

05/07/2018

Oracle database

data project

written BY:

Dylan cattelan,

cedric montes,

anthony luque,

anais banos

cesi exia a3

Table of Contents

[Oracle database 2](#_Toc518573182)

[MCD – Modèle de conception de données 2](#_Toc518573183)

[Color, Variant, Texture 2](#_Toc518573184)

[CandyCost / Candy 3](#_Toc518573185)

[Packaging, Palette, Cardboard 3](#_Toc518573186)

[CandyReferences 3](#_Toc518573187)

[Orders 3](#_Toc518573188)

[Country 3](#_Toc518573189)

[Stock 3](#_Toc518573190)

[Machine 3](#_Toc518573191)

[MLD – modèle logique de données 4](#_Toc518573192)

[MPD – Modèle physique de données 4](#_Toc518573193)

[Data Generator 4](#_Toc518573194)

[UML 4](#_Toc518573195)

[rights and privileges 5](#_Toc518573196)

[Data generator 5](#_Toc518573197)

[ETL - talend 5](#_Toc518573198)

iNTRODUCTION

To introduce this report, I would like to explain the situation.

During this week, we need to replicate the operation of a candy factory. Thereby, we had to conceive, develop and deploy an Oracle DB and a Data Generator.

We can split our work in two main parts:

* Oracle DB
* Data Generator

The Oracle DB concerns the internal operation of the factory, and the Data Generator simulates the external flows of the company like orders.

# Oracle database

## MCD – Modèle de conception de données

This schema describes how we thank our database, there are different link with associations between them. We used the MERISE conventions.

(picture)

We can see above our database design schema, in this case we are in a relational Database.

It contains the following 13 tables:

* Color
* Variant
* Texture
* Stock
* Cardboard
* Palette
* Packaging
* Machine
* CandyCost
* Candy
* CandyReferences
* Orders
* Country

To explain we will group some tables according to their characteristics.

### Color, Variant, Texture

We chose to create specific tables for these data, to anticipate their evolution. For example, if the company wants to add another color, this new addition will be managed in a naturel way with a new Id in the corresponding table.

### CandyCost / Candy

In this case, we decided to split these data in two tables because of the different sources of the data. CandyCost group all the Commercial Data concerns the different costs, like Manufacturing or Box price. The second one concerns the composition of the candy with the different quantities of components.

### Packaging, Palette, Cardboard

### CandyReferences

This table will group all the unique candy references, we chose to add an Id in this table to facilitate the use and lighten the different documents.

### Orders

This table will be used to store order information like the order Date or the client name. It can be important to have a specific table. It makes easier when we search information about a specific order. It avoids us to join some tables, in our case all information are store clearly in an optimal table.

### Country

Country Table groups the different possibilities of country shipping. And we can see it is link with the packaging table we have explain above. Thanks to the packaging\_id we can know how the package is delivered in each country.

### Stock

Stock it is the only table without connection, because information which are store in concern raw material supply and it is manage by the generator.

### Machine

We group Manufacturing Machine and Conditioning machine in Machine table, we use foreign keys to differentiate the two. In the first case, if the variant\_id\_FK is not null that show us it is a manufacturing machine, at the opposite if the packaging\_id\_FK is not null and the variant\_id\_FK is null (it is the same way, but it is just the opposite of the precedent case) we can say we are talking about a conditioning Machine.

## MLD – modèle logique de données

(picture)

This is a logic schema, that means there is no association table, but there are the 13 main tables with the different primary and foreign keys which shows us how they are linked to the other one.

That is the same explanation as MCD.

## MPD – Modèle physique de données

Cf annex

That is totally different from the previous explanations, because we built it according to the programming language. This schema represents the previous one with the adapted syntax.

As we can see, there are different part in this SQL script.

On the one hand, we have the creation part, this is used to create all the necessary tables for the database to function properly.

On the other hand, we have commands for insert data into the previous tables, it is used to insert a dataset into the database and it is called a databank.

This databank will be used to do our calculations and our data generations.

# Data Generator

## UML

# rights and privileges

As an introduction to this part, we decided to build 2 users, for security reasons.  
The first one is called “generator” it will be able to SELECT any tables and to INSERT and UPDATE only the following tables:

- Orders

- CandyReferences

- Stock  
Concerning the talend user, he will only get a SELECT available on every tables of the Oracle database, no modifications are directly allowed so he won’t have any others rights.

## Data generator

## ETL - talend