# SeaTRANSPORTER INC.

# 

SOLUTION CONCEPTS AND PROPOSAL DOCUMENT

# Overview:

Business background

Business problem of current project is find out amount of transportation for each customer’s company by months.

So we can highlight some main objects:

- Ships

|  |  |
| --- | --- |
| Ships | Description |
| Ship unique id  Weight  Height  Water volume  Max cargo | Ship unique identifier  Ship weight  Ship height  Ship water volume  Ship max cargo |

* Client company

|  |
| --- |
| Client company |
| Company name  Company office region  Company office country  Company office city  Company office address  Company office fax  Company office telephone  Company email  Company contact person |

* Ports

|  |  |
| --- | --- |
| Ports | Description |
| Port identifier  Pier num  Pier identifier  Port contact person  Port contact telephone  Port address  Port city  Port country  Port region | Unique port identifier  Pier number in port  Pier unique identifier  Contact person in port |

* Products

|  |  |
| --- | --- |
| Products | Description |
| Unique product identifier  Product name  Product description  Product category  Income coefficient | Coefficient that shows the revenue from one ton of cargo |

* Insurance

|  |  |
| --- | --- |
| Insurance | Description |
| Unique document code  Risk type  Insurance cost  Insurance company name | Type of risk when company agrees to transport some products |

Sum up business problem:

1. Amount transportation for each customer’s company;
2. Calculate pct depart/arrive products for each company per months;
3. Calculate income of each ship for each transportation by customer’s company per months.

Pct depart/arrive products will calculated as amount of departed products divided by amount of arrived products (data taken from source file look [source table structure](#_Source_tables_structure)).

Income from each ship– income coefficient of each product multiply by amount arrived products.

Benefits

DWH establishment will increase company profit. Analytic department will work more effective. It means that 40 percent may be fired. Moreover, executive department will get fresh reports with actual data for effective company’s management. DWH helps to react much more quickly on some business problems and mistakes.

## Requirements:

### Business requirements

First of all let’s describe measures:

Pct depart/arrive products will calculated as amount of departed products divided by amount of arrived products (data taken from source file look [source table structure](#_Source_tables_structure)).

Income from each ship (in USD) – income coefficient of each product multiply by amount arrived products.

Amount transportation for each company to highlight most important customers.

In future will build reports based on this business problems.

Its important to make historical respective, that why we need to store ten years ago from current date data. The business data model is a “point-in-time” model.

### Technical requirements

Users:

System has to have several users:

Sysadmin – must have all access to all layers of our system;

DWHmanager – must have access to object schema to manage it;

Reporter – must have only read ability to datamarts for report building and so on.

Roles:

Sysadmin – all grants (as sysdba);

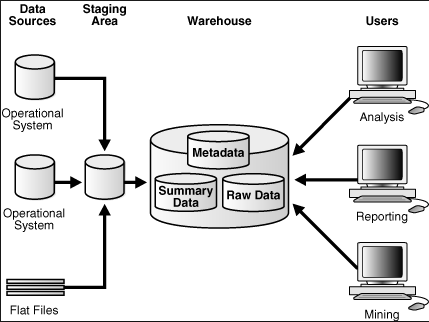
DWmanager – DML, DDL grants, create user grants for DWH tables and views;

Reporter – select from allowed view.

Languages: System have to support two languages: English and Dutch.

Data granularity: month

DWH consistent:



Data sources:

In current system data sources are flat files (.tab) with data from customer’s systems. Flat files consists important for business information.

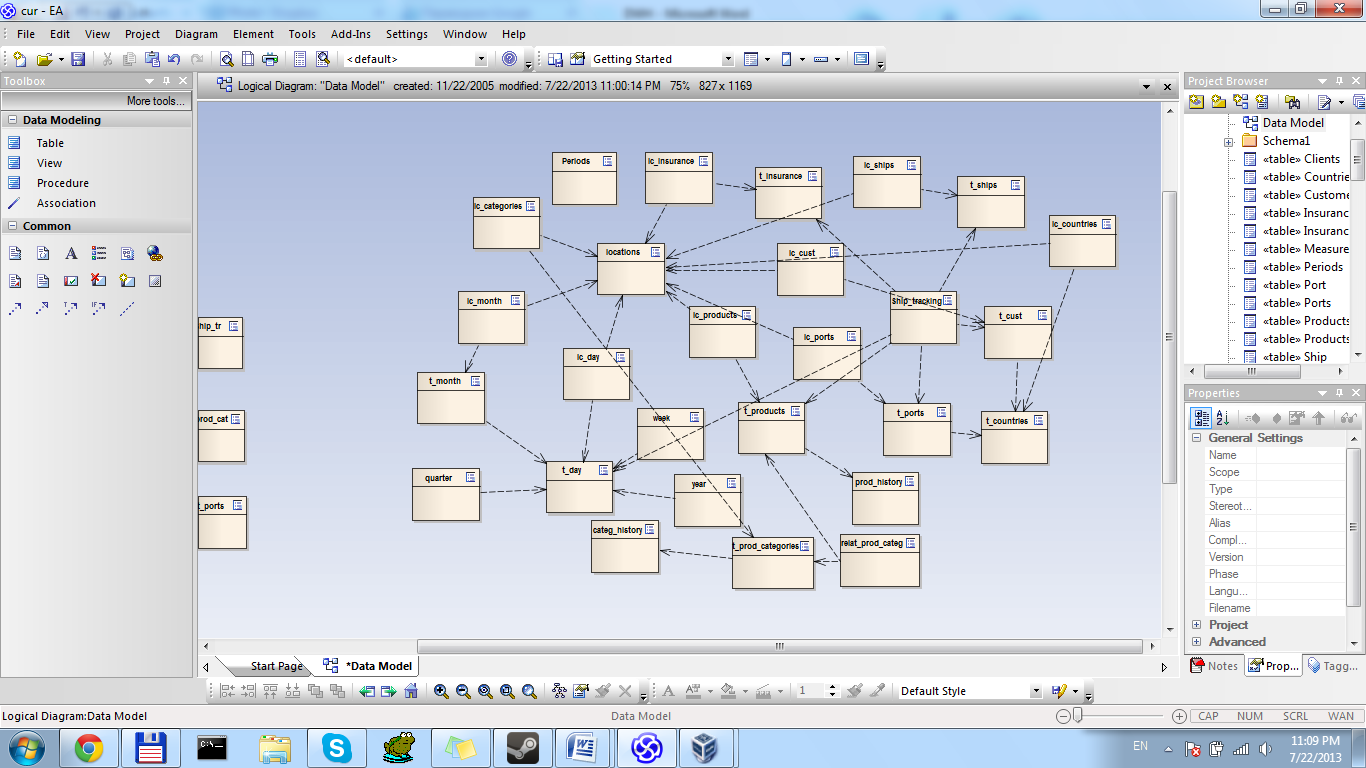
For additional information look [source tables structures](#_Source_tables_structure).

Stage area:

That area will store tables from external sources (flat files). It makes several important functions:

1. Store all data;
2. Check data integrity;
3. Format some data;
4. Create a hierarchy and history;

Approximate view of stage area data:



Properties:

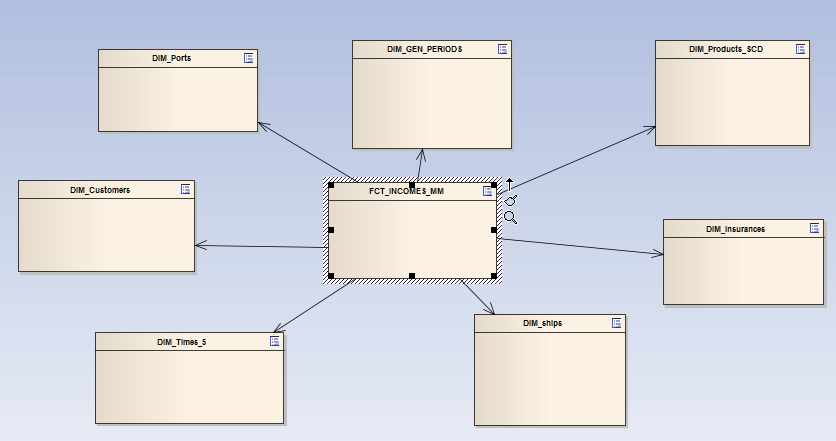
Default tablespace

ts\_sa\_data\_01 (SIZE 500M AUTOEXTEND ON NEXT 50M SEGMENT SPACE MANAGEMENT AUTO NOLOGGING

Cleansing area

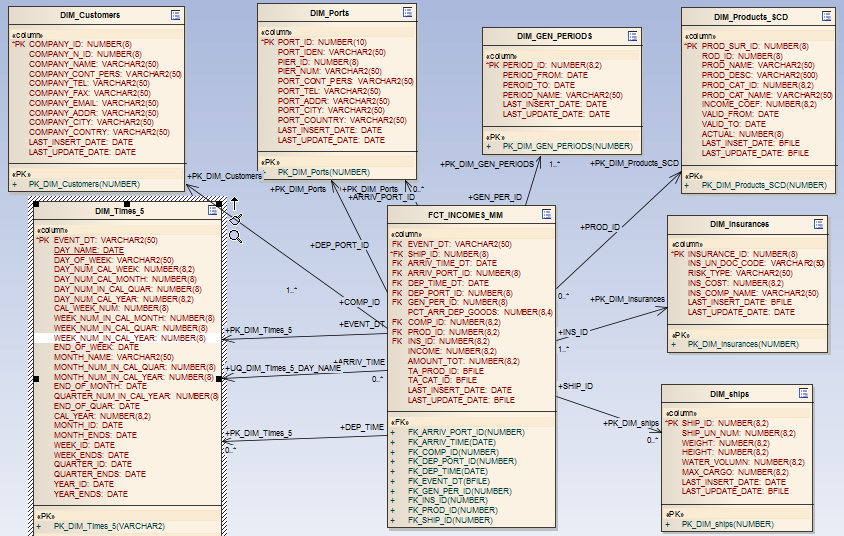
Warehouse:

Logical and physical start schema:

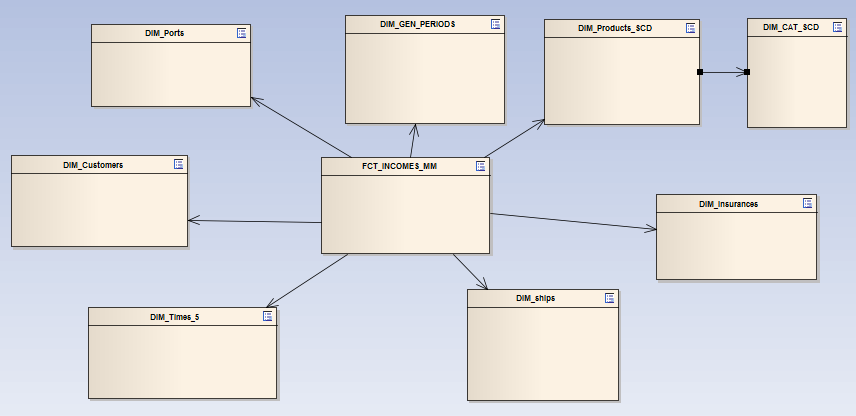


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dimensions | Type | Size | Merged dimensions | Description |
| DIM\_CUSTOMERS | SCD1 | Small | DW.T\_CUST  DW.LC\_CUST  DW.T\_COUNTRIES  DW.LC\_COUNTRIES | Contains information about customer companies |
| DIM\_PORTS | SCD1 | Small | DW.T\_PORTS  DW.T\_COUNTRIES  DW.LC\_PORTS  DW.LC\_COUNTRIES | Contains all information about ports |
| DIM\_TIMES\_5 | SCD1 | Big | DW.T\_DAYS  DW.T\_WEEKS  DW.T\_MONTHS  DW.T\_QUARTERS  DW.T\_YEARS  DW.LC\_DAYS  DW.LC\_MONTHS | Contains time |
| DIM\_SHIPS | SCD1 | Small | DW.T\_SHIPS  DW.LC\_SHIPS | Contains all information about company’s ships |
| DIM\_PRODUCTS\_SCD | SCD2  hybrid | Big | DW.T\_PRODUCTS  DW.LC\_PRODUCTS  DW.LC\_PROD\_CATEGORIES  DW.T\_PROD\_CATEGORIES  DW.T\_PROD\_HISTORY  DW.T\_CAT\_HISTORY  DW.T\_PROD\_CAT\_LINKS | Contains all information about products which are transported. |
| DIM\_INSURANCES | SC1 | Small | DW.T\_INSURANCES | Contains all info about insurances. |
| DIM\_GEN\_PERIODS | SC1 | Small | DW.T\_GEN\_PERIODS | Contains information how to group our facts |

Physical start schema



Logical snowflake schema:



Properties:

Default tablespace:

ts\_dw\_data\_01 (SIZE 300M AUTOEXTEND ON NEXT 100M SEGMENT SPACE MANAGEMENT AUTO LOGGING

Cleansing area

Time key:

In current project we will use complex time key. It will contain 2 columns:

* Month name (event\_DT) – month name.

## Solution sketch

### Source tables structure

There are several data tables:

1. Cust\_comp.tab

This file contains information about customer companies:

|  |  |
| --- | --- |
| Customer companies | |
| Attribute | Description |
| Company name | Company full name |
| Company office region | Geographical region of the company’s office |
| Company office country | Country name where company’s office is |
| Company office city | City name where company’s office is |
| Company office address | Street and number of the house where company’s office is |
| Company office fax | Fax number of the office |
| Company office telephone | Telephone number |
| Company email | Company email |
| Company contact person | Pearson who contracted with our company |

1. Ports.tab

This file contains information about all ports that makes a deal with our company to hold, load and restore our ships.

|  |  |
| --- | --- |
| Ports | |
| Attribute | Description |
| Port identifier | Unique port identifier |
| Pier num | Number of pier which reserved for our ships |
| Pier identifier | Unique pier identifier |
| Port contact person | Pearson who contracted with our company |
| Port contact telephone | Port telephone |
| Port address | Street of port |
| Port city | City where port is situated |
| Port country | Country where port is situated |
| Port region | Geographical region where port is situated |

1. Insurance.tab

This file contains information about insurances of my ship with products.

|  |  |
| --- | --- |
| Insurance | |
| Attribute | Description |
| Unique document code | Unique document identifier |
| Risk type | Type of risk |
| Insurance cost | Cost of insurance |
| Insurance company name | Company who supply insurances |

1. Products.tab

This file contains information about products which transported by our company.

|  |  |
| --- | --- |
| Products | |
| Attribute | Description |
| Unique product identifier | Unique product identifier |
| Product name | Product name |
| Product description | Product description |
| Income coefficient | Income coefficient per one ton transported goods |
| Prod\_cat\_id | Reference on product category |

1. Prod\_categories.tab

Contains all information about product categories

|  |  |
| --- | --- |
| Products | |
| Attribute | Description |
| Unique\_category\_identifier | Unique product category identifier |
| Product\_category\_name | Category name name |

1. Ship tracking

This table contains information about each transaction.

|  |  |
| --- | --- |
| Ship tracking | |
| Attribute | Description |
| Ship\_id | Unique ship identifier of each which transport goods |
| Prod\_id | Product which transported |
| Insur\_id | Insurance of ship with products |
| Ar\_time | Time when ship was arrive into destination port |
| Ar\_port\_id | Unique arrival port identifier |
| Dep\_time | Time when ship was depart from destination port |
| Dep\_port\_id | Depart port identifier |
| Amount\_arrived\_goods | Amount of transported goods which was transported to destination point |
| Amount\_departed\_goods | Amount of transported goods which was transported from departure point |

1. Ships

This table contains information about all ships in our company.

|  |  |
| --- | --- |
| Ships | |
| Attribute | Description |
| Ship unique id | Unique ship identifier |
| Weight | Weight of ship |
| Height | Height of ship |
| Water volume | Water volume of ship |
| Max cargo | Max weight that ship can transport |

### Summarize data plan

Measures in fact table have to group by month.

Description:

Pct depart/arrive products will calculated as amount of departed products divided by amount of arrived products. All necessary data have to taken for source table “Ship tracking.tab” from columns “Amount depart goods” and “Amount arrive goods”.

Function for FCT\_INCOMES\_MM.PCT\_ARR\_DEP\_GOODS:

Measure type: no additive

Income from each ship– income coefficient of each product multiply by amount arrived products. All necessary data have to taken for source tables “Ship tracking” from column “Amount arrive goods” and “Products” from column “Income coefficient”.

Function for FCT\_INCOMES\_MM.INCOME:

Measure type: additive

AMOUNT\_TOT is additive column which is calculated by grouping each ship with its own properties by months(EVENT\_DT)

## Dimension types and hierarchies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dimensions | Type | Size | Merged dimensions | Description |
| DIM\_CUSTOMERS | SCD1 | Small | DW.T\_CUST  DW.LC\_CUST  DW.T\_COUNTRIES  DW.LC\_COUNTRIES | Contains information about customer companies |
| DIM\_PORTS | SCD1 | Small | DW.T\_PORTS  DW.T\_COUNTRIES  DW.LC\_PORTS  DW.LC\_COUNTRIES | Contains all information about ports |
| DIM\_TIMES\_5 | SCD1 | Big | DW.T\_DAYS  DW.T\_WEEKS  DW.T\_MONTHS  DW.T\_QUARTERS  DW.T\_YEARS  DW.LC\_DAYS  DW.LC\_MONTHS | Contains time |
| DIM\_SHIPS | SCD1 | Small | DW.T\_SHIPS  DW.LC\_SHIPS | Contains all information about company’s ships |
| DIM\_PRODUCTS\_SCD | SCD2  hybrid | Big | DW.T\_PRODUCTS  DW.LC\_PRODUCTS  DW.LC\_PROD\_CATEGORIES  DW.T\_PROD\_CATEGORIES  DW.T\_PROD\_HISTORY  DW.T\_CAT\_HISTORY  DW.T\_PROD\_CAT\_LINKS | Contains all information about products which are transported. |
| DIM\_INSURANCES | SC1 | Small | DW.T\_INSURANCES | Contains all info about insurances. |
| DIM\_GEN\_PERIODS | SC1 | Small | DW.T\_GEN\_PERIODS | Contains information how to group our facts |

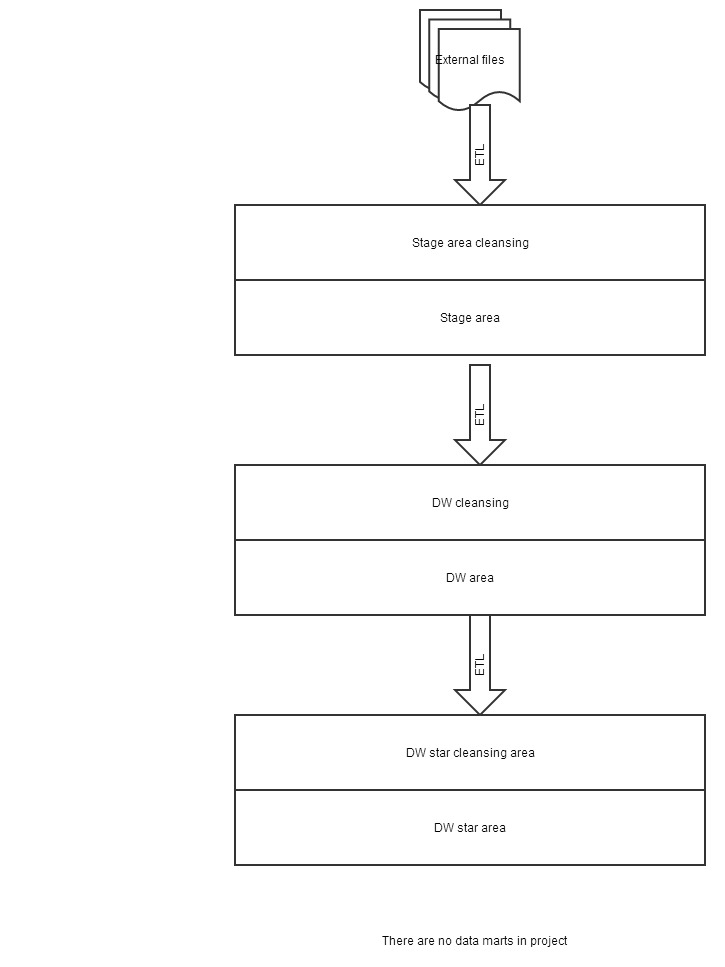
Hierarchies:

|  |  |  |  |
| --- | --- | --- | --- |
| Products | | | |
| Name | Level code | Level description | Natural level key |
| Products | Prod | All product in category | Prod\_id |
| Categories | Categ | All categories | Prod\_cat\_id |

Times

|  |  |  |  |
| --- | --- | --- | --- |
| Times | | | |
| Name | Level code | Level description | Natural level key |
| Days | Day | All days in week | Day\_id |
| Weeks | Week | All weeks in month | Week\_id |
| Months | Month | All months in quarter | Month\_id |
| Quarters | Quarter | All quarters in year | Quarter\_id |
| Years | Year | All years | Year\_id |

## Dataflow diagram



## Partitions

There is one table that should be partitioned: DW.T\_MONTHS.

It’s should be range partitions by EVENT\_DT column by years and some subpartitions(EVENT\_DT) by hash (composite partitioning):

PARTITION BY RANGE (EVENT\_DT)

subpartition by hash (EVENT\_DT) subpartitions 4

(

PARTITION part1 VALUES LESS THAN(sysdate,'dd/mm/yyyy'))

(

subpartition part1\_sub1,

subpartition part1\_sub2,

subpartition part1\_sub3,

subpartition part1\_sub4

),

PARTITION part2 VALUES LESS THAN(to\_date(maxvalue,'dd/mm/yyyy'))

(

subpartition part2\_sub1,

subpartition part2\_sub2,

subpartition part2\_sub3,

subpartition part2\_sub4

)

)

## Parallel execution

In current project we don’t beed any parallel executions because our customer hasn’t any money for multiprocessor system. We need to use only one processor.