date	timestamp			Yes	
closed_date	timestamp			Yes	
visits	numeric	6	0	No	
critical_ind	char	1		Yes	
mdfd_dttm	date			No	
remarks	char	40		Yes	
closed_flag	integer			Yes	
esc_level	integer			Yes	
esc_dttm	timestamp			Yes	

Primary Key : call_no.

Table Name: Call_dtl

Column Name	Data Type	Width	Dec	Null	Default
call_no	char	12		No	
sl_no	numeric	6	0	No	
unit_code	numeric	6	0	No	
unit_sl_no	numeric	14	0	Yes	
prob_code	numeric	6	0	Yes	
prob_desc	varchar	50		No	
status	numeric	6	0	No	
solution	varchar	40		Yes	
serv_prov	numeric	6	0	Yes	

Primary Key : call_no & sl_no

Table Name : Call_allocation

Column Name	Data Type	Width	Dec	Null	Default
call_no	varchar	12		No	
emp_code	numeric	6	0	No	
allc_date	timestamp			No	

Pimary Keys : call_no,emp_code &allc_date

Table Name: Customer_end_location

Column Name	Data Type	Width	Dec	Null	Default
cust_code	char	9		No	
account_code	char	5		No	
cust_addr1	varchar	40		No	
cust_addr2	varchar	40		Yes	
cust_addr3	varchar	40		Yes	
city_code	numeric	6	0	No	
stat_code	numeric	6	0	No	
pin	char	6		Yes	

privileged_ind	char	1		No
site_locn	char	1		No
territory	char	2		Yes
sent_down_ind	char	1		Yes
lst_no	varchar	20		Yes
lst_date	date			Yes
cst_no	varchar	20		Yes
cst_date	date			Yes
mdfd_dttm	date			No
dummy_account	varchar	40		Yes
category	numeric	6	0	Yes

Primary Key : cust_code

Table Name: Cust_priority

Column Name	Data Type	Width	Dec	Null	Default
category	integer			No	
level	integer			No	
duration	numeric	6	2	No	

Primary Key : category & level

Table Name: Code_master

Column Name	Data Type	Width	Dec	Null	Default
code_type	numeric	6	0	No	
code_value	numeric	6	0	No	
code_desc	varchar	40		No	
code_sname	varchar	4		Yes	
code_ind	varchar	1		Yes	
code_state	varchar	1		No	
mdfd_dttm	timestamp			Yes	

Primary Keys : code_type & code_value

Table Name: Employee

Column Name	Data Type	Width	Dec	Null	Default
emp_code	numeric	6	0	No	
emp_name	varchar	40		Yes	
emp_grade	numeric	6	0	No	
emp_addr1	varchar	40		Yes	
emp_addr2	varchar	40		Yes	
emp_sname	char	6		No	
emp_skill_area	varchar	40		Yes	
emp_group	char	10		Yes	
emp_phone	char	15		Yes	
emp_dsgn	varchar	40		Yes	
emp_type	numeric	6	0	No	
emp_stat	char	1		No	
emp_locn	numeric	6	0	Yes	
escalate_to	numeric	6	0	Yes	
emp_amc_norm	numeric	6	2	Yes	
mdfd_dttm	date			Yes	

Primary Key : emp_code

Table Name: Esc_call_out

Column Name	Data Type	Width	Dec	Null	Default
call_no	char	12		No	
eac_type	char	1		No	
esc_date	date			No	
esc_to_emp	numeric	6	0	No	
esc_to_locn	numeric	6	0	Yes	
closed_date	date			Yes	
user_crtd	numeric	6	0	No	
mdfd_dttm	date			No	

Primary Keys :call_no & esc_type

Table Name: Esc_call_suggested_activities

Column Name	Data Type	Width	Dec	Null	Default
call_no	char	12		No	
activity_sl_no	numeric	6	0	No	
activity_info	varchar	160		No	
mdfd_dttm	date			No	
user_crtd	numeric	6	0	No	

Primary Key : call_no & activity_sl_no

Table Name: Esc_duration

Column Name	Data Type	Width	Dec	Null	Default
duration	integer			No	

Table : Esc_person_at_levels

Column Name	Data Type	Width	Dec	Null	Default
id	numeric	6	0	No	
name	varchar	25		Yes	
designation	numeric	6	0	Yes	
loc_code	numeric	6	0	Yes	
e_mail	varchar	35		Yes	
mailbox_name	varchar	35		Yes	
resp_flag	char	1		Yes	

Primary Key : id

Table Name : Escalated_call

Column Name	Data Type	Width	Dec	Null	Default
call_no	char	12		No	
call_date	date			No	
mc_sl_no	char	14		Yes	
esc_to_emp	numeric	6	0	No	
excl_onsite	char	1		No	
stat_code	numeric	6	0	No	
esc_type	char	1		No	
to_be_att_date	date			Yes	
chargeable	char	1		No	
qualified	numeric	6	0	No	
esc_date	date			Yes	
hw_sw	char	1		Yes	
call_type	numeric	6	0	No	
first_att_date	date			Yes	
closed_date	date			Yes	
account_id	char	5		Yes	
cust_addr1	varchar	40		No	
cust_addr2	varchar	40		Yes	
cust_addr3	varchar	40		Yes	
esc_to_locn	numeric	6	0	Yes	

Primary Key : call_no

Table Name: Escalated_call_dtl

Colmn Name	Data Type	Width	Dec	Null	Default
sl_no	numeric	6	0	No	
call_no	char	12		No	
unit_code	numeric	6	0	No	
unit_sl_no	numeric	6	0	Yes	
prob_code	numeric	6	0	Yes	
prob_desc	varchar	50		No	
status	numeric	6	0	No	
solution	varchar	30		Yes	
serv_prov	numeric	6	0	Yes	

Primary Key : sl_no & call_no

Table Name: Item_master

Column Name	Data Type	Width	Dec	Null	Default
item_code	char	10		No	
item_desc	varchar	25		No	
item_dtldesc	varchar	40		Yes	
item_level	integer			No	
return_ind	char	1		No	
item_class	char	1		Yes	
lot_size	integer			Yes	
uom	integer			No	
active_state	char				'A'
import_state	char	1		No	'D'
war	numeric	5	3	No	
proc_ind	char	1		No	'Y'
crtd_dttm	date			Yes	
mdfd_dttm	date			Yes	
item_type_code	integer			Yes	
repair_state	char	1		No	
item_sl_ind	char	1		Yes	

Primary Key : item_code

Table Name : Mif_configuration

Column Name	Data Type	Width	Dec	Null	Default
mc_item_sl_no	char	14		No	
manufacturer	char	10		No	
prin_or_mach	char	10		No	

Primary Key : mc_item_sl_no

Table Name : Mif_detail

Column Name	Data Type	Width	Dec	Null	Default
mc_item_sl_no	varchar	14		No	
mif_sl_no	numeric	6	0	No	
item_code	varchar	10		No	
item_sl_no	varchar	14		Yes	
mif_stat	numeric	6	0	No	
serv_stat	numeric	38	0	No	
brought_back_ind	varchar	1		No	
stand_by_ind	varchar	1		No	
item_level	numeric	6	0	No	

date_of_install	timestamp			Yes
hir_signed_ind	varchar	1		No
ecn_level	varchar	15		Yes
enhanced_ind	varchar	1		No
hir_no	varchar	20		Yes
hir_date	timestamp			Yes
chargeable_ind	varchar	1		Yes
parent_sl_no	numeric	38	0	No
service_type	varchar	1		Yes
wty_contract_no	varchar	20		Yes
start_date	date			Yes
end_date	date			Yes
wty_contract_ext_no	varchar	20		Yes
wty_contract_type	numeric	6	0	Yes
mdfd_dttm	timestamp			No
despatch_locn	numeric	6	0	Yes
dc_no	varchar	20		Yes
dc_date	timestamp			Yes

Primary Key: mc_item_sl_no & mif_sl_no

Table Name : Mif_header

Column Name	Data Type	Width	Dec	Null	Default
mc_item_sl_no	char	14		No	
item_code	char	10		No	
ship_to_cust	char	9		No	
bill_to_cust	char	9		No	
order_by_cust	char	9		No	
supported_by_locn	numeric	6	0	No	
attached_to_locn	numeric	6	0	No	
ordered_to_locn	numeric	6	0	No	
billed_by_locn	numeric	6	0	No	
mif_stat	numeric	6	0	No	
serv_stat	numeric	6	0	No	
stop_serv_ind	char	1		No	
brought-back_ind	char	1		No	
networked_ind	char	1		Yes	
stand_by_ind	char	1		No	
wipro_mc_ind	char	1		Yes	
division_code	numeric	6	0	No	
aux_ser_sl_no	char	14		Yes	
opf_no	varchar	20		Yes	
opt_date	date			Yes	
opt_type	numeric	6	0	Yes	

hir_no	varchar	20		Yes
hir_date	date			Yes
hir_signed_ind	char	1		No
date_of_install	date			Yes
despatch_locn	numeric	6	0	Yes
dc_no	varchar	20		Yes
dc_date	date			Yes
sold_by	char	1		Yes
service_type	char	1		Yes
wty_contract_no	varchar	20		Yes
start_date	date			Yes
end_date	date			Yes
wty_contract_ext_no	varchar	20		Yes
wty_contract_type	numeric	6	0	Yes
der_cust_type	numeric	6	0	No
mdfd_dttm	date			No
item_desc_dummy	varchar	25		Yes
account_code_dummy	varchar	5		Yes
ship_to_dummy	varchar	40		Yes
ordered_by_cust_dummy	varchar	40		Yes
account_name_dummy	varchar	40		Yes

Primary Key :mc_item_sl_no

Table Name : Office_time_rep

Column Name	Data Type	Width	Dec	Null	Default
morn_st_time	numeric	5	2	No	
morn_end_time	numeric	5	2	Yes	
aft_st_time	numeric	5	2	Yes	
aft_end_time	numeric	5	2	Yes	

Table Name : Place_master

Column Name	Data Type	Width	Dec	Null	Default
place_code	numeric	6	0	No	
place_type	char	1		No	
place_name	varchar	40		No	
parent_code	numeric	6	0	No	
mdfd_dttm	date			Yes	

Primary Key : place_code & place_type

Table Name: Spare_indent_dtl

Column Name	Data Type	Width	Dec	Null	Default
indent_no	char	12		No	
item_code	char	10		No	
qty_indented	numeric	6	0	No	
qty_issued	numeric	6	0	Yes	
qty_canc	numeric	6	0	Yes	
closed_canc_date	date			Yes	
mdfd_dttm	date			No	
user_crtd	numeric	6	0	No	

Primary Key : indent_no & item_code

Table Name : Spare_Indent_hdr

Column Name	Data Type	Width	Dec	Null	Default
indent_no	char	12		No	
call_no	char	12		No	
indent_date	date			No	
emp_code	numeric	6	0	No	
status	numeric	6	0	No	
mdfd_dttm	date			No	
user_crtd	numeric	6	0	No	

Primary Key : indent_no

Table Name : Unit

Column Name	Data Type	Width	Dec	Null	Default
unit_code	numeric	6	0	No	
problem_code	numeric	6	0	No	

Primary Key : unit_code & problem_code

Table Name : Fe_attendance

Column Name	Data Type	Width	Dec	Default
locn	numeric	6	0	No
fe_date	date			No
fe_days_on_roll	numeric	6	2	Yes
fe_days_on_leave	numeric	6	2	Yes
fe_days_on_metro	numeric	6	2	Yes
fe_days_on_remote	numeric	6	2	Yes
fe_days_on_training	numeric	6	2	Yes
fe_days_on_special_assignment	numeric	6	2	Yes
fe_days_assigned_in	numeric	6	2	Yes
fe_days_assihned_out	numeric	6	2	Yes
hpr_engg	numeric	6	2	Yes

Table Name : Location

Column Name	Data Type	Width	Dec	Null	Default
locn_code	numeric	6	0	No	
locn_name	varchar	40		No	
locn_addr1	varchar	40		No	
locn_addr2	varchar	40		No	
locn_addr3	varchar	40		No	
city_code	numeric	6	0	No	
stat_code	numeric	6	0	No	
pin	numeric	6	0	Yes	
reporting_to	numeric	6	0	No	
locn_type	char	2		No	
locn_sname	char	2		Yes	
mdfd_dttm	date			Yes	

Table Name : Site_parms

Column Name	Data Type	Width	Dec	Null	Default
curr_locn_code	numeric	6	0	No	
dbms_name	char	1		No	

auto_ex_to_emp	numeric	6	0	Yes
year_code	char	2		No
def_xfer_mode_to_ho	numeric	6	0	Yes
def_xfer_mode_to_ro	numeric	6	0	Yes
def_xfer_mode_to_ao	numeric	6	0	Yes
def_xfer_mode_to_fe	numeric	6	0	Yes
def_xfer_mode_to_fr	numeric	6	0	Yes
def_esc_norm_is_hrs	numeric	6	0	Yes
def_high_call_norm_per_mnth	numeric	6	0	Yes
def_data_out	varchar	10		Yes
def_data_in	varchar	10		Yes

SYMIX

Table Name : lindemln

Field Description	Type	Width
Fin. Year	Numeric	4
Dem. Type	Alpha	2
From Whse	Alpha	4
To Whse	Alpha	4
Doc.No	Alpha	8
Emp/Fran Code	Alpha	7
Customer Code	Alpha	7
Purpose	Alpha	25
SlNo	Numeric	3
Demand No	Numeric	6
Item Code	Alpha	30
Item Description	Alpha	40

Input Design

Input design is the process of converting the user originated inputs into a computer based format. The goal of the input design is to make the data entry easier, logical and free from errors. Errors in the input data are controlled by input design.

This application has been developed in a user friendly manner. The forms have been designed in such a way that during the processing the cursor is placed in the position where the data must be entered.

The user is also provided with an option of selecting an appropriate input from the list of values. Validation are made for each and every data entered. Whenever the user types erroneous data, error message is displayed and user can move to the next field only after entering the correct data.

Dropdown data window has been provided for the list of valid data for certain fields where the selection can be made by the user.

Output Design

An application is successful only when it can produce efficient and effective reports. The reports generated must be useful to the management and for the history reference. For the user/operator//management, reports provide a permanent hard copy of transactions occured. Careful consideration has been given in designing the reports as it helps in decision making.

All the reports generated are concise with the necessary information.The reports can be viewed on the screen, printed or can be stored in a file.

The following are the reports generated:

1. Incoming Escalated Calls
2. Outgoing Escalated Calls

CONCLUSION

CONCLUSION

The Limitation in the E-mail Interface is that the system cannot continuously monitor the Pending Calls and Escalate immediately. The System escalates the Pending Calls only at the elapse of the time fixed to check for Pending Calls. So, if the Call exceeds the duration specified in the Norm just after the last escalation, only during the next escalation the Call will be escalated, which causes delay.

Other than this, the system works fine and is flexible to include any modification required in future.

SCOPE OF THE PROJECT

SCOPE OF THE PROJECT

This Project can be extended to receiving the customer service requests through mail and registering the problems automatically. The various files to be sent from one location to another can be done by sending them as attachment files through mail and performing the required processes. This can be extended to have a complete interface between CSS and SYMIX avoiding duplication of work and sharing of data between two databases.

BIBLIOGRAPHY

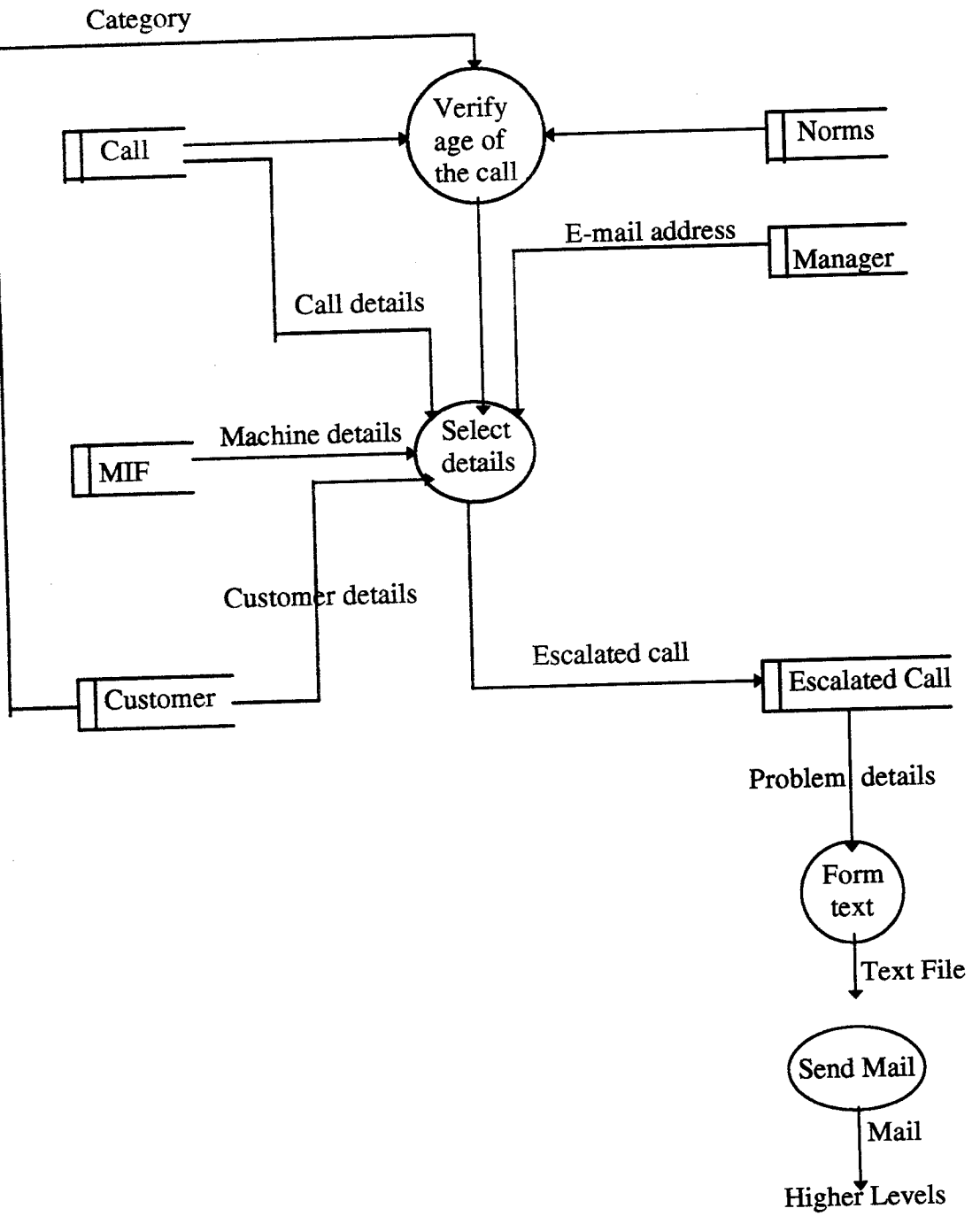
BIBLIOGRAPHY

1. UNIX Network Programming - W.Richard Stevens, Prentice Hall of India.
2. TCP/IP for DUMMIES - Marshall Wilensky & Candace Leiden, Comdex Computer Publishing.
3. Short Term Intensive Course on Unix Networking & Network Programming - Indian Institute of Science, Bangalore.
4. An Open Interface for Network Programming under Microsoft Windows - Martin Hall ,Mark Towfiq and others.
5. Berkeley UNIX System Calls & Interprocess Communication - Lawrence Besaw.
6. Application Development using PowerBuilder 4.0 - Rajkumar Sedani, BPB Publications.
7. PowerBuilder 5.0 Manuals.

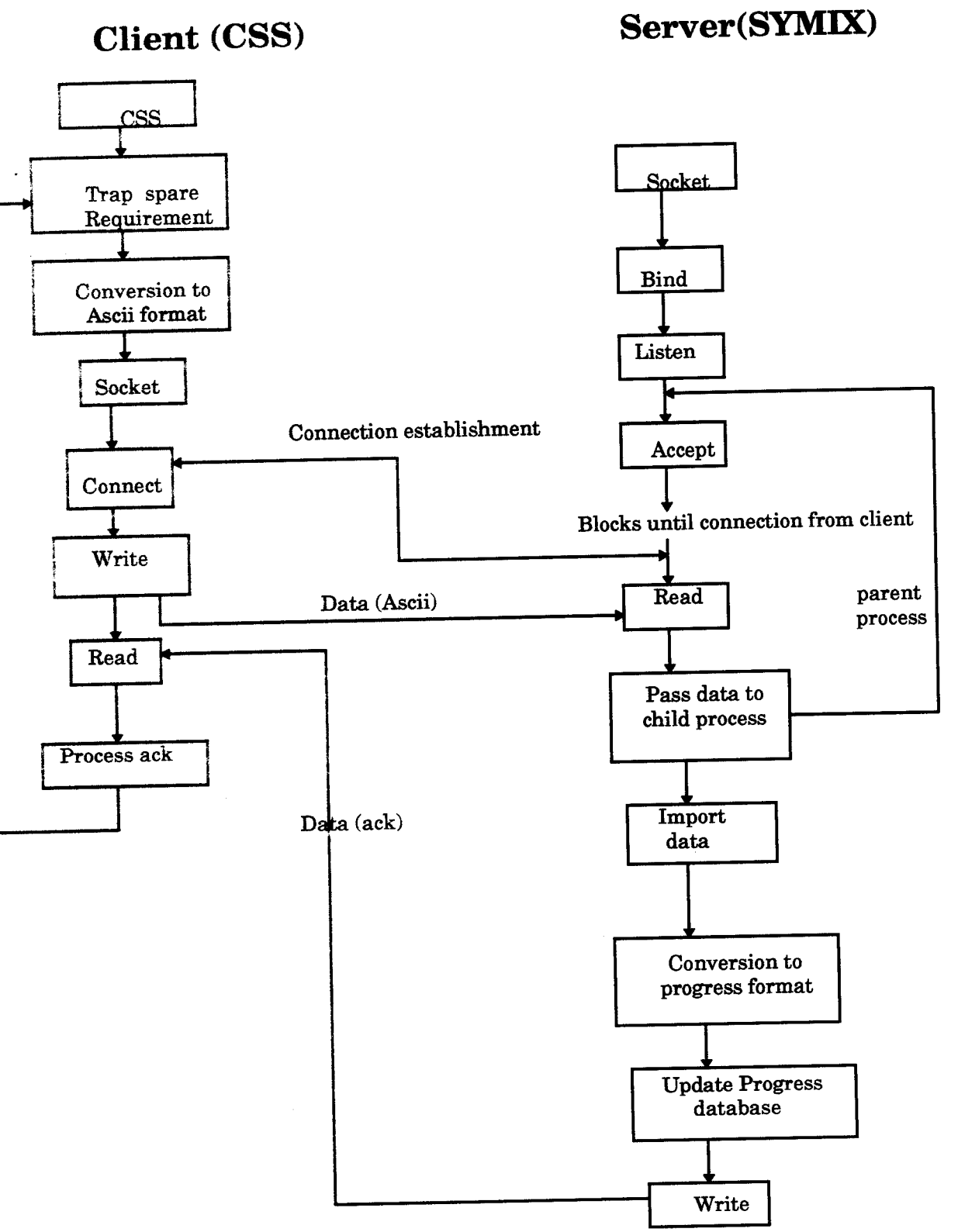
APPENDIX

FLOWCHARTS

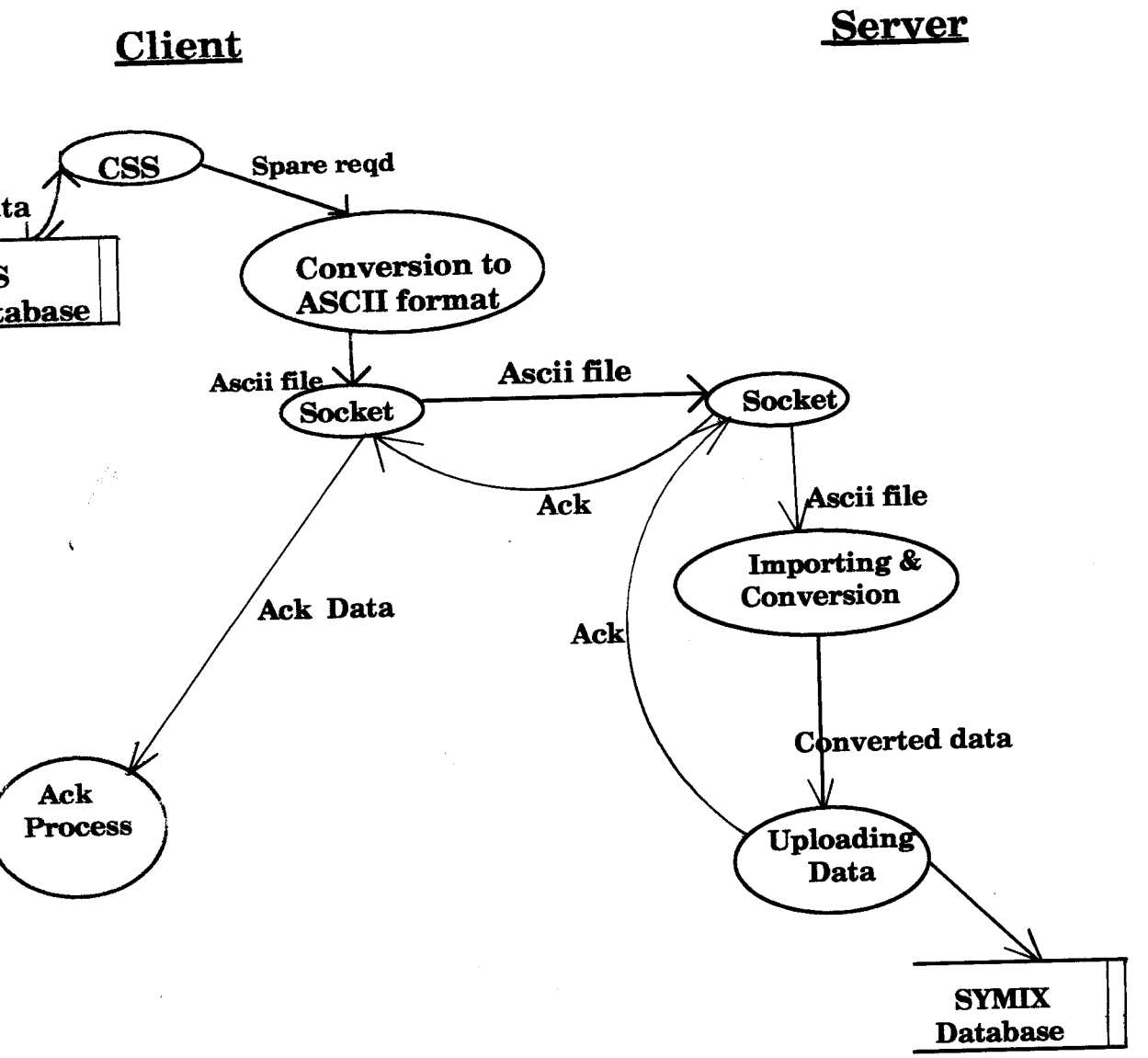
Call Escalation - Data Flow Diagram



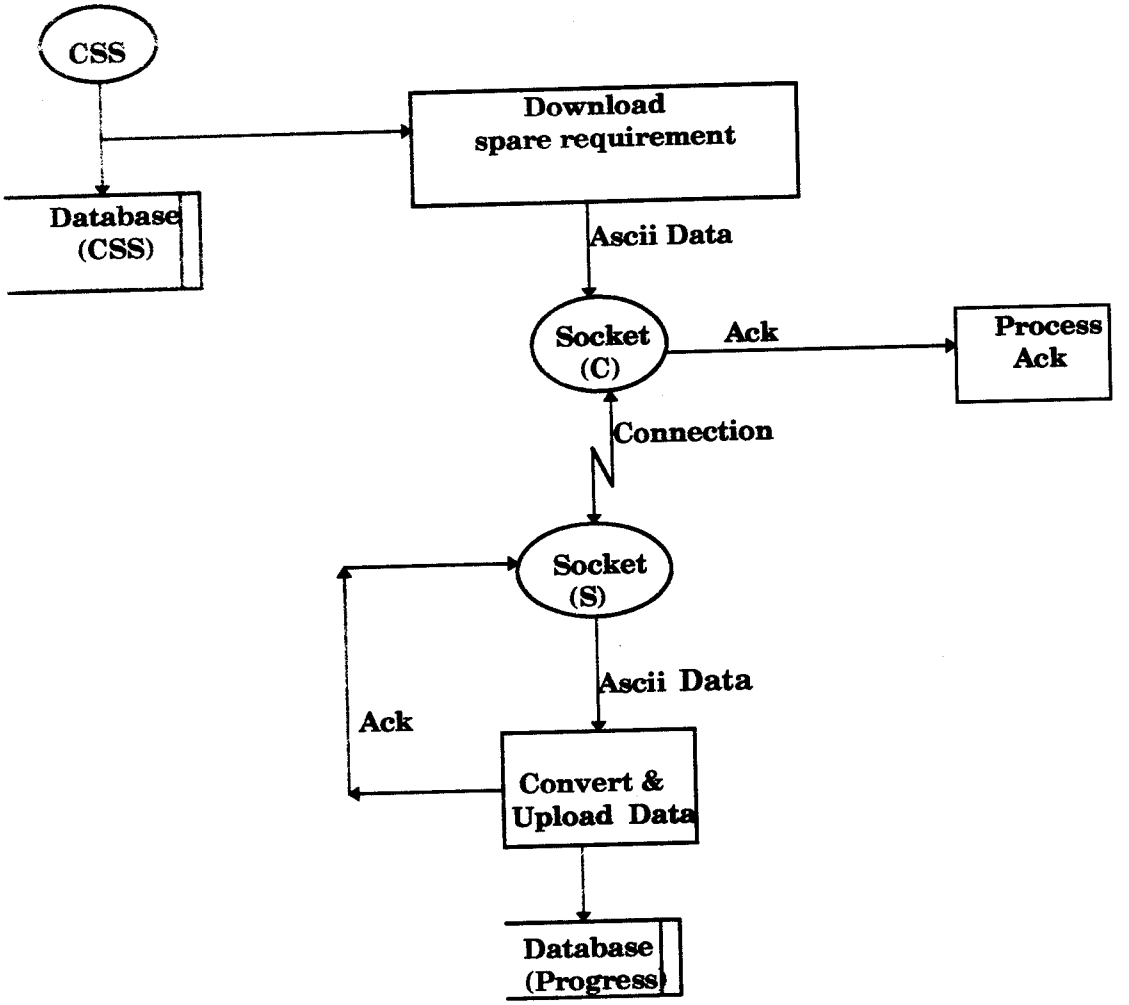
System Flow Diagram



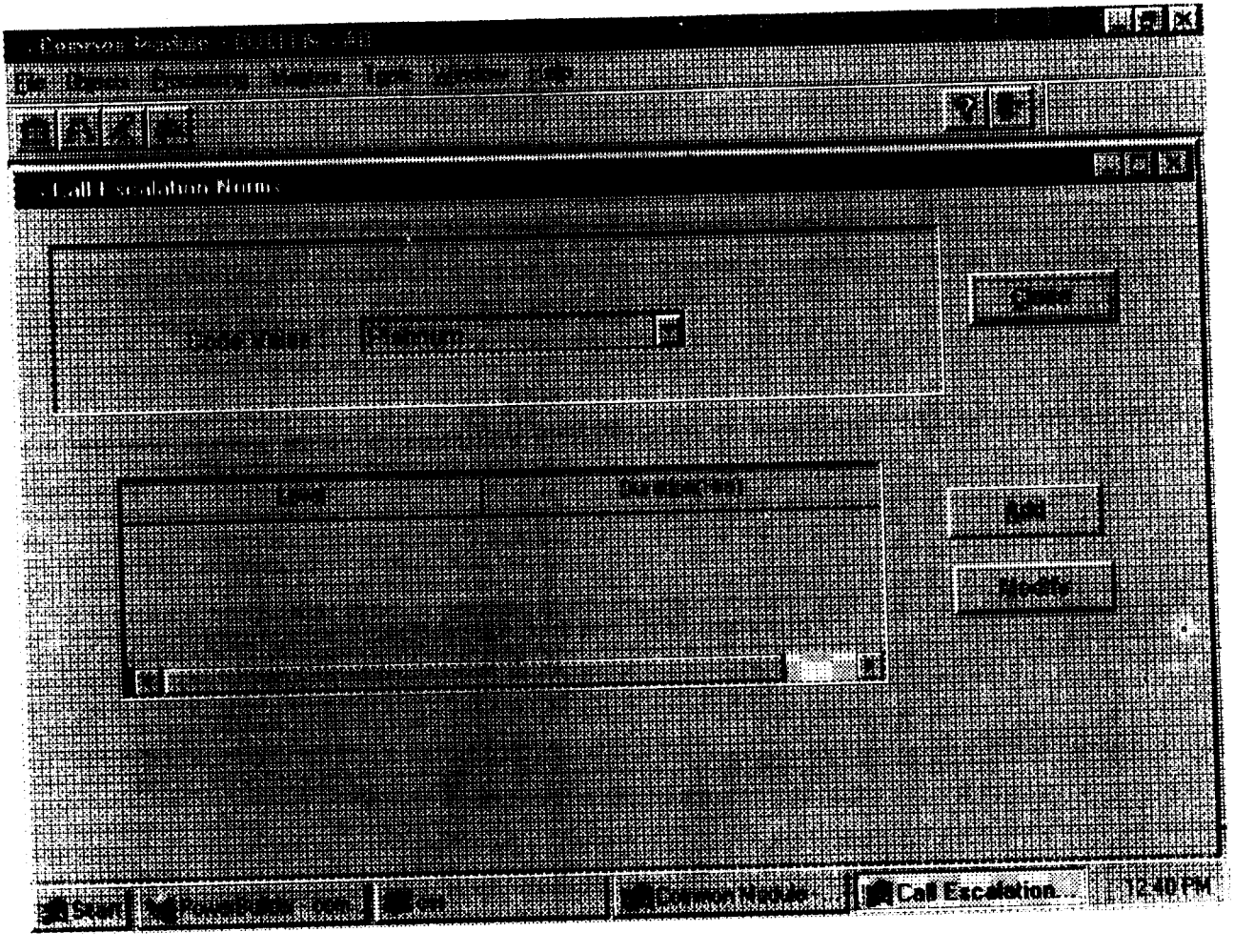
DATA FLOW DIAGRAM

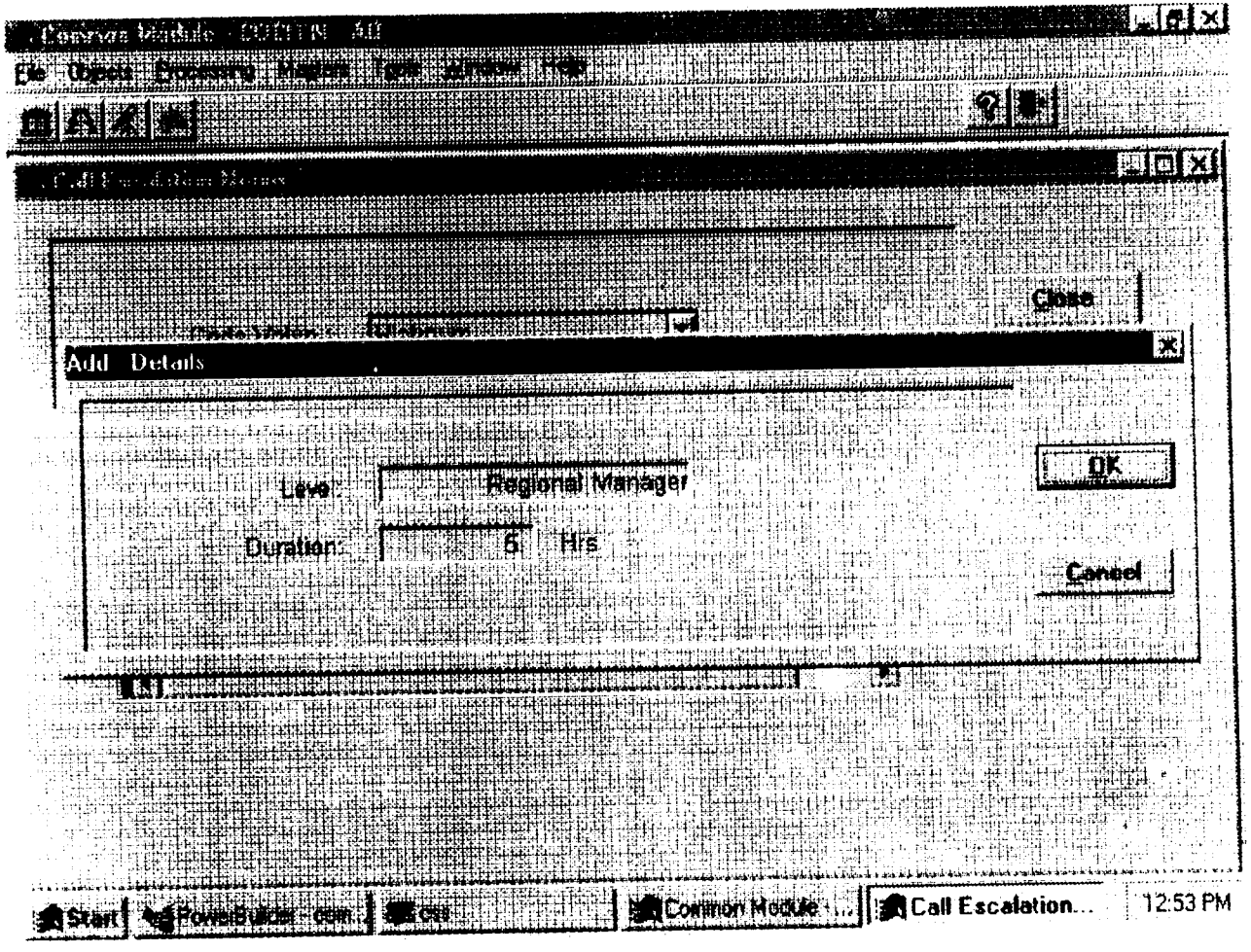


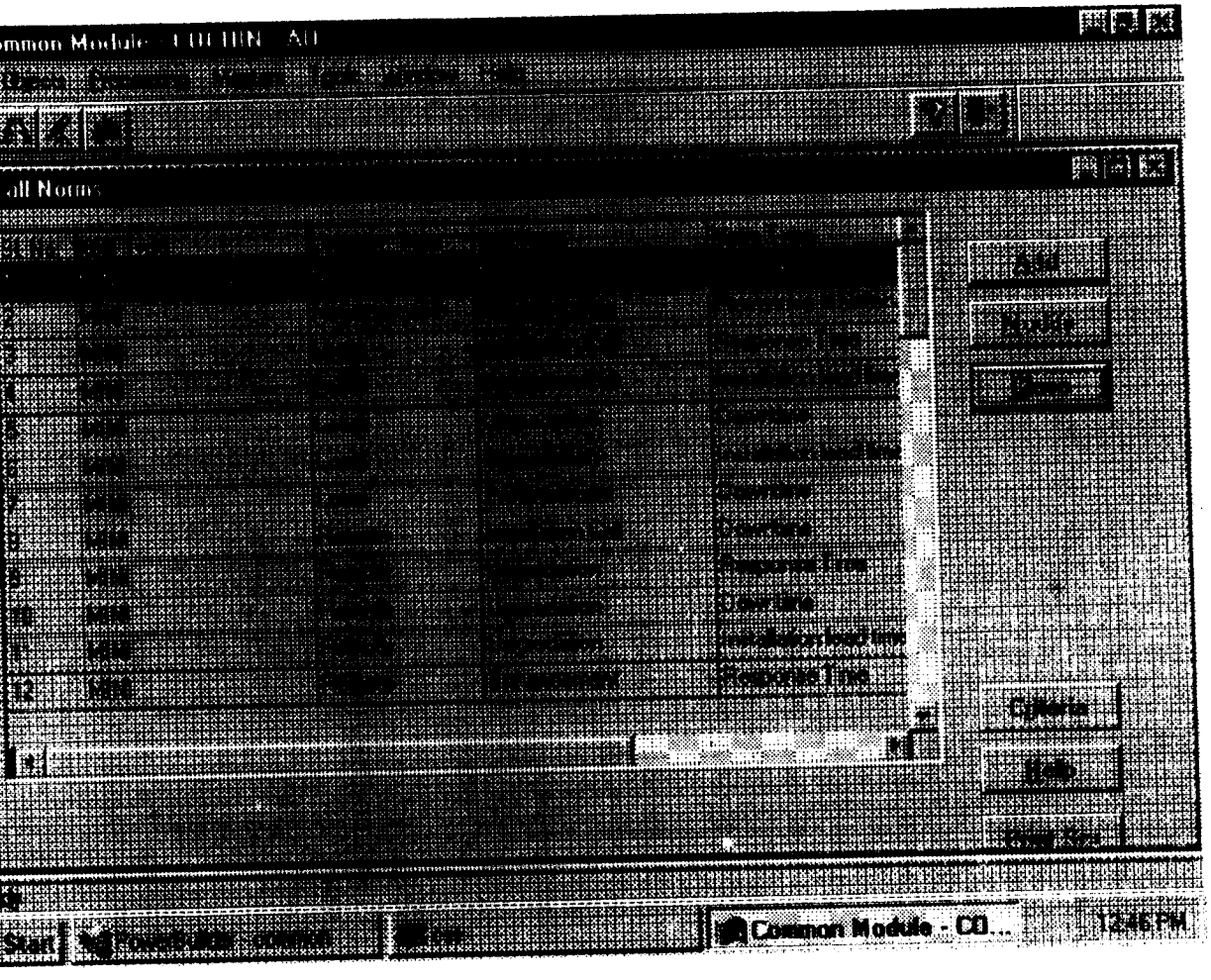
Data Flow Diagram

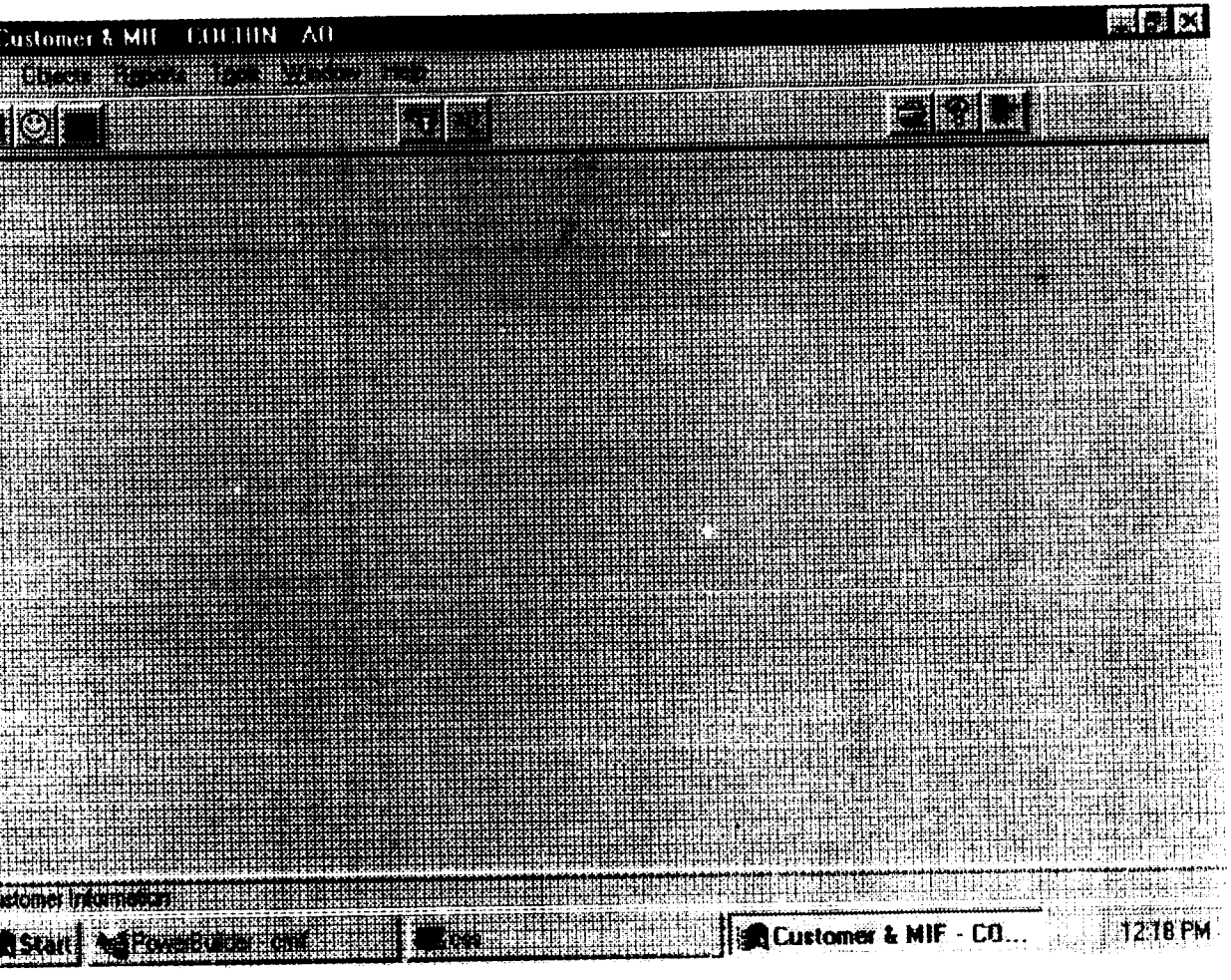


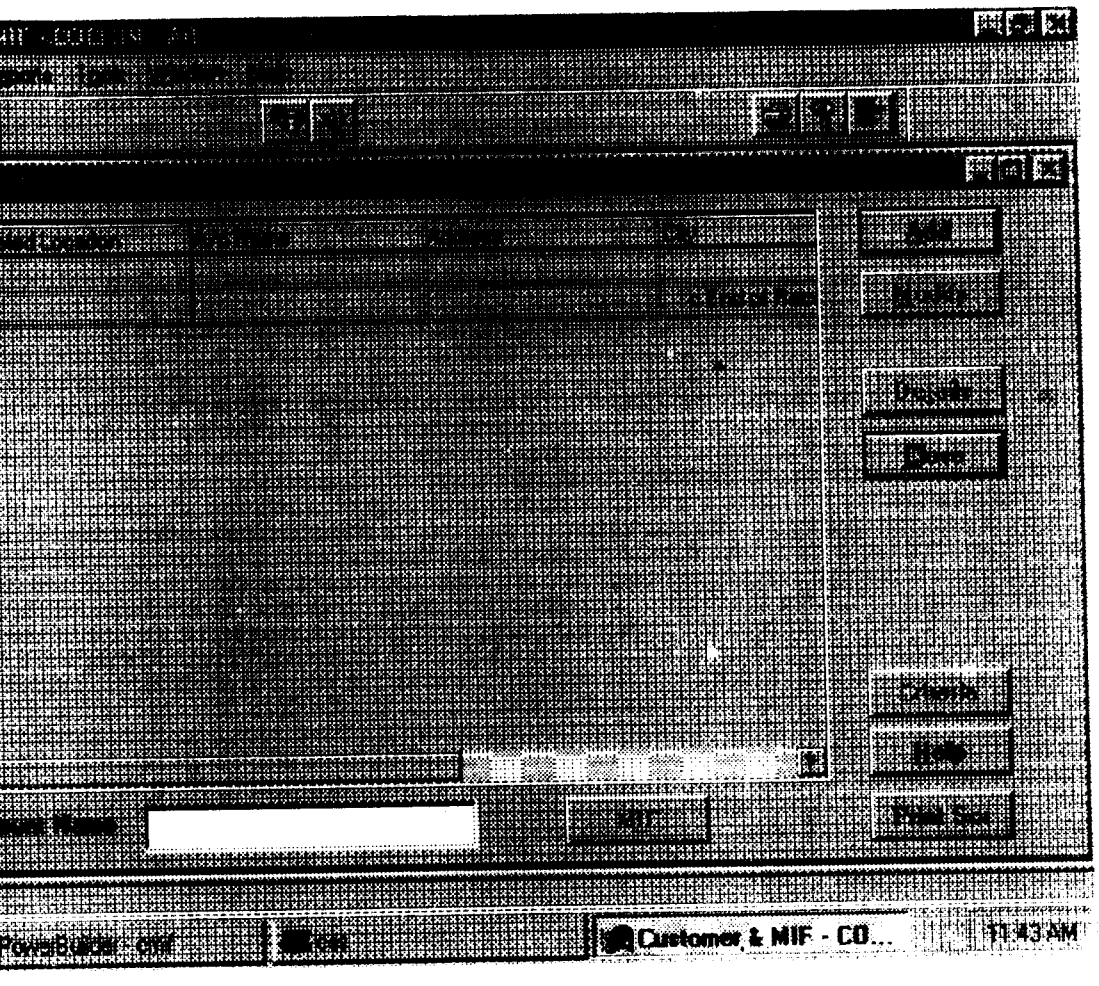
REPORTS

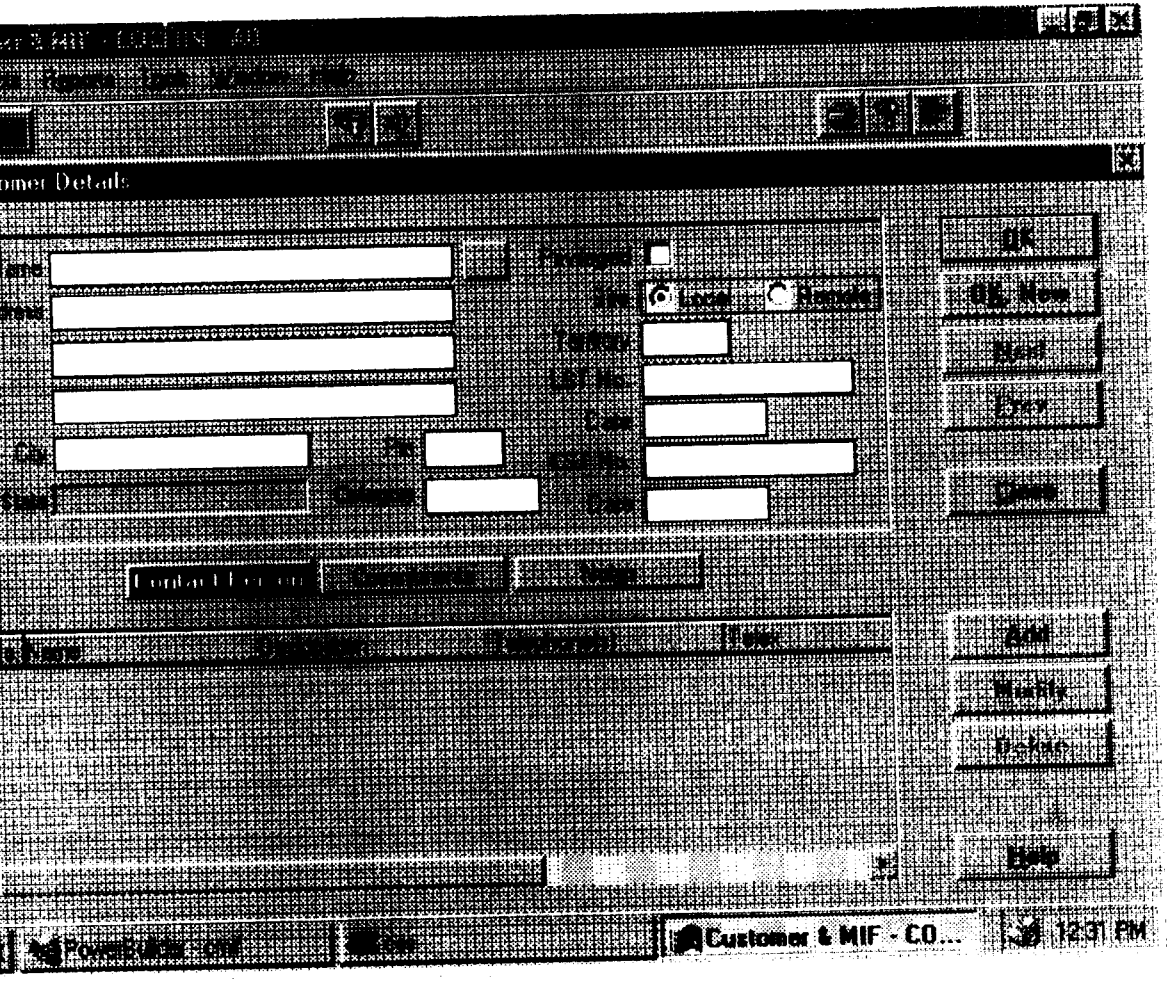



Close
Add Details
Level:
Regional Manager
Duration:
6
Hrs
OK
Cancel
Start
Common Module - ...
Call Escalation...
12:53 PM

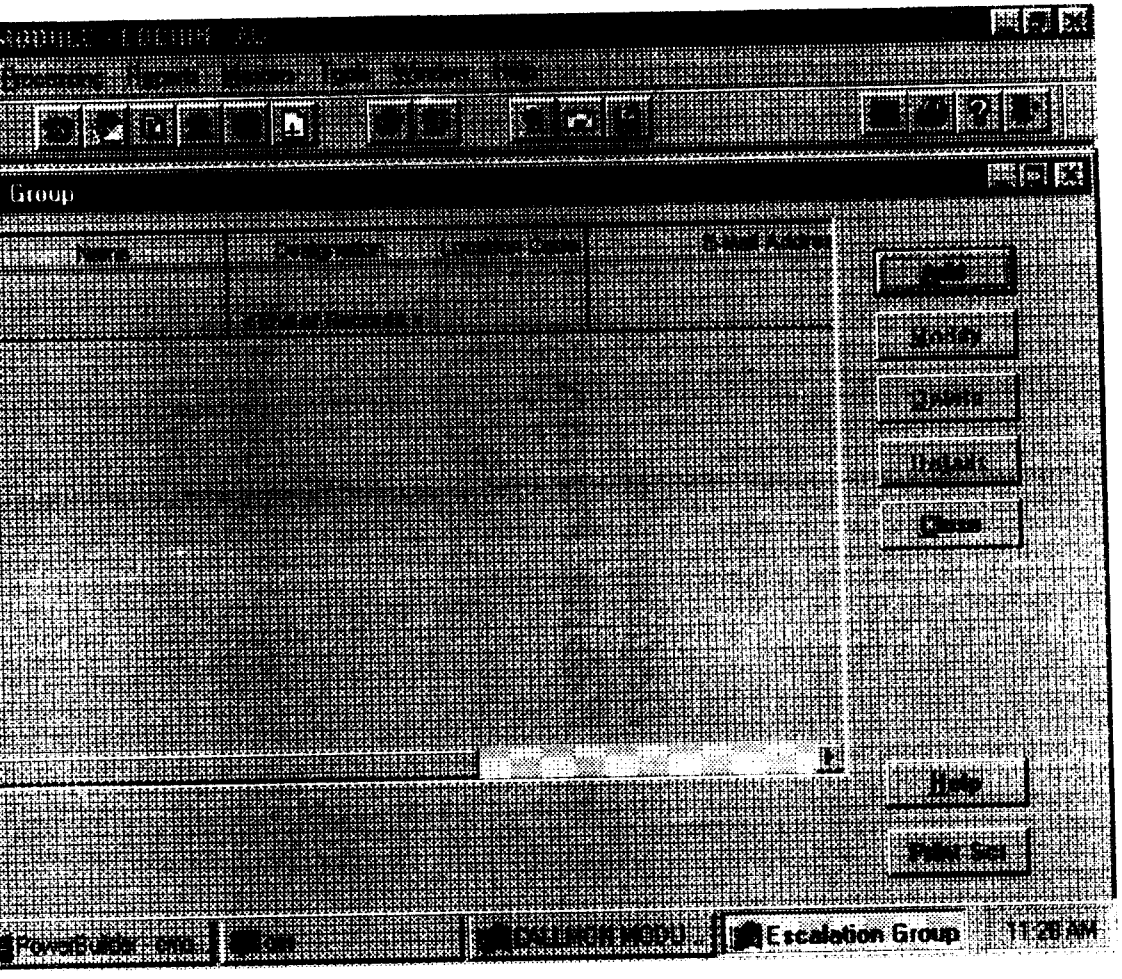

Common Module - CO...
12:46 PM

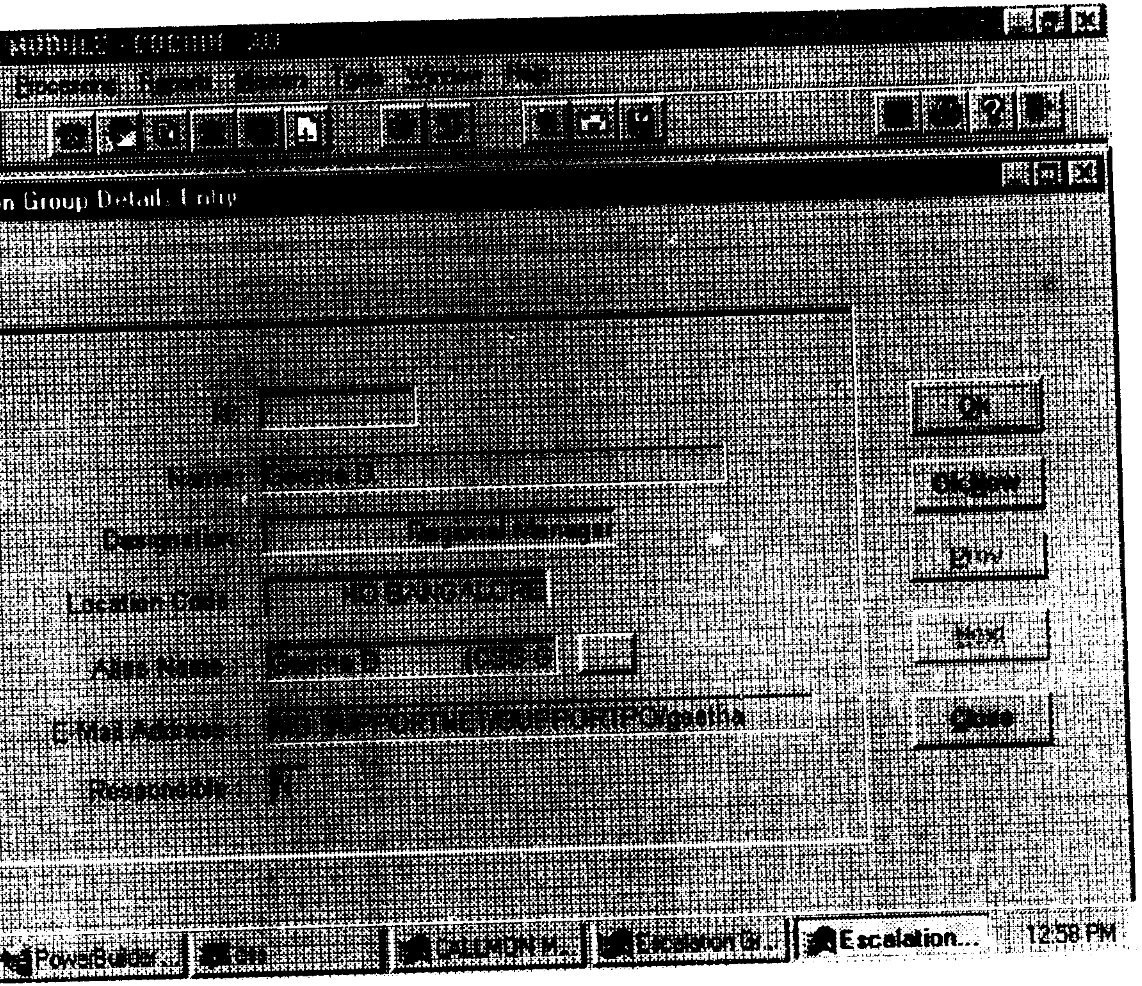

Customer & MIF - CO...
12:16 PM


Customer & MIF - CD...
11:43 AM


omer Details
OK
OK, New
Next
Prev
Close
Add
Modify
Delete
Local
Remote
Customer & MIF - CO...
12:31 PM

CALLMON MODULE
11:26 AM


Group
Escalation Group
11:28 AM



n Group Detail Entry
Name
Designation
Location Code
Alias Name
E-Mail Address
Responsible
Regional Manager
Ok
Close
Escalation...
12:58 PM

Wipro Infotech Limited , XXXXXXX
Customer Support System
Incoming Escalated Calls Report as of XX/XX/XXXX

Account Name	Address	Call No & Date	Call Type	Escalated Type	First Attn Date	Esc To Emp	Mc Sl No	Unit Problem

ion From:

Wipro Infotech Limited , XXXXXXX
Customer Support System
Outgoing Escalated Calls Report as of XX/XX/XXXX

ccount Name	Address	Call No & Date	Call Type	Escalated Type	First Attn Date	Esc To Emp	FE Allc	Mc & Type	Sl No	Unit	Problem	Problem Status

ion To: