Rapport du projet en BDD évoluées

HISTORIQUES DES NAISSANCES EN 2015 EN FRANCE

Maëlle Brassier, Florian Lardy et Ophélie Marinier
UNIVERSITE DE NANTES | [ADRESSE DE LA SOCIETE]

Table des matières

I. Présentation du projet 2
A. Notre sujet 2
B. Présentation de nos 2 datasets 2
II. La conception de notre entrepôt de données 3
A. Présentation de notre entrepôt 3
B. Intégration et transformation de nos données 5
III. Rajout de tuples à notre entrepôt 8
IV. Nos requêtes OLAP 8
V. Problèmes rencontrés 11
VI. Conclusion 12

I. Présentation du projet

A. Notre sujet

Au cours de notre première année de master, un projet en bases de données évoluées nous a été donné. L'objectif de celui-ci est de réaliser un entrepôt de données à partir de données réelles.

Pour trouver des datasets intéressants, nous avons recherché sur le site OpenDataSoft.com afin de pouvoir travailler avec de réelles données. Nous avons fait attention que nos données soient historisées pour nos analyses. Notre premier dataset porte sur l'historique des naissances en 2015 en France (https://public.opendatasoft.com/explore/dataset/les-naissances-en-2015/)
et notre deuxième dataset sur les départements français en 2015 également (https://public.opendatasoft.com/explore/dataset/contours-simplifies-des-departements-francais-2015/).
Les naissances comportent 793819 enregistrements et les départements comportent 101 enregistrements.

Concernant les outils utilisés nous avons choisi de faire une base de données relationnelle avec MySQL et d'ensuite utiliser Talend pour l'intégration de nos données.

B. Présentation de nos 2 datasets

Pour commencer, nous avons trouvé un dataset qui répertorie l'historique des naissances en 2015 en France. Celui-ci contient un grand nombre d'attributs :

- Sexe de l'enfant : masculin ou féminin
- Année de naissance de l'enfant
- Mois de naissance de l'enfant
- Département de naissance de l'enfant
- Jour de reconnaissance du père
- Mois de reconnaissance du père
- Année de reconnaissance du père
- Jour de reconnaissance de la mère
- Mois de reconnaissance de la mère
- Année de reconnaissance de la mère
- Jour de reconnaissance conjointe des parents
- Mois de reconnaissance conjointe des parents
- Année de reconnaissance conjointe des parents
- Âge de la mère dans l'année de naissance de l'enfant
- Age exacte de la mère à la naissance de l'enfant
- Indicateur du lieu de naissance de la mère (1 : née en France métropolitaine, 2 : née dans un DOM, 3 : née dans un COM, 4 : née à l'étranger).
- Situation professionnelle de la mère : salariée, inconnue, retraitée ou inactive, Nsalariée.
- Indicateur de nationalité de la mère (1 : française, 2 : étrangère).
- Département de domicile de la mère
- Tranche de commune du lieu de domicile de la mère
- Age du père dans l'année de naissance de l'enfant
- Age exacte du père à la naissance de l'enfant
- Indicateur du lieu de naissance du père (1 : né en France métropolitaine, 2 : né dans un DOM, 3 : né dans un COM, 4 : né à l'étranger)
- Situation professionnelle du père : salarié, inconnu, retraité ou inactif, Nsalarié.
- Indicateur de nationalité du père (1 : français, 2 : étranger).
- Année de mariage des parents

- Comparaison des dates anniversaires de mariage des parents et de naissance de l'enfant (né hors mariage, naissance survenue avant l'anniversaire de mariage, naissance survenue le même jour ou après l'anniversaire du mariage, Jugement déclaratif de naissance).
- Conditions de l'accouchement (jugement déclaratif de naissance, dans un établissement spécialisé, autre).
- Nombre d'enfants issus de l'accouchement
- Durée écoulée depuis l'événement précédent (soit enfant né hors mariage, premier né ou jugement déclaratif de naissance, ou soit le nombre d'années écoulées depuis le mariage ou la naissance précédente).
- Origine du nom de l'enfant (Origine du nom non connue (Jugement déclaratif de naissance), Père, Mère, Père-mère, Mère-Père ou Autre).

Ce dataset va nous permettre de faire des analyses sur l'âge des parents, leurs situations professionnelles, leurs localisations dans la France, leurs origines, l'influence du mariage sur les naissances, etc.
Pour complémenter nos informations, nous avons pu lier ce premier dataset avec un deuxième qui répertorie les départements français en 2015. En effet, ils contiennent tous les deux un attribut département donc le lien s'est fait facilement. Voici donc les quelques attributs qui complètent notre entrepôt :

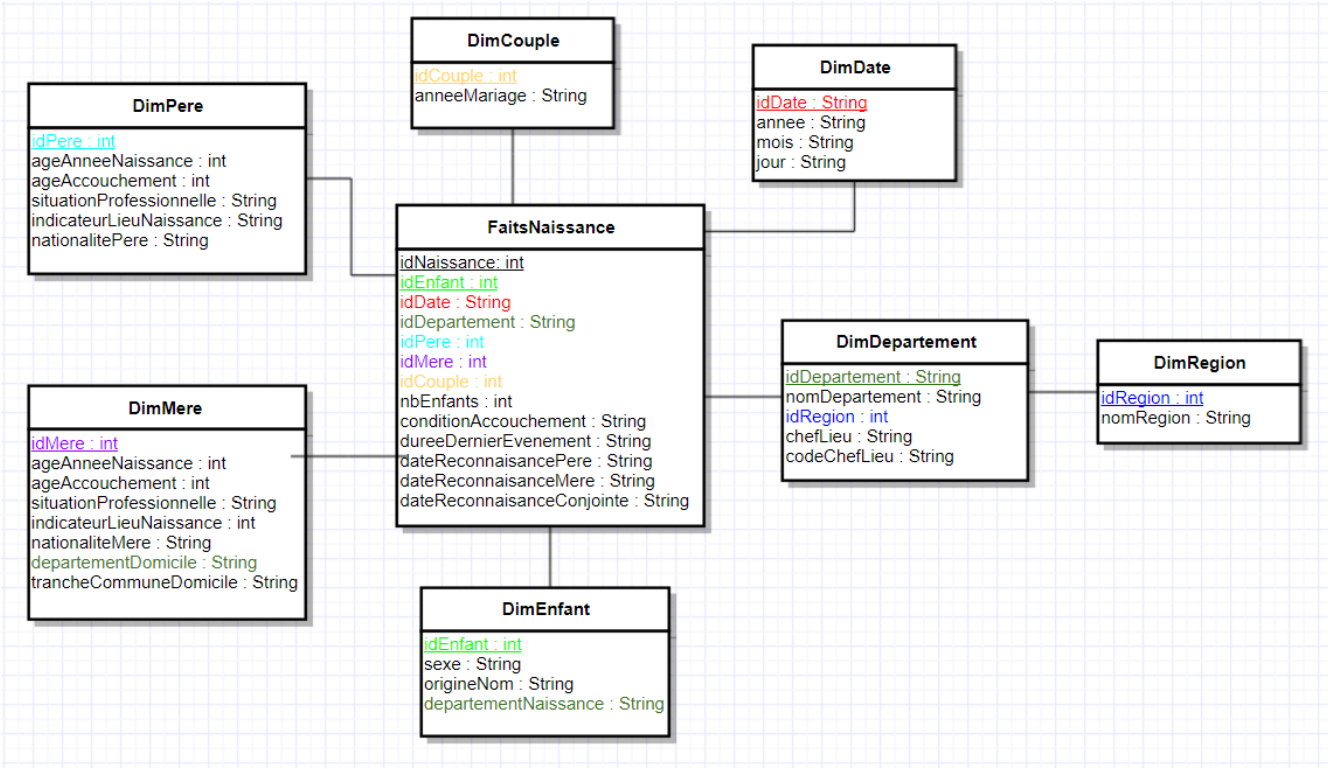
- Nom du département
- Nom du chef-lieu
- Code de la région
- Nom de la région
- Code du département
- Code du chef-lieu

II. La conception de notre entrepôt de données

Nous allons vous montrer notre démarche de conception de notre base de données.

A. Présentation de notre entrepôt

Voici ci-dessous le schéma de notre entrepôt de données.



Nous avons fait une conception du schéma en étoile de notre entrepôt de données sauf pour les tables DimDepartement et DimRegion qui sont en flocon.
Notre choix de grain est la naissance d'un enfant en France en 2015.

En ce qui concerne nos tables de dimensions, nous en avons sept :

- La table DimDate qui contient les jours, les mois et les années.
- La table DimPere avec les informations liées au père telles que son âge pendant l'année de naissance de l'enfant, son âge exact au moment de l'accouchement, sa situation professionnelle, son lieu de naissance et sa nationalité. Sa clé primaire est
- La table DimMere avec les informations liées à la mère, c'est-à-dire les même attributs que la table DimPere mais avec en plus son département de domicile et la tranche de la commune de son domicile.
- La table DimCouple qui relie un homme et une femme par un identifiant de couple et une année de mariage.
- La table DimEnfant avec son sexe, l'origine de son nom et son département de naissance.
- La table DimDepartement qui contient le numéro et le nom du département, le numéro et le nom du chef-lieu, et le numéro de la région du département
- La table DimRegion fait correspondre le numéro de la région avec son nom.

Notre table des faits appelée FaitsNaissance contient les identifiants de toutes les tables de dimensions et quelques attributs supplémentaires comme le nombre d'enfants par accouchement, les conditions de la naissance, la durée écoulée depuis l'événement précédent, la date de reconnaissance de l'enfant du père, de la mère et de la conjointe.
En ce qui concerne nos contraintes, nous nous sommes dit que les données dans nos datasets étaient déjà triées et nettoyées et donc nous n'avions pas besoin d'en rajouter énormément.

Les clés primaires :

La clé primaire de la table DimDate est idDate qui est une chaine de caractère représentant la concaténation du jour, du mois et de l'année. En effet, nous avons du les concaténer pour vérifier que l'on ai pas des doublons de dates.
La clé primaire de la table DimMere est idMere.
La clé primaire de la table DimPere est idPere.
La clé primaire de la table DimMere est idMere.
La clé primaire de la table DimCouple est idCouple.
La clé primaire de la table DimEnfant est idEnfant.
La clé primaire de la table DimDepartement est idDepartement.
La clé primaire de la table DimRegion est idRegion.
La clé primaire de la table FaitsNaissance est idNaissance et idEnfant. En effet, comme une naissance peut être multiple (des jumeaux, triplés, etc) nous avons choisi une clé primaire double afin d'avoir la naissance d'un seul enfant par ligne de la table des faits.

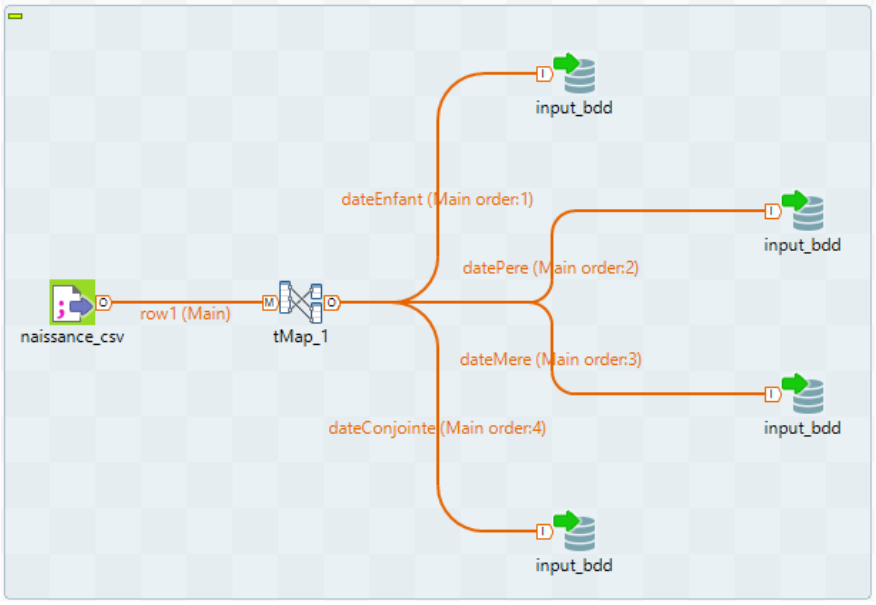
Les clés étrangères :
Celles-ci sont reliées par un code couleur (voir le schéma de l'entrepôt).

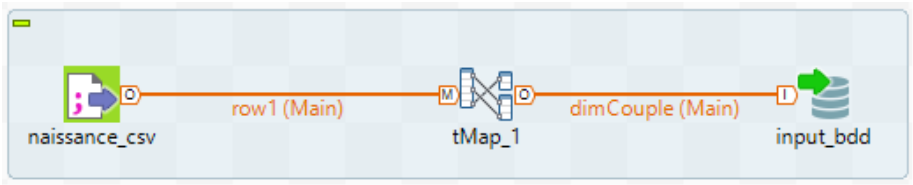
B. Intégration et transformation de nos données

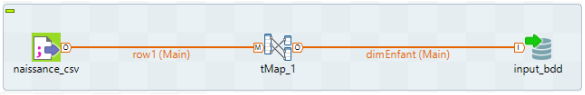
Comme nous l'avons dit dans la présentation de notre sujet, nous avons utilisé Talend pour l'intégration de nos données. Cependant nous avons rencontré quelques problèmes comme le fait qu'il n'y ait pas d'identifiant dans notre dataset de naissances. Les naissances multiples posent également problème. En effet, nous ne pouvions pas avoir deux attributs en auto incrémentation sur la clé primaire double de la table des faits. Nous avons donc créé un script de transformation de nos données dans le fichier CSV des naissances pour supprimer les naissances multiples. Ce script se nomme *ConversionCSV.java.*

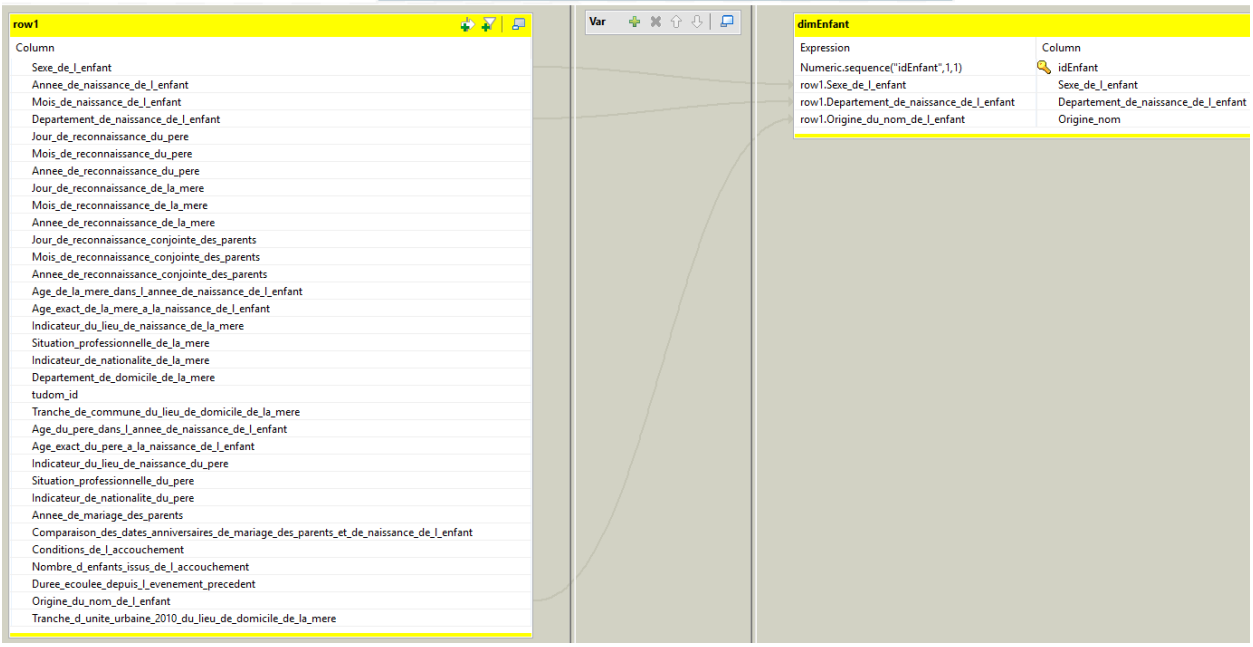
Intégration des tables de dimensions :

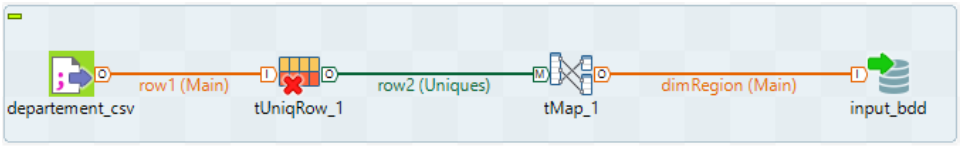
Pour la dimension Date il nous fallait créer une date pour la naissance de l'enfant et pour les reconnaissances du père, de la mère et conjointe des parents. Nous avons utilisé la concaténation des attributs jour, mois et année de ces dates respectives.

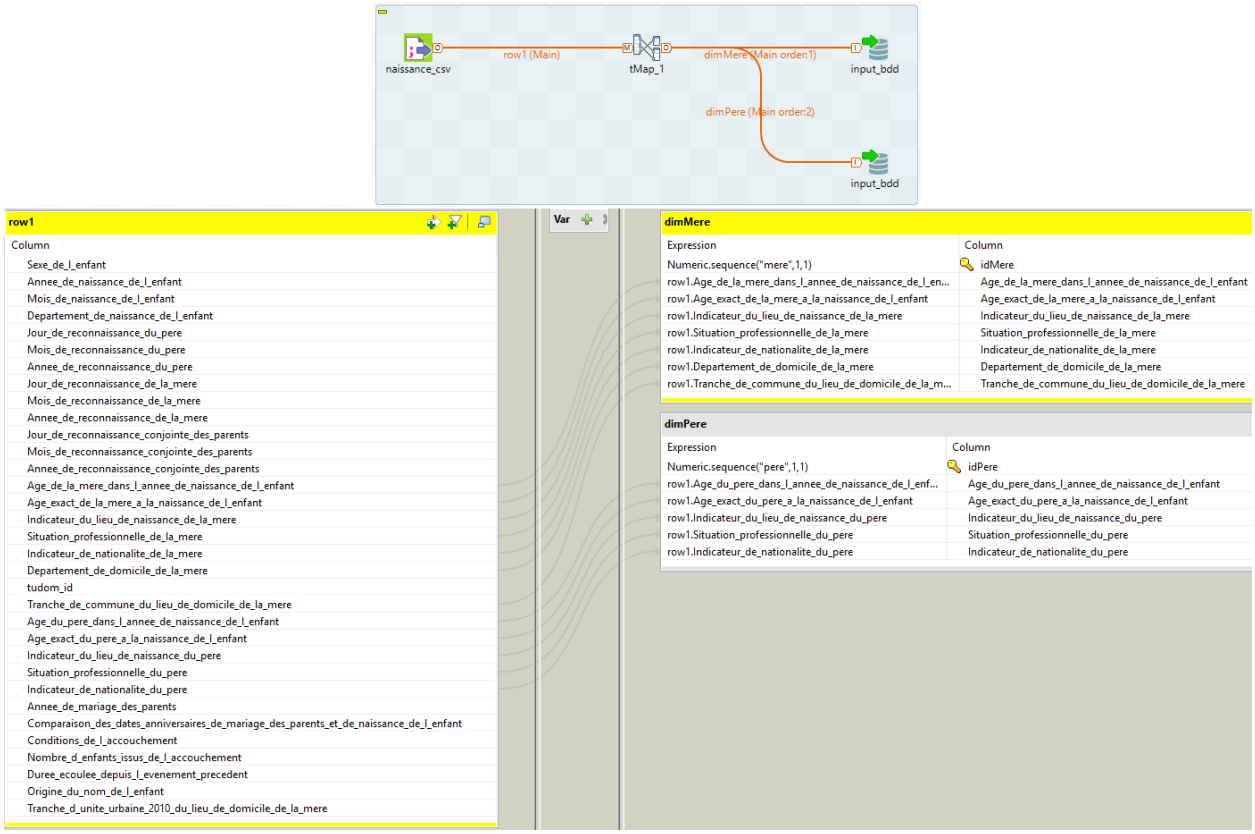



naissance_csv
row1 (Main)
tMap_1
dimCouple (Main)
input_bdd

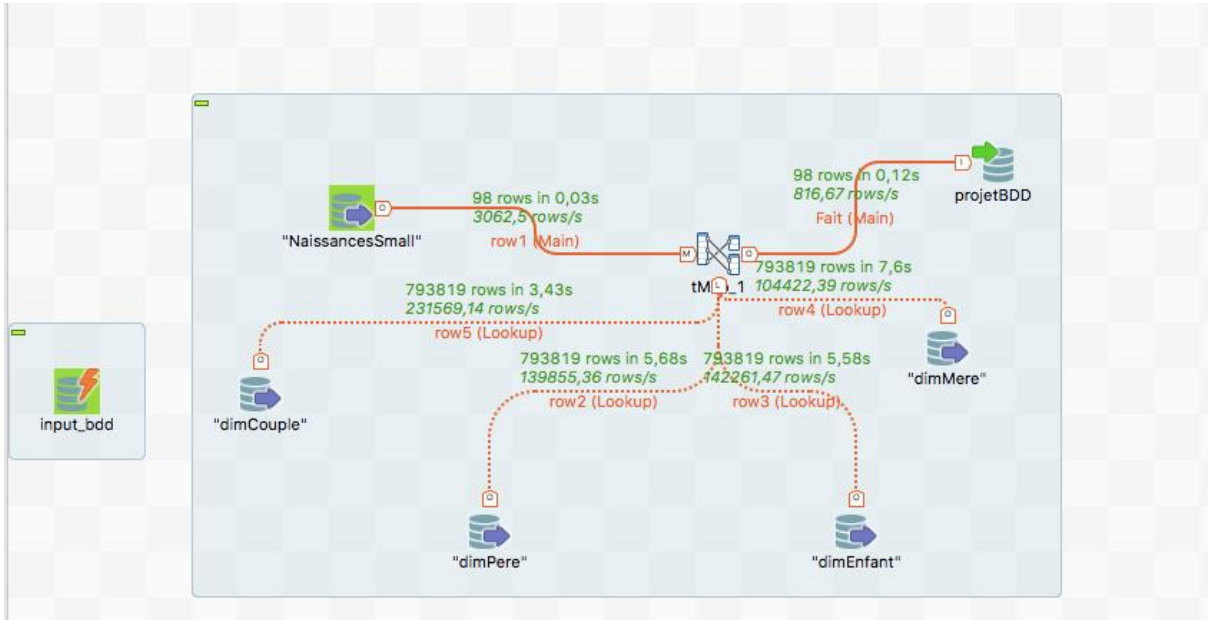

naissance_csv
row1 (Main)
tMap_1
dimEnfant (Main)
input_bdd


row1
Column
Sexe_de_l_enfant
Annee_de_naissance_de_l_enfant
Mois_de_naissance_de_l_enfant
Departement_de_naissance_de_l_enfant
Jour_de_reconnaissance_du_pere
Mois_de_reconnaissance_du_pere
Annee_de_reconnaissance_du_pere
Jour_de_reconnaissance_de_la_mere
Mois_de_reconnaissance_de_la_mere
Annee_de_reconnaissance_de_la_mere
Jour_de_reconnaissance_conjointe_des_parents
Mois_de_reconnaissance_conjointe_des_parents
Annee_de_reconnaissance_conjointe_des_parents
Age_de_la_mere_dans_l_annee_de_naissance_de_l_enfant
Age_exact_de_la_mere_a_la_naissance_de_l_enfant
Indicateur_du_lieu_de_naissance_de_la_mere
Situation_professionnelle_de_la_mere
Indicateur_de_nationalite_de_la_mere
Departement_de_domicile_de_la_mere
tudom_id
Tranche_de_commune_du_lieu_de_domicile_de_la_mere
Age_du_pere_dans_l_annee_de_naissance_de_l_enfant
Age_exact_du_pere_a_la_naissance_de_l_enfant
Indicateur_du_lieu_de_naissance_du_pere
Situation_professionnelle_du_pere
Indicateur_de_nationalite_du_pere
Annee_de_mariage_des_parents
Comparaison_des_dates_anniversaires_de_mariage_des_parents_et_de_naissance_de_l_enfant
Conditions_de_l_accouchement
Nombre_d_enfants_issus_de_l_accouchement
Duree_ecoulee_depuis_l_evenement_precedent
Origine_du_nom_de_l_enfant
Tranche_d_unite_urbaine_2010_du_lieu_de_domicile_de_la_mere
Var
dimEnfant
Expression
Column
Numeric.sequence("idEnfant",1,1)
idEnfant
row1.Sexe_de_l_enfant
Sexe_de_l_enfant
row1.Departement_de_naissance_de_l_enfant
Departement_de_naissance_de_l_enfant
row1.Origine_du_nom_de_l_enfant
Origine_nom


departement_csv
row1 (Main)
tUniqRow_1
row2 (Uniques)
tMap_1
dimRegion (Main)
input_bdd



Intégration de la table des faits TableFaitsNaissance :



III. Rajout de tuples à notre entrepôt

Voici quelques tuples que nous avons ajouté à notre entrepôt de données.

INSERT INTO faitnaissance
(idNaissance, idEnfant, idDate, idDepartement, idPere, idMere, idCouple, nbEnfant, conditionAccouchement, dureeDernierEvenement, dateReconnaissancePere, dateReconnaissanceMere, dateReconnaissanceConjointe)

VALUES (780000, 780000, '09022015', '41', 780000, 780000, 780000, 2, 'dans un établissement spécialisé', '04', '12022015', '12022015', '15022015'),
(780000, 780001, '09022015', '41', 780000, 780000, 780000, 2, 'dans un établissement spécialisé', '04', '12022015', '12022015', '15022015'),
(780001, 780002, '10092015', '13', 780001, 780001, 780001, 1, 'autre', '09', '11092015', '11092015', '13022015'),
(780002, 780003, '25082015', '44', 780002, 780002, 780002, 3, 'dans un établissement spécialisé', '02', '26082015', '26082015', '27082015'),
(780002, 780004, '25082015', '44', 780002, 780002, 780002, 3, 'dans un établissement spécialisé', '02', '26082015', '26082015', '27082015'),
(780002, 780005, '25082015', '44', 780002, 780002, 780002, 3, 'dans un établissement spécialisé', '02', '26082015', '26082015', '27082015'),
(780003, 780006, '02052015', '37', 780003, 780001, 780003, 1, 'jugement déclaratif de naissance', '01', '04052015', '04052015', '04052015'),
(780004, 780007, '24122015', '36', 780004, 780003, 780004, 2, 'autre', 'enfant né hors mariage, premier né ou jugement déclaratif de naissance', '29122015', '29122015', '01012015'),
(780004, 780008, '24122015', '36', 780004, 780003, 780004, 2, 'autre', 'enfant né hors mariage, premier né ou jugement déclaratif de naissance', '29122015', '29122015', '01012015'),
(780005, 780009, '13062015', '33', 780005, 780004, 780005, 1, 'dans un établissement spécialisé', '05', '13062015', '23062015', '23062015'),
(780006, 780010, '30102015', '95', 780006, 780005, 780006, 1, 'autre', '12', '01112015', '011122015', '05112015'),
(780007, 780011, '17032015', '14', 780007, 780006, 780007, 1, 'dans un établissement spécialisé', 'enfant né hors mariage, premier né ou jugement déclaratif de naissance', '19032015', '17032015', '18032015');

Nous avons voulu montrer qu'il était possible d'ajouter des naissances multiples de jumeaux ou même de triplés.

IV. Nos requêtes OLAP

-- Requête 1 : Nombre d'enfants par sexe en fonction du mois.

```
SELECT D.Mois, E.Sexe_de_l_enfant, COUNT(E.idEnfant)
        FROM faitNaissance N, dimEnfant E, dimDate D
        WHERE N.idEnfant = E.idEnfant AND D.idDate = N.idDate
        GROUP BY D.Mois, E.Sexe_de_l_enfant WITH ROLLUP ;
```

Cette requête renvoie le nombre d'enfants nés tel mois en fonction de leur sexe. Cette requête peut être intéressante pour définir les mois les plus propices aux naissances ou à l'accouplement. Les compagnies

(de publicité par exemple) pourront adapteur leur stratégie en fonction de ces mois ainsi que du sexe de l'enfant.

-- Requête 2 : Nombre d'enfants par sexe en fonction du département

```
SELECT P.idDepartement, E.Sexe_de_l_enfant, COUNT(E.idEnfant)
        FROM faitNaissance N, dimEnfant E, dimDate D, dimDepartement P
        WHERE N.idEnfant = E.idEnfant AND N.idDate = D.idDate AND P.idDepartement =
        N.idDepartement
        GROUP BY P.idDepartement, E.Sexe_de_l_enfant ;
```

-- Requête 2-bis : Rang des départements en fonction du nombre d'enfants

Si on considère que count_enfant est le nombre d'enfants par département trouvé par la requête suivante :

```
SELECT P.idDepartement, COUNT(E.idEnfant)
FROM faitNaissance N, dimEnfant E, dimDate D, dimDepartement P
WHERE N.idEnfant = E.idEnfant AND N.idDate = D.idDate AND P.idDepartement =
N.idDepartement
GROUP BY P.idDepartement;
```

On peut faire un rang des départements qui ont le plus d'enfants.

```
SELECT P.idDepartement, count_enfant,
        DENSE_RANK() OVER (ORDER BY count_enfant DESC)
        FROM faitNaissance N, dimEnfant E, dimDate D, dimDepartement P
        GROUP BY idDepartement;
```

-- Requête 3 : Grouping des naissances par mois et par nombre d'enfants issus de l'accouchement

```
SELECT D.Mois, N.nbEnfant, COUNT(N.nbEnfant) as count_enfant
        FROM dimEnfant E, faitNaissance N, dimDate D
        WHERE E.idEnfant = N.idEnfant AND N.idDate = D.idDate
        GROUP BY D.Mois, N.nbEnfant WITH CUBE
```

-- Requête 4 : L'influence de la situation professionnelle sur les conditions d'accouchement

```
SELECT D.Mois, E.Sexe_de_l_enfant, COUNT(E.idEnfant)
        FROM faitNaissance N, dimEnfant E, dimDate D
        WHERE N.idEnfant = E.idEnfant AND D.idDate = N.idDate
        GROUP BY D.Mois, E.Sexe_de_l_enfant WITH ROLLUP ;
```

-- Requête 5 : L'influence de la situation professionnelle sur le temps de reconnaissance de l'enfant

```
SELECT M.Situation_professionnelle_de_la_mere, P.Situation_professionnelle_du_pere,
N.conditionAccouchement, COUNT(E.idEnfant),
GROUPING_ID(M.Situation_professionnelle_de_la_mere,
M.Situation_professionnelle_de_la_mere, N.conditionAccouchement) as grp
FROM faitNaissance N, dimEnfant E, dimDate D, dimMere M, dimPere P
WHERE N.idEnfant = E.idEnfant AND M.idMere = N.idMere AND N.idDate = D.idDate AND
P.idPere = N.idPere
                GROUP BY M.Situation_professionnelle_de_la_mere,
P.Situation_professionnelle_du_pere, N.conditionAccouchement WITH ROLLUP;
```

-- Requête 6 : Comparer les âges des deux parents

```
SELECT idEnfant, ABS(Pr.Age_exact_du_pere_a_la_naissance_de_l_enfant-
M.Age_exact_de_la_mere_a_la_naissance_de_l_enfant) AS diff FROM dimPere Pr, dimMere M,
faitNaissance N WHERE Pr.idPere = N.idPere AND M.idMere = N.idPere;
```

--Requête 6-bis : Grouping des différences des âges des parents ainsi que le nombre de personnes associées

```
SELECT ABS(Pr.Age_exact_du_pere_a_la_naissance_de_l_enfant-
M.Age_exact_de_la_mere_a_la_naissance_de_l_enfant) AS differenceAge,
count(ABS(Pr.Age_exact_du_pere_a_la_naissance_de_l_enfant-
M.Age_exact_de_la_mere_a_la_naissance_de_l_enfant)) AS countDifferenceAge
FROM dimPere Pr, dimMere M, faitNaissance N
WHERE Pr.idPere = N.idPere AND M.idMere = N.idPere
GROUP BY differenceAge
ORDER BY countDifferenceAge desc;
```

-- Requête 7 : les naissances où au moins un des parents est retraité ou inactif

```
SELECT idMere, idPere
FROM dimMere, dimPere
WHERE dimMere.Situation_professionnelle_de_la_mere = "retraitée ou inactive" AND
dimPere.Situation_professionnelle_du_pere = "retraité ou inactif"
Group By Age_de_la_mere_dans_l_annee_de_naissance_de_l_enfant,
Age_du_pere_dans_l_annee_de_naissance_de_l_enfant;
```

-- Requête 8 : les 10 premiers enfants nés au mois de janvier

```
SELECT idEnfant, idDate
FROM faitNaissance
WHERE SUBSTR (idDate, 3, 2) = “01”
LIMIT 10;
```

-- Requête 9 : TOP 5 durée depuis le dernier événement

```
SELECT dureeDernierEvenement,COUNT(N.dureeDernierEvenement) AS count_duree
FROM faitNaissance N, dimEnfant E, dimDate D, dimDepartement P
WHERE N.idEnfant = E.idEnfant AND N.idDate = D.idDate AND P.idDepartement =
N.idDepartement
GROUP BY N.dureeDernierEvenement ORDER BY count_duree DESC LIMIT 5;
```

-- Requête 10 : Durée entre l’année de mariage et la naissance

```
SELECT CAST(SUBSTR(N.idDate,5,4) AS UNSIGNED) -
CAST(C.Annee_de_mariage_des_parents AS UNSIGNED) as difference_marriage_naissance,
count(CAST(SUBSTR(N.idDate,5,4) AS UNSIGNED) - CAST(C.Annee_de_mariage_des_parents AS
UNSIGNED)) as count_diff
FROM dimDate D, faitNaissance N, dimCouple C, dimEnfant E
WHERE N.idEnfant = E.idEnfant AND N.idDate = D.idDate AND N.idCouple = C.idCouple
AND C.Annee_de_mariage_des_parents NOT LIKE "0000"
GROUP BY difference_marriage_naissance;
```

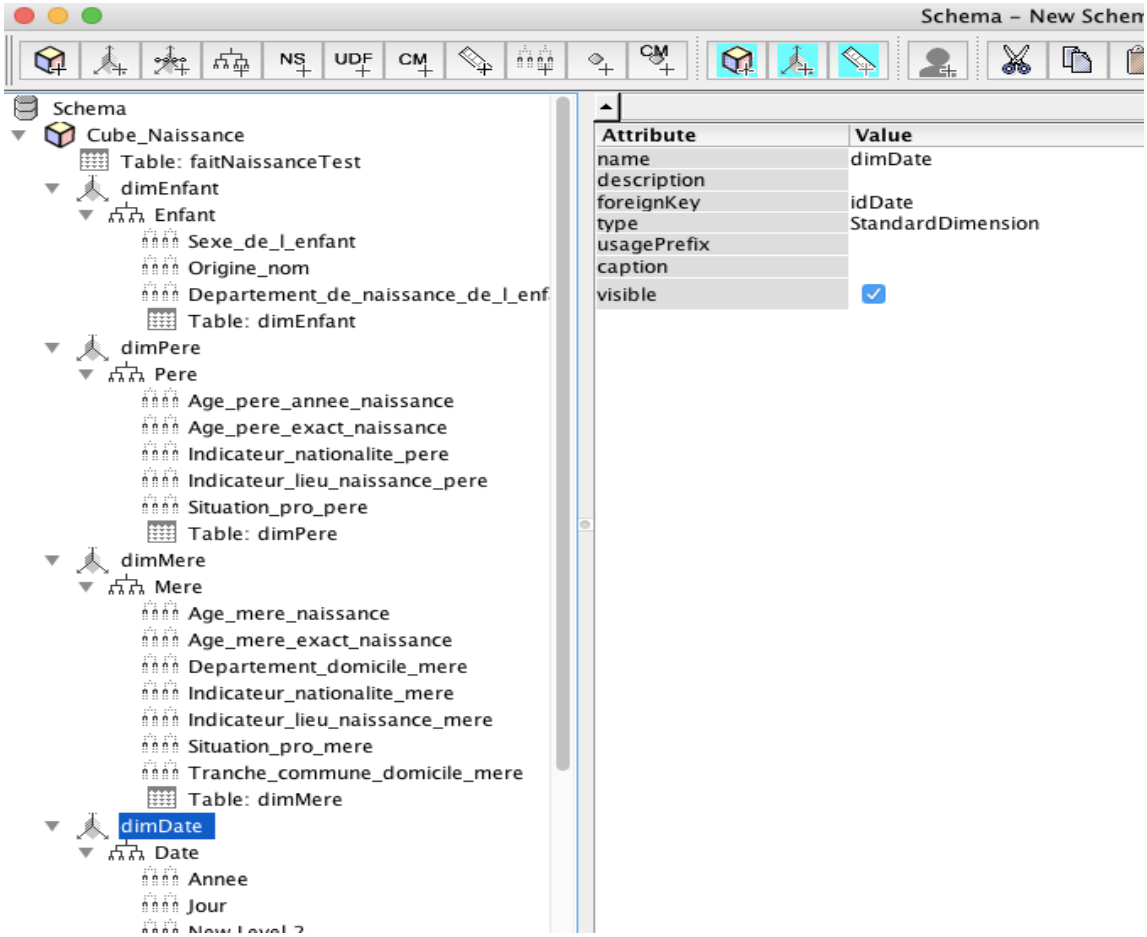
Exemple de résultats sous PhpMyAdmin (ici, pour la requête 4). Nous avons créé une table faitNaissancesTest avec 98 tuples car celle de base était bien trop lourde.

Mois	Sexe_de_l_enfant	COUNT(E.idEnfant)
01	féminin	4
01	NULL	4
02	féminin	7
02	masculin	2
02	NULL	9
03	féminin	1
03	masculin	7
03	NULL	8
04	féminin	1
04	masculin	3
04	NULL	4
05	féminin	5
05	masculin	5
05	NULL	10
06	féminin	8
06	masculin	3
06	NULL	11
07	féminin	4
07	masculin	2
07	NULL	6
08	féminin	2
08	masculin	7
08	NULL	9
09	féminin	11
09	masculin	3
09	NULL	14
10	féminin	4
10	masculin	6
10	NULL	10
11	féminin	8
11	masculin	2
11	NULL	10
12	féminin	1
12	masculin	2
12	NULL	3
NULL	NULL	98

V. Problèmes rencontrés

- Le manque d'identifiants dans le csv des naissances. Nous avons de ce fait du créer manuellement (via Talend) des identifiants pour la plupart des tables (idNaissances, iDPere, idCouple..)
- Suppression des naissances multiples dans notre fichier .csv.
- MySQL ne prend pas en charge les GROUP BY CUBE, RANK, TOP, c'est-à-dire la plupart des requêtes que nous devions faire.

A la suite de notre intégration Talend, nous avions décidé d'utiliser la technologie Pentaho qui nous aurait ainsi permis de visualiser notre cube et pouvoir analyser les données. Pour commencer, nous avons utilisé Schema Workbench afin de créer ce cube.



Ce cube nous a ressorti un schema XML (présent sur GitHub).

Il suffisait ensuite d'utiliser Pentaho avec un plugin tel que Pivot4j, cependant, ce dernier ne renvoyait qu'une simple page blanche sans aucune fonctionnalité. Pourtant, toutes nos configurations étaient faites. Certains utilisateurs ont assisté à la même erreur dernièrement (il y a moins de 4 jours), donc nous ne savons pas si le problème vient de nous ou bien de Pentaho directement.

Nous nous sommes donc rapatriés sur MySql et nous avons créé quelques requêtes en utilisant le plus possible les fonctions disponibles. Pour les fonctions intégrées uniquement sur oracle (telle que RANK), nous avons simulé des requêtes en écrivant ce qui nous semblait juste.

VI. Conclusion

Ce projet nous a permis de constater tout d'abord que le pré-traitement des données est une étape importante. En effet, toutes les données réelles (et open-source) ne sont pas forcément bien conçues. De plus, nous aurions également pu nous interroger d'avantage sur le choix des technologies. Si nous étions conscients que MySql ne prend pas en charge les requêtes OLAP, nous aurions probablement opté pour Oracle comme système de gestion de base de données.

Néanmoins, ce projet nous a appris de nouvelles choses telles que l'utilisation de Talend pour l'intégration des données, qui peut être un avantage conséquent lors d'un projet informatique. Nous avons pu aussi mettre en pratique nos connaissances concernant la conception des bases de données

dimensionnelles ainsi que le schéma en étoile/ flocon. Et de ce fait, nous donner une meilleure idée de ce qu'est et comment fonctionne un entrepôt de données.