



Projet de data warehouse

Conception et Realisation d'une data warehouse

Réalisé par :

Ourahou Mohamed

Ahammad Abdellatif

Encadré par :

Prof. HILAL imane

Année universitaire : **2021/2022**



Table des matières

1	Introduction	4
2	Notions généraux	5
2.1	Data Warehouse	5
2.2	Utilisations du Data Warehouse	5
2.3	Propriétés d'un Data Warehouse	6
2.4	Architecture d'un Data Warehouse	6
3	Conception et réalisation d'une data warehouse pour la base de données Salika	8
3.1	Problématique	8
3.2	Schéma de la base données	8
3.3	Schéma du Data Warehouse	9
3.4	Alimentation du Data Warehouse	13
3.4.1	Présentation d'outil Talend	13
3.4.2	Alimentation des dimensions	13
3.4.3	La table de faits payment :	17
3.5	Visualisation des résultats	18
3.5.1	Définition d'util Tableau	18
3.5.2	Visualisations	18
4	Conclusion	26

Table des figures

2.1	Data Warehouse Architecture	7
3.1	sakila database	9
3.2	Le schéma du data warehouse	12
3.3	La fenêtre principale du Talend	13
3.4	mapping	14
3.5	L'alimentation	14
3.6	L'alimentation	14
3.7	Le mapping	15
3.8	L'alimentation	15
3.9	L'alimentation	16
3.10	Le mapping	16
3.11	L'alimentation	16
3.12	la dimension payment après la combinaison par count aggregator	17
3.13	mapping	17
3.14	L'alimentation du data warehouse (table du faits)	18
3.15	le nombre de film achetés par jour	19
3.16	le nombre de film achetés par mois	19
3.17	le revenue des films achetés par année	20
3.18	le revenue des film achetés par client	21
3.19	le nombre des film achetés par client	22
3.20	les films qui génèrent meilleurs revenues	23
3.21	les films les plus achetés	24
3.22	les meilleurs magasins	25

L'entreposage de données est un phénomène né de l'énorme quantité de données électroniques stockées ces dernières années et du besoin urgent d'utiliser ces données pour atteindre des objectifs qui vont au-delà des tâches routinières liées au traitement quotidien. Dans un scénario typique, une grande entreprise a de nombreuses succursales, et les cadres supérieurs doivent quantifier et évaluer la façon dont chaque succursale contribue à la performance globale de l'entreprise. La base de données de l'entreprise stocke des données détaillées sur les tâches effectuées par les succursales. Pour répondre aux besoins des managers, des requêtes sur-mesure peuvent être émis pour récupérer les données requises. Pour que ce processus fonctionne, la base de données des administrateurs doivent d'abord formuler la requête souhaitée (généralement une requête SQL agrégée) après avoir étudié attentivement les catalogues de bases de données. Ensuite, la requête est traitée. Cela peut prendre quelques heures en raison de l'énorme quantité de données, de la complexité des requêtes et des effets simultanés d'autres requêtes de charge de travail régulières sur les données. Enfin, un rapport est généré et transmis à cadres supérieurs sous la forme d'un tableur.

Il y a de nombreuses années, les concepteurs de bases de données ont réalisé qu'une telle approche n'était guère réalisable, parce qu'il est très exigeant en termes de temps et de ressources, et qu'il n'atteint pas toujours les résultats souhaités. De plus, un mélange de requêtes analytiques avec des requêtes de routine transactionnelles ralentit inévitablement le système, et cela ne répond pas aux besoins des utilisateurs de l'un ou l'autre type de requête. Les processus d'entreposage de données avancés d'aujourd'hui séparent le traitement analytique en ligne (OLAP) du traitement transactionnel en ligne (OLTP) en créant un nouveau référentiel d'informations qui intègre des données de base provenant de diverses sources, organise correctement les formats de données, puis rend les données disponibles pour l'analyse et l'évaluation visant à la planification et à la prise de décision.

2.1 Data Warehouse

Data Warehouse ou (l'entreposage de données) est un ensemble de méthodes, de techniques et d'outils utilisés pour soutenir les travailleurs du savoir—cadres supérieurs, directeurs, gestionnaires et analystes—pour effectuer des analyses de données qui aident à effectuer des processus de prise de décision et à améliorer les ressources d'information.

2.2 Utilisations du Data Warehouse

Pour comprendre le rôle et les propriétés utiles de l'entreposage de données complètement, il faut d'abord comprendre le besoins qui l'ont fait naître. En 1996, R. Kimball résumait efficacement quelques affirmations fréquemment soumis par les utilisateurs des systèmes d'information classiques.

- "Nous avons des tas de données, mais nous ne pouvons pas y accéder !" Cela montre la frustration de ceux qui sont responsables de l'avenir de leurs entreprises mais qui n'ont pas d'outils techniques pour les aider à extraire les informations requises dans un format approprié.
- "Comment des personnes jouant le même rôle peuvent-elles obtenir des résultats substantiellement différents ?" nombreuses bases de données sont généralement disponibles, chacune consacrée à un Zone commerciale. Ils sont souvent stockés sur différents supports logiques et physiques qui sont pas intégré conceptuellement. Pour cette raison, les résultats obtenus dans chaque entreprise sont susceptibles d'être incompatibles.

- "Nous voulons sélectionner, regrouper et manipuler les données de toutes les manières possibles !" les processus de La prise de décision ne peuvent pas toujours être planifiés avant que les décisions ne soient prises. Les utilisateurs finaux ont besoin un outil convivial et suffisamment flexible pour effectuer des analyses ad hoc.
- "Montrez-moi ce qui compte !" Examiner les données au niveau de détail maximal n'est pas seulement inutile pour les processus de prise de décision, mais est également vouée à l'échec, car elle ne permet pas aux utilisateurs de se concentrer sur des informations significatives.
- "Tout le monde sait que certaines données sont erronées !" C'est un autre point sensible. Un appréciable pourcentage de données transactionnelles n'est pas correct ou n'est pas disponible. Il est clair que on ne peut pas obtenir de bons résultats si on base l'analyses sur des données incorrectes ou incomplètes.

2.3 Propriétés d'un Data Warehouse

- *accessibilité* aux utilisateurs peu familiarisés avec l'informatique et les structures de données ;
- *intégration* des données sur la base d'un modèle d'entreprise standard ;
- *flexibilité* des requêtes pour maximiser les avantages obtenus à partir des informations existantes ;
- *la concision des informations* permettant des analyses ciblées et efficaces ;
- *représentation multidimensionnelle* offrant aux utilisateurs une vue intuitive et gérable d'information ;
- *exactitude et exhaustivité* des données intégrées
- *Orientée sujet* car les entrepôts de données s'appuient sur des concepts, tels que les clients, les produits, les ventes et les commandes. Au contraire opérationnel les bases de données reposent sur de nombreuses applications spécifiques à l'entreprise.

2.4 Architecture d'un Data Warehouse

Les propriétés d'architecture suivantes sont essentielles pour un système d'entrepôt de données :

1. **Séparation** : Le traitement analytique et transactionnel doit être séparé autant que possible.
2. **Évolutivité** : Les architectures matérielles et logicielles doivent être faciles à mettre à niveau car le volume de données à gérer et à traiter et les besoins d'utilisateurs , qui doivent être satisfaits, augmentent progressivement.

3. **Extensibilité** : L'architecture doit pouvoir héberger de nouvelles applications et technologies sans repenser l'ensemble du système.
4. **La surveillance** : des accès de sécurité est essentielle en raison des données stratégiques stockées dans des entrepôts de données.
5. **Administrabilité** : La gestion de l'entrepôt de données ne devrait pas être trop difficile.

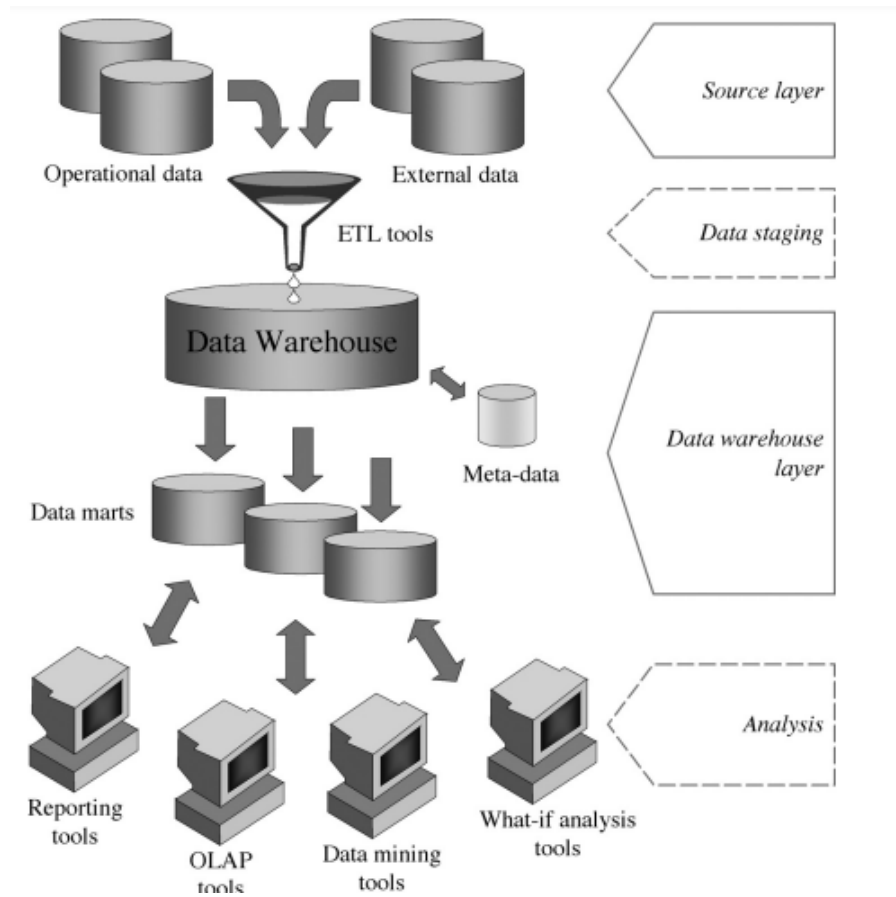


FIGURE 2.1 – Data Warehouse Architecture

Conception et réalisation d'une data warehouse pour la base de données Salika

3.1 Problématique

la société Dell met un magasin de DVD en ligne qui distribue des films dans un DVD. Elle décide de réaliser une base de données afin de faire le suivi des achats des DVD et des paiements effectués par des clients. Elle veut faire un suivi indépendant de chacun des paiements par jour, par mois, par année, mais aussi avoir la possibilité d'un suivi global.

3.2 Schéma de la base données

La base de données **Sakila** est conçue pour représenter un magasin de location de DVD. La base de données Sakila emprunte toujours des films et noms d'acteurs de l'exemple de base de données Dell. **Sakila sample database** a été conçue pour remplacer la base de données **world**, également fourni par Oracle.

Le développement de la base de données Sakila a commencé au début de 2005. Les premières conceptions étaient basées sur la base de données utilisée dans le livre blanc Dell Three Approaches to MySQL Applications on Dell PowerEdge Servers.

Alors que la base de données Dell a été conçu pour représenter un magasin des DVD en ligne.

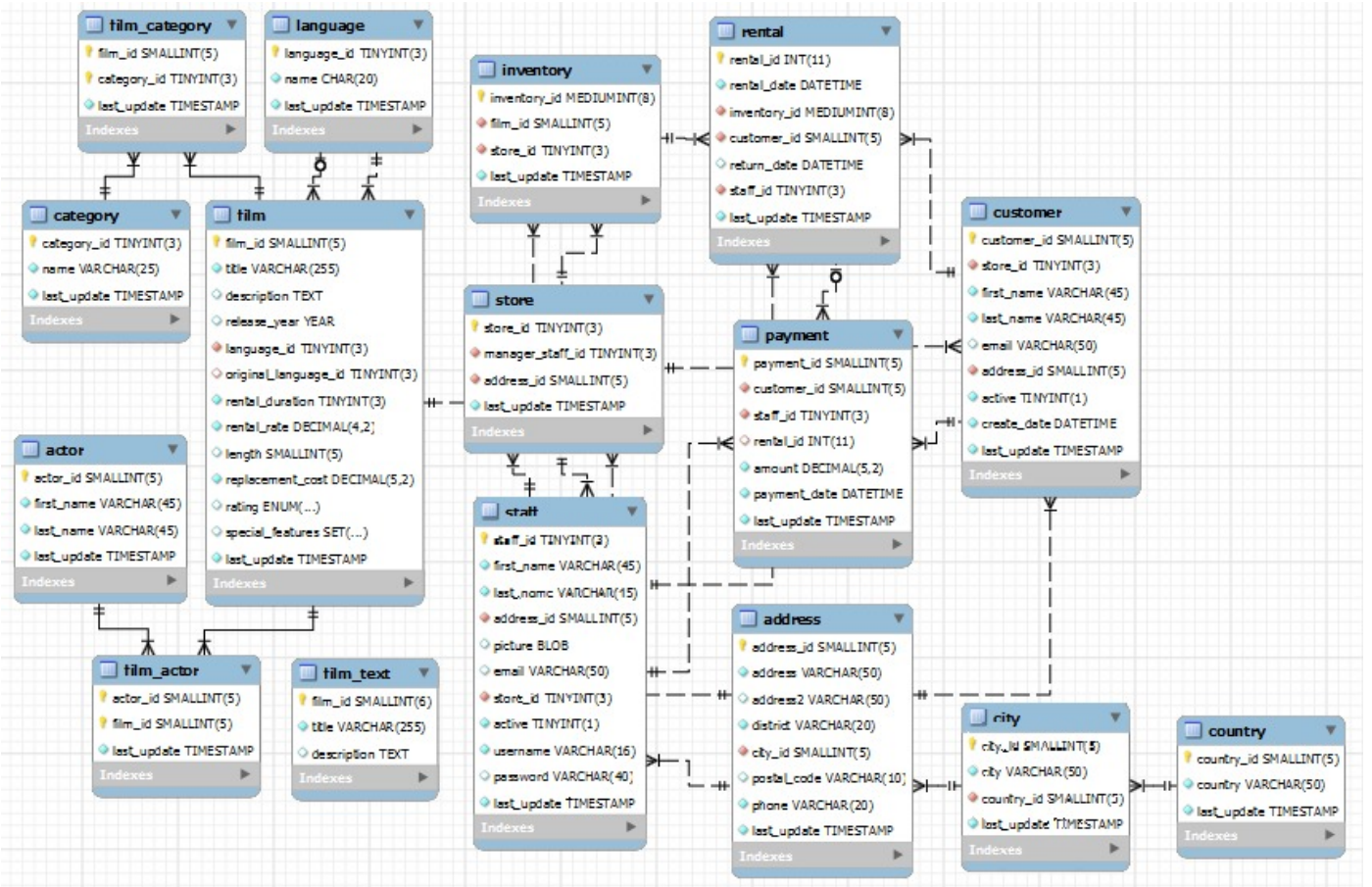


FIGURE 3.1 – sakila database

3.3 Schéma du Data Warehouse

Les responsables de la décision veulent avoir facilement la réponse aux questions suivant :

1. Quel est le revenue des films achetés par jour ?
2. Quel est le revenue des films achetés par mois ?
3. Quel est le revenue des films achetés par année ?
4. Quel est le revenue des films achetés par client ?
5. Quel est le nombre de film achetés par client ?
6. Quels sont les films qui génèrent le meilleurs revenues ?
7. Quels sont les films les plus achetés ?
8. Quels sont les meilleurs store (magazins) ?

Pour répondre aux questions de la décision on va considérer just les tableaux du schéma de la base de données salika qui vont contribuer à y répondre. Alors le schéma de data warehouse contient :

1. **La dimension Film** : est une liste de tous les films potentiellement en stock dans les magasins.

Les attributs:

- *film_id* : Une clé primaire de substitution utilisée pour identifier de manière unique chaque film dans la table.
- titre : Le titre du film.
- description : une courte description ou un résumé de l'intrigue du film.
- *release_year* : l'année de sortie du film.

2. **La dimension customers** : contient une liste de tous les clients. La dimension customer est référencée dans la table de faits payment à l'aide d'une clé étrangère.

Les attributs :

- *customer_id* : clé primaire de substitution utilisée pour identifier de manière unique chaque client dans la table.
- *fullname* : le nom et prénom du client.
- email : L'adresse e-mail du client.
- active : Indique si le client est un client actif. Définir ceci sur FALSE si le client n'est pas actif

3. **La table du faits payment** : enregistre chaque paiement effectué par un client, avec des informations telles que le montant et l'achat payée (le cas échéant).

Les attributs

- *payment_id* : une clé primaire de substitution utilisée pour identifier de manière unique chaque paiement.
- *customer_id* : Le client dont le solde est appliqué au paiement. Ceci est une clé étrangère référence à la dimension customer.
- *staff_id* : le membre du personnel qui a traité le paiement. Il s'agit d'une référence de clé étrangère au dimension staff.
- *stor_id* : Il s'agit d'une référence de clé étrangère au dimension stores.
- *film_id* : Il s'agit d'une référence de clé étrangère au dimension film.
- amount : Le montant du paiement.
- count : le nombre de films achetés pour chaque paiement.
- *date_id* : designe la date du paiement Il s'agit d'une référence de clé étrangère au dimension date.

4. **La dimension staff** : répertorie tous les membres du personnel, y compris les informations sur l'adresse e-mail et le nom complet.

La dimension staff fait référence aux table de faits payment à l'aide de clé étrangère *staff_id*

Les attributs

- *staff_id* : Une clé primaire de substitution qui identifie de manière unique le membre du staff.
- *full_name* : Le nom complet du personnel.
- *email* : L'adresse e-mail du membre du personnel.

5. **La dimension store** : répertorie tous les magasins du système.

Les attributs:

- *store_id* : Une clé primaire de substitution qui identifie de manière unique le store.
- *postal_code* : code postal de magasin.
- *address* : désigne l'adresse du magasin.
- *district* : La région d'une adresse, cela peut être un état, une province, une préfecture, etc.
- *phone* : le numéro de tél de magasin.
- *staffFullName* : le nom complet du responsable de magasin.

6. **La dimension date** : répertorie tous les dates des opérations effectuées dans le système.

Les attributs:

- *date_id* : Une clé primaire de substitution qui identifie de manière unique les dates.
- *day* : désigne le jour d'opération.
- *month* : désigne le mois d'opération.
- *Year* : désigne l'année d'opération.

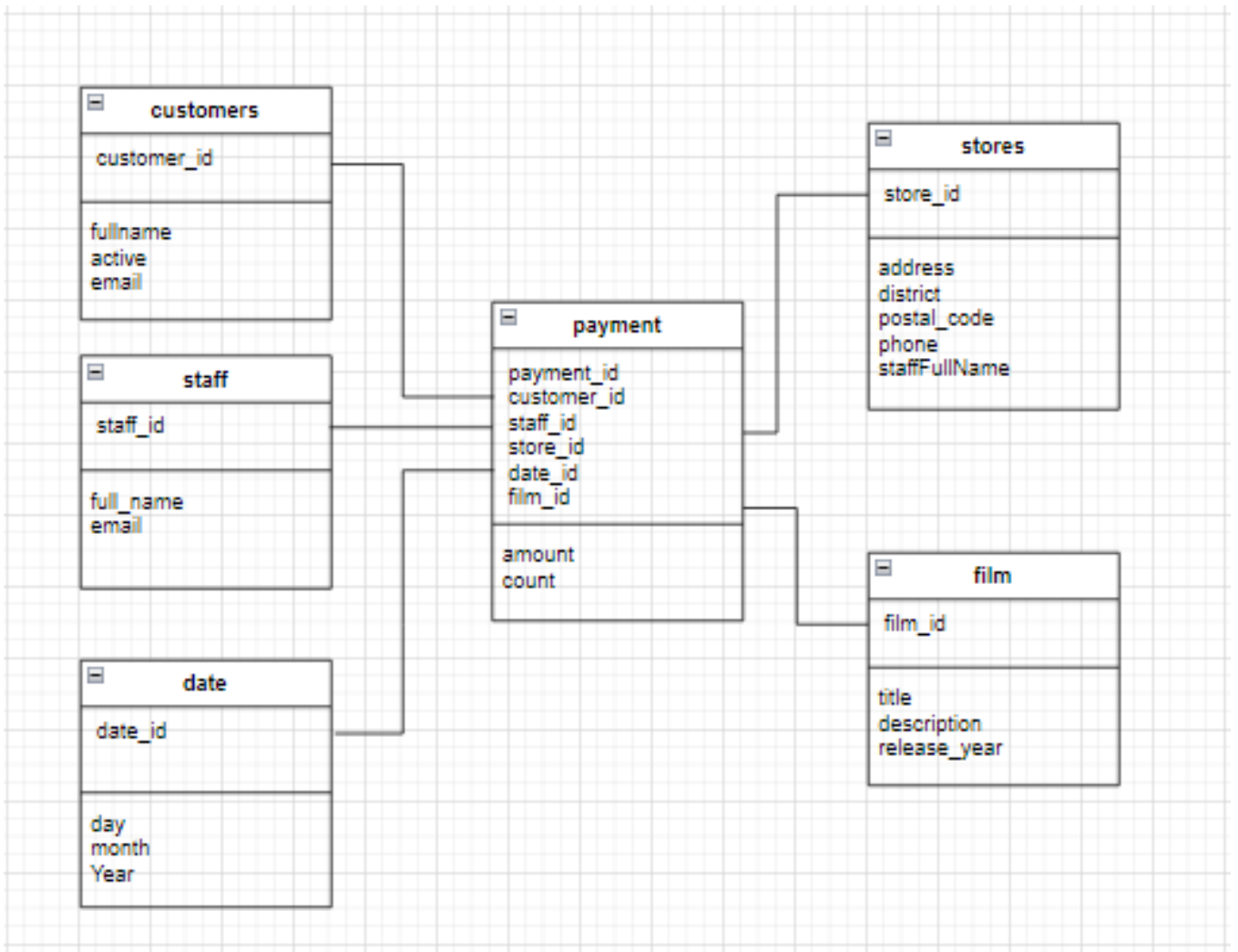


FIGURE 3.2 – Le schéma du data warehouse

3.4 Alimentation du Data Warehouse

3.4.1 Présentation d'outil Talend

Talend est un **ETL** (Extract Transform and Load) qui permet d'extraire des données d'une source, de modifier ces données, puis de les recharger vers une destination. La source et la destination des données peuvent être une base de données, un service web, un fichier csv. et bien d'autres...

Talend peut donc être utilisé dans n'importe quel contexte où des données sont véhiculées.

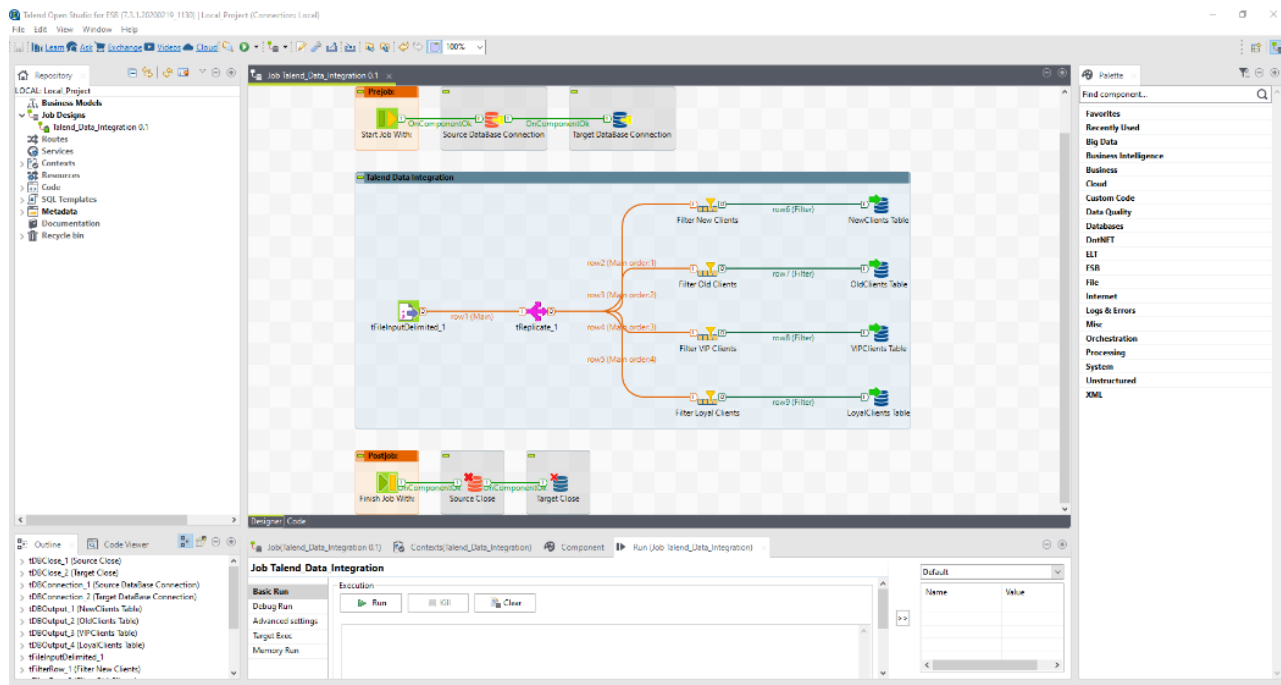


FIGURE 3.3 – La fenêtre principale du Talend

3.4.2 Alimentation des dimensions

— *La dimension film :*

row1	Var	out1
Column		Expression
nb		
film_id		row1.film_id
title		row1.title
description		row1.description
release_year		row1.release_year
language_id		
original_language_id		
rental_duration		
rental_rate		
length		
replacement_cost		
rating		
last_update		
		Column
		film_id
		title
		description
		release_year

FIGURE 3.4 – mapping

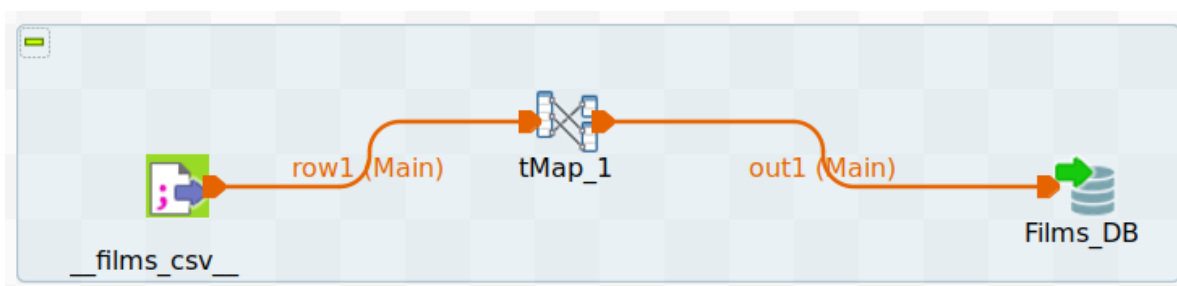


FIGURE 3.5 – L'alimentation

— *La dimension customers :*

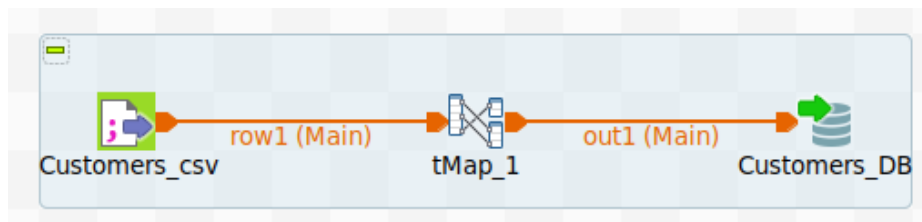


FIGURE 3.6 – L'alimentation

— *La dimension store :*

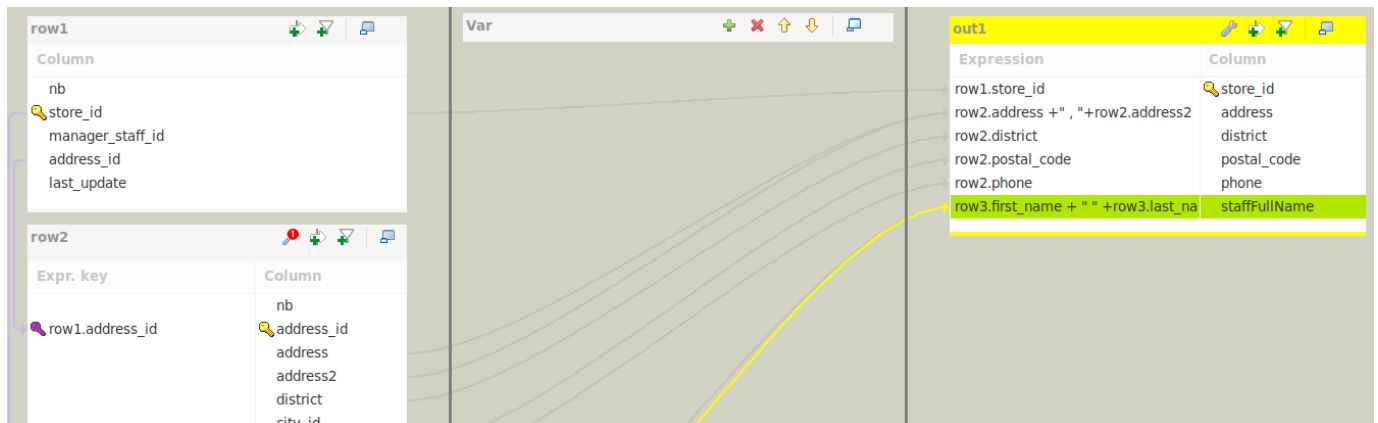


FIGURE 3.7 – Le mapping

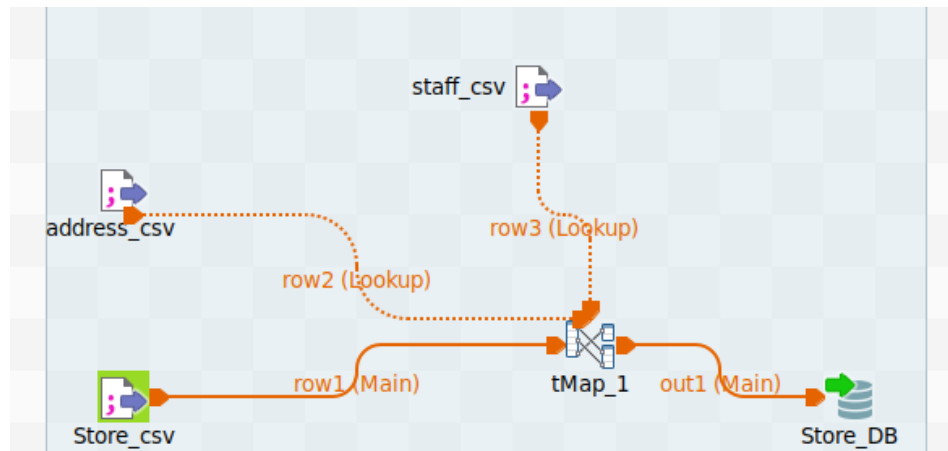


FIGURE 3.8 – L'alimentation

— *La dimension staff :*

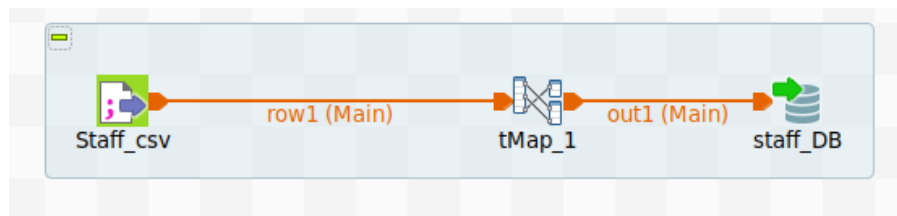


FIGURE 3.9 – L'alimentation

— *La dimension date :*

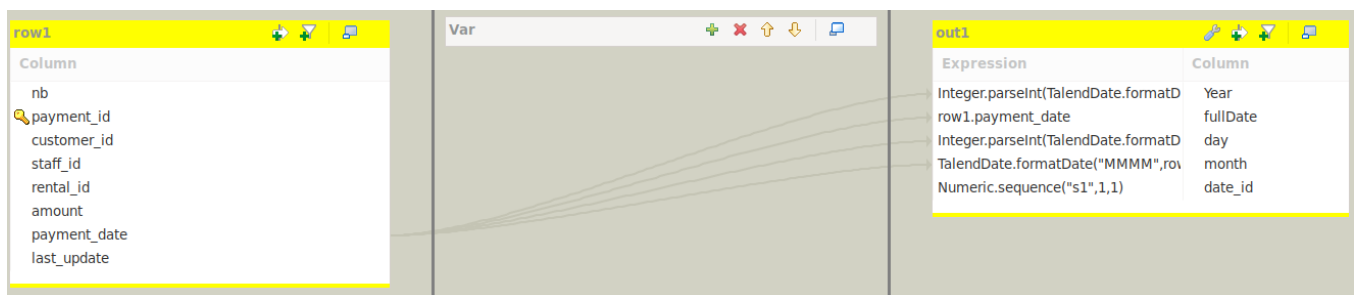


FIGURE 3.10 – Le mapping

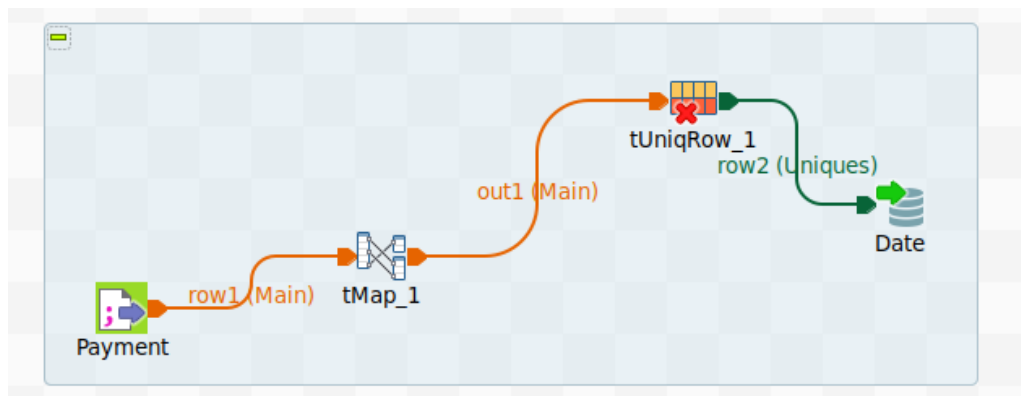
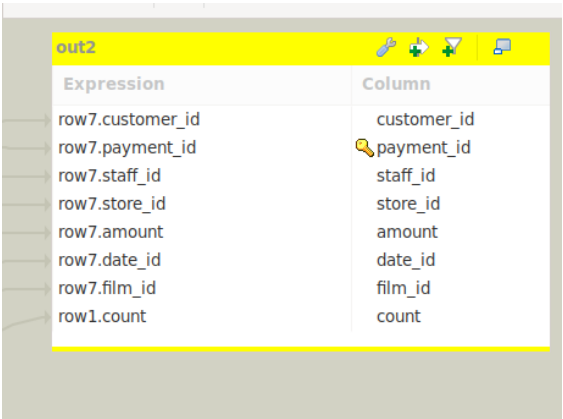


FIGURE 3.11 – L'alimentation

3.4.3 La table de faits payment :



Expression	Column
row7.customer_id	customer_id
row7.payment_id	payment_id
row7.staff_id	staff_id
row7.store_id	store_id
row7.amount	amount
row7.date_id	date_id
row7.film_id	film_id
row1.count	count

FIGURE 3.12 – la dimension payment après la combinaison par count aggregator

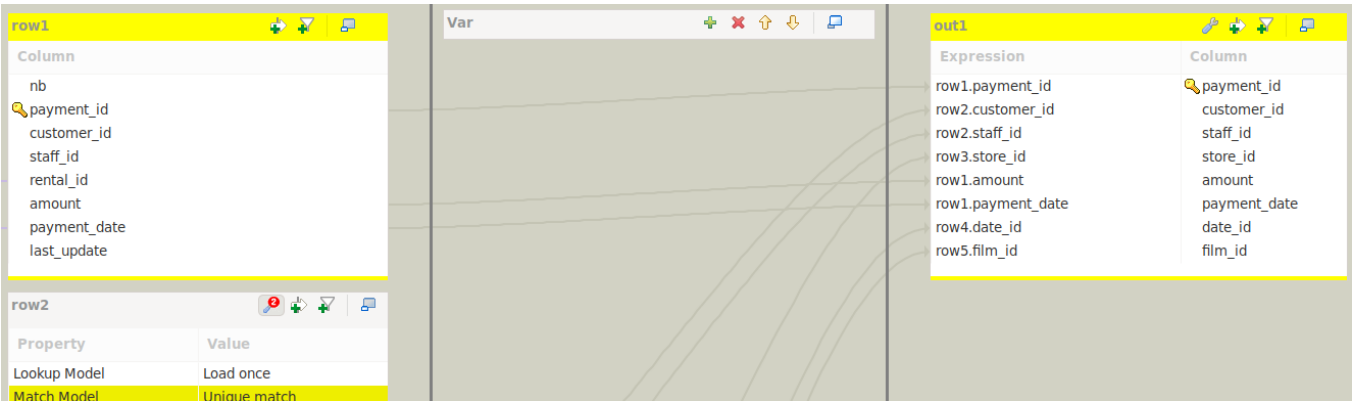


FIGURE 3.13 – mapping

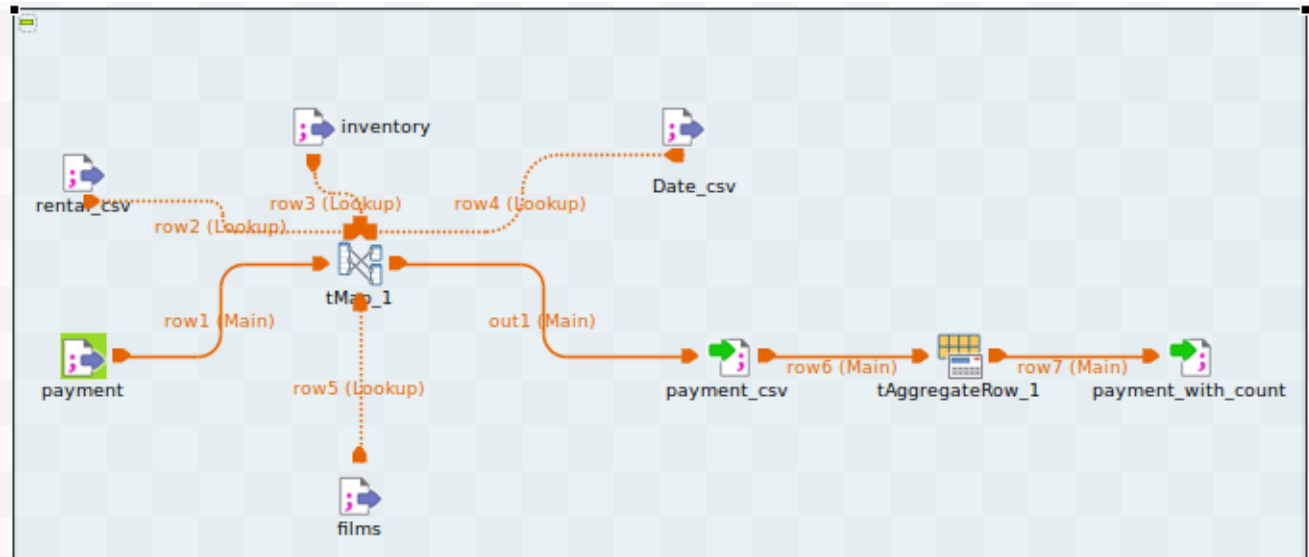


FIGURE 3.14 – L’alimentation du data warehouse (table du faits)

3.5 Visualisation des résultats

3.5.1 Définition d’outil Tableau

Tableau est souvent vu comme un outil de BI et de data visualisation mais en réalité, il est bien plus que cela. Tableau s’utilise principalement en tant qu’outil de data visualisation, mais apporte d’autres fonctionnalités, qui permettent de connecter, traiter et modéliser les données grâce aux différents éléments de la suite. Il permet de refaire toute la chaîne des données, ou qui peut être mis en complément d’autres outils (par exemple, une entreprise qui souhaite utiliser son propre ETL à la place de Tableau Prep Builder).

Tableau s’utilise aussi en ligne grâce à **Tableau Server**, Online et Public, les services cloud et On Premise de Tableau. Cela permet de visionner et partager ses tableaux de bords au sein de son entreprise, ou en dehors de celle-ci. Ces versions serveurs facilite le partage des analyses aux utilisateurs, sans avoir besoin d’installer Tableau sur leurs machines.

3.5.2 Visualisations

La visualisation des données obtenu permet de répondre aux questions proposés dès le départ par le décideur.

- Quel est le revenu des films achetés par jour ?

Feuille 1

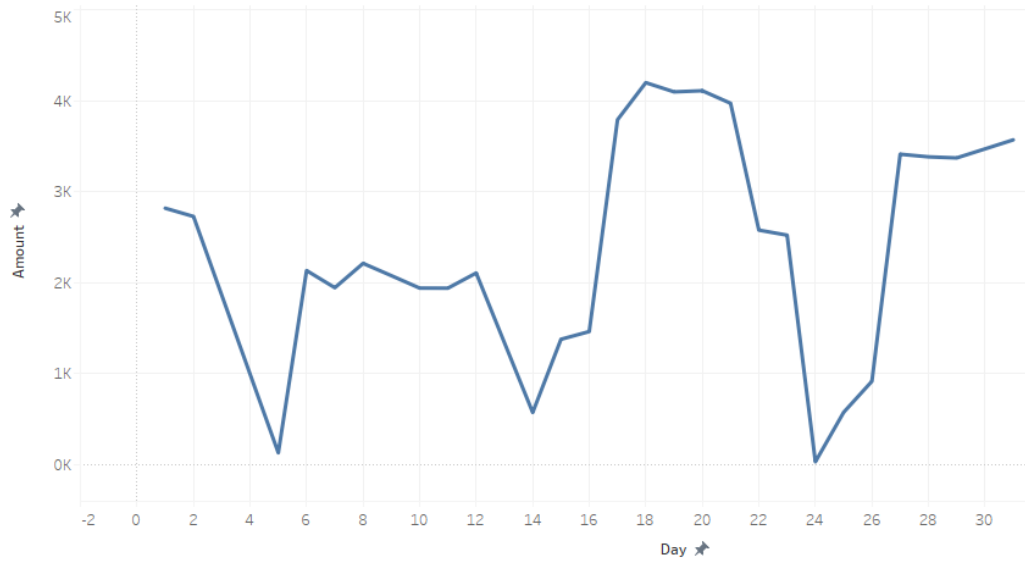


FIGURE 3.15 – le nombre de film achetés par jour

— Quel est le revenue des films achetés par mois ?

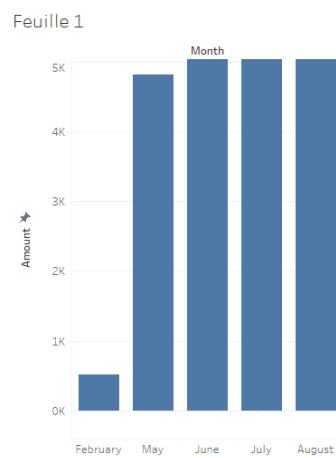


FIGURE 3.16 – le nombre de film achetés par mois

— Quel est le revenue des films achetés par année ?

Feuille 1

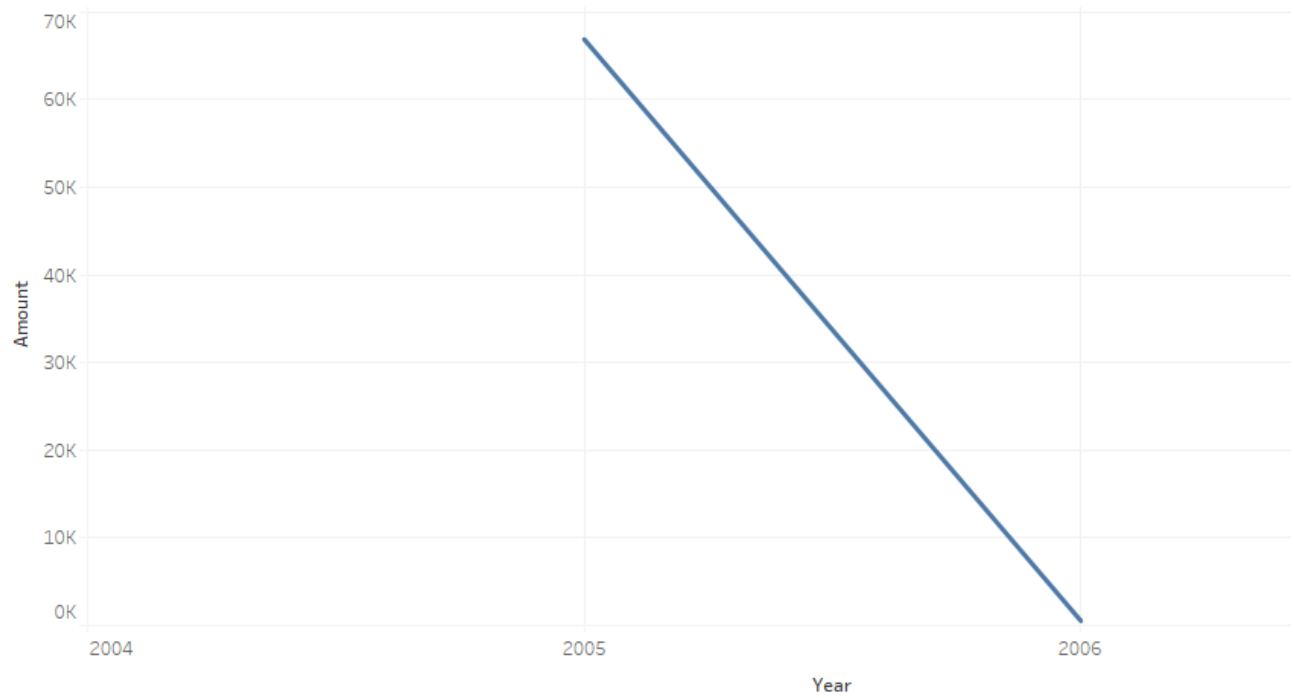


FIGURE 3.17 – le revenue des films achetés par année

— Quel est le revenu des films achetés par client ?

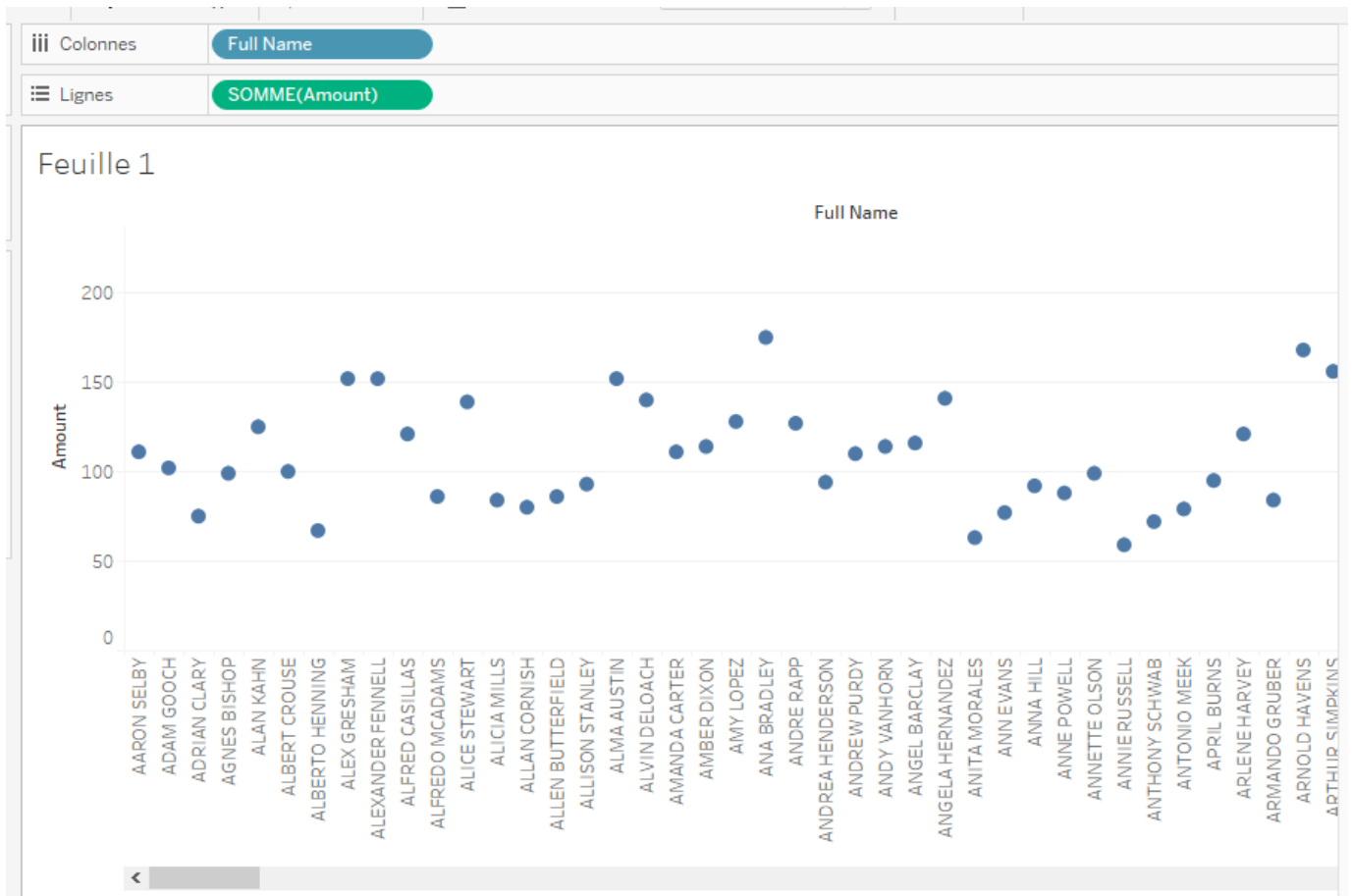


FIGURE 3.18 – le revenu des film achetés par client

— Quel est le nombre des films achetés par client ?

Feuille 1

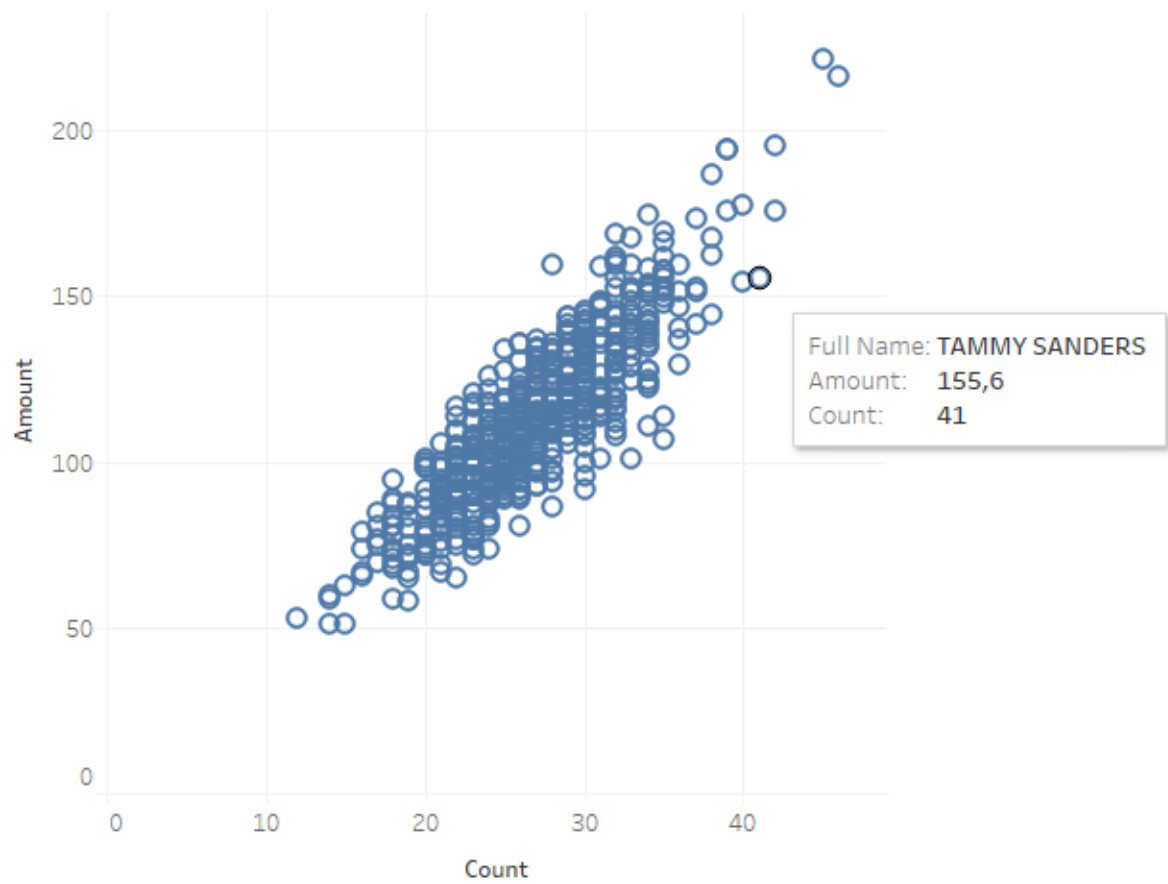


FIGURE 3.19 – le nombre des film achetés par client

— Quels sont les films qui génèrent meilleurs revenus ?

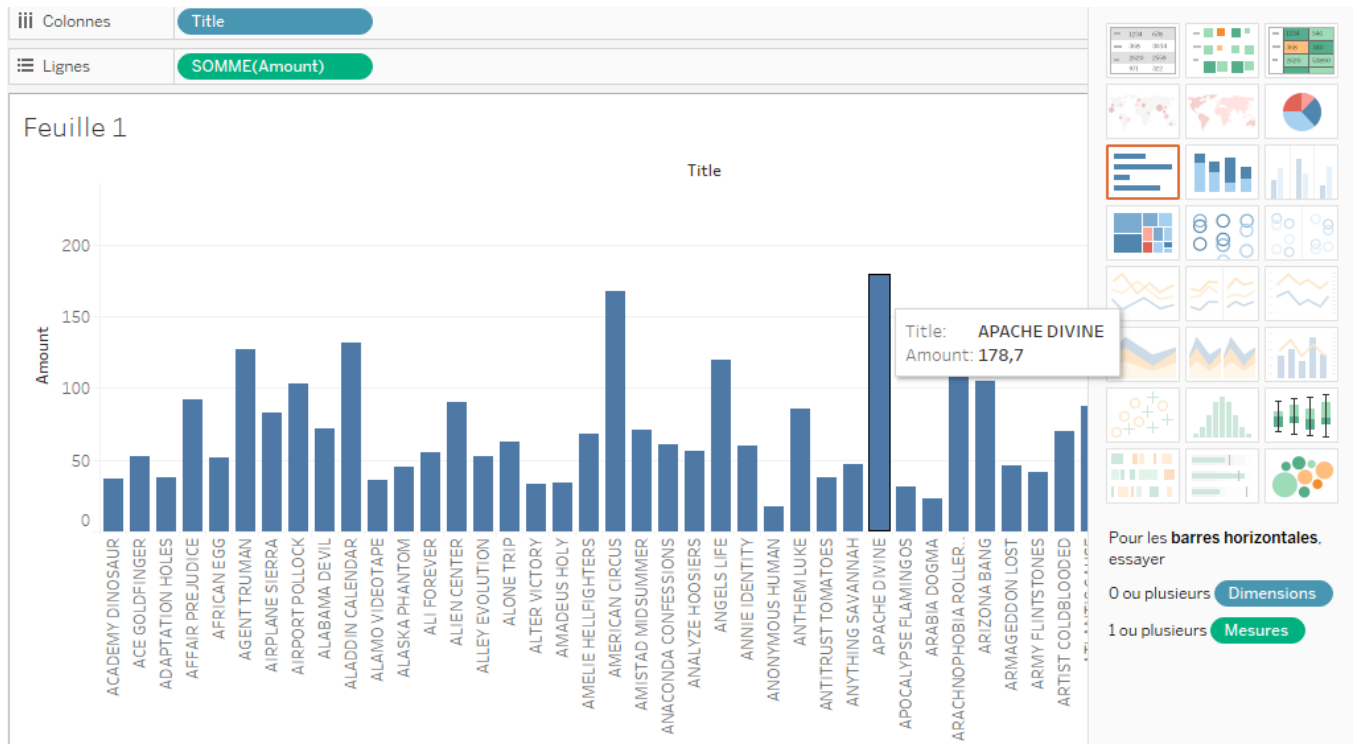


FIGURE 3.20 – les films qui génèrent meilleurs revenus

— Quels sont les films les plus achetés ?

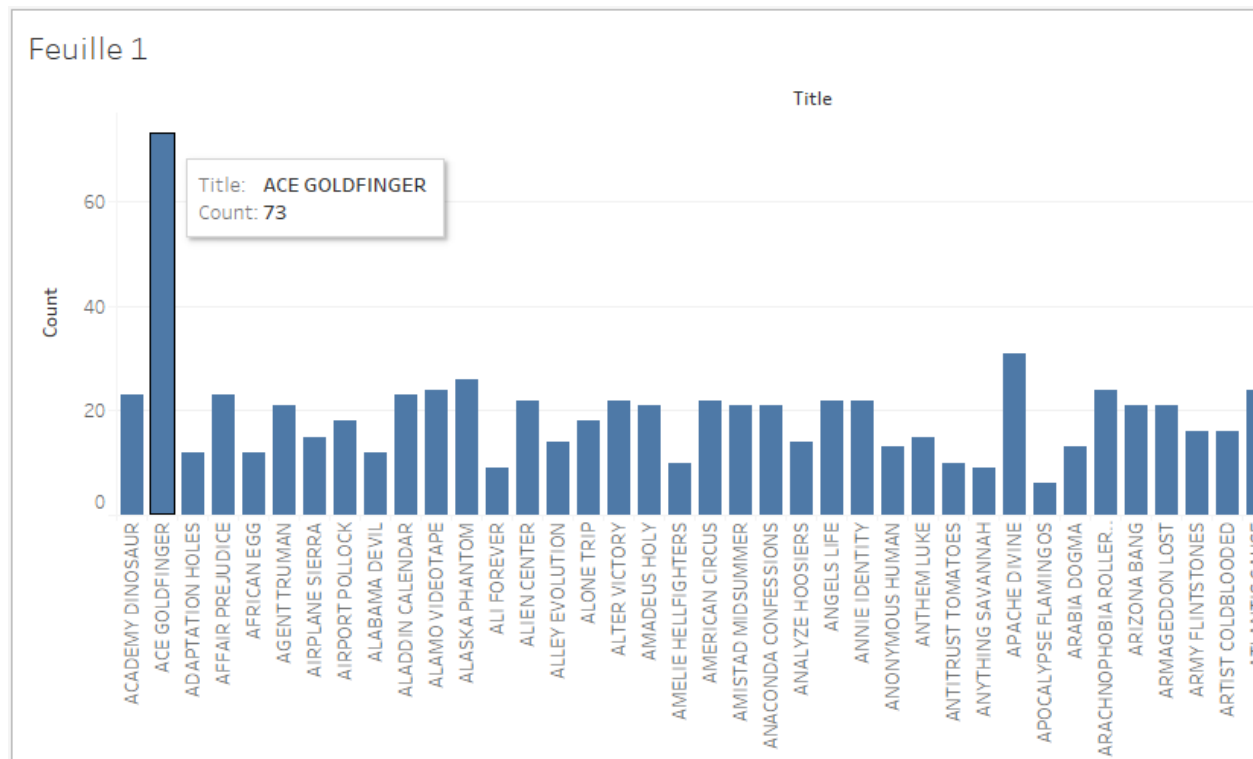


FIGURE 3.21 – les films les plus achetés

— Quels sont les meilleurs store (magazins) ?

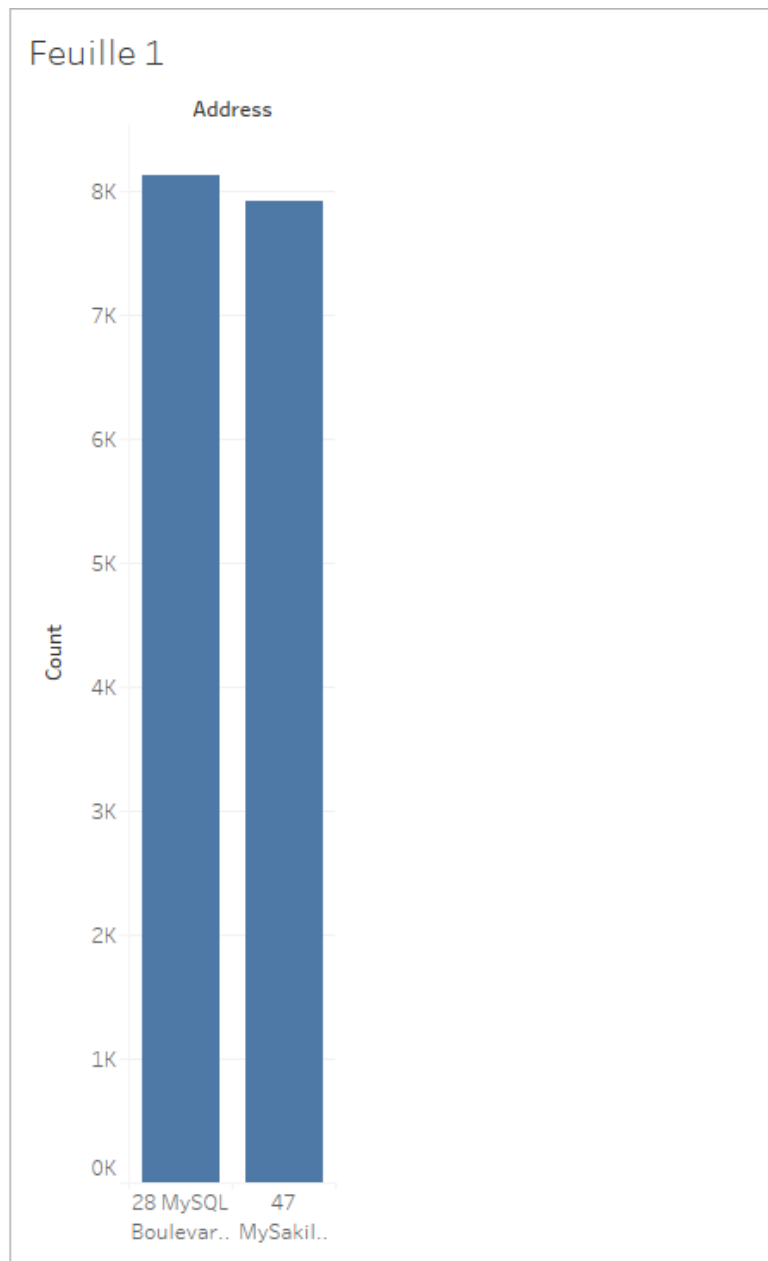


FIGURE 3.22 – les meilleurs magasins

L'intégration des données dans un Data Warehouse a permis de les regrouper et de les rendre homogènes après quelques transformations. Ces données sont intégrées dans l'entrepôt à l'aide du logiciel Talend. A travers ce travail de conception et de réalisation, nous avons pu constater qu'utiliser une approche mixte est plus avantageux pour répondre aux attentes des utilisateurs tout en exploitant au mieux les données générées par les systèmes opérationnels afin de pouvoir anticiper des besoins non exprimés. L'objectif qui a été d'instaurer un Data Warehouse pour contribuer à l'accélération du mécanisme de traitement des déclarations des films au sein de la société Dell a donc été atteints. Toutefois, une des principales raisons de l'échec d'un projet d'entrepôt de données est le manque d'entretien. Sans entretien adéquat, les résultats souhaités sont presque impossibles à atteindre depuis l'entrepôt.

Bibliographie

- [1] <https://dev.mysql.com/doc/sakila/en/>
- [2] <https://www.tableau.com/support/releases>
- [3] <https://www.talend.com/fr/products/talend-open-studio/>