



M1 INFORMATIQUE AIGLE

HMIN122M

MINI-PROJET : ENTREPÔTS DE DONNÉES

Rapport

Bachar RIMA

Joseph SABA

Tasnim SHAQURA MUHAMMAD

Jérémy BOURGIN

8 novembre 2018

Table des matières

1	Introduction	2
1.1	Problématiques	2
1.2	Actions et opérations	2
1.3	Exemples de requêtes analytiques possibles	3
2	Modélisation	5
3	Implémentation	9
4	Conclusion	10

Chapitre 1

Introduction

1.1 Problématiques

Dans le cadre du mini-projet du module **HMIN122M**, nous avons décidé de modéliser un entrepôt de données pour le réseau de transport public de Montpellier, *tam-voyages*. Pour ce faire, nous avons proposé des *data marts* formant le *data warehouse* et permettant de réaliser des requêtes analytiques sur un ensemble important de données. Cette modélisation permettra ainsi de mettre en œuvre un outil d’analyse permettant de bien répondre aux problématiques suivantes :

1. Comment *tam-voyages* pourront-ils profiter de la fréquentation de leurs véhicules en se basant sur la circulation du réseau¹ pour améliorer la qualité de leur service (e.g. *nombre de véhicules à envoyer sur une ligne pendant une certaine période de la journée*) ?
2. Comment *tam-voyages* pourront-ils suivre l’évolution et la maintenance de leurs matériaux de manière à réduire les dépenses qui y sont associées ?

Ces problématiques seront ainsi adressées en analysant les actions et opérations effectuées par *tam-voyages*, notamment en choisissant celles qui paraissent les plus pertinentes et les plus importantes en termes de données intégrées et flexibilité des critères d’analyse.

1.2 Actions et opérations

Les actions/opérations effectuées par *tam-voyages* considérées :

— Les voyages.

1. en particulier en examinant les lignes de tramway et les bus

- La maintenance de véhicules.
- Les ventes de tickets et les abonnements.
- Les amendes.

Les actions considérées, par ordre d'importance :

1. « voyages ».
2. « ventes ».
3. « maintenance ».
4. « amendes ».

Les actions les plus pertinentes à analyser vis-à-vis les problématiques avancées sont « voyages » et « maintenance » qu'on traitera de la manière suivante :

voyages : modèle en étoile détaillé.

maintenance : modèle en étoile *moins* détaillé, en particulier le modèle intitulé "*periodic snapshot*".

1.3 Exemples de requêtes analytiques possibles

1. exemples de requêtes analytiques pour l'action « voyages » :
 - le nombre de voyageurs par bus, utilisant des tickets pour le mois de juillet.
 - le prix moyen par voyage pendant les vacances de Noël de 2018. :
 - le nombre de voyageurs abonnés par ligne pour chaque voyage pour les deux derniers mois.
 - l'arrêt le plus fréquenté par toutes les lignes de circulation.
2. exemples de requêtes analytiques pour l'action « maintenance » :
 - le nombre de bus maintenus pour le mois de septembre 2018.
 - les X mécaniciens les plus expérimentés convoqués pour la maintenance des bus l'année précédente.
 - les X premières véhicules nécessitant le plus de maintenance pour les 6 dernier mois.
3. exemples de requêtes analytiques pour l'action « ventes » :
 - le nombre d'abonnés ayant plus que 26 ans pour le mois d'août 2018.
 - le nombre d'abonnés par date de naissance pour l'année 2018.
 - les types d'abonnement les plus fréquents pour l'année 2018.
4. exemples de requêtes analytiques pour l'action « amendes » :

- les lignes qui ont générées le plus d'amendes pour les deux derniers mois.
- les lignes les plus contrôllées de la semaine dernière.
- le nombre des abonnés qui ont reçu des amendes par ligne, l'avant-midi.
- la somme total d'amendes rapportée par type de voyageur par ligne pour le dernier mois.

Chapitre 2

Modélisation

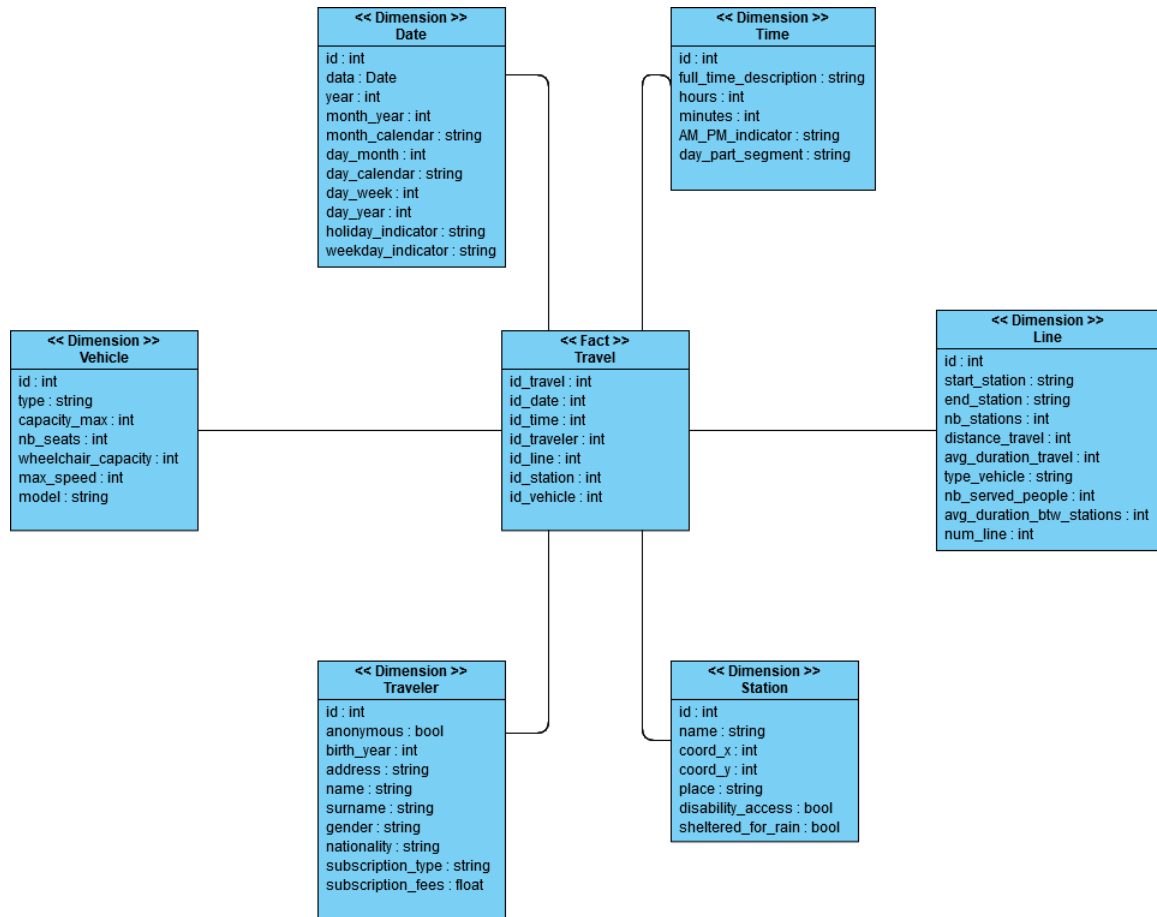


FIGURE 2.1 – modèle en étoile de l'action « voyages »

La table des voyages n'admet aucune mesure ; on se contentera d'utiliser les informations fournies par les dimensions qui suffiront largement pour analyser la fréquentation des différentes lignes et véhicules correspondant.

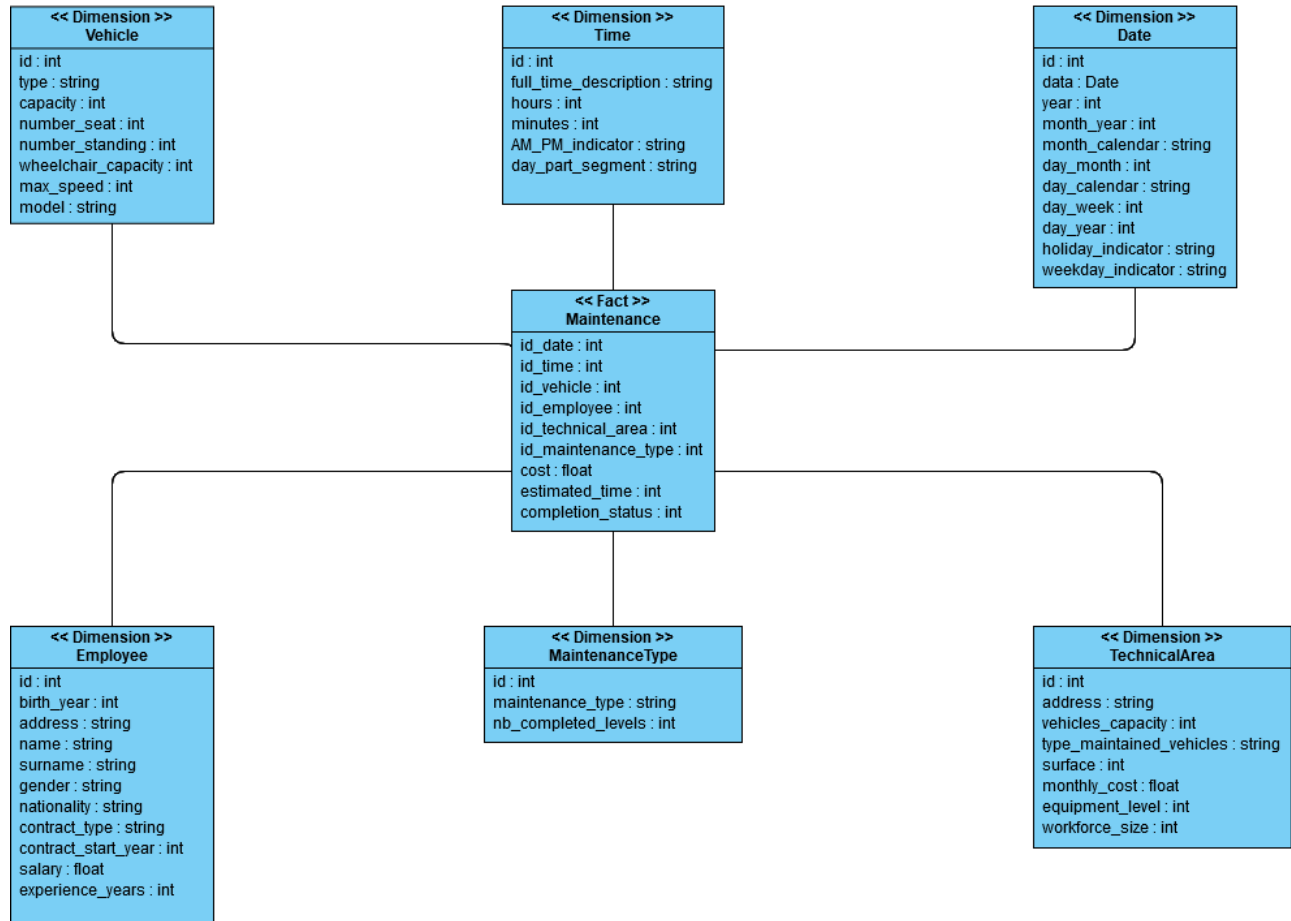


FIGURE 2.2 – modèle en étoile de l'action « maintenance »

Les mesures de la table des maintenances sont :

- **maintenance_duration** : mesure semi-additive désignant la durée de maintenance d'un véhicule
- **completion_status** : mesure non additive désignant l'état d'une opération de maintenance pour un véhicule donné

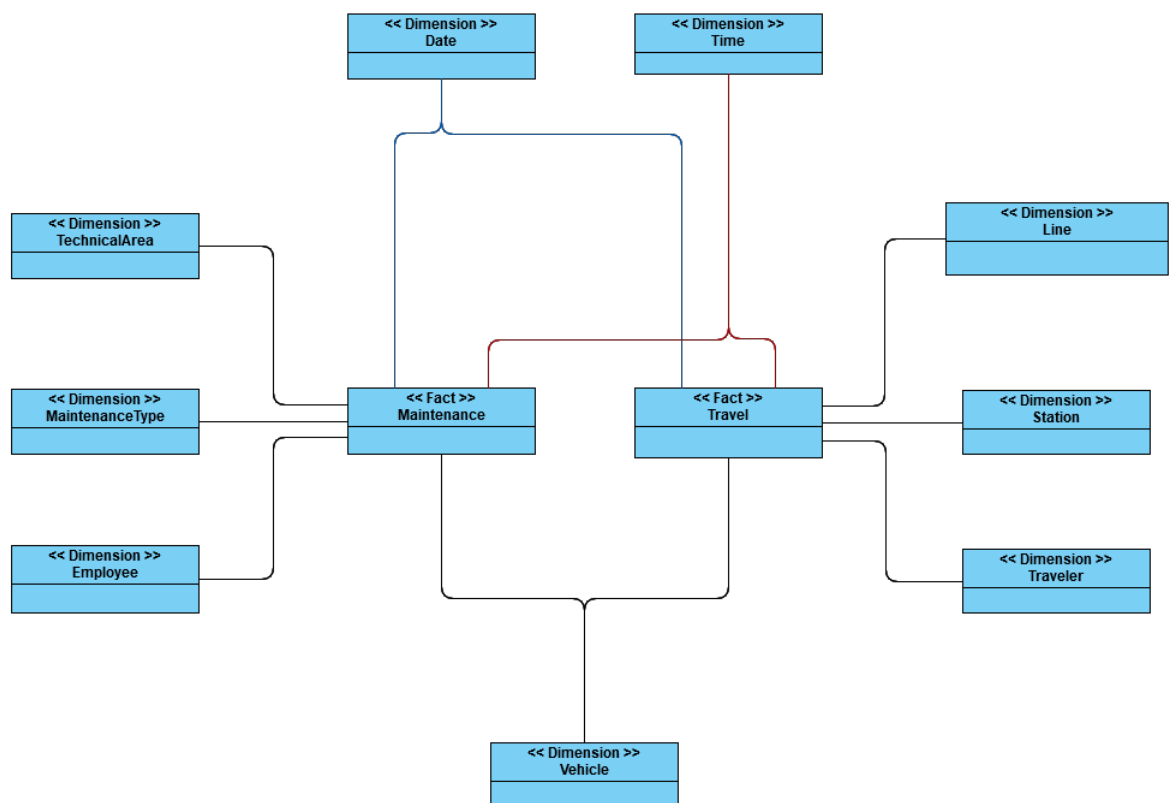


FIGURE 2.3 – le *data warehouse* résultant

Remarques

- l'attribut **id_travel** de la table **Travel** est la clé primaire utilisée pour identifier un voyage (*dimension dégénérée*).
- l'attribut **nb_served_people** de la table **Line** désigne le nombre de passagers desservis par la ligne.
- l'attribut **place** de la table **Station** désigne l'endroit où se trouve la station (avenue *X*, rue *Y*, ...).
- l'attribut **wheelchair_capacity** de la table **Vehicle** désigne la capacité théorique maximale de personnes handicapées et de leurs fauteuils roulants.
- la table **Traveler** est une dimension qui contient deux dimensions corrélées (les abonnés et le voyageur anonyme non abonné utilisant un ticket) :
 1. si le voyageur est **abonné**, alors on traite le tuple correspondant en tant qu'un **voyageur concret** dont les informations sont à notre disposition.
 2. sinon, tous les **voyageurs non abonnés** seront représentés par un seul tuple.
 3. cette décision de corrélation est utilisée pour éviter la normalisation et l'introduction d'une superclasse abstraite étendue par les classes désignant les voyageurs abonnés et non abonnés.
 4. nous utilisons ainsi l'attribut **anonymous** afin de distinguer les deux types de voyageurs. En effet, **anonymous** valera *true* quand le voyageur est non abonné, sinon il valera *true*.
 5. le tuple du **voyageur non abonné** contiendra ainsi des **valeurs nulles** pour les attributs décrivant un **voyageur abonné**.
- l'attribut **equipment_level** de la table **TechnicalArea** désigne le niveau de matériaux disponibles au local et peut prendre une valeur entre 1 (*pas assez équipé*) et 5 (*très bien équipé*).

Chapitre 3

Implémentation

Chapitre 4

Conclusion