VIETNAM NATIONAL UNIVERSITY HO CHI MINH CITY

UNIVERSITY OF ECONOMICS AND LAW

FACULTY OF INFORMATION SYSTEMS

FINAL PROJECT

PRODUCTION SCHEDULE & INVENTORY CONTROL SUPPORTING SYSTEM

Subject: Business Intelligence & Decision Supporting System

Instructor: MSc Nguyen Van Ho

MSc Nguyen Van Tuyen

Tutors: BSc Nguyen Anh Nhat

Group: OBD

Members of group:

Bùi Phương Nga	K194060857
Nguyễn Hoàng Tuyết Nhi	K194060863
Lê Thị Diễm Quỳnh	K194060869
Nguyễn Đỗ Thanh Thùy	K194060873
Nguyễn Thu Vân	K194060880

Ho Chi Minh city, 26th May 2022

LIST OF GROUP 03 – K19406C MEMBERS AND CONTRIBUTION

No.	**Student Name**	**Student ID**	**Contribution**
1	Bùi Phương Nga	K194060857	100%
2	Nguyễn Hoàng Tuyết Nhi	K194060863	100%
3	Lê Thị Diễm Quỳnh	K194060869	100%
4	Nguyễn Đỗ Thanh Thùy	K194060873	100%
5	Nguyễn Thu Vân	K194060880	100%

Table of Contents

ACKNOWLEDGEMENTS 6
CHAPTER 1: INTRODUCTION 7
1.1. Introduction about company 7
1.2. Reason for the Project 7
1.3. Business case for the project 8
1.4. Objectives of the project 10
1.4.1. General Objectives 10
1.4.2. Specific Objectives and Research Questions 10
1.5. Object and Scope of the project 11
1.5.1. In scope 11
1.5.2. Out of scope 11
1.6. Value and desired outcome of the project 11
1.7. Structure of project 11
CHAPTER 2: THEORETICAL BASIS 13
2.1. Overview about BI 13
2.1.1. What is BI? 13
2.1.2. BI Architecture 13
2.2. ETL Process 13
2.2.1. What is ETL? 13
2.3. Data warehouse and Data mart 14
2.3.2 Who needs Data warehouse and Data mart? 14
2.3.3 Advantages and disadvantages of Data warehouse 15
2.3.4 Snowflake and Star schemas 15
2.4 KPIs 15
2.4.1 KPIs Definition 15
2.4.2 The advantages and disadvantages of KPIs 15
2.4.3 Categories of Production KPIs 16
2.5 MDX language for analyzing multidimensional data and OLAP 18
2.5.1 What is MDX language? 18
2.5.2 OLAP technique 18
2.5.3 MDX method and structure 20

CHAPTER 3: REQUIREMENTS ANALYTICS AND INTRODUCTION TO BI SOLUTION 21
3.1. Business processes of Production 21
3.1.1 Production department 21
3.1.2 The purpose of Production 21
3.1.3 Production process 21
3.2. Data source and challenges 22
3.3. Business Requirements Analysis 23
3.4. IT requirements Analysis (IT & Infrastructure) 24
3.5 Comparative Analysis of BI and Data Visualization Tools 24
3.5.1 Surveying and evaluation 24
3.5.2 Proposing BI solution for the project 25
CHAPTER 4: BUILDING DATA WAREHOUSE AND INTEGRATING DATA 27
Introduction 27
4.1. Designing Data Warehouse 28
4.1.1. Bus Matrix 28
4.1.2 Master Data 28
4.1.3 Transaction Data 28
4.1.4. Fact and dimension tables 30
4.2. ETL processes 37
4.2.1. Dimension Table's ETL Process 37
4.2.2. Fact Table's ETL Process 42
CHAPTER 5: DATA ANALYTICS 48
5.1. Data analytics with SSAS technology 48
5.1.3. Building KPIs system with SSAS 55
CHAPTER 6: VISUALIZATION AND FORECASTING OR PREDICTIVE MODEL 62
6.1. Data analysis with Power BI 62
6.1.1 Production 62
6.2.2 Inventory 76
CHAPTER 7: CONCLUSION AND FUTURE WORKS 87
7.1. Results 87

7.2. Limitations ... 87
7.3. Future works ... 87
REFERENCES ... 88

ACKNOWLEDGEMENTS

First and foremost, we would like to express my sincere gratitude and appreciation to my lecturer, PhD. Nguyen Van Ho for providing an overview and detailed knowledge of Business Intelligence and Decision Support System. His patience, enthusiasm, and immense knowledge has inspired us a lot. He helped us every time we had any problems. He always patiently explained and recommended solutions to resolve problems so that we could find suitable ways and have gained new knowledge through his guidance. We are really grateful for what he has offered us.

Second, we would like to thank Mr. Nguyen Van Tuyen who supported us in the practical part, helped my team use the tools fluently, and gained the necessary skills in the Data Analyze career.

Finally, we would like to thank Mr. Nguyen Anh Nhat - assistant professor of the subject, who has supported us not only in knowledge but also in experience about data visualization and career as well as encouraged us in spirit to successfully complete this project.

Sincere thanks.

CHAPTER 1: INTRODUCTION

1.1. Introduction about company

This research uses a fictitious company provided by Microsoft namely Adventure Works Bicycles, Inc. Adventure Works is a fictional bicycle wholesaler. The company has 97 different brands of bikes that grouped into three categories: mountain bikes, road bikes, and touring bikes. Moreover, Adventure Works also manufacture some of its own components. Several components, accessories and clothing are purchased from outside from vendors.

Adventure Works is not only selling bicycles, but it also provides accessories, clothing, and components. The accessories available such as bottles, bike racks, brakes, etc. The available clothing such as caps, gloves, jersey, etc. For the components, Adventure Works sells brakes, chains, derailleurs, etc. Many of those things are made by vendors, so Adventure Works stand as a reseller. Adventure Works serve the customer globally, including Australia, Canada, France, and Germany, United Kingdom, and United States. There are 2 business models in Adventure Works which are retail stores that sell bikes, and internet sales that serve individual customers. Usually Adventure Works sells in bulk to retail stores, which acts as resellers for its products.

To run the business activities, Adventure Works has a total of 290 employees that included in some functions such as sales, production, purchasing, engineering, finance, information services, marketing, shipping and receiving, and R&D. The customers of Adventure Works include over 700 stores and over 19000 individuals worldwide and its vendors are quantified around 100 vendors companies that supply raw materials, accessories, clothing, and components. Even though Adventure Works is fictional, it is designed as a realistic case, the same as a real company in industry. Adventure Works provides a database and data warehouse that covers business processes from sales, material management, production, finance, and human capital management. Therefore, the researcher uses this fictional company as the case study to develop Self-service BI system.

1.2. Reason for the Project

The main issues facing Adventure Works were the availability of an effective reporting system. Due to the lack of real-time and accurate reports, the organization struggled to track their performance and did not know what areas of the process need to be improved.

Table below is the Adventure Works's SWOT Analysis.

Strengths	Weaknesses
Strong distribution network (over 700 stores and over 19000 individuals	Not very good at product demand forecasting, often struggle to identify

worldwide)	how much inventory is needed
Strong relationships with many different vendors (about 100 vendors companies)	Cost in inventory management is quite high leading to the rise of products' price
A wide selection of goods (bikes, accessories, clothing,...)	
Opportunities	Threats
The ability to spread fixed costs over a larger area as sales increase, increasing profit margins.	Competitive rivalry in the bicycle industry is high. Competitors like Giant Bicycle, Trek Bicycle, Scott, Konacan,...can offer similar products quickly
The gas price is rising quickly, so there would be a better chance for people to purchase bicycles	

SWOT analysis of AdventureWorks

In order to overcome weaknesses and stay competitive in today's market, they need the capability to make better decisions, especially in production planning and inventory management. After evaluating, Adventure Works has decided to implement BI solutions to help them solve problems. That is also the reason for doing this project.

1.3. Business case for the project

Project Business Case			
Project Name	Production schedule & inventory control supporting system		
Project Sponsor	Van-Ho Nguyen	Project Manager	Thu-Van Nguyen
Date of Project Approval	March 1st, 2022	Last Revision Date	March 1st, 2022

Contribution to Business Strategy	The company strategy is to achieve maximum efficiency in manufacturing goods, but the current situation does not reflect this. And the new BI system will ensure to support manufacturers make better decisions based on historical data.
Options Considered	Options considered included: 1. Finding a new way to make schedule for producing goods 2. Inventory control support.
Benefits	1. Creating interactive dashboards 2. Helping managers monitor production and inventory status 3. Supporting executive managers' decisions on production adjustions
Time Scales	The system will take 2 months to implement
Cost	1. Server hardware: $50,000 2. Software licensing (SQL Server & SharePoint): $13,000 3. External resource labor: $100,000 4. Internal resource reassigned labor: $300,000 5. Training & other misc costs: $50,000 6. Project Manager: $100,000 7. Total cost estimate: $613,000
Expected Return on Investment	Year 1: 0 Year 2: 100,000 Year 3: 200,000 Year 4: 300,000 Year 5: 500,000

Risks	1. Team members have never experienced any BI project. 2. Skills are limited but can be improved. 3. Project lacks resources such as human capital, etc.

Business case of project

1.4. Objectives of the project

1.4.1. General Objectives

There are 5 primary objectives that we expected to achieve after doing the project:

- Saving expenses on manufacturing management.
- Improve decision-making.
- Forecasting, predicting, and meeting market demand.
- Automatically reports.
- Preventing out of stock or inventory redundancy.

1.4.2. Specific Objectives and Research Questions

More specifically, the organization wants to focus on answering the following questions:

Production:

- What is the current production status of the company? How to help company improve production planning?
    - How many items are ordered, stocked, completed, and scrapped?
    - Is the quantity of these items increasing or decreasing over time?
    - Is there any correlation between completed quantity and scrapped quantity, or between the ordered and stocked quantity?
    - What is the total or the rate of scrapped products by quarter, year?
    -
    - How much waste or cost caused by scraping?
    - What is the difference between the actual & planned production time as well as the actual & planned cost? How large are these differences?

Inventory:

- What is the current inventory status of the company? How to utilize the capacity of the warehouse efficiently?
    - What is the total of inventory value by location and by product category?
    - How many items are in stock, running-low or out of stock?
    - How long does each item usually stay in stock?
    - Which products are often below the safety level?
    - Which location in the warehouse was used the most and when?

1.5. Object and Scope of the project

The main objective of this project is to build a BI solution to boost operation of the production department with a production schedule & inventory control supporting system. Besides, the project scopes are defined as follows. The project is planned to be carried out from March to May 2022 while the time of data is between 2011 to 2014. The space scope is a BI solution for production schedule and stock control of finished goods.

1.5.1. In scope

Within the scope, the expected result includes a data warehouse, analytical model, and intuitive dashboard, therefore, there are detailed scopes to be concerned. The data warehouse is built with Kimball's approach and galaxy. Researched processes involve planning and scheduling production, and stock control, which mainly focuses on finished goods. Also, useful information is presented with Power BI dashboards.

1.5.2. Out of scope

Along with the aforementioned scopes, scopes that are out of reach are listed below:

- The proposed system supports other departments than production.
- The studied process involves external movement of inventory, usage of products and release of new products.
- The data on raw materials are scrutinized.

1.6. Value and desired outcome of the project

After analyzing the company's objective and its available data, we hope we can achieve these things after completing this project

- Understanding clearly the requirements of the company.
- Designing and building a data warehouse meet the company's expectations.
- Creating interactive dashboards to present information allowing key decision-makers to know what's going on at a simple glance and monitor the overall production process.
- Analyzing data and finding some insights to recommend for managers what needs to optimize and improve production efficiency to drive continuous improvement.

1.7. Structure of project

Chapter 1: Introduction

In this chapter, we explore information about the company Adventure Works and define the general view of the project by using some documents or templates.

Chapter 2: Theoretical basis

We will introduce and explain some business intelligence methodology and concepts that will be applied in this project.

Chapter 3: Requirements analytics and introduction to BI solution

We determine requirements and prepare the BI solution.

Chapter 4: Building data warehouse and integrating data

In this chapter, we will analyze data and design a data warehouse. Then, we perform the ETL process.

Chapter 5: Data analytics

Based on the data model that is built in the previous step, we use some tools to analyze data.

Chapter 6: Visualization

After analyzing the data, we build graphs, charts for data visualization.

Chapter 7: Conclusion and future works

Finally, we conclude the project and identify direct further development.

CHAPTER 2: THEORETICAL BASIS

2.1. Overview about BI

2.1.1. What is BI?

BI also known as business intelligence is a type of technology that supports organizations in understanding the past, allowing them to make better decisions, take action, and forecast the future. Where BI refers to skills, procedures, technology, or apps that help people make better decisions. Business intelligence (BI) combines business analytics, data mining, data direction, tools and infrastructure, and deployment to assist businesses in making data-driven decisions.

2.1.2. BI Architecture

A BI Architecture defines the innovation concepts and data the board and examination procedures that support an organization's BI initiatives, as well as the specific stages and gadgets that will be delivered. It serves as an innovation diagram for acquiring, sorting out, and supervising BI data, and then making the data available for analysis, information representation, and detailing. A good BI Architecture also includes methods for monitoring how the technology components are being used.

2.1.3 Advantage of BI in enterprises

BI helps businesses master content accurately, achieve good results, thereby supporting analysis, data mining, forecasting of service price trends, customer behavior, detection of users, etc. have the ability to buy goods to set out appropriate sales plans to increase the competitiveness of the business.

Some practical benefits can be mentioned such as:

- Helping companies use information efficiently and accurately to adapt to the constantly changing and fiercely competitive business environment.
- Maximum support for senior personnel in making quick, timely and effective sales decisions
- Create competitive advantages, increase opportunities to find and control sales opportunities
- Supporting internal users in evaluating, improving, and optimizing capabilities as well as operational processes of the organization

2.2. ETL Process

2.2.1. What is ETL?

ETL is a type of data integration that refers to the three steps (extract, transform, load) used to blend data from multiple sources. It's frequently used to construct a data warehouse. Data is collected (extracted) from a source system, converted (transformed) into an analyze able format, and placed (loaded) into a data warehouse or other system throughout this process. ELT (extract, load, transform) is a different but related method for pushing processing to the database for better speed.

2.2.2. Why do we need ETL?

ETL tools break down data silos and make it easy for data scientists to access and analyze data, and turn it into business intelligence. In short, ETL tools are the first essential step in the data warehousing process that even KPI lets you make more informed decisions in less time.

2.2.3. ETL Process

Data was taken from an OLTP database, modified to fit the Data warehouse schema, and fed into the Data warehouse database during the ETL process. Data from non-OLTP systems, such as text files, legacy systems, and spreadsheets, was often included into data warehouses. ETL was frequently a sophisticated blend of procedure and technology that ate up a large amount of the data warehouse development work and necessitated the expertise of business analysts, database designers, and application developers. The ETL process was not a one-time event. The data warehouse would be updated on a regular basis when data sources changed. Also, when the company changed, the data warehouse system needed to alter – in order to preserve its value as a decision-making tool – and as a result, the ETL changed and matured as well. The ETL methods were created with flexibility in mind. The success of a data warehouse project required a reliable, well-designed, and documented ETL infrastructure. An ETL system was made up of three processes that were characterized as follows:

1. Extract - the process of reading data from a specified source database and extracting a desired subset of data.
2. Transform - the process of converting the extracted/ acquired data from its previous form into the form it needed to be in so that it could be placed into another database. Transformation occurred by using rules or lookup tables or by combining with other data.
3. Load - the process of writing the data into the target database.

2.3. Data warehouse and Data mart

2.3.1. What are Data warehouse and Data mart?

A data warehouse is a data management system that helps an organization's business intelligence and analytics. Large volumes of data, particularly historical data, are frequently stored in data warehouses. A data warehouse's data is typically derived from a variety of sources, including application log files and transaction applications. A data warehouse is a storage facility for structured data with a well-defined function.

A data mart is a basic type of data warehouse that focuses on a single subject or business lines, such as sales, finance, or marketing. Data marts use fewer sources than data warehouses because of their emphasis. Internal operating systems, a central data warehouse, and external data can all be used as data mart sources.

2.3.2 Who needs Data warehouse and Data mart?

A data warehouse takes a long time to set up, especially in a large company, because it collects data from a variety of internal and external sources. As a result, data warehouses are required for all types of users, including: Those who make decisions based on a large amount of data, or people who want to access info using simple technologies also use it. It's also necessary for individuals who want to take a methodical approach to making decisions.

By having a comparably small subject of data, data marts are meant to satisfy the demands of specialized groups. While a data mart can still have millions of records, its goal is to give the most relevant data to business users in the quickest amount of time. A data mart powers corporate intelligence and analytics, which help departments make better decisions. Teams can employ targeted data insights to achieve their objectives. The enterprise benefits from quicker business processes and better productivity as teams uncover and extract useful data in less time.

2.3.4 Snowflake, Star and Galaxy schema

Star Schema in data warehouse, in which the center of the star can have one fact table and a number of associated dimension tables. It is known as star schema as its structure resembles a star. The Star Schema data model is the simplest type of Data Warehouse schema. It is also known as Star Join Schema and is optimized for querying large data sets.

Snowflake Schema in data warehouse is a logical arrangement of tables in a multidimensional database such that the ER diagram resembles a snowflake shape. A Snowflake Schema is an extension of a Star Schema, and it adds additional dimensions. The dimension tables are normalized which splits data into additional tables.

Galaxy Schema contains four facts tables that share dimension tables between them. It is also called Fact Constellation Schema. The schema is viewed as a collection of stars hence the name Galaxy Schema.

2.4 KPIs

2.4.1 KPIs Definition

KPIs give executives the chance to communicate the mission and focus of the organization to investors, team members, and other stakeholders. As KPIs filter through an organization, they must grab employees' attention to make sure that everyone is moving together in the right direction.

2.4.2 The advantages and disadvantages of KPIs

Organizations gain greatly from the use of KPIs. To begin with, KPIs assist in comprehending difficult situations. Many firms may certainly agree on two things: efficiency is crucial, and time is money. KPIs are extremely valuable for evaluating the various intertwined processes and activities that occur in businesses. They can help managers save time and improve efficiency.

The second is used to set goals and track how well they are being met. Setting goals is an essential management technique. Goals are only useful if they can be measured both in terms of execution and progress. This is why KPIs are the preferred instrument for goal-setting and progress tracking: they help clarify the procedures required to achieve specific objectives and are best integrated into wider strategic concerns.

The third goal is to improve communication efficiency. Communication takes time, especially in situations that are complicated and unpredictable. Communication is critical in a VUCA world to ensure rapid adaptability to changing conditions. KPIs can boost efficiency in communication when a sound (team) mental model is in place. In other words, agreeing on KPIs may improve cooperation, morale, and production when groups share understandings and visions of how to accomplish particular tasks.

2.4.3 Categories of Production KPIs

A manufacturing Key Performance Indicator (KPI) or metric is a well defined and quantifiable measure that the manufacturing industry uses to gauge its performance over time.

Below is a list of the top KPIs used by manufacturing leaders to create sustainable manufacturing excellence and follow lean manufacturing processes.

No	Evaluation criteria	Explanation
1	Throughput	This is probably one of the most fundamental KPIs for the manufacturing industry while also arguably one of the most important. The Throughput KPI measures the production capabilities of a machine, line, or plant; also known as how much they can produce over a specified time period.
2	Cycle Time	In the manufacturing industry, cycle time is the average amount of time it takes to produce a product. The cycle time metric can be used to measure the time it takes to manufacture a completed product, each individual component of the final product, or even go as far as to include delivery to the end user. Thus, cycle time can be used to analyze overall efficiency of

		a manufacturing process on the macro scale, as well as determine inefficiencies on a micro scale. Cycle time = Process End Time – Process Start Time
3	Demand Forecasting	This manufacturing metric is used by companies to estimate the amount of raw materials they will require to meet future customer demand. This metric can be a little bit trickier for companies to fully utilize, as it is highly dependent on uncontrollable external factors. The basic formula is as follows: Projected Customer Demand = Raw Materials * Production Rate
4	Inventory Turns	This is a measure of how many times inventory is sold over a specific time period and helps indicate resource effectiveness. Low ratio numbers indicate poor sales and excessive inventory, while high ratio numbers represent strong sales or insufficient inventory. Inventory Turns = Cost of Goods Sold / Avg. Inventory
5	Production Attainment	This production performance metric measures production levels over a specific time period and calculates what percentage of the time a target production level is achieved.
6	Cash to Cash Cycle Time	This is a time-based manufacturing KPI metric. It measures the amount of time it takes from an initial cash outlay for raw materials, inventory, or a manufacturing plant until the company receives cash from its customers for its products. This KPI is typically measured in days. Cash to Cash Cycle Time = Inventory Sale Date – Inventory Purchase Date
7	Avoided Cost	The avoided cost manufacturing metric is an estimate of how much money you saved by spending money. Avoided Cost = Assumed Repair Cost + Production Losses – Preventative Maintenance Cost

2.5 MDX language for analyzing multidimensional data and OLAP

2.5.1 What is MDX language?

MDX stands for Multidimensional Expressions, a syntax for constructing multidimensional objects in OLAP databases as well as searching and manipulating multidimensional data. The purpose of MDX is to make querying data from multidimensional databases more simple and intuitive.

MDX is a query language for OLAP databases, much like SQL is for relational databases. MDX is a SQL extension that allows you to use queries and scripts to access multidimensional data. MDX queries to access data contained in a SQL Server Analysis Server cube by delivering facts related to dimensions.

To identify and extract data from a single cell or a block of cells in a cube, MDX uses a reference system called tuples. Tuples identify individual cells as well as larger groups of cells in the cube using dimensions and members.

2.5.2 OLAP technique

OLAP stands for Online Analytical Processing, and it is a type of software that allows users to analyze data from various databases at the same time. It's a tool that allows analysts to obtain and analyze company data from various perspectives.

2.5.2.1 OLAP process

There are four types of analytical OLAP processes:

- Roll-up: There are two techniques to accomplish the Roll-up operation: reducing dimensions and climbing up hierarchy.
- Drill-down: Drill-down data is broken down into smaller chunks. It's the opposite of the rollup technique. It's possible to accomplish by descending down the hierarchy and adding a dimension.
- Slice and dice:

The slice operation creates a sub-cube by selecting a single dimension from the main OLAP cube. For example, you can perform a slice by highlighting all data for the organization's first fiscal or calendar quarter (time dimension).

The dice operation isolates a sub-cube by selecting several dimensions within the main OLAP cube. For example, you could perform a dice operation by highlighting all data by an organization’s calendar or fiscal quarters (time dimension) and within the U.S. and Canada (location dimension).

- Pivot

The pivot function turns the current cube view to show a different representation of the data, allowing for dynamic multidimensional data displays. The OLAP pivot function is similar to the pivot table functionality in spreadsheet software such as Microsoft Excel, however although pivot tables in Excel may be difficult to use, OLAP pivots are easier to use (less experience is required) and provide better query performance.

2.5.2.2 Advantages and disadvantages of OLAP

Advantages of OLAP

- It may be used to extract or mine data, analyze data, and report on connections between data elements.
- To establish an OLAP multidimensional database, we may utilize ODBC (Open Database Connectivity) to import data from an existing relational database.
- Because trend analysis does not require all transactional data, an OLAP database does not need to be as massive as a data warehouse.

Disadvantages of OLAP

- OLAP requires organizing data into a star or snowflake schema. These schemas are complicated to implement and administer.
- Cannot have large number of dimensions in a single OLAP cube
- Any modification in an OLAP cube needs a full update of the cube. This is a time-consuming process.

2.5.3 MDX method and structure

2.5.3.1 MDX

MDX is used for Analysis Services multidimensional database, in which enables users to define multidimensional objects, query and manipulate multidimensional data in OLAP databases. MDX helps to get access to data from multiple dimensional databases effortlessly and efficiently.

Expressions of MDX are units of syntax, resolving to values or objects. They include functions that return a single value, a set of expressions, ect. Besides, they allow to wrote statements such as CREATE GLOBAL CUBE, ALTER CUBE, DROP MMEBER so as to create, drop and manipulate objects in the OLAP database.

2.5.3.2. MDX structure

We can use Multidimensional Expressions (MDX) to query multidimensional data or create MDX expressions within a cube, it helps to understand multidimensional concepts and terms. An Analysis Services cube consists of measures, dimensions, and dimension attributes.

CHAPTER 3: REQUIREMENTS ANALYTICS AND INTRODUCTION TO BI SOLUTION

3.1. Business processes of Production

3.1.1 Production department

A production department's role is to ensure the process of turning raw materials into products is performed efficiently and accurately. The department performs five functions to assist this process.

- Establish standards in regard to the quality and the quantity of the products being made. Typically, these standards are placed throughout the process, not just at the beginning or end.
- Work with the purchasing department to ensure enough materials are on the production line and to ensure the replacement of any broken equipment.
- Work with the design and technical department to ensure the product is built to the correct specifications and to place any new designs or changes to the product onto the line.
- Collaborates with the works department to ensure there is a proper workforce available to check the quality of the product and make any necessary repairs to any equipment that breaks.

The production department does not work directly with the Sale department. However, it remains in contact with marketing to ensure the products are attractive and desired by customers.

3.1.2 The purpose of Production

The goal of production is converting raw materials and other inputs into finished goods or services. Improving the efficiency of the production or assembly line in between the processes of production, so that it can meet the output targets set by company management and ensure finished products offer consumers the best value and quality.

3.1.3 Production process

The first step in the production is planning. Planning process is a stage in determining several things in this process. Such as what kind of products will be made, how many raw materials are used, how much cost you need, and how much labor you require to carry out production. After that, the company will also design the shape of the goods because companies need information and knowledge about the types of goods to produce, their needs, and the company's ability to carry out production activities to create a good plan.

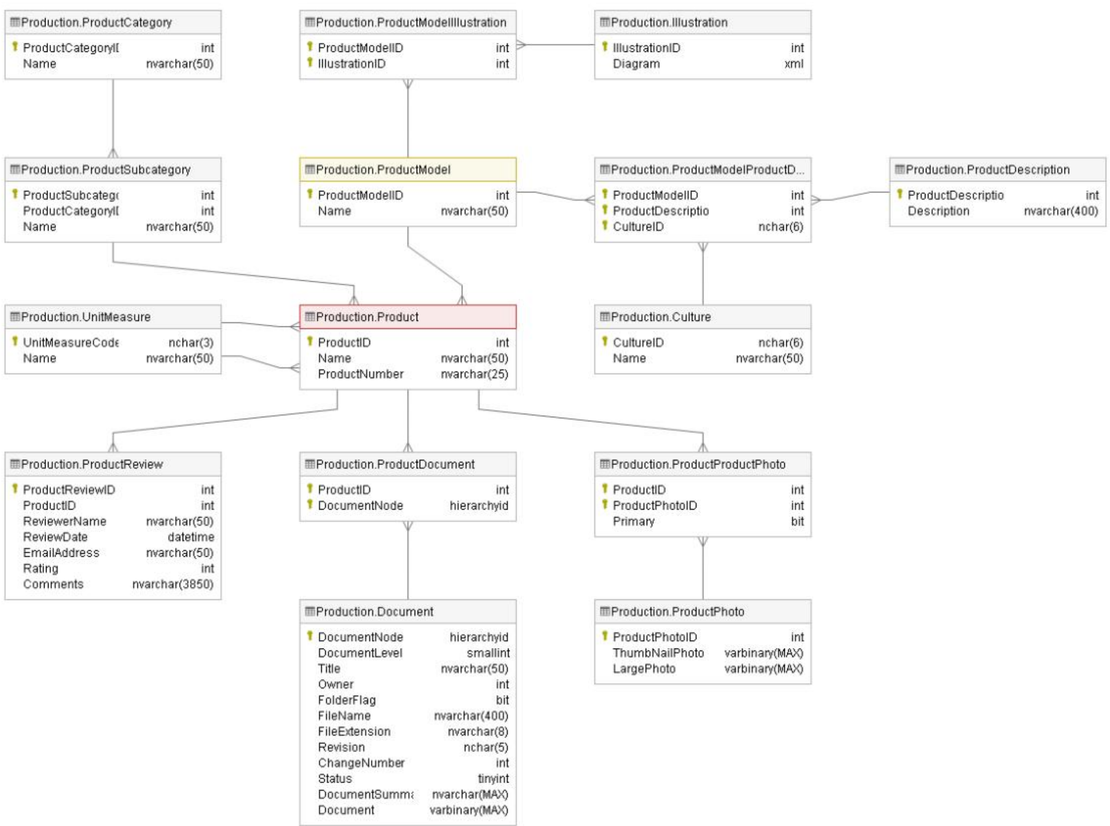
Next, we execute the routing activity. Routing flow is an activity to determine and determine the sequence of activities of this process. The focus is at this stage, from the initial processing of raw materials, forming, polishing, finishing, and quality control to the distribution of manufactured goods.

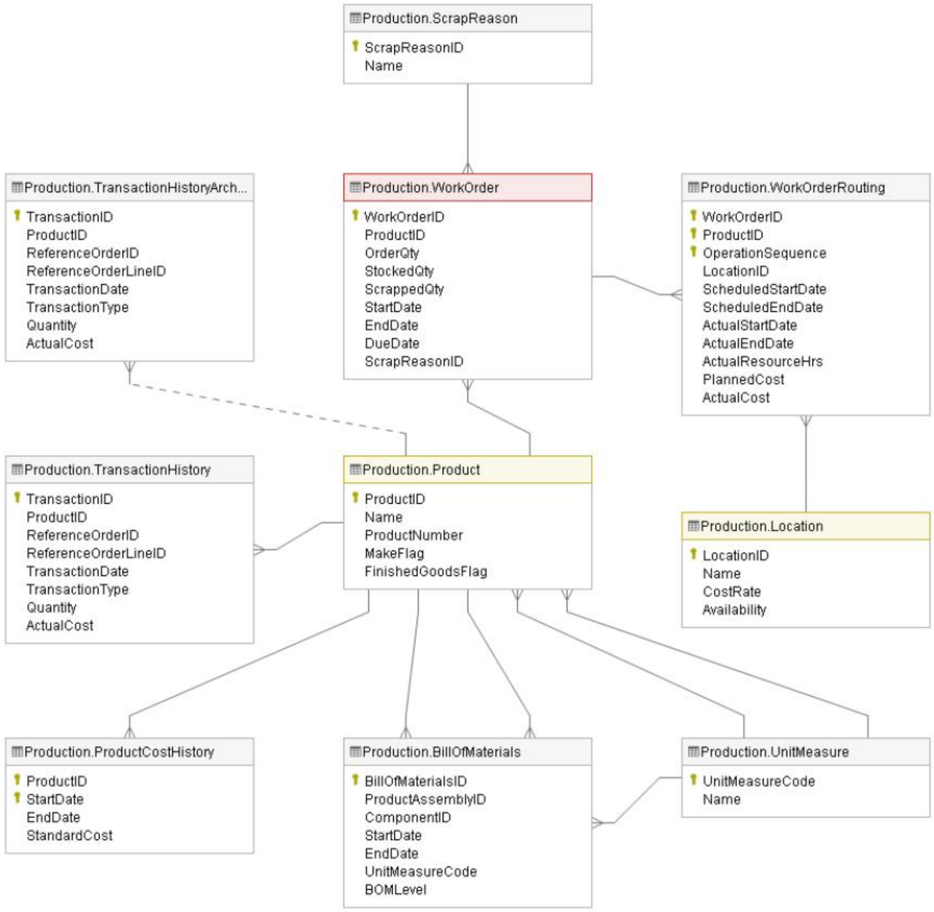
The third step is scheduling. Scheduling is an activity to determine and determine when production must be carried out after the flow is made. In its implementation, scheduling considers the working hours of workers and the length of each production flow.

Finally, finished goods are moved to storage following dispatching directions. Dispatching an activity to determine and establish a process of giving orders to start production after the production schedule is set.

3.2. Data source and challenges

In this project, we used AdventureWorks database whose version is OLTP 2019 to build the BI solution. This version is considered appropriate for the project and online transaction processing workloads. An advantageous point is that this data source provides data that illustrates processes of several departments such as human resources, sales, production and purchasing. Moreover, data is gathered, well-formatted and has clear relationships, which helps reduce a remarkable amount of time working on raw data. According to the business requirements and objectives, production data is the main concern of the research.





Schema of production is very informative with data on products, manufacturing and inventory. However, utilization of the whole data is not optimal for the project so that a number of data fields could be left out and the rest that is helpful will be useful for the further steps.

Apart from the plus points, there are some clear-cut drawbacks. Despite the version of the year 2019, data time is between 2011 and 2014, meaning that the gap of time is too long. The project can be carried out but its accuracy may be undefined. Another limitation is that some data fields are ambiguous in explanation from AdventureWorks Data Dictionary. It would be confusing if the business process is not well understood.

3.3. Business Requirements Analysis

Based on the company's objective, we identify some of the requirements from Adventure Works as follows:

- The company needs to be able to access the data in one single location and in a format that can be easily manipulated.
- The system will be able to display figures and charts that are clear, intuitive, and user-friendly.
- The business wants to have a holistic picture of the production performance and inventory status. So they require the dashboard to display some key metrics and KPI.

- They also likes to see some analysis of trends or patterns of data to help them plan, and control risks, as well as make a more reasonable production schedule.

3.4. IT requirements Analysis (IT & Infrastructure)

Based on the existing IT infrastructure available in the company, Adventure Works intends to use SQL Server Management Studio (SSMS) and SQL Server Data Tools (SSDT) and SSAS (SQL Server Analysis Services) for developing the BI solution. In addition, they also want to use Microsoft Power BI to generate self-servicing reports because Power BI is quite easy to use even for non IT experts so that top and middle level management are able to explore the business data their own on demand BI reports without reliance on the IT department.

3.5 Comparative Analysis of BI and Data Visualization Tools

3.5.1 Surveying and evaluation

As regards BI tools, it is important to choose a suitable tool. Power BI and Tableau are considered for the project.

Power BI is developed by Microsoft with primary focus on business intelligence. It is part of the Microsoft Power Platform and is a collection of software services, apps, and connectors that are able to work together to turn unrelated sources of data into coherent, visually immersive and interactive insights. Data may be input directly from a database, webpage, or structured files such as spreadsheets, CSV, XML, and JSON. Similarly, Tableau provides a powerful and fast growing data visualization tool for business intelligence. It enables simplifying raw data in an easily understandable format.

	Power BI	Tableau
Full-featured Free Version	Yes	Separate tool
Development Environment	Desktop	Desktop
R and Python Supported	Yes	Yes
Dynamic Cross-filtering	Yes	Yes
AI-enabled Analytics	Yes	Yes
Search Analytics with NLP	Yes	No
Data Prep Tools	Yes	Separate tool
Data Modeling Tools	Yes	Separate tool
Preferred Data Model	Star-schema	Flat
Database Independent	Yes	Yes
Built in Row Level Security	Yes	Yes
Mixed Model Types	Yes	No
Third-party Data Model Access	Yes	No
Commenting & Collaboration	Yes	Yes
Embedded Analytics	Yes	Yes
Open-source Custom Visualizations	Yes	No
Native Mobile App	Yes	Yes

Comparison of PowerBI and Tableau

Both of them have advantages and disadvantages themselves. However, Power BI is more appropriate owing to its features that meet the project requirements.

3.5.2 Proposing BI solution for the project

There are a variety of ways for implementing a BI solution. This may vary significantly depending on the company's size and finances. In this case, we propose the project will go through five phases:

- Define the requirements
- Collect the data
- Clean the data
- Analyze the data
- Visualize and share findings

Define the requirements

The team will identify the requirements from stakeholders and validate the data needed to achieve business objectives as well as establish the KPIs required to be measured. This phase is very important because it forms the foundation for the next phases.

Collect the data

Once the requirements are gathered and objectives are set, we then source the data and transform it into a form that will provide the required intelligence to the users. It is important to identify the data available in the source that can meet the project objectives. As mentioned earlier, the source business data is stored in the form of a relational database using the SQL server. Based on the project requirement, it was decided that the data from the source would be transformed and stored into a data warehouse. One of the most commonly used techniques for designing a data warehouse is dimensional modeling (Kimball and Ross, 2013). We will follow Kimball's four steps to dimensional modeling:

- Select the business process.
- Declare the grain.
- Identify the dimensions.
- Identify the facts.

Clean the data

In this step, we will eliminate bad data (redundant, incomplete, or incorrect data) and begin to create high-quality data for the best business intelligence.

Analyze the data

Data that is processed, organized and cleaned would be ready for the analysis. In this phase, we will apply some type of analysis such as

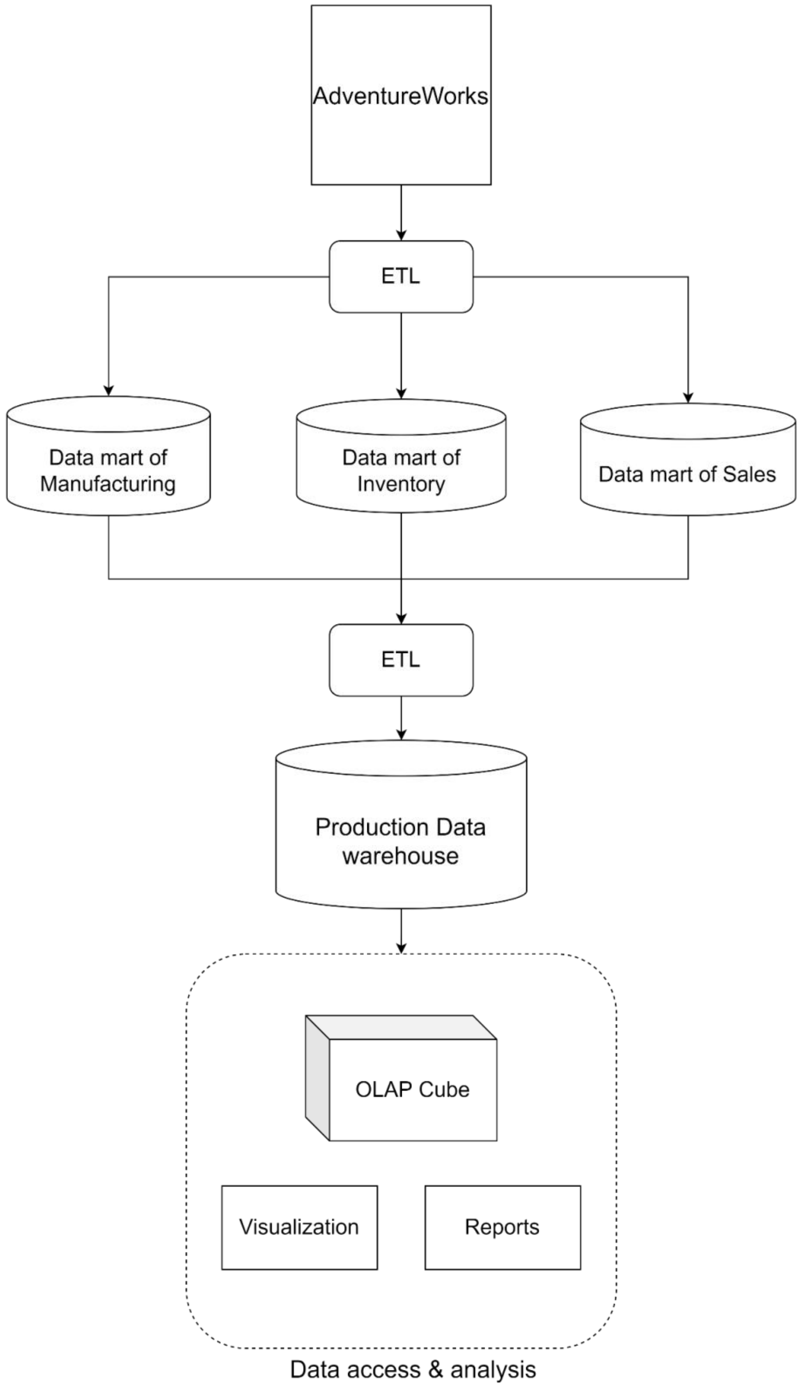
- Descriptive analysis to identify what has already happened
- Diagnostic analytics to understand why something has happened and help the company draw correlations between the issue and factors that might be causing it

Visualize and share findings

The final phase is sharing the insights that we have found in the previous phase. To present it clearly and unambiguously, we use reports, dashboards, and interactive visualizations. Nowadays, there are many tools for data visualization but we will choose Power BI desktop version as mentioned before.

CHAPTER 4: BUILDING DATA WAREHOUSE AND INTEGRATING DATA

Introduction



Research process of the project

As chosen approach was Kimball's, we conducts ETL process to gather the data from AdventureWorks. Then, data marts are designed and built with fact and dimensional tables based on the subject of the research. After that, data fields are extracted, transformed into suitable type. Loading data is carried out with SQL Server Integration Services to combine all data marts into a data warehouse. After gather enough data for the project, analytics are performed with OLAP, visualization and reports in order to help managers make decisions better and more quickly.

4.1. Designing Data Warehouse

4.1.1. Bus Matrix

	Date	Product	Work order	Routing	Scrap	Location
Production	x	x				
Scheduling	x	x	x	x		
Quality checking	x	x			x	
Work planning	x	x	x	x		
Inventory	x	x				x

Bus Matrix table

4.1.2 Master Data

Master data is defined by a few central characteristics that distinguish it from transaction data in particular: It has a high level of statistics, which means that it rarely changes and is mostly valid in the long term. Since master data is used by several divisions of a company, it is highly relevant for all business processes.

Our master data is the data used to build the Dimension tables in the data warehouse, including the tables:

Object	**Description**
Product	Products sold or used in the manufacturing of sold products.
ProductSubCategory	Product subcategories.

ProductCategory	High-level product categorization.
Location	Product inventory and manufacturing locations.
ScrapReason	Manufacturing failure reasons lookup table.

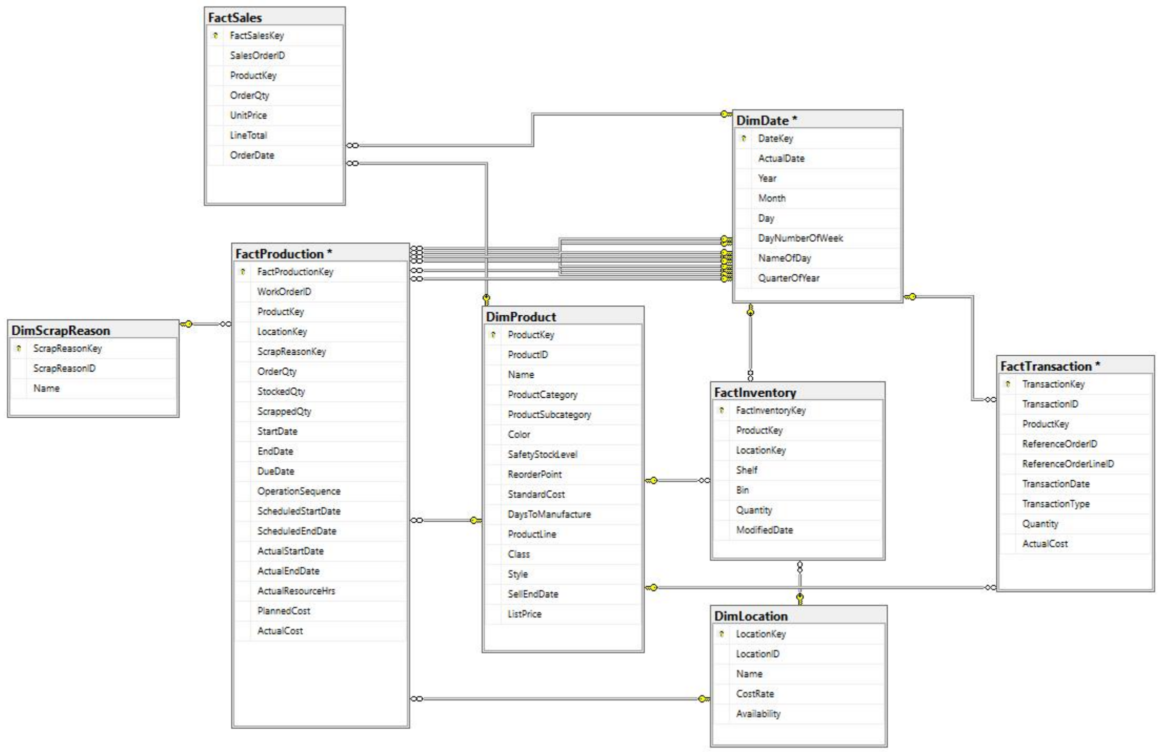
4.1.3 Transaction Data

Transaction data relates to the transactions of the organization and includes data that is captured. It is also used to construct fact tables in the data warehouse, including tables:

Object	**Description**
WorkOrder	Manufacturing work orders. Details of each production order, represented by specific products, required quantity, stocked quantity, scrapped quantity, start date, end date and due date of work order.
WorkOrderRouting	Contains detailed data about production processes, transactions displayed in order of the production process, planned start and end dates, actual start and end dates, hours production used, estimates and actual production costs at specific times.
ProductInventory	Product inventory information. The transaction data is displayed for each product in an amount, in any location,

	shelf and bin

4.1.4. Fact and dimension tables



Data model

The figure shows the data warehouse model which contains two data marts (Production and Inventory). We have five dimensional tables and two fact tables. Below is a detailed description for each of them.

FactProduction table contains information about manufacturing work orders and work order details

	Column Name	Data type	Nullable	Description
PK	ProductionKey	int		Primary key for production fact table. Auto increment column
FK	FactProductKey	int		Product identification number. Foreign key to

				DimProduct.ProductKey
FK	LocationKey	int	N	Location identification number. Foreign key to DimLocation.LocationKey
FK	ScrapReasonKey	int	N	ScrapReason identification number. Foreign key to DimScrapReason.ScrapReason Key
	WorkOrderID	int		Identity for WorkOrder records.
	OrderQty	int		Product quantity need to build.
	StockedQty	int		Quantity built and put in inventory. Computed: isnull([OrderQty]-[ScrappedQty],(0))
	ScrappedQty	smallint		Quantity that failed inspection
FK	StartDate	int		Work order start date.
FK	EndDate	int		Work order end date.
FK	DueDate	int		Work order due date.
	OperationSequence	smallint		Indicates the manufacturing process sequence.
FK	ScheduledStartDate	int		Planned manufacturing start date.
FK	ScheduledEndDate	int		Planned manufacturing end date.
FK	ActualStartDate	int	N	Actual start date.

FK	ActualEndDate	int	N	Actual end date.
FK	ActualResourceHrs	decimal(9, 4)	N	Number of manufacturing hours used.
	PlannedCost	money		Estimated manufacturing cost.
	ActualCost	money	N	Actual manufacturing cost.

FactProduction table

FactInventory: storages inventory description includes storage compartment, storage container, and quantity of products in the inventory location

	Column	Data Type	Nullable	Description
PK	InventoryKey	int		Primary key for DimInventory table. Autoincrement column
FK	ProductKey	int		
FK	LocationKey	int		
	Shelf	nvarchar(10)		Storage compartment within an inventory location.
	Bin	tinyint		Storage container on a shelf in an inventory location.
	Quantity	smallint		Quantity of products in the inventory location.
FK	ModifiedDate	int		Date and time the record was last updated. Default: getdate()

FactInventory table

FactTransaction table contains information about product transaction history and Transaction history archive.

	Column	Data Type	Nullable	Description
PK	TransactionKey	int		Primary key for Transaction fact table. Auto increment column
	TransactionID	int		Purchase order, sales order, or work order identification number.
FK	ProductKey	int		
	ReferenceOrderID	int		Purchase order, sales order, or work order identification number.
	ReferenceOrderLineID	int		Line number associated with the purchase order, sales order, or work order.
FK	TransactionDate	int		Date and time of the transaction
	TransactionType	nchar(1)		W = Work Order, S = Sales Order, P = Purchase Order
	Quantity	int		Product quantity.
	ActualCost	money		Product cost.

FactTransaction table

FactSale table contains information about Sale order detail.

	Column	Data Type	Nullable	Description
PK	FactSalesKey	int		Primary key for Sale fact table. Auto increment column
	SalesOrderID	int		
FK	ProductKey	int		
	OrderQty	smallint		Quantity ordered per product.

	UnitPrice	money		Selling price of a single product.
	LineTotal	numeric(38, 6)		Per product subtotal.
FK	OrderDate	int		Dates the sales order was created

FactSale table

DimDate includes the format of date so that we can visualize facts and figures over the time in the calendar. In this table, we use the data of Year, Month, Day, DayNumberOfWeek, NameOfDay, QuarterOfYear to be easily statistical and intuitive guarantee in data mining. The PK_Date format is automatically saved from the fact table.

	Column Name	Data type	Nullable	Description
PK	DateKey	int		Primary key for Date Dimension table. Auto increment column
	ActualDate	int		
	Year	int		
	Month	int		
	Day	int		
	DayNumberOfWeek	int		Example: Sunday is 1, Monday is 2 and so on
	NameOfDay	int		Example: Monday, Tuesday
	QuarterOfYear	int		

DimDate table

DimScrapReason: contains the scrap reasons of product. Using the information in this table can help us sort, statistics the amount of excess or broken goods.

	Column Name	Data type	Nullable	Description
PK	ScrapReasonKey	int		Primary key for Scrap Reason Dimension table. Auto increment column
FK	ScrapReasonID	smallint		Scrap reason identification number
	Name	nvarchar(50)		Failure description.

DimScrapReason table

DimProduct storages product information sold or used in the manufacturing of sold products.

	Column Name	Data type	Nullable	Description
PK	Product Key	int		Primary key for Product Dimension table. Autoincrement column
	ProductID	int		Product identification number
	Name	nvarchar(50)		Name of the product.
	ProductCategory	nvarchar(50)		Category description.
	ProductSubcategory	nvarchar(50)		Subcategory description.

	Color	nvarchar(15)	N	Product color.
	SafetyStockLevel	smallint		Minimum inventory quantity
	ReorderPoint	smallint		Inventory level that triggers a purchase order or work order
	StandardCost	money		Standard cost of the product
	DaysToManufacture	int		Number of days required to manufacture the product
	ProductLine	nchar(2)		4 lines: R = Road, M = Mountain, T = Touring, S = Standard
	Class	nchar(2)	N	3 classes: H = High, M = Medium, L = Low
	Style	nchar(2)	N	3 styles: W = Womens, M = Mens, U = Universal
	SellEndDate	int	N	Date the product was no longer available for sale.
	ListPrice	money		Selling price

DimProduct table

DimLocation: storages inventory location and location description include cost and the availability of them.

	Column	Data Type	Nullable	Description

PK	LocationKey	int		Primary key for DimLocation table. Autoincrement column
	LocationID	smallint		Inventory location identification number.
	Name	nvarchar(50)		Location description
	CostRate	smallmoney		Standard hourly cost of the manufacturing location
	Availability	decimal(8,2)		Work capacity (in hours) of the manufacturing location

DimLocation table

4.2. ETL processes

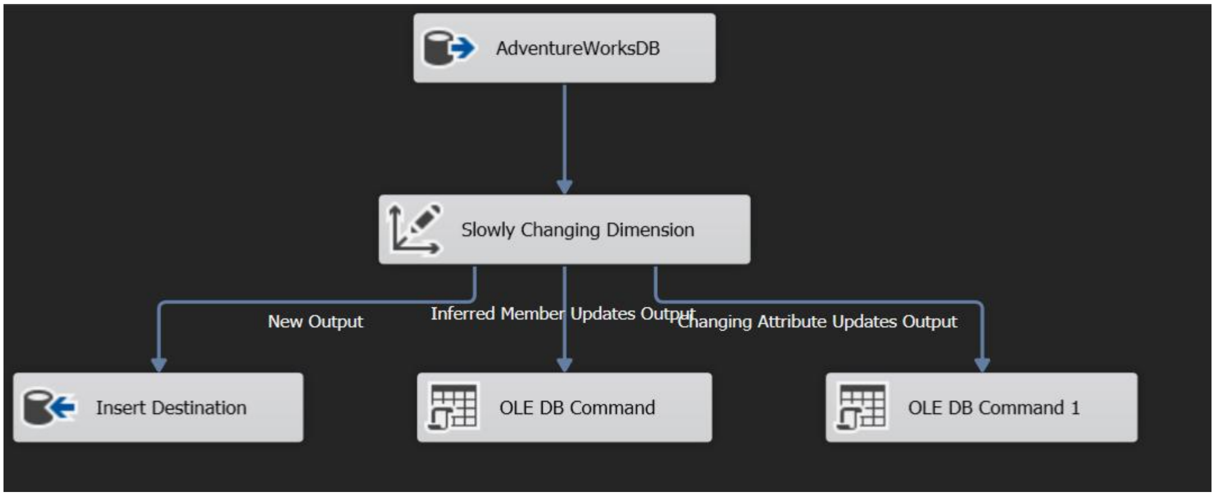
After defining the data warehouse model, the next step we need to perform is populating data warehouse. The tools to transfer data from the source table to the target table we used is SSIS - a platform for data integration and workflow applications.

4.2.1. Dimension Table's ETL Process

Each dimension table will have

- The primary key (surrogate key): This is an identity (1,1) column, and the values are unique. It joins with the foreign key in the fact table and allows to connect those two tables to ensure the integrity. It has no business meaning and used to maintain a hierarchy.
- The natural key (domain key): A descriptor of the data. It is based on attributes that exist. The relationship of the surrogate key and natural key may be one o one or many to one (in slowly changing dimension)
- Descriptive attributes (textual, numeric): describes a non-unique characteristic of an entity instance

Besides, dimension tables have the similar ETL process using Slowly Changing Dimension to store and manage both current and historical data over time in a data warehouse.



Data task flow of dimension tables

4.2.1.1. DimProduct Table

Initially, in AdventureWorks database, we built a DimProductTemp table to get attributes and their data types. Then in data warehouse ProductionDW, we created a DimProduct table with the same design as the DimProductTemp. We added the surrogate key as the primary key of the table, they are unique and not null.

Below is the query we used to build “DimProductTemp” table:

```
SELECT ProductID,
        P.Name,
        C.Name ProductCategory,
        S.Name ProductSubcategory,
        (CASE WHEN Color is null THEN 'Undefined' ELSE Color END) Color,
        ROUND(StandardCost, 2) StandardCost,
        ListPrice,
        DaysToManufacture,
        (CASE WHEN ProductLine is null THEN 'Undefined' ELSE ProductLine END) ProductLine,
        (CASE WHEN Class is null THEN 'N/A' ELSE Class END) Class,
        (CASE WHEN Style is null THEN 'N/A' ELSE Style END) Style,
        ReorderPoint,
        SafetyStockLevel,
        (CASE WHEN SellEndDate IS NULL THEN 0 ELSE FORMAT(SellEndDate, 'yyyyMMdd') END)SellEndDate
INTO DimProductTemp
FROM Production.Product P
    LEFT JOIN Production.ProductSubcategory S ON P.ProductSubcategoryID = S.ProductSubcategoryID
    LEFT JOIN Production.ProductCategory C ON C.ProductCategoryID = S.ProductCategoryID
```

Design of DimProduct table:

	Column Name	Data Type	Allow Nulls
🔑	ProductKey	int	☐
	ProductID	int	☐
	Name	nvarchar(50)	☐
	ProductCategory	nvarchar(50)	☑
	ProductSubcategory	nvarchar(50)	☑
	Color	nvarchar(15)	☑
	StandardCost	money	☐
	ListPrice	money	☐
	DaysToManufacture	int	☐
	ProductLine	nvarchar(9)	☑
	Class	nvarchar(3)	☑
	Style	nvarchar(3)	☑
	ReorderPoint	smallint	☐
	SafetyStockLevel	smallint	☐
	SellEndDate	int	☑

Data designing of DimProduct table

After running we have the result:

	ProductKey	ProductID	Name	ProductCategory	ProductSubcategory	Color	SafetyStockLevel	ReorderPoint	StandardCost	DaysToManufacture	ProductL
1	1	680	HL Road Frame - Black, 58	Components	Road Frames	Black	500	375	1059.31	1	R
2	2	706	HL Road Frame - Red, 58	Components	Road Frames	Red	500	375	1059.31	1	R
3	3	707	Sport-100 Helmet, Red	Accessories	Helmets	Red	4	3	13.09	0	S
4	4	708	Sport-100 Helmet, Black	Accessories	Helmets	Black	4	3	13.09	0	S
5	5	709	Mountain Bike Socks, M	Clothing	Socks	White	4	3	3.40	0	M
6	6	710	Mountain Bike Socks, L	Clothing	Socks	White	4	3	3.40	0	M
7	7	711	Sport-100 Helmet, Blue	Accessories	Helmets	Blue	4	3	13.09	0	S
8	8	712	AWC Logo Cap	Clothing	Caps	Multi	4	3	6.92	0	S
9	9	713	Long-Sleeve Logo Jersey, S	Clothing	Jerseys	Multi	4	3	38.49	0	S
10	10	714	Long-Sleeve Logo Jersey, M	Clothing	Jerseys	Multi	4	3	38.49	0	S
11	11	715	Long-Sleeve Logo Jersey, L	Clothing	Jerseys	Multi	4	3	38.49	0	S
12	12	716	Long-Sleeve Logo Jersey, XL	Clothing	Jerseys	Multi	4	3	38.49	0	S
13	13	717	HL Road Frame - Red, 62	Components	Road Frames	Red	500	375	868.63	1	R

Query executed successfully. | DESKTOP-7TS1P30\SQLEXPRESS ... | DESKTOP-7TS1P30\admin ... | ProductionDW(final) | 00:00:00 | 504 rows

Data of DimProduct table

4.2.1.2. DimLocation Table

Initially, we built a DimLocationTemp table to get attributes and their data types. We then designed a DimLocation table in our data warehouse called ProductionDW with the same design as the DimLocationTemp.

Query to build "DimLocationTemp" table in AdventureWorks database:

```sql
SELECT LocationID,
       Name as LocationName,
       CostRate,
       Availability
INTO DimLocationTemp
FROM Production.Location
```

Design of DimLocation:

	Column Name	Data Type	Allow Nulls
🔑	LocationKey	int	☐
	LocationID	smallint	☐
	LocationName	nvarchar(50)	☐
	CostRate	smallmoney	☐
	Availability	decimal(8, 2)	☐

Data designing of DimLocation table

After running we have the result:

	LocationKey	LocationID	Name	CostRate	Availability
1	15	1	Tool Crib	0.00	0.00
2	16	2	Sheet Metal Racks	0.00	0.00
3	17	3	Paint Shop	0.00	0.00
4	18	4	Paint Storage	0.00	0.00
5	19	5	Metal Storage	0.00	0.00
6	20	6	Miscellaneous Storage	0.00	0.00
7	21	7	Finished Goods Storage	0.00	0.00
8	22	10	Frame Forming	22.50	96.00
9	23	20	Frame Welding	25.00	108.00
10	24	30	Debur and Polish	14.50	120.00
11	25	40	Paint	15.75	120.00
12	26	45	Specialized Paint	18.00	80.00
13	27	50	Subassembly	12.25	120.00
14	28	60	Final Assembly	12.25	120.00

Query executed successfully. | DESKTOP-7TS1P30\SQLEXPRESS ... | DESKTOP-7TS1P30\admin ... | ProductionDW(final) | 00:00:00 | 14 rows

Data of DimLocation table

4.2.1.3. DimScrapReason Table

Initially, we built a DimScrapReasonTemp table to get attributes and their data types. We then designed a DimLocation table in our database called ProductionDW with the same design as the DimScrapReasonTemp.

Design DimScrapReasonTemp table in AdventureWorks database based on the query:

```sql
SELECT ScrapReasonID,
       Name AS ScrapReason
INTO DimScrapReasonTemp
FROM Production.ScrapReason
```

Design of DimScrapReason table

	Column Name	Data Type	Allow Nulls
🔑	ScrapReasonKey	int	☐
	ScrapReasonID	smallint	☐
	ScrapReason	nvarchar(50)	☐

Data designing of DimScrapReason table

After running, we have the result:

	ScrapReasonKey	ScrapReasonID	Name
1	1	1	Brake assembly not as ordered
2	2	2	Color incorrect
3	3	4	Drill pattern incorrect
4	4	5	Drill size too large
5	5	6	Drill size too small
6	6	3	Gouge in metal
7	7	7	Handling damage
8	8	8	Paint process failed
9	9	9	Primer process failed
10	10	10	Seat assembly not as ordered
11	11	11	Stress test failed
12	12	12	Thermoform temperature too high
13	13	13	Thermoform temperature too low

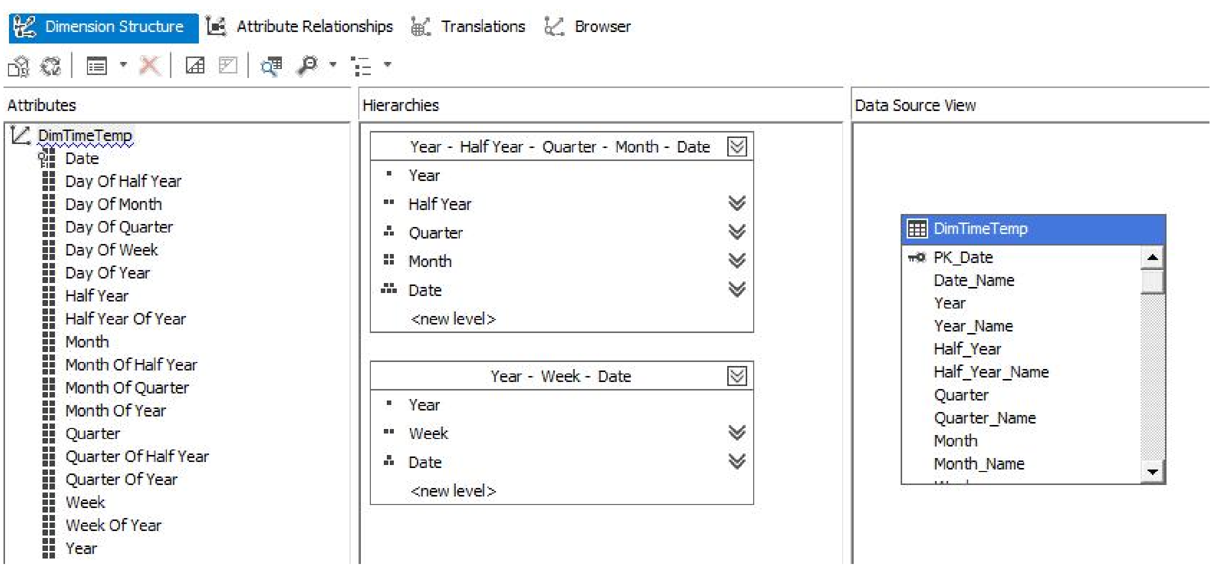
Query executed successfully. | THUYNDT-IS\THUYNDT (15.0 RTM) | THUYNDT-IS\HP (58) | ProductionDW | 00:00:00 | 16 rows

Data of DimScrapReason table

4.2.1.3. DimDate Table

There are many ways to create a date dimension table. In this case, we will use Analysis Services Multidimensional and Data Mining Project to create a DimTimeTemp table.

Dimension structure of DimTimeTemp:



Dimension structure of DimTimeTemp

After running, we have the result of DimTimeTemp:

Results | Messages

	PK_Date	Date_Name	Year	Year_Name	Half_Year	Half_Year_Name	Quarter	Quarter_Nan
1	2006-01-01 00:00:00.000	Sunday, January 01 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
2	2006-01-02 00:00:00.000	Monday, January 02 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
3	2006-01-03 00:00:00.000	Tuesday, January 03 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
4	2006-01-04 00:00:00.000	Wednesday, January 04 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
5	2006-01-05 00:00:00.000	Thursday, January 05 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
6	2006-01-06 00:00:00.000	Friday, January 06 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
7	2006-01-07 00:00:00.000	Saturday, January 07 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
8	2006-01-08 00:00:00.000	Sunday, January 08 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
9	2006-01-09 00:00:00.000	Monday, January 09 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2
10	2006-01-10 00:00:00.000	Tuesday, January 10 2006	2006-01-01 00:00:00.000	Calendar 2006	2006-01-01 00:00:00.000	Semester 1, 2006	2006-01-01 00:00:00.000	Quarter 1, 2

Query executed successfully. | LAPTOP-SI3K33UD (15.0 RTM) | LAPTOP-SI3K33UD\vannt ... | ProductionDW_Temp | 00:00:00 | 5,479 rows

Data of DimTimeTemp table

Then, based on DimTimeTemp we create a DimDate that is suitable for our project. Below is the query:

```
SELECT
    FORMAT(PK_Date,'yyyymmdd') DateKey,
    PK_Date AS ActualDate,
    Year(PK_Date) Year,
    Month(PK_Date) Month,
    Day(PK_Date) Day,
    DATEPART(WEEKDAY, PK_Date) DayNumberOfWeek,
    DATENAME(WEEKDAY, PK_Date) NameOfDay,
    DATEPART(QUARTER, PK_Date) QuarterOfYear
INTO DimTime
FROM [dbo].[DimTimeTemp]
```

The result of DimDate:

	DateKey	ActualDate	Year	Month	Day	DayNumberOfWeek	NameOfDay	QuarterOfYear
1	20060101	2006-01-01	2006	1	1	1	Sunday	1
2	20060102	2006-01-02	2006	1	2	2	Monday	1
3	20060103	2006-01-03	2006	1	3	3	Tuesday	1
4	20060104	2006-01-04	2006	1	4	4	Wednesday	1
5	20060105	2006-01-05	2006	1	5	5	Thursday	1
6	20060106	2006-01-06	2006	1	6	6	Friday	1
7	20060107	2006-01-07	2006	1	7	7	Saturday	1
8	20060108	2006-01-08	2006	1	8	1	Sunday	1
9	20060109	2006-01-09	2006	1	9	2	Monday	1
10	20060110	2006-01-10	2006	1	10	3	Tuesday	1
11	20060111	2006-01-11	2006	1	11	4	Wednesday	1

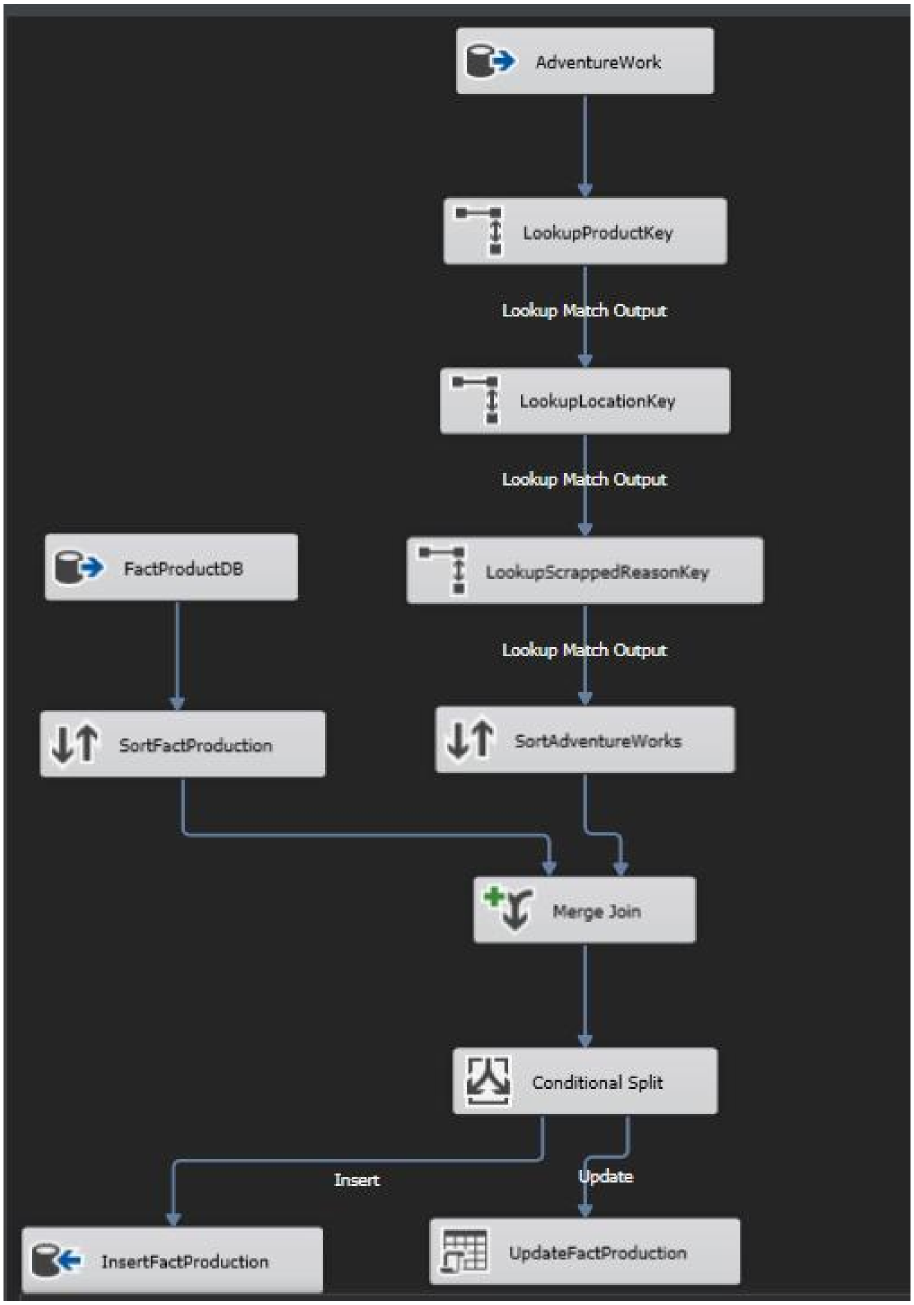
Query executed successfully. | LAPTOP-SI3K33UD (15.0 RTM) | LAPTOP-SI3K33UD\vannt ... | ProductionDW_Temp | 00:00:00 | 1,000 rows

Data of DimDate table

4.2.2. Fact Table's ETL Process

4.2.2.1 Table FactProduction

Table FactProduction is extracted and transformed with SSMS and then loaded with SSIS. Below is the data task flow loading data from AdventureWorks into data warehouse. FactProduction is lookup with ProductKey, LocationKey, ReasonKey in that order.



Data flow task of FactProduction

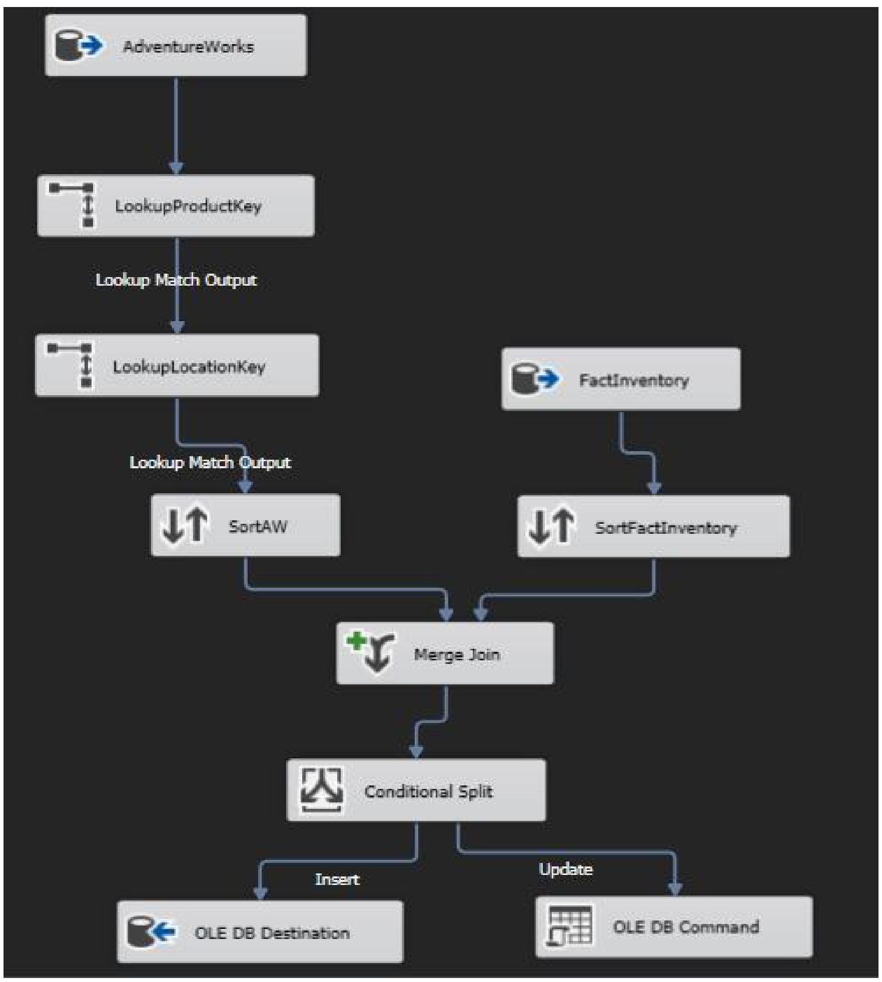
	ProductionKey	WorkOrderID	ProductKey	ScrapReasonKey	OrderQty	StockedQty	ScrappedQty	StartDate	EndDate	DueDate	LocationKey
1	71523	1	22	NULL	8	8	0	20110603	20110613	20110614	NULL
2	72132	2	25	NULL	15	15	0	20110603	20110613	20110614	NULL
3	72157	3	26	NULL	9	9	0	20110603	20110613	20110614	NULL
4	72474	4	29	NULL	16	16	0	20110603	20110613	20110614	NULL
5	72658	5	30	NULL	14	14	0	20110603	20110613	20110614	NULL
6	72696	6	32	NULL	16	16	0	20110603	20110613	20110614	NULL
7	72724	7	33	NULL	4	4	0	20110603	20110613	20110614	NULL
8	73714	8	38	NULL	19	19	0	20110603	20110613	20110614	NULL
9	74334	9	41	NULL	2	2	0	20110603	20110613	20110614	NULL
10	74608	10	42	NULL	3	3	0	20110603	20110613	20110614	NULL
11	74881	11	43	NULL	1	1	0	20110603	20110613	20110614	NULL
12	75371	12	45	NULL	1	1	0	20110603	20110613	20110614	NULL

Query executed successfully. | THUYNDT-IS\THUYNDT (15.0 RTM) | THUYNDT-IS\HP (70) | ProductionDW | 00:00:00 | 69,501 rows

Data of FactProduction

4.2.2.2 Table FactInventory

Similarly, data from FactInventory is loaded. However, some errors have come into view so that the data warehouse is not loaded as we expected.



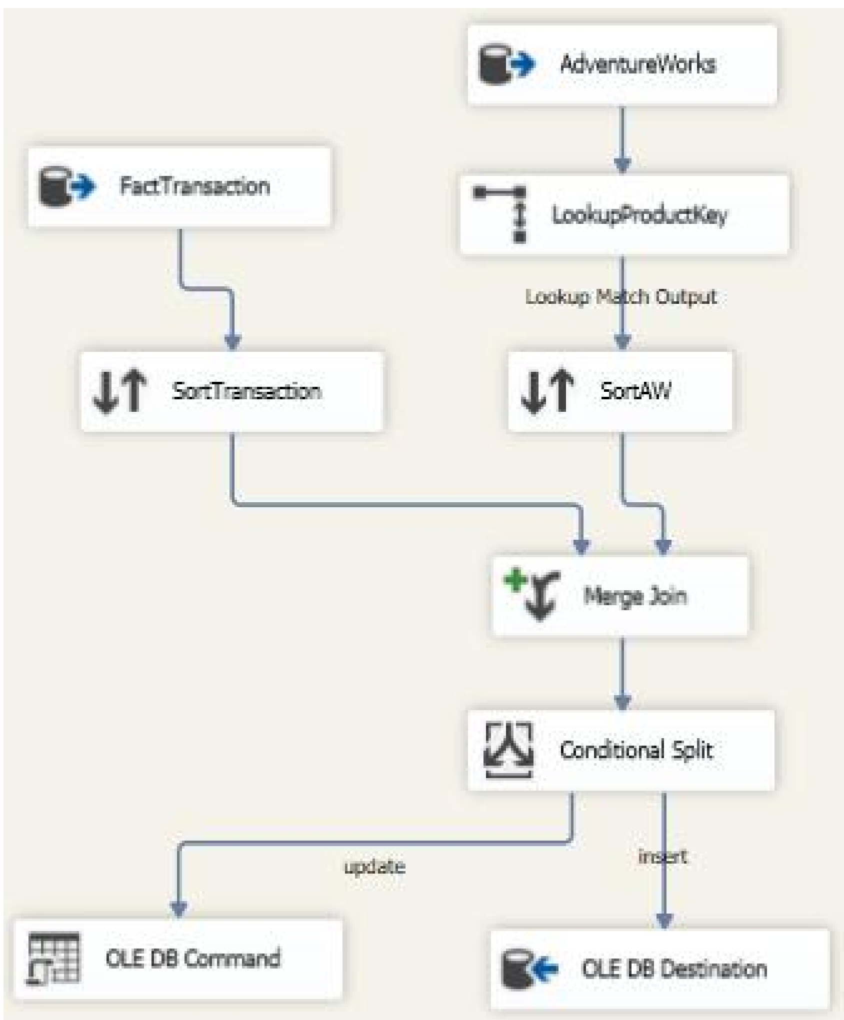
Data flow task of FactInventory

	FactInventoryKey	ProductKey	LocationKey	Shelf	Bin	Quantity	ModifiedDate
1	1002	3	21	N/A	0	288	20110501
2	1003	4	21	N/A	0	324	20110501
3	1004	5	21	N/A	0	180	20110501
4	1005	6	21	N/A	0	216	20110501
5	1006	7	21	N/A	0	216	20110501
6	1007	8	21	N/A	0	288	20110501
7	1008	9	21	N/A	0	144	20110501
8	1009	10	21	N/A	0	180	20110501
9	1010	11	21	N/A	0	216	20110501
10	1011	12	21	N/A	0	252	20110501
11	1012	43	22	N/A	0	121	20110501
12	1013	43	23	N/A	0	161	20110501
13	1014	43	24	N/A	0	148	20110501
14	1015	43	25	N/A	0	107	20110501
15	1016	43	27	N/A	0	160	20110501
16	1017	43	28	N/A	0	137	20110501
17	1018	44	22	N/A	0	105	20110501
18	1019	44	23	N/A	0	145	20110501
19	1020	44	24	N/A	0	132	20110501

Query executed successfully. | THUYNDT-IS\THUYNDT (15.0 RTM) | THUYNDT-IS\HP (74) | ProductionDW | 00:00:00 | 1,069 rows

Data of FactInventory

4.2.2.3 Table FactTransaction



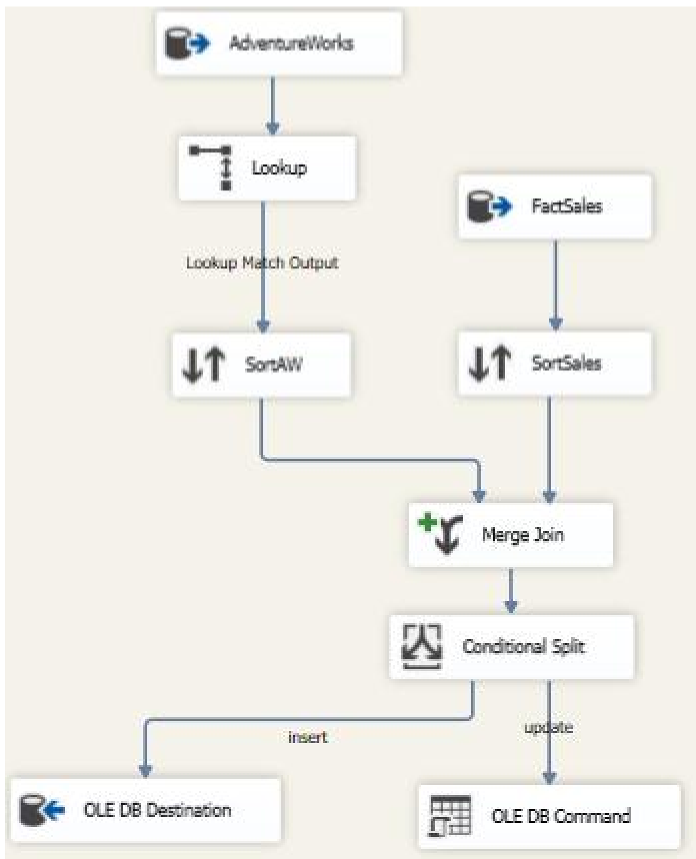
Data flow task of FactTransaction

	TransactionKey	TransactionID	ProductKey	ReferenceOrderID	ReferenceOrderLineID	TransactionDate	TransactionType	Quantity	ActualCost
1	1	104802	1	42112	0	20130804	W	1	0.00
2	2	113481	1	44520	0	20130901	W	1	0.00
3	3	127803	1	48969	0	20131018	W	2	0.00
4	4	134283	1	50260	0	20131101	W	1	0.00
5	5	145886	1	54082	0	20131206	W	1	0.00
6	6	151916	1	56248	0	20131229	W	1	0.00
7	7	159651	1	58311	0	20140118	W	1	0.00
8	8	172559	1	62221	0	20140226	W	2	0.00
9	9	187491	1	65856	0	20140402	W	1	0.00
10	10	192045	1	67443	0	20140416	W	1	0.00
11	11	202055	1	69957	0	20140509	W	1	0.00
12	12	106485	1	42714	0	20130811	W	1	0.00
13	13	117619	1	46086	0	20130917	W	1	0.00
14	14	123642	1	47294	0	20131001	W	1	0.00

Query executed successfully. | THUYNDT-IS\THUYNDT (15.0 RTM) | THUYNDT-IS\HP (70) | ProductionDW | 00:00:01 | 202,696 rows

Data of FactTransaction

4.2.3.4 Table FactSales



Data flow task of FactSales

	FactSalesKey	SalesOrderID	ProductKey	OrderQty	UnitPrice	LineTotal	OrderDate
1	1	43875	3	2	20.1865	40.373000	20110701
2	2	47028	3	7	20.1865	141.305500	20120630
3	3	49118	3	4	20.1865	80.746000	20121231
4	4	51400	3	1	34.99	34.990000	20130612
5	5	53459	3	6	20.994	125.964000	20130731
6	6	55095	3	1	34.99	34.990000	20130827
7	7	58715	3	1	34.99	34.990000	20131027
8	8	44745	3	3	20.1865	60.559500	20111031
9	9	47398	3	2	20.1865	40.373000	20120731
10	10	47973	3	3	20.1865	60.559500	20120930
11	11	50311	3	4	20.1865	80.746000	20130330
12	12	53266	3	1	34.99	34.990000	20130727
13	13	55812	3	1	34.99	34.990000	20130906

Query executed successfully. | THUYNDT-IS\THUYNDT (15.0 RTM) | THUYNDT-IS\HP (71) | ProductionDW | 00:00:00 | 121,317 rows

Data of FactSales

CHAPTER 5: DATA ANALYTICS

5.1. Data analytics with SSAS technology

We utilize SSAS technology (SQL Server Analysis Service) to build a cube and analyze data from data warehouse “ProductionDW”.

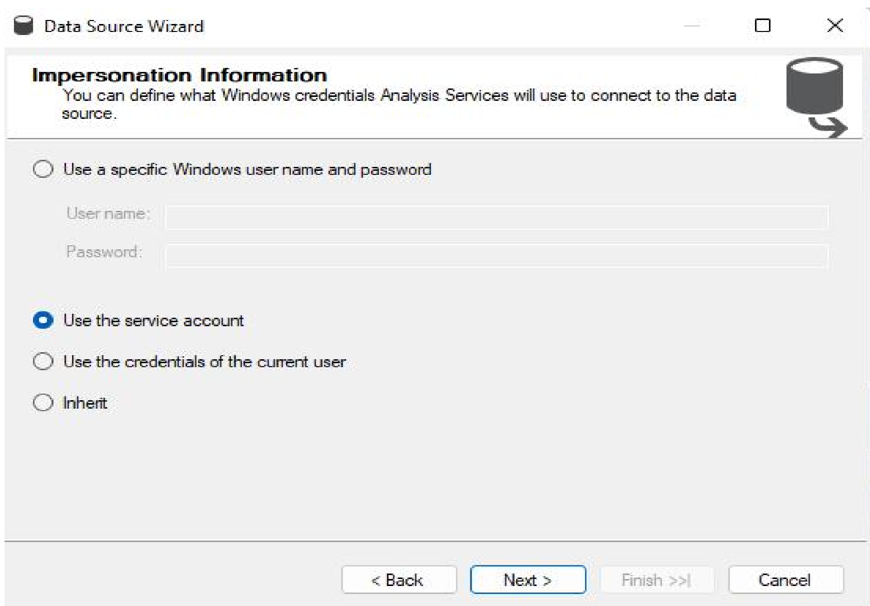
There are 5 steps to implement an Analysis Services database. Below is a detailed description for each step.

Step 1: Create a data source

Firstly, we created an Analysis Services project in SQL Server 2019 Business Intelligence Development Studio and named the project “ProductionDW_SSAS”.

Then, we connect the project to the ProductionDW sample data warehouse on a local instance of SQL Server 2019.

After setting up the connection, we configure the impersonation information as the figure below

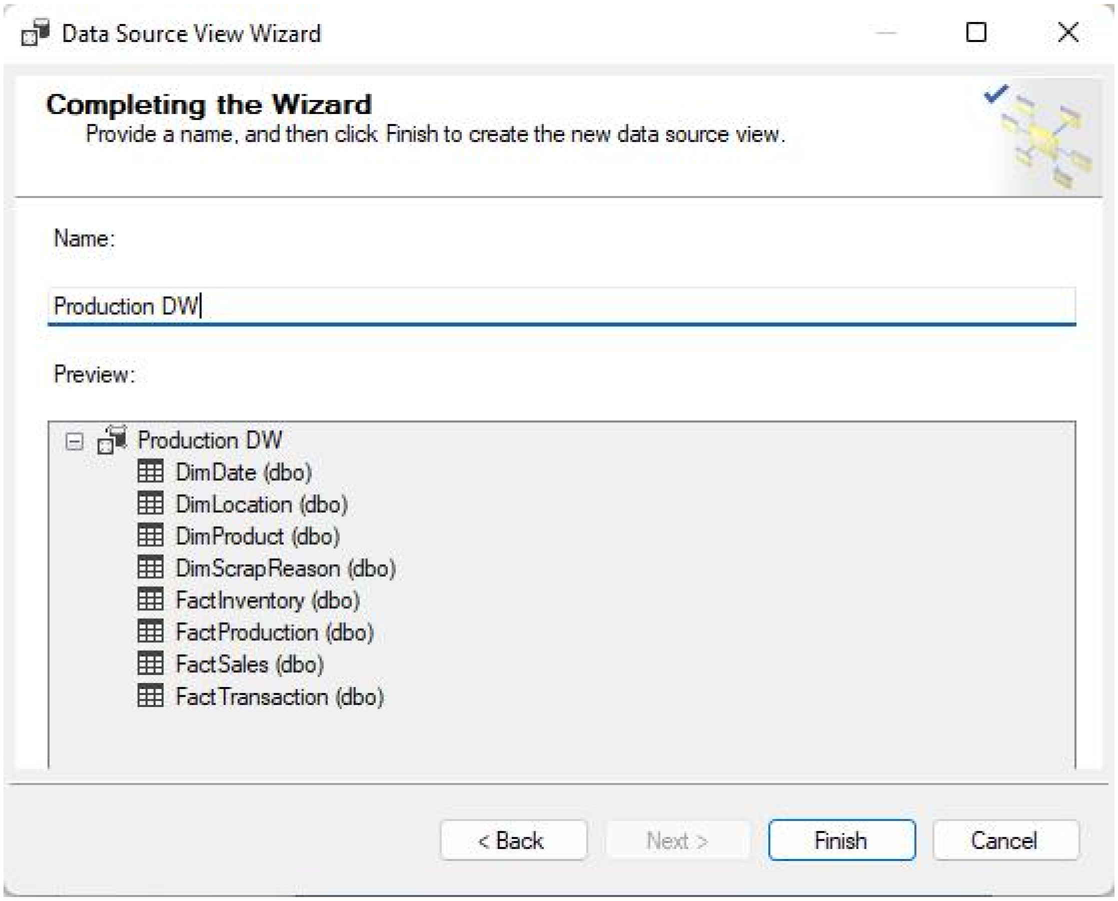


Impersonation information

After that, data sources connected successfully.

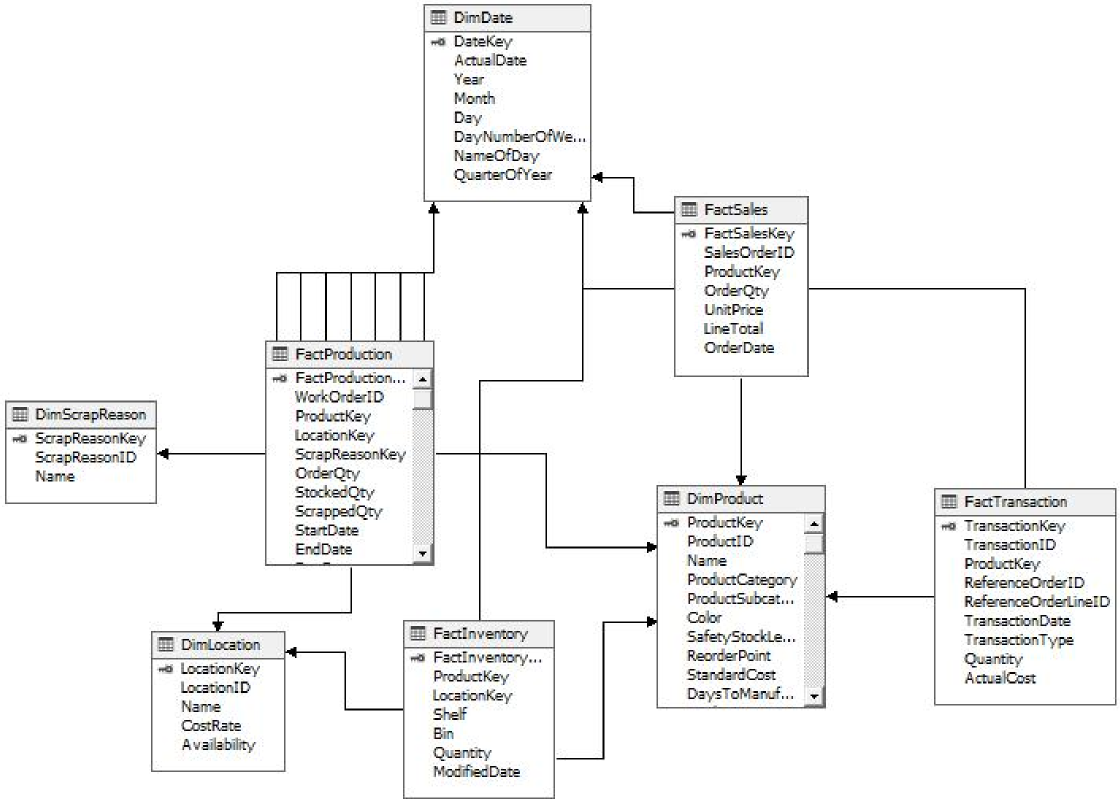
Step 2: Create a data source view

For the purposes of this project, we selected four fact tables (FactInventory, FactProduction, FactSales and FactTransaction) and four dimension tables (DimLocation, DimDate, DimProduct and DimScrapReason)



Data source view wizard

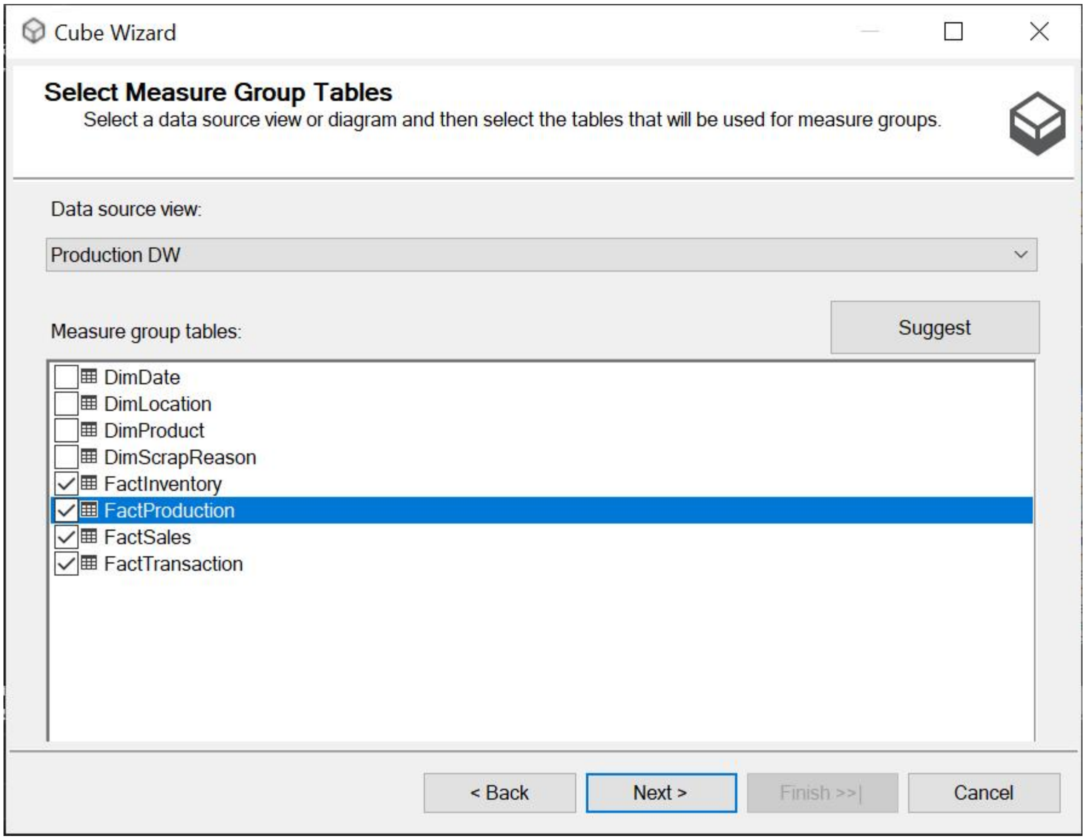
After clicking the Finish button, SQL Server Analysis Services creates the data source view and opens a window in the design surface that displays a detailed view, as shown in the figure.



Data warehouse view

Step 3: Create a cube

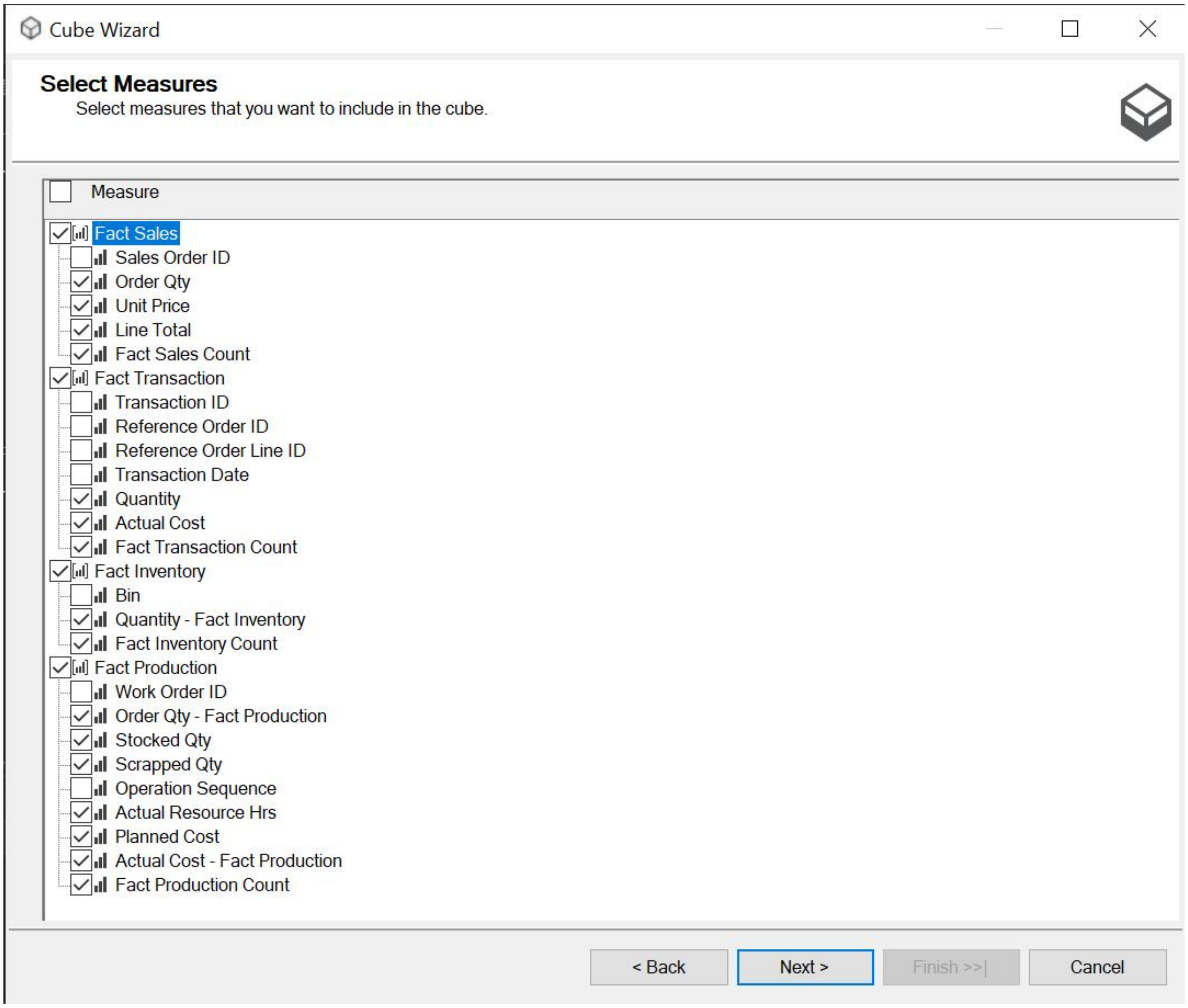
From the data source view, we build a cube that is a multi-dimensional data source which has four fact tables, including: FactInventory, FactProduction, FactSales and FactTransaction. Each table is a measure group table. Measure groups are used to associate dimensions with measures.



Fact tables in cube

Continue, we select measures that contain quantifiable data, details are as follow:

- From FactSales: OrderQuantity, UnitPrice, LineTotal.
- From FactTransaction: Quantity, ActualCost.
- From FactInventory: Quantity.
- From FactProduction: OrderQuantity, StockedQuantity, ScrappedQuantity, ActualResourceHrs, PlannedCost.



Measures included in the cube

Next, we pick dimension tables to include in the cube: DimProduct, DimDate, DimLocation, DimScrapReason.

When completing the wizard, SQL Server Analysis Services will generate the cube and open it in the designer surface. The figure below shows the Cube Designer for the Production DW project.

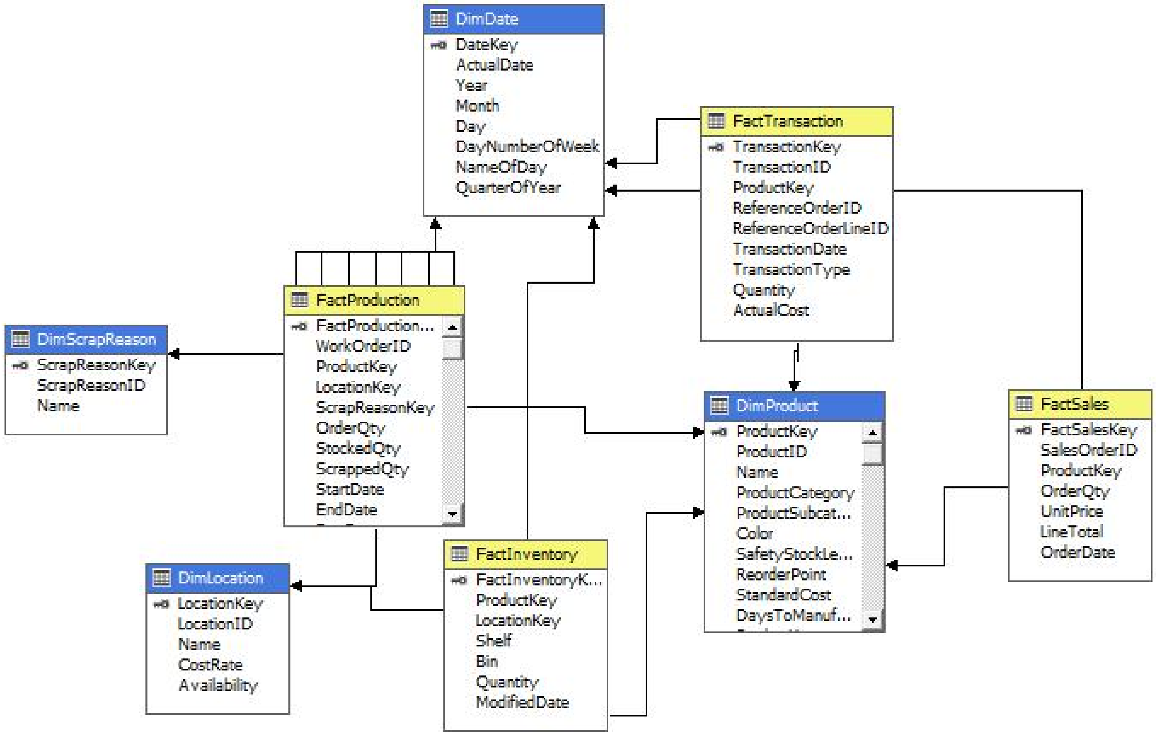


Figure: The ProductionDW cube in Cube Designer

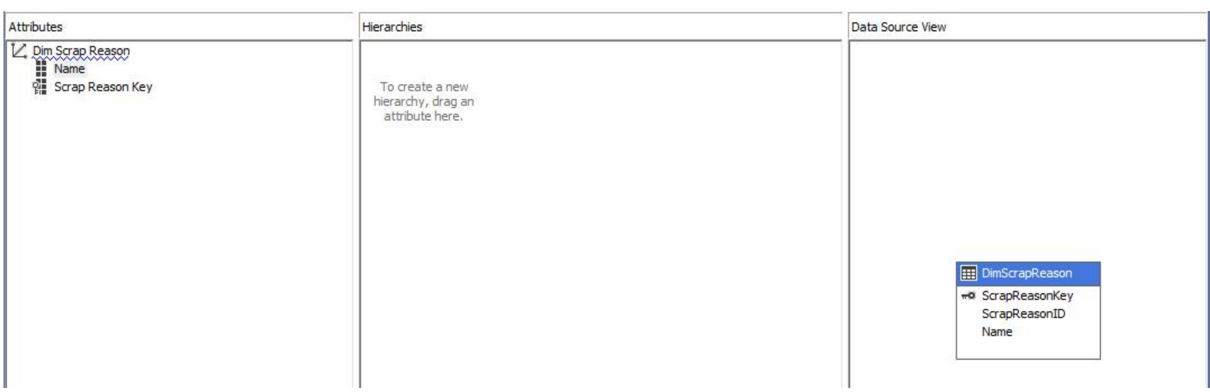
Step 4: Edit Dimension

The next step is to specify the attributes to include in the dimension tables. We double click on the name of dimension, then select every attributes that are suitable for the project, and drag the selected properties from the Data Source View to Attributes. Figures below show the attributes we have selected for every dimension.

Dimension Structure of DimProduct

Dimension Structure of DimDate

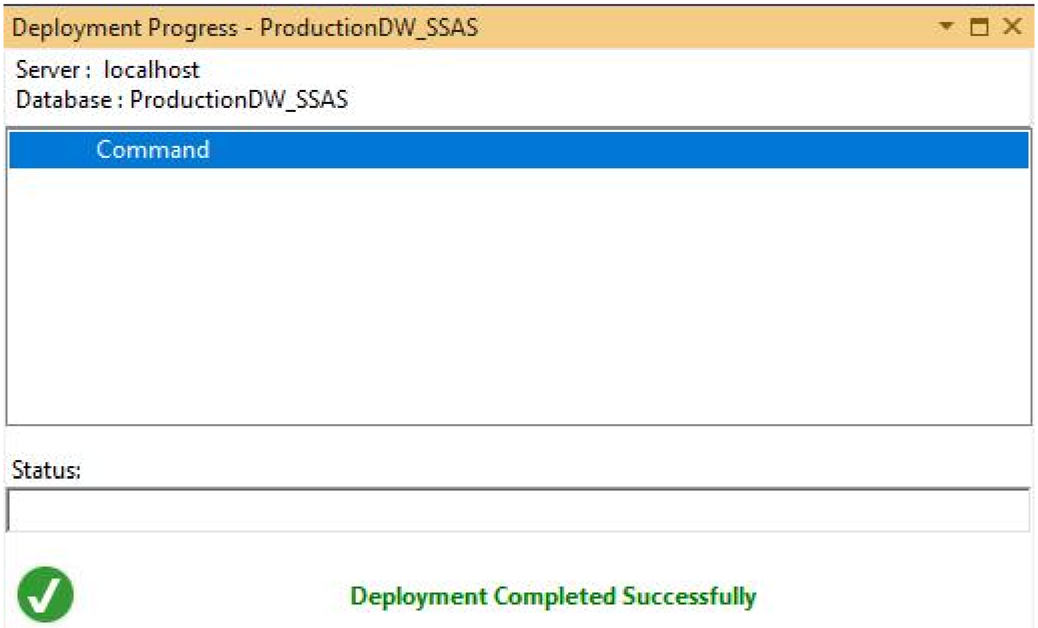
Dimension Structure of DimLocation



Dimension Structure of DimScrapReason

Step 5: Deploy the database

The final step is to deploy the database to the SQL Server instance. Right-click the project name in Solution Explorer, and then click Deploy. The database is deployed with no problems, we receive a message saying that the deployment is complete.



Deployment Completed Successfully

5.1.3. Building KPIs system with SSAS

5.1.3.1 Production Volume

Production volume measures the total amount your company can produce over time. This KPI tracks the total number of products manufactured over a set period of time and focuses on total output.

KPI name: KPIManufacturingQuantity

Value expression: [Measures].[Stocked Qty]

Goal expression:

CASE WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Bikes] THEN 40000

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Components] THEN 200000

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Accessories] THEN 0

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Clothing] THEN 0

END

Status expression

CASE

```
WHEN
KPIVALUE('KPIManufacturingQuantity')/KPIGOAL('KPIManufacturingQuantity') >
0.7 THEN 1

WHEN
KPIVALUE('KPIManufacturingQuantity')/KPIGOAL('KPIManufacturingQuantity') <=
0.7 THEN -1

END
```

Trend expression

```
CASE

WHEN ISEMPTY(PARALLELPERIOD([End Date].[Year].[Year],1,[End

Date].[Year]))THEN 0

WHEN [Measures].[Stocked Qty] > (PARALLELPERIOD([End
Date].[Year].[Year], 1, [End Date].[Year].currentmember),[Measures].[Stocked Qty])
THEN 1

WHEN [Measures].[Stocked Qty] = (PARALLELPERIOD([End
Date].[Year].[Year], 1, [End Date].[Year].currentmember), [Measures].[Stocked Qty])
THEN 0

ELSE -1

END
```

Results:

Row Labels	KPIManufacturingQuantity	KPIManufacturingQuantity Goal	KPIManufacturing Quantity Status	KPIManufacturing Quantity Trend
2011				
Bikes	7831	40000		
Components	118854	200000		
2012				
Bikes	25758	40000		
Components	415445	200000		
2013				
Bikes	37348	40000		
Components	602703	200000		
2014				
Bikes	19183	40000		
Components	298697	200000		
Grand Total	1525819			

Manufacturing quantity KPI

Row Labels	KPIManufacturingQuar	KPIManufacturingQuantity	KPIManufacturingQuantity Status	KPIManufacturingQuantity Trend
2011				
1				
Bikes		40000		
Components		200000		
2				
Bikes	501	40000		
Componen	7563	200000		
3				
Bikes	3121	40000		
Componen	47595	200000		
4				
Bikes	4209	40000		
Componen	63696	200000		
2012				
1				
Bikes	3950	40000		
Componen	58762	200000		
2				
Bikes	5931	40000		
Componen	94327	200000		
3				
Bikes	8498	40000		
Componen	142956	200000		
4				
Bikes	7379	40000		
Componen	119400	200000		
2013				
1				
Bikes	7041	40000		
Componen	111392	200000		
2				
Bikes	8266	40000		
Componen	136565	200000		
3				
Bikes	11432	40000		
Componen	188094	200000		
4				
Bikes	10609	40000		
Componen	166652	200000		
2014				
1				
Bikes	10001	40000		
Componen	154488	200000		
2				
Bikes	9182	40000		
Componen	144209	200000		
3				
Bikes		40000		
Components		200000		
4				
Bikes		40000		
Components		200000		
Grand Total	1525819			

Manufacturing cost KPI

We set the KPIs for each product category as follows:

Bikes: 40,000$ and Components: 200,000$

Looking at the table, we can see that bikes are often below the goal (except for 2013), while components have a better result, it exceeds the goal 3 years in a row (2011, 2012, 2013).

When it comes to trend, manufacturing volume in 2012 and 2013 both higher than the previous year. But in 2014, we just have the data for the first 6 months so we cannot know that the trend is actually decreasing or increasing.

5.1.3.2 Manufacturing cost

KPI name: KPIManufacturingCost

Value expression: [Measures].[Actual Cost - Fact Production]

Goal expression:

CASE WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Bikes] THEN 350000

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Components] THEN 800000

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Accessories] THEN 0

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Clothing] THEN 0

ELSE 0

END

Status expression:

CASE WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Bikes] THEN 350000

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Components] THEN 800000

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Accessories] THEN 0

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Clothing] THEN 0

ELSE 0

END

Trend:

CASE

WHEN ISEMPTY(PARALLELPERIOD([End Date].[Year].[Year],1,[End

Date].[Year]))THEN 0

WHEN [Measures].[StandardCostInventory] > (PARALLELPERIOD([End Date].[Year].[Year], 1, [End Date].[Year].currentmember), [Measures].[StandardCostInventory]) THEN 1

WHEN [Measures].[StandardCostInventory] = (PARALLELPERIOD([End Date].[Year].[Year], 1, [End Date].[Year].currentmember), [Measures].[StandardCostInventory]) THEN 0

ELSE -1

END

Row Labels	Stocked Qty	KPIManufacturingC	KPIManufacturingCo	ManufacturingCost St	KPIManufacturingCost Trend
⊟2011					
Bikes	7831	46795	350000	●	↑
Components	118854	234050.25	800000	●	↑
⊟2012					
Bikes	25758	122010	350000	●	↑
Components	415445	596854.25	800000	●	↑
⊟2013					
Bikes	37348	248332	350000	●	↑
Components	602703	1030700.25	800000	◆	↑
⊟2014					
Bikes	19183	196245	350000	●	↓
Components	298697	623163.25	800000	●	↓
Grand Total	**1525819**	**3098150**	**0**	◆	↑

Manufacturing cost KPI

Manufacturing cost KPI will show us the trend and status of manufacturing costs over 4 years (2011 to 2014). We set the KPIs for each product category as follows:

Bikes: 350000$ and Components: 800000$

In terms of status, the results show that in 2011 and 2012 both Bikes and Components products ensure that the cost does not exceed the KPI (it means that the production costs for the categories are lower than the target). However, by 2013, the production of Components had exceeded the cost (1030700.25$ > 800000$), causing the status to exceed the target. By the next year (2014), production costs were low again to ensure that they remained below the planned cost.

Regarding the trend of manufacturing cost, almost from 2011 to 2013, the cost tended to increase (especially in 2013 there was a very strong increase in the production of Components), but by 2014 there was a trend of decreasing costs in both categories. The production department should review and manage the status of production to ensure this trend of cost reduction, or a slight increase to remain within the original cost.

5.1.3.3 Scrap Rate

Scrap rate measures the amount of scraped products compared to the total of products produced. It enables the production department to keep track numbers of products which are deemed scrap. Additionally, it provides the company with insight in production run and identifies an inefficient process. The lower the ratio it has, the better manufacturing performs.

Scrap rate = # of Scrap Units / Total # of Units

Calculation: [ScrapRate]

Expression:[Measures].[Scrapped Qty]/[Measures].[Stocked Qty]

KPI name: KPIScrapRate

Value expression: [Measures].[ScrapRate]

Goal Expression:

CASE WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Bikes] THEN 0.0015

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Components] THEN 0.002

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Accessories] THEN 0

WHEN [Dim Product].[Product Category] IS [Dim Product].[Product Category].&[Clothing] THEN 0

ELSE 0.001

END

Row Labels	KPIScrapRate	KPIScrapRate Goal	KPIScrapRate Status	KPIScrapRate Trend
2011				
Bikes	0.001276976	0.0015	△	↑
Components	0.001901493	0.002	△	↑
2012				
Bikes	0.002251728	0.0015	◆	↑
Components	0.001923239	0.002	△	↑
2013				
Bikes	0.001552961	0.0015	◆	↓
Components	0.001788609	0.002	△	↓
2014				
Bikes	0.001146849	0.0015	△	↓
Components	0.002373643	0.002	◆	↑
Grand Total	**0.001939942**	**0.001**	◆	↑

Scrap rate KPI

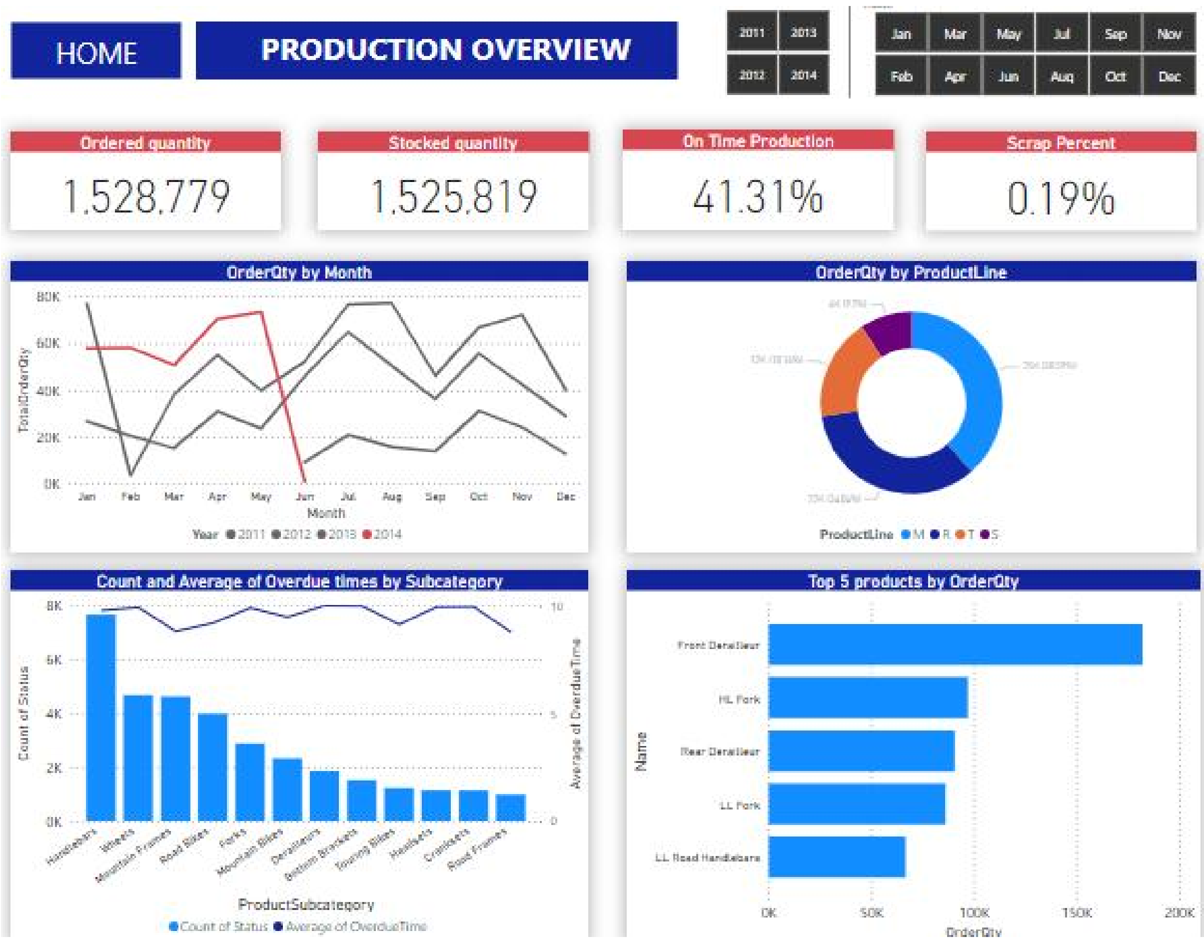
According to the table above, we see that the product category "Bikes and Components only has yellow and red status, which means that the KPI is not met. Although not meeting the KPI as in 2011, this is a good sign because the actual ScrapRate rate is lower than the ScrapRateGoal KPI, which means that the percentage of broken products appearing is less than expected. However, in 2012, the ScrapRateTrend KPI increased, and the actual ScrapRate of both categories also increased, but the ScrapRate of Bikes exceeded the ScrapRateGoal, i.e., the percentage of damaged goods was higher than expected. This year, the ScrapRate of Components also increased but did not exceed ScrapRateGoal, so it still shows yellow. In 2013, the ScrapRateTrend in two categories began to decline, but the ScrapRate of Bikes continued to exceed the set scrap rate goal. By 2014, the ScrapRateTrend of Bikes decreased, resulting in the ScrapRate of Bikes also falling lower than that of ScrapRateGoal, the error rate during product production in the Bikes category was lower than expected. However, with category Components, ScrapRateTrend increases, ScrapRate shows up in red, which means that the ScrapRateGoal KPI is not reached and in fact exceeded.

CHAPTER 6: VISUALIZATION AND FORECASTING OR PREDICTIVE MODEL

6.1. Data analysis with Power BI

6.1.1 Production

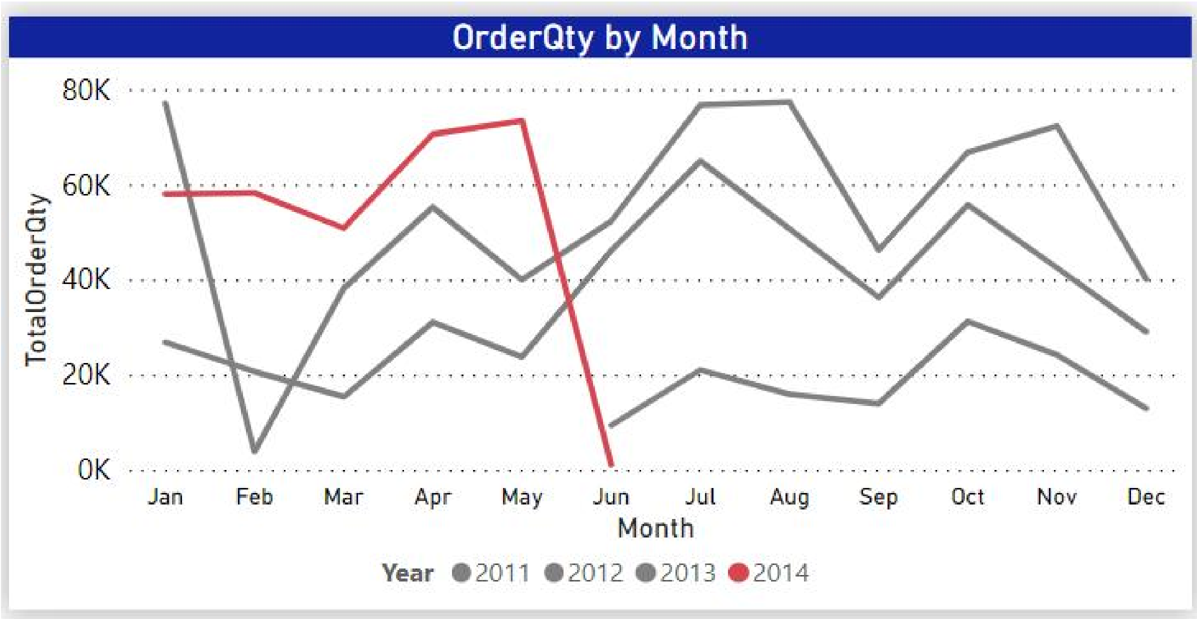
6.1.1.1 Production overview



Production overview dashboard

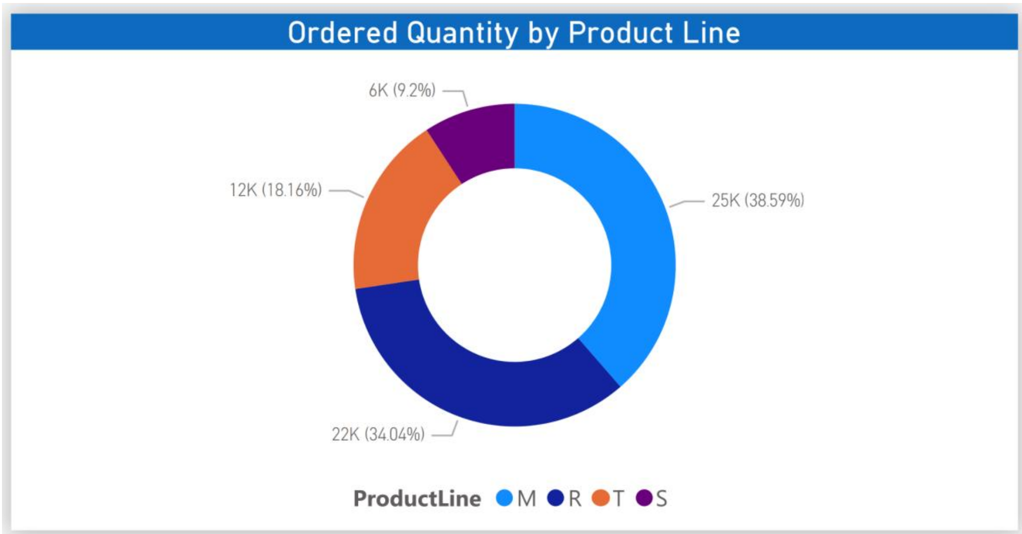
This dashboard shows a general overview about the production status. It consists of four charts: ordered quantity trend, ordered quantity by product line, average of overdue time by product subcategory, and top 5 ordered products. Moreover, we have slicers to let users filter by year and month.

The first section of dashboard are 4 card visuals help the company evaluate operational efficiency of production. As we can see in the dashboard, ordered quantity and stocked quantity are approximately equal to each other, and the scrap rate is quite small (about 0.19%). On the other hand, the percent of time when production goals were met is quite low, just about 41.31%. The reason for this could be the planning is poor, or the work performance is not good. So to mitigate this problem, company should consider some ways to increase productivity at manufacturing such as: Update processes and technology, commit to scheduled maintenance, train and educate employees or organize the workspace,....



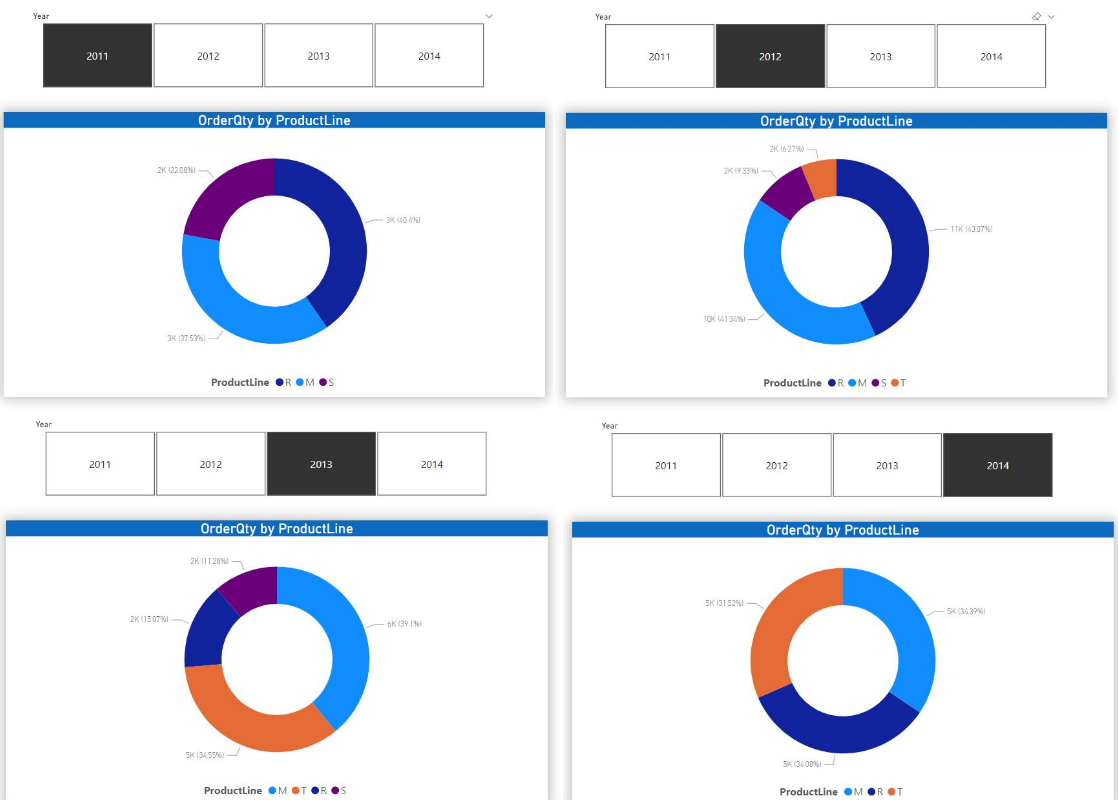
Line chart about ordered quantity

The chart illustrates the total of ordered quantity over the years, which helps managers monitor and compare ordered quantity of the recent year (2014) with the previous years (2011, 2012, 2013), which helps managers monitor and compare ordered quantity of the recent year (2014) with the previous years (2011, 2012, 2013). This chart also show for the managers that order quantity is often high in January, April, July and October while it will be low in February, June, September and December. Based on that, we can predict the trend of or ordered quantity will increase rapidly in the first month of quarter and decrease in the next month, but after that it will increase slightly in the last month. Following this trend, production department will build production planning that maintain amount of product in stock at safety line and in the last month of quarter, we need to promote production.



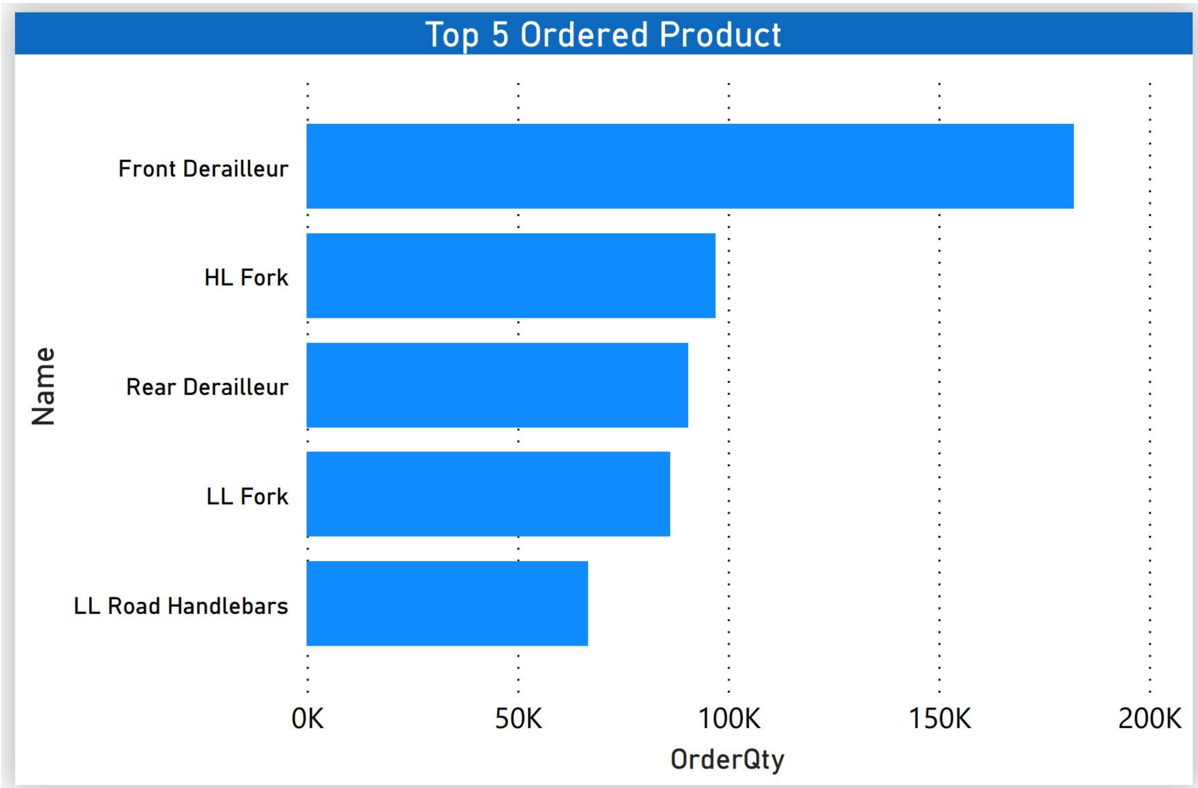
Ordered quantity by product line chart

The chart illustrates ordered quantity by product line, including: R = Road, M = Mountain, T = Touring, S = Standard. According to the chart, road was the largest line and standard was the least narrow line. In all types of product line there was only minimal change from 2012 to 2013. In 2011 and 2014, standard line was not ordered.



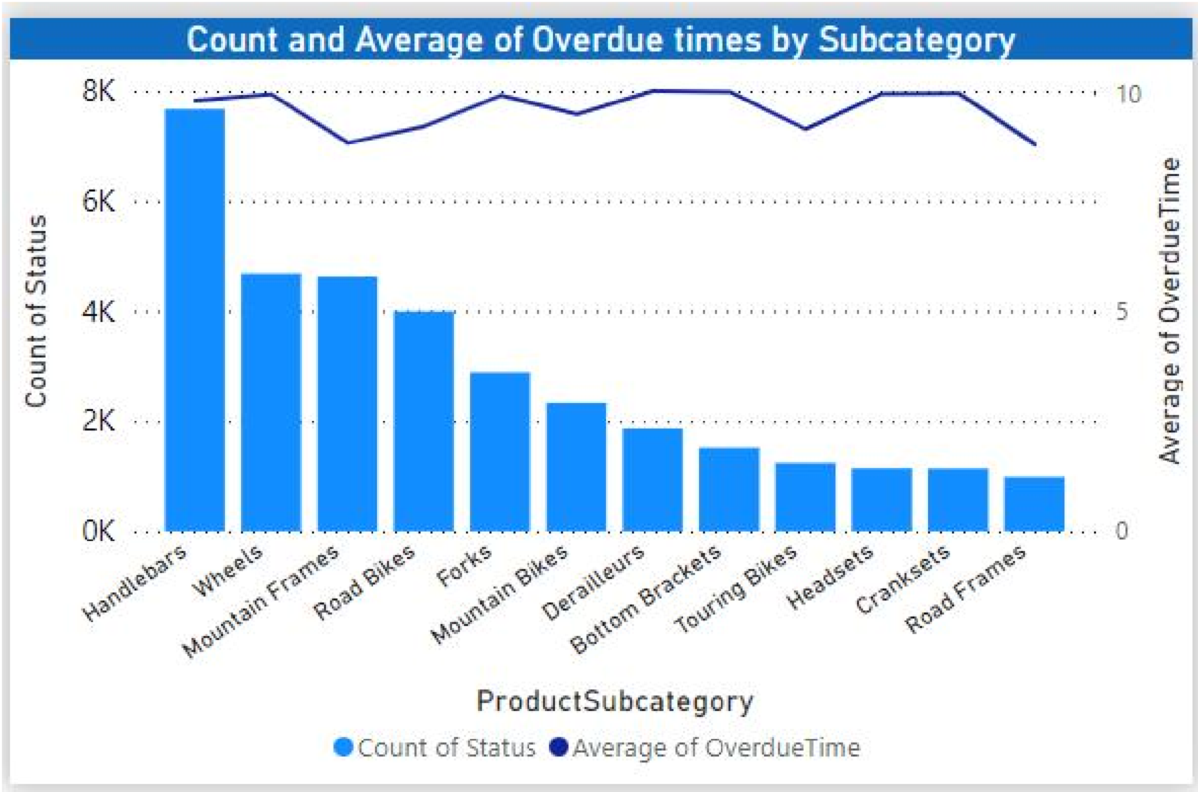
Ordered quantity by product line from 2011 to 2014 report

Based on this report, touring line increased enormously from 2012 to 2014. Road line and mountain line changed marginally. In 2014, all lines were approximately equal. This remaining maybe a good point for balance on product. However, we can not predict whether the demand for order will outperform or decline.



Top 5 ordered products chart

This chart shows the top 5 popular products including: Front Derailleur, HL Fork, Rear Derailleur, LL Fork and LL Road Handlebars. Front Derailleur is the highest consumed product at 182,162 items. Over 4 years, Front Derailleur was the most popular product. So, we need to promote production of this product.



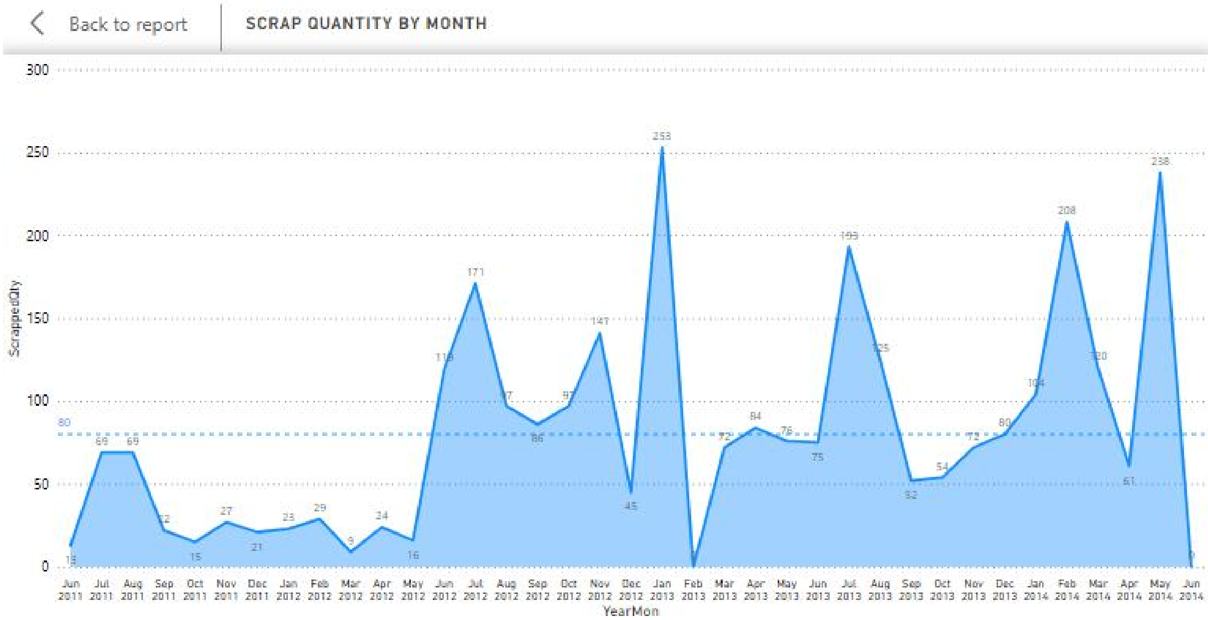
Overdue time by product subcategory chart

The chart displays overdue time of product subcategories. Monitoring the overdue times of products can support production department to evaluate planning process and improve operational efficiency. According to that chart, Handlebars was the item that are often overdue (about 7671 times). Handlebars also was the third ordered product so production department must to improve this issue by upgrading the production system. Average of overdue time was from 8 to 10 days and only minimal change between different subcategories. The problems about missing deadline in production can come from planning process estimated manufacturing capabilities in an incorrect way and production system worked inefficiently.

6.2.1.2 Scrap details

Scrap details dashboard

The first line chart of the dashboard gives us an overview of the amount of scrap over each month/year. In addition, we also combine a line of average scrap quantity of the production department, this will help us to assess whether the production situation (scrap quantity) is good or not.



Scrap quantity by Month

Scrap quantity has its increase and decreases depending on the socio-economic situation of that time. However, in general, scrap quantity increases in the 3rd and 4th quarters of each year. This can be influenced by the number of production orders of the company.

One solution given from the charts results is that the company should focus on the working hours of machines and workers, especially during the peak production time of the 3rd and 4th quarters every year. Since production quantity is directly proportional to scrap quantity, if we want to reduce scrap quantity, we need to focus on fixing and reducing the reasons.

After analyzing the scrap quantity over time, we proceed to find out the top 5 products with the most scrap quantity as well as the top 5 scrap reasons. Attached to these quantities is the corresponding cost. The combination of line and column charts will make it easy to visualize how the amount of scrap will affect the actual costs that the company will incur.

Top 5 products have the most scrap quantity

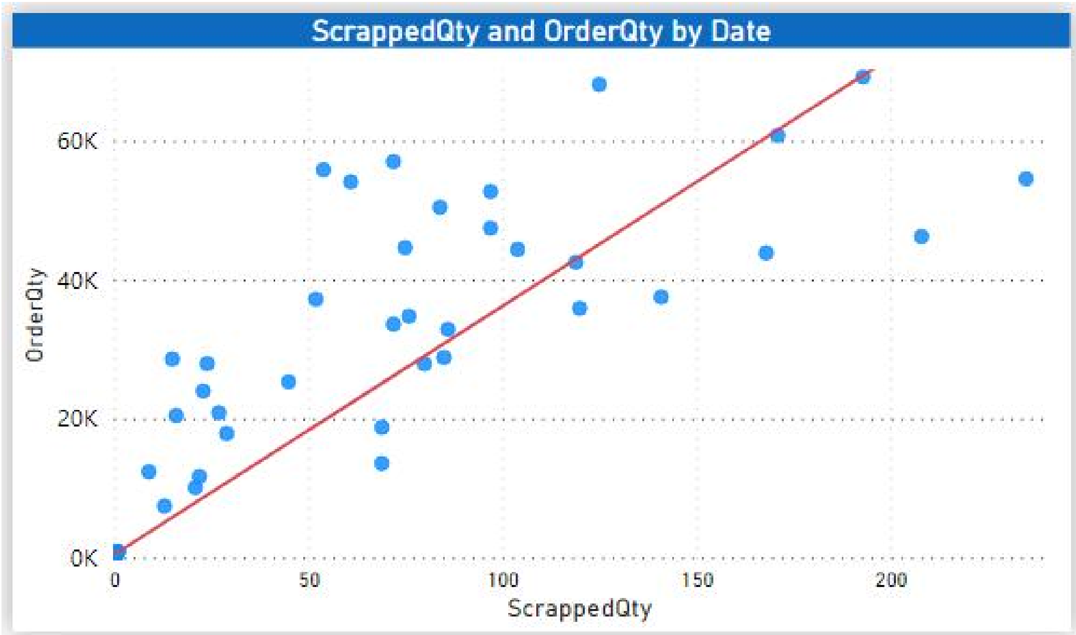
The top 5 products with the most scrap quantity belong to the product category of Component. The chart shows that each scrap quantity of different products can entail different cost ratios. For example, the product "Front Derailleur" has a higher scrap quantity than "ML Crankset" but in terms of cost it is lower than "ML Crankset".

From that, we want to conclude that the production department needs to pay attention to improving the quality of products that affect actual production costs, such as "ML Crankset" (because it is not the top 1 about scrap quantity but generates much higher scrap costs), as well as products "LL Fork", "Front Derailleur",…

Top 5 scrap reasons

The top 5 reasons for the situation of scrap include: leading in quantity is the reason "Trim length too long" (26.65% of the total), the following 4 reasons have the same amount in total (approx. 15 – 20%) including "Thermoform temperature too low/high", "Color incorrect" and "Brake assembly not as order".

Besides considering the quantity, the cost for these reasons should also be taken into consideration, from the results of the chart, we suggest that Adventureworks should pay careful attention to the production process for many items. Reasons for large cost loss such as "Color incorrect" and "trim length too long", etc. need to be given the most optimal solution (this strategy needs to consider aspects such as operation sequence, production time, expected cost, nature of scrap reasons,...).

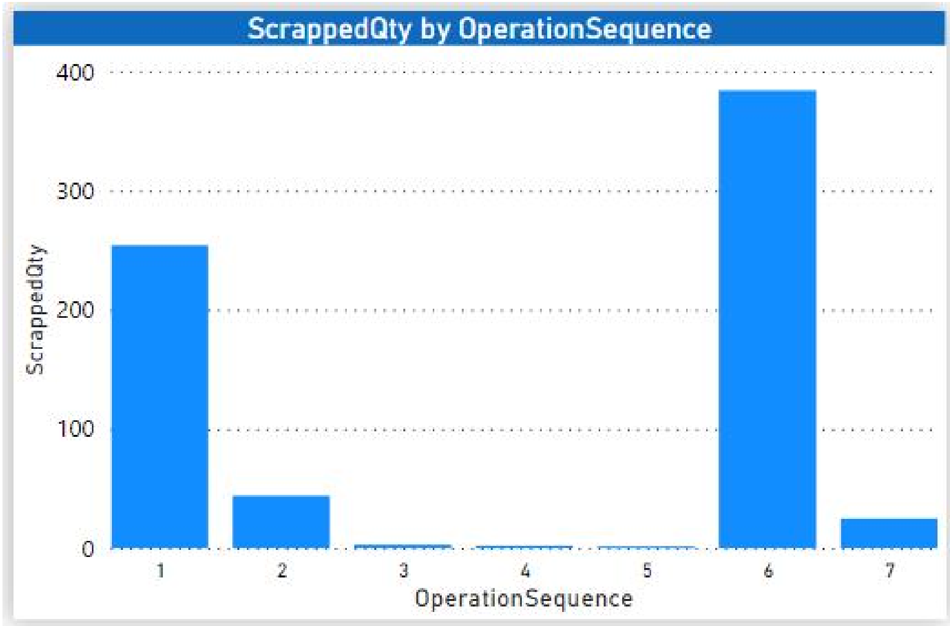


The correlation between Scrapped quantity and Order quantity by Date

This is a scatter plot, it shows the correlation between the two variables Scrap quantity and Order quantity. In addition to observing the dots (dispersion point, intersection point between 2 variables) to see the correlation, we can observe the trend line (red line).

This means that when the Order quantity increases, the Scrapped quantity also increases.

Besides analyzing the influence of product quantity as well as scrap reasons, we realize that it is necessary to study the production process. Specifically, the operation sequence and the company's monthly production cycle.



Scrap quantity by Operation sequences

This column chart shows the total scrap quantity of each operation sequence. We can see that sequence 6 accounts for the majority of the volume with 1827 scrap products. Next is sequence 1 with 567 products. And sequences 2 and 7 range from 168 to 179, while sequences 3, 4, and 5 still have the lowest level of fewer than 20 products. Since the production quantity at each sequence is also proportional to the scrap quantity, when the production quantity is large, the scrap quantity is also large. Looking at the chart, we can understand that sequence 6 is the key sequence because it has to work continuously, but the scrap quantity is also quite a lot, this can affect the revenue of each product related to this sequence.

It can be understood that sequence 6 is the place to focus on producing the "Component" product category, so with a large amount of order quantity, the scrap quantity will certainly increase. Besides, scrap reasons for this number are also quite diverse and account for almost 80% of the number of top 5 reasons.

One thing that we want to recommend to AW company is to focus on fixing (or limiting) the reasons that lead to scrap (top 5 reasons), every product that is reduced in scrap quantity will also decrease the waste costs. Not only sequence 6 but also other operation sequences.

6.2.1.3 Cost Details

The Cost Details page consists of 4 charts that show an in-depth analysis of the production cost and help determine specific issues.

Dashboard about Cost Detail

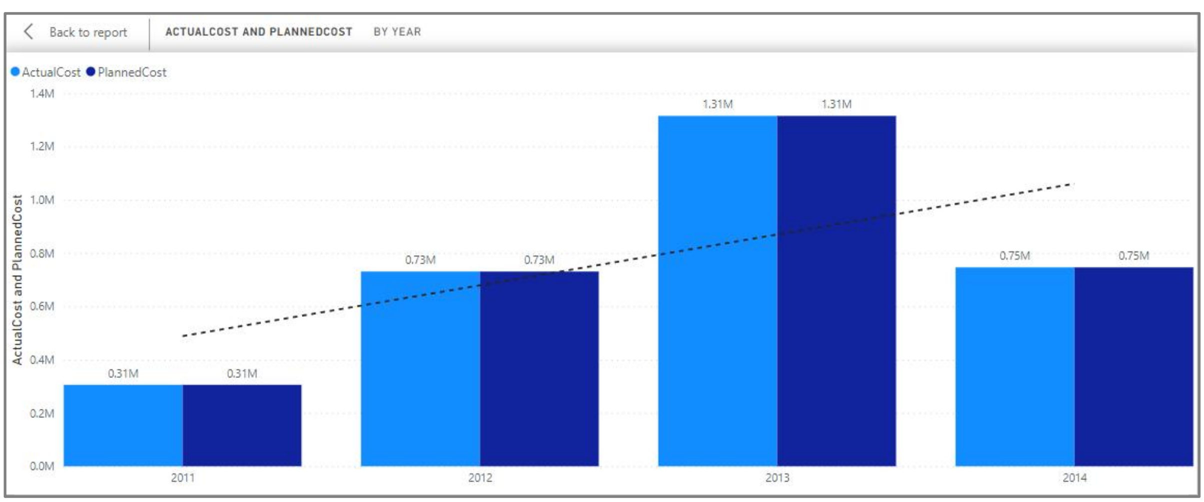
Scap Cost - Product Matrix

Scrap Cost - Product Matrix

Name	Bikes	Components	Total
Trim length too long	17,291.64	27,588.69	**44,880.33**
Color incorrect	11,419.23	29,465.86	**40,885.09**
Thermoform temperature too low	10,478.80	25,648.37	**36,127.17**
Primer process failed	1,597.79	29,438.65	**31,036.44**
Thermoform temperature too high	8,869.47	21,576.34	**30,445.81**
Gouge in metal	15,827.28	14,174.60	**30,001.88**
Brake assembly not as ordered	4,046.54	24,379.55	**28,426.09**
Wheel misaligned	6,828.93	20,728.90	**27,557.83**
Drill size too large	18,595.65	8,392.26	**26,987.91**
Drill pattern incorrect	3,734.60	22,436.47	**26,171.07**
Drill size too small	11,969.13	10,039.15	**22,008.28**
Stress test failed	12,420.51	8,192.38	**20,612.89**
Trim length too short	5,840.91	14,669.47	**20,510.38**
Paint process failed	4,480.03	12,432.13	**16,912.16**
Handling damage	3,755.94	11,210.46	**14,966.40**
Seat assembly not as ordered	2,433.55	11,557.83	**13,991.38**
Total	**139,590.00**	**291,931.11**	**431,521.11**

Matrix table of scrap cost for product category

The Matrix table shows the reason where exactly the waste is costing money to the company and due to which reasons. The first column provides the reason for scrap reason, while the other two columns are divided into two categories Bikes and Components. The cost is conditionally formatted showing which portion is costing more and the reason for it. There are subtotals on rows, columns and grand total for waste money.



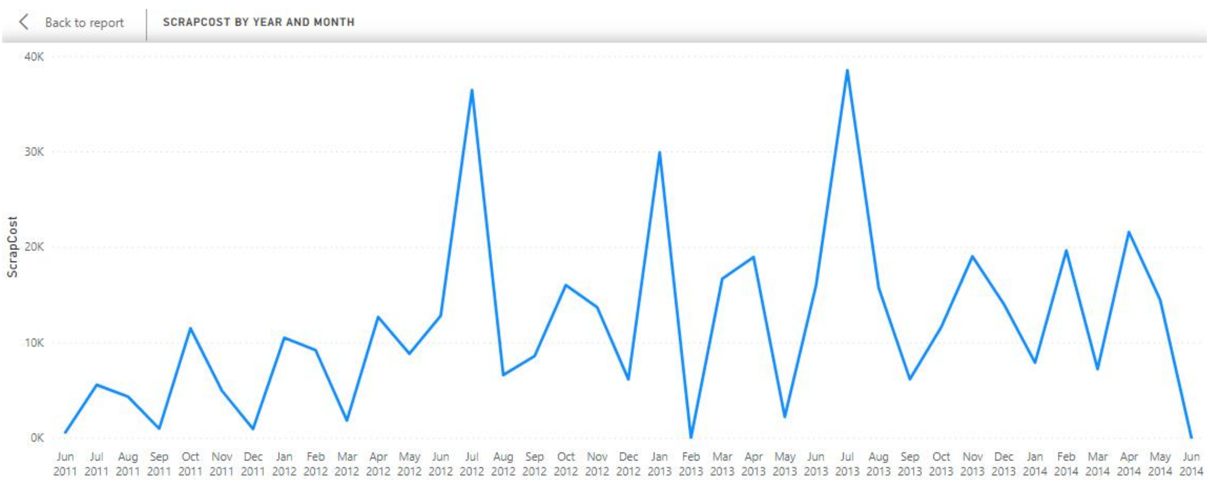
Actual Cost and Planned Cost by Year

Line and clustered column chart about actual and planned cost by year

With this chart, we can see that the actual cost is equivalent to the planned cost. There are two hypothesis here:

- Hypothesis 1: The company does very well in financial planning so that cost variance always equals 0, which means the company is never over or under the budget. That is a very good sign so the company should continue to keep up the good work.
- Hypothesis 2: For some reasons, the data of company is not reflecting the right problems. Because there are many products that are overdue but the cost and resource hours still constant. So the company should track cost more carefully.

Looking at the chart, we can also see the trend of cost over years. It increases about 500,000$/year. Based on that, we can predict that cost in the next 6 months of 2014 will increase about 1,000,000 if company does not have a solution to optimize their production process and reduce scrap quantity.



Line chart about scrap cost be year and month

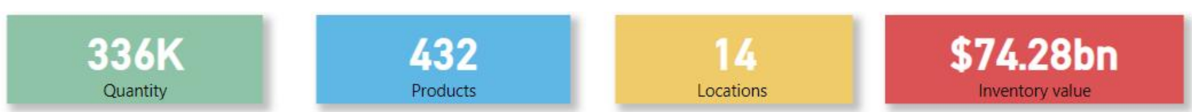
Comparing scrap cost on different years shows us a pattern. The peaks often occur in July, October, and January. Now, the company is in June, and base on the data of history, we can predict that the next month will have an increase in cost. Knowing this can help company planning better in the next month.

6.2.2 Inventory

6.2.2.1 Inventory Overview

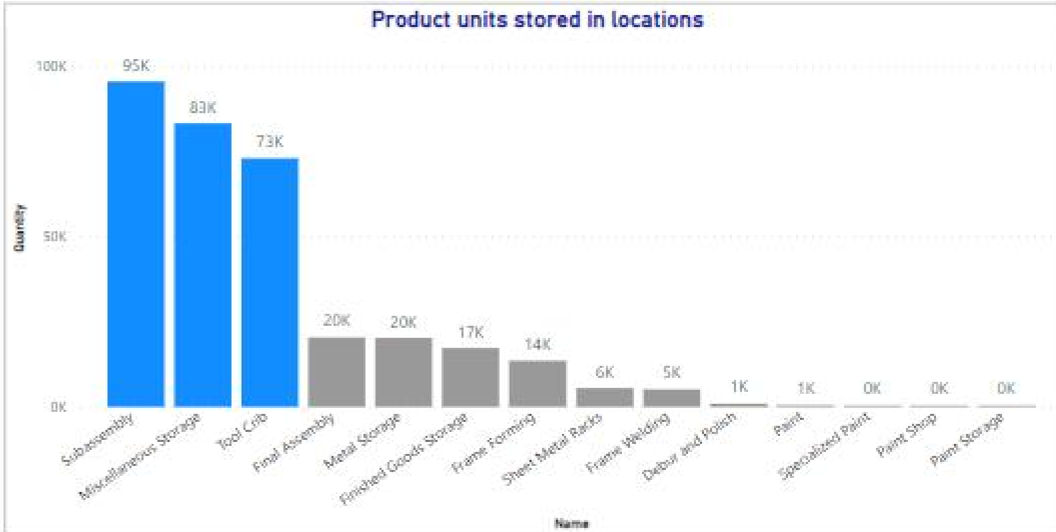
Inventory overview

With the Inventory Overview dashboard, AWs company will have an overview of the current status of inventory through cards and charts.



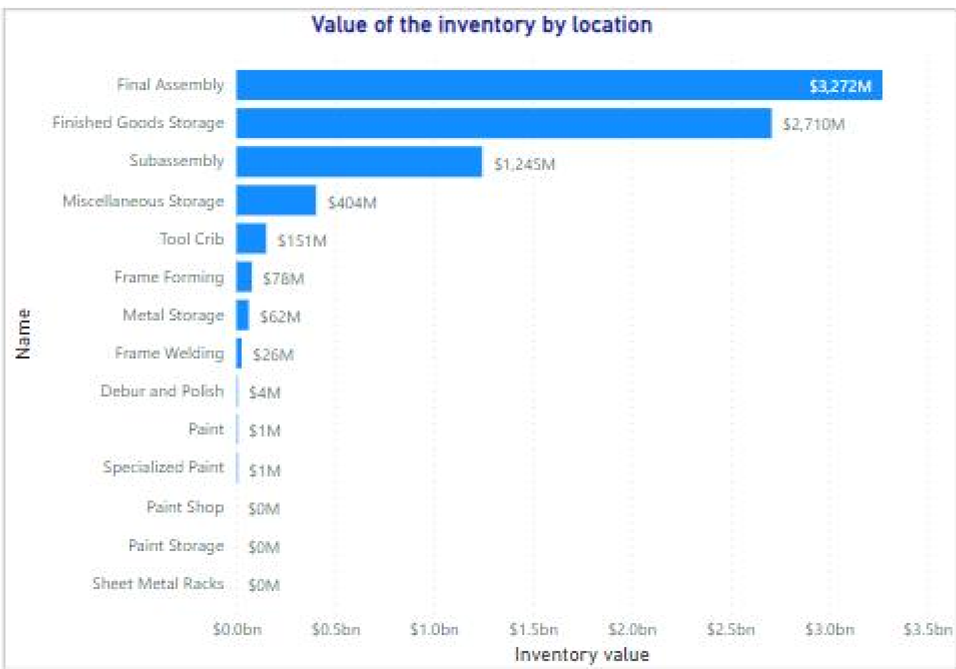
Cards Visual

The 4 cards represent Quantity, Products, Locations and Inventory Value, respectively. Through these 4 cards, we can see the quantity of the inventory, the total number of products, the total number of Locations and the value of the inventory (Inventory value) over time. In addition, with 3 filters: Year (filter parameters by time are the years from 2006-2020); Product Category (filter parameters according to each category of inventory), and Product Subcategory (filter parameters according to each subcategory of the inventory). inventory).



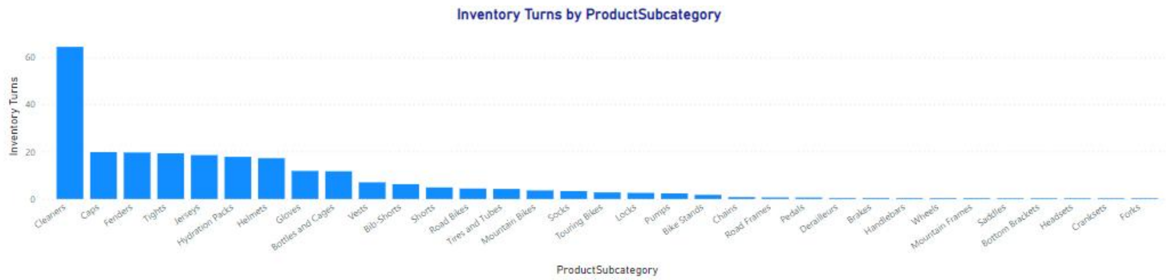
Product units stored in locations

With a clustered column chart Product units stored in locations, we will see the number of products distributed by each location.



Value of the inventory by location

The stacked bar chart Value of the inventory by location shows the value of the inventory for specific locations.



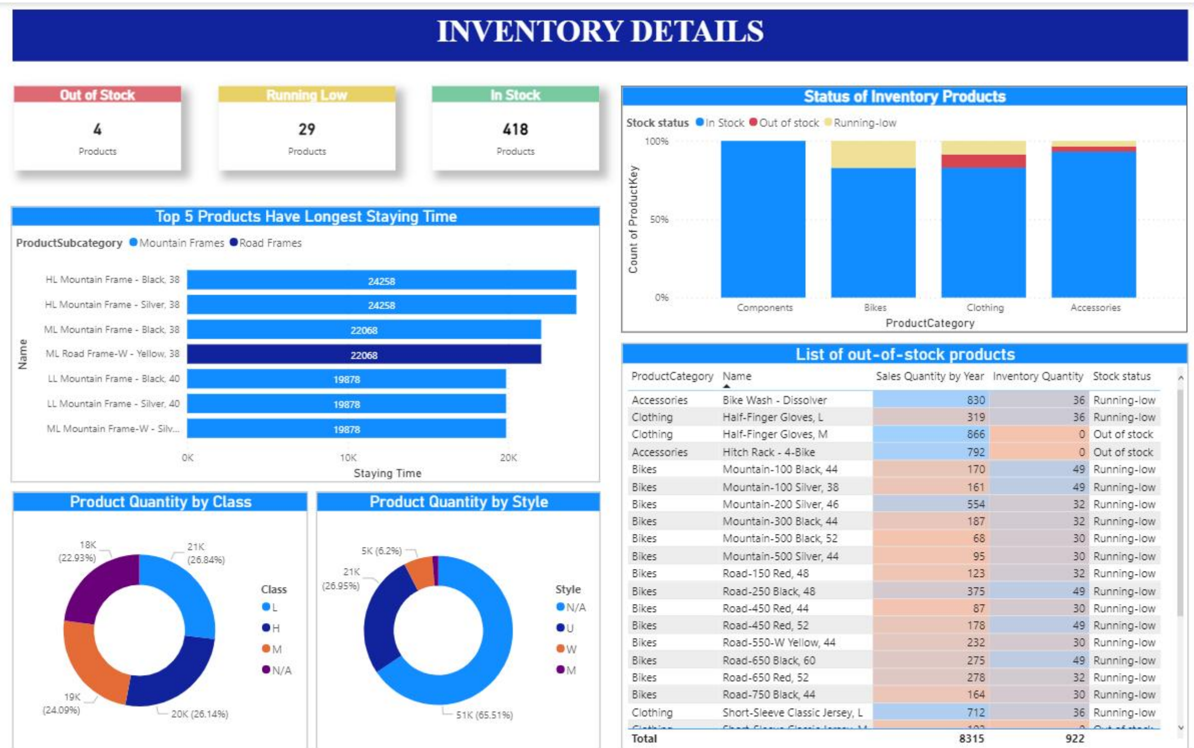
Inventory Turns by ProductSubcategory

Inventory Turns by Product Subcategory is a stacked column chart that displays inventory turnover by product category. Inventory turnover shows the frequency of product sales of the store and can assess whether a product is selling quickly or slowly, thereby making it easier to make decisions to manage the store's anti-lost goods. From the chart above, we can see that the subcategory Cleaner has the highest index, which means that the store is selling goods very quickly and demand for that item is very large. Besides, with other subcategories like Headsets, Forks, Cranksets,... have a low index, and zero means weak sales as well as low demand for the product.

With the above stacked bar chart and clustered column/bar chart, managers will have a visual view of the distribution of inventory and products according to each location, thereby being able to assess which locations are used a lot and which locations are overloaded or rarely used. Along with that, the inventory turnover chart indicator will help administrators capture trends and customer needs to adjust the purchase plan and stockpile products more reasonably.

6.2.2.2 Inventory Details

In addition to dashboard of inventory overview, one of inventory details is plotted so as to provide managers more useful information in decision making.



Dashboard of inventory details

The dashboard demonstrates detailed information and metrics, showing the state of inventory such as stock status of products, characteristics of inventory goods and which sellable products stay in inventory for an extended period of time. It is believed that the aforementioned information would help the production department to control the inventory more efficiently and wisely.

In order to give information at a glance, the above scorecards are put at the top of the page. Then, it is followed by other graphs.

List of out-of-stock products

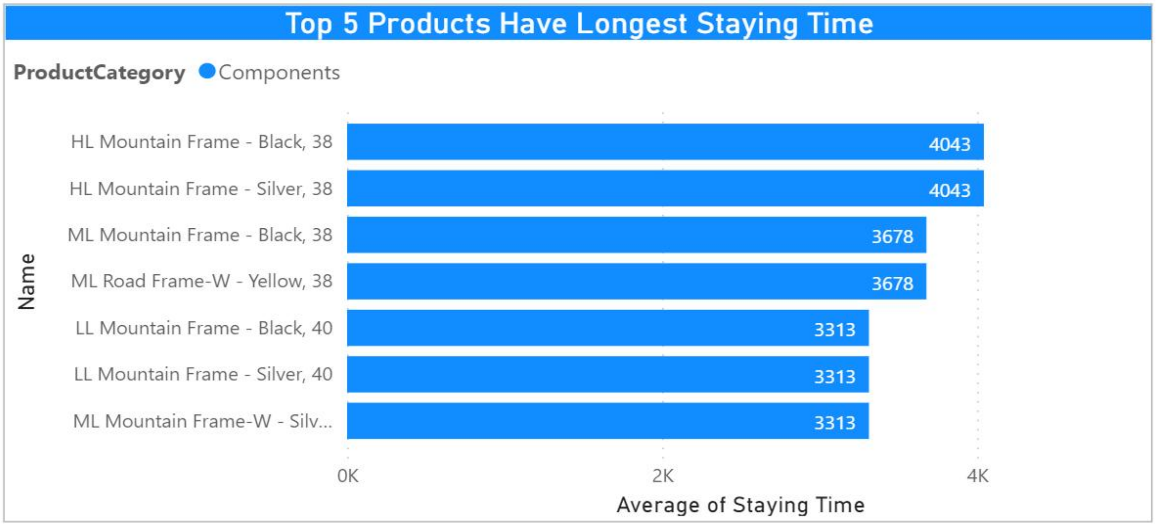
ProductCategory	Name	Sales Quantity by Year	Inventory Quantity	Stock status
Accessories	Bike Wash - Dissolver	830	36	Running-low
Clothing	Half-Finger Gloves, L	319	36	Running-low
Clothing	Half-Finger Gloves, M	866	0	Out of stock
Accessories	Hitch Rack - 4-Bike	792	0	Out of stock
Bikes	Mountain-100 Black, 44	170	49	Running-low
Bikes	Mountain-100 Silver, 38	161	49	Running-low
Bikes	Mountain-200 Silver, 46	554	32	Running-low
Bikes	Mountain-300 Black, 44	187	32	Running-low
Bikes	Mountain-500 Black, 52	68	30	Running-low
Bikes	Mountain-500 Silver, 44	95	30	Running-low
Bikes	Road-150 Red, 48	123	32	Running-low
Bikes	Road-250 Black, 48	375	49	Running-low
Bikes	Road-450 Red, 44	87	30	Running-low
Bikes	Road-450 Red, 52	178	49	Running-low
Bikes	Road-550-W Yellow, 44	232	30	Running-low
Bikes	Road-650 Black, 60	275	49	Running-low
Bikes	Road-650 Red, 52	278	32	Running-low
Bikes	Road-750 Black, 44	164	30	Running-low
Clothing	Short-Sleeve Classic Jersey, L	712	36	Running-low
Total		**8315**	**922**	

Stock status of each product category and list of low stock products

It is vital to show the ratio of stock status in every single product category. Almost all products are in stock, particularly components whose products are completely available. It is to say that component stocks are managed well or they have not been sold and have stayed in inventory for a long time. Whereas, other categories have a small amount of running-low and nonexistent products.

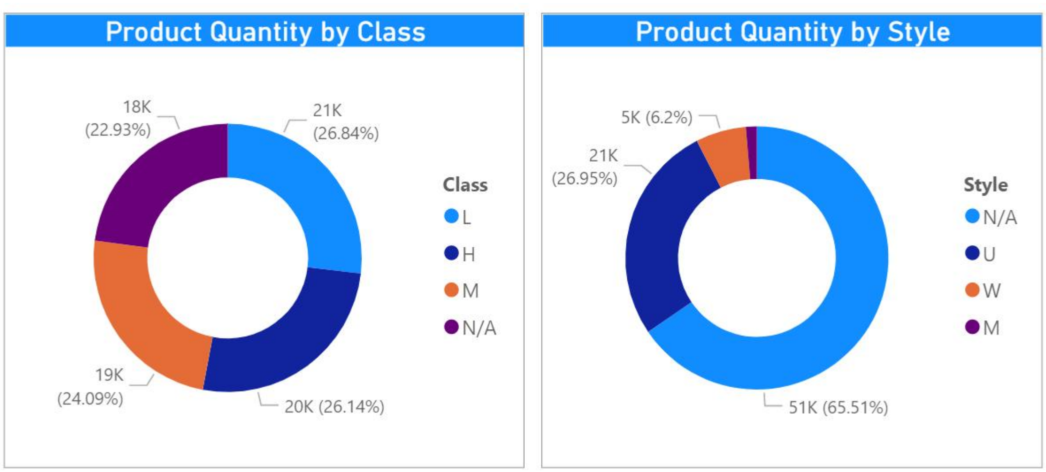
For more details, a list of products that are out of stock or running low is given so that managers can compare the average annual sales amount to the current balance of inventory to make decisions for future plans. The products with low stock and high sales should be continuously restocked, in contrast, the production department has to take into consideration renewing. For example, “Half-Finger Gloves, M” and “Hitch Rack - 4-

Bike” have positive sales but they are currently out of stock, therefore, they should be restocked to meet the market demand and not miss any opportunities. Besides, products that have low order quantity need more examination in adjusting the number of inputs.



Top 5 products staying at inventory for a long period of time

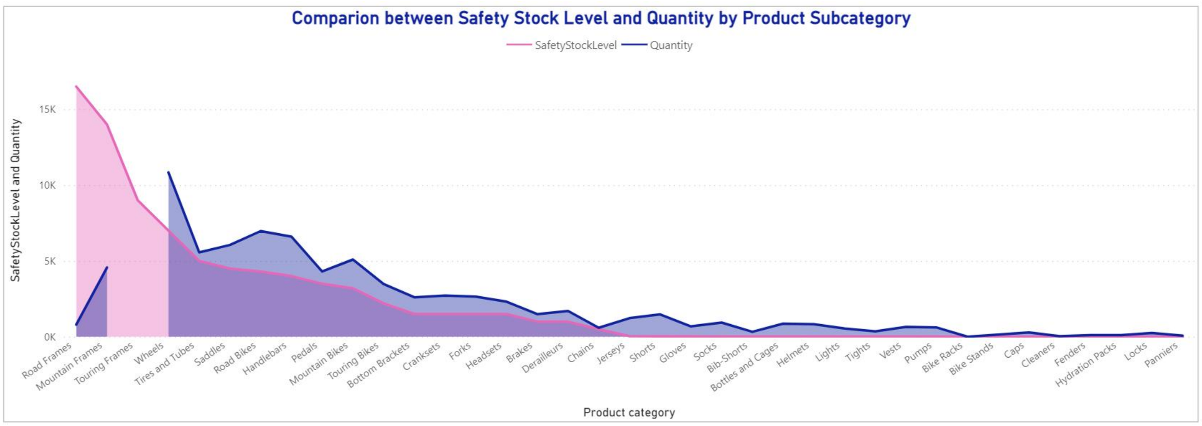
As stock can not tell if products that are available are well controlled. There is a case that products have high inventory days. In other words, these products were not turned into sales quickly. The reason could be poor sales performance or purchase of too much inventory. Having too many inefficient goods can be disadvantageous to AdventureWorks. As mentioned above, component stocks were managed well or they had not been sold and have stayed in inventory for a long time although all products in the category are in stock. Then the above figure gives the answer that the top five products having the highest inventory days are components, which reach a high of 4034 days (about 11 years) . Hence, the components were not administered properly. Those products should be boosted in sales to enhance the inventory performance.



Proportion of products in terms of class and style

The charts would deliver information of how much class and style account for in inventory. Regarding the ratio of product classes, there were no significant differences in 4 classes, which are from 0.22 to 0.268. The distribution of classes was quite balance.

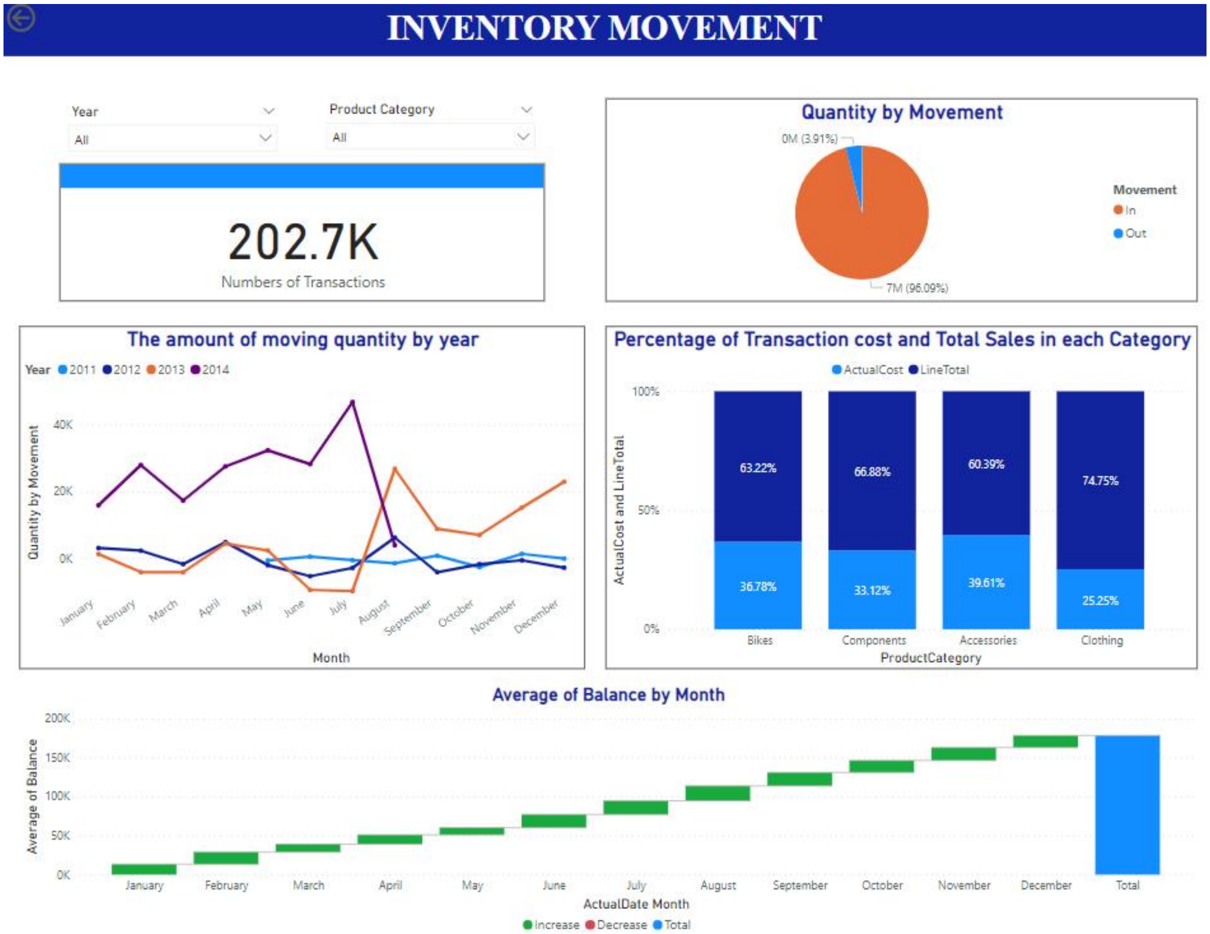
Taking a closer look at figure of style, the quantity of products whose style were undefined represented the highest percentage, followed by that of universal products, women's products in that order. The number of men products was the least product style stored in the inventory.



Comparison between safety stock level and quantity in subcategories

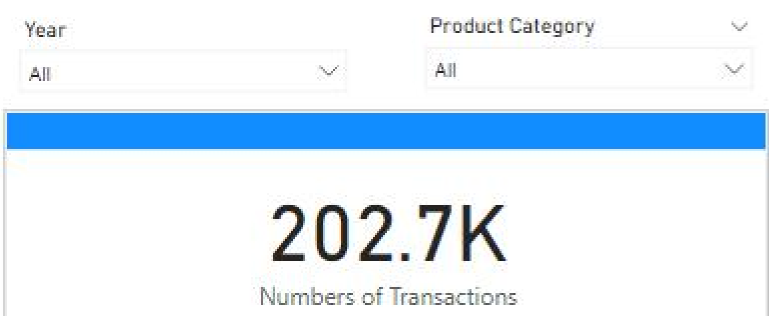
Safety stock level illustrates the minimum inventory quantity. It is to say that inventory quantity should be higher than safety stock level. From the area chart, almost subcategories stored goods in a safe level. However, some categories did not meet the safety level, having a huge gap such as road frames, mountain frames. In particular, touring frames has no quantity. These products should be planned to produce and purchase. The production department can manage the inventory better based on these metrics combined with demand of customers.

6.2.2.3 Inventory Movement



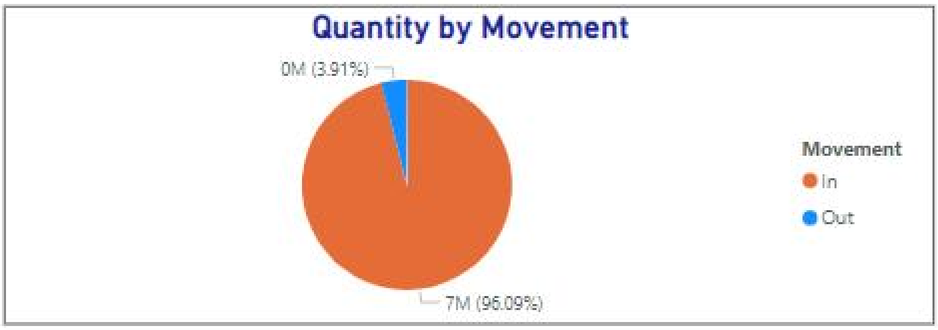
Inventory Movement Dashboard

The Inventory Movement describes the transactions that occur with the inventory by year and by Product Category.



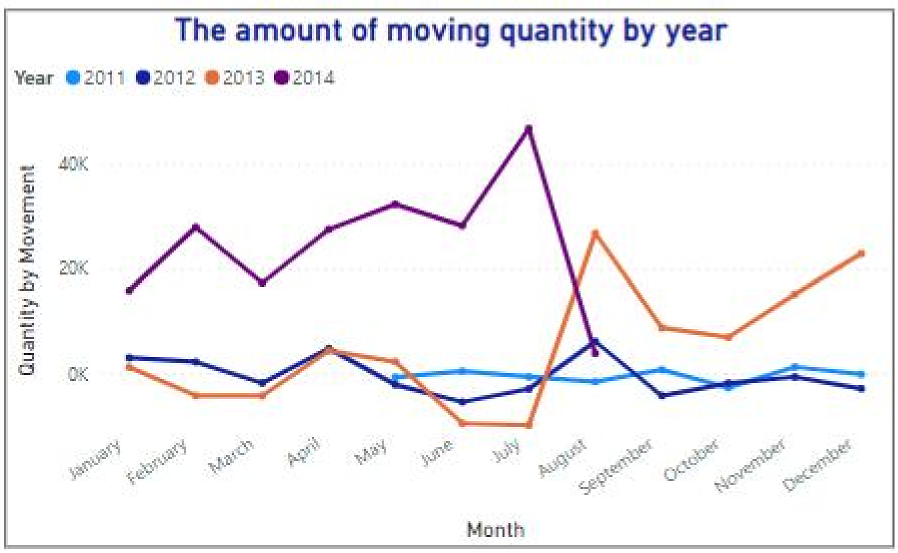
Cards Visual

With a card and a filter, users can see the number of transactions and filter data by year (Filter Year) or by product type (filter Product Category).



Quantity by Movement

The Pie chart, Quantity by Movement shows the ratio between the In and Out movement of the inventory. Movement In represents Work Order transaction, Out movement represents Purchase Order. For each Category, the movements are very different, with Clothing and Accessories having almost no In movement, with Bikes In and Out balanced, and Components with the majority of In movement. For example, the amount of In and Out can be seen to be very different. From 2011 to 2014, all transactions of Accessories and Clothing were Out movements.



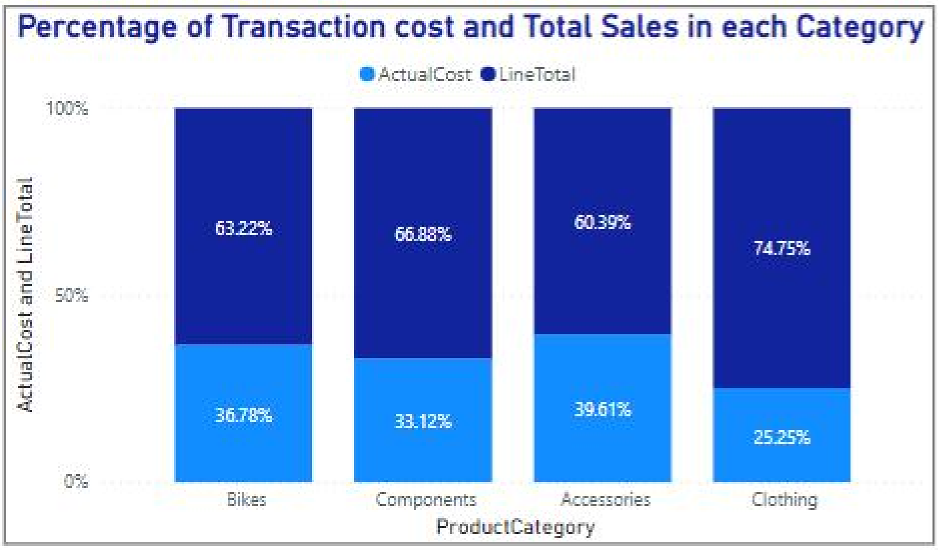
The amount of moving quantity by year

The line chart The amount of moving quantity by year shows the quantity fluctuations of inventory transactions by month of the year. Quantity by movement is sum of goods in and outs in a period. At the monthly points, if that point is above 0, the amount of goods in that is greater than the figure of goods out should show a positive value, and the "In" greater than "Out" movement volume also shows the excess in inventory compared to actual demand, and for points below the 0 mark, the amount of Out greater than In movement is equivalent to the quantity of goods in short supply compared to demand.

Notably, the numbers of moving quantity in Q1 and Q3 in 2013 experienced negative values. It is can say that:

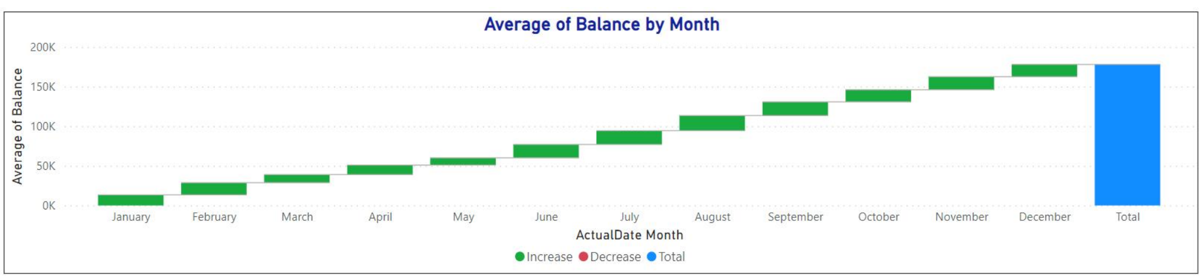
- Positive impact
- Negative impact

In addition, according to the volatility of the line chart, the administrator can predict the trend of demand combined with the number of movements, thereby making more appropriate decisions about the production of goods or purchasing more inventory.



Percentage of Transaction cost and Total Sales in each Category

The stacked column chart Percentage of Transaction cost and Total Sales in each Category shows the ratio of Actual Cost (product cost) and Line total. With this chart, we can see how much the product cost (inventory) accounts for compared to the selling price, which shows the effect of inventory and transactions inventory on total sales.



Average of Balance by Month

The waterfall chart, Average of Balance by Month shows the rise and fall of the inventory balance. It can be seen that with the average balance always increasing every month, it shows that in transactions, transaction In is always larger than transaction Out, so this index does not decrease. This proves that there is always an inventory, whether it is redundant or full.

So, from the charts in the Inventory Movement dashboard, we can predict the monthly trend of product consumption, adjust production, purchase, and sale plans, and store goods accordingly from inventory movements.

CHAPTER 7: CONCLUSION AND FUTURE WORKS

7.1. Results

Overview of the results, we have completed the project and achieved the following results:

- Identify the goals, research results, and the importance of business intelligence (BI) solution for businesses
- Description, overview analysis of Adventure Cycle Works, organizational models, data storage, processing and analysis of BI solution.
- Complete the given objectives for the BI solution, and answer the questions set out to solve the problem.
- Learn and use tools to support the development of BI solutions such as PowerBI, Visual Studio SSAS, Microsoft SQL Server Management,....
- Improve teamwork and research skills
- Responsible and positive at work
- Have professional ethics in the job

7.2. Limitations

During the implementation of the project, we still have the following limitations.

First, because of limited knowledge and time, we have not been able to perform MDX query as planned. This partially affects the outcome of the project. Although the results have not really gone into the analysis, we have also achieved certain success.

Second, due to lack of experience, the construction of data warehouse and implementation of ETL process was overtime. Spending more time on this, delays the project and shortens the execution time of the following sections.

Finally, because the scope of the project is quite wide and the difficulty of knowledge is very large, the results are not really perfect. The results of the dashboard analysis are right in focus but have not yet gone into economic operations.

7.3. Future works

First of all, we continue research on MDX query and data storytelling to more complete data processing and visualization.

Second, we research deeper into economic requirements to design flexible dashboards and meet business requirements with the goal of supporting better decision making. Moreover, we want to extend interactive features in reports and nested reports.

Finally, we will learn more about the BI solution and expand the project both horizontally and in depth.

REFERENCES

[1] SkyPoint CSG (2021). Business Intelligence Tools: A Pros and Cons Comparison Chart. Retrieved from:

https://www.csgpro.com/blog/business-intelligence-tools-comparison-chart/

[2] Ez Data Munch - Abhishek Sharma (2018). Define KPIs for Successful Business Intelligence. Retrieved from:

https://ezdatamunch.com/define-kpis-successful-business-intelligence/

[3] Red Gate Hub - Robert Sheldon (2010). Five Basic Steps for Implementing an Analysis Services Database. Retrieved from:

https://www.red-gate.com/simple-talk/databases/sql-server/bi-sql-server/five-basic-steps-for-implementing-an-analysis-services-database/

[4] Microsoft Docs (2022). Key Concepts in MDX (Analysis Services). Retrieved from:

https://docs.microsoft.com/en-us/analysis-services/multidimensional-models/mdx/key-concepts-in-mdx-analysis-services

[5] Idera. MDX - Idera Glossary. Retrieved from:

https://www.idera.com/glossary/mdx

[6] Hashmicro - Vania Marsha Kristiani (2021). Production Process in Business: Definition, Types, and Characteristics. Retrieved from:

https://www.hashmicro.com/blog/what-is-production-process/

[7] The Global Treasurer (2013). The Pros and Cons of KPIs. Retrieved from:

https://www.theglobaltreasurer.com/2013/11/19/the-pros-and-cons-of-kpis/

[8] ScienceDirect Topics. Transactional Data - an overview. Retrieved from:

https://www.sciencedirect.com/topics/computer-science/transactional-data