



T.C

ISTANBUL ALTINBAS UNIVERSITY

Department Information Technology, Master Degree

**Using Agile Methodologies to Develop Self-Service Business Intelligence
Solution**

Khadija I Mohamed Almujrab

Assoc.Prof. Dr.Oguz Bayat

Oguzbayat@altinbas.edu.tr

Istanbul-2018

Using Agile Methodologies to Develop Self-Service Business Intelligence Solution

By

Khadija I Mohamed Almujrab

Master Degree, Information Technology

Submitted to the Graduate Faculty of
Altınbaş Universities in partial fulfillment
of the requirements for the degree of
Master of Information Technology

ALTINBAŞ UNIVERSITY

Istanbul-2018

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Oguz Bayat

Supervisor

Examining Committee Members

Assoc. Prof. Dr. Oguz Bayat	Electrical and Electronics Engineering , Altınbaş University	_____
Asst. Prof. Dr. Dogu Cagdas ATILLA	Electrical and Computer Engineering, Altınbaş University	_____
Asst. Prof. Dr .Adil Deniz Dunu	Physical Education and Sports Teaching , Marmara University	_____

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Oguz Ata

Head of Department

Assoc. Prof. Dr. Oguz Bayat

Director

Approval Date of Graduate School of
Information Technology : ____/____/____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Khadija Almuhrab

[Signature]

ACKNOWLEDGEMENTS

I would like to express my gratitude to Assoc. Prof. Dr. Oguz Bayat for the invaluable comments on the thesis.

My thanks to all my teachers for their guidance, times, and their invaluable comments on my project.

I would like to express my deepest appreciation to my Family, my brothers, my sisters, my husband and all my friends for their unconditional support.



ABSTRACT

Using Agile methodologies to Develop Self-Service Business Intelligence Solution

Khadija I Mohamed Almujrab

M.S, Information Technology, Altınbaş Üniversitesi-Turkey

Supervisor: Assoc.Prof. Dr.Oguz Bayat

Date: November, 2018

Pages: 73

The market is changed in very dramatically manner, and the decision makers need a more of abstract views about their business, a new model of Business Intelligence appeared to fill their limitations is called Self-Service BI, but it didn't solve the problem; this study is searching about choosing the most appropriate Agile development methodology and adapt it to build the Self-Service BI solution, a wholesale company as a case study has been tested to answer this hypothesis

Keywords: Business Intelligence, Self-Service BI, Agile, Self-service, Agile Methods

TABLE OF CONTENTS

	<u>Pages</u>
LIST OF FIGURES	x
LIST OF ABBREVIATIONS	xii
1. INTRODUCTION	1
1.1 PROBLEM DEFINITION.....	2
1.2 OUR WORK:	3
1.3 RELATED WORK.....	5
1.4 WHAT IS BUSINESS INTELLIGENCE (BI)?.....	7
1.4.1 Business Intelligence Solution Types:.....	7
1.4.2 Self-Service BI definition.....	8
1.4.3 QlikView:	10
2. SYSTEM DEVELOPMENT METHODS	11
2.1 INTRODUCTION:.....	11
2.2 SYSTEM DEVELOPMENT METHODS:	11
2.3 WHAT IS AGILE METHODOLOGY?.....	13
2.4 TRADITIONAL DEVELOPMENT METHODOLOGIES LIMITATIONS:	15
2.5 AGILE IN BI:.....	17
2.6 AGILE CHARACTERISTICS IN BI:	18
2.7 AGILE METHODOLOGY IN THE SELF SERVICE BI:.....	19
2.8 AGILE METHODS:.....	20
2.8.1 Scrum:.....	20
2.8.2 Extreme Programming XP:	21
2.8.3 Crystal method:.....	21
2.8.4 Feature-Driven Development (FDD):	22
2.8.5 Dynamic Systems Development (DSDM):	22
2.8.6 Adaptive Software Development (ASD):.....	23

2.9	WHAT AGILE METHOD TO USE?	23
2.10	ASD PHASES:	25
2.10.1	Speculation:	25
2.10.1.1	Business Understanding:.....	25
2.10.1.2	Data Understanding:	26
2.10.1.3	Objectives/Hypotheses Setting:	26
2.10.1.4	Data Preparation/ETL:	26
2.10.2	Collaboration:	27
2.10.2.1	Solutions Development Process:	27
2.10.3	Learning:.....	27
2.10.3.1	Evaluation:	27
2.10.3.2	Deployment.....	28
2.11	AGILE BI CHALLENGES:.....	28
3.	THE APPLICATION PART:	30
3.1	FIRST PHASE: SPECULATION:.....	31
3.1.1	Business Understanding:	31
3.1.1.1	Case study Statement.....	31
3.1.1.2	Company Workflow	31
3.1.1.3	The business Problem.....	33
3.1.1.4	Deliver the Knowledge.....	33
3.1.2	Data Understanding:.....	34
3.1.2.1	Software Used:.....	34
3.1.2.2	Business Process, Data Sources:	35
3.1.3	Hypotheses/Objectives Setting:	38
3.1.4	Data Preparation/ ETL (Extract, Transfer, Loading):	38
3.1.4.1	Comparing the Data Model Building Methods:	38
3.1.4.2	Build the Data Models:	40

3.2	SECOND PHASE: COLLABORATION:	49
3.3	THIRD PHASE LEARNING:.....	54
3.3.1	Evaluation:.....	54
3.3.2	Deployment:	54
4.	COMPARISON WITH OTHER WORKS:	55
4.1	TIME TO GET VALUE:.....	55
4.2	PERFORMANCE IN THE IMPLEMENTATION:.....	55
5.	CINCLUSION:.....	57
	REFERENCES.....	58

LIST OF TABLES

	<u>Pages</u>
Table 1.1: Diminsions and its enablers	6
Table 3.1: Comparing the data model building methods [28]	39



LIST OF FIGURES

	<u>Pages</u>
Figure 1.1: BI vendors [19].....	2
Figure 1.2: Case study hierarchy	4
Figure 1.3: Sales analysis application example	10
Figure 2.1 Agile methodology steps. [8]	15
Figure 3.1: ASD-BI methodology [11].....	30
Figure 3.2: ASD-Self Service BI methodology	30
Figure 3.3: "St_Hist" page in edit script window in QlikView	42
Figure 3.4: "St_det" page in edit script window in QlikView	42
Figure 3.5: "SalesMan" page in edit script window in QlikView.....	43
Figure 3.6: "Stock" page in edit script window in QlikView	43
Figure 3.7: "Units" page in edit script window in QlikView	44
Figure 3.8: "Clients" page in edit script window in QlikView	45
Figure 3.9: "LTYD Sales 2011" page in edit script window in QlikView	45
Figure 3.10: "Calendar" page in edit script window in QlikView	46
Figure 3.11: Sales data model.....	47
Figure 3.12: Purchase data model.....	47

Figure 3.13: Accounting data model.....	48
Figure 3.14: Inventory data model.....	48
Figure 3.15: Metadata sheet in the sales analysis application	50
Figure 3.16: Goods sheet in the sales analysis application.....	51
Figure 3.17: Goods sales table building.....	52
Figure 3.18: Clients sheet in the sales analysis application.....	53
Figure 3.19: AVG sales per customer chart building	53
Figure 3.20: Time sheet in the sales analysis application.....	54

LIST OF ABBREVIATIONS

BI	:	Business Intelligence
Self-Service BI	:	Self-Services Business intelligence
XP	:	Extreme Programming
FDD	:	Feature-Driven Development
DSDM	:	Dynamic Systems Development
ASD	:	Adaptive Software Development
ASD-BI	:	Adaptive Software Development for Business Intelligence



1. INTRODUCTION

In the current changed business environment, collecting the data and analyze it quickly, give the organizations their competitive advantage, increase their market share or give organizations chance to avoiding expected losses (for example analyzing the sentiment for the social media users help to predict the feelings about specific product or service).

More people collecting and searching for data (like the government and non-government data), but the winner is who will be able to come with useful conclusions quickly (where combining many data sourced together help the company to understand the changing in the market). Business Intelligence gives you these conclusions.

Business Intelligence is defined as "applications and technologies for, gathering, storing, analyzing and providing access to data to help make better business decisions. The managerial view of Business Intelligence (BI) is getting the right information to the right people at the right time so they can make decisions that ultimately improve enterprise performance." [2]

But BI don't fulfill their promise to provide the data to the decision maker as we mentioned above, also there are mostly misunderstanding between the IT department and the end users, whereas the IT department may not fully understand how the data should look when an end user requests it or what that user is trying to get out of the data; and the It department don't satisfy all the changed information needs for the departments managers and the knowledge workers.

Forrester in their 2012 report about evaluation of Self-Service Business Intelligence (BI) vendors, [3] wrote "We maintain that in an ideal BI environment, 80% of all BI requirements should be carried out by the business users themselves". The solution as Evelson wrote is: "let the users make their BI applications, so nothing is lost in translation and that the core of the Self-Service BI".

But using just the self-service BI didn't solve the problem, where according to Gartner [1], building the Self-Service BI solutions by using traditional methodologies (like waterfall methodology) is one of the reasons that projects fails.

We suppose that building the Self-Service BI solutions by using one of the agile method will help to solve the problem of Self-Service BI projects failure.

1.1 PROBLEM DEFINITION

Gartner bifurcates the Self-Service BI platforms (or Gartner called data discovery tools) market into interactive visualization tools and search-based data discovery tools. [1]

As seen in the following Gartner reports for the BI providers: [19]



Figure 1.1: BI vendors [19]

In fact choosing the right Self-Service BI platform doesn't mean we get a successful Self-Service BI solution, because it solves part of the changed requirements problem and not all, whereas the business user may has a question in his mind or he needs a new chart to understand specific problem, the Self-Service BI platform enables him to do that if these requirements included in the data model or enabled by the solution expressions, but for example what if the business users need to connect to a new data source to get a new feed or want to do a complex analysis with

special formula, such these requirements appear after the user gets the BI solution ready to use, and adapting the BI solution to fit such these requirements after the deployment is costly and takes time; because We need to re-build the solution from the start to adapt the changes, test, and deploy it; and sometimes the organization don't adapt these changes, and over the time and with increasing the number of such requirements that lead to disregard the BI solution.

Traditional development methodologies have a lot of limitations when we want to build Self-Service BI projects.

For example the waterfall methodology is a software development life cycle, in this methodology we collect the user's requirements, and then build the design which fit the requirements, turning to developers, then to the testing phase. [4]

The limitation of this method that it doesn't enable the users to participate in the development process, it just give them the final solution, and any new requirements or changes appear will cost a lot of money and time, which make the methodology not appropriate for the BI projects.[4]

Our view to solving the above problem that we need a method that enable the decision makers to share in the developing processes for the Self-Service BI solutions which later will support them in taking the decision. And we suppose that The Agile development methodology fill the above mentioned limitations of the traditional development methodologies where it lets the user to contribute in project building.

But there are many methods types in the Agile, and the framework of each one different from the others, and not all of them useful to build Self-Service BI solutions.

our approach to solve the problem is comparing the existing agile development methodologies, and discuss their ability to use in building successful Self-Service BI solutions (where according to Gartner 70 to 80 percent of Business Intelligence projects fail for several reasons [1]) . And test it by building a Self-Service BI solution for a real life case study to see if it solved the above problem or not.

1.2 OUR WORK:

“A business analyst can draw conclusions in partnership with IT much faster using Power BI in conjunction with the agile development methodology than with the old technique of traditional IT.” [7] Jeff Melton, Senior IT Solution Manager, Microsoft

This study will be combined from two parts:

Theoretical part: a comparison study which discuss the developing methodologies, the Agile methodologies, using the Agile methods in developing Self-Service BI solutions , which one of the methods is the most appropriate to develop Self-Service BI solution, The ability to use one of the Agile methods to develop Self-Service BI theoretically

Practical part: discuss the ability to develop a successful Self-Service BI solution by using the above chosen method, by solving a real data analysis problem for a real life case study.

Ken collier suggested the following solution:" it is better to fail fast and adapt than to fail late after the budget is spent". [10]

As we saw in the figure (1) the power of the Agile methodology come from the iterative, and decomposing the work, where the whole solution dividing to small pieces and enabling the users to share in solution building, Where after taking the requirements from the users and build the pieces of the solution, that encourages him to add a new requirements according to the appeared or expected needs which facing him; that's all happened through the building phase and before the deployment; which represent the iterative in the Agile methods.

Where the iterative is: developing software through multiple repetitions and that done for each software release or subsystem. [11]

The Agile methods are different due many specifications like the number of iterations, number of meetings and other specifications. [11]

In the normal cases the Agile methods are using in developing the softwares, but this study will benefit from it to build the Self-Service BI solutions, we will use it in a case study for a mid-size company which have the following hierarchy, we will work on a wholesale company because it has hierarchy similar to the following, it doesn't have a lot of complex hierarchy which hard to understand like the manufacturing companies. And it uses a relational database SQL.



Figure 1.2: Case study hierarchy

Where we can divide the whole Self-Service BI solution according to the departments or data sources, and applying the chosen method from the theoretical part in developing the application for each department.

We will use to develop the solution tool called a QlikView where it is one of the leaders in the Self-Service BI market according to a survey done by Gartner in 2018 and it's recommended in many studies. [9]

1.3 RELATED WORK

Using the Agile methodology in the BI projects discussed in many literature, where Ken Collier [22] wrote "data community wasn't paying attention to the benefits of agility, while the software community wasn't paying attention to the unique challenges of database development and systems integration". Where he shed the light on using the Agile methodologies in the traditional BI projects.

But in some cases it didn't contributed in Success the BI projects where many Agile BI projects are failed, the reason according to Moss et. Al, the Agile didn't apply in the correct form, and they conditioned that we should use agile methodologies with rigor in BI projects. [21]

The Self-Service BI is different from the traditional BI from many sides, and has many different characteristics.

in some literature there are a mixing between concepts Agile BI and using Agile as a development methodology, in article for TDWI organization, the Agile BI defined as "Agile BI enables data access when it is required through the development of an infrastructure that supports real-time, diverse analytics access, etc" the author in his article didn't mean developing the solution by Agile methods, but he meant the Agility itself in delivering the data ; and the solutions that's support that such flexible infrastructure. [23]

In 2014 Daniel Rönnow in his study Develop a theoretical conception of standards that can be followed for the enablers to governance the mechanisms for the Agile development in the Self-Service BI. The framework developed consists of four dimensions: organizational, process, technical and external, he just added the external dimension on the previous work, as seen in the following figure: [20]

Table 1.1: Diminsions and its enablers

Dimension	Enablers
Organisational enablers	Usage and support Management support Business alignment Flexibility
Process enablers	Resources User participation Team skills and interaction Iterative development
Technical enablers	User access Data quality Architecture
External enablers	Legislation

The author took a case study of a Swedish bank, he collected data by interviewing the employees of the case study; the collected data analyzed to find the enablers for BI service agilely developing Self-Service BI applications in the organization. Where he divided the groups into the four above dimensions. And He proposed structure of the Agile development program of Self-Service BI applications

According to the author “The study was conducted at a case study company about to establish such program of Self-Service BI, hence has an established program not been included in the study.” The author presented hypothetical enablers of an agile development program of Self-Service BI, Where the author didn’t explain the development in the implementation, he just put a theoretical conception for success criteria which can be followed, and the conception is assisted from studying just one organization “Swedish bank” which may makes some problems in the implementation in other organizations.

In article for Microsoft team in 2015 they advised to use the Agile methodologies to develop the Self-Service BI projects. And they said that they got a many benefits from doing that; Jeff Melton Senior IT Solution Manager Microsoft suggested to use the Agile methodologies to develop the Self-Service BI projects done by Power BI a (Microsoft Self-Service BI platform to build the projects) instead of using Agile methodologies with the traditional BI, and he said that doing that helping to give better understanding to the business. [7]

The article was very short and has not enough information about the used methods, or the implementation method that can be followed to access these benefits.

Data classification is an important factor in development Process and can help in making decisions by collecting the data and analysis it. [32]

To the best of our knowledge there is no previous literature discussed using the Agile methods and their different characteristics in Self-Service BI project implementation. In this study the Agile methods will be discussed in details, and the most appropriate one for the specific Self-Service BI characteristics will be used to build a real life solution.

1.4 WHAT IS BUSINESS INTELLIGENCE (BI)?

There are many definitions for BI, but the famous one is the IBM definition:

"Business Intelligence is the application of putting the right information into the hands of the right user at the right time to support the decision-making process". [12]

In general BI is any tool or process that helps the decision makers to extract knowledge from the stored data, which helps him to take the fast and good decision.

1.4.1 Business Intelligence Solution Types:

We can package the BI solutions in the following groups:

1) Standalone BI tool:

This solution type used by the small and medium organizations, it can integrate with the database on any platform (windows, Linux...), and it doesn't need a Data Warehouse or OLAP cube to work. This solution can produce: static and Ad-hoc reports. Static report is a query that returns from the database regular information without any interchange from the user.

Ad-hoc report is a report that the user can configure its variables to get a required result; such reports enable the users to explore database contents without any experience with database technical issues like the relational theory and key constraints, they focus more on providing the users with static data.

Such a solution can be built by the organization IT team, or they can buy it as whole package from solutions vendors.

2) Integrated with Data Warehouse (DW):

Ralph Kimball has defined the data warehouse as “a copy of transaction data, specifically structured for query and analysis”. [11]

This type is used in the large organizations; by using it we can make multidimensional analysis, with the ability to drill down to get detailed data and to drill up to get aggregated data. In this solution type we can make the following analysis applications:

- Reporting process: like the previous reports types.
- OLAP Cube analysis: it's a technology enables the user or the analyst to view the information from different dimensions to understand the transaction data and extract the knowledge from it.
- Data mining: The process of discovering interesting information from the hidden data that can either be used for future prediction and/or intelligently summarizing the details of the data [11] it can be used in all the life domains, and can be applied on many different data types such the medical data like in the research done by in 2018 on classification the medical data. [32] .
- Dashboards, scorecards, Key Performance Indicators (KPI's)

This solution is built by BI team professional on special BI platform. And the users of it must have a minimum knowledge about the explored data to get the maximum benefit.

3) Self-Service BI (Business Discovery):

A new age of BI (We will discuss it in the next section)

4) Advanced BI solution:

Contains many new solutions like the open source BI, BI as a service, Cloud BI and others...

The first two types represent the traditional BI, and the third solution represent the new BI solution age or what Gartner called it Data Discovery solution. [13]

1.4.2 Self-Service BI definition

In fact the changing does not occur just in the BI market, it's changing all over the technology world, the consumer behavior is changing in very important way, he depends more on himself by using the search engine and social networks to make decisions, and depends more on the mobile and its applications. [14]

TDWI (The Data Warehousing Institute) defined it as "The facilities within the BI environment that enable BI users to become more self-reliant and less dependent on the IT organization. These facilities focus on four main objectives: easier access to source data for reporting and analysis, easier and improved support for data analysis features, faster deployment options such as appliances and cloud computing, and simpler, customizable, and collaborative end-user interfaces".[15]

The gist of Self-Service: it allows business decision makers to build their own business performance reports, instead of waiting, sometimes for weeks, for IT resources to deliver reports, as has been traditional in the BI world. However, organizations that makes BI tools more readily accessible to a larger number of decision makers report faster delivery of reports and models. [16]

The following figure shows a Self-Service BI analysis application example for the sales which contains chart, and reports made by the sales manager itself after preparing the appropriate solution by the Self-Service BI developers: [17]

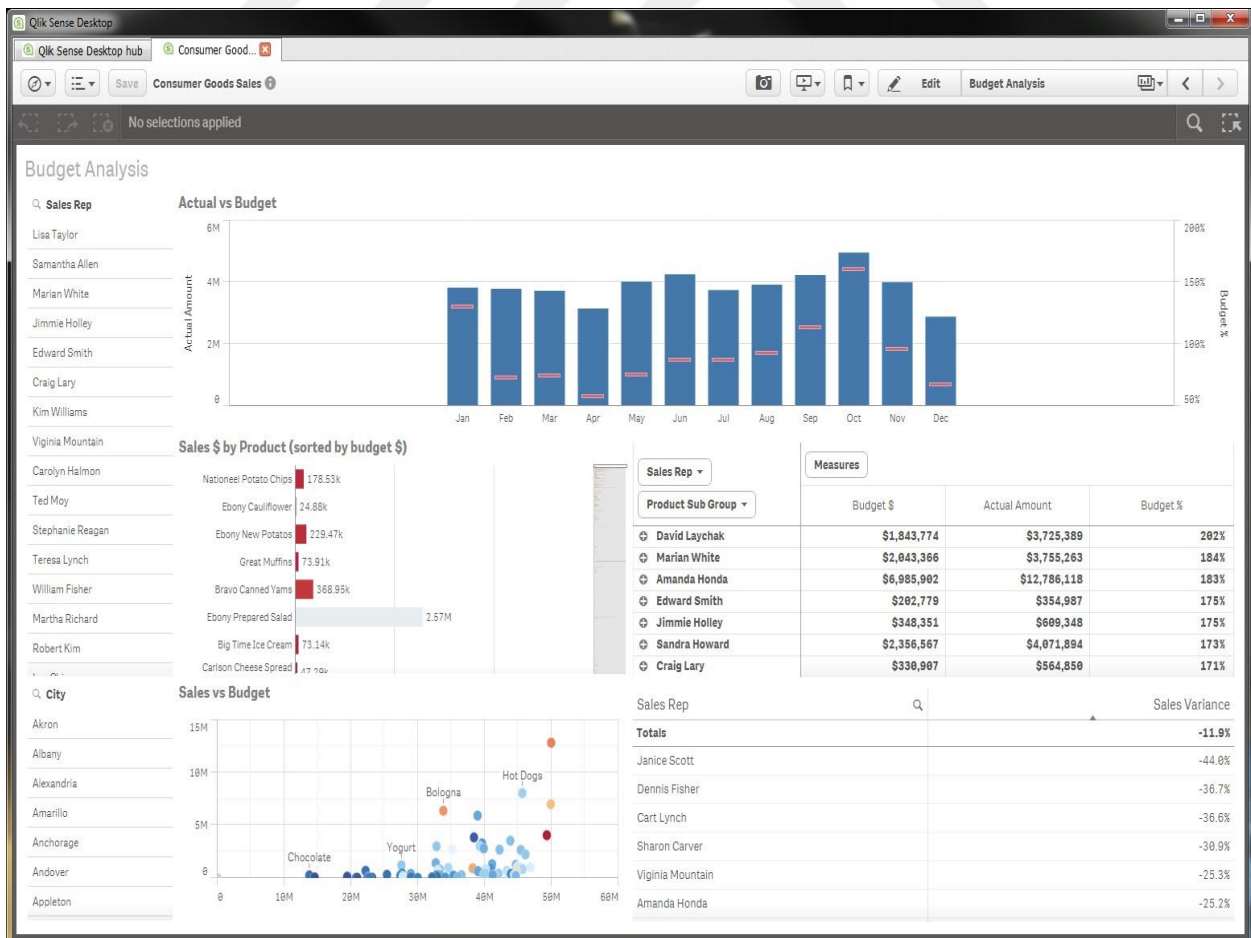


Figure 1.3: Sales analysis application example

1.4.3 QlikView:

QlikTech is the vendor name and QlikView is the name of their product, it's a Self-service BI platform used to develop the solutions; its enable the BI developer to access heterogeneous databases (for example in our case the wholesale companies use SQL databases), preprocess the data, built the data marts, load the data onto the application, develop interactive reports, OLAP cubes, maps, charts, KPIs and other analysis applications, with infinite power usability and efficiency. As seen in figure (3). Also in 2014 Gartner report QlikView ranked as better than many international BI vendors like Microsoft. As seen in figure (1).

2. SYSTEM DEVELOPMENT METHODS

2.1 INTRODUCTION:

In fact choosing the right Self Service BI platform doesn't mean we get a successful Self Service BI solution, because it solves part of the changed requirements problem and not all, whereas as mentioned before the business user may has a question in his mind or he needs a new chart to understand specific problem, the Self Service BI platform enables him to do that if these requirements included in the data model or enabled by the solution expressions, but for example what if the business users need to connect to a new data source to get a new feed or want to do a complex analysis with special formula, such these requirements appear after the user gets the BI solution ready to use, and adapting the BI solution to fit such these requirements after the deployment is costly and takes time; because We need to re-build the solution from the start to adapt the changes, test, and deploy it; and sometimes the organization don't adapt these changes, and over the time and with increasing the number of such requirements that lead to disregard the BI solution.

Ken collier suggested the following solution: "it is better to fail fast and adapt than to fail late after the budget is spent". [5] which mean we should give the user executable application that encourages him to expect the requirements which may face in the future and then adapt this application to handle these requirements; that all happened through the building phase and before the deployment. And that represent the core difference between the traditional building BI solution method and a new one.

2.2 SYSTEM DEVELOPMENT METHODS:

In general there are many methods to develop the systems; in their book Systems Analysis and Design book Gary B. Shelly, Harry J. Rosenblatt mentioned that the most popular development methods are: [24]

Structured analysis, Object-Oriented (O-O) analysis, and Agile/Adaptive methods

They define them as:

2.1. Structured Analysis method:

Represents the system in terms of data and the process that act upon that data. System development is organized into phases, with deliverables and milestones to measures progress".

We use The Software Development Lifecycle (SDLC) waterfall model consists of five phases, to plan, analyze, design, implement, and support an information system.

2.2. Object-Oriented (O-O) analysis method:

View the system in terms of objects that combine data and process. The objects represent actual people, things, transactions, and events. Compared to structured analysis, O-O phases tend to be more interactive; cannot use the waterfall model or the model that stresses greater iteration.

2.3. Agile/Adaptive methods:

Attempt to develop a system incrementally by building a series of prototypes and constantly adjusting them to user requirements as the Agile process continues, developers revise, extend and merge earlier versions into the final product. And Agile approach emphasizes continuous feedback and each incremental step is affected by what was learned in the prior steps.

In general the methods used in the BI projects are Structured Analysis and Agile/Adaptive methods.

Notice: The SDLC describes activities and functions that all systems developers perform, regardless of which approach they use. [24]

The SDLC has many models, Murray Cantor in his book Object-Oriented Project Management with UML wrote the lifecycle models are: waterfall, spiral, rapid application development (time-boxes), and controlled iteration. [25]

He wrote " All lifecycles have scoping, design, implementation, and verification activities, but they differ in how they schedule and organize the practice of these activities". [25]

The famous and the most used model is the Waterfall model: introduced by Barry Boehm in the 1970s, it was the first widely adopted lifecycle model, this model consist of five major phases:

Take the requirements and plan the project

- Design the model
- Write the code
- Deployment and integration
- Test and maintenance

In this model we cannot start any phase until finished the previous one.

Spiral model: The spiral model is an elaborate form of the waterfall model introduced by Barry Boehm in an article published in IEEE Computer in May 1988. Boehm points out that the builds maybe thought of as iterations through four high-level phases: [25]

- Determining objectives: Product definition, determination of business objects, constraints.
- Evaluation of alternatives: Risk analysis, prototyping.
- Product development: Detailed design, code, unit test, integration.
- Plan the next iterations: Customer evaluation, design planning, implementation, and customer delivery.

The spiral model develops the system in iterations, and through the iteration it follows the waterfall model, and it used mostly in the risk analysis and in the big projects.

In fact development techniques changed rapidly to fit the changed environment, we called the previews models as the traditional development methodologies and the Agile methodology is the new generation, because it comes after the structured analysis to handle the limitation of the traditional methodologies.

In our research we focused on the development mythologies for one of the software projects type which is the Business Intelligence project and in specific the Self Service BI one, and we searched to find the methodology which most fit to build such projects.

2.3 WHAT IS AGILE METHODOLOGY?

Alnoukari in his book describes the Agile methodology as "the new software development age which handle the failure reasons of the traditional one, it is coming after using the traditional methodologies for many years, to change the process-based software development methodologies that used in the past, that are characterized with heavy documentation, strong emphasis on the process, and less communication with customers". [11]

The Agile methods appeared before 2000, but In 2000, a group of the most influential application software developers convened in Salt Lake City and formed the Agile Alliance. In this meeting the developers sharing and comparing their Agile styles of software development and they put the Agile Manifesto which represents the Agile projects guide and decision making. The Agile Manifesto reads:

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value: [5]

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

In fact the Agile is more than software development methodology, it's a way of thinking, and it's a culture; personally I saw people out of the software development domain, applying it in their projects.

In general we can describe the Agile software development methodology characteristics as following: [11]

3.1 Incremental:

According to Agile method the system is divided to subsystem, then each subsystem divided according to a specific method such as functionalities or user stories, each function coded and tested to present to the end user as executable software.

3.2. Iterative:

The word iterative means developing software through multiple repetitions and that done for each software release or subsystem. [11] We work in short iterations that are generally one to three weeks long, and never more than four weeks. [5]

3.3. Self-Organizing team:

The Agile gives the team the autonomy to organize itself to best complete the work items. [6]

3.4. Collaboration:

It's the most important Agile characteristic, Agile method maximize the interaction between people, it focusses on the people and process not just on the process like the traditional methods. The collaboration done between project managers, sponsors, users, developers; the developers collaborate with the users and get their feedback after each iteration to adapt the software, the project management collaborates with the development team to orient them, and the developer collaborate between them and share their knowledge, this collaboration atmosphere is an important success factor.

3.5. Production quality:

In each software release the developers deliver to the user fully tested and accepted software, they don't deliver a prototype.

3.6. Value-driven development:

The Agile methodology by itself is a method to get our goal, our goal is to deliver a product that serve the user needs, the business users don't care about the method that I use, the code, the software architecture or the documentation, they need to solve their business problems in the best and fast way.

And that what the Agile method deliver, whereas every iteration must produce at least one new user-valued feature, also Agile methodology less focus on the high degree of ceremony and emphasizes a sufficient amount of ceremony to meet the practical needs of the project. [5]

The following figure explains the agile methodology characteristics, where each sprint represents a subsystem which developed iteratively to give a value to the user.

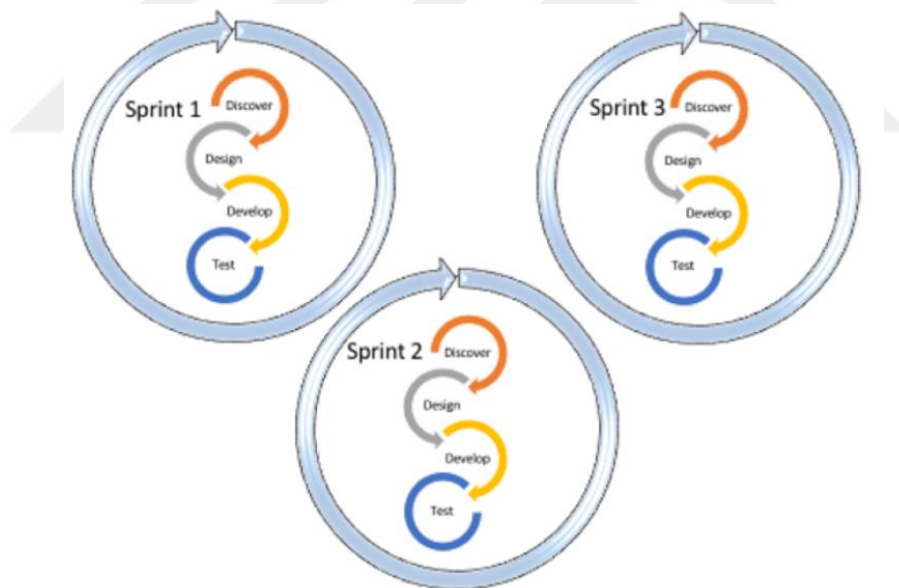


Figure 2.1: Agile methodology steps. [8]

2.4 TRADITIONAL DEVELOPMENT METHODOLOGIES LIMITATIONS:

As mentioned before the Traditional development methodologies faced many limitations when used to build the BI solution, this was emphasized by Deanne Larson in the "Business Intelligence and Agile Methodologies for Knowledge-Based Organizations" book when wrote:"

Research has demonstrated that waterfall lifecycles and traditional software development practices are not successful in BI. Software and hardware do not provide organizations value pertaining to BI; it is the use of information". [11]

But what are the limitations of waterfall lifecycles and other traditional methods which make them not successful in BI:

4.1. As we know the waterfall cycle contain five phases:

We collect the user's requirements, and then build the design which fits the requirements, turning to developers, then to the testing phase.

Each of these phases doesn't start until we finished the previous one. Here we have two problems:

Users might not be able to describe their needs until they can see examples of features and functions: [24]

At first in the BI projects, the development team get the requirements from the end users which are still fuzzy in their minds, and the end users cant specify what they want from the BI projects, so after the project has finished and they can see the user interface and BI features, the requirements triggered in their minds, so he asked the development team to adapt the BI project to fit the requirements which have been clearer in their minds.

The second problems: as we know in today environment the requirements are changed in dramatically manner, so a new requirements appear, the development team take the requirements from the end users and go to continue all the development process, and in general the waterfall model takes a long time to be finished, after finishing the users got the final solution to use it, but according to the development time a new requirements appear, and the end user want to adapt the BI solution to fit these new requirements.

That made the main problem here the high cost of adapting in the waterfall methods, especially in later phases: [24]

For example if any change appears in the test phase, we should alter design, code, documentation, integration with the finished features etc.

Of course the requirements can change in the earlier phase and that don't need a hard work as in the later phases.

4.2. Don't participate the end user in the development process:

The end user is not part of the project in the traditional development methods, he just gives the requirements and makes the acceptance test, but he doesn't participate in the building phase and other phases.

That don't enthuse the end user to use the BI solution, because he doesn't feel like he shared in the solution building, In spite of the shares in taking the solution fail responsibility.

4.3. It takes too long time:

The waterfall cycle takes too long time to deliver an executable BI solution to the end use.

Traditional development includes timelines that can extend to years before producing working software. [11]

4.4. Traditional development includes larger teams of individual specialists. [11]

4.5. Developers didn't understand the requirements:

That is a big problem face the waterfall development methodology, the developers take the requirements and build the BI solution upon it, and when they deliver the solution after all development phases the user say "that is not the solution that I want".

2.5 AGILE IN BI:

As we see we face many limitations when we use the traditional development methods to develop the BI projects, the question here can we use the new development generation (Agile methods) in the BI projects?, and does it gives successful BI project?, if we know that the Agile method created at first to develop the software applications, and the BI projects different than the software applications, ken collier describe the difference as " we are dealing with large volumes of data, our efforts involve systems integration, customization, and programming". [5].

Also Deanne Larson wrote:"the BI projects include stumbling blocks differs the BI projects from the software applications such: fuzzy requirements; lacking an understanding about how data is created and used; data quality is not measured or known; source system constraints dictate design and service levels; developing based on perceptions of data; results are not demonstrated in a timely manner; and working with a lack of trust between IT and business stakeholders". [11]

In fact there were no really interesting in applying Agile in the BI projects, key collier wrote" data community wasn't paying attention to the benefits of agility, while the software community wasn't paying attention to the unique challenges of database development and systems

integration". But along the past years there are more interesting in engaging Agile in the Database, Data warehouse and BI projects, and there are more than one author write in this topic like Scott Ambler, Ralph Hughes, and ken Collier.

Here it's important to mention the newest and important TDWI survey's result about using Agile in the BI and data warehouse projects, which covered 408 responses (business and IT executives, VPs and directors of BI, analytics, and data warehousing, business and data analysts, application managers and other professionals), a total of 295 completed every questions; (research which published in the Q1 2013), they wanted to know whether organizations have been successful in using Agile techniques and disciplines for BI and data warehouse to improve productivity , quality, customer satisfaction, costs, and other metrics.

And they found that the concept of using Agile in the BI and data warehouse projects is amateur. Just over half of respondents in the research study were found to have at least one year's experience with Agile, with 11% indicating that they have over five years' experience. And the researchers found that organizations that have gotten past the early stages with Agile appear to be sticking with it. At the same time just under half of responds have either no experience with Agile (25%) or less than a year (23%). [26]

This survey shows the percentage of organizations which follow the Agile roles rigidly, and do it by the experience, but there are many organizations (this was emphasized by the interviews which made by the TDWI team) follow that goals of Agile development but are not yet rigorously applying this method. [26]

In general there are more interesting in using Agile methodology especially with increasing the evidences which emphasized that Agile approaches lead to higher project success rates, whereas the last survey which made by Scott Ampler in 2010 showed that Agile methods in IT continued to show better results. [5]

Accordingly, we can use the Agile method in the BI projects, and with some adoption on the Agile practices we can meet the challenges. [5]

2.6 AGILE CHARACTERISTICS IN BI:

In general, we can list the following characteristics of the Agile method when it uses to build the BI solution:

6.1. Handle changed requirements:

The requirements in the BI projects change in dynamic and fast way more than any other software projects. In fact, Forrester says, nearly three-quarters (70 percent) of survey respondents say their requirements change on a monthly, daily, or even hourly basis. [27]

The Agile method solves the most important problem in the BI projects, which is changed requirements, the user gives the initial requirements to the development team at start of the project, and after each iteration he can feedback the new requirements which face hem to the development team, to alter the release.

6.2. Documentation:

As mentioned above the user doesn't care about heavy documentation, he cares about the applicable software which solves his business problems, on the other hand the heavy documentation for the fast changed requirements is uneconomical.

Agile method cares more about code writing and present applicable BI software to the user than heavy documentation, (that doesn't mean Agile don't make any documentation, there are some activities that are still important to document, such as documenting your enterprise data model). But it makes the helpful one, this process reduces cost and time of the heavy documentation, which in turn accelerate the delivery time.

6.3. Collaboration:

High collaboration level in the Agile BI helps the developers to get better understanding for the data and business requirements and to make helpful solution.

Users, stakeholders, and developers must share project ownership and work together daily throughout the project: that create a full trust between all them because all share the responsibility to fail not just the development team.

2.7 AGILE METHODOLOGY IN THE SELF SERVICE BI:

The Self-Service BI projects is a new age of BI projects, it has all BI project characteristics beside its own characteristic, so there is nothing special mind us from applying Agile methodology in the Self-Service BI projects. Self-Service BI has some special characteristics we should care about when applying Agile methodology:

- In the Self-Service BI projects we don't just develop BI features and introduce them to the business user to test them (like dashboards), but we should build a BI application, which is the

most important characteristic, and this application should not give the business users a fixed state, but it should enable him to interactive, explore the data, discover trends, turn data into knowledge and get insight upon data.

In this study we built that data model according to the Kimball method upon the business process (we will explain it later), each business process represent a data mart or layer, we used the Agile method to build a data mart in each iteration, then present it to the business user to make the accepting test, and each data mart include many BI features like dashboards and key performance indicators and analysis applications.

For this reason some of the Agile methods don't fit the Self Service BI projects such as SCRUM method (we explain it later) which depend on very small iteration, in this method we develop each BI feature in each data mart alone and present it to the business users to accept it, if it accepted the business user, we should integrate it with the exists BI features, and that will add a new tasks (integration between components) which cost the development team a time and effort.

- Highly interactive between the development team and business users, business user engaged in all development process, because they build the application to use it by them, so it is important to participate the business users in all application development phases: from the data stage to build the user interfaces.

2.8 AGILE METHODS:

There are many Agile methods, and all of them share the same Agile concept, but they different in the practices of each method, Therefore, we discuss these methods and explain how are they different: [6]

2.8.1 Scrum:

Ken Schwaber first described Scrum in 1996; the term is borrowed from Rugby: “[A] Scrum occurs when players from each team huddle closely together...in an attempt to advance down the playing field” [6]

Scrum process defines three main phases: pre-game, development and post-game: [18]

In the pre-game the team holds all the work in "release backlog", then tasks are selected from the release backlog and placed into the “sprint backlog,” and each sprint has a goal. [6]

Development: Team members choose the tasks they want to work on and begin development. [6]
Post-game: the team meets to analyze project progress and demonstrate the current system. [6]
Ken collier describe it as "Scrum is based on the rhythm of working in two- to four-week iterations called sprints and short (15-minute) daily synchronization meetings called scrums". [5]
What differentiate scrum is the small iteration which gives frequent executable software, small team which gives high commendation and the meeting which corrects the work and solves problems.

2.8.2 Extreme Programming XP:

Extreme Programming is important Agile Method Introduced by Beck and Jeffries, in 1998. [6]

What differentiate this method are:

Metaphor: Guide all development with a simple shared story of how the whole system works. [18]

Simple design: The system should be designed as simply as possible at any given moment. [18]

Refactoring: As developers work, the design should be evolved to keep it as simple as possible. [6]

Pair programming: All production code is written with two programmers at one machine. [18]

Continuous Integration: Developers integrate new code into the system as often as possible. [6]

Collective ownership: Anyone can change any code anywhere in the system at any time. [18]

40-hour week: Work no more than 40 hours a week as a rule. [18]

XP is advised to use when: [5]

Team size: from 2-10.

Iteration length: 2 weeks.

Support for distributed teams: Because of XP's focus on community and co-location, distributed teams are not supported.

2.8.3 Crystal method:

Alistair Cockburn introduced Crystal Methods in 2002, Crystal depends on reducing the documentation and asynchronous communication (e-mail threads), to increase the Face to face collaboration. [5]

Crystal focuses on people, interaction, community, skills, talents, and communication as first order effects on performance. Process remains important, but secondary. [6]

Cockburn's methods are named "crystal" to represent a gemstone, the crystal has many clear degrees, the most Agile version is Crystal Clear. The version of crystal one use depends on the number of people involved, which translates into a different degree of emphasis on communication. [6]

Crystal is advised to use when: [6]

Team size: it accommodates any team size

Iteration length: Up to 4 months for large

Support for distributed teams: Crystal Methodologies have built in support for distributed Teams.

2.8.3 Feature-Driven Development (FDD):

Feature Driven Development arose in the late 1990s from collaboration between Jeff DeLuca and Peter Coad. [6]

FDD consists of five sequential processes during them the design and building of the system is carried out: Develop an overall model, Build a Feature list, Plan by Feature, Design by Feature, and Build by Feature. While the first three processes are sequential, the next two (Design and Build) are the iterative part of the FDD. (A feature is a small, client valued function). [18]

Feature Driven Development is advised to use when: [6]

Team size: varies depending on the complexity of the feature at hand.

Iteration length: Up to two weeks.

Support for distributed teams: FDD is designed for multiple teams and while it does not have built-in support for distributed environments, it should be adaptable.

2.8.5 Dynamic Systems Development (DSDM):

Dynamic Systems Development Method is not so much a method as it is a framework; it was developed in 1994 by Stapleton in the United Kingdom as a framework for rapid application development. [11]

DSDM has underlying principles that include active user interaction and frequent deliveries. The Three major phases of DSDM: functional iteration, design and build iteration and implementation, are themselves iterative processes. [18]

So DSDM suggests making a series of prototypes in short cycles to gradually solve the problems that are not precisely defined in advance or not stable during time (it allows handling changed requirements). [18]

2.8.6 Adaptive Software Development (ASD):

Adaptive Software Development (ASD) was developed by Jim Highsmith and published in 2000, this method strongly encourages incremental and iterative Development and It focus on produce executable features rather than on the tasks to produce them. [18]

The Plan, Build, Revise lifecycle phases are adapted to fit the ASD characteristics as following: [11]

Speculation: this phase instead plan phase. Speculation phase recognizes the uncertain nature of complex problems.

It's more about explored the business problems and requirements and understands them.

Collaboration: this phase replace the Build phase. Collaboration gain the ability of people to build successful solution and share ideas to solve problems which face them.

Learning: this phase replaces the Revise phase. The customer's feedbacks gain the Team experience, and this knowledge can help development team to build successful projects.

2.9 WHAT AGILE METHOD TO USE?

In the following we will compare between the Agile methods according to their ability to use in the Self-Service BI projects:

Scrum:

BI/DW projects in general have many characteristics differ them than the traditional system projects, the Agile methods which are common to use in the traditional systems like Scrum method, face difficulties when we want to apply them in the BI projects.

David Stodder the director of TDWI research for BI, wrote in the Q1, 2013 research "the specific challenges of BI/DW which can include the need for longer cycles", which contradict with the Scrum principles, and he wrote "make it important for organizations to find scrum masters who have experience not only with Agile method but also with BI/DW development" if they want to use Scrum method.

In the same research the organizations experienced with BI/DW Agile development often find that they need some of related work functions rather than strict Scrum cycles, so that team members have full sprints to address specific requirements. [26]

Extreme Programming XP:

Need a lot of programmers, where all production code is written with two programmers at one machine

Crystal method:

Used in the long iterations where the iteration length is Up to 4 months for large projects. And it increases the face to face communications which makes Overflow of requirements from the business users

Feature-Driven Development (FDD):

Just the design and build are iterative where the other first three phases are sequential. The user can adapt just the code or design, but for example if a new requirement appears it not taken into consideration, and it's better for the distributed teams.

Dynamic Systems Development (DSDM):

It's appropriate for business projects, but in this method a many prototype must provide to the business user to define the problem and its solution in the most appropriate way, which cost a lot of work and drowns the users in details.

Adaptive Software Development (ASD):

The most appropriate one to the BI projects, where it focus in producing exactable features to the users , and appropriate for the business problems

ASD methodology is the most fit between other Agile methodologies to the BI projects in general, but when we go through the details of the BI project development, it has some weakness in some points like the data sources and destination.

Alnoukari suggests a model called it ASD-BI, he introduced it as "a model is built in a way to consider the strengths of the previous models, and avoid their weaknesses". [11]

Alnoukari suggests the data sources and the repositories for the model as the following: [11]

- ASD-BI data model is using data warehouse/data marts as the core data source for any BI application: The model doesn't use the data sources without cleaning and standardized format.

- ASD-BI model store a metadata repository which is "a separate data storage that is used to describe the data stored in the data warehouse/data marts".
- ASD-BI stores a knowledge repository which is "used to store and maintain the knowledge discovered for future use". In our study we can store the reports templates to use by the business users.

In fact Alnoukari used this model in the data mining projects, but we think that the ASD-BI is fit the Self Service BI characteristics which mentioned above where in it we can build analysis applications in the speculation phase and enables the interactive between people in the collaboration phase. (explained in the next chapter)

2.10 ASD PHASES

2.10.1 Speculation:

In this phase, we conduct all the project initiation tasks including: determining the objectives, mission, requirements of each cycle, determining the optimal number of cycles and time box for each cycle, making preliminary size and scope estimation, and identifying the high risk items early in the project [11].

The outcome of this phase are: data marts with data processed, cleansed and ready for use in the application building.

2.10.1.1 Business Understanding:

This process helps to understand the business requirements, to expect number of users and their location, and the number of subjects that might be addressed (e.g., sales, finance, purchasing, HR, etc.).

In general understanding the business helps the development team to recognize the goal of BI project in the organization, what are the departments which really need decision support system that help them to make an added value to their works?

How can the development team understand business?

The development team should understand the business activities such as: the organization mission, vision, strategy, and goals.

They should understand business strategy plan and what accomplish from it, understand the organization hierarchy, how each business process interact with others business process, what are the roles and responsibilities of the people in the organization and what are the reports and analytic required?.

2.10.1.2 Data Understanding:

In this process the development team identify the data sources, what are the data systems, data flow and data description.

2.10.1.3 Objectives/Hypotheses Setting:

After the development team understand the business and data, they can define the main goals from the project, and what are the analytic applications which required to get the project goals?

2.10.1.4 Data Preparation/ETL:

The term ETL (Extract/Transform/Load) is mostly used with BI projects, whereas data preparation is commonly used with knowledge discovery processes, [11] After the development team understand the requirements and applications needed, they can build the data model which they will use it later as a data sources to solution:

In this process they build data marts as following:

- Coupling Business Requirements and develop a first draft of the dimensional model.
- This model identifies the fact table, associated dimensions, and attributes.
- Identify relationships between tables.

And prepare data to be a standard data sources as following:

- Define Data Staging Requirements: The data staging process has three major steps: Extraction, Transformation and Load
- Define Staging Procedures
- Define Extract Services
- Define Data Transformation Services
- Define Data Loading Services
- Define Data Staging Job Control Services

2.10.2 Collaboration:

In this phase the development team members share their knowledge to solve the business problems, and develop the solution.

The author of this method wrote "ASD-BI collaboration phase consists only of the modeling/data mining process". [11]

But because our project is not data mining project we will replace it with the solutions building process.

2.10.2.1 Solutions Development Process:

The goal of this process is delivering applicable software with fewer defects, in shortest delivery times and with better values to the customers.

In this process we develop:

- The analytic applications
- The user's interfaces

2.10.3 Learning:

In this phase, the development team focus on quality review, the author wrote "Postmortem sessions and project status reviews are needed to evaluate each cycle's progress and the whole project status".

In other word in this phase they make the verification "Are you building it right?" and validation test "Are you building the right thing?"

ASD-BI learning phase consists of two process steps:

2.10.3.1 Evaluation:

This process deals with the business users feedback, they get the application and verify if it met their requirements, in this phase the business users suggest the changes and they can suggest a new requirements to add to the application.

also technical reviews are conducting during this process. [11]

2.10.3.2 Deployment

This is the endpoint of each ASD-BI cycle. The knowledge discovered by the modeling/data mining step should be organized and presented in a way that the customer can use. [11]

2.11 AGILE BI CHALLENGES:

Through our search and conversation with developers, who used the Agile method in their BI projects, I found that the Agile method has many challenges, and it sometimes drive the BI project to failure:

11.1. The development team uses the concept of Agile but they don't apply the method rules in rigidly:

In the Q1, 2013 TDWI research, Interviews conducted for this research study suggested that interest in Agile is stronger than experience, some respondents said that their organizations are trying to follow that goals of Agile development but are not yet rigorously applying this method. [26]

In fact using Agile without following the Agile rules in rigidly delivers the BI projects to fail, this was emphasized by the co-author of a BI greatest methodological guide (Moss & Atre) when condition that we should use Agile methodologies with rigor in BI projects to success. [11]

11.2. Overflow of requirements:

When the business users see their requirements translated to executable software in fast time, that encourages them to ask for more requirements and not in necessary they need this additional requirements, and they are still asking for more until the entire team is swamped with additional work, in the research about the system features used by the business users they found that 45 percent of features were never used and only 20 percent of features were used often. [5]

Therefore, the development team should be careful about the business users frequently adaption on the requirements, following the specified time boxes, and the project manager must know when to refuse some of the additional requirements.

11.3. Documentations level:

As mentioned above, Agile method minimize the documentations needed in the project comparing with the traditional methods, but that didn't mean we don't need any documentation,

there are a specific level required to be as archive when we want to make any development in the future.

11.4. Team skills:

The development team must have high level of communication and technical skills.



3. THE APPLICATION PART:

The following figure represent The ASD- BI methodology which made by Alnoukari to make to use in the Data mining projects [11]

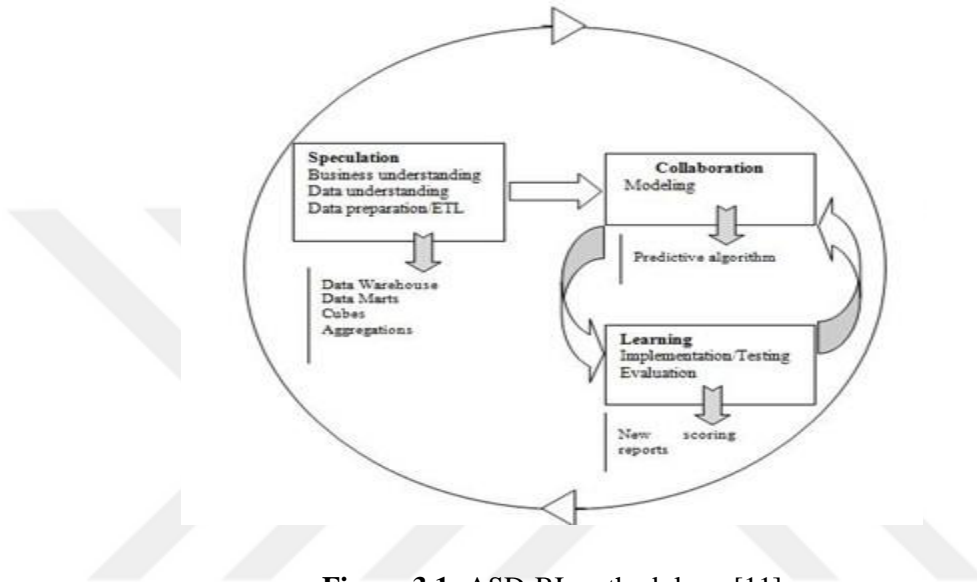


Figure 3.1: ASD-BI methodology [11]

In this study the ASD-BI method is configured to use in building the Self Service BI solution, as shown in following figure.

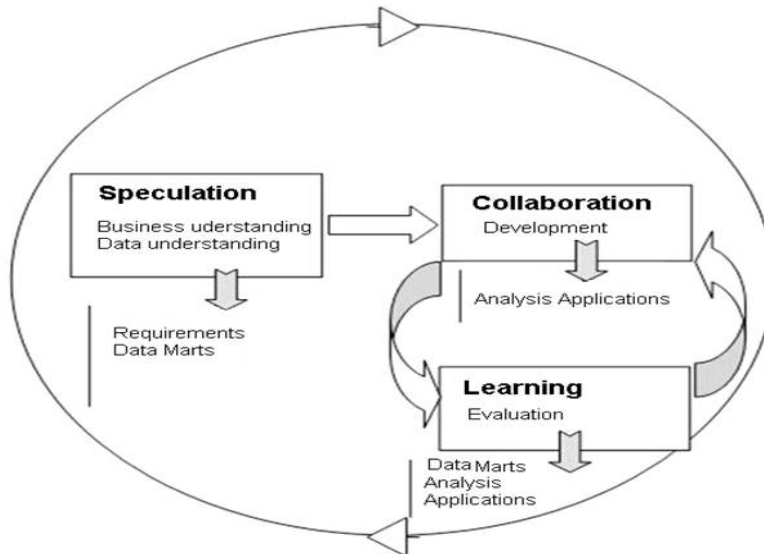


Figure 3.2: ASD-Self Service BI methodology

3.1 FIRST PHASE: SPECULATION

This is the most important phase in the methodology and it takes the longest time to be prepared comparing with other phases, in this phase we understood the business, defined the business problem, defined the company business process, understood the data, and defined the overall project hypotheses and objectives.

The outcome of this phase is the company data marts: sales, purchase, inventory and accounting data marts, these data marts should build in the adaptive way to integrate the new challenges.

3.1.1 Business Understanding:

The main objective of this session is to understand the business problem, and elaborate the initial hypotheses for each data marts

In this session we defined the:

3.1.1.1 Case Study Statement

The problem statement is illustrated by a case study on a wholesale company; such companies buy the goods from the manufacturing and agents and distribute them to all retailers.

3.1.1.2 Company Workflow

To start with, we explained the company's workflow and architecture to understand the business case and to focus on the problem:

Wholesale company is a type of business projects that buy the products from the manufactures and deliver them to the retailers through a distribution channel.

The wholesale company has many departments: purchasing, warehouse, sale, accounting, marketing, distribution.

The wholesale company has a general manager in the top management level and departments mangers in the second management level.

The general manager takes the strategic decision for the company, organize, coordinate and monitor the company works, set the plans to reach the company's goals.

The departments managers take the tactical decisions, put the tactical plans which serve the strategic plan, each department manager is responsible for its department, coordinate monitor and supervise on all the process and human resources in its department.

Company workflow as following:

- The purchasing manager determine what the items needed to be ordered (he determines the required products upon the required items in the market, decrease in the inventory or the vendors offers).
- The purchasing manager demand a purchase orders from the certain agents.
- The warehouse receive the required items from the vendors, check it and the inventory manager get a report by all the entered items to the warehouse and he decide to return any damaged products or unnecessary products.
- The purchase manager in the head office gives the sales manager a total numbers from each product entered the warehouse.
- The marketing department put a marketing strategy, they have a promotion strategy and they give offers to the clients.
- The Sales department receives the order from client and turns the invoice to the accounting department for the invoice value is added to the customer's account and turns a copy of the invoice to the warehouse section.
- The inventory department Get the products out of the warehouse and shipped via shipping channels to the customer.
- The accounting department handles all the company financial process (such as: get client payments, deliver usual schedule reports to the general manager about company financial state, and give the employee salaries...).

The wholesale company stores all the previous daily transactions in database.

3.1.1.3 The business Problem

The main problem is that: the company has a huge amount of stored data, and they use one of the traditional Business Intelligence solutions types, which doesn't analyze all the stored data; also the information which they get from this analysis don't support the managers in the second level in the best way, and don't enable them to discover the business in the right way to make the best and fast decisions.

So those managers asked the IT team to build analysis applications for them to respond their requirements, but responding to such emergency requirements late the IT team plan and tasks, also the time which required to understand and build the managers requirements by IT delay the decisions making process, and sometimes the IT misunderstand the managers requirements so they deliver not the required analysis; or the managers export the data to a spreadsheet like Excel to make their analysis by themselves, such solution has many limitations such security and consistency .. and delay the decision making process. In general we can consider the previous state the main reason for Business Intelligence projects fails.

3.1.1.4 Deliver the Knowledge

In our case study we can notice how the Business Intelligence solution can fall short in providing the value which expected by the decision makers in the company. The static reports which produced by the stock system and the accounting system which embedded in the company system, besides the advanced reports which produced by the IT department to fill the general manager demand, don't use all the stored data also they provide low aggregation level and don't allow other management levels to discover the business and share their vision about departments.

As mentioned before in our study we focused on the middle managers and the knowledge workers, with enable all the management levels to benefit from the solution outputs, through sharing the knowledge between all decision makers to create a clear vision about the company state in all management levels.

Whereas the middle managers must access all the data which relates to their departments and use the knowledge discovery tools which give them a complete vision for their departments, help them to make the better decisions and give them the ability to transfer the vision to the general manager.

For example, when the sales manager analyzes the sales for specific product according to the previous years, he can predict the quantity which needed in the coming quarter, so he can advise the purchase manager to get the best price for the predicted quantity.

And in the same time the purchase manager after analyzing the average prices for this product from the previous purchase data, he can predict the total cost of this deal and can share this knowledge with the financial manager to detect the cash statement.

We can drop this knowledge sharing process on the entire company department and for all business process.

However, the Self Service BI gives the department managers and knowledge workers applications to do the previous activity with lower cost than traditional BI.

Accordingly, when the information are exchanged between departments that create valuable information integration, also it makes the departments managers more productive, and can benefit the senior managers when they set the strategic plans. Because it gives them accurate and insightful information.

In general Self Service BI provides easy access to the information for the mid-level management and base level, and flexibility to integrate information, with effective cost for the company.

In our case study we studied the requirements of each department manager and knowledge worker, we used the appropriate tool with the previous advantages, and we built the analysis application to meet the need of each one of them.

3.1.2 Data Understanding:

In this session we specified the software used by the company, the business process for our case and determine their data sources; as following:

3.1.2.1 Software Used:

The company used an accounting and stock system, the accounting system handle all the accounting process, and the stock system handle all the transactions related with products.

These systems have a central relational database to store the daily transactions data, with SQL database management system, and they used a user interface developed by PHP programming language which enables the users to store and retrieve the data from the database

3.1.2.2 Business Process, Data Sources:

In this section we defined the business process for our case study, because we used them later to build the data models according to Kimball method.

Kimball defined the business process "it is a natural business activity performed in your Organization that typically is supported by a source data collection system". [4]

3.2.1. Purchase Process:

In this process the company buys the products from the agents, the company records the following data about this process in the purchase invoice form:

- Invoice number: represent the invoice ID, it's a unique number.
- Order number: represent the purchase order which submitted by the purchase manager to the agents to inform them about the needed products.
- Supplier: the company gives the suppliers accounts a general initial accounting number (4011-1), this account contains many classes like the domestic suppliers or foreign suppliers (275 represent the account number for the domestic suppliers).
- Entry name: the ID for the employee who enter the data to the company system.
- Description: any more information explains the invoice.
- Date: represent the getting products date.
- Details: in this invoice sections, we have many records (according if the invoice has different items), this records contain details about the purchased products such as:

The product Symbol, the expiry date, supplier: it's a unique number identity each supplier in the database, quantity from each item, the offer which we get according the purchased quantity, discount on price for purchased quantity (like 22, 5 %), price for each unit, total cost, and total cost after discount.

At the bottom of the invoice we have the total cost of the invoice, the discount on the total of the invoice, and the total after the tax

3.2.2. Inventory:

In this business process the incoming products are received, stored in the appropriate manner to save them from damaging, and the outgoing products are delivered by the distributing channels to the clients.

The inventory business process is an important one, and the activity in this process produce respect amount of data, but in our case study what important for the management is to identify the available quantity from each product item. Therefore, the inventory team record the incoming and outgoing products' information and their quantities from the: purchase, sales, return sales and return purchases invoices.

Processing this invoices' data helps the company management to stay informed about the available quantity from each product type and their state.

3.2.3. Sales:

This business process interests in selling products to clients, the company records the following data about this process in the sell invoice form:

- Invoice number: represent the invoice ID.
- Client: the company allocates a general initial account number for the all clients (4111-1), which on turn contains many classes like the (regular client with account number 2310).
- Entry name: the ID for the employee who enter the data to the company system.
- Date: represents the date of selling the products.
- Salesman: the salesman ID.
- Description: any more information explains the invoice.
- Address: client address.
- The details section: each record in the detail section include the symbol name, expiry date, supplier ID, the sold quantity, percent discount on price, sold price, total , total price after discount.

At the bottom of the invoice we have the total invoice value, the discount on the total invoice value, and the total after the added value tax.

3.2.4. Accounting:

The company stores all the accounting transaction in the General Ledger, and as known the General ledger is the process of prepare and record the accounting transactions to the appropriate fiscal period and accounts.

General Ledger contains data about the purchasing & accounts payable ,invoicing, accounts receivable, and fixed assets.

In fact the general ledger (G/L) is a core of accounting system, because it ties together the detailed information about purchasing, payables (what you owe to others), receivables (what others owe you), also Expenses and revenue.

In this business process the company records the total value for each purchase invoice as a Financial commitment by the Corporation for Supplier without details (details handled in the purchase process),and recorded the total value for each sell invoice as a Financial commitment by the client for the company without details.

In general The General Ledger identify each account type (for example, asset, liability, equity, income, or expense) by a digits, and each account type contains all the related accounts with numbering scheme For example, account numbers from 1,000 through 1,999 might be asset accounts, whereas account numbers ranging from 2,000 to 2,999 may identify liabilities.

For example: the company identifies the Financial fixed assets account by a 22 digits, All accounts that fall under this category are identified by a number from 221 to 228, and the account with number 221 which refers to the lands account contains four accounts from 2211 to 2214.

The General Ledger saves all the transactions which happened to any account, and it provides the company with totally credit or debit value for each account to build the budget.

The following example explains how the data is stored in the General Ledger:

This is a daily transaction, it's a payment from our company to one of the supplier by a cheque, the system by this journal entry stored in the general journal data like : the date , transaction ID , supplier account ID (the beneficiary from this payment) , the description in the General Ledger

(the Account that the payment will decrease its totally credit), the payment value in the debit section, and in the credit section the system save data about (the Account that the payment will increase its totally credit).

So what saved in the General Ledger from the journal entries in the general journal are the decreasing and increasing in the accounts total.

In the same manner the accounting system handle all the payment from the clients, salaries to the employees, Expenses and revenue.

3.1.3 Hypotheses/Objectives Setting:

In this session we benefited from the past sessions to get the project goal "which is deliver the right information to the right knowledge seek in the right time".

We made interviews with the department managers to collect their requirements and to understand how they want to benefit from the stored data.

And we added some analysis applications, that we think they help us to achieve the project goal.

In fact, gathering the requirements is very important process, and if we gather the right requirements which represent the real business questions, we can define the answer for these questions, consequently we can build successful solution.

We gathered the following requirements by making interviews with the department's managers.

3.1.4 Data Preparation/ ETL (Extract, Transfer, Loading):

After we got the requirements now we can build the data marts or data models, we built four data marts for each business process (Sale, Purchase, Inventory, and Accounting).

3.1.4.1 Comparing the Data Model Building Methods:

As mentioned in the first chapter we built the data model according to the Kimball method, (the alternative method is Anemi method); in fact these two methods are used to build the Data Warehouse model, they share many characteristics like the ETL process, queries by the users, and others.

In her article Mary Breslin (DW builder) compared the essential features between Kimball and Anemi methods and posted the following table: [28]

Table 3.1: Comparing the data model building methods [28]

	Inmon	Kimball
Methodology and architecture		
Overall approach	Top-down Bottom	Bottom-up
Architectural structure	Enterprisewide (atomic) data warehouse "feeds" departmental databases	data warehouse Top-down Bottom-up Data marts model a single business process ;enterprise consis tency achieved through data bus and conformed dimensions
Complexity of the method	Quite complex	Fairly simple
Comparison with established development methodologies	Derived from the spiral methodology	Four-step process, a departure from RDBMS methods
Discussion of physical design	Fairly thorough	Fairly light
Data modeling		
Data orientation	Subject- or data-driven	Process Oriented
Tools	Traditional (ERDs ,DISs)	Dimensional modeling, a departure from relational modeling
End users accessibility	Low	High
Philosophy		
Primary audience	IT professionals	End users
Place in the organization	Integral part of the Corporate Information Factory (CIF)	Transformer and retainer of operational data
Objective	Deliver a sound technical solution based on Deliver a sound technical solution based on	Deliver a solution that makes it easy for end users to directly query the data and still get reasonable response times

What important to our case study is comparing between these two methods from the data model building side. Because we don't want to build a DW but we want to build a data marts; in the following, a brief explaining for some features posted in the table above to understand the difference between the two methods and why we use the Kimball method: [28]

- Anemi method use the top down approach: whereas this approach used the Entity Relationship Diagrams (ERD) to explore the entities, their attributes and relations between entities; The corporate ERD is the sum of all department ERDs.

Kimball's approach bottom-up approach is recommend building one data mart per business process. The sum of all data marts is the organization's data warehouse.

- The team in Anemi method derives subsequent departmental databases using the code and processes which they created when they developed earlier departmental databases.

Whereas in the Kimbal method the team used the Dimensional modeling which begins with tables rather than entity-attribute data models such as ERDs. The tables are either fact tables or dimension tables.

- The Anemi method tools and methods can be actively used only by IT professionals; the Kimbal method Dimensional modeling tools can be actively used by users with some special training.

Accordingly, there are many differences between the two methods, and we select the Kimball method, because it doesn't need IT professionals who special in the ERDs and other relational database tools; also it enables the business users to share in the building process (which as mentioned previously represent an important role to success the solution).

Besides, it's easier than Anemi method in building and modeling; the important thing is when we use the Kimball method we can reduce the redundant values, Because we build the data mart for one business process, in contrast, in the Anemi method we need to use the same process in many departments, for example if we followed the Anemi method we need the sold products process data in the sales data mart and in the marketing data mart; but in the Kimball method we avoid save the same data twice.

For these reason we select the Kimball method and we think the Anemi method is more fit to the big organizations than the small and midsize.

3.1.4.2 Build the Data Models:

our case has a central SQL server database which built by the accounting system company as a commercial version, so the dependencies between the tables are hidden and the columns in the tables have unclear names, and as mentioned above we want to build four data models (marts) for the business process (sales, purchase, accounting, inventory), as example we will explain the sales data mart here; We have specific requirements, therefore we must extract specific columns from the database and transfer them with their data to build the sales data mart.

After scanning the database, we found that the:

- "St_hist" table contains the general data from the data sources such: invoice id, date, salesman ID, invoice total amount.
- The "St_Det" table contains the detailed invoice data such: invoice id, date, item id, purchase price, selling price, discount, onhand quantity, sold quantity, and option quantity.

- The "stock" table contains the following data: item id, country of origin, good name, onhand quantity.
- The "units" table contains: item id, selling price.
- The "peopleInof" table contains: client Id, classification of the clients.
- The "LYTD Sales 2011" table contains: month, sales amount in 2011.
- The "SalesMan" table contains: sales man id, sales man name.
- The "ClientsInformation" contains: client ID , client address, client name

these columns names are not the real columns names as stored in the database, but we explored the database and renamed the columns to make them more understandable by the business users, also to enable the association between tables; because if two tables have the same columns which have the same data element values type, they should have the same columns name to associate by QlikView engine.

After the columns needed to extract from the database determined, we can build the data model for the sales business process.

To build the model we connect to the database and extract the needed tables with the required columns and the data which needed to fulfill the requirements; that done in QlikView in the edit script window as following:

" St_Hist" page:

It's good to mention that in each data model, we specify one page in the Edit Script for extracting each table's columns; that will help us to follow the errors and to adapt in the future; as we will see in the following:

We open the "Edit Script" page from the QlikView start page, and start to write the extract and load code to extract the tables and columns from the database:

At first in the St_Hist page we set the variables names, then we connect to the SQL database (in general QlikView can connect to ODBC, OLE DB, web files and table files to extract data from them); then we select from the "St_Hist" table (which contained the general information for each invoice) the needed columns, as shown in the following figure:

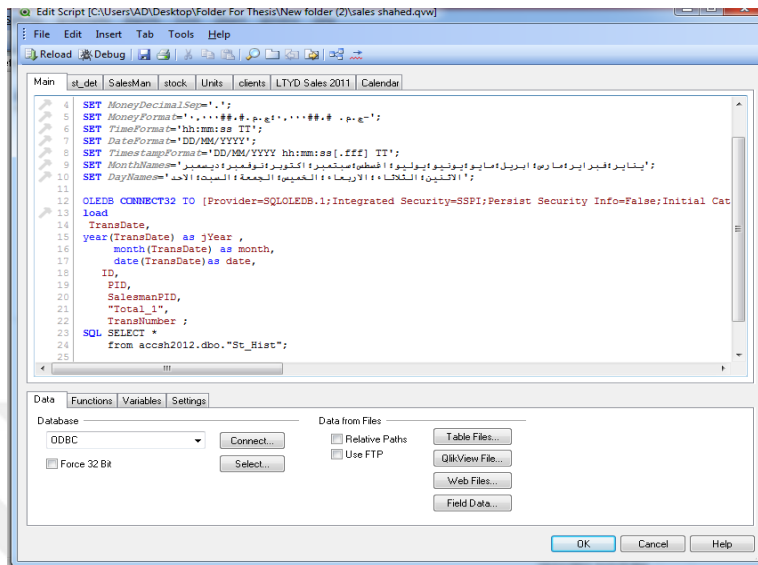


Figure 3.3: "St_Hist" page in edit script window in QlikView

"St_det" page:

In this page we select the columns which needed from the " St_Det" table which contains the invoices details, as we see we put a condition in the SQL statement, this condition retrieve just the sales and return sales invoices's details, as shown in following figure:

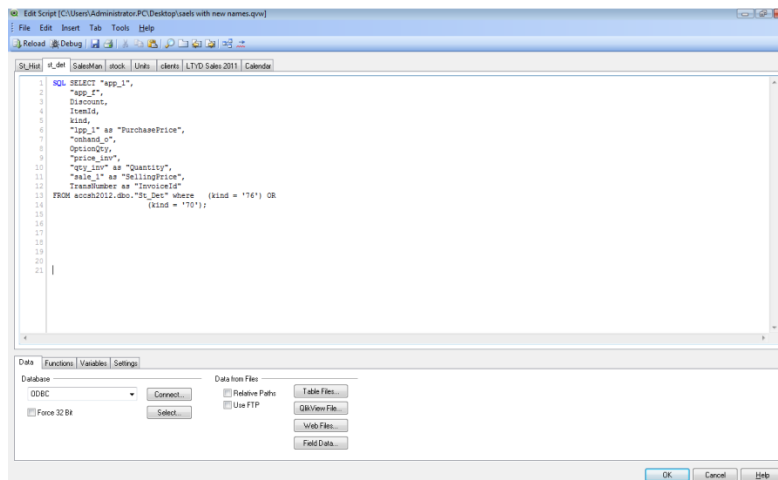


Figure 3.4 "St_det" page in edit script window in QlikView

"SalesMan" page:

In this page we select the Sales men data from the SalesMan tables, as shown in following figure:

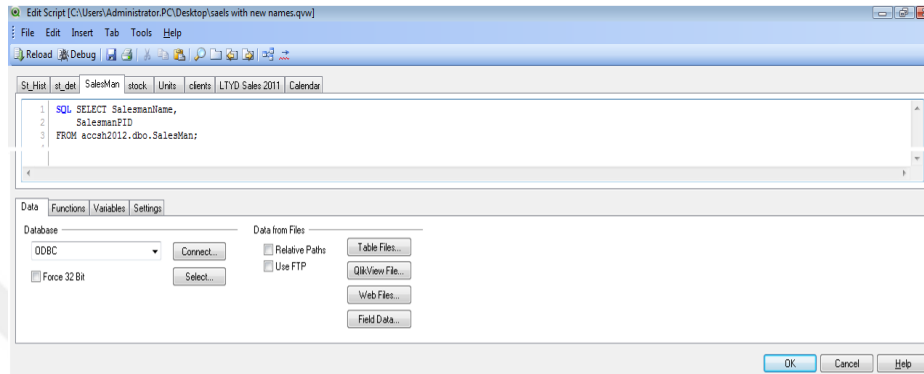


Figure 3.5: "SalesMan" page in edit script window in QlikView

"Stock" page:

In this page we select the stock data from the "Stock" table and start to rename the columns to have the same names in other tables; for example we change the "id" column name to be "ItemId" to associate with the "ItemId" in the "St_Det" table.

Also we changed some columns name to identified them and to be more clear for the business users in the future; for example we change the "name" column in the "Stock" table to be "good name" to identify it later, as shown in following figure:

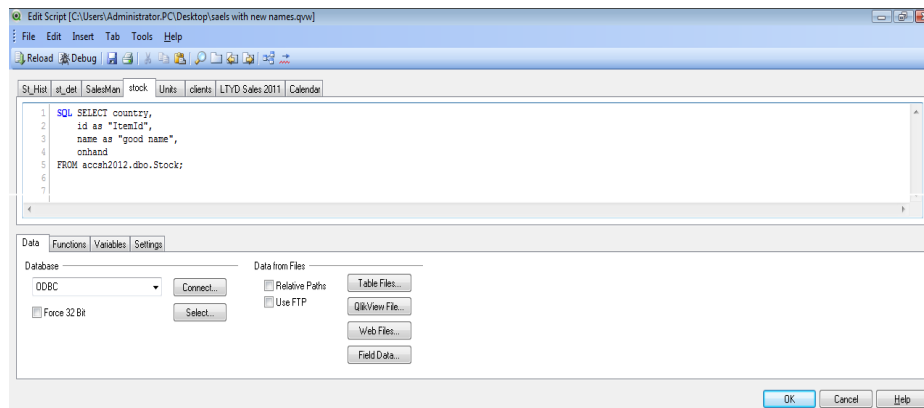


Figure 3.6: "Stock" page in edit script window in QlikView

"Units" page:

In the Units page we selected a columns which complete the stock information like the selling price, also we changed the "id"column name to be "ItemId" to associate with the "ItemId" column in the "St_Det" and "Stock" tables, as shown in following figure:

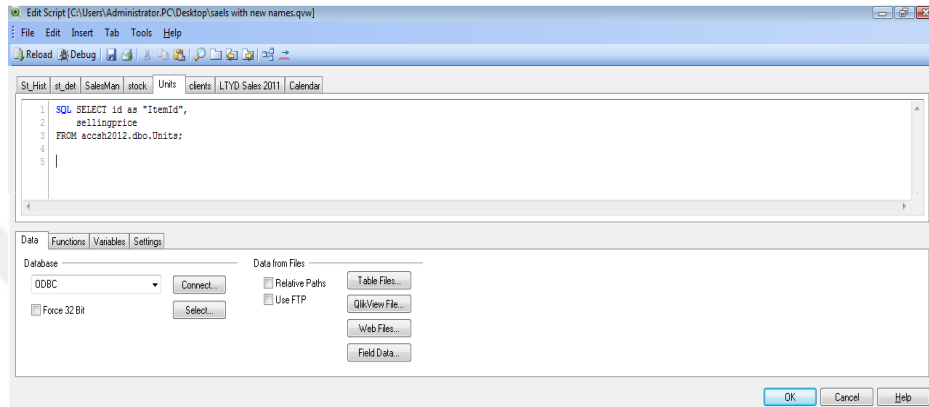


Figure 3.7: "Units" page in edit script window in QlikView

"Clients" page:

In the clients page we selected some columns which relate with contain clients data from the "ClientsInofrmation" and "PeopleInfo" tables.

Also we changed some columns name to associate with other columns in other tables; for example we changed the "code" column name to be "PID" to associate with "PID" columns in the "PeopleInfo" and "St_Hist" tables, as shown in following figure:

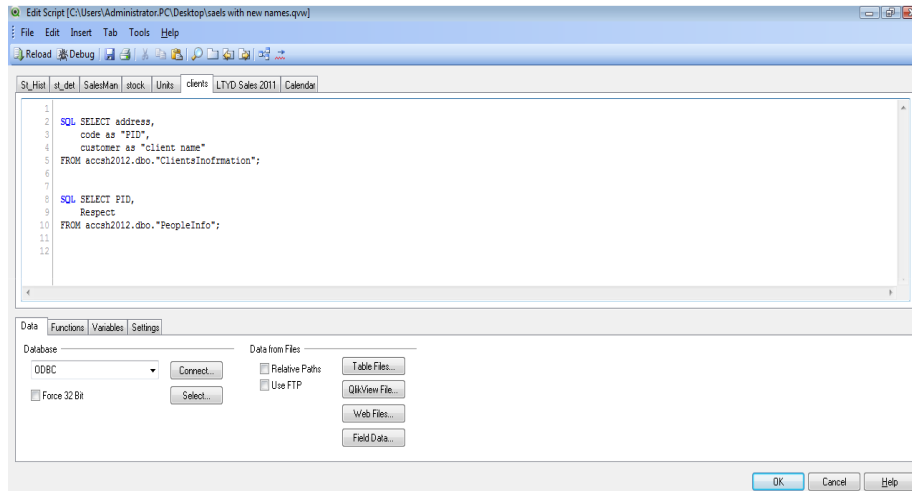


Figure 3.8: "Clients" page in edit script window in QlikView

"LTYD Sales 2011" page:

In this page we selected the sales data from the previous year as shown in following figure:

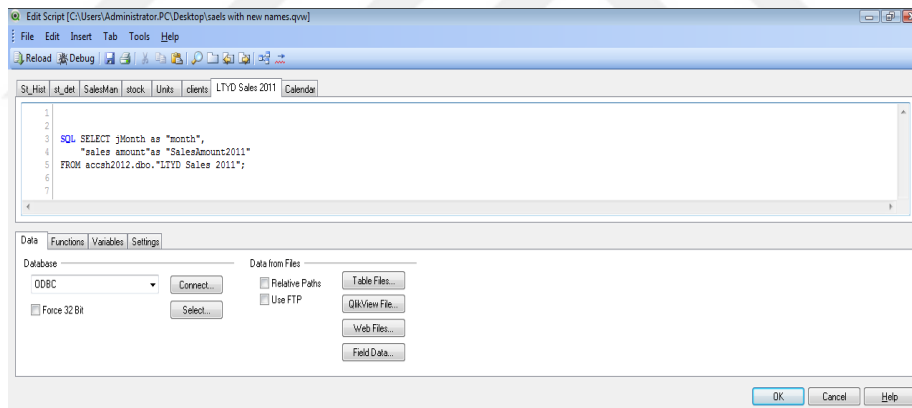


Figure 3.9: "LTYD Sales 2011" page in edit script window in QlikView

"Calendar" page:

In this page we determine the "V_MIN_DATE" and "V_MAX_DATE" variables also we create a Calendar temp table to unified the date form, as shown in following figure:

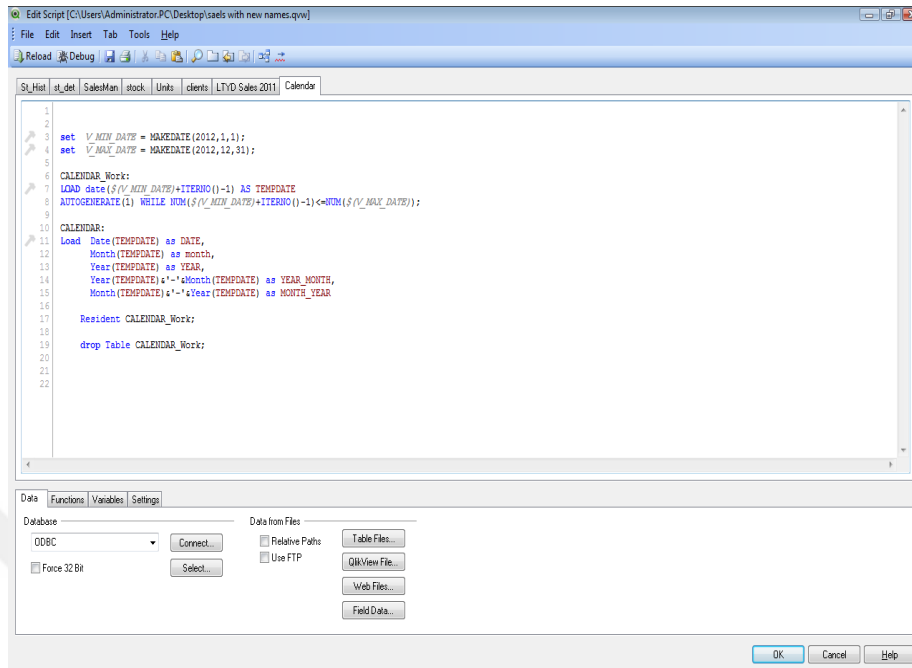


Figure 3.10: "Calendar" page in edit script window in QlikView

After writing the extract code and debugging it, the data are reloaded from the SQL database to the memory; by pressing the Relod button in the Edit Script which start to load the data

After the loading process the model has been built, and it shown in the table viewer window, as shown in following figure:

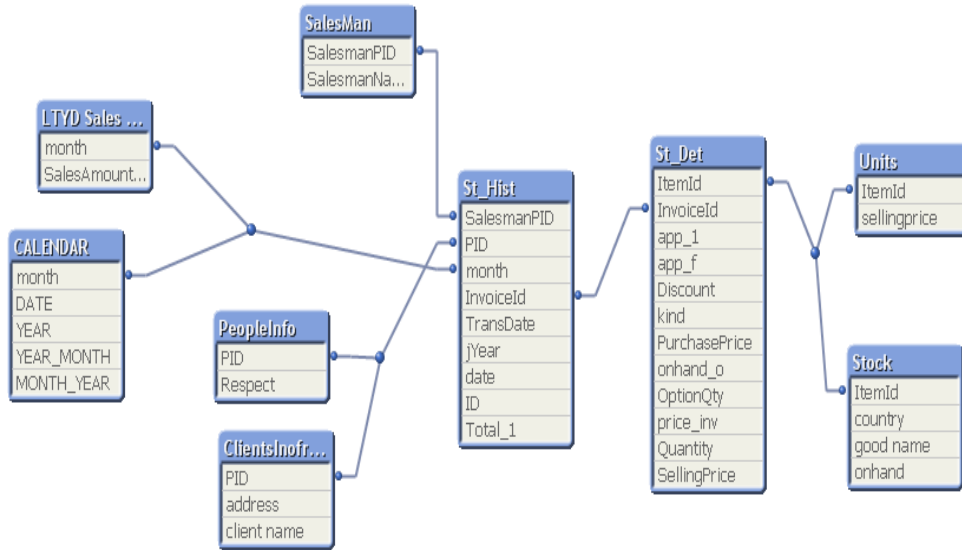


Figure 3.11: Sales data model

Similar processes done to build the other business processes models, and the models will shown as following:

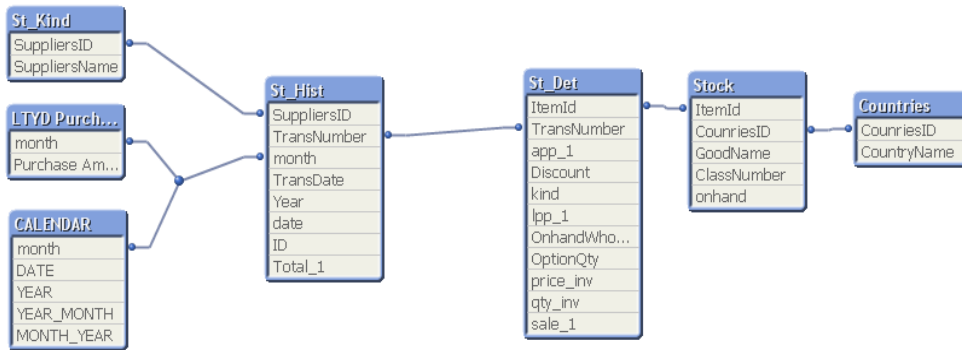


Figure 3.12: Purchase data model

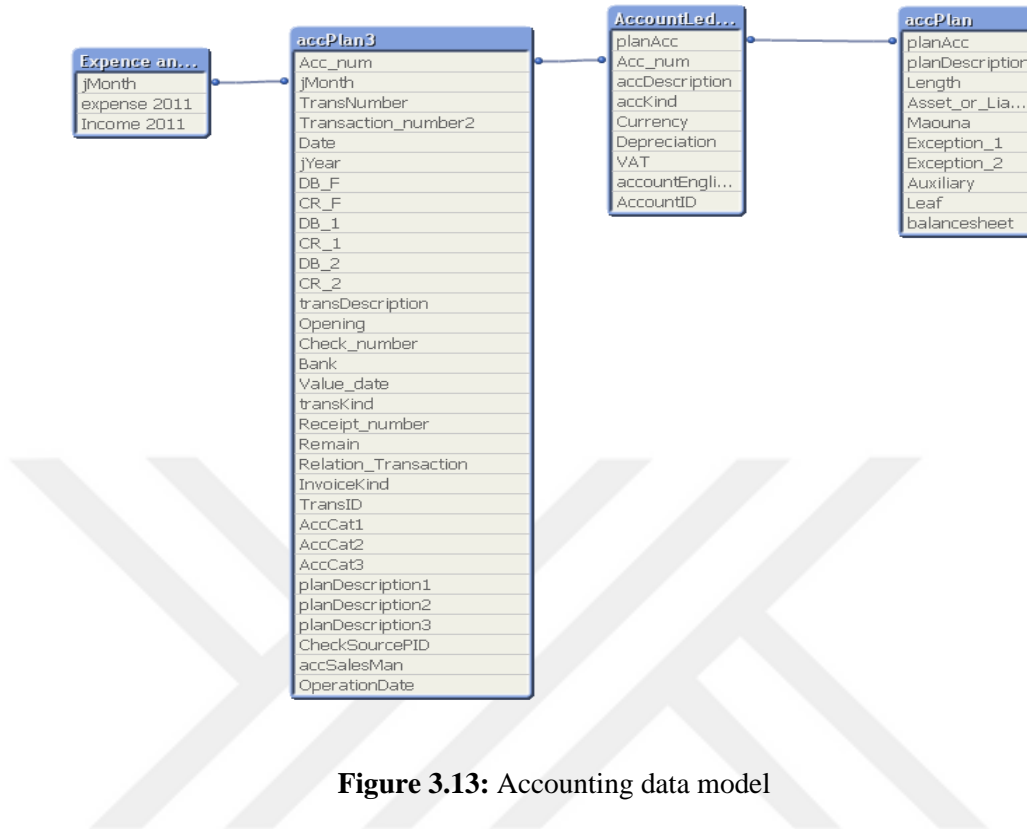


Figure 3.13: Accounting data model

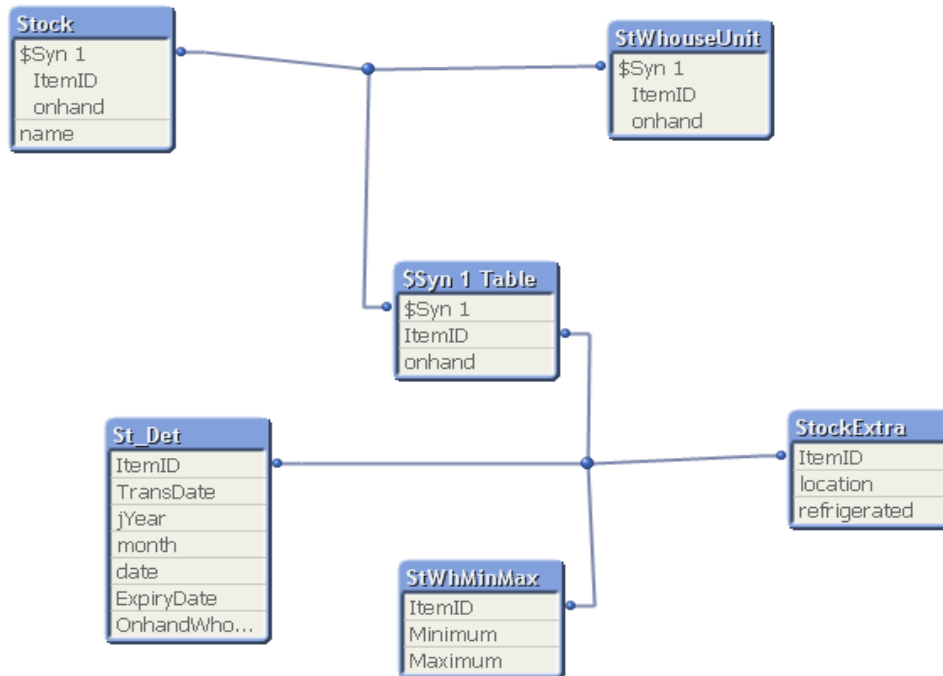


Figure 3.14: Inventory data model

3.2 SECOND PHASE: COLLABORATION:

This is the core phase of the ASD-BI process model, in this phase we build the analysis applications like ad-hoc reports, dashboards, charts, Key performance indicators.

As mentioned above we built four solutions (a solution for each business process), and we divided each solution to many sheets, each sheet contains a specific related analysis applications (charts and tables...).

In the following we explain some of the analysis applications in the sales solutions, how we built them and how can they benefit the sales managers and the salesmen in the sales department:

Sales solution:

The Sales solution has many sheets; at first we will start with the metadata sheet, in this sheet we declared the variables, which used in defining the Expressions that needed to build the analysis applications, it is like the measures in the fact table.

Note: we can declare the variables in the Edit Script page, but we do that in the metadata sheet because that is easier to adapt in the future and to group all the variables in one table.

The following variables are examples on the variables which used in the Sales solution:

- $eGrossSales = \text{sum}(\text{If}(\text{kind}=70,(\text{sale}_1 * \text{qty_inv}), \text{Null}()))$

This equation written in the Edit Expression window which provide a lot of functions and fields name to build the equation.

The eGrossSales variable return the total of all the sales without decrease the sales returns. the equation above means that when the invoice kind = 70 which represent the sales invoice, multi the sales price with the quantity sold, else return null.

- $eMargin = \text{sum}(\text{If}(\text{kind} = 70, ((\text{sale}_1 - \text{lpp}_1) * \text{qty_inv}), \text{Null}())) - \text{sum}(\text{If}(\text{kind}=76, ((\text{sale}_1 - \text{lpp}_1) * \text{qty_inv}), \text{Null}()))$

This variable returns the margin for the gross sales by minus the purchase price from the sales price and multi the result with quantity sold, then calculate the lose margin from the sales return. and minuse it from the gross sales margin.

- $ePromotion = \text{sum}(\text{If}(\text{kind}=70, (((\text{Discount} * \text{price_inv}) / 100) + (\text{OptionQty} * \text{sale_1})), \text{Null}())) - \text{sum}(\text{If}(\text{kind}=76, (((\text{Discount} * \text{price_inv}) / 100) + (\text{OptionQty} * \text{sale_1})), \text{Null}()))$

This variable calculate the total of percentage discount and quantity discount which given to the sales invoices minus the returned promotion from the sales returns.

The following figure shows the metadata sheet which contains the variables grouped in one table:

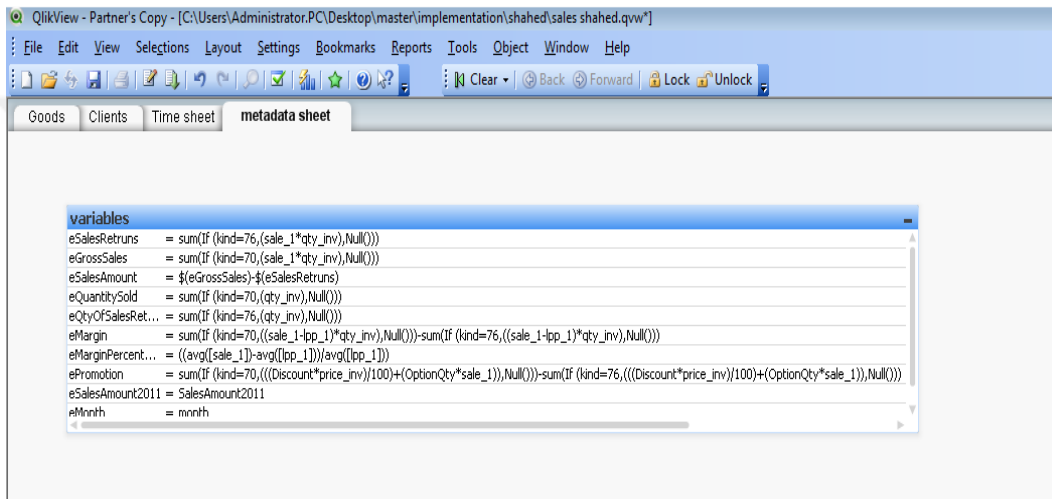


Figure 3.15: Metadata sheet in the sales analysis application

after we explained the metadata sheet, we will explain some of the analysis applications in the sales solution:

Goods sheet:

This is the first sheet in the sales solution, It contains the analysis applications which help to analyse the goods data. The following print screen represents the Good sheet which contains good table and charts visualize the information in the table and comparing between them.

The good sales table in the following figure presents information about the following dimensions: goods countries origin, good name, selling price, item id, the quantity on hand from each item.

And the following expressions (the expressions are divided to three Drop-down lists as following): (gross sales, sales amount, sales return), (quantity sold, quantity of sales returns),(margin percentage, margin) and promotion. As shown in following figure:

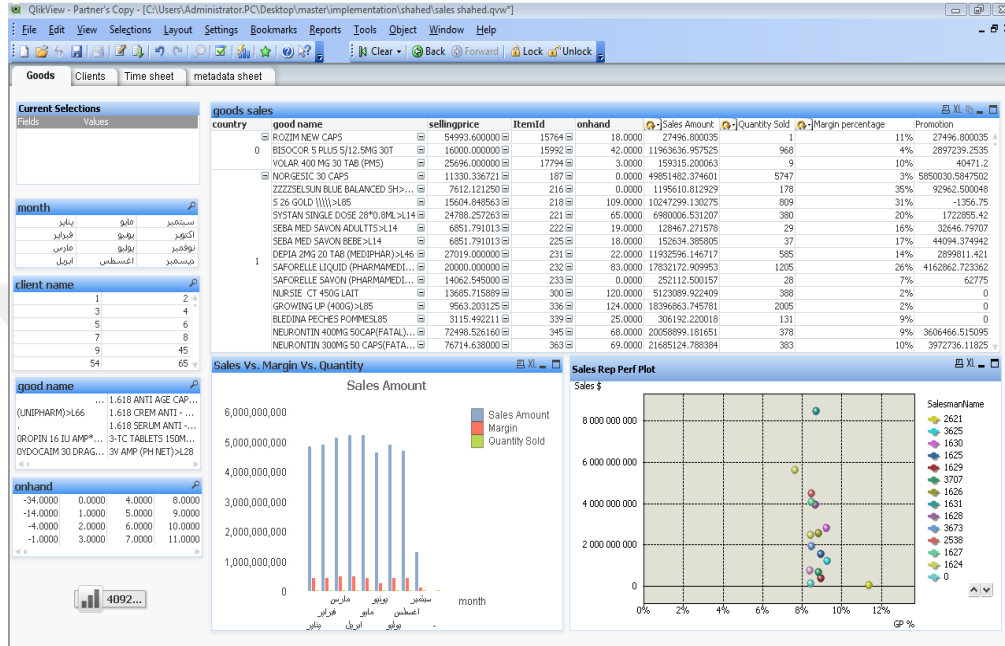


Figure 3.16: Goods sheet in the sales analysis application

When the sales manager select any good, he can know what the selling price, quantity available, sales amount, Sales returns amount, net sales amount, Quantities sold, quantities of sales returns, The margin amount, the margin percentage, the promotion; which related with this good.

Also when the sales manager select any country, he can know all the previous information aggregated for all the goods according to this country.

We built the previous table by create a pivot table (to aggregate the goods according countries), then we select the dimensions which we need to measure, and define the expressions which we need to calculate the dimensions data according them. As shown in following figure:

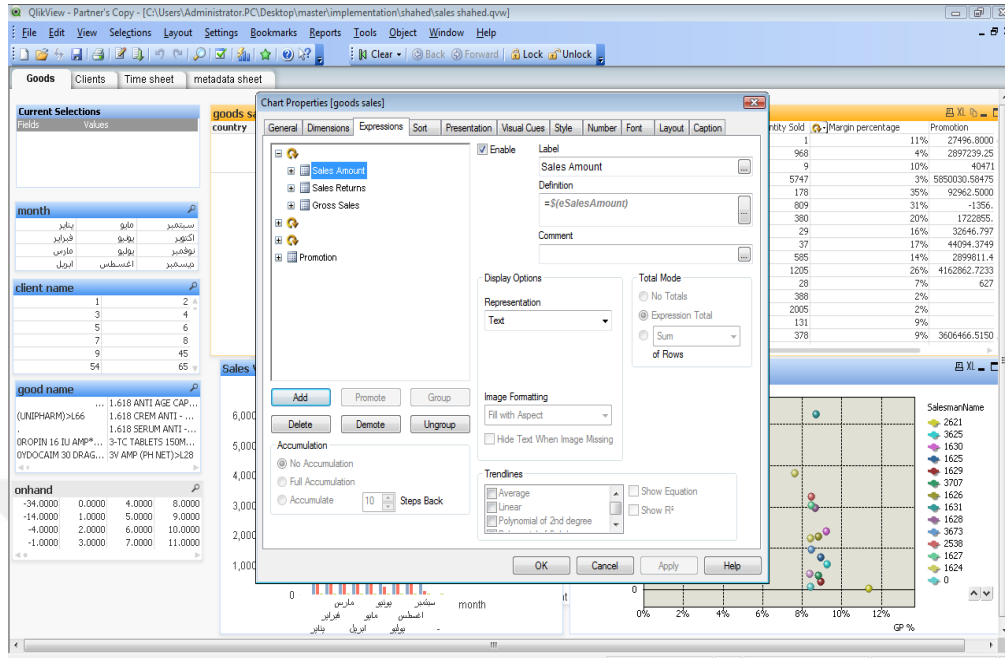


Figure 3.17: Goods sales table building

Clients sheet:

In this sheet the analysis applications which related with the clients have been built, for example we built a pivot table similar to the previous one has the following dimensions:

Respect (which represent the client class), client ID, client name, the sales men names who sold the goods to the clients, and the measures which mentioned in the good sheet table.

Also in the same sheet we have built a charts to visualize the data and to enable the sales manager to compare between the information, as shown in the following figure:

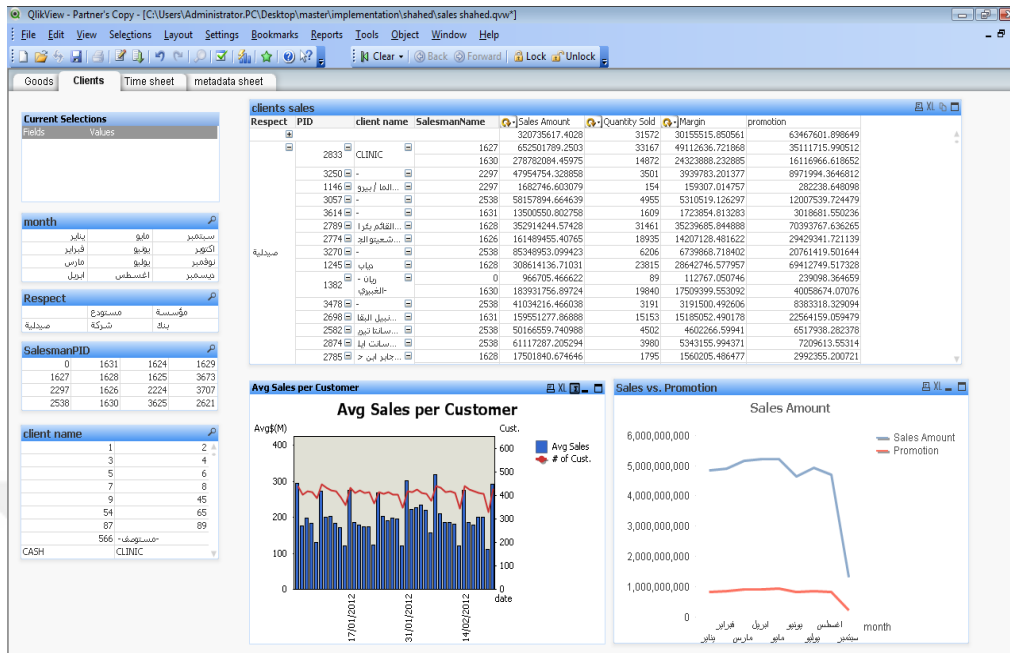


Figure 3.18: Clients sheet in the sales analysis application

For example the AVG sales per customer is a Meko chart has a date dimension, and the average sales, clients count as expressions; the following figure shows how it built.

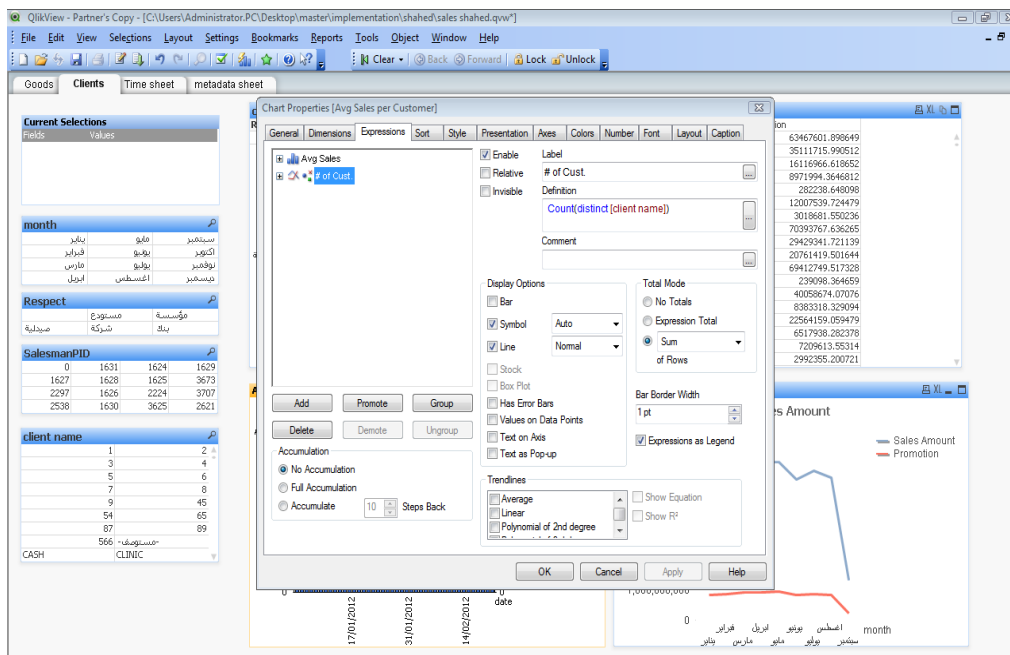


Figure 3.19. AVG sales per customer chart building

Time sheet:

This sheet contains some charts visualize the data, makes some comparison according months and years, and makes prediction for the coming months, as shown in the following figure :

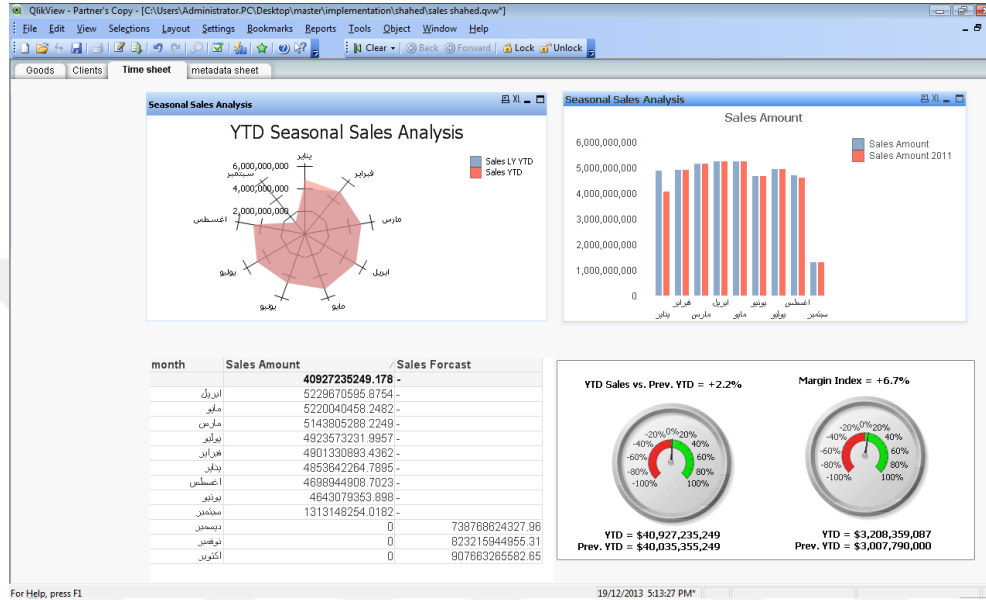


Figure 3.20: Time sheet in the sales analysis application

3.3 THIRD PHASE: LEARNING:

This phase contains two sessions:

3.3.1 Evaluation:

In this phase we get the feedback from the users about the data mart applications and if they have new requirements we can configure the solution to fit them.

In this project after the collaboration phase, and the business users saw the analysis applications, they asked for new requirements, we turned again to the collaboration phase and built them.

This discussion with the solution's users reduces the optimal number of cycles needed to complete the project. Where the number of cycles in this project were two.

3.3.2 Deployment:

In this final phase we deliver to each analysis team the accepted analysis solution.

4. COMPARSION WITH OTHER WORKS

4.1 TIME TO GET VALUE:

The company wants a solution which has a short implementation time or short payback time.

According to the survey which made by IDC QlikView's average implementation time by using the traditional development methodology was 82 days, or 12 weeks [29], but the implementation time which we used to completely build the solution by using the Adapted ASD-BI and with QlikView platform was 60 days. Which mean using the suggested method in building the Self-Service BI solutions reduced the building time in average 12 days.

4.2 PERFORMANCE IN THE IMPLEMENTATION:

the Business intelligence solution which used previously in the company are depend on the static reports designed in PHP programming language, where they are queries on the database designed to answer specific questions such (the sales amount in month, quarter or year), in each time the report asked by the managers the user interface connected with the database on the hard to retrieve the data.

With the new solution, Qlikview in the loading process load all the data from the database and saved it in the memory in compressed way by using special data compression algorithm . For example in our solution the size of the used database in the hard is 3,396,697 KB but when we load the extracted data on Qlikview the size is 62,800 KB. So the needed time to retrieve the requested report are reduced in noticed way. Where it Compress data from 20% to 90 % of its size, depending on the nature of the data being compressed [31].

QlikView stored the data set in-memory and that every data point associated with another data points across the data set, it doesn't use queries, the data set is ready to answer any request by the end user whatever the question and without any customization, after the data is retrieved, its aggregated on the fly by the aggregation engine then presented to the end user in real time. [30]

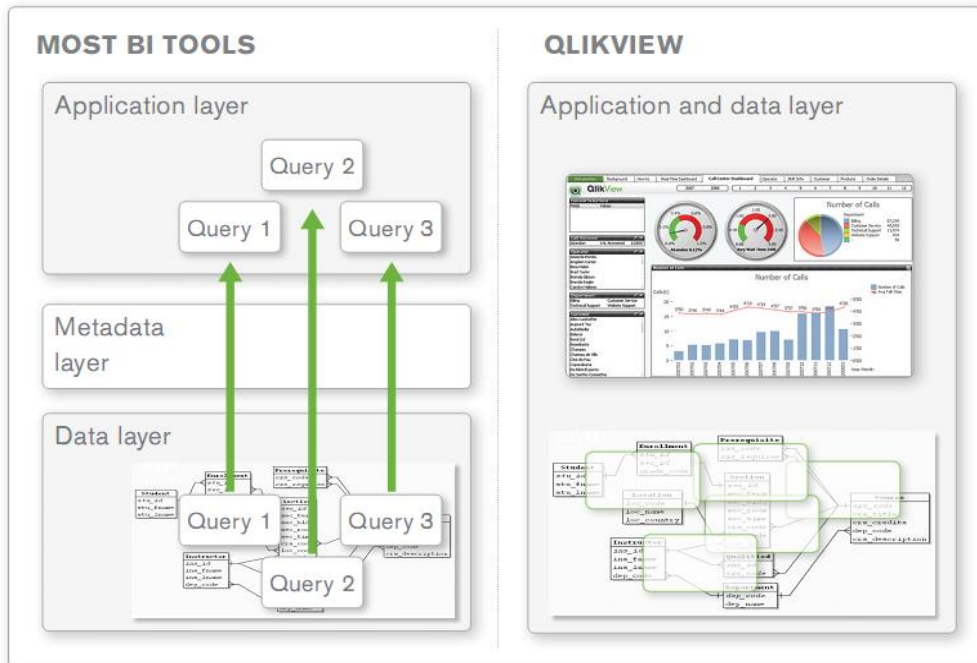


Figure 4.1: Retrieving information in QlikView [30]

5. CONCLUSION

This research discussed using the Agile methodology in developing the Self-Service BI solutions, and it's presented an important theoretical and practical contributions. In the theoretical part we explored the concept of Self Service BI. Also we explored the Agile method as a development methodology to build the Self Service BI solution, and all its methods, we compared between Agile methodologies based on set of criteria, and we suggested the Adaptive Software Development (ASD) to use. Then we adapt the methodology which developed by Alnoukry to be appropriate to develop the Self-Service BI solutions.

In the practical part we success in building a Self Service BI solution by using QlikView platform and the adapted ASD-BI method; this solution has two sides: technical side which is using Self-Service BI platform and Agile methodology; and the management side which is about delivering the information to the manager in the middle level and the knowledge workers to make the right decisions in the right time.

The future research could be:

Extend the solution to meet the increasing in the company activities which attach with increasing in the stored data, and increasing in the need for more information.

Looking for a lower cost solution:

With the accelerated development in the BI domain, maybe the company in the future will have many choice to choose between them as a lower cost alternative to the exists solution; Such as: open source tools (reporting, OLAP, databases, ETL, and predictive analytics) and cloud-based BI services...

The adapted ASD-BI model can be tested on another domains such the manufacturing or engineering fields where the data sources and departments different, the business needs rapidly changed, and the requested analysis applications are totally different from the suggested in the wholesale company mentioned above.

Also as a future work the model can be generalize to be a general analysis model for the wholesale companies which handle all the expected analysis requirements for the mangers in all levels.

REFERENCES

- [1] Hagerty John, L. Sallam Rita, Richardson James, 2012, "*Magic Quadrant for Business Intelligence Platforms Report* ", Gartner report, ID: G00225500.
- [2] Rouhani Saeed, Asgari Sara, Mirhosseini Seyed Vahid, 2012, "*Review Study: Business Intelligence Concepts and Approaches*" Whitepaper, MehrAlborz University, American Journal of Scientific Research, ISSN 1450-223X Issue 50 (2012), pp. 62-75.
- [3] Evelson Boris, June, 12, 2012, "*The Forrester Wave: Self-Service Business Intelligence Platforms, Q2 2012*" Report, Forrester Research Inc, 71902.
- [4] Cohen David, Lindvall Mikael, Cost Patricia, 2003. "*An Introduction to Agile Methods*" Book, university of Maryland, Elsevier Inc, ISSN: 0065-2458/DOI10.1016/S0065-2458(03)62001-2
- [5] B. Shelly Gary , J. Rosenblatt Harry, J. Cashman Thomas, March 16, 2005, "*Systems Analysis and Design*" book, Publisher: Course Technology; ISBN-10: 0619255102.
- [6] Stodder David, First Quarter 2013, "Achieving GreATER Agility with Business intelligence" report, TDWI research, <http://tdwi.org/research/2013/01/tdwi-best-practices-report-achieving-greater-agility-with-business-intelligence.aspx>
- [7] Microsoft IT Showcase . <https://msdn.microsoft.com/en-us/library/mt433042.aspx>. August 2015
- [8] <http://www.ownerteamconsult.com/what-capital-project-teams-need-to-know-about-information-technology-part-2/>
- [9] Gartner, *Magic Quadrant for Analytics and Business Intelligence Platforms*, Cindi Howson, Alys Woodward, Carlie J. Idoine, James Laurence Richardson, Joao Tapadinhas, Rita L. Sallam, 26 February 2018
- [10] B. Shelly Gary , J. Rosenblatt Harry, J. Cashman Thomas, March 16, 2005, "*Systems Analysis and Design*" book, Publisher: Course Technology; ISBN-10: 0619255102.
- [11] El Sheikh Asim Abdel Rahman, Alnoukari Mouhib, 2012 "*Business Intelligence and Agile Methodologies for Knowledge-Based Organizations*" Book, Published by: Business Science Reference, HD38.7.B8715 2012
- [12] Reinschmidt Joerg, Françoise Allison, 2000, "*Business Intelligence Certification Guide*" Book, International Technical Support Organization, SG24-5747-00.

- [13] L.Sallam Rita, Richardson James, Hagerty John, Hostmann Bill, January 2011, " *Magic Quardan for Business Intelligence platforms*" research, Gartner RAS core research, RA3 05032011.
- [14] CITO Research, May 2011, " From Reports to Discovery": A New Era in Business Intelligence" research, sponsored by QlikView, QlikView.com.
- [15] Imhoff Claudia, White Colin, Third Quarter 2011," *Empowering Users to Generate Insights Self-Service Business intelligence*" Report, TDWI research, TDWI.org.
- [16] Shaw Rochelle, January 2011 Issue, "what is Business Intelligence?" research, database trends and applications website,<http://www.dbta.com/Articles/Editorial/Trends-and-Applications/What-is-Business-Intelligence-73502.aspx>
- [17] <https://ometisuk.wordpress.com/2015/09/16/the-different-levels-of-self-service-what-is-self-service-bi-5-levels-of-qlik-sense/>
- [18] Howard Dresner, May, 2012, "*Wisdom of Crowds TM Business Intelligence Market Study*" report, Dresner Advisory Services, LLC, 2012 edition.
- [19] Sallam, Rita L., et al. "Magic quadrant for business intelligence and analytics platforms." Gartner RAS core research notes. Gartner, Stamford, CT (2014).
- [20] Rönnow, Daniel. "*A structural framework of an agile development program of self-service business intelligence.*" (2014).
- [21] Moss, Larissa T., and Shaku Atre. *Business intelligence roadmap: the complete project lifecycle for decision-support applications*. Addison-Wesley Professional, 2003
- [22] COLLIER KEN, July 2011, " *AGILE analytics a value-driven approach to Business Intelligence and data warehousing* " book, Pearson Education, Inc., ISBN-13: 978-0-321-50481-4.
- [23] James E. Powell , Q&A: Agile and Self-Service BI , 1.21.2014, <https://tdwi.org/Articles/2014/01/21/Agile-and-Self-Service-BI.aspx?Page=1>
- [24] Cantor Murray, August, 1998, " *Object-Oriented Project Management with UML*" book, Publisher: Wiley, ISBN-10: 0471253030.
- [25] Cohen David, Lindvall Mikael, Costa Patricia, 2004, " *An introduction to agile methods*" report, computer scines bibliography joutnals: *Advances in Computers*, Volume 62, isbn 0-12-012162-x.

- [26] Swoyer Stephen, May 5, 2010, " Making the Case for Agile Business Intelligence" article, TDWI research, <http://tdwi.org/Articles/2010/05/05/Agile-BI.aspx?Page=1>
- [27] Stojanovic Zoran, Dahanayake Ajantha, Sol Henk , 2003, " *Modeling and Architectural Design in Agile Development Methodologies* " whitepaper, Faculty of Technology, Policy and Management Delft University of Technology, <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.108.3616>
- [28] Breslin Mary, 2004," *Data Warehousing Battle of the Giants: Comparing the Basics of the*" article, Business Intelligence Journal, vol. 9, no 1, 2004.
- [29] Woodward Alys, October 2009," *Time to Value and ROI from BI: The QlikView Customer Experience* " white paper, IDC research, # IDCWP16R3.
- [30] Monash Curt, june 2010, " The underlying technology of QlikView " blog, a Monash research corporation, <http://www.dbms2.com/2010/06/12/the-underlying-technology-of-qlikview/>
- [31] Driver Erica, Sep 6, 2011, "*The Secret Sauce*" Topic, community.qlikview.com, <http://community.qlikview.com/blogs/theqlikviewblog/tags/in-memory>
- [32] Salman, I., Ucan, O. N., Bayat, O., & Shaker, K. (2018). *Impact of Metaheuristic Iteration on Artificial Neural Network Structure in Medical Data*. Processes, 6(5), 57.