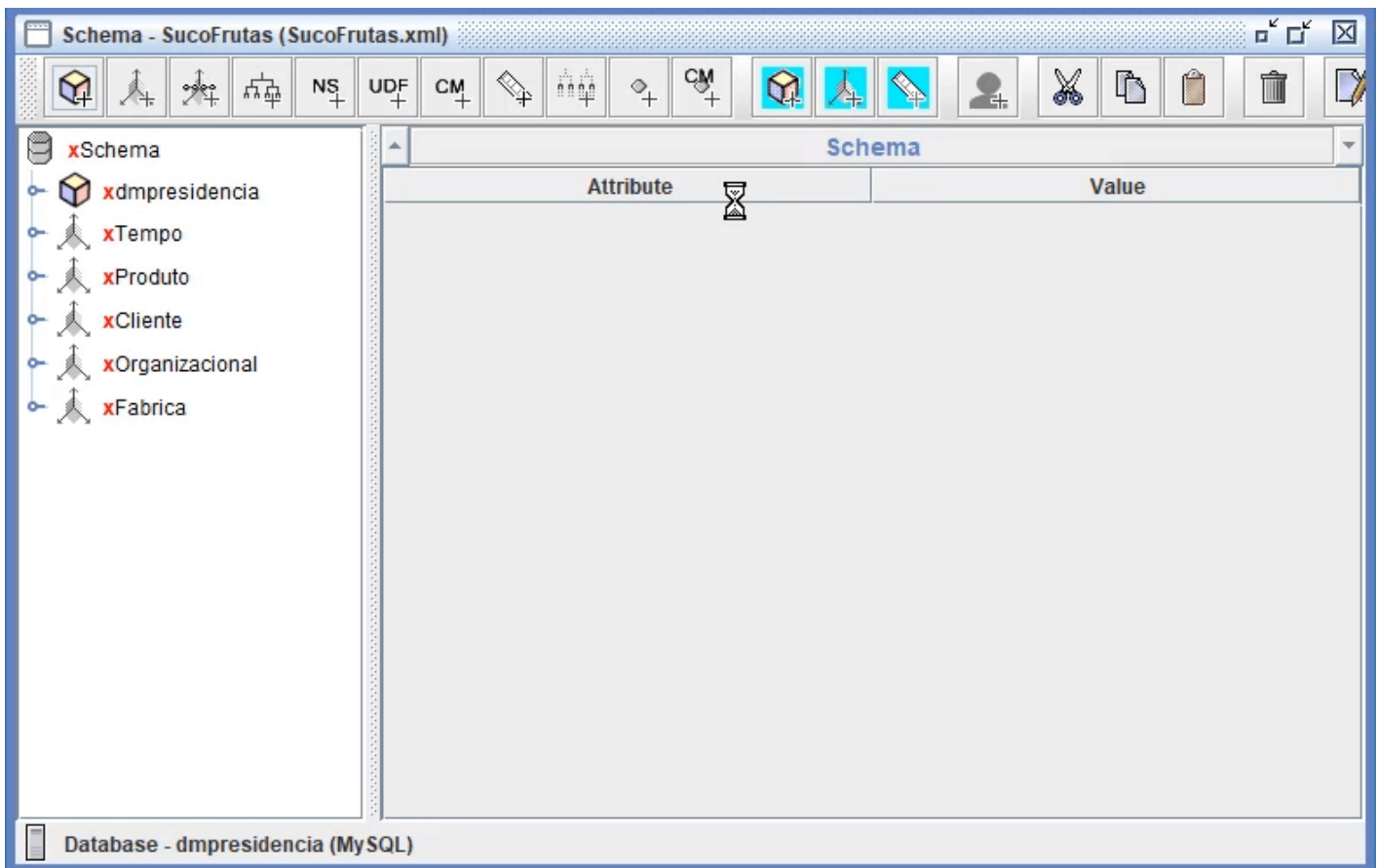


1) O primeiro passo para restringir o acesso à base OLAP é criar regras (*roles*) na configuração do cubo OLAP. Para isso, no **Pentaho Workbench**, abra o XML **SucosFrutas.xml**:



2) Clique com o botão direito do mouse sobre o esquema e selecione **Add Role**. Crie a *role* chamada **Administrator**:

Role for 'SucoFrutas' Schema	
Attribute	Value
name	Administrator

3) Crie uma outra regra, chamada **Sudeste**:

Role for 'SucoFrutas' Schema	
Attribute	Value
name	Sudeste

4) E mais uma chamada **Authenticated**:

Role for 'SucoFrutas' Schema	
Attribute	Value
name	Authenticated

5) Clique com o botão direito do mouse sobre a regra **Administrator** e escolha **Add Schema Grant**. Inclua:

Schema Grant	
Attribute	Value
access	all

6) Faça o mesmo para **Authenticated**, mas a seleção será diferente:

Role for 'SucoFrutas' Schema	
Attribute	Value
name	Authenticated

7) Já para a regra **Sudeste**, crie um *Schema Grant* como mostrado abaixo:

Schema Grant	
Attribute	Value
access	all_dimensions

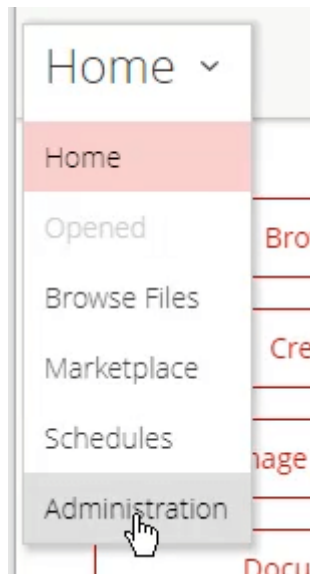
8) Abaixo deste *Schema Grant*, adicione um *Cube Grant*:

Cube Grant	
Attribute	Value
access	all
cube	dmpresidencia

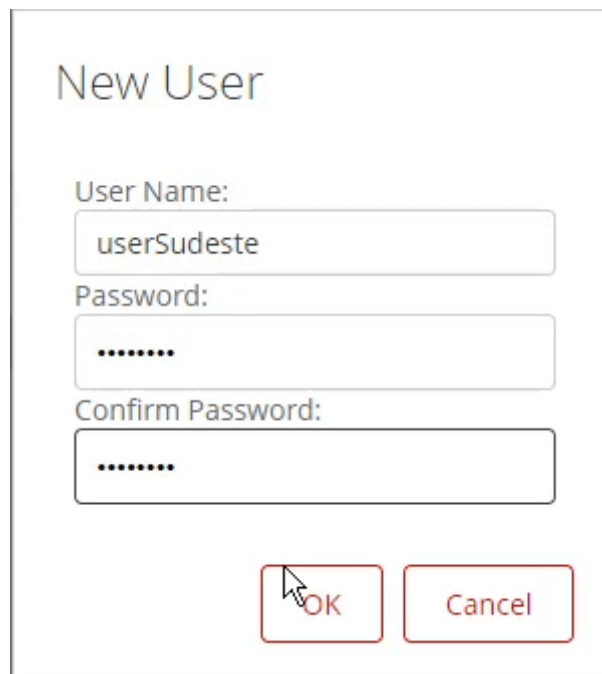
9) Abaixo de *Cube Grant*, crie um *Hierarchy Grant*:

Hierarchy Grant	
Attribute	Value
access	custom
hierarchy	[Cliente.Cliente]
topLevel	[Cliente.Cliente].[Regiao]
bottomLevel	
rollupPolicy	

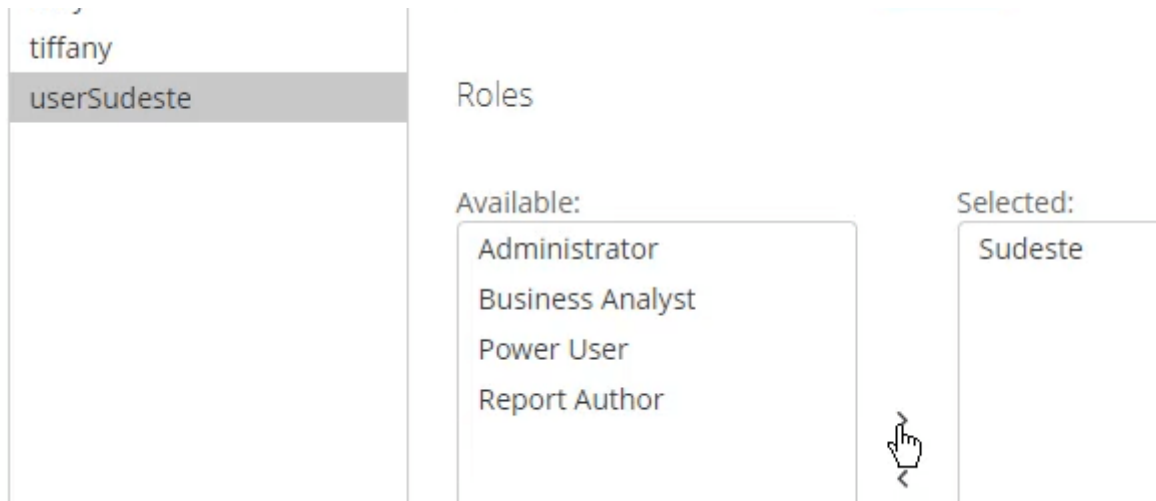
- 10) Salve o XML e publique-o no **Pentaho Server**, não esquecendo de, ao abrir novamente o **Pentaho User Console**, escolher no menu a opção **Tools --> Refresh --> Mondrian Schema Cache**.
- 11) Para garantir que a mudança será registrada, pare e inicialize o servidor Pentaho.
- 12) Ainda no **Pentaho User Console**, clique em **Home --> Administration**:



- 13) Em **Users & Roles**, na aba **Manage Users**, clique no botão **+** e adicione um novo usuário, chamado **userSudeste**:



- 14) Com o usuário **userSudeste** selecionado, adicione a regra **Sudeste** a ele:



15) Vá no **Saiku** e crie a visão abaixo:

Medidas

Faturamento

Colunas

Linhas

Região

Cliente

Região	Faturamento
Centro Oeste	97.145.592,33
Nordeste	12.672.441,51
Sudeste	94.618.730,62
Sul	32.543.988,78

Note que, ao selecionar todas as regiões, o relatório exhibe as quatro regiões contidas na base.

16) Salve esta visão e faça *logout* do **Pentaho User Console**.

17) Faça *login* com o usuário **userSudeste**, vá no **Saiku** e selecione a mesma visão salva. Você verá:

Medidas

Faturamento

Colunas

Linhas

Região

Cliente

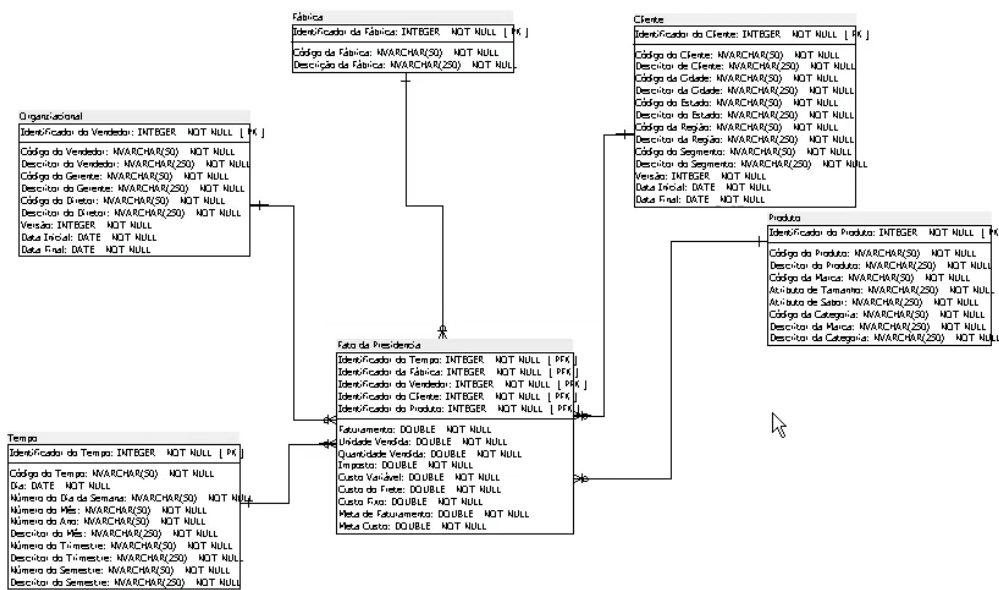
Região	Faturamento
Sudeste	94.618.730,62

Ou seja, os dados estão limitados.

18) Teste o uso de tabelas agregadas. Levando em consideração que:

- Os *logs* do **OLAP Pentaho** devem ter sido ativados
- Toda a tabela agregada deve ter, obrigatoriamente, um campo de contador de linha

19) Abra o **SQL Power Architecture** e acesse o modelo do *Data Mart*:



20) Crie uma nova tabela agregada, como mostrado abaixo:

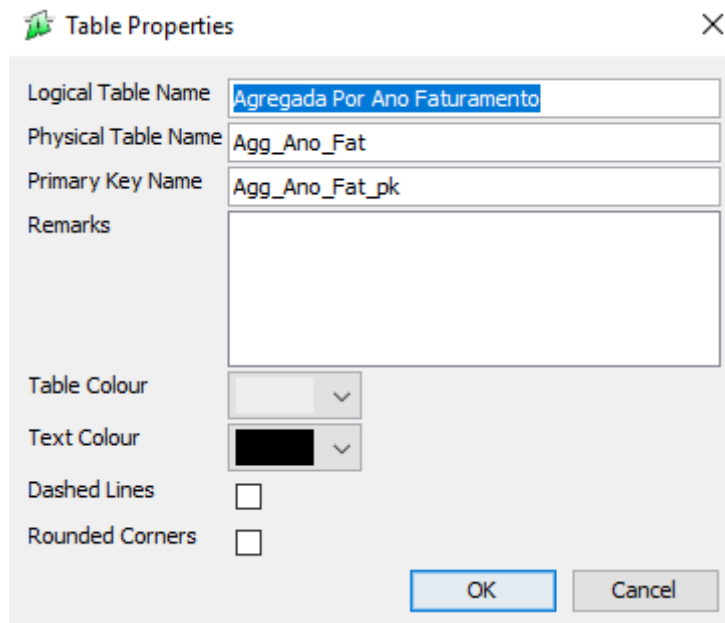


Table Properties

Logical Table Name: Agregada Por Ano Faturamento

Physical Table Name: Agg_Ano_Fat

Primary Key Name: Agg_Ano_Fat_pk

Remarks:

Table Colour: [Color Selection]

Text Colour: [Color Selection]


Dashed Lines: ☐

Rounded Corners: ☐

OK Cancel

21) Crie os seguintes campos:

- Logical Name: **Ano**
- Physical Name: **Cod_Ano**
- In Primary Key: **Sim**
- Type: **VARCHAR**
- Precision: **50**

 Column Properties of Ano ×

Source for ETL Mapping
None Specified

Logical Name
Ano

Physical Name
Cod_Ano

☒ In Primary Key

Type
VARCHAR

Precision ☒ 50 **Scale** ☐ 0

Allows Nulls
☒ No

Auto Increment
☒ No

Default Value
☒

Sequence Name (Only applies to target platforms that use sequences)
Agg_Ano_Fat_Cod_Ano_seq

- *Logical Name:* **Identificador do Vendedor**
- *Physical Name:* **ID_Vendedor**
- *In Primary Key:* **Sim**
- *Type:* **INTEGER**

Column Properties of Identificador do Vendedor

Source for ETL Mapping
None Specified

Logical Name
Identificador do Vendedor

Physical Name
ID_Vendedor

☒ In Primary Key

Type
INTEGER

Precision ☐ 0 Scale ☐ 0

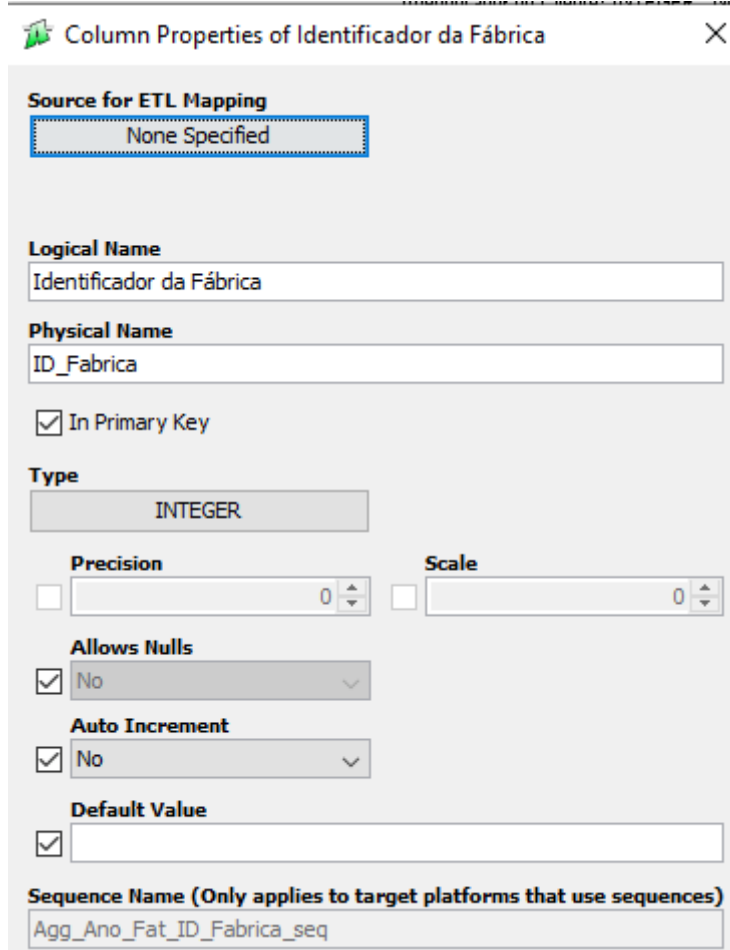
☒ Allows Nulls
No

☒ Auto Increment
No

☒ Default Value

Sequence Name (Only applies to target platforms that use sequences)
Agg_Ano_Fat_ID_Vendedor_seq

- Logical Name: **Identificador da Fábrica**
- Physical Name: **ID_Fabrica**
- In Primary Key: **Sim**
- Type: **INTEGER**



Column Properties of Identificador da Fábrica

Source for ETL Mapping
None Specified

Logical Name
Identificador da Fábrica

Physical Name
ID_Fabrica

☒ In Primary Key

Type
INTEGER

Precision
☐ 0

Scale
☐ 0


Allows Nulls
☒ No

Auto Increment
☒ No

Default Value
☒

Sequence Name (Only applies to target platforms that use sequences)
Agg_Ano_Fat_ID_Fabrica_seq

- *Logical Name:* **Identificador do Cliente**
- *Physical Name:* **ID_Cliente**
- *In Primary Key:* **Sim**
- *Type:* **INTEGER**

 Column Properties of Identificador do Cliente ✕

Source for ETL Mapping
None Specified

Logical Name
Identificador do Cliente

Physical Name
ID_Cliente

☒ In Primary Key

Type
INTEGER

Precision
☐ 0

Scale
☐ 0


Allows Nulls
☒ No

Auto Increment
☒ No

Default Value
☒

Sequence Name (Only applies to target platforms that use sequences)
Agg_Ano_Fat_ID_Cliente_seq

- *Logical Name:* **Identificador do Produto**
- *Physical Name:* **ID_Produto**
- *In Primary Key:* **Sim**
- *Type:* **INTEGER**

 Column Properties of Identificador do Produto ×

Source for ETL Mapping
None Specified

Logical Name
Identificador do Produto

Physical Name
ID_Produto

☒ In Primary Key

Type
INTEGER

Precision
☐

Scale
☐

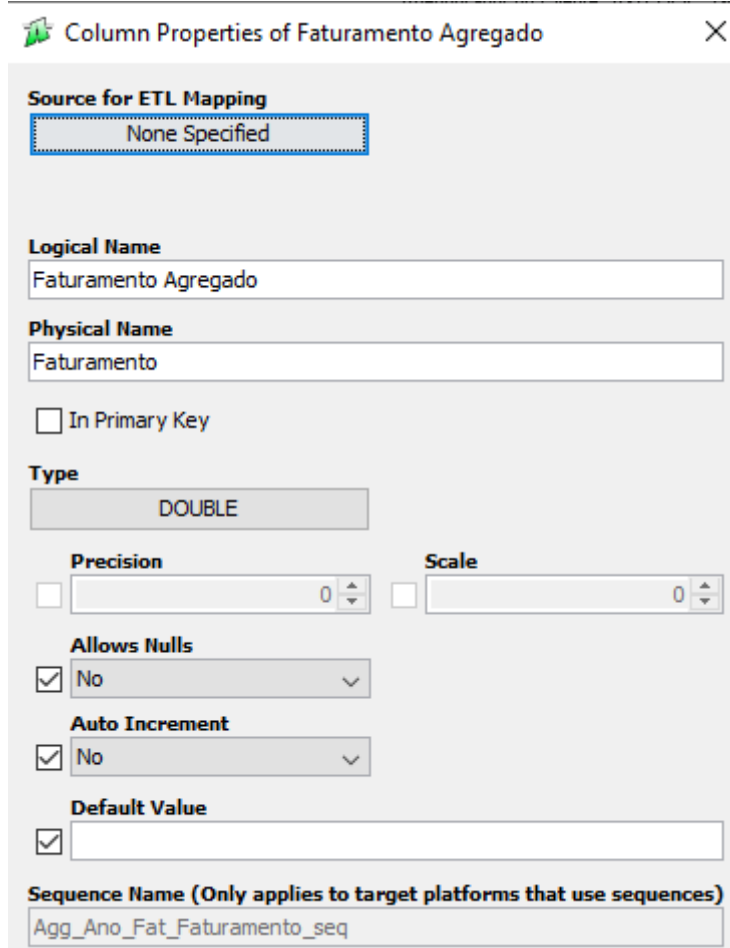
☒ **Allows Nulls**
No

☒ **Auto Increment**
No

☒ **Default Value**

Sequence Name (Only applies to target platforms that use sequences)
Agg_Ano_Fat_ID_Produto_seq

- *Logical Name:* **Faturamento Agregado**
- *Physical Name:* **Faturamento**
- *In Primary Key:* **Não**
- *Type:* **DOUBLE**



Column Properties of Faturamento Agregado

Source for ETL Mapping
None Specified

Logical Name
Faturamento Agregado

Physical Name
Faturamento

☐ In Primary Key

Type
DOUBLE

Precision
☐ 0

Scale
☐ 0

Allows Nulls
☒ No

Auto Increment
☒ No

Default Value
☒

Sequence Name (Only applies to target platforms that use sequences)
Agg_Ano_Fat_Faturamento_seq

- *Logical Name:* **Contador de Linhas**
- *Physical Name:* **ContadorLinha**
- *In Primary Key:* **Não**
- *Type:* **Integer**

Column Properties of Contador de Linhas ✕

Source for ETL Mapping
None Specified

Logical Name
Contador de Linhas

Physical Name
ContadorLinha

☐ In Primary Key

Type
INTEGER

Precision
☐ 0

Scale
☐ 0

Allows Nulls
☒ No

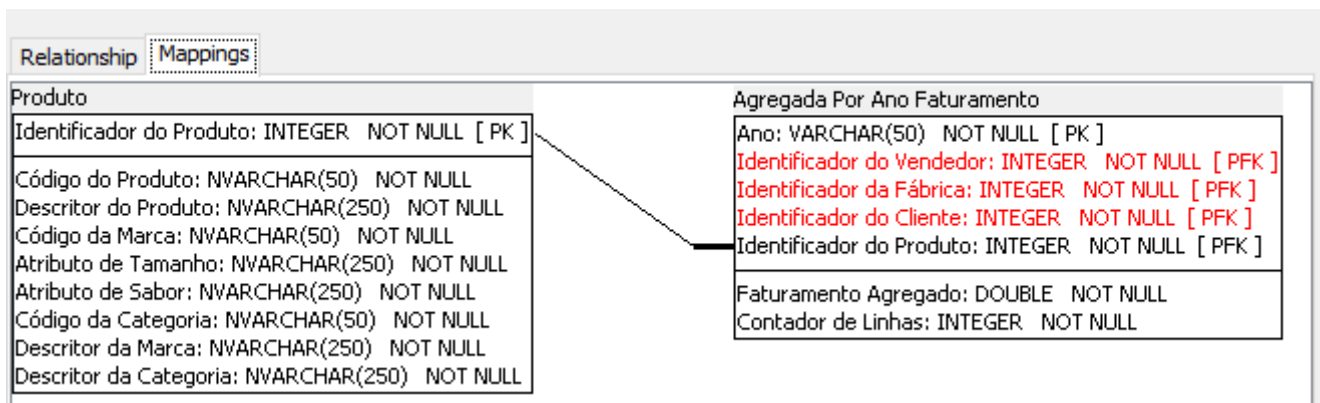
Auto Increment
☒ No

Default Value
☒

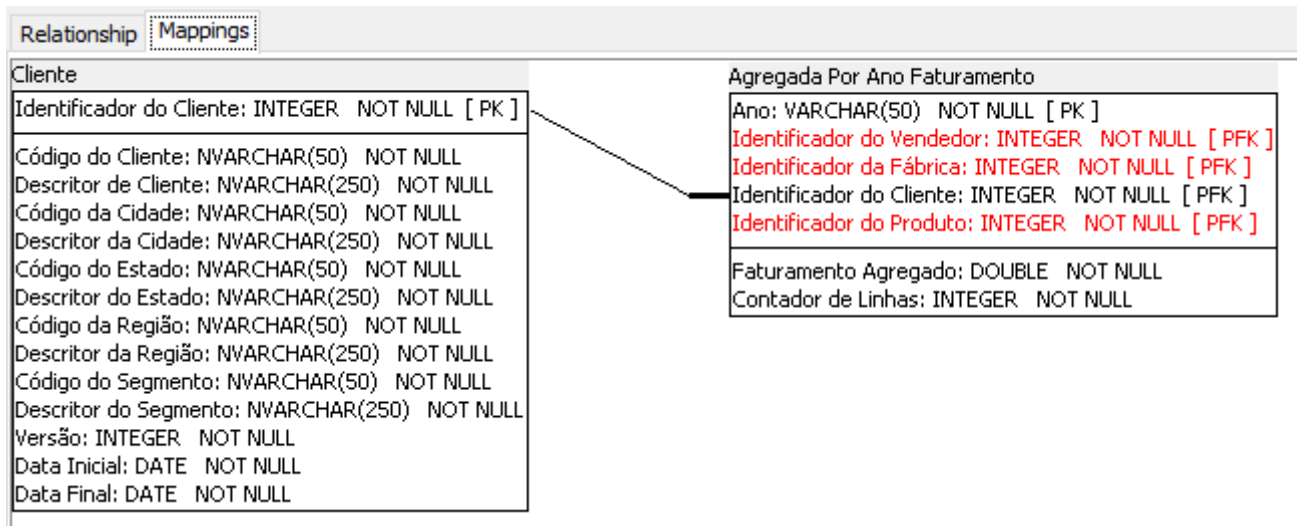
Sequence Name (Only applies to target platforms that use sequences)
Agg_Ano_Fat_ContadorLinha_seq

22) Com a tabela criada faça as ligações:

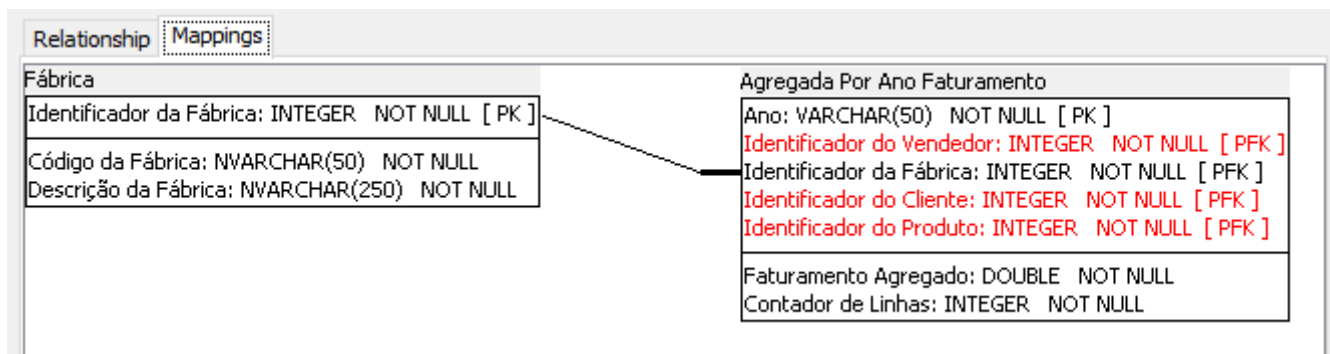
- **Identificador de Produto (Tabela Dim_Produto) - Identificador de Produto (Tabela Agg_Ano_Fat)**



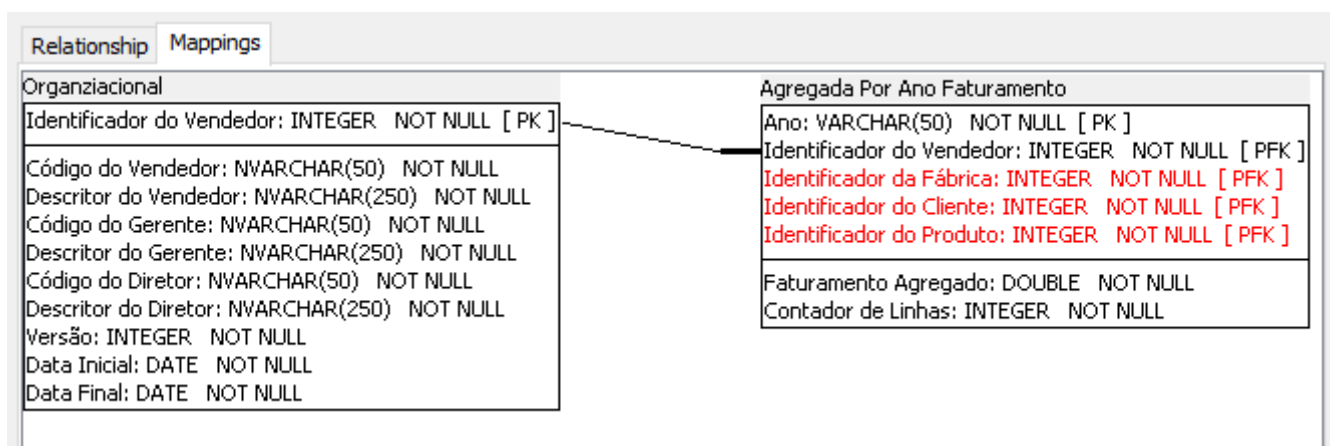
- **Identificador do Cliente (Tabela Dim_Cliente) - **Identificador do Cliente(Tabela Agg_Ano_Fat)**



- **Identificador da Fábrica (Tabela Dim_Fabrica) - **Identificador da Fabrica (Tabela Agg_Ano_Fat)**



- **Identificador do Vendedor (Tabela Dim_organizacional) - **Identificador do Vendedor (Tabela Agg_Ano_Fat)**



23) Gerando os comandos SQL, filtre apenas os comandos que estão relacionados com esta nova tabela. Se todos os nomes de suas tabelas e índices forem corretos, execute:

```
CREATE TABLE Agg_Ano_Fat (
    Cod_Ano VARCHAR(50) NOT NULL,
```

```
ID_Vendedor INT NOT NULL,  
ID_Fabrica INT NOT NULL,  
ID_Cliente INT NOT NULL,  
  
ID_Produto INT NOT NULL,  
Faturamento DOUBLE PRECISION NOT NULL,  
ContadorLinha INT NOT NULL,  
PRIMARY KEY (Cod_Ano, ID_Vendedor, ID_Fabrica, ID_Cliente, ID_Produto)  
);
```

```
ALTER TABLE Agg_Ano_Fat ADD CONSTRAINT dim_organizacional_agg_ano_fat_fk  
FOREIGN KEY (ID_Vendedor)  
REFERENCES Dim_Organizacional (ID_Vendedor)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION;
```

```
ALTER TABLE Agg_Ano_Fat ADD CONSTRAINT dim_produto_agg_ano_fat_fk  
FOREIGN KEY (ID_Produto)  
REFERENCES Dim_Produto (ID_Produto)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION;
```

```
ALTER TABLE Agg_Ano_Fat ADD CONSTRAINT dim_cliente_agg_ano_fat_fk  
FOREIGN KEY (ID_Cliente)  
REFERENCES Dim_Cliente (ID_Cliente)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION;
```

```
ALTER TABLE Agg_Ano_Fat ADD CONSTRAINT dim_fabrica_agg_ano_fat_fk  
FOREIGN KEY (ID_Fabrica)  
REFERENCES Dim_Fabrica (ID_Fabrica)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION;
```

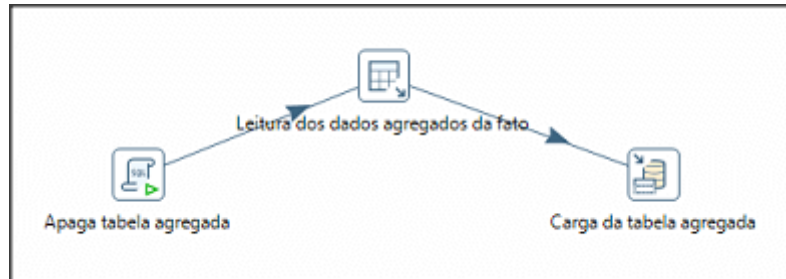
[COPIAR CÓDIGO](#)

24) Com a tabela criada, chegou a hora de criar o processo de ETL para incluir dados na tabela agregada. Para isso, no **Pentaho Data Integration**, crie uma transformação chamada **agg_fat_ano**.

25) Na área de Canvas, adicione os seguintes componentes

- **Execute SQL script**, que terá o nome **Apaga tabela agregada**
- **Table input**, que terá o nome **Leitura dos dados agregados da fato**
- **Insert / update**, que terá o nome **Carga da tabela agregada**

Que resultará no seguinte esquema:



26) Em **Apaga tabela agregada**, preencha as seguintes propriedades:

Execute SQL script

Step name: **Apaga tabela agregada**

Connection: **Data Mart Presidência** [Edit... New... Wizard...]

SQL script to execute. (statements separated by ;) Question marks will be replaced by arguments.

DELETE FROM AGG_ANO_FAT

Line 1 Column 0

☐ Execute for each row?

☐ Execute as a single statement

☐ Variable substitution

☐ Bind parameters?

☐ Quote Strings?

Parameters :

#	Field name to be used as argument
1	

Field to contain insert stats:

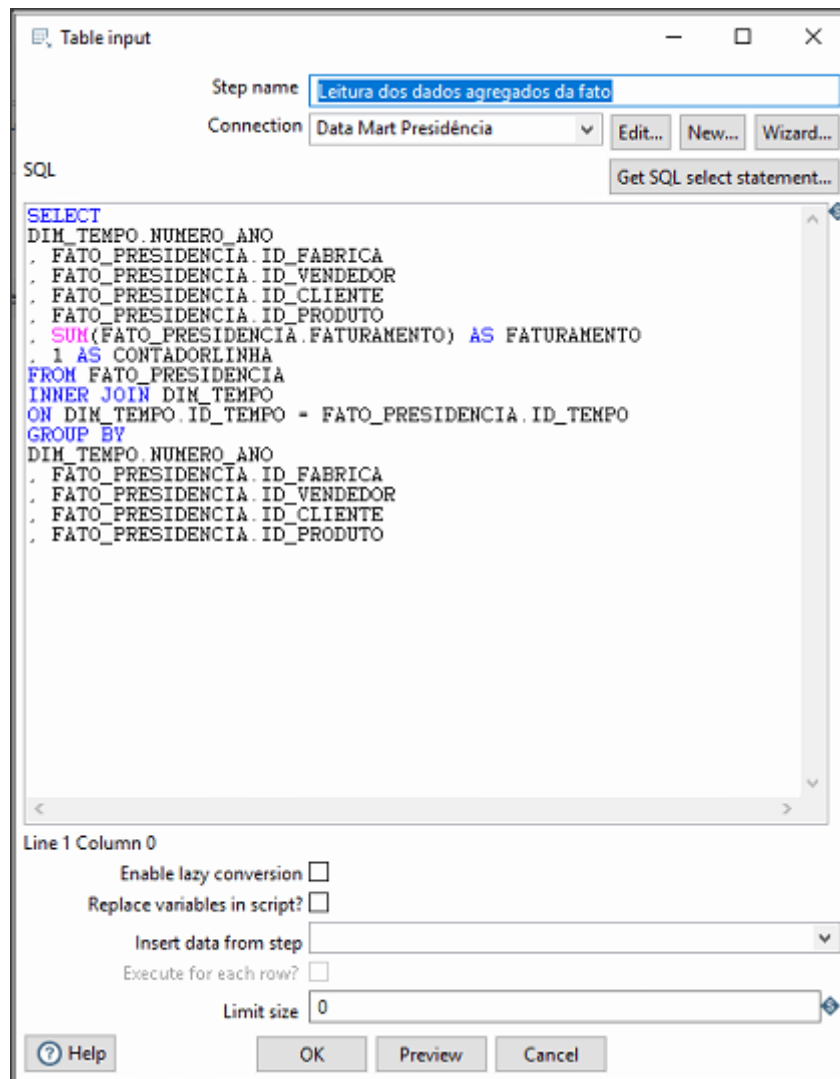
Field to contain Update stats:

Field to contain Delete stats:

Field to contain Read stats:

[Help] [OK] [Cancel] [Get fields]

27) Em **Leitura dos dados agregados da fato**, preencha as seguintes propriedades:



A consulta a ser efetuada é descrita abaixo:

```
SELECT
DIM_TEMPO.NUMERO_ANO
, FATO PRESIDENCIA.ID_FABRICA
, FATO PRESIDENCIA.ID_VENDEDOR
, FATO PRESIDENCIA.ID_CLIENTE
, FATO PRESIDENCIA.ID_PRODUTO
, SUM(FATO PRESIDENCIA.FATURAMENTO) AS FATURAMENTO
, 1 AS CONTADORLINHA
FROM FATO PRESIDENCIA
INNER JOIN DIM_TEMPO
ON DIM_TEMPO.ID_TEMPO = FATO PRESIDENCIA.ID_TEMPO
GROUP BY
DIM_TEMPO.NUMERO_ANO
, FATO PRESIDENCIA.ID_FABRICA
, FATO PRESIDENCIA.ID_VENDEDOR
```

```
, FATO_PRESIDENCIA.ID_CLIENTE
, FATO_PRESIDENCIA.ID_PRODUTO
```

[COPIAR CÓDIGO](#)

28) Em **Carga da tabela agregada**, preencha as seguintes propriedades:

Step name:

Connection:

Target schema:

Target table:

Commit size:

Don't perform any updates: ☐

The key(s) to look up the value(s):

#	Table field	Comparator	Stream field1	Stream field2
1	ID_Cliente	=	ID_CLIENTE	
2	ID_Fabrica	=	ID_FABRICA	
3	ID_Produto	=	ID_PRODUTO	
4	ID_Vendedor	=	ID_VENDEDOR	
5	Cod_Ano	=	NUMERO_ANO	

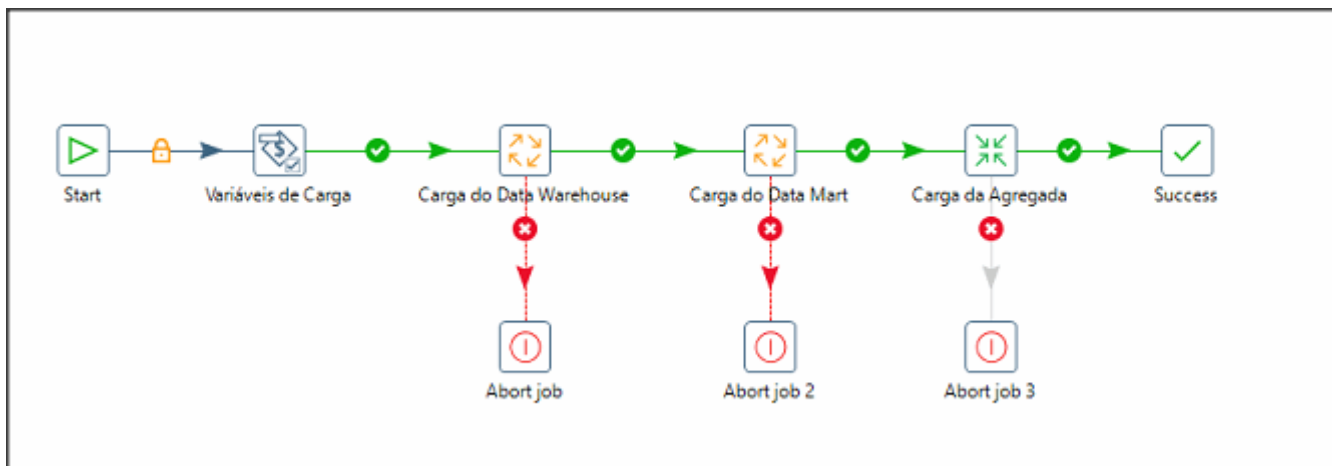
Update fields:

#	Table field	Stream field	Update
1	ContadorLinha	CONTADORLINHA	Y
2	Faturamento	FATURAMENTO	Y
3	ID_Cliente	ID_CLIENTE	N
4	ID_Fabrica	ID_FABRICA	N
5	ID_Produto	ID_PRODUTO	N
6	ID_Vendedor	ID_VENDEDOR	N
7	Cod_Ano	NUMERO_ANO	N

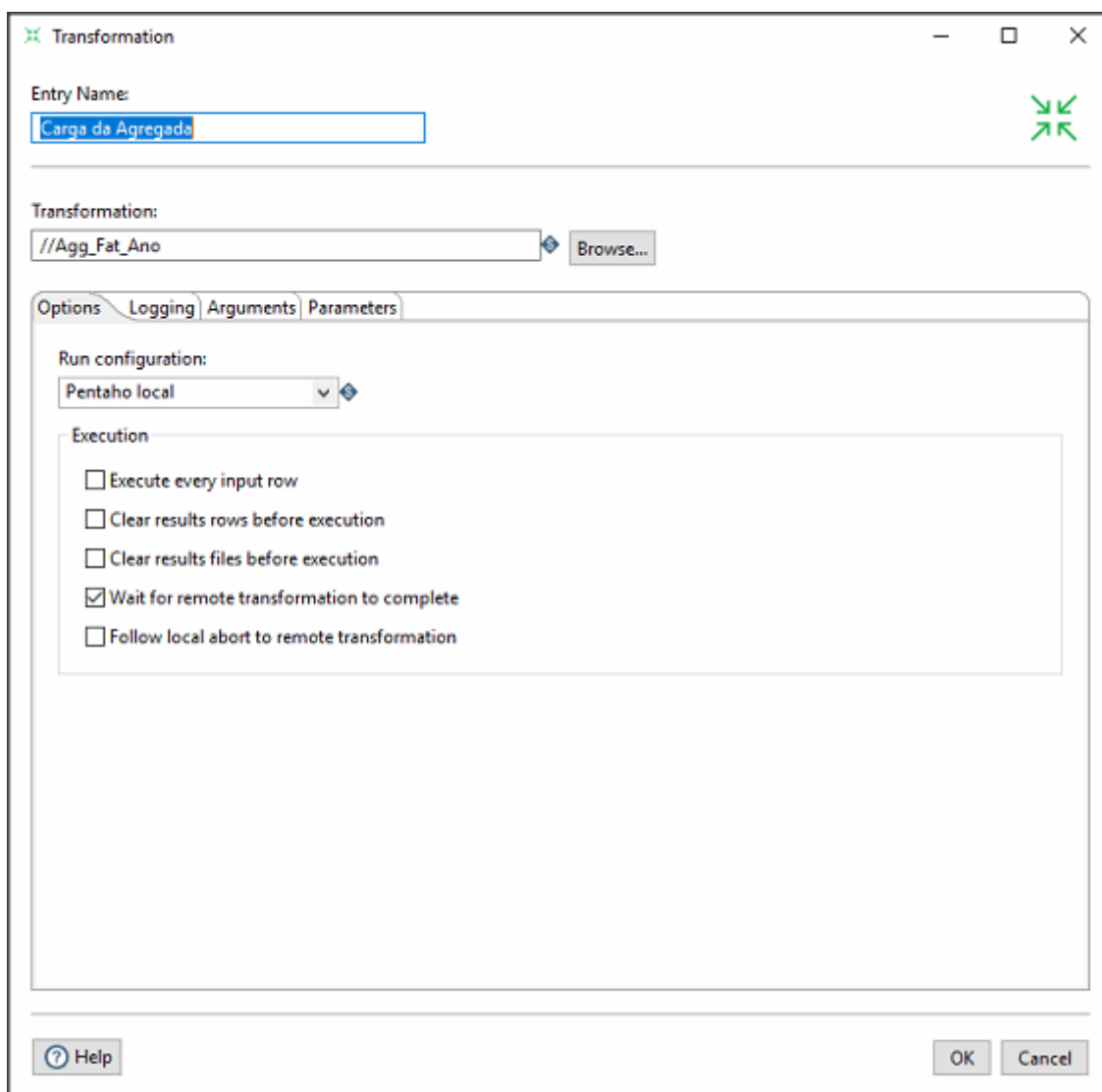
29) Salve e execute a transformação.

30) Abra o *job* **JobCargaDwDm**.

31) Inclua, no final do fluxo do *job*, a carga do *Data Mart*, como mostrado abaixo:



32) Em **Carga da Agregada**, preencha as seguintes propriedades:



33) Salve o *job*.

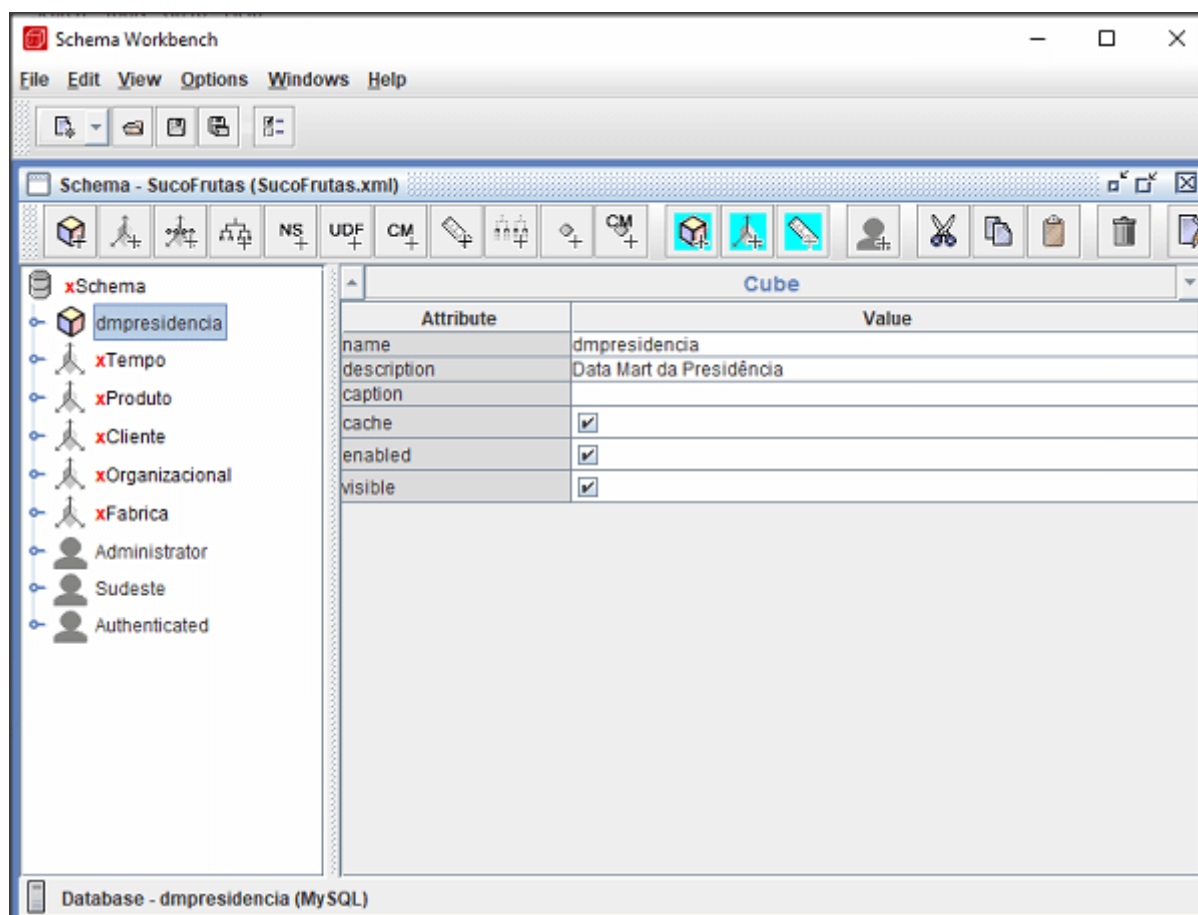
34) Acesse o **HeidiSQL** e veja o conteúdo da tabela agregada, executando:

```
Select * from agg_ano_fat
```

COPIAR CÓDIGO

agg_ano_fat (7x5,367)						
Cod_Ano	ID_Vendedor	ID_Fabrica	ID_Cliente	ID_Produto	Faturamento	ContadorLinha
2013	1	2	16	1	20,261.255249023438	1
2013	1	2	16	2	16,337.998901367188	1
2013	1	2	16	3	15,079.169860839844	1
2013	1	2	16	4	23,185.965454101562	1
2013	1	2	16	5	23,165.207763671875	1
2013	1	2	16	6	13,218.384216308594	1
2013	1	2	16	7	20,357.919555664062	1
2013	1	2	16	8	33,018.79821777344	1
2013	1	2	16	9	13,723.132568359375	1
2013	1	2	16	10	17,196.186096191406	1
2013	1	2	16	11	15,779.344543457031	1
2013	1	2	16	12	14,799.834655761719	1
2013	1	2	16	13	11,455.542602539062	1

35) Agora, configure a tabela agregada no **Pentaho Workbench**:



36) Devido a um problema na versão do **Pentaho Workbench** abra, usando um editor de texto, o arquivo **SucosFrutas.xml**. localizado em **C:\treinamento\Eschema**.

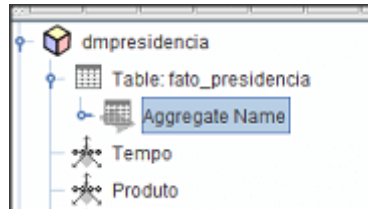
37) Procure pela *tag <Table> .

38) Edite o arquivo de maneira que você tenha:

```
<Table name="fato_presidencia">
  <AggName name="agg_ano_fat" ignorecase="true">
</Table>
```

COPIAR CÓDIGO

39) Salve o arquivo e volte ao **Pentaho Workbench**. Você terá:



Aggregate Name	
Attribute	Value
name	agg_ano_fat
ignorecase	<input checked="" type="checkbox"/>
approxRowCount	

40) Continue a configuração da agregação. Abaixo de *Aggregate Name*, inclua uma **Aggregate Fact Count**:

Aggregate Fact Count	
Attribute	Value
column	ContadorLinha

41) Crie, também abaixo da *Aggregate Name*, uma **Aggregate Measure**:

Aggregate Measure	
Attribute	Value
column	Faturamento
name	[Measures].[Faturamento]
rollupType	

42) Também crie um **Aggregate Level**:

Aggregate Level	
Attribute	Value
column	Cod_Ano
name	[Tempo.Calendario].[Ano]
collapsed	<input checked="" type="checkbox"/>
captionColumn	
ordinalColumn	
nameColumn	

43) E crie quatro **Aggregate Foreign Key**:

Aggregate Foreign Key	
Attribute	Value
factColumn	ID_Fabrica
aggColumn	ID_Fabrica

Aggregate Foreign Key	
Attribute	Value
factColumn	ID_Vendedor
aggColumn	ID_Vendedor

Aggregate Foreign Key	
Attribute	Value
factColumn	ID_Cliente
aggColumn	ID_Cliente

Aggregate Foreign Key	
Attribute	Value
factColumn	ID_Produto
aggColumn	ID_Produto

Observação: Aqui é importante que a forma com que você registra o nome dos campos seja igual aos campos declarados na definição desta mesma chave no OLAP. Por exemplo:

ID_Cliente (com I, D e C maiúsculos), deve ser igual aqui:

Aggregate Foreign Key	
Attribute	Value
factColumn	ID_Cliente
aggColumn	ID_Cliente

Na **Dimension Usage** da base OLAP:

Dimension Usage for 'dmpresidencia' Cube	
Attribute	Value
name	Cliente
foreignKey	ID_Cliente
source	Cliente
level	
usagePrefix	
caption	
visible	<input checked="" type="checkbox"/>

E na base **MySQL**:

Columns: ➕ Add ➖ Remove ⬆ Up ⬇ Down							
#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default
1	ID_Cliente	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default
2	Cod_Cliente	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
3	Desc_Cliente	VARCHAR	250	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
4	Cod_Cidade	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL

44) Salve e publique o XML **SucosFrutas.xml**.

45) Veja se os *logs* do OLAP Pentaho estão ativados. Para isso, verifique os valores no arquivo **mondrian.properties** (em **C:\treinamento\pentaho-server\pentaho-solutions\system\mondrian**):

```
# Allow the use of aggregates
mondrian.rolap.aggregates.Use=true
mondrian.rolap.aggregates.Read=true
```

COPIAR CÓDIGO

46) E no arquivo **log4j.xml**, em **C:\treinamento\pentaho-server\tomcat\webapps\pentaho\WEB-INF\classes**. Descomente o conteúdo das seguintes seções:

```
<!-- ===== -->
<!-- Special Log File specifically for Mondrian -->
<!-- ===== -->

<!-- ===== -->
<!-- Special Log File specifically for Mondrian MDX Statements -->
<!-- ===== -->

<!-- ===== -->
<!-- Special Log File specifically for Mondrian SQL Statements -->
<!-- ===== -->
```

COPIAR CÓDIGO

47) Pare o serviço Tomcat do Pentaho Server, vá em **c:\treinamento\pentaho-server\tomcat\logs** e apague os arquivos **mondrian_sql.log** e **mondrian_mdx.log**. Em

seguida, inicialize o serviço novamente.

48) No **Pentaho User Console**, crie e execute a seguinte análise pelo **Saiku**:

Ano	Faturamento
2013	54.587.992,47
2014	77.213.072,90
2015	105.179.687,87

49) Verifique no **mondrian_sql.log** que a consulta optou pela tabela de fato original:

```
I
1 [select `dim_tempo`.`Numero_Ano` as `c0`, sum(`fato_presidencia`.`Faturamento`) as
```

50) Troque a consulta por esta:

Medidas

Faturamento

Colunas

Fábrica.Fábrica

Fábrica

Linhas

Ano

Calendário

51) Já o **Pentaho OLAP** optou pela agregada:

```
executing sql [select count(distinct `Cod_Fabrica`) from `Dim_Fabrica`]
rica`.`Cod_Fabrica` as `c0`, sum(`agg_ano_fat`.`Faturamento`) as `m0` from `Dim_Fabrica`
```