HTTP Analyser

MOREAU Benjamin

Novembre 2015

Table des matières

A) Introduction

Lorsque nous naviguons sur le web, la majorité des sites que nous consultons envoie des données à un ou plusieurs sites tiers à l'aide de requêtes HTTP :*Hypertext Transfer Protocol*. Il est possible d'observer ces échanges à l'aide d'outils spécifiques appelés *Sniffers* mais ce type de logiciels ne permet souvent pas de conserver ces données sur le long terme, et encore moins d'avoir des informations spécifiques sur les sites qui émettent et reçoivent les données.

Mon objectif est donc de concevoir un entrepôt de donnée sous *MYSQL* permettant de stocker l'ensemble des requêtes effectuées sur un navigateur. Cette Base de données devra être capable de mettre en relation les requêtes avec une date, un navigateur internet et surtout des informations sur les sites. Un tel entrepôt peux servir à faire des statistiques intéressantes sur les sites ou pays récoltant le plus de données sur le web.

Dans ce dossier, je vais tout d'abord expliquer de quelle façon j'ai récolté et transformé mes données, puis, dans un second temps, je présenterai le schéma et l'optimisation de la base de données. Enfin, J'effectuerai quelques requêtes *SQL* pour illustrer le fonctionnement de mon entrepôt.

B) Collecte et transformation des données

Afin de récolter l'ensemble des requêtes HTTP lors de mes navigations, j'utilise le plugin *lightbeam* pour *Mozilla Firefox*. L'export des données se fait en *CSV* où chaque ligne représente une requête avec : la date, le nom du site source de la requête et celui d'arrivée, et d'autres informations qui seront présentes dans la table des faits.

L'objectif de cet entrepôt est aussi d'avoir des informations précises sur les noms de domaine présents dans la table des faits. Pour cela, j'ai développé, à l'aide du langage *AutoIt*, un automate ou *script*, récoltant des informations pour chaque site contenu dans mon fichier *CSV*. Ce dernier s'exécute quotidiennement et effectue une requête sur `https://who.is` pour chaque nouveau site présent. Les informations sont alors récoltées dans la page *HTML* à l'aide d'expressions régulières.

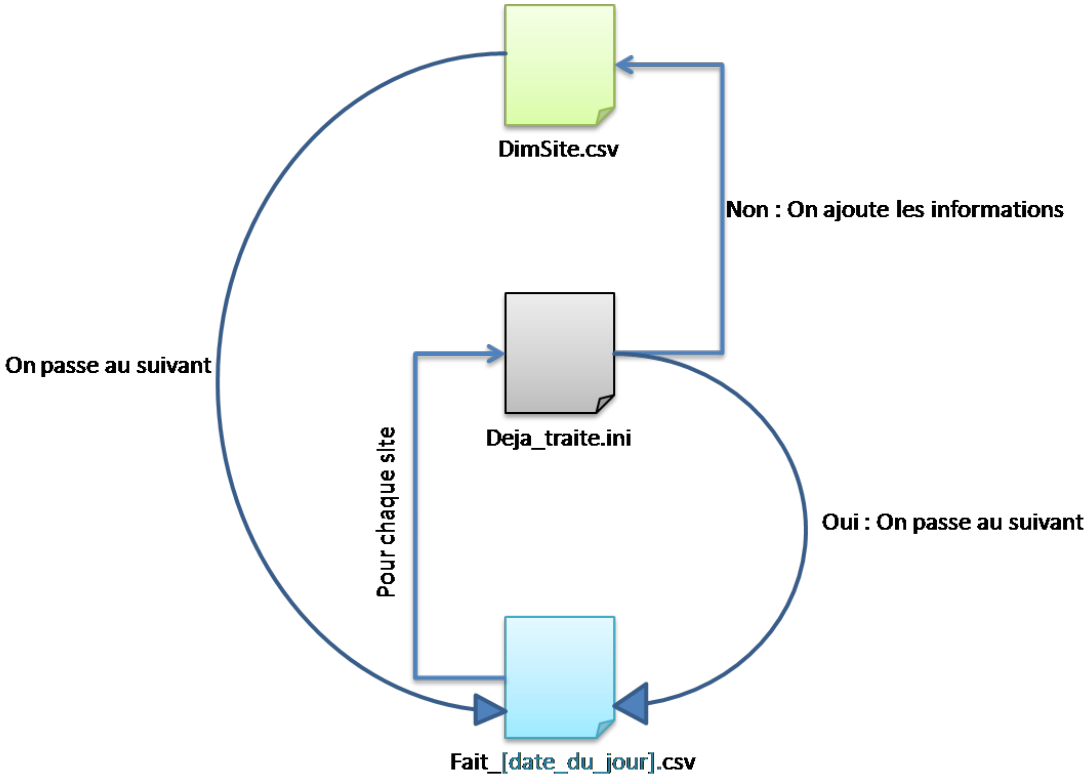

FIGURE 1 – Fonctionnement de l'automate

Ces données sont ensuite envoyées dans la table *DimSite* de mon entrepôt.

L'objectif de cet entrepôt est d'analyser le nombre de requêtes reçues ou envoyées par ces sites. La granularité de la table de fait est donc l'ensemble des requêtes envoyées ou reçues. Il est possible que plusieurs requêtes identiques soient faites le même jour. Il faut donc que la table des faits est un attribut clé identifiant les requêtes, autorisant donc les doublons.

C) Schéma de l'entrepôt

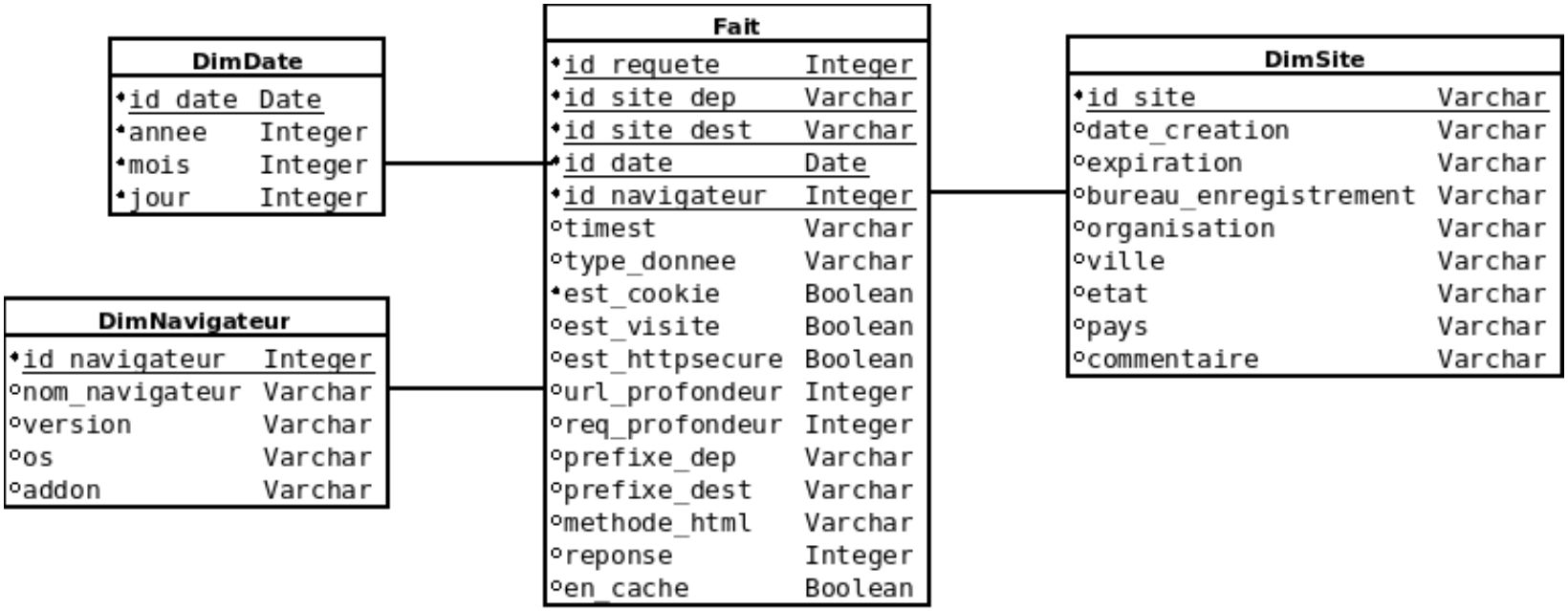

FIGURE 2 – Schéma de l'entrepôt de données

Afin d'optimiser les jointures entre la table *Fait* et *DimSite*, les attributs *id_site_dep* et *id_site_dest* on été indexés.

Notons qu'il aurait été possible d'agréger les requêtes dans la table des faits en supprimant l'attribut *id_requete* et en rajoutant un attribut *nb_req* représentant le nombre de requêtes identiques effectuées par jour.

D) Requêtes OLAP

Dans le but de réaliser des statistiques sur mon entrepôt, les requêtes suivantes ont été écrites :

```
-- nombre de requetes par jour, mois et annee avec sous totaux

SELECT annee, mois, jour, COUNT(*) AS nb_req
FROM Fait NATURAL JOIN DimDate
GROUP BY mois, jour WITH ROLLUP;
```

```
/* nombre de requetes envoyees vers des tiers depuis le site Facebook
 par jour, mois et total avec sous-totaux: */

SELECT annee, mois, jour, COUNT(*) AS nb_req_facebook_vers_tiers
FROM Fait NATURAL JOIN DimDate
WHERE (Fait.id_site_dep = 'facebook') AND (Fait.id_site_dest != Fait.id_site_dep)
GROUP BY mois, jour WITH ROLLUP;
```

```
-- nombre de requetes envoyees vers chaque pays, regions et ville:

SELECT pays, etat, ville, COUNT(*) AS nb_req_recu
FROM Fait AS F JOIN DimSite AS S ON F.id_site_dest = S.id_site
GROUP BY pays, etat, ville WITH ROLLUP
```

```
-- classement des sites, non visites, recevant le plus de requetes:
-- Fonction RANK() OVER() non disponible en MYSQL

SELECT id_site, COUNT(*) AS nb_data_recu, bureau_enregistrement, organisation
FROM Fait AS F JOIN DimSite AS S ON F.id_site_dest = S.id_site
WHERE (F.id_site_dest != F.id_site_dep)
GROUP BY id_site
ORDER BY nb_data_recu DESC
LIMIT 20
```

```
-- classement des sites envoyant le plus de requetes vers des sites tiers:

SELECT id_site, COUNT(*) AS nb_data_envoy, bureau_enregistrement, organisation
FROM Fait AS F JOIN DimSite AS S ON F.id_site_dep = S.id_site
WHERE (F.id_site_dest != F.id_site_dep)
GROUP BY id_site
ORDER BY nb_data_envoy DESC
LIMIT 3
```

```
-- classement des pays qui recoivent le plus de requetes:

SELECT pays, COUNT(*) AS nb_data_recu
FROM Fait AS F JOIN DimSite AS S ON ((F.id_site_dest = S.id_site)
AND (F.id_site_dest != F.id_site_dep))
WHERE (pays != 'NULL')
```

```
7 GROUP BY pays
8 ORDER BY nb_data_recu DESC
9 LIMIT 10
```

E) Conclusion

Cet entrepôt permet donc de stocker et d'étudier l'ensemble des requêtes effectuées entre les sites web visités. A l'heure actuelle, seul les dimensions *DimDate* et *DimSite* sont exploitées mais la dimension *DimNavigateur* pourrait elle aussi donner des résultats intéressants notamment l'attribut *addon* permettant de comparer l'efficacité de différents logiciels empêchant soi-disant ces échanges, non volontaires de la part de l'utilisateur. Il aurait aussi été intéressant de faire des analyses volumétriques des requêtes en ajoutant un attribut de taille(Ko) dans la table principale.