Automation Framework Development for Oracle Retail Products

Submitted By

Disha V. Bhatt

15MCEC04

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2017

Automation Framework Development for Oracle Retail Products

Major Project

Submitted in fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering

Submitted By

Disha V.Bhatt

(15MCEC04.)

Guided By

Prof. Vipul Chudasama

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2017

Certificate

This is to certify that the major project entitled **"Automation Framework Development for Oracle Retail Products"** submitted by **Disha Bhatt (Roll No: 15MCEC04)**, towards the fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering of Nirma University, Ahmedabad, is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Vipul Chudasama
Guide & Assistant Professor
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Priyanka Sharma
Professor,
Coordinator M.Tech - CSE
Institute of Technology,
Nirma University, Ahmedabad

Dr. Sanjay Garg
Professor and Head,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr Alka Mahajan
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Disha Bhatt**, Roll. No. **15MCEC04.**, give undertaking that the Major Project entitled "**Automation Framework Development for Oracle Retail Products**" submitted by me, towards the fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

_____

Signature of Student

Date:

Place:

Endorsed by

Mr.Vipul Chudasama

(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Prof. Vipul Chudasama**, Assistant Professor, Computer Science Department, Institute of Technology, Nirma University, Ahmedabad for his valuable guidance and continual encouragement throughout this work. The appreciation and continual support he has imparted has been a great motivation to me in reaching a higher goal. His guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

It gives me an immense pleasure to thank **Dr. Sanjay Garg**, Hon'ble Head of Computer Science and Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his kind support and providing basic infrastructure and healthy research environment.

A special thank you is expressed wholeheartedly to **Dr Alka Mahajan**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Disha Bhatt

15MCEC04.

Abstract

In today's I.T. industry,mainly Retail Market, large number of retail products are developed every day to perform manual jobs such as billing,calculating stocks and ordering new ones. Once these products are developed and tested to a satisfactory level, tedious tasks of deploying this product on customers site comes into picture. Retail product companies has big customers like "Shoppers Stoppe" and "HyperCity". These retail products are to be deployed across all the branches of the customers site.These branches may be spread across the globe. As a result of which it is impossible for a human to sit and install these products on each machine customer needs. Also these products uses layered approach i.e. it needs various components like database, dashboards,workbooks etc.Thousands of code files working with each other in proper synchronization makes such products successful.As a result of which "Automation" comes into picture. An automation can create a installer for such huge products and install them parallely on thousands of hosts. The automated installer performs checks at various checkpoints to make sure that all other components required to proceed with installation are present and up on the hosts.If not,it also installs them and then proceeds. An automation script can also be helpful in various other tasks such as compiling large number of code files, or running quality checks against them.This will save a lot of time,money and effort required to perform those jobs manually.

Abbreviations

RPAS	Retail Predictive Application Server.
FC	Fusion Client.

Contents

List of Figures

Chapter 1

Introduction

1.1 Project Definition

Automation Framework Development for Oracle Retail RPAS [Retail Predictive Application Server] Fusion Client.

1.2 Motivation

When some new features of software application is introduced, new build is given to the customers, but before delivery of new software product to customer , it undergoes the process of testing. Performing Testing manually during each and every release increases production cost as well as a tiresome process and also it depends upon the skills of individuals which affects the quality of software product.

Solution is to perform these tasks using scripts which give many advantages like :

- Reduces Production time-Integration systems executes the Job of automation and so it can also run job during nights as well for long hours. System triggers the automated task so production time will be reduced.
- Reduces human errors-Machines performing the repeated tasks are less likely to make errors com-pared to Humans.
- Increases Productivity-Reduction in Production time and Errors will automatically increases Pro-ductivity.

- Re-usability-Automated scripts can be reused to perform similar kind of tasks.
- Accuracy and High Quality-Decreased errors will improve final product accuracy and Quality.

1.3 Terms in Definition

1.3.1 Automation

Automation is defined as"A set of recorded steps re-played to perform same task again".The project performs automation of [**?**]:

- Pulling of code from source control
- Compiling Code
- Packing the artifacts
- Enable deploying of package using WebLogic
- Running varoius Testscripts against them

Advantages of Automation

- Automation is process of allowing repetitive and tedious tasks to be done by itself
- Some tedious tasks can be made independent using automation
- Economic method for manual testing.Reduces cost of maintainence
- Quality check can be kept for softwares using automation scripts

Chapter 2

Literature Survey

2.1 Overview of RPAS

The RPAS Fusion Client is a web-based rich client for the Retail Predictive Application Server (RPAS) platform developed using the latest Oracle Application Development Framework (ADF). Planning is one of the most important and complex processes in a retail business. It typically involves a detailed set of activities that need to be followed as part of a workflow. Unlike the RPAS Windows-based Classic Client, the Fusion Client includes a taskflow feature that provides a robust workflow capability to make each planning activity easier to track and maintain. The Fusion Client uses the same RPAS server as the RPAS Windows-based Classic Client. In addition to the enhanced user experience, the Fusion Client provides access to a larger number of users and a greater degree of platform independence. The taskflow also allows you to switch between solutions and domains without logging out and back in.

2.2 RPAS Platform Components

1. RPAS Server

   RPAS server consists of two main components :

   (a) RPAS Domain

   (b) Domain Daemon

2. RPAS Client

3. Configuration Tools

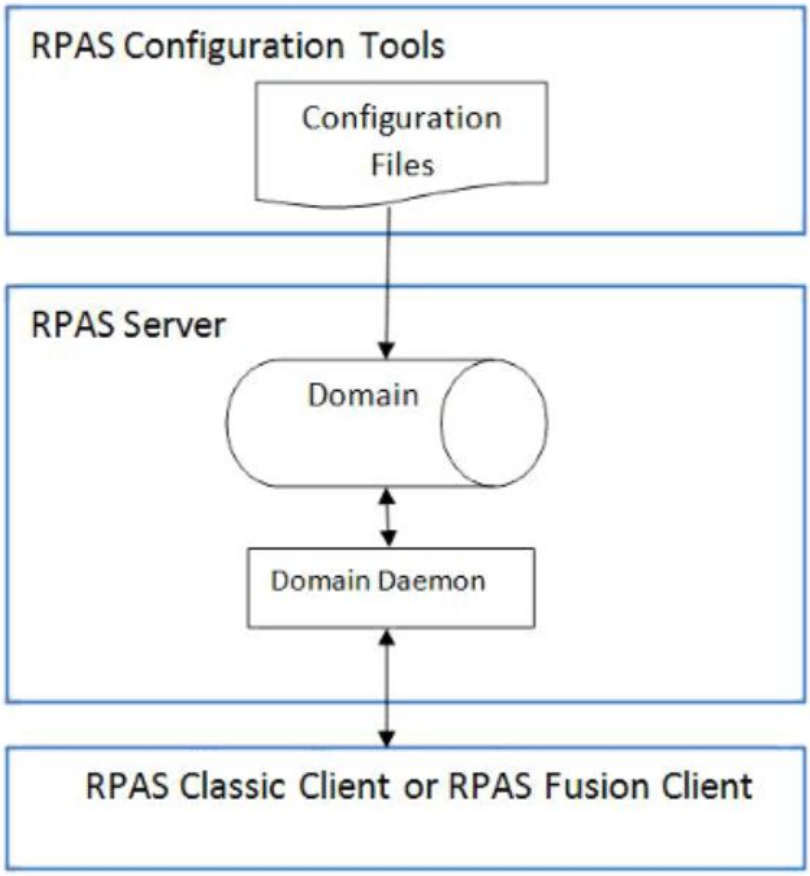

Figure 2.1: RPAS Platform Components

2.3 RPAS Server

- Main Component of RPAS server is :

  1. RPAS Domain
  2. RPAS Domain Daemon

- RPAS Domain are actually the server side files and directories that houses data used by RPAS database.
- RPAS Domain uses Multi-dimensional database to store data used by RPAS.
- RPAS Server also stores the "Platform Code base" that provides a set of utilities to interact with the domain.

2.3.1 RPAS Domain Daemon

Basically,RPAS domain daemon is a middle ware between RPAS client and Domain.It can be defines as follows:

- Process that enables communication channel between RPAS Client and RPAS Domain
- It is a server side utility that will wait for request from Client[RPAS Client] on specific port
- After request is received,server process is started to which client connects to

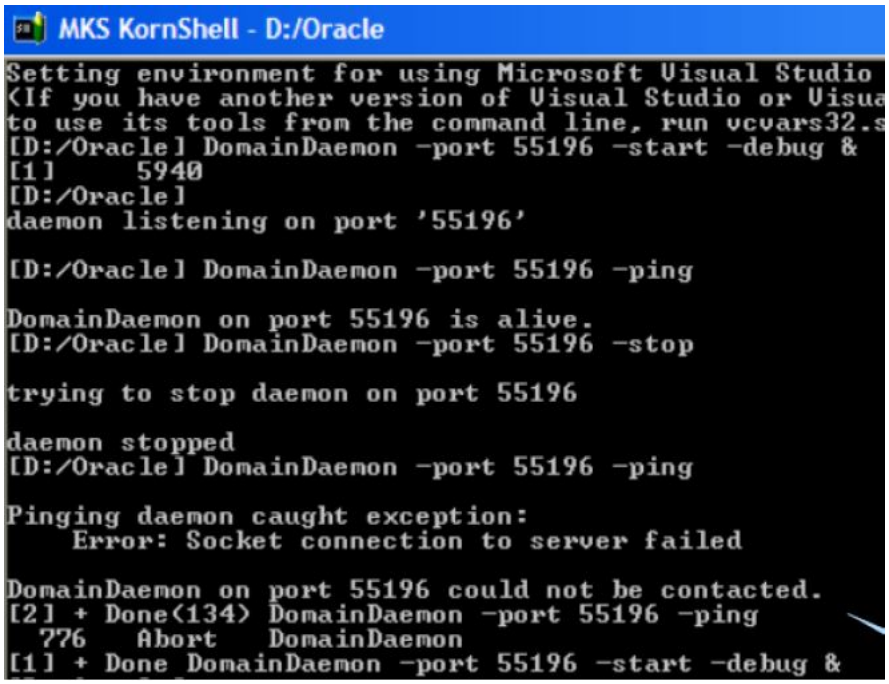

Figure 2.2: Domain Daemon

2.3.2 Multi-Dimensional Databases

Consider following example that shows data representation in two dimensions : It shows "Sales Volume for a Printing Company"

Forming a multi-dimensional view of the two dimensional relation :

Following is the analysis of Multi-dimensional database:

- Data Arrays are used to store data in Multi-Dimensional Databases
- It stores data in the form of Multi-Dimensional matrix

Product	Location	Sales
Copy Paper	Store A	47
Copy Paper	Store B	3
Photo Paper	Store A	12
Photo Paper	Store B	44

Figure 2.3: Multidimension Database

- Each axis is internally defined to be of a specific type of data.
- In the above figure,,X axis represents Location, Y axis shows Product name while the values in cells are the actual sales value
- Here only 4 cells are used to store 4 values

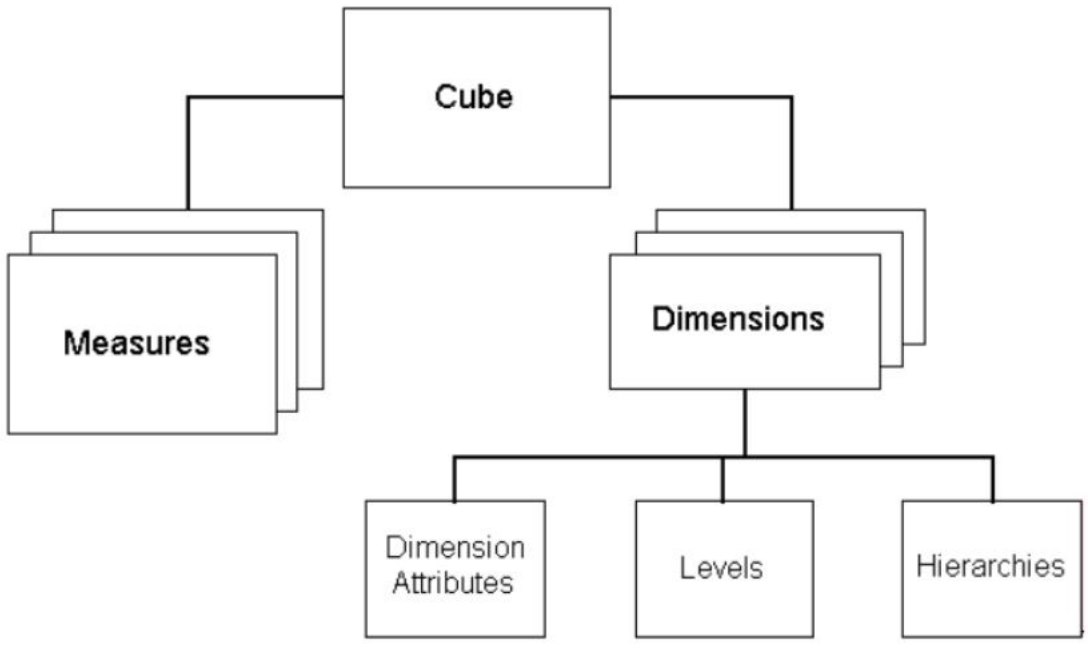

Figure 2.4: Multi-Dimensional View

DATA ARRAYS

Data Arrays are :

- RPAS database uses Data Arrays to store data

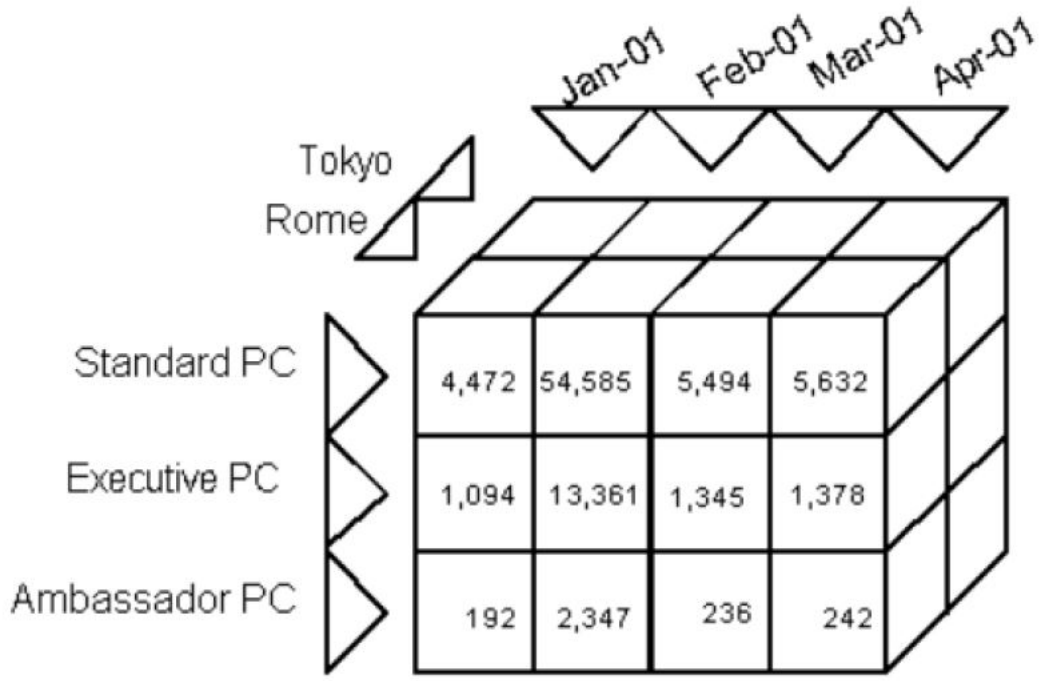

Figure 2.5: Multi-Dimensional Data Cube

- Each axis in multi-dimensional data array is called a "Dimension"
- Above example has 2 axis - Location , Products.
- Each and every entries within the dimension is called "Position"
- In above example as shown dimension Location has two positions named as Store A and Store B

HIERARCHIES

- "Top-to-bottom" set-up of Super-sub relationships between dimensions and measures
- A hierarchy is away to show multi-views of data
- Aggregation means in viewing data, analysts uses hierarchies to understand patterns at one level, moves to lower levels to conclude reasons for these patterns, and go back to higher levels to make out what effect this patterns can have on larger scale
- For Example,in the above data cube figure, plan is made according to months. This all plans can be combined for an annual plan.

AGGREGATION AND SPREADING

- Aggregation and Spreading are used to define relationships between dimensions at each level
- In above example,retailer can plan at a lower dimension "week" and this plans can easily be aggregated at higher level "month"
- Similarly, retailer can also plan at a high level "year" and plans can be spread down to lower level "months"

POSITIONS

- Each dimension within a hierarchy is made up of any number of positions.Each child dimension must roll up to a parent dimension
- Similarly, every position in the child dimension must also roll up to positions in parent dimension
- Each child position goes up to only one parent dimension but a parent position can contain more than one child position. "One-to-Many" relationship

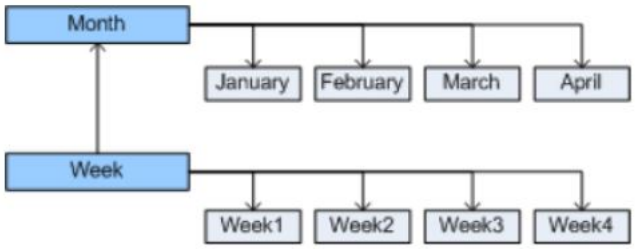

Figure 2.6: Positions

2.4 RPAS Client

RPAS Client can be defined as:

- RPAS Client allows user to access and manipulate data in the domain of RPAS Server through a windows GUI
- RPAS Client allows user to build a personal scaled down version of domain using user-defined data limitations.
- These personal copies offers various views of data stored in the domain database.

2.5 RPAS Configuration Tools

RPAS Configuration tools can be defined as :

- Provides a flexible mean to build and configure RPAS based applications with customer specific parameters
- Provides a user-friendly,streamlined interface to utilize RPAS functionality
- A template can also be created so that same parameters can be used in other modifications.

Chapter 3

HUDSON Continuous Integration System

3.1 Hudson Basics

The term **"Continuous Integration system"** means a system that can integrate changes into the presently available code and re ect its changes dynamically. It can also notify all the users of presently available system about the changes. The main job of a continuous integration system is to monitor the execution of repeated job.

Hudson performs two basic jobs as follows[5]:

- Testing and Building of software projects continuously : It makes easier for a developer to integrate changes into the product. It also helps users to obtain a fresh build every time.This can lead to increase in productivity.
- Monitoring execution of externally running jobs : One can receive regular e-mails that captures output of currently running builds. Hudson can capture those output and notify a user when something is wrong.

Features of Hudson:

1. Installation[6] : Installing Hudson is an easy process. A user needs to download a latest "jar" and run following command in windows command prompt:
   java - jar <absolutepathtojar >- - httpPort <Port No >
   The default port used by Hudson is "8080".One can customize the port number using argument "httpPort".

2. Ease of Configuration: Hudson can be easily configure using its GUI. There are various options in Hudson GUI to create a new job,add a new slave,con

   gure plugins and nodes etc.

3. Shows a summary of changes: Hudson can summarize all the changes that were made to a particular build.

4. Links to builds: Hudson can create links like "Last successful build". If some changes goes wrong than such links can be used to restore the build to a work- ing version.

5. Plugin support: Hudson also provides third party plugins for various purposes.

6. Customizable environments: Hudson allows a user to mention customizable paths for environment variables like JAVA,ANT while con

   guring a job.

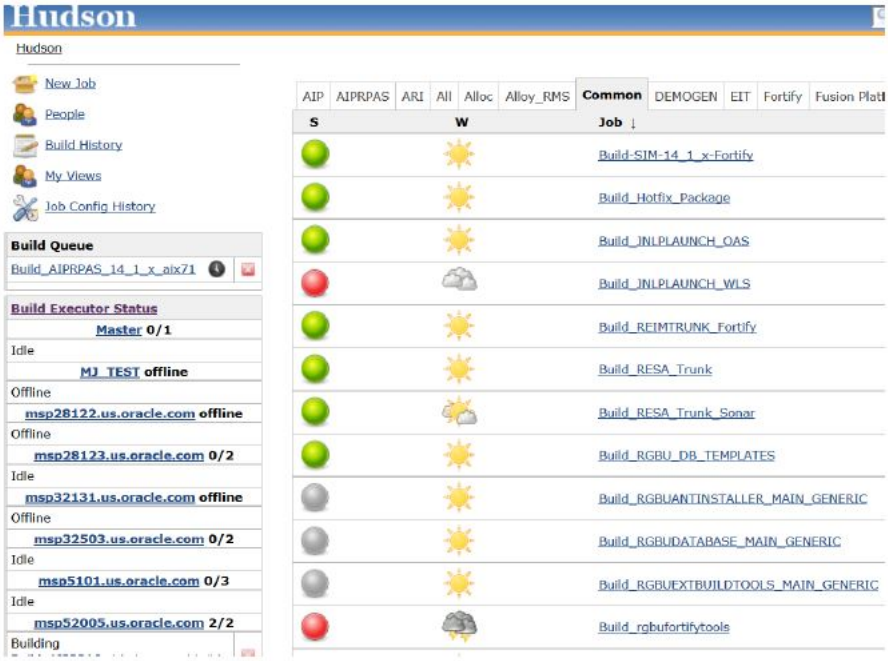

Figure 3.1: Hudson GUI Snapshot

3.1.1 Hudson Master Slave Configuration

Hudson has a "master-slave" configuration. The tasks to perform for a working Hudson slave are as follows:

1. Check if JAVA jdk is higher than version 1.6 using command "java -version" in windows cmd.

2. Install Hudson master using "java -jar" command.

3. Once Hudson master is installed with a message "successfully installed", open a browser and type "http://localhost:(port no)".By default it is 8080.

Install Hudson master as a windows service

One needs to install Hudson master as a start up service of windows so that it can start without typing the above link in browser next time windows is loaded. The steps to follow are :

1. Once Hudson is started, click the manage Hudson link and look for the "Install as Windows Service" link on the management page.

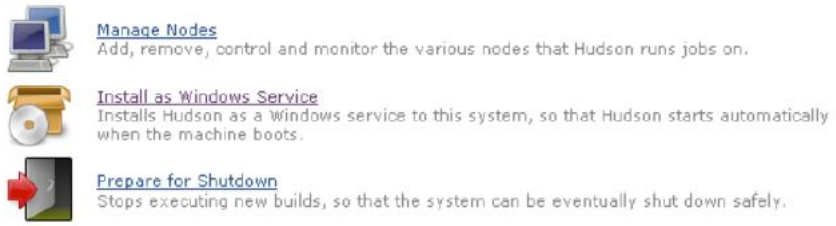

Figure 3.2: Hudson UI Snapshot

2. Once clicked, it guides us to a installation screen as follows:

Figure 3.3: Hudson Installation Screen

3. Choose an already created directory on a local disk where Hudson gets installed.This becomes "HUDSON HOME" and will be used to store data files and programs.

4. Once Hudson is successfully installed as a windows service, an admin needs to restart the Hudson server. A page like below appears:

5. To check if all the steps have been executed successfully, go to "start panel -¿ Services". A page like below appears:

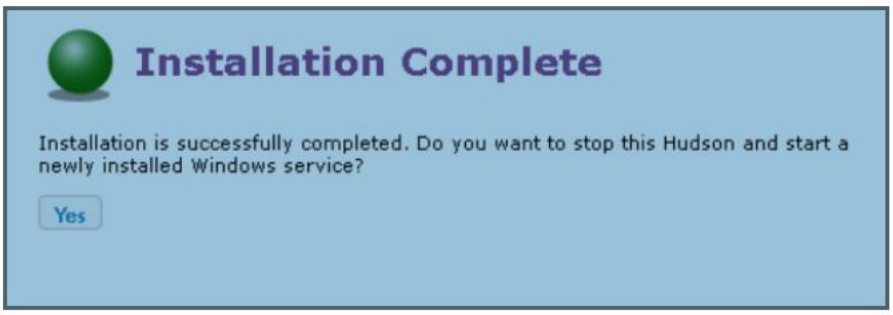

Figure 3.4: Hudson Restart Screen

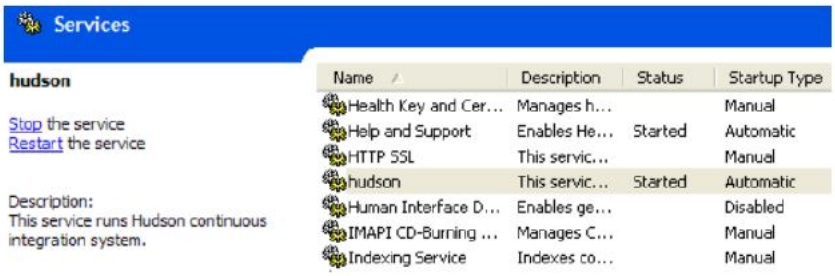

Figure 3.5: Start up services on windows

Configure and Install Hudson Slave

Hudson also allows us to configure a slave node on which jobs can be run. To configure a slave follow steps :

1. On master Hudson click Manage Jenkins - >Manage Nodes.A screen as following appears:

2. Click New Node - >Enter Node Name.

3. Click Dumb Slave and press OK on a screen as follows:

4. Fill details as follows. Here, Remote FS Root is "a slave directory on master Hudson".Give proper already existing path on a local machine that is being configured as a slave of master Hudson.

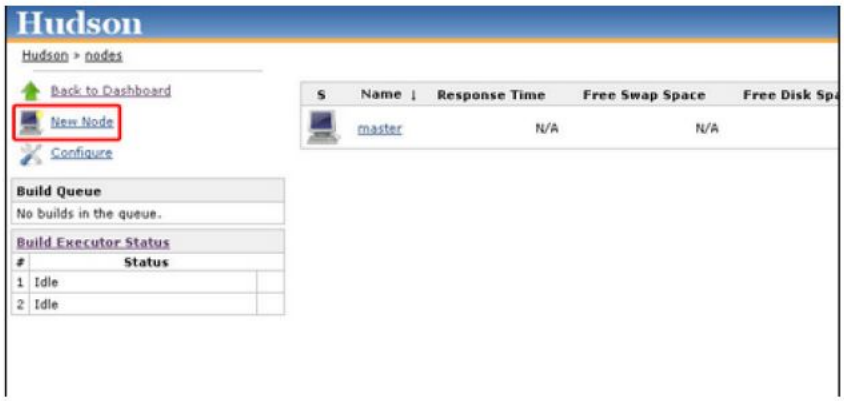

Figure 3.6: Configuring Hudson Slave

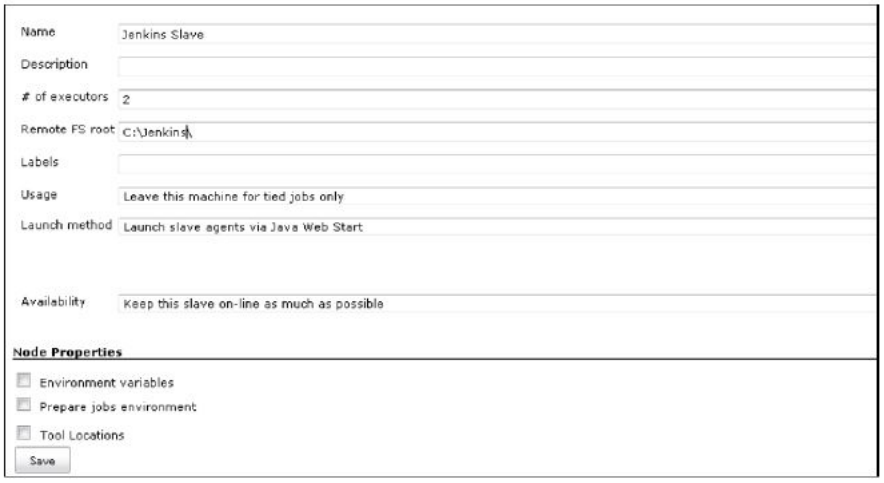

Figure 3.7: Configuring Hudson Slave

5. Connect this newly created machine to master as follows:

   - Launch a browser on slave machine.
   - Go to URL "http://localhost:(portno)"
   - Go to Manage Jenkins-¿ Manage Nodes.
   - Click on newly created slave.A window like below appears:
   - Click on launch button or copy the URL and run it in command prompt of slave machine. A jnlp agent gets downloaded. Running it following slave window appears:
   - Eve Hudson slave can be installed as "Slave as a service" by clicking on "Install as Windows Service".

Figure 3.8: Hudson Slave Launch Window

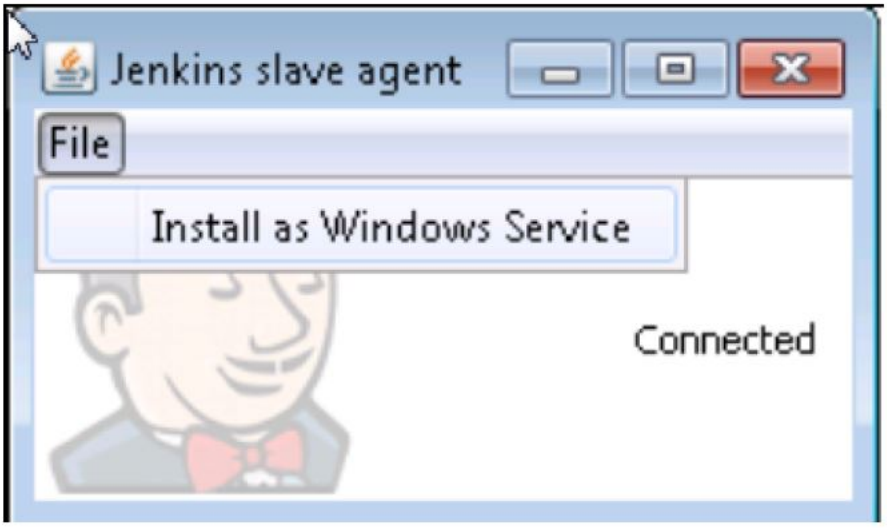

Figure 3.9: Hudson Slave

3.1.2 Configuring a new JOB on Hudson slave

A job on Hudson slave is actually a script or a snippet of code that a user need to run on slave machine to serve some purpose. To con

gure such job follow following steps:

1. Go to URL "http://localhost:(portno)"
2. Click on "New Job".A screen as shown in Figure 3.19 below appears :
3. Enter the details such as "name of the job" and click "build a free style project".Click "save". It leads us to a "Configuration page" as follows:
4. Here click on "Restrict where project is to be run". The address of slave node is to be selected from the drop down menu here.This means our build will run on the slave node selected.

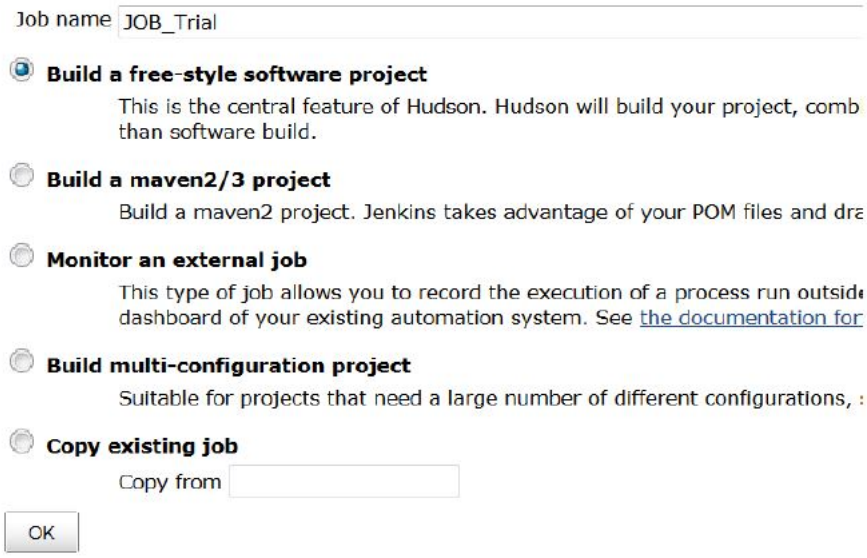

Figure 3.10: A new job on Hudson

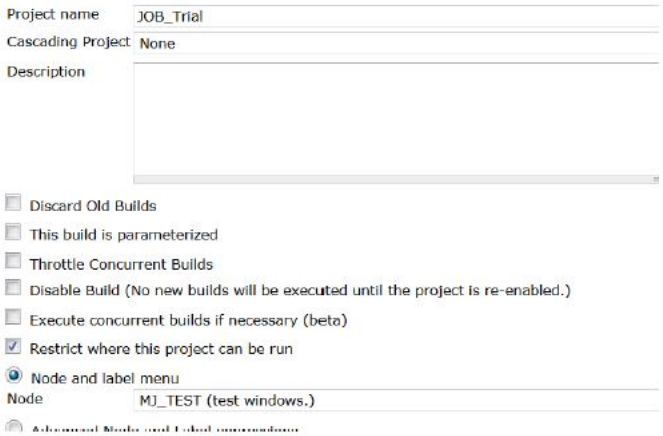

Figure 3.11: JOb Configuration Screen

5. Now one needs to specify the path of the build

   le on slave in "Execute build" option as follows:

Figure 3.12: JOb Configuration Screen

6. Finally click "Save". A job if successfully con

   gured leads to following screen:

Figure 3.13: JOb Configuration Screen

7. The "Workspace" option when expanded shows all files that are kept under "Remote FS Root" on slave machine.

8. "Recent changes" shows the summary of changes in the build since last successful build has taken place.

9. "Permanent links" are links like "Last successful build" that gives a link for last build that was executed successfully.

Chapter 4

SYSTEM OVERVIEW

4.1 Oracle Retail Predictive Application Server (RPAS)

RPAS is the foundation for Oracle Retail Predictive Solutions. The Oracle Retail Predictive Solutions gives number of products which generates forecasts and develops trading plans by analysing customer behavior. These products use predictive technology to examine historical data and predict future behaviour. There are number of applications that are part of the Oracle Retail solution footprint, such as Oracle Retail Demand Forecasting, Oracle Merchandise Financial Planning, Oracle Assortment Planning, Oracle Item Planning, Oracle Size Profile Optimization, Oracle Replenishment Optimization and Oracle Advanced Inventory Planning.

Figure 4.1: RPAS Architecture

What makes our Predictive Analytic Solutions unique is our underlying database architecture or as we call it, the Predictive Application Server.
The Oracle Retail Predictive Solutions run from a common platform called the Oracle Retail Predictive Application Server that includes features such as:

- Aggregation and spreading of data.
- Product, business and time location hierarchies
- Multidimensional databases.
- Workbooks and Worksheets that manipulates and displays data
- Wizards for creating and formatting workbooks and worksheets

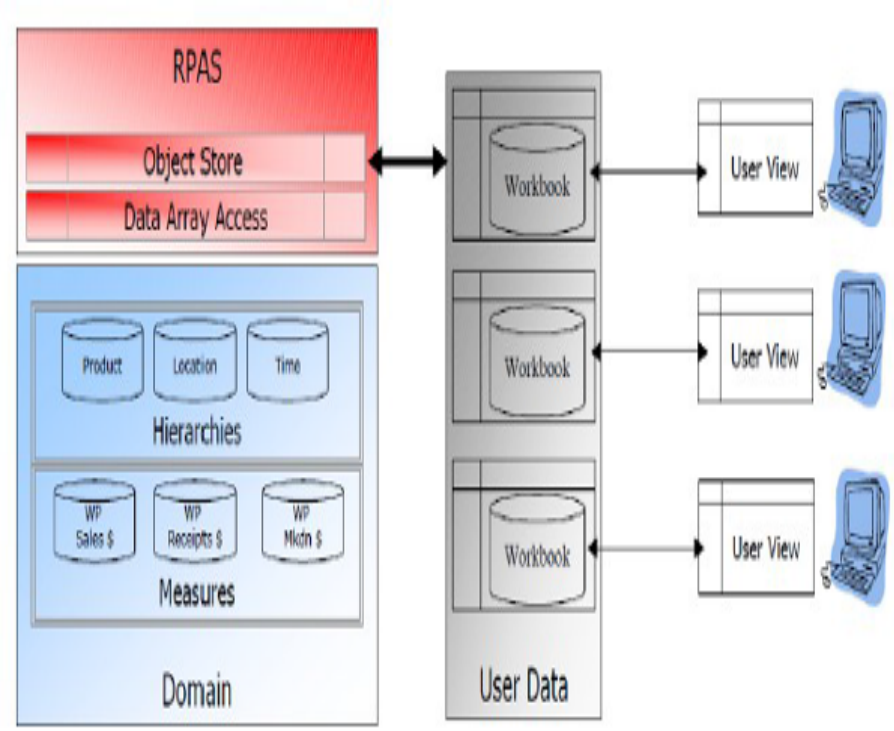

Figure 4.2: Operating Architecture

4.2 Domains

The RPAS Platform was designed using a multidimensional database instead of a relational database. A multidimensional database gives an application a fully normalized table definition (i.e. all coordinates of the cube are addressable even if no value currently exists at an address), and the ability to store a small percentage of the total rows in the table. All missing values are assumed to have a single value (NA value). Usually, the missing value is the most common value. This allows an application to get a value from any possible address in the table and only store those values that are required. RPAS refers to a fully normalized table as an array. An array is a multi-dimensional table where

the positions and values are stored together. Currently, RPAS uses a BTree structure to store key/value pairs of an array. A multidimensional database provides very high-speed access to individual cells within an array. A multidimensional structure accepts the addition of new dimensions while providing the ease of data analysis. A three dimensional multidimensional array (2x2x2) contains 8 cells rather than a two dimensional relational table that contains 32 data cells.

Logically, a domain is the storage mechanism for an RPAS application. Physically, a domain is a directory structure on a file subsystem. Domains are where the master data persists for the application server. When an RPAS client application connects to an RPAS server, a domain is specified to indicate what data database should be used.

A configuration tool allows an environment (domain) to be configured. A hierarchy is a top-to-bottom set up of parent-child relationships between elements of the same type. Hierarchies provide a means to define relationships between dimensions (aggregates, roll-ups, and alternate roll-ups) and groups belonging to the same entity (for example, Time = years, months, weeks, and days).

The following hierarchies are created automatically and cannot be deleted even from Configuration Tools:

1. Product [PROD]
2. Calendar [CLND]
3. Location [LOC]

These three hierarchies are the pillow of RPAS-based solutions and cannot be removed, but we can add more to support business process. There is no limit on number of hierarchies supported so one can add as many hierarchies as required by the business process.

Along with the Hierarchies, we can also add as many dimension as required for the given hierarchy. Dimensions are the components of hierarchy which define its structure

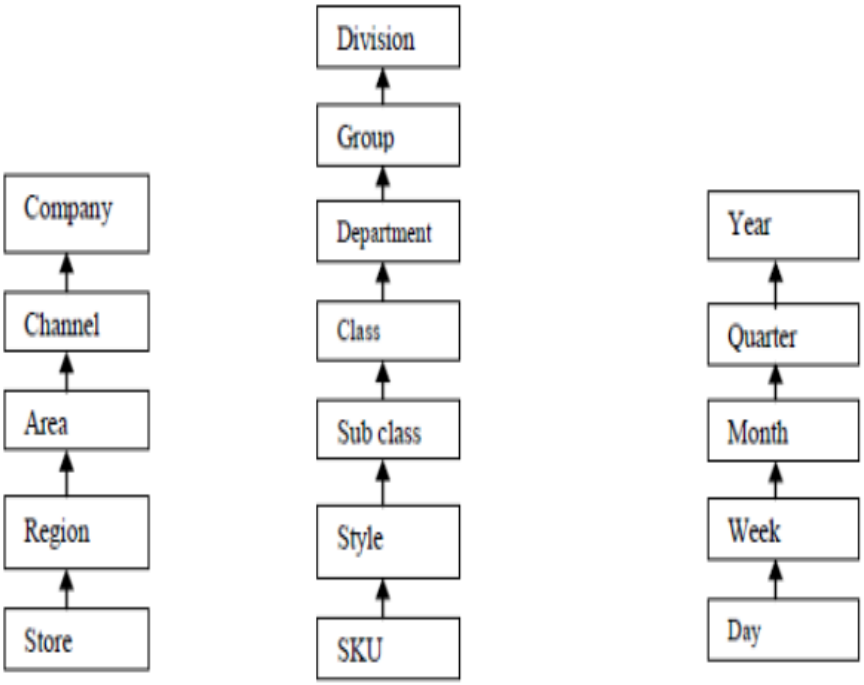

Figure 4.3: Hierarchies

as well as rollup for given hierarchy. For example, the dimensions for a calendar hierarchy can be year,month,week or day , or they can be accounting periods.

A measure is a property on which calculations (e.g., sum, count, minimum, maximum, average) can be made. A measure represents a business metric.The lowest level for any measure is its base intersection defined in configuration tool.Measure is only editable at or above its base intersection.The measure is only stored at the base intersection in domain. E.g. the number of umbrella sold in Manipal in January. Here the intersection is class-region-month.

In RPAS, An ordered collection of rule is called rule group. Rule groups can be of four different types: refresh,load, calculate and commit. Rules are expressions that describe the relationship between measures.The expression states the relation between different measures and the values are displayed on different measures based on this calculation.These rules form the basis and values are updated once the calculate is clicked.Expression are written in such a way that by doing calculation on single measure,other measure are also evaluated using some arithmetic calculation.

RPAS Configuration Tool is used for configuration of a domain.Configuration tool provides the structure of domain.The configuration Tool provides various functionalities like creating new hierarchy, deleting hierarchy, create a dimension, specify hierarchy properties, defining dimension properties, delete a dimension,define measures, editing dimension,defining measure properties, creating rule, deleting rule, creating rule group, deleting rule group, adding rules to rule group, adding an expression to a rule, deleting an expression from a rule etc.

How is a domain for a solution built? The solution for a specific business requirement is configured via the RPAS Configuration tools. This configuration is saved as xml file .This configuration defines the solution and is used to build the domain. In order to build the domain, the data needed is provided in the form of .ovr files. After the configuration is built and the files are available, a domain can be built on the UNIX server. RPAS Installer takes as input these xml file and .ovr files and builds the domain. A solution can use a subset of the dimension and hierarchies defined within the domain.

There are two main types of domain structures. These are simple and global. Simple domains are domains that stand alone by themselves. Simple domains could be used when the data size (i.e. the size of hierarchy positions and the density of the data) is small. Simple domains could also be used when the number of users for a given domain is less. A global domain environment refers to a collection of domains built with a single master domain and multiple local domains. The local domains must be partitioned on a single non-calendar dimension.

Data is stored in 2 ways in global domain :

1. Master or Global Domain
2. Across Local Domain

4.3 Client

The RPAS client is a Windows based GUI that is currently supported on Windows XP and equivalents. It is built using MFC as an infrastructure. A workbook is an easily

manipulated easily viewed multidimensional framework that perform interactive business functions in configured solution. To display data in viewable format,workbooks can contain many number of worksheets.To view and analyse business functions,workbooks and worksheets are used. RPAS uses a series of different wizards to obtain information for building workbook.Entire data of domain is divided and available to user in form of Workbook; so think about the most logical flow for the wizards. Wizard allows end user to make choices in workbook.

A user will access the domain via the RPAS Client. When in the domain, a user will access a workbook wizard specific to the workbook template they want to build. After the user makes their selections, RPAS will build the workbook. The building of a workbook includes building the structure of the workbook, loaded the current data from the domain, running the calculations for non-loaded measures and data is aggregated to meet the needs of the view in the workbook. After the workbook is built, it is available to a user to make updates and changes. A user is required to save the workbook in order for it to be used later. When saving a workbook, the changes are still isolated to the individual workspace (i.e. workbook). In order for other users to see the updates and for new workbooks to contain that updated data, the workbook must be committed back to the domain.

The Workbook Designer in RPAS Configuration Tool allows for the creation selection, and integration of the various components of a workbook template, which is a pre-designed workbook that is formatted for RPAS users to view and manipulate data. It contains worksheets,tabs,wizards, rule groups and workflow processes. The workbook designer allows you to specify which wizards will be used to build the workbooks.

4.4 Server

The RPAS server platform is currently supported on Solaris, HP-UX, AIX, and NT. Support of NT is only for demo and development purposes. The server is multiple instances of a process that contains a rule engine, a calculation engine, and RPAS Core. A Domain-Daemon process is used to start multiple RpasDBServer processes. One RpasDBServer process exists for each client application. The client application communicates with the

DomainDaemon on a predefined port ID. When a client application requests a server process to start, the DomainDaemon starts the RpasDBServer process, dynamically generates a port ID, and sends the port ID to the client and RpasDBServer. This port ID is used for the lifetime of the relationship between the client and the RpasDBServer process. Since all communication is initiated from the client application, the client can detect a missing server process if the server process terminates unexpectedly. The server process periodically checks for the existence of a client application. If the client application is missing due to an unexpected shutdown, the server process stops. If the DomainDaemon is terminated for any reason, all new client-server connections are halted, but all existing connections can continue.

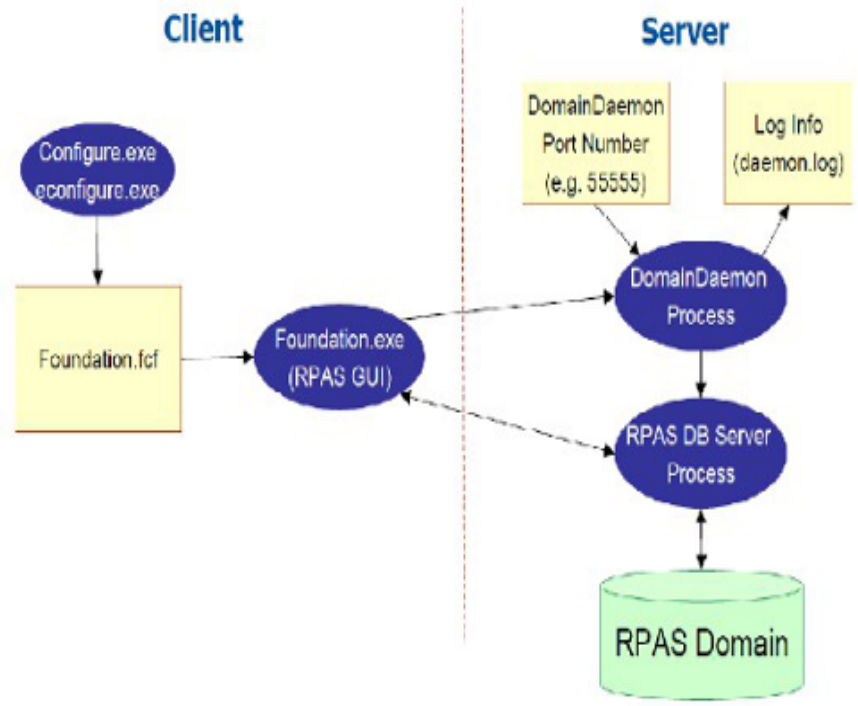

Figure 4.4: Client/Server Interaction

The Rule Engine and the Calculation Engine are key parts of the RPAS system. The RPAS Rule Engine manages the relationship between expressions, rules, and rule groups. A rule is defined from one or more expressions. If a rule has a single expression, the left hand side of the expression is always a destination and cannot be modified by the user interface. If a rule contains inverse expressions, then all but one measure within an expression can be modified and used to drive a calculation sequence. A rule group

consists of a series of related rules that defines the relationship between measures. The rule engine is able to generate a list of ordered expressions from a rule group. It also can determine when a list of expressions is constrained and no further edits are allowed to a set of measures.

The RPAS calculation engine is a very flexible and powerful engine that is built to support OLAP [OnLine Analytical Processing] type of calculations against a multidimensional model. RPAS supports two different forms of relationships between cells: Measure relationships which requires rules and expressions and Hierarchical relationships which requires aggregation and spreading.RPAS Functions provides mechanism to perform operations within an expression that are executed and controlled by the calculation engine. Most functions have only 1 output. The calculation engine executes and controls the evaluation of a function.

The RPAS core is the remaining portion of the RPAS library. The RPAS core provides support functionality for domains, workbooks, calculations, communication, and storage.

The RPAS storage layer consists of a multidimensional database, which comprises a BTree and dimensional information. The bridge library contains C++ objects that are shared between the client and the server. The RpasDbServer uses the communications library with the bridge library to implement a request broker. The client application manages communications under a database interface layer.

4.5 Domain Daemon

It enables the communication between RPAS domains and RPAS Client.Once we start the Domain Daemon,we can access the data of domain from RPAS Client.Domain Daemon is server side utility.It will wait for request from Client.After receiving the request,it will start process on specific port which allows client and server to communicate.

4.6 Client Server Communication

A DomainDaemon process is used to start RpasDBServerprocess. One RpasDBServer process exists for each client application. As in Figure 1.8, the Client application communicates with the DomainDaemon on a predefined port ID . When a Client application requests a server processes to start, the DomainDaemon starts the RpasDBServer process, dynamically generates a port ID, and sends the port ID to the Client and the RpasDB-Server. This port ID is used for the lifetime of the relationship between the client and the RpasDBServer processes. Since all communication is initiated from the client application, the Client can detect a missing server process even if the server is terminated unexpectedly.

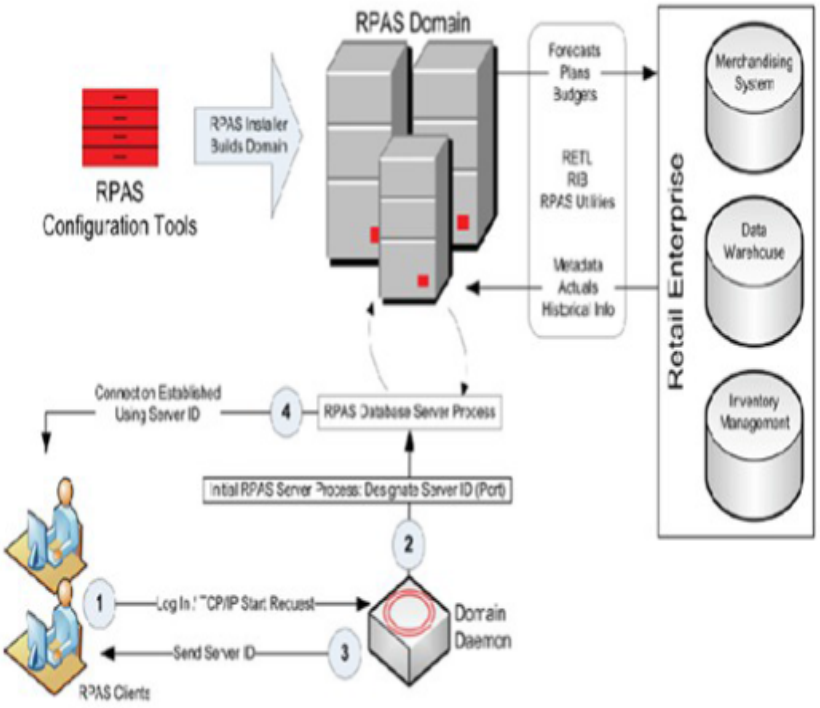

Figure 4.5: Client/Server Communication

The server process periodically checks for the existence of the Client application. If the Client application is missing due to an unexpected shutdown, the server process stops. Ideally, if the DomainDaemon is terminated for any reason, all new Client Server connections are halted, but all existing connections can continue. In practice, the existing connection also terminates if due to some reason, the DomainDaemon terminates. The DomainDaemon must be started and all the connections must be established again to

regain the connection.

4.7 RPAS Architecture

The Oracle Retail Predictive Solutions are products tha generates forecasts and develops trading plans to analyze customer behavior. These products use predictive technology to examine historical data and predict future behavior. The underlying Database Architecture makes the Predictive Analytic Solutions unique.

The Oracle Retail Predictive Solutions run from a common platform called the Oracle Retail Predictive Application Server that includes features such as:

- Aggregation and spreading of data
- Time,Product and business location hierarchies.
- Multidimensional databases
- Workbooks and worksheets for displaying and manipulating forecast data
- Wizards for formatting workbooks and worksheets
- Quick menus,Menus and toolbars for working with forecast and sales data
- Exception Management and user friendly alerts.

The architectural design gives the overall view of the RPAS server. As shown in Figure 3.5, each user may create a set of workbooks and store it locally in the system. The workbook gets the data from the RPAS server. The data is stored in domain. The domain consists of measures and hierarchies. The workbooks use the data from the data arrays.

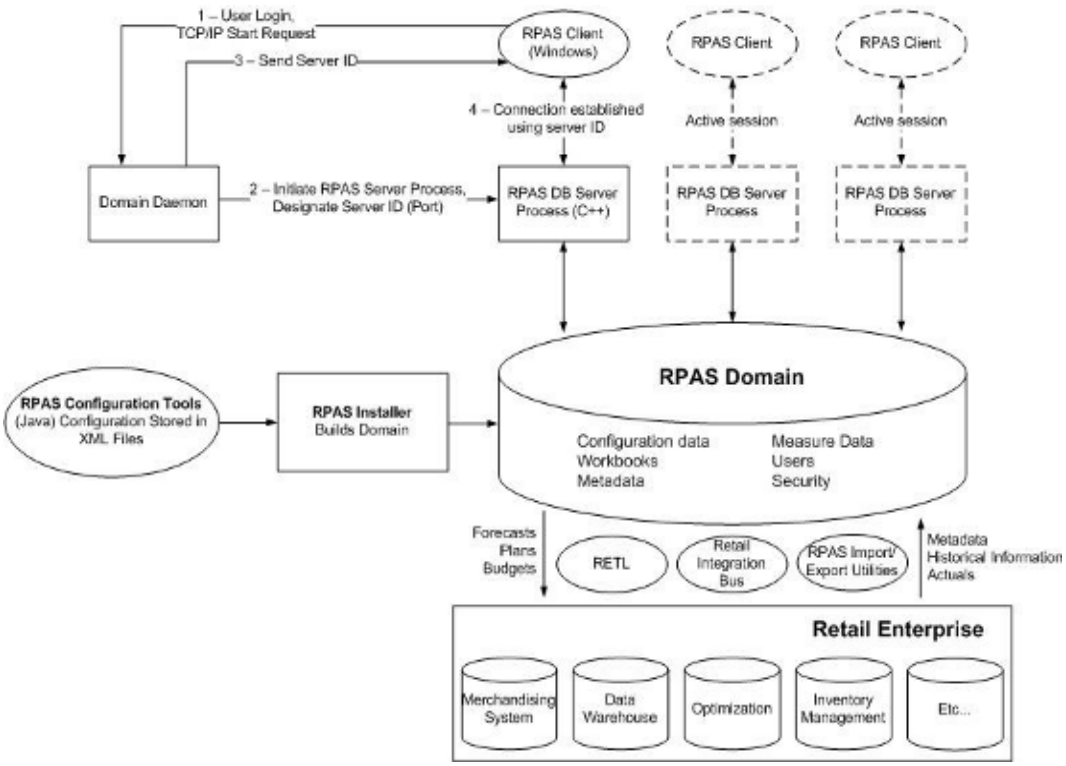

Figure 4.6: RPAS Architecture

Chapter 5

TEST AUTOMATION

Automation can be performed with the help of any Automation Tool.Task of these tools is to take predefined steps and returns the result as a success or failure.Test script should able to determine the success or failure based on the steps given as an input.

These automated test scripts once created,needs to be run or playback as an when required which reduces lot of manual efforts and hence decrease cost

Following are the benefits of Automation :

- Increased Test Speed, Test Efficiency and Software Quality
- Decreased Costs and shorter Time to Market
- Increased Confidence
- Greater Test Coverage

The project aims at learning and understanding the functionalities of the product and automate the test cases to increase testing efficiency and save testing time. The automated testcases must handle various scenarios so that the test cases can be executed with little or no manual assistance and hence save testing effort. Using Oracle's internal automation tool, the testcases in Advanced inventory planning are automated. Firstly, the functionality of the product and the work flow with in the product must be analysed and understood . Then identify the steps that can be automated and identify the reusable units . Reusable automation scripts are written(keywords) and using the keywords, the test cases are automated.

5.1 Automation Testing

A pre-defined sequence of steps to write and to execute testcases using and automation tool is called automation testing.As an when require, we need to run the testcases which requires no manual intervention.

Each functionality has different number of testcases to cover the different scenarios,all that testcases are grouped together to form a TestSuite

5.2 Benefits :

- Repeatability
- Greater consistency
- Reduction of repetitive work
- Ease of access of information about tests or testing

5.3 Why we need Automated Testing?

Each and Every software product undergoes testing.Every company has the team to perform Manual Testing.Everyone tries to catch the defect before it goes in the hand of customer.Once the software goes in to the hands of customer,some or the other defects are reappears which needs to be fixed and retested

Hence,Automation Testing helps us to retest the bug easily and efficiency.

In Manual Testing,some one has to sit in front of desktop and analyze various scenarios,validating whether the given product should able to produce the expected results in all the scenarios.This tedious task should be performed on each and every release and for every patch release which leads to rework.Creating Automated Test Script make the task easier and efficient.Every time the new version come we need to run the scripts which saves lot of time

5.4 Importance of an Automation Testing

5.4.1 Automated Software Testing Saves Money and Time

Software tests have to be repeated often during development cycles to ensure quality. Every time source code is modified software tests should be repeated. For each release of the software it may be tested on all supported operating systems and hardware configurations. Manually repeating these tests is costly and time consuming. Once created, automated tests can be run over and over again at no additional cost and they are much faster than manual tests. Automated software testing can reduce the time to run repetitive tests from days to hours. A time savings that translates directly into cost savings.

5.4.2 Testing Improves Accuracy

Even the most conscientious tester will make mistakes during monotonous manual testing. Automated tests perform the same steps precisely every time they are executed and never forget to record detailed results.

5.4.3 Increase Test Coverage

With the help of Automated Testcases ,many complex and long scripts can be run easily which increases the coverage and thereby save time and manual effort.Many long and complex scripts which is impossible and time consuming when performed manually, can be run quickly.Manually validating same scripts on different configuration is a tedious task which can be accomplished through automated test cases.

5.4.4 Automation Does What Manual Testing Cannot

Automated Test cases can create a thousand of virtual users and makes the task of controlled web application test to perform in no time.Manually creating the same scenario and test with thousands of users is not possible so automated test cases helps in such scenario.

5.4.5 Automated QA Testing Helps Developers and Testers

Automated testcases once created can be used by tester as well as developer.Before giving the product to QA team,developers should make sure that any small changes should not break any of the remaining functionality.so automated scripts once created can be run

at the developer's end and notifies developer if failure occurs which saves lot of time and resources

5.4.6 Automated Software Testing Tool

Different types of Software Testing tool like Oracle Synergy,Open Script,Test Complete,Selenium etc are used by Software Testing Department to automate the test cases and increase efficiency

Chapter 6

SYNERGY AUTOMATION TOOL

Synergy is the software tool which is used to automate the Test-scripts. Synergy is developed using the Languages like JAVA, JAVASCRIPT and PYTHON.

Synergy provides UNIFIED AUTOMATION PLATFORM that provides :

- Easy and fast way of adding new automation plug-ins for new technologies
- Common test execution and recording architecture across all automation plug-ins
- Common automation integrated development environment (IDE) across all automation plug-ins
- HPQC integration for metrics reporting

6.1 Synergy Test Suites

A Test Suite is a collection of tests that validate whether a software program exhibit a specific behaviour and any supporting constructs/information neccessary for the execution of contained tests

Synergy Test Suite contain a following component :

- Test Cases
- Folders
- User Keywords
- Resources Links

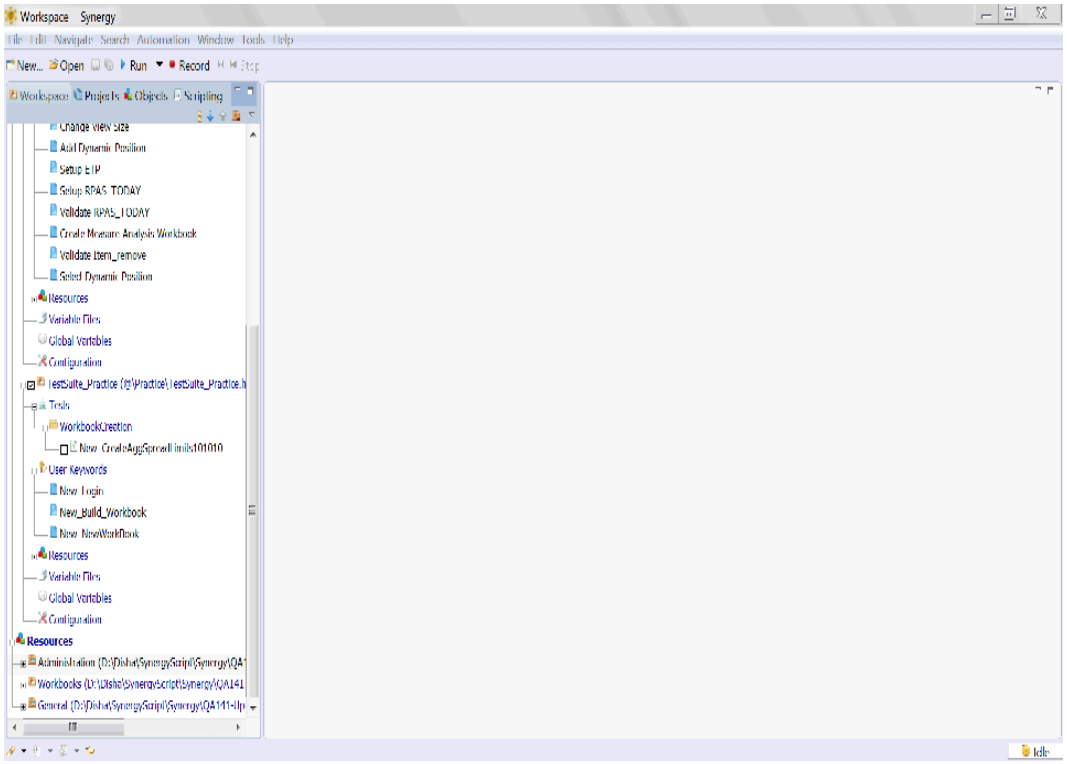

Figure 6.1: Synergy Workspace

- Global Variables
- Configuration Settings

6.1.1 Test Suite Creation

A test suite is created by selecting the Automation -¿ Test Suite wizard accessible from the New button in the Synergy main toolbar or under File -¿ New.
Once you have selected the new Test Suite wizard, you will be prompted for a test suite name and location.

6.1.2 Test Suite Configuration

Each test suite has configuration information associated with it. The configuration information may be accessed by double clicking on the Configuration of your test suite in the Workspace tab.

Under the Execution Settings section, you can configure test suite setup and teardown keywords.

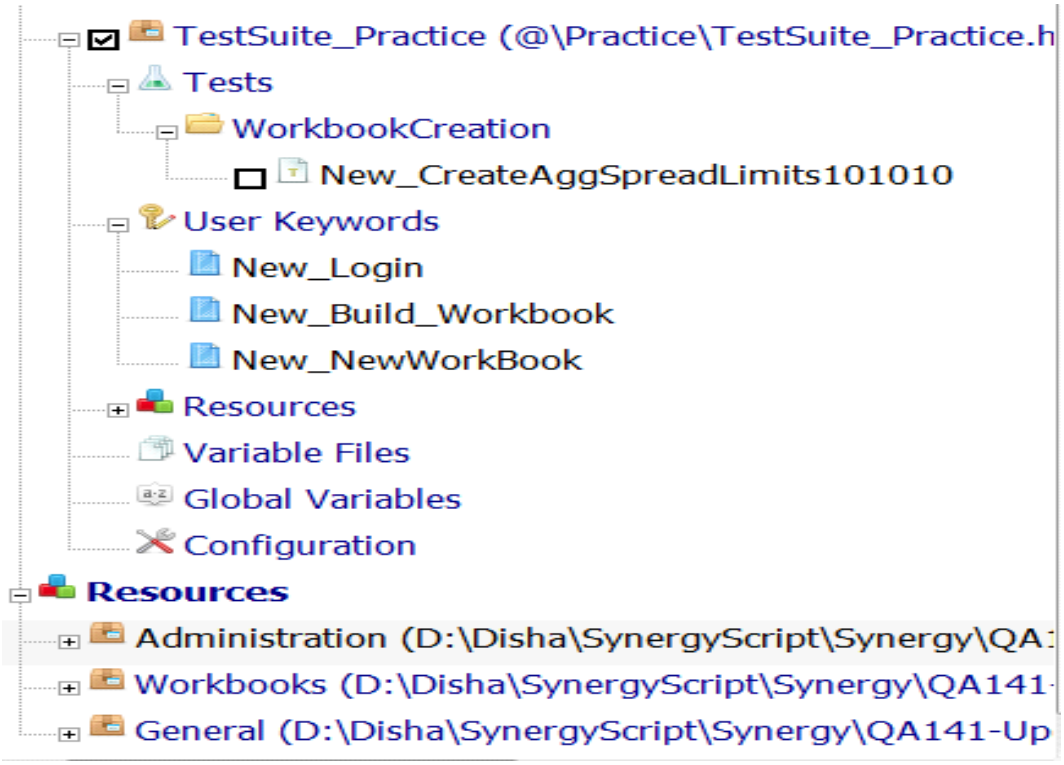

Figure 6.2: Synergy Components

6.1.3 Suite Setup Keyword

A test suite setup keyword can be used to perform any operations that may be needed before the execution of tests such as environment setup, database connection setup, data aggregation, etc.

A test suite setup keyword runs only once when the test suite is executed, before any tests contained in the test suite have run.

If a test suite has a suite setup, the setup is executed before any of the test cases.

If the suite setup passes, test execution continues as normal.

If the suite setup fails, none of the test cases are executed.

Suite setups are often used for setting up the test environment.

Since tests are not run if the suite setup fails, suite setups that verify that the environment is in a sate in which the tests can be executed are extremely useful.

6.1.4 TestSuite Teardown

A test suite teardown keyword can be used to perform any cleanup or error recovery operations that may be needed after the execution of tests has completed.
A test suite teardown keyword runs only once when the test suite has finished executing regardless of whether the tests contained in the test suite have run successfully.

If the testsuite includes Suite Teardown at the end,whatever steps are mentioned in Teardown are executed at the end of the Testsuite regardless of the success or failure of the testcases.

If the suite teardown fails, all tests in the suite are marked failed afterwards.
Suite teardowns are commonly used for cleaning up the test environment after execution.
To ensure all teardown tasks are performed, all the keywords used in the teardown are executed even if some of them fail

6.2 Automation Library

Keywords provided by Synergy are organized into automation libraries.
Automation libraries contain those lowest-level keywords, often called library keywords, which actually interact with the AUT.
All test cases always use keywords from some library, often through higher-level user keywords.
Before any keyword provided by Synergy can be used in your test cases, you must explicitly import the library that provides that keyword into your test suite configuration.

Automation libraries can be imported by clicking on the Import library... button which will then present you with an import wizard associated with that library.
For most libraries, you can simply click Finish in the wizard in order to import the library.
For some libraries, however, you may have to provide additional parameters (such as database connection information, for example) on various import wizard screens.

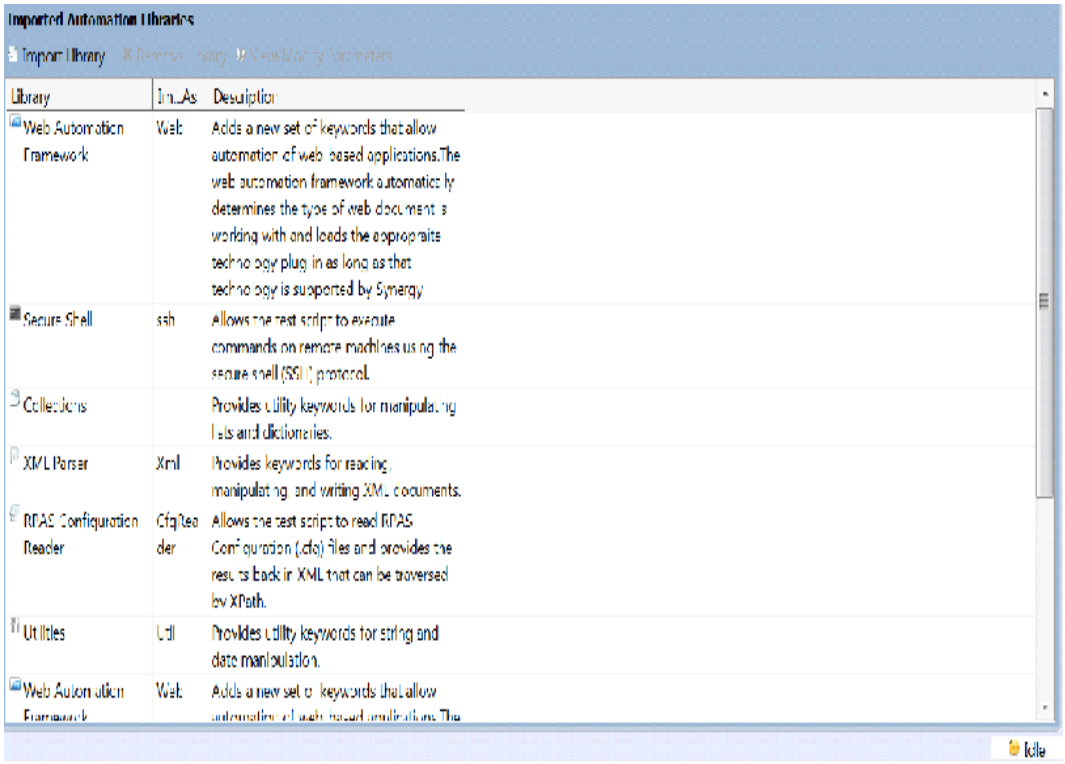

Figure 6.3: Synergy Automation Library

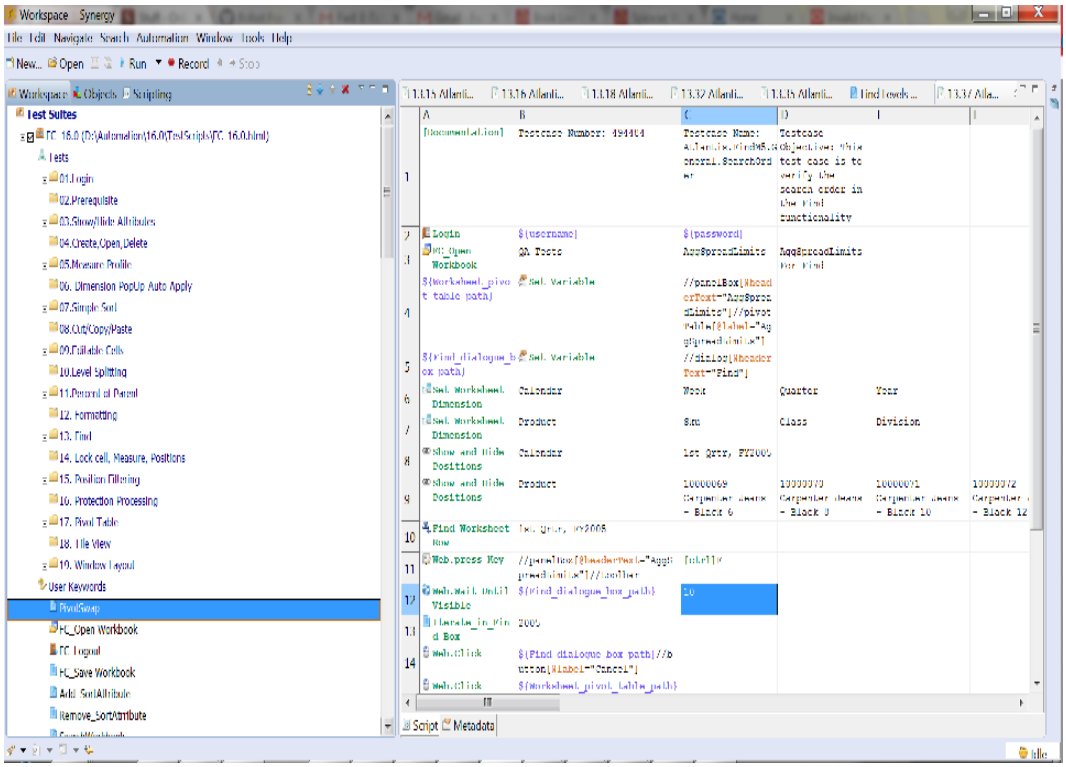

Figure 6.4: Functionalities of FC automated

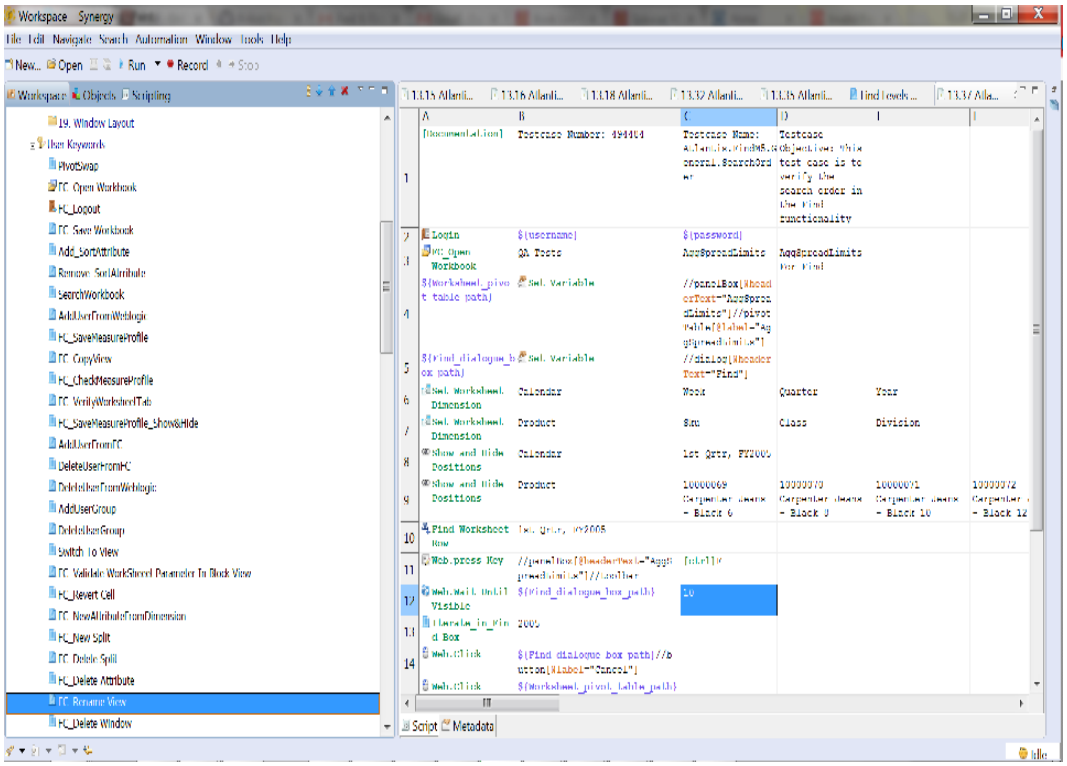

Figure 6.5: Synergy Keywords

Chapter 7

Software Used

7.1 Tortoise SVN

To run a build or to modify some code, we need to download it from a central repository or a subversion repository. This process is called "Checkout".
Basically a "check-out" is downloading a latest version of the required script from central database.
Similarly, "check-in" is submitting your code to central repository after modifications. A software used for this purpose is "Tortoise SVN".

7.1.1 Tortoise SVN Installation

Pre-requisites

Tortoise SVN runs on Windows Vista or higher and is available in both 32-bit and 64-bit operating system.

Installation

Tortoise SVN comes with an simple installer. Double click on the install.exe and follow the instructions.Restart your machine after installation.

7.1.2 Tortoise SVN usage

- Once system is rebooted,right click on desktop and select "SVN checkout".A window like following appears:
- Mention URL from where code needs to be downloaded in "URL Repository".
- Mention local path to save the file in "Checkout Directory".

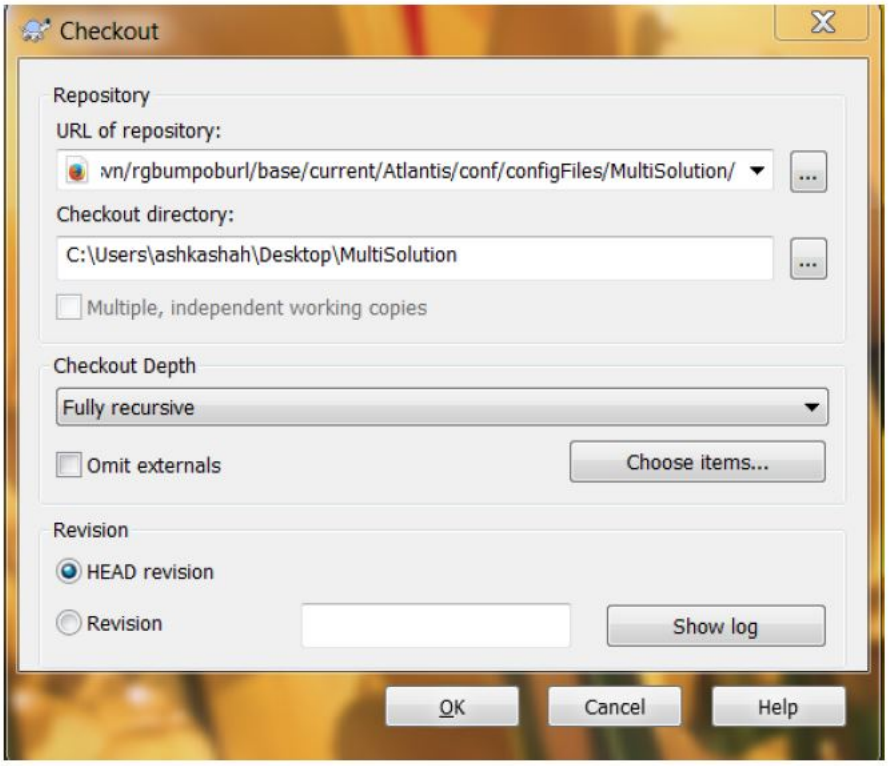

Figure 7.1: SVN Checkout

- If one needs to checkout folders and sub-folders then click "Recursive depth".

Chapter 8

Conclusion

Software testing is the process of evaluation a software item to detect differences between given input and expected output. Testing assesses the quality of the product.Manual Testing requires some one to sit and do the repetitive task of installing Weblogic and hooking Fusion Client and domain into it. Automated software testing is the best way to increase the effectiveness, efficiency and coverage of your software testing.The process of installing Weblogic Server 12C, creating a weblogic domain on it, installing a Fusion Client on the weblogic domain and hooking multiple product domains on it in 17 different languages and running all the Regression Test on it has been completely automated and used by the Internal Team of Oracle.

Oracle Internal Automation Tool (Synergy) is used to automate testcases in RPAS apps. It provides excellent reporting and provide good debugging options that allow the user to pinpoint and fix failures quickly. By automating the testcases, they can be executed repeatedly with little or no manual effort. This saves time and increase quality.

Bibliography

[1] Shotts, William E., The Linux Command Line: A complete Introduction, No Starch Press USA, first edition, ISBN-10: 1593273894

[2] Stevens, Richard w., Advanced Programming in the UNIX Environment, Addison-Wesley USA, second edition, ISBN-10: 0321525949

[3] Oracle Retail, http://www.oracle.com/us/industries/retail/solutions/index.html

[4] Frank Munz, Oracle WebLogic Server 12c: Distinctive Recipes: Architecture, Development and Administration, Munz More, 2013 Hill Osborne, 2013