****

**AutoChance**

**BI Solution Concept and Proposal**

**Prepared by:** [Aliaksandr Rymasheuski](mailto:aliaksandr_rymasheuski@epma.com)

**Issue date:** 25/07/2013

**1. Overview**

**1.1. Business Background**

Company ‘AutoChance’ (next the Company) has 9 years experience in auto business and it offices are located in different countries all over the world. The Company is well-known in this business and has a huge database of customers.

The Company buys cars from people who need money immediately. It pays 80-90 percent of the car’s market value, and the car’s owner gets money on the day of the contract. The Company just has to find a customer who will buy car at the market price.

**1.2. Benefits**

The Company needs to make detailed analysis of the sales by regions and countries to find out the most profitable models of cars. Find out parts of the population on which the Company’s sales will be oriented for future planning promotions and advertising courses.

**2. Requirements**

**2.1. Business Requirements**

|  |  |
| --- | --- |
| N/N | Requirements |
| B1 | The system should store the information about sales and buying cars |
| B2 | The expected time of execution of the analytical query: 1 min |
| B3 | Maximum allowed execution time of the analytical query: 5 min |
| B4 | Total profit rep month by country should be calculated at the last Friday of each month |

**2.2. Technical Requirements**

|  |  |
| --- | --- |
| N/N | Requirements |
| T1 | The system should be updated every Sunday at a fixed time: from 23.00 to 0.00 local time of the region where the office is located |
| T2 | The system should be located on 8 servers. These servers need to be as close as possible to the offices to quickly response |
| T3 | Backup of the system should be made every first day of the month |
| T4 | The expected growth of data per month: 1 GB |
| T5 | The system should have high information security |

**3. Solution Sketch**

**3.1. Source Tables structure**

Company has database that contains the following information:

* customers – all information about people who use the Company’s services.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Passport number | First name | Last name | Country | City | Adress | Gender |
| <customer’s passport number (unique)> | <customer’s first name> | <customer’s last name> | <country where customer lives> | <city where customer lives> | <customer’s adress> | <customer’s gender (M/F)> |

* cars – all information about cars

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Car stand number | Previous owner | Brand | Model | Year of production | Color | Date of purchase | Cost | Status |
| <unique car stand number> | <number of customers passport> | <brand of the car> | <model of the car> | <year when car was made> | <car’s color> | <date when car was bought> | <Cost of the car> | <in stock/sold> |

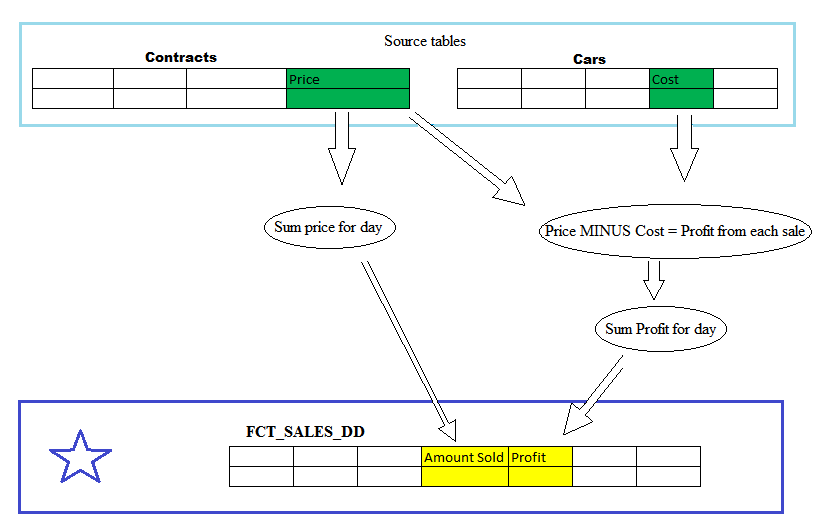
* employees – information about the Company’s employees

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Employee id | First name | Last name | Position | Office | Salary |
| <employee’s unique number> | <employee’s first name> | <employee’s last name> | <employee’s job position> | <employee’s office> | <employee’s salary> |

* contracts - information about transactions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Time | Contract number | Car stand number | Customer passport number | Employee id | Price |
| <when transaction was made> | <unique contract number> | <which car was sold or bought> | <customer’s passport> | <employee who was made transaction> | <total price of the car> |

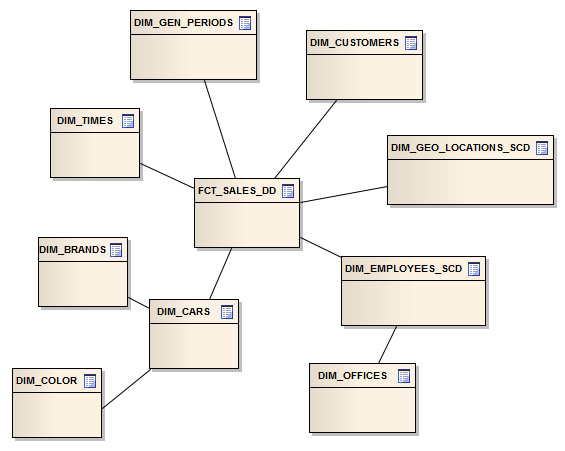
**3.2. Summarize Data Plan**

****

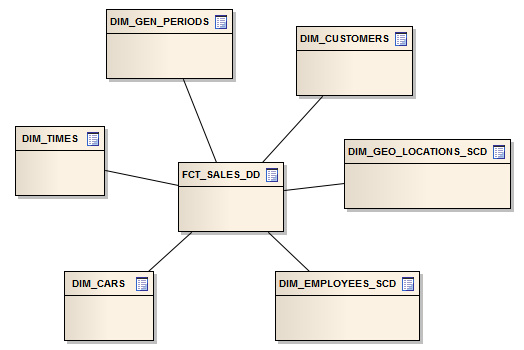
**4. DWH Solution Concept**

**4.1. Logical Diagram**

There are two methods to design the data warehouse: star-schema and snowflake-schema.

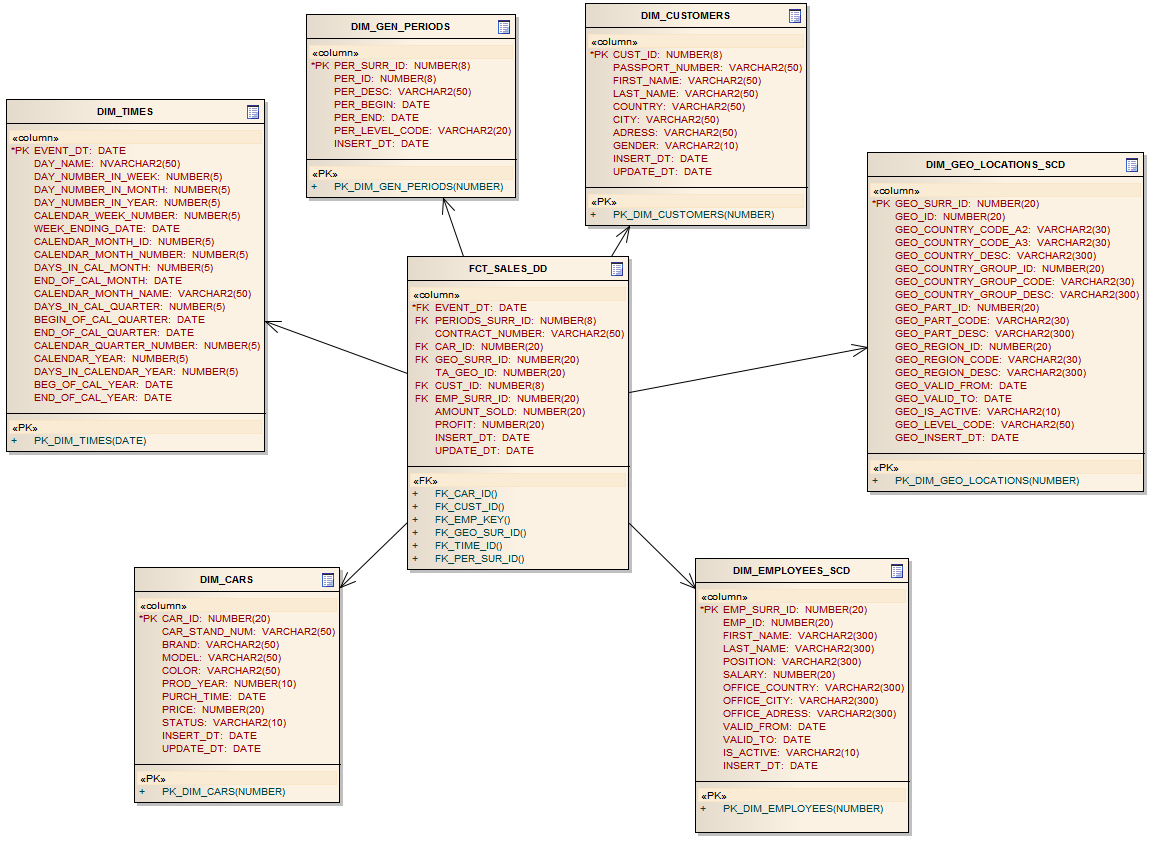


Snowflake logical diagram



Star logical diagram

The snowflake approach saves some storage space but queries against the snowflake will require additional joins, potentially affecting performance. Also more tables must be loaded by the ETL process. So, the star schema is better option.

**4.2. Physical diagram**

Star physical diagram

**4.3. Dimensions**

**4.3.1. Dimensions Types Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Size | Merged Dimensions | Descriptions |
| DIM\_TIMES | SCD1 | BIG | T\_DAYS, T\_WEEKS, T\_MONTHS, T\_QUARTERS,  T\_YEARS | Contains all time gradation to make continuous time in the system |
| DIM\_GEO\_LOCATIONS\_SCD | SCD2 | SMALL | T\_COUNTRIES  T\_CNTR\_GROUPS  T\_CNTR\_SUB\_GROUPS  LC\_CNTR\_GROUPS  T\_GEO\_TYPES  T\_GEO\_SYSTEMS  LC\_GEO\_SYSTEMS  T\_GEO\_PARTS  T\_GEO\_REGIONS  T\_GEO\_OBJECTS  T\_CNTR\_GROUP\_SYSTEMS  LC\_CNTR\_GROUP\_SYSTEMS  LC\_CNTR\_SUB\_GROUPS  LC\_ GEO\_PARTS  LC\_COUNTRIES  LC\_ GEO\_REGIONS | Contains all information about countries, with different levels of grouping |
| DIM\_EMPLOYEES\_SCD | SCD2 | SMALL | T\_EMPLOYEES,  T\_OFFICES,  LC\_EMPLOYEES,  LC\_OFFICES | Contains information about employees and offices with saving of the changing history |
| DIM\_CUSTOMERS | SCD1 | BIG | T\_CUSTOMERS,  LC\_CUSTOMERS | Contains information about customers |
| DIM\_CARS | SCD1 | BIG | T\_CARS,  LC\_BRANDS,  LC\_COLORS | Contains information about cars |
| DIM\_GEN\_PERIODS | SCD1 | SMALL | T\_PERIODS,  LC\_PERIODS | Contains information about different type periods |

**4.3.2. Dimensions Hierarchies**

**DIM \_TIME:**

**Hierarchy DAY –MONTH-QUARTER-YEAR**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | LEVEL\_CODE | LEVEL\_DESC | LEVEL\_NATURAL\_KEY |
| DAYs | DAY | Store all day at the calendar | DAY\_ID |
| MONTHs | MONTH | Store all months at the calendar year | MONTH\_ID |
| QUARTERs | QUARTER | Store all quarters at the calendar year | QUARTER\_ID |
| YEARs | YEAR | Store all years | YEAR\_ID |

**Hierarchy DAY-WEEK-MONTH-YEAR**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | LEVEL\_CODE | LEVEL\_DESC | LEVEL\_NATURAL\_KEY |
| DAYs | DAY | Store all day at the calendar | DAY\_ID |
| WEEKs | WEEK | Store all weeks at the calendar year | WEEK\_ID |
| MONTHs | MONTH | Store all months at the calendar year | MONTH\_ID |
| YEARs | YEAR | Store all years | YEAR\_ID |

**DIM\_CARS:**

**Hierarchy CAR-BRAND-MODEL**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | LEVEL\_CODE | LEVEL\_DESC | LEVEL\_NATURAL\_KEY |
| CARs | CAR | Store car’s stand number | CAR\_ID |
| BRANDs | BRAND | Store information about car’s brand | BRAND\_ID |
| MODELs | MODEL | Store all models of the cars | MODEL\_ID |

**DIM\_EMPLOYEES:**

**Hierarchy EMPLOYEE-OFFICE**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | LEVEL\_CODE | LEVEL\_DESC | LEVEL\_NATURAL\_KEY |
| EMPLOYEEs | EMPLOYEE | Store employee’s full name, id, salary and position | EMPLOYEE\_ID |
| OFFICEs | OFFICE | Store information about offices | OFFICE\_ID |

**DIM\_GEO\_LOCATIONS:**

**Hierarchy COUNTRY-COUNTRY\_GROUP-PART-REGION**

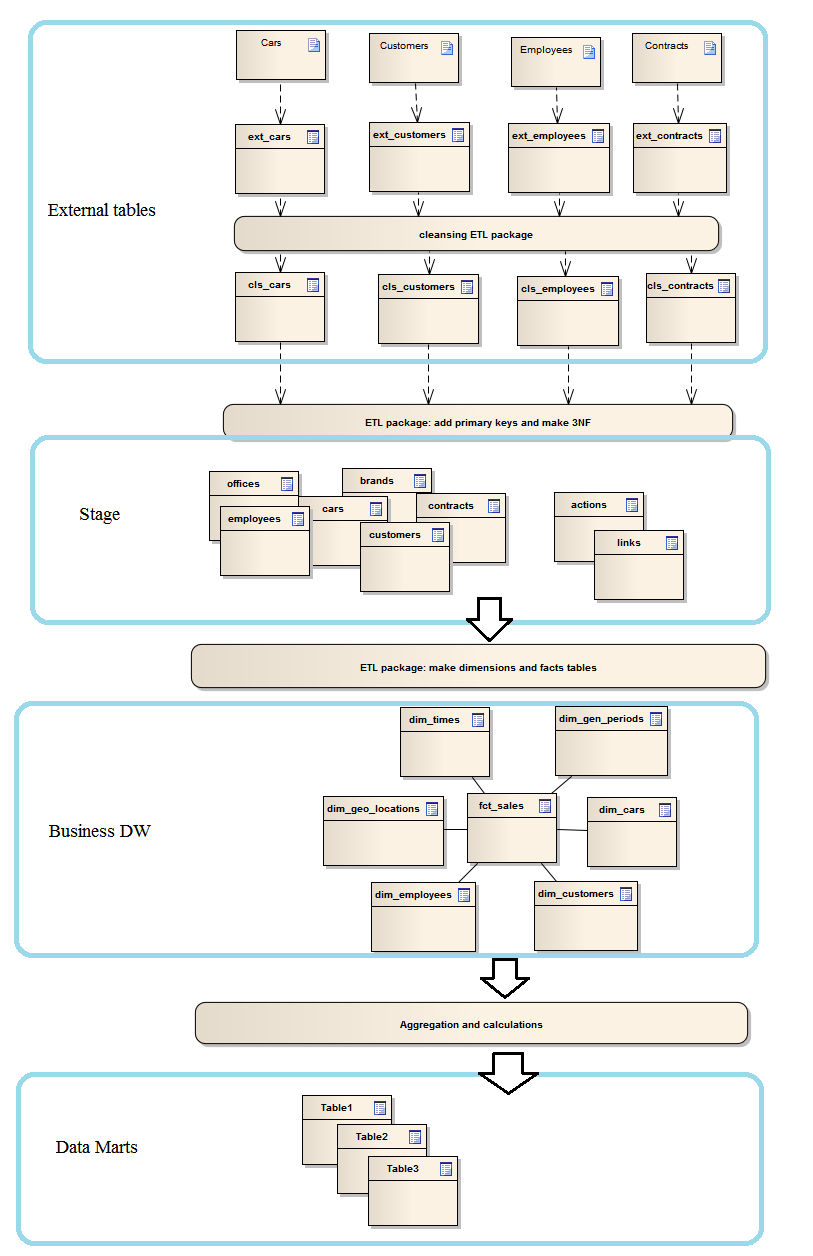
|  |  |  |  |
| --- | --- | --- | --- |
| Name | LEVEL\_CODE | LEVEL\_DESC | LEVEL\_NATURAL\_KEY |
| COUNTRies | COUNTRY | Store all countries in regions | COUNTRY\_ID |
| COUNTRY\_GROUP s | COUNTRY\_GROUP | Store all country groups in region | COUNTRY\_GROUP \_ID |
| PARTs | PART | Store all parts of the world | PART\_ID |
| REGIONs | REGION | Store all global regions | REGION\_ID |

**4.4. Facts**

**4.4.1 Facts Aggregations**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Code | Table Name | Additive | Descriptions |
| Amount of revenue | AMOUNT\_SOLD | FCT\_SALES | + | Calculate total amount of revenue per day |
| Amount of profit | PROFIT | FCT\_SALES | + | Calculate total profit of revenue per day |

**4.5. Dataflow Diagram**

****

**4.6. Partitioning rules**

The Company has a huge amount of customers. To increase availability and reduce administrative burden we make decision that table "customers" should be partitioned. We will use hash partitioning (8 partitions). Hash key column will be “customer\_id” because this column is unique and it will provide for a good spread of the rows across partitions.

To improve performance we make decision: table that contains information about contracts should be partitioned by range. The first partition of the table will contain all rows from the last two years. The second partition will contain all rows with old information.

**4.7. Strategy of Parallel execution**

Partitioned table “Customers” should be stored on different hard drives to provide effective execution of parallel queries. It will increase process of reading data and thus will improve effect of parallel of large queries.

All DML operations with large amounts of data should be executed in parallel for efficient use of time and resources.

Parallel DDL is a great opportunity for DBA to use the full capabilities of the hardware. The parallel DDL should be used for loading external tables.

Finally, all tables in DWH should be paralleled.