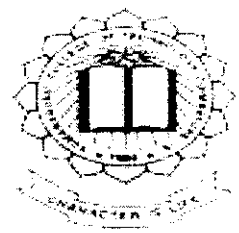


IMPLEMENTATION OF EDI USING SAP AT ROOTS INDUSTRIES

By

D. PRADEEP
Reg. No. 71203621033

Of

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements
For the award of the degree

Of

MASTER OF COMPUTER APPLICATION

June, 2006

BONAFIDE CERTIFICATE

Certified that this project report titled **IMPLEMENTATION OF EDI USING SAP AT DOTS INDUSTRIES** is the bonafide work of Mr. D. Pradeep who carried out the search under my supervision. Certified further, that to the best of my knowledge the ork reported herein dose not form part of any other project report or dissertation on the asis of which a degree or award was conferred on a earlier occasion on this or any her candidate.

Faculty Guide

Head of the Department

ubmitted for the viva-voce examination held on 24.06.06

Internal Examiner

External Examiner

ROOTS

SL. No. : 1146

Date : 16/05/2006

PROJECT / INPLANT TRAINING / INTERNSHIP CERTIFICATE

This is to certify that Mr. / ~~Ms.~~ D. PRADEEP

MCA III year student of KUMARAGURU COLLEGE OF

TECHNOLOGY has done / ~~undergone~~ / ~~a Project~~ / ~~Inplant training~~ / ~~Internship~~ on

" IMPLEMENTATION OF EDI USING SAP

in our ROOTS INDUSTRIES LIMITED during

the period from JAN 2006 to JUNE 2006

During this period his / ~~her~~ conduct was GOOD

(KAVIDASAN)
GENERAL MANAGER - CORPORATE HRD.

ABSTRACT

Implementation of EDI using SAP being developed for data transfer between oots industries and Original Equipment Manufactures such as Ford (North America) at oots industries.

SAP an acronym for System Application and Products for data processing. It tightly integrated, large-scale business application.

EDI is the electronic exchange of business documents between the computer pplications of business partners, using standard format (EDIFACT & ANIS X12) over a ommunication network. EDI is also called paperless exchange.

This implementation carried out to transfer master's data and transaction ata by using BDC programming, ALE technologies and produce report using SAP script.

In the first module used to transferring master data from legacy system to AP database. Then next two modules are carried out by ALE. The Outbound process ends masters and transaction data in EDI format to business partners. The Inbound rocess receives EDI document from a business partner and create SAP document from .

The final module creates graphical view and produce report for the orresponding process.

ACKNOWLEDGEMENT

It is beyond the comprehension of mere elegance of word to acknowledge someone who has been the guiding spirit behind the dissertation. I am very much grateful to former principal Dr K.K. Padmanabhan, Principal Dr Joseph V. Thanikal and Dean Dr. S. Thangasamy for his excellent support.

With profound reference and high regards I record my indebtedness Gratitude to Mr. M. Gururajan, Head of the Department, Mr. A. Muthukumar, Course Coordinator, Mrs. V. Geetha, Project Coordinator, and to my guide Mr. N. Jayakanthan, Lecture, Department of Computer Application, Kumaraguru College of Technology for their guidance throughout my project work.

My heartfelt gratitude to Mr. O A. Balasubramanium, Head - IT and Mr.M.Santhosam, Associate Head- IT, ROOTS Industries Limited, for their valuable guidance and their patience, without which this project would not have been completed. I take this opportunity to thank all the Associates in ROOTS Industries Limited, Coimbatore for their help and cooperation, which has led to the successful completion of this project work.

And I would like to thank my parents, friends, and all those who helped me in this project and whose names are leftover.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

SAP - System Application and Products

EDI - Electronic Data Interchange

ALE - Application Link Enable

BDC - Batch Data Communication

ABAP/4- Advanced Business Application Programming/4 (ABAP/4

CHAPTER 1

INTRODUCTION

This chapter is organized into two parts. The first part deals with the organization profile. It provides a brief insight into the history of the organization and the products. The second part gives an introduction about the project.

1.1 COMPANY PROFILE- ROOTS INDUSTRIES LTD

ROOTS Industries Ltd. is a leading manufacturer of **HORNS** in India and the 11th largest Horn Manufacturing Company in the world.

Headquartered in Coimbatore - India, ROOTS has been a dominant player in the manufacture of Horns and other products like **Castings** and **Industrial Cleaning Machines**.

Since its establishment in 1970, ROOTS has had a vision and commitment to produce and deliver quality products adhering to International Standards.

With a strong innovative base and commitment to Quality, Roots Industries Limited has occupied a key position in both international and domestic market as suppliers to leading OEMs and after market.

Similar to products, Roots has leading edge over competitors on strong quality system base. Now, RIL is the first Indian Company and first horn manufacturing company in the world to get ISO/TS 16949certification based on effective implementation of QS 9000 and VDA 6.1 system requirement earlier.

RIL has entered into technical collaboration with Robert Bosch, SA to further enhance the technical competence. Roots' vision is to become a world-class company manufacturing world-class product, excelling in human relation.

1.2 PROJECT OVERVIEW

1.2.1 Implementation of EDI - Introduction

Implementation of EDI using SAP being developed for data transfer between Roots industries and Original Equipment Manufactures such as Ford (North America) at Roots industries.

SAP an acronym for System Application and Products for data processing. It is tightly integrated, large-scale business application.

EDI is the electronic exchange of business documents between the computer applications of business partners, using standard format (EDIFACT & ANIS X12) over a communication network. EDI is also called paperless exchange.

This implementation carried out to following three transfer master's data and two transaction data by using BDC programming, ALE technologies and produce report using SAP script.

1.2.2 Material Master (MATMAS)

General material data applicable to the entire company is stored at client level. This includes, for example, the material group, base unit of measure, material descriptions, and conversion factors for alternative units of measure.

1.2.3 Customer Master (DEBMAS)

Specifications that control how an account is process. The master record contains data that controls how business transactions are recorded and processed by the system. It also includes all the information about a customer that you need to be able to conduct business with him or her.

1.2.4 Vendor master (CREMAS)

Data in vendor master records controls how transaction data is posted and processed for a vendor. The vendor master record also contains all the data you require to do business with your vendors.

The master record is used not only in Accounting but also in Materials Management. By storing vendor master data centrally and sharing it throughout your organization, you only need to enter it once. You can prevent inconsistencies in master data by maintaining it centrally. If one of your vendors changes their address, you only have to enter this change once and your accounting and purchasing departments will always have up-to-date information.

1.2.5 Delivery Schedule (DELFOR01)

Used to provide for customary and established business practice relative to the transfer of forecasting/material release information between organizations. The planning schedule transaction may be used in various ways or in a combination of ways, such as: (1) a simple forecast; (2) a forecast with the buyer's authorization for the seller to commit to resources, such as labor or material; (3) a forecast that is also used as an order release mechanism, containing such elements as resource authorizations, period-to-date cumulative quantities, and specific ship/delivery patterns for requirements that have been represented in "buckets," such as weekly, monthly, or quarterly.

The order release forecast may also contain all data related to purchase orders, as required, because the order release capability eliminates the need for discrete generation of purchase orders.

1.2.6 Shipping Notification (DESADV)

Used to list the contents of a shipment of goods as well as additional information relating to the shipment, such as order information, product description, physical characteristics, type of packaging, marking, carrier information, and configuration of goods within the transportation equipment. The transaction set enables the sender to describe the contents and configuration of a shipment in various levels of detail and provides an ordered flexibility to convey information.

The sender of this transaction is the organization responsible for detailing and communicating the contents of a shipment or shipments, to more receivers of the transaction set. The receiver of this transaction set can be any organization having an interest in the contents of a shipment or information about the contents of a shipment.

1.2.7 EDI Component and Functionality

Implementation of EDI using SAP being developed for data transfer between Roots industries and Ford . The whole system is modularized into four.

- ✓ Transfer from legacy system
- ✓ Receive from another system (inbound)
- ✓ Transmit to another system (outbound)
- ✓ Graphical view and report creation.

The functionality of each module is discussed in detail in the later part of the report.

CHAPTER 2

REQUIREMENT ANALYSIS

A complete understanding of the requirement is essential for the success of software development. The software scope, initially established by the system engineer and refined during the project planning, is refined in detail. Model of the required data, information and control flow, and operational behavior are created. Alternative solution are analyzed and allocated to various software elements. This chapter presents the problem statement as definitive statement to be solved. The feasibility study evaluates the viability of the project and presents the recommended strategy adopted for the development

2.1 PROBLEM STATEMENT

The main objective of the portal is to using EDI to transfer data from legacy system to SAP database. And then using ALE technology, transfer and receive data between Roots plants and ford industries at North America.

2.2 FEASIBILITY ANALYSIS

Feasibility is he measures of how beneficial or practical the development of information system will be to an organization. Once the problem is explained the feasibility study is to be done to test whether the product is achievable. The feasibility study describes the degree of the usefulness of the product to the organization. The feasibility study can be divided into four phases. They are as follows:

2.2.1 Existing System

➢ Outbound Process

Using Desktop EDI as third party software, enter data manual and pass through business partners.

➢ Inbound Process

Retrieves business document from partners, using M.S Excel as front end tool pass the data to SQL server as a data base.

➢ Report

Using M.S Excel produced report.

Limitation of Existing System

- ➢ Is highly inefficient and Laborious
- ➢ Cannot be tracked easily
- ➢ Include redundant data entry at various points
- ➢ Manual maintenance
- ➢ Cost high

2.2.2 Proposed System

✓ Legacy to SAP Database

Read the master data from legacy system (Excel, Notepad) using field mapping, conversion rules and BDC programming create inbound IDoc. Using batch input process transfer to SAP database.

✓ Outbound Process

The outbound process send document from SAP system to business partners. The outbound process has following process. They are.

- Create logical system to client
- Set up an RFC destination
- Port Definition
- Maintaining the distribution model
- Generate partner profile
- Distributing the model
- Pushing the data

✓ Inbound Process

The inbound process simply reveres of the outbound process. The inbound process receives EDI document from business partner and create SAP document from it. The inbound process has following process. They are.

- EDI transmission is received
- EDI document is convert into IDoc
- The IDoc is transferred to SAP system
- The application document is creation
- Generation application document can be view

Advantage of Proposed System

- ✓ Reduced data entry errors
- ✓ Reduced inventories and better planning
- ✓ Better business process
- ✓ Standard means of communication
- ✓ Availability of data in electronic form
- ✓ Reduced processing cycle time

2.2.3 Technical Feasibility

Technical Feasibility is a measure of practically of a specific technical solution and the availability of technical resource and expertise. This deals with the study of building within the pre-established cost and schedule bounds, the technology that exits to develop all elements of the system, system reliability on proven technologies, the possibility of defining the interfaces, performance and functional aspects, analysis of technical resources, risk associated with the technologies. Feasibility study on quality of the elements of the elements of the system, system's external environment, and system communications is performed.

The tool can be developed with the existing technology, which already install at Root industries and Ford Company. The individual modules are to be developed by SAP. RDBMS Server (Oracle9i) - on IBM-AIX Servers was chosen as the backend engine, because of the huge data volume to be handled. The tool works effectively even for huge amount of data.

2.2.4 Economic Feasibility

Economic Feasibility is a measure of the cost- effectiveness of a project or solution. The System has been designed to work for any type of EDI document and it manages voluminous data. Since the effort to develop the product was found to be feasible, the development presents a good investment for the organization. Hence the above system is economically feasible.

2.3 SOFTWARE REQUIREMENT SPECIFICATION (SRS)

The purpose of this SRS is to document the requirement of the system in a defined format based on the feasibility study. This is an elaboration of requirement analysis and must be validated and approved by the customer with respect to the customer needs. It specifies the required behavior of a system in terms of input data,

required processing, output data, operational scenarios and interfaces. The development environment and system perspective give the view of the system.

2.3.1 Purpose of the project

In today's global business environment, it is crucial for organization to manage the document related to the various technologies. Implementation of EDI provides a complete stored data from legacy system to SAP Oracle database. Transfer and receive EDI data from one R/3 system to another R/3 system using ALE outbound and inbound process respectively.

2.3.2 Scope of the project

The scope of the project is to convert requirement specified by the user into functional requirement and implement the same in the system. It involve the following stage Requirement Analysis, Functional Specification, Design, Coding, and Testing.

2.3.3 Hardware Requirements

Intel P-IV above 1.4 GHz.

512 MB RAM.

17" or 15" VGA Monitor.

114 IBM Keyboard.

IBM two button scroll mouse.

80GB HDD.

52X CD-ROM.

1.44 FDD.

2.3.4 Software Requirements

SAP R/3 IDES Version 4.6C

SAP GUI Presentation Server.

Microsoft office-on WIN 2K e-business Clients.

SAP Application Servers (Training, Developing, Testing).

Sap Database Server.

RDBMS Server (Oracle9i) - on IBM-AIX Servers.

2.3.5 Functional Requirements

Introduction

This part deals with the necessary requirements for the software to make all the functions highly performable.

List of inputs

- Selection screen for data to be transfer
- Transaction code for data transfer
- Giving business partner number.

Information Processing

- Create application document,
- IDoc is generated,
- IDoc converted to EDI standard
- EDI is transmitted to business

List of outputs

- Display status
- Graphical view overall state

2.4 SOFTWARE OVERVIEW

2.4.1 SAP R/3

SAP was founded in 1972 in Walldorf, Germany. It stands for Systems Application and Products in Data processing. It is the name of both online financial and administrative software and the company that developed it. SAP is the world premier provider of client server business solutions for which it is so well known today. The main purpose of using standard business application software like SAP is to reduce the amount of time and money spent on developing and testing all the programs. The main advantage of using SAP as the company's ERP system is that SAP has high level of integration among its individual applications which guarantee consistency of data throughout the system and company itself.

SAP is made up of individual, integrated software modules that must be customized to an organization's business processes. The modules are Finance/Accounting (FI), Controlling (CO), Project Systems (PS), Funds Management (FM), Materials Management (MM), Sales Distribution (SD), Office and Communications (OC), Human Resources (HR), Fixed Assets Management (AM), Production Planning (PP), Quality Management (QM) and Warehouse Management(WM). These individual SAP modules perform various system tasks and function as a team, compiling and integrating information across departments and automating information management. As SAP is fully integrated, we can think more globally about information transfer.

MODULES IN SAP

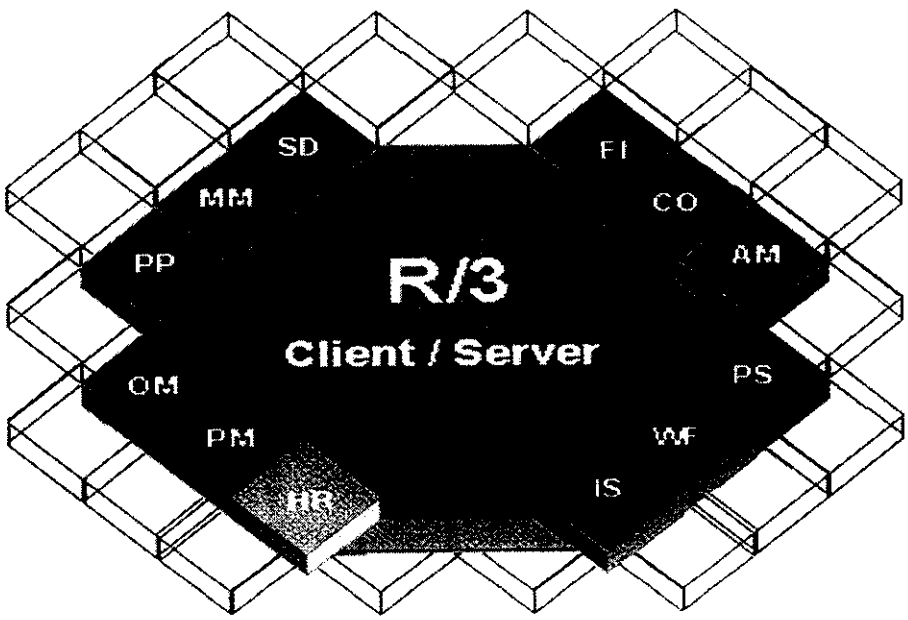


Figure c 2.1 Modules in sap

SAP R/3 Basis System

SAP R/3 is the third generation set of highly integrates software modules that perform common business functions based on n-n leading practice. It takes care of any enterprise however diverse it may be in operations, spread over the world geographically. R/3 is an integrated system that provides us one application for all the processes.

The primary reason to use R/3 is that there are many administrative systems performing different and, in some cases, overlapping functions. Often these different systems cannot share data, requiring us to do more work. Secondly, we need a standardized, efficient system that will give us better information in "real time," better supports new programs and the growth and development of the Health System. Thirdly, present systems have not remained current with new technologies.

Architecture of R/3 basis System

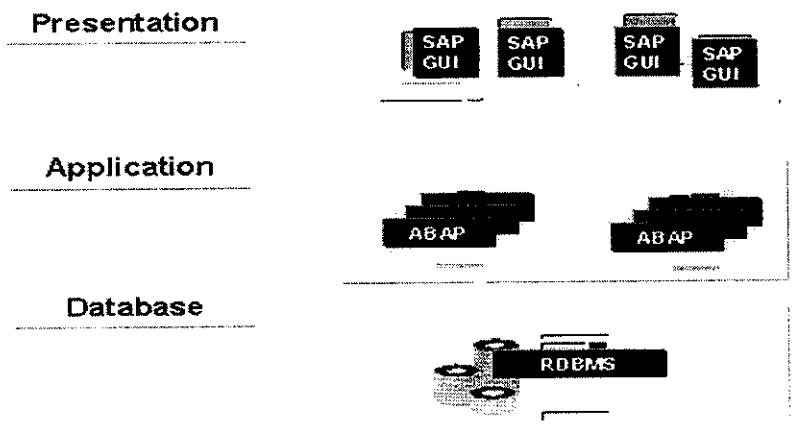


Figure c 2.2 Architecture of R/3 basis System

The R/3 Basis system is a three-tier client/server system. The classic configuration of an R/3 System contains the following software layers:

- **Database Layer**

This layer consists of a central database system containing all the data in the R/3 System. The database system has two components-the database management system (DBMS), and the database itself. SAP does not manufacture its own database. In stead, the R/3 System supports the database systems from other suppliers. The database contains the master data, transaction data from business applications and all data for the entire R/3 System.

- **Application Layer**

The application layer consists of one or more application servers and a message server. Each application server contains a set of services used to run the R/3 system. The message server is responsible for communication between the application servers. It passes requests from one application server to another within the system. It also contains information about application server groups and the current load balancing within them.

- **Presentation Layer**

The presentation layer contains the software components that make up the SAPgui (graphical user interface). This layer is the interface between the R/3 system and its users. The R/3 system uses the SAPgui to provide an intuitive graphical user interface for entering and displaying data. The presentation layer sends the user's input to the application server, and receives data for display from it. While a SAPgui component is running, it remains linked to a user's terminal session in the System.

Characteristics of SAP R/3

- Rich in functionality & tight integration-across functions.
- International orientation.
- Highly configurable.
- Highly secure data handling.
- Seamless handling of db.
- Minimize data redundancy, max data consistency.
- Capitalize on economics of scale, e.g. purchasing.
- Business practices world wide are incorporated.

2.4.2 ABAP/4

Advanced Business Application Programming/4 (ABAP/4) is the fourth generation programming developed by SAP to develop application programs. ABAP/4 is the backbone SAP R/3.

Interfaces from non-SAP systems can be managed through ABAP/4.

The ABAP Workbench component is a fully-fledged development environment for applications in the ABAP language. With it, you can create, edit, test, and organize application developments. It is fully integrated in the R/3 Basis system and, like other R/3 applications, is itself written in ABAP. The actual processing logic of an application

Program is written in ABAP-SAP's own programming language. The ABAP processor executes the processing logic of the application program, and communicates with the database interface.

All R/3 application programs, along with parts of the R/3 Basis system, are written in the ABAP Workbench using ABAP, SAP's programming language. The individual components of application programs are stored in a special section of the database called the R/3 Repository. The R/3 Repository serves as a central store for all of the development objects in the R/3 System. The programs are run within the application layer of the R/3 System. ABAP programs communicate with the database management system of the central relational database (RDBMS), and with the graphical user interface (SAPgui) at presentation level.

2.4.3. ORACLE 9i

A database server is the key to solving the problem of information management.

Databases provide an electronic location to store information in a specific manner, which can be accessed to make business decisions for your customers and business. Oracle 9i is used as the database server. The Oracle RDBMS is an enormous environment with unlimited potential. Oracle9i helps to develop, create and modify database objects, access information on remote databases, and deploy database applications into any Oracle environment. It reliably manages a larger amount of data in a multi-user environment so that many users can concurrently access the same data, delivering high performance. It also prevents unauthorized access and provides efficient solutions for failure recovery. Today, the Oracle tools are among the best and most powerful development tools available.

2.4.4 ALE (Application Link Enable)

ALE-enable processes and master data object have to be configured to implement a distributed system. SAP doesn't know how you want to distribute your process and data, how many systems are involved, and what data you want to exchange between systems. The first task in configuring any ALE scenario is to put the basic ALE infrastructure in place, which is like building a bridge .you must connect the end before traffic can move over it. The bridge is built once and remains the same, regardless of the type of traffic. You are simply providing the basic infrastructure for two system to recognize each other and communicate with each other, including logical system name, an RFC destination, and port definition.

CHAPTER 3

SYSTEM DESGIN AND DEVELOPMENT

This is the pivotal point in the system development life cycle. User requirement have been identified. Information has been gathered to verify the problem and evaluate the existing system. A feasibility study has been conducted to review alternative solutions and provide cost/benefit justification. The culmination of the study is a proposal summarizing the finding and recommending a candidate system for the user. It translates the system requirements into an operational system.

Since the reason behind the candidate system made sense, the system is now ready to undergo the design phase. The design as Award puts, is "a solution, a 'how to' approach, compares to analysis, a 'what is' orientation".

System design is the process of planning a new system or to replace or the complement of the existing system. The design is based on limitations of the exiting system and the requirement specification that are gathered in phase of system analysis.

System design primarily involves a logical design and then the physical construction of the system. During the logical design of the system, the detailed specifications for the new system have been provided by describing its features, procedures that meets the system requirements.

The requirement of the data was identified after the identification of the objectives. After identifying the objectives, performance the code design of the input documents, output formats and the frequency of report are finalized.

3.1 DESIGN CONCEPTS

Several concepts such as abstraction, refinement, modularity, software architecture, control hierarchy, structural partitioning, data structures, software produced, information hiding are kept in mind during the design phase. Each concept provides the designer with the formulation from which more sophisticated design method can be applier.

- Modularity (both in program and data) and the concept of abstraction enable the designer to simplify and reuse the software components.
- Refinement provides a mechanism for representing successive layers of functional detail.
- Programs and data structure contributes to an overall view of software architecture, while procedure provides the details necessary for algorithm implementation.
- Information hiding and functional independence provide heuristics for achieving effective modularity.

Design Principles

Design process is a set of iterative steps, which enable the designer to describe all aspects of the software to build. The main principles in software design are

- The design process should not suffer from 'tunnel vision'.
- The design should be traceable to the analysis model.
- The design should not reinvent the wheel.
- The design should exhibit uniformity and integration.
- The design should be reviewed to minimize the conceptual errors.
- The design should minimize the intellectual distance between the software and the problem, as it exits in the real world.

Architecture design

Architectural design involves identification of software components, decoupling and decomposing them into processing modules, conceptual data structure and specifying relationship among the components.

The Implementation of EDI system has been developed as a three- tier architecture application. The 3 tiers or layers are.

1) View Layer

This layer contains the Dialog programmings, Smart Form are used for designing the GUI for view Master data.

2) Middle layer

This layer is responsible for connecting to the database and processing the data in the database. The implementation of EDI system uses ABAP programming and ALE to process the requests.

3) Data layer

This layer contains the database which stores the information required for the application. In the Implementation of EDI system Sap Database Server, RDBMS Server (Oracle9i) - on IBM-AIX Servers

3.2 MODULARIZATION

Modularity is the single attribute of software of software that allows a program to be intellectually manageable. A reader cannot easily grasp monolithic software. The five criteria that enable us to evaluate a design method with respect to its ability to define effective modular design are modular decomposability which provides a systematic mechanism for decomposing, Modular compos ability which

enables exiting design components to be assembled into a new system, Modular understandability which enables us to understand as a single unit, Modular continuity which forces the developers to make change to individual modules rather than to entire system, and modular protection, which does not allow the aberrant conditions occurring in the module to affect the entire system. All these factors are considered in decomposing this system into modules that are integrated to satisfy the problem requirements.

3.2.1 TRANSFER FROM LEGACY TO SAP

This module used to read the master data from legacy system (Excel, Notepad) using field mapping, conversion rules and BDC programming create inbound IDoc. Using batch input process transfer to SAP database.

3.2.2 TRANSFER DATA FROM SAP TO SAP (Out Bound)

The outbound process send document from SAP system to business partners. The outbound process consists of following process. They are Create logical system to client, Set up an RFC destination, Port Definition, Maintaining the distribution model, Generate partner profile, distributing the model, pushing the data

3.2.3 RECEIVE DATA FROM SAP TO SAP (In Bound)

The inbound process simply reveres of the outbound process. The inbound process receives EDI document from business partner and create SAP document from it. The inbound process consists of following process. They are EDI transmission is received, EDI document is convert into IDoc, The IDoc is transferred to SAP system, the application document is creation, and Generation application document can be view

3.2.4 Graphical view and Report

In this module used to produce classical report and interaction report by using SAP script. By help of using Smart Form able to generate graphical representation the status of the process.

3.3 DATAFLOW DIAGRAMS (DFD)

Data flow diagram is a graphical representation that depicts information flow and the transform that are applied as data from input to output. The DFD can be used to represent a system or software at any level of abstraction.

3.3.1 DFD Level 0 – Context Flow Diagram

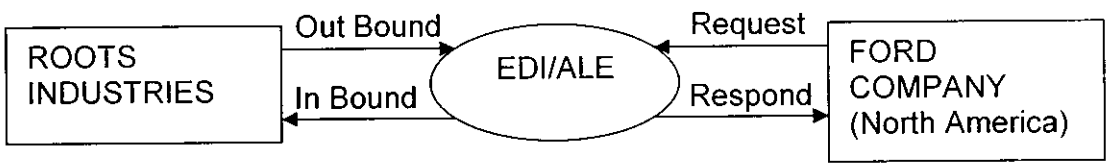
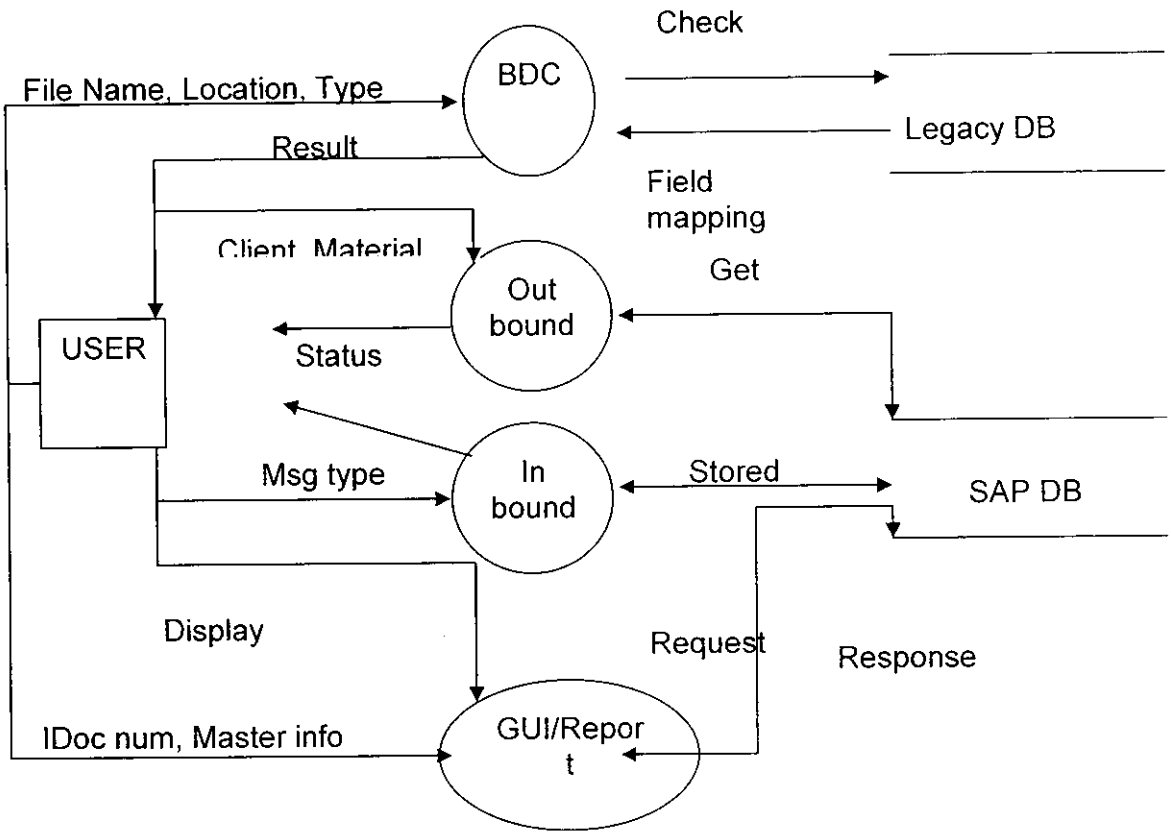
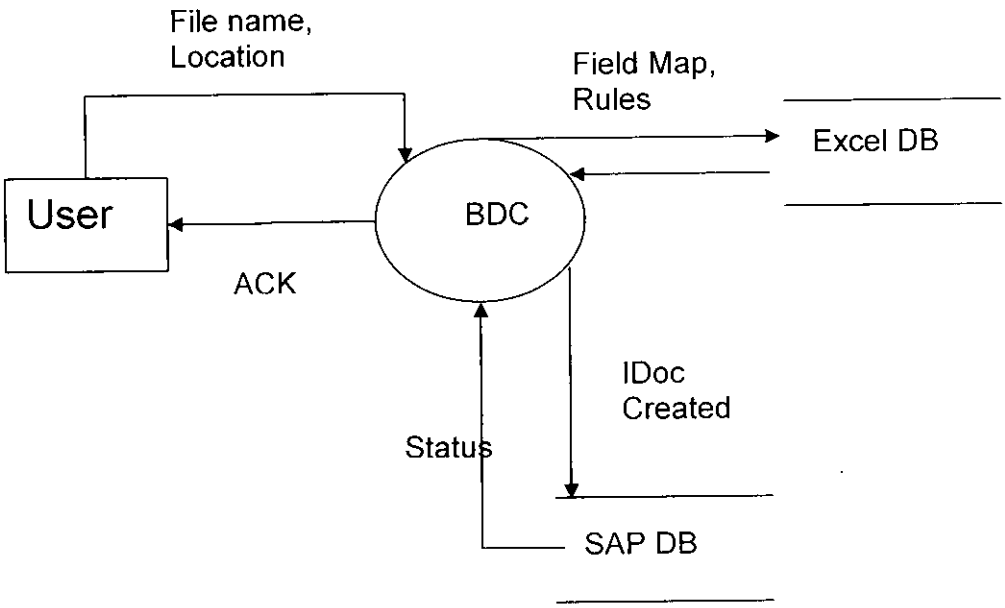


Figure c 3.1 DFD Level 0 – Context Flow Diagram

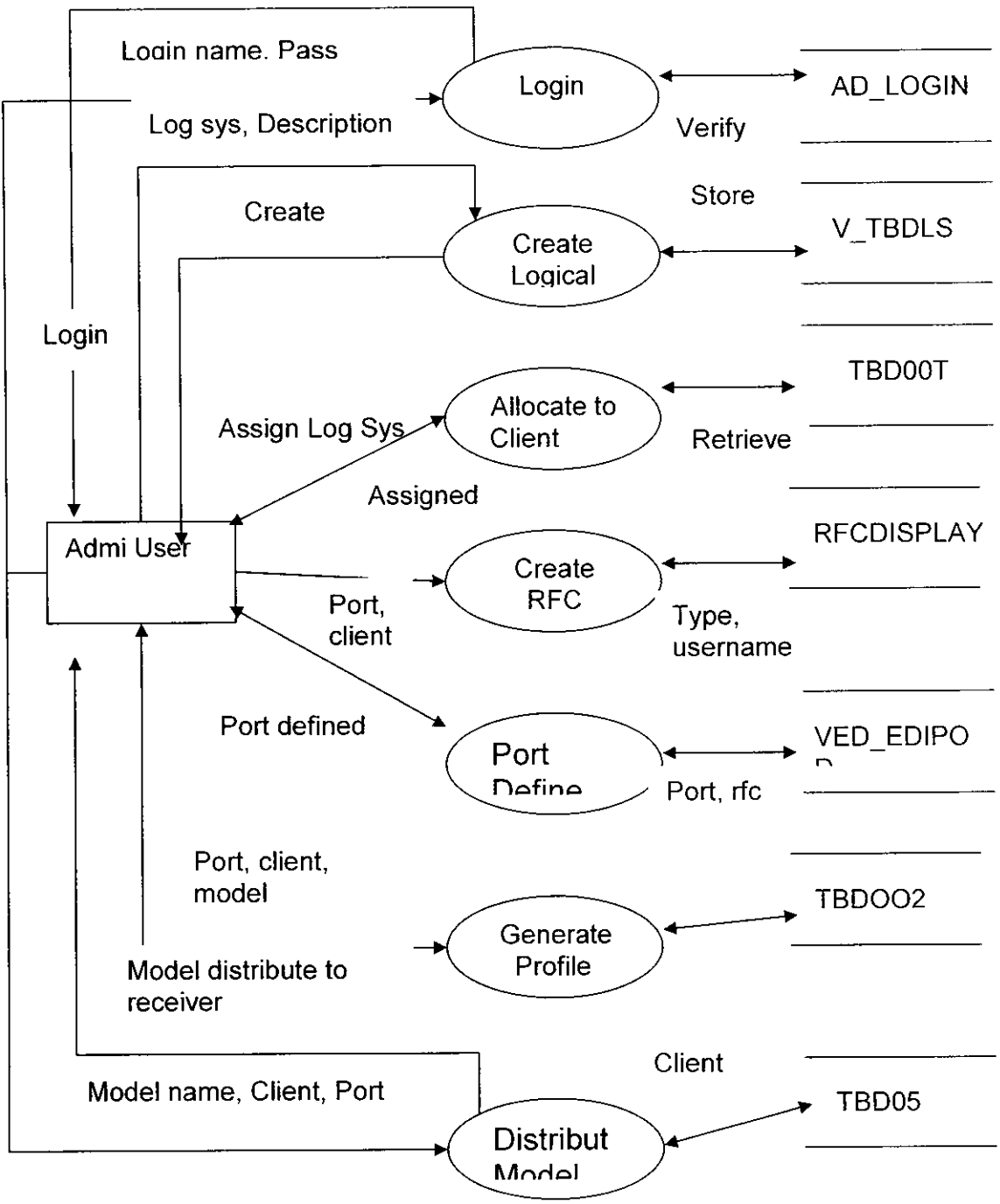
3.3.2 DFD Level 1 – Implementation Of EDI



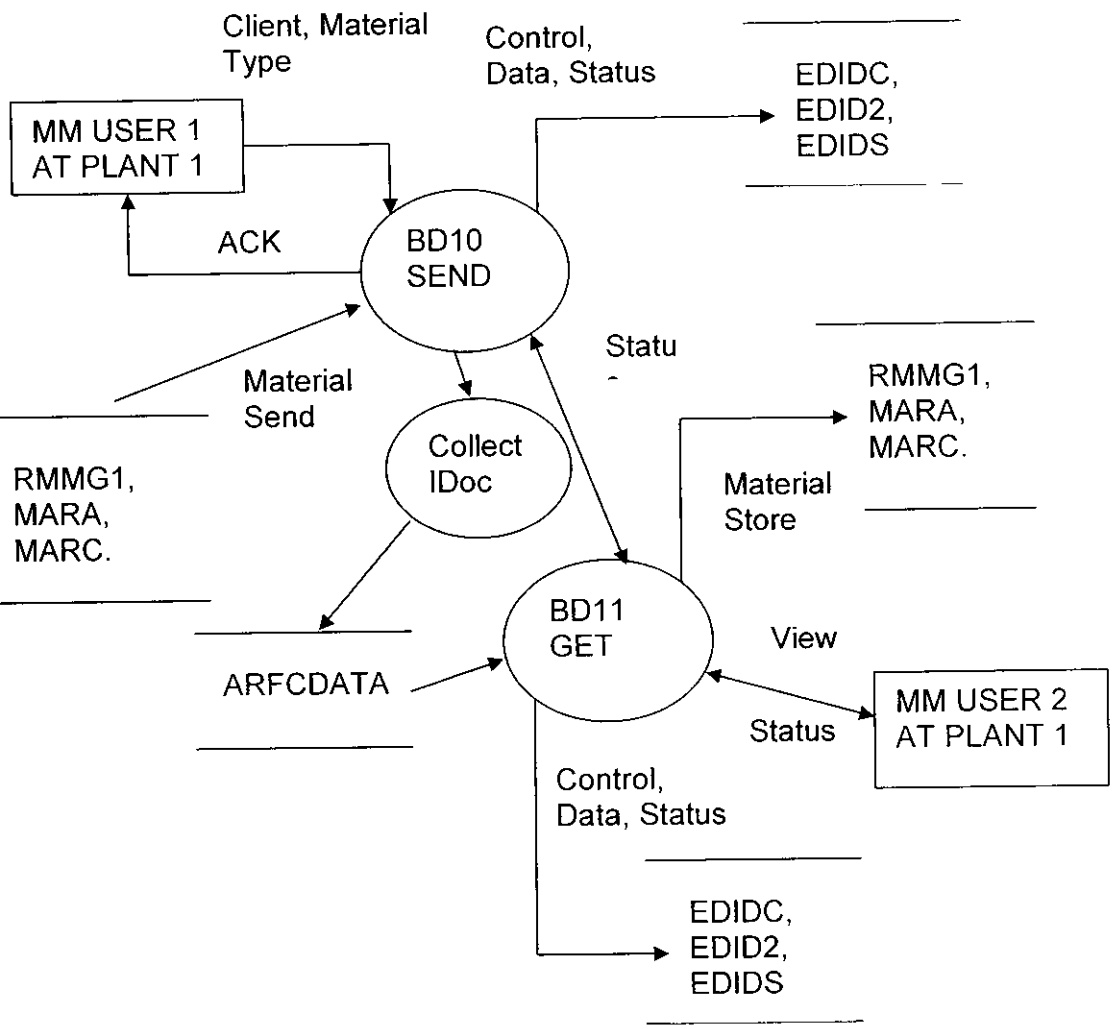
3.3.2 DFD of Transfer from Legacy data to SAP Data Base



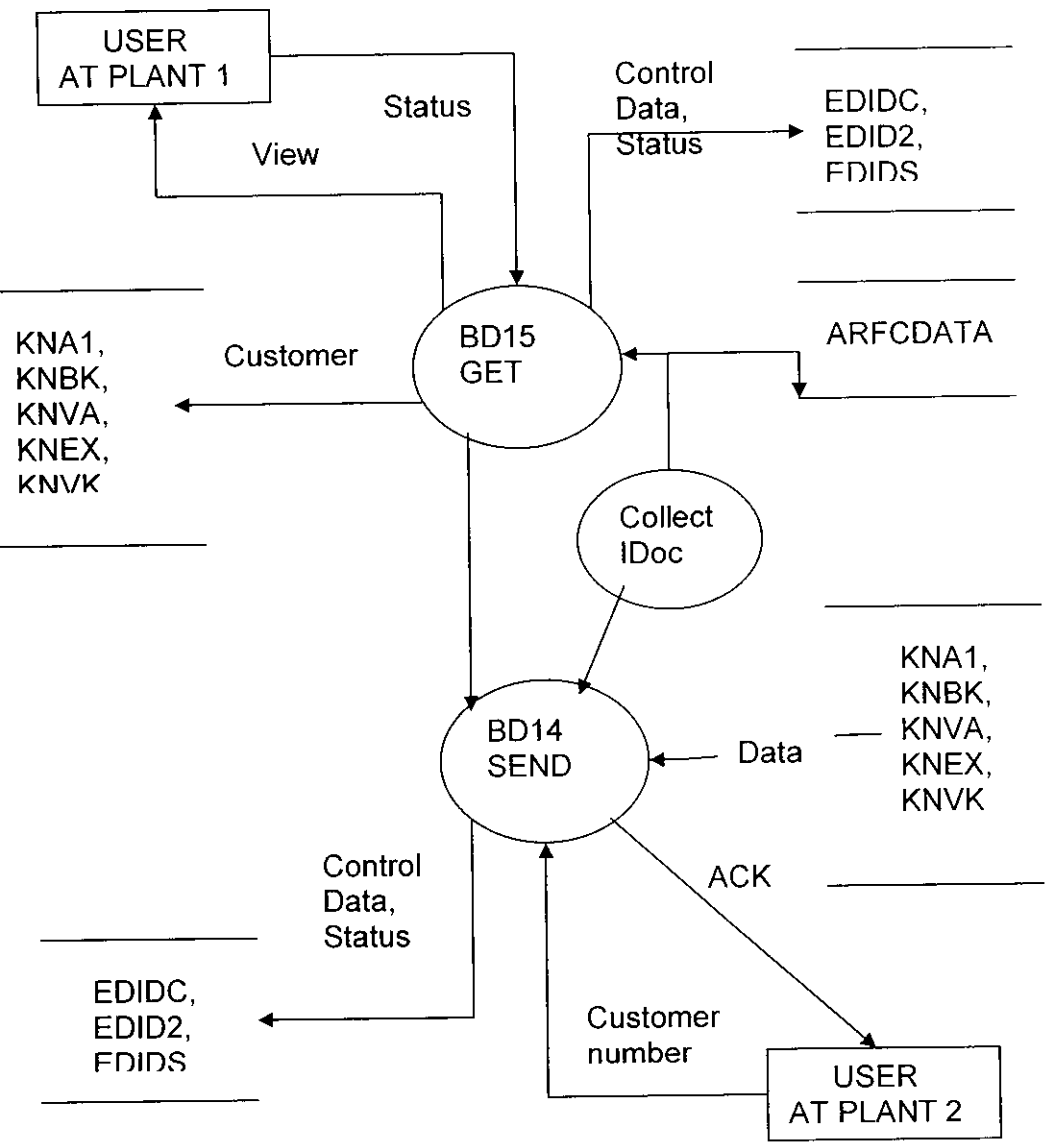
3.3 DFD- Configure of ALE for INBOUND and OUTBOUND



3.4 DFD Material Master Transfer



3.4 DFD Customer Master Error!



3.4 DATABASE DESIGN

The general theme behind a database is to handle information as an integrated whole. A database is a collection of inter-related data store with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information necessary, quick, inexpensive and flexible for the user. The organization of data in a database aims to achieve 3 main objectives:

- Data Integration
- Data independence
- Data integrity

Normalization: Data structure is refined through a process called normalization. Data are grouped in the simplest way possible so that change can be made with a minimum of impact on the data structure. When too many attributes are grouped together to form entities, some attributes are found to be entities themselves. Further normalization of these entities into attributes linked by common data elements to form relationships improves the effectiveness of the DBMS. The Implementation of EDI uses Oracle 9i sever as database to store and handle information related to the proposed system.

Table Schemas for EDI

* Means Primary Key

Table: 3.1 KNA1- General Data in customer master.

Field	Type	Description
Mandt *	Clnt(3)	Client
Kunnr *	Char (10)	Customer number
Land1	Char (3)	Country key
Name1	Char (35)	Name1
Name2	Char (35)	Name2
Ort01	Char (35)	City
Pstlz	Char (10)	Postal
Regio	Char (3)	Region
Sortl	Char (10)	Sort field
Stras	Char (35)	House number
Adrnr	Char (10)	Address
MCOD1	Char (35)	Search term for match code search

Table: 3.2 KNBK Customer Master (Bank Details)

Field	Type	Description
Mandt *	Clnt (3)	Client
Kunnr *	Char (10)	Customer number
Banks *	Char (3)	Bank country key
Bankl *	Char (15)	Bank key
Bankm *	Char (15)	Bank account number
Bvtyp	Char (4)	Partner bank key
Xezer	Char (1)	Indicator
Koinh	Char (60)	Account holder name

Table: 3.3 KNVA Customer Master Loading Points

Field	Type	Description
Mandt *	Clnt (3)	Client
Kunnr *	Char (10)	Customer number
Ablad *	Char (25)	Uploading point
Lfdnr	Numc (2)	Sequential number
Knfak	Char (2)	Customer's factory calendar
Defab	Char (1)	Default unloading point
Wand	Char (3)	Goods receiving hour
Moab1	Tims(6)	Goods receipt times

able: 3.4 KNEX Customer Master: Legal Control - Sanctioned Party List

eld	Type	Description
andt *	Clnt (3)	Client
unnr *	Char (10)	Customer number
doco	Char (1)	ID: TDO boycott list for export control
ndex *	Char (3)	Country key for export control in customer master
doda	Dats (8)	Date of last check of tdo list for export control
dnco	Char (8)	Id: sdn boycott list for export control
dnda	Dats (8)	Date of last check of sdn list for export control
hrco	Char (8)	D: customer boycott list for export control
hrda	Dats (8)	Date of last check in inter. Boycott list for exp. Control
ldck	Dats (8)	Validity start date of entry in sanctioned party list
ldst	Dats (8)	Validity end date of entry in sanctioned party list

ables:3.5 KNVK Customer Master Contact Partner

ield	Type	Description
Mandt *	Clnt (3)	Client
Parnr	Nums (3)	Number of contact person
Kunnr *	Char (10)	Customer number
Adrnd	Char (10)	Business address
Adrnp	Char (15)	Home address
Abtpa	Char (20)	Contact person's department at customer
Pafkt	Char (1)	Contact person function
Parvo	Char (5)	Partner's power of attorney
Parla	Char (10)	Partner language

able: 3.6 RMMG1 Mat. Master Maintenance

ield	Type	Description
ATNR *	Char (18)	Material number
ERKS	Char (4)	Plant
GORT	Char (4)	Storage location
UKRS	Char (4)	Company Code
WKEY	Char (4)	Valuation area
WTAR	Char (10)	Valuation type
KORG	Char (4)	Sales organization
PRAS	Lang (1)	Language key
UKRS_VKO	Char (4)	Company Code
IBRSH	Char (1)	Industry sector
ITART	Char (4)	Material type
ISPR	Char (4)	Material: MRP profile
UNNR	Char (10)	Customer number

able:3.7 MARA General Material Data

ield	Type	Description
MATNR *	Char (18)	Material number
MANDT *	Clnt (3)	Client
ERSDA	Dats (8)	Creation date
ERNAM	Char (10)	Name of Person who Created the Object
AEDA	Dats (8)	Date of last change
AENAM	Char (10)	Name of person who changed object
BISMT	Char (18)	Old material number
MEINS	Unit (3)	Base unit of measure
BSTME	Unit (3)	Order unit
ZEINR	Char (22)	Document number
ZEIAR	Char (3)	Document type
ZEIVR	Char (3)	Document version
WRKST	Char (48)	Basic Material
NORMT	Char (18)	Industry Standard Description
LABOR	Char (3)	Laboratory/design office
EKWSL	Char (4)	Purchasing value key
BRGEW	Quan (13)	Gross weight
NTGEW	Quan (13)	Net weight
GEWEI	Unit (3)	Weight Unit
VOLUM	Quan (13)	Volume

able:3.8 MARC Plant Data for Material

eld	Type	Description
ANDT *	Clnt (3)	Client
ATNR *	Char (18)	Material number
ERKS *	Char (4)	Plant
STAT	Char (15)	Maintenance status
VORM	Char (1)	Flag Material for Deletion at Plant Level
WTTY	Char (1)	Valuation category
CHAR	Char (1)	Batch management indicator (internal)
MSTA	Char (2)	Plant-Specific Material Status
ESKZ	Char (2)	Procurement Type
OBSL	Char (2)	Special procurement type
INBE	Quan (13)	Reorder point
ISBE	Quan (13)	Safety stock

able: 3.9 LFA1 Vendor Master

ield	Type	Description
MANDT *	Clnt (3)	Client
IFNR	Char (10	Account number of vendor or creditor
AND1	Char (3)	Country key
NAME1	Char (35)	Name 1
NAME2	Char (35)	Name 2
ORT01	Char (35)	City
ORT02	Char (35)	District
PFACH	Char (10)	P.O. Box
REGIO	Char (3)	Region (State, Province)
SORTL	Char (10)	Sort field
STRAS	Char (35)	House number and street
ADRNR	Char (10)	Address
MCOD1	Char (10)	Search term for match code search

able:3.10 LFB1 Vendor Master- company code

eld	Type	Description
ANDT *	Clnt (3)	Client
FNR *	Char (10)	Account number of vendor or creditor
JKRS *	Char (4)	Company Code
ERNR	Char (8)	Personnel Number
RDAT	Char (8)	Date on which the Record Was Created
RNAM	Char (12)	Name of Person who Created the Object
PERR	Char (1)	Posting block for company code
OEVM	Char (1)	Deletion Flag for Master Record
UAWA	Char (3)	Key for sorting according to assignment numbers
KONT	Char (10)	Reconciliation Account in General Ledger
EGRU	Char (4)	Authorization group
ZSKZ	Char (2)	Interest calculation indicator

ables: 3.11 V_TBDLS Logical Systems

ield	Type	Description
OGSYS*	Char (10)	Logical system
TEXT	Char (40)	Text (40 characters)

ables: 3.12 TBD00T View of Distribution Model

ield	Type	Description
IANDT *	Clnt (3)	Client
ANGU	Lang(1)	Language Key
USTMODEL	Char(30)	View of ALE model
IODELTEXT	Char(70)	Short description of view of distribution model

ables:3.13 RFCDISPLAY Display structure for RFCDES maintenance

ield	Type	Description
FCDEST *	Char (32)	Logical destination
FCTYPE *	Char (1)	Type of entry in RFCDES
FCTYTEXT	Char (32)	RFC connection type
FCLOCK	Char (1)	Display type of destination
AVESERVER	Char (1)	Save target host as hostname
FCLOGON	Char (1)	Logon screen for RFC logon
RFCOPT	Char (1)	Maintain asynchronous RFC
FCHOST	Char (100)	Name of target host
FCSERVICE	Char (2)	Service used
FCALIAS	Char (32)	Alias name for destination
FCCLIENT	Char (3)	Explicit user ID
FCUSER	Char (12)	Explicit language

ables:3.14 VED_EDIPOD Definition of file ports

ield	Type	Description
MANDT *	Clnt (30)	Client
ORT *	Char (10)	Port
ORTTYP	Char (1)	Port type
ERSIO	Char (1)	Version of control record
ESCRI	Char (50)	Port description
CTRIG	Char (1)	Flag
HELLDIR	Char (70)	Name of a directory
HELLSCRIP	Char (30)	Command file
UTPUTDIR	Char (70)	Name of a directory
UTPUTFILE	Char (30)	File name for outbound data
UTPUTFUNC	Char (30)	Name of function module
NPUTDIR	Char (30)	Name of a directory
NPUTFILE	Char (12)	Name of function module
TATUSDIR	Char (30)	Name of a directory
TATUSFILE	Char (30)	File name for status records
UTPATHM	Char (1)	Logical or physical directory?

ables:3.15 TBD05 Distribution model for message types

eld	Type	Description
ANDT *	Clnt (30)	Client
USTMODEL *	Char (10)	View of ALE model
NDSYSTEM	Char (10)	Sender
ESTYP	Char (30)	Message type
CVSYSTEM	Char (10)	Receiving logical system
OMBINENUM	Numc(4)	Filter group
OUNTER	Numc(4)	Filter counter
TYPE	Char(1)	Filter type

ables: 3.16 EDIDC Control record (IDoc)

ield	Type	Description
ANDT *	Clnt (30)	Client
OCNUM *	Numc (16)	IDoc number
TATUS	Char(2)	Status of IDoc
OCTYP	Char(8)	IDoc type
IRECT	Char(1)	Direction for IDoc transmission
CVPOR	Char(2)	Partner type of receiver
CVPRT	Char(10)	Partner number of receiver

ABLE: 3.17 EDID2 IDoc Data Record from 3.0C onwards

ield	Type	Description
ANDT *	Clnt (30)	Client
OCNUM *	Numc (16)	IDoc number
OUNTER *	Numc (3)	Counter in cluster table
EGNUM *	Numc (6)	Number of SAP segment
EGNAM	Char(8)	Name of SAP segment
SGNUM	Numc (6)	Number of the hierarchically
LEVEL	Numc (2)	Hierarchy level
TINT2	Int (2)	Length field for VARC field
DATA	Lchr(1000)	Application data

ables: 3.18 EDIDS Status Record (IDoc)

ield	Type	Description
ANDT *	Clnt (30)	Client
OCNUM *	Numc (16)	IDoc number
OUNTER *	Numc (3)	Counter in cluster table
OGTIM *	Tims(6)	Time of status information
REDAT	Dats (8)	Date status record was created
RETIM	Tims (8)	Time status record was created
TATUS	Char(8)	Status of IDoc
NAME	Char(12)	User name
EPID	Char(30)	Program name
OUTID	Char(30)	Name of subroutine

3.5 DEVELOPMENT APPROACH

The two most commonly used approaches for the development and testing of software program are: The traditional "Bottom- up" approach and the contemporary "Top-down "approach. The Implementation of EDI using SAP was developed using the latter approach. The Top-down approach involves the identification of a model of the organization's information needs and them the design of subsystems is based on this model.

The complete document sharing the report of the development phase was sent to the management for their approval. The reports of the development phase will be reviewed by the management team and any restrictions are then rectified.

CHAPTER 4

SYSTEM TESTING

System testing is most vital activity that has to be enforced in any system development. This could be done parallel during the development phase and after implementation. The feedback received from this testing was carefully examined of further enhancement.

It is the part of testing where the entire implementation is tested. This testing is performed with the requirement document as the reference and the goal is to see whether the application meets the requirement.

4.1 White Box Testing

White box testing also referred to as glass- box software testing. It is a test case design that would use the 'program control flow' structure to derive software test cases. The software engineer can derive white box software testing using following guidelines.

- All independent paths within a module have been exercised.
- All loops are executed at their boundaries and within their operational bounds
- All internal data structure is exercised to assure their validity.

All the independent paths in the four modules of EDI implementation were tested and found consistent.

4.2 Black Box Testing

Black box testing enable the software engineer to derive set of conditions that will fully exercise all requirements for EDI design code. Also note that black box testing "is not an alternative "to white box testing. Rather it is a compulsory approach that it is likely to uncover a different CLASS OF ERRORS that what white box testing methods reveal. Black box testing reveal attempts to reveal errors in the following software work areas:

- Interface of inputs
- Database access
- Initialization and termination

The all interface in EDI were tested to find whether relevant screen for shown accordingly. The database connectivity and data retrieval were tested and found consistent.

4.3 Acceptance Testing

Acceptance testing involves planning and execution of the functional tests, performance test and stress tests in order to demonstrate that the implemented system satisfies its requirements. It is not unusual for two sets of acceptance tests to be run those developments by the quality Assurance group and those developed by the customer. The EDI was tested for load handling and data handling and found successful in handling large amounts of data and users.

4.4 Alpha Testing

A third person who just has the knowledge and the working capacity of the system conducts the alpha test at the developer's process. The developer 'looks over the shoulder' of the user and records the errors and usage problems. The user in turn gives General discomforts, which may be mended to make the system little better in a way of efficiency and user friendly.

The EDI was handed over to a user and was asked to navigate through the system and try all possibilities and all the requirements were satisfied.

4.5 Beta Testing

After alpha testing is done the developed process is given to other solution partners to check for errors. After this testing has been done then the product move to production system.

The EDI process was undergone beta testing and found error free and was move to production system.

CHAPTER 5

SYSTEM IMPLEMENTATION

Implementation includes all those activities that take place to creating a transaction code for each transaction process. Transport the process from development system to the production system by using transport option. A proper implementation is essential to provide a reliable system to meet the requirements of the customers and the company. An improper transportation may affect success of the process.

5.1 Implementation process

The implementation plan includes a description of all the activities that must occur to launch the development process and to put it into operation. It identifies the personnel responsible for the activities and prepares a time chart for transporting. The implementation plans consist of the following steps.

- Lists all files that have to be transport
- Identifies all data required to build new files during the upload.
- List all new document and procedures that go into the new system

The implementation plan should anticipate possible problem and must be able to deal with them. The usual problem may be missing document, mixed data formats between current files, errors in data translation, missing data etc.. The implementation started after all the modules of the process are integrated and tested together.

5.2 Future Enhancement

Enhancement is always an important and necessary activity in the life of the portal developed. Any system development should always provide room to accept any changes and further inclusions.

This may occurs so as to suit the new trend and additional facilities required by the user. The further enhancements are to be carried out in such a way that it should not affect existing facilities adds more attractive features in the process.

The future enhancements involves the following features

- Enhance the ALE configuration to following message
  - ORDERS Purchase order/order
  - DEBADV Debit display
  - REMADV Payment advice

- Using change pointer concepts in ALE
- Enhance report and graphical view

CHAPTER 6

CONCLUTION

By taking up this project for ROOTS Industries titled as Implementation of EDI using SAP I attained an extensive knowledge of how to use the existing new technologies to create a product based on the client requirements. This project is aimed to provide assistance for the employees of ROOTS Industries to satisfy their knowledge search. The programming techniques used in the design of the system provide a scope for further expansion and implementation of any changes, which may occur in future. Maximum care and concentration has been focused to troubleshoot this project. This design can be applied for any organization under similar circumstance.

Invaluable experience has been gained in these areas of users system and database design, system testing and implementation. Since the requirements of any organization and their standards are changing day to day the system has been designed in such a way that its scope and boundaries could be expanded in future with little modification. The portal has been thoroughly tested and all the client needs are assured. A key issue in security has been checked for consistency. The tool has been successfully launched by giving rights to authorized users. Users can gain access to the portal by subscribing to the technology of his choice. The project has been taken up with at most dedication and it has been developed in such a way to accommodate any type of change.

APPENDIX 1

SNAPSHOT OF SCREENS

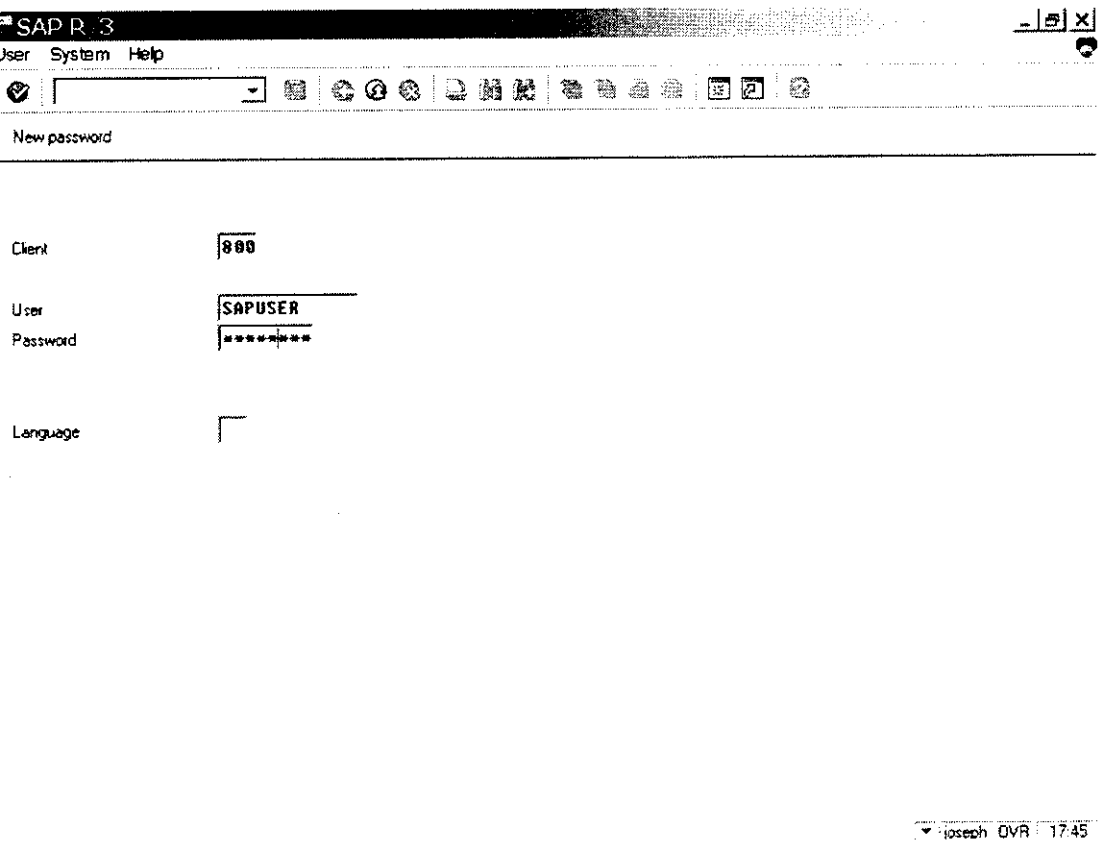


Figure A 1.1 Logon Screen

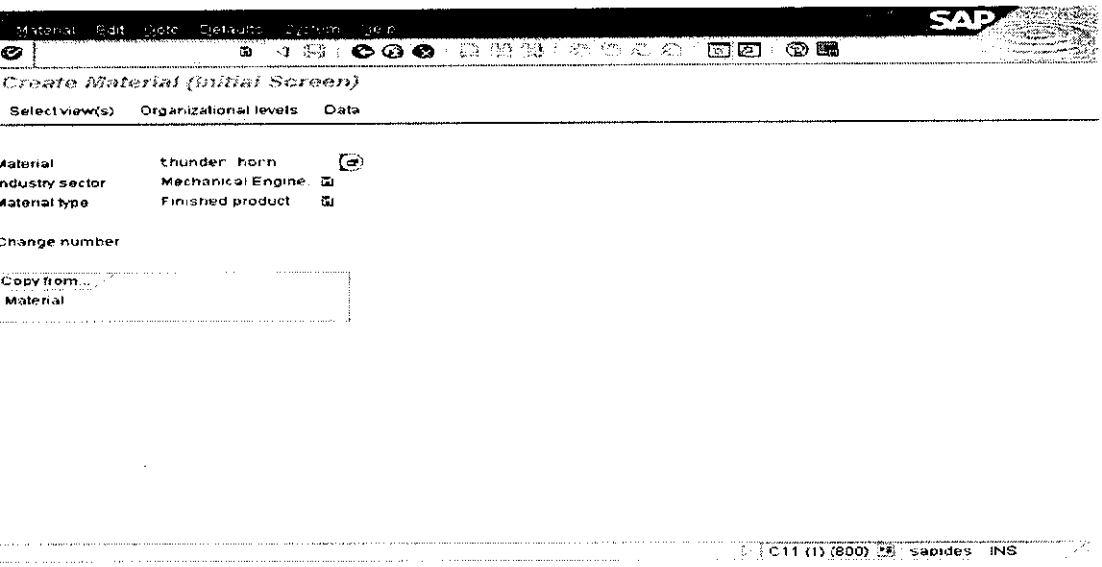


Figure A 1.2 Create Material

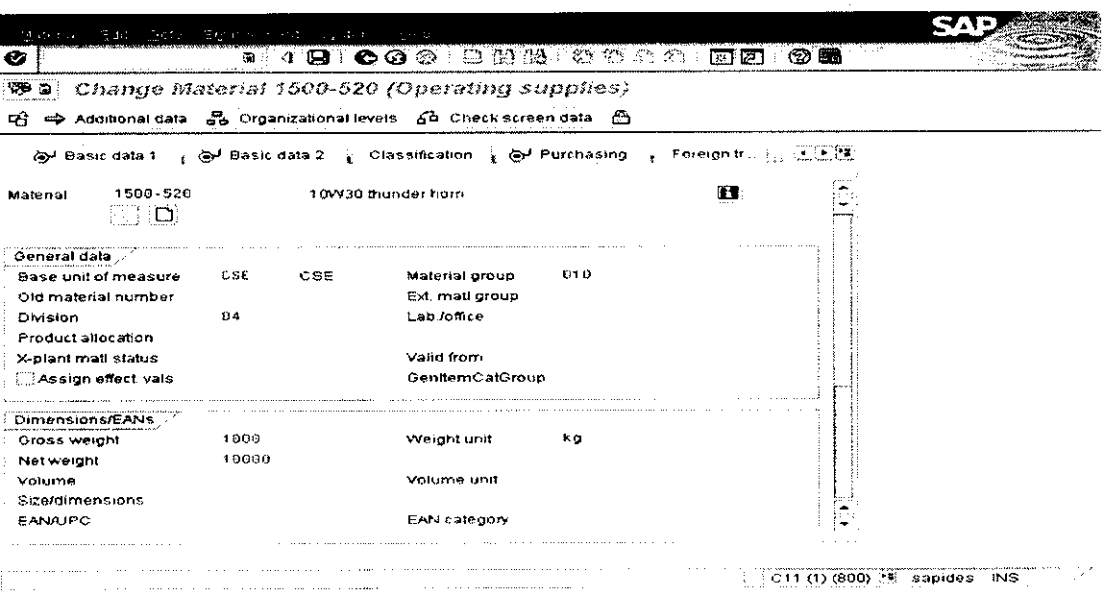


Figure A 1.3 Change view of material Basic data

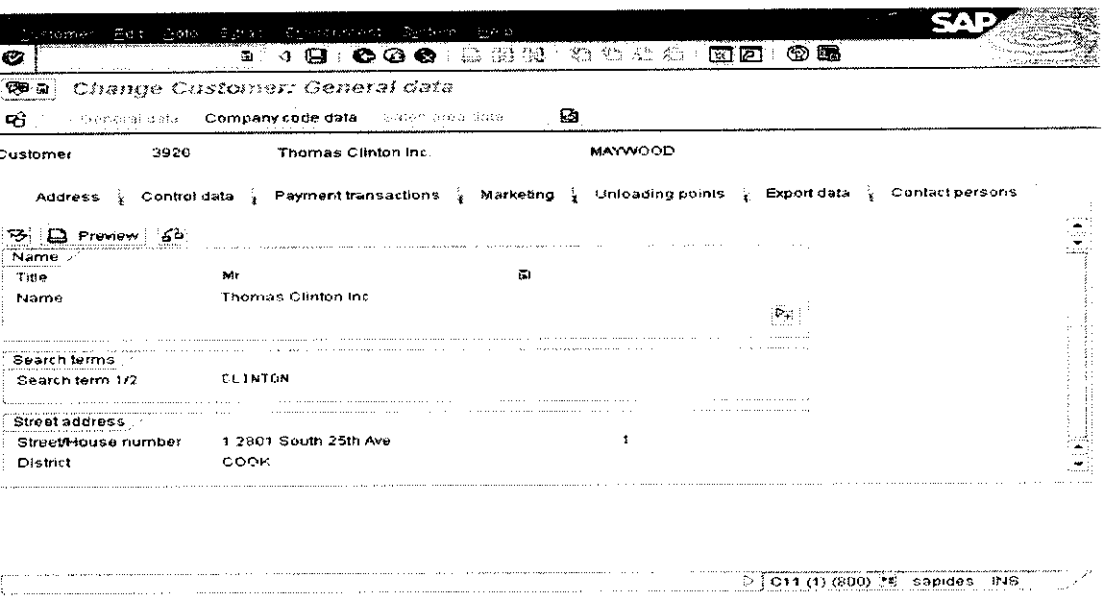


Figure A 1.4 Create customer

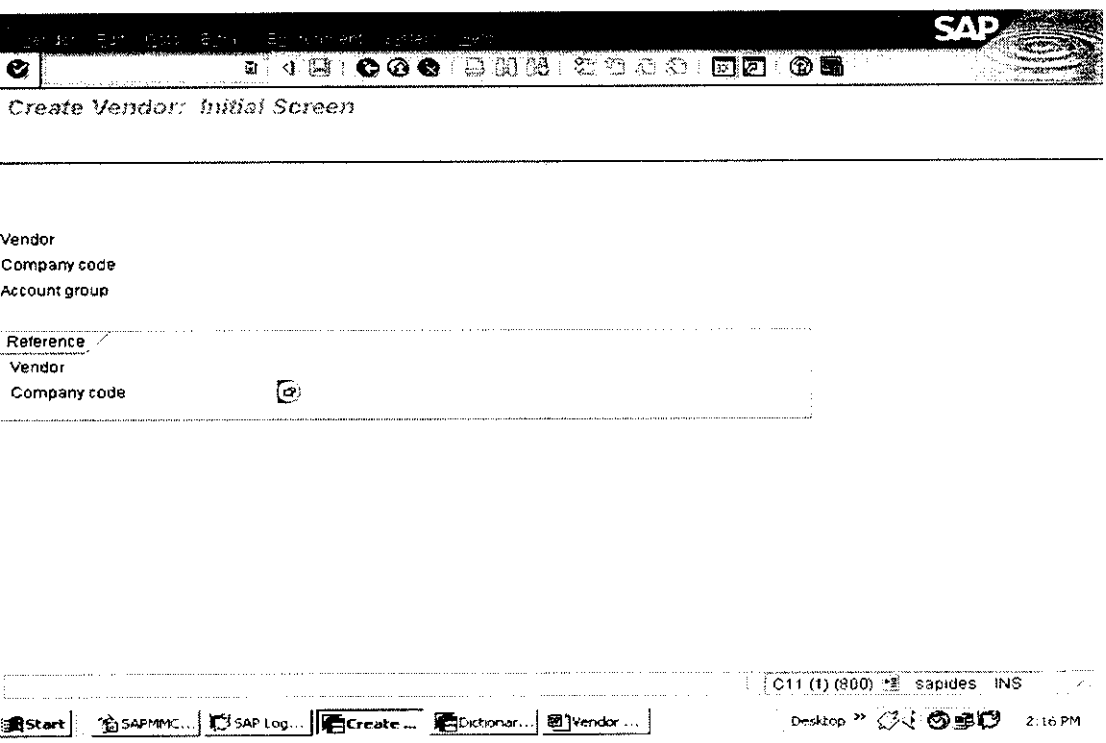


Figure A 1.5 Create Vendor

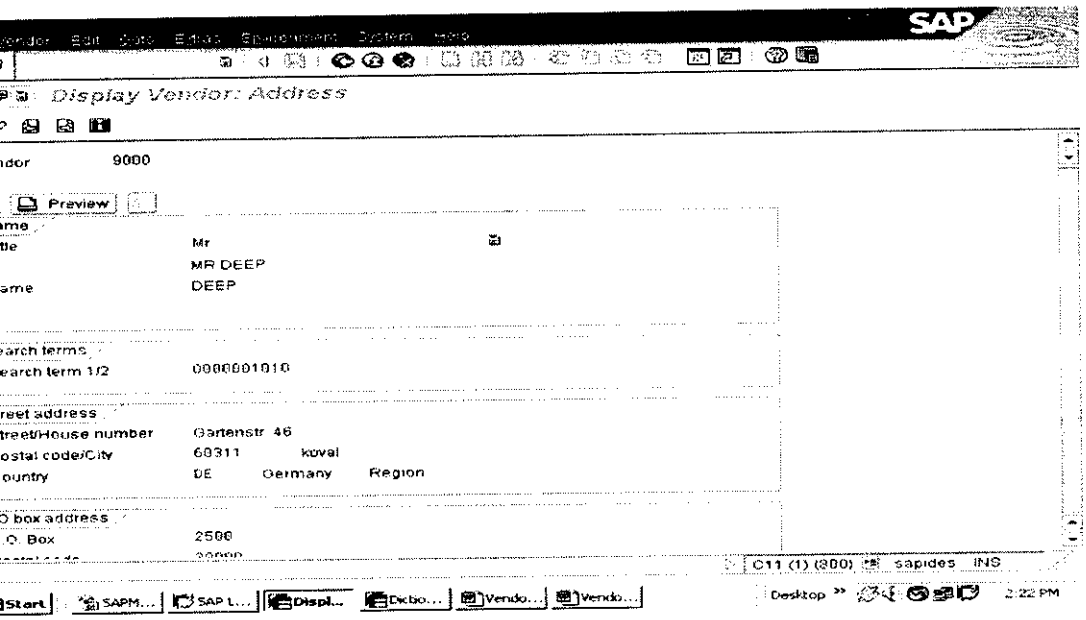


Figure A 1.6 Display of vendor address

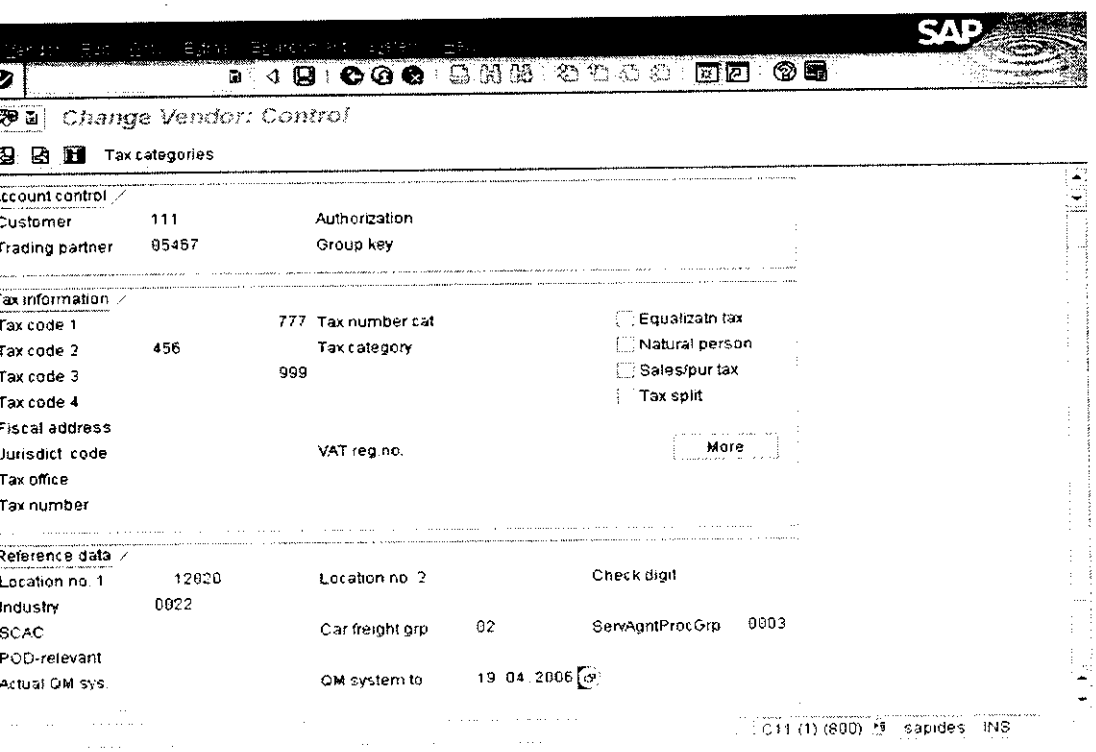


Figure A 1.7 vendor control

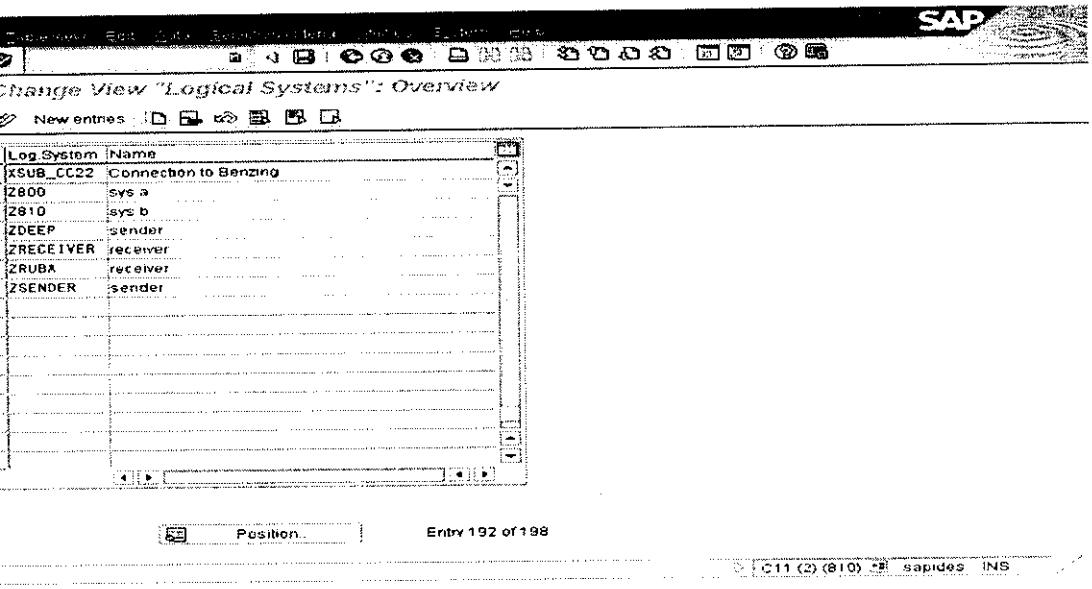


Figure A 1.8 Create Logical System

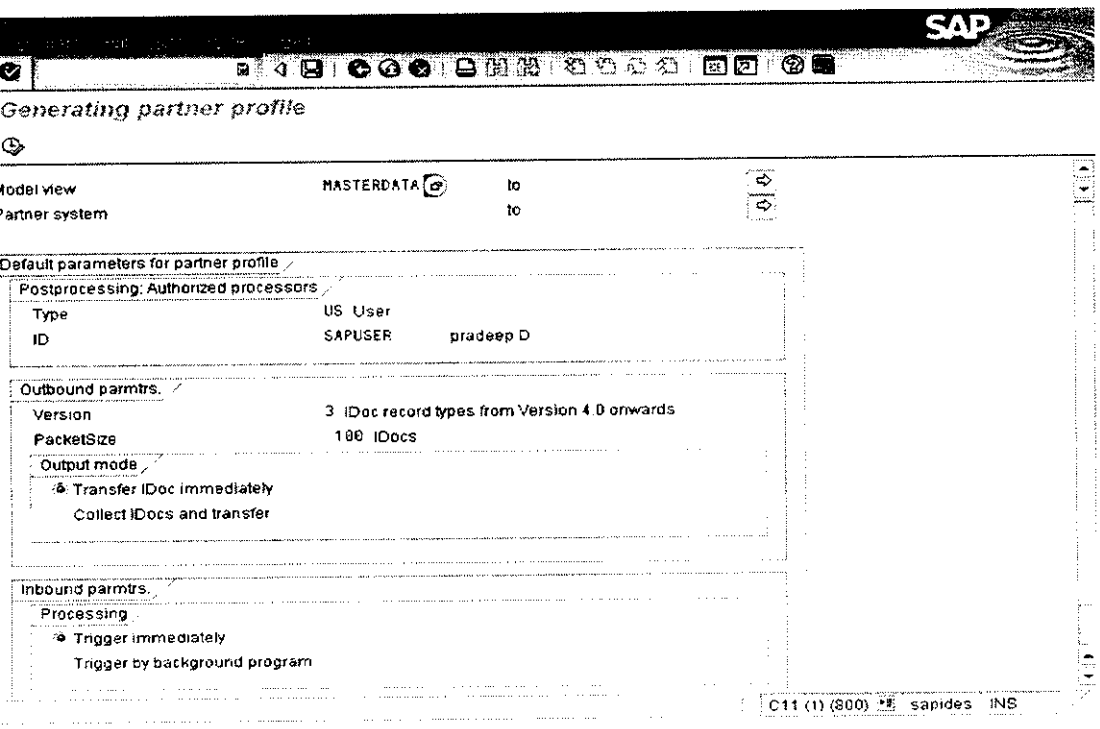


Figure A 1.9 Generate Partner Profile

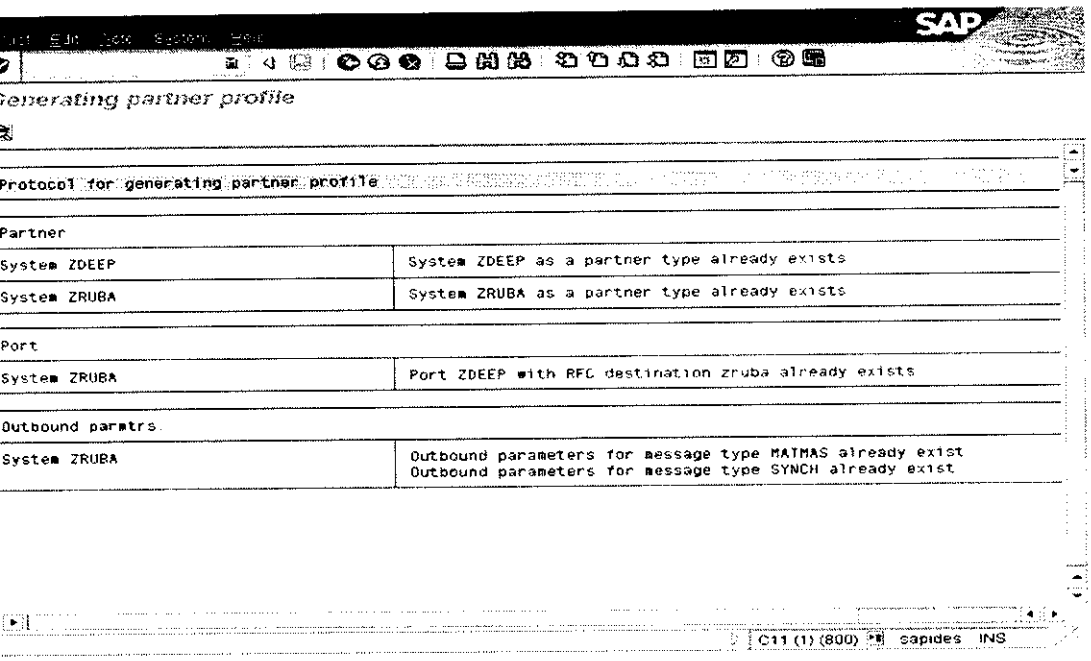


Figure A 1.10 Result of Generation of Partner Profile

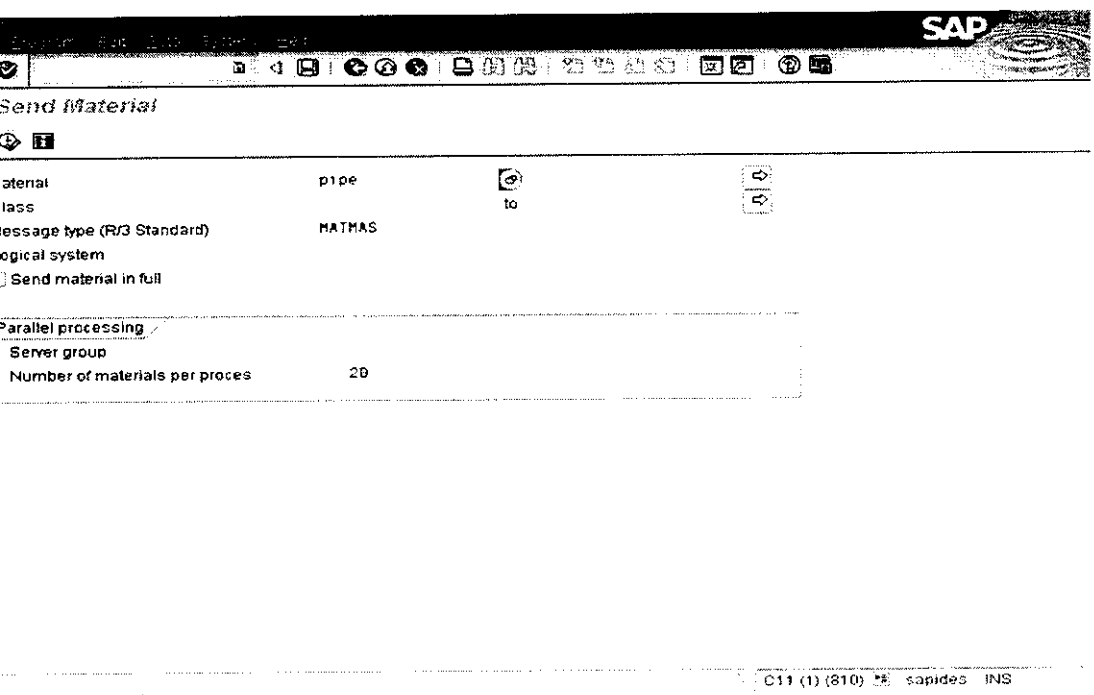


Figure A 1.11 Sending MATMAS Data

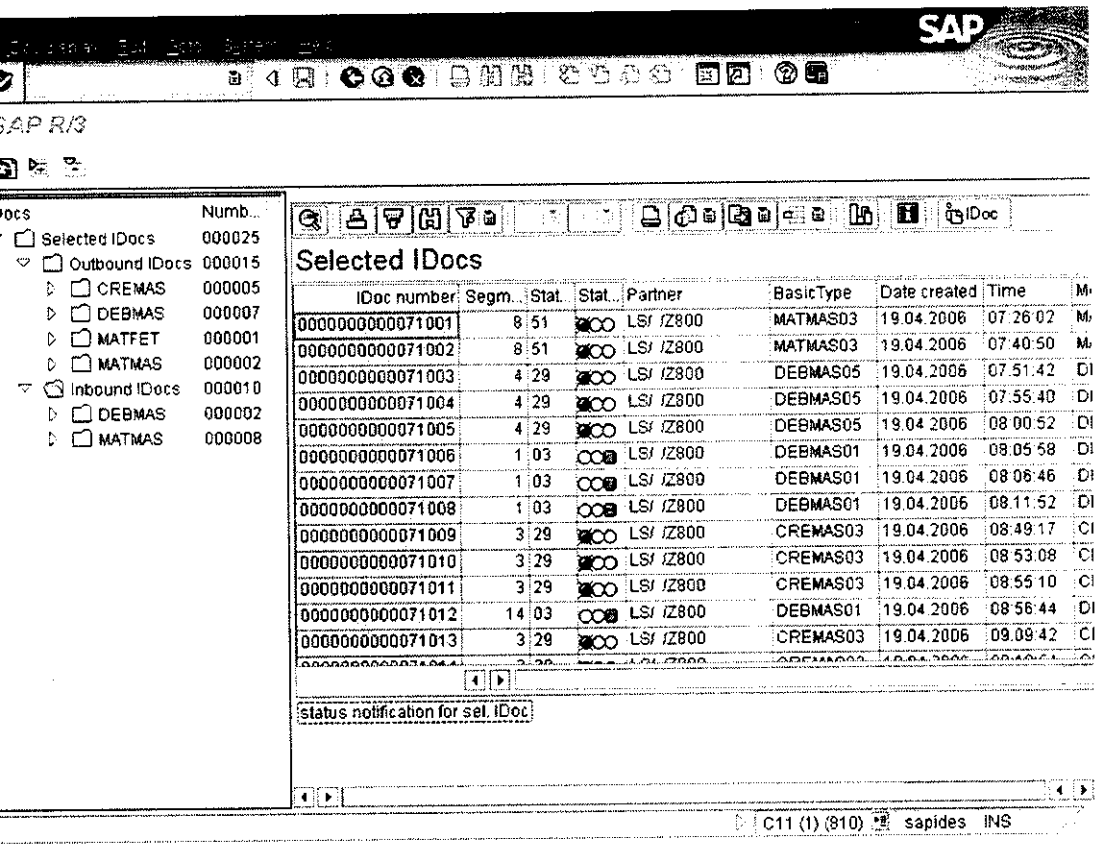


Figure A 1.12 Graphical view of IDoc Status

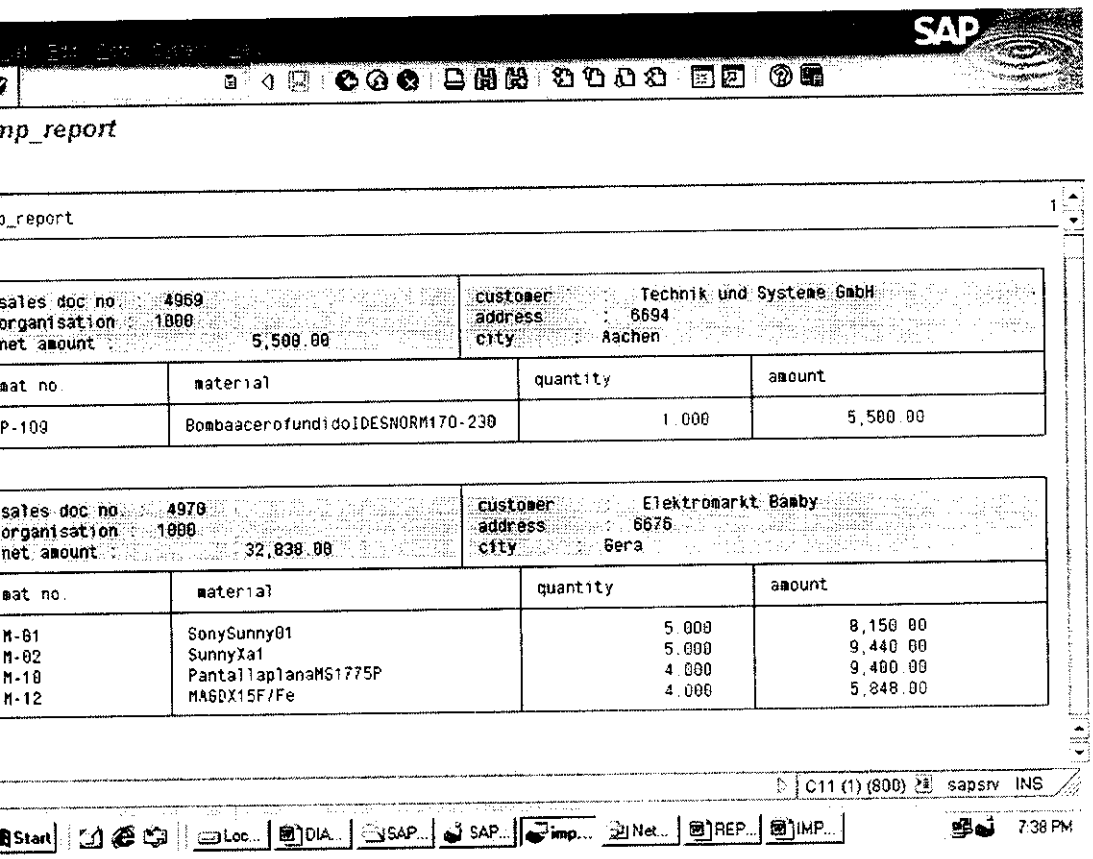


Figure A 1.13 General Material Report

REFERENCES

1. Arvind Nagpal with John Pitlak (2002) ALE, EDI, &IDoc Technologies for SAP.

   Prentice Hall of India Private limited, New Delhi-110 001.

2. Werner Hertleif, Christoph Wachter (2003) SAP Smart Forms

   SAP PRESS is an imprint of Galileo Press, New York, Bonn.

3. Palo Alto SAP (1999) Script made easy, SAP labs , California.

4. Alexis Leon, ERP Demystified Tata McGraw-Hill publishing Company Ltd, Delhi.

5. Jens Stumpe, Joachim Orb SAP Exchange Infrastructure.

   SAP PRESS is an imprint of Galileo Press, New York, Bonn.

6. ALE configuration Tutorial

   http :// www.sapgenie.com

7. Detail about IDoc

   http:// www.idoc.de

8. SAP R/3 IDoc Cookbook for EDI and Interfaces

   http:// www.logosworld.com