STREAMLINING OF LANKA BULK DEPOTS' STOCK PLANNING

P.D.N. Rukshani

129165A

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa, Sri Lanka for the partial fulfillment of the requirements of the Master of Science in Information Technology.

November 2016

Declaration

I hereby declare that the thesis submitted by me is based on original works carried out by me. Any reference to work done by any other person or institution or any material obtained from other sources have been duly cited and referenced. I further certify that the dissertation has not been submitted in any form for any other degree or diploma at any university or other institution of tertiary education.

.....

P. D. N. Rukshani

Date: 09.12.2016

I certify that the above particulars given are true and correct to the best of my knowledge.

UOM Verified Signature

Mr. Saminda Premarathna

Supervisor

Date: 21/12/2016

Dedication

I dedicate my dissertation work to my family with a special feeling of appreciation to my loving mother who is the strength of my life.

Acknowledgement

I would like to extend my sincerest gratitude to all the persons who helped and supported me to get ahead with my research works.

Firstly, Mr. Saminda Premarathna, Senior Lecturer, University of Moratuwa as supervisor for the support, advices, comments given by him to succeed my research. And also I would also like to thank all the lecturers and the staff of the faculty of Information technology, University of Moratuwa for their assistance throughout the course.

Secondly, I wish to thank the head of the department and the staff members of Ceylon Petroleum Storage Terminals Limited Information Technology function for supporting me to complete my research. Special thanks should go to Mr. Janak Dharmaprema for providing me essential materials related to fuel distribution process.

At last but not least, I extend my heartiest gratitude to my husband, my little baby methuki and my mother for their unconditional love and encouragement. I would not have been able to complete this thesis without their support during the last two years.

Abstract

Petroleum has been used since ancient times, and is now important across the society. As the world continues to industrialize rapidly, demand is increasing day by day. At the same time unit cost of petroleum products also increasing very frequently.

Even though CPSTL has been automated recently using SAP ERP system, stock planning has not been utilized yet. Therefore, delays in product delivery, over/under scheduling, lack of monitoring/ control mechanisms, damage of company prestige, creates windows for various malpractices/congestion, unnecessary costs incurred, improper utilization of resources is experiencing currently. As a result, CPSTL may lose its power of fuel storing and distribution.

These problems were brought up by many proposals, including the 'Chief Internal Audit' report in many occasions. Clearly, the stock planning for island wide LBDs' has to be addressed as a top priority issue, since it affects CPSTL as an organization.

To address the above issues, stock planning for the bulk depots can be improved by the utilization of ERP system. That is, the stock planning should be carried out while considering the storage capacities, available space, expected sales for the day and various other factors with respect to the depot. Bulk depots have to be replenished with respective materials while satisfying the demand without over scheduling.

I believe that, the stock planning for the bulk depots can be enhanced by the utilization of ERP system concerning the system took over as of now. This will improve operational efficiency, reduce information delays and errors, satisfy the island wide demand, speedup stock transfers, improve customer service and improve overall productivity.

Table of Contents

List of Tables

List of Figures

Abbreviation

CPC	Ceylon Petroleum Corporation
CPSTL	Ceylon Petroleum Storage Terminals Limited
EPR	Enterprises Resource Planning
LBD	Lanka Bulk Depots
IT	Information Technology
BTR	Bulk transfers
SAP	Systems Applications and Products in Data Processing
ERP	Enterprise Resource Planning
CRM	Customer Relationship Management
SD	Sales & Distribution
MM	Materials Management
FI	Finance Management
DEV	Development System
QAS	Quality Assurance System
PRD	Production System
ABAP	Advanced Business Application Programming
ALV	ABAP List Viewer
OOD	Object Oriented Design
OOA	Object Oriented Analysis
DM	Distribution Manager

Chapter 1 – Introduction

1.1 Introduction

Since the inception of LBDs many decades ago, the conventional manual method of maintaining stock books and conveying LBD stock details to CPSTL distribution was functional. This method had various drawbacks, bottlenecks and disadvantages. To overcome these issues, it's must to streamline the stock planning for the bulk depots improved by the utilization of ERP system.

1.2 Background and Motivation

Keeping in mind the end goal to satisfy the nation's fuel prerequisite, the fuel items are being imported as crude oil or finished products. Kolonnawa and Muthurajawela oil installations are in charge of the capacity and dissemination of these refined and finished products. The oil installations are being replenished by the tankers and the CPC Sapugaskanda Refinery. Lanka bulk depots have been set up island wide, with a specific end goal to supply the demand of the nation's fuel prerequisite. The mass stations are being recharged by the aforementioned oil installations by Rail (wagon) and Road (Hired and CPC browsers).

In fuel distribution apart from main installations LBDs' acts a major role. Because of workers' states of mind practices still CPSTL not ready to handle an immaculate stock planning process. Even though CPSTL has been automated recently using SAP ERP system, stock planning has not been utilized yet. Therefore, delays in product delivery, over/under scheduling, lack of monitoring/ control mechanisms, damage of company prestige, creates windows for various malpractices/congestion, unnecessary costs incurred, improper utilization of resources is experiencing currently.

To address the above mentioned issues, stock planning for the bulk depots can be improved by the utilization of ERP system. That is, stock planning carried out (top up the tanks), while considering the storage capacities, available space, and expected sales for the day various other factors with respect to the depot. Bulk depots have to be replenished with respective materials while satisfying the demand without over scheduling.

I believe that, the stock planning for the bulk depots can be enhanced by the utilization of ERP system concerning the system took over as of now. With this proposed solution, tanks will be finished up as for accessibility and request as opposed to outsider inclusions. Streamlined process will improve operational efficiency, reduce information delays and errors, satisfy the island wide demand, speedup stock transfers, improve customer service and improve overall productivity. At last, it will lead to streamlining the ongoing process and make better fuel, storing and distribution process and becomes the most efficient petroleum terminal operator in South Asia as per CPSTL mission [1; 2].

1.3 Aim and Objective

Aim

The aim of this research is streamlining LBDs' stock planning process.

Objectives

The objective of this research is to propose that, the stock planning for the bulk depots improved by the utilization of ERP system, with respect to the procedure followed currently.

- Studying the existing LBD stock planning process.
- Identify the downsides, bottlenecks and drawbacks in the present system.
- Design and develop the proposed solution.
- Evaluate the usage, adaptation, benefits, and impacts of proposed solution.
- Improve LBDs' operational efficiency, reduce information delays and errors, satisfy the island wide demand, speedup stock transfers, improve customer service and improve overall productivity.

To satisfy the above objectives I proposed a solution which contains, processes such as

- BTR deliveries (wagon and tank lorries).
- Daily Quota Allocation.

And the features, such as,

- Conclusive Report on all aspects of stock planning for performance checking.
- Value added services such as sending bulk SMS to the customers [9;10].

- Availability of data for further analysis and managerial decision making.

1.4 Structure of the Dissertation

Chapter 2 describes the current issues in LBDs' stock planning, which is about the downsides, bottlenecks and drawbacks in the manual process. Chapter 3 is on Technology Adapted, which describe ABAP and its features which I used to develop the proposed solution. Chapter 4 is the Approach, which is how to use ABAP as a programming language to streamline the LBDs' stock planning process. The conceptual solution that fulfills the proposed solution is described in chapter 5, which is analysis and design. Next chapter is Implementation, how to develop the modules described in design chapter. Chapter 7 is the evolution, which is about the testing proposed system against objectives. At last it is an overall discussion about the research and further works that I wish to carry out, conclusion and future works.

1.5 Summary

The customary manual strategy for keeping up stock books and passing on LBD stock details to CPSTL distribution was functional. This strategy had different downsides, bottlenecks and drawbacks. These issues were raised by numerous workforces, including the Chief Internal Auditor CPSTL in numerous events. Obviously, the stock planning for island wide LBDs' must be tended to as a top need issue, since it affects CPSTL as an organization.

Chapter 2 – Current Issues in LBDs' Stock Planning

2.1 Introduction

The revolution of the computer and its appliance in the last century has brought several changes to business and organizations everywhere all over the world. IT is one of the most excellent applications of computer discovery, which concerns nearly all areas of the business. By considering this circumstance SAP ERP system has been acquainted with the distribution process. Be that as it may not fully automate for all the areas of stock planning operations. In this way various issues have been confronted amid most recent couple of years.

2.2 Manual Stock Planning Process

Order Book

The customer's indent is captured in the order book. Once place an order it is added to an excel sheet or a hand written paper to recognize the aggregate orders have accomplished for the day.

ORDER BOOK

06.09.2013

							90 OCT	LAD	LK	LFO
A/C NO	CUSTOMER NAME	LOCATION	KM	TIME	ORDER NO	OUTBOUND NO				
100243	U.K.BUDDHADASA SILVA	TALAWAKELLE	24	1.45	237225	401688266		6600		

Figure 2.1: Order Book

Drawbacks

- Different formats

  In general figure 2.1 is the format of the order book. Be that as it may, it is different from depot to depot.

- Different methods

  Some depots practicing creating open orders in earlier days and it is adding to following day order book. But all the open orders may not turn out to be genuine requests. In this way superfluous orders will be added to order book and as a result total order count or the daily total sale might be off base.

Depot stock position

The depot stock position was prepared by the LBD superintendents and sent to the Kolonnawa distribution function via fax/mail. This is a filled description about the stock positions, including today's sales, yesterday's sales and receipts and staff and bowser positions.

STOCK POSITION

DEPOT: PERADENIYA JUNCTION : 31.07.2013

Product	Usable Stock (Ltr.)	Today Sales(Ltr.)	Yesterday Sales CPC	LIOC	No. of un discharged Wagon in siding	
Lanka Petrol 90	427400	204600	132000	13200	BMT	-
Lanka Petrol 95	-	13200	-	-		-
Lanka Auto Diesel	1122500	306900	303600	19800	BGT	-
Lanka Super Diesel	-	13200	-	-	-	-
Lanka Kerosene	167800	42900	-	-	BKT	-

(2) YESTERDAYS'S RECEIPT: 30.07.2013.

	PRODUCT					
	LP 90 OCT	LAD	LK	LSD	95 OCT	LFO
(1) BY RAIL	52740	52740	-	-	-	-
(2) BY ROAD	105600	264000	26370	-	-	-

(3)STAFF POSITION

DS – A5	01
Asst. DS – A7	01
Ex. Asst. – A7	01
Depot Supervisor	01
Clerk	18
Typists	04
Record keepers	02
Store keepers	01
Office aids	03
Security Officer	01
Asst. Sec. Officers	01
Gate Checkers	06
Security Asst.	10
Drivers	18
Porters	19
Mechanics	01
Asst. Mechanics	05
G. Laborers	16
Tank Gougers	02
Fireman	01

(4) BOWSER POSITION

CPSTL LORRIES		
LORRY NO	CAPACITY	CONDITION
GC-181	13200	WORKING
GC- 1287	13200	working
GC - 0178	13200	working
LB -2142	6600	working
GE - 7754	13200	working
GE - 0880	13200	working
GC. - 7960	13200	working
226 - 7027	19800	Working
226-1963	19800	working
GC-1569	13200	U/R kolo.
GC-2018	13200	U/R kolo

Last Filling ; 19.05
Last Wagon /BTR : 13.30 /18.35
Closing time : 19.35

Prepared by : 12894

Figure 2.2: Deport stock position

Drawbacks

- Different formats

  Same as the order book stock position also have altered format as for the depots.

- Different methods

  Sales details are depending on the order book. On the off chance that the order book has diverse technique stock positions additionally changed.

- Accuracy level of the data

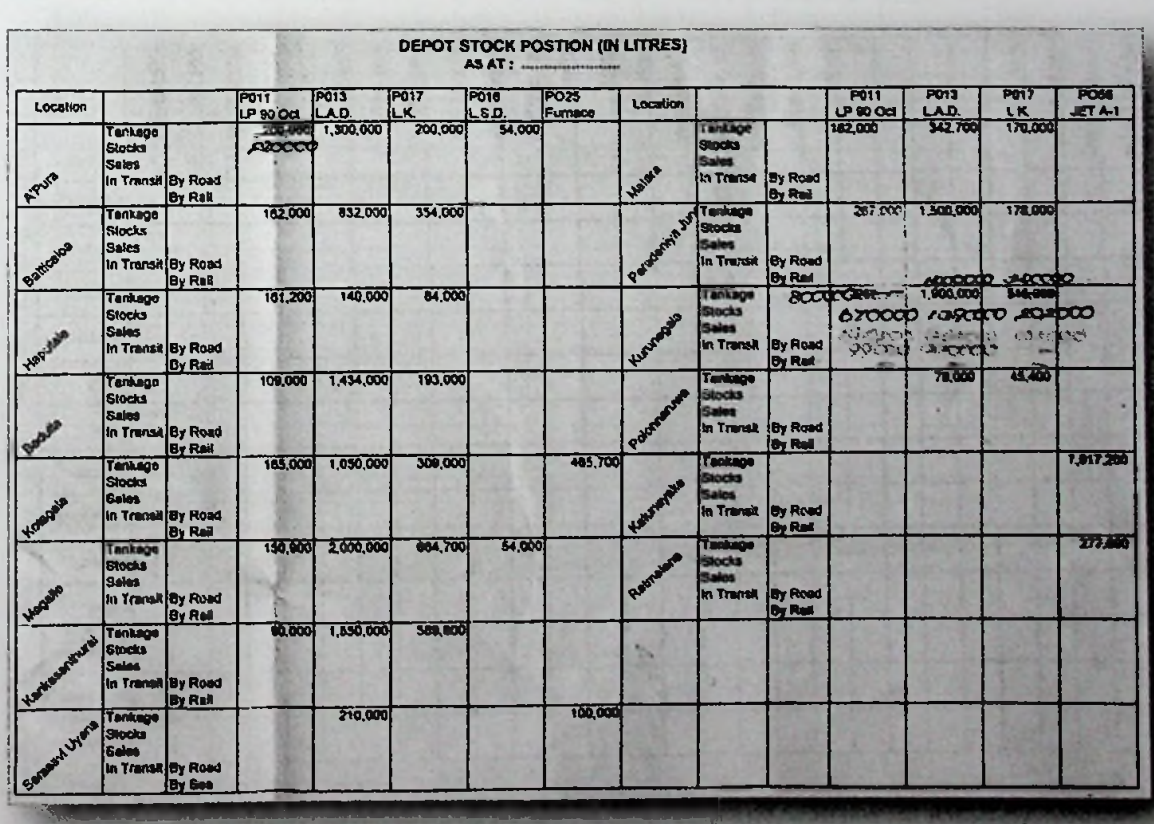
  The data updated in the stock position cannot assure 100% precise and up to date.

- No specific time

  There is no explicit time to send stock details to the main installation. It sent at different times and may not be sent at all if the responsible person is on leave.

Island-wide depot sock position

Island-wide LBD Stock position was maintained by the Kolonnawa Wagon Section on a cumbersome stock book. All the depot stock planning was performed based on this manually updated information.



DEPOT STOCK POSTION (IN LITRES)
AS AT : .....

Location			P011 LP 90 Octane	P013 L.A.D.	P017 L.K.	P018 L.S.D.	P025 Furnace
A'Pura	Tankage			1,300,000	200,000	54,000	
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Batticaloa	Tankage		182,000	832,000	354,000		
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Hapitale	Tankage		161,200	140,000	84,000		
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Peradeniya Jun	Tankage						
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Kurunegala	Tankage						
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Polgahawela	Tankage						
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Kelaniya	Tankage						
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Ratmalana	Tankage						
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					

Location			P011 LP 90 Octane	P013 L.A.D.	P017 L.K.	P056 JET A-1
Matara	Tankage		182,000	342,700	170,000	
	Stocks					
	Sales					
	In Transit	By Road				
		By Rail				
Peradeniya Jun	Tankage		267,000	1,500,000	178,000	
	Stocks					
	Sales					
	In Transit	By Road				
		By Rail				
Kurunegala	Tankage			1,900,000	146,000	
	Stocks					
	Sales					
	In Transit	By Road				
		By Rail				
Polgahawela	Tankage			78,000	45,400	
	Stocks					
	Sales					
	In Transit	By Road				
		By Rail				
Kelaniya	Tankage					7,917,200
	Stocks					
	Sales					
	In Transit	By Road				
		By Rail				
Ratmalana	Tankage					277,600
	Stocks					
	Sales					
	In Transit	By Road				
		By Rail				

Kankasenthurai	Tankage		60,000	1,550,000	589,600		
	Stocks						
	Sales						
	In Transit	By Road					
		By Rail					
Sapugaskanda/ Uyana	Tankage			210,000		100,000	
	Stocks						
	Sales						
	In Transit	By Road					
		By Sea					

Figure 2.3: Island-wide depot stock position summary

Stock in-transit

Stocks in-transit was maintained daily by the wagon section of the Kolonnawa distribution function, upon the data sent by the LBDs.

STOCK IN TRANSIT AS ON.....

Depot	Lanka Petrol 90 Octane										
	13200	19800	26400	33000	22275	22460	26370	45400	49000	50000	Total(Lt)
A'pura											
IRD A'pura											
Badulla											
Batticaloa											
Magalle											
Haputhale											
K.K.S											
Kotagala											
Kurunagala											
Matara											
Peradeniya											
S/Uyana											
Vauniya											

Depot	Lanka Auto Diesel(Lt)										
	13200	19800	26400	33000	22275	22460	26370	45400	49000	50000	Total(Lt)
A'pura											
IRD A'pura											
Badulla											
Batticaloa											
Magalle											
Haputhale											
K.K.S											
Kotagala											
Kurunagala											
Matara											
Peradeniya											
S/Uyana											
Vauniya											

Depot	Lanka Kerosene										
	13200	19800	26400	33000	22275	22460	26370	45400	49000	50000	Total(Lt)
A'pura											
IRD A'pura											
Badulla											
Batticaloa											
Magalle											
Haputhale											
K.K.S											
Kotagala											
Kurunagala											
Matara											
Peradeniya											
S/Uyana											
Vauniya											

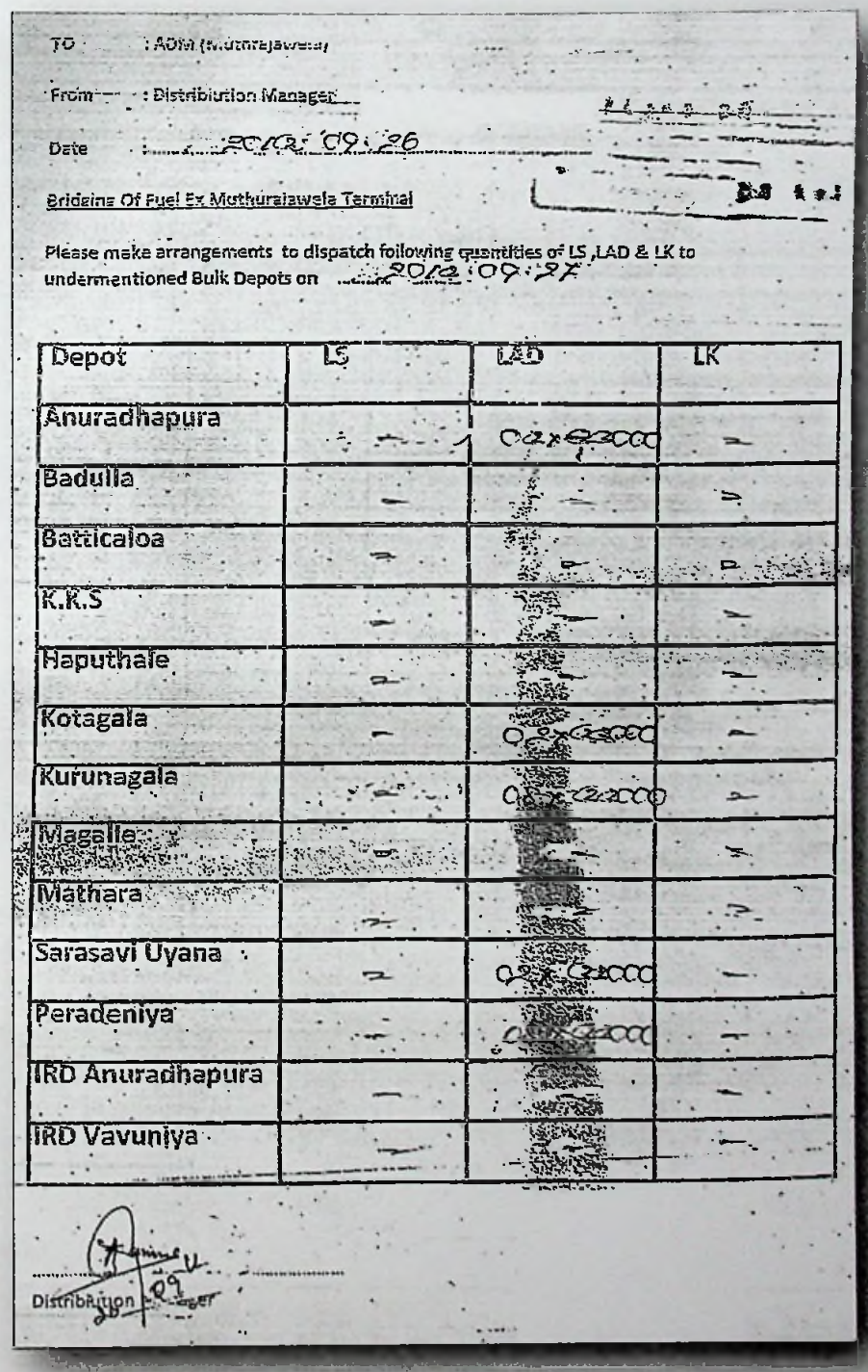
Figure 2.4: Stock in-transit

Drawbacks

- No real-time information.
- Unable to enforce any control mechanism.
- No centralized information to be accessed by different plants.

Bridging of fuel

Distribution Manager sent manually prepared instructions to Muthurajawela terminal, LBD Anuradhapura and IRD Vavuniya via fax for the bridging/replenishing of LBDs.



TO : AOM (Muthurajawela)

From : Distribution Manager

Date : 2012:09:26

Bridging Of Fuel Ex Muthurajawela Terminal

Please make arrangements to dispatch following quantities of LS ,LAD & LK to undermentioned Bulk Depots on 2012:09:27

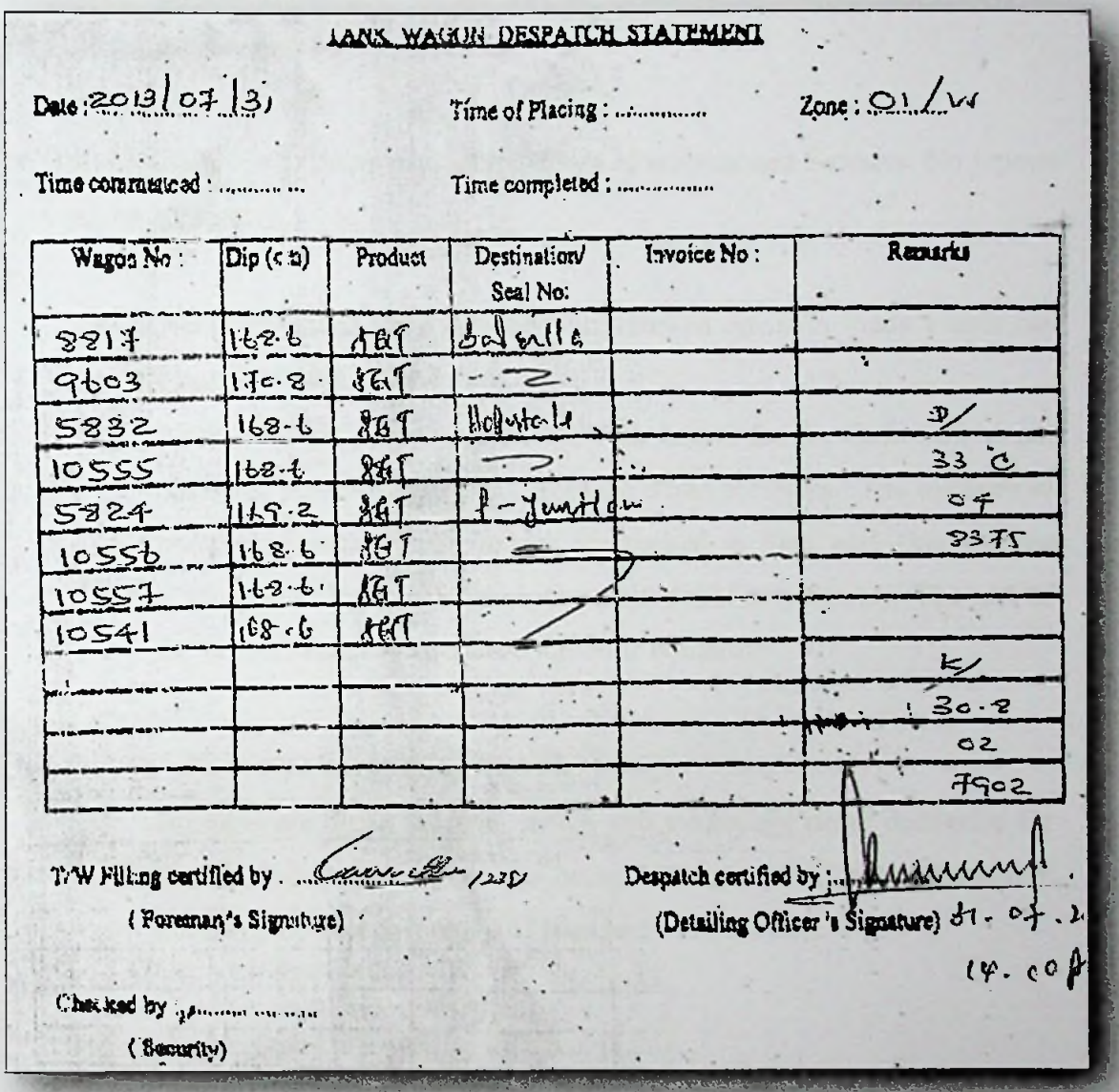
Depot	LS	LAD	LK
Anuradhapura			
Badulla			
Batticaloa			
K.K.S			
Haputhale			
Kotagala			
Kurunagala			
Magalle			
Mathara			
Sarasavi Uyana			
Peradeniya			
IRD Anuradhapura			
IRD Vavuniya			

Distribution Manager

Figure 2.5: Bridging of fuel

Wagon dispatch statement

Fuel wagon dispatch statements and instructions are carried out by manual forms.



LANKA WAGON DESPATCH STATEMENT

Date : 2013/07/31 Time of Placing : ..... Zone : 01/W

Time commenced : ..... Time completed : .....

Wagon No :	Dip (cm)	Product	Destination/ Seal No:	Invoice No :	Remarks
8817	162.6	AGT	Badulla		
9603	170.8	AGT	"		
5832	168.6	AGT	Hofuta-la		D/
10555	168.6	AGT	"		33 C
5824	165.2	AGT	K-Junction		04
10556	168.6	AGT	"		3375
10557	168.6	AGT			
10541	168.6	AGT			
					K/
					30.8
					02
					7902

T/W Filling certified by 1235

(Foreman's Signature)

Despatch certified by .....

(Detailing Officer's Signature) 31.07.2

14.00 h

Checked by .....

(Security)

Figure 2.6: Wagon dispatch statement

Drawbacks

- Non availability of statistics/reports.
- No availability of information for forecasting identification or optimization.

2.3 Drawbacks of Present System

- Over scheduling of Bulk loads.

  Over-scheduling of BTRs can be observed in the currently adopted procedure for stock planning.

- No method of monitoring or tracking down of wagons and bowsers. No reports can be generated.

- Due to over-scheduling, bowsers are being parked on main roads which can lead to congestion and malpractices.

  As an example, the overscheduled tank lorries for the Galle bulk depot are being parked 13km away from the depot premises. The distance to the place where excess lorries are parked is 3km and 1km for the Matara bulk depot and Peradeniya bulk depot respectively. The rest of the depots seem to experience the same scenario.

- Bowers and wagons not arriving expected date.

  Bowsers are being delayed, which will eventually delay deliveries for the other depots. Wagons are being held up due to over-scheduling, delaying the cheapest mode of transport of bulk loads.

- Different procedures followed by different plants.

  Different procedures are being followed by different depots in handling of sales orders and providing sales information to the wagon section.

- No restriction / controlling mechanism for the scheduling of bulk loads.

- Oil shortages are being deducted from the transportation bill of the hired bowsers.

  Lorries parked outside the depot premised can lead to oil shortages due to various malpractices and weather conditions. The shortages from

hired bowsers are being deducted from their transportation bill, which is an issue the transporters have brought up several times.

Bulk Depot	Penalty charge (Rs.)
Badulla	95,166.00
Galle	16,846.80
Haputhale	44,881.00
Matara	136.034.00
Peradeniya	4,848.00

Table 2.1: BTR penalty charges totals for the period 01/08/2013 to 31/08/2013

These problems were brought up by many personnel, including the 'Chief Internal Audit' report (figure 2.7) in many occasions. Clearly, the stock planning for island-wide LBDs have to be addressed as a top priority issue, since it affects CPSTL as an organization.

Memo

To: Chairman

From: Chief Internal Auditor

CC: DGM(F), Manager IT, Manager Distribution

Date: April 27, 2011

Re: Data on product transfers to depots and their delays in taking into the Storage Tanks.

We have checked the BTR's issued and the delays on the receipts of products in the receiving location

The total intransit loads delayed for the period from April 2010 to March 2011 for more than one day exceeds 8,000 loads.

We have not highlighted the delays for one day.

Summarized details as follows

Depot	Delay of 2 days	Delay of more than 2 days within 5 days	Delay of more than 5 days within 10 days	Delay of more than 10 days within 15 days	Delay of more than 15 days and more
Anuradhapura	732	389	65	3	10
IRD Anuradhapura	614	226	18	1	3
Badulla	484	566	138	14	6
Batticaloa	245	563	174	9	1
Galle	362	48	-	-	2
Haputale	339	281	94	11	1
Kankesanthurai	363	40	4	6	47
Kotagala	223	126	13	1	-
Kurunegala	714	261	18	3	6
Matara	884	147	-	-	34
Peradeniya	588	270	13	-	11
Sarasavi Uyana	67	7	-	-	-

The numbers are as indicated in the SAP. This table indicates higher intransit delays are at Anuradhapura – 1230 loads, Badulla – 1200 loads, Batticaloa - 1000 loads. Both Kurunegala & Matara delay are 1000 loads at each location. At Anuradhapura 732 loads have got delayed for 2 days. This needs to be monitored for an efficient operation. The data also indicates there have been loads delayed without discharging for 3 months.

For efficient distribution system these should not have been taken place. We have not identified the reasons for these delays. The managers have to identify the reasons and take corrective action.

Details of the schedules have been sent to relevant managers on SAP mail. Summarized details are attached for your information. Products transfer to depot and IRDs needs to be monitored for an efficient distribution.

Chief Internal Auditor

Cc: Manager (Internal Audit)

Figure 2.7: Internal audit report

2.4 Summary

The stock positions of the depots are being gathered via the traditional methods of telephone and fax. The material replenishment of the depot is being done via wagons and tank lorries. Stock book records are being created and maintained. To overcome the issues faced due to manually process, it needs to be automated with the assistance of the SAP ERP system. Appliance of new solution into distribution process can improve operational efficiency, reduce information delays and errors, improve customer service and improve overall productivity.

Chapter 3 – Technology Adapted

3.1 Introduction

From a business applications software perspective, SAP is nearly all things to nearly all businesses. SAP's application software foundation is built on the concepts of specialization and integration. ABAP is a programming language developed by SAP for programming commercial applications in the SAP environment. As a programming language, it has improved in recent years. Additions to ABAPs' database perception it has implemented more functional and concise syntax, powerful.

3.2 SAP System Overview

SAP ERP is an enterprise resource planning software developed by the German company [3, 4]. SAP is the largest enterprise applications provider and one of the largest software companies worldwide. Although SAP and its enterprise competitors are all distinctly different from one another, they are markedly similar as well [5]. Dissimilar to a large number of its rivals, SAP has for the most part become natural and have only a couple of critical acquisitions added to its repertoire.

SAP offers a wide range of ERP applications including Sales & Distribution, Materials Management, Production Planning, Logistics Execution, and Quality Management, Financial Accounting, Management Accounting, Financial Supply Chain Management, Payroll and e-Recruiting [6]. Apart from this CPSTL purchased SD, MM, FI and Payroll modules.

3.2.1 System Architecture

At the top is the Presentation server, which is any input device or the user interface that can be used to control ABAP based SAP framework such as SAP GUI, web browser or a mobile device. The Presentation layer communicates with the Application server is the 'brains' of the SAP system, where all the central processing takes place or the ABAP programs are executed. The Application server, in turn, communicates with the Database layer. The database layer consists of a database system which the central dataset of an ABAP-based SAP system is stored. The main role of ABAP programs in

the application layer is the processing and formatting of data from the database layer and its transfer to, and receipt of user input from the presentation layer [7].

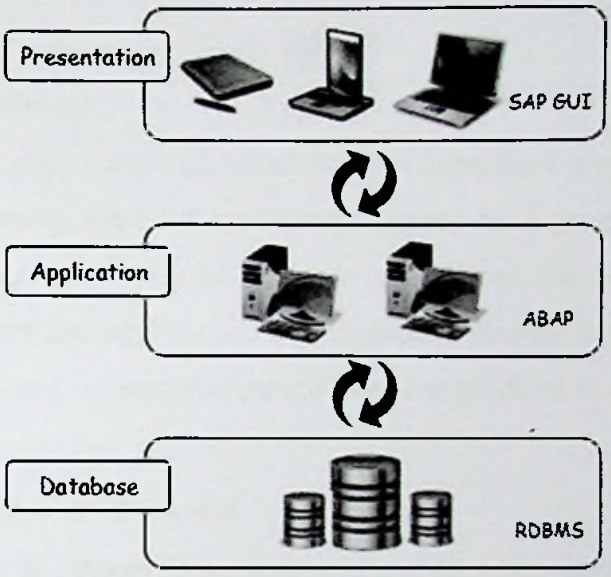


Figure 3.1: 3-Tire Client/Server Architecture

The database is kept on a separate server, essentially for performance reasons, additionally for security. Communication happens between each layer of the system, from the Presentation layer to Application server, to the Database [8].

3.2.2 Landscape Architecture

All the initial developments and the testing done on the Development system, which ensures other systems, are not affected. Once developments are at a stage where they may be ready to be tested by someone within the company whose role is to carry out testing or an external source, developments moved into the testing system [8].

Normally, no customizations at all are done on the testing system; it is just used for testing. If everything passes through the Testing system, move the customizations into the production environment. When code enters the Production environment, this is the stage at which it is turned on, and used within the business itself.

Landscape architecture separated not just for development purpose. The quantity of the data in the Production system may be too great to be used in the development

environment. Also, it could be due to security reasons. Companies do not want developers to see live production data, for data security reasons. Normally, then, the Development and Testing system have their own set of data to work with.

3.3 ABAP Programming

ABAP is high-level, object oriented, structured and imperative programming language design for programming the SAP application server. As a programming language, ABAP has an abstraction between the business applications, the operating system and database. This ensures that applications do not depend directly upon a specific server or database platform and can easily be ported from one platform to another.

3.3.1 ABAP Runtime environment

ABAP programs are not stored in separate external files like Java or C++ programs, whereas resides inside the SAP database. In the database all ABAP code exists in two forms: source code, which can be viewed and edited with the ABAP Workbench tools; and generated code, a binary representation somewhat comparable with Java bytecode.

ABAP programs execute under the control of the runtime system, which is part of the SAP kernel. The runtime system is responsible for processing ABAP statements, controlling the flow logic of screens and responding to events (such as a user clicking on a screen button); in this respect it can be seen as a Virtual Machine comparable with the Java VM. A key component of the ABAP runtime system is the Database Interface, which turns database-independent ABAP statements ("Open SQL") into statements understood by the underlying DBMS ("Native SQL"). The database interface handles all the communication with the relational database on behalf of ABAP programs; It also contains extra features such as buffering of tables and frequently accessed data in the local memory of the application server.

3.3.2 Types of ABAP Programming

As in other programming languages, an ABAP program is either an executable unit or a library, which provides reusable code to other programs and is not independently executable.

ABAP distinguishes two types of executable programs:

- Reports

  Reports, as the name would suggest, follow a relatively simple programming model whereby a user optionally enters a set of parameters and the program then uses the input parameters to produce a report in the form of an interactive list.

- Module pools

  Module pools define more complex patterns of user interaction using a collection of screens. The term "screen" refers to the actual, physical image that the user sees.

The non-executable program types are:

- INCLUDE modules

  An INCLUDE module gets included at generation time into the calling program. This is often used to subdivide large programs.

- Subroutine pools

  Subroutine pools contain ABAP subroutines (blocks of code enclosed by FORM/ENDFORM statements and invoked with PERFORM).

- Function groups

  Function groups are libraries of self-contained function modules (enclosed by FUNCTION/ENDFUNCTION and invoked with CALL FUNCTION).

- Object classes

  Object classes define a set of methods and attributes and which is similar to Java classes.

- Interfaces

Interfaces contain "empty" method definitions which is similar to Java interfaces. Any class implementing the interface must provide explicit code.

- Type pools

  Type pools define collections of data types and constants.

3.3.3 ABAP Special features

SMARTFORMS is used to generate formatted documents which can contain formatting objects such as logos, tables and so on. It looks like a pre-printed document.

ALV stands for **ABAP List Viewer,** per-defined report format in SAP. It unifies and simplifies the use of lists which have a better look and feel. ALV report consists of some pre-defined options like sort, filters, sum, downloading, printing, changing the layout structure and many more.

Apart from printing data, ABAP facilitates to download data in many formats such as **Excel** worksheet, **Text** file, **Word** file and **PDF** documents.

SAP provided with some plotting functions to represent data in **Graphs** and **Charts**. This can use to analyze and compare given set of data.

With the help of mobile partner, SAP provides short message service (**SMS**) to the external parties.

3.4 Why SAP for streamlining LBDs' stock planning process?

A few years before, the fuel distribution process has been automated using the SAP ERP system. All the bulk depots also operated based on SAP environment. Centralized database will be responsible for all the data. Due to security reasons higher management does not want to transfer internal data into external parties. Therefore cannot go for separate environment when streaming existing process.

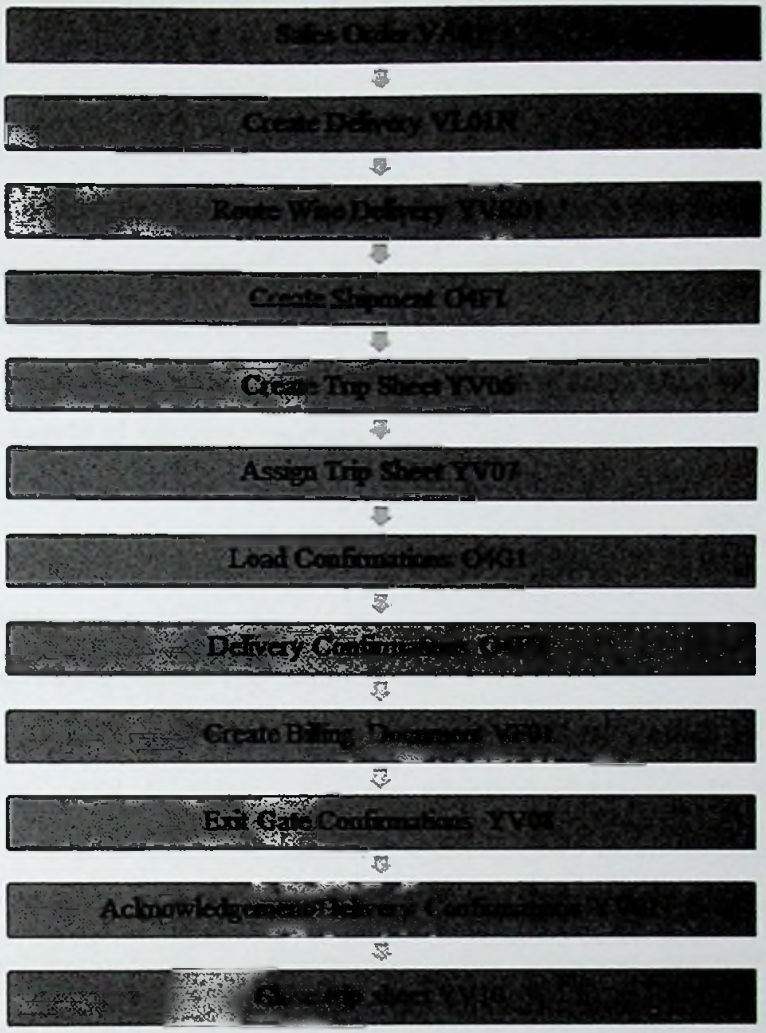


Figure 3.2: Bulk loading process running on SAP

Figure 3.2 shows the bulk loading process currently handling via SAP. The proposed solution will be affected to shipment creation. If there is no space available in tank system will restrict shipment creation. Unnecessary order creations will be restrict and therefore, as a result, it will improve LBDs' operational efficiency, reduce information delays and errors, satisfy the island wide demand, speedup stock transfers, improve customer service and improve overall productivity.

3.5 Summary

Each software component or application within the SAP family of products and services come across a particular need, facilitating day-to-day financial and resource management, addressing product lifecycle planning, supporting internal company procurement, interconnecting different systems to ease integration headaches, enabling customer relationship management, and so on.

Chapter 4 - Approach

4.1 Introduction

My approach is streamlining stock planning for the bulk depots by utilization of ERP system, with respect to the procedure followed currently. After studying the existing LBD stock planning process, identify the downsides, bottlenecks and drawbacks in the process. Thereafter design and develop the solution and evaluate the usage, adaptation, benefits, and impacts of proposed solution.

4.2 Proposed solution

4.2.1 Processes

- BTR deliveries (wagon and tank lorries).

  Enforcements have been taken to record BTR arrivals in LBDs against the BTR In-sliding using the ABAP module pool. Daily dip posting (which is already available in existing system) and vehicle status has to be done as a pre-posting. Once complete these two processes, update BTR In-Sliding (BTR slid in LBD premises) and then daily stock positions sent as a SAP mail from the LBD superintendent to distribution manager.

  Facilitate to save data into local files such as Excel worksheet, Text file or Word file and BTR In-Transit report can generate as a PDF file.

- Daily Quota Allocation.

  Prior to update the BTR in-transits and allocate daily quota, SAP background job, a simple ABAP report, will be run automatically at midnight and update the daily demand. This job will be run as a safe method. If any case LBD or distribution manager fails to update the BTR in-transits or daily quota, system will restrict sales order creations unless this background process runs.

When the distribution manager received stock position SAP mail from the LBD superintendent, distribution manager will allocate daily quota for depots, with respect to the space available in tanks. Quota allocation is a SAP dialog program and display data in a ALV report, which allocates maximum loads that can be sent to any depot from one or more oil installations Kolonnawa, Muthurajawela and bulk depots Anuradhapura and IRD Vavniya.

To exceed the allocated quota, need special authorization from higher management. "Authorize BTR with exceeding quota limit" is an ALV report with default ALV menu options, which authorized exceeded BTRs.

Permission has been given to re-allocate daily quota. This is again a blue print of daily quota allocation. If any case higher management wants to change the allocated quotations can use this re-allocation.

A separate ABAP panel is provided to update stock allocation percentage. This is also a combination of dialog program and ALV reporting. This program is used to update bowser, wagon, expected sales, quota and total allocation percentages which used in quota allocations.

Quota allocation effected to the shipment creation which via SAP standard transaction. Customization has done to modify the program with respect to the space availability. If there is no quota allocated or available to the plant/depot, the system will restrict shipment creation.

4.2.2 Features

- Conclusive Report on all aspects of stock planning for performance checking.

  Sales order book

  This is a simple ABAP report, same as hand written order book, but the only difference is all the depots are using the same methodology due to

automation. Order book consists todays sales and carry-forwards for one week, whereas the broad-forwards and same day orders will not consider.

Stock planning report for bulk depots

This is a summarized description of the LBDs stock planning process. This report contains tank details, expected and previous day sales details, stock In-Transits, stock In-Sliding and scheduled BTRs. Apart from LBD stock details, can view plant summary and island wide summary.

Bowser/Wagon delay report

Grace period has been defined for the transportation based on the distance from the source location. As an example, any bowser should not take more than 3 days to reach the Haputhale depot from Kolonnawa main installation and for wagon it is a day. Therefore, once place an order, the order should reach the depot within that grace.

The *'CGR Fuel Wagon Delay Letter'* prepared by the system automatically as a letter to the Additional General Manager, CGR by the Distribution Manager, CPSTL asking for explanation. Here I am using ABAP Smartforms to format the letter, same as the original written letter.

Bowser calling list

This is an ALV report, shows the overview on how the oil installation has managed to replenish the LBDs using bowsers of different capacities.

Performance indicators

The *'Key Performance Indicators'*, *'All locations performance'* and *'Plant Performance'* report options associates with different key performance indicators on which the performance of a plant can be

measured/gauged upon. This is an ALV list viewer with a list of menu options such as sorting, filtering, downloading as well as mailing.

Storage capacity utilization

A line graph generated (maximum for a month) to check the daily, monthly utilization of storage capacity with respect to the plants (depot) and the materials. This is mainly for managerial decision making purposes.

BTR release status

This is a simple ALV report to view BTR status, which has been sent to authorize exceeded quota limits. This is for LDB employees and the superintendent to check the status until update the exceeded quota by the higher management.

Monitor BTR status

This report is also a simple ALV report to monitor BTR status, such as release, receipts and discharge with release-receipts and release-discharge time differences. From this report LBD superintendent as well as the higher management can monitor the time taken to complete the delivery process.

Product order pattern

This is an ALV report to identify the order pattern of the depot or main installation. Selections are available for material, customer, plant, vehicle and the date.

Sales analysis report

This is an ole graph to identify the pat sales details. Daily analysis displayed in a line graph and monthly is a bar graph. Both graphs available excel download option.

- Value added services such as sending bulk SMS to the customers.

  Facilitate short message services using SAP dialog programs to update customers with new product released, promotions, news alerts.

- Availability of data for further analysis and managerial decision making.

  All necessary data will be updated to the SAP master and customized databases to use in the future attempt such as wagon scheduling, tanker operation and so on.

4.3 Summary

SAP offers a wide range of ERP applications and SD module is responsible when considering LBD operations. It facilitates as variety of reporting and dialog programing to streamline the stock planning process for the bulk depots.

The next chapter will be described how BTR delivery and daily quota allocation processes as well as mentioned features design in different modules.

Chapter 5 –Analysis and Design

5.1 Introduction

Systems analysis is a problem solving technique that decomposes a system into its component pieces for the end goal of the studying how well those component parts function and communicate to accomplish their purpose [11; 12]. Requirement analysis is the procedure of deciding user expectations for a new or modified product. Before analyzing the system, the requirements should be gathered by using the fact finding techniques, such as interviews, observations and documentations. These requirements must be quantifiable, relevant and detailed [13].

Software design is an iterative process through which requirements are interpreted into a blueprint for developing the product. At first, the plan delineates an all-encompassing perspective of software. That is, the design is represented at a high level of abstraction a level that can be specifically followed to the specific system objective and more detailed data, functional, and behavioral requirements [14]. The main goal of this design approach is defining objects and establishing relationship between classes by sending and receiving messages [15].

5.2 Requirement Gathering Techniques

Gathering of requirements, which is the most critical part in the analysis phase, should be properly accomplished using face gathering techniques such as observation, interview and document analysis.

5.2.1 Observing existing system

This is particularly useful when gathering data on current processes. May discover, for instance, that a few individuals have their work routine down to such a habit that they have a hard time explaining what they do [16].

The observation covers the investigation of users in its characteristic natural surroundings. By watching users such as LBD employees and bowser drivers and porters, a procedure stream, pain points, cumbersome steps and opportunities can be determined by an analyst for development. Observation can either be passive or active.

Passive observation gives better feedback to refine requirements on the same hand active perception works best to obtain a comprehension over a current business process. Analysts can utilize any of these ways to uncover the implicit requirements that are often overlooked in stock planning process [17]. Observing has played a significant role when learning island wide stock position and the bridging of fuel from the respective oil installation.

5.2.2 Interviews

Interviews of users are vital in making brilliant software. Without knowing the desires and objective of the users, are exceedingly unrealistic to satisfy them. Additionally need to comprehend the point of view of each interviewee, keeping in mind the end goal to legitimately address and measure their inputs. Like a decent reporter, listening is a quality that helps an amazing analyst to increase better esteem through a meeting when contrasted with an average analyst [17].

Face-to-face meetings are constantly best, on the grounds that they give a superior method for revealing the issue behind the issue. Once in a while, however, face-to-face meetings with users are not doable [18].

A number of interviews were conducted with the higher management, including DM, LBD superintendent and a few employees who are highly involved in stock planning process. Due to the lack of understanding of the system, difficulties arose only in interviewing rather than prior implementation and introducing the system to address their requirements.

5.2.3 Document Analysis

This is a vital information gathering method. Assessing the documentation of a present environment can help when making AS-IS process documents and furthermore, when driving the gap analysis for perusing of the migration projects.

In today's reality, will also be determining the requirements that drove making of an existing framework a starting point for reporting every present requirement. Chunks of data are mostly buried in present documents that help in putting inquiries as a piece of

validating the requirement completeness [17]. The existing manual process was fully paper base. Therefore, the appropriate documents were available for the analysis.

5.3 Object oriented design

Object oriented design is concerned with developing an object oriented model of a software system to implement the identified requirements. In OOD define, objects and building up a relationship between classes to solve a problem that was identified and documented during OOA [12]. Please refer *Appendix A* for more details in OOD.

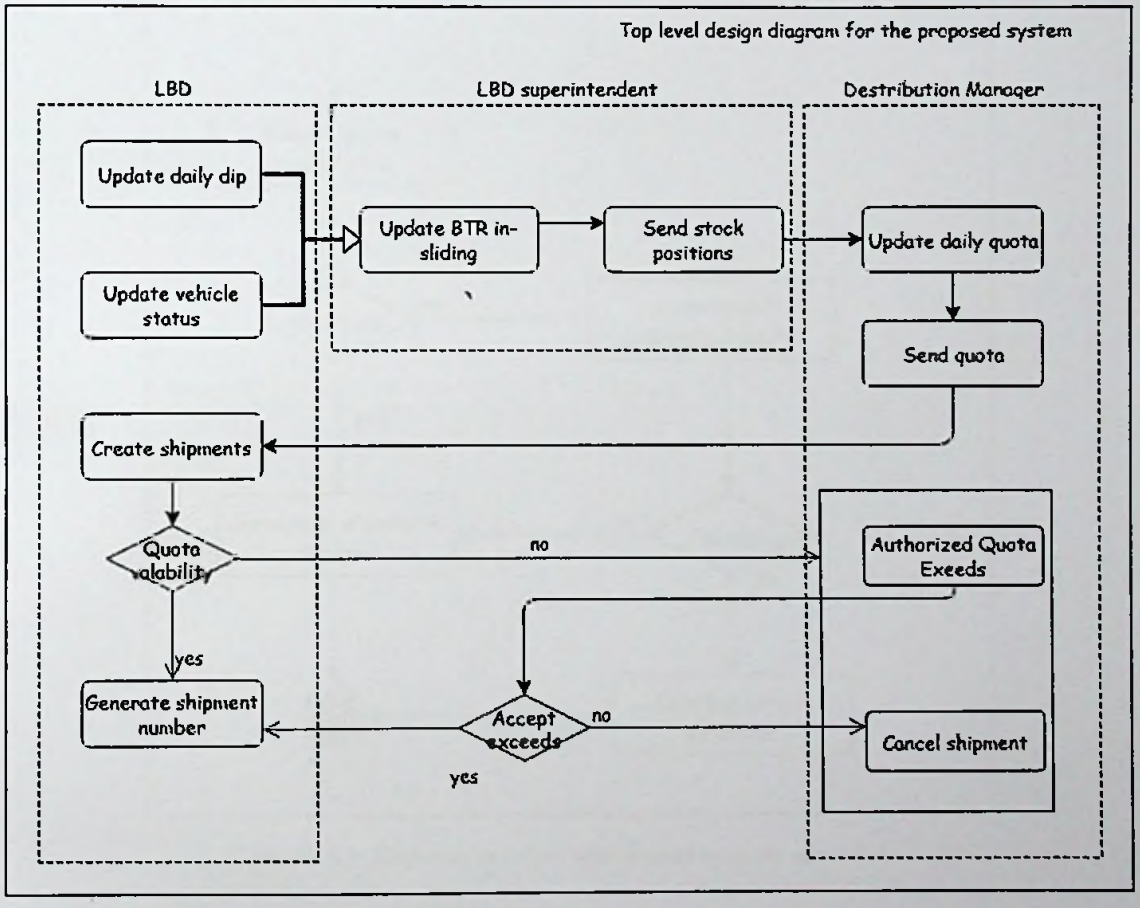


Figure 5.1: Top level design of stock planning process

Figure 5.1 displayed the top level design of the stock planning process, which explains the architecture or the overview of the entire system that would be used for developing. As can be seen here, LDB employee posts daily dip and vehicle status. Then the LBD superintendent updates BTR In-Sliding and sends the stock positions to distribution manager. DM updates the quota with as for the space availability of the tanks and send allocated quota back to LBD superintendent. Thereafter bulk depots will be

replenished with respective materials with respect to the quota availability. If there is no quota allocated or available to the plant/depot, the system will restrict shipment creation. All things considered DM can be authorized exceeding quota. If so LBD will be able to replenish further with respective materials.

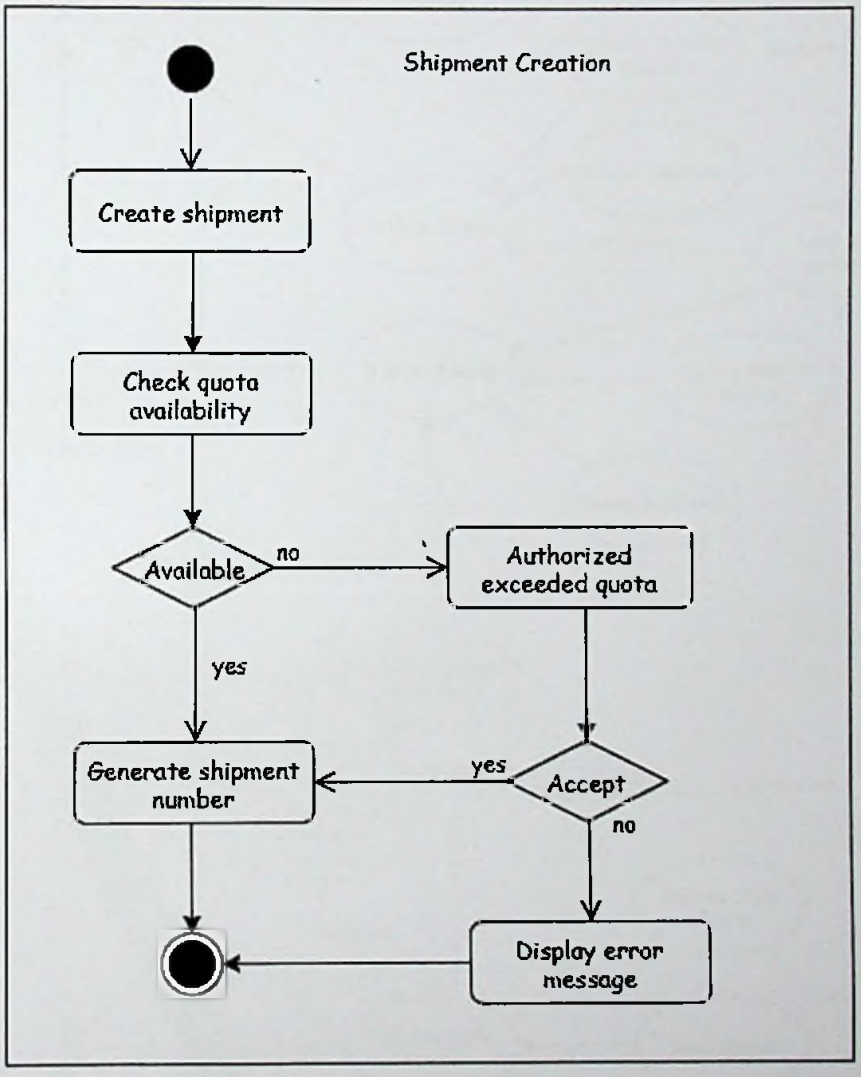


Figure 5.2: Shipment creation with respect to quota allocation

Figure 5.2 displays a state diagram for shipment creation (create bulk schedule). As can be seen above figure when creating shipments, the system will check the quota availability for the depot. If there is no quota available for the main installation to receiving depot, system will send the shipment details to the higher management authorization. Anyway, the system will not allow exceeding the quota limit unless authorized by higher management.

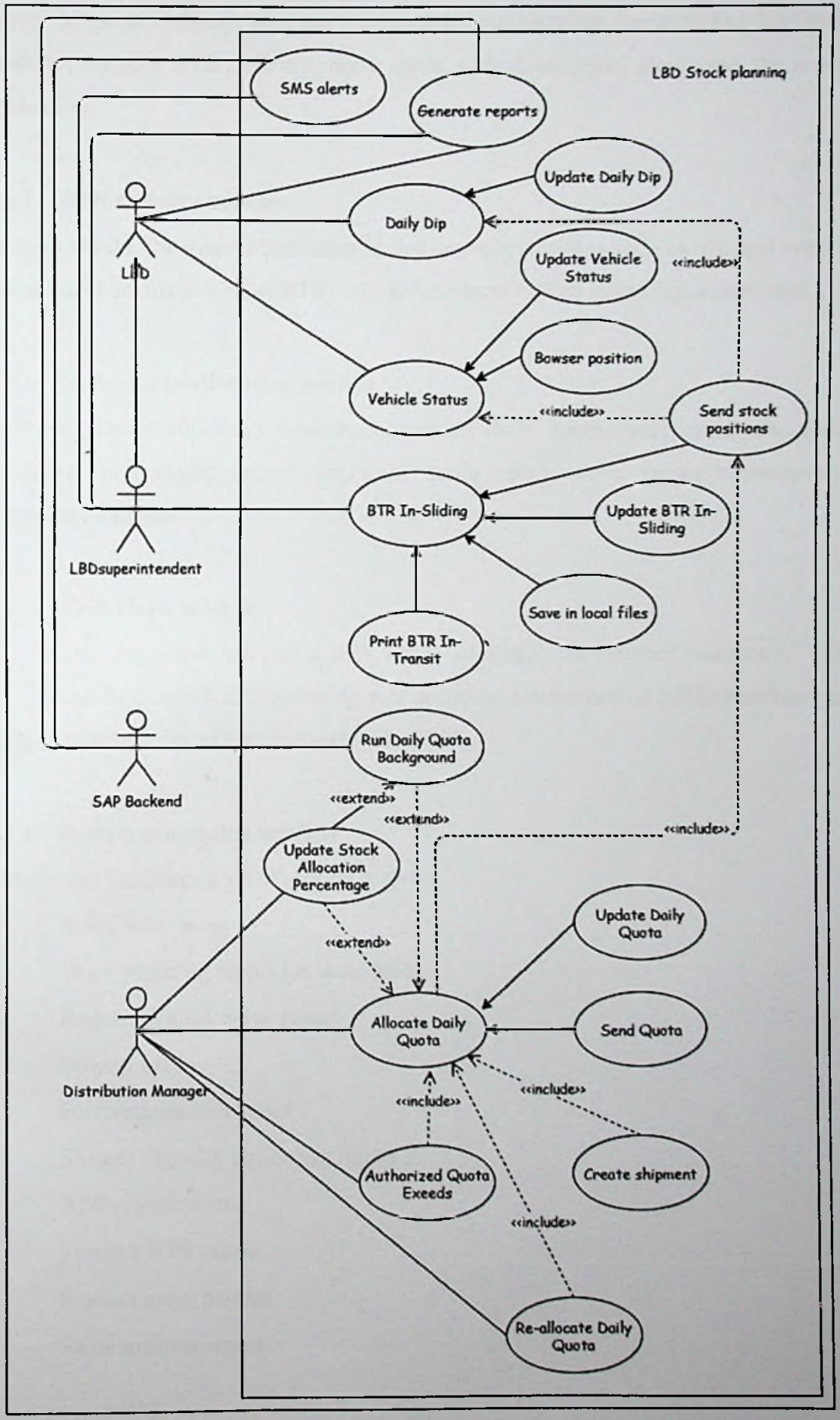


Figure 5.3: Class diagram for LDB stock planning

Figure 5.3 is an elaboration of the top level design. Here can be identified four main modules, namely BTR delivery, daily quota allocation, SMS alerts and the report generation.

5.3.1 BTR delivery module

In this module, the system facilitates to update daily postings such as dip and vehicle status, and tracking down the BTRs which have been parked in the depot premises.

5.3.2 Daily quota allocation module

The daily quota allocation module consists of daily quota background job, quota allocation percentage, quota allocation, quota re-allocation, quota exceeds and shipment creations.

5.3.3 SMS alerts module

The system facilitates sending a bulk set of messages for relevant customers. This facility can be used when introducing new products, promotions or public notifications such as holidays, out of services and many more.

5.3.4 Report generation module

The system facilitates a set of reports such as,

Sales order book

Stock planning report for bulk depots

Bowser/Wagon delay report

Bowser calling list

Performance indicators

Storage capacity utilization report

BTR release status

Monitor BTR status

Product order pattern

Sales analysis report

These reports are generated for authorized users according to their needs. This can be used to monitor the ongoing process and take managerial decisions when desirable.

5.4 Database design

Database design is the process of producing a detailed data model which describes how the data elements relate to one another of a database [19]. Well plan database improves performance, parallel data usage, data integrity, ease of use and reduce redundancy.

SAP uses embedded Oracle database management system. ABAP Dictionary (SE11) is a transaction where the developer can manage all ABAP dictionary objects (tables, views, data types, type groups, domains, search helps, lock objects). Data browser (SE16) is a place where the developer can manage and see the content of a database table. Generate table maintenance (SE54) uses to create and maintain the maintenance view of a table/view. Figure 5.4 describes the database design for the proposed system. Hence this is part of the existing process (customization), here I use SAP standard tales such as T001W (Plants/Branches), MARA (General Material Data), OIGV (TD Vehicle Header) and KNA1 (General Data in Customer Master).

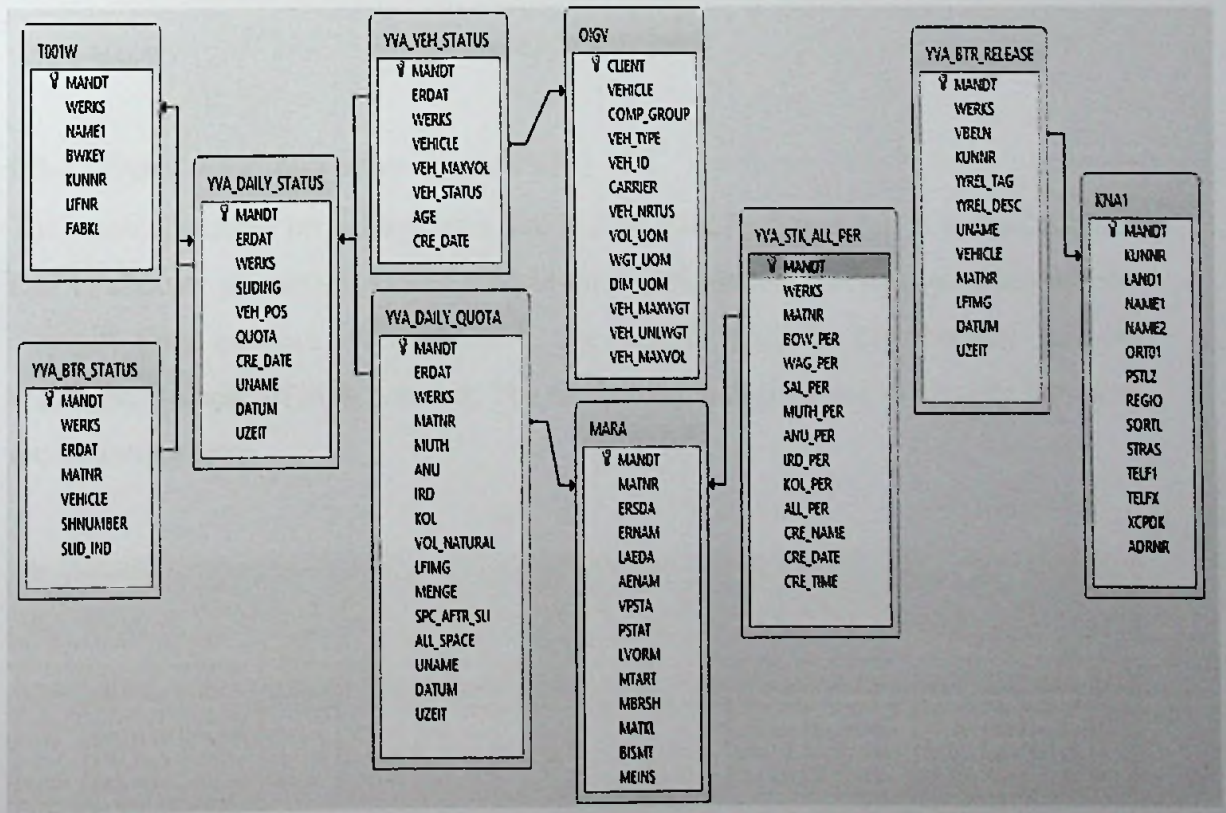


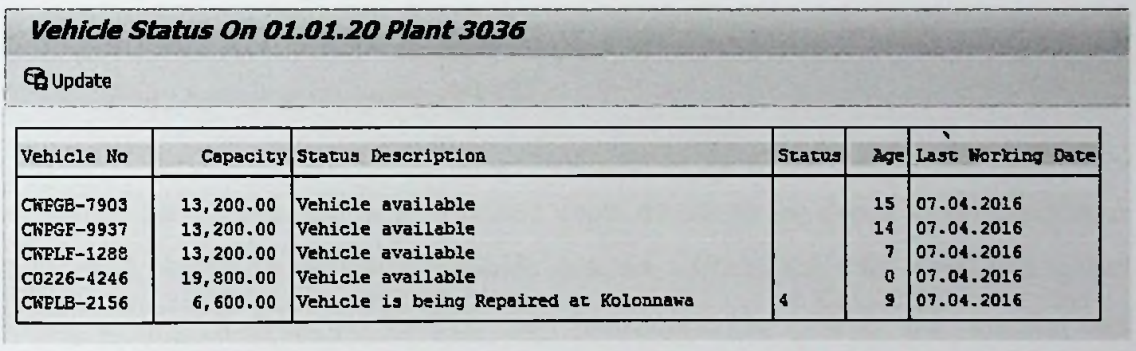
Figure 5.4: Database design

5.5 Interface design

The objective of user interface design is to make the user's interaction as simple and efficient as possible, as far as accomplishing user goals. Hence interactivity in between the system and the user is managed by the interface, it is necessary to have a user friendly design.

5.5.1 Vehicle status (YV28)

The vehicle status interface is displayed by figure 5.5, which allow users to update the temporary status of the vehicle. The vehicle can be available/not available, to be deleted, is being repaired at depot/kolonnawa or sold. Here the system will not allow updating vehicle status, if the stock position has been sent to the DM.

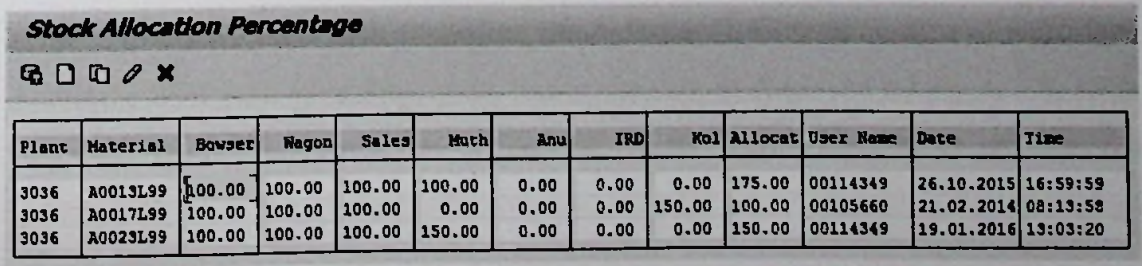


Vehicle No	Capacity	Status Description	Status	Age	Last Working Date
CWPGB-7903	13,200.00	Vehicle available		15	07.04.2016
CWPGF-9937	13,200.00	Vehicle available		14	07.04.2016
CWPLF-1288	13,200.00	Vehicle available		7	07.04.2016
CO226-4246	19,800.00	Vehicle available		0	07.04.2016
CWPLB-2156	6,600.00	Vehicle is being Repaired at Kolonnawa	4	9	07.04.2016

Figure 5.5: Vehicle status

5.5.2 Stock allocation percentage (YV30)

The stock allocation percentage interface is displayed by figure 5.7, which allows the DM to allocate percentage values used in quota calculation. These allocations will be changed time to time. Particularly in the regular periods, distributions will be expanded, because of high demand. The fields, date and time use to identify the most recent assignments.



Plant	Material	Bowser	Wagon	Sales	Muth	Anu	IRD	Kol	Allocat	User Name	Date	Time
3036	A0013L99	100.00	100.00	100.00	100.00	0.00	0.00	0.00	175.00	00114349	26.10.2015	16:59:59
3036	A0017L99	100.00	100.00	100.00	0.00	0.00	0.00	150.00	100.00	00105660	21.02.2014	08:13:58
3036	A0023L99	100.00	100.00	100.00	150.00	0.00	0.00	0.00	150.00	00114349	19.01.2016	13:03:20

Figure 5.6: Stock allocation percentage

5.5.3 BTR In-Transit (YV25)

The BTR In-Transit interface is displayed by figure 5.6, which allows the LBD superintendent to update the BTR status. After updating, the status can be sent as stock positions by clicking on the mail button on the menu bar. This screen facilitates selecting all rows at once, save data in a local file and print in-transits too.

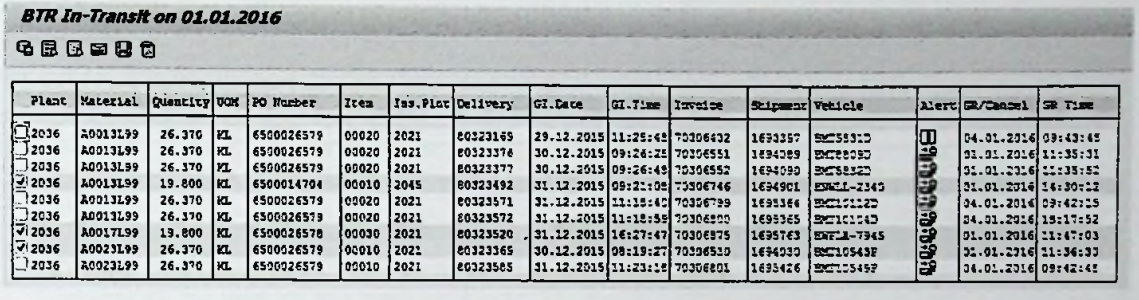


Figure 5.7: BTR In-Transit

5.5.4 Daily operational quota (YV26)

The daily operational quota interface is displayed by figure 5.8, which allows the DM to update daily quota and send allocated quota details to the depot superintendent as SAP mail. Based on the space available in tanks, identify the total quota and spread them among Kolonnawa (3021) and Muthurajawela (3045) oil installations. Kankasanthure BD will replenish with Anuradapura (3078) or IRD vavniya (3031) depots. Therefore quota allocations are available for those depots too. The system will not allow exceeding the space availability when allocating quota.

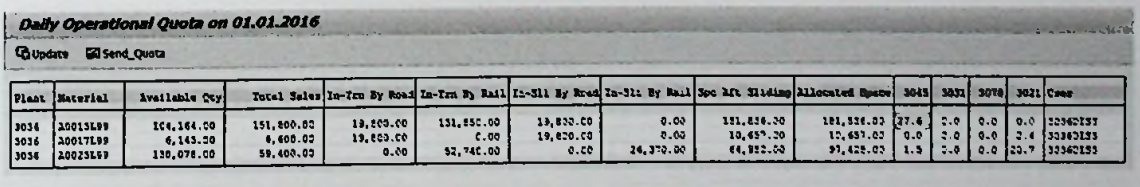
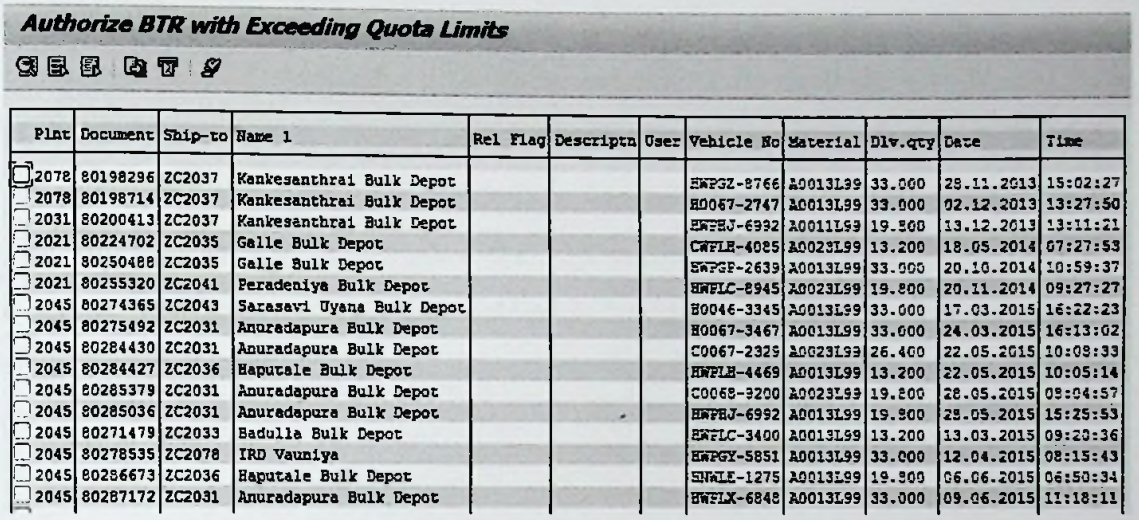


Figure 5.8: Daily operational quota

5.5.5 Authorize BTR with exceeding quota limits (YV29)

Authorize BTR with exceeding quota limits interface is displayed by figure 5.9. After allocating quota, shipments can be created. In the shipment creation process, the system will check the quota availability and will not allow exceeding the allocation.

Therefore need special authorization to approve the exceed quotas. Authorizations can be given by click on the confirming button on the menu.



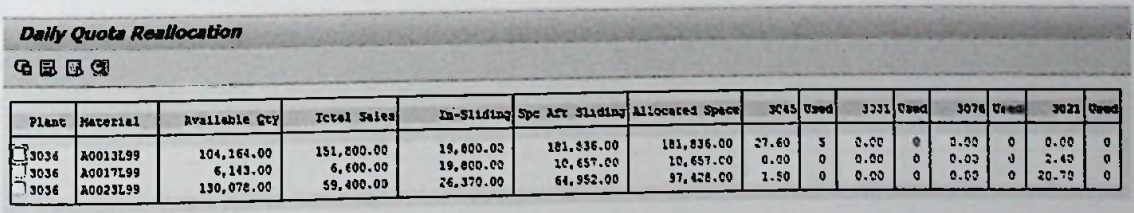
Authorize BTR with Exceeding Quota Limits

Plat	Document	Ship-to	Name 1	Rel Flag	Descriptn	User	Vehicle No	Material	Dlv.qty	Date	Time
2078	80198296	ZC2037	Kankesanthrai Bulk Depot				EWPGZ-8766	A0013L99	33.000	28.11.2013	15:02:27
2078	80198714	ZC2037	Kankesanthrai Bulk Depot				H0067-2747	A0013L99	33.000	02.12.2013	13:27:50
2031	80200413	ZC2037	Kankesanthrai Bulk Depot				EWPEJ-6992	A0011L99	19.800	13.12.2013	13:11:21
2021	80224702	ZC2035	Galle Bulk Depot				CWPLE-4085	A0023L99	13.200	18.05.2014	07:27:53
2021	80250488	ZC2035	Galle Bulk Depot				EWPGF-2639	A0013L99	33.000	20.10.2014	10:59:37
2021	80255320	ZC2041	Peradeniya Bulk Depot				HWPLC-8945	A0023L99	19.800	20.11.2014	09:27:27
2045	80274365	ZC2043	Sarasavi Uyana Bulk Depot				H0046-3345	A0013L99	33.000	17.03.2015	16:22:23
2045	80275492	ZC2031	Anuradapura Bulk Depot				H0067-3467	A0013L99	33.000	24.03.2015	16:13:02
2045	80284430	ZC2031	Anuradapura Bulk Depot				C0067-2329	A0023L99	26.400	22.05.2015	10:08:33
2045	80284427	ZC2036	Haputale Bulk Depot				HWPLE-4469	A0013L99	13.200	22.05.2015	10:05:14
2045	80285379	ZC2031	Anuradapura Bulk Depot				C0068-9200	A0023L99	19.800	28.05.2015	09:04:57
2045	80285036	ZC2031	Anuradapura Bulk Depot				HWPBJ-6992	A0013L99	19.800	28.05.2015	15:25:53
2045	80271479	ZC2033	Badulla Bulk Depot				EWPLC-3400	A0013L99	13.200	13.03.2015	09:20:36
2045	80278535	ZC2078	IRD Vauniya				HWPGY-5851	A0013L99	33.000	12.04.2015	08:15:43
2045	80286673	ZC2036	Haputale Bulk Depot				EWPLE-1275	A0013L99	19.800	06.06.2015	06:50:34
2045	80287172	ZC2031	Anuradapura Bulk Depot				HWPLX-6848	A0013L99	33.000	09.06.2015	11:18:11

Figure 5.9: Authorize BTR with exceeding quota limits

5.5.6 Daily quota re-allocation (YV31)

The daily quota re-allocation interface is displayed by figure 5.10, which allows the DM to re-allocate quota. Reallocation should be possible after the quota allocation. Once allocated quota, quota allocations will be updated in the master files and this will directly fetch data from master files. Like in quota allocation, re-allocation will not be sent to the depot superintendent. This interface is essentially designed to utilize the un-utilized quotas. The system will permit to lessen quotas up to utilized sums.



Daily Quota Reallocation

Plant	Material	Available Qty	Total Sales	In-Sliding	Spc Art Sliding	Allocated Space	3045	Used	3331	Used	3076	Used	3021	Used
3036	A0013L99	104,164.00	151,800.00	19,800.00	181,836.00	181,836.00	27.60	5	0.00	0	0.00	0	0.00	0
3036	A0017L99	6,143.00	6,600.00	19,800.00	10,657.00	0.00	0	0	0.00	0	0.00	0	2.40	0
3036	A0023L99	130,078.00	59,400.00	26,370.00	64,952.00	97,428.00	1.50	0	0.00	0	0.00	0	20.70	0

Figure 5.10: Daily quota re-allocation

5.6 Summary

Analysis underlines an examination of the issues and requirements, as opposed to an answer and design underlines a conceptual solution that fulfills the requirements, as opposed to its usage.

In this chapter I used object oriented analysis and design modeling techniques such as use case diagrams, state diagrams, data flow diagrams and so on to analyze, model, design the key modules such as LBD delivery module and daily quota allocation module. More details of analysis and design is in *'Appendix A – Design documentation'*, *'Appendix B – User documentation'* and *'Appendix C – Management reports'*.

The next chapter will be described the implementation details of BTR delivery and daily quota allocation module that is stated in the design diagram.

Chapter 6 - Implementation

6.1 Introduction

Implementation is the acknowledgment of an application, or execution of a plan, idea, model, design, outline, detail, standard, algorithm, or strategy. This chapter will be conveyed the major tasks involved in the stock planning process, the overall resources needed such as hardware, software and the flowcharts, algorithms, pseudo codes, code segments used in the design chapter.

6.2 Software and hardware requirement

Hence this research is a part of the current SAP system, there is no need of exceptional software or hardware to be installed. To keep the stream of the thesis underneath I briefly explains the existing.environment.

SD, MM, FI and HR modules of the SAP ERP system have being purchased, configured and utilized. The SAP uses an embedded Oracle database management system. The SAP servers are currently using SUSE Linux as the operating system. Windows Server 2003 is being used in the SAN (Storage Area Network).

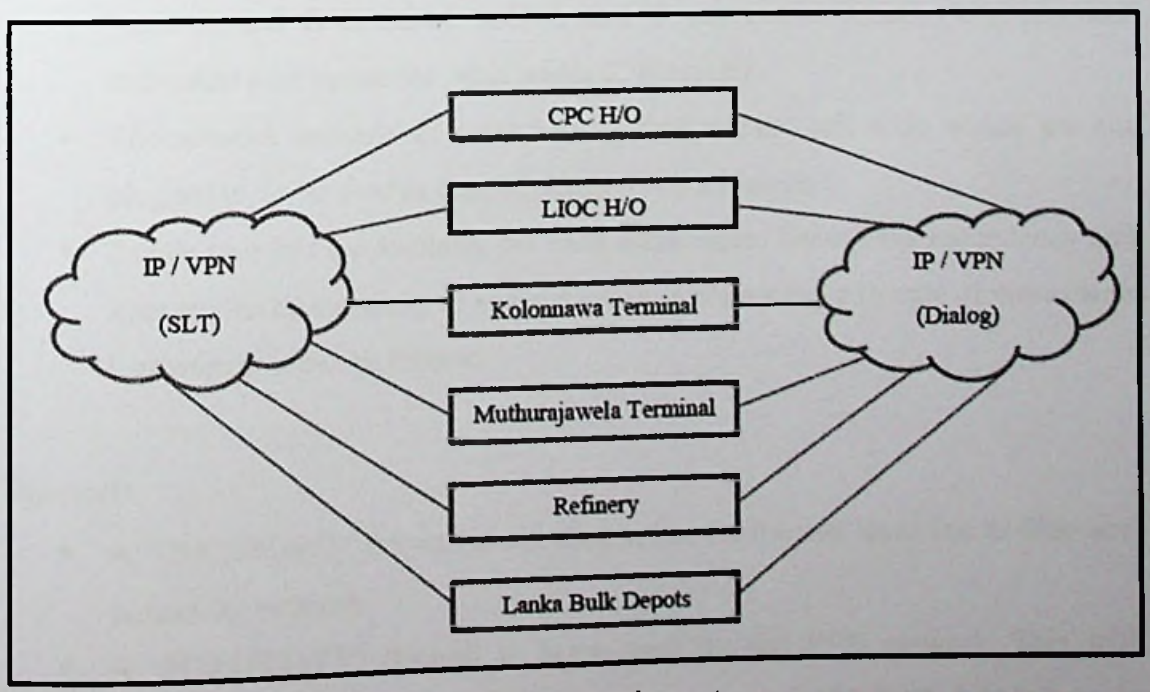


Figure 6.1: Network Infrastructure

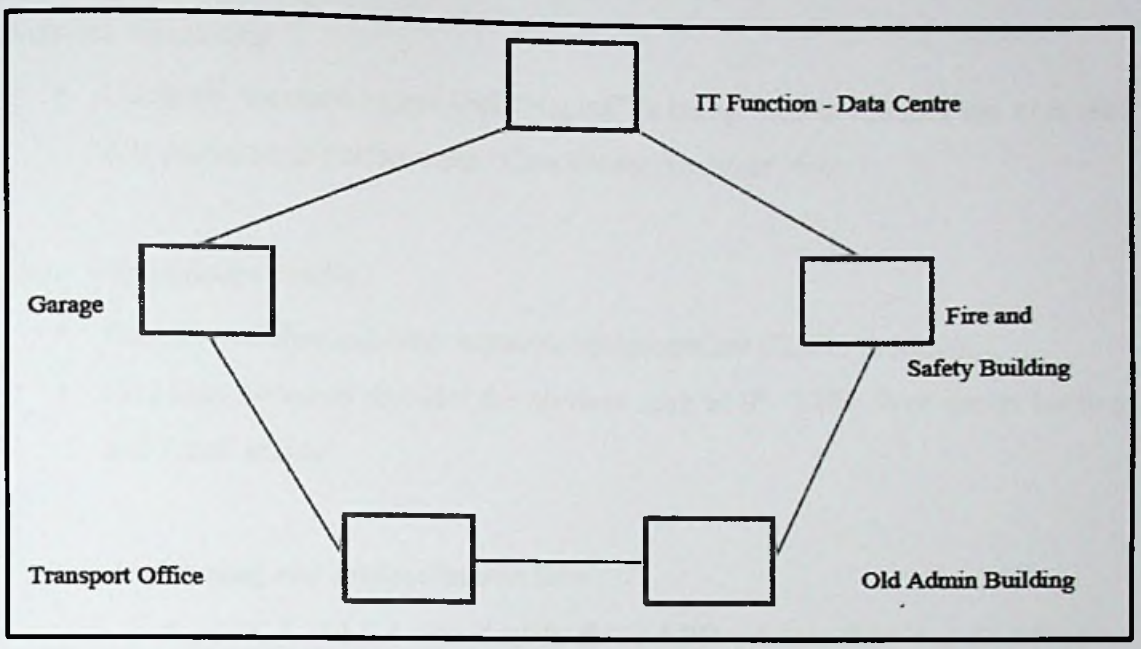


Figure 6.2: Fiber optic network

As can be seen in figure 6.2, Kolonnawa terminal fiber backbone consists of 5 major nodes. The network topology may physically seem like a ring network. But it is a logical star network. Local Area networks are being created function wise according to the requirement.

- This network has redundancy to make sure uninterrupted connectivity with all the nodes in case of a physical network failure. (i.e. If the connection between Node A and B is failed, still A and B can communicate via a physical redundant path across the other nodes C, D and E)
- The network consists of extra backup routers per each node which are not plugged in, to be used in case of physical router failure.
- 2 node switches are available per each major node. These have redundancy and inter switch connectivity to support versatile connectivity in case of unforeseen hardware or network failures.

Firewalls

- A "CHCKPOINT" firewall is installed before for internet lease line to filter and protect the network.
- A "FORTEGATE" firewall is being used for the VPN network. This will monitor and filter traffic being passed through the VPN, SAP, LAN, Servers and Internet.

Network monitoring

- Currently the open source tool "Nagios" is being used to monitor the network. It is proposed to purchase the "Checkpoint Analyzer" tool.

Other infrastructure details

- Routers switches and other network equipment are CISCO products.
- Sri Lanka Telecom provides the services such as IP / VPN, Web server hosting and Email server.

6.3 Stock planning and replenishment flow

As seen in figure 6.3 and 6.4 can identify flows LBD process flow and Distribution function process flow. Once complete the LBD flow stock positions will sent to DM and then, distribution function flow can be started.

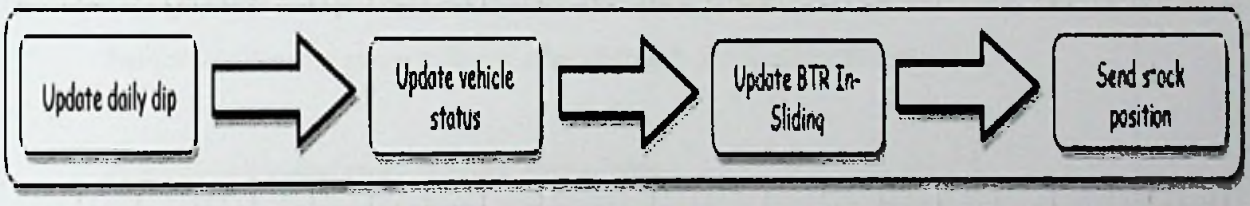


Figure 6.3: LBD process flow

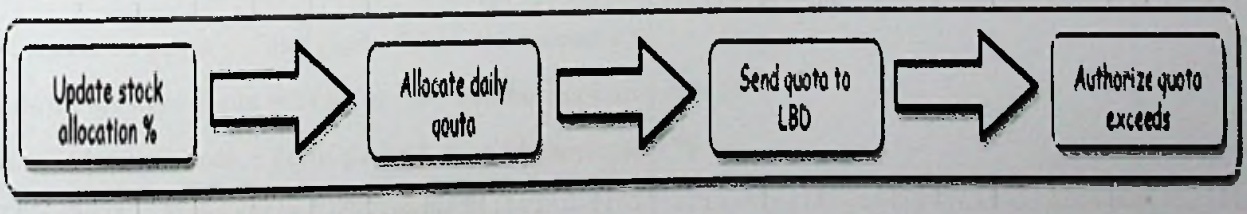


Figure 6.4: Distribution function process flow

6.4 Pseudo code

Here is the pseudo code used for daily quota calculation which used in quota allocation module.

```
Available Capacity = Tank Capacity – Available Quantity

IF Expected sales % > 0
        Total sales = Expected sales for the day * Expected sales % + Collections + Same day
```

orders (Previous day)

ELSE

Total sales = Expected sales for the day + Collections + Same day orders (Previous day)

ENDIF

IF *Stock in-sliding by road % > 0*

*BTR in-sliding = Stock in-sliding by rail + Stock in-sliding by road * Stock in-sliding by road %*

ELSE

BTR in-sliding = Stock in-sliding by rail + Stock in-sliding by road

ENDIF

Space after sliding = Available Capacity + Total sales – BTR in-sliding

IF *Available Quantity > Total sales* ***AND*** *Stock allocation % > 0*

*Available/Allocated space = Space after sliding * Stock allocation %*

ELSE

Available/Allocated space = Space after sliding

ENDIF

Total quota = Available/Allocated space / 6600

*Kolonnawa quota = Total quota * Kolonnawa quota %*

*Muthurajawela quota = Total quota * Muthurajawela quota %*

*Anuradapura quota = Total quota * Anuradapura quota %*

*IRD quota = Total quota * IRD quota %*

6.5 Code segments

The eye catching code segments developed in the system have been mentioned below by briefly describing their functionality. More details about the coding can be seen in the *'Appendix E – Code Listing'*.

6.5.1 Check authorization for the plant

Here is the code segment used to check the authorization for the selected plant/depot to update/view the report. 'WERKS' is the field name for the plant. Transaction code SU21 is used to maintain the authorization objects. Even though the developers used the authorization objects while coding, the SAP BASIS team is responsible for granting the access in the production environment. This code segment is common to almost every module hence it is must to check the authorization as a top priority.

```
      AUTHORITY-CHECK OBJECT 'Z:VKK_REPS'
                      ID 'SPART' DUMMY
                      ID 'VKBUR' DUMMY
                      ID 'VKORG' DUMMY
                      ID 'VSTEL' DUMMY
                      ID 'VTWEG' DUMMY
                      ID 'WERKS' FIELD i_werks-werks.

*--- Set a flag if authorization check fails for any value .
      IF sy-subrc <> 0 .
        MESSAGE e000(yv01) WITH 'Authorization failed for the '
                                ' Selection Criteria ' .
        EXIT.
      ENDIF .
```

6.5.2 ALV list display

This code segment is common to all modules. Most of the dialog programs and reports are based on ALV list display. Field categories and events are considered as subprograms.

```
*&---------------------------------------------------------------------*
*&      Form  PREVIW_DATA
*&---------------------------------------------------------------------*
*       Preview data
*----------------------------------------------------------------------*
FORM PREVIW_DATA .
  SORT i_output BY rel_date rel_time matnr oivbeln.
  i_layout-info_fieldname = 'COLOR'.
```

```
  PERFORM add_fieldcat.
  PERFORM add_events.

  CALL FUNCTION 'REUSE_ALV_LIST_DISPLAY'
    EXPORTING
      I_CALLBACK_PROGRAM       = SY-REPID
      I_CALLBACK_PF_STATUS_SET = 'S_BTR'
      I_CALLBACK_USER_COMMAND  = 'USER_COMMAND'
      IT_FIELDCAT              = I_FCAT
      IS_LAYOUT                = I_LAYOUT
      IT_EVENTS                = I_EVENTS
    TABLES
      T_OUTTAB                 = I_OUTPUT
    EXCEPTIONS
      PROGRAM_ERROR            = 1
      OTHERS                   = 2.
  IF SY-SUBRC <> 0.
  ENDIF.
ENDFORM.                    " PREVIW_DATA
```

The following is the code fragment used to create field list. Along these lines each time a designer needs to change the program if needs to include or alter new field. In that case can use 'REUSE_ALV_FIELDCATALOG_MERGE' inbuilt class to merge the field with existing table structure. In general both the methods are used to define field names.

```
*&---------------------------------------------------------------------*
*&      Form  ADD_FIELDCAT
*&---------------------------------------------------------------------*
*       Update ALV field list
*----------------------------------------------------------------------*
FORM ADD_FIELDCAT .
  DEFINE append_fcat.
    wa_fcat-fieldname = &1.
    wa_fcat-seltext_m = &2.
    wa_fcat-icon = &3.
    wa_fcat-outputlen = &4.
    APPEND wa_fcat TO i_fcat.
```

```
    CLEAR wa_fcat.
  END-OF-DEFINITION.

  CLEAR i_fcat.
  REFRESH i_fcat.
  append_fcat 'WERKS' 'Plant' '' 6.

ENDFORM.                    " ADD_FIELDCAT
```

```
*&      Form  build_fieldcat
*&---------------------------------------------------------------------*
*       -->CT_FIELDCAT  text
*----------------------------------------------------------------------*
FORM build_fieldcat TABLES   ct_fieldcat TYPE slis_t_fieldcat_alv.

  DATA: ls_fieldcat TYPE slis_fieldcat_alv.

  CALL FUNCTION 'REUSE_ALV_FIELDCATALOG_MERGE'
    EXPORTING
      i_program_name         = 'YVTRU29_MAINT_BTR_RELEASE'
      i_internal_tabname     = 'CT_SELECT'
      i_structure_name       = 'YVS_BTR_RELEASE'
      i_client_never_display = 'X'
    CHANGING
      ct_fieldcat            = ct_fieldcat[]
    EXCEPTIONS
      inconsistent_interface = 1
      program_error          = 2
      OTHERS                 = 3.
  IF sy-subrc NE 0.
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
            WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
  ENDIF.

ENDFORM.                    "build_fieldcat
```

The following code segment is used to create events used in ALV list. SAP allows adding headers and footers to ALV using field events.

```
*&---------------------------------------------------------------------*
*&       Form  ADD_EVENTS
*&---------------------------------------------------------------------*
*       Add events
*----------------------------------------------------------------------*
FORM ADD_EVENTS .
  CALL FUNCTION 'REUSE_ALV_EVENTS_GET'
    EXPORTING
      i_list_type = 0
    IMPORTING
      et_events    = i_events[].

  READ TABLE i_events WITH KEY name = slis_ev_top_of_page
                            INTO wa_event.
  IF sy-subrc = 0.
    MOVE 'TOP-OF-PAGE' TO wa_event-form.
    APPEND wa_event TO i_events.
  ENDIF.
```

6.5.3 PDF download via SAP

Y_PDF_DOWNLOAD is a reusable class, used to PDF download via SAP. This code segment is used when generating BTR In-Transit report in BTR delivery module.

Class consists of four class method SELECT_FILE, GET_PRINT_PARAMETERS, DOWNLOAD_PDF and ALV_DISPLAY. SELECT_FILE class method use to select a file or location which is going to save the PDF. Set printer parameters such as layout, line count, etc. in the method called GET_PRINT_PARAMETERS and download generated PDF file into the selected file using the method DOWNLOAD_PDF. And here is another method call ALV_DISPLAY which is used to list data in ALV and can use this file to directly download and generate a PDF.

Class interface: Y_PDF_DOWNLOAD

```
METHOD SELECT_FILE.
  CALL METHOD cl_gui_frontend_services=>file_save_dialog
    EXPORTING
      window_title      = 'Select a File'
      default_extension = 'pdf'
```

```
      default_file_name = g_default_file_name
      file_filter       = '*.pdf'
    CHANGING
      filename          = g_filename
      path              = g_path
      fullpath          = g_fullpath
      user_action       = g_user_action.
ENDMETHOD.
```

```
METHOD GET_PRINT_PARAMETERS.
  TYPES: t_pripar      TYPE pri_params,
         t_arcpar      TYPE arc_params.

  DATA: g_pripar     TYPE t_pripar,
        g_arcpar     TYPE t_arcpar,
        g_valid TYPE c.

 "Read, determine, change spool print parameters and archive paramete
rs
  CALL FUNCTION 'GET_PRINT_PARAMETERS'
    EXPORTING
      in_archive_parameters  = g_arcpar
      in_parameters          = g_pripar
      layout                 = g_layout
      line_count             = g_line_count
      line_size              = g_line_size
      no_dialog              = 'X'
    IMPORTING
      out_archive_parameters = g_arcpar
      out_parameters         = g_pripar
      valid                  = g_valid
    EXCEPTIONS
      archive_info_not_found = 1
      invalid_print_params   = 2
      invalid_archive_params = 3
      OTHERS                 = 4.
  IF g_valid <> space AND sy-subrc = 0.
    g_pripar-prrel = space.
    g_pripar-primm = space.
    NEW-PAGE PRINT ON
```

```
    NEW-SECTION
    PARAMETERS g_pripar
    ARCHIVE PARAMETERS g_arcpar
    NO DIALOG.
  ENDIF.
ENDMETHOD.
```

```
METHOD DOWNLOAD_PDF.
  NEW-PAGE PRINT OFF.
  CALL FUNCTION 'ABAP4_COMMIT_WORK'.

  DATA: g_bytecount TYPE i,
        g_spoolid TYPE tsp01-rqident,
        g_jobname TYPE tbtcjob-jobname,
        g_jobcount TYPE tbtcjob-jobcount,
        it_pdf TYPE STANDARD TABLE OF tline.

  IF g_src_spoolid IS INITIAL.
    g_src_spoolid = sy-spono.
  ENDIF.

  IF g_fullpath IS INITIAL.
    g_fullpath =  select_file( g_default_file_name ).
  ENDIF.

  IF g_user_action = 0 .
    "Convert spool to PDF
    CALL FUNCTION 'CONVERT_ABAPSPOOLJOB_2_PDF'
      EXPORTING
        src_spoolid   = g_src_spoolid
        no_dialog     = ' '
      IMPORTING
        pdf_bytecount = g_bytecount
        pdf_spoolid   = g_spoolid
        btc_jobname   = g_jobname
        btc_jobcount  = g_jobcount
      TABLES
        pdf           = it_pdf.

    "Download PDF file C Drive
```

```
    CALL FUNCTION 'GUI_DOWNLOAD'
      EXPORTING
        filename                = g_fullpath
        filetype                = 'BIN'
      TABLES
        data_tab                = it_pdf
      EXCEPTIONS
        FILE_WRITE_ERROR        = 1
        NO_BATCH                = 2
        GUI_REFUSE_FILETRANSFER = 3
        INVALID_TYPE            = 4
        NO_AUTHORITY            = 5
        UNKNOWN_ERROR           = 6
        HEADER_NOT_ALLOWED      = 7
        SEPARATOR_NOT_ALLOWED   = 8
        FILESIZE_NOT_ALLOWED    = 9
        HEADER_TOO_LONG         = 10
        DP_ERROR_CREATE         = 11
        DP_ERROR_SEND           = 12
        DP_ERROR_WRITE          = 13
        UNKNOWN_DP_ERROR        = 14
        ACCESS_DENIED           = 15
        DP_OUT_OF_MEMORY        = 16
        DISK_FULL               = 17
        DP_TIMEOUT              = 18
        FILE_NOT_FOUND          = 19
        DATAPROVIDER_EXCEPTION  = 20
        CONTROL_FLUSH_ERROR     = 21
        OTHERS                  = 22.
    IF SY-SUBRC = 0.
      MESSAGE s000(yv01) WITH 'PDF create succesfully'.
    ELSE.
      MESSAGE e000(yv01) WITH 'Error'.
    ENDIF.
  ELSE.
    MESSAGE e000(yv01) WITH 'Please select a valid file'.
  ENDIF.

ENDMETHOD
```

```
METHOD ALV_DISPLAY.
  DATA: it_print type slis_print_alv.

  it_print-print = 'x'.
  it_print-no_print_listinfos = 'x'.
  it_print-no_change_print_params = 'X'.
  sy-batch = 'X'.

  CALL FUNCTION 'REUSE_ALV_LIST_DISPLAY'
    EXPORTING
      I_CALLBACK_PROGRAM = I_REPID
      IT_FIELDCAT        = IT_FIELDCAT
      IT_SORT            = IT_SORT
      IS_LAYOUT          = IT_LAYOUT
      IT_EVENTS          = IT_EVENTS
      IS_PRINT           = IT_PRINT
    TABLES
      T_OUTTAB           = IT_OUTTAB
    EXCEPTIONS
      PROGRAM_ERROR      = 1
      OTHERS             = 2.
  IF SY-SUBRC <> 0.
  ENDIF.

ENDMETHOD.
```

6.5.4 Generating graphs

A SAP inbuilt function is used to generate ole graphs. Once passed the relevant data, the system will automatically create the graph. Below is the related code.

```
  CALL FUNCTION 'GFW_PRES_SHOW'
    EXPORTING
      container         = 'CONTAINER'
      presentation_type = gfw_prestype
      header            = v_header
      x_axis_title      = x_axis_title
      y_axis_title      = y_axis_title
    TABLES
      values            = values
```

```
      column_texts      = column_texts
    EXCEPTIONS
      error_occurred    = 1
      OTHERS            = 2.

  IF SY-SUBRC <> 0.
    LEAVE PROGRAM.
  ENDIF.
```

6.5.5 Downloading ole graph to excel file

Even though excel download is common in SAP programming; ole graph downloading is very new. Frist creates the excel worksheet, then pass the data and thereafter generate the graph. See the below code segment for more details.

```
* Start Excel
  IF v_excel-header = space OR v_excel-handle = -1.
    CREATE OBJECT v_excel 'EXCEL.APPLICATION'.
  ENDIF.

* Get list of workbooks, initially empty
  CALL METHOD OF v_excel 'Workbooks' = v_books.
  SET PROPERTY OF v_excel 'Visible' = 1.
  GET PROPERTY OF v_books 'Application' = v_app .
  SET PROPERTY OF v_app   'SheetsInNewWorkbook' = 1 .
  CALL METHOD OF v_books 'Add' = v_book.
  GET PROPERTY OF v_excel 'ACTIVESHEET' = v_worksheet.
  SET PROPERTY OF v_worksheet 'Name'  = v_sheet_name.

* Fill data
  CALL METHOD cl_gui_frontend_services=>clipboard_export
    IMPORTING
      data                 = i_excel_data[]
    CHANGING
      rc                   = v_rc
    EXCEPTIONS
      cntl_error           = 1
      error_no_gui         = 2
      not_supported_by_gui = 3
      OTHERS               = 4.
```

```
* Setting & formatting the title
  CALL METHOD OF v_excel 'Cells' = v_cell1
    EXPORTING
    #1 = 1
    #2 = 1.
  SET PROPERTY OF v_cell1 'Value' = 'Sales Analysis Report'.

  GET PROPERTY OF v_cell1 'Font' = v_font .
  SET PROPERTY OF v_font 'Underline' = 2 .
  SET PROPERTY OF v_font 'Bold' = 1 .

  CALL METHOD OF v_excel 'Cells' = v_cell1
    EXPORTING
    #1 = 3
    #2 = 1.

  IF p_dsum = 'X'.
    SET PROPERTY OF v_cell1 'Value' = 'Day'.
  ELSE.
    SET PROPERTY OF v_cell1 'Value' = 'Month'.
  ENDIF.

  GET PROPERTY OF v_cell1 'Font' = v_font .
  SET PROPERTY OF v_font 'Bold' = 1 .

  CALL METHOD OF v_excel 'Cells' = v_cell1
    EXPORTING
    #1 = 3
    #2 = 2.
  SET PROPERTY OF v_cell1 'Value' = 'Sales Qty'.

  GET PROPERTY OF v_cell1 'Font' = v_font .
  SET PROPERTY OF v_font 'Underline' = 2 .

* Paste excel
  CALL METHOD OF v_excel 'Cells' = v_cell1
    EXPORTING
    #1 = 4
    #2 = 1.
  CALL METHOD OF v_excel 'Cells' = v_cell2
```

```
    EXPORTING
    #1 = 4
    #2 = 2.
  CALL METHOD OF v_excel 'Range' = v_range
    EXPORTING
    #1 = v_cell1
    #2 = v_cell2.
  CALL METHOD OF v_range 'Select'.
  CALL METHOD OF v_worksheet 'Paste'.

*  Draw chart.
  CALL METHOD OF v_excel 'Cells' = v_cell1
    EXPORTING
    #1 = 3
    #2 = x.
  CALL METHOD OF v_excel 'Cells' = v_cell2
    EXPORTING
    #1 = m
    #2 = 5.
  CALL METHOD OF v_excel 'Range' = v_range
    EXPORTING
    #1 = v_cell1
    #2 = v_cell2.

  CALL METHOD OF v_range 'Select' .
  GET PROPERTY OF v_app 'Charts' = v_charts .
  CALL METHOD OF v_charts 'Add' = v_chart .
  CALL METHOD OF v_chart 'Activate' .
  SET PROPERTY OF v_chart 'ChartType' = v_charttype.
  CALL METHOD OF v_chart 'SetSourceData'
    EXPORTING
    #1 = v_range
    #2 = 2.

  SET PROPERTY OF v_chart 'HasTitle' = 1 .
  GET PROPERTY OF v_chart 'ChartTitle' = v_charttitle .
  GET PROPERTY OF v_charttitle 'Characters' = v_charttitlechar .
  SET PROPERTY OF v_charttitlechar 'Text' = v_header.

* Locate the chart onto the current worksheet &b activate current she
et
```

```
  CALL METHOD OF v_excel 'WorkSheets' = v_worksheet
    EXPORTING
    #1 = v_sheet_name.
  CALL METHOD OF v_worksheet 'Activate' .
  CALL METHOD OF v_chart 'Location'
    EXPORTING
    #1 = 2
    #2 = v_sheet_name.

* Free Excel objects
  FREE OBJECT: v_books,
               v_book,
               v_excel.
```

6.5.6 Sending mails

Sending mails are new to SAP programming. Stock positions in BTR module and daily quota allocations in quota allocation module are sent to the respective managers with attachment are using this code segment. Sending mail is possible to set of recipients with multiple attachments. PACKING_LIST contains the packing details such as body content type, attachment type, attachment name, etc. CONTENTS_BIN is the table use to add attachment content and CONTENTS_TXT consists of the email body message.

```
*&---------------------------------------------------------------------*
*&      Form  SEND_MAIL
*&---------------------------------------------------------------------*
*       Send mail      -     Function module to send mail to Recipients
*----------------------------------------------------------------------*
*  CALL FUNCTION 'SO_NEW_DOCUMENT_ATT_SEND_API1'
    EXPORTING
      DOCUMENT_DATA              = W_DOCUMENT_DATA
      PUT_IN_OUTBOX              = 'X'
      COMMIT_WORK                = 'X'
    IMPORTING
      SENT_TO_ALL                = G_SENT_TO_ALL
    TABLES
      PACKING_LIST               = I_PACKING_LIST
      CONTENTS_BIN               = I_ATTACHMENT
```

```
      CONTENTS_TXT                  = I_BODY_MSG
      RECEIVERS                     = I_RECEIVERS
    EXCEPTIONS
      TOO_MANY_RECEIVERS            = 1
      DOCUMENT_NOT_SENT             = 2
      DOCUMENT_TYPE_NOT_EXIST       = 3
      OPERATION_NO_AUTHORIZATION    = 4
      PARAMETER_ERROR               = 5
      X_ERROR                       = 6
      ENQUEUE_ERROR                 = 7
      OTHERS                        = 8.

  IF SY-SUBRC = 0 .
    MESSAGE s000(yv01) WITH 'Mail has been Successfully Sent.'.
  ELSE.
    MESSAGE e000(yv01) WITH 'Error: Cannot send Mail.'.
  ENDIF.
ENDFORM.                    " SEND_MAIL
```

6.5.7 Send SMS via SAP

Here is the reusable code segment which uses to send SMS via SAP in SMS alerts module. It consists of a class interface call Y_SEND_SMS. In the class interface there are three methods called CHECK_ACCESS which used to check the accessibility to send mails, VALIDATE_MOB_NO used to validate the mobile no which is going to send SMS and SEND_SMS. Usage of this class can be seen in *'Appendix E - Broadcast messages to customers'*.

Function Module: YV_SEND_SMS

```
FUNCTION YV_SEND_SMS.
*"----------------------------------------------------------------------
*"  IMPORTING
*"     VALUE(I_TYPE) TYPE  SO_OBJ_TP
*"     VALUE(I_SUBJECT) TYPE  SO_OBJ_DES
*"     VALUE(I_TEXT) TYPE  SOLI_TAB
*"     VALUE(I_NUMBER) TYPE  AD_PAGNMBR
*"     VALUE(I_LENGTH) TYPE  SO_OBJ_LEN OPTIONAL
*"----------------------------------------------------------------------
  DATA: c_send_sms TYPE REF TO y_send_sms.
```

```
  CALL METHOD c_send_sms->validate_mob_no
    EXPORTING
      i_number     = i_number
    RECEIVING
      i_new_number = i_number.

  IF i_number IS NOT INITIAL.
    CALL METHOD c_send_sms->send_sms
      EXPORTING
        i_type    = i_type
        i_subject = i_subject
        i_text    = i_text
        i_number  = i_number
        i_length  = i_length.
  ENDIF.

ENDFUNCTION.
```

Class Interface: Y_SEND_SMS

```
METHOD VALIDATE_MOB_NO.
*&---------------------------------------------------------------------
*&          DEVELOPER: Rukshani                DATE: 11.10.2014
*&       TRANSPORT NO: DEVK909241
*&        DESCRIPTION: Validate Mobile no before send the SMS
*&---------------------------------------------------------------------
  DATA: v_number TYPE AD_PAGNMBR.

  v_number = i_number.
  i_new_number = ''.

  IF v_number CS '-'.
    REPLACE '-' WITH '' INTO v_number.
  ENDIF.

  IF v_number CS '/'.
    REPLACE '/' WITH '' INTO v_number.
  ENDIF.

  CONDENSE v_number NO-GAPS.
```

```
  IF v_number+0(1) = '0'.
    v_number = v_number+1(9).
  ENDIF.

  IF strlen( v_number ) = 9.
    i_new_number = v_number.
  ENDIF.
ENDMETHOD.
```

```
METHOD SEND_SMS.
*&---------------------------------------------------------------------
*&          DEVELOPER: Rukshani              DATE: 11.10.2014
*&       TRANSPORT NO: DEVK908241
*&        DESCRIPTION: Send SMS to valid mobile no
*&---------------------------------------------------------------------
* Data definition
  DATA: lo_send_request TYPE REF TO cl_bcs,
        lo_document TYPE REF TO cl_document_bcs,
        lo_dockey TYPE soodk,
        lo_recipients TYPE bcsy_re,
        lo_bcs_exception TYPE REF TO cx_bcs,
        i_service TYPE ad_pagserv,
        i_recipient TYPE REF TO if_recipient_bcs,
        v_length TYPE SO_OBJ_LEN,
        wa_text LIKE LINE OF i_text.

  v_length = i_length.
  IF v_length IS INITIAL.
    LOOP AT i_text INTO wa_text.
      v_length = v_length + STRLEN( wa_text ).
    ENDLOOP.
  ENDIF.

  TRY.
*   handle request
      lo_send_request = cl_bcs=>create_persistent( ).

*   set 'send immediately'
      lo_send_request->set_send_immediately( '' ).
```

```
*   create document
    lo_document = cl_document_bcs=>create_document(
                  i_type        = i_type
                  i_text        = i_text
                  i_language    = sy-langu
                  i_subject     = i_subject
                  i_length      = v_length ).

*   add document to send request
    CALL METHOD lo_send_request->set_document( lo_document ).

*   document key
    lo_dockey-objtp = lo_document->get_doctp( ).
    lo_dockey-objyr = lo_document->get_docyr( ).
    lo_dockey-objno = lo_document->get_docno( ).

    TRY.
*     create SMS/pager recipient
      CALL METHOD cl_cam_address_bcs=>create_sms_address
        EXPORTING
          i_service = i_service
          i_number  = i_number
        RECEIVING
          result    = i_recipient.

      CATCH cx_address_bcs INTO lo_bcs_exception.
        result = lo_bcs_exception->get_text( ).
        EXIT.
    ENDTRY.

    TRY.
*     add recipient to send request
      CALL METHOD lo_send_request->add_recipient
        EXPORTING
          i_recipient = i_recipient.

      CATCH cx_send_req_bcs INTO lo_bcs_exception.
        result = lo_bcs_exception->get_text( ).
        EXIT.
    ENDTRY.
```

```
*   add recipient with its respective attributes to send request
      CALL METHOD lo_send_request->add_recipient
        EXPORTING
          i_recipient = i_recipient
          i_express   = 'X'.

*   check if document has recipients at all
      lo_recipients = lo_send_request->recipients( ).
      IF lo_recipients IS INITIAL.
        result = 'Send request has no receiver'.
        EXIT.
      ELSE.
*       sms has been sent
        CALL METHOD lo_send_request->send( ).
        COMMIT WORK.
        CONCATENATE 'SMS has been sent to' i_number INTO result.
      ENDIF.

    CATCH cx_bcs INTO lo_bcs_exception.
*   error messages
      result = lo_bcs_exception->get_text( ).
      EXIT.
  ENDTRY.

ENDMETHOD.
```

6.6 Summary

In this chapter I described about software/hardware requirements, stock planning work flow, pseudo codes and a set of code segments used to develop the proposed solution. Here I mentioned only selected code segments from each module. More details of coding is in *'Appendix E – Code listing'*.

Next chapter is the evolution, or in the general testing proposed system against objectives.

Chapter 7 - Evaluation

7.1 Introduction

After the implementation, next is the evaluation, making judgments on the findings. Proper testing needs to be carried out to ensure the item acts the way we need it to and to ensure the item is worked according to client prerequisites the system.

7.2 System test plan and test cases

Since the development began, the test plan proceeded by testing the system units. In the wake of performing the unit testing, next integration testing was done. This can distinguish the mistakes and the required usefulness of the units after integration. The final stage of the testing is system testing, which is done after the completion of the system to check the functionalities. This performed at the client's environment with client's original data.

Constructing test cases are the most critical part of the testing technique. Legitimately arranged test cases ought to be able to confirm the important system component functionality. In this manner, to confirm all the functionalities, there should be appropriately arranged test cases for every single capacity. Some test cases designed for major system modules are as follows: (Please refer Appendix D for all the test cases furthermore, comes about).

7.3 Test cases for BTR delivery module

7.3.1 Test cases for Vehicle status

Table 7.1 describes the test cases related to vehicle status update.

ID	Test case description	Test step	Expected result
1	Validate user inputs	Click 'Display/Update' button without creation date	Display an error message "Please Enter Valid Date"
2	Validate user inputs	Click 'Display/Update' button without plant	Display an error message "Please enter a valid plant"
3	Validate user inputs	Click 'Display/Update' button without valid plant	Display an error message "Please enter a valid plant"

4	Validate user inputs	Click 'Display/Update' button with correct creation date and the unauthorized plant	Display an error message "Authorization failed for the selection criteria"
5	Validate user inputs	Click 'Display/Update' button with valid creation date and the authorized plant	Successfully display the vehicle status details/report.
6	Validate keyboard enter press	Enter invalid status on 'status' field and press keyboard enter	Display an error message "Please enter valid status"
7	Validate keyboard enter press	Enter valid status on 'Status' field and press keyboard enter	Status will be updated will entered value and change grid color into green
8	Grid double click	Double click on 'Status Description/Status' field copy without position the cursor on the status	Display a warning message "Position the cursor on a line in the list"
9	Grid double click	Double click on 'Status Description/Status' field and select a status and cancel	No changes will be happening
10	Grid double click	Double click on any other field apart from 'Status Description/Status' field.	Nothing will happen
11	Grid double click	Double click on 'Status Description/Status' field and select a status and copy	Status will be updated will entered value and change grid color into green
12	Update status	Change vehicle status and click on 'Update'	Pop-up will be displayed "These changes cannot be reverted. Do you want to update vehicle status?"
13	Update status	Click 'Yes' on above message	Display a success message "Save changes successfully"
14	Update status	Try to update data after sending the stock positions	Display an error message "Data has been already sent for this plant"

Table 7.1: Test cases for vehicle status

7.3.2 Test cases for BTR In-Transit

Table 7.2 describes the test cases related to BTR In-Transit update.

ID	Test case description	Test step	Expected result
1	Validate user inputs	Click 'Display' button without creation date	Display an error message "Please Enter Valid Date"
2	Validate user inputs	Click 'Display' button without plant	Display an error message "Please enter a valid plant"
3	Validate user inputs	Click 'Display' button without valid plant	Display an error message "Please enter a valid plant"
4	Validate user inputs	Click 'Display' button with correct creation date and the unauthorized plant	Display an error message "Authorization failed for the selection criteria"
5	Validate user inputs	Click 'Display' button with valid creation date and the authorized plant	Successfully display the BTR In-Transit data.
6	Update BTR status	Click 'Update' without selecting any entry	Display an error message "Select at least one record to update."
7	Update BTR status	Select few entries and click on 'Update'	Pop-up will be displayed "Do you want to update BTR in-transit status?"
8	Update BTR status	Click 'Yes' on above message	Display a success message "Save changes successfully"
9	Update BTR status	Click 'Update' without selecting already updated entry/entries	Display an error message "Selected records were already updated."
10	Send stock positions	Try to send stock positions without updating	Display an error message "Please update data before sending stock positions."
11	Send stock positions	Update stock positions and try to send stock positions without dip posting	Display an error message "Please update data before sending stock positions."
12	Send stock positions	Update stock positions, dip and try to send stock positions without vehicle status update	Display an error message "Please update vehicle stats before sending stock

			positions."
13	Send stock positions	Update stock positions, dip and vehicle status. Now try to send stock positions	Display a success message "Mail has been successfully sent." And mail will be sent to respective persons with the stock position attachment.
14	Send stock positions	Try to resend the stock positions	Display an error message "Stock positions have been already sent."

Table 7.2: Test cases for BTR In-Transit

7.4 Test cases for Daily quota allocation module

7.4.1 Test cases for stock allocation percentage

Table 7.3 describes the test cases related to stock allocation percentage.

ID	Test case description	Test step	Expected result
1	Validate user inputs	Click 'Display Data' button without 'Plant' & 'Material'	The grid should display all the data without any filter
2	Validate user inputs	Click 'Display Data' button with valid 'Plant' & empty 'Material'	The grid should display only the data related to selected plant
3	Validate user inputs	Click 'Display Data' button with valid 'Plant' & 'Material'	The grid should display only the data related to the selected plant & material
4	Validate user inputs	Click 'Display Data' button without valid 'Plant'	Display an error message "Please enter valid plant"
5	Validate user inputs	Click 'Display Data' button with unauthorized 'Plant'	Display an error message "Authorization failed for the selection criteria"
6	Validate percentages	Enter 'Bowser sliding percentage' greater than 1000	Display an error message "Invalid Bowser Sliding Per centage"
7	Validate percentages	Enter 'Wagon sliding percentage' greater than 1000	Display an error message "Invalid Wagon Sliding Per

			centage"
8	Validate percentages	Enter 'Sales percentage' greater than 1000	Display an error message "Invalid Sales Percentage"
9	Validate percentages	Enter 'Muthuragawela quota percentage' greater than 1000	Display an error message "Invalid Muthurajawela Quota Percentage"
10	Validate percentages	Enter 'Anuradhapura quota percentage' greater than 1000	Display an error message "Invalid Anuradhapura Quota Percentage"
11	Validate percentages	Enter 'IRD quota percentage' greater than 1000	Display an error message "Invalid IRD Quota Percentage"
12	Validate percentages	Enter 'Kolonnawa quota percentage' greater than 1000	Display an error message "Invalid Kolonnawa Quota Percentage"
13	Validate percentages	Enter 'Allocation percentage' greater than 1000	Display an error message "Invalid Allcation Percentage"
14	Update data	Click 'Update' button after making any changes	Display a success message "Save changes successfully"
15	Create/Copy	Try to create/copy with empty plant	Display an error message "Plant cannot be empty"
16	Create/Copy	Try to create/copy with invalid plant	Display an error message "Please enter valid Plant"
17	Create/Copy	Try to create/copy with empty material	Display an error message "Material cannot be empty"
18	Create/Copy	Try to create/copy with invalid material	Display an error message "Please enter valid Material"
19	Create/Copy	Try to create/copy existing entry	Display an error message "selected combination already exist"
20	Create/Copy	Try to create/copy valid entry	The grid should display newly added data
21	Edit	Click on 'Edit' button and repeat	Results should be same as

		the test cases 6 to 13	test cases 6 to 13
22	Delete	Select an entry and click 'Delete'	Deleted entry should be removed from the grid

Table 7.3: Test cases for stock allocation percentage

7.4.2 Test cases for daily operational quota

Table 7.4 describes the test cases related to daily operations quota.

ID	Test case description	Test step	Expected result
1	Validate user inputs	Click 'Display' button without operational date	Display an error message "Please Enter Valid Date"
2	Validate user inputs	Click 'Display' button without plant	Display an error message "Please enter a valid plant"
3	Validate user inputs	Click 'Display' button with valid operational date and the authorized plant	Successfully display the daily operational quota details.
4	Update quota	Try to update quota with negative values	Display an error message "Quota cannot be negative"
5	Update quota	Try to update quota which exceeds the total amount	Display an error message "Total quantity cannot be > available space"
6	Update quota	Tyr to update quota before updating BTR sliding	Display an error message "Please update BTR sliding before allocate quota"
7	Update quota	Update quota with valid values and click 'update' button	Display a success message "Save changes successfully"
8	Send quota	After updating quota, send quota	Display a success message "Mail has been Successfully Sent"
9	Send quota	Try to resend quota again	Display an error message "Daily quota has been already sent"
10	Edit quota	Try to edit already updated data	Display an error message "Cannot edit already updated data"

Table 7.4: Test cases for daily operations quota

7.5 Summary

Software testing has to be done during the development, after integration as well as after completing the full process. Testing is a verification and validation process.

Verification is the process to ensure the item fulfills the conditions forced toward the begin on the improvement stage. Validation is the process to ensure the item fulfills the predetermined prerequisites toward the end of the improvement stage.

Unit testing, integration testing and system testing carried out at different stages to justify the functionalities. Once the product moves into the client's hand, it should work as the customer expects.

Chapter 8 – Conclusion and Future Works

8.1 Introduction

SAP is the largest enterprise applications provider and one of the largest software companies worldwide. SAP offers a wide range of ERP applications including Sales & Distribution, Materials Management and Finance management and so on.

A business process is an activity or set of activities that will accomplish a specific organizational goal. When managing a business process IT is an essential partner and its applications can manage business with greater accuracy and efficiency. Adoption of IT resources allows companies to maintain a competitive advantage over their rivals.

CPSTL also carried out their business process with more prominent precision and effectiveness. Thusly SAP is a vital accomplice. Implementing SAP ERP System is a big hurdle that CPSLT passed a few years back. Now it is great opportunity to round out the crevices with existing framework and utilizations.

8.2 Conclusion

In fuel distribution apart from main installations LBDs' acts a major role and streamlining of LBDs' stock planning processes is another milestone that CPSTL carryover. Un-utilized stock planning leads to various issues such as, delays in product delivery, over/under scheduling, lack of monitoring/ control mechanisms, damage of company prestige, creates windows for various malpractices/congestion, unnecessary costs incurred, improper utilization of resources is experiencing currently.

Stock planning carried out to overcome these issues, while considering the storage capacities, available space, expected sales for the day various other factors with respect to the depot. Then bulk depots replenished with respective materials while satisfying the demand without over scheduling.

Stock planning for the bulk depots enhanced by the utilization of ERP system concerning the system took over as of now. After the implementations tanks ended up

as for accessibility and requested as opposed to outsider inclusions. Ultimately, improved operational efficiency, reduced information delays and errors, satisfied the island wide demand, speedup stock transfers, improved customer service and improved overall productivity. The data have been recorded and maintained and now available for further analysis and managerial decision making.

8.3 Future works

Here only I focused on LBD stock planning process only. Scheduling also needs to be optimized by utilizing SAP ERP system.

Provision needs to be provided to track and monitor the transportation mode (bowsers and wagons) receipts and dispatches. The optimal utilization of the available modes will mean the utilization to maximal usage of transportation and minimal delay. A scheduling process needs to be carried out using a more scientific way with respect to the manual system which is being followed today. Since, wagons are the cheapest mode of transport; this will result in significant reduction of fuel transportation costs.

8.4 Summary

Petroleum has been used since ancient times, and is presently essential crosswise over society. As the world keeps on industrializing quickly, demand is increasing day by day. At the same time unit cost of petroleum products also increasing very frequently.

As of now LBDs' stock planning has been used utilizing the SAP ERP framework. The following step is scheduling transportation modes using a more scientific way with respect to the manual system which is being followed today. From that point, I trust the progressing fuel dissemination procedure might run easy way.

References

[1] http://www.cpstl.lk.html

[2] http://www.ceypetco.gov.lk/index.html

[3] https://en.wikipedia.org/wiki/Enterprise_resource_planning

[4] https://en.wikipedia.org/wiki/SAP_ERP

[5] Anderson, G., W. (2011), *"SamsTeachYourself SAP in 24 Hours"*, pp 7 – 10, 800 East 96th Street, Indianapolis, Indiana, 46240 USA

[6] Boeder, J., Groene, B. (2014), *"The Architecture of SAP ERP"*

[7] http://help.sap.com/abapdocu_70/en/ABENABAP_OVERVIEW.htm

[8] Moxon, P. (2012), *Beginners Guide to SAP ABAP*

[9] http://scn.sap.com/docs/DOC-58032

[10] http://scn.sap.com/docs/DOC-57910

[11] Bentley, L., D., *"Systems Analysis and Design for the Global Enterprise"*, pp160, 7th edition.

[12] Whitten, J., L., Bentley, L., D., Dittman, K., C. (2004), *"Systems analysis and design methods"*, pp 162, 5th edition.

[13] Sommerville, I. (2007), *"Software Engineering"*, 8th edition, Addison-Wesley.

[14] Roger, S., P. (2005), *"Software Engineering"*, pp 226, 6th edition.

[15] Whitten, J., L., Bentley, L., D. (2007), *"Systems Analysis and Design methods"*, 7th edition.

[16] http://www.techrepublic.com/blog/10-things/10-techniques-for-gathering-requirements

[17] http://www.brighthubpm.com/project-planning/60264-techniques-used-in-business-requirements-gathering

[18] http://epf.eclipse.org/wikis/openup/core.tech.common.extend_supp/guidances/guidelines/req_gathering_techniques_8CB8E44C.html

[19] https://en.wikipedia.org/wiki/Database_design

Use case diagrams with details

Here is the use case diagram in detail shown in Chapter 5. To understand it easily and clearly, it was categorized into two modules and each module explained with use case descriptions. Each module is comprised of several management reports which described in Appendix C.

LBD stock planning DFD

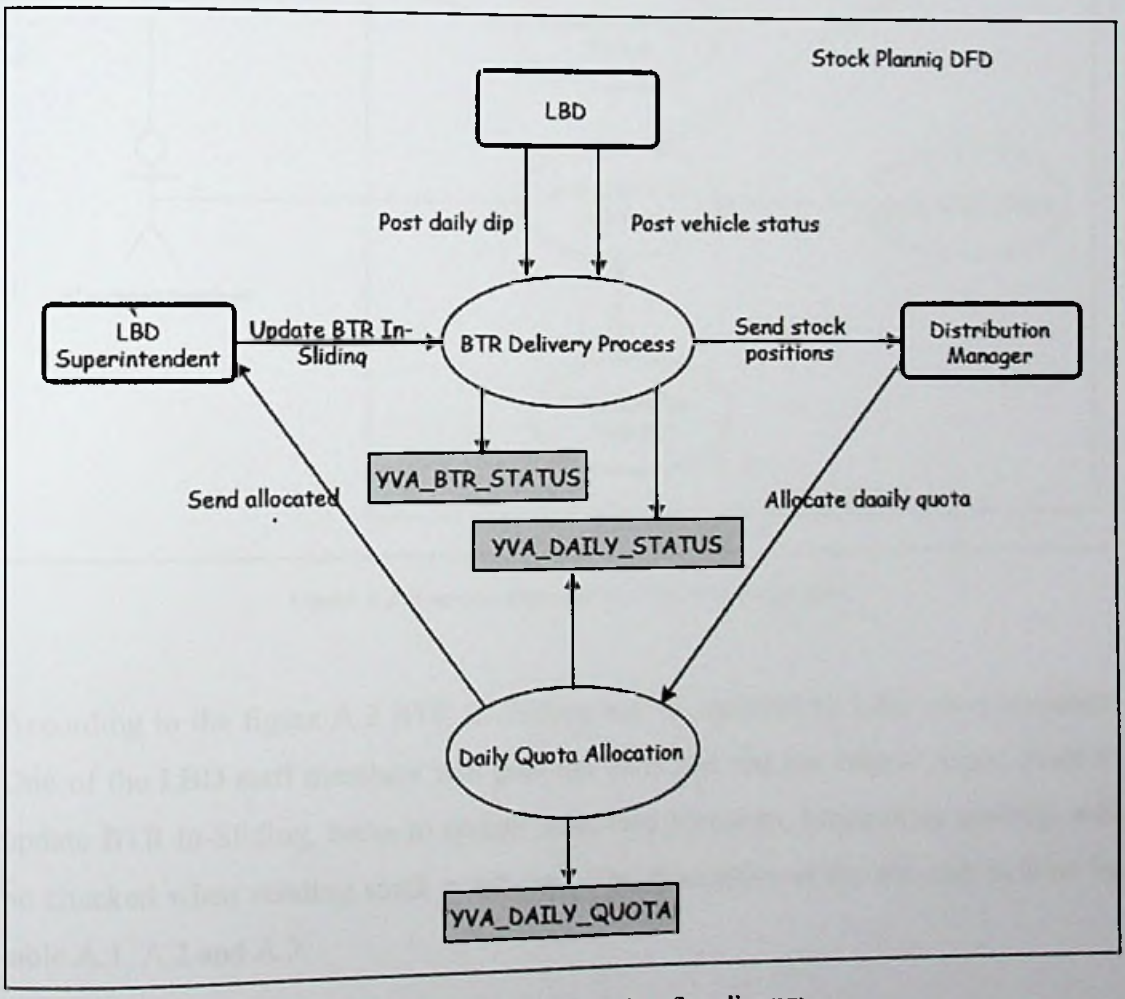


Figure A.1: Stock planning data flow diagram

Figure A.1 displays stock planning data flow diagram in brief. LDB employees post the daily dip and vehicle status. Then the LBD superintendent updates BTR In-Sliding and send the stock positions to distribution manager. DM updates the quota with respect to the space availability of the tanks and send allocated quota back to LBD

superintendent. Thereafter bulk depots will be replenished with respective materials with respect to the quota availability.

BTR delivery module

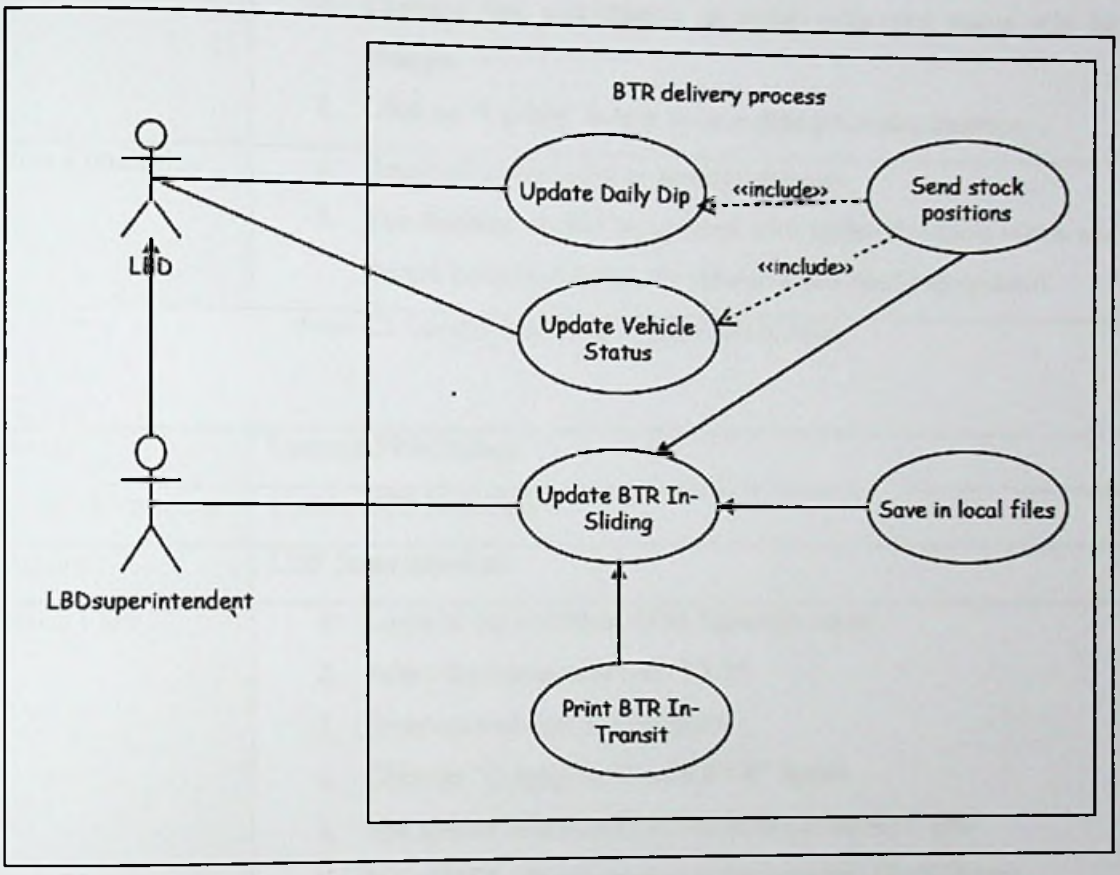


Figure A.2: Use case diagram for BTR delivery process

According to the figure A.2 BTR In-Sliding will be updated by LBD superintendent. One of the LBD staff members will post the daily dip and the vehicle status. Prior to update BTR In-Sliding, better to update these two processes, hence these postings will be checked when sending stock positions. The description of the use case is done by table A.1, A.2 and A.3.

Name	UpdateVehicleStatus
Brief Description	Update LBD vehicles' current status
Actors	LDB Employee
Basic Flow	1. Login to the system as LDB Employee 2. Select the transaction code YV28 3. Enter created date and the plant

	4. Click on "Update" button 5. The system will display all the vehicles belongs to the selected plant in an ALV grid 6. Click on "Status" tab and select the current status 7. Changed line will display in green color and status will be changed 8. Click on "Update" button to save changes to the database.
Post Conditions	1. Received a successful or an error message 2. The database should be updated with updated vehicle status and should be indicated that the vehicle status has been updated

Table A.1: Use case description - Update vehicle status

Name	UpdateBTRInSliding
Brief Description	Update BTR In-Sliding
Actors	LDB Superintendent
Basic Flow	1. Login to the system as LDB Superintendent 2. Select the transaction code YV25 3. Enter created date and the plant 4. Click on "Display In-Transit BTR" button 5. The system will display all the BTRs in an ALV grid 6. Select BTRs already parked at the premises (In-Slidings) 7. Update database by clicking on "Update BTR Ststus" button
Post Conditions	1. Received a successful or an error message 2. Updated records will be displayed in green color and the error records in red color 3. The database should be updated with the sided BTRs

Table A.2: Use case description - Update BTR In-Sliding

Name	SendStockPosition
Brief Description	Send/Mail stock positions to Distribution Manager
Actors	LDB Superintendent
Precondition	1. Update daily dip 2. Update vehicle status
Basic Flow	1. Run the use case UpdateBTRInSliding

	2. Click on "Send Mail" button to send stock positions to DM
Post Conditions	1. Received a successful or an error message 2. The database should be updated to indicate that the stock positions have been sent

Table A.3: Use case description - Send stock positions

Daily quota allocation module

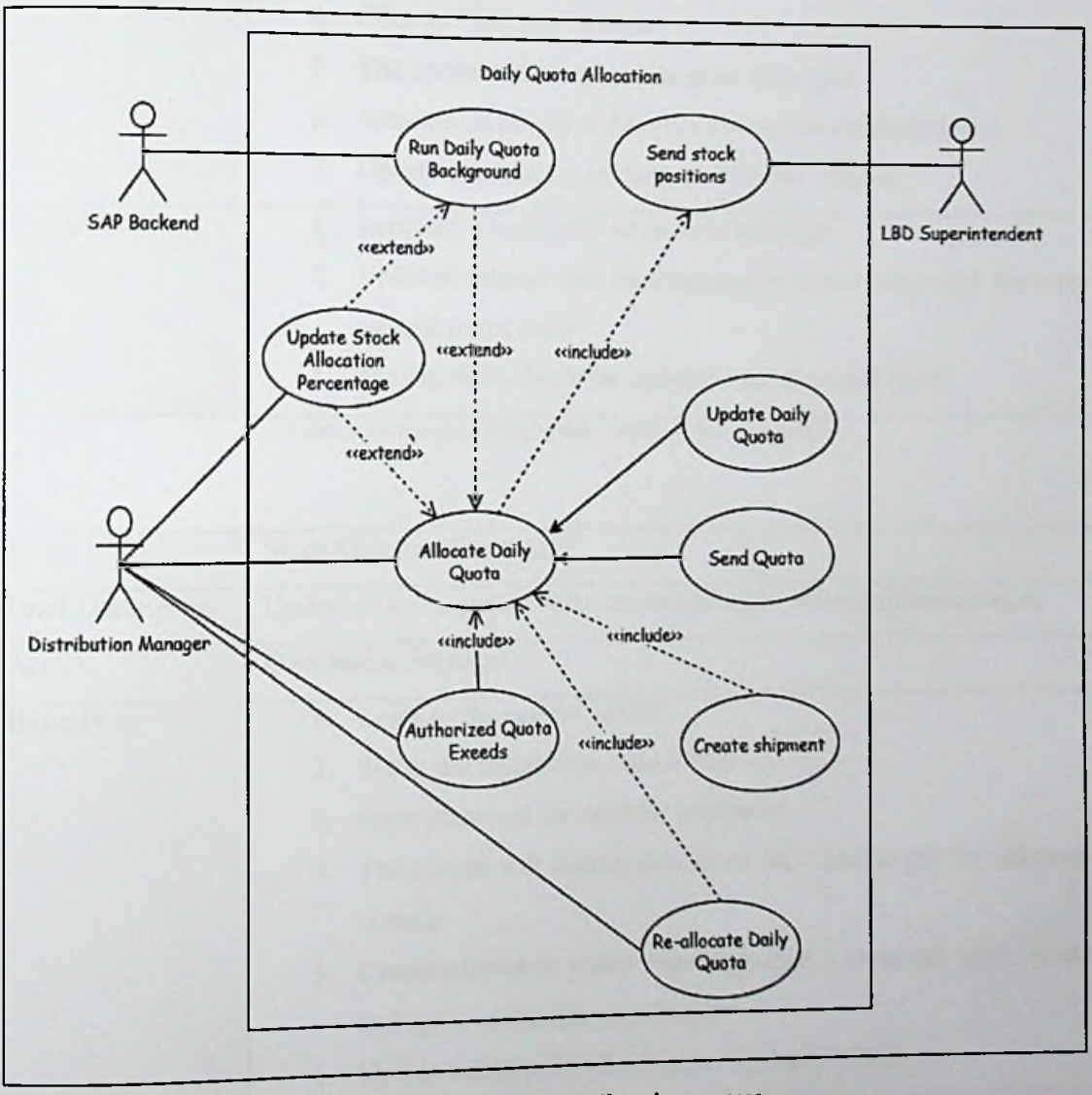


Figure A.3: Daily quota allocation process

As seen in figure A.3, stock positions should be sent by the depot superintendent prior to allocate quota. If the DM wants to change the allocations, it can be done via re-allocate daily quota. If there is no quota allocated or available for the plant/depot, the system will restrict shipment creation. The description of the use case is done by table A.4 A.5 and A.6.

Name	UpdateDailyQuota
Brief Description	Update daily quota
Actors	Distribution Manager
Precondition	1. Send stock positions
Basic Flow	1. Login to the system as DM 2. Select the transaction code YV26 3. Enter operational date and the plant 4. Click on "Display" button 5. The system will display data in an ALV grid 6. Allocate quota which does not exceed the available space 7. Update database by clicking on "Update" button
Post Conditions	1. Received a successful or an error message 2. Updated records will be displayed in green color and the error records in red color 3. The database should be updated with allocated quota

Table A.4: Use case description - Daily quota allocation

Name	StockAllocationPercentage
Brief Description	Update slides, expected sales, quota and space allocation percentages
Actors	Distribution Manager
Basic Flow	1. Login to the system as DM 2. Select the transaction code YV30 3. Enter plant and the material (optional) 4. The system will display data in an ALV grid as per the selection criteria 5. Create/edit/delete slides (bowser/wagon), expected sales, quota and space allocation percentages 6. Update database by clicking on "Update" button
Post Conditions	1. Received a successful or an error message 2. Updated records will be displayed in green color and the error records in red color 3. The database should be updated with updated percentages

Table A.5: Use case description - Stock allocation percentages

Name	ReallocationDailyQuota
Brief Description	Reallocate daily quota
Actors	Distribution Manager
Precondition	1. Update daily quota
Basic Flow	1. Login to the system as DM 2. Select the transaction code YV31 3. Enter operational date and the plant 4. Click on "Display" button 5. The system will display data in an ALV grid 6. Reallocate quota which does not lessen used quota 7. Update database by clicking on "Update" button
Post Conditions	1. Received a successful or an error message 2. Updated records will be displayed in green color and the error records in red color 3. The database should be updated with new allocations

Table A.6: Use case description -Daily quota reallocation

LDB employee, LBD superintendent and Distribution manager played a significant role in the stock planning process. User documentation will be helped to get an idea about the process and its usage for these user groups.

Once login to the system as a LDB employee, he/she (a user who has the authority to update the dip) has to be updated the daily dip as the first thing of the day. This is not new to LDB employees; hence dip posting already available in the old system. Next is the vehicle status updating. Daily dip, vehicle status and stock allocation percentage can be considered as master files in stock planning process, hence these are the files directly effecting stock planning customized process.

Vehicle status

Enter the transaction code YV28. Selection screen will be displayed as per the figure B.1. Enter the creation date and the plant and click on to display data.

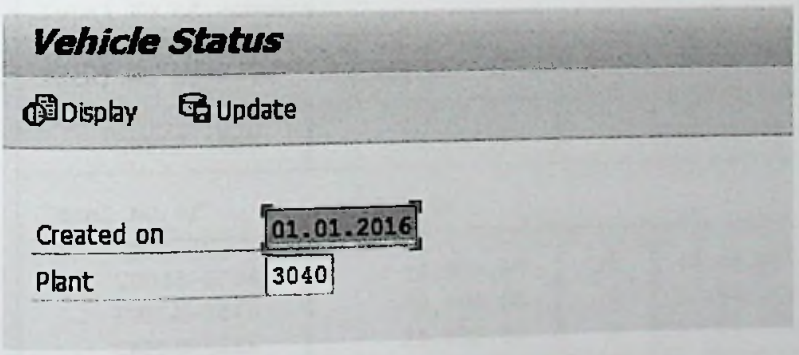


Figure B.1: Vehicle status selection screen

The next screen is vehicle status details. Now update the status by clicking on status tab of the selected entry as can be seen in figure B.2. If you know the status code, simply just enter the code number. Validations are available for invalid status codes. Here you will not allow changing the status, if the stock positions have been sent. Click on to update vehicle status. Click on (in the selection screen) to view the bowser positions (figure B.3) which has been updated.

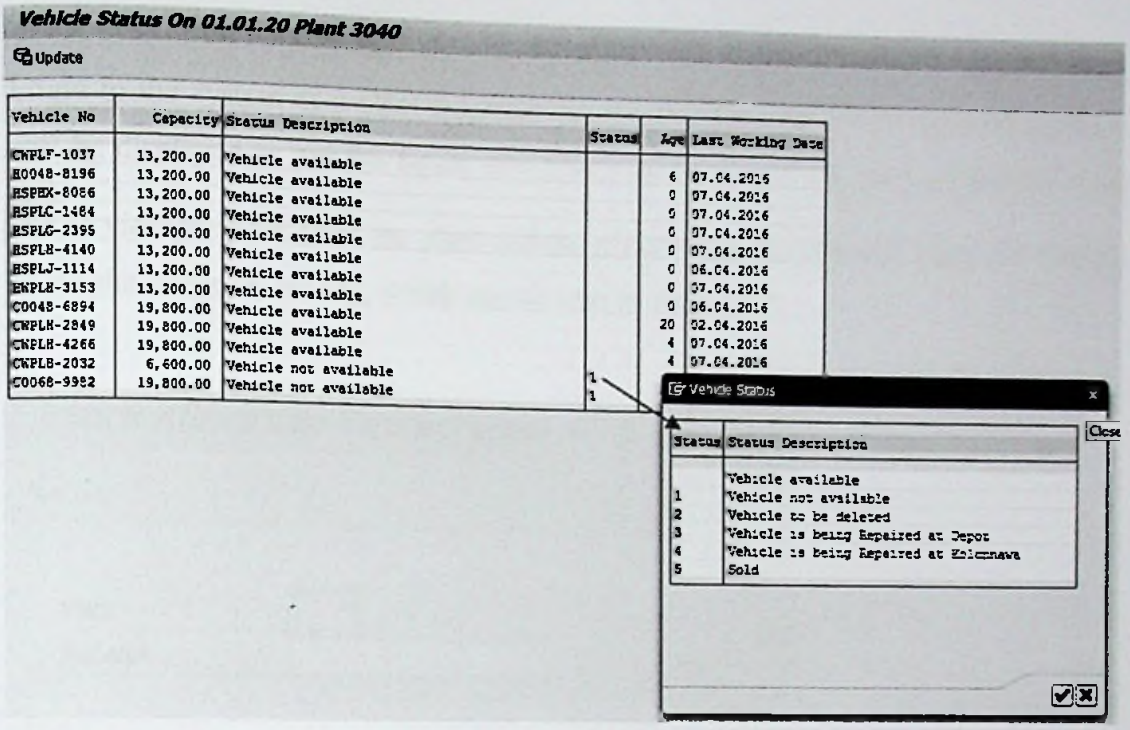


Figure B.2: Vehicle status detail screen

CPSTL Tank Lorries

Vehicle	Capacity	Age	Last Working
Vehicle available Total No of Bowsers:			4
Total No of	13,200.00		1
CWPLF-1037	13,200.00	6	07.04.2016
Total No of	19,800.00		3
C0048-6894 CWPLH-2849 CWPLH-4266	19,800.00 19,800.00 19,800.00	20 4 4	02.04.2016 07.04.2016 07.04.2016
Vehicle not available Total No of Bowsers:			2
Total No of	6,600.00		1
CWPLB-2032	6,600.00	8	12.12.2014
Total No of	19,800.00		1
C0068-9982	19,800.00	17	30.05.2015

Figure B.3: Bowser position of CPSTL/Hired tank lorries

Stock allocation percentage

Login to the system as a DM and enter the transaction code YV30. Selection screen will be displayed as per the figure B.4. Enter the plant and the material and click on  to view the data. Here the plant and the material are an optional. Data will display as per the selection criteria, which can be seen in figure B.5.

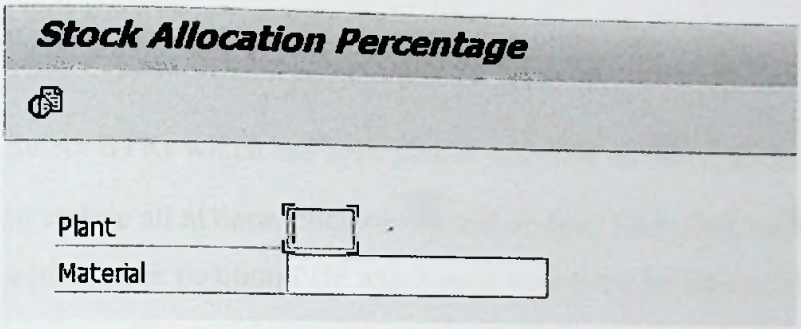
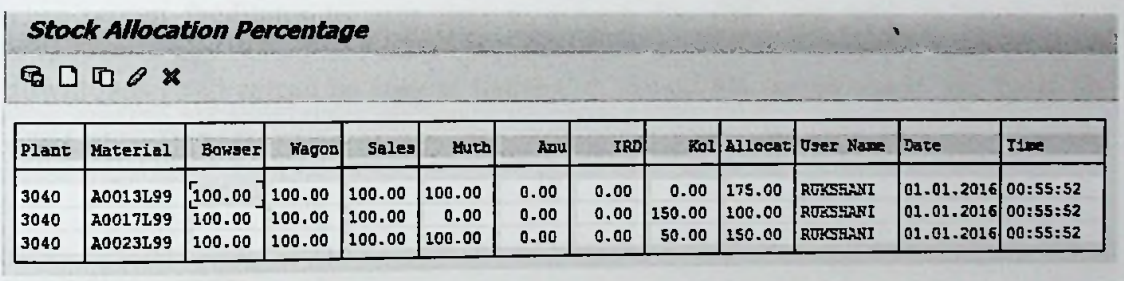


Figure B.4: Stock allocation percentage selection screen



Plant	Material	Bowser	Wagon	Sales	Muth	Anu	IRD	Kol	Allocat	User Name	Date	Time
3040	A0013L99	100.00	100.00	100.00	100.00	0.00	0.00	0.00	175.00	RUKSHANI	01.01.2016	00:55:52
3040	A0017L99	100.00	100.00	100.00	0.00	0.00	0.00	150.00	100.00	RUKSHANI	01.01.2016	00:55:52
3040	A0023L99	100.00	100.00	100.00	100.00	0.00	0.00	50.00	150.00	RUKSHANI	01.01.2016	00:55:52

Figure B.5: Stock allocation percentage detail screen

Once you make the changes, click on  to save changes to the master files. Here you have the facility to add new (), duplicate (), edit () and delete () entries. Note that the changes will move into the history file to access old data.

BTR In-Transit

Login to the system as a LBD superintendent and enter the transaction code YV25. Selection screen will be displayed as per the figure B.6. Enter the creation date and the plant and click on 'Display' button. All the In-Transits (figure B.7) will be displayed in a grid.

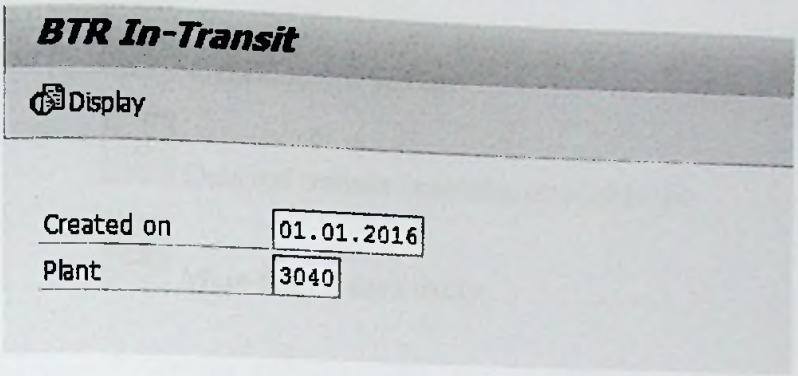
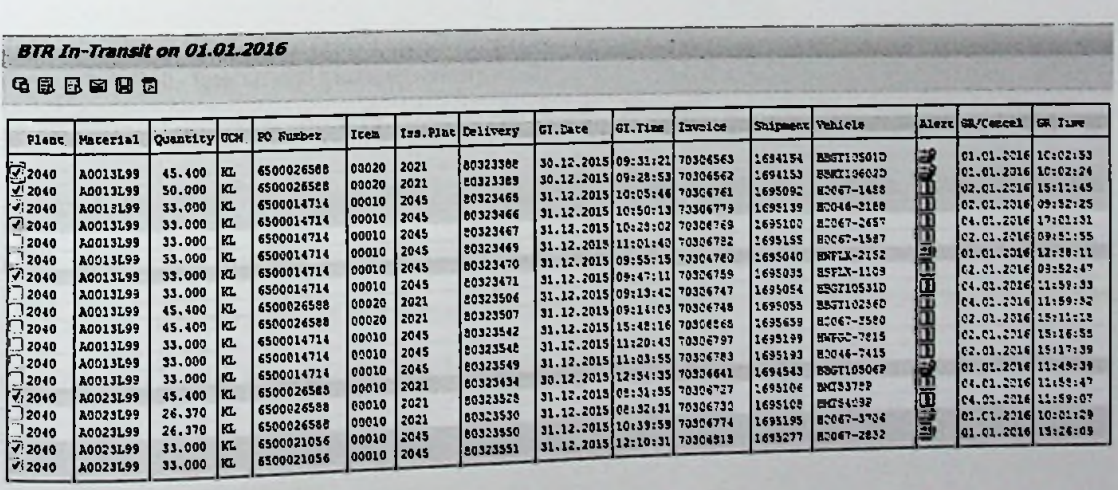


Figure B.6: BTR In-Transit selection screen

Select BTRs which has been parked and click on to update BTR status. If you want to update all at once, click on and update. Then click on to send stock positions with a stock position PDF attachment which can be seen in '*Appendix C - Management Reports*' figure C.1.

This screen facilitates to save data in a local file () and BTR In-Transit PDF downloads () as can be seen in figure C.2. Local file can be a text file, excel file, word file or HTML file or else can be saved in the clipboard.



Plant	Material	Quantity	UOM	PO Number	Item	Iss.Plnt	Delivery	GI.Date	GI.Time	Invoice	Shipment	Vehicle	Alert	GR/Cancel	GR Time
2040	A0013L99	45.400	KL	6500026568	00020	2021	80323388	30.12.2015	09:31:21	70306563	1694154	BBGT135015		01.01.2016	10:02:53
2040	A0013L99	50.000	KL	6500026528	00020	2021	80323389	30.12.2015	09:28:53	70306562	1694153	BBGT136023		01.01.2016	10:02:24
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323465	31.12.2015	10:05:46	70306761	1695092	HC067-1488		02.01.2016	15:11:45
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323466	31.12.2015	10:50:13	70306779	1695139	HC046-2188		02.01.2016	09:52:25
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323467	31.12.2015	10:25:02	70306769	1695102	HC067-2457		04.01.2016	17:01:31
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323469	31.12.2015	11:01:40	70306782	1695155	HC067-1587		02.01.2016	09:52:55
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323470	31.12.2015	09:55:15	70306760	1695040	BBFLX-2152		01.01.2016	12:39:11
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323471	31.12.2015	09:47:11	70306759	1695035	BSFLX-1109		02.01.2016	09:52:47
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323506	31.12.2015	09:13:42	70306747	1695054	BBGT105310		04.01.2016	11:59:33
2040	A0013L99	45.400	KL	6500026588	00020	2021	80323507	31.12.2015	09:14:03	70306748	1695055	BBGT102360		04.01.2016	11:59:32
2040	A0013L99	45.400	KL	6500026588	00020	2021	80323542	31.12.2015	15:48:16	70306568	1695659	HC067-2580		02.01.2016	15:11:12
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323548	31.12.2015	11:20:43	70306797	1695199	HWFGC-7815		02.01.2016	15:16:55
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323549	31.12.2015	11:03:55	70306783	1695193	HC046-7415		02.01.2016	15:17:39
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323634	30.12.2015	12:54:35	70306641	1694543	BBGT105046		01.01.2016	11:49:39
2040	A0023L99	45.400	KL	6500026583	00010	2021	80323528	31.12.2015	09:31:55	70306727	1695106	BMCS372P		04.01.2016	11:59:47
2040	A0023L99	26.370	KL	6500026588	00010	2021	80323530	31.12.2015	08:32:31	70306732	1695108	BMCS423P		04.01.2016	11:59:07
2040	A0023L99	26.370	KL	6500026588	00010	2021	80323550	31.12.2015	10:39:59	70306774	1695195	HC067-3706		01.01.2016	10:01:20
2040	A0023L99	33.000	KL	6500021056	00010	2045	80323551	31.12.2015	12:10:31	70306819	1695277	HC067-2832		01.01.2016	15:26:09
2040	A0023L99	33.000	KL	6500021056	00010	2045									

Figure B.7: BTR In-Transit detail screen

The notations used as alerts;

Delivered In-Transits

Returned documents

Canceled documents

Delayed transits (exceeds grace period)

More than 7 days delay

Daily operational quota

After sending Stock position, the next step is daily quota allocation. Login to the system as a DM and enter the transaction code YV26. Selection screen will be displayed as per the figure B.8. Enter the operational date and the plant and click on 'Display' button.

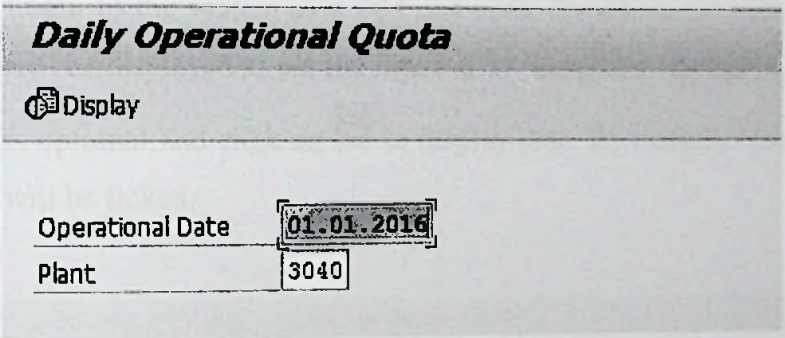


Figure B.8: Daily operational quota selection screen

As can be seen in figure B.9, availability, total sales, In-transits, In-Sliding and allocated space details will be displayed in the grid. Quota allocations are based on allocation percentages. If allocation percentages were not updated prior to this, all the quotas will be allocated to the Kolonnawa installation as a default.

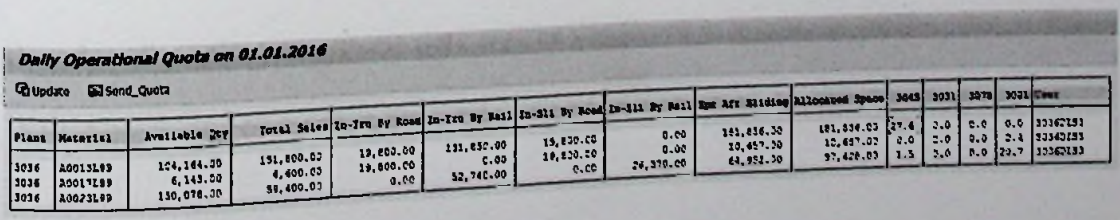


Figure B.9: Daily operational quota

After updating the quota, send allocated quota to the respective terminals to make arrangements to dispatch the updated material quantities as can be seen in figure B.10

To : Deputy Distribution Manager (Muthurajawela)
From : Distribution Manager (CPSTL)
Date : 18.04.2016

Bridging Of Fuel Ex Muthurajawela Terminal

Please make arrangements to dispatch the following quantities of LS, LAD & LK to under mentioned Bulk depots on 01.01.2016

Depots	LS	LAD	LK
Matara Bulk Depot	89,760.00	76,560.00	13,200.00

Distribution Manager (CPSTL)
W.M.T.W.A. Bandara

CC: DGM Operation
Assistant Distribution Manager (Muthurajawela)
Wagon section

Figure B.10: Bridging of fuel

Authorize BTR with exceeding quota limits

Login to the system as a DM and enter the transaction code YV29. Selection screen will be displayed as per the figure B.11. Enter the Transportation planning point, which is optional and click on to display data. By default, 'Only Pending Authorization' will be ticked.

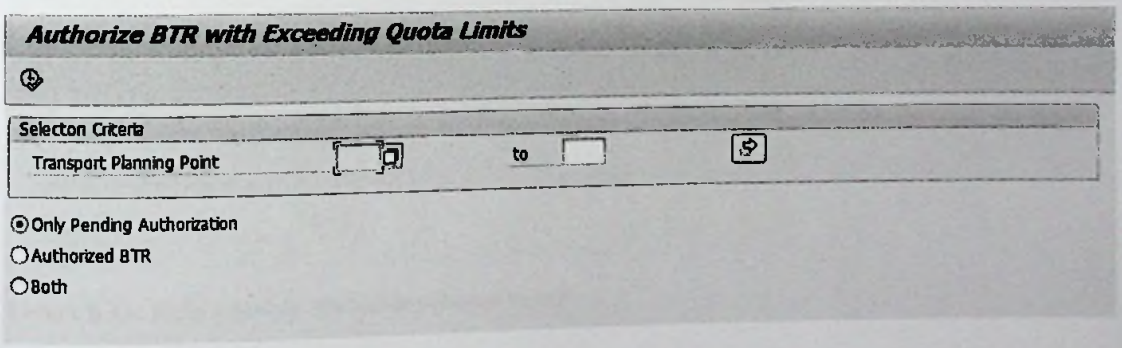
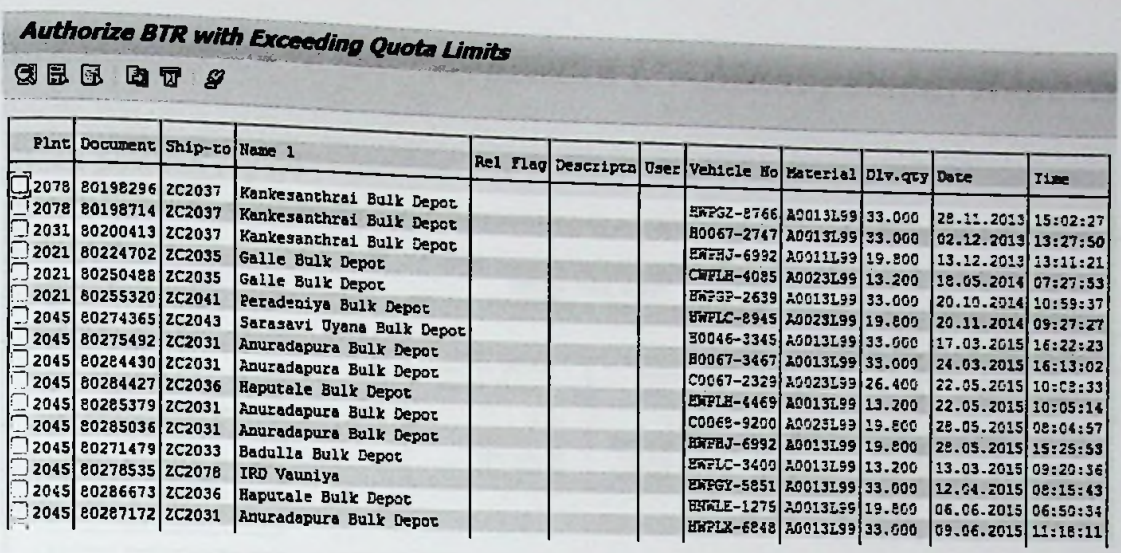


Figure B.11: Authorize BTR with exceeding quota limits selection screen

Based on the selection criteria data will be displayed in a grid, as can be seen in figure B.12. Select one or more entries and click on to allow an exceeded quota.



Plnt	Document	Ship-to	Name 1	Rel Flag	Descriptn	User	Vehicle No	Material	Dlv.qty	Date	Time
2078	80198296	ZC2037	Kankesanthrai Bulk Depot				[redacted]	A0013L99	33.000	28.11.2013	15:02:27
2078	80198714	ZC2037	Kankesanthrai Bulk Depot				[redacted]	A0013L99	33.000	02.12.2013	13:27:50
2031	80200413	ZC2037	Kankesanthrai Bulk Depot				[redacted]	A0011L99	19.800	13.12.2013	13:11:21
2021	80224702	ZC2035	Galle Bulk Depot				[redacted]	A0023L99	13.200	18.05.2014	07:27:53
2021	80250488	ZC2035	Galle Bulk Depot				[redacted]	A0013L99	33.000	20.10.2014	10:59:37
2021	80255320	ZC2041	Peradeniya Bulk Depot				[redacted]	A0023L99	19.800	20.11.2014	09:27:27
2045	80274365	ZC2043	Sarasavi Uyana Bulk Depot				[redacted]	A0013L99	33.000	17.03.2015	16:22:23
2045	80275492	ZC2031	Anuradapura Bulk Depot				[redacted]	A0013L99	33.000	24.03.2015	16:13:02
2045	80284430	ZC2031	Anuradapura Bulk Depot				[redacted]	A0023L99	26.400	22.05.2015	10:08:33
2045	80284427	ZC2036	Haputale Bulk Depot				[redacted]	A0013L99	13.200	22.05.2015	10:05:14
2045	80285379	ZC2031	Anuradapura Bulk Depot				[redacted]	A0025L99	19.800	28.05.2015	08:04:57
2045	80285036	ZC2031	Anuradapura Bulk Depot				[redacted]	A0013L99	19.800	28.05.2015	15:25:53
2045	80271479	ZC2033	Badulla Bulk Depot				[redacted]	A0013L99	13.200	13.03.2015	09:20:36
2045	80278535	ZC2078	IRD Vauniya				[redacted]	A0013L99	33.000	12.04.2015	08:15:43
2045	80286673	ZC2036	Haputale Bulk Depot				[redacted]	A0013L99	19.800	06.06.2015	06:50:34
2045	80287172	ZC2031	Anuradapura Bulk Depot				[redacted]	A0013L99	33.000	09.06.2015	11:18:11

Figure B.12: Authorize BTR with exceeding quota limits detail screen

Daily quota re-allocation

Login to the system as a DM and enter the transaction code YV31. Selection screen will be displayed as per the figure B.13. Enter the operational date and the plant and click on to display data.

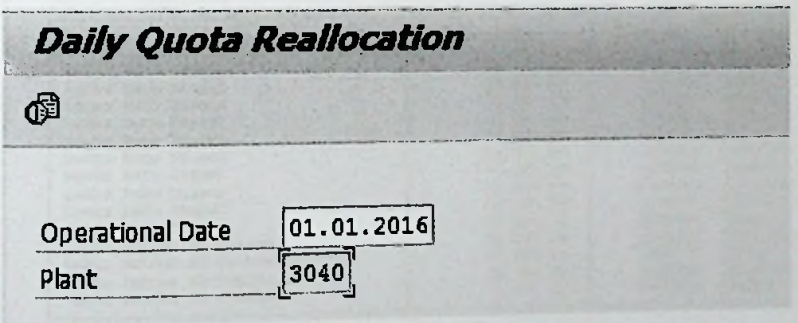
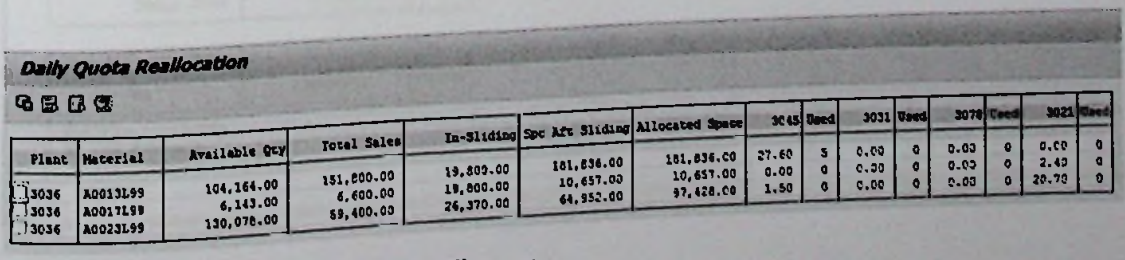


Figure B.13: Daily quota re-allocation selection screen

The daily quota re-allocation interface is displayed by figure B.14, which allows the DM to re-allocate quota which is possible after the quota allocation. Change quota which is not exceeding the used sum and update.



Plant	Material	Available Qty	Total Sales	In-Sliding	Spc Aft Sliding	Allocated Space	3045	Used	3031	Used	3078	Used	3021	Used
3036	A0013L99	104,164.00	151,800.00	19,800.00	181,836.00	181,836.00	27.60	5	0.00	0	0.00	0	0.00	0
3036	A0017L99	6,143.00	6,600.00	18,800.00	10,657.00	10,657.00	0.00	0	0.00	0	0.00	0	2.40	0
3036	A0023L99	130,078.00	59,400.00	26,370.00	64,952.00	97,428.00	1.50	0	0.00	0	0.00	0	20.70	0

Figure B.14: Daily quota re-allocation detail screen

Stock position report

STOCK POSITION REPORT: 01/01/2016
MATARA BULK DEPOT
Tel No: 0412222085 , Fax : 0412224690

Product	Usable Stock Stock (Ltrs)	Today Sales (Ltrs) CPC	Today Sales (Ltrs) LIOC	Yesterday Sales (Ltrs) CPC	Yesterday Sales (Ltrs) LIOC
Lanka Petrol 90 Octane	0.00	0.00	0.00	0.00	0.00
Lanka Auto Diesel	183,564.00	250,800.00	26,400.00	271,208.00	112,200.00
Lanka Kerosene	118,196.00	13,200.00	0.00	13,200.00	0.00
Lanka Petrol 92 Octane	125,713.00	138,600.00	26,400.00	85,800.00	26,400.00

IN-TRANSIT/IN-SLIDING BTR

Product	Qty (Ltrs)	Issue Plant	Shipment	Vehicle	
Lanka Auto Diesel	33,000.00	2045	1695193	KC046-7415	
Lanka Auto Diesel	33,000.00	2045	1695199	KKPGC-7815	
Lanka Auto Diesel	33,000.00	2045	1695659	KC067-3588	
Lanka Auto Diesel	45,400.00	2021	1695055	KKGT10256D	
Lanka Auto Diesel	45,400.00	2021	1695054	KKGT10531D	
Lanka Auto Diesel	33,000.00	2045	1695038	KSVLX-1109	
Lanka Auto Diesel	33,000.00	2045	1695040	KNVLX-2152	*
Lanka Auto Diesel	33,000.00	2045	1695155	KC067-1587	
Lanka Auto Diesel	33,000.00	2045	1695100	KC067-2657	
Lanka Auto Diesel	33,000.00	2045	1695139	KC046-3188	*
Lanka Auto Diesel	33,000.00	2045	1695092	KC067-1488	*
Lanka Auto Diesel	50,000.00	2021	1694153	KKKT10602D	*
Lanka Auto Diesel	45,400.00	2021	1694154	KKGT10501D	*
Lanka Petrol 92 Octane	45,400.00	2021	1694543	KKGT10506D	*
Lanka Petrol 92 Octane	26,370.00	2021	1695106	KKTS3789	
Lanka Petrol 92 Octane	26,370.00	2021	1695108	KKTS4099	
Lanka Petrol 92 Octane	33,000.00	2045	1695195	KC067-3706	*
Lanka Petrol 92 Octane	33,000.00	2045	1695277	KC067-2832	*

* - Indicated the BTR in-sliding for the day.

YESTERDAYS RECEIPTS

Product	Qty (Ltrs)	Shipment	Vehicle
Lanka Auto Diesel	33,000.00	1694037	KC044-8887
Lanka Auto Diesel	33,000.00	1694225	KC067-1355
Lanka Auto Diesel	33,000.00	1694087	KKPGP-4935
Lanka Auto Diesel	26,400.00	1695016	C0067-2040
Lanka Auto Diesel	26,400.00	1695020	C0046-9990
Lanka Auto Diesel	26,400.00	1695069	C0067-2406
Lanka Auto Diesel	33,000.00	1694085	KC067-2923
Lanka Auto Diesel	45,400.00	1693062	KKGT10258D
Lanka Auto Diesel	45,400.00	1693057	KKGT10519D
Lanka Auto Diesel	33,000.00	1694038	KC067-1351
Lanka Auto Diesel	45,400.00	1693233	KKGT10236D
Lanka Petrol 92 Octane	33,000.00	1695195	KC067-3706
Lanka Petrol 92 Octane	33,000.00	1694983	KKKNJ-8773
Lanka Petrol 92 Octane	33,000.00	1694826	KSVLX-2107

BOWSER STATISTIC

CPSTL Tank Lorries

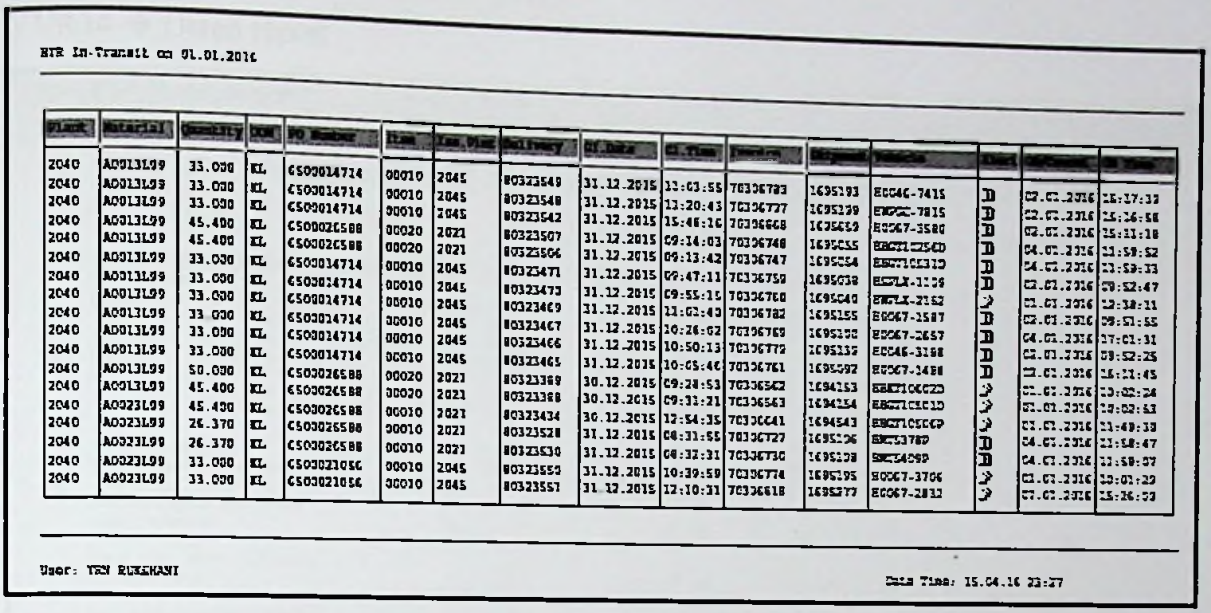
No	Capacity	Years
Vehicle available No. of Bowsers:		4
C0048-6894	19,800.00	20
CWPLP-1037	13,200.00	6
CWPLH-2849	18,800.00	4
CWPLH-4266	19,800.00	4
Vehicle not available No. of Bowsers:		2
C0048-9982	19,800.00	17
CWPLH-2032	6,600.00	8

Hired Tank Lorries

No	Capacity	Years
Vehicle available No. of Bowsers:		7

Figure C.1: Stock position report

BTR In-Transit report



BTR In-Transit on 01.01.2016

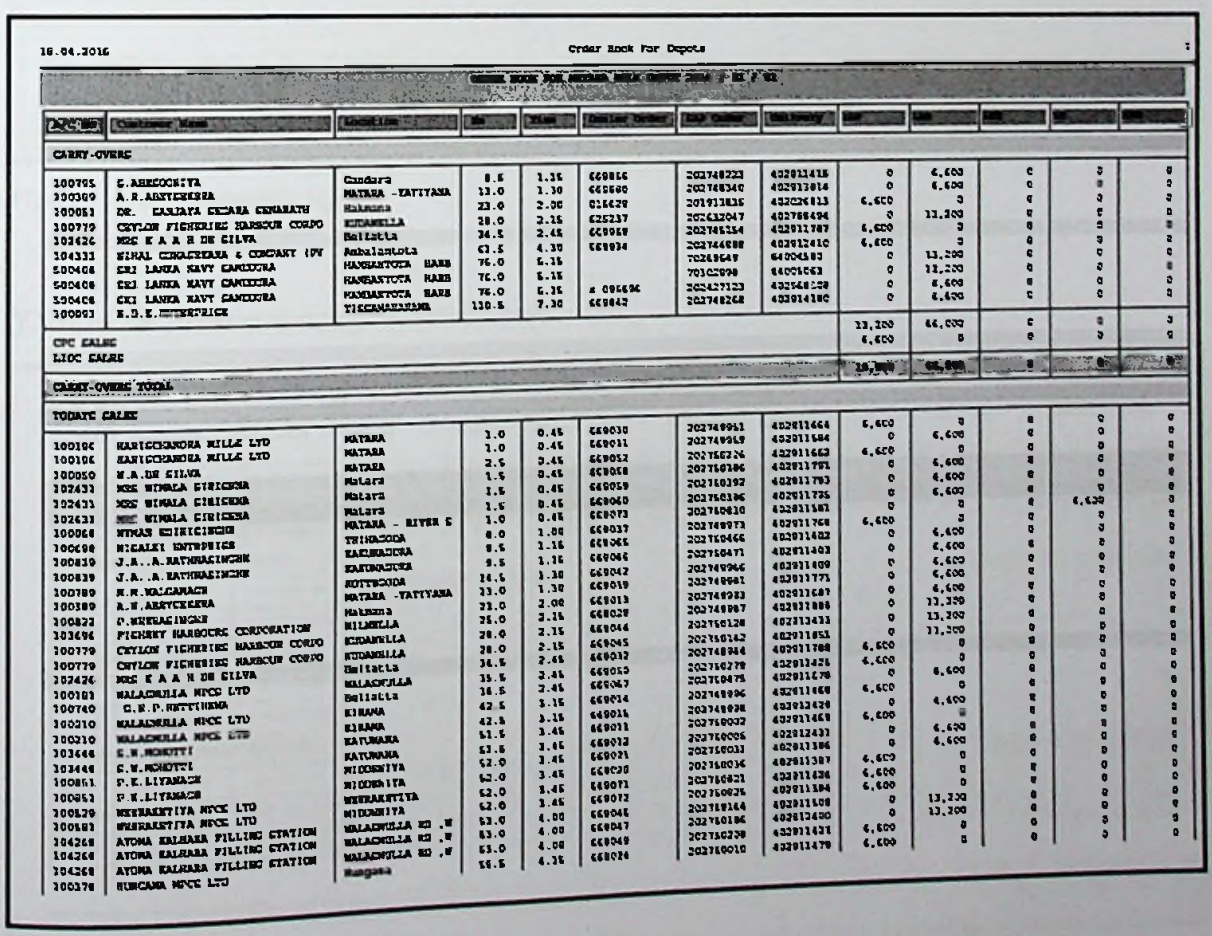
Plant	Material	Quantity	UOM	PO Number	Item	Loc. Plan	Delivery	SI Date	SI Time	Invoice	Shipment	Vehicle	Class	GRN Number	GR Date
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323549	31.12.2015	13:03:55	70336783	1695193	EC046-7415	A		02.01.2016 15:17:33
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323548	31.12.2015	13:20:43	70336727	1695239	EC046-7815	A		02.01.2016 15:16:58
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323542	31.12.2015	15:48:16	70336668	1695659	EC067-3580	A		02.01.2016 15:11:18
2040	A0013L99	45.400	KL	6500025588	00020	2021	80323507	31.12.2015	09:14:03	70336748	1695255	EBCT125CD	A		04.01.2016 11:59:52
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323506	31.12.2015	09:13:42	70336747	1695254	EBCT125CD	A		04.01.2016 11:59:33
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323471	31.12.2015	09:47:11	70336750	1695038	EC7LX-1128	A		02.01.2016 09:52:47
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323473	31.12.2015	09:55:15	70336760	1695040	EBCLX-2152	A		01.01.2016 12:38:11
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323469	31.12.2015	11:03:43	70336782	1695155	EC067-1587	A		02.01.2016 09:51:55
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323467	31.12.2015	10:26:02	70336769	1695132	EC067-2657	A		04.01.2016 17:01:31
2040	A0013L99	33.000	KL	6500014714	00010	2045	80323466	31.12.2015	10:50:13	70336772	1695137	EC046-3188	A		02.01.2016 09:52:25
2040	A0013L99	50.000	KL	6500026588	00020	2021	80323465	31.12.2015	10:05:46	70336761	1695092	EC067-1488	A		02.01.2016 15:21:45
2040	A0013L99	45.400	KL	6500026588	00020	2021	80323369	30.12.2015	09:28:53	70336542	1694153	EBCT106CCD	A		01.01.2016 13:02:24
2040	A0023L99	45.400	KL	6500026588	00010	2021	80323368	30.12.2015	09:31:21	70336543	1694154	EBCT105CCD	A		01.01.2016 12:02:43
2040	A0023L99	26.370	KL	6500025588	00010	2021	80323434	30.12.2015	12:54:35	70336641	1694543	EBCT105CCD	A		01.01.2016 11:49:38
2040	A0033L99	26.370	KL	6500025588	00010	2021	80323528	31.12.2015	08:31:55	70336727	1695106	EBCT125D	A		04.01.2016 11:58:47
2040	A0023L99	33.000	KL	6500021056	00010	2045	80323550	31.12.2015	10:39:59	70336774	1695195	EC067-3706	A		01.01.2016 13:01:29
2040	A0023L99	33.000	KL	6500021056	00010	2045	80323557	31.12.2015	12:10:31	70336818	1695277	EC067-2812	A		07.01.2016 15:26:59

User: TEN RUKSHANI

Date Time: 15.04.16 23:27

Figure C.2: BTR In-Transit report

Sales order book



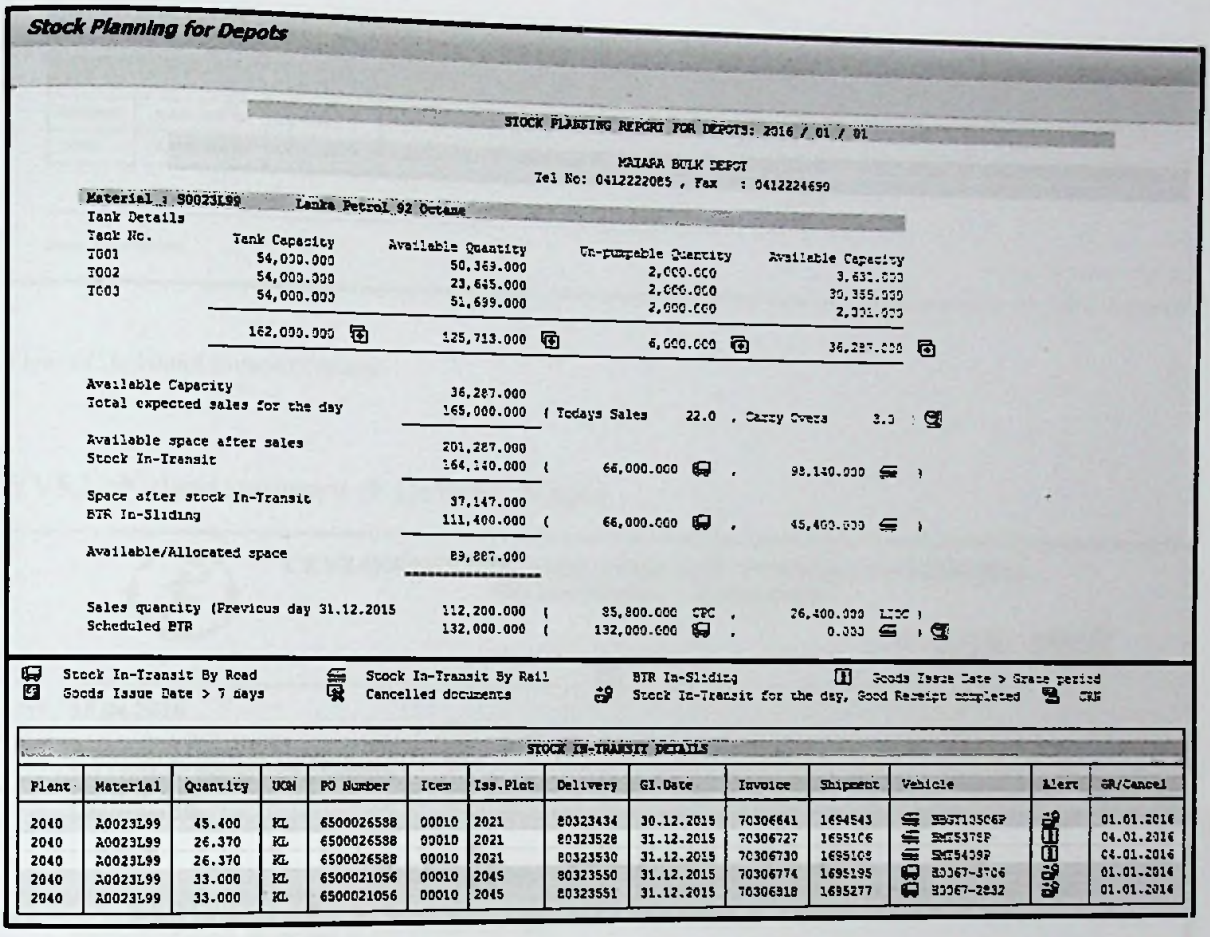
16.04.2016 Order Book For Depots

Customer	Customer Name	Location	Qty	Class	Dealer Order	SAP Order	Delivery	Ref	Ref	Ref	Ref	Ref
CARRY-OVERS												
100795	C.ABEYSOORIYA	Gandara	8.5	1.15	6698 16	2027482223	4029114 15	0	6,600	0	0	0
100369	A.R.ABEYSEKERA	MATARA -YATIYANA	13.0	1.30	6685 90	2027483 40	4029130 16	0	6,600	0	0	0
100051	DR. SANJAYA SENARA SENARATH	Matara	23.0	2.00	6714 29	2019133 35	4032261 13	6,600	0	0	0	0
100779	CEYLON FISHERIES HARBOUR CORPO	KUDAMILLA	28.0	2.15	6252 37	2027432 47	4027864 94	0	13,200	0	0	0
102426	MRS K A A N DE SILVA	Ballatta	34.5	2.45	6699 69	2027463 54	4029117 87	6,600	0	0	0	0
104331	SINAL CONSTRUCTION & CONSULT (PV	Ambalangoda	63.5	4.30	6699 34	2027444 88	4029124 10	6,600	0	0	0	0
500406	SRI LANKA NAVY CANTEEN	HANGANTOTA HARB	76.0	5.15		70283649	6400438 3	0	13,200	0	0	0
500406	SRI LANKA NAVY CANTEEN	HANGANTOTA HARB	76.0	5.15		7012 0098	6400106 3	0	13,200	0	0	0
500406	SRI LANKA NAVY CANTEEN	HAMBANTOTA HARB	76.0	5.15	a 095696	2024271 23	4025482 20	0	6,600	0	0	0
500406	SRI LANKA NAVY CANTEEN	TRINCOMALEE HARB	130.5	7.30	6698 42	2027482 68	4029141 80	0	6,600	0	0	0
100093	M.D.E. [illegible]ERICK							13,200	66,000	0	0	0
CPC SALES								6,600	0	0	0	0
LIOC SALES								19,800	66,000	0	0	0
CARRY-OVERS TOTAL												
TODAYS SALES												
100190	HARISCHANDRA MILLS LTD	MATARA	1.0	0.45	6698 30	2027499 51	4029116 64	6,600	0	0	0	0
100190	HARISCHANDRA MILLS LTD	MATARA	1.0	0.45	6698 31	2027499 59	4029116 84	0	6,600	0	0	0
100050	M.A.DE SILVA	MATARA	2.5	0.45	6698 52	2027562 24	4029116 63	6,600	0	0	0	0
102431	MRS WIMALA SIRISENA	MATARA	1.5	0.45	6698 58	2027567 06	4029117 51	0	6,600	0	0	0
102431	MRS WIMALA SIRISENA	Matara	1.5	0.45	6698 59	2027610 97	4029117 93	0	6,600	0	0	0
102431	MRS WIMALA SIRISENA	Matara	1.5	0.45	6698 60	2027567 06	4029117 25	0	0	0	6,600	0
102431	MRS WIMALA SIRISENA	Matara	1.5	0.45	6698 61	2027610 10	4029115 67	0	0	0	0	0
100068	NIMAL SIRISINGHE	MATARA - RIVER S	1.0	0.45	6698 73	2027499 73	4029117 68	6,600	0	0	0	0
100698	MIGALEI ENTERPRISE	THIHAGODA	8.0	1.00	6699 37	2027610 66	4029114 82	0	6,600	0	0	0
100839	J.A. .A.RATNAYAKE	KACHCHAIDEVA	9.5	1.15	6699 45	2027567 71	4029114 03	0	6,600	0	0	0
100839	J.A. .A.RATNAYAKE	KANTHARUWA	9.5	1.15	6699 66	2027499 66	4029114 09	0	6,600	0	0	0
100839	J.A. .A.RATNAYAKE	KOTTEGODA	14.5	1.30	6699 42	2027499 81	4029117 71	0	6,600	0	0	0
100789	M.M.WALGAMACH	MATARA -YATIYANA	13.0	1.30	6699 19	2027499 83	4029116 87	0	6,600	0	0	0
100389	A.H.ABEYSEKERA	Matara	23.0	2.00	6698 13	2027482 67	4029117 88	0	13,200	0	0	0
100822	P.NIWASINGHE	HILMILLA	25.0	2.15	6698 29	2027562 28	4029114 33	0	13,200	0	0	0
103696	FISHERY HARBOURS CORPORATION	KUDAMILLA	28.0	2.15	6698 44	2027562 42	4029117 53	0	13,200	0	0	0
100779	CEYLON FISHERIES HARBOUR CORPO	KUDAMILLA	28.0	2.15	6698 45	2027483 44	4029117 88	6,600	0	0	0	0
100779	CEYLON FISHERIES HARBOUR CORPO	Ballatta	34.5	2.45	6698 32	2027562 78	4029124 25	6,600	0	0	0	0
102426	MRS K A A N DE SILVA	Ballatta	34.5	2.45	6698 53	2027610 75	4029117 78	0	6,600	0	0	0
100101	MALAGHILLA MPCS LTD	MALAGHILLA	38.5	2.45	6698 67	2027483 96	4029114 68	6,600	0	0	0	0
100740	G.H.P.HETTIHENA	Ballatta	42.5	3.15	6699 14	2027483 98	4029124 28	0	6,600	0	0	0
100210	MALAGHILLA MPCS LTD	KIRAMA	42.5	3.15	6699 15	2027499 32	4029114 69	6,600	0	0	0	0
100210	MALAGHILLA MPCS LTD	KIRAMA	42.5	3.45	6699 11	2027610 02	4029114 69	0	6,600	0	0	0
103446	S.M.MOHOTTI	KATUNARA	51.5	3.45	6699 12	2027610 06	4029124 33	0	6,600	0	0	0
103446	S.M.MOHOTTI	KATUNARA	51.5	3.45	6699 21	2027567 33	4029113 86	0	0	0	0	0
100861	P.K.LIYANAGE	NIDOMIYA	52.0	3.45	6699 20	2027610 36	4029113 87	6,600	0	0	0	0
100861	P.K.LIYANAGE	NIDOMIYA	52.0	3.45	6699 71	2027567 21	4029114 26	6,600	0	0	0	0
100861	P.K.LIYANAGE	NIDOMIYA	52.0	3.45	6699 72	2027610 25	4029114 84	0	0	0	0	0
100179	WEERARATIYA MPCS LTD	WEERARATIYA	52.0	3.45	6699 46	2027483 44	4029115 08	0	13,200	0	0	0
100183	WEERARATIYA MPCS LTD	NIDOMIYA	53.0	4.00	6699 47	2027483 86	4029124 00	0	13,200	0	0	0
104268	ATOMA KALAWARA FILLING STATION	MALAGHILLA RD ,W	53.0	4.00	6699 49	2027562 30	4029114 31	6,600	0	0	0	0
104268	ATOMA KALAWARA FILLING STATION	MALAGHILLA RD ,W	53.0	4.00	6699 24	2027610 10	4029114 79	6,600	0	0	0	0
104268	ATOMA KALAWARA FILLING STATION	MALAGHILLA RD ,W	53.5	4.15								
100378	HUNCAMA MPCS LTD	Hungama										

Figure C.3: Sales order book

Stock planning summary for bulk depots

YVR34 → Detail report



Stock Planning for Depots

STOCK PLANNING REPORT FOR DEPOTS: 2016 / 01 / 01

MAIARA BULK DEPOT
Tel No: 0412222085 , Fax : 0412224650

Material : S0023L99 Lanka Petrol 92 Octane

Tank Details

Tank No.	Tank Capacity	Available Quantity	Un-pumpable Quantity	Available Capacity
T001	54,000.000	50,369.000	2,000.000	3,631.000
T002	54,000.000	23,645.000	2,000.000	30,355.000
T003	54,000.000	51,699.000	2,000.000	2,301.000
	162,000.000	125,713.000	6,000.000	36,287.000

Available Capacity	36,287.000	
Total expected sales for the day	165,000.000	(Todays Sales 22.0 , Carry Overs 3.0)
Available space after sales	201,287.000	
Stock In-Transit	164,140.000	(66,000.000 , 98,140.000)
Space after stock In-Transit	37,147.000	
BTR In-Sliding	111,400.000	(66,000.000 , 45,400.000)
Available/Allocated space	89,887.000	
Sales quantity (Previous day 31.12.2015	112,200.000	(85,800.000 CPC , 26,400.000 LIOC)
Scheduled BTR	132,000.000	(132,000.000 , 0.000)

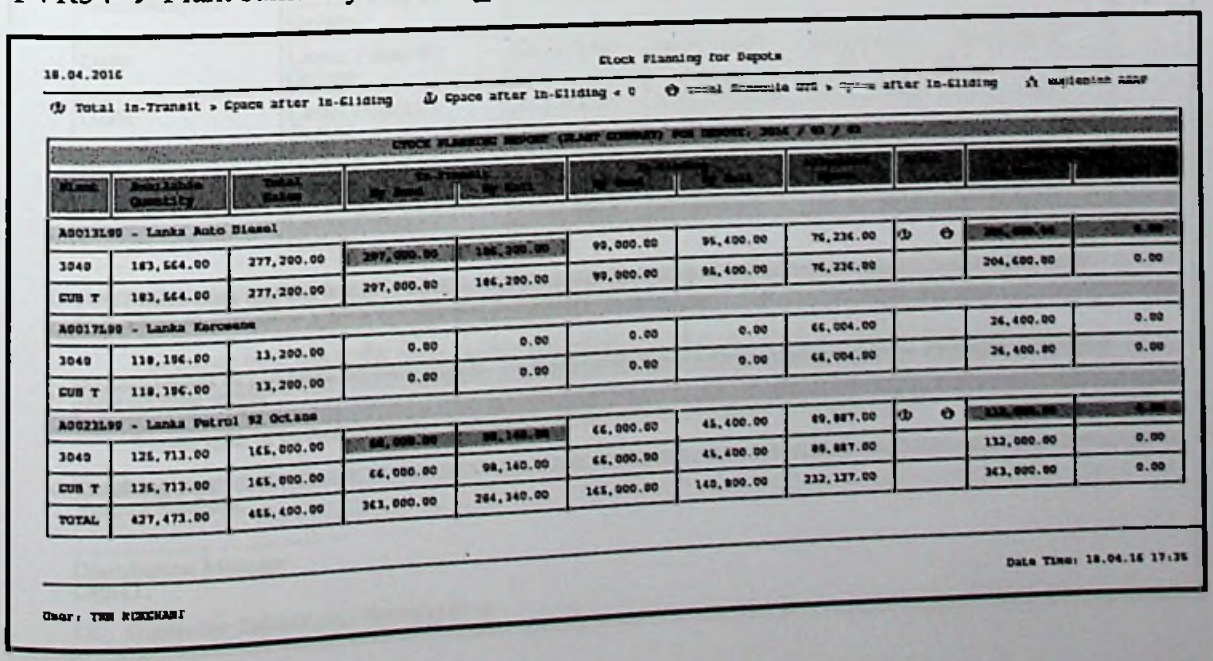
Stock In-Transit By Road | Stock In-Transit By Rail | BTR In-Sliding | Goods Issue Date > Grace period
Goods Issue Date > 7 days | Cancelled documents | Stock In-Transit for the day, Good Receipt completed | CRN

STOCK IN-TRANSIT DETAILS

Plant	Material	Quantity	UOM	PO Number	Item	Iss.Plat	Delivery	GI.Date	Invoice	Shipment	Vehicle	Alert	GR/Cancel
2040	A0023L99	45.400	KL	6500026588	00010	2021	80323434	30.12.2015	70306641	1694543	SBST105062		01.01.2016
2040	A0023L99	26.370	KL	6500026588	00010	2021	80323528	31.12.2015	70306727	1695106	SMT5375P		04.01.2016
2040	A0023L99	26.370	KL	6500026588	00010	2021	80323530	31.12.2015	70306730	1695108	SMT5439P		04.01.2016
2040	A0023L99	33.000	KL	6500021056	00010	2045	80323550	31.12.2015	70306774	1695195	83067-3706		01.01.2016
2040	A0023L99	33.000	KL	6500021056	00010	2045	80323551	31.12.2015	70306818	1695277	83067-2832		01.01.2016

Figure C.4: Stock planning report

YVR34 → Plant summary → Plant_Sum



Stock Planning for Depots

18.04.2016

Total In-Transit > Space after In-Sliding | Space after In-Sliding < 0 | Total Allocate Qnt > Space after In-Sliding | Maximum Allocation

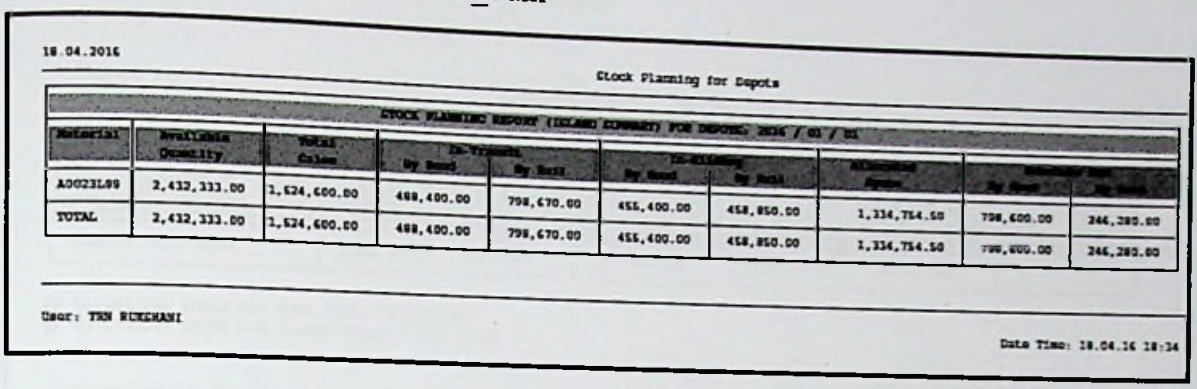
Plant	Available Quantity	Total Sales	In-Transit By Road	In-Transit By Rail	BTR In-Sliding	Allocated Quantity	Space after In-Sliding		
A0013L99 - Lanka Auto Diesel									
3040	183,564.00	277,200.00	297,000.00	186,200.00	99,000.00	95,400.00	76,236.00	204,600.00	0.00
SUB T	183,564.00	277,200.00	297,000.00	186,200.00	99,000.00	95,400.00	76,236.00	204,600.00	0.00
A0017L99 - Lanka Kerosene									
3040	118,196.00	13,200.00	0.00	0.00	0.00	0.00	66,004.00	26,400.00	0.00
SUB T	118,196.00	13,200.00	0.00	0.00	0.00	0.00	66,004.00	26,400.00	0.00
A0023L99 - Lanka Petrol 92 Octane									
3040	125,713.00	165,000.00	66,000.00	98,140.00	66,000.00	45,400.00	89,887.00	132,000.00	0.00
SUB T	125,713.00	165,000.00	66,000.00	98,140.00	66,000.00	45,400.00	89,887.00	132,000.00	0.00
TOTAL	427,473.00	455,400.00	363,000.00	284,340.00	165,000.00	140,800.00	232,127.00	363,000.00	0.00

Data Time: 18.04.16 17:35

User: THR RUKSHANI

Figure C.5: Plant summary report

YVR3 →Island summary → Island_Sum



18.04.2016

Stock Planning for Depots

STOCK PLANNING REPORT (ISLAND SUMMARY) FOR DEPOT: 2016 / 01 / 01

Material	Available Quantity	Total Value	In-Transit		In-Pipeline		Allocated Sales	Delivered Sales	
			By Road	By Rail	By Road	By Rail		By Road	By Rail
A0023L89	2,432,333.00	1,624,600.00	488,400.00	798,670.00	455,400.00	458,850.00	1,334,754.50	798,600.00	246,280.00
TOTAL	2,432,333.00	1,624,600.00	488,400.00	798,670.00	455,400.00	458,850.00	1,334,754.50	798,600.00	246,280.00

User: TRN RUKEMANI

Date Time: 18.04.16 18:34

Figure C.6: Island summary report

YVR3 →Island summary → Delayed_Wagon

CEYLON PETROLEUM STORAGE TERMINALS LIMITED
Oil Installation - Kolonnawa

Tel: 2572307 / 2691643
2694482 / 2532122

18.04.2016

Additional General Manager (Operating)
Sri Lanka Railway Headquarters,
PO Box 355,
Colombo 10.

Dear Sir,

Ref: CGR Fuel Wagon Delays for 01.01.2016

It has been noted that the following wagons, which were due to our bulk depots have not yet reached/delayed, according to the scheduled dates.

Due Destination	Material	Good Issue Date	Wagon No.	Scheduled Good Receipt Date	Actual Good Receipt Date
Anuradhapura	Lanka Petrol 92 Octane	30.12.2015	BBGT10235P	01.01.2016	04.01.2016
Galle	Lanka Petrol 92 Octane	31.12.2015	BMT5388P	01.01.2016	02.01.2016
Galle	Lanka Petrol 92 Octane	31.12.2015	BBGT10262P	01.01.2016	02.01.2016
Matara	Lanka Petrol 92 Octane	31.12.2015	BMT5378P	01.01.2016	04.01.2016
Matara	Lanka Petrol 92 Octane	31.12.2015	BMT5409P	01.01.2016	04.01.2016

*** Please note that the items in bold, refers to deliveries which are more than 7 day delay and items mark as * refers to deliveries which are delayed but placed after 7 days.

I re-iterate that, due to the delay in planning wagons, we have encountered difficulty in planning replenishment of products for bulk depots. Therefore I have no alternative, but to deduct shortages in wagons planned after 48hrs.

Please look in to this matter immediately and fax your response as soon as possible.

Thanking you,
Yours Faithfully,

Distribution Manager
CPSTL

CC: Controller Rabukkana / Nawalapitiya

Figure C.7: Wagon delay report

YVR3 → Island summary → Delayed_Bowser

18.04.2016

Stock Planning for Depots

1

Delayed Bowser Report 01.01.2016

[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
Badulla Bulk Depot	Lanka Petrol 92 Octane	31.12.2015	H0067-2508	01.01.2016	

Deliveries which are more than 7 day delay
Deliveries which are placed after 7 day delay

User: THN RUKSHANI

Date Time: 18.04.16 18:37

Figure C.8: Bowser delay report

Storage capacity utilization report

YVR82 → Local File → Spreadsheet

STORAGE CAPACITY UTILIZATION REPORT

REPORT PERIOD: 2016/01/01 TO 2016/01/15

Plant / Product	Plant Name / Product Name	Tot. Capacity (Kl)	Tot. Out of Servic	Ava. Capacity (Kl)	Ava. Stock (Kl)	Util(%)
3021	Kolonnawa Terminal					
S0011L99	Lanka Petrol 90 Octane	1,222,830.00	0	1,222,830.00	0	0
S0012L99	Lanka Industrial Kerosene	31,545.00	0	31,545.00	12,831.08	40.675
S0013L99	Lanka Auto Diesel	1,377,270.00	187,500.00	1,189,770.00	268,707.30	19.51
S0016L99	Lanka Super Diesel	242,160.00	0	242,160.00	74,870.27	30.918
S0017L99	Lanka Kerosene	516,585.00	0	516,585.00	83,544.68	16.172
S0018L99	Lanka Petrol 95 Octane	375,435.00	49,395.00	326,040.00	135,442.39	36.076
S0021L99	Lanka Chemical Naptha	303,750.00	0	303,750.00	323,521.97	106.509
S0023L99	Lanka Petrol 92 Octane	1,198,185.00	15,717.00	1,182,468.00	234,192.95	19.546
S0026L99	Lanka Fuel Oil 800 Sec.	1,037,175.00	61,155.00	976,020.00	236,301.09	22.783
S0029L99	Lanka Fuel Oil 1500 Sec.(High Sulpher)	1,151,475.00	177,840.00	973,635.00	0	0
S0039L99	Lanka Fuel 1500 Oil Sec.(Low sulpher)	500,880.00	0	500,880.00	0	0
S0056L99	JET A1	568,920.00	186,420.00	382,500.00	190,739.10	33.527
3045	MuthurajawelaTerminal					
S0011L99	Lanka Petrol 90 Octane	590,505.00	590,505.00	0	0	0
S0013L99	Lanka Auto Diesel	2,575,860.00	161,610.00	2,414,250.00	1,642,484.14	63.764
S0017L99	Lanka Kerosene	648,270.00	0	648,270.00	149,291.78	23.029
S0023L99	Lanka Petrol 92 Octane	1,075,500.00	74,355.00	1,001,145.00	636,729.99	59.203
S0026L99	Lanka Fuel Oil 800 Sec.	322,770.00	0	322,770.00	253,885.01	78.658
S0039L99	Lanka Fuel 1500 Oil Sec.(Low sulpher)	807,360.00	0	807,360.00	20,667.30	2.56

Figure C.9: Storage capacity utilization spreadsheet

YVR82 → Plant → 3021

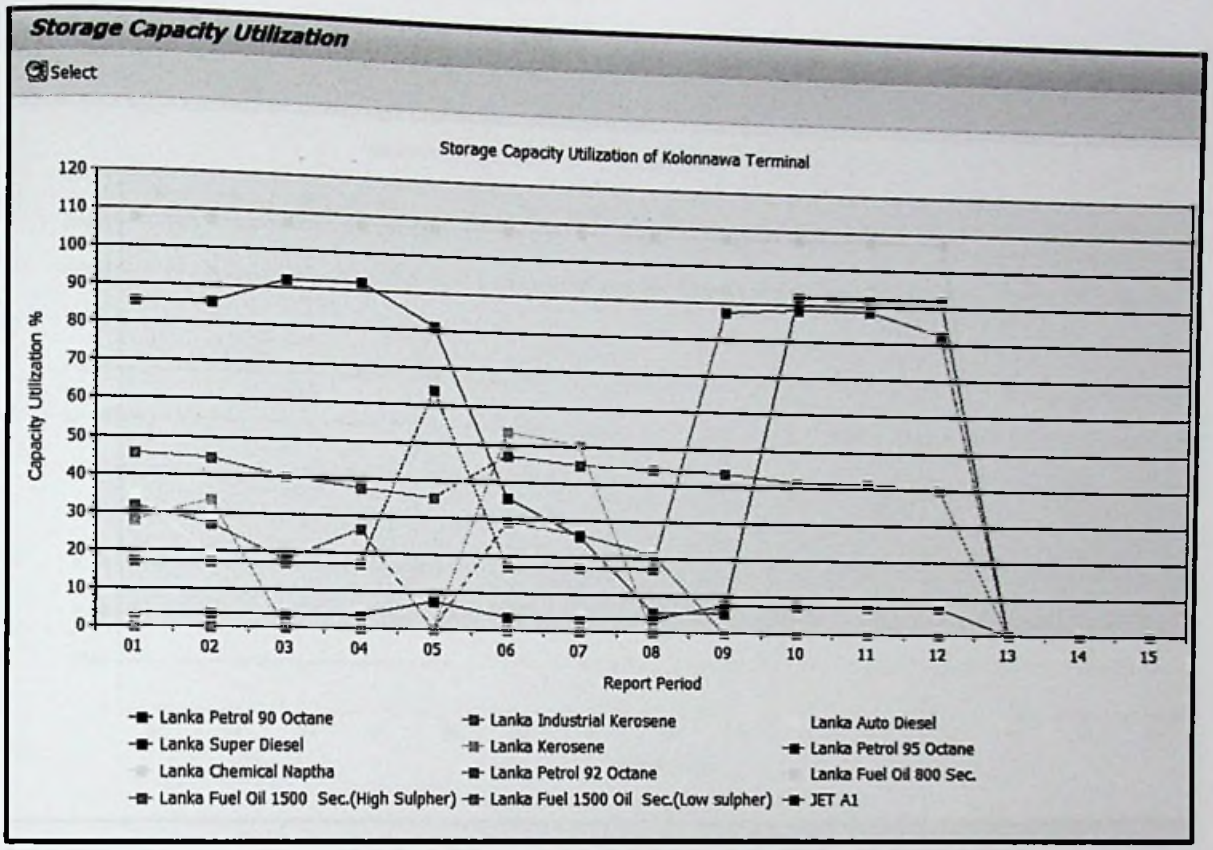


Figure C.10: Storage capacity utilization of Kolonnawa terminal

YVR82 → Material → S0023L99

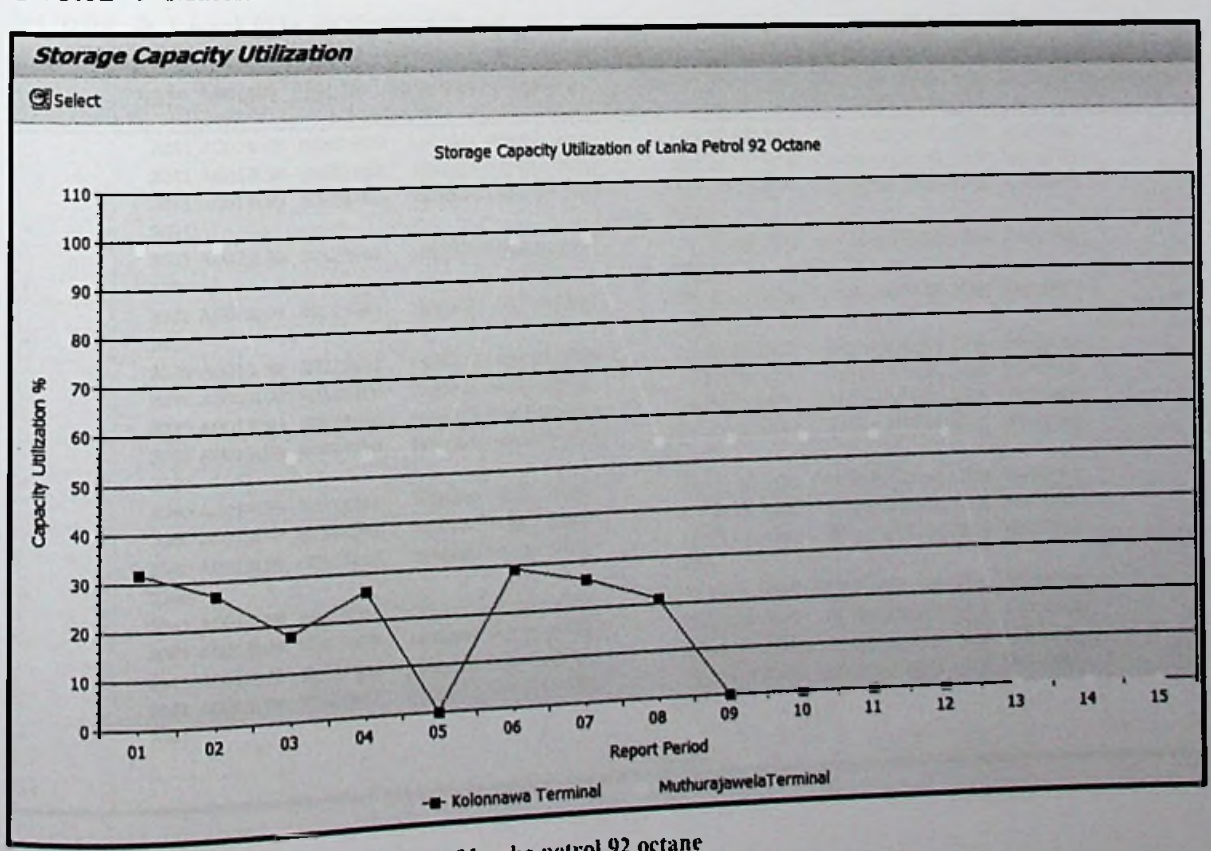


Figure C.11: Storage capacity utilization of Lanka petrol 92 octane

YVR82 → Plant & Mat → 3021/S0023L99

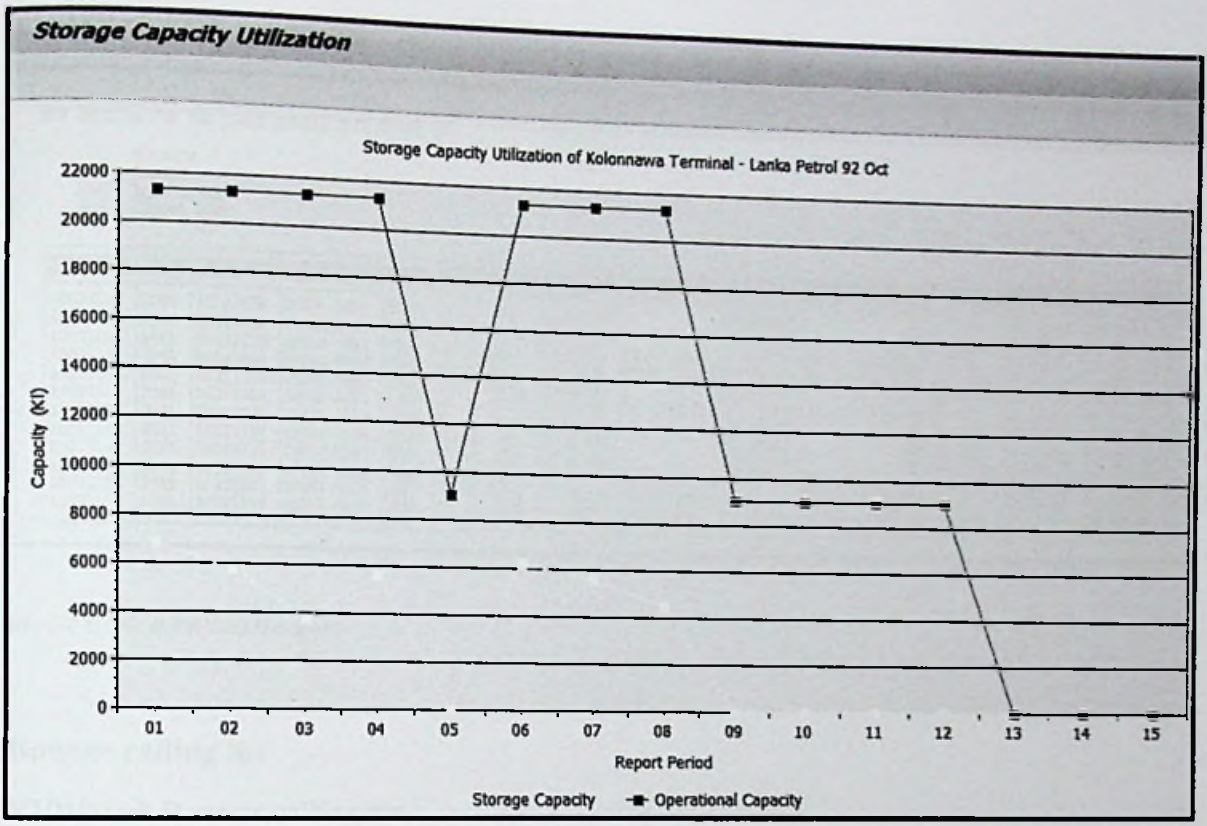


Figure C.12: Storage and operational capacity

BTR release status report

YVR88 → Local File → Spreadsheet

	Ship-to	Material	Doc. No	Shipment	Vehicle No	Quantity	Exit Date	Exit Time	Auth. Date	Authorized	Descriptio
	2031	A0013L99	80287995	1516167	H0067-3774	33	13.06.2015	16:31:08	01.01.2016	3021DIS1	
	2031	A0013L99	80323782	1696593	HNWLE-4384	19.8	01.01.2016	16:14:40	01.01.2016	3021DIS1	
	2031	A0013L99	80323783	1696589	HNWLC-8103	19.8	01.01.2016	16:29:53	01.01.2016	3021DIS1	
*	2031					72.6					
	2033	A0023L99	80323642	1695874	H0067-3774	33	01.01.2016	8:31:32	01.01.2016	3021DIS1	
*	2033					33					
	2038	A0013L99	80323741	1696510	HWPLX-0853	33	01.01.2016	15:31:41	01.01.2016	3021DIS1	
*	2038					33					
	2039	A0013L99	80323117	1696576	HWPLX-0386	33	01.01.2016	16:12:44	01.01.2016	3021DIS1	
	2039	A0013L99	80323266	1696389	H0067-2306	33	01.01.2016	14:31:01	01.01.2016	3021DIS1	
	2039	A0013L99	80323534	1696572	HWPGP-6871	33	01.01.2016	16:15:36	01.01.2016	3021DIS1	
	2039	A0013L99	80323759	1696369	HNWLH-1601	19.8	01.01.2016	13:44:33	01.01.2016	3021DIS1	
*	2039					118.8					
	2040	A0013L99	80323541	1696240	H0067-0876	33	01.01.2016	13:31:57	01.01.2016	3021DIS1	
	2040	A0013L99	80323543	1696169	H0067-1355	33	01.01.2016	12:58:58	01.01.2016	3021DIS1	
	2040	A0023L99	80323197	1696380	HSPLX-0808	33	01.01.2016	13:34:26	01.01.2016	3021DIS1	
*	2040					99					
	2041	A0013L99	80323601	1696272	HSGJB-6033	33	01.01.2016	13:02:36	01.01.2016	3021DIS1	
	2041	A0013L99	80323657	1696206	HWPGA-7584	33	01.01.2016	13:19:30	01.01.2016	3021DIS1	
	2041	A0013L99	80323660	1696330	H0067-2580	33	01.01.2016	15:15:07	01.01.2016	3021DIS1	
	2041	A0013L99	80323662	1696232	HSPGR-5288	33	01.01.2016	13:29:21	01.01.2016	3021DIS1	
*	2041					132					
**						488.4					

Figure C.13: BTR release status report

Monitor BTR status

YVR89 → PDF download

BTR details for the plant Matara Bulk Depot-CPC Period: 01.01.2016

Bowsers
Wagons
Return BTR
Cancel BTR

Material	Tns.A	Document	Vehicle No	BTR	Release	Rel.Time	Receipt	Rec.Time	Rel-Rec Dif	Discharge	Dis.Time	Rec-Dis Dif
A0023L99	2045	80323650	H0325-1035	TBWO	01.01.2016	08:01:56		00:00:00		01.01.2016	17:38:46	00:09:36:50
A0013L99	2021	80323733	C0046-6129	TBWO	01.01.2016	09:54:52		00:00:00		01.01.2016	15:26:37	00:05:31:45
A0023L99	2045	80323195	HSPLX-0961	TBWO	01.01.2016	09:59:03	02.01.2016	07:13:00	00:21:13:57	02.01.2016	15:16:33	01:05:17:30
A0013L99	2045	80323665	H0044-8887	TBWO	01.01.2016	10:12:26	02.01.2016	07:13:00	00:21:00:34	02.01.2016	16:50:25	01:06:37:59
A0013L99	2021	80323731	C0067-2406	TBWO	01.01.2016	10:14:25		00:00:00		01.01.2016	16:13:15	00:05:58:50
A0023L99	2045	80323649	HWPGS-5000	TBWO	01.01.2016	10:15:59	02.01.2016	07:13:00	00:20:57:01	02.01.2016	11:02:09	01:00:46:10
A0013L99	2021	80323732	C0067-2040	TBWO	01.01.2016	10:32:21		00:00:00		01.01.2016	14:07:43	00:03:35:22
A0013L99	2021	80323737	C0046-9990	TBWO	01.01.2016	10:47:14		00:00:00		01.01.2016	17:36:51	00:06:49:37
A0017L99	2021	80323725	C0067-2581	TBWO	01.01.2016	11:13:55		00:00:00		01.01.2016	17:39:15	00:06:25:20
A0013L99	2045	80323543	H0067-1355	TBWO	01.01.2016	12:47:57	02.01.2016	07:13:00	00:18:25:03	02.01.2016	18:42:37	01:05:54:40
A0013L99	2045	80323541	H0067-0876	TBWO	01.01.2016	13:25:08	02.01.2016	07:13:00	00:17:47:52	02.01.2016	16:26:27	01:03:01:19
A0023L99	2045	80323197	HSPLX-0808	TBWO	01.01.2016	13:25:55	02.01.2016	07:13:00	00:17:47:05	02.01.2016	15:10:34	01:01:44:39

Figure C.14: BTR detail report

Bowser calling list

YVR84 → Bowser calling list

Bowser Calling List

	Date	Plant	Material	Total Quantity	Quota	13,200	19,800	26,400	33,000
	01.01.2016	3031	A0017L99	22,440.00	3.40	2	1	1	1
	01.01.2016	3031	A0023L99	170,940.00	25.90	13	9	6	5
	01.01.2016	3033	A0017L99	32,340.00	4.90	2	2	1	1
	01.01.2016	3034	A0017L99	32,340.00	4.90	2	2	1	1
	01.01.2016	3035	A0017L99	42,900.00	6.50	3	2	2	1
	01.01.2016	3036	A0017L99	15,840.00	2.40	1	1	1	0
	01.01.2016	3036	A0023L99	136,620.00	20.70	10	7	5	4
	01.01.2016	3038	A0017L99	69,960.00	10.60	5	4	3	2
	01.01.2016	3039	A0017L99	37,620.00	5.70	3	2	1	1
	01.01.2016	3040	A0017L99	26,400.00	4.00	2	1	1	1
	01.01.2016	3041	A0017L99	23,760.00	3.60	2	1	1	1
	01.01.2016	3041	A0023L99	26,400.00	4.00	2	1	1	1
*	01.01.2016			637,560.00	96.60	47	33	24	19
	31.12.2015	3031	A0013L99	79,200.00	12.00	6	4	3	2
	31.12.2015	3031	A0017L99	35,640.00	5.40	3	2	1	1
	31.12.2015	3031	A0023L99	105,600.00	16.00	8	5	4	3
	31.12.2015	3033	A0017L99	51,480.00	7.80	4	3	2	2
	31.12.2015	3034	A0017L99	19,140.00	2.90	1	1	1	1
	31.12.2015	3035	A0017L99	63,360.00	9.60	5	3	2	2

Figure C.15: Bowser calling list

Performance indicators

YVR84 → Key performance indicator

Key Performance Indicator

Date	Plant	Material	Total Quantity	Quota	Used	U13,200	U19,800	U26,400	U33,000	Outboun	KPI
01.01.2016	3031	A0017L99	22,440.00	3.40	3	0	1	0	0	3	
01.01.2016	3031	A0023L99	170,940.00	25.90	24	0	0	6	0	24	
01.01.2016	3033	A0017L99	32,340.00	4.90	0	0	0	0	0	0	
01.01.2016	3034	A0017L99	32,340.00	4.90	4	0	0	1	0	4	
01.01.2016	3035	A0017L99	42,900.00	6.50	0	0	0	0	0	0	
01.01.2016	3036	A0017L99	15,840.00	2.40	0	0	0	0	0	0	
01.01.2016	3036	A0023L99	136,620.00	20.70	0	0	0	0	0	0	
01.01.2016	3038	A0017L99	69,960.00	10.60	0	0	0	0	0	0	
01.01.2016	3039	A0017L99	37,620.00	5.70	5	0	0	0	0	0	
01.01.2016	3040	A0017L99	26,400.00	4.00	4	0	0	1	0	5	
01.01.2016	3041	A0017L99	23,760.00	3.60	0	0	0	1	0	4	
01.01.2016	3041	A0023L99	26,400.00	4.00	4	0	0	0	0	0	
* 01.01.2016			637,560.00	96.60	44	0	1	1	0	4	
31.12.2015	3031	A0013L99	79,200.00	12.00	12	0	0	9	1	44	
31.12.2015	3031	A0017L99	35,640.00	5.40	3	0	1	3	0	12	
31.12.2015	3031	A0023L99	105,600.00	16.00	16	0	0	0	0	3	
31.12.2015	3033	A0017L99	51,480.00	7.80	3	0	1	4	0	16	
31.12.2015	3034	A0017L99	19,140.00	2.90	0	0	0	0	0	3	

Figure C.16: Key performance indicator

YVR84 → Plant performance

Plant Performance

Plant	Material	Total Quantity	Quota	Used	Outboun	KPI	First Shp.	First Load	Day Op	Op.Time
3031	A0017L99	22,440.00	3.40	3	3		10:24:06	11:15:28	✔	09:12:03
3031	A0023L99	170,940.00	25.90	24	24		07:14:17	08:32:14	✔	09:12:03
* 3031		193,380.00	29.30	27	27					
3033	A0017L99	32,340.00	4.90	0	0		00:00:00	00:00:00	✔	07:57:46
* 3033		32,340.00	4.90	0	0					
3034	A0017L99	32,340.00	4.90	4	4		08:20:04	09:56:22	✔	07:53:40
* 3034		32,340.00	4.90	4	4					
3035	A0017L99	42,900.00	6.50	0	0		00:00:00	00:00:00	✔	08:11:27
* 3035		42,900.00	6.50	0	0					
3036	A0017L99	15,840.00	2.40	0	0		00:00:00	00:00:00	✔	09:55:31
3036	A0023L99	136,620.00	20.70	0	0		00:00:00	00:00:00	✔	09:55:31
* 3036		152,460.00	23.10	0	0					
3038	A0017L99	69,960.00	10.60	0	0		00:00:00	00:00:00	✔	07:25:27
* 3038		69,960.00	10.60	0	0					
3039	A0017L99	37,620.00	5.70	5	5		11:35:24	12:43:22	✔	07:15:33
* 3039		37,620.00	5.70	5	5					
3040	A0017L99	26,400.00	4.00	4	4		08:38:38	11:19:59	✔	23:09:45
* 3040		26,400.00	4.00	4	4					
3041	A0017L99	23,760.00	3.60	0	0		00:00:00	00:00:00	✔	07:17:58
3041	A0023L99	26,400.00	4.00	4	4		10:02:21	11:43:41	✔	07:17:58
* 3041		50,160.00	7.60	4	4					
***		637,560.00	96.60	44	44					

Figure C.17: Plant performance

YVR84 → All location performance

All Locations Performance

Plant	Op.Time	First Shp.	Op/Shp Dif	First Load	Shp/Load D	Last Load	Last BTR R	Last Dip A
2021	07:30:00	06:07:25	00:00:00	08:13:12	02:05:47	19:10:53	11:58:09	20:47:08
2031	09:12:03	09:02:55	00:00:00	08:46:02	23:43:07	16:59:19	20:26:06	20:44:47
2033	07:57:46	08:20:42	00:22:56	09:29:36	01:08:54	16:01:59	18:22:36	17:01:39
2034	07:53:40	09:12:33	01:18:53	10:11:21	00:58:48	18:40:19	20:16:43	20:10:08
2035	08:11:27	08:28:32	00:17:05	10:01:50	01:33:18	17:52:09	17:44:56	17:46:41
2036	09:55:31	09:58:41	00:03:10	10:27:39	00:28:58	20:43:44	18:38:04	20:46:21
2037	07:18:21	07:57:49	00:39:28	12:20:20	04:22:31	18:38:02	18:09:19	19:08:00
2038	07:25:27	10:19:33	02:54:06	10:44:52	00:25:19	19:42:32	17:41:35	18:58:35
2039	07:15:33	07:17:30	00:01:57	08:46:24	01:28:54	17:17:10	16:35:10	18:07:00
2040	23:09:45	09:03:15	00:00:00	09:52:23	00:49:08	17:29:06	17:39:15	18:26:03
2041	07:17:58	07:52:06	00:34:08	08:31:45	00:39:39	17:53:36	17:27:33	17:38:48
2043	06:56:43	08:21:52	01:25:09	08:37:25	00:15:33	16:50:45	16:55:03	17:09:34
2045	07:30:00	06:03:58	00:00:00	07:39:57	01:35:59	17:14:52	00:00:00	13:57:10
2078	00:00:00	10:27:33	10:27:33	13:43:49	03:16:16	16:07:46	15:03:51	16:36:44

Figure C.18: All location performance

Sales analysis

YVR97 → Daily sales report

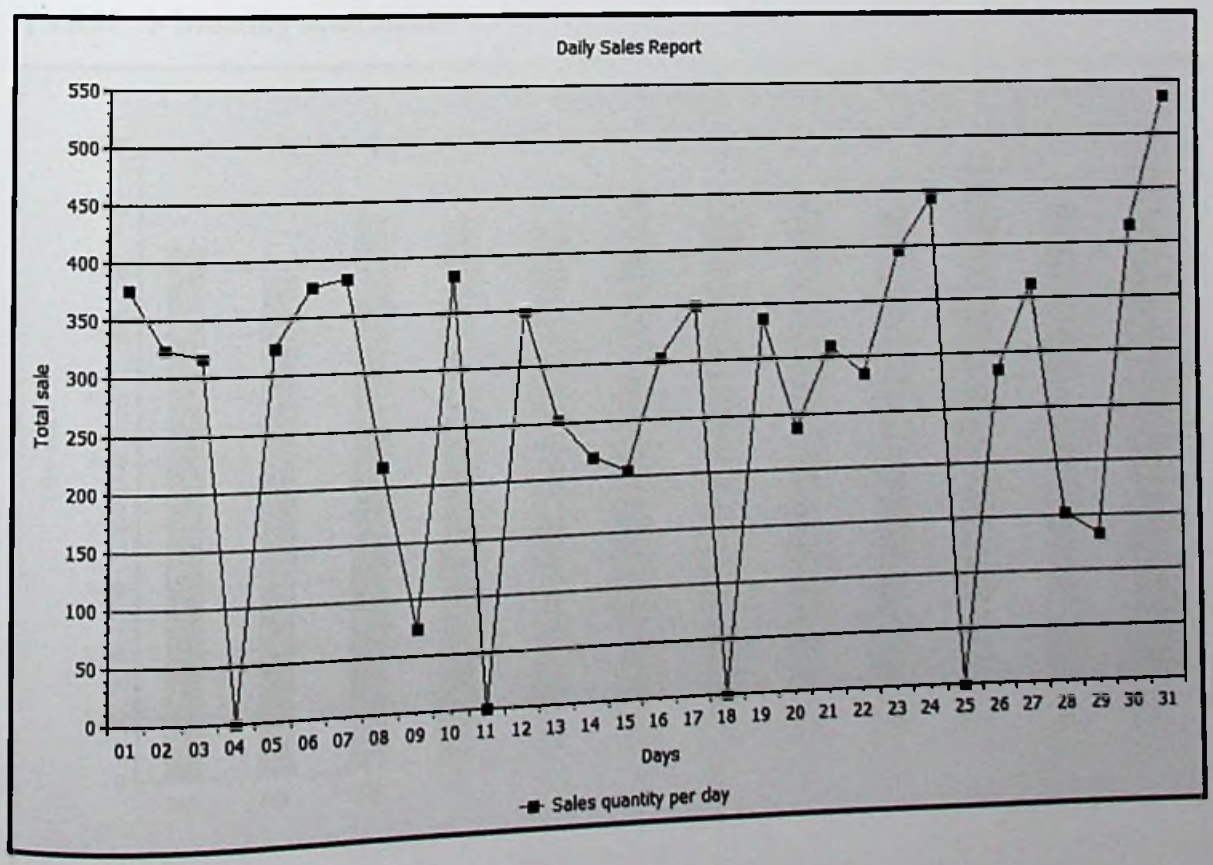


Figure C.19: Daily sales report

Sales Analysis Report

Day	Sales Qty
1	377.003
2	324.06
3	317.582
4	0
5	324.237
6	376.731
7	383.535
8	218.387
9	73.015
10	383.738
11	0
12	350.446
13	251.416
14	218.374
15	205.225
16	304.41
17	350.46
18	0
19	337.494
20	238.132
21	310.728
22	284.682
23	396.777
24	443.336
25	0
26	284.748
27	363.954
28	152.479
29	132.645
30	416.416
31	535.225

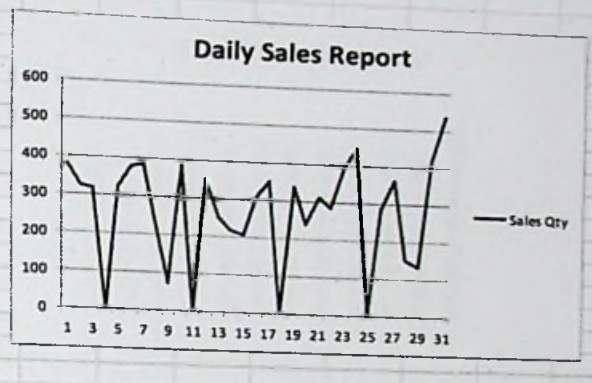


Figure C.20: Daily sales report - excel download

YVR97 → Monthly sales report

Figure C.21: Monthly sales report

Sales Analysis Report

Month	Sales Qty
JAN	8355.235
FEB	7902.853
MAR	8806.293
APR	8724.101
MAY	8632.356
JUN	8981.001
JUL	8727.481
AUG	8883.986
SEP	8865.76
OCT	9554.105
NOV	8819.356
DEC	9983.625

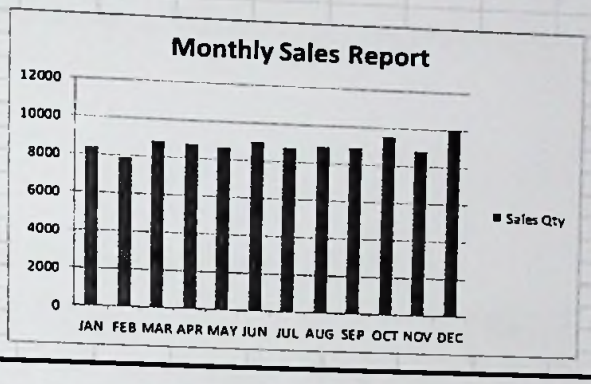


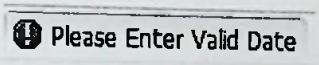
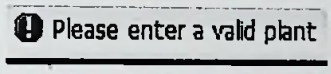
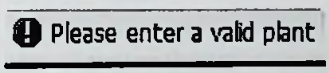
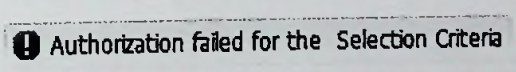
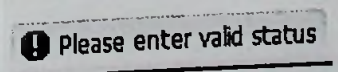
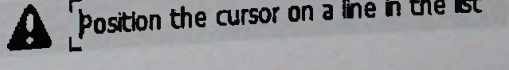
Figure C. 22: Monthly sales report - Excel download

Appendix D – Test Results

Most important test cases at the evaluation stage along with test results are contained in this chapter.

Test results for vehicle status

The test cases with results belong to vehicle status is displayed by table D.1.

ID	Test step	Expected result	Status
1	Click 'Display/Update' button without creation date	Display an error message "Please Enter Valid Date" 	✓
2	Click 'Display/Update' button without plant	Display an error message "Please enter a valid plant" 	✓
3	Click 'Display/Update' button without valid plant	Display an error message "Please enter a valid plant" 	✓
4	Click 'Display/Update' button with correct creation date and the un-authorized plant	Display an error message "Authorization failed for the selection criteria" 	✓
5	Click 'Display/Update' button with valid creation date and the authorized plant	Successfully display the vehicle status details/report.	✓
6	Enter invalid status on 'Status' field and press keyboard enter	Display an error message "Please enter valid status" 	✓
7	Enter valid status on 'Status' field and press keyboard enter	Status will be updated will entered value and change grid color into green	✓
8	Double click on 'Status Description/Status' field copy without position the cursor on the status	Display a warning message "Position the cursor on a line in the list" 	✓

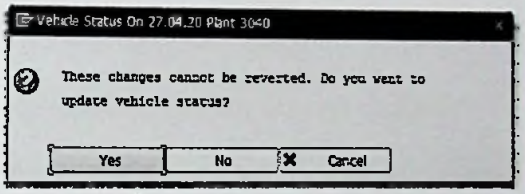
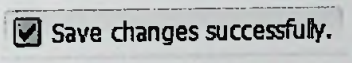
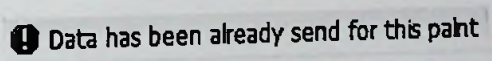
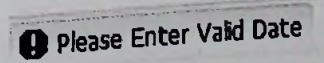
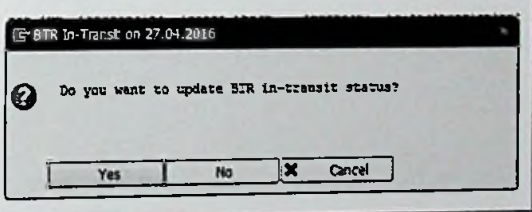
9	Double click on 'Status Description/Status' field and select a status and cancel	No changes will be happening	✓
10	Double click on any other field apart from 'Status Description/Status' field.	Nothing will happen	✓
11	Double click on 'Status Description/Status' field and select a status and copy	Status will be updated will entered value and change grid color into green	✓
12	Change vehicle status and click on 'Update'	Pop-up will be displayed "These changes cannot be reverted. Do you want to update vehicle status?" 	✓
13	Click 'Yes' on above message	Display a success message "Save changes successfully" 	✓
14	Try to update data after sending the stock positions	Display an error message "Data has been already sent for this plant" 	✓

Table D.1: Test results for vehicle status

Test results for BTR In-Transit

The test cases with results belong to BTR In-Transit is displayed by table D.2.

ID	Test step	Expected result	Status
1	Click 'Display/Update' button without creation date	Display an error message "Please Enter Valid Date" 	✓
2	Click 'Display/Update' button without plant	Display an error message "Please enter a valid plant"	✓

		Please enter a valid plant	
3	Click 'Display/Update' button without valid plant	Display an error message "Please enter a valid plant" Please enter a valid plant	✓
4	Click 'Display/Update' button with correct creation date and the un-authorized plant	Display an error message "Authorization failed for the selection criteria" Authorization failed for the Selection Criteria	✓
5	Click 'Display/Update' button with valid creation date and the authorized plant	Successfully display the BTR In-transit data.	✓
6	Click 'Update' without selecting any entry	Display an error message "Select at least one record to update." Select at least one record to update.	✓
7	Select few entries and click on 'Update'	Pop-up will be displayed "Do you want to update BTR in-transit status?" 	✓
8	Click 'Yes' on above message	Display a success message "Save changes successfully" Save changes successfully.	✓
9	Click 'Update' selecting already updated entry/entries	Display an error message "Selected records were already updated." Selected records were already updated.	✓
10	Try to send stock positions without updating	Display an error message "Please update data before sending stock positions." Please update data before sending stock positions.	✓
11	Update stock positions and try to send stock positions without dip posting	Display an error message "Please update data before sending stock positions."	✓

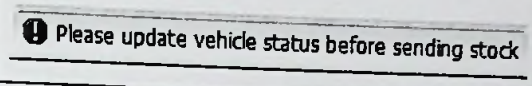
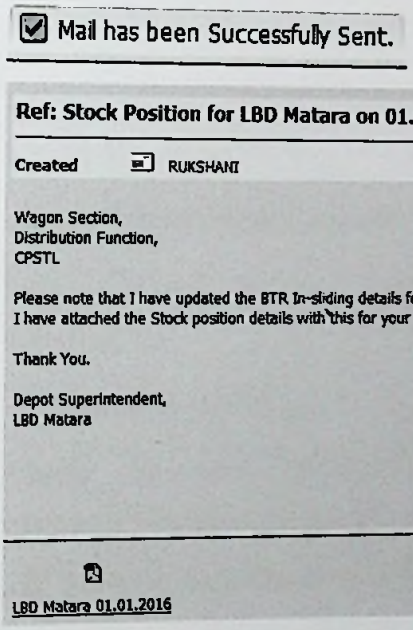
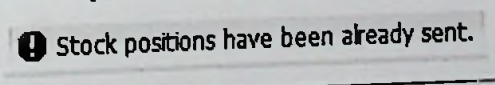
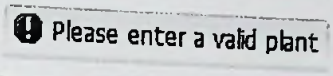
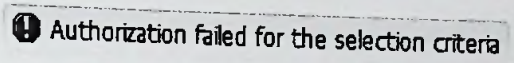
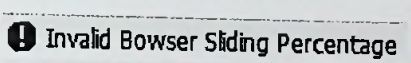
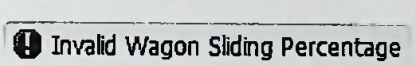
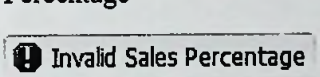
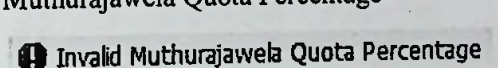
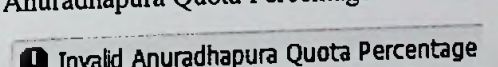
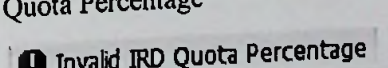
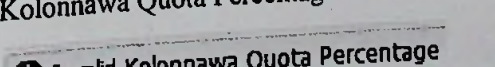
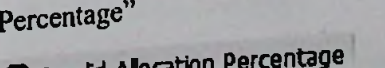
		Please update dip before sending stock positions.	
12	Update stock positions, dip and try to send stock positions without vehicle status update	Display an error message "Please update vehicle status before sending stock." 	✓
13	Update stock positions, dip and vehicle status. Now try to send stock positions	Display a success message "Mail has been successfully sent." And mail will be sent to respective persons with the stock position attachment. 	✓
14	Try to resend the stock positions	Display an error message "Stock positions have been already sent." 	✓

Table D.2: Test results for BTR In-Transit

Test results for stock allocation percentage

ID	Test step	Expected result	Status
1	Click 'Display Data' button without 'Plant' & 'Material'	The grid should display all the data without any filter	✓
2	Click 'Display Data' button with valid 'Plant' & empty 'Material'	The grid should display only the data related to selected plant	✓
3	Click 'Display Data' button with	The grid should display only the data related	✓

	valid 'Plant' & 'Material'	to the selected plant & material	
4	Click 'Display Data' button without valid 'Plant'	Display an error message "Please enter valid plant" 	✓
5	Click 'Display Data' button with unauthorized 'Plant'	Display an error message "Authorization failed for the selection criteria" 	✓
6	Enter 'Bowser sliding percentage' greater than 1000	Display an error message "Invalid Bowser Sliding Percentage" 	✓
7	Enter 'Wagon sliding percentage' greater than 1000	Display an error message "Invalid Wagon Sliding Percentage" 	✓
8	Enter 'Sales percentage' greater than 1000	Display an error message "Invalid Sales Percentage" 	✓
9	Enter 'Muthuragawela quota percentage' greater than 1000	Display an error message "Invalid Muthurajawela Quota Percentage" 	✓
10	Enter 'Anuradhapura quota percentage' greater than 1000	Display an error message "Invalid Anuradhapura Quota Percentage" 	✓
11	Enter 'IRD quota percentage' greater than 1000	Display an error message "Invalid IRD Quota Percentage" 	✓
12	Enter 'Kolonnawa quota percentage' greater than 1000	Display an error message "Invalid Kolonnawa Quota Percentage" 	✓
13	Enter 'Allocation percentage' greater than 1000	Display an error message "Invalid Allcation Percentage" 	✓

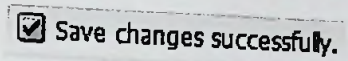
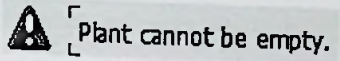
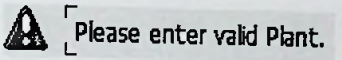
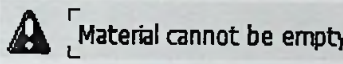
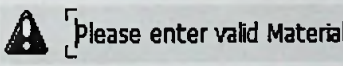
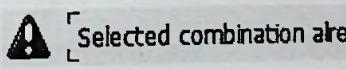
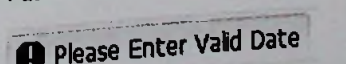
14	Click 'Update' button after making any changes	Display a success message "Save changes successfully" 	✓
15	Try to create/copy with empty plant	Display an error message "Plant cannot be empty" 	✓
16	Try to create/copy with invalid plant	Display an error message "Please enter valid Plant" 	✓
7	Try to create/copy with empty material	Display an error message "Material cannot be empty" 	✓
18	Try to create/copy with invalid material	Display an error message "Please enter valid Material" 	✓
19	Try to create/copy existing entry	Display an error message "selected combination already exist" 	✓
20	Try to create/copy valid entry	The grid should display newly added data	✓
21	Click on 'Edit' button and repeat the test cases 6 to 13	Results should be same as test cases 6 to 13	✓
22	Select an entry and click 'Delete'	Deleted entry should be removed from the grid	✓

Table D.3: Test results for stock allocation percentage

Test results for daily operational quota

ID	Test step	Expected result	Status
1	Click 'Display' button without operational date	Display an error message "Please Enter Valid Date" 	✓
2	Click 'Display' button without	Display an error message "Please enter a	✓

	plant	valid plant" Please enter a valid plant	
3	Click 'Display' button with valid operational date and the authorized plant	Successfully display the daily operational quota details.	✓
4	Try to update quota with negative values	Display an error message "Quota cannot be negative" Quota cannot be negative	✓
5	Try to update quota which exceeds the total amount	Display an error message "Total quantity cannot be > available space" Total quantity cannot be > available space	✓
6	Tyr to update quota before updating BTR sliding	Display an error message "Please update BTR sliding before allocate quota" Please update BTR sliding before allocate quota	✓
7	Update quota with valid values and click 'update' button	Display a success message "Save changes successfully" Save changes successfully.	✓
8	After updating quota, send quota	Display a success message "Mail has been Successfully Sent" Mail has been Successfully Sent.	✓
9	Try to resend quota again	Display an error message "Daily quota has been already sent" Daily quota has been already sent.	✓
10	Try to edit already updated data	Display an error message "Cannot edit already updated data" Cannot edit already updated data	✓

Table D.4: Test results for daily operational quota

Appendix E – Code Listing

Major code segments for anyone who is interested in referring the functionality of the system are contained in this document.

Broadcast messages to customers

This code segment is used to send single or multi line SMS to set of customers at once. Here selection criteria are available for the customer, sales organization, distribution channel and the division.

```
*&---------------------------------------------------------------------*
*& Report  YV_SEND_SMS
*&---------------------------------------------------------------------*
*
*      PROGRAM DESCRIPTION: Sending SMS through SAP
*                DEVELOPER: Rukshani
*            CREATION DATE: 17.10.2014
*               DER NUMBER: DEVK908336
*                    NOTS:  Need special authorizations to send SMS
*                           [ Authority Object 'Z:SPC_AUTH' & YVA_SMS ]
*---------------------------------------------------------------------*
*

*---------------------------------------------------------------------*
* REVISION HISTORY-----------------------------------------------------
*
*        REVISION NO:   R001
*          DEVELOPER:   Rukshani      DATE:   27.11.2014
*       TRANSPORT NO:   DEVK908356
*        DESCRIPTION:   Broadcast message to many users (CPC/LICC)
*---------------------------------------------------------------------*
REPORT  YV_SEND_SMS NO STANDARD PAGE HEADING.

*---------------------------------------------------------------------*
* INCLUDES -----------------------------------------------------------
INCLUDE yixxx_report_standard_sd.

*---------------------------------------------------------------------*
* TABLES - SAP TABLES-------------------------------------------------
TABLES: knvv, sscrfields.

*---------------------------------------------------------------------*
* INTERNAL TABLES-----------------------------------------------------
DATA: BEGIN OF i_knvv OCCURS 0,
  kunnr LIKE knvv-kunnr,
  END OF i_knvv,

  BEGIN OF i_kna1 OCCURS 0,
    kunnr  LIKE kna1-kunnr,
    telf2 LIKE kna1-telf2,
```

```
    END OF i_kna1,

  BEGIN OF i_err_list OCCURS 0,
    kunnr  LIKE kna1-kunnr,
    telf2 LIKE kna1-telf2,
    END OF i_err_list.

DATA: it_note TYPE STANDARD TABLE OF txw_note,
      wa_note LIKE LINE OF it_note,
      i_text TYPE soli_tab,
      wa_text TYPE soli_tab WITH HEADER LINE,
      i_length TYPE so_obj_len.

* FLAGS-----------------------------------------------------------------*
* VARIABLES-------------------------------------------------------------*

* SELECTION SCREEN
SELECTION-SCREEN BEGIN OF BLOCK b1.
SELECTION-
SCREEN BEGIN OF LINE.                          " Body of sms single-
line/multi-line
SELECTION-SCREEN COMMENT (33) FOR FIELD p_body.
PARAMETERS: p_body TYPE  soli-line.
SELECTION-SCREEN PUSHBUTTON 81(4) text-001 USER-COMMAND cli2 .
SELECTION-SCREEN END OF LINE.

SELECTION-SCREEN SKIP.

SELECT-OPTIONS: p_kunnr FOR knvv-kunnr NO INTERVALS,       " Customer
                p_vkorg FOR knvv-
vkorg NO INTERVALS,        " Sales Organization
                p_vtweg FOR knvv-
vtweg NO INTERVALS,        " Distribution Channe
                p_spart FOR knvv-spart NO INTERVALS.       " Division

SELECTION-SCREEN END   OF BLOCK b1.

*---------------------------------------------------------------------*
* INITIALIZATION--------------------------------------------------------
INITIALIZATION.
* Check special authorizations to run the report...
  PERFORM tcode_auth_check.
*---------------------------------------------------------------------*
* AT SELECTION SCREEN---------------------------------------------------
AT SELECTION-SCREEN.
  IF sscrfields-ucomm = 'CLI2'.
    CALL FUNCTION 'TXW_TEXTNOTE_EDIT'
      TABLES
        t_txwnote = it_note.
  ENDIF.
*---------------------------------------------------------------------*
* TOP OF PAGE-----------------------------------------------------------
* END OF PAGE-----------------------------------------------------------
* START OF SELECTION----------------------------------------------------
```

```
* AT USER-COMMAND-------------------------------------------------------*
* END OF SELECTION------------------------------------------------------*

END-OF-SELECTION.
* Check special authorisations to send SMS
  AUTHORITY-CHECK OBJECT 'Z:SEC_AUTH'
                  ID 'ACTVT' FIELD '16'.
  IF sy-subrc <> 0 .
    MESSAGE e225(yv01).
  ENDIF.

* Add text message...
  IF it_note[] IS INITIAL AND p_body IS NOT INITIAL.
    APPEND p_body TO it_note.
  ENDIF.

  IF it_note[] IS INITIAL.
    MESSAGE s000(yv01) WITH 'Please type your message that you wish t
o send'  DISPLAY LIKE 'E'.
    EXIT.
  ENDIF.

  LOOP AT it_note INTO wa_note.
    i_length = i_length + STRLEN( wa_note ).
    MOVE-CORRESPONDING wa_note TO wa_text.
    APPEND wa_text TO i_text.
  ENDLOOP.

* Get customer numbers which are going to send messages
  SELECT kunnr FROM knvv INTO TABLE i_knvv
    WHERE kunnr IN p_kunnr
      AND vkorg IN p_vkorg
      AND vtweg IN p_vtweg
      AND spart IN p_spart.

  SELECT kunnr telf2 FROM kna1 INTO TABLE i_kna1
    FOR ALL ENTRIES IN i_knvv
    WHERE kunnr = i_knvv-kunnr.

  DELETE ADJACENT DUPLICATES FROM i_kna1 COMPARING kunnr.

  PERFORM send_sms.
*&---------------------------------------------------------------------*
*&      Form  SEND_SMS
*&---------------------------------------------------------------------*
*       Call Y_SEND_SMS to send SMS
*----------------------------------------------------------------------*
FORM SEND_SMS.
  DATA: v_sms TYPE yy_sms,
        v_error TYPE string,
        c_send_sms TYPE REF TO y_send_sms,
        i_number TYPE ad_pagnmbr.
```

```
  CREATE OBJECT c_send_sms.

  CALL METHOD c_send_sms->check_access
    RECEIVING
      sms = v_sms.

  IF v_sms = 'X'.
    LOOP AT i_kna1.
      i_number = i_kna1-telf2.
      CALL METHOD c_send_sms->validate_mob_no
        EXPORTING
          I_NUMBER     = i_number
        RECEIVING
          I_NEW_NUMBER = i_number.

      IF i_number IS NOT INITIAL.
        CALL METHOD c_send_sms->send_sms
          EXPORTING
            I_TYPE    = 'RAW'
            I_SUBJECT = ''
            I_TEXT    = i_text
            I_NUMBER  = i_number
            I_LENGTH  = i_length.

        IF sy-subrc <> 0.
          MOVE-CORRESPONDING i_kna1 TO i_err_list.
          APPEND i_err_list.
        ENDIF.
      ELSE.
        MOVE-CORRESPONDING i_kna1 TO i_err_list.
        APPEND i_err_list.
      ENDIF.
    ENDLOOP.
  ELSE.
    MESSAGE e225(yv01).
  ENDIF.

  IF i_err_list[] IS INITIAL.
    MESSAGE s223(yv01).
  ELSE.
    WRITE:/ 'Unable to send message(SMS) to following customer(s)'.
    SKIP.
    WRITE:/ 'Customer', 15 'Phone No'.

    LOOP AT i_err_list.
      WRITE:/ i_err_list-kunnr, 15 i_err_list-telf2.
    ENDLOOP.
  ENDIF.
                                  " SEND_SMS
ENDFORM.
```

Stock planning

This is the code segment used in stock planning report. This is an include file and which used in BTR In-Transits and daily quota allocations too.

```
*&---------------------------------------------------------------------*
*&  Include           YV_STOCK_PLANNING
*&---------------------------------------------------------------------*
* PROGRAM DESCRIPTION: All functions used in YV25, YV26, YV534 & YV52
0
*                DEVELOPER:  Rukshani
*           CREATION DATE:  21.10.2013
*              DEV NUMBER:  DEVK907056
*---------------------------------------------------------------------*
INCLUDE YV_MATERIAL_MAPPING.

*&---------------------------------------------------------------------*
*&       Form  GET_TANK_NUMBERS
*&---------------------------------------------------------------------*
*       Storage Location Data for All Plants & Material
*---------------------------------------------------------------------*
FORM GET_TANK_NUMBERS TABLES p_matnr p_werks.
  CLEAR i_mard.
  REFRESH i_mard.

  SELECT MATNR LGORT WERKS FROM mard
  INTO CORRESPONDING FIELDS OF TABLE i_mard
    WHERE matnr IN p_matnr AND werks IN p_werks AND lgort LIKE 'T0%'.

  DELETE i_mard WHERE ( werks = '3000' OR werks = '3021' OR werks = '
3045' OR werks = '3032' ).

ENDFORM.                    " GET_TANK_NUMBERS
*&---------------------------------------------------------------------*
*&       Form  GET_TANK_CAPACITY
*&---------------------------------------------------------------------*
*       Get tank capacity
*---------------------------------------------------------------------*
FORM GET_TANK_CAPACITY .
```

```
  CLEAR i_oiisock.
  REFRESH i_oiisock.

  LOOP AT i_mard.
    val_full_text = ''.
    val_plant = i_mard-werks.
    val_tank = i_mard-lgort.
    CONCATENATE val_doller val_plant val_tank val_text INTO val_full_
text.

    SELECT SINGLE socnr kapaz
           INTO CORRESPONDING FIELDS OF i_oiisock2
           FROM oiisock
           WHERE socnr = val_full_text.

    IF i_oiisock2-socnr NE ''.
      i_oiisock-matnr = i_mard-matnr.
      i_oiisock-socnr = i_oiisock2-socnr.
      i_oiisock-kapaz = i_oiisock2-kapaz.
      APPEND i_oiisock.
    ENDIF.
    CLEAR i_oiisock2.

  ENDLOOP.

  IF i_oiisock[] IS NOT INITIAL.
    SELECT werks name1
      FROM t001w
      INTO TABLE i_werks
      FOR ALL ENTRIES IN i_oiisock
      WHERE werks = i_oiisock-socnr+2(4).

    DELETE i_werks WHERE ( werks = '3000' OR werks = '3021' OR werks
= '3045' OR werks = '3032' ).

*   Get Dead stock..
    SELECT socnr unpkz maxout FROM oiisocb1
    INTO TABLE i_oiisocb1
    FOR ALL ENTRIES IN i_oiisock
    WHERE socnr = i_oiisock-socnr.
  ENDIF.
```

```
  IF i_werks[] IS INITIAL.
    MESSAGE e015(yv01).
  ENDIF.

ENDFORM.                              " GET_TANK_CAPACITY
*&---------------------------------------------------------------------*
*&       Form  GET_TANK_QTY
*&---------------------------------------------------------------------*
*       Get Tank Quantity
*----------------------------------------------------------------------*
FORM GET_TANK_QTY USING p_date.
  IF i_mard[] IS NOT INITIAL.
    SELECT matnr werks lgort trndate vol_natural FROM yma_outturn
    INTO CORRESPONDING FIELDS OF TABLE i_yma_outturn
    FOR ALL ENTRIES IN i_mard
    WHERE lgort = i_mard-lgort AND
    matnr = i_mard-matnr AND
    werks = i_mard-werks AND
    trndate = p_date AND
    opendip = 'Y'.
  ENDIF.

ENDFORM.                              " GET_TANK_QTY
*&---------------------------------------------------------------------*
*&       Form  GET_TANK_STATUS
*&---------------------------------------------------------------------*
*       Get Tank details which is Under maintains
*----------------------------------------------------------------------*
FORM GET_TANK_STATUS USING p_date.

  IF i_oiisock[] IS NOT INITIAL.
    SELECT werks matnr lgort status
      FROM yvm_lgort_stat
      INTO TABLE i_lgort_stat
      FOR ALL ENTRIES IN i_oiisock
      WHERE werks = i_oiisock-socnr+2(4)
      AND matnr = i_oiisock-matnr
      AND ( date_start <= p_date AND date_finish = '00000000' OR
            date_start <= p_date AND date_finish = '' ).
```

```
  ENDIF.

ENDFORM.                              " GET_TANK_STATUS
*&---------------------------------------------------------------------*
*&      Form  CHANGE_MAT_PLANT
*&---------------------------------------------------------------------*
*       Change material first character A to S and Plant 2 to 3
*----------------------------------------------------------------------*
FORM CHANGE_MAT_PLANT.

  LOOP AT i_werks.
    IF i_werks-werks = '3031'.        "Add KKS Petrol sale to A'pura
      READ TABLE i_matnr WITH KEY matnr = v_safe_map_mat_lp." 'S0011L
99'.  R017
      IF sy-subrc = 0.
        add_kks = 1.
      ENDIF.
    ENDIF.

    IF i_werks-werks = '3078'.        "Add KKS 13/17 sale to IRD
      READ TABLE i_matnr WITH KEY matnr = 'S0013L99'.       " R013
      IF sy-subrc = 0.
        add_kks = 1.
      ENDIF.

      READ TABLE i_matnr WITH KEY matnr = 'S0017L99'.       " R013
      IF sy-subrc = 0.
        add_kks = 1.
      ENDIF.
    ENDIF.

    CONCATENATE '2' i_werks-werks+1(3) INTO p_plant.
                                                    " cpc plant
    r_werks-sign = 'I'.
    r_werks-option = 'EQ'.
    r_werks-low = p_plant.
    APPEND r_werks.

    CONCATENATE '4' i_werks-werks+1(3) INTO p_plant. " lioc plants
    l_werks-sign = 'I'.
    l_werks-option = 'EQ'.
```

```
    l_werks-low = p_plant.
    APPEND l_werks.

    APPEND r_werks TO cpc_lioc_werks.
    APPEND l_werks TO cpc_lioc_werks.
  ENDLOOP.

  IF add_kks = 1.                                                    "R013
    cpc_lioc_werks-sign = 'I'.
    cpc_lioc_werks-option = 'EQ'.
    cpc_lioc_werks-low = '2037'.
    APPEND cpc_lioc_werks.

    cpc_lioc_werks-sign = 'I'.
    cpc_lioc_werks-option = 'EQ'.
    cpc_lioc_werks-low = '4037'.
    APPEND cpc_lioc_werks.
  ENDIF.

  LOOP AT i_matnr.
    CONCATENATE 'A' i_matnr-matnr+1(17) INTO p_mat.
    r_matnr-sign = 'I'.
    r_matnr-option = 'EQ'.
    r_matnr-low = p_mat.
    APPEND r_matnr.
  ENDLOOP.

ENDFORM.                    " CHANGE_MAT_PLANT
*&---------------------------------------------------------------------*
*&      Form  SELECT_MAT_DOCS
*&---------------------------------------------------------------------*
*       Fetch data for given Plant & Material
*----------------------------------------------------------------------*
FORM SELECT_MAT_DOCS USING p_date.

  IF NOT i_matnr[] IS INITIAL.
    SELECT sptag mblnr zeile lifnr
      FROM s571
      INTO TABLE i_s571
      WHERE werks IN r_werks
```

```
      AND matnr IN r_matnr
      AND bwart IN r_bwart
      AND sptag IN r_date
      AND ssour EQ ' '
      AND vrsio = '000'
      AND spmon = ' '
      AND spwoc = ' '
      AND spbup = ' '.


  IF sy-subrc EQ 0.
    SORT i_s571 BY sptag mblnr zeile.
    DELETE ADJACENT DUPLICATES FROM i_s571 COMPARING ALL FIELDS.
  ENDIF.

  IF i_s571[] IS NOT INITIAL.
    LOOP AT i_s571.
      i_mseg_input-mblnr = i_s571-mblnr.
      i_mseg_input-zeile = i_s571-zeile.
      i_mseg_input-mjahr = i_s571-sptag+0(4).
      i_mkpf_input-mblnr = i_s571-mblnr.
      i_mkpf_input-mjahr = i_s571-sptag+0(4).
      APPEND i_mseg_input.
      APPEND i_mkpf_input.
      CLEAR : i_mseg_input,i_mkpf_input.
    ENDLOOP.

    SORT i_mkpf_input BY mblnr mjahr.
    SORT i_mseg_input BY mblnr zeile mjahr.
    DELETE ADJACENT DUPLICATES FROM i_mkpf_input COMPARING ALL FIEL
DS.
    IF NOT i_mkpf_input[] IS INITIAL.
      SELECT mblnr mjahr cpudt budat cputm bldat bktxt tcode
       INTO TABLE i_mkpf FROM mkpf
       FOR ALL ENTRIES IN i_mkpf_input
           WHERE mblnr = i_mkpf_input-mblnr

           AND mjahr = i_mkpf_input-mjahr.
    ENDIF.

    IF NOT i_mseg_input[] IS INITIAL.
```

```
        SELECT mblnr zeile bwart mjahr xauto  matnr umwrk werks lifnr
 menge
         meins smbln smblp sjahr oic_mot ebeln ebelp grund kunnr oic_
truckn
              bwtar oivbeln
           APPENDING  TABLE i_mseg
                    FROM mseg FOR ALL ENTRIES IN i_mseg_input
                                        WHERE  mblnr  = i_mseg_input-
mblnr
                                        AND    mjahr  = i_mseg_input-
mjahr
                                        AND zeile = i_mseg_input-
zeile.
      ENDIF.

      DELETE i_mseg[] where bwart = '641' and xauto <> 'X'.
      DELETE i_mseg[] where bwart = '647' and xauto <> 'X'.
      CHECK i_mseg[] IS NOT INITIAL.

      SORT i_mkpf BY mblnr mjahr.
      LOOP AT i_mseg.
        READ TABLE i_mkpf WITH KEY mblnr = i_mseg-mblnr
                                   mjahr = i_mseg-
mjahr BINARY SEARCH.
        IF sy-subrc = 0.
          MOVE-CORRESPONDING i_mseg TO i_output.
          i_output-werks = i_mseg-werks.
          i_output-reswk = i_mseg-umwrk.
          i_output-budat = i_mkpf-budat.
          i_output-cputm = i_mkpf-cputm.
          i_output-belnr  = i_mseg-oivbeln.
          APPEND i_output. CLEAR : i_output, i_mseg.
        ENDIF.
      ENDLOOP.
    ENDIF.

    i_issue[] = i_output[].
    i_cancel[] = i_output[].
    DELETE i_output[] WHERE bwart = '101'.
    DELETE i_output[] WHERE budat > p_date.
    DELETE i_issue[] WHERE bwart <> '101'.
```

```
    DELETE i_cancel[] WHERE bwart <> '642'.

    PERFORM get_cancelled_doc USING p_date.

  ENDIF.
ENDFORM.                              " SELECT_MAT_DOCS
*&---------------------------------------------------------------------*
*&      Form  FILL_RANGES
*&---------------------------------------------------------------------*
*       Fill ranges used to fetch data
*----------------------------------------------------------------------*
FORM FILL_RANGES USING p_date.

  CALL FUNCTION 'MONTH_PLUS_DETERMINE'      " get data only for 2 mon
ths
    EXPORTING
      MONTHS  = -1
      OLDDATE = p_date
    IMPORTING
      NEWDATE = s_date.

  r_date-low = s_date.
  r_date-high = sy-datum.
  r_date-option = 'BT'.
  r_date-sign = 'I'.
  APPEND r_date.

  r_bwart-sign = 'I'.
  r_bwart-option = 'EQ'.
  r_bwart-low = '641'.
  APPEND r_bwart.
  CLEAR r_bwart.

  r_bwart-sign = 'I'.
  r_bwart-option = 'EQ'.
  r_bwart-low = '642'.
  APPEND r_bwart.
  CLEAR r_bwart.

  r_bwart-sign = 'I'.
```

```
  r_bwart-option = 'EQ'.
  r_bwart-low = '643'.
  APPEND r_bwart.
  CLEAR r_bwart.

  r_bwart-sign = 'I'.
  r_bwart-option = 'EQ'.
  r_bwart-low = '644'.
  APPEND r_bwart.
  CLEAR r_bwart.

  r_bwart-sign = 'I'.
  r_bwart-option = 'EQ'.
  r_bwart-low = '647'.
  APPEND r_bwart.
  CLEAR r_bwart.

  r_bwart-sign = 'I'.
  r_bwart-option = 'EQ'.
  r_bwart-low = '648'.
  APPEND r_bwart.
  CLEAR r_bwart.

  r_bwart-sign = 'I'.
  r_bwart-option = 'EQ'.
  r_bwart-low = '101'.
  APPEND r_bwart.
  CLEAR r_bwart.

ENDFORM.                              " FILL_RANGES
*&---------------------------------------------------------------------*
*&      Form  GET_CANCELLED_DOC
*&---------------------------------------------------------------------*
*       Identify cancelled documents
*----------------------------------------------------------------------*
FORM GET_CANCELLED_DOC USING p_date.

  DATA: BEGIN OF i_canc_doc OCCURS 0,
        smbln LIKE mseg-smbln,
        smblp LIKE mseg-smblp,
```

```
        sjahr LIKE mseg-sjahr,
        END OF i_canc_doc.

  IF NOT i_output[] IS INITIAL.
    SELECT smbln smblp sjahr
     INTO TABLE i_canc_doc
       FROM mseg  FOR ALL ENTRIES IN i_output
        WHERE
        smbln = i_output-mblnr AND
        smblp = i_output-zeile AND
        sjahr = i_output-mjahr .
  ENDIF.

* CRN details
  SELECT invo rplant erdat invnl sy_date FROM yva_crn_inv
  INTO CORRESPONDING FIELDS OF TABLE i_crn_inv
  WHERE erdat IN r_date.

  DELETE i_crn_inv WHERE invo+0(3) <> '007'.

  IF i_crn_inv[] IS NOT INITIAL.
    SELECT vbelv vbeln FROM vbfa
    INTO CORRESPONDING FIELDS OF TABLE i_crn_vbfa
    FOR ALL ENTRIES IN i_crn_inv
    WHERE vbeln = i_crn_inv-invo.
  ENDIF.

  LOOP AT i_output.
    READ TABLE i_canc_doc WITH KEY smbln = i_output-mblnr
                                   smblp = i_output-zeile
                                   sjahr = i_output-mjahr .

    IF sy-subrc = 0.
      i_output-status = 'X'.
    ELSE.
      READ TABLE i_issue WITH KEY belnr = i_output-belnr.
      IF sy-subrc = 0.
        IF i_issue-budat < p_date.
          i_output-status = 'X'.
        ENDIF.
      ENDIF.
    ENDIF.
```

```
    READ TABLE i_cancel WITH KEY belnr = i_output-belnr.
    IF sy-subrc = 0.
      i_output-status = 'X'.
    ENDIF.

* CRN details reference to invoice
    READ TABLE i_crn_vbfa WITH KEY vbelv = i_output-belnr.
    IF sy-subrc = 0 .
      READ TABLE i_crn_inv WITH KEY invo = i_crn_vbfa-vbeln.
      IF sy-subrc = 0 .
        IF i_crn_inv-sy_date < p_date.
          i_output-status = 'X'.                          " Delete
        ELSEIF i_crn_inv-sy_date = p_date.
          i_output-status = 'R'.                          " Return
        ELSE.
          i_output-
status = 'RI'.                        " Return but include stock
        ENDIF.
        i_output-erdat = i_crn_inv-sy_date.
      ENDIF.
    ENDIF.

    MODIFY i_output.

  ENDLOOP.

  DELETE i_output WHERE status = 'X'.
ENDFORM.                              " GET_CANCELLED_DOC
*&---------------------------------------------------------------------*
*&       Form  POPULATE_SHIPMENT
*&---------------------------------------------------------------------*
*        Populate shipments
*----------------------------------------------------------------------*
FORM POPULATE_SHIPMENT USING p_date.

  IF i_output[] IS NOT INITIAL.

    SELECT vbelv vbeln vbtyp_n erdat erzet FROM vbfa
      INTO TABLE i_vbfa
```

```
        FOR ALL ENTRIES IN i_output
        WHERE vbtyp_n IN ('r', 'M', 'h', 'S', 'N') AND          "vbtyp_
n = 'r'          " Transport
              vbtyp_v = 'J' AND
              vbelv = i_output-belnr.

    i_vbfa_inv[] = i_vbfa[].
    i_vbfa_s[] = i_vbfa[].

    DELETE i_vbfa WHERE vbtyp_n <> 'r'.          " Transport
    DELETE i_vbfa_inv WHERE vbtyp_n <> 'M'.      " Delivery

    " Cancel - h - Good issue c, S - Credit memo c, N - Inv.c
    DELETE i_vbfa_s WHERE vbtyp_n = 'r'.         " Transport
    DELETE i_vbfa_s WHERE vbtyp_n = 'M'.         " Delivery

  ENDIF.

  IF i_vbfa_inv[] IS NOT INITIAL.
    SELECT vbeln oic_truckn FROM vbrp
        INTO CORRESPONDING FIELDS OF TABLE i_inv                 "
Get vehicle No..
        FOR ALL ENTRIES IN i_vbfa_inv
        WHERE vbeln = i_vbfa_inv-vbeln.
  ENDIF.

  IF i_inv[] IS NOT INITIAL .
    SELECT vehicle veh_type FROM oigv
      INTO CORRESPONDING FIELDS OF TABLE i_oigv                  " G
et vehicle Type..
      FOR ALL ENTRIES IN i_inv
      WHERE vehicle = i_inv-oic_truckn.
  ENDIF.

  LOOP AT i_output.
    READ TABLE i_vbfa WITH KEY vbelv = i_output-belnr.
    IF sy-subrc = 0.
      i_output-shnumber = i_vbfa-vbeln.
    ENDIF.

    READ TABLE i_issue WITH KEY belnr = i_output-belnr .
```

```
    IF sy-subrc = 0.
      i_output-
status = 'D'.
                                        " Deliver Intransit
      i_output-erdat = i_issue-budat.
      i_output-erzet = i_issue-cputm.
    ENDIF.

    READ TABLE i_vbfa_s WITH KEY vbelv = i_output-
belnr. " Canceled documents..
    IF sy-subrc = 0.
      IF i_vbfa_s-erdat < p_date.
        i_output-status = 'X'.                          " Delete
      ELSEIF i_vbfa_s-erdat = p_date.
        i_output-status = 'C'.                          " Cancel
      ELSE.
        i_output-
status = 'CI'.                      " Cancel but include stock
      ENDIF.
      i_output-erdat = i_vbfa_s-erdat.
      i_output-erzet = i_vbfa_s-erzet.
    ENDIF.

    READ TABLE i_vbfa_inv WITH KEY vbelv = i_output-belnr.
    IF sy-subrc = 0.
      READ TABLE i_inv WITH KEY vbeln = i_vbfa_inv-vbeln.
      IF sy-subrc = 0.
        i_output-vbeln = i_inv-vbeln.
        i_output-vehicle  = i_inv-oic_truckn.
        READ TABLE i_oigv WITH KEY vehicle = i_inv-oic_truckn..
        IF sy-subrc = 0.
          i_output-veh_type = i_oigv-veh_type.
        ENDIF.

* CRN details reference to invoice
        READ TABLE i_crn_inv WITH KEY invo = i_inv-vbeln.
        IF sy-subrc = 0 .
          IF i_crn_inv-sy_date < p_date.                " Delete
            i_output-status = 'X'.
          ELSEIF i_crn_inv-sy_date = p_date.            " Return
            i_output-status = 'R'.
          ELSE.
```

```
            i_output-
status = 'RI'.
                                        " Return but include stock
          ENDIF.
          i_output-erdat = i_crn_inv-sy_date.
        ENDIF.
      ENDIF.
    ENDIF.

    MODIFY i_output.
    CLEAR : i_output, i_vbfa.

  ENDLOOP.

  SORT i_output BY werks matnr belnr.
  DELETE i_output[] WHERE vehicle IS INITIAL.
  DELETE i_output WHERE status = 'X'.

ENDFORM.                    " POPULATE_SHIPMENT
*&---------------------------------------------------------------------*
*&      Form  GET_SALES
*&---------------------------------------------------------------------*
*       Get sales for all plants (2-cpc, 4-lioc) and materials (A)
*----------------------------------------------------------------------*
FORM GET_SALES .
  IF i_vbrk[] IS INITIAL.
    SELECT vbeln FKDAT FKART
        FROM vbrk CLIENT SPECIFIED
        INTO TABLE i_vbrk
        WHERE FKDAT = w_pre_date
        AND mandt = sy-mandt
        AND ( FKART = 'ZODD' OR FKART = 'ZCMR' OR FKART = 'ZFBE')
        AND FKSTO <> 'X'.
  ENDIF.

  IF i_vbrk[] IS NOT INITIAL.
    IF i_vbrp_cpc[] IS INITIAL .
      SELECT vbeln vstel matnr fklmg
          FROM vbrp
          INTO CORRESPONDING FIELDS OF TABLE i_vbrp_cpc
          FOR ALL ENTRIES IN i_vbrk
```

```
            WHERE vbeln = i_vbrk-vbeln
            AND vstel IN r_werks
CPC plants                                                    " 2 -
            AND matnr IN r_matnr.

      SORT i_vbrp_cpc BY vstel matnr.
    ENDIF.

    IF i_vbrp_lioc[] IS INITIAL .
      SELECT vbeln vstel matnr fklmg
        FROM vbrp
        INTO CORRESPONDING FIELDS OF TABLE i_vbrp_lioc
        FOR ALL ENTRIES IN i_vbrk
        WHERE vbeln = i_vbrk-vbeln
        AND vstel IN l_werks                                  " 4 -
LIOC plants
        AND matnr IN r_matnr.

      SORT i_vbrp_lioc BY vstel matnr.
    ENDIF.
  ENDIF.

ENDFORM.                    " GET_SALES
*&---------------------------------------------------------------------*
*&      Form  CHECK_HOLIDAYS
*&---------------------------------------------------------------------*
*       Identify holidays
*----------------------------------------------------------------------*
FORM CHECK_HOLIDAYS  USING    P_W_PRE_DATE
                     CHANGING P_H_INDICATOR.

  CALL FUNCTION 'DATE_CONVERT_TO_FACTORYDATE'
    EXPORTING
      CORRECT_OPTION                = '-'
      DATE                          = p_w_pre_date
      FACTORY_CALENDAR_ID           = 'CS'
    IMPORTING
      WORKINGDAY_INDICATOR          = p_h_indicator
    EXCEPTIONS
      CALENDAR_BUFFER_NOT_LOADABLE = 1
```

```
      CORRECT_OPTION_INVALID        = 2
      DATE_AFTER_RANGE              = 3
      DATE_BEFORE_RANGE             = 4
      DATE_INVALID                  = 5
      FACTORY_CALENDAR_NOT_FOUND    = 6
      OTHERS                        = 7.
  IF SY-SUBRC <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*         WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.
ENDFORM.                    " CHECK_HOLIDAYS
*&---------------------------------------------------------------------*
*&       Form  BTR_STATUS
*&---------------------------------------------------------------------*
*       Get sliding BTR s
*----------------------------------------------------------------------*
FORM BTR_STATUS USING p_date.
  SELECT * FROM yva_btr_status
    INTO TABLE i_btr_status
    WHERE werks IN r_werks
    AND erdat = p_date
    AND matnr IN r_matnr.

ENDFORM.                    " BTR_STATUS
*&---------------------------------------------------------------------*
*&       Form  GET_TODAYS_SALES
*&---------------------------------------------------------------------*
*       Get todays sales
*----------------------------------------------------------------------*
*  -->  p1        text
*  <--  p2        text
*----------------------------------------------------------------------*
FORM GET_TODAYS_SALES USING p_date.

  DATA: r_all_werks TYPE EFG_TAB_RANGES ,
        r_mat TYPE EFG_TAB_RANGES.

  r_all_werks[] = cpc_lioc_werks[].
  r_mat[] = r_matnr[].
```

```
  CALL FUNCTION 'Y_TODAYS_SALES'
    EXPORTING
      P_DATE       = p_date
      P_PRE_DATE   = w_pre_date
      R_ALL_WERKS  = r_all_werks
      R_MATNR      = r_mat
    TABLES
      I_LIKP       = i_likp
      I_COLLECTNS  = i_collectns
      I_SAMEDAY    = i_sameday
      I_CF_BF      = i_cf_bf
      I_LIPS       = i_lips
      I_OUTPUT     = i_result.

ENDFORM.                    " GET_TODAYS_SALES
*&---------------------------------------------------------------------*
*&      Form  GET_PRVDAY_SALES
*&---------------------------------------------------------------------*
*       Get previous day sales
*----------------------------------------------------------------------*
FORM GET_PRVDAY_SALES .
  IF i_sameday[] IS NOT INITIAL.
    SELECT vbeln vbelv FROM vbfa
      INTO CORRESPONDING FIELDS OF TABLE i_sdvbfa
      FOR ALL ENTRIES IN i_sameday WHERE vbeln = i_sameday-vbeln.

    IF i_sdvbfa[] IS NOT INITIAL.
      SELECT vbeln erdat FROM vbak
        INTO CORRESPONDING FIELDS OF TABLE i_sdvbak
        FOR ALL ENTRIES IN i_sdvbfa WHERE vbeln = i_sdvbfa-vbelv.
    ENDIF.
  ENDIF.

ENDFORM.                    " GET_PRVDAY_SALES
*&---------------------------------------------------------------------*
*&      Form  GET_CUSTOMER_NAME
*&---------------------------------------------------------------------*
*       Get customer details
*----------------------------------------------------------------------*
FORM GET_CUSTOMER_NAME .
```

```
  IF i_result[] IS NOT INITIAL.
    SELECT kunnr name1 ort01
      FROM kna1
      INTO TABLE i_kunnr
      FOR ALL ENTRIES IN i_result
      WHERE kunnr = i_result-kunnr.
  ENDIF.

ENDFORM.                    " GET_CUSTOMER_NAME
*&---------------------------------------------------------------------*
*&      Form  GRACE_PERIOD
*&---------------------------------------------------------------------*
*       Initialized grace periods for Delayed PTR Report
*----------------------------------------------------------------------*
FORM GRACE_PERIOD .
  DEFINE append_grace_p.
    i_grace_p-werks = &1.
    i_grace_p-wagon = &2.
    i_grace_p-bowser = &3.
    APPEND i_grace_p.
    CLEAR i_grace_p.
  END-OF-DEFINITION.

  CLEAR i_grace_p.
  REFRESH i_grace_p.

  append_grace_p '2031' 2  1.
  append_grace_p '2033' 3  1.
  append_grace_p '2034' 2  1.
  append_grace_p '2035' 1  1.
  append_grace_p '2036' 3  1.
  append_grace_p '2037' 0 1.
  append_grace_p '2038' 3  1.
  append_grace_p '2039' 1  1.
  append_grace_p '2040' 1  1.
  append_grace_p '2041' 2  1.
  append_grace_p '2042' 2  1.
  append_grace_p '2078' 2  1.

ENDFORM.                    " GRACE_PERIOD
```

```
*&---------------------------------------------------------------------*
*&       Form  GET_DAILY_STATUS
*&---------------------------------------------------------------------*
*       Get daily status for plants
*----------------------------------------------------------------------*
FORM GET_DAILY_STATUS USING p_date.

  SELECT *
       FROM yva_daily_status
       INTO CORRESPONDING FIELDS OF TABLE i_daily_status
       WHERE erdat = p_date.

ENDFORM.                    " GET_DAILY_STATUS
*&---------------------------------------------------------------------*
*&       Form  STOCK_POSITION_DATA
*&---------------------------------------------------------------------*
*       Write stock position details
*----------------------------------------------------------------------*
FORM STOCK_POSITION_DATA USING p_werks p_matnr.
  CONCATENATE '3' p_werks+1(3) INTO ps_werks.
  CONCATENATE 'S' p_matnr+1(16) INTO ps_matnr.
  CONCATENATE '2' p_werks+1(3) INTO pa_werks.
  CONCATENATE 'A' p_matnr+1(16) INTO pa_matnr.

READ TABLE i_oiisock WITH KEY socnr+2(4) = ps_werks matnr = ps_matnr.
  IF sy-subrc = 0.
*      Tank capacity qty. details
  LOOP AT i_oiisock WHERE socnr+2(4) = ps_werks AND matnr = ps_matnr.
       val_tank = i_oiisock-socnr+6(4).

       READ TABLE i_yma_outturn WITH KEY werks = ps_werks matnr = ps_m
atnr lgort = val_tank.
       IF sy-subrc = 0.
         mul_val = i_yma_outturn-vol_natural * 1000.
       ENDIF.

       READ TABLE i_oiisocb1 WITH KEY socnr = i_oiisock-socnr.
       IF sy-subrc = 0.
         unpm_stk = i_oiisocb1-unpkz.
       ENDIF.
```

```
      READ TABLE i_lgort_stat WITH KEY werks  = ps_werks matnr = ps_m
atnr lgort = val_tank.
      IF sy-subrc = 0 .
*          Inactive tanks....
      ELSE.

        READ TABLE i_oiisocb1 WITH KEY socnr = i_oiisock-socnr.
        IF sy-subrc = 0 AND i_oiisocb1-maxout > 0.
          auth_tot_qty = i_oiisocb1-maxout.
        ELSE.
          auth_tot_qty = i_oiisock-kapaz.
        ENDIF.

        avl_cap = auth_tot_qty -
 mul_val.                    " Autorized Capacity
        tot_vol_natural = mul_val + tot_vol_natural.        " Availab
le Qty
        val_tot_qty = auth_tot_qty + val_tot_qty.           " Total T
ank Capacity
        tot_unpm_stk = unpm_stk + tot_unpm_stk.             " Unpumpa
ble Qty
        tot_avl_cap = avl_cap + tot_avl_cap.                " Availab
le Capacity = Tank Capacity - Ava Qty

        CLEAR: val_tank, mul_val, unpm_stk, avl_cap, auth_tot_qty.
      ENDIF.

    ENDLOOP.
    diff_qty = val_tot_qty - tot_vol_natural.    " Available Capacity

*     Todays sales
    IF pa_matnr = v_map_mat_lp.  " 'A0011L99'. R002
      v_matnr = v_map_mat_xlp.   " 'A0041L99'. R002
    ELSEIF pa_matnr = 'A0013L99'.
      v_matnr = 'A0043L99'.
    ELSE.
      v_matnr = ''.
    ENDIF.

    LOOP AT i_result WHERE werks+1(3) = p_werks+1(3) AND ( matnr = pa
```

```
_matnr OR matnr = v_matnr ).
      IF i_result-alert = 'CF' OR i_result-alert = 'CC' OR i_result-
alert = 'NC'.
        IF i_result-werks+0(1) = '2'.
          day_sales_cpc = day_sales_cpc + i_result-lfimg.
        ELSE.
          day_sales_lioc = day_sales_lioc + i_result-lfimg.
        ENDIF.
      ENDIF.
    ENDLOOP.

    day_sales_cpc = day_sales_cpc * 1000.
    day_sales_lioc = day_sales_lioc * 1000.
    day_sales_qty = day_sales_cpc + day_sales_lioc.

*     Previous day sales
    LOOP AT i_vbrp_cpc WHERE vstel+1(3) = p_werks+1(3) AND ( matnr =
pa_matnr OR matnr = v_matnr ).
      prv_sales_cpc = prv_sales_cpc + i_vbrp_cpc-fklmg.
    ENDLOOP.

    LOOP AT i_vbrp_lioc WHERE vstel+1(3) = p_werks+1(3) AND ( matnr =
 pa_matnr OR matnr = v_matnr ).
      prv_sales_lioc = prv_sales_lioc + i_vbrp_lioc-fklmg.
    ENDLOOP.

    prv_sales_cpc = prv_sales_cpc * 1000.
    prv_sales_lioc = prv_sales_lioc * 1000.
    prv_sales_qty = prv_sales_cpc + prv_sales_lioc.

*   Append data to internal
    i_stock_pos-matnr = p_matnr.
    i_stock_pos-tank_cap = val_tot_qty.
    i_stock_pos-ava_qty = tot_vol_natural.
    i_stock_pos-unpm_qty = unpm_stk.
    i_stock_pos-ava_cap = tot_avl_cap.
    i_stock_pos-day_sales_cpc = day_sales_cpc.
    i_stock_pos-day_sales_lioc = day_sales_lioc.
    i_stock_pos-prv_sales_cpc = prv_sales_cpc.
    i_stock_pos-prv_sales_lioc = prv_sales_lioc.
    APPEND i_stock_pos.
```

```
    CLEAR i_stock_pos.

    CLEAR: val_tot_qty, tot_vol_natural, tot_unpm_stk, tot_avl_cap, d
iff_qty.
    CLEAR: day_sales_qty, day_sales_cpc, day_sales_lioc.
    CLEAR: prv_sales_qty, prv_sales_cpc, prv_sales_lioc.

  ENDIF.
ENDFORM.                    " STOCK_POSITION_DATA
*&---------------------------------------------------------------------*
*&      Form  GET_VEH_STATUS
*&---------------------------------------------------------------------*
*       Get vehicle status for plants
*----------------------------------------------------------------------*
FORM GET_VEH_STATUS USING p_date.
  SELECT werks vehicle veh_maxvol veh_status age
    FROM yva_veh_status
    INTO CORRESPONDING FIELDS OF TABLE i_veh_status
    WHERE erdat = p_date.

  SORT i_veh_status BY vehicle veh_status.
ENDFORM.                    " GET_VEH_STATUS
*&---------------------------------------------------------------------*
*&      Form  STATUS_DESC
*&---------------------------------------------------------------------*
*       Get vehicle status description
*----------------------------------------------------------------------*
FORM STATUS_DESC.
* Get status description
  DATA n TYPE n VALUE 0.

  DO 6 TIMES.
    IF n = 0.
      veh_status = ''.
    ELSE.
      veh_status = n.
    ENDIF.

    CALL FUNCTION 'QC04_DOMAIN_TEXT_GET'
      EXPORTING
```

```
        i_domain_name = 'OIG_VEHSTA'
        i_language    = sy-langu
        i_domvalue_l  = veh_status
      IMPORTING
        e_ddtext      = status_text
      EXCEPTIONS
        no_data_found = 01.

    i_status_text-veh_status  = veh_status.
    i_status_text-status_text = status_text.
    APPEND i_status_text.
    CLEAR: i_status_text, status_text.
    n = n + 1.
  ENDDO.
ENDFORM.                          " STATUS_DESC
*&---------------------------------------------------------------------*
*&       Form  DAILY_QUOTA
*&---------------------------------------------------------------------*
*       Get Daily Quota details..
*----------------------------------------------------------------------*
FORM DAILY_QUOTA  USING  P_DATE.

  SELECT * FROM yva_daily_quota
   INTO TABLE i_yva_quota
     WHERE erdat = p_date AND matnr IN r_matnr.

  DELETE i_yva_quota WHERE uname = 'BASIS_BKG'.
ENDFORM.                          "DAILY_QUOTA
*&---------------------------------------------------------------------*
*&       Form  GET_STK_ALL_PER
*&---------------------------------------------------------------------*
*       DEVK907336: Get Stock allocation percentages - RS 23/12/2013
*----------------------------------------------------------------------*
FORM GET_STK_ALL_PER.

  SELECT * FROM yva_stk_all_per INTO TABLE i_stk_all_per.

ENDFORM.                          "GET_STK_ALL_PER
```

GLOSSARY

Dip - Physical quantity measurement of Tanks.

BTR - Bulk transfers. Transferring bulk loads from the main installation to depot or one depot to another.

BTR In-Transit - BTRs departed from the source location (main installation or depot) however, not yet came to the destinations (depot). That is loads are in the middle of the transporting process.

BTR In-Slides/sliding - BTRs parked/parking at the depot premises.

Stock position – Stock position is detail about the depot stocks. This consists of stock details, bowser positions and the staff availability. The ultimate objective is to identify the space availability of the tanks. Distribution manager makes decisions based on stock positions when allocating daily quota for the plant/depot.

Daily quota - Allocate quota with respect to the space availability, which is going to send from the main installation to the depot. With the quota allocation, restrict depots from over exceeding tanks capacities. Therefore, tanks can be top-up the with respect to the space availability.

Shipment creation - Link between transportation modes (bowser/wagon) and the outbound (transfer document). That is assign vehicle which is going to transfer the bulk loads. But here can transfer can be done up to the allocated amount (i.e Quota * 6600).

Sales order book - Is the document describes the depot daily demand and the expected sales for the day.