Carnet de bord :

07/03 : découverte du projet et mise en route (power BI + regarder le code parrallele coordinate : <http://bl.ocks.org/jasondavies/1341281>)

08/03 : chercher à comprendre le code du parrallele coordinate et début de code pour rendre le code compatible à n’importe quel fichier csv

09/03 : suite de la compatibilité csv -> objectif atteint quand le séparateur de colonne est la virgule (Excel définit la virgule ou le double point selon les paramètres)

Début de recherche sur le drag & drop pour importer un fichier.

10/03 : Drag n drop OK, toujours des problèmes de format de fichier… (si le texte comprend des virgules par ex) : code du drag n drop : <http://www.html5rocks.com/en/tutorials/file/dndfiles/>

|  |  |  |
| --- | --- | --- |
| Library | jqGrid | jsGrid |
| Sort | Par une ou plusieur colone  Less/greater/equal sont possibles | Une seule ligne a la fois |
| Filter | Autofilter par liste, par mot et par date | Autofilter par liste, par mot |
| Modification du tableau | Column reordering by D n’ D  Modif des données possible directement sur le tableau | Ordre des ignes et données modifiable directement sur le tableau |
| Hierarchy | oui | ? |
| Data | From json, xml, javascript Array |  |
| Selection | Possibilité de selectionner des lignes |  |
| Grouping | Oui |  |
| Code |  | Peu de démo  Très bien documenté |

11/03 : telechargement de jqGrid et affichage du tableau avec autofilter sur chaque colone : Reste a mettre en place l’interaction graph/tableau

Option envisagée -> la selection sur le graph donne en sortie les coordonnées min et max pour l’élément selectionné, il faudrait relier ces valeurs a un filtre sur le tableau… (implique la mise en place d’un filtre « greater than » et « lower than »).

14/03 : J’ai finalement réussi a extraire les données de la barre de recherche du tableau (directement dans la creation du tableau avec l’event loadComplete). Il me faut maintenant importer ces données dans la fct brush qui definit la zone ou les traits sont affichés en bleu (ou pas). Elle test les differents criteres, jusqu’au dernier, où toutes les options sont traitées (label1/option1 : OK -> label2, sinon label1/option2…etc…). La fonction renvoie true si on affiche, false sinon.

15/03 : 

Les données de recherches peuvent etre extraites de n’importe où, et sont donc exploitable. J’ai réussi à formater le filtre de « brush » et du tableau pour qu’ils soient compatible et associable. Une reche dans le tableau et donc maintenant visible sur le graph mais ne se declanche pas toute seule : la fonction brush qui tri le graph selon les infos qu’on lui donne ne se déclanche qu’au clic… (pour le moment).

-J’envisage donc 2 solutions : simuler un clic par programmation (mais me semble compliqué)

-relancer la fontion brush apres une recherche (cela provoque pour le moment une erreur car tous les parametre de la fonction ne sont pas encore configuré.

16/03 : Avant : 2 fonction aui s'envoyaient des filtres, qu'il fallait donc actualiser en permanance. Cela marchait jusqu'a la derniere etape, ou le tableau appelait le graph, qui lui meme appelait le tableau...etc. J'ai donc créé une fonction intermediaire qui prend en compte a source du changement, et réactualise tout le reste.

Pour le moment : ok dans le sens tableau ->graph, reste a faire l'autre sens.

17/03 : Creation d’une fonction Gathered et d’une fonction reload qui ont pour but de faire l’intermédiaire entre les differents elements du dashboard (tableau et graph pour le moment). Pas encore fonctionnel.

Prise en compte des fichier .json en entrée.

18/03 : Interaction tableau/graph ok pour les valeur num

Bugs rencontrés :

-Insertion dans le FilterTable des données du brush apres chaque reload du tableau (tout temps) ce qui créait des contraintes en permanace (et donc affichage nul car trop de contrainte) : differenciation des filtres insérés par l’utiliateur (option onSearch) des filtres envoyés par le brush

-non relache des filtres apres un brush : les données etaient toutes sauvegardées dans le FilterTable ,quand on refaisait une recherche apres un brush : a la relache du brush, les filtres etaient toujours dans FilterTable. Solution : fonction de differenciation qui retire de FilterTable les filtres provenant du brush.

21/03 : <http://bl.ocks.org/syntagmatic/3150059> <- parrallel coordinate pas trop mal !

Base de données importante : ca ram.

Utilisation d’un plugin de jqgrid pour un column choser sympa (qui permet de supprimer des lignes du tableau)

Import de csv avec des parser « ; » OK (meme si aeroport.csv pose toujours pb)

Une couleur par ligne OK : fonction couleur qui definit aléatoirement un code rgb(modification possible pour rendre ca plus harmonieux : limiter une couleur dans un intervalle restreint)

22/03 : fin des interactions tableau/graph a deux element : le tableau et le graph se repondent !!!

Ajout du nombre de ligne encore active et fonction resizable sur le tableau.

Objectifs 1 :

~~- génération de la datagrid avec autofilter~~

~~- faire interagir le graphique et la datagrid~~

~~- régler l'opacité des lignes non selectionnées~~

~~- mettre des couleurs différentes sur les lignes selectionnées~~

-  insertion d'un compteur

~~- insertion d'un controle pour le upload de fichier~~

~~- couleur par ligne (graph + tab)~~

~~- enlever des colonnes (graph)~~

~~-column choser (tab)~~

Passage aux objectifs 2 : adaptation d’un bar chart existant pour l’integrer au code :

-nom des variables

-type des variables

-place dans le dashboard (au debut : superposition des deux graphs… --‘)

Le bar chart est desormais integré au code mais n’est pas encore adabtable aux diverse entrées (que cars pour le moment) et doit encore etre mis en interaction avec le reste du dashbord.

Objectifs 2 :

~~-ajouter la couleur de la ligne dans le tableau~~

~~-ajouter un autre graphique pour une interaction a + que 2~~

-commencer à réfléchir à un bookmark qui lie l’utilisateur, le fichier et les filtres (filtre tableau, graph Sert colonnes cachées) => dans l’optique d’un webservice, lié directement avec l’algo en python, il serait intéressant de voir si on peut enregistrer ces données coté serveur, de facon light (ce qui implique une interaction client-serveur//ajax//php)

23/03 : le Bar Chart est integré au graphique et interactif avec le tableau et le parallel coordinate. (unilateral dans le sens « vers le BarChart » -> par de filtre sur le BC)

Pour cela j’ai dû créer une fction filtre qui affiche ou non les barres (taille de la barre normale ou 0). Réflexion faite, j’aurais aussi pu modifier la couleur (normal ou blanc [voir gris clair comme sur le // coordinate]). Le changement n’est pas fait parce que le format de données au niveau des couleurs n’est pas le meme qu’au niveau de la taille de la barre (mais serait faisable sans trop de soucis).

Problème rencontré : au dela du svg que je ne connais pas encore tres bien, il m’a fallu reloader le BC a chaque chgmt du filtre. Pour cela, j’utilise la fction reload (déjà créée pour l’interaction a 2) dans lequel je supprime le BC (a l’aide des classes bien choisies et non pas en supprimant toutes les classes « g » comme fait pour le reset ) et le recrée.

J’ai aussi commencer la fonction permettant de lier la couleur des lignes du // coordinate au tableau. Pour cela j’ai du modifier la structure du programme en rendent le tableau plus accessible (avant il était coincé dans une macro fonction). J’ai aussi sauver eles couleurs dans une variable objet comprenant la couleur et les data associées. Reste a coder la fonction permettant de lier la couleur aux data et de l’afficher sur le tableau.

24/03 : L’affichage des couleurs sur le tableau est assez simple mais la reconnaissance donnée tableau <-> donnée //coordinate a posé probleme car le programme ajoute un element a l’objet lors de la sauvegarde (directement au moment du choix des couleurs). J’ai donc du passer par une sauvegarde au format « string » qui fonctionne. Le tableau comprend maintenant une case avec la couleur du trait !

Debut de dechiffrage de code pour le zoomable circle. Le code appelle des données déjà hierarchisée (par le mot clef « children ») : <https://bl.ocks.org/mbostock/7607535>

Idem avec le code de zoomable treemap qui n’utilise pas de base de données hiérarchisée, pour voir comment est lue et ordonnée cette base de données : <http://bl.ocks.org/ganeshv/6a8e9ada3ab7f2d88022>

29/03 : Tentative de comprehension du code ZC et TM afin de comprendre la hierarchisation des données. A defaut de comprendre la globalité, j’ai reussi a extraire la partie « hierarchisation » du code du TM pour pouvoir y integrer le code du ZC. Le probleme est que j’ai maintenant des données hierarchisées, mais pour le ZC, elles doivent etre comprises dans un « parent » qui englobe le tout (ca ca va) avec des données adequates (la ca pose probleme) afin de dessinner le cercle global (le contour). Tentative en hard codant des valeurs : le code les refuse et me donne NaN. Tentative en ajoutant un niveau en plus (definit comme parent) : probleme de la valeur qu’on y met (meme si ce n’est pas fondamental) mais surtout, cela fonctionne plus ou moins jusqu’au moment ou je passe dans ma fonction qui trace le ZC, et ou il me renvoie a nouveau NaN… (alors qu’avant il m’affiche des valeurs…)

30/3 : Continuation sur les hierarchies. A defaut de créer une hierarchie directement, je la créé dans une variable qui servira de reference. Cette variable est ensuite uilisée par le programme comme lors d’un appel par url (ce qui était fait precedemment). Cela me permet donc de me servir d’un fichier plat pour créer zoomable circle et bubble. Les mots clef key et value sont encore a supprimer du programme.

Concernant la datagrid, utilisation du model : <http://www.guriddo.net/demo/treegridjs/> ou <http://www.guriddo.net/demo/guriddojs/grouping/multiple_columns_header/index.html>, reste a comprendre comment est definie la hierarchie (dans la bdd ici) et voir comment l’adapter a mes données d’entrée.