BTECH 451

Orion Health – Product Quality and Reliability

Priyal Bangia
Pban49

Contents

ABSTRACT

It is critical to pay attention to the complaints which companies receive and utilize information to discover product performance within the market in order ensure the quality of the product is maintained by management. The aim of this project is to improve product quality and reliability through a Business Intelligence approach. This report discusses the issue of reports being manually created as well as data being stored in an unreadable, unstructured form within Orion Health who provide products to many clinics and hospitals locally and globally. In order to have real time reports, the data has to be retrieved manually on a regular basis. The types of different reports created and maintained is limited due to the manual process. This consumes a lot of time and energy, it gets difficult for Complaint Administrators and Data Analytics to carry out the process to provide meaningful data to management in a fast paced environment. Many more companies follow a similar manual procedure who are also not using the best approaches for data retrieval and reporting.

A proposed solution to this is to automate data retrieval using an Extract, Transform, Load process as well as providing some useful and interactive automated reports which reduces the effort that goes into reporting from the current process. This helps uncover historical trends about products and therefore the quality of products can be focused on. Different tools for storing data, carrying out the ETL process and reporting are analysed and the best suited is used for implementation.

INTRODUCTION

This project is entirely based on a Business Intelligence approach to improve product quality and reliability. BI will aid in transforming raw data into meaningful and useful information to make it easier for analyse in order to create better decision making on a product basis. BI will be able to handle enormous amounts of unstructured data to help identity, develop and create new strategic opportunities and provides historical and current business operation views for tracking. This project is aimed at providing Orion Health with an improved data retrieval and reporting service using a suitable software. The project focuses on transforming the data into a structured form where data is easy to interpret and querying (data retrieval) is quicker.

People comprehend data better through visualizations as compared to just reading numbers in rows and columns. So by visualizing data, you are able to more effectively ask and answer important questions such as “Which versions realised of a product have which issues”

“What is driving growth” and “Which areas are getting the most complaints therefore what part of the implementation are we lacking in and need to focus on”. With the ability to quickly answer questions, your data becomes a competitive advantage instead of an underutilized asset [5]. Data is no longer just a bunch of rows and columns. It's actually something more meaningful *which can* help quickly discover patterns in your data that unveil trends and reveal unexpected insights in order to have the focus on correcting the right area as well as prioritising how much work and effort needs to be put into certain products or issues with the product. For this Orion to get a clear broad and in depth view of the product performance, it is essential to retrieve requested reports as fast as possible.

The goal is to automate data retrieval and reporting by performing an ETL process using SSIS to extract data from the database and transform it in a way where it is much easier to analyse the structure itself. Reports will be generated and the users will be given flexibility to select what they want a report of rather than creating new reports for similar things. To make it easier to get a broad view of the operations an interactive Dashboard is considered.

THE COMPANY

This begin when founder Ian McCrae and CEO of Orion Health left his role at Ernst & Young as a Business Consultant in 1992. He wanted a job which was more interesting, challenging and would be able to make a difference in the world. Internet was not used in those days. Together with people who had a similar vision, formed a consulting firm where they obtained various IT projects related to Auckland District Health Board who required computerization and to connect all hospitals. Using their ingenuity they places microwave dishes on the roof of the hospitals to facilitate data sharing. The outcome came to be a level of inter-connectedness which had not been carried out before. The benefits and savings made by interacting was huge. They started focusing on innovation where the team focus on coming up with creative ideas and the success of projects opened up pathways for many other projects locally and globally in countries such as Australia, Spain, France, USA and many more. Orion Health offices are located in 23 countries with over 1,100 employees, the company grew at a speed of approximately 30% per year and is evolving more and more in the future.

Orion Health is now a leading global provider of HIE and Healthcare integration solutions. Products and implementations which were built by this company are used by 30+ countries.

Orion health implements and provides various facilities to hospitals such as collaborative care to securely and easily exchange clinical information with other organizations, electronic health record, and hospital information system and efficiency tools. Also another product called Orion health mobile for iPhone or apple devices provides complete flexibility for healthcare professionals. All these products give an outcome of increased quality, efficiency, accuracy, collaboration of data and saved costs/risks as well as improved patient safety and experience.

Nothing is perfect therefore because so many countries use products implemented by Orion Health, they are expected to receive a large number of complaints or issues clients are facing with the products. I now help look into using the information to keep up with the competitive market. Using BI it is possible to gain insights to understand and analyse the product performance and opportunities at a deeper level leading to scalability and flexibility to grow. This is only done if Orion Health data is utilized in a way to monitor the performances of their products and also create new innovative versions based on previous information as well as have the ability to make better strategic decisions for the better. However, currently a manual tradition approach is being used which is restricted and will not provide opportunities to grow in the long term.

THE PROBLEM

Currently, the major issue with the process is that the reports are being manually generated which takes a lot of time, energy and is very tedious. The cause of this being the unstructured and unreadable data stored in the support system. When new complaints arrive, the reports are not automated accordingly which means they are not in real time. The Complaints Administrator presents these reports to management on a weekly-monthly basis but before that they have to update the reports. The process of having up to date reports depends heavily on retrieving the updated data from the support system again. The reporting process here is not a simple task as it would normally be, it applies numerous steps to achieve. This continuous manual reporting process becomes difficult for the Complaint Administrators and Data Analytics to carry out the process to provide meaningful data to management.

To update a previously created report, a search feature is used to select the data required from the support system then it is required to be transformed and exported into excel. This consumes time because this has to be done for every separate report. This process also has to be carried out on detailed reports on request. When required data is exported into excel and if you require other, all that data must be queried and transformed again because you do not have the flexibility once the data is already exported into excel. The number of reports created and maintained is limited due to the manual retrieval of data for every report created hence, not being able to monitor product performance to enhance product quality and reliability.

The current process is explained where it is easier to discover that this is a lengthy process which uses an out dated approach having huge disadvantages.

Current Process

The reporting process begins with issues coming in on a support application which Orion Health uses (support tracker) which have to undergo an investigation. Before an investigation a Regional analysts will complete a Pre-investigation questionnaire for each of the issues that has come in by reading through what the customer has written. The result of this questionnaire determines if this may be considered as an alleged problem is legitimate or not. An investigation is carried out then the regional analysts completes a post investigation questionnaire based on the investigation that was executed for the issue and its fix. The result of this questionnaire determines if the issue is legitimate or not (for example – a bug) which is also represented by a 1 or 0 as you can see in figure 1.

Documentid	Eventid	Type	Questio...	Questio...	Questio...	Questio...	Questio...	Questio...	Result	CompletedDa
43f6f8b4-5819-48ff-a107-06f22a5ac6b8	40466	PRE	0	0	0	NULL	NULL	NULL	0	2013-12-19
4bd2a30c-c9ab-4056-b331-576ebdae6a...	38394	POST	1	1	0	0	0	0	1	2014-08-01
5	41784	PRE	0	0	0	0	NULL	NULL	0	2014-05-15
542ed489-059c-49b5-ad0a-019fa2bc6324	38497	PRE	0	1	0	0	NULL	NULL	1	2014-08-01
57773ecd-bb6e-4dac-b397-31258aeded...	38340	PRE	0	1	0	0	NULL	NULL	1	2014-08-01
6	41784	POST	1	1	1	1	1	1	1	2014-06-15
7	41777	PRE	0	0	0	0	NULL	NULL	0	2014-07-04
72ee5350-1d75-443d-92b7-cdb51ed37...	38406	PRE	0	0	0	0	NULL	NULL	0	2014-08-01
8	41777	POST	1	1	0	0	0	0	0	2014-07-04
8db402e3-3fd6-4ede-904a-eae2b76cdc9f	39254	PRE	0	1	0	0	NULL	NULL	1	2014-04-04
9	41792	PRE	0	1	0	0	NULL	NULL	1	2014-05-01
9752aeb8-15f0-4235-b248-a95288ac65...	40567	PRE	0	0	0	0	NULL	NULL	0	2014-08-04
98fd8777-c7b4-48d0-a02f-1662d08faad2	40575	PRE	0	0	0	0	NULL	NULL	0	2014-08-04
9bc3477e-9753-4b37-943f-e82fd3ebe9df	40561	PRE	0	0	0	0	NULL	NULL	0	2014-09-02
a0c7d138-5eaf-4004-91d1-6497c6f60826	40575	POST	1	1	0	0	0	0	1	2014-08-04
a3fd801e-9fd8-4b2a-bd51-32b09a850298	40545	POST	1	1	0	0	0	0	1	2014-08-04
a4cce4bd-9a52-4dd7-9ca5-f37c04ff067e	39785	PRE	0	0	0	0	NULL	NULL	0	2014-08-15
bd2ae7f7-dc32-4657-9b2e-92c3d7809789	40542	POST	1	1	0	0	0	0	1	2014-08-04
c7c1b2f8-2443-4b84-80f6-ae49eb53769a	39785	POST	1	1	1	1	1	1	1	2014-08-15
d845970d-ff27-4310-96bb-35790a47e8c1	40598	POST	1	1	0	0	0	0	1	2014-08-04
e30751fc-c7ab-4401-8f3c-e853a6b05213	39251	PRE	0	1	0	0	NULL	NULL	1	2014-04-04
e5356d28-7c4d-4b75-80db-e21a612ee...	40562	PRE	0	0	0	0	NULL	NULL	0	2014-08-04
f3df1e43-4ea8-4d6d-b2fb-c63709d2fd90	40542	PRE	0	0	0	0	NULL	NULL	0	2014-08-04

Figure 1

A pre and post entry is made separately for each eventid(Complaint issued). Documentid (for the questionnaires) is stated to provide a unique primary key with the eventid. Therefore there are two rows for every eventid representing pre and post results.

Interpreting this data is not efficient. For example to find out if a the complaint with the eventid is actually considered as a complaint or not the Analyst has to find the eventid then look at the result for the pre investigation(0), then the same for the post investigation(1) then determine from these two values. If either of the results is 1, the ticket issued is considered as a legitimate complaint however if Pre is 0 and Post result is NULL, this cannot be determined because either one has to be 1 but the post questionnaire has not been carried out so it is incomplete. The use of this data can be structured in a better form where interpretation can be made easier where the analyst will not have to search over hundred rows of data. Data is also redundant as there are two row for every eventid.

After a ticket is determined as a complaint it is read through by the Global Consultant for that product/application. They will 'code' this complaint down to three levels:

- Problem Code, i.e. Environment, configuration, design etc.
- Problem Area, specific area of the application
- Problem Detail, the specific detail within that area

This enables any analysis to be done over either a broad spectrum or highly detailed. These actions are essential for one to do as in today's world, technology is not capable to do so but the following process is where the issue occurs.

Once these processes are done, the Quality Consultant and Complaints Administrator uses the support application to query the expected output, the data then needs to be manually filtered and manipulated then exported into an csv file then into an excel file where the consultant has to manually create graphs, tables and analytics using excel or tableau. The graphs are then pasted onto a word document which is taken to meetings or passed onto department managers.

As the data is updated on the support system, the whole process needs to be carried out again therefore the query needs to be written out again which then is manipulated/transformed and exported into a csv file then excel or where the graphs are updated or tableau where the new versions of the graphs are created again. The Quality Consultant and Complaints Administrator carries out this cycle continuously and also when the management requires up-to-date information at the point of time or even whether is it yearly, monthly or weekly.

Orion health is a fast paced work environment so this manual process is not the best approach. Time is a huge concern for large businesses. Tasks should be done as quickly as possible in order to carry on with other tasks, this becomes a concern when some tasks are to be done sequential and dependent on the other for example, the manager needs to analyse a report for a product then decides to request for a more detailed report related to the product before making any strategic or even operational decisions. The problem related to reporting using the current system discussed in more detail in the implementation section of this report so it is easier to understand the difference.

Related Work

Accessing, organizing, and using data to support research and quality improvement projects are evolving challenges for hospital systems. As is common in health care, a system-wide relational database linking laboratory, pharmacy, and administrative data did not exist. Because the vendor's did not provide an easily accessed clinical data warehouse for research and quality improvement purposes, the authors developed their own.

Existing data stored in a hospital transactional server have enormous potential to improve the health care quality, innovation and performance measurements. The authors report the development of a clinical data warehouse that they created by importing data from the information systems of various hospitals. One of their key goals is the development of a hospital information system that could recognize and report trends in antimicrobial use and resistance.

An essential infrastructure requirement was an information management system designed to detect, track, and report the occurrence of antimicrobial-resistant organisms and to quantify antimicrobial use [10]. The data warehouse developed had the ability to store data from imported electronic data from existing hospital information systems and also data collected from non-electronic sources (e.g. patient medical records using optical character recognition). Data was imported from 13 different computing such as laboratory, medical records, and emergency department. To allow for communication they had developed a local area network and an intranet Web site that linked investigators and information. Orion health is already on the same LAN therefore communication will be easier.

The selected data elements are stored in Microsoft SQL tables which allows for automated data retrieval. The Data Transformation Services function of the SQL server enabled open database connectivity linkages to different database platforms (e.g., application databases from Pharmacy and Laboratory) [10]. This automated data extraction from each primary server to the newly developed data warehouse. To minimize any potential impact on the primary transactional server, the extraction step was scheduled [10]. Two of the biggest challenges in the planning process were accommodating the security and confidentiality.

Simple queries were was performed readily which provided them with descriptive statistics but analysing and displaying patient level data in a complex arrangement and applying business intelligence models or of variables required more advanced programming. The authors had considered to purchase an Application Service Provider, an application used for specific applications however none met their requirements. Business specifications outlining the requirements was made and solicited bids from four consulting firms[10] but they chose to develop internal resources by employing a statistician and recruiting a master's-level graduate student [10]. The system was then used for complex monitoring to measure

antimicrobial use, to detect hospital-acquired bloodstream infections, to measure the cost of infections, and to detect antimicrobial prescribing errors [10].

Another paper discusses the term BI which encompasses software for extraction, transformation and loading [11], data warehousing, database query and reporting. Managing an organisation requires access to information in order to monitor activities and assess performance. There is an increasing competitive global market which is rapidly changing and rapid shortening product life cycles therefore identifying what information is contained within the organisation is essential and can be a challenge as the information systems collect large amounts of data. Business enterprises are necessary to analyse accurate and timely data about financial operations, customers and products using familiar business terms to raise opportunities by gaining analytical insight into business problems [12]. Business are seeing the advantage and now building BI systems that supports business analysis and decision making which lead to a more clear understanding of their operations and compete in the marketplace. Thus, BI is advantageous as it turns operational information into a business asset that leads strategic decisions and improves performance for the company and their clients [12].

The first phase of the implementation was the development of prototypes after an extensive system analysis. They have focused their second phase of the implementation on the actual deployment of BI with legacy system. The project intends to create a central source of information that delivers strategic business knowledge worldwide in a consistent, timely manner by the creation of a comprehensive data warehouse focusing on order processing, inventory analysis, purchasing, sales and service [11].

Research states that the project begins with the analysis of Information requirements and evaluation of key performance indicators that define the overall business drivers which leads to examining supporting business processes to determine the architecture and develop a data model accordingly. To meet the specific requirements, BI system is implemented using the following software:

- RODIN – Highly advanced data warehouse management system which supports features such as extract, transform and load data into the data warehouse.
- Data Tracker – supports subject-area data marts with dynamic query and drill down analysis capabilities through ad-hoc.
- Crystal reports – Provides detailed analysis and reporting capabilities from the data warehouse when further investigation is required.

This BI implementation provided the opportunity to keep staff aware of the latest developments, to alert staff, to allow staff to respond to the alert. The information is available instantaneously. The authors use RODIN whereas I use SSIS for the ETL process in Business Intelligence Development Studio due to the ease of use and familiarity compared to RODIN which may be too advanced for a normal person to operate and maintain. Crystal Reports is a Business Intelligence application marketed to small businesses by SAP SE which is used to create reports from various data sources such as Oracle databases, Microsoft SQL Server databases, Business Objects and more. Here, the main ability and feature is to be able to integrate data from a range of sources however I decided to use SSRS for simplicity

as there is only one data source to be dealt with therefore this ability is not necessary and is not important although this may be beneficial if the business uses many different data sources to store their data.

Another paper focuses on two key points which is firstly valid, integrated, in time data and the other is the means which will transform the data into decision information. In support of the two key points the authors considered using a BI Technical framework which consists of operational applications Tier, Data warehouse Tire, Corporate performance and Enterprise BI Suites Tier. The data Warehouse includes more than just a Data Warehouse such as Data Mining, OLAP (Online Analysis Processing)[13]. Data Warehouse is important in my case as well as it is a large repository of historical data pertaining to an organisation. The complexity of queries required in order to support the OLAP application makes it difficult to implement using standard relational database technology therefore I decided not to continue on with this approach for this reason and also because they use Decision Tree Induction, K-Means, Genetic Algorithms for data mining strategies which gets complicated but can however be considered for future use once the Complaints Administrator is use to SSIS and SSRS which is user friendly and easy to work with.

By providing a business with a wider view of their own information allows them to understand markets, customers and the data that shapes the business. This makes the organisation more competitive and agile by empowering all employees to make faster and better decisions. To support this many Enterprise solutions and platforms have been developed including IBM DB2, MSMinging, Microsoft SQL Server, Oracle, Business Objects and many more. Oracle 10g, Oracle Express, Oracle Darwin Data Mining are all included in the Oracle BI enterprise solution which are used for 3 BI technologies, data warehousing, OLAP and Data mining[13]. MSMiner developed by the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology [13] which is a data mining platform for decision support. This platform consists of six modules: OLAP, ETL, Data mining, Metadata server, data visualization, some which are useful in my case as well but I have chosen to go with Microsoft SQL Server because the current version and upcoming versions are expected heavily for its BI capabilities [13] and reaches significant enhancements in OLAP and data integration.

Moving on with the architecture, the article “Method and architecture for automated optimization of ETL throughput in data warehousing applications” focuses on architectures to automatically optimize and throughput of the data extraction/transforming (ETL) process [12]. This architecture has two aspects, the component aspect which indicated that all the transformations used are built upon other transformation components and a pipeline aspect. The pipeline is used in order to provide users the building blocks to effectively construct pertinent and functionally sophisticated transformations [12]. The way in which a pipeline function is that each transformation pushes data into another transformation component or it either pulls data from another transformation component therefore preforms push/pull operations.

Figure 2 is one example of independent pipelines of transformation components for parallel executions. Also shows the arrangement of transformation components that the users have selected from the extensive set of transformation components and the arrow shows the

mapping defined by the transformation engine server. Parallelism can be achieved such that one single thread can be dedicated to each of the tasks and more with a process dedicated to the tasks. This level of Parallelism can be advantageous but after investigating further into the data I have learnt towards using SSIS where a package consists of a control flow. A control flow defines a workflow of tasks to be executed, often a particular order which is more useful with the data I will be handling. A data flow within the package makes as data flows between the selected entities (sources, transformations, destinations) which is perfect for this ETL process as well as the Complaints Administrator for whom it will be easier and faster to understand how data flows between the source and destination and the transformations done between these entities in case they need to change/add flows in the future whereas I agree that pipelines will be helpful but I feel that tasks can be done in the same manner without it.

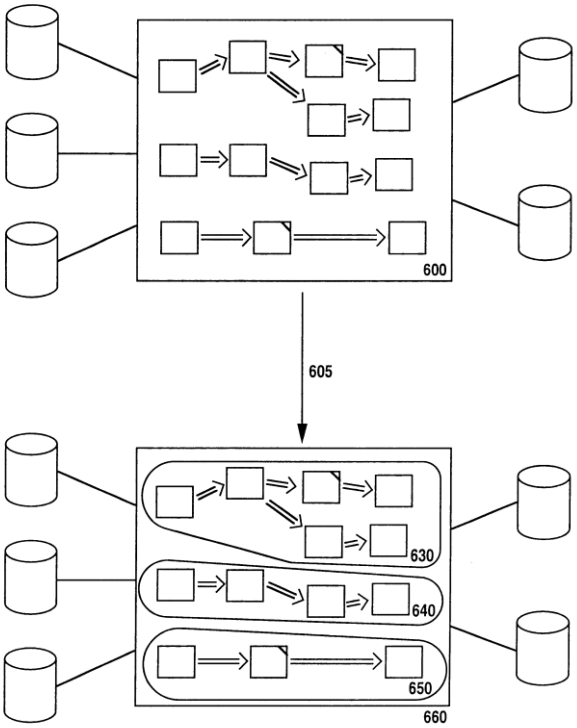


Figure 2

Object Oriented Data Structure

Different O-O implementations have different data structures unlike the relational data structure where the data structure is formalized. When object oriented programming gained prominence, there was attempts to get rid of relational databases and put Object databases in place but these attempts were not successful. Now Object-relational mapping algorithms are used where Object Oriented co-exists with an underlying relational database. [1] OO consists of object structure, type constructor and object identity.

This interest was initiated by requirements in new areas of database applications. Some applications call for data modelling capabilities not supported by traditional models of databases or current implementations of database management systems (relational databases). [2]

Unlike relational databases, the data structures of objects in an object-oriented database are encapsulated (hidden) and cannot be manipulated directly by generalised procedures. This is because generalised procedures are only possible if the data structure is known and uniform over all objects, eg. the relation or table in the relational model is a known and uniform structure, allowing generalised procedures such as query operations to manipulate tabular structures independently of actual contents. Instead, each ADO presents a public set of methods that operate over its private data structures. This allows great flexibility in the design and definition of objects while at the same time allows object capabilities to be shared and re-used amongst persistent objects.
In relation to this project I personally thought Object Oriented programming can become more time consuming and quite technical as you are required to make new classes, methods, define memory in which a lot more coding is necessary if anything changes after the project for example, new tables, relations. The Quality Consultant and Complaint Administrator is the one to be carrying out this job after and it is more beneficial to take an approach which is more simple and something that is easier to learn such as approaches traditional approaches relating to relational databases rather than to have them learn something completely new, there may be more chance of errors made.

Software Tools Considered

Informatica

The products of this software development company is based on data integration, areas such as Cloud Data Integration, Data Quality, Data virtualization, ETL, Data Replication and more. These features form a toolset for maintaining data warehouses. Informatica ETL tool ‘power center” offers many features of data transformation such as lookups, sorting/ aggregating data, transforming data with functions. Using this platform the ETL process is simplified, allows you to initiate ETL Projects in a cost effective manner. Comparatively this has greatest flexibility of your ETL process with the ability to extract more enterprise data types.
Informatica allows easy training and tool availability for software companies, most of the other ETL tools lack in this aspect. Some tools are very expensive whereas some just have to face challenges such as debugging and ease of use, looking at it in this perspective, Informatica is an ideal ETL tool. Success deployment ratio is quite high, close to 100%. During a ETL tool survey 2014, Informatica gained 4 out of 5 stars over all. For a physical table storage, the rows are partitioned.

Oracle

Oracle Warehouse Builder (OWB) is another ETL tool incorporated in Oracle which allows you to build, manage and maintain data integration processes. Further it offers capabilities for data auditing, graphical environment for relational/dimensional modelling.
Oracle Business Intelligence Enterprise Edition 11g, provides all the capabilities of that of BI Publisher so I considered this thereafter. Not only just reporting this supports many capabilities including OLAP Analytics and Enterprise reporting, access, analysis and integrated web environment. Provides a common infrastructure for all these things as well [4].

Reporting based features:

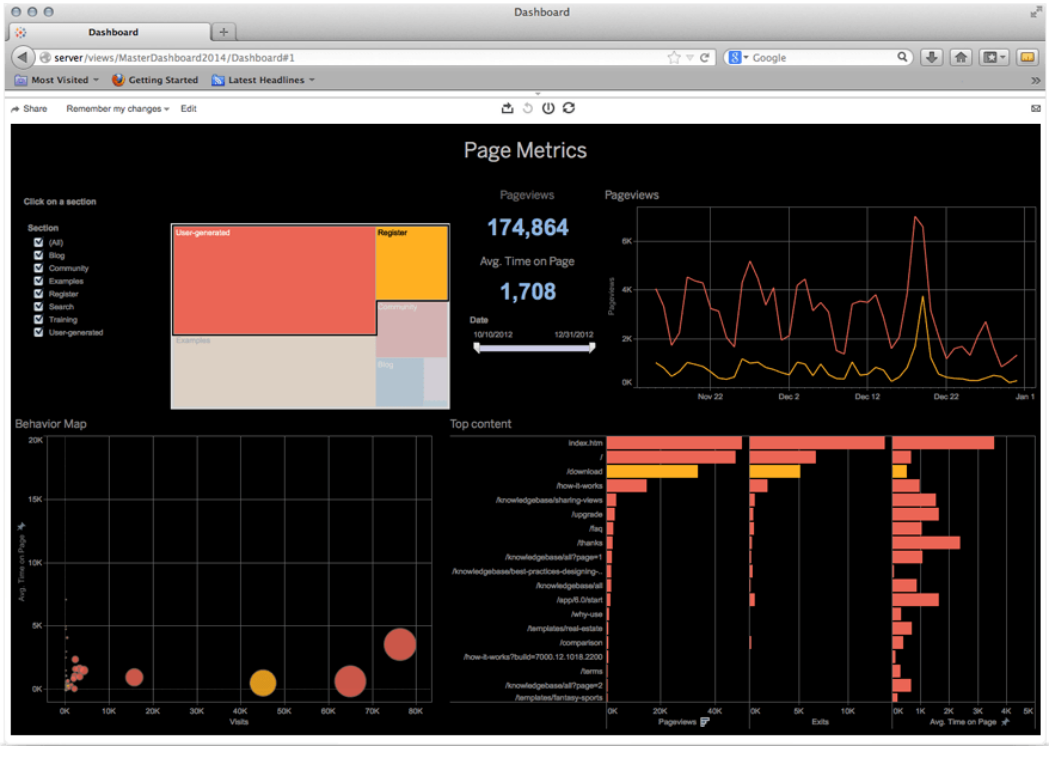
- Interactive Dashboard – Provides interactive dashboards and reports with a rich variety of visualizations. Can drill, navigating, pivot and filter. The end user is working with prompts, tables, graphics and reports in a web architecture.
- Self-serve Interactive Reporting – Enables users to create new analysis from scratch or modify existing analysis without any help form IT.
- Enterprise reporting – This is delivered through BI Publisher itself. Creation of highly formatted templates, reports. Report layouts can be easily generated directly onto a web browser using BI Publisher layout in conjunction. This does not require approrientry design studio, additional costs and reduced learning curve. BI Publisher is solely focused on reporting. Data can be combined from separate data sources into a single Data Model sharable across many reports. In this case, this capability can be beneficial as I will be using Microsoft SQL Server where the data will be stored in a star schema structure. Not an ETL tool but it can manipulate logical schemas instead of physical schemas. BI Publisher is able to sort the data and format it but this is for report purposes so Microsoft SQL server is necessary for where the data is stored and BI Publisher can be used on the surface for reporting. Output formats for the document generation can be selected between interactive, HTML, Excel, Powerpoint, PDF and batch report generation [3].

  The reports can be viewed online, sent to particular printers, fax or emails as compared to before where the Quality Consultant and Complaint Administrator will have carried out a process of first creating the graphs and tables on excel then copy and paste these onto Microsoft Word then format it then finally print/email it. Even better, reports can be scheduled. This saves time in case a product manager requests for in-depth details regarding their product and has a huge impact on time when weekly reports are already schedules to be generated, the administrator does not need to update the database first as that is going to be automated unlike the current system there is a need to update the data then create updated reports on top of that.

The layout can support 150 languages and includes automatic internationalization of date and time which can be helpful for collaboration with overseas branches of Orion Health.

Tableau

Tableau's aim is to equip people to help them see and understand data. This is a software which helps build reports and visualizations in tableau desktop then publish to tab server for sharing. A range of capabilities relating to visual analytics in areas such as data discovery, data visualization, Business Dashboards, Mobile Business Intelligence and Time Series Analysis. Time Series Analysis plays a huge role in business decisions where you are able to perform time comparisons, find trends and analyse by time, week, month or year.
A huge advantage that comes along with Tableau is the ease of use, the software is very user friendly. It is not a must that the user has to have technical skills in order to use this effectively, people such as business users, senior management are able to use this as well. A reason for this is that you are able to create queries without having to physically code due to a drag and drop facility. Whereas SSIS is more developer centric reporting application. Similar to Oracle and unlike the current process in place updates are done automatically so you are able to get up to date data with a live connection and also on a schedule if defined. Another feature which is similar was to be able to share live on the web, using Tableau mobile devices can also be used for this purpose. It would be more interactive and professional for the administrator to create this and share the web Brower around to the department regarding their product rather than printed pieces of paper of graphs posted on a word document. Orion health uses this software as well as Excel so I thought it would be good to use Tableau and explore more features.



SOLUTION

My goal is to automate both report and data retrieval which will free up a lot more resource, time and effort. On this note the accuracy will also increase and the ability to explore more factors which can be used to support product quality. To achieve this, initially I have decided to build a Data Warehouse because of the following reasons:

- The data is stored within an operational systems (OLTP) is not easily accessed.
- The data in OLTP is constantly changing due to incoming complaint tickets but the report that are generated on excel are not retrieving up-to-date data
- Diverse forms of data stored in different platforms or dissimilar formats
- OLTP systems are not designed for analysis or even end user-analysis
- Can be deficient in historical data

This data warehouse will store current and historical data that are used for creating trending reports for senior management reporting such as annual and quarterly comparisons. I decided to carry out an Extract, Transform, Load (ETL) process to load the data warehouse regularly so that it can serve its purpose of facilitating business analysis effectively. Then I researched the different ETL techniques and also different tools to carry out this process which were discussed in the Related Tools section of this report. I came to a conclusion to use Microsoft SQL Server to carry out the ETL process using SSIS which I talk about next in the report. Why this approach was selected will be discussed as this is the software I have decided to use throughout my project. The second half of my project dealt with reporting and researching about the different tools and approach for reporting which reveal more about the products and benefits to management.

Selected Tools

Microsoft SQL server

This is a commercial relational database management system whose function is to store and retrieve data as requested. Access is similar but it is a database which is more of a database management system that permits you to build an interactive UI where the end user is able to enter data and report, however Microsoft SQL Server does not have a front end but allows you to store data. SQL Server manages many services such as relational databases, Integration services, Analysis Services and Reporting Services which is one of the reasons why I think this can be an effective management system in relation to this project. Data is stored in a database.

The SQL server upon it is able to support the different types of data types (text, numeric, float, char, Boolean). Larger volumes of data can be stored including objects such as video, text and photographs. Access has the same capability but it can handle only 1Gigabyte of data as compared to Microsoft SQL server which can handle terra bytes.

Concurrent access is taken care of by SQL server in two modes, first being pessimistic concurrency and the other, optimistic concurrency. Locks are used when there is need to control access in the pessimistic mode. Optimistic concurrency control differs in the sense that it allows a new version of a row to be created when the row is updated instead of replacing or overriding the row. This can be useful when multiple access into the system is being made within Orion Health.

Many more features exist as mentioned earlier but not all are necessary to function on the database, it depends on what you are looking for. The add-on services that I will be looking at is to do with data integration and reporting. Microsoft SQL Server supports better security features such as strong authentication, efficient access control, permission management tools and better performance which is why I choose to keep with Microsoft SQL server instead of Access due to the large amount of confidential data Orion Health wishes to store.

Business Intelligence Development Studio

BIDS is the IDE (Integrated Development Environment) for Microsoft utilizing the Reporting, integrating services which means it is a software that provides facilities to programmers for software development. It is equipped with service specific extensions such as projects for reporting services will use this type of extension and things such as data mining structures will use extensions corresponding to Analysis services.

SSIS (SQL Service Integration Services)

DTS was built upon to contain more transformation features and was replaced by SSIS in 2005. This allows for ETL capabilities for SQL Servers. SSIS is consists of GUI tools to create workflows such as extracting data, querying data, transforming data, exporting data to destination. Used entirely for Data integration and workflow applications. To avoid out problem of taking too much time for reporting which concerned the data stored not kept up to date, SSIS resolves this issue as it used to automate maintenance for SQL server databases and updates. Using this, data will be kept up-to-date rather than having the Admin to update every time a report needs to be generated causing the whole cycle to take place continuously.

Because Business Intelligence Development studio is the IDE for integrated services, SSIS can be developed though this. There may be various data sources from which the data needs to be considered, this will not be an issue when using SSIS as one is permitted to use heterogeneous data sources so this is beneficial for the future when necessary. This can also be applied in my case where I was given separate tables with rows and columns in a csv format, I was able to import and preform SSIS operations effectively using Business Intelligence Development Studio.

However SSIS is more developer centric as compared to software such as Tableau but Microsoft SQL Server gives the ability to store data, preform data integration using SSIS and also reporting services using SSRS all in one place whereas if I was to use Tableau I would still require a place for storing data and then use Tableau only as a surface. Similar case with Oracle, I did not go ahead with that because Orion Health is already using Microsoft SQL Server to perform their transactional data even though if it is not in the best structure so it would be best to use the same platform and prove the system from there rather than to shift all the data into an Oracle platform therefore I decided not to use Oracle Business Intelligence Enterprise for the ETL process as this would also be a cross platform just like using Tableau with Microsoft SQL Server. So to keep everything all together in an effective and efficient manner.

SSRS (SQL Server Reporting Services)

This service generates reports in a RDL file from the data collected from the SQL server database. Microsoft Visual Studio produces the same outputs as long as Business Intelligence Development Studio is installed. We are not limited to RDL files when it comes to output, RDL file can be rendered in any other format such as PDF, XML, CSV and more.

Implementation

Creating database/ETL Process using SSIS

Using Microsoft SQL Management Studio I connected to my own server using (local)\SQL.EXPRESS in order to create a database as I did not have permission to use Orion Health server. For the ETL process to take place later I decided to create two databases as shown in figure 1.2, one for staging where the database 'BTechNew' will store raw data extracted from the support system and the other for the destination database 'BTechNewDest' where the transformed data will be loaded into during the ETL process, however it is empty initially. The CSV files provided are imported to the BTechNew

database.

These are the 6 tables I have been working with as shown in Figure 1.3. Data has been extracted from different data sources in order to create this database. The tables have been pulled from the support system by querying out the columns to make a table which is then exported into a csv file due to the fact that the data in the support system is raw, unstructured and kept in an unreadable manner. These flat files are then used to import into Microsoft SQL Server where they use different data organisation and format. In many cases this is the most challenging aspect of ETL, since extracting data correctly sets the stage for the success of subsequent processes.

Instead of the website being linked to the support system, data entered can be linked to Microsoft SQL Server so the data is kept all together in a structured form which will automatically take part in the ETL process leading to efficient data retrieval.

An intrinsic part of the extraction involves data validation to confirm if the data pulled from the sources have the correct values (such date as the type for Arrival/Pre-completed columns). In case the data fails the validation rules it is rejected entirely. The rejected data is reported on the error message for further analysis to identify and rectify the in-correct records. My sample data had a lot of issues with the types where I had worked on resolving this for a long time while trying out various approaches. The columns with dates were set to the type Date but data was not being imported due to loss of information therefore in the end I set up an empty table with the field type set to Date and finally the data was appended to the table successfully.

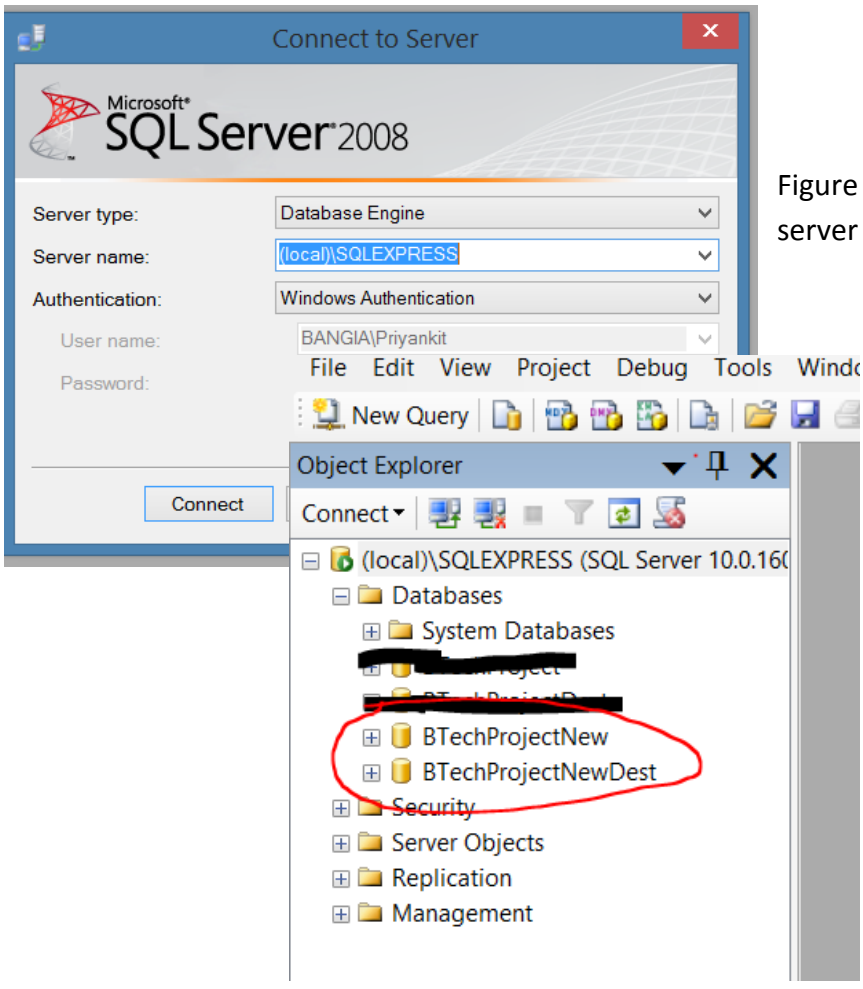


Figure 1.1 – Connecting to local server

Figure 1.2 – staging database containing information from support system and destination database 'BTechProjectNewDest'

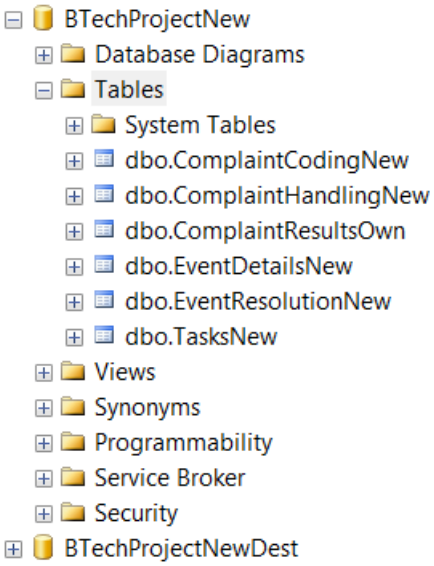


Figure 1.3 – Imported CSV files into Microsoft SQL Server which were exported from the support system using queries to create these tables.

Analysis of important data the Complaint Administrator uses for reporting and analysis purposes was done by viewing their current reports and also communicating. These tables contain metadata about certain things which is also shown by the following images of the tables (dimension tables – CSV formatting) which you will have to zoom into see the details.

EventDetails- As shown in figure 2.1 the table EventDetails contains all the information about a particular ticket (complaint) that has been made which is given a unique id, the eventid. After carrying out an analysis from the current reports which the Complaint Administrator generates on a monthly/weekly basis I analysed which information was used and which was not. A lot of important information is embedded within this table such as the application (product) the issue is regarding with the corresponding version number of the product as well the responsible team, diagnosis, arrival of the ticket and which clientid to see who it came from is stated. There is some data that is not majorly used for reporting such as Clientid, RequestType(whether it is an issue or a technical enquiry), level of issue, state(closed or open) and synoposis on the issue is not used when reporting to the management. There are two fields related to client and supportStaff for example clientid, Client, SupportStaff and Support Staffid where the names themselves can be taken forward, it is unnecessary for the id's to be included in the final destination although the Id's can be dug out from the dimension tables when required.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	eventId	applicationId	Application	versio	Client	RequestType	Priority	Responsible	State	Diagnosis	Arrival	synopsis	contactPersonName	clientId	SupportStaff	SupportStaffId
2	ABCDE	00:00.0	RHA	5.2.0	VIS	Issue Report	Level 2 - Production Degradation	NA Region	Closed	Other	49:28.5	Rhapsody Production Crash	Mano M	abc	Michael Stanford	michael.s
3	BCDEF	00:00.0	RHA	5.2.0	BAN	Issue Report	Level 3 - Functional Impact	NA Region	Resolved	Product Bug	27:06.1	Rhapsody 5.2 Crash 12/31/2012	Christof C	abc	Terry Pan	terry.p
4	HIJKL	00:00.0	RHA	3.2.0	BAN	Issue Report	Level 1 - Production Failure	EMEA Region	Closed	Environment Issue	57:29.6	rhapsody failure	Dasarsh D	def	Terry Pan	terry.p
5	IJKLM	00:00.0	RHA	5.2	VIS	Technical Question	Level 4 - No Production Impact	NA Region	Closed	Not Specified	32:13.5	Question on comm point icon in Management Console	Sid S	ghi	Tony D	tony
6	JKLMN	00:00.0	RHA	5.1	SAT	Technical Question	Level 4 - No Production Impact	NA Region	Closed	Config Advice	12:13.1	Want to test messages	Saad S	jkl	Michael Stanford	michael.s
7	MNOPQ	00:00.0	RHA	5.2.0	JOT	Issue Report	Level 3 - Functional Impact	EMEA Region	Closed	Environment Issue	38:21.0	Rhapsody not processing messages	Rina R	mn	Terry Pan	terry.p
8	OPQRS	00:00.0	RHA	4.0.0	SAT	Issue Report	Level 3 - Functional Impact	EMEA Region	Closed	Environment Issue	01:18.6	Interface problem?	Dave D	opq	Tony D	tony
9	CDEFG	00:00.0	RHA	5.4	JOT	Issue Report	Level 3 - Functional Impact	EMEA Region	Closed	Product Bug	10:41.8	CAnnot add new segments to HL7 with database looku	Saketh S	res	Samuel Wills	samw
10	KLMNO	00:00.0	RHA	3.4.0	CHR	Issue Report	Level 4 - No Production Impact	NA Region	Closed	Account Requests	46:34.2	License file sent not working	Mandy M	tuv	Michael Stanford	michael.s
11	PQRST	00:00.0	RHA	5.4.0	CHR	Technical Question	Level 3 - Functional Impact	EMEA Region	Closed	Release request	30:29.5	NHS Communication Point	Kunaal K	wxy	Terry Pan	terry.p
12	GHIJK	00:00.0	RHA	5.4	RIN	Technical Question	Level 4 - No Production Impact	NA Region	Closed	Product Bug	52:13.7	SNMP MIB and Documentation	Ashwin A	sdk	Samuel Wills	samw
13	FGHIJ	00:00.0	RHA	5.1.0	VAD	Technical Question	Level 1 - Production Failure	EMEA Region	Closed	Environment Issue	51:58.9	we have a production interface down and need urgen	Nizam N	kse	Michael Stanford	michael.s
14	EFGHI	00:00.0	RHA	5.2.0.(	DAV	Issue Report	Level 1 - Production Failure	EMEA Region	Resolved	Product Bug	51:02.6	Incorrect Date in Acknowledgements	Lawrence L	knf	Felisa Pan	fel
15	DEFGH	00:00.0	RHA	3.4.0	RIN	Issue Report	Level 3 - Functional Impact	EMEA Region	Closed	Environment Issue	24:53.5	rhapsody 3 not processing trak adt messages	Xu X	mwo	Terry Pan	terry.p
16	LMNOP	00:00.0	RHA	5.4.0	SIV	Issue Report	Level 3 - Functional Impact	NA Region	Closed	Other	13:39.7	JavaScript	Michael M	wee	Michael Stanford	michael.s
17	GHIJK	00:00.0	RHA	5.5.0	SAA	Issue Report	Level 4 - No Production Impact	NA Region	Suspended - Client Feedback	Not Specified	45:40.0	Unclean Shutdown Detected	Mano M	boo	Michael Stanford	michael.s
18	DEFGH	36	RHA	5.5.1	NIZ	Issue Report	Level 4 - No Production Impact	NA Region	Resolved	Product Bug	12:16.3	Server app aborts	Christof C	mei	Tony D	tony
19	RSTUV	138	LAB	8.3	NIZ	Technical Question	Level 4 - No Production Impact	Projects Admir	In Progress	Not Specified	45:34.5	Interpreted value from HL7 not update in HIS	Dasarsh D	pll	Felisa Pan	fel
20	STUVW	150	RAD	8.3.13	SAA	Issue Report	Level 2 - Production Degradation	HIS Region	Suspended - Client Feedback	Not Specified	10:25.8	Cannot send to PACS	Saketh S	bbb	Terry Pan	terry.p

Figure 2.1

ComplaintHandling – Contains information about the complaint handling for example if the complaint has been taken action on and resolved the issue as shown in Figure 2.2. From analysis VigilanceReporting and field of Action of a complaint is looked at most even though it is not used in reporting.

	EventId	vigilanceReporting	fieldAction	complaintClosure
1	15922	00:00.0	null	1
2	16900	00:00.0	1	1
3	20870	00:00.0	1	1
4	21454	00:00.0	1	1
5	21892	00:00.0	1	1
6	22059	00:00.0	1	1
7	24602	00:00.0	1	1
8	24791	00:00.0	1	1
9	27748	00:00.0	0	1
10	28636	00:00.0	1	1
11	29293	00:00.0	1	1
12	29583	00:00.0	1	1

Figure 2.2

ComplaintCoding contains vital information related to complaints and is heavily used for reporting and analysis. Relates to when the Global Consultant manually codes a complaint into categories such as ‘Envrionment’, ‘Defect’, ‘Maintenance’, ‘By Design’ and things as such in the ‘problemCode’ field. Problem Area(browser, IDE, backup) is broken down into more detail from problem code and problemDetail from ProblemArea. These 3 fields are used for reporting to see the types of problems occurring with versions of products in order to compare. When using this information for weekly reports, description of problem are and detail, ‘problemAreaDes’ and ‘problemDetailDesc’ as well as the last modified is not used. Figure 2.3 shows the structure of the table, the data has been taken out due to confidential data.

	A	B	C	D	E	F	G	H
1	EventId	problemCode	problemCodeDesc	problemArea	problemAreaDesc	problemDetail	problemDetailDesc	lastModified
2								
3								
4								
5								
6								
7								
8								
9								

Figure 2.3 – ComplaintCodingNew Table

Tasks – This table elaborates more on the investigation which is not used and is not as useful unless there is need for in depth analysis of the completion of the pre/post investigations and if the complaint is being handled at the time. To begin with, the eventid column was empty in the ComplaintResult table so I used the tasks table to link the documentId as shown in figure 2.4 to find the corresponding eventid to move on futher with the ComplaintResults table.

EventId	taskType	taskStat...	taskStatusNa...	documentId
40562	Post Investigation	4	Completed	2b0b5705-76e8-4240-9191-eff68cae7624
40562	Pre Investigation	4	Completed	e5356d28-7c4d-4b75-80db-e21a612ee8f1
40561	Post Investigation	1	New	NULL
40561	Pre Investigation	4	Completed	9bc3477e-9753-4b37-943f-e82fd3ebe9df
9227	Ticket Coding	4	Completed	28fc3298-a7b5-4e89-bffd-c22dfe906545
8981	Pre Investigation	1	New	a845cbae-ac2c-4f6e-a8ad-9f95168f521e
64	Complaint Handling	1	New	ComplaintHandlingnull
5950	Post Investigation	4	Completed	36bb205a-3460-45be-81c2-d03509f24c42
5950	Pre Investigation	4	Completed	2e46c2df-78dd-4654-bd06-2a6ff40d18f3
57	Pre Investigation	1	New	ddb45615-9190-41b7-93cd-0fbdd5603c27
41786	Pre Investigation	1	New	c7b1cff9-3aa8-4d23-bd5f-7b138129c96f
41785	Post Investigation	1	New	74e14b6b-e7e3-41c5-97b2-fb423a88e499
41785	Pre Investigation	4	Completed	b3347ea7-a98e-4a52-84ea-506aaa9fb713
41619	Complaint Handling	1	New	NULL
41619	Post Investigation	1	New	506a3938-7aa3-448e-9471-63b8585c69...
41619	Pre Investigation	4	Completed	f608b0a9-430f-473b-873d-44a34ed53ba3

Figure 2.4

ComplaintResult – Contains information about the complaint preforming an analysis by having done a pre investigation which involves answering a questionnaire where the result is recorded and another questionnaire after investigation has gone into the issue, hence ‘post’ Questionnaire. The result of these two entries decide the ‘Result’. Figure 2.5 supports the current process when described earlier. Completed date is not required for reporting or looked at unless further investigation is required.

Documentid	Eventid	Type	Questio...	Questio...	Questio...	Questio...	Questio...	Questio...	Result	CompletedDa
43f6f8b4-5819-48ff-a107-06f22a5ac6b8	40466	PRE	0	0	0	NULL	NULL	NULL	0	2013-12-19
4bd2a30c-c9ab-4056-b331-576ebdae6a...	38394	POST	1	1	0	0	0	0	1	2014-08-01
5	41784	PRE	0	0	0	0	NULL	NULL	0	2014-05-15
542ed489-059c-49b5-ad0a-019fa2bc6324	38497	PRE	0	1	0	0	NULL	NULL	1	2014-08-01
57773ecd-bb6e-4dac-b397-31258aeded...	38340	PRE	0	1	0	0	NULL	NULL	1	2014-08-01
6	41784	POST	1	1	1	1	1	1	1	2014-06-15
7	41777	PRE	0	0	0	0	NULL	NULL	0	2014-07-04
72ee5350-1d75-443d-92b7-cdb51ed37...	38406	PRE	0	0	0	0	NULL	NULL	0	2014-08-01
8	41777	POST	1	1	0	0	0	0	0	2014-07-04
8db402e3-3fd6-4ede-904a-eae2b76cdc9f	39254	PRE	0	1	0	0	NULL	NULL	1	2014-04-04
9	41792	PRE	0	1	0	0	NULL	NULL	1	2014-05-01
9752aeb8-15f0-4235-b248-a95288ac65...	40567	PRE	0	0	0	0	NULL	NULL	0	2014-08-04
98fd8777-c7b4-48d0-a02f-1662d08faad2	40575	PRE	0	0	0	0	NULL	NULL	0	2014-08-04
9bc3477e-9753-4b37-943f-e82fd3ebe9df	40561	PRE	0	0	0	0	NULL	NULL	0	2014-09-02
a0c7d138-5eaf-4004-91d1-6497c6f60826	40575	POST	1	1	0	0	0	0	1	2014-08-04
a3fd801e-9fd8-4b2a-bd51-32b09a850298	40545	POST	1	1	0	0	0	0	1	2014-08-04
a4cce4bd-9a52-4dd7-9ca5-f37c04ff067e	39785	PRE	0	0	0	0	NULL	NULL	0	2014-08-15
bd2ae7f7-dc32-4657-9b2e-92c3d7809789	40542	POST	1	1	0	0	0	0	1	2014-08-04
c7c1b2f8-2443-4b84-80f6-ae49eb53769a	39785	POST	1	1	1	1	1	1	1	2014-08-15
d845970d-ff27-4310-96bb-35790a47e8c1	40598	POST	1	1	0	0	0	0	1	2014-08-04
e30751fc-c7ab-4401-8f3c-e853a6b05213	39251	PRE	0	1	0	0	NULL	NULL	1	2014-04-04
e5356d28-7c4d-4b75-80db-e21a612ee...	40562	PRE	0	0	0	0	NULL	NULL	0	2014-08-04
f3df1e43-4ea8-4d6d-b2fb-c63709d2fd90	40542	PRE	0	0	0	0	NULL	NULL	0	2014-08-04

Figure 2.5

EventResolution – specifies when a complaint has been resolved and is used just for analysis purposes but not reporting.

	EventId	ResolvedDate
1	41705	2014-05-15
2	41706	2104-05-15
3	41707	2014-05-15
4	41708	2014-05-15
5	41709	2014-05-16
6	41711	2014-05-16
7	41714	2014-05-17
8	41715	2104-05-15
9	41717	2014-05-15
10	41721	2014-05-17
11	41722	2014-05-15

Figure 2.6

Using all the tables, I decided to use a Snowflake schema(shown in figure 3.1) where the data can be seen as a blueprint of the structure of the database to keep formed integrity. I have used one fact table that refers to 6 other dimension tables (the 6 tables in the previous image) which makes it easier maintain compared to a star schema. Another reason why I decided to use a Snowflake schema is because as compared to a Star schema they are suitable for small data warehouse. The dimensions are the descriptive attributes related to fact data and the fact table holds the important information relating to complaints as you will see in an output image later on in the report. The fact table is also only found within the destination table as it contains all the important fields together which were talked about earlier during the analysis stage, it extracts the data from the sources(other dimensions)

then transforms the data and loads it into the 'FactNewDest' table in the destination ready for use as illustrated by SSIS features in figure 3.2 and figure 3.2.1 and 3.2.2 in more detail. I decided to have all those information together in one place so it will be easier and time friendly for the Complaints administrator so she would not have to query out the information required and used daily from other tables using joins.

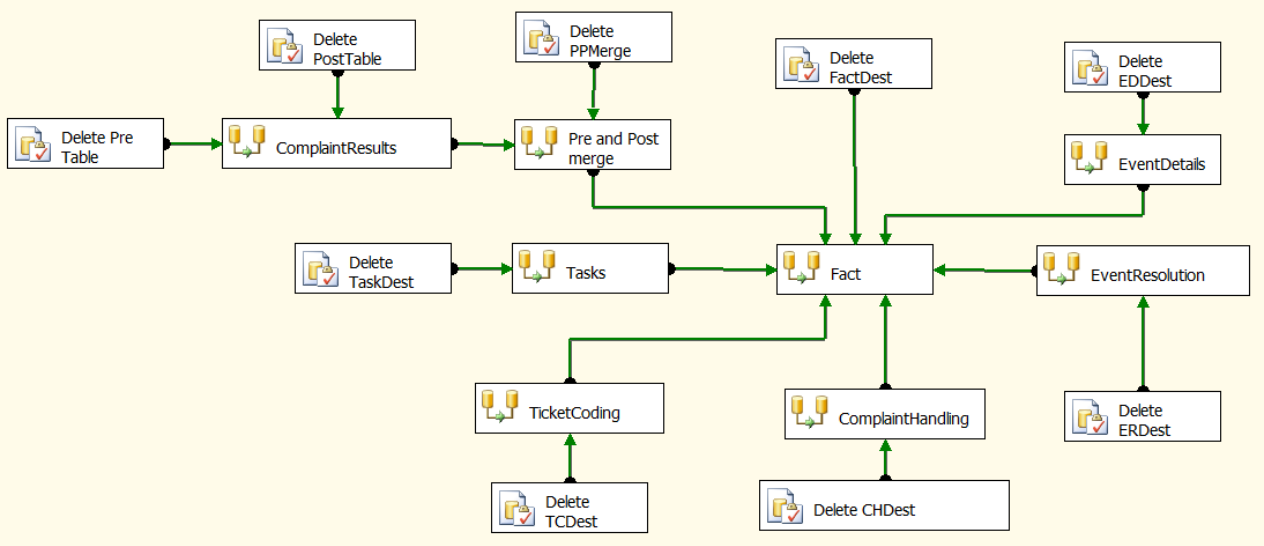


Figure 3.1

Fact table consists of low level of uniform detail whereas the dimensions define a range of characteristics. For example in this case the following dimensions are as described previously:

- Complaint Handing related details such as closer, field of action.
- Complaint results – consists of the pre and post results depending on the questionnaire with the results and date/time completed.
- Tasks – states the type of task (either pre/post investigation) and weather the task has been completed with a corresponding documented.
- Event Details – this means attributes related to an complaint that has come in, includes details such as application it concerns, client, request type (technical/report), priority of the complaint from 1-4 (Production failure to No production impact), diagnosis, synopisis, staff responsible for this issue and their id's.
- Event Resolution – the date a problem has been resolved of that eventid.
- And lastly, Ticket Coding which expands more on the problem code (DEFECT, BYDES etc) for the complaint. Expresses what it stand for and the problem details of a complaint.

Attributes which were required for daily reports that are being generated consists within the fact table and therefore queries required for reporting will be used against this information rather than the individual tables themselves. Figure 3.4 displays the structure of the output (not the real output due to confidential data). My company supervisor carried out the process of obtaining the output using the current system by manually querying and I used SSIS. Both the outcomes matched out with the right amount of rows and transformed columns. There is a huge benefit to my approach because as complaints come in, the database is automatically updated which is going to be working towards the goal of automated data retrieval and reporting rather than the manual process of gaining updated data for reporting each time. The fact table is automatically updated as well so all the important fields are placed altogether therefore the data that has been extracted to create reports will also be automatically updated. Figure 3.3 is actually the last stage of the SSIS implementation which shows the SQL query for retrieving the valid information within the fact table using Joins on the 6 table. Data within Figure 3.4 is not accurate as it is a structure of what fields I wish to retrieve due to confidential purposes and you may have to zoom-in in order to see details for the outcome.

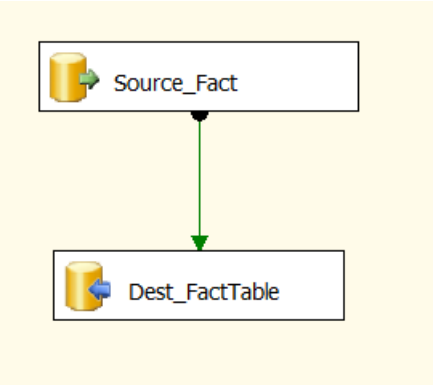


Figure 3.2 – Extracting data from all sources in the staging database then loading it to the destination in the destination database as shown in more detail below.

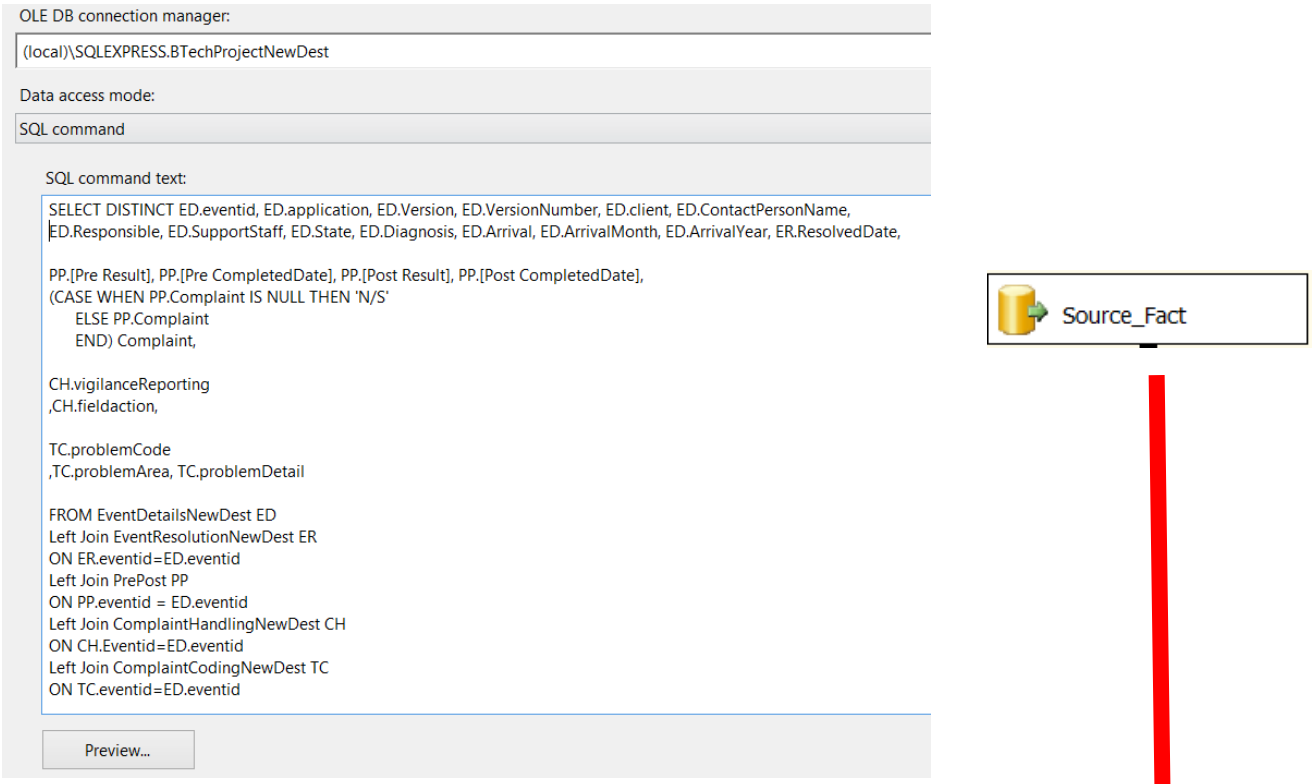


Figure 3.2.1 – uses connection as the 2nd database to extract data to use in The fact table.

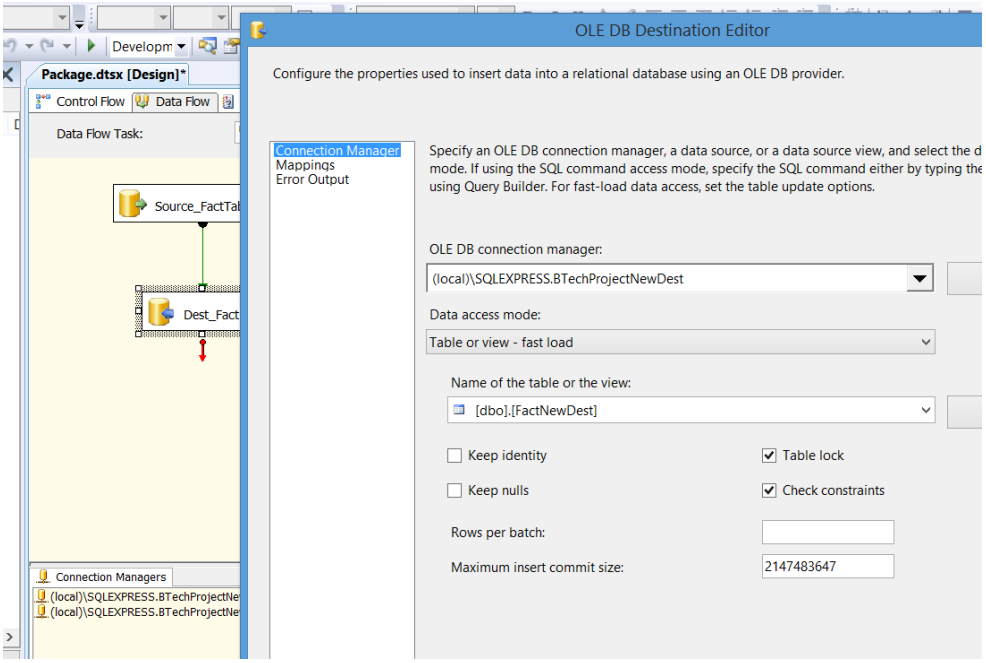
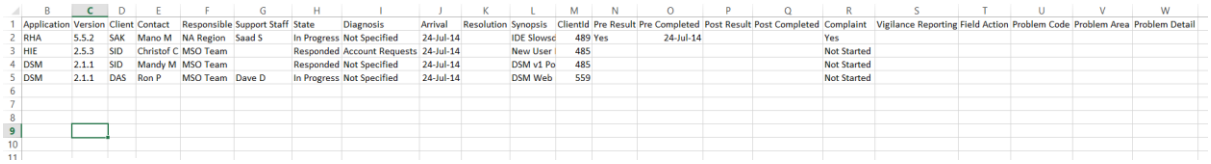


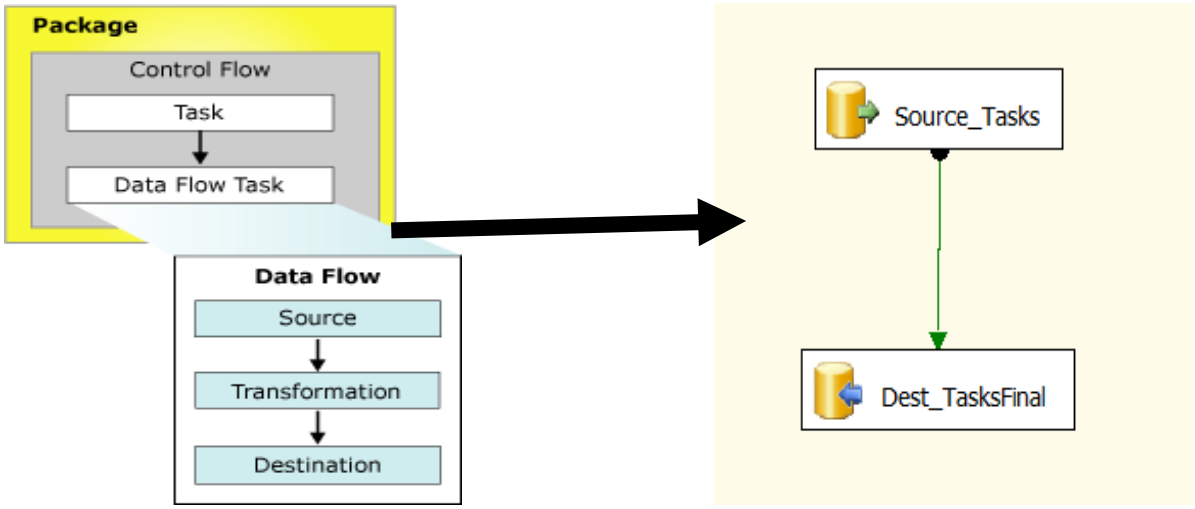
Figure 3.2.2 – The extracted data from the previous image is now loaded into the destination database ‘BTechProjectNewDest’. A new table is created called FactNewDest within this database where all the data will be stored after extracting and transforming.



	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Application	Version	Client	Contact	Responsible	Support Staff	State	Diagnosis	Arrival	Resolution	Synopsis	ClientId	Pre Result	Pre Completed	Post Result	Post Completed	Complaint	Vigilance Reporting	Field Action	Problem Code	Problem Area	Problem Detail
2	RHA	5.5.2	SAK	Mano M	NA Region	Saad S	In Progress	Not Specified	24-Jul-14		IDE Slowsc	489	Yes	24-Jul-14			Yes					
3	HIE	2.5.3	SID	Christof C	MSO Team		Responded	Account Requests	24-Jul-14		New User	485					Not Started					
4	DSM	2.1.1	SID	Mandy M	MSO Team		Responded	Not Specified	24-Jul-14		DSM v1 Po	485					Not Started					
5	DSM	2.1.1	DAS	Ron P	MSO Team	Dave D	In Progress	Not Specified	24-Jul-14		DSM Web	559					Not Started					
6																						
7																						
8																						
9																						
10																						
11																						

Figure 3.4

Coming back a few steps, control flow contains workflow of tasks to be executed. However, there a need for transformation between the Extract and Load steps with are done in the Data flow for a table. Data flow defines the flow of data from source to destination where data flows can also flow between your selected entities, source, transformation, and destination.



Looking at the following images for the complaintResult table(shown in figure 2.5) that specifies the type, pre investigation, post investigation (different rows for the same complaint), the outcome for each question, the result (if it an actual issue) done by the Regional Analyst. Because the data is redundant and structured in a very inefficient manner, data from the tasks and complaintResults table is required to be extracted then transformed to what it should be displayed as in the final outcome and stored it into the tasks table. There is also an inner join to obtain the eventid between the two tables as initially they are not stated in the complaintResult table. This transformation is used to ensure better understanding of the data by just looking at it and ease when using queries for the final output.

The transformation that took place was to have pre and post columns for one eventid (shown in figure 3.7) with the data entry of yes or no according to the result. For example the pre column will contain a yes if the result from the questionnaires was 1 or no if 0 as well as a derived complaint column to state whether the ticket (complaint) is legitimately a

complaint or not. In the current system if the analyst are wanting to know whether a ticket issued was considered as a complaint or not they are required to fetch the eventid to see the result of the Pre investigation then fetch the eventid where the type is post for the post investigation and consider both the results in order to decide. However after the transformation the data is cleansed and therefore the analyst will only be required to gain the eventid they are after, the derived complaint column will state clearly whether this is a complaint or not. This allows for easy interpretation and also avoids human reading errors. To do this I had tried two approaches to split the complaintResult table into two tables, one with eventid and Pre investigation and the completed date, same for the post investigation.

The first approach I took was to create a view of the Pre Investigations in Microsoft SQL Server Management Studio where the type is only "PRE" as stated in the WHERE Clause which gives the eventid's which have performed a pre investigation. Using this SQL query I have chosen to extract the eventid (using an inner join with documentid from the tasks table to identify the eventid to the corresponding rows in the ComplaintResults table), the result (of type pre investigations renames to Pre Investigation):

	EventId	Type	Result	CompletedDate
1	38340	PRE	1	2014-08-01
2	38394	PRE	1	2014-08-01
3	38406	PRE	0	2014-08-01
4	38497	PRE	1	2014-08-01
5	39251	PRE	1	2014-04-04
6	39251	PRE	1	2014-04-04
7	39254	PRE	1	2014-04-04
8	39785	PRE	0	2014-08-15
9	40466	PRE	0	2013-12-19
10	40542	PRE	0	2014-08-04
11	40545	PRE	0	2014-08-04
12	40561	PRE	0	2014-09-02

Figure 3.5 – Result of this query.

```
SELECT DISTINCT t.EventId, cr.Result, cr.CompletedDate
FROM dbo.Tasks AS t INNER JOIN
    dbo.ComplaintResults AS cr ON cr.documentId = t.documentId
WHERE (cr.Type = 'PRE')
```

I also create a view where the type is only "POST" from the ComplaintResults table as stated in the WHERE Clause which gives me only the eventid's which have performed a post investigation. Using this SQL query I have chosen to extract the eventid (using an inner join with documentid from the tasks table to identify the eventid to the corresponding rows in the ComplaintResults table), the result (of type post investigations renames to Pre Investigation):

	EventId	Type	Result	CompletedDate
1	38394	POST	1	2014-08-01
2	38406	POST	1	2014-08-15
3	39785	POST	1	2014-08-15
4	40542	POST	1	2014-08-04
5	40545	POST	1	2014-08-04
6	40562	POST	1	2014-08-04
7	40575	POST	1	2014-08-04
8	40598	POST	1	2014-08-04
9	41768	POST	0	2013-02-26
10	41773	POST	1	2014-07-04
11	41774	POST	0	2014-08-01
12	41777	POST	0	2014-07-04

```
SELECT DISTINCT t.EventId, cr.Result, cr.CompletedDate
FROM dbo.Tasks AS t INNER JOIN
    dbo.ComplaintResults AS cr ON cr.documentId = t.documentId
WHERE (cr.Type = 'POST')
```

Figure 3.6 –Result of the query.

	Eventid	Pre Res...	Pre CompletedD...	Post Res...	Post CompletedD...	Complaint
1	38340	Yes	2014-08-01	NULL	NULL	Yes
2	38394	Yes	2014-08-01	Yes	2014-08-01	Yes
3	38406	No	2014-08-01	Yes	2014-08-15	Yes
4	38497	Yes	2014-08-01	NULL	NULL	Yes
5	39251	Yes	2014-04-04	NULL	NULL	Yes
6	39254	Yes	2014-04-04	NULL	NULL	Yes
7	39785	No	2014-08-15	Yes	2014-08-15	Yes
8	40466	No	2013-12-19	NULL	NULL	I/C
9	40542	No	2014-08-04	Yes	2014-08-04	Yes
10	40545	No	2014-08-04	Yes	2014-08-04	Yes
11	40561	No	2014-09-02	NULL	NULL	I/C
12	40562	No	2014-08-04	Yes	2014-08-04	Yes
13	40567	No	2014-08-04	NULL	NULL	I/C
14	40575	No	2014-08-04	Yes	2014-08-04	Yes
15	40598	No	2014-08-04	Yes	2014-08-04	Yes
16	41768	Yes	2013-02-26	No	2013-02-26	Yes
17	41773	No	2014-06-04	Yes	2014-07-04	Yes

Figure 3.7 – Outcome stored in the Tasks table.

Figure 3.7 shows the outcome of the following query in the tasks table in the destination database. I preformed left joins on the two views created above, Pre and Post. This means using the Pre table, for every eventid that has a corresponding Post result and date specified in the post table will be displayed otherwise that means the post investigation has not taken place and remains null. The Also the 1's and 0's are now displayed as 'Yes' or "No' in the pre investigation and post investigation columns. The following query uses another case to derive another column called Complaint which is based on if any of the pre investigation or post investigation states a yes as described in the current process. The columns were selected and left joins were formed between the two views using the SQL Query:

```
SELECT DISTINCT
        pre.EventId, (CASE pre.[Result] WHEN 1 THEN 'Yes' WHEN 0 THEN 'No' ELSE 'No'
END) AS PRE, pre.CompletedDate AS [Pre Completed],
        (CASE post.[Result] WHEN 1 THEN 'Yes' WHEN 0 THEN 'No' ELSE 'No' END) AS POST,
post.CompletedDate AS [Post Result], (CASE WHEN post.[Result] = 1 OR
        pre.[Result] = 1 THEN 'Yes' ELSE 'Not Started' END) AS Complaint
FROM  dbo.Pre AS pre
LEFT OUTER JOIN dbo.Post AS post
ON post.EventId = pre.EventId
```

Figure 4.1 and 4.2 compares the old table to the transformed table. As you can see data can be interpreted more easily as compared to the old table where for example you were told to find whether a complaint with the eventid 38406 is complaint or not. In the old table you will be required to look up these eventid's and find the PRE then the Result for this which is 0, meaning no but however a complaint is legitimate if either of the questionnaires results to be 1 therefore you then look up the same id but where type is POST and check the result which is 1 therefore taking both the results into consideration, consider this as a legitimate complaint. The data is not structured in an easy to read manner, there is unnecessary data which is not used in between where the analyst has to keep their eye on the row and move along the columns to the result column towards the end. Human reading errors are highly possible in this case. However using the same example on the transformed data the analyst can straight away identify that the complaint is stated as yes overall and also have the ability to see if the PRE and POST investigations were initially yes or no (changed from 1's and 0's in the horizontal frame to vertical). In cases where the pre evaluation has taken place but not the post evaluation, the 'Complaint' is programmed to state incomplete and where neither of the questionnaires have taken place, in other words NULL, the complaint is stated as Not Started.

	Documentid	Eventid	Type	Questio...	Questio...	Questio...	Questio...	Questio...	Questio...	Result	CompletedDat
1	57773ecd-bb6e-4dac-b397-31258aededc9	38340	PRE	0	1	0	0	NULL	NULL	1	2014-08-01
2	4bd2a30c-c9ab-4056-b331-576ebdae6a39	38394	POST	1	1	0	0	0	0	1	2014-08-01
3	2ab470a4-2ef8-48dd-8b05-d4516976273e	38394	PRE	0	1	0	0	NULL	NULL	1	2014-08-01
4	09e3bb35-89ab-484e-9f07-5f72b1e17163	38406	POST	1	1	1	1	1	1	1	2014-08-15
5	72ee5350-1d75-443d-92b7-cdb51ed37338	38406	PRE	0	0	0	0	NULL	NULL	0	2014-08-01
6	542ed489-059c-49b5-ad0a-019fa2bc6324	38497	PRE	0	1	0	0	NULL	NULL	1	2014-08-01
7	42b3858a-4d58-4321-8fc4-11657a5aecac	39251	PRE	0	1	0	0	NULL	NULL	1	2014-04-04
8	e30751fc-c7ab-4401-8f3c-e853a6b										
9	8db402e3-3fd6-4ede-904a-eae2b7										
10	a4cce4bd-9a52-4dd7-9ca5-f37c04f										
11	c7c1b2f8-2443-4b84-80f6-ae49eb5										
12	43f6f8b4-5819-48ff-a107-06f22a5a										
13	bd2ae7f7-dc32-4657-9b2e-92c3d7										
14	f3df1e43-4ea8-4d6d-b2fb-c63709d										

Figure 4.1- old table

	Eventid	Pre Res...	Pre CompletedD...	Post Res...	Post CompletedD...	Complaint
1	38340	Yes	2014-08-01	NULL	NULL	Yes
2	38394	Yes	2014-08-01	Yes	2014-08-01	Yes
3	38406	No	2014-08-01	Yes	2014-08-15	Yes
4	38497	Yes	2014-08-01	NULL	NULL	Yes
5	39251	Yes	2014-04-04	NULL	NULL	Yes
6	39254	Yes	2014-04-04	NULL	NULL	Yes
7	39785	No	2014-08-15	Yes	2014-08-15	Yes
8	40466	No	2013-12-19	NULL	NULL	I/C
9	40542	No	2014-08-04	Yes	2014-08-04	Yes
10	40545	No	2014-08-04	Yes	2014-08-04	Yes
11	40561	No	2014-09-02	NULL	NULL	I/C
12	40562	No	2014-08-04	Yes	2014-08-04	Yes
13	40567	No	2014-08-04	NULL	NULL	I/C
14	40575	No	2014-08-04	Yes	2014-08-04	Yes
15	40598	No	2014-08-04	Yes	2014-08-04	Yes
16	41768	Yes	2013-02-26	No	2013-02-26	Yes
17	41773	No	2014-06-04	Yes	2014-07-04	Yes

Figure 4.2- transformed table

Second approach for this process – this is where I changed my star schema to a snowflake schema(as shown in figure 5.1) where I split the Pre and post tables and have them in a form of a table in the destination database rather than creating views for the two using queries as shown in figure 5.2. This data is pushed into another control flow where the two tables are merged to create the final transformed ComplaintResults table in the destination database. Data Flow Transformations from SSIS have been used directly to take advantage and reduce the SQL queries.

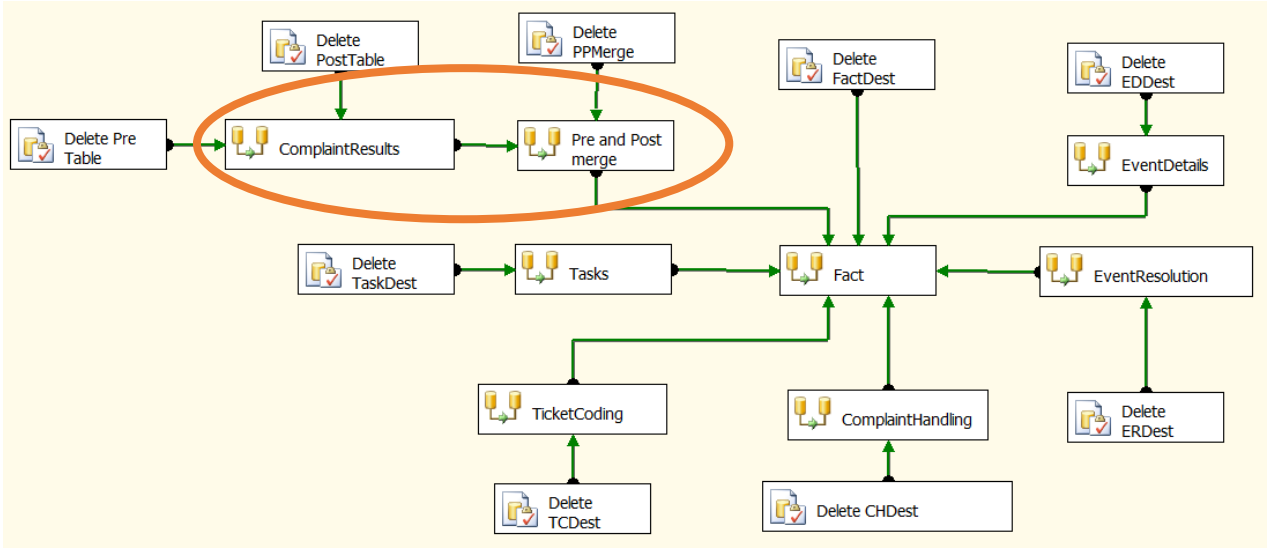


Figure 5.1

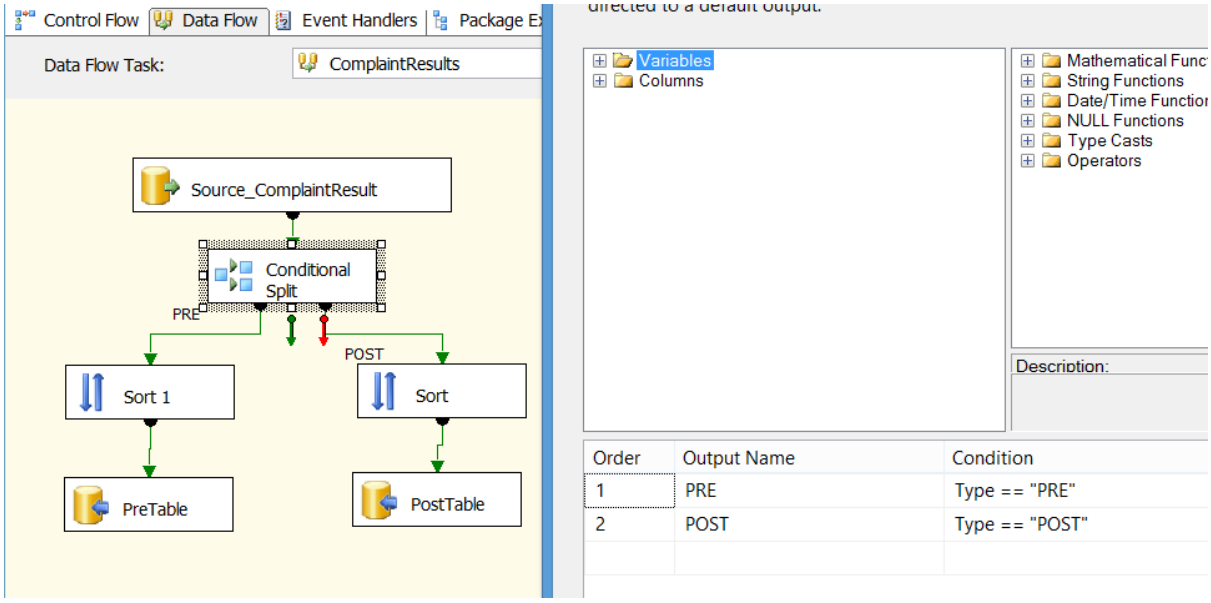


Figure 5.2 – Data flow of the ComplantResult Data flow task from the Control Flow.

EventId	Type	Result	CompletedDate
38340	PRE	1	1/08/2014 12:00:00 a.m.
38394	PRE	1	1/08/2014 12:00:00 a.m.
38406	PRE	0	1/08/2014 12:00:00 a.m.
38497	PRE	1	1/08/2014 12:00:00 a.m.
39251	PRE	1	4/04/2014 12:00:00 a.m.
39251	PRE	1	4/04/2014 12:00:00 a.m.
39254	PRE	1	4/04/2014 12:00:00 a.m.
39785	PRE	0	15/08/2014 12:00:00 a.m.
40466	PRE	0	19/12/2013 12:00:00 a.m.
40542	PRE	0	4/08/2014 12:00:00 a.m.
40545	PRE	0	4/08/2014 12:00:00 a.m.
40561	PRE	0	2/09/2014 12:00:00 a.m.
40562	PRE	0	4/08/2014 12:00:00 a.m.
40567	PRE	0	4/08/2014 12:00:00 a.m.
40575	PRE	0	4/08/2014 12:00:00 a.m.
40598	PRE	0	4/08/2014 12:00:00 a.m.
41768	PRE	1	26/02/2013 12:00:00 a.m.

Figure 5.2.1

The same goal is achieved as described previously but here the data extracted from the ComplaintResult table from the source/staging database flows into the data transformation task ‘Conditional Split’ where if the rows meet the conditions, if the type is of pre investigation then those rows flow into the ‘PreTable’ Table in the destination database and the same in the case of Post investigation. After this data flow Figure 5.2.1 shows the resulting table for the PreTable stored in the destination database. I also validated the tables by manually counting the number of pre and post rows there should be from the raw data to ensure the accuracy. For further transformations to achieve the outcome this data then flows into the ‘PrePost’ control flow. To fix this, I thought to get rid of the type column and rename the Results column to Pre Investigation then to change the 1s and 0s to ‘Yes’, ‘No’ otherwise at this stage ‘NotDone’ if the field is left NULL to show if the result of the pre investigation is considered as a complaint or not straight way without having to search deeper into the data as shown by Figure 5.3. Same goes for post invatigation. This process is carried out using a query in the “PreTable’ data source and similarly in the ‘PostTable’ Data Source illustrated by figure 5.4 as well a preview of what the data looks like after the changes have been made.

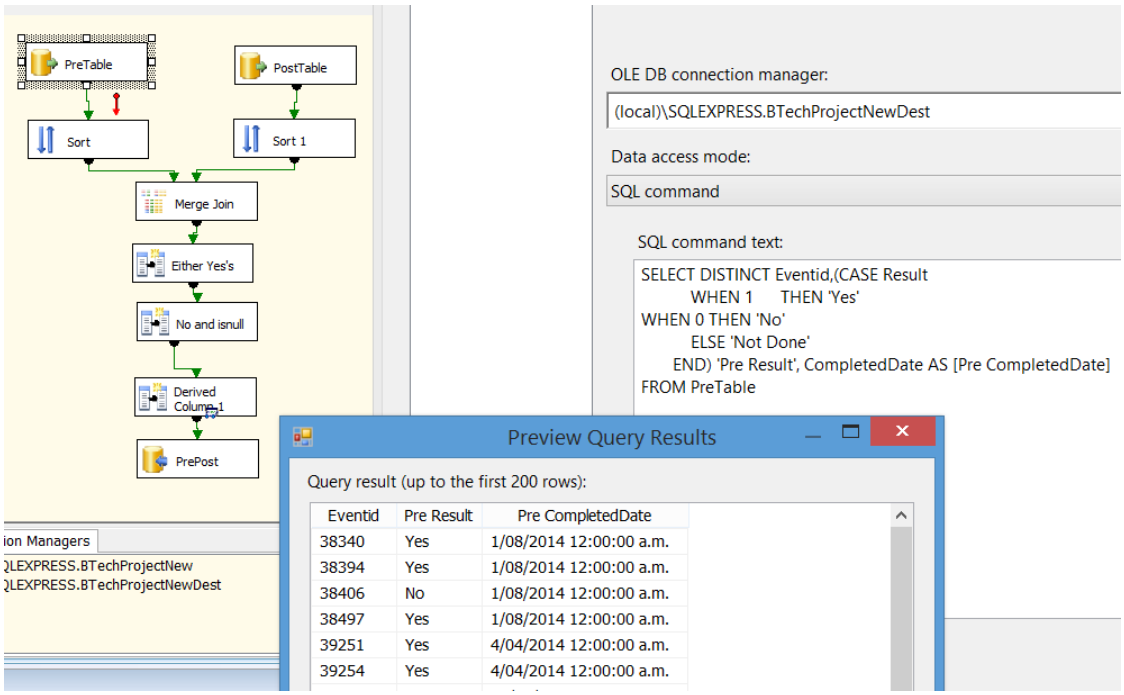


Figure 5.3 – shows the flow of data and the Extract Transform Load Process in the “PrePost” data flow as well as the “PreTable” properties.

Within the PrePost control flow, I used two sources, ‘PreTable’ and ‘PostTable’ to perform an left join to merge the two tables which returns all rows from the ‘PreTable’ with the matching rows in the ‘PostTable’. If there is no corresponding entry in the “PostTable” for an eventid meaning a Post investigation has not been carried out the field will be considered as NULL. By having done this the data is now cleansed and easier to understand. The Merge Join Transformation Editor in figure 5.4 illustrates the join.

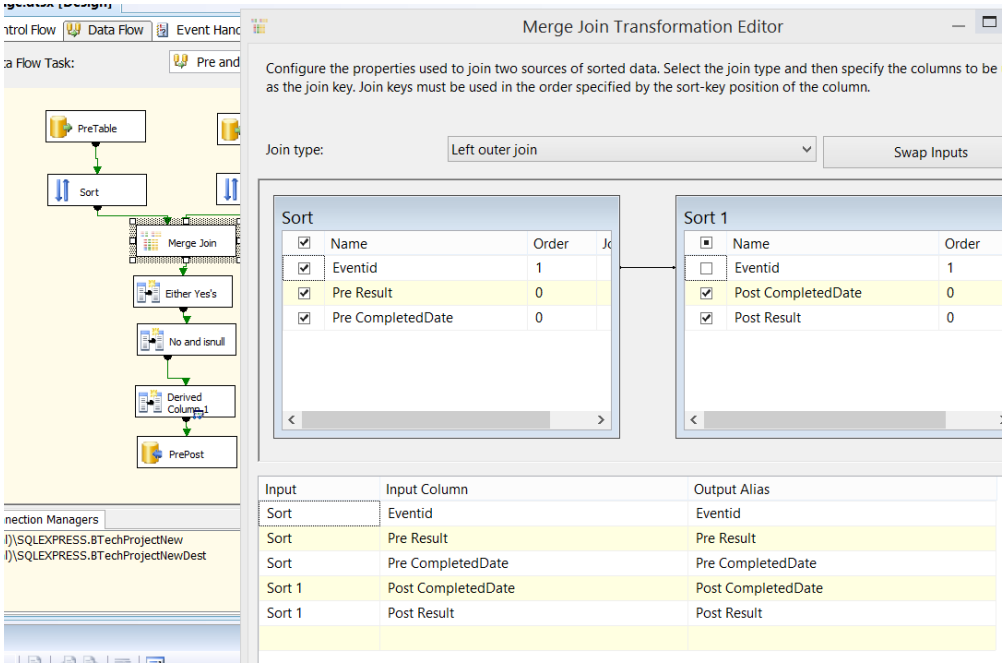
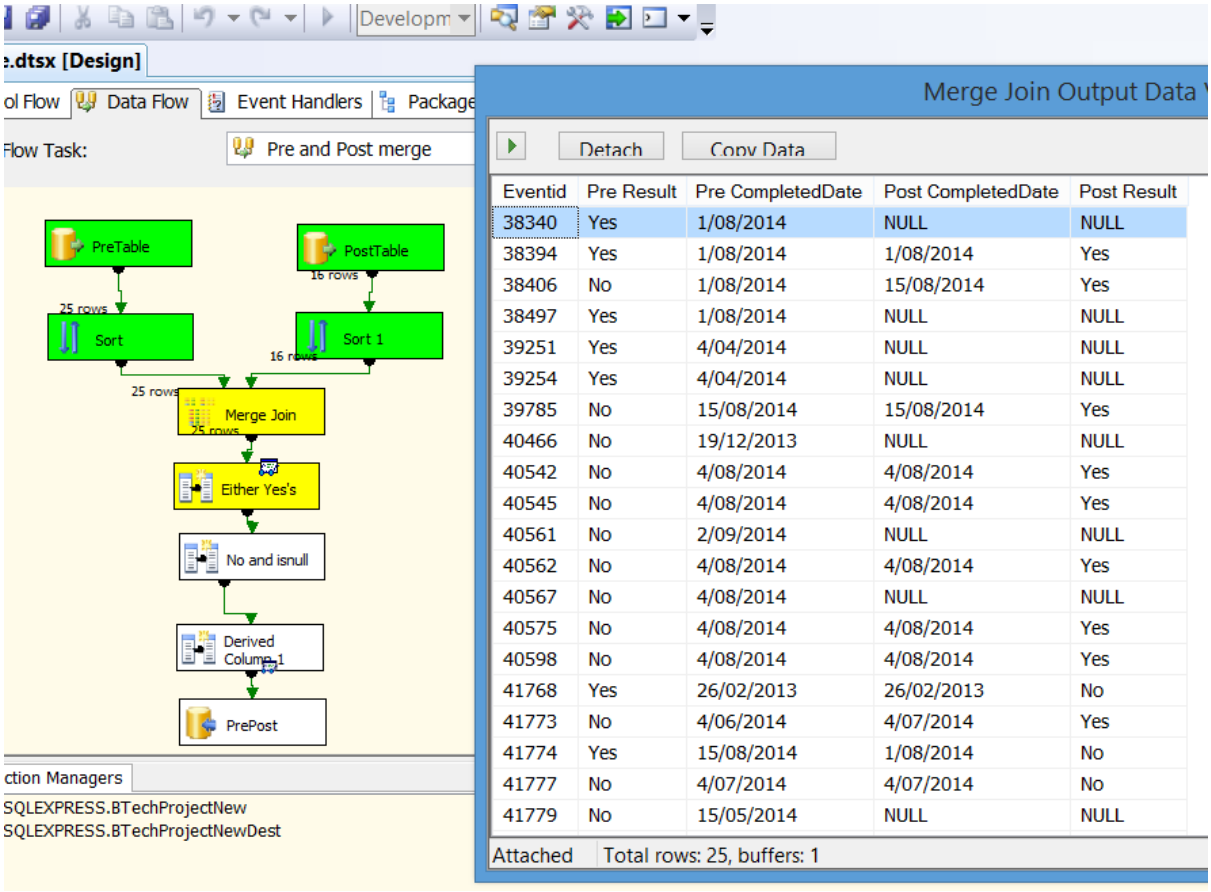


Figure 5.4 –The Merge Join Transformation editor which is preforming a left join with tables.



Eventid	Pre Result	Pre CompletedDate	Post CompletedDate	Post Result
38340	Yes	1/08/2014	NULL	NULL
38394	Yes	1/08/2014	1/08/2014	Yes
38406	No	1/08/2014	15/08/2014	Yes
38497	Yes	1/08/2014	NULL	NULL
39251	Yes	4/04/2014	NULL	NULL
39254	Yes	4/04/2014	NULL	NULL
39785	No	15/08/2014	15/08/2014	Yes
40466	No	19/12/2013	NULL	NULL
40542	No	4/08/2014	4/08/2014	Yes
40545	No	4/08/2014	4/08/2014	Yes
40561	No	2/09/2014	NULL	NULL
40562	No	4/08/2014	4/08/2014	Yes
40567	No	4/08/2014	NULL	NULL
40575	No	4/08/2014	4/08/2014	Yes
40598	No	4/08/2014	4/08/2014	Yes
41768	Yes	26/02/2013	26/02/2013	No
41773	No	4/06/2014	4/07/2014	Yes
41774	Yes	15/08/2014	1/08/2014	No
41777	No	4/07/2014	4/07/2014	No
41779	No	15/05/2014	NULL	NULL

Figure 5.5

The data displayed in Figure 5.5 is what the data looks like after the two tables undergo a left merge, however there are many little things to be transformed in the data such deriving a ‘Complaint’ column that specifies if the ticket issued is really a complaint depending on the investigations. Using a Derived Column Transformation named ‘Either Yes’s’ derives the Complaint column which derives on the bases of the expression shown in the Derived Column Transformation Editor in figure 5.6 which means if either of the ‘Pre Investigation’ or ‘Post Investigation’ columns contain a ‘Yes’ then this is a Complaint. Also it may get confusing for other instances, for example, if the result for the Pre Investigation is 0(no) and the post investigation has not taken place (in other words, is NULL) therefore we cannot determine if this is a complaint from this incomplete information as compared to the pre investigation resulting to be ‘Yes’. Another Derived Transformation is used for this case named ‘No and isNull’ and considers the complaint as ‘I/C’ (Incomplete) for such cases as illustrated in figure 5.7.

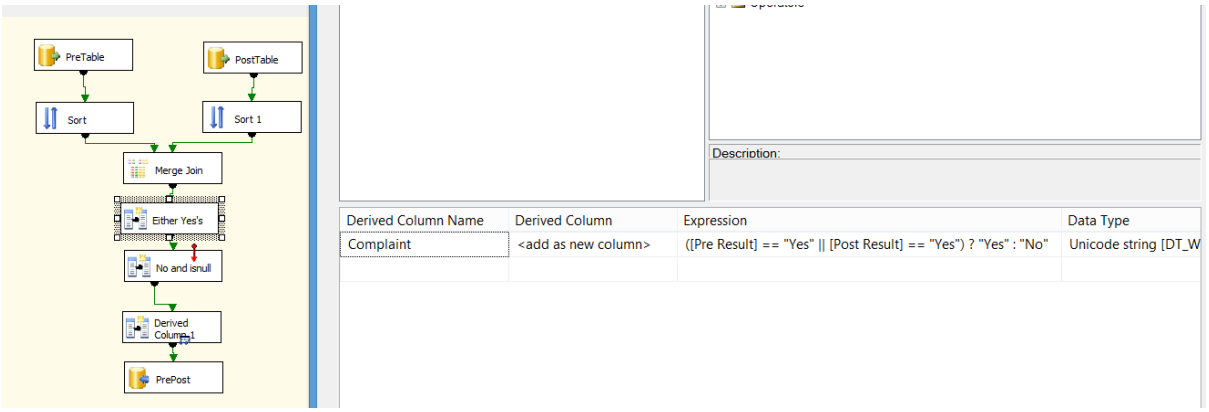


Figure 5.6 – 'Either Yes's' Derived Transformation Editor.

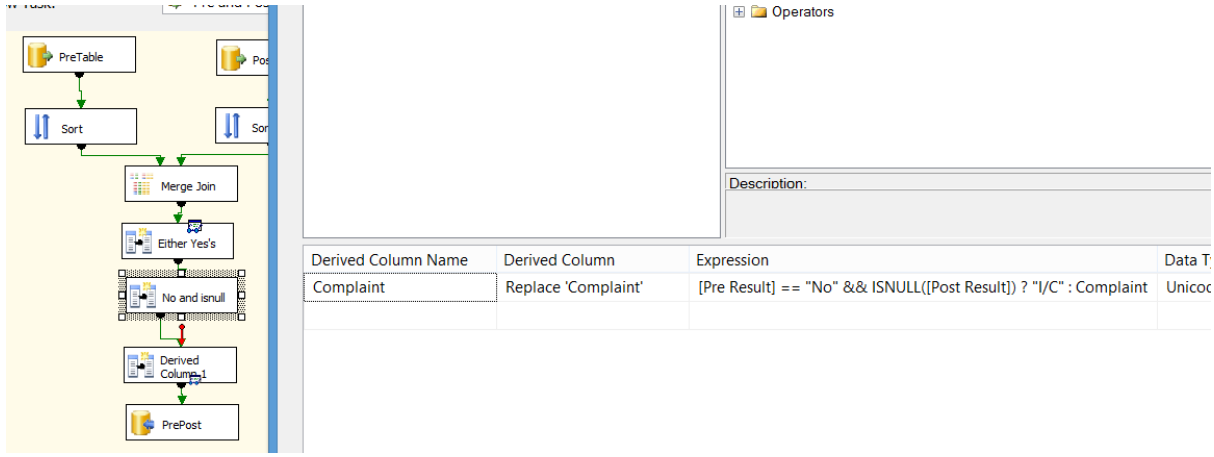
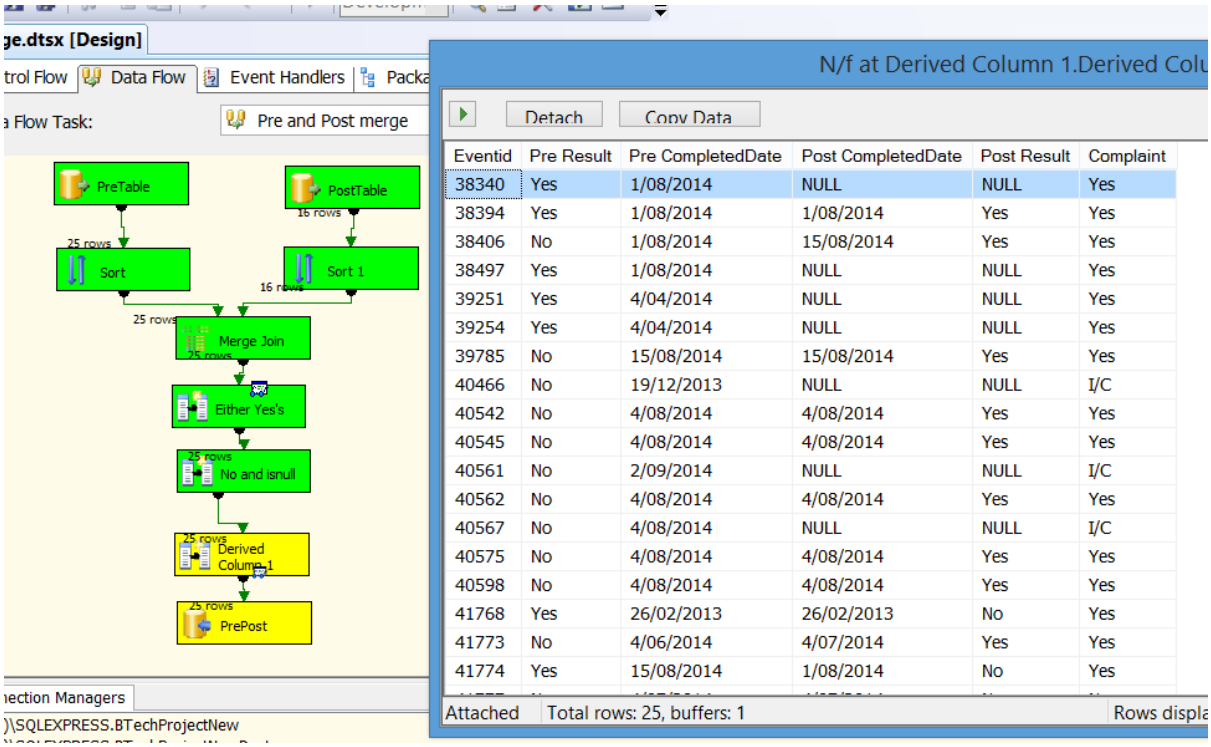


Figure 5.7– 'No and isnull' Derived Transformation Editor. No and isnull means what if it's a No for pre investigation and the post investigation is NULL.



Eventid	Pre Result	Pre CompletedDate	Post CompletedDate	Post Result	Complaint
38340	Yes	1/08/2014	NULL	NULL	Yes
38394	Yes	1/08/2014	1/08/2014	Yes	Yes
38406	No	1/08/2014	15/08/2014	Yes	Yes
38497	Yes	1/08/2014	NULL	NULL	Yes
39251	Yes	4/04/2014	NULL	NULL	Yes
39254	Yes	4/04/2014	NULL	NULL	Yes
39785	No	15/08/2014	15/08/2014	Yes	Yes
40466	No	19/12/2013	NULL	NULL	I/C
40542	No	4/08/2014	4/08/2014	Yes	Yes
40545	No	4/08/2014	4/08/2014	Yes	Yes
40561	No	2/09/2014	NULL	NULL	I/C
40562	No	4/08/2014	4/08/2014	Yes	Yes
40567	No	4/08/2014	NULL	NULL	I/C
40575	No	4/08/2014	4/08/2014	Yes	Yes
40598	No	4/08/2014	4/08/2014	Yes	Yes
41768	Yes	26/02/2013	26/02/2013	No	Yes
41773	No	4/06/2014	4/07/2014	Yes	Yes
41774	Yes	15/08/2014	1/08/2014	No	Yes

Figure 5.8 – Transformed ComplaintResult Table.

As with the previous approach, the same transformations were achieved using different ways. I preferred to keep this second approach using Transformation features in SSIS even though it makes the data flow look more intense it will be easier to follow and understand what each feature is doing as compared to using pure SQL Code for the Complaint administrator. I have tried to limit the coding level low in this project to make it less technical for the Complaints administrator and would prefer logical functioning rather than coding. Whereas queries are all clustered together and harder to understand what is happening where. SSIS allows the ability to use the features and move step by step making it easier to follow which is an advantage as things can be edited, added, removed in the future easily.

Now when executed the data is extracted from the sources(within the first database), transformed and loaded into destination(second database). There is an issue with the execution is taking place, the rows being loaded is appended onto the corresponding tables. For example when you execute the ETL process 100 rows are being loaded into a table, the next time you execute the process those 100 rows should get updated if there are any changes or if more data has been added those rows should be added to the destination. However, there will be 200 rows in the destination table now instead. This shows that same data is being appended at the end of the destination table and at a point in time the process will be very slow due to the transformations being carried out on a large amount of data. Even when a primary key is enabled in the table there will be an error when executed because the data will not take in data of the same primary key therefore the process will not be carried out at all. To overcome this issue I had taken two approaches once again and selected the best one in terms of efficiency.

My first approach involved using a lookup transformation. When executed I ensure the lookup transform preforms lookups in the destination table to check if all the fields, row by row is the same (matching) as the source data and the fields are replaced with the new data coming from the source if the data has been changed or append new entries into the destination table. When the amount of data increases, the transformation has to go through more and more fields, row by row, column by column and therefore the time in order to carry out this process increases as the lookup transformation is used on all the tables after other transformations as well. The efficiency decreases due to this fact. Therefore I decided to use my second approach in my project where I delete the corresponding destination table before the data is Extracted, Transformed and Loaded into the destination table.

Previous data will be loaded into the database so no data will be lost. All the changes/updates made to the entries as well as the new data will be contained in the destination when loaded rather than looking up every field to check if it is still the same otherwise replaces the old value. For all the tables I use a Control Flow Item called Execute SQL Task to truncate the table. The reason I have separate Execute SQL Tasks for each of the tables rather than the tables all together is because the tasks are carried out in an order therefore in case of an error other tables which have not been reached get deleted from the destination.

Even when using the second approach, to validate the data I have used a lookup transformation again to match the eventid from the data source to the eventid in the destination table. For example, the FactDest table in the destination database has been cleared and now the lookup transformation checks to see any matches in the destination table where we can ensure the table is empty as you can see in figure 5.9 the source has 19 rows and there are 19 non-matches rows which is then loaded into the FactDest table. The Lookup Transformation ensures the ETL process is doing what it is meant to and data is flowing I a proper manner. I am also able to take out the Lookup transformation as I have ensured this ETL process is working therefore less time during the process, increasing efficiency of the system.

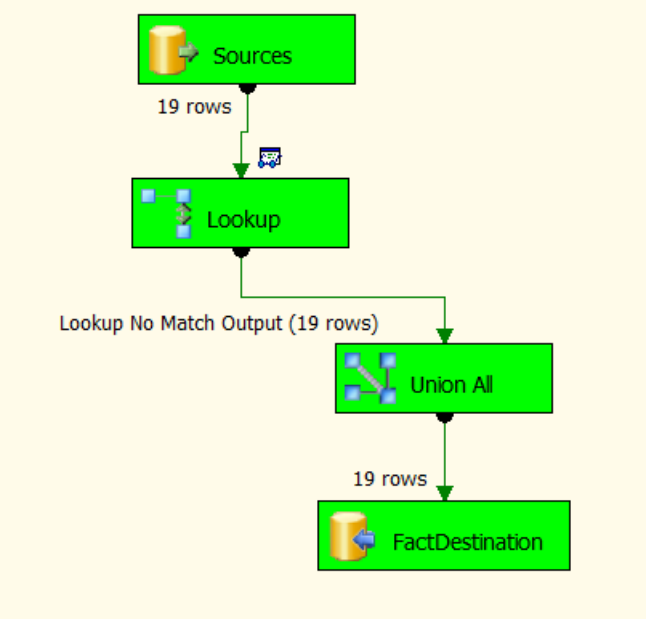


Figure 5.9

Reporting

Now the ETL process has been dealt with, I used SSRS for reporting. I generated reports after getting to understand what sorts of analysis the Complaint Administrator undergoes and the types of information they present to the management weekly. The following reports are used on a weekly basis where the data was required to be queried and transformed then exported every single week to excel for reporting due to the data in excel not being up to date every week.

- Tickets over time (displayed as a bar chart by month) with complaint status (i.e. Yes, No, Incomplete, Not Started) displayed by colour
    - For application RHA and CON
    - With PAL (Client) tickets filtered to be removed from the display
    - With Diagnosis type “duplicate” filtered to be removed from the display
    - With the Responsible team MSO region filtered to be removed from the display (Responsible refers to the team responsible for the complaint)

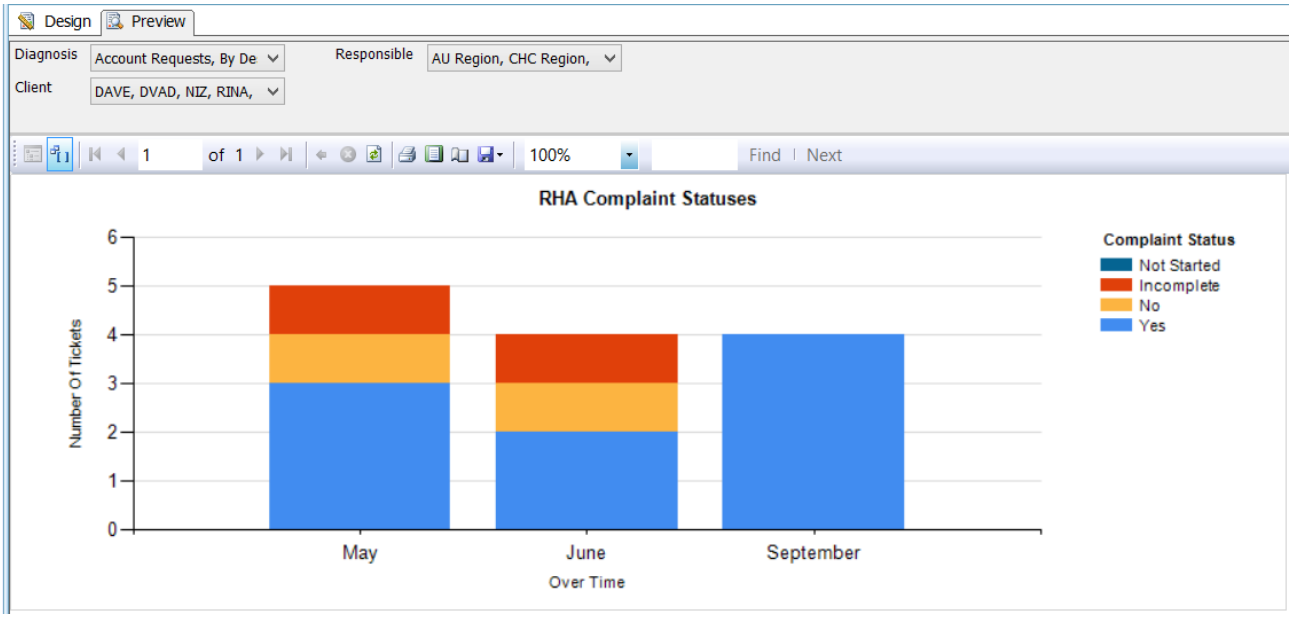


Figure 6.1 – Tickets Overtime by Complaint Status for the RHA product.

This report is of high importance because it shows the product manager an overview of the total amount of complaints coming in which is differentiated by complaint statuses so the progress can be easily interpreted as well for example, the manager can push the analyst to work faster if there are a lot of incomplete investigations or if a large amount of complaints have not been looked at. This works as a guideline

which heads towards helping keep track of product complaints for monthly comparison. If the number of legitimate complaints is starting to increase then the Complaints Administrator is able to point out these facts to the product manager to focus on. More information can be dug out which is most probably contained in the following reports I have generated otherwise information can be easily provided due to the structures and easy to interpret data from the SSIS stage. I have created a chart for RHA and also CON as shown in Figure 6.1 based on these requirements.

Instead of providing the Complaints Administrator with a normal report where they do not include data such as Client - PAL, Diagnosis – duplication and the MSO region responsible team, Parameters are used for filtering. Firstly, a new datasets for 'Responsible' with distinct value is created so the Administrator has the ability to choose from the different teams. For example in figure 6.2, the dataset contains all the teams there by using the query displayed. Next, the Parameter is created and linked to the corresponding datasets of 'Responsible' which means that now there will be a Responsible parameter displayed on top of the report which is a drop down list containing the data from the Responsible dataset. Moving from design to preview mode a parameter must be selected in order for the report to be generated therefore I selected all the options available as the default option as you can see on the top in figure 6.1 so the Administrator can view everything then de-select what region they do not want to see the data for.

To have this fully functioning on the chart for the corresponding data from the parameters selected, the filters of the chart need to be set in the Chart Properties. Figure 6.3 explains this. For example, after moving into the design mode if the Administrator does not want to include the complaints that relate to the PAL region, they can just deselect it from the responsible parameter as illustrated in figure 6.4. I have made Parameters for Diagnosis, Responsible and Client as it will be easier to fiddle around with the data to find out in depth information and trends rather than having to create new reports based on this data therefore this is easier and quicker.

After getting a hang of using and implementing parameters I wanted to make this report generic rather than having separate reports for the two products which is limited so I created another parameter for product which makes this report more generic because the product you want to see the information about can be selected, there is no need to go into a different report, it can be done in one place. I have created all these parameters throughout the other reports as well as it makes the report generic so the Administrator is able to find out more about the organisation and uncover trends and data by using combination of these.

In order to generate these reports the Month of Arrival was necessary because "Arrival" contains a datetime format but it necessary to plot the complaints by Month then Year which were both derived using the calculated derived fields in the dataset. Regarding other reports version number of products are in 3 digits, for

example 5.1.4 or 5.1.5 but we would want to group the digits in 2 digits which is also derived in the data set for all the reports as displayed in figure 6.5 and 6.6

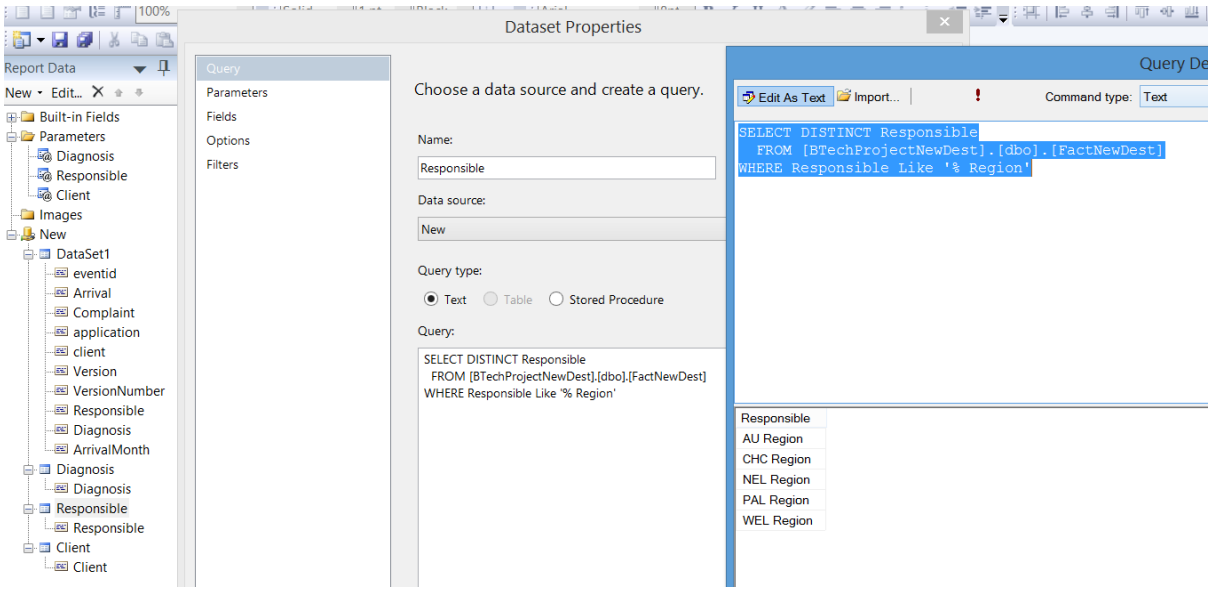


Figure 6.2 – 'Responsible' Dataset where the values will be used in the parameters for filtering.

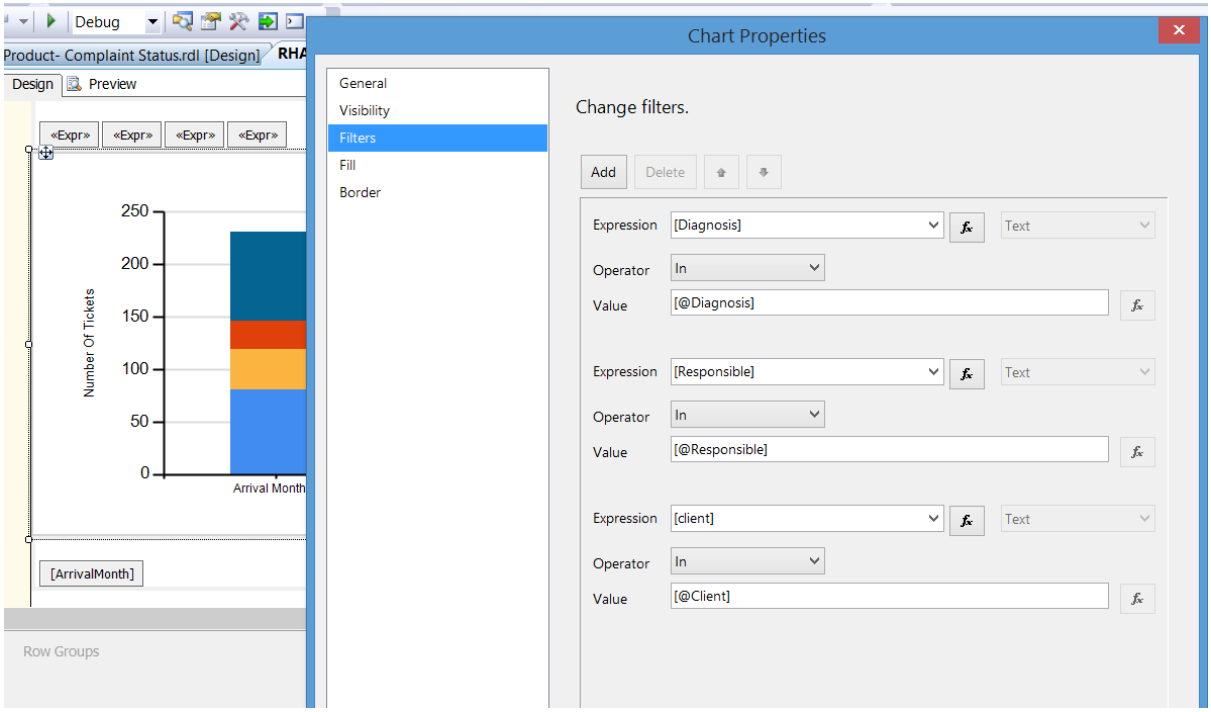


Figure 6.3 – Chart Properties to use the Parameters on the data

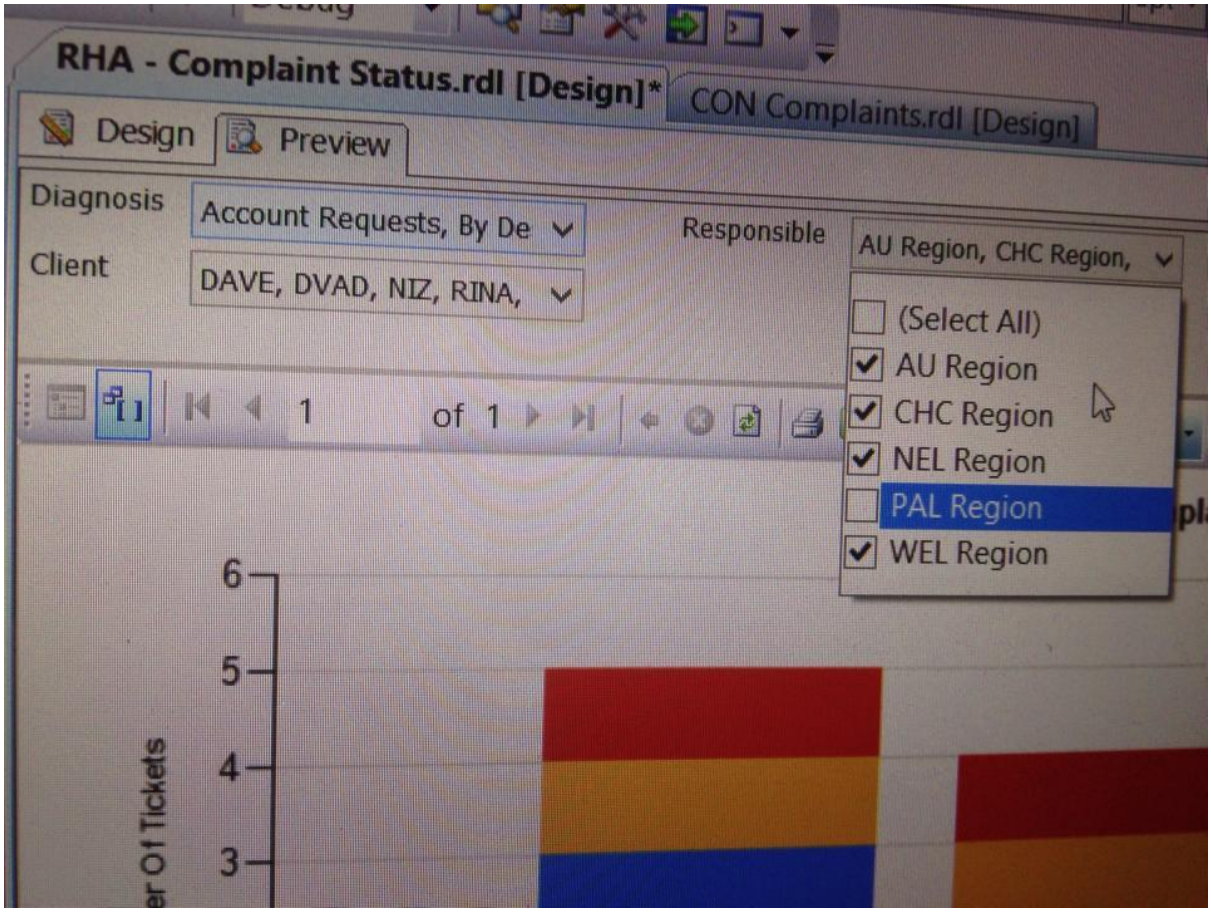


Figure 6.4 – Parameters

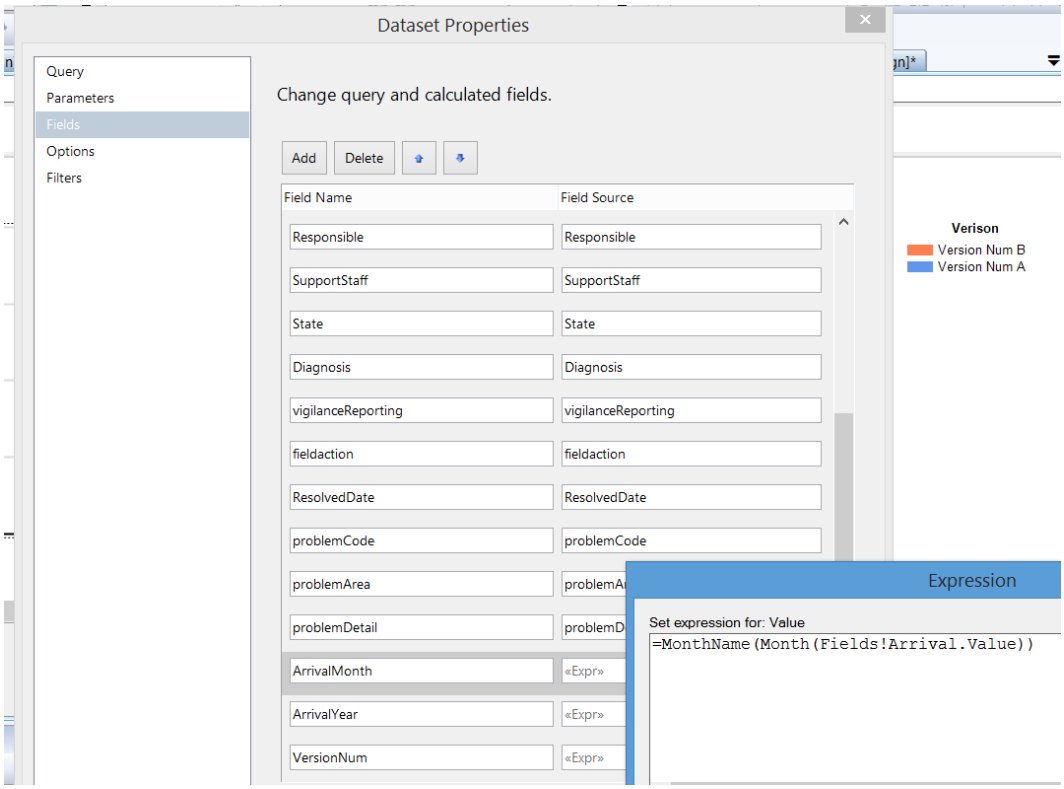


Figure 6.5

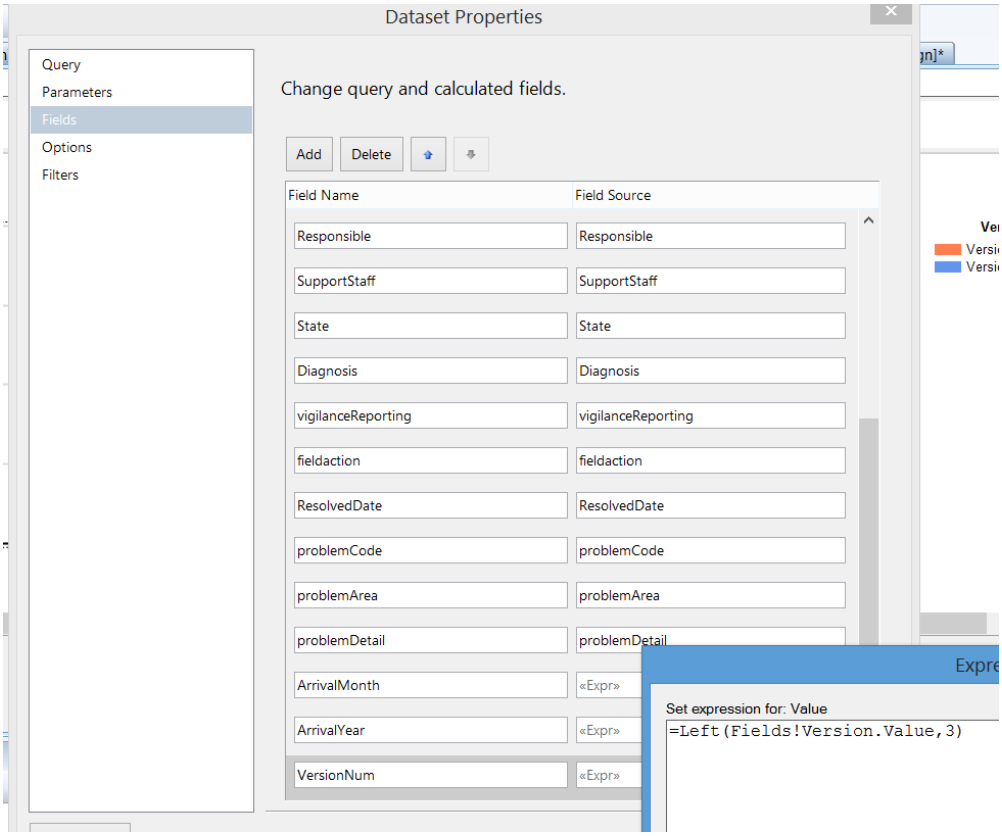


Figure 6.6

- Tickets over time (displayed as a bar chart by month) with only complaint status yes
    - Same filters as previous

A General report about the different products is used for meetings to compare total number of complaints per month to track performance and Quality by product as shown by Figure 7.

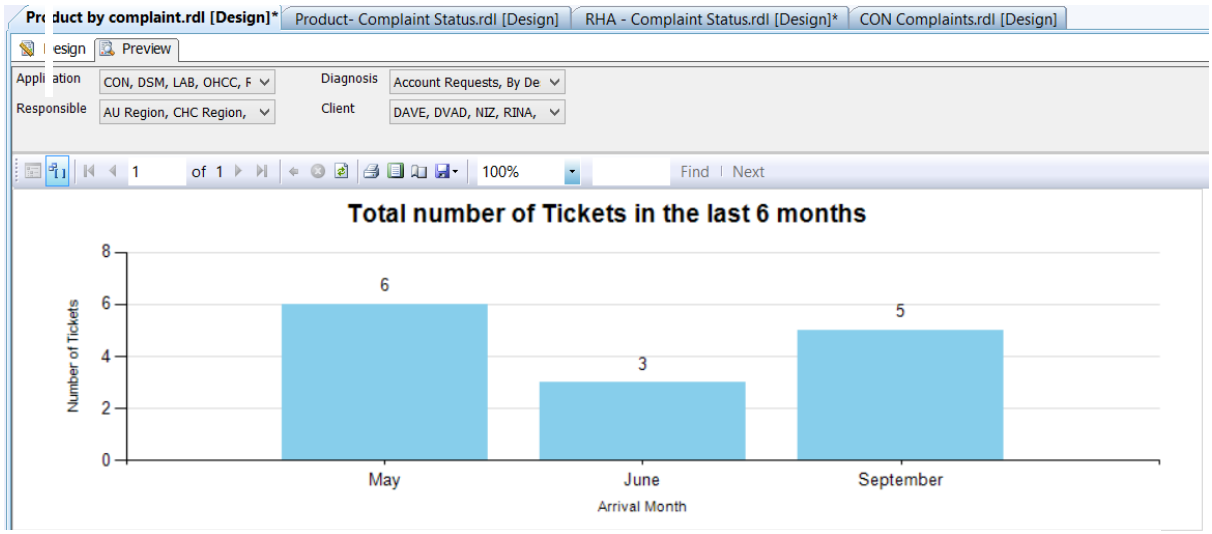


Figure 7

- Tickets over time (displayed as a bar chart by month) with only complaint status yes AND version displayed by colour
    - ➢ Same filters as previous

This report displays and goes in depth in terms of version numbers for the products therefore if the product RHA as shown in figure 8.2 was selected, this would illustrate the total number of complaints per month but also shows how many complaints each of the product versions received monthly which can be compared on a monthly basis. This is critical data Managers would love to be informed about to see how the versions of the product they are responsible for is doing in the market and which version is getting more complaints. This also identifies if the complaints for versions have decreased over the months after a version has been looked into and improved as well as which versions of the product needs to be looked into and spent time and effort on due to the high number of complaints. Product Quality is maintained by virtualizing this information and helps keep on track with Product Quality and Reliability.

On the other hand the Complaints administrator can keep up to date with the products and the versions, if they notice such patterns they are able to inform the manager for example if the Manager wishes to see the different versions a particular client has made a complaint about over the months/weeks so they are able to dig into the versions and solve the issues they were having shown by an example in figure 8.3 where Client SVISH (e.g. A hospital in Mt Eden) made a complaint about version 3.3 in June and again in September which has not been resolved therefore is critical to look further into to have satisfied customers. Many other uses of this report is possible due to being generic rather than restricted saving the Administrators time to create more reports which can be generated using these parameters using this as a template for the information required. As compared to the old system there is no need to query out this information from the support system then to transform and export the data in order to generate the reports which can be done instantly using these interactive reports.

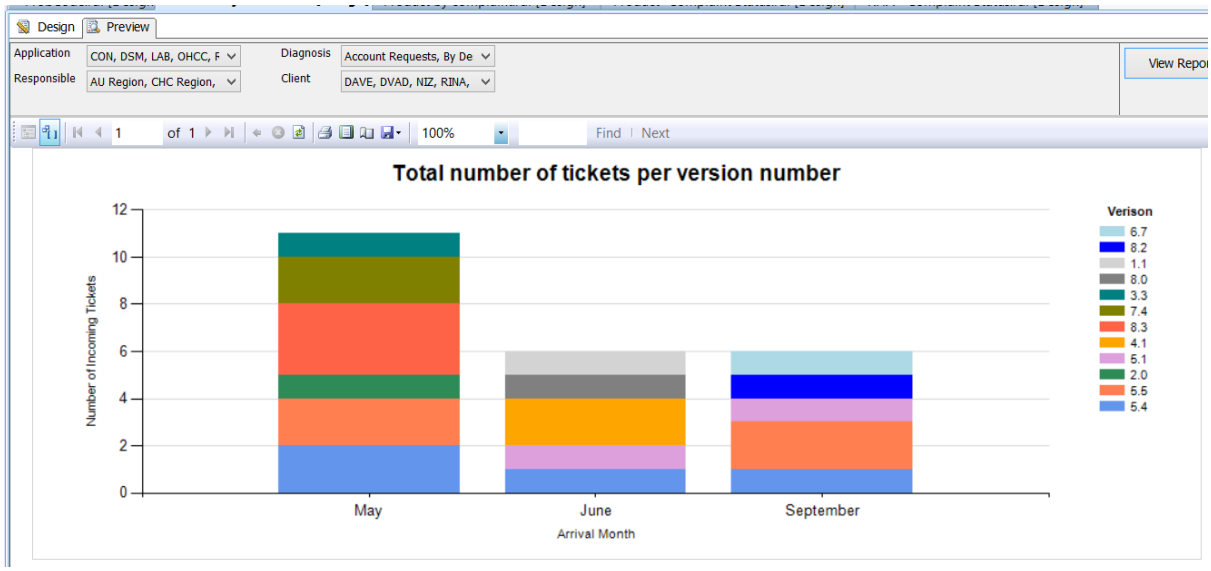


Figure 8.1 – Total volume of tickets split by Version number of products.

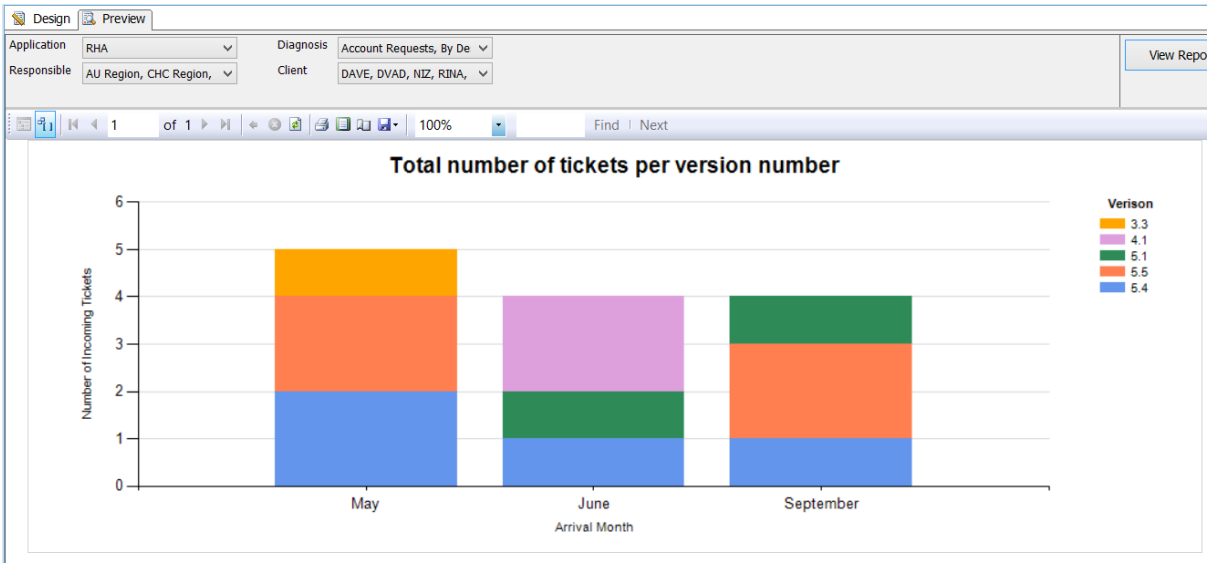


Figure 8.2

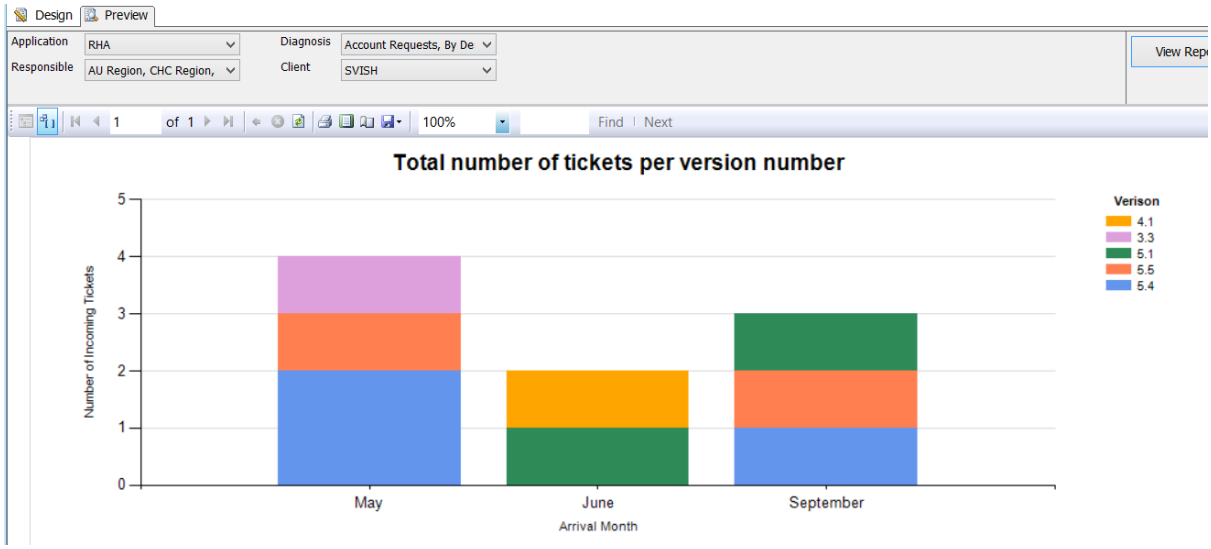


Figure 8.3

- Tickets for each version number for each Problem Code for the last 6 months: Looks over the overall complaints separated by Versions during the time categorised by problem code (Defect, Bug, Configuration – Used PC1, PC2 etc for confidential purposes)

    - Same filters as previous

  Complaint Versions Over Time: Versions are released for a products, this looks at the versions of an application (product) that the complaints are raised for each problem code such as defect, configuration, and environment so the organisation knows whether they are lacking mostly in configuration which leads to decisions like putting more resources for configuration of products or if it requires hiring more staff. Figure

9.1 shows an overall view of this, a product can be selected from the default selection of all the products in the 'Application' parameter illustrated in 9.2. This shows in a clear view of what is going on for each product. For example in figure 9.2 within the RHA product you can clearly see that Version 5.5 has an issue in the problem code 3 and 4 field (can be defect and configuration) whereas 3.3 is doing much better having an issue within the PC1 field (can be Design) but also by just looking at that chart the manager can see that the product RHA is getting higher number of complaints for the PC1 field(more issues with Design of products). This report of use and can quickly uncover areas for different products therefore this is once again generic therefore is generated depending on the selected product rather than having to go through the old system to generate these reports for the different products then update all of them on a weekly/monthly basis which will definitely take up a lot of time.

The administrator can go more in depth with this report to make reporting and analysis easier. For example, by clicking anywhere on the bar chart will take you to an even more detailed chart which is shown in figure 9.3 of Problem Area. Problem area is broken down from Problem Code such as in Defect the problem areas may be engine, hardware etc therefore the version number and product parameters are automatically taken and another report for problem area is generated based on that. For example in figure 9.1 and 9.2, I clicked on the circled area (Version 4.1) takes you to an automatically generated report which breaks down the problem code into a more detailed chart for product RHA which version 4.1 as seem in figure 9.3 which allows the administrator to see the types of areas which the version is not doing well in and needs focus on.

If this was to be created with the old system, the information of what the product and version number is noted then queries are used in the support system to filter out data based on that information as well as filtering it according to the problem code which you are interested in is considered then the data is transformed and exported. This process has to be done continuously for this report and for any others the same process has be carried out, it is difficult for the administrator to have this sort of flexibility in the old system instantaneously. Having the ability to do so leads towards improving the quality of products and better strategic decisions.

Figure 9.3 is a Pareto chart which contains both bars and a line graph. I did not have enough data in order for the cumulative line to be displayed in figure 9.3 so figure 9.3.2 illustrates an example of the Pareto chart when there is more data. Taking Figure 9.3.2 as the example, the individual values are represented in descending order by bars and the cumulative total is represented by the line. The right vertical axis is the cumulative percentage of the total number of tickets. This represents the most common sources of defect, the highest bar representing the highest occurring type of problem area is the most frequent reason for customer complaints. This shows management if they want to reduce their product complaint by 80% they must focus to on the 1st three issues. The use of this chart will support improve product quality as we can see the contributing causes to a problem which can be

targeted to improve and focus on to reduce complaints leading to improve product quality and reliability.

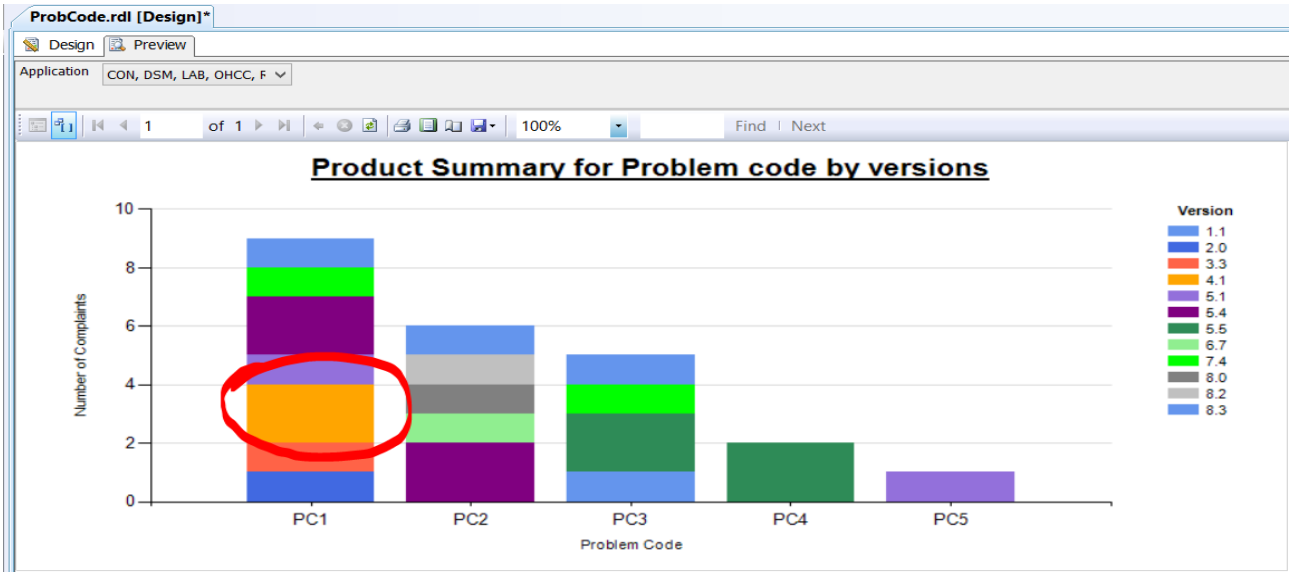


Figure 9.1

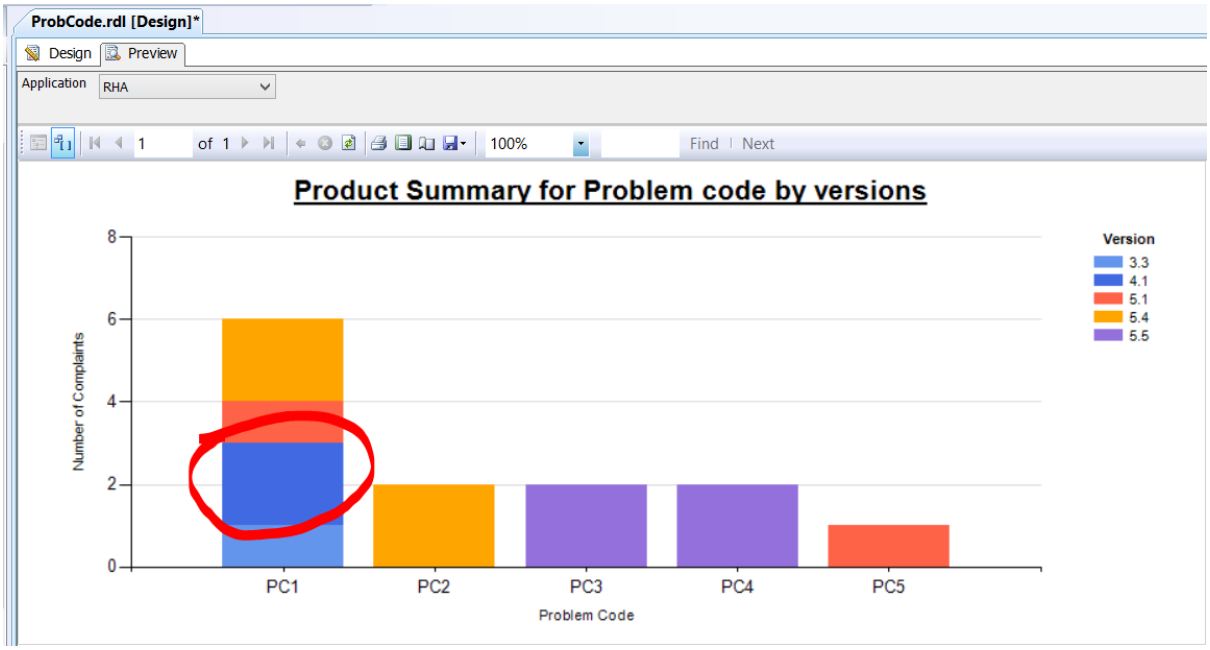


Figure 9.2

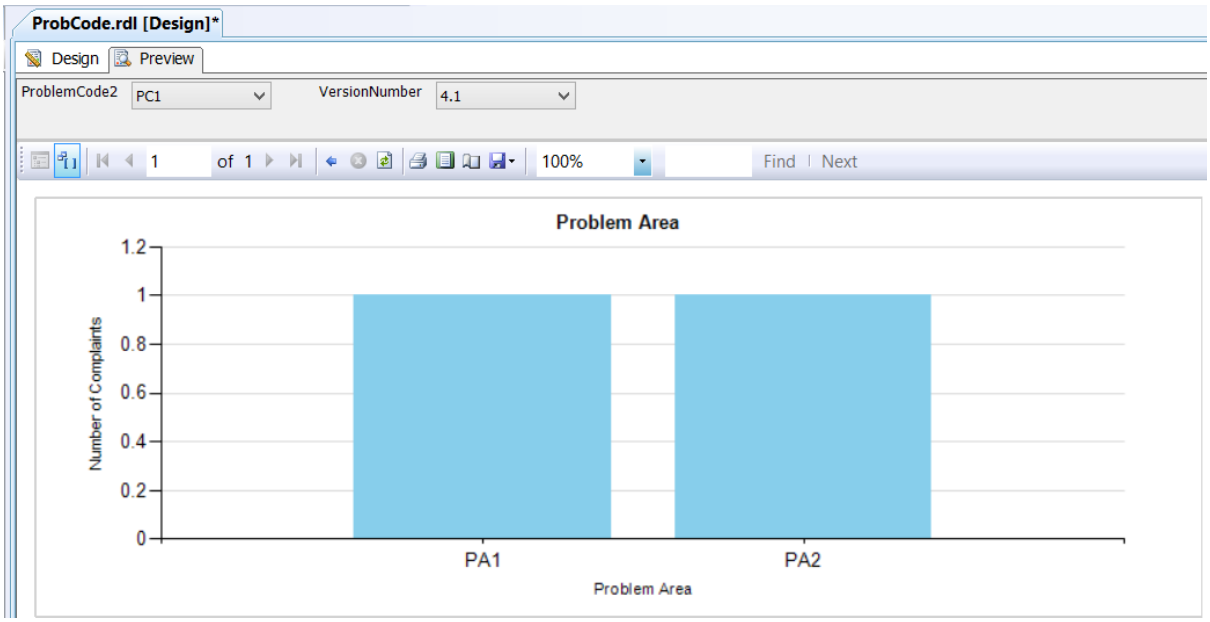


Figure 9.3

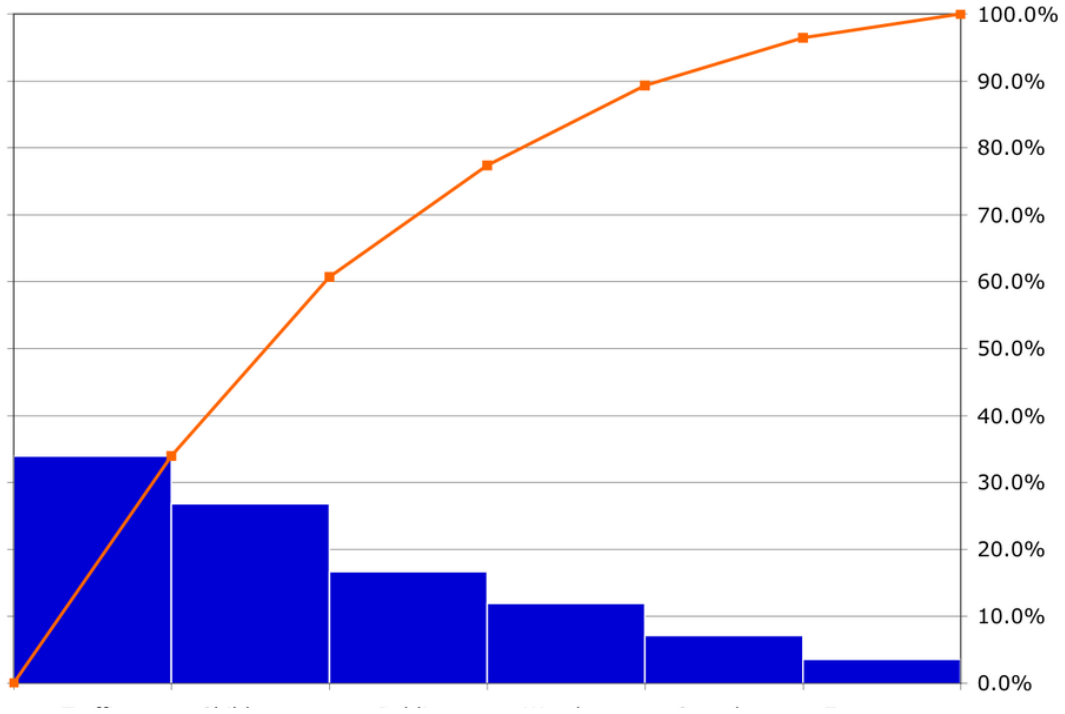


Figure 9.3.2 – Pareto chart. Left axis has also been taken out.

Note: All the columns are kept at that width so it is easier to see, where there is more data the graphs will look like this in terms of the appearance - column Width and colour scheme which was decided to be blue, green, red, purple and the shades around these colours for variety. The axis have been anonymized due to confidentiality.

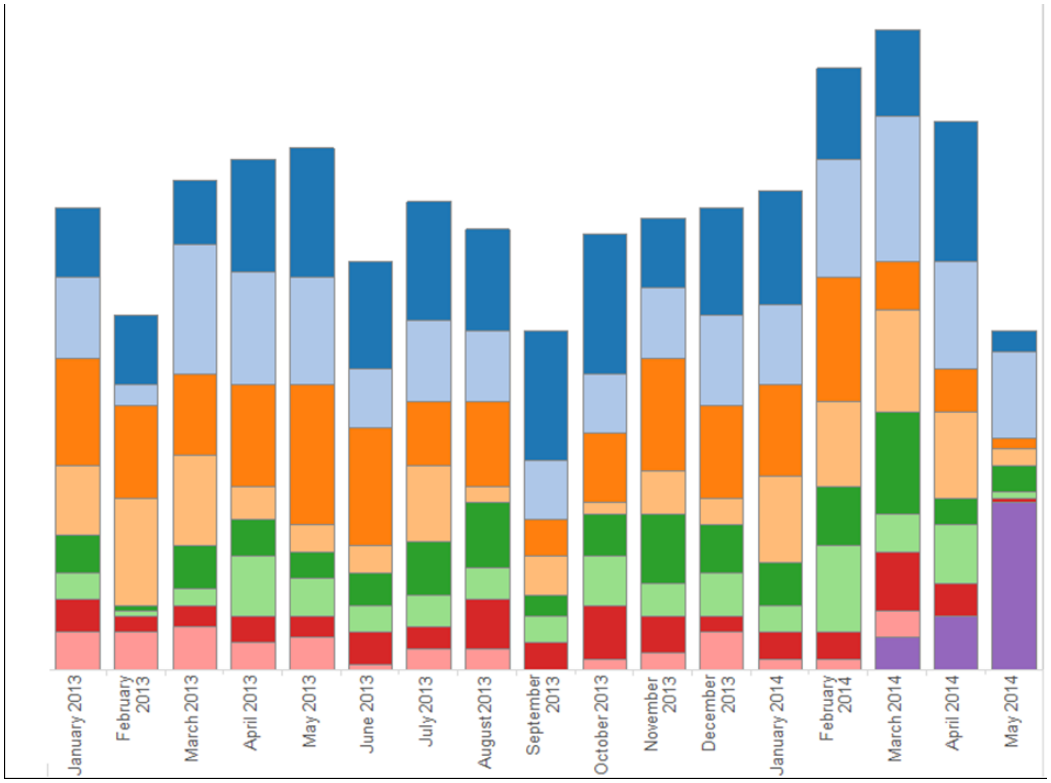


Figure 8.1 – Complaint Status over time for a product

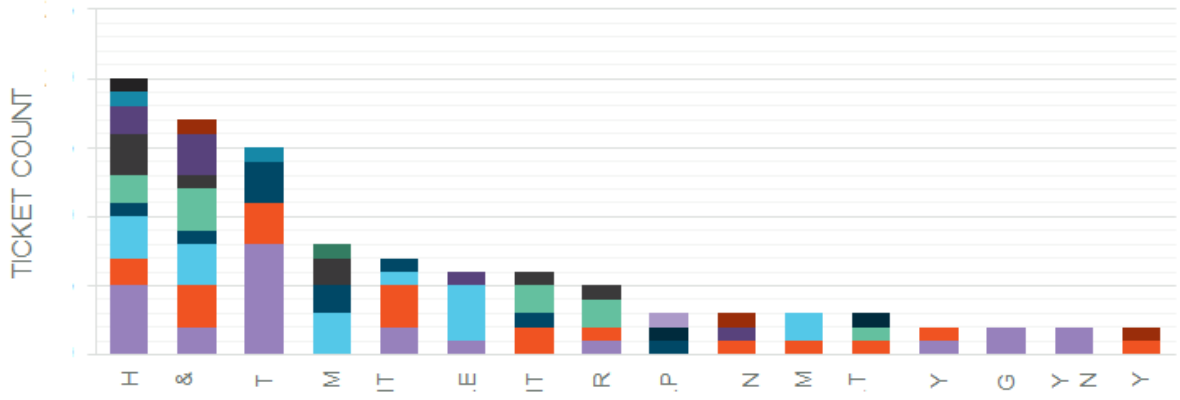


Figure 8.2- Number of Tickets of a particular product for each problem code

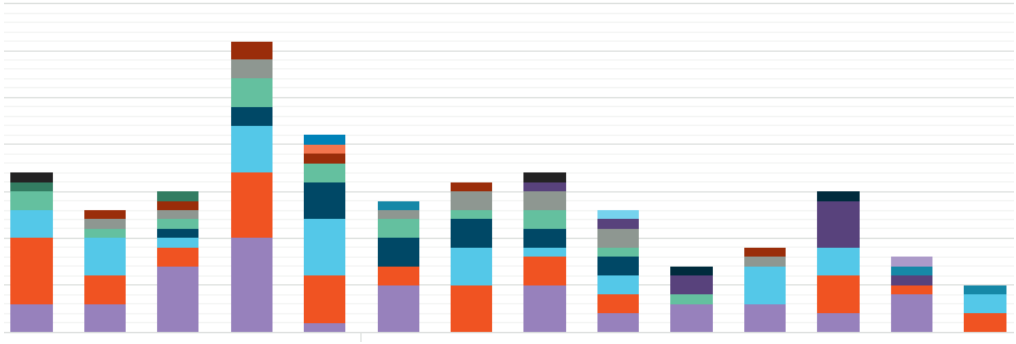


Figure 8.3 –Number of tickets for grouped by Problem Area for a particular problem code and product.

Dashboard

My last implementation as illustrated by Figure 10 is an Interactive Dashboard, an easy to read, single page, real-time user interface showing a graphical representation of the current status and historical trend of Orion Health's key performance indicators which may enable instantaneous and informed decision to be made at a glance[14]. This can also be considered as a progress report. This is also linked to the database which allows the report to be constantly updated as the data is updated. I decided to the number of complaints per product for each month during the last 3 months to see the progress and compare complaints on a product basis. Just by looking at the dashboard one can interpret if the product is decreasing in the number of complaints from the last 3 months or vice versa. By clicking on any part of the pie chart in the top row will take in the product you clicked on as a parameter and generate a corresponding report of the complaints every month for the product differentiated by the version number as seem before. For example if I click on the RHA part of the pie chart I will get a detailed chart of RHA as seen previously in Figure 8.2.

In the second row of Figure 10 I chose to display two of Orion Health's main products which I was told to generate the reports on before making them generic and flexible. These two pie charts are also interactive. You are able to click in any of the two pie charts where another report will be generated taking in the product (RHA or CON depending on which chart you clicked on) and giving a more detailed view of the problem code by versions shown in figure 9.2. Recalling from earlier on this report is also interactive and can drill down into the problem areas of the problem code and version you click on therefore a high level of detail is available to look at as well as simple in the dashboard which gives a simple over view of what problem area is getting more complaints over all for the RHA and CON product in the last month.

The Administrator like to see who is responsible for what therefore I created a basic overview in the bottom left hand table in terms of product so get an idea of which team is handling which product most at the time. When clicked on the graph, this shows the overall history of the product as shown in figure 7 but the product is used as the parameter in that report.

Lastly it shows the load on each of the teams in that month to show productivity so even when management views this they can assign more work to the teams having less workload and reduce the workload on teams or even hire more staff depending on the load.

Using this Dashboard, the Complaints Administrator may see a broader view of the organisation then drill down into detailed data. This level of detail is buried deep within the organisation or either unavailable to the management and senior executives. For this to be a successful Dashboard it depends on the metrics which are selected to be monitored and as discussed with the Complaints Administrator these are the Key Performance Indicators. Factors within this dashboard and also the other reports allow managers to monitor the contributions of the various departments.

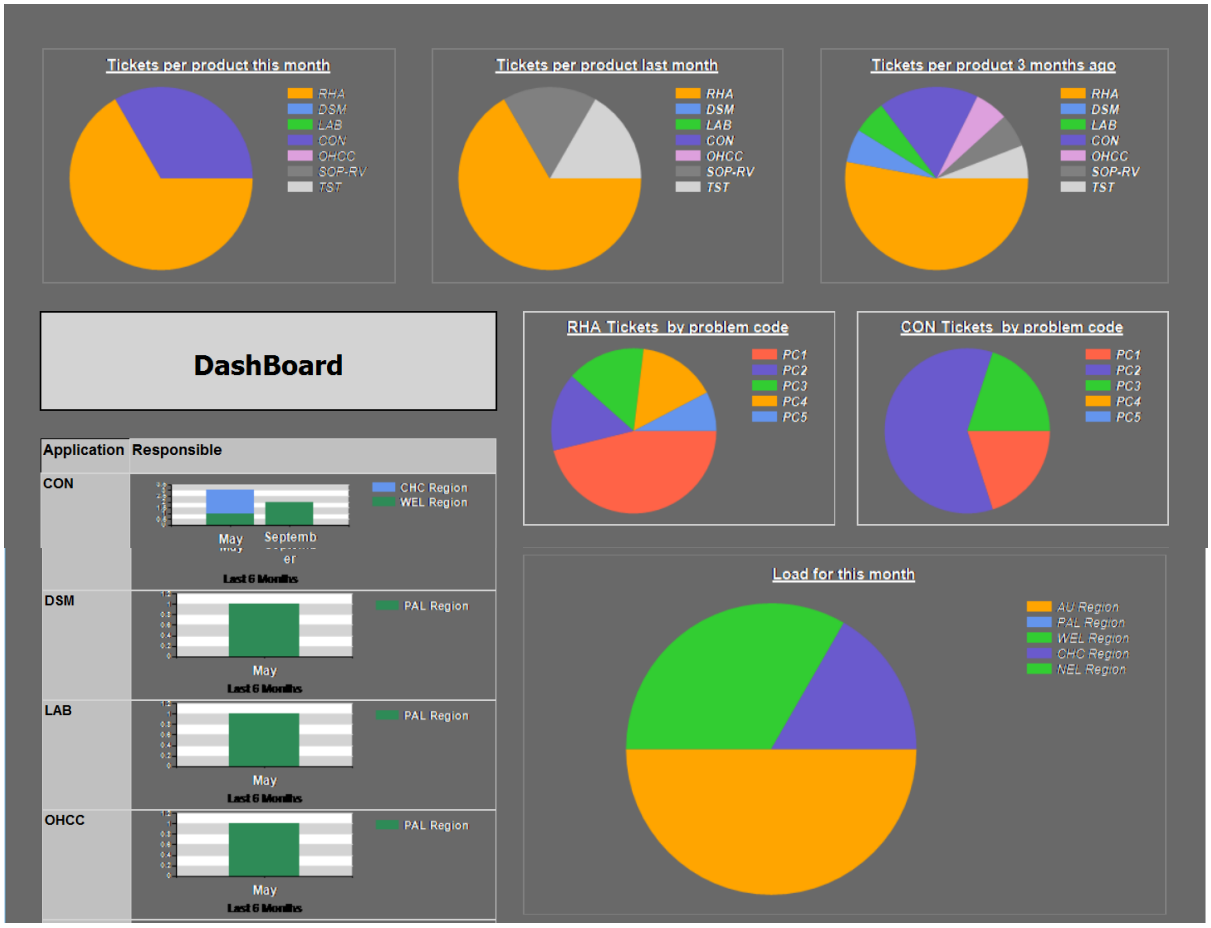


Figure 10

CONCLUSION

Reports are generated on a weekly/monthly basis to take into meetings where complaints are discussed with managers. For these the data is firstly updated manually then queried and transformed to import into a CSV file and finally into Excel or Tableau to create graphs and tables. These graphs are then placed in word for analysis, meetings and to be passed on to managers on request. This process is very time consuming as it is and also when this cycle has to be carried out by the Quality Consultant and Complaint Administrator on a regular basis. A lot of reports are going to be generated related to different aspects. The final reports are used to:

- Display trends over time, details and arising issues at Complaint meetings with product management
- Generate Metrics for Management
- Compare Development priorities with the main complaints in support and provide a feedback loop to ensure the right issues are being tackled by development.
- Give feedback to regional and global support teams on improvement
- A wide variety of other analytics

The Quality Consultant and Complaint Administrator get held up carrying out these steps when this can be made more quicker and efficient so resources. By avoiding these unnecessary processes can give the Quality Consultant and Complaint Administrator to proceed with other important tasks that have to be completed to improve products and services. Orion Health is a huge company so there are many tasks to be done, this inefficient process can be dealt with so many software platforms which are available.

Having a fast process for data retrieval and reporting will help Orion Health in decrease defects and improving their products. The generation of these types of analytics, everybody is able to look at the same data in the same way and they can always drill down and see what they want to look at, decision-making happens much faster [6]. Product managers are able to ask for more detailed reporting based on the product they are responsible therefore having the data structured and the important data together in one place (Fact Table) makes data retrieval easier and faster. Even just by interpreting the data itself may show significant changes during a time period for each of the products, components or services that Orion Health provides. Using data visualizations for reporting will help uncover deeper factors and shed light on the service the product has been providing over a time frame e.g. if the problem is in fact something other than what the product manager has interpreted to be but complaints are still arising due to the fact that the real concern is another factor. Time and effort may have been wasted on improving aspects which do not affect the product.

With the dramatic expansion of information technology, and the desire for increased competitiveness in corporations, there has been an increase in the use of computing power to produce unified reports which join different views of the enterprise in one place [15]. Dashboards have become increasingly more sophisticated thanks to new and more

advanced information technology and increased availability of data [16]. The Dashboard will provide the ability to measure and track company performance as well as pinpoint areas where the company is doing well or not for example product wise, this is only due to up to date data and automated reporting used.

The Dashboard focuses attention on the discussed metrics that affect Orion Health's success. The complaint Administrator does not have to flick through pages and pages of unnecessary data which is not meaningful. Drill downs have been used in order to focus attention in specific areas.

The automated data retrieval, data cleansing gives the ability to use great tool for automated reporting which gives the ability to identify and correct negative trends, measure efficiency and inefficiencies, ability to generate detailed reports showing new trends and having flexibility using parameters, saves time compared to querying out data then exporting it to create other reports as in the current system. A dashboard allows sharing of key information between different departments in the company to ensure that they are all aligned toward the company's core value proposition [14]. Having automated data retrieval and automated reporting will help monitor product performance leading towards improving product quality and reliability.

Bibliography

1. Relational Database Model. (2014). [online] Available at: http://myweb.lmu.edu/dondi/share/db/relational1.pdf [Accessed 14 Aug. 2014].
2. Oriented Object Modelling. (2014). [online] Available at: http://coronet.iicm.edu/is/scripts/lesson04.pdf [Accessed 14 Aug. 2014].
3. BI Public. (2014). [online] Available at: http://www.oracle.com/technetwork/middleware/bi-publisher/overview/bipublisherdatasheet-129370.pdf?ssSourceSiteId=ocomen [Accessed 14 Aug. 2014].
4. BI Enterprise Edition. (2014). [online] Available at: http://www.oracle.com/us/bi-enterprise-edition-plus-ds-078848.pdf [Accessed 14 Aug. 2014].
5. Data Visualization. (2014). [online] Available at: http://www.tableausoftware.com/solutions/data-visualization [Accessed 14 Aug. 2014].
6. Tableau. (2014). [online] Available at: http://www.tableausoftware.com/learn/stories/seagate-innovates-putting-data-hands-business-users [Accessed 14 Aug. 2014].
7. Tableau. (2014). [online] Available at: http://www.tableausoftware.com/learn/stories/seagate-innovates-putting-data-hands-business-users [Accessed 14 Aug. 2014].
8. Tableau VS Microsoft. (2014). [online] Available at: http://viz-analyis.blogspot.co.nz/2011/10/tableau-vs-microsoft-bi.html [Accessed 14 Aug. 2014].
9. Oracle. (2014). [online] Available at: http://www.oracle.com/us/bi-enterprise-edition-plus-ds-078848.pdf [Accessed 14 Aug. 2014].
   Wisniewski, M. F., Kieszkowski, P., Zagorski, B. M., Trick, W. E., Sommers, M., Weinstein, R. A., & Chicago Antimicrobial Resistance Project. (2003). Development of a clinical data warehouse for hospital infection control. *Journal of the American Medical Informatics Association*, *10*(5), 454-462.
10. Curt Hall, “Data Warehousing for Business Intelligence”, March 1999, http://www.cutter.com/itreports/RP68E.pdf
11. Gangadharan, G. R., & Swami, S. N. (2004, June). Business intelligence systems: design and implementation strategies. In *Information Technology Interfaces, 2004. 26th International Conference on* (pp. 139-144). IEEE.
12. Informatica Corporation, (2001). Method and *architecture* for automated optimization of ETL throughput in data warehousing applications. 6208990.
13. Zeng, L., Xu, L., Shi, Z., Wang, M., & Wu, W. (2006, October). Techniques, Process, and Enterprise Solutions of Business Intelligence. In *SMC* (pp. 4722-4726).
14. Clark, B. H., Abela, A. V., & Ambler, T. (2006). Behind the wheel.
15. Marketing Management, 15, 19-23.
16. Wikipedia, (2014). *Business reporting*. [online] Available at: http://en.wikipedia.org/wiki/Business_reporting [Accessed 29 Oct. 2014].

17. Anon, (2014). [online] Available at: http://www.rkm-research.com/downloads/Why-Dashboard.pdf [Accessed 29 Oct. 2014].