Usando Banco de dados colunares em projetos de BI

George Tavares

Banco de dados colunares

- Banco de dados tradicionais são organizados em linhas, porém análise de dados , usando group by star schema, acessa os dados em colunas.
- Solução : criar índices - Duplica dados e resolve somente para as colunas que foram criado índice
- Por que não armazenar somente os índices então, esquecer as linhas?

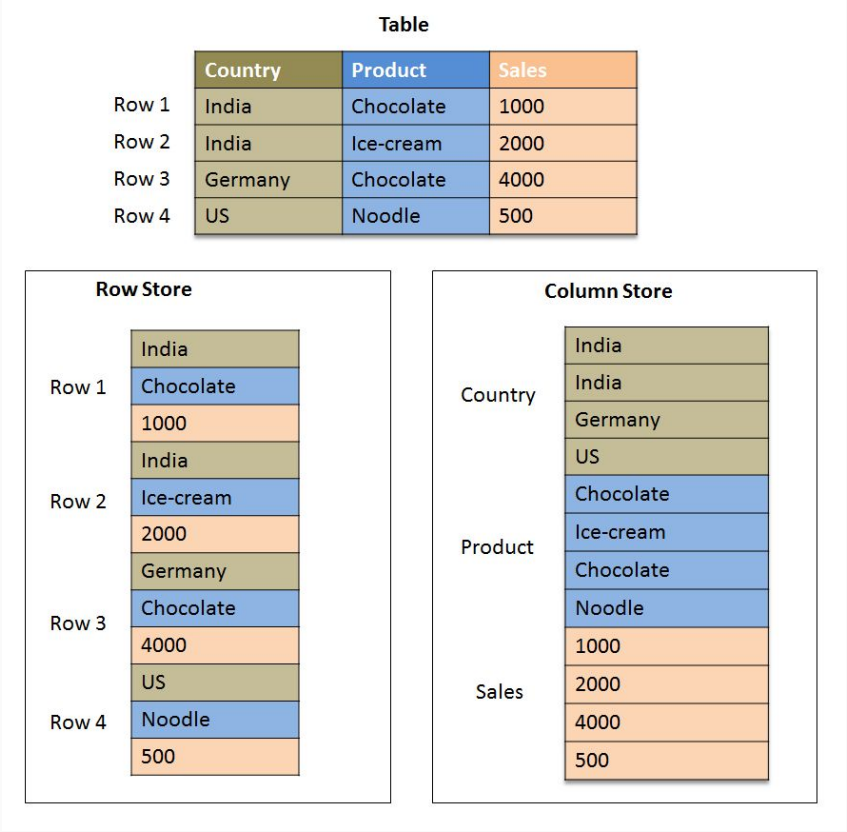
Organização dos dados

Open Source:

- MonetDB
- MariaDB ColumnStore (InfoBright)
- Postgres Citus DB

Comercial

- SAP Hana
- HP Vertica
- SQL Server columnstore index
- Oracle IM column store



saphanatutorial.com/column-data-storage-and-row-data-storage-sap-hana/

Fragmento da base de dados de censo americano de 1990 - 2.458.285 registros - 69 colunas

```
caseid,dAge,dAncstry1,dAncstry2,iAvail,iCitizen,iClass,dDepart,iDisabl1,iDisabl2,iEnglish,iFeb55,iFertil,dHispanic,dHour89,dHours,iImmigr,dIncome1,dIncome2,dIncome3,dIncome4,dIncome5,dIncome6,dIncome7,dIncome8,dIndustry,iKorean,iLang1,iLooking,iMarital,iMay75880,iMeans,iMilitary,iMobility,iMobillim,dOccup,iOthrserv,iPerscare,dPOB,dPoverty,dPwgt1,iRagechld,dRearning,iRelat1,iRelat2,iRemplpar,iRiders,iRlabor,iRownchld,dRpincome,iRPOB,iRrelchld,iRspouse,iRvetserv,iSchool,iSept80,iSex,iSubfam1,iSubfam2,iTmpabsnt,dTravtime,iVietnam,dWeek89,iWork89,iWorklwk,iWWII,iYearsch,iYearwrk,dYrsserv
10000,5,0,1,0,0,5,3,2,2,1,0,1,0,4,3,0,2,0,0,1,0,0,0,0,10,0,1,0,1,0,1,4,2,2,3,0,2,0,2,1,4,3,0,0,0,3,1,0,3,22,0,3,0,1,0,1,0,0,0,5,0,2,1,1,0,11,1,0
10001,6,1,1,0,0,7,5,2,2,0,0,3,0,1,1,0,1,0,0,0,0,1,0,0,4,0,2,0,0,0,1,4,1,2,2,0,2,0,2,2,4,2,1,0,0,1,1,0,2,10,0,1,0,1,0,1,0,0,0,1,0,2,1,1,0,5,1,0
10002,3,1,2,0,0,7,4,2,2,0,0,1,0,4,4,0,1,0,1,0,0,0,0,0,1,0,2,0,4,0,10,4,1,2,4,0,2,0,2,1,4,2,2,0,0,0,1,0,2,10,0,6,0,1,0,1,0,0,0,2,0,2,1,1,0,10,1,0
10003,4,1,2,0,0,1,3,2,2,0,0,3,0,3,3,0,1,0,0,0,0,0,0,1,4,0,2,0,2,0,1,4,1,2,2,0,2,0,2,1,2,2,0,0,0,1,1,0,2,10,0,4,0,1,0,1,0,0,0,1,0,1,1,1,0,10,1,0
10004,7,1,1,0,0,0,0,2,2,0,0,3,0,0,0,0,0,0,0,0,1,0,0,0,0,0,2,2,0,0,0,4,1,2,0,0,2,0,2,0,2,1,4,0,1,0,0,0,6,0,2,22,0,1,0,1,0,1,0,0,3,0,0,0,2,2,0,5,6,0
10005,1,1,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,4,0,0,0,1,0,0,0,0,0,2,0,4,0,2,0,121,0,0,1,0,10,1,0,0,2,0,1,0,0,0,0,0,0,0,0,0,4,0,0
10006,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,4,0,0,0,1,0,0,0,0,0,2,1,0,0,2,0,121,0,0,1,0,10,1,0,0,2,0,0,0,0,0,0,0,0,0,0,0,4,0,0
10007,4,1,2,0,0,6,0,2,2,0,0,4,0,5,5,0,2,1,0,0,0,0,0,0,9,0,2,0,0,0,11,4,1,2,3,0,2,0,2,1,2,3,1,0,0,0,1,0,3,10,0,1,0,1,0,1,0,0,0,0,0,2,1,1,0,11,1,0
10008,6,1,1,0,0,1,0,2,2,0,0,7,0,1,0,0,1,0,0,0,0,0,0,0,7,0,2,2,0,0,0,4,1,2,6,0,2,0,2,1,2,2,1,0,0,0,6,0,2,10,0,1,0,1,0,1,0,0,3,0,0,1,1,2,0,10,1,0
10009,3,1,12,0,0,1,0,2,2,0,0,0,0,3,0,0,1,0,0,0,0,0,0,0,4,0,2,2,4,0,0,2,1,2,6,0,2,0,2,1,0,2,2,0,0,0,3,0,2,10,0,6,1,1,1,0,0,0,1,0,0,1,1,2,0,8,1,1
10010,5,11,1,0,0,1,3,2,2,0,0,0,0,3,3,0,3,0,0,0,0,0,0,0,6,0,2,0,0,0,1,4,1,2,1,0,2,0,2,1,0,4,0,0,0,1,1,0,4,21,0,1,0,1,0,0,0,0,0,4,0,2,1,1,0,11,1,0
10011,4,0,1,0,0,1,5,2,2,0,0,3,0,1,1,0,1,0,0,0,0,0,0,0,7,0,2,0,0,0,1,4,1,2,1,0,2,0,2,1,3,2,1,0,0,1,1,0,2,10,0,1,0,1,0,1,0,0,0,2,0,2,1,1,0,10,1,0
10012,1,1,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,0,0,0,0,0,0,0,0,0,2,0,0,0,2,0,112,0,0,1,0,10,1,0,0,1,0,0,0,0,0,0,0,0,0,2,0,1,0,0
```

https://archive.ics.uci.edu/ml/datasets/US+Census+Data+%281990%29

Field	Type	Null	Key	Default	Extra
caseid	bigint(20)	YES		NULL	
dAge	bigint(20)	YES		NULL	
dAncstry1	bigint(20)	YES		NULL	
dAncstry2	bigint(20)	YES		NULL	
iAvail	bigint(20)	YES		NULL	
iCitizen	bigint(20)	YES		NULL	
iClass	bigint(20)	YES		NULL	
dDepart	bigint(20)	YES		NULL	
iDisabl1	bigint(20)	YES		NULL	
iDisabl2	bigint(20)	YES		NULL	
iEnglish	bigint(20)	YES		NULL	
iFeb55	bigint(20)	YES		NULL	
iFertil	bigint(20)	YES		NULL	
dHispanic	bigint(20)	YES		NULL	
dHour89	bigint(20)	YES		NULL	
dHours	bigint(20)	YES		NULL	
iImmigr	bigint(20)	YES		NULL	
dIncome1	bigint(20)	YES		NULL	
dIncome2	bigint(20)	YES		NULL	
dIncome3	bigint(20)	YES		NULL	
dIncome4	bigint(20)	YES		NULL	
dIncome5	bigint(20)	YES		NULL	
dIncome6	bigint(20)	YES		NULL	
dIncome7	bigint(20)	YES		NULL	
dIncome8	bigint(20)	YES		NULL	
dIndustry	bigint(20)	YES		NULL	
iKorean	bigint(20)	YES		NULL	
iLang1	bigint(20)	YES		NULL	
iLooking	bigint(20)	YES		NULL	
iMarital	bigint(20)	YES		NULL	
iMay75880	bigint(20)	YES		NULL	
iMeans	bigint(20)	YES		NULL	
iMilitary	bigint(20)	YES		NULL	
iMobility	bigint(20)	YES		NULL	
iMobillim	bigint(20)	YES		NULL	

Field	Type	Null	Key	Default	Extra
dOccup	bigint(20)	YES		NULL	
iOthrserv	bigint(20)	YES		NULL	
iPerscare	bigint(20)	YES		NULL	
dPOB	bigint(20)	YES		NULL	
dPoverty	bigint(20)	YES		NULL	
dPwgt1	bigint(20)	YES		NULL	
iRagechld	bigint(20)	YES		NULL	
dRearning	bigint(20)	YES		NULL	
iRelat1	bigint(20)	YES		NULL	
iRelat2	bigint(20)	YES		NULL	
iRemplpar	bigint(20)	YES		NULL	
iRiders	bigint(20)	YES		NULL	
iRlabor	bigint(20)	YES		NULL	
iRownchld	bigint(20)	YES		NULL	
dRpincome	bigint(20)	YES		NULL	
iRPOB	bigint(20)	YES		NULL	
iRrelchld	bigint(20)	YES		NULL	
iRspouse	bigint(20)	YES		NULL	
iRvetserv	bigint(20)	YES		NULL	
iSchool	bigint(20)	YES		NULL	
iSept80	bigint(20)	YES		NULL	
iSex	bigint(20)	YES		NULL	
iSubfam1	bigint(20)	YES		NULL	
iSubfam2	bigint(20)	YES		NULL	
iTmpabsnt	bigint(20)	YES		NULL	
dTravtime	bigint(20)	YES		NULL	
iVietnam	bigint(20)	YES		NULL	
dWeek89	bigint(20)	YES		NULL	
iWork89	bigint(20)	YES		NULL	
iWorklwk	bigint(20)	YES		NULL	
iWWII	bigint(20)	YES		NULL	
iYearsch	bigint(20)	YES		NULL	
iYearwrk	bigint(20)	YES		NULL	
dYrsserv	bigint(20)	YES		NULL	

Mysql

```
MariaDB [test]> select dAge,count(*) from census group by dAge order by 1;
+------+----------+
| dAge | count(*) |
+------+----------+
|    0 |    32169 |
|    1 |   441248 |
|    2 |   242511 |
|    3 |   370955 |
|    4 |   404535 |
|    5 |   312825 |
|    6 |   331258 |
|    7 |   322784 |
+------+----------+
8 rows in set (21.54 sec)
```

Mysql - indice no dAge

```
MariaDB [test]> select dAge,count(*) from census2 group by dAge order by 1;
+------+----------+
| dAge | count(*) |
+------+----------+
|    0 |    32169 |
|    1 |   441248 |
|    2 |   242511 |
|    3 |   370955 |
|    4 |   404535 |
|    5 |   312825 |
|    6 |   331258 |
|    7 |   322784 |
+------+----------+
8 rows in set (1.65 sec)
```

Mysql - indice no dAge

```
MariaDB [test]> select dAge,iSex,count(*) from census2 group by dAge,iSex order by 1,2;
+------+------+----------+
| dAge | iSex | count(*) |
+------+------+----------+
|    0 |    0 |    16419 |
|    0 |    1 |    15750 |
|    1 |    0 |   225807 |
|    1 |    1 |   215441 |
|    2 |    0 |   124344 |
|    2 |    1 |   118167 |
|    3 |    0 |   185246 |
|    3 |    1 |   185709 |
|    4 |    0 |   198566 |
|    4 |    1 |   205969 |
|    5 |    0 |   152989 |
|    5 |    1 |   159836 |
|    6 |    0 |   157523 |
|    6 |    1 |   173735 |
|    7 |    0 |   130707 |
|    7 |    1 |   192077 |
+------+------+----------+
16 rows in set (20.91 sec)
```

MonetDB

```
sql>select dAge,count(*) from census group by dAge order by 1;
+------+--------+
| dage | L3     |
+======+========+
|    0 |  32169 |
|    1 | 441248 |
|    2 | 242511 |
|    3 | 370955 |
|    4 | 404535 |
|    5 | 312825 |
|    6 | 331258 |
|    7 | 322784 |
+------+--------+
8 tuples (23.096ms)
```

MonetDB

```
sql>select dAge,iSex,count(*) from census group by dAge,iSex order by 1,2;
+------+------+--------+
| dage | isex | L4     |
+======+======+========+
|    0 |    0 |  16419 |
|    0 |    1 |  15750 |
|    1 |    0 | 225807 |
|    1 |    1 | 215441 |
|    2 |    0 | 124344 |
|    2 |    1 | 118167 |
|    3 |    0 | 185246 |
|    3 |    1 | 185709 |
|    4 |    0 | 198566 |
|    4 |    1 | 205969 |
|    5 |    0 | 152989 |
|    5 |    1 | 159836 |
|    6 |    0 | 157523 |
|    6 |    1 | 173735 |
|    7 |    0 | 130707 |
|    7 |    1 | 192077 |
+------+------+--------+
16 tuples (38.761ms)
```

MonetDB

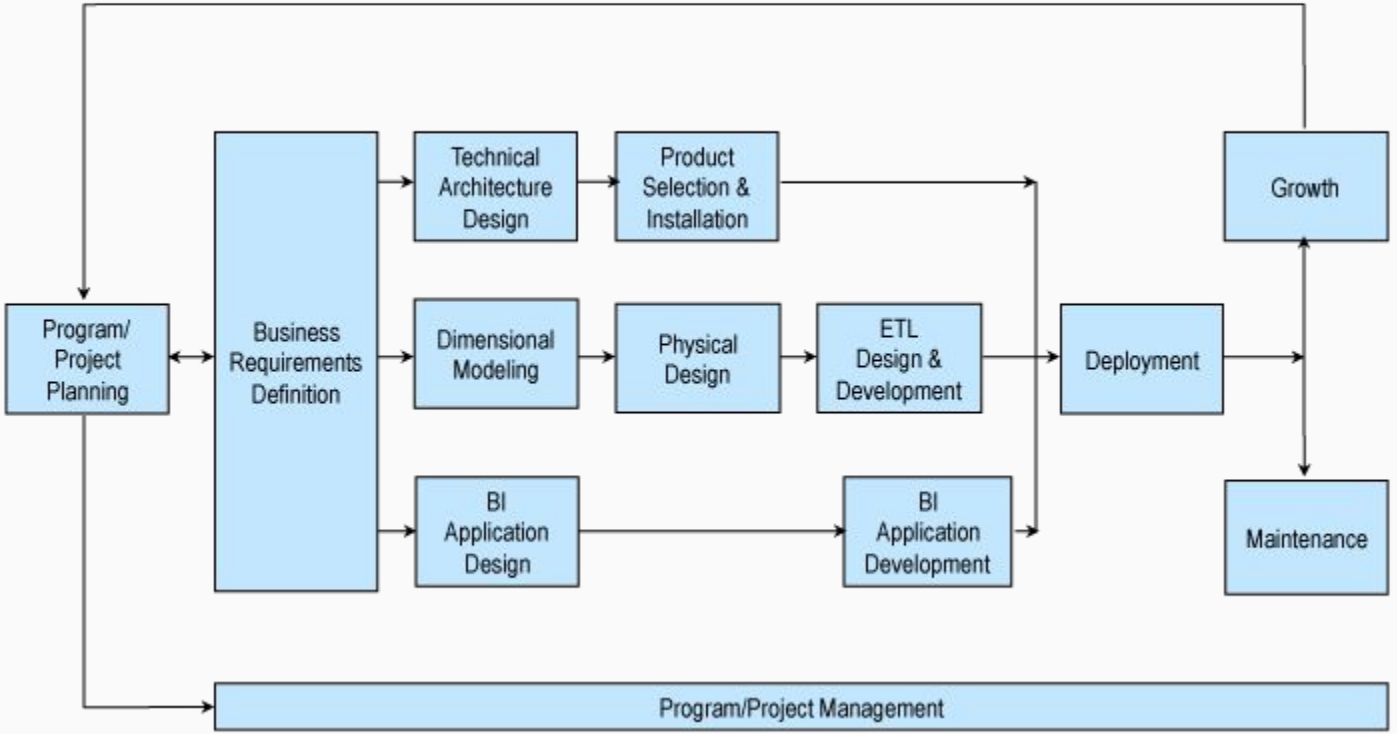
```
sql>select dPOB,dOccup,iClass,dIndustry,count(*) from census group by
dPOB,dOccup,iClass,dIndustry order by 1,2,3,4;

+------+--------+--------+-----------+--------+
| dpob | doccup | iclass | dindustry | L6     |
+======+========+========+===========+========+
|    0 |      0 |      0 |         0 | 927701 |
|    0 |      1 |      1 |         1 |   1156 |
|    0 |      1 |      1 |         2 |   1462 |
|    0 |      1 |      1 |         3 |   6382 |
|    0 |      1 |      1 |         4 |  35298 |
|    0 |      1 |      1 |         5 |  10166 |
|    0 |      1 |      1 |         6 |   6530 |
…
|    6 |      8 |      9 |        12 |     32 |
+------+--------+--------+-----------+--------+
2152 tuples (77.587ms)
```

Não é a bala de prata.

Para atualizar uma linha, é muito mais custoso.
Mesmo o processo de carga de dados deve ser feito em lote,tudo de uma vez.
Alterações geralmente são registradas em deltas e posteriormente integradas nas colunas.

Ciclo de vida de projeto BI/DW

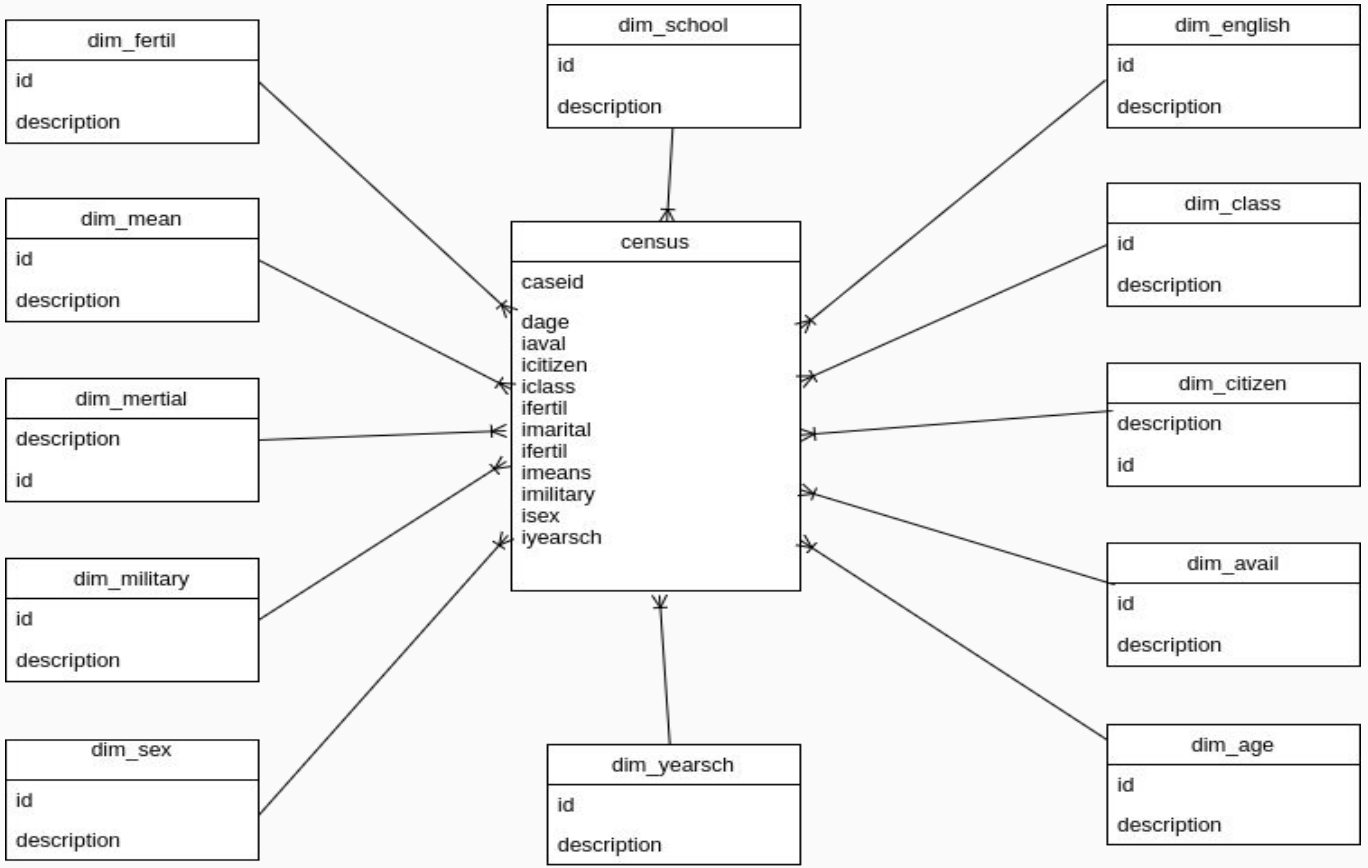


http://www.kimballgroup.com/data-warehouse-business-intelligence-resources/kimball-techniques/dw-bi-lifecycle-method/

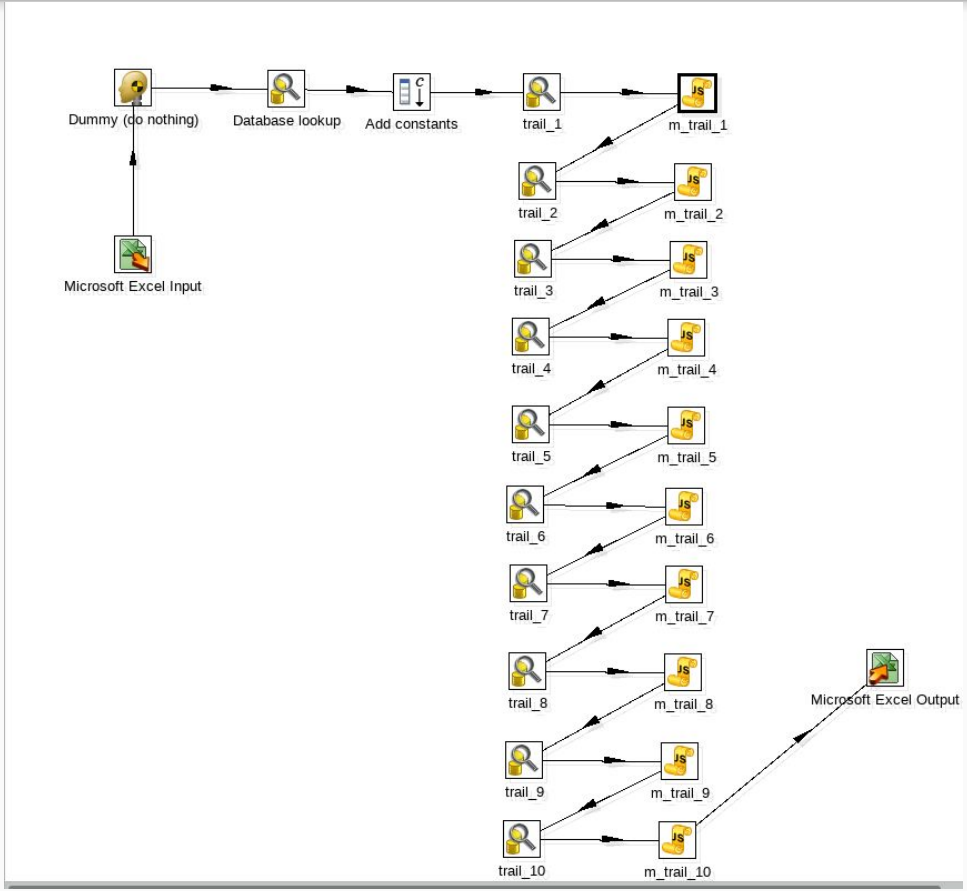
Seleção de ferramentas

Banco de Dados	MonetDB	SAP Hana SQL Server column index
Ferramenta OLAP	Mondrian	Analysis Server
Ferramenta ETL	Pentaho Data Integration	Integration Services
Ferramenta Visualização	Saiku BA Pentaho BA	Tableau PowerBI

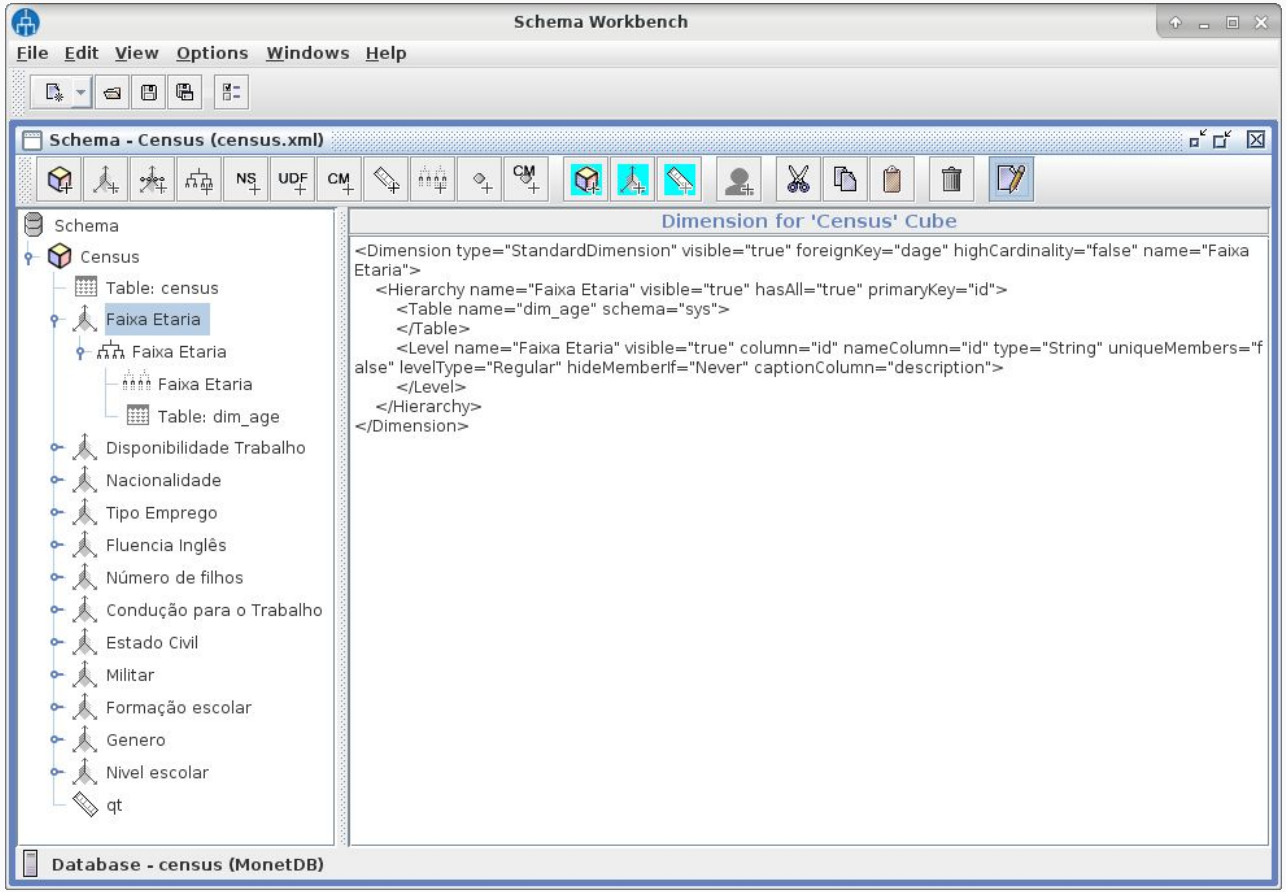
Modelo ER - Star Schema



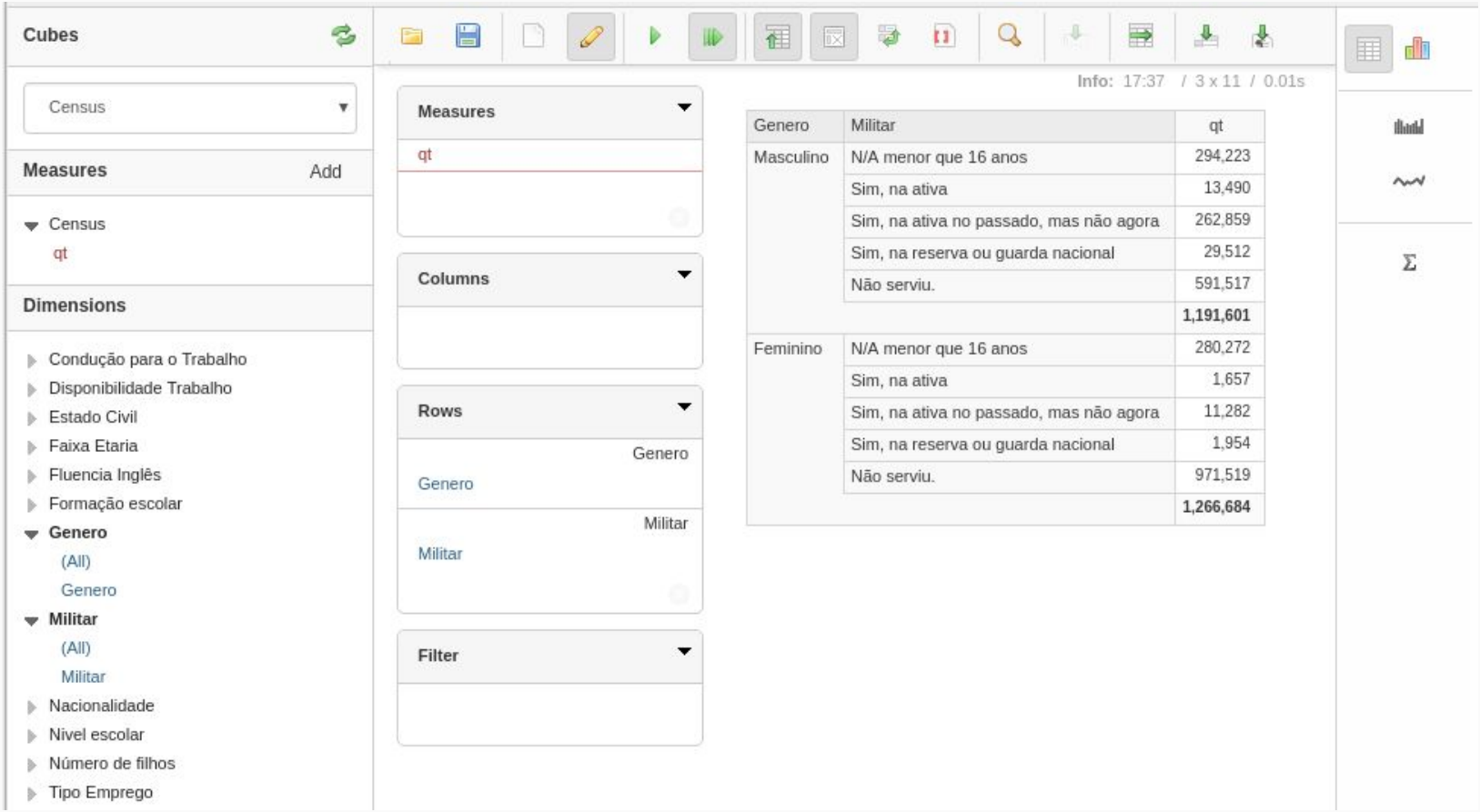
ETL Data Integration



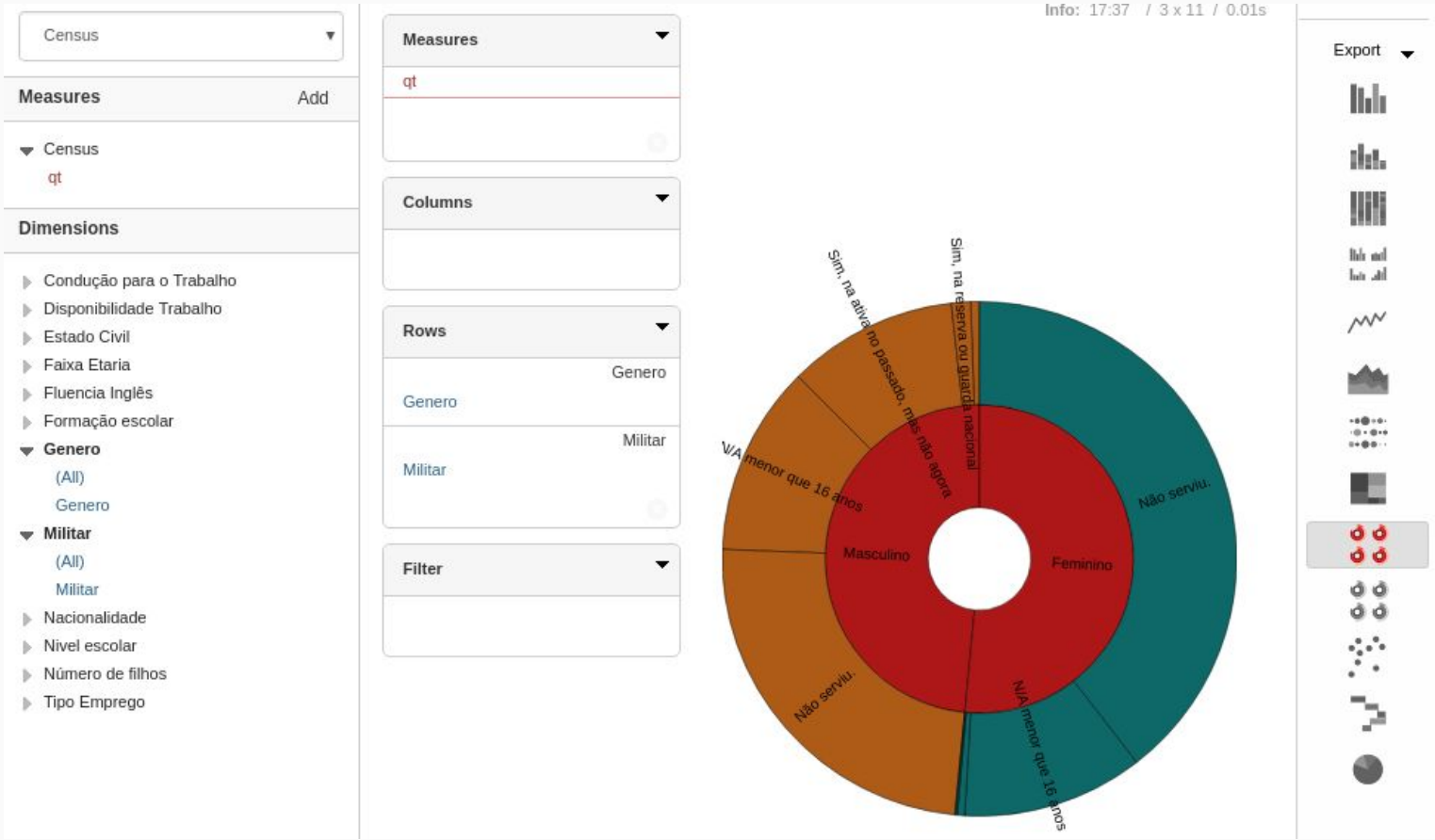
Modelagem do Cubo



Visualização Saiku 1

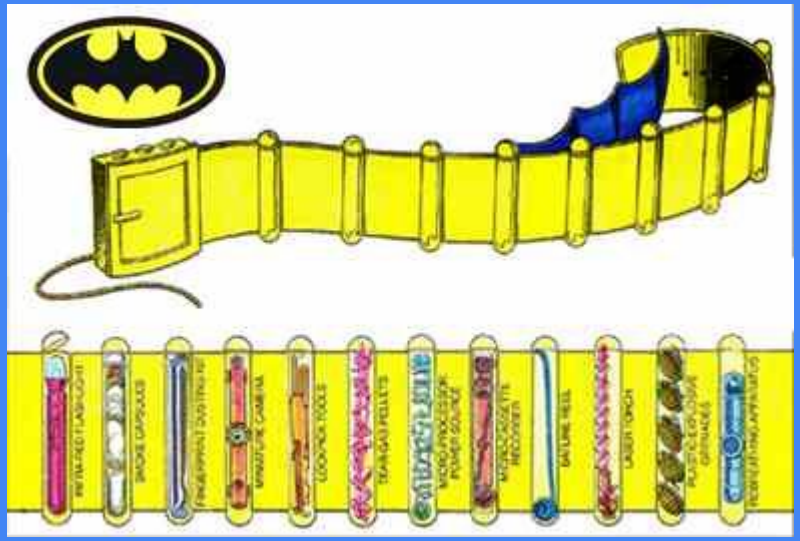


Visualização Saiku 2



Mais recursos para seu cinto de utilidades.

Não podemos usar a marreta para tudo



http://smokingpot.org/os-10-gadges-de-filmes-que-eu-queria-ter/

Obrigado!

George Tavares
@tanquetav
https://github.com/tanquetav

https://github.com/tanquetav/census