

Diagrammes et histogrammes avec étiquettes automatiques

Introduction

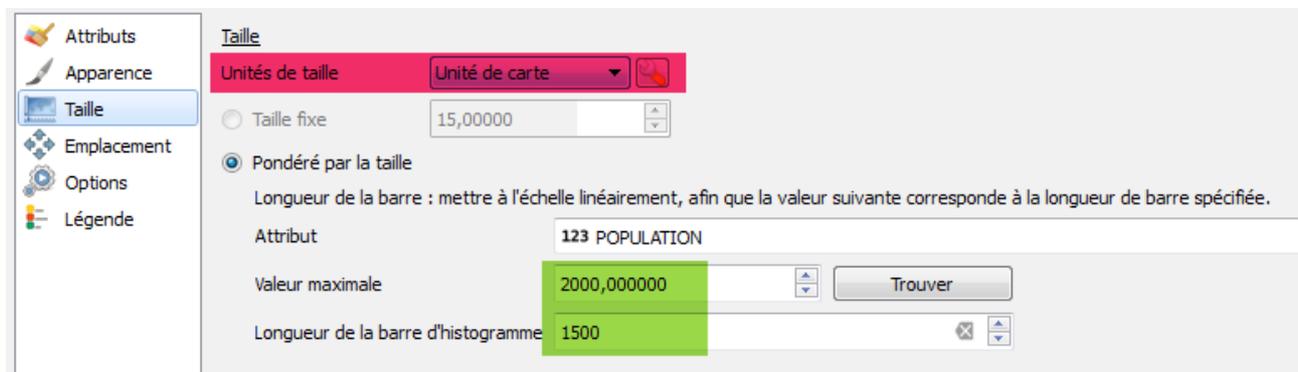
A ce jour, il n'est pas possible de produire de diagrammes ou histogrammes avec étiquettes sous QGIS. Cette fiche présente des astuces pour parvenir à un résultat avec un placement automatique. Il est possible de dériver les idées présentées ici pour adapter le rendu à ses propres besoins. Toute remarque permettant d'améliorer ou compléter cette fiche est la bienvenue. Il est possible d'envisager une fonction python permettant de raccourcir les écritures pour les générateurs de géométrie dans le cas des diagrammes.

Jeu de données utilisé

Nous utilisons la table des communes de la BdCarto fournie dans les Formations à distance (FOAD) *QGIS prises en main* et *QGIS perfectionnement*, dans laquelle nous avons ajouté deux colonnes supplémentaires 'Pop_2030' et 'pop-2050' pour lesquelles nous avons utilisé la fonction rand() pour générer aléatoirement des valeurs dérivées de la colonne 'POPULATION'. Nous nous appuyons sur ces 3 colonnes, à titre d'illustration, sans nous préoccuper de la légitimité du choix de la représentation, qui pourrait à juste titre être critiquée.

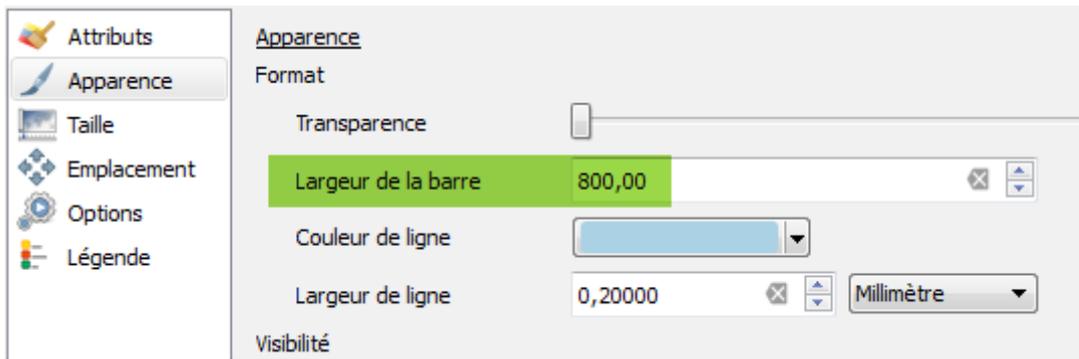
Histogrammes avec étiquettes

Dans un premier temps nous produisons les histogrammes avec une taille choisie en unité de carte (fonction du zoom pour avoir un bon rendu)

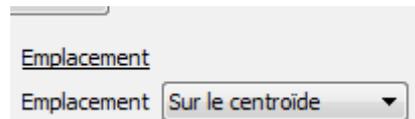


Il est important de noter l'échelle de longueur des barres de l'histogramme. Ici 1500/2000 soit 15/20 (**echelle_barre** = 15/20)

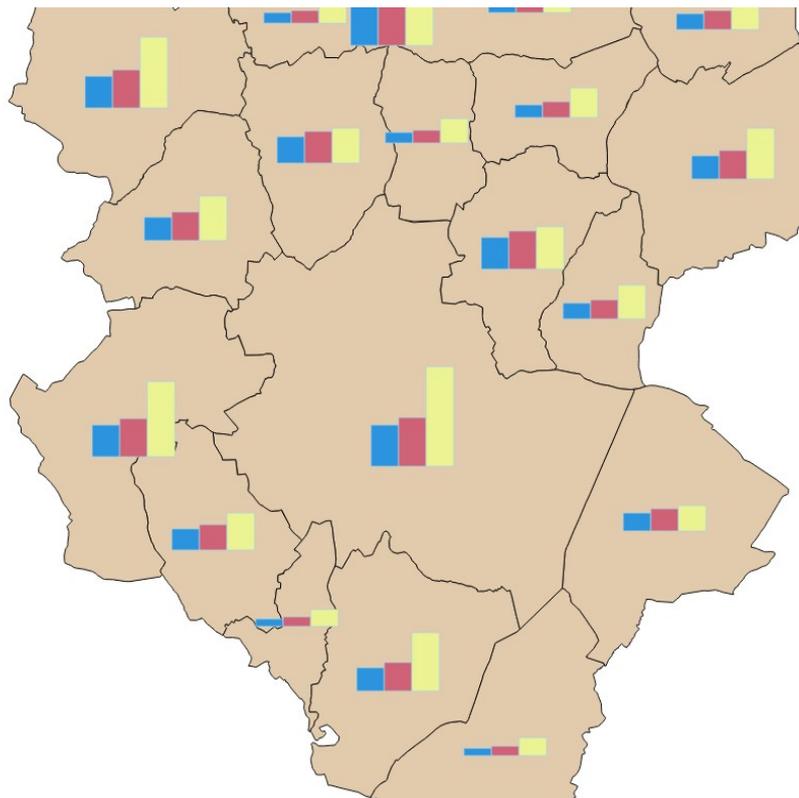
Nous fixons également la largeur des barres dans l'onglet 'apparence' (l'unité est la même que pour l'onglet taille (donc ici en unité de carte).



Nous imposons l'emplacement aux centroïdes des polygones :



le résultat est le suivant :



Nous créons ensuite des étiquettes avec un retour à la ligne entre chaque champ

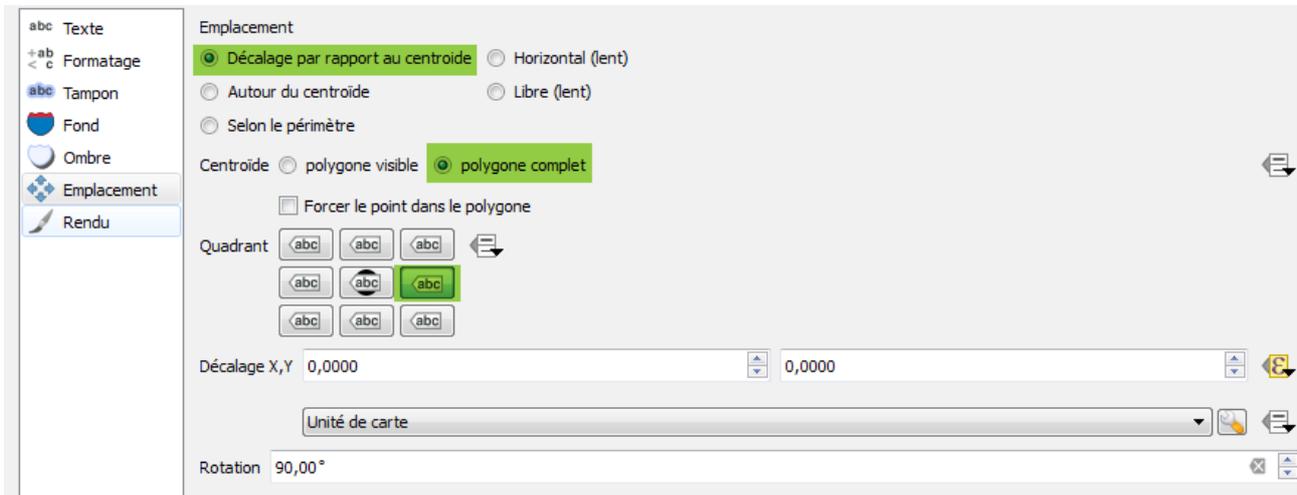
Exemple :

```
to_string( "POPULATION" ) || '\n' ||
```

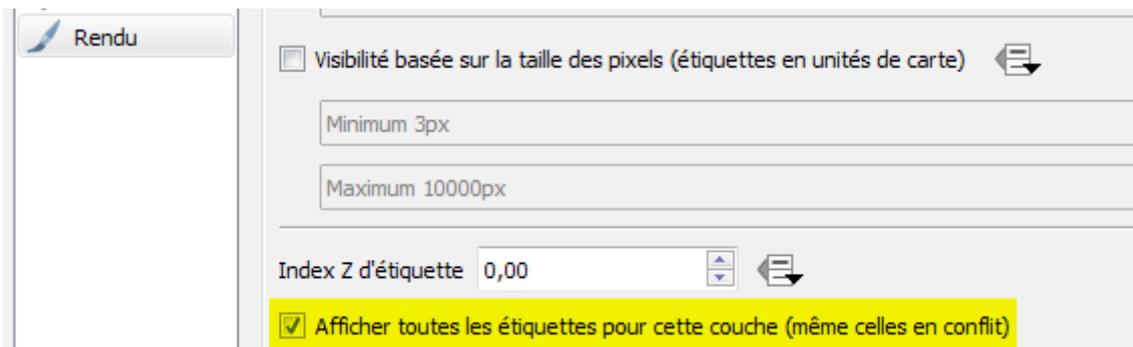
```
to_string( "Pop_2030" ) || '\n' ||
```

```
to_string( "pop_2050" )
```

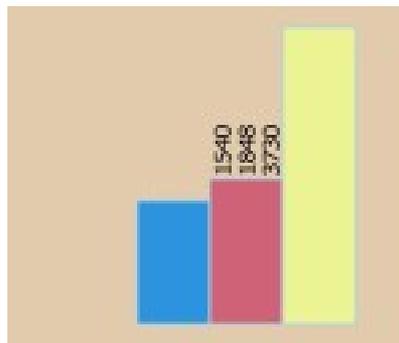
On impose un positionnement de type 'décalage par rapport au centroïde avec option 'polygone complet' et un quadrant comme indiqué ci-dessous. On impose également une rotation de 90°



On active l'option de rendu



A ce stade les étiquettes ne sont pas encore bien positionnées :



On va maintenant imposer un décalage avec la formule suivante pour calculer le décalage en Y

$$\max(\text{champ1}, \text{champ2}, \dots) * \text{échelle_barre}/2$$

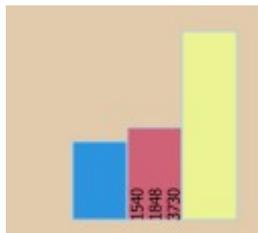
ce qui nous donne dans notre cas

```
to_string( 0) || ';' ||  
to_string( max("POPULATION", "pop_2030", "pop_2050")*15/40)
```

en effet la valeur attendu est (décalage_X, décalage_Y) et on prend 0 pour décalage en X
Cette formule est à placer dans le décalage calculé :

Décalage X,Y 0,0000 0,0000

ce qui nous donne :

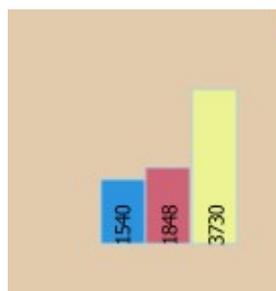


Il reste à affiner en jouant sur la hauteur de ligne :

abc Texte
+ab Formatage
< c
abc Tampon
Fond
Ombre
Emplacement
Rendu

Formatage
Lignes multiples
Retour à la ligne sur le caractère
Hauteur de ligne 1,00 ligne
Alignement Au centre
 Nombres formatés

et éventuellement la taille des caractères pour obtenir :



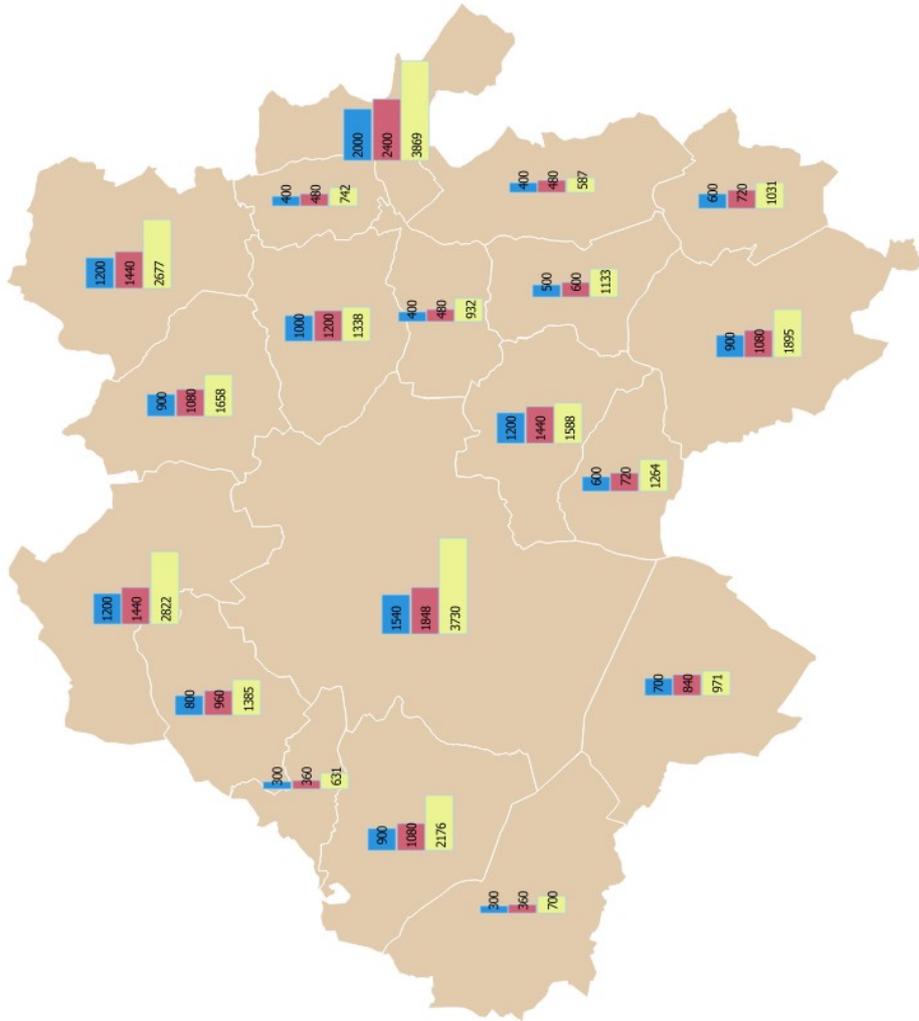
On pourra jouer sur le z-index de l'histogramme si certaines étiquettes n'apparaissent pas.

Index Z du diagramme -1,00

On peut également modifier légèrement la formule de décalage des textes pour conserver un espace :

```
to_string( 0) || ';' ||  
to_string( max("POPULATION", "pop_2030", "pop_2050")*15/40 - 100)
```

Le résultat est alors le suivant :

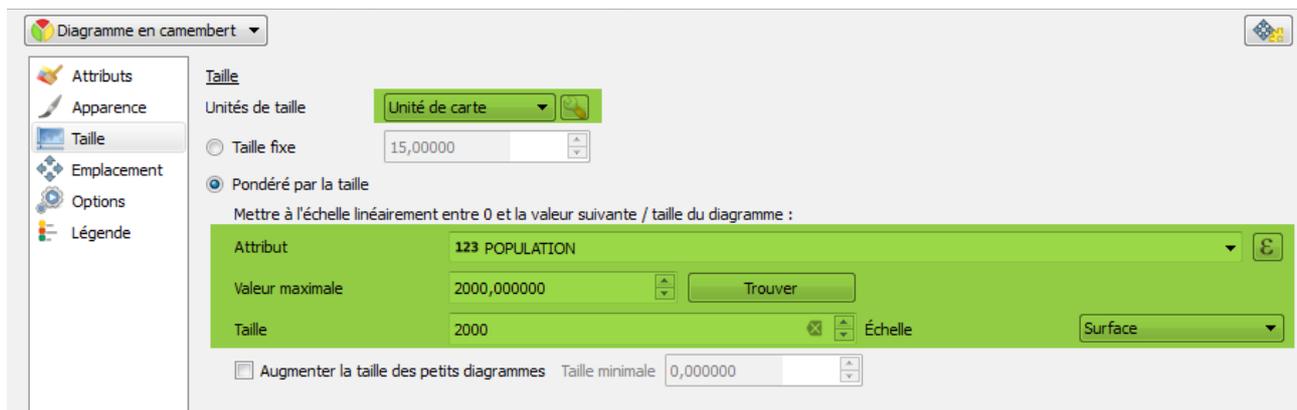


Diagrammes en Camembert avec étiquettes automatiques

Des solutions ont été publiées sur internet pour réaliser des cartes de diagrammes avec étiquettes. Voir par exemple [ici](#) pour un déplacement manuel de 'diagramme texte', ou [ici](#) pour l'utilisation de diagramme SVG généré par un autre moyen que QGIS.

Nous nous intéressons ici à un exemple avec placement automatique sous QGIS.

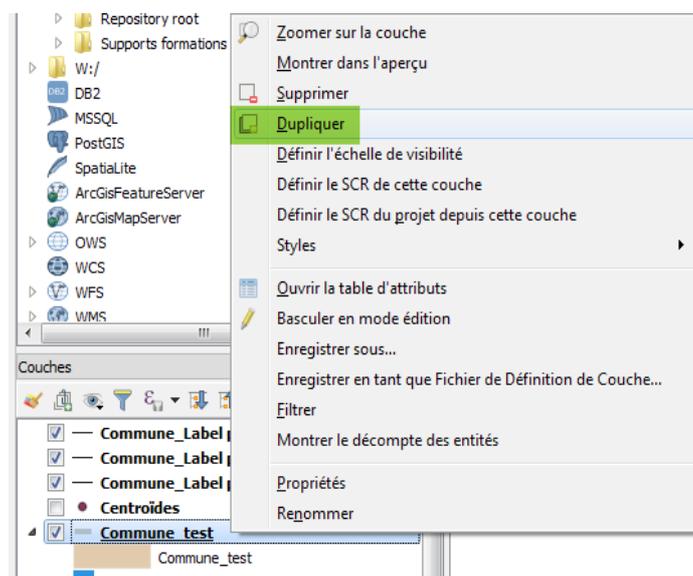
Nous supposons que les diagrammes sont pondérés par la taille selon la variable 'POPULATION' avec une **échelle_diagramme** de 1 (2000/2000) et selon la 'Surface'.



Les attributs ayant été choisis dans l'ordre ci-dessous :

Attributs utilisés		
Attribut	Couleur	Légende
"POPULATION"		POPULATION
"Pop_2030"		Pop_2030
"pop_2050"		pop_2050

La représentation des étiquettes et des traits de rappels nécessite autant de couches dupliquées à partir de la couche de départ qu'il y a de variables à représenter (3 dans notre cas).

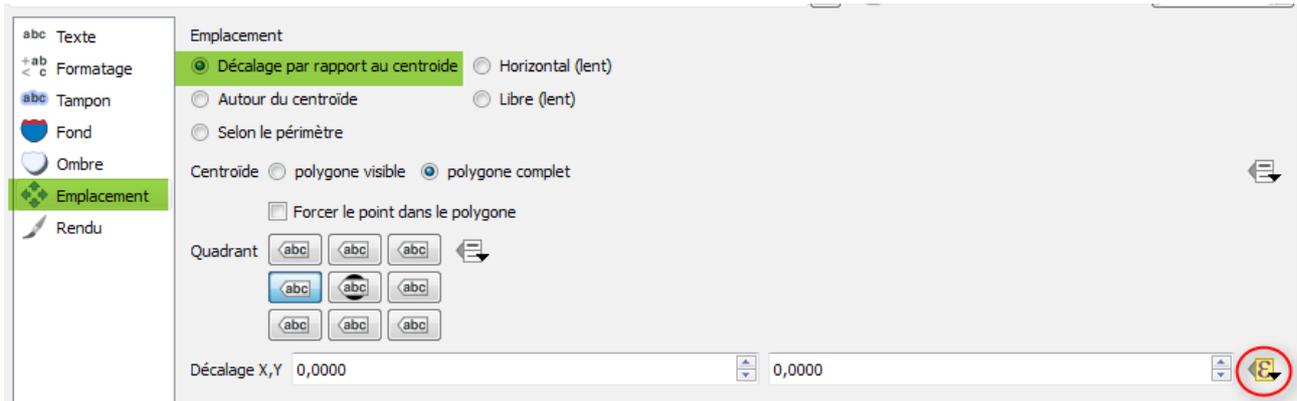


Nous renommons les couches de la façon suivante :

- Commune_Label population
- Commune_Label pop2030
- Commune_Label pop2050

(nous allons voir un peu plus loin pourquoi la représentation de ces trois couches est linéaire)

Pour la couche 'commune_label_population' dans l'onglet *étiquettes* nous étiquetons avec la variable 'POPULATION'. Pour l'emplacement nous indiquons un 'décalage par rapport au centroïde' que nous fixons à l'aide d'une formule :



la formule est la suivante :

```
to_string( ((sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2) + 250)
* cos(pi()/2 + pi() * ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))))
|| ';' ||
to_string( -((sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2) + 250)
* sin(pi()/2 + pi() * ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))))
```

que nous reprenons ci-dessous pour faciliter les *copier/coller*:

```
to_string( ((sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2) + 250)
* cos(pi()/2 + pi() * ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))))
|| ';' ||
to_string( -((sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2) + 250)
* sin(pi()/2 + pi() * ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))))
```

Explication pour la valeur du rayon retenu :

$((\sqrt{("POPULATION"/ \text{maximum}("POPULATION"))} * 2000/2) + 250)$

↑
champ de pondération pour la taille des diagrammes

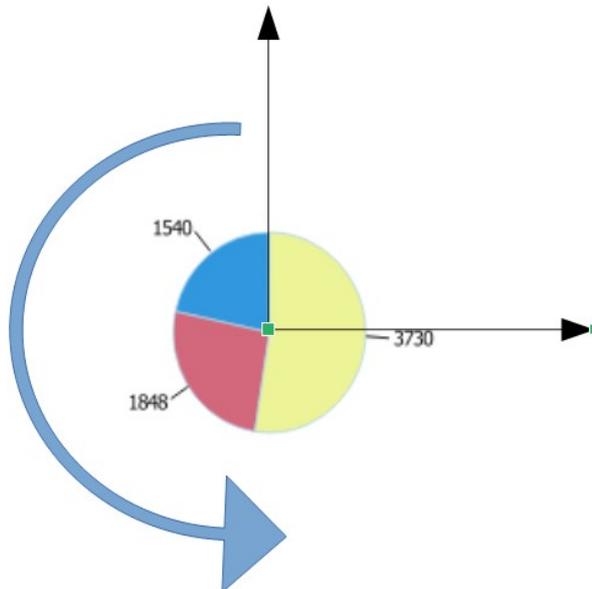
↑
taille du diagramme en unité de carte pour la valeur maxi du champ de pondération de taille

↑
espace supplémentaire en unité de carte pour que le label ne touche le trait de rappel

La fonction sqrt (racine carrée) est utilisée, car on a choisi une échelle variant suivant la

surface pour la taille des diagrammes.

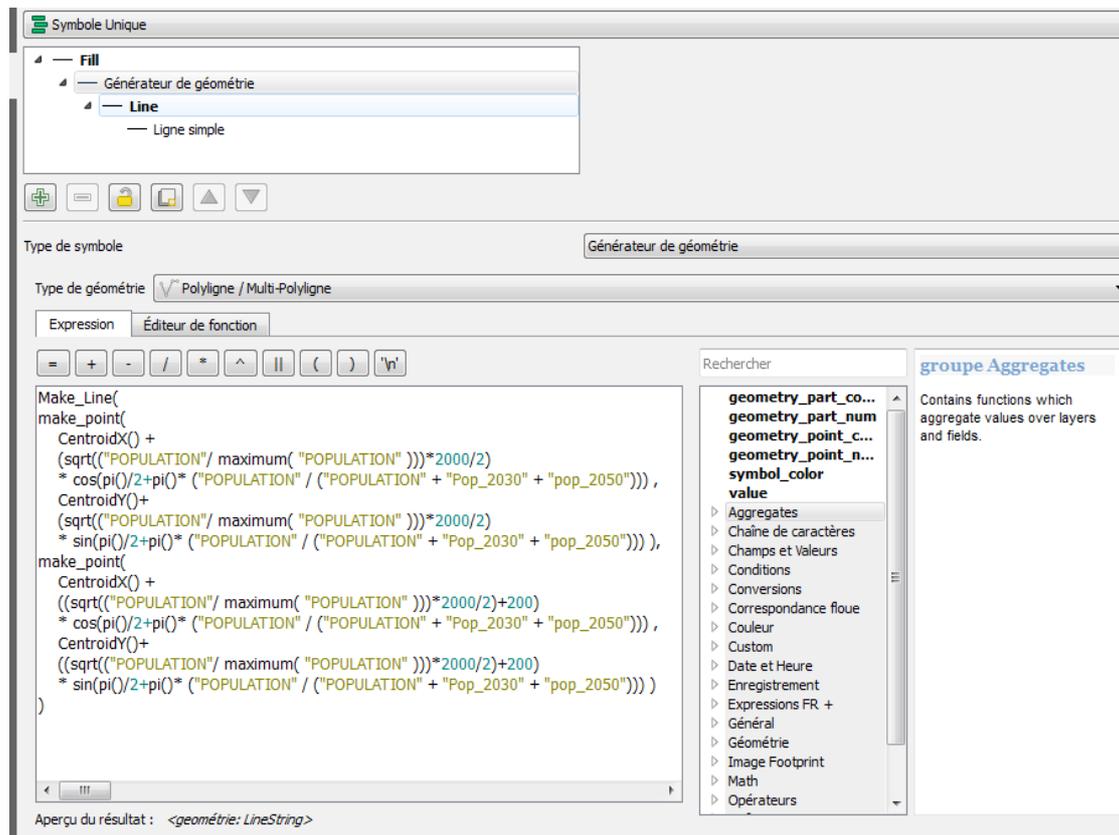
Explication pour la valeur des coordonnées exactes tenant compte d'un placement autour du diagramme selon un angle :



L'angle est compté depuis l'axe des Y dans le sens anti-horaire. Un tour complet vaut 2π radians et représente la somme des valeurs des variables (ici POPULATION + pop2030 + pop2050). La première étiquette est placée selon l'angle $\left(\frac{2\pi * (v_1/2)}{v_1+v_2+v_3+\dots}\right)$

Nous utilisons ensuite les générateurs de géométrie pour représenter le trait de rappel de la 1ère variable dans le style de la couche '*commune_label_population*' :

Le générateur de géométrie utilise peu ou prou les mêmes formules que ci-dessus :



L'expression utilisée est la suivante :

```

Make_Line(
make_point(
  CentroidX() +
  (sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2)
  * cos(pi()/2+pi()* ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))),
  CentroidY()+
  (sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2)
  * sin(pi()/2+pi()* ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))),
make_point(
  CentroidX() +
  ((sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2)+200)
  * cos(pi()/2+pi()* ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))),
  CentroidY()+
  ((sqrt(("POPULATION"/ maximum( "POPULATION" ))) * 2000/2)+200)
  * sin(pi()/2+pi()* ("POPULATION" / ("POPULATION" + "Pop_2030" + "pop_2050"))))
)
)

```

De la même façon on va générer les étiquettes et les traits de rappels pour les autres couches représentant les autres variables. Il faut juste penser à changer le nom de la variable dans l'onglet *Étiquette* et généraliser l'angle avec la formule ci-dessous pour la variable v_i :

$$\left(\frac{2\pi * (v_1 + \dots + v_i / 2)}{v_1 + v_2 + v_3 + \dots} \right)$$

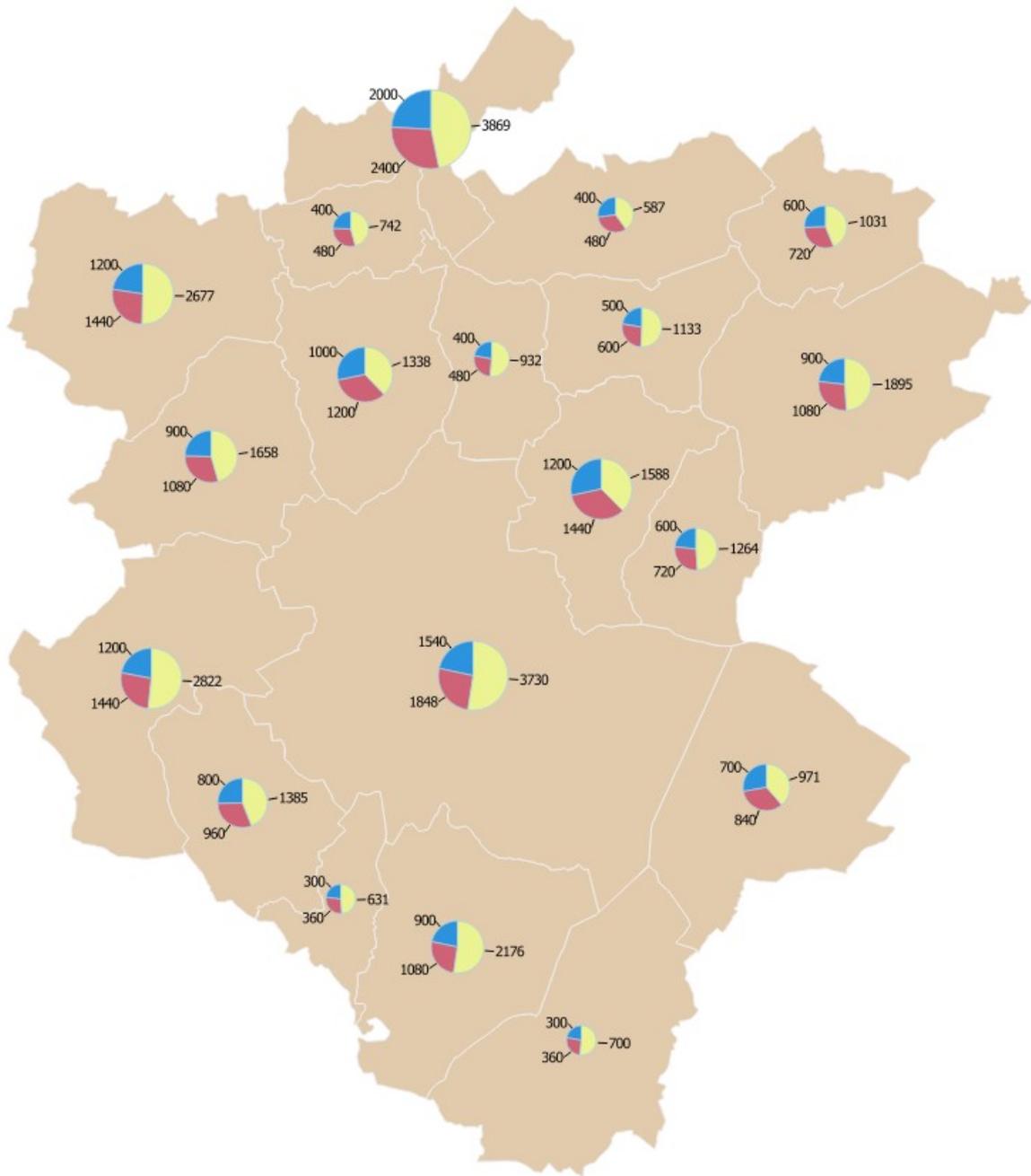
Ceci nous donne par exemple pour notre 3ème variable pop2050: un décalage X, Y de :

```
to_string( ((sqrt(("POPULATION"/ maximum( "POPULATION" )))*2000/2)+250)
    * cos(pi()/2+2*pi()* ("POPULATION"+ "Pop_2030"+"pop_2050"/2) / ("POPULATION" + "Pop_2030"
+ "pop_2050"))))
    || ' ; ||
to_string( -((sqrt(("POPULATION"/ maximum( "POPULATION" )))*2000/2)+250)
    * sin(pi()/2+2*pi()* ("POPULATION"+ "Pop_2030"+"pop_2050"/2) / ("POPULATION" + "Pop_2030" +
"pop_2050")) )
```

et comme formule de générateur de géométrie :

```
Make_Line(
make_point(
    CentroidX() +
    (sqrt(("POPULATION"/ maximum( "POPULATION" )))*2000/2)
    * cos(pi()/2+2*pi()* ("POPULATION"+ "Pop_2030"+ "pop_2050"/2)/ ("POPULATION" + "Pop_2030"
+ "pop_2050"))),
    CentroidY()+
    (sqrt(("POPULATION"/ maximum( "POPULATION" )))*2000/2)
    * sin(pi()/2+2*pi()* ("POPULATION"+ "Pop_2030"+ "pop_2050"/2) / ("POPULATION" + "Pop_2030"
+ "pop_2050"))),
make_point(
    CentroidX() +
    ((sqrt(("POPULATION"/ maximum( "POPULATION" )))*2000/2)+200)
    * cos(pi()/2+2*pi()* ("POPULATION"+ "Pop_2030"+ "pop_2050"/2) / ("POPULATION" + "Pop_2030"
+ "pop_2050"))),
    CentroidY()+
    ((sqrt(("POPULATION"/ maximum( "POPULATION" )))*2000/2)+200)
    * sin(pi()/2+2*pi()* ("POPULATION"+ "Pop_2030"+ "pop_2050"/2) / ("POPULATION" + "Pop_2030"
+ "pop_2050"))))
)
```

ceci nous donne au final le résultat page suivante :



Il faut cependant noter que le décalage à droite ou à gauche du trait de rappel pour les étiquettes dépend de leur écriture à droite ou à gauche du diagramme. Ici nous voyons que les deux premières étiquettes sont toujours à gauche et la 3ème à droite. Nous avons donc fixé pour les deux premières :

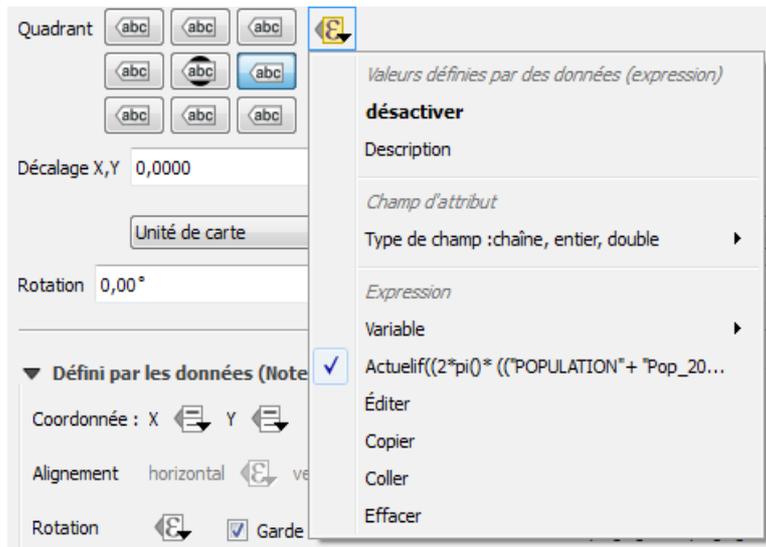


et



pour la 3ème.

Dans un cas plus général, il faut déterminer si l'angle utilisé est plus petit ou plus grand que π . On utilise alors un quadrant dépendant d'une expression :



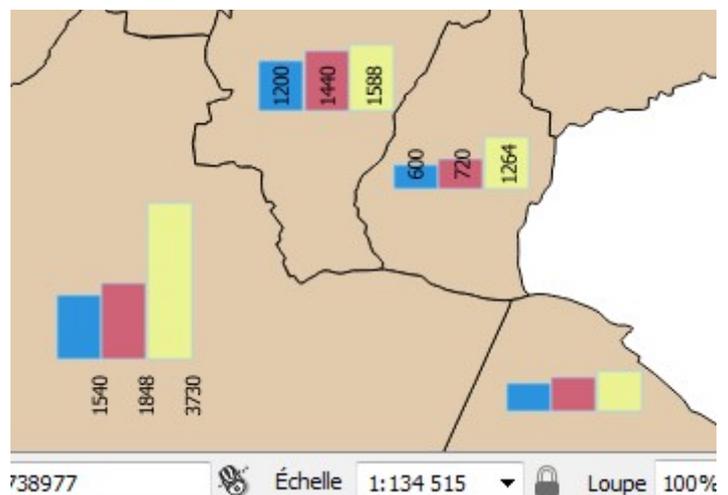
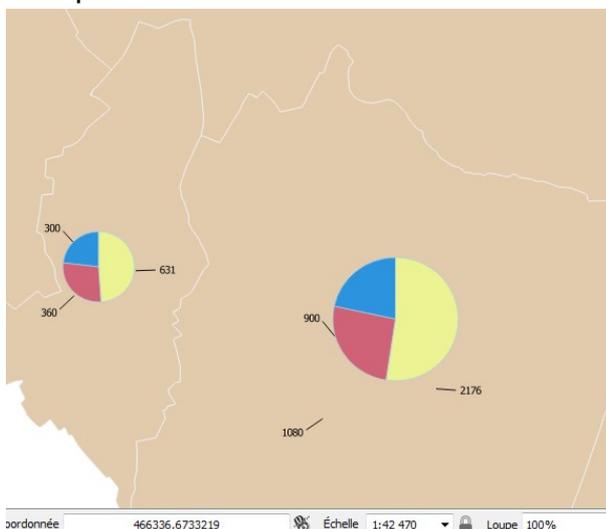
Avec des formules du type (ici pour la 3ème variable *pop2050*) :

`if((2*pi)* ((\"POPULATION\" + \"Pop_2030\" + \"pop_2050\"/2)/ (\"POPULATION\" + \"Pop_2030\" + \"pop_2050\")))>pi(), 5,3)`

positionnement automatique des diagrammes en cas de polygones non entièrement visible :

Sous QGIS 2.16, il n'est pas possible d'imposer un positionnement des diagrammes au centroïde si le polygone n'est pas entièrement visible :

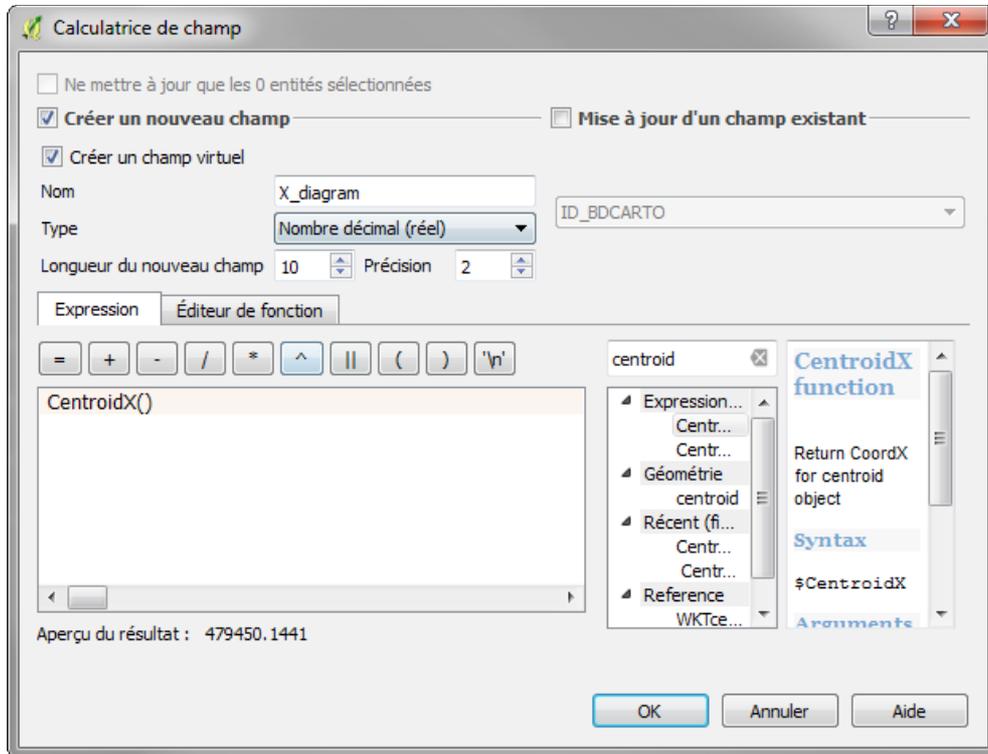
exemple :



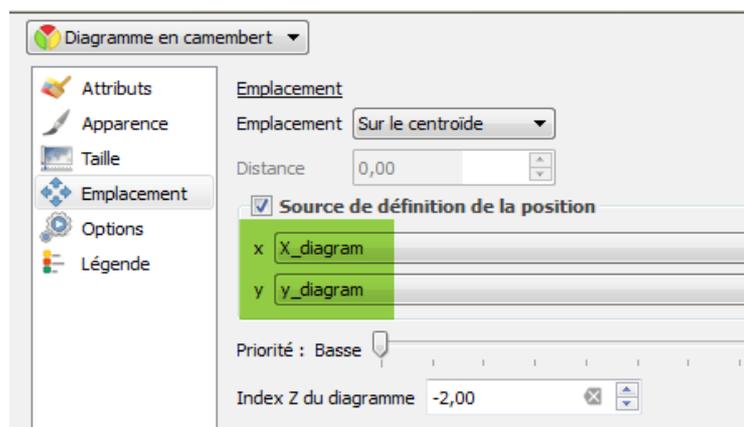
Les étiquettes sont alors mal positionnées (voir ci-dessus) pour les polygones partiellement visibles.

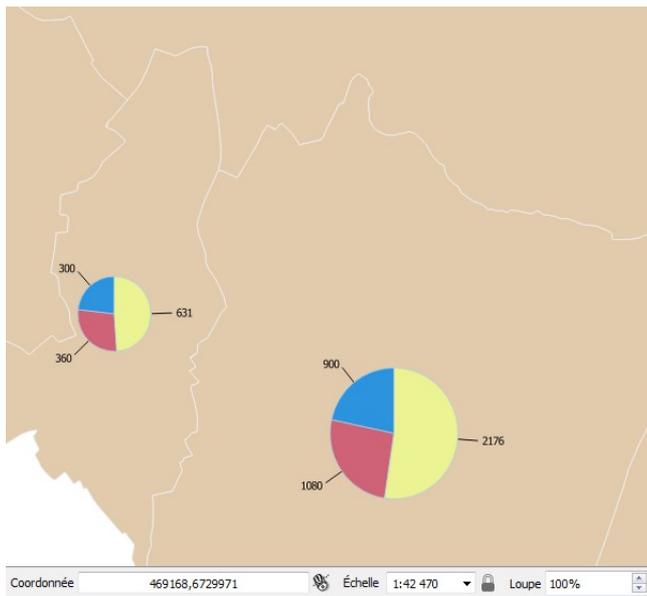
Pour imposer le positionnement au centroïde on peut créer des champs virtuels (stockés dans le fichier projet) qui seront les sources de positionnement des diagrammes :

exemple ; création du champ virtuel $x_diagram$:



On crée de même $y_diagram$, puis on indique que la position du diagramme dans la couche de départ est dépendant de ces variables :





La représentation est maintenant correcte.

