

Compléments sur la Rose des Vents

Paramétrer la petite rose et les critères complémentaires

1

La Rose des Vents repose sur un couple de fichiers :

- Le fichier **radar.xml**
- le fichier **PetiteRoue.js**

Tous deux situés dans le répertoire racine de GaïaMundi

Le fichier PetiteRoue.js contient le paramétrage des branches de la petite roue de la rose des vents et le paramétrage des critères de sélection complémentaire

Langues de version

Le fichier petite roue peut exister en plusieurs langues. La version du fichier de la langue en court porte le nom **PetiteRoue.js**

Les autres versions portent le nom : **PetiteRoue langue.js**

Pour changer de langue, il suffit de renommer les fichiers.

Paramétrage des critères de pré sélection

Le fichier PetiteRoue.js comprend deux sections.

1. La première section concerne le paramétrage de la petite rose
2. La deuxième section concerne le paramétrage des filtres complémentaires

The image shows two parts of a configuration interface. The top part, titled 'Focus', displays a circular diagram with 'All' in the center and various criteria radiating outwards: population, nationalité, scolarisé, diplôme, actifs, occupés, emploi, navette, travail-domicile, études-dor, entreprise, salaire, établissements, revenu, équipement, famille, ménage, logement, et immigré. Below this diagram, it says 'nd_filtre : All'. The bottom part, titled 'filtres complémentaires', contains a form for additional filtering. It includes a section for 'Lieu de recensement' with radio buttons for 'résidence', 'travail', and 'Indifférent'. Below that is a section for 'Sur l'emploi et l'activité' with checkboxes for 'CSP', 'Secteurs', 'CDI CDD ...', 'Forme contrat', 'Salarié/non', 'Indépendants', 'âges', and 'Sexe'. At the bottom, there is a text input field labeled 'Règles pour la recherche', an 'envoyer' button, and an 'annuler filtre complémentaire' button.

1 - Première section : paramétrage de la Petite Rose

```
var NumeroPlateforme="altercarto.fr : Visionneuse"
```

```
//-----Module Petite Rose-----
```

```
libelaxeB[libelaxeB.length]=new Array("population","*","*", "fertilityrate","Lifeexpectancy")
axescoteB[axescoteB.length]="1"
coulB[coulB.length]="green"
```

```
libelaxeB[libelaxeB.length]=new Array("health","Lifeexpectancy","*","UNHIV","medical","immunization")
axescoteB[axescoteB.length]="1"
coulB[coulB.length]="green"
```

Chaque branche de la petite rose est représentée par un bloc de trois lignes comme ceux présentés ci-dessus.

La première ligne comprend les critères de définition du champ correspondant à cette branche de la petite rose. La fonction logique appliquées à la collection de critères est le ou. Pour signifier « sauf tel critère », un point d'exclamation ! précède le critère,

Certains de ces critères sont apparent, d'autres non, selon la règle suivante :

- Les trois premiers critères compris dans l'Array() sont visibles sauf s'il sont remplacés par une asterix *

On peut donc définir un thème par une série de critères dont un à trois sont visibles et les autres invisibles.

Les deux autres lignes sont inactives dans cette version. Mettre les valeurs « 1 » dans la deuxième ligne et « green » dans la troisième.

Pour que la présélection soit efficace et pertinente, chaque petite Rose doit être paramétrée en fonction du contenu effectif des fichiers contenus dans la plateforme

La construction d'une batterie de critères pour définir un champ est un exercice méticuleux. Il vaut parfois mieux faire des compromis et avoir des redondances plutôt que de trop fermer la sélection et risquer d'occulter des fichiers.

2 - Deuxième section : paramétrage des filtres complémentaires

La deuxième section comporte 3 parties :

- La fabrication du tableau** de critères complémentaires
- La fonction annulation** de tous les critères complémentaires
- La fonction exécution** des critères complémentaires sélectionnés

a. Fabrication du tableau de critères complémentaires

Le code est composé d'un entête de tableau et d'une fin de tableau. Entre les deux se situe l'espace d'insertion des codes et paramètres nécessaires à l'introduction des sélecteurs de critères (bouton radio et checkbox)

//-----SECTION II-----module critères complémentaires-----

```
var TableCompSelect=""
TableCompSelect+='<table style="border:1px dotted black;background-
color:transparent;width:310px;height:300px;font-size:10px;font-family:arial">'
TableCompSelect+='<tr><td style="text-align:center;background-color:#E6E6E6">'
TableCompSelect+='<big><b>filtres complémentaires</b></big><br/> <i>(Attention : chaque bouton
ajoute une condition logique "et" au filtre focus)</i>'
TableCompSelect+='<br/>Panier : '
TableCompSelect+='<input onclick=";SuiteCheck()" type="radio" name="panier"
id="pa_1"></input>oui'
TableCompSelect+='<input onclick="SuiteCheck()" type="radio" name="panier" id="pa_2"
value="checked"></input>non'
TableCompSelect+='</td></tr><tr><td style="text-align:center">'
//***** Début des paramètres insérés par l'utilisateur *****

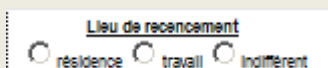
(...)

//***** fin des paramètres insérés par l'utilisateur *****
TableCompSelect+='</td></tr><tr><td style="text-align:center;background-color:#E6E6E6">'
TableCompSelect+='<br/>'
TableCompSelect+='<span style="; color: rgb(153, 153, 255);" title="Insérez les mots clés séparés par
un espace. Pour former une chaîne de mots, reliez les avec le tiret bas \'_\'. Pour indiquer \'sauf\' :
placez \'!\' devant le mot clé ou la chaîne_de_mots" ><b><u>Règles pour la
recherche</u></b></span><br/>'
TableCompSelect+='<textarea name="T1" id="T1_1" style="width:90%;height:40px"></textarea>'
TableCompSelect+='<input type="button" onclick="SuiteCheck()" value="envoyer"></input><br/>'
TableCompSelect+='<input type="button" onclick="annuleCheck()" value="annuler filtre
complémentaire"></input>'
TableCompSelect+='<br/><br/><br/><br/><br/>'
TableCompSelect+='</td></tr>'
TableCompSelect+='</table>'
```

On introduit les critères selon deux modalités :

A - question à choix multiple et à réponse unique.

On utilise pour cela les boutons radios
comme dans l'exemple suivant :



Les boutons radio d'une telle question ont tous la même valeur pour leur attribut name=

Ils ont en revanche un identifiant id= différent

Ce qui, dans cet exemple, donne le code suivant, en incluant le titre de la question :

```
TableCompSelect+='<b><u>Lieu de recensement</u></b><br/>'

TableCompSelect+='<input onclick=";SuiteCheck()" type="radio" name="F1" id="F1_1"></input>résidence'
TableCompSelect+='<input onclick="SuiteCheck()" type="radio" name="F1" id="F1_2"></input>travail'
TableCompSelect+='<input onclick="SuiteCheck()" type="radio" name="F1" id="F1_0"></input>indifférent'
```

B - question à choix multiple et à réponses multiples.

On utilise pour cela les checkbox comme dans l'exemple suivant :

Sur l'emploi et l'activité

☐ CSP ☐ Secteurs ☐ "CDI CDD ..." ☐ Forme contrat

☐ Salarié/non ☐ Indépendants ☐ âges ☐ Sexe

Dans ce mode, chaque checkbox a un attribut name= différent et un identifiant id= différent.

Ce qui, pour cet exemple, donne le code suivant, en incluant le titre de la question:

```
TableCompSelect+<=<b><u>Sur l'emploi et l'activité</u></b><br/>
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F2" id="F2_1"></input>CSP'
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F3" id="F3_1"></input>Secteurs'
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F4" id="F4_1"></input>CDI CDD'
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F5" id="F5_1"></input>Forme  
contrat<br/>
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F6" id="F6_1"></input>Salarié/non'
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F7" id="F7_1"></input>Indépendants'
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F8" id="F8_1"></input>âges'
```

```
TableCompSelect+<=<input onclick="SuiteCheck()" type="checkbox" name="F9" id="F9_1"></input>Sexe'
```

Séparateur de questions

Entre deux questions on introduit un séparateur qui introduit un liseré gris comme dans l'exemple ci-dessous où les deux questions précédentes ont été introduites entre les entêtes haut et bas du tableau

filtres complémentaires
(chaque critère ajoute une condition logique "et" au filtre focus)

Parler : ☐ oui ☒ non

Lieu de recensement
☐ résidence ☐ travail ☐ indifférent

Sur l'emploi et l'activité

☐ CSP ☐ Secteurs ☐ "CDI CDD ..." ☐ Forme contrat

☐ Salarié/non ☐ Indépendants ☐ âges ☐ Sexe

Règles pour la recherche

envoyer

annuler filtre complémentaire

(Suite page suivante)

b. Fonction annule les critères complémentaires

Elle se présente sous la forme suivante :

```
function annuleCheck(){
```

```
//*****Début insertion des critères *****
```

```
document.getElementById("F1_0").checked=false  
document.getElementById("F1_1").checked=false  
document.getElementById("F1_2").checked=false
```

Question 1 à réponse unique

```
document.getElementById("F2_1").checked=false  
document.getElementById("F3_1").checked=false  
document.getElementById("F4_1").checked=false  
document.getElementById("F5_1").checked=false  
document.getElementById("F6_1").checked=false  
document.getElementById("F7_1").checked=false  
document.getElementById("F8_1").checked=false  
document.getElementById("F9_1").checked=false
```

Question 2 à réponses multiples

```
//*****Début insertion des critères *****
```

```
document.getElementById("T1_1").value=""  
document.getElementById("T1_1").innerHTML=""  
document.getElementById("pa_2").checked=true // panier non  
document.getElementById("pa_1").checked=false  
SuiteCheck()  
}
```

Chaque ligne représente un bouton ou une check box.
Tous les boutons et checkbox sont représentés.

(Suite page suivante)

C. Fonction exécution des critères complémentaires sélectionnés

Elle se présente sous la forme suivante :

```
function FiltreCompET(){
var transit=""
motcleET=""
//*****Début insertion des critères*****

if(document.getElementById("F1_0").checked==true){}
if(document.getElementById("F1_1").checked==true){motcleET+="résiden (LR)**"}
if(document.getElementById("F1_2").checked==true){motcleET+="lieu_de_travail (LT)**"}

if(document.getElementById("F2_1").checked==true){motcleET+="cadres intermédiaire**"}
if(document.getElementById("F3_1").checked==true){motcleET+="industrie commerce**"}
if(document.getElementById("F4_1").checked==true){motcleET+="CDD déterminé**"}
if(document.getElementById("F5_1").checked==true){motcleET+="contrat**"}
if(document.getElementById("F6_1").checked==true){motcleET+="salarié**"}
if(document.getElementById("F7_1").checked==true){motcleET+="indépendant employeur**"}
if(document.getElementById("F8_1").checked==true){motcleET+="_ans**"}
if(document.getElementById("F9_1").checked==true){motcleET+="femme homme**"}

//*****Début insertion des critères*****

if(document.getElementById("pa_1").checked==true){}
if(document.getElementById("pa_2").checked==true){motcleET+="!panier**"}
transit=document.getElementById("T1_1").value
transit=transit.split(" ")

for(g=0;g<transit.length-1;g++){
motcleET+=transit[g]+"**"
}
motcleET+=transit[transit.length-1]
var lg=motcleET.length
//alert(motcleET.substring(motcleET.length-2,motcleET.length))
if(motcleET.substring(motcleET.length-2,motcleET.length)!="**"){
motcleET=motcleET.substring(0,motcleET.length-2)
}
}
```

Comme dans la fonction précédente, il y a une ligne pour chaque objet bouton radio ou checkbox.

On peut avoir des boutons sans critères {} comme le bouton « indifférent » dans l'exemple qui a pour seule fonction de décocher les deux autres boutons de la question à réponse unique.

Pour les autres boutons radio ainsi que pour les checkbox on a une exécution du type suivant :
{motcleET+="indépendant employeur**"}

C'est là que l'on inscrit les critères correspondant à chaque bouton ou Checkbox.

Ici, la syntaxe des critères est la suivante :

critère1 critère2 critère3 critèrex**

soit une liste des critères séparés par un espace avec ** en terminaison

ici « l'espace » correspond à la fonction logique « ou » tandis que « ** » correspond à une fonction logique « et » appliquée à la ligne de critères, ce qui équivaut à **(critère 1 ou critère2 ou...) et**

Notez que pour avoir le terme « ans » comme critère de repérage des âges, en français, on a mis « _ans », le caractère « _ » étant interprété ici comme un espace, de sorte que le mot artisans ne sera pas pointé par le moteur de recherche.

Comptes utilisateurs et accès public

Comme évoqué dans le manuel d'usage, il ya trois statuts d'utilisateur de la Rose des vents:

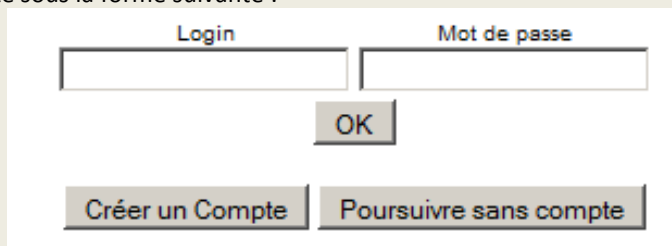
1. **Les adhérents de la plateforme** (c.a.d. ceux qui possèdent un mot de passe pour la plateforme). Le statut d'adhérent de la plateforme s'acquière par cooptation dans l'équipe qui anime la plateforme GaïaMundi. Pour que le mot de passe d'adhérent soit opératoire dans la rose des vents ; il doit entrer dans la Rose des Vents à partir de la plateforme GaïaMundi (Menu Edition-> Rose des Vents). Les registres de mots de passe utilisateur public et adhérent sont en effet indépendants. Les login et mots de passe adhérent ne sont pas inscrits automatiquement dans le registre des utilisateurs publics. Si un adhérent veut travailler sur la plateforme à partir de l'entrée publique (ajouter des données à un dossier carte ou créer un panier accessible à partir de la plateforme avec sont identifiant d'adhérent), il doit créer un compte public dont le login doit être le mot de passe d'adhérent (**login public = mot de passe adhérent**).
2. **Les titulaires d'un compte public sur la Rose des vents.** L'ouverture d'un compte public se réalise en ligne lorsque les animateurs de la plateforme ont fait le choix d'ouvrir cette possibilité en mettant en ligne l'interface d'ouverture des comptes publics. (Cf détail ci-après). Le compte public permet à son titulaire de créer des dossiers paniers qu'il peut ensuite exploiter avec la Rose des vents (Cf Manuel) sans toutefois pouvoir intervenir sur l'architecture de données de la plateforme. La frontière entre l'espace public et la plateforme est étanche pour les titulaires de comptes publics.
3. **Les simples visiteurs** (sans login ni mot de passe).Le statut de simple visiteur s'acquiert par le seul fait de se connecter à la Rose des Vents quand les animateurs de la plateforme ont fait le choix de communiquer son adresse.

Ouverture de la Rose des vents au public.

L'ouverture de la Rose des Vents au public s'opère via une interface ad 'hoc qui permet de créer un compte public (utilisateur public) ou d'accéder directement à la Rose des vents (utilisateur simple) sans s'enregistrer).

Pour cela il faut mettre en ligne (Intranet ou Internet) le fichier « AccueilSDM.html » situé dans le répertoire racine de GaïaMundi.

L'interface se présente sous la forme suivante :



Soit on dispose d'un compte : il faut alors saisir logon et mot de passe.

Soit on choisit d'accéder à la Rose des Vents sans compte (Poursuivre sans compte)

Soit un choisit de créer un compte. Dans ce cas, la page d'accueil propose alors le formulaire suivant :

Login	Mot de passe	Adresse mail
<input type="text"/>	<input type="text"/>	<input type="text"/>
OK		
<div>Revenir en arrière</div> <div>Poursuivre sans compte</div>		

La validation du formulaire entraîne l'ouverture de la Rose des Vents avec les autorisations associée au statut d'utilisateur public.

Adressage directe de la Rose des vents

Si votre plateforme est située sur un réseau intranet ou sur un serveur en ligne pouvez rendre publique votre Rose des Vents en rendant publique son adresse ou en insérant celle-ci dans un lien.

L'adresse est constituée comme dans l'exemple suivant ou « SuiteCairo-V1-1-Candidate-UNDP » est le répertoire racine de la plateforme GaïaMundi

Adressage direct sans passer par l'interface de création de compte

Adressage pour un utilisateur disposant d'un compte (public ou adhérent). Ici l'identifiant du compte est « identifiant ».

<http://localhost/SuiteCairo-V1-1-Candidate-UNDP/LaRoseDesVents-1.html?util=identifiant>

- Si vous mettez un identifiant utilisateur inscrit dans le registre des adhérents, vous autorisez l'écriture de données dans les dossiers cartes de la plateforme.
- Si vous voulez mettre un mot de passe public, vous devez d'abord créer ce compte avec le formulaire d'enregistrement ou bien mettre le paramètre **url ?util=public** dans l'adresse. **L'identifiant « public »** est en effet un compte public universel qui permet à tous les internautes de créer des paniers sans pour autant pouvoir accéder à aux dossiers cartes de la plateforme. Dans ce cas les paniers ne sont pas personnalisés.

NB. Le compte « public » n'est pas un compte adhérent de la plateforme. Vous pouvez toutefois récupérer les données des Paniers de « public » à partir de la plateforme en prenant l'option **transfert Panier Public** dans le menu Edition->gestion des données ;

Attention, si vous enregistrez le mot de passe « public » dans les mots de passe adhérents de la plateforme, vous ouvrez la possibilité aux utilisateurs publics d'écrire dans la plateforme.

Adressage pour les seuls utilisateurs simples :

<http://localhost/SuiteCairo-V1-1-Candidate-UNDP/LaRoseDesVents-1.html>

ceci inhibe la possibilité de sauvegarder les fichiers de panier dans GaïaMundi et en particulier, cela inhibe la possibilité d'ajouter des données à un dossier carte existant.

Administration manuelle des Paniers et des registres d'utilisateurs


Physiquement, les paniers de sauvegarde des données collectées par les utilisateurs sur la Rose des vents sont des répertoires générés dans le répertoire racine de GaïaMundi. Leur nom a la forme :

PANIERS-identifiant_utilisateur

Pour un utilisateur donné, les paniers sont placés dans son répertoire dans des sous répertoires dont le nom est le celui attribué au panier dans le formulaire de création du panier dans la Rose des Vents.

L'enregistrement d'un panier par un utilisateur génère un fichier de registre des paniers de l'utilisateur placé dans le répertoire racine de GaïaMundi et dont le nom prend la forme suivante :

mapsILIADEPanier-identifiant_utilisateur.js comme dans l'exemple :

 mapsILIADEPanier-admin.js

Après une déclaration de variable « *var sommableEchelleP=new Array();* », le fichier comprend un bloc de 15 lignes pour chaque panier :

```
mapXP[mapXP.length]="00100"; //Code Maille
mapXP[mapXP.length]=""; // vide
mapXP[mapXP.length]=""; // vide
mapXP[mapXP.length]="principal-1234"; // racine du nom du fichier de données (attribution automatique)
mapXP[mapXP.length]="PANIERS-admin/essai3"; // chemin du panier : ici essai3 est le nom du répertoire panier
var echelle="Monde"; // contour
var maille="country"; // niveau géographique = maille
var commentTitrecarte="troisième essai de panier sur le Monde"; // nom du panier tel qu'il apparaît dans les menus
var commentaireschamp="mlkjlklkj"; // explicitation du contenu de panier
var theme=new Array(); // on laisse tel quel
var autrescommnt=""; // on laisse tel quel
sommableEchelleP[sommableEchelleP.length]= new Array("non précisé","non précisé","2"); // rempli par défaut. On peut modifier ces paramètres manuellement selon la règle définie au chapitre saisie des informations détaillées de la fiche de présentation de la carte dans le manuel d'insertion de nouvelles cartes de GaïaMundi
var originalauthors=""; // auteur du panier : facultatif puisqu'on a l'identifiant
var otherauthors=new Array(); // on laisse tel quel
libelmapP[libelmapP.length]=new Array(echelle,maille,commentTitrecarte,commentaireschamp,theme,autrescommnt,originalauthors,otherauthors); // on laisse tel quel
```

Comment se dessine la géographie des fichiers paniers ?

Chaque dossier panier dans le répertoire « PANIERS-identifiant_utilisateur » comprend trois fichiers

- Principal-1234.html
- Principal-1234.js
- DATA-mapoocord.js

Ce dernier fichier est l'opérateur de connexion entre les différentes fonctions d'affichage et d'insertion dans les différentes interfaces de transfert. Il comporte en particulier une information importante (**en rouge dans l'exemple**) : l'URL du dossier carte de provenance des premières données insérées dans le panier, c'est-à-dire le chemin url du dossier carte d'origine des premières colonnes de données. La première insertion de données définit en effet le contour et la maille du fichier panier.

Ce chemin permet d'accéder aux données géographiques permettant de dessiner, en dynamique, la carte d'origine dans la visionneuse de Rose des vents.

Ces données sont au format de fichier **.js**, ce qui permet de les incorporer quelque soit le serveur sur lesquelles elles se trouvent (le fichier de données géographique est du type DATA-SVG2.js).

Exemple de contenu du fichier DATA-mappocoord.js

```
mappocoord[0]="principal-1234.html" //" fichier de données sujet"
mappocoord[1]="""
mappocoord[2]="1" //
mappocoord[3]="1" //
mappocoord[4]="zz" // titre du pznirt
mappocoord[5]="0" //?
mappocoord[6]="PANIERS-admin/essai3" //répertoire du fichier principal
mappocoord[7]=" PANIERS-admin/essai3" //répertoire du fichier complémentaire
mappocoord[8]="0" //
mappocoord[9]="0" //
mappocoord[10]="1"
mappocoord[11]="http://localhost/SuiteCairo-V1-1-Candidate-UNDP/UNeducation"
```

Cependant, si la carte d'origine des premières données placées dans le panier comporte plus de 3100 aires géographiques, la Rose des Vents n'affiche pas de carte à son ouverture. Comme dans le cas général (Cf. ci-après), pour les cartes de plus de 3100 aires, elle affiche la liste des cartes disponibles à la même maille. Charge à l'utilisateur de choisir la carte dans laquelle il veut visualiser les données du panier.

Compléments sur le nommage des dossiers cartes dans GaïaMundi au regard des règles de sélection et d'affichage dans la Rose des Vents.

La base cartographique de GaïaMundi et la logique de sélection des cartes.

Comme indiqué dans le manuel de fabrication des cartes, on dispose de trois modèles types de cartes svg

- Le modèle de cartes jusqu'à 258 aires géographiques
- Le modèle de cartes jusqu'à 800 aires
- Le modèle de cartes jusqu'à 3100 aires

A la base, le choix du modèle découle du fait que pour limiter le temps de téléchargement des cartes, on utilise une matrice dans laquelle les données de la carte seront incorporées dynamiquement lors de l'appel de la carte. Avec la généralisation du haut débit, ce parti pris a perdu de son acuité. Cependant même si dans GaïaMundi et PageCarto les cartes sont maintenant « en dur », c'est-à-dire que les données géographiques, d'azimut, etc.. sont déjà dans le fichier et non plus incorporées en dynamique, le principe initial a été maintenu pour diverses raisons, parmi lesquelles les contraintes du projet la Rose des Vents ont été décisives.

Avant d'être un dispositif de cartographie, la Rose des vents est un dispositif d'administration de données dans un réseau de plateformes. Avant d'être un outil en d'accès au public et un outil d'investigation dans la géographie statistique, la visionneuse de la Rose des Vents avait pour finalité de voir comment une donnée se comportait à l'épreuve des situations géographiques. Il s'agissait moins de travailler sur les situations géographiques avec des données que l'inverse, c'est-à-dire de comprendre les qualités de la données comme indicateur à partir de sa projection cartographique.

Dans cette logique d'administration de données en réseau, pour la partie mutualisées, le parti pris suivant

a été adopté : à chaque fois que l'on mobilise une source de données pour les façonner au format GaïaMundi, et les intégrer dans les plateformes de ressources partagées, on privilégie deux critères dans la collecte de la donnée :

- Avoir l'échelle la plus large
- et la maille la plus fine de façon à pouvoir réutiliser les données façonnées dans n'importe quelle sous partie de l'échelle initiale et dans n'importe quelle maille géographique recomposable à partir de la plus fine.

Dans le cas des données du recensement rendues disponibles en libre téléchargement, la maille communale apparaît dans comme une maille la plus fine mobilisables à l'échelle la plus grande : la France. On se trouve alors avec des fichiers de données correspondant à un très grand nombre d'aires géographiques. Près de 3700 pour les communes de France.

Il n'est bien entendu pas possible d'afficher une carte de 37000 aires géographiques. Le seuil de 3100 a été arrêté comme le niveau limite à la fois en termes technique (temps d'affichage) et en termes de lisibilité.

Il fallait donc mettre en place un répertoire étendu de cartes à des échelles infra, permettant de visualiser les données. On a ainsi constitué un dispositif couvrant l'ensemble de la géographie nationale à la maille communale, à la maille cantonale, zone d'emploi, département, région.

Une géographie IRIS est aussi en cours de production en fonction des besoins des utilisateurs, de même que la géographie mondiale emboîtée (Monde – Etats - Régions selon différentes définitions / gouvernorats...) est aussi en cours d'élaboration en fonction des besoins des utilisateurs.

Cependant l'emboîtement des mailles via des tables de passage n'est pas toujours possible. Certaines données ne sont en effet pas sommables, comme les salaires moyens, les pourcentages bruts, ...

Aussi a-t-il fallu mettre en place un dispositif de contrôle invitant l'utilisateur à faire des choix non seulement en fonction de ses buts (choix de telle géographie) mais aussi en fonction de contraintes liées aux données, aux machines (temps d'affichage etc...) et à la lisibilité.

D'où le dispositif de sélection pour afficher les données d'un fichier que l'on a choisi dans la Rose des Vents.

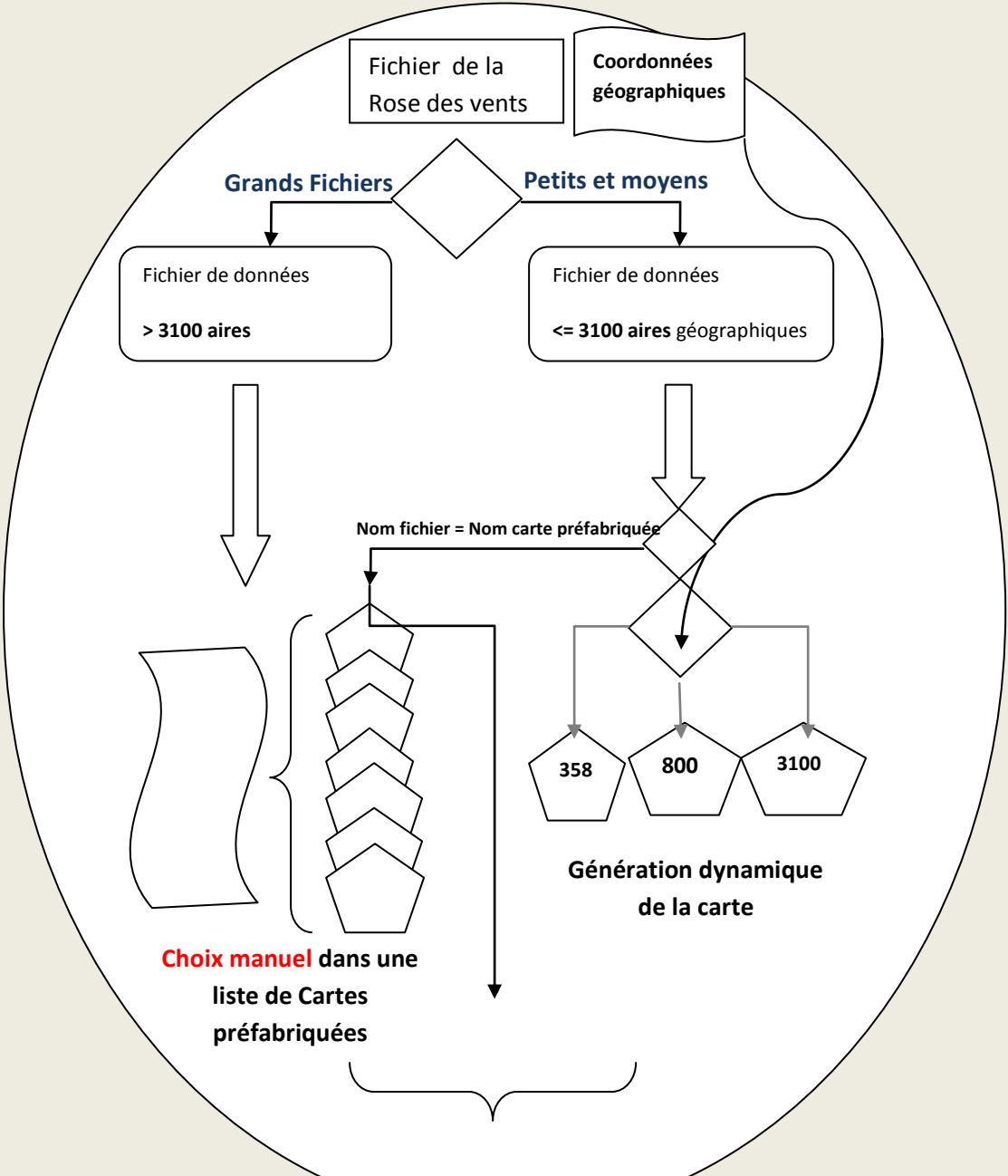
Ce dispositif devait aussi prendre en compte un autres contraintes liées au projet même du réseau : rendre accessible des données construites à des échelles « sur mesure » : par exemple un découpage spécifique autour d'une agglomération pour lequel les auteurs de la carte ont collecté des données locales qui n'existent pas à une échelle supérieure.

Il en résulte le schéma de fonctionnement suivant :

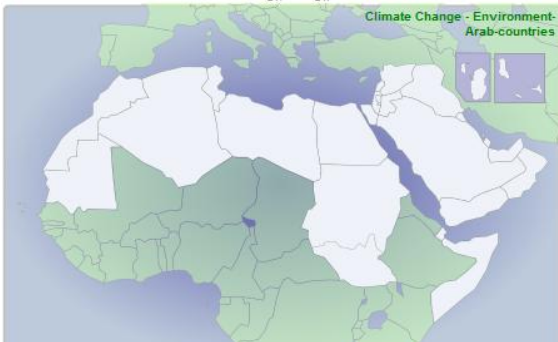
Selon la taille du fichier choisi (plus ou moins 3100 aires) dans la Rose des vents, la gestion de l'affichage des cartes suit l'une ou l'autre des deux voies suivantes :

Voie de gauche pour les grands fichiers. Aucune carte n'est affichée automatiquement. **L'utilisateur doit choisir dans une liste de cartes** à la même maille que celle du fichier. un sélecteur lui propose aussi de choisir une maille emboîtée. S'il choisit une telle maille, une nouvelle liste de carte s'affiche correspondant à nouvelle maille. Le sélecteur indique l'information sur la sommabilité des données. L'utilisateur peut passer outre lorsqu'il y a une indication de non sommabilité (mais c'est très risqué) ou lorsque l'information n'est pas renseignée ou l'indication indique à la prudence. Pour permettre ces choix il y a en arrière plan une banque de carte préfabriquées pour la Rose des Vents. Elle est située dans le répertoire **CartoPanier** dans le répertoire GaïaMundi

Voie de droite pour les petits et moyens fichiers. La carte s'affiche automatiquement. Pour cela, en fonction de la taille du fichier, elle mobilise une matrice située dans la banque de carte de la Rose des Vents (répertoire CartoPanier). Cette matrice ne contient aucune coordonnées géographiques ni étalon pour indexer les données. Les coordonnées géographiques sont récupérées dans le fichier DATA-SVG2 situé dans le dossier carte de la plateforme sur laquelle est situé le fichier téléchargé dans la Rose des Vents puis injectées dans la matrice d'affichage de la carte. le format JavaScript de ce fichier permet de le télécharger quel que soit le serveur sur lequel est placée la plateforme où se trouve le fichier. Quant à l'étalon permettant de rapprocher les données de la carte, il est extrait du fichier de données (les trois premières colonnes) et injecté dans la matrice d'affichage de la carte



Affichage dans la visionneuse



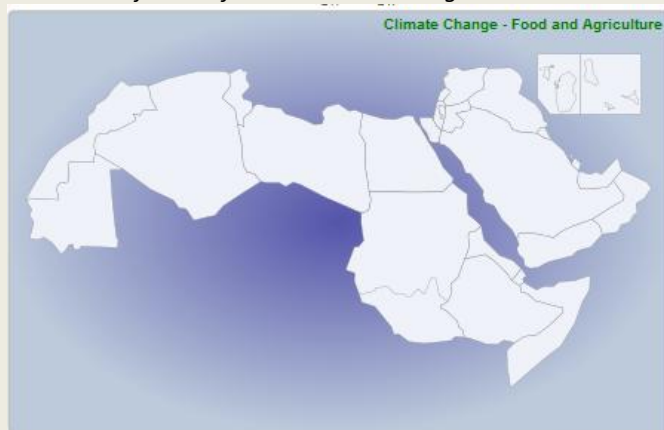
Un autre fichier est téléchargé en même temps que le fichier DATA-SVG2.js des coordonnées géographiques : le fichier DATA-Union.js qui contient les paramètres des unions géographiques des aires de la cartes (par exemple l'union entre les deux parties de l'Égypte séparées par le canal de Suez)

Il existe un cas de bifurcation de la voie de droite vers la voie de gauche. C'est lorsque tout ou partie du nom du fichier choisi contient le nom d'une carte de la banque des cartes préfabriquées.

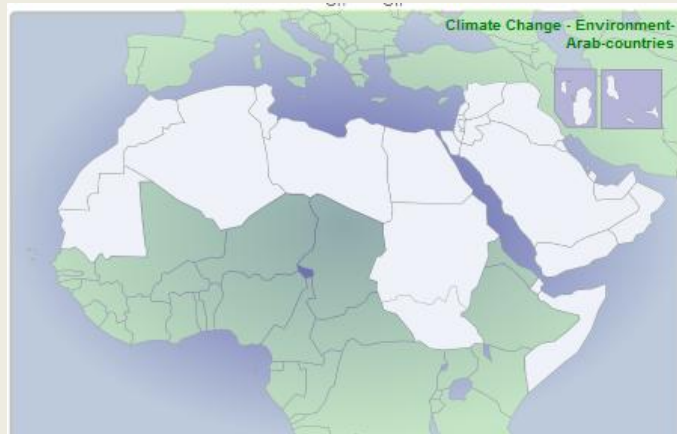
Il est plus simple de donner **un exemple** :

Les animateurs d'une plateforme qui traite des pays arabes ont souhaité réaliser une carte avec en trame de fond les pays environnant et différentes indications non spécifiquement en rapport avec les données statistiques.

La carte brute réalisée à parti des coordonnées de contour des pays arabes contenues dans le fichier DATA-SVG2.js lié au fichier « Food and Agriculture » est la suivante



Tandis que la carte préfabriquée placée dans la banque CartoPanier est la suivante :



En réalité la carte préfabriquée est nommée « Arab-countries ». C'est elle qui a été choisie automatiquement pour le fichier « Climate Change – Environnement – Arab-countries » parce que le nom de ce fichier contient le nom « Arab-Countries » de la carte préfabriquée.

Ainsi, lorsqu'on dispose de cartes préfabriquées spéciales comme celle-ci, on peut adopter systématiquement la convention de nommage des dossiers carte de la plateforme concernée pour que les cartes sélectionnées choisissent automatiquement cette carte plutôt que d'afficher la carte minimale des contours réalisée avec le fichier de coordonnées géographiques.

Un autre Manuel précisera comment construire les cartes préfabriquées de la Rose des Vents et comment paramétrer la gestion automatique des mailles emboîtées ainsi que la liste des cartes dans laquelle l'utilisateur doit choisir dans le cas des grands fichiers de données.