Database Systems

Table of Contents

Database System Overview (Week 1) 5

[Company Scenario 5](#_Toc497926946)

[Mission Statement 5](#_Toc497926947)

[Goals 6](#_Toc497926948)

[Critical Success Factors 7](#_Toc497926949)

[Quantitative and Qualitative Variables 8](#_Toc497926950)

[Alignment between the Solution and the Mission and Goal 9](#_Toc497926951)

[Entity Relationship Model (Week2) 12](#_Toc497926952)

[Subjects of Interest (proposed entities) 12](#_Toc497926953)

[Business Unit Master 12](#_Toc497926954)

[Customer Master 12](#_Toc497926955)

[Item Master 12](#_Toc497926956)

[Item Branch 12](#_Toc497926957)

[Sales Order Header and Detail File 13](#_Toc497926958)

[Business Rules 13](#_Toc497926959)

[Entity-Relationship Model 13](#_Toc497926960)

[Business rule to a Data Model component 13](#_Toc497926961)

[List of Tables and Entity Relationship 13](#_Toc497926962)

[Data Model for Sales Order Entry 18](#_Toc497926963)

[Normalization 20](#_Toc497926964)

[PT Matahari Application of ER Model 21](#_Toc497926965)

[Structured Query Language (SQL) Scripts (Week3) 23](#_Toc497926966)

[Database Definition Language (DDL) 23](#_Toc497926967)

[Database Manipulation Language (DML) 24](#_Toc497926968)

[Reports/Queries 27](#_Toc497926969)

[PT Matahari Application of DDL, DML, and Reports 29](#_Toc497926970)

[Database Administration Plan (Week4) 31](#_Toc497926971)

[Database Administration Plan 31](#_Toc497926972)

[Transaction Processing in Sales Order Entry 31](#_Toc497926973)

[Database Security Procedure 33](#_Toc497926974)

[Backup Plan and Recovery Model 34](#_Toc497926975)

[PT Matahari Application of Database Administration 36](#_Toc497926976)

[Future Database System Implementation Plan (Week5) 37](#_Toc497926977)

[Object-Oriented, Object-Relational Database System, and Web-based Database System 37](#_Toc497926978)

[Impact 37](#_Toc497926979)

[Data Warehouse Implementation 38](#_Toc497926980)

[Distributed Database Considerations 39](#_Toc497926981)

[Types of Business Intelligence from the Database 39](#_Toc497926982)

[Decision Making Based on BI 40](#_Toc497926983)

[Benefits of Data Warehousing 40](#_Toc497926984)

[ROI on BI Initiatives 41](#_Toc497926985)

[Competitive Advantage 41](#_Toc497926986)

[Productivity 41](#_Toc497926987)

[Data Warehouse Challenges 41](#_Toc497926988)

[Data Collection 41](#_Toc497926989)

[Resources 42](#_Toc497926990)

[Hidden problem 42](#_Toc497926991)

[PT Matahari Application of Future Database Systems 42](#_Toc497926992)

[References 44](#_Toc497926993)

# Database System Overview

## Company Scenario

PT Paint Matahari (Matahari) sells many types of paint and paint products in Indonesia to meet rising demand in both residential and commercial buildings, which Nippon Paint Japan supplies. Recently city council in Malang mandates to paint all commercial buildings every three years to cope with stain caused by tropical weather. At the same time, people spend more money for painting because the disposable income went up in recent years. Naturally, new paint shops enter the paint market, and they target same consumer groups. To survive the tougher completion, Matahari signed a new agreement with Nippon Paint to produce paints in Malang. It is necessary steps to make this retain paint shop more sustainable. Meanwhile, as long-term expansion plan, Matahari seeks for a possible opportunity in the area of interior design because most of the demand for paint is coming from the interior design companies which provide labor, transportation, and other building materials.

Many advised Matahari to implement modern database system to cope with the various expectation from stakeholders. Currently, Matahari maintains the information of business parties and daily transactions using Microsoft Access which is difficult for any staffs to search and update. So Matahari expects that the result database system runs with good performance and friendly user interface. And this system should meet the availability, reliability, and scalability in the long run.

## Mission Statement

When planning database system, the first step is defining the mission statement for the project to describe the purpose and goal of it (Connolly & Begg, 2015). So this mission statement is to answer what is the purpose of Matahari, why Matahari needs a database system, and will this database system meet the Matahari’s requirement. Hence, missing statement can represent as below,

*To enhance better customer satisfaction through fast, real-time, accurate inventory management with sophisticated pricing, and to plan and forecast the demand and supply more realistic based on the transaction history.*

## Goals

The goal of the Matahari database system is to have a competitive edge against current and future competitors using timely planning and forecasting and to give better customer experience. In turn, this data system enriches the relationship between business parties and foundation to expand business itself. The primary Matahari’s goal for database system projects is to make the business more profitable by giving better customer service and analyzing daily business through timely and accurate data. To meet these objectives of Matahari which aims for,

* To computerize business parties information – supplier, customers, employees, agents, and brokers
* To manage all item and inventories in the system after standardizing it with visual catalog
* To record all transactions including purchase, sales, inventory, and commission information
* To enable to have inventory availability checking real-time basis
* To have customer retainage rate over 90% with better customer service and discount
* To increase the sales volume 20% for commercial use

## Critical Success Factors

Table 1-1 below is to demonstrate the critical success factors to meet the goals we set above,

**Table 1-1**

*Goals and Success Critical Factors*

|  |  |
| --- | --- |
| **Objective (Goal)** | **Critical Success Factor** |
| To computerize business parties information – supplier, customers, employees, agents, and brokers | To develop software application and its storage for Address Book, Customer Master, Supplier Master |
| To manage all item and inventories in the system after standardizing it with visual catalog | To create Item Master with image repository with columns to store cross-reference information |
| To record all transactions including purchase, sales, inventory, and commission information | To build Sales Order, Procurement, and Inventory Management system with tables for Sales, Procurement, and Inventory ledger |
| To enable to have inventory availability checking real-time basis | To record Item Location file which records as-is available quantity in various steps |
| To have customer retainage rate over 90% with better customer service and discount | To build sophisticated pricing structure with database tool or using software application |
| To increase the sales volume 20% for commercial use | To promote the product using reliable data and comparison with competitors |

*Note*: This project excludes the Manufacturing of paint material.

Based on above mission and goal which repeat the business case for Matahari, here list up critical success factors which are crucial to make this database system project justified.

* Review mission and goal – to set proper business process and requirement
* Strong project management and resource commitment – to make this project roll
* Support from top management – to have financial and moral support
* Detailed plan including training
* In detail, the result database system performs with good performance and friendly user interface. And this system should meet the availability, reliability, and scalability.

## Quantitative and Qualitative Variables

Likewise another project, the success measure can be the timeline of schedule, the scope of project, budget, and quality of work (Inc, n.d.). The Same measurement can be applied to Matahari’s database system implementation project using both quantitative and qualitative measurement. Some of the qualitative measurement can be what is the look-and-feel from the users, does a new system increases the productivity at work, does system error-prone, is proper security measures in place, and do business partners like the system. On the other hand, quantitative measurement can be how quick the database response, is there any congestion, the size of the database, the number of transactions, and the utilization of database system resources. Detail represents Table 1-2,

**Table 1-2**

*Determining Quantitative and Qualitative Variables*

|  |  |  |
| --- | --- | --- |
| **Objective (Goal)** | **Measure Indicator**  **(Quantitative/Qualitative)** | **Source** |
| To computerize business parties – supplier, customers, employees, agents, and brokers | *Quantitative*: A centralized data repository record all business entities.  *Qualitative*: Users satisfaction through reliable and fast search using various input value | Customer Master, Supplier Master, Employee Master (Database)  Users’ adaptability with proper user interface |
| To manage all item and inventories in the system after standardizing it with visual catalog | *Quantitative*: Online availability is feasible  *Qualitative*: Accurate and timely inventory commitment | Item master, Item Ledger, and Item Location (Database)  Customer satisfaction survey |
| To record all transactions including purchase, sales, inventory, and commission information | *Quantitative*: The number of applications enables user to enter transactions  *Qualitative*: Timely response to the customers’ request | Sales Order Entry, Purchase Order Entry, Inventory Management (Database)  Customer satisfaction survey |
| To enable to have inventory availability checking real-time basis | *Quantitative*: The accuracy of system and physical stock (cycle count)  *Qualitative*: Accuracy of online inventory | Item Ledger vs. Item Location Report (Database)  Integrity reports (Database) |
| To have customer retainage rate over 90% with better customer service and discount | *Quantitative*: Count actual transaction using Sales Ledger  *Qualitative*: Customer service satisfaction survey | Customer Master and Customer Ledger (Database)  Customer satisfaction survey |
| To increase the sales volume 20% for commercial use | *Quantitative*: Analyze the total value based on Customer Ledger  *Qualitative*: Customer satisfaction survey | Customer Ledger (Database)  Customer satisfaction survey |

## Alignment between the Solution and the Mission and Goal

The database system implementation for Matahari highly rely on the proper steps and activities as below (Connolly & Begg, 2015),

* 1. *Panning*: to proceed the project most efficiently and effectively through management decision. This stage defined the goals and objectives.
  2. *System definition*: the scope and boundary including the number of tables, columns, user-interfaces. At this stage, define the technical specification.
  3. *Requirement Collection*: mission detail above based on the consensus of stakeholders; management and end-users including sales and customer service personnel.
  4. *Analysis*: to meet the system requirement made based on functional specification and technical specification.
  5. *Design*: how to access/store data based on requirement above with simple and beautiful user interface.
  6. *Prototyping*: conference room pilot (CRP) is a common way to perform in the implementation phase.
  7. *Implementation*: the actual system construction based on the specification. The functional specification which is a requirement and technical specification on how to execute it.
  8. *Testing*: test thoughtfully all systems and feedback to the implementer before go-live to minimize the maintenance efforts. Especially when the system is implemented using 3rd party software vendor, this stage determines the success of implementation. Testers are to document all functionalities and steps including missing basic features.
  9. *Maintenance*: the phase includes the change in internal and external requirements and preferences. New or enhanced needs are incorporated into base functionality when needed.

All steps of the database system development lifecycle are important to implement a database system for Matahari successfully. Hence, all stakeholders to learn an overview of this development lifecycle to cooperate and collaborate with each other because this implementation leads higher market share and leaves room for business expansion beyond Malang Indonesia.

# Entity Relationship Model

## Subjects of Interest (proposed entities)

A sales order needs information on the customer, item, credit history, item cost, item price, and on-hand quantity information. When building simpler business rule, basic entries comprise below,

### Business Unit Master

This entity is to hold information the characteristics of cost and profit center in expanding the business. And the income statement represents per business unit. This unit is more for the financial transaction. On the other hand, there can be Branch/Plant Constant file to indicate or maintain inventory at this level. All business units including warehouse belong to the company. In detail, this business unit master is a child of company constant which determines the unit for the balance sheet, or the destination of assets.

### Customer Master

This table holds information of customers which belong to address book master. This file includes the credit limit, payment terms, and other pricing rules. In detail, the customer master belongs to Address Book master that holds all stakeholders including, company, employee, customer, supplier, and other types of service providers with different search type.

### Item Master

This entity holds all stock and non-stock inventories including detail specification of the item to sell and purchase.

### Item Branch

This entity contains Item Master by Branch/Plant constant to hold different sales policy per shops. So availability, item cost, and item price can have different values.

### Sales Order Header and Detail File

Sales Order Header file to store information on customer and its rule including business dates where detail file contains information on items to sell.

## Business Rules

Some of the business rules are defined to maximize the customer’s satisfaction and to keep minimum level of inventory as below,

* Sales order header comprises business unit (cost/profit center), the customer (Sold-To), Ship To, dates, tax information based on Ship To, and payment terms.
* Sales order detail contains branch/plant (which is a business unit) which indicates the source of inventory, item, quantity, and another item related information
* The relationship between sales order header and detail is 1 to many

## Entity-Relationship Model

### Business rule to a Data Model component

*Each business unit includes multiple customers, but each customer can have only one business unit which is a cost center or warehouse.*



Business Unit



Customer



*One Sales Order Header which store customer’s information can have multiple Sales Order Detail which to hold inventory related information.*



Sales Order Header



Sales Order Detail



### 

### List of Tables and Entity Relationship

For easier understanding and for allowing future extension, the name of tables will be,

It is good to implement the table naming – starts with F (File), the first 2 digits indicate system code (00 for global constant, 01 for Address Book, 03 for Account Receivable (AR), 04 for Account Payable (AP), 41 for Inventory, and 42 for Sales Order), and the last digits describe the contents of data (00 for controlling data, 01 for master file, 02 for balance file, 11 for transaction file, 19 for ledger file).

Based on above naming convention businessUnit table (F0006) has multiple customerMaster (F0301) records,

businessUnit (F0006)

**VARCHAR (12)**

**BusinessUnit**

**PK**

Description

VARCHAR**(30)**

Company

CHAR (5)

Language

CHAR(2)

customerMaster (F0301)

**NUM(8)**

**AddressNo**

**PK**

**VARCHAR(12)**

**Business Unitsiness**

**Unit**

**FK1**

Basic master table comprises business unit and company to describe the organization, PT Matahari as below,

|  |  |  |
| --- | --- | --- |
| **businessUnit (F0006)** | | |
| **PK** | **BusinessUnit** | **VARCHAR (12)** |
| FK | Company | CHAR (5) |
|  | BUType | CHAR (1) |
|  | Description | VARCHAR (30) |
|  | Description2 | VARCHAR (30) |
|  | CategoryCode1 | VARCHAR (8) |
|  | CategoryCode2 | VARCHAR (8) |

|  |  |  |
| --- | --- | --- |
| **companyConstant (F0010)** | | |
| **PK** | **Company** | **CHAR (5)** |
|  | Name | VARCHAR (30) |
|  | CurrencyCode | CHAR (3) |
|  | FYBeginDate | DATE (6) |
|  | FiscalYear | NUM (4) |

Address book resides on top of customer master which is repository for all business parties including stakehoders. This table stores employee, customer, supplier, and other business parties,

|  |  |  |
| --- | --- | --- |
| **addressBookMaster (F0101)** | | |
| **PK** | **AddressNumber** | **NUM (8)** |
| FK | BusinessUnit | CHAR (12) |
|  | TaxID | CHAR (15) |
|  | LegalName | VARCHAR (45) |
|  | SearchType | CHAR (1) |

|  |  |  |
| --- | --- | --- |
| **customerMaster (F0301)** | | |
| **PK** | **AddressNumber** | **NUM (8)** |
| **PK** | **BusinessUnit** | **CHAR (12)** |
|  | CreditLimit | NUM (15) |
|  | PaymentTerms | CHAR (3) |
|  | PaymentInstrument | CHAR (2) |

Address book contains additional information as below,

|  |  |
| --- | --- |
| **Tables to Create** | **Description** |
| addressDate (F0116) | With primary key using addressNumber and Date to makeup address detail lines |
| whosWho (F0111) | With primary key with addressNumber and sequence number to enter multiple entities |
| phoneNumber (F0115) | To store phone and fax information |
| eMailAddress (F01151) | To store email address and preference for email itself |
| postalCode (F0117) | To store postal code to print it out |

*Note: the relationship between addressBookMater and tables listed above is 1 to many.*

|  |  |  |
| --- | --- | --- |
| **customerMaster (F0101)** | | |
| **PK** | **AddressNumber** | **NUM (8)** |
| **PK** | **BusinessUnit** | **CHAR (12)** |
|  | CreditLimit | NUM (15) |
|  | PaymentTerms | CHAR (3) |
|  | PaymentInstrument | CHAR (2) |

Where Item information can be,

|  |  |  |
| --- | --- | --- |
| **itemMater (F4101)** | | |
| **PK** | **ItemShort** | **NUM (8)** |
|  | 2ndItemNumber | VARCHAR (25) |
|  | 3rdItemNumber | VARCHAR (25) |
|  | Description 1 | VARCHAR (30) |
|  | Description 2 | VARCHAR (30) |
|  | UnitOfMeasure | CHAR (2) |
|  | UOM2 | CHAR (2) |
|  | UOM3 | CHAR (2) |
|  | SearchText | VARCHAR (30) |

|  |  |  |
| --- | --- | --- |
| **itemBranch (F4102)** | | |
| **PK** | **ItemShort** | **NUM (8)** |
| **PK** | **BranchPlant** | **CHAR (12)** |
|  | 2ndItemNumber | VARCHAR (25) |
|  | 3rdItemNumber | VARCHAR (25) |
|  | Description 1 | VARCHAR (30) |
|  | Description 2 | VARCHAR (30) |
|  | UnitOfMeasure | CHAR (2) |
|  | UOM2 | CHAR (2) |
|  | 3rdItemNumber | VARCHAR (25) |
|  | Description 1 | VARCHAR (30) |
|  | Description 2 | VARCHAR (30) |
|  | UnitOfMeasure | CHAR (2) |
|  | UOM2 | CHAR (2) |
|  | UOM3 | CHAR (2) |
|  | SearchText | VARCHAR (30) |

|  |  |  |
| --- | --- | --- |
| **itemLoction (F41021)** | | |
| **PK** | **ItemShort** | **NUM (8)** |
| **PK** | **BranchPlant** | **CHAR (12)** |
| **PK** | **Location** | **CHAR (12)** |
|  | onHandQuantity | NUM (15,4) |
|  | otherQuantity | NUM (15,4) |

*Note: Item branch file overrides information from item master which can be specific to from branch for PT Matahari.*

|  |  |  |
| --- | --- | --- |
| **itemCost (F4105)** | | |
| **PK** | **ItemShort** | **NUM (8)** |
| **PK** | **BranchPlant** | **CHAR (12)** |
| **PK** | **CostType** | **CHAR (2)** |
|  | UnitCost | NUM (15,4) |

|  |  |  |
| --- | --- | --- |
| **itemPrice (F4106)** | | |
| **PK** | **ItemShort** | **NUM (8)** |
| **PK** | **BranchPlant** | **CHAR (12)** |
|  | UnitOfMeasure | CHAR (2) |
|  | CurrencyCode | CHAR (3) |
|  | PriceEffectiveDate | DATE (6) |
|  | UnitPrice | NUM (15,4) |

*Note: itemCost file contains the purchase price including all logistic cost whereas item price file the price to sell.*

Above master tables suffice to enter sales order which comprises,

|  |  |  |
| --- | --- | --- |
| **salesOrderHeader (F4201)** | | |
| **PK** | **Company** | **CHAR (5)** |
| **PK** | **OrderType** | **CHAR (2)** |
| **PK** | **OrderNumber** | **NUM (8)** |
| FK | SoldTo | NUM (8) |
| FK | ShipTo | NUM (8) |
| FK | BusinessUnit | CHAR (12) |
|  | CurrencyCode | CHAR (3) |
|  | ExchangeRate | NUM (15, 7) |
|  | OrderDate | Date (6) |
|  | RequestDate | Date (6) |
|  | ShipDate | Date (6) |
|  | PaymentTerms | CHAR (3) |
|  | PaymentInstrument | CHAR (2) |
|  | CustomersOrder | VARCHAR (25) |
|  | Reference | VARCHAR (25) |

|  |  |  |
| --- | --- | --- |
| **salesOrderDetail (F4211)** | | |
| **PK** | **Company** | **CHAR (5)** |
| **PK** | **OrderType** | **CHAR (2)** |
| **PK** | **OrderNumber** | **NUM (8)** |
| **PK** | **LineNumber** | **NUM (7,3)** |
| FK | SoldTo | NUM (8) |
| FK | ShipTo | NUM (8) |
|  | CurrencyCode | CHAR (3) |
|  | ExchangeRate | NUM (15, 7) |
|  | OrderDate | Date (6) |
|  | RequestDate | Date (6) |
|  | ShipDate | Date (6) |
|  | PaymentInstrument | CHAR (2) |
|  | CustomersOrder | VARCHAR (25) |
|  | Reference | VARCHAR (25) |
| FK | BranchPlant | CHAR (12) |
| FK | ItemShort | NUM (8) |
|  | 2ndItemNumber | VARCHAR (25) |
|  | 3rdItemNumber | VARCHAR (25) |
|  | Description1 | VARCHAR (30) |
|  | Description2 | VARCHAR (30) |
|  | UnitOfMeasure | CHAR (3) |
|  | OrderQuantity | NUM (15,4) |
|  | UnitCost | NUM (15,4) |
|  | ExtendedCost | NUM (15,4) |
|  | UnitPrice | NUM (15,4) |
|  | ExtendedPrice | NUM (15,4) |

*Note: Some columns may appear to be redundant, but this allows the user to override each information because the request for individual lines can be different from the header.*

### Data Model for Sales Order Entry

Based on basic data above, the data model of sales order entry depicted as below,



Above data model has relation between each table as below,

* 1 – One Item Mater can have many Item Branch file
* 2 – One Item branch may have multiple Item price data per different Unit Of Measure or effect date
* 3 – One branch can have multiple item cost depending on costing type
* 4 – each item branch can have multiple item locations to hold an on-hand quantity
* 5 – One sales order detail line contains 1 item branch
* And multiple line items make one sales order header. Circled number 6 through 9 shows actual flow of data.

### Normalization

Normalization is a database design technique which starts with examining the relationships between tables where the table contains columns with datatype and size (Conolly & Begg, 2015). Here we review First normal form, Second normal form, and lastly Third normal form as below,

*First normal form (1NF):*

* Eliminate duplicative columns from the same table: each table is designed not to have redundant columns within a table.
* Create separate tables for each group of related data and identify each row with a unique column or set of columns (the primary key): in building address book, which comprises Address Date, Contact Information, Phone Number, and Email address in separate tables.

*Second normal form (2NF):*

* Remove subsets of data that apply to multiple rows of a table and place them in separate tables: all master tables with 1 to many relationships are branched out. For instance, one Item Branch can have many Item Price, Item Cost,
* Create relationships between these new tables and their predecessors through the use of foreign keys: In building customer master table, address number and business unit are foreign keys which read from address book master and business unit master respectively.
* Eliminate Redundant Data: minimize the duplication of columns in separate tables with relationship except for the business requirement which requires overriding or different column values for the same column name.

*Third normal form (3NF):*

* Remove columns that are not dependent upon the primary key: Each table in above section to store unique information to meet the business requirement.
* Eliminate data not dependent on the key: Each row in a single table is different from each other.

## PT Matahari Application of ER Model

A Relational Database Management System (RDBMS) is the due course for PT Matahari. Because RDBMS has the benefit of controlling redundancy, Enforcing integrity using business rules, Consistency, Data Sharing, Using Structured Query Language (SQL), Higher security than ODBC databases, Data Model, and Concurrency Control (Thakur, n.d.). This advantage is prominent when adding, updating, deleting the stored data let alone easier to change the data type.

The table 2-1 represents the goal of database implementation project for PT Matahari,

**Table 2-1**

*Database Model to fulfill the goal for PT Matahari*

|  |  |
| --- | --- |
| **Goal** | **Database Model** |
| To computerize business parties information – supplier, customers, employees, agents, and brokers | Add multiple tables to store specific business parties’ information.  *Address book*: containing the common information for all entities  *Employee/Customer/Supplier Master*: detailed entities based on the search type which contains business specific rule and preference |
| To manage all item and inventories in the system after standardizing it with visual catalog | Multiple tables contain information. This approach is essential to growing business along the way.  *Branch/Plant constant*: information about warehouses or where inventory is stored  *Item Master*: Item-specific information or item specification  *Item Branch*: The specification of item in a specific Item/Branch  *Item Location*: To store real-time availability  *Item Ledger*: to record all transactions for a specific item for auditing or planning |
| To record all transactions including purchase, sales, inventory, and commission information | Transaction tables comprises header and detail  *Header*: customer/supplier specific information and allow to override master information  *Detail*: item-specific information which defaults item branch information which user can override |

*Note: In building tables, it is important to have a proper naming convention. Depending on the nature of transaction here defines system code. For instance, the address book starts with 01. And other two digits contain the contents of data. For example, 00 for constant or controlling data, 01 for master data, 11 for transaction data, 19 for ledger or history, and 02 for balance data.*

Other goals of PT Matahari are related to the software applications comprises,

• To enable to have inventory availability checking real-time basis

• To have customer retainage rate over 90% with better customer service and discount

• To increase the sales volume 20% for commercial use

# Structured Query Language (SQL) Scripts

## Database Definition Language (DDL)

The DDL is used to specify the database schema, or it is the language to create data for data where we call the overall description of the database as database schema including controlling access the data (Connolly & Begg, 2015). Currently, PT Matahari does not have any Relational Database Management System (RDBMS) because most of the daily job is done using Microsoft Excel and manual creation of invoice. However to make the business as a sustainable and competitive, here we are implementing RDBMS based on entity relationship we have reviewed earlier.

It is proper to have multiple schemas to have reasonable software development cycle. For instance, it starts with pttestdata to create the schema in a testing environment then Conference Room Pilot (CRP), User Acceptance (UA), Quality Assurance (QA), and Production (e.g., ptproddta schema for this). Below represents some of the execution of testing environment,

-- CHECK THE EXISTENCE OF DATABASE pttestdta OR EXPAND OBJECT EXPLORER FOR THE SAME

IF EXISTS(SELECT name FROM sys.databases

WHERE name = 'pttestdta')

DROP DATABASE pttestdta

GO

CREATE DATABASE [pttestdta]

GO

-- SWITCH TO NEW DATABSAE pttestdta

USE pttestdta

-- Create company constant which is the unit for Asset

CREATE TABLE companyConstant

(

Company CHAR (5) NOT NULL PRIMARY KEY,

CompanyName VARCHAR (30),

CurrencyCode CHAR (3),

FYBeginDate DATE,

FiscalYear NUMERIC (4)

);

-- Create business unit master

CREATE TABLE businessUnit

(

BusinessUnit CHAR (12) NOT NULL PRIMARY KEY,

Company CHAR (5) NOT NULL FOREIGN KEY REFERENCES CompanyConstant(Company),

BUType CHAR (2),

Description VARCHAR (30),

Description2 VARCHAR (30),

CategoryCode1 VARCHAR (8),

CategoryCode2 VARCHAR (8)

);

-- Create Address Book Master with AddressNumber as a primary key

CREATE TABLE addressBookMaster

(

AddressNumber NUMERIC (8) NOT NULL PRIMARY KEY,

BusinessUnit CHAR (12),

TaxID CHAR (15),

LegalName VARCHAR (45),

SearchType CHAR (1)

);

To enable Sales Order Entry, companyConstant, businessUnitMaster, addressBook, customerMaster, itemMaster, itemBranch, itemLocation, itemPrice, itemCost, salesOrderHeader, and salesOrderDetail tables are created in pttestdta (Refer DDL.sql for full list).

## Database Manipulation Language (DML)

Now it is time to populate rows in tables created above using DML statements to manipulate (insert/add, update/change, delete, and select/view). The purpose of DML is to retrieve and update data where update comprises insert, update and delete. Basic report in the following section describes detail statements for select query.

Firstly, populate the data meant for the organization PT Matahari as below,

-- Insert company master (this table is future extension)

INSERT INTO companyConstant VALUES ('00001','PT Matahari', 'IDR', '2017-01-01', 2017)

-- Insert Business Unit 3 records (shop and warehouse)

INSERT INTO businessUnit VALUES (' SHOP','00001', 'BU for Shop', 'Income Statement', 'S', ' ')

INSERT INTO businessUnit VALUES (' WARE','00001', 'BU for Warehouse', 'Balance Sheet', 'W', '001')

INSERT INTO businessUnit VALUES (' 0000','00001', 'Header BU', 'Balance Sheet', 'W', '001')

-- Insert Address Book Master 20 different customers

INSERT INTO addressBookMaster VALUES (4211, ' SHOP', '238794511', 'Ayu Alfiaturrohmah', 'C')

INSERT INTO addressBookMaster VALUES (4232, ' SHOP', '669874157', 'Test Child Support', 'C')

INSERT INTO addressBookMaster VALUES (4233, ' SHOP', '695412540', 'TCapital Systems', 'C')

INSERT INTO addressBookMaster VALUES (4234, ' SHOP', '641313130', 'TCustom Brokers', 'C')

INSERT INTO addressBookMaster VALUES (6001, ' SHOP', '744441965', 'Ririn Ekowati', 'E')

INSERT INTO addressBookMaster VALUES (4343, ' SHOP', '000861965', 'Nippon Paint', 'V')

Repeat this action for the tables created using DDL above. Here we review some of useful query for update and delete as below,

-- Give more credit limit for address number 4242 and 4244

UPDATE customerMaster

SET CreditLimit = 2000

WHERE AddressNumber IN (4242, 4244)

At once itemPrice data are populated, update UnitOfMeasure column value based on itemMaster.UnitOfMeasure as below,

-- update unit of measure based on itemMaster

UPDATE itemPrice

SET UnitOfMeasure = t1.UnitOfMeasure

FROM itemPrice t0, itemMaster t1

WHERE t0.ItemShort = t1.ItemShort

Then update PaymentInstrument and PaymentTerms of salesOrderHeader and salesOrderDetail file based on customerMaster data as below,

-- update payment terms and instrument from customerMaster

UPDATE salesOrderHeader

SET

PaymentInstrument = t1.PaymentInstrument, PaymentTerms = t1.PaymentTerms

FROM

salesOrderHeader t0

INNER JOIN customerMaster t1

ON t0.SoldTo = t1.AddressNumber and t0.PaymentTerms = t1.PaymentTerms

Update certain column values based on referencing tables and delete unnecessary data. To maintain higher integrity implement Transaction Processing using BEGIN TRAN, ROLLBACK TRAN, and COMMIT TRAN,

-- Delete data from itemLocation table which is a child for ItemBranch

-- 1. Verify existing data from itemLocationFile

SELECT \* FROM itemLocation;

BEGIN TRAN

DELETE FROM itemLocation

WHERE ItemLocation = '3.A.a';

COMMIT TRAN;

-- 2. Now Delete Item Location based on the short item number deleted above

BEGIN TRAN

DELETE FROM itemBranch

WHERE ItemShort IN (60100, 60118, 60126, 60134, 60142)

IF (@@ERROR <> 0) BEGIN

PRINT 'locking or unexpected error.'

ROLLBACK TRAN

END

COMMIT TRAN

Then manipulate salesOrderDetail.OrderDate to create Crosstab report in the following section. Some functions from standard library is used as below,

-- 1. Update salesOrderDetail.UnitPrice based on itemPrice.UnitPrice

UPDATE salesOrderDetail

SET

UnitPrice = t1.UnitPrice

FROM

salesOrderDetail t0

INNER JOIN itemPrice t1

ON ( t0.ItemShort = t1.ItemShort AND t0.BranchPlant = t1.BranchPlant);

-- 2. Update salesOrderDetail.UnitCost based on itemCost.UnitCost

UPDATE salesOrderDetail

SET

UnitCost = t1.UnitCost

FROM

salesOrderDetail t0

INNER JOIN itemCost t1

ON ( t0.ItemShort = t1.ItemShort AND t0.BranchPlant = t1.BranchPlant);

-- 3. Calculate extend cost and extended price

UPDATE salesOrderDetail

SET ExtendedCost = OrderQuantity \* UnitCost, ExtendedPrice = OrderQuantity \* UnitPrice;

-- 4. Check whether data got updated correctly

SELECT OrderNumber, LineNumber, SecondItemNumber, OrderQuantity, ExtendedCost, ExtendedPrice from salesOrderDetail

A full list of table population is attached (Refer to DML.sql for detail).

## Reports/Queries

Here we review possible reports which help daily operation for PT Matahari. This report starts with Crosstab report. Crosstab queries let you store your data in a normalized manner but also let you produce pivoted, the de-normalized output from that data. In other words, crosstab queries let you rotate rows to columns to see different summaries of the source data (Ben-Gan, 2000).

-- Crosstab01 - Sales Order Count by Month and Year

SELECT

MONTH(OrderDate) AS OrderMonth,

SUM(CASE YEAR(OrderDate)

WHEN 2015 THEN 1

ELSE 0

END) AS YR2015,

SUM(CASE YEAR(OrderDate)

WHEN 2016 THEN 1

ELSE 0

END) AS YR2016,

SUM(CASE YEAR(OrderDate)

WHEN 2017 THEN 1

ELSE 0

END) AS YR2017

FROM salesOrderDetail

GROUP BY MONTH(OrderDate)

ORDER BY MONTH(OrderDate)

Which yields output below,

|  |  |  |  |
| --- | --- | --- | --- |
| OrderMonth | YR2015 | YR2016 | YR2017 |
| 3 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 |
| 8 | 9 | 0 | 0 |
| 9 | 0 | 3 | 0 |
| 10 | 0 | 0 | 10 |

Sales Order Detail file can generate another Crosstab reports using,

-- 2. Crosstab Total Amount per customer

SELECT

SoldTo,

SUM(CASE YEAR(OrderDate)

WHEN 2015 THEN ExtendedPrice

ELSE 0

END) AS Yr2015,

SUM(CASE YEAR(OrderDate)

WHEN 2016 THEN ExtendedPrice

ELSE 0

END) AS Yr2016,

SUM(CASE YEAR(OrderDate)

WHEN 2017 THEN ExtendedPrice

ELSE 0

END) AS Yr2017

FROM salesOrderDetail

GROUP BY SoldTo

ORDER BY SoldTo

Which yields output below,

|  |  |  |  |
| --- | --- | --- | --- |
| SoldTo | Yr2015 | Yr2016 | Yr2017 |
| 4241 | 4929.4000 | 954.9000 | 3319.6000 |
| 4242 | 2464.7000 | 0.0000 | 4174.5000 |
| 4243 | 0.0000 | 1609.8000 | 1709.8000 |

Same can generate multiple reports based on data populated,

-- Select All Business Unit and Customer

SELECT t0.BusinessUnit, t1.AddressNumber, t2.LegalName

FROM businessUnit t0, customerMaster t1, addressBookMaster t2

WHERE (t0.BusinessUnit = t1.BusinessUnit AND t1.AddressNumber = t2.AddressNumber )

-- Show the list of customer who bought white color pain

SELECT t0.SoldTo, t0.SecondItemNumber, t1.Description1

FROM salesOrderDetail t0

INNER JOIN itemMaster t1

ON t0.ItemShort = t1.ItemShort

WHERE t1.Description1 like 'White%' Order By SoldTo;

--- Select total order quantity per customer from salesOrderDetail

SELECT SoldTo, COUNT (\*) AS OrderCount, SUM (ExtendedPrice) AS OrderTotal

FROM salesOrderDetail

GROUP BY SoldTo ORDER BY SoldTo

The last example returns report as below,

|  |  |  |
| --- | --- | --- |
| SoldTo | OrderCount | OrderTotal |
| 4241 | 14 | 11768.6000 |
| 4242 | 8 | 6639.2000 |
| 4243 | 4 | 3319.6 |

The output varies based on the relationship and WHERE clause uses. The output can be useful in conjunction with any visualization tools. For instance, Microsoft Excel can convert table as below,

The attachment REPORT.sql gives full implementation.

## PT Matahari Application of DDL, DML, and Reports

The goal of relational database implementation for PT Matahari is,

* To computerize business parties information – supplier, customers, employees, agents, and brokers: For this case study, address book records are created along with customer master using different search types. For instance, C for the customer, V for the vendor, E for employee and so on.
* To manage all item and inventories in the system after standardizing it with visual catalog: Item Location File is to store on-hand quantity which gets updated actual shipment of sales order takes place.
* To record all transactions including purchase, sales, inventory, and commission information: Simplified approach accomplished how to enter sales order what are the columns to compute the needs.

Based on the relationship database created, multiple reports are generated Crosstab with Total Order Amount yearly per customers, the relationship between Business Unit and its customers, customers who bought white shade paint, total sales quantity and amount per customer, and the number of items per category code defined. All these reports enable better customer service through reports generated using transaction history and better planning by maintaining and better planning by checking the supply and demand for inventories.

It is important for PT Matahari to have proper documentation which enables true Business Intelligence (BI). According to Gartner, Business Intelligence (BI) is common terms including the software applications, infrastructure and tools, and possibly best practice which enable access to and analysis information to improve and optimize the business decisions and its performance (Gartner, n.d.).

# Database Administration Plan

## Database Administration Plan

The database administration plan follows its tasks as described in Table 4.1 – Database Administration Plan,

**Table 4.1** – Database Administration Plan

|  |  |
| --- | --- |
| **Plan** | **Detail for PT Matahari Paint** |
| Evaluate and Plan DBMS | Oracle database servers because Oracle supplies the engineered system which is made up of both software and hardware including network if needed. |
| Install, Create, and Open Database | Installation of Oracle Database based on the mission and goal for PT Matahari |
| Define Security and Integrity Constraint | Allow only named user before extending the functionality to the Internet and implement Transaction Processing when it requires higher integrity. For instance, Sales Order Creation, Ship Confirmation, Sales Update, Purchase Order Creation, Purchase Order Receipt, Voucher Match and so on. |
| Tune Database Performance | To handle concurrent processing, perform proper performance test. Gather Oracle Statistics regularly, add table indices, and put more memory if needed. |
| Backup | Oracle Real Application Clusters (Oracle RAC) is a clustered version of Oracle Database based on a comprehensive high-availability stack that can be used as the foundation of a database cloud system as well as a shared infrastructure, ensuring high availability, scalability, and agility for any application.  And back up to Oracle Cloud because it is cheaper, simpler, and scalable (Oracle, n.d.) |
| Download and Install Patches | Most of the patches are related to performance and security so timely apply the patches are important without having downtime. |

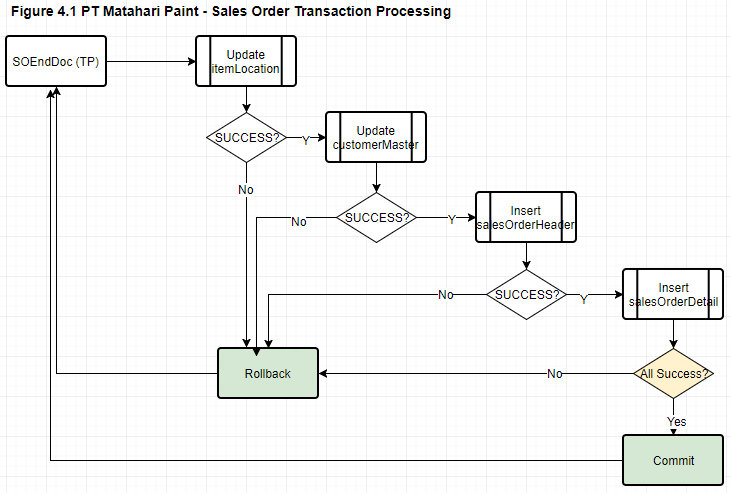
*Note*: Above plan is combined literature of Connolly & Begg (2015) and Oracle Database Administration Guide for Oracle 12C update 2.

It is important to choose the software which DBA is familiar with and the database which supports multiple cloud options for now.

## Transaction Processing in Sales Order Entry

PT Matahari needs to have transaction processing in the area: Sales Order Entry (to allocate available inventory properly and to update customer’s credit information properly), Shipment Confirmation (when shipping items to deliver to the customer), Invoicing, and Cache receipt in Sales. On the other hand, Purchase Order Entry (to increase item and determine the availability of item including the cost of goods), Purchase Order Receipt (receiving of physical goods), and Voucher Match, and cash payment.

Oracle defines a transaction is a logical, atomic unit of work which comprises multiple SQL statements. A transaction groups SQL statements so that they are either all committed, which means they are applied to the database, or all rolled back, which means they are undone from the database. Oracle Database assigns every transaction a unique identifier called a transaction ID (Oracle, n.d.). Figure 4.1 depicts how to maintain the integrity of data related with.



It validates the value entered before End Doc function gets called for customer and item, then this routine commits the transaction to database: 1. Relieve Quantity On Hand, 2. Subtract Available credit amount. Above figure is to comply with the basic properties of Atomicity, Consistency, Isolation, and Durability (ACID). Each can be translated as ‘All of Nothing,' higher integrity, isolate transactions using record reservation which records who is in use, and the committed data is permanent with recovery mechanism (Oracle, n.d.).

## Database Security Procedure

The importance of database security is paramount because the data stored in the database for any organization is sensitive or restrictive. All the procedure and counter-measure against security threats have to be spelled out on the way to the hardening database system to prevent data loss, leakage, or unauthorized access to the database.

Any information systems which includes database system has security threats: Unauthorized Access, Unauthorized Tempering, and Denial of Service. These items are equivalent to provisions for user access control, user authentication, and availability (Oracle, n.d.). Table 4.2 described in detail.

**Table 4.2** – Database Security

|  |  |
| --- | --- |
| **Area** | **Procedure** |
| Authentication | A means for the server to verify client’s identity. For example, a client gives a password to Directory Server during an LDAP bind operation. |
| Password policies | Defines the criteria that a password must satisfy to be considered valid, for example, age, length, and syntax. And enforce the user to change it periodically. |
| Encryption | Encryption protects the privacy of information. Implement encryption for all the endpoints. |
| Access Control | Tailors the access rights granted to different directory users and provides a means of specifying required credentials or bind attributes. |
| Account inactivation | Disables a user account, group of accounts, or an entire domain. Reject all authentication attempts automatically.  Inactivation is crucial when there is any change in employment. |
| Secure Socket Layer (SSL) | SSL maintains the integrity of information. If encryption and message digests are applied to the information being sent, the recipient can determine that it was not tampered with during transit. This is important when it opens the business in the public network. |
| Auditing | Allows you to determine if the security of your directory has been compromised. For example, you can audit the log files maintained by your directory.  These factors get important because of machine learning technology nowadays. |

The detail steps start from analyzing the security needs comprise,

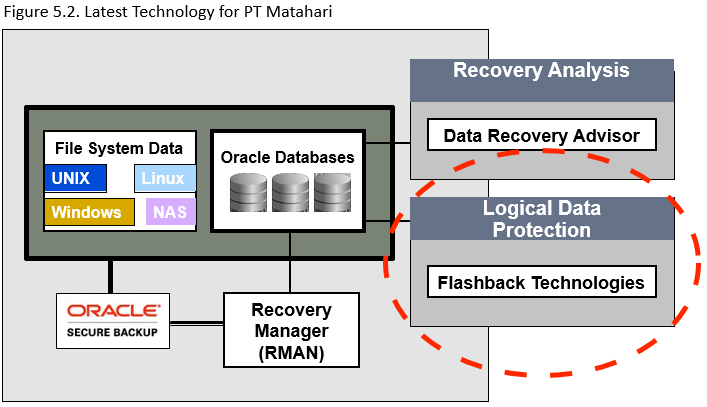
* Providing users with access to the information they need to perform their jobs using role-based access control.
* Protecting sensitive employee or company data from general access. Either through proper settings of business view or row security when it is applicable.
* Ensuring data integrity through transaction processing which requires data manipulation for multiple files especially when it affects inventory and item cost.

In expanding the business to the Internet, which allows the public network to access database servers, directly and indirectly, data privacy and data integrity will be the main concern. Same has to be mitigated through,

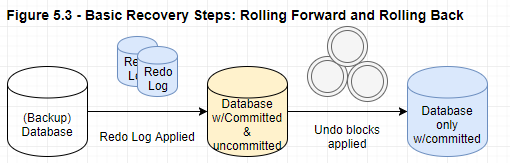
* Encrypting data
* Using certificates to sign data
* Encrypting data transfers

## Backup Plan and Recovery Model

Point-in-time recovery is essential to sustain the daily business. According to Oracle, the maximum availability architecture provides Data Recovery Advisor for analysis, Flashback Technology for moving back in time, Active Data Guard for failover, and Recovery Manager and Oracle Security Backup (Oracle, n.d.). Figure 5.2 depicts the architecture of these features and functionality.



As suggested earlier, PT Matahari plans to implement transaction processing in multiple applications. When all internal routines have committed successfully rolling forward has to take place, and any subroutine has failed to update then rolling back need occurring. Below Figure 5.3 shows simplified routine for rolling forward and rolling backward using redo log file and undo log file using the Oracle Recovery Manager (RMAN) (Connolly & Begg, 2015).



*Note*: this figure is adapted from Oracle database documentation for all releases. Undo for rollback and redo for commit.

## PT Matahari Application of Database Administration

The goal of database implementation project is to computerize all master and transaction data, to maximize the customer satisfaction through accurate and timely inventory management with competitive price management.

First of all, transaction processing is crucial to have ACID database properties by putting a set of SQL statements within Transaction Processing boundaries. However, transaction processing can result in slower performance because of database blocking and deadlock when it is not implemented properly. To have better performance limit timeout for transaction processing, gather Oracle statistics regularly, add more indices when needed. Another approach is to implement record reservation not to have deadlock and isolate a transaction from other process and user.

For the security, PT Matahari need considering physical database security, logical database security using a firewall, restrict data access, and auditing for all change made. The security encompasses privileged user control (and privilege analysis), masking, encryption, redaction, database firewall, and proper audit data and event logs.

Backup plan and a recovery model is an essential factor for PT Matahari not to have any disruption of business caused by unavailability of the database system. To have an economical backup, backup to Oracle Cloud is an option for PT Matahari because it cost less than conventional offline backup like magnetic tapes. And to make full use of DBMS which provided by Oracle, server-managed backup through RMAN is advised.

# Future Database System Implementation Plan (Week5)

## Object-Oriented, Object-Relational Database System, and Web-based Database System

The benefit of the relational database system is that it has simple datatypes, a powerful query language. The Object-oriented programming language gain popularity which requires other types of the database system, which is an object-oriented database system. Hence this system can handle complex data types, integration with a programming language, and high performance in nature. Since then there is an effort to have the benefits of both relationship database system and object-oriented database system, which is the object-relational system. Hence object-relational database system can have complex data types, powerful query languages, and high protection (Zaiane, 1998).

On the other hand, with the advent of the Internet, the trend is towards using the Internet for the internal and external application. Commonly, HTML user, HTML Server and Java Application Server (JAS) through the Internet and JAS talks to database system using JDBC. The web-based database system is a database application designed to managed and accessed through the Internet allowing the collection of infinite amounts of data from an infinite amount of sources (North, n.d.).

### Impact

PT Matahari’s business expansion plan includes the interior design business which requires different data types. For instance, to present the result of the design, interior designer draw a certain part of the house which simulates 3-D view for the potential client. This types of drawing require Object-Oriented Relational database system to accommodate any update in a certain part updates it as a whole. Meanwhile, many households may not have a personal computer, but the smartphone and tablets are commonplace because of the nature of network in Indonesia. So it is crucial to have web-based database system which is free-form data entry for the customer based on the catalog from PT Matahari. So these future database system is essential along with future expansion plan.

## Data Warehouse Implementation

The primary purpose of a data warehouse is to maintain historical data and to analyze the information to gain a better understanding of the business and to improve it. Hence, the data warehouse is different from online transaction processing (OLTP) systems. Commonly the characteristics of the data warehouse can be subject oriented, integrated, nonvolatile, and time variant as Willam Inmon has suggested (Oracle, n.d.).

The change or addition of data warehousing depends on the contrasting OLTP and data warehousing environment features because data warehouses are not exclusively in third normal form (3NF) which tries to minimize the repetition of columns for a certain table. Hence, table 5.1 represents the probable changes from relational database to data warehouses,

**Table 5.1** – Data warehouse vs. OLTP

|  |  |  |
| --- | --- | --- |
|  | **OLTP** | **Data Warehouse** |
| Workload | Supports predefined operations | Need to accommodate ad hoc queries and data analysis. Hence, it is crucial to optimize to have the best performance |
| Data Modification | Issuing DML is daily routine | Periodic Extract, Transform, and Load (ETL) using bulk data modification technique |
| Schema Design | Fully normalized schema and guarantee consistency | Partially denormalized schemas to optimize query and analytical performance (because join is always most expensive) |
| Typical Operations | Limited number of data retrieval | Query can read millions of rows for a single request |
| Historical data | Limited period of data | Years of data because the primary purpose of data warehouse to have historical analysis and reports |

*Note*: This table uses same literature from Oracle.

## Distributed Database Considerations

A distributed database is a logically interrelated collection of shared data but physically distributed over a computer network (Connolly & Begg, 2015). The advantages of a distributed database which reflects the physical, organizational structure; data sharing between remote sites; better reliability, availability, and performance; modular database growth, and facilitate integration.

Above benefit translates into its considerations because the distributed database system can be expensive than the centralized one.

* System performance
* Security, concurrency, consistency, and availability
* The amount of data needed for query
* Management strategy for cross-system security, monitoring and tuning, problem handling, database backup and recovery, and change control
* Distributed database design decision (IBM, n.d.).

## Types of Business Intelligence from the Database

Business Intelligence (BI) is the analysis of an organization’s information to make business decision commonly based on database warehouse. Some technique can be,

* Analytic SQL: ranking, moving averages, cumulative sums, ratio-to-reports, and period-over-period comparisons
* OLAP: to provide native multidimensional storage and rapid response times in analyzing data across multiple dimensions. OLAP makes it possible for data analysts to obtain answers to complex, iterative queries during an interactive session.
* Advanced Analytics: for predictive analytics, data mining, text mining, statistical analysis, advanced numeric computations, and interactive graphics inside the database (Oracle. n.d).

### Decision Making Based on BI

We have described that the purpose of data warehousing and its output business intelligence above. In given case study retail store, PT Matahari Paint has long-term sustainable business expansion plan when positioning itself in this market properly. Likewise other planning enterprise solutions, BI solutions can produce,

* Faster reporting, analysis, or planning: this is possible even without knowing DML statement itself
* More accurate reporting, analysis or planning
* Improve data quality: since this is the result of data warehouse or data mart

In turn, this system can have,

* Better business decision: naturally decision is made based on refined data and statistics
* Improve customer satisfaction by predicting the behavior of customer or the group of customers
* Reduce cost and increase revenues (BI-SURVEY, n.d.)

## Benefits of Data Warehousing

We defined a data warehousing is a subject-oriented, integrated, and nonvolatile collection of data to have a better business decision for business owners and managers which has advantage below when it implemented successfully.

### ROI on BI Initiatives

A study by the International Data Corporation (IDC) has reported that data warehouse projects accomplished an average three-year Return on Investment (ROI) of over 400% though it is an expensive project to implement (Connolly & Begg, 2015). So this figure should be encouraging factor to implement a data warehouse.

### Competitive Advantage

When proper BI tools are implemented on top of data warehouse, which enables the business owner to uncover the pattern of data which was unavailable, unknown information previously on customers, trends, and demands. Now PT Matahari has sufficient information to take action prior to any competitors’ action in Malang in Indonesia.

### Productivity

The integrated database on top of a relational database is the consistent, subject-oriented, and sufficient amount of historical data. Hence, this types of information yields more substantive, accurate, and consistent analysis (Connolly & Begg, 2015).

## Data Warehouse Challenges

Where it is about Big Data or the repository of the output from various enterprise solutions (e.g., Enterprise Resource Planning (ERP), Supply Chain Management (SCM), and so on), the data warehouse has some challenge to face.

### Data Collection

The first step of Extract, Transform, and Load (ETL) is a most important factor to have successful data warehouse implementation is the quality of data since the data is coming from inconsistent data from disparate sources. Inconsistent data, duplicate data, logic conflicts, missing data result in quality challenges (Wentzlaff, 2014).

### Resources

Business agility can be accomplished using data warehouse at the cost of resources which occupies both time and space. As the number of data sources and data types increases, the complexity of extraction and integration requires much higher physical resources including both memory and hard disk (Strange, 2001).

### Hidden problem

Other hidden problem can be extensive when the information comes from multiple data sources which have a different structure, description, heterogeneous, and inconsistent. All the business decision can be made based on good quality data. So focusing points have to be ensuring acceptable data quality; ensuring acceptable performance; reconciliation of data.

## PT Matahari Application of Future Database Systems

The initial goal of PT Matahari Paint is building a relational database system which functions like Enterprise Resource Planning (ERP) which enables the integrated business processes, one central database which holds daily transactions with good user interface and with rich views and reports. In turn, this implementation project makes it possible to have accurate inventory management, customer management, and order management using transaction processing for concurrency of data. Then proper gathering of information enables better customer service which translates daily transaction into customer satisfaction and makes room for future extension.

As a due course, the expansion of business requires the distributed database system which locates the physical database in the network unless PT Matahari rents the Cloud service as Platform as a Service (PaaS). In exploring the possible business opportunity in the area of interior design, the object-relational database system can be implemented because it requires extended data types and dynamic update of a certain change made. On the other hand, the web-based database system is considered to hold all feedback from customers and business parties.

On the other hand, there will be time to have proper analysis based on huge amount of master and transactional data collected along with other systems. Hence, a data warehouse shall enable PT Matahari Paint to have a competitive edge against its’ competitor in having timely, accurate business intelligence tools for better business decision.

# References

Ben-Gan, Itzik (2000). Dynamic Crosstab Queries, Produce pivoted, denormalized output from normalized data, Retrieved October 22, 2017, from <http://sqlmag.com/t-sql/dynamic-crosstab-queries>

BI-SURVEY. (n.d.). The Benefits of Business Intelligence… Why Do Organizations Need BI? Retrieved November 7, 2017, from <https://bi-survey.com/benefits-business-intelligence>

Burke, A. (n.d.). Goal vs. Mission. Retrieved October 6, 2017, from <http://smallbusiness.chron.com/goal-vs-mission-37563.html>

Connolly, T. & Begg, C. (2015).Database Systems: *A Practical Approach to Design, Implementation, and Management*, Sixth Edition. Upper Saddle River, New Jersey: Pearson Education.

eTutorials. (n.d.). Defining the Mission Objectives. Retrieved October 6, 2017, from <http://etutorials.org/SQL/Database+design+for+mere+mortals/Part+II+The+Design+Process/Chapter+5.+Starting+the+Process/Defining+the+Mission+Objectives/>

Gartner. (n.d.). IT Glossary, Business Intelligence (BI). Retrieved October 20, 2017, from <https://www.gartner.com/it-glossary/business-intelligence-bi>

IBM. (n.d.). Data considerations for a distributed relational database. Retrieved November 7, 2017, from <https://www.ibm.com/support/knowledgecenter/en/ssw_ibm_i_61/ddp/rbal1dataconsid.htm>

North, M. (n.d.). What Is a Web Databbase? Retrieved November 6, 2017, from <https://www.techwalla.com/articles/what-is-a-web-database>

Oracle. (n.d.). Access Control, Authentication, and Encryption. Retrieved October 31, 2017, from <https://docs.oracle.com/cd/E19901-01/817-7607/aci.html>

Oracle. (n.d.). Business Intelligence. Retrieved November 7, 2017, from <https://docs.oracle.com/database/122/CNCPT/topics-for-database-administrators-and-developers.htm#CNCPT6543>

Oracle. (n.d.). Database Backup. Retrieved October 30, 2017, from <https://cloud.oracle.com/database_backup>

Oracle. (n.d.). Database Data Warehousing Guide. Retrieved November 11, 2017, from <https://docs.oracle.com/database/121/DWHSG/concept.htm#DWHSG-GUID-452FBA23-6976-4590-AA41-1369647AD14D>

Oracle. (n.d.). Oracle Database Backup-and-Recovery Best Practices and New Features. Retrieved October 31, 2017, from <http://www.oracle.com/technetwork/database/availability/311394-132335.pdf>

Oracle. (n.d.). Transactions. Retrieved October 30, 2017, from <https://docs.oracle.com/cloud/latest/db112/CNCPT/transact.htm#CNCPT016>

Pozin, I. (2012). 6 Ways to Measure the Success of Any Project. Retrieved October 6, 2017, from <https://www.inc.com/ilya-pozin/6-ways-to-measure-the-success-of-any-project.html>

Strange, K. (2001). The Challenges of Implementing a Data Warehouse to Achieve Business Agility. Retrieved November 7, 2017, from <http://www.rocklandsolutions.com/docs/gartner_dw_study.pdf>

Thakur, D. (n.d.). What is DBMS? Advantages and Disadvantages of DBMS. Retrieved October 18, 2017, from <http://ecomputernotes.com/fundamental/what-is-a-database/advantages-and-disadvantages-of-dbms>

Wentzlaff, A. (2014). 7 Challenges to Consider when Building a Data Warehouse. Retrieved November 7, 2017, from <http://www.onapproach.com/7-challenges-consider-building-data-warehouse/>

Zaiane, O. (1998). Comparison of Object-Oriented and Object-Relational Databases. Retrieved November 6, 2017, from <http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chapter9/node13.html>