

Le Concept d'Echange Matriciel

Dans un planning d'organisation, l'affectation de ressources peut obéir à une répartition matricielle.

Ces ressources peuvent correspondre à des personnes affectées sur des sites, des voitures affectées à des employés, du matériel affecté dans des salles,

Pour des questions didactiques, prenons l'exemple d'affectation de personnes.

A un instant t, la matrice peut être décrite comme telle:

t	site 1	site 2	site 3	site 4	site 5	...
id 1		x				
id 2	x					
id 3			x			
id 4			x			
id 5				x		
id 6	x					
id 7				x		
....						

Forme croisée

t	site 1	site 2	site 3	site 4	site 5	...
id 1	0	1	0	0	0	
id 2	1	0	0	0	0	
id 3	0	0	1	0	0	
id 4	0	0	1	0	0	
id 5	0	0	0	1	0	
id 6	1	0	0	0	0	
id 7	0	0	0	1	0	
....						

Forme booléenne

Ces deux représentations indiquent par exemple, qu'à l'instant t, id 1 est affecté sur le site 2.

Tout serait simple si la forme d'un tel tableau était conservé. Il suffirait d'une permutation booléenne pour réaffecter les personnes.

Hors, sur une interface séduisante, il n'est pas envisageable d'utiliser une telle forme pour représenter les affectations. Ainsi, plusieurs ressources vont être rassemblées, dans une case, pour chacune des affectations. La description de la matrice prendra alors cette forme:

t	site 1	site 2	site 3	site 4	site 5	...
Ressources	id 2	id 1	id 3	id 5		
	id 6		id 4	id 7		

Il faut donc, maintenant, trouver le moyen de permuter une ressource d'une affectation à une autre.

LA PROBLEMATIQUE :

Comment sur une interface graphique créée avec FileMaker, permuter facilement une personne d'un site à un autre ?

Les Solutions d'Interface FileMaker

- La zone d'édition

Méthode inutilisable car trop lente.

- Le menu local ou la liste déroulante

Possible, mais méthode un peu rébarbative quand il y a beaucoup de ressources à gérer. Il faut effacer la personne sur le site 1 puis la réaffecter sur le site 2.

- Le Glisser-Déposer

Excellente méthode, pour les petites interfaces, mais elle est en défaut lorsque l'interface est large. En effet, le fait de déplacer un item sur la largeur de l'interface, tout en maintenant le bouton de souris enfoncé, est fatigant pour la main.

De plus, il y a un risque de relâcher l'item, par inadvertance, sur une mauvaise case.

La pratique du Glisser-Déposer sur un iPad peut également s'avérer sportive, voire impossible.

- L'Echange Matriciel

J'ai imaginé cette technique pour qu'en 2 clics, une personne puisse être réaffectée d'un site à un autre. Elle est utilisable pour de grandes interfaces avec de nombreuses cases d'affectation. FileMaker 12, grâce aux apports des fonctions SQL, a apporté une nette simplification, par rapport à cette technique que j'ai développée sur Filemaker 11.

Conception de l'Echange Matriciel

Le fichier joint à cet article reprend les éléments techniques décrits ci-dessous.

Plusieurs tables sont nécessaires, mais c'est sur les tables Matrice et Calendar, que va se faire le travail de l'échange matriciel.

Table Matrice

La matrice (forme croisée) est codée par une table dédiée.

Chaque enregistrement code l'intersection ligne-colonne où se situe le x ou le 1 de la représentation matricielle:

1 enregistrement = 1 instant t --- 1 personne --- 1 site

3 rubriques sont nécessaires au codage de la matrice dans FileMaker 12:

- **Date** : code l'instant t de la matrice
- **Site** : identifie la colonne
- **Personne** : identifie la ligne

Dans le version FM11, j'utilisais une rubrique supplémentaire. Vous la trouverez dans le fichier en référence:

- **Item** : permet le codage de l'intersection ligne-colonne sous la forme **site # personne** . C'est une rubrique calculée Indexée. Cette rubrique **Item** servait à l'affichage de la matrice et au travail de l'échange matriciel. Grâce à 2 fonctions personnalisées simples, les données de la matrice étaient extraites pour leurs affichages (*Liste.filter#* extrait une liste

des rubriques **Item** de chaque site, ensuite *#extrait* extrait pour chaque **Item**, la rubrique personne après le #).

Table Personne

Cette table permet de répertorier les ressources. 1 rubrique **Nom** est suffisante.
Cette table pourrait aussi être dénommée Matériel, Voiture, Elèves, ...

Table Sites

Cette table permet de répertorier les sites d'affectation des ressources. 1 rubrique **Nom** est suffisante.

Table Calendar

Cette table peut être assimilée à une table de jointure interne. Elle sert de moteur pour l'affichage de la matrice.

Une rubrique **Date** est créée dans la table Calendar. Cette Date va servir de clé de liaison pour chacune des dates de la table Matrice.

Cette table possède également plusieurs rubriques multivaluées calculées non mémorisées permettant d'extraire et d'afficher les données de la matrice. Ces rubriques multivaluées sont dénommées **site1 MV**, **site2 MV**, ... Dans les noms de ces rubriques site, seul le chiffre compte. Il sert de référence pour identifier les sites dans leur liste.

La GRANDE innovation de FileMaker 12 est, dans ce cas, l'apport de la fonction ExecuterSQL (Requête SQL; ...)

Grâce à cette fonction, il est extrêmement simple d'aller extraire les données matricielles.

La syntaxe des requêtes prend alors la forme:

```
ExecuterSQL (
    "SELECT personne FROM Matrice WHERE site =? AND dat =? "; "" ; ¶; sites[1] ; date
)
```

La puissance de cette fonction rend même le lien entre Calendar et Matrice sur la rubrique Date inutile. Seul le script décrit ci-dessous a besoin de ce lien.

note: pour des raisons de syntaxe SQL, il est obligatoire que la rubrique Date de la table Matrice soit renommée dat (car Date est une fonction native sql).

L'affichage des données se fait sous la forme de rubriques Multivaluées. Leur utilisation est obligatoire, car à la différence d'un affichage en Liste, c'est le seul moyen d'identifier un contenu de rubrique sous un clic de souris.

Table Interface

Cette table sert juste à piloter la table Calendar par une rubrique globale DATE.FILTER

Le Modèle Interface

C'est sur ce modèle qu'est réalisé l'échange matriciel. L'occurrence de table Calendar est affiché dans un table externe. L'utilisation des rubriques multivaluées permet une grande souplesse d'affichage. Il peut être décider qu'un site N n'aura que 3 cases affichées , car ne pouvant accueillir que 3 ressources.

Le Script Echange Matriciel

Ce script est la clé du mécanisme de l'échange matriciel.

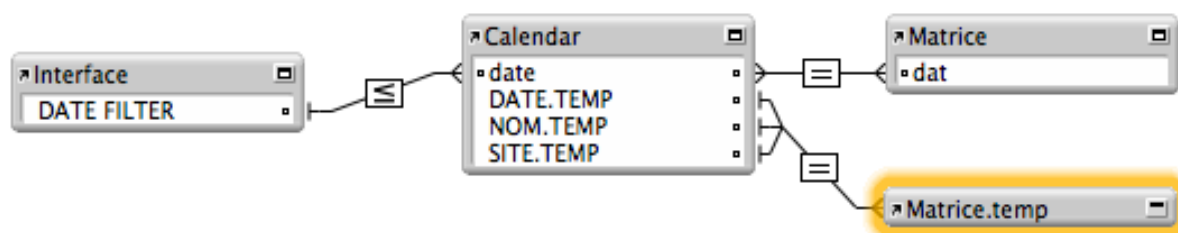
Il est appelé par le déclencheur de script **SurEntréeObjet**

Ce simple clic sur une rubrique identifie l'intersection de la matrice par le date, la ressource, le site.

Ces données sont stockées temporairement dans des rubriques globales de la table Calendar **DATE.TEMP, SITE.TEMP, NOM.TEMP**

La sélection est identifiée par formatage conditionnel.

Les rubriques globales permettent d'isoler l'enregistrement matriciel sélectionné grâce à l'occurrence de table ci-dessous:



Le script opère ensuite "sa magie" lors du 2ème clic.

Si le clic de souris se fait sur une case vide ayant la même date que la sélection, le script ré-affecte le nouveau site de l'enregistrement isolé dans l'occurrence **Matrice.temp**, et l'Echange Matriciel se produit !

Les rubriques globales sont ensuite effacées.

Les questions possibles:

Comment ajouter une date à la matrice?

Un script du fichier de démonstration fait ce travail:

- 1- ajout d'un enregistrement de la table Calendar avec une nouvelle date
- 2- avec cette même date, boucle de création d'enregistrements dans la table Matrice pour chacune des ressources. Dans cette boucle, il est IMPERATIF, d'affecter pour chacun des enregistrements, un site par défaut. Si cela n'est pas fait, alors les ressources ne seront pas affichées dans la matrice, et donc non pilotables.

Comment ajouter une ressource ?

Un script, facilement programmable, permet ce travail. Il suffit de créer des enregistrements dans la table matrice avec le nom de la ressource, les dates souhaitées, et, impérativement, un site par défaut.

Quid de la puissance de calcul et de la vitesse d'affichage ?

Dans le fichier de démonstration, le même modèle est utilisé pour la version desktop et iPad. Ce modèle affiche 118 cases cliquables. Le rendu nécessite de nombreux calculs. Les performances sont donc en retrait sur l'iPad.

Afin d'optimiser le fonctionnement sur iPad, il est donc important de développer la matrice en fonction des ressources à affecter. Ceci concerne le nombre de répétition des calculs multivaluées de la table Calendar, ainsi que le nombre de répétitions des rubriques multivaluées affichées sur le modèle.