Rapport NF26 TD Alimentation avec la Plateforme Pentaho

Tianyang CAI Longen ZHAO

I Introduction

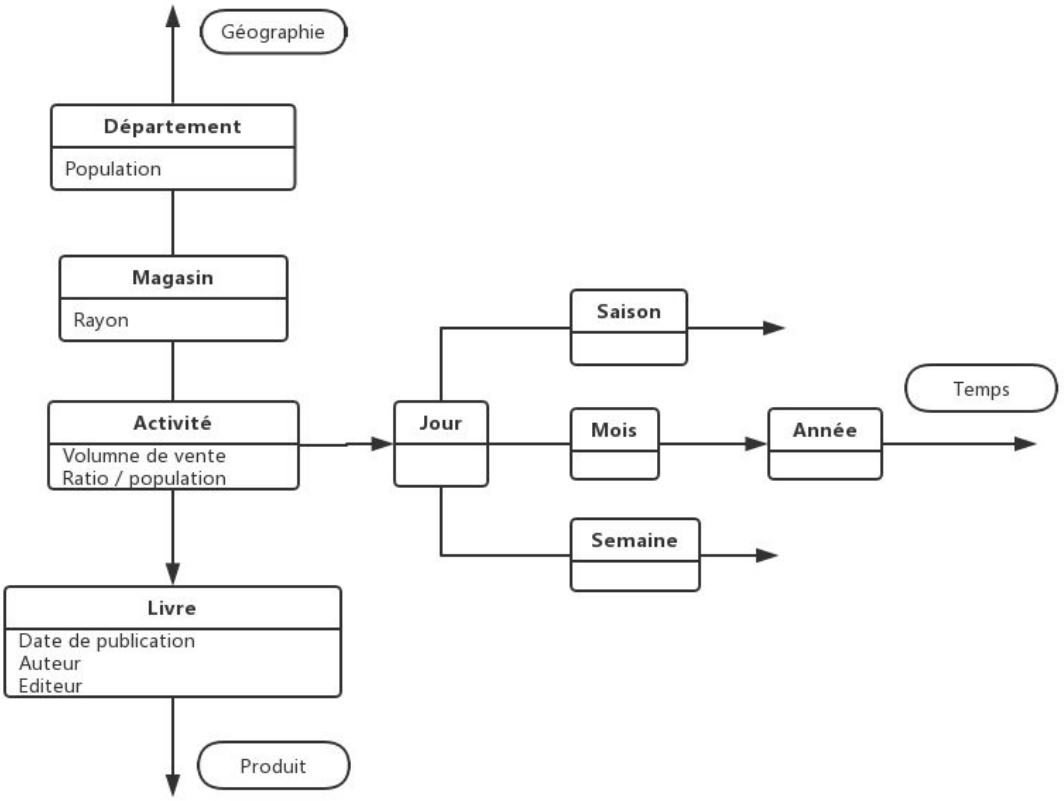
L'entreprise 'Fantastic' souhaite faire une étude large sur les ventes de l'année passée afin de prendre des orientations stratégiques nouvelles : ouverture de nouveaux magasins, fermeture ou transfert de magasins mal implantés, extension territoriale à de nouveaux départements français, réorganisation des directions, réorientation du marketing, élargissement ou réduction du catalogue, etc.

La question posée est donc : **quels sont les facteurs sur lesquels on pourrait jouer pour augmenter les ventes ?**

Ainsi, toute la modélisation et l'analyse des données de notre projet s'articule autour des ventes des produits de l'entreprise 'Fantastic', en même temps, analyser la relation entre les ventes et les divers facteurs: **l'organisation du rayonnage des magasins, jours de la semaine, certaines périodes de l'année, magasins ou départements, population, certaines livres etc.**

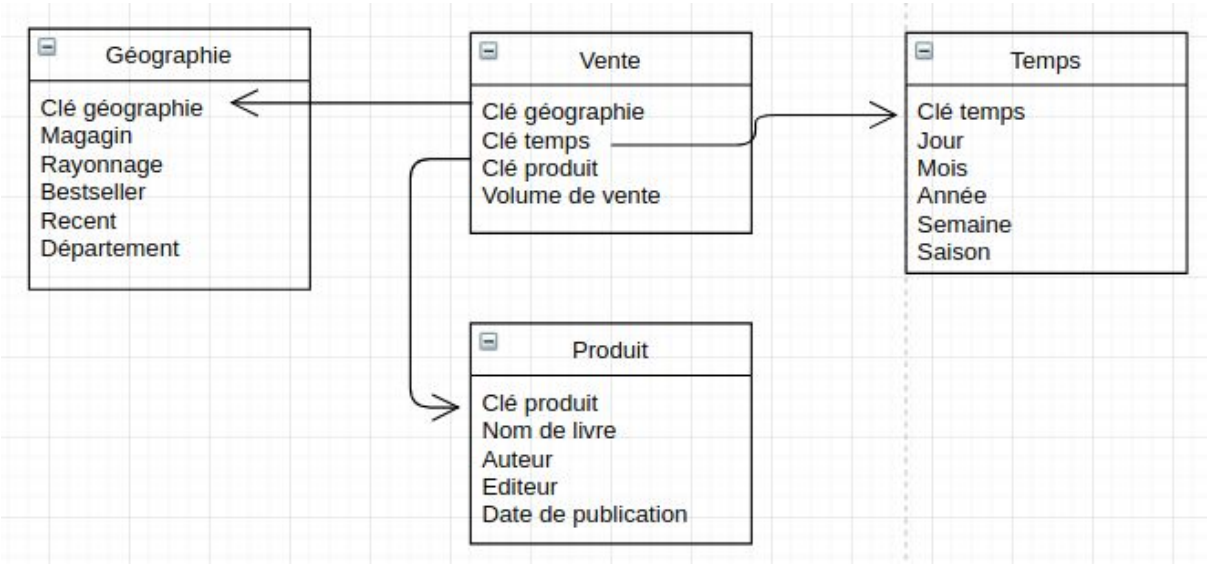
II Présentation des dimensions

Après avoir analysé les besoins du client, nous avons conçu trois dimensions pour construire notre DWH : **Temp**, **Géographie**, **Produit.**



Il faut noter que les deux entités **magasin** et **département** doivent être placées dans la même dimension **Géographie.**

Le MCD ayant été conçu indépendamment de toute contrainte d'implémentation, il va falloir définir le modèle correspondant à sa mise en œuvre opérationnelle. Ainsi, nous avons conçu le schéma étoile en analysent les données réels dans les fichiers comme une préparation pour la construction de notre DWH.



III Construction des dimensions et justifications

Dans l'étape construction des transformations pour le processus ETL des données, nous avons construit pour chaque dimension une transformation.

1. Dimension du temp

1.1 Date de début

En analysant des données brutes, nous avons constaté que toutes les transactions dans le fichier 'fantastic' avaient lieu en 2014. Par conséquent, nous fixons la date initiale au 1er janvier 2014. Ensuite, en limitant le nombre de jours générés à 365 jours, on a tout d'abord défini ainsi tous les journée dans l'année 2014.Ici, nous devons faire l'attention que le format de la date doit être défini comme **yyyy-MM-jj** selon les exigences du sujet.

1.2 Taille de l'étape

Nous avons simplement fixé le pas à un jour, afin de ne rater aucun jour de l'année 2014.

1.3 la calculation de la date

Ensuite, afin de mettre en œuvre la structure de la dimension **TEMP** définie précédemment, nous devons effectuer quelque nouveaux calculs à la date existante.

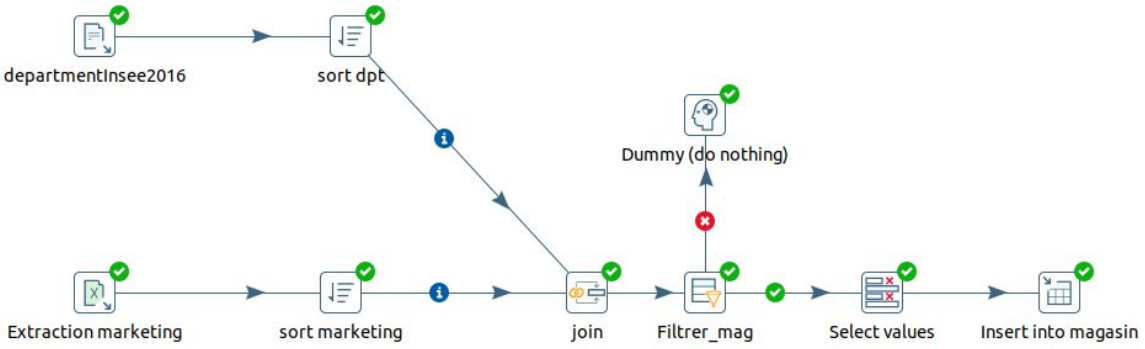
1	CurrentDate	Date A + B Days	START_DAY	Days_Since		Date		
2	Year	Year of date A	CurrentDate			Integer	4	0
3	Month	Month of date A	CurrentDate			Integer	2	0
4	DayOfWeek	Day of week of date A	CurrentDate			Integer	1	0
5	Quarter	Quarter of date A	CurrentDate			Integer	1	0

Les cinq colonnes de données dans la figure précédent correspondent au jour, l'année, le mois, la semaine et le saison dans notre dimension **TEMP**.

1.4 Insert des données dans le table TEMP

Enfin, nous renommons chaque colonne et les stockons dans la table de temps de la base de données.

2. Dimension de la géographie



2.1 Importe des fichiers

Pour la transformation de la dimension de la géométrie, nous avons importer le fichier département, où il y des données de la population de chaque département, et le fichier marketing, où il y a des données des magasins.

2.2 Jointure

Pour faire la jointure entre la table des magasins et celle des départements, il nous faut d'abord trier les données : les deux tables sont triées par rapport aux nom du département. Ensuite nous réalisons la jointure dans l'étape "**Join**".

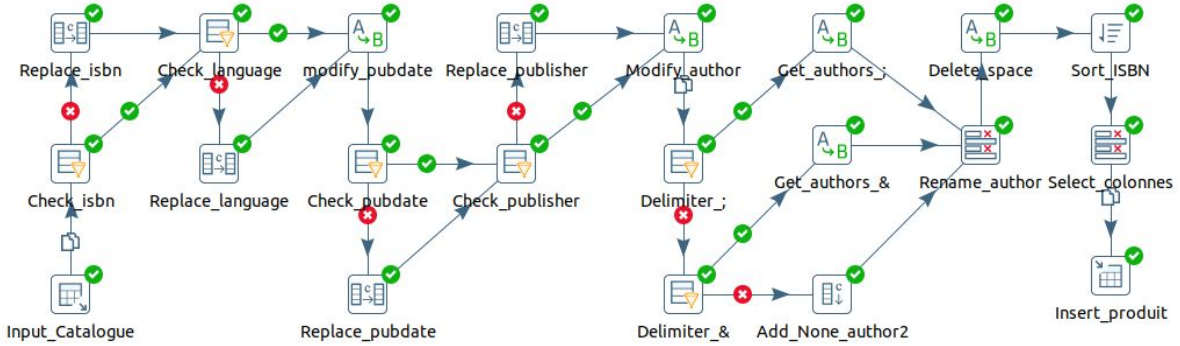
2.3 Filtre du nom des magasins

Après la jointure, pour que les données des magasin satisfassent le besoin d'être en forme "M+numéro de magasin", nous avons fait un filtre pour éliminer les mauvais nom de magasin dans l'étape "**Filter_magasin**". Dans l'exécution de la transformation, nous n'avons pas trouvés les erreur dans le noms des magasins.

2.4 Stockage dans la base de données

Finalement, nous pouvons choisir les colonnes dont nous avons besoin, pour éviter les doublonnage, pour les importer dans la base de données comme la dimension du géométrie.

3. Dimension du produit



3.1 Import des données dans la base de données

Pour manipuler les données de la catalogue, qui sont les informations des livres et qui sont stockées dans la base de données, nous avons fait un SQL, pour sélectionner tous les données de la table “Catalogue” comme données initiales dans notre transformation.

3.2 Vérification et correction du ISBN

Nous allons vérifier les données colonne par colonne pour corriger ou supprimer les erreur dans les données. Pour la colonne ISBN, où les données doivent être composées de 13 chiffres, nous utilisons l’expression régulière dans **“Check_isbn”** pour vérifier si l’ISBN de chaque ligne corresponde à ce critère ou non. Si oui, qui veut dire il n’y a pas de problème dans la donnée, la ligne peut passer à l’étape suivant. Sinon, nous avons décidé d’utiliser une chaîne de caractère spécial pour montrer que l’information dans ce colonne n’est pas bonne. Donc nous les remplaçons par 13 zéros : “0000000000000” dans **‘Replace_isbn”** pour montrer l’erreur.

3.3 Vérification et correction du language

Dans la colonne langage, il y a des null que nous devons traiter. Donc dans **“Check_language”,** nous utilisons l’expression régulière pour si les informations corresponde aux chaine de caractères ou non. Si non, qui veut dire que les données sont null, nous les remplaçons dans **“Replace_language”** par “unk” pour montrer qu’il a une donnée manquante.

3.4 Vérification et correction du pubdate

Il y a deux choses à faire pour la colonne pubdate. Premièrement, nous devons changer le format du pubdate dans **“modify_pubdate”** : enlève tous les informations en utilisant l’expression régulière à partir de la caractère “T” qui est le séparateur entre jour et de l’heure. Ensuite, nous vérifions que le jour est bien en format demandé dans

“Replace_pubdate” pour transformer tous les pubdates qui ont des mauvais informations en “0000-00-00”.

3.5 Vérification et correction du publisher

Il y a deux cas à traiter dans la colonne publisher : il y a des “?” dans certains lignes et il y aussi des null. Nous souhaitons les tout modifier dans l’étape **“Check_publisher”.** Donc en utilisant expression régulière, nous changeons tous les publisher, qui ne commence pas par un caractère ou un numéro, en “unk”.

3.6 Correction et séparation d’auteur

Pour les auteurs des livres, il faut d'abord supprimer les mauvais information : il y des chiffre dans les ligne. Nous les supprimons dans l’étape **“Check_author”** en utilisant l’expression régulière.

1163	302	978...	Le Chercheur De Pistes	1818-1883, Aimard Gustave & Bibliobazaar
1164	299	978...	La Loi De Lynch	1818-1883, Aimard Gustave & Bibliobazaar

Ensuite pour séparer les auteurs dans deux colonnes, il y a 3 cas que nous devons traiter différemment :

1) Livre qui a un seule auteur

333	20	978...	Monsieur De Camors	Bibliobazaar & 1821-1890, Feuillet Octave

2) Livre qui a plus que 1 auteurs et les auteurs sont séparés par “;”

201	2970	918	[Livre des lunes-1] 16 Lunes	Garcia,Kami; Sthol,Margaret

3) Livre qui a plus que 1 auteurs et les auteurs sont séparés par “&”

302	2088	978...	Les Maîtres Du Vortex	Smith, Edward Ellis & Watkins

Ce qu’il faut prendre en compte est qu’il peut y avoir plus que 2 auteurs :

446	2642	978...	Deus Irae	Dick, Philip K. & Zelazny, Roger & Daniels, Luke

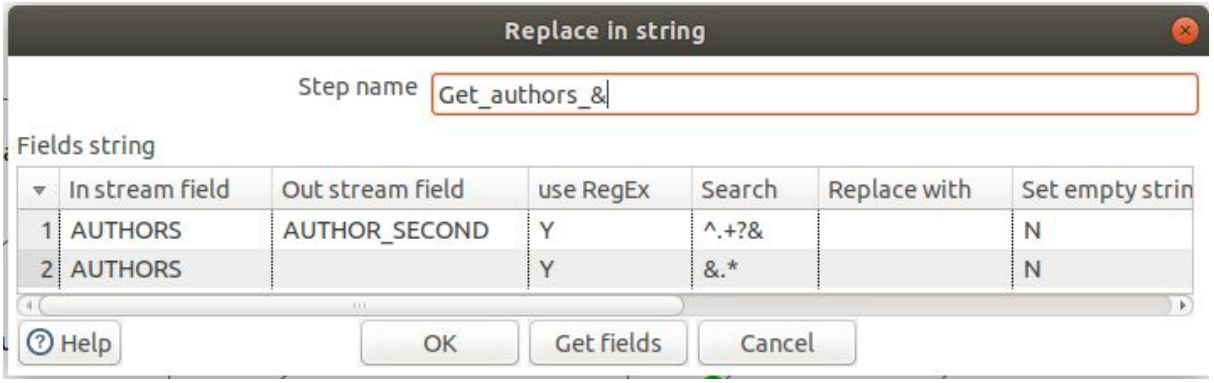
Dans ce cas, par rapport aux cahier de charge, nous devons mettre le premier auteur dans une colonne et les autres dans une autre colonne.

Pour le faire, nous vérifions d’abord le type de délimiteur, dans **“Delimiter_;”** et **“Delimiter_&”** en cherchant l'existence de ces délimiteurs. S’il y en a pas, nous ajoutons une nouvelle colonne AUTEUR_SECOND et initialisons tous les valeur par “None” dans **“Add_None_auteur2”**.

Pour les livres qui ont plus que 1 auteur, nous avons utilisé la méthode “Splite fields” au début, mais cela ne marche pas quand il y a 3 ou plus d’auteurs : seulement le deuxième auteur apparaît dans le deuxième colonne ajoutée. Donc il faut plus de colonne pour prendre tous les auteurs, qui n’est pas pratique et pas scalable s’il y a plus d’auteur.

Pour résoudre ce problème, nous décidons d’utiliser l’expression régulière pour séparer les auteurs. En utilisant “Replace in string” dans l’étape **“Get_authors_;”** et **“Get_authors_&”** , nous pouvons extraire la partie jusqu’à ‘**&**’ et la partie restant en utilisant le remplacement : nous remplaçons le premier auteur par string vide pour avoir les autres auteurs et pareille

pour le contraire. Finalement nous renommons la colonne AUTHOR par AUTHOR_FIRST dans l'étape **"Rename_author"**.

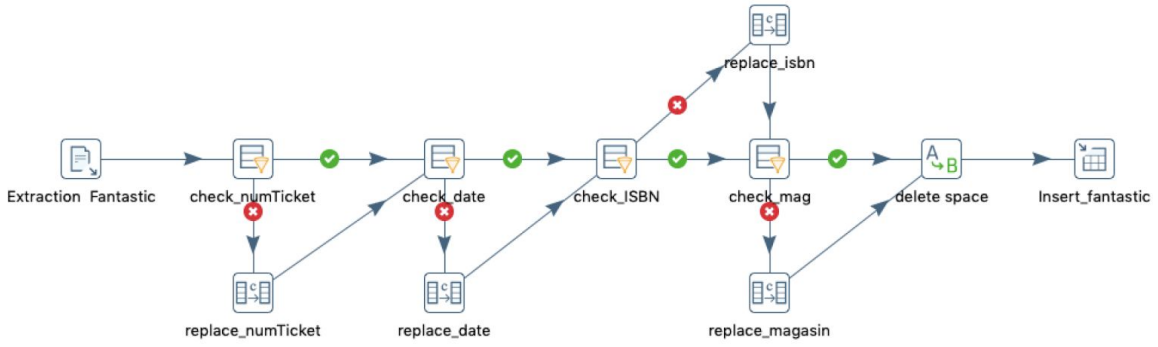


Comme il y a souvent des espace avant et après les délimiteurs, il faut supprimer les espace au début de AUTEUR_SECOND et les espace à la fin de AUTEUR_FIRST. Nous le fait dans l'étape "**Delete_space**" en utilisant expression régulière.

3.7 Insert des données dans une table.

Finalement, nous avons fait un tri sur le colonne ISBN pour mettre les données en ordre et les insérons dans une table dans la base de données.

4. Données des ventes



Le fichier fantastique stocke toutes les informations de transaction. En tant que le lien entre chaque dimension (**TEMP**, **MAGASIN**, **PRODUIT**), il occupe une position très importante. Nous devons donc faire très attention au traitement de ses données.

4.1 Vérification et correction du numéro du ticket

Selon la définition du sujet, le numéro de ticket doit être une chaîne de 9 chiffres. En utilisant l'expression régulière '**[0-9]{9}**', nous avons filtré tous les donnée qui ne répondaient pas aux exigences et les remplacé par 111111111.

4.2 Vérification et correction du data

La filtrage de la date est différente du numéro de ticket. Il a des exigences strictes en matière de format de date : yyyy-MM-dd. Seulement nous avons unifié tous les formats de date dans notre base de données, les statistiques de données dans le table **FAIT** ne causeront pas d'erreurs. En conséquence, nous utilisons des expressions régulières : '**[0-9]{4}-[0-9]{2}-[0-9]{2}',** et les remplacé par 2015-1-1 qui est une date en dehors de 2014.

4.3 Vérification et correction du ISBN

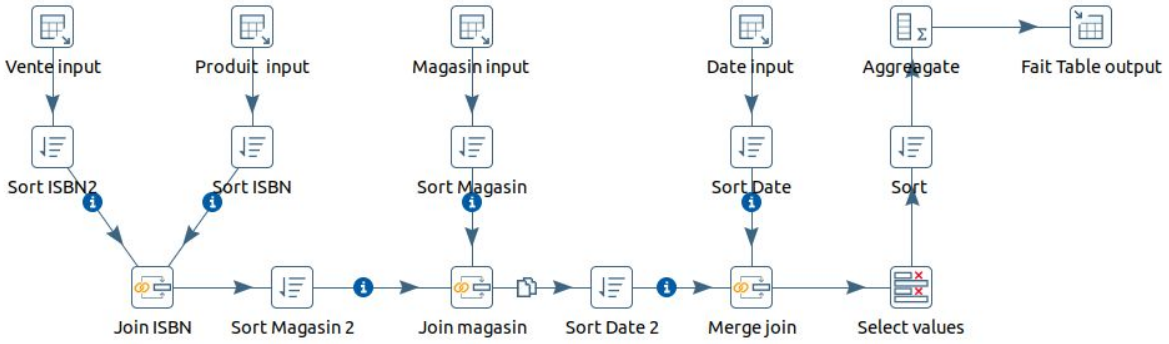
Au début, nous avons commis une erreur dans le remplacement de l'ISBN : comme ce qu'on a fait dans le traitement sur ISBN dans la **Dimension de la géographie,** tous les ISBN non conformes sont remplacés par treize zéro.

Mais cela va connecter les deux données potentiellement non liées dans une donnée apparemment correcte mais factice. Il affectera sérieusement les résultats de notre table **FAIT**. Donc ici nous le remplaçons par treize 1.

4.4 Vérification et correction du magasin

En utilisant l'expressions régulières: '**M\d+'** , nous avons filtré tous les données qui n'ont pas commencé par '**M'** et suivis de nombres et les remplacé par **M0** qui représente une magasin n'exist pas.

5. Table des faits



Une fois que tous les dimensions de données sont prêtes, nous passons à la création de la table des faits. Pour le faire, il suffit d'importer tous les dimensions de données que nous avons traitées, ensuite les trier et faire des jointure entre eux. Finalement, nous choisissons les colonnes dont nous avons besoin, par rapport aux faits : magasin, jour et ISBN. En faisant le tri et l'agrégation, nous pouvons avoir la volume de chaque produit vendu chaque jour dans chaque magasin.

IV Evaluation du résultat

1 . Dimension de la géographie

Dans l'importe des données de la géographie, la seule vérification que nous avons fait est de vérifier le nom des magasins, et nous avons constaté qu'il y a pas d'erreur dans les nom des magasin. Donc cette vérification n'a pas d'impact sur le résultat final.

2. Dimension du produit

Execution Results

Logging | Execution History | Step Metrics | Performance Graph | Metrics | Preview data

	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)
1	Input_Catalogue	0	0	1443	1443	0	0	0	0	Finished	2.6s	545
2	Check_isbn	0	1443	1443	0	0	0	0	0	Finished	2.6s	551
3	Replace_isbn	0	295	295	0	0	0	0	0	Finished	2.6s	112
4	Check_language	0	1443	1443	0	0	0	0	0	Finished	2.6s	550
5	Replace_language	0	7	7	0	0	0	0	0	Finished	2.6s	3
6	modify_pubdate	0	1443	1443	0	0	0	0	0	Finished	2.6s	549
7	Check_pubdate	0	1443	1443	0	0	0	0	0	Finished	2.6s	549
8	Replace_pubdate	0	0	0	0	0	0	0	0	Finished	2.6s	0
9	Check_publisher	0	1443	1443	0	0	0	0	0	Finished	2.6s	548
10	Replace_publisher	0	310	310	0	0	0	0	0	Finished	2.6s	118
11	Modify_author	0	1443	1443	0	0	0	0	0	Finished	2.6s	547
12	Delimiter_;	0	1443	1443	0	0	0	0	0	Finished	2.6s	546
13	Delimiter_&	0	1441	1441	0	0	0	0	0	Finished	2.6s	545
14	Get_authors_&	0	116	116	0	0	0	0	0	Finished	2.6s	44
15	Add_None_author2	0	1325	1325	0	0	0	0	0	Finished	2.6s	501
16	Get_authors_;	0	2	2	0	0	0	0	0	Finished	2.6s	1
17	Rename_author	0	1443	1443	0	0	0	0	0	Finished	2.6s	545
18	Delete_space	0	1443	1443	0	0	0	0	0	Finished	2.7s	544
19	Sort_ISBN	0	1443	1443	0	0	0	0	0	Finished	2.7s	542
20	Select_colonnes	0	1443	1443	0	0	0	0	0	Finished	2.7s	541
21	Insert_produit	0	1443	1443	0	1443	0	0	0	Finished	3.1s	471

Par rapport aux statistiques d'exécution, nous avons pu constater qu'il n'y a quand même 20 pourcent de ISBN qui ne sont pas en format demandé et qui ont été changé en "0000000000000". Dans ce cas il y a des ventes dont nous pouvons pas trouver les produit quand nous faisons des jointure. Comme mentionné précédemment, puisque nous avons faire les ISBNs erroné de la table de vente "1111111111111" et il est différent que le valeur que nous avons mis dans la table de produit, il n'y a pas de risque qu'il y a des mauvais jointures.

Comme pour tous les mauvais données dans la table, nous le changeons avec une valeur qui signifie qu'il y avait une erreur, nous ne perdre aucun lignes non nécessairement. Donc pour ces modifications que nous avons fait, il n'y a pas d'impact sur l'étude des faits.

3. Donnée des ventes

Execution Results

Logging | Execution History | Step Metrics | Performance Graph | Metrics | Preview data

#	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)	input/output
1	Extraction Fantastic	0	0	200000	200000	0	1	0	0	Finished	11.6s	17,286	-
2	Calculator	0	0	0	0	0	0	0	0	Finished	0.0s	0	-
3	check_numTicket	0	200000	200000	0	0	0	0	0	Finished	12.3s	16,223	-
4	replace_numTicket	0	7698	7698	0	0	0	0	0	Finished	12.3s	624	-
5	check_date	0	200000	200000	0	0	0	0	0	Finished	12.6s	15,819	-
6	Replace in string	0	0	0	0	0	0	0	0	Finished	0.0s	0	-
7	replace_date	0	7660	7660	0	0	0	0	0	Finished	12.6s	606	-
8	check_ISBN	0	200000	200000	0	0	0	0	0	Finished	14.7s	13,574	-
9	replace_isbn	0	44918	44918	0	0	0	0	0	Finished	14.7s	3,048	-
10	check_mag	0	200000	200000	0	0	0	0	0	Finished	14.8s	13,556	-
11	replace_magasin	0	6540	6540	0	0	0	0	0	Finished	14.8s	443	-
12	Replace in string 2	0	0	0	0	0	0	0	0	Finished	0.0s	0	-
13	delete space	0	200000	200000	0	0	0	0	0	Finished	14.8s	13,518	-

Selon les statistiques dans la figure précédent, nous avons une grande quantité de données （surtout ISBN）qui ne répondent pas à nos exigences, nous les avons donc remplacées.（7698+7660+6540+44918 = 66756）Presque 33% de données sont inutiles.

Donc au début nous avons modifié les données origine pour faire des données devenir utiles. Par exemple: (Magasin : 13 => M13, Date : 14-05-6 => 2014-05-06). Mais à la fin, nous avons trouvé que c'était une très grave erreur. Comme il n'existe pas aucune preuve permettant de prouver la crédibilité des nouvelles données, il s'agit d'un opération de falsification de données qui affectera sérieusement le résultat final.

Nous avons donc abandonné l'idée de modifier les données d'origine. Cela peut perdre d'une grande quantité d'informations de données., mais au moins nous pouvons garantir l'authenticité de nos statistiques dans notre table des faits.

4. Table des faits

Execution Results

Logging | Execution History | Step Metrics | Performance Graph | Metrics | Preview data

	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)
1	Date input	0	0	365	365	0	0	0	0	Finished	0.9s	414
2	Vente input	0	0	200000	200000	0	0	0	0	Finished	5mn 59s	556
3	Magasin input	0	0	150	150	0	0	0	0	Finished	0.5s	311
4	Sort ISBN2	0	200000	200000	0	0	0	0	0	Finished	6mn 0s	555
5	Produit input	0	0	1443	1443	0	0	0	0	Finished	2.9s	495
6	Sort ISBN	0	1443	1443	0	0	0	0	0	Finished	2.9s	500
7	Join ISBN	0	201443	155082	0	0	0	0	0	Finished	6mn 0s	559
8	Sort Date	0	365	365	0	0	0	0	0	Finished	0.9s	428
9	Sort Magasin 2	0	155082	155082	0	0	0	0	0	Finished	6mn 0s	430
10	Sort Magasin	0	150	150	0	0	0	0	0	Finished	0.5s	332
11	Join magasin	0	155232	149034	0	0	0	0	0	Finished	6mn 1s	430
12	Sort Date 2	0	149034	149034	0	0	0	0	0	Finished	6mn 1s	412
13	Merge join	0	149399	143347	0	0	0	0	0	Finished	6mn 1s	413
14	Select values	0	143347	143347	0	0	0	0	0	Finished	6mn 1s	396
15	Sort	0	143347	143347	0	0	0	0	0	Finished	6mn 11s	385
16	Aggreagate	0	143347	131936	0	0	0	0	0	Finished	6mn 12s	384
17	Fait Table output	0	131936	131936	0	131936	0	0	0	Finished	6mn 13s	353

En faisant des jointures entre les tables, nous avons d'abord environs un quart des données de produit perdues à cause des données manquant dans ISBN. Ensuite, comme nous avons changer tous les données de magasin manquantes en "M0", qui est un nom de magasin qui n'existe pas, il y a des ligne de données perdu après la jointure interne. C'est le même pour les jours, nous avons changer les mauvais données du jour en première jour de l'année 2015. Donc il y a des ligne perdu aussi dans la jointure.