

## LES NOUVEAUTÉS DE L'ODS GRAPHICS EN VERSION 9.3

Nous allons présenter dans cet article des fonctionnalités qui appartenaient jusqu'à présent au module SAS/GRAPH®, et qui sont maintenant déportées dans le module Base SAS®, à compter de la version 9.3 de SAS. Vous pourrez ainsi désormais :

- réaliser des graphiques statistiques de façon automatique,
- mais aussi les modifier facilement, via l'interface ODS Graphics Editor,
- ou via du code spécifique pour en généraliser la production
- et les créer via des clics de souris avec ODS Graphics Designer, et en récupérer ensuite le code pour exécution dans SAS.



### Caractéristiques :

Catégorie : SAS/GRAPH®

OS : Windows

Version : 9.3

Vérifié en décembre 2011

### Introduction à l'ODS GRAPHICS

L'une des grandes nouveautés de SAS 9.3 est la mise à disposition de l'ODS Statistical GRAPHICS (appelé plus communément ODS GRAPHICS) dans Base SAS. En mode interactif, il est maintenant activé par défaut. S'agissant d'ODS, son utilisation implique d'avoir choisi une destination ODS pour l'affichage des résultats. La destination Listing (soit la fenêtre Output (ou Sortie) de SAS) est donc exclue. C'est pourquoi, dans une session SAS 9.3 démarrée en interactif, la destination par défaut n'est plus Listing mais désormais le HTML, qui est l'une des huit destinations supportant l'ODS GRAPHICS ([accès à la liste complète](#)).

En conséquence, tous les graphiques supplémentaires (qui jusqu'en version 9.2, s'affichaient uniquement quand ODS GRAPHICS ON ; était utilisé) seront automatiquement présents dans les sorties réalisées avec la version 9.3. Sont concernées plus de 80 procédures de Base SAS, SAS/STAT®, SAS/ETS® et SAS/QC® ([accès à la liste complète](#)), contre environ 60 en version 9.2.

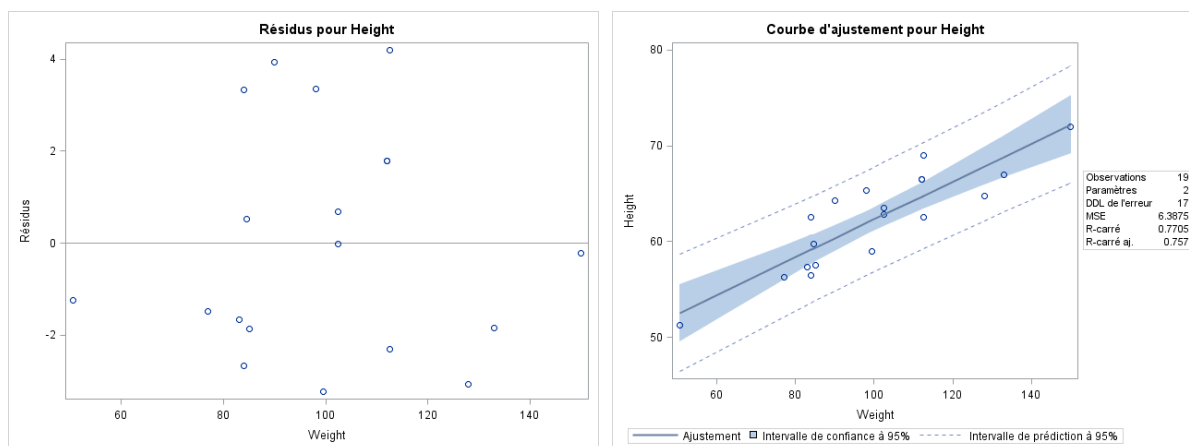
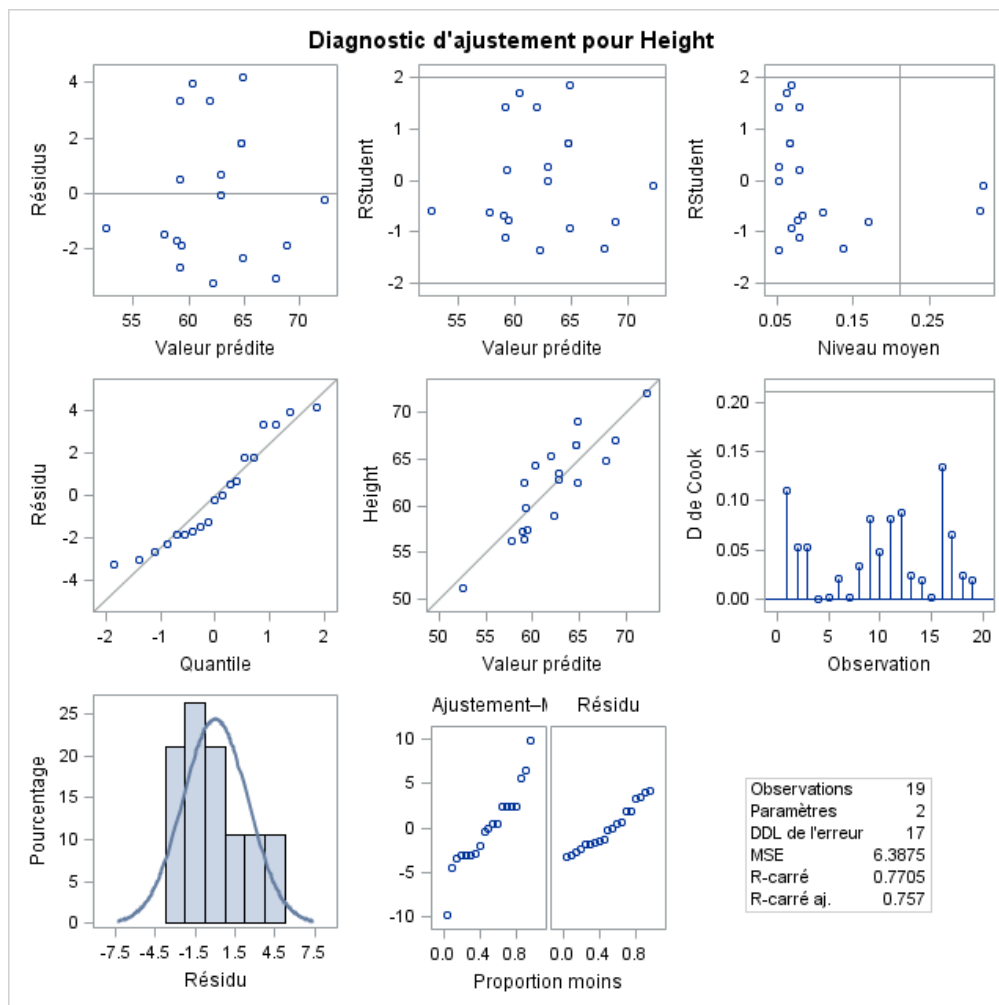
Enfin, (toujours en mode interactif) le style par défaut utilisé pour l'affichage des résultats est désormais HTMLBLUE (contre DEFAULT dans les versions précédentes). Il a été mis au point spécialement pour l'ODS GRAPHICS, avec des couleurs bleues (contre grises et bleues auparavant), des styles de tableaux différents, des types de lignes et des symboles différents dans le cas de représentation de groupes. Ces différences sont illustrées sur [cette page](#) (au niveau des figures 21.19 et 20, puis 21.27 et 28 et enfin 21.35 et 36).

### Illustration de l'ODS GRAPHICS

Prenons l'exemple de la procédure REG et d'une analyse de variance classique.

```
proc reg data=sashelp.class;
model height=weight;
run;
quit;
```

Ce programme, en plus d'afficher les résultats habituels (le tableau d'analyse de variance et des statistiques), donne désormais (en version 9.3) trois graphiques supplémentaires : un diagnostic d'ajustement pour la variable à expliquer, un nuage des points des résidus et la courbe d'ajustement. Avant la version 9.3, pour les obtenir, il était nécessaire d'exécuter au préalable « ODS GRAPHICS ON ; ».



### **ODS GRAPHICS vs les graphiques traditionnels**

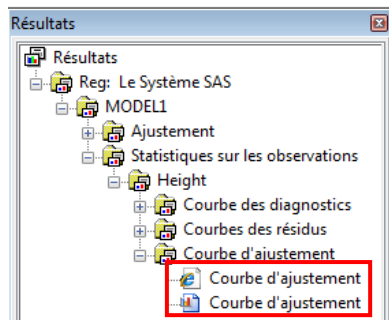
Les graphiques traditionnels, qui existent depuis toujours dans SAS, sont stockés dans des entrées de type GRSEG dans des catalogues. Leur apparence est contrôlée via des options, que ce soit pour le device à utiliser (jpeg, gif, activex, etc), pour les polices, les tailles, et via des instructions spécifiques pour définir les axes (axis), les légendes (legend), les symboles (symbol), etc.

Les graphiques produits avec l'ODS ne sont plus stockés dans des catalogues SAS. Dans le cas de documents HTML, ce sont immédiatement des images, au format PNG (par défaut). Quant à leur apparence, elle est contrôlée par les styles ODS et les templates.

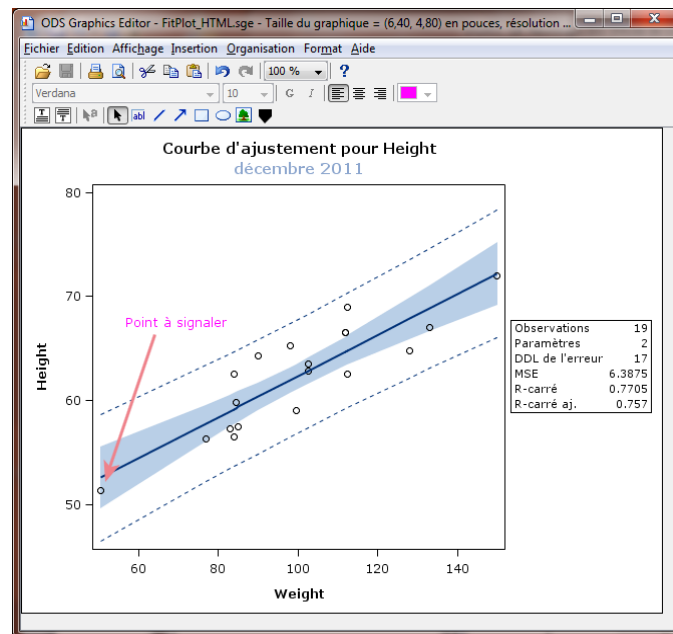
## Modification des graphiques

Dans le cas d'un besoin ponctuel de modification d'un graphique, un éditeur est disponible directement dans SAS. Il est nécessaire d'exécuter au préalable : « ODS HTML SGE=ON ; ». Ainsi l'image, au format SGE, est visible et directement accessible dans la fenêtre Résultats. En double-cliquant dessus, le graphique est ouvert dans ODS Graphics Editor.

Sur le même exemple, ouvrons le graphique contenant la courbe d'ajustement, pour y apporter quelques modifications.



Le premier item « Courbe d'ajustement » permet simplement à un accès direct sur la page HTML au graphique (sashtml.htm#IDX6 par exemple). Le second est bien le fichier sge.



Une fois dans l'éditeur, il est aisé d'ajouter une flèche pour signaler un point, ainsi que du texte, ou encore un titre supplémentaire.

Les graphiques modifiés dans ODS Graphics Editor peuvent ensuite être sauvegardés sous deux formats : PNG ou SGE. Les images peuvent ultérieurement être éditées dans ODS Graphics Editor, mais le format SGE présente l'avantage de pouvoir modifier les personnalisations déjà faites. Sur l'exemple ci-dessus, la couleur rose pourrait être changée, le texte 'Point à signaler' pourrait être supprimé ou modifié.

## Le code Graph Template Language (GTL)

La modification manuelle pouvant rapidement atteindre ses limites, il existe également une technique pour modifier le graphique obtenu pour qu'il persiste à chaque ré-exécution de la procédure. Elle consiste à utiliser le « Graph Template Language » ou GTL. Chaque procédure produisant des graphiques ODS se base sur un 'template' propre. Nous allons maintenant voir sur un exemple comment identifier ce template, le modifier et l'utiliser, pour altérer, de façon récurrente, un type de graphique produit par défaut dans SAS.

Cet exemple se base sur la procédure REG (le même qu'en début d'article). Nous allons nous intéresser au graphique représentant, par un nuage de point, la variable à expliquer par la variable explicative, et où sont superposées la courbe d'ajustement, les intervalles de confiance à 95% et les bornes de prédiction inférieures et supérieures à 95%.

### 1- Comment identifier le nom du template utilisé par la proc REG ?

Grâce à l'instruction « ODS TRACE ON ; » à positionner avant la proc REG. Des informations sont ajoutées au niveau du journal (ou log), dont, entre autres :

```
Sortie ajoutée :
-----
Nom :          FitPlot
Libellé :      Courbe d'ajustement
Modèle :      Stat.REG.Graphics.Fit
Chemin :      Reg.MODEL1.ObswiseStats.Height.FitPlot
```

Le nom du template qui nous intéresse est donc Stat.REG.Graphics.Fit.

## 2- Comment éditer le code correspondant ?

La première méthode consiste à exécuter :

```
proc template;  
    source Stat.REG.Graphics.Fit;  
run;
```

Le code source sera affiché dans la fenêtre Journal et pourra être copié/collé dans l'éditeur pour modification.

La seconde méthode consiste à utiliser le navigateur de modèles, par clic droit sur « Résultats » dans la fenêtre du même nom > Modèles puis en dépliant Sashelp.Tmplmst (la bibliothèque standard de modèles SAS) et en navigant jusqu'à l'objet Fit. Un double-clic sur Fit permet d'accéder au code, qu'il sera ainsi possible de copier/coller dans l'éditeur.

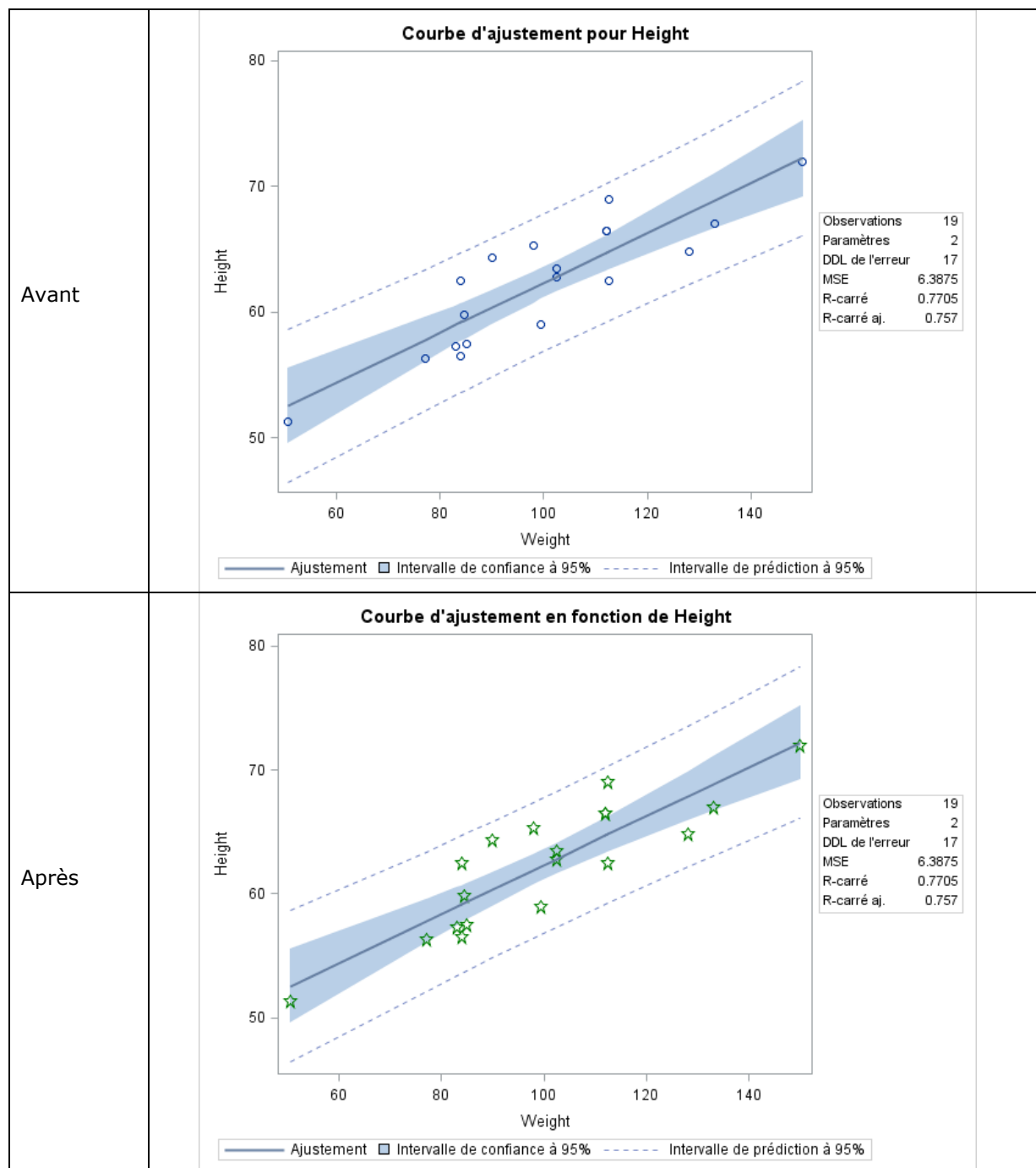
## 3- Exemple de modification de code

Voici le début du template gérant le graphique « Courbe d'ajustement ».

```
proc template;  
    define statgraph Stat.Reg.Graphics.Fit;  
        notes "Fit Plot";  
        dynamic _DEPLABEL _DEPNAME _MODELLABEL _SHOWSTATS _NSTATSCOLS _SHOWNObs  
            _SHOWTOTFREQ _SHOWNParm _SHOWEDF _SHOWMSE _SHOWRSquare _SHOWAdjRSq  
            _SHOWSSE _SHOWDepMean _SHOWCV _SHOWAIC _SHOWBIC _SHOWCP _SHOWGMSEP  
            _SHOWJP _SHOWPC _SHOWSBC _SHOWSP _NObs _NParm _EDF _MSE _RSquare  
            _AdjRSq _SSE _DepMean _CV _AIC _BIC _CP _GMSEP _JP _PC _SBC _SP  
            _PREDLIMITS _CONFLIMITS _XVAR _SHOWCLM _SHOWCLI _WEIGHT _SHORTXLABEL  
            _SHORTYLABEL _TITLE _TOTFREQ;  
        BeginGraph;  
            entrytitle halign=left textattrs=GRAPHVALUETEXT _MODELLABEL halign=  
                center textattrs=GRAPHTITLETEXT _TITLE " pour " _DEPNAME;  
            layout Overlay / yaxisopts=(label=_DEPLABEL shortlabel=_SHORTYLABEL)  
                xaxisopts=(shortlabel=_SHORTXLABEL);  
            if (_SHOWCLM=1)  
                BANDPLOT limitupper=UPPERCLMEAN limitlower=LOWERCLMEAN x=_XVAR  
                    / fillattrs=GRAPHCONFIDENCE connectororder=axis name=  
                        "Confidence" LegendLabel=_CONFLIMITS;  
            endif;  
            if (_SHOWCLI=1)  
                if (_WEIGHT=1)  
                    SCATTERPLOT y=PREDICTEDVALUE x=_XVAR / markerattrs=(size=0)  
                        datatransparency=.6 yerrorupper=UPPERCL yerrorlower=  
                            LOWERCL name="Prediction" LegendLabel=_PREDLIMITS;  
                else  
                    BANDPLOT limitupper=UPPERCL limitlower=LOWERCL x=_XVAR /  
                        display=(outline) outlineattrs=GRAPHPREDICTIONLIMITS  
                            connectororder=axis name="Prediction" LegendLabel=  
                                _PREDLIMITS;  
                endif;  
            endif;  
            SCATTERPLOT y=DEPVAR x=_XVAR / markerattrs=GRAPHDATADEFAULT  
                primary=true rolename=( _tip1=OBSERVATION _id1=ID1 _id2=ID2 _id3=  
                    ID3 _id4=ID4 _id5=ID5 ) tip=(y x _tip1 _id1 _id2 _id3 _id4 _id5  
                    );  
            SERIESPLOT y=PREDICTEDVALUE x=_XVAR / lineattrs=GRAPHFIT  
                connectororder=xaxis name="Fit" LegendLabel="Ajustement";  
            if (_SHOWCLI=1 OR _SHOWCLM=1)  
                DISCRETELEGEND "Fit" "Confidence" "Prediction" / across=3  
                    HALIGN=CENTER VALIGN=BOTTOM;  
            endif;  
        /* fin du code non copié ici*/  
    end;
```

Nous allons en modifier quelques éléments :

- Le titre, en changeant 'pour' par 'en fonction de'
- Les symboles utilisés pour représenter les points



Le code à modifier a été surligné dans le code ci-dessus. Il a été changé comme suit :

- Pour le titre du graphique :

```
entrytitle halign=left textattrs=GRAPHVALUETEXT MODELLabel halign=center
textattrs=GRAPHTITLETEXT _TITLE " en fonction de " DEPNAME;
```

- Pour le type de symbole, leur taille et leur couleur :

```
SCATTERPLOT y=DEPVAR x=_XVAR / markerattrs=(size=10px color=green symbol=star) primary=true
rolename=(_tip1=OBSERVATION _id1=ID1 _id2=ID2 _id3=ID3 _id4=ID4 _id5=ID5) tip=(y x _tip1 _id1
_id2 _id3 _id4 _id5);
```

#### 4- Le stockage de ses propres templates.

Rappelons que les templates standards de SAS sont stockés dans la bibliothèque sashelp. Par défaut, les templates modifiés seront quant à eux stockés dans la bibliothèque sasuser. L'instruction « ODS PATH show ; » permet de le vérifier (le résultat est affiché dans le journal) :

La liste ODS PATH actuelle est :

1. SASUSER.TEMPLAT(UPDATE)
2. SASHELP.TMPLMST(READ)

Une autre bibliothèque peut être spécifiée avec l'instruction « ODS PATH ». En voici un exemple :

```
ODS PATH mytempl.templat(update) mytemp.templat(read) sashelp.tmplmst(read);
```

L'ordre des bibliothèques a toute son importance. Ici, SAS recherchera des templates d'abord dans la bibliothèque mytempl, et ensuite dans la bibliothèque sashelp.

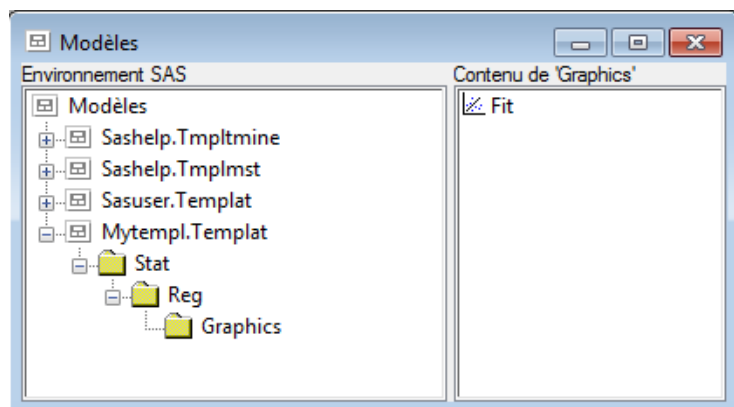
Dans notre cas, le template Stat.REG.Graphics.Fit existe dans les deux bibliothèques. Et c'est donc bien celui stocké dans mytempl qui sera utilisé.

#### 5- La suppression de ses propres templates.

Pour que les graphiques soient à nouveau produits avec le template par défaut de SAS, il est nécessaire de supprimer le template, portant le même nom, qui est stocké dans la bibliothèque mytempl.

Cette suppression peut être réalisée :

- soit via l'interface de SAS, dans la fenêtre Modèles (cf point 2-), avec le bouton 'Suppr' sur l'objet Fit. Cette méthode permet de choisir un template en particulier (en l'occurrence, dans le cas présent, seul un template est présent, mais nous pourrions en avoir plusieurs).



- soit via du code :

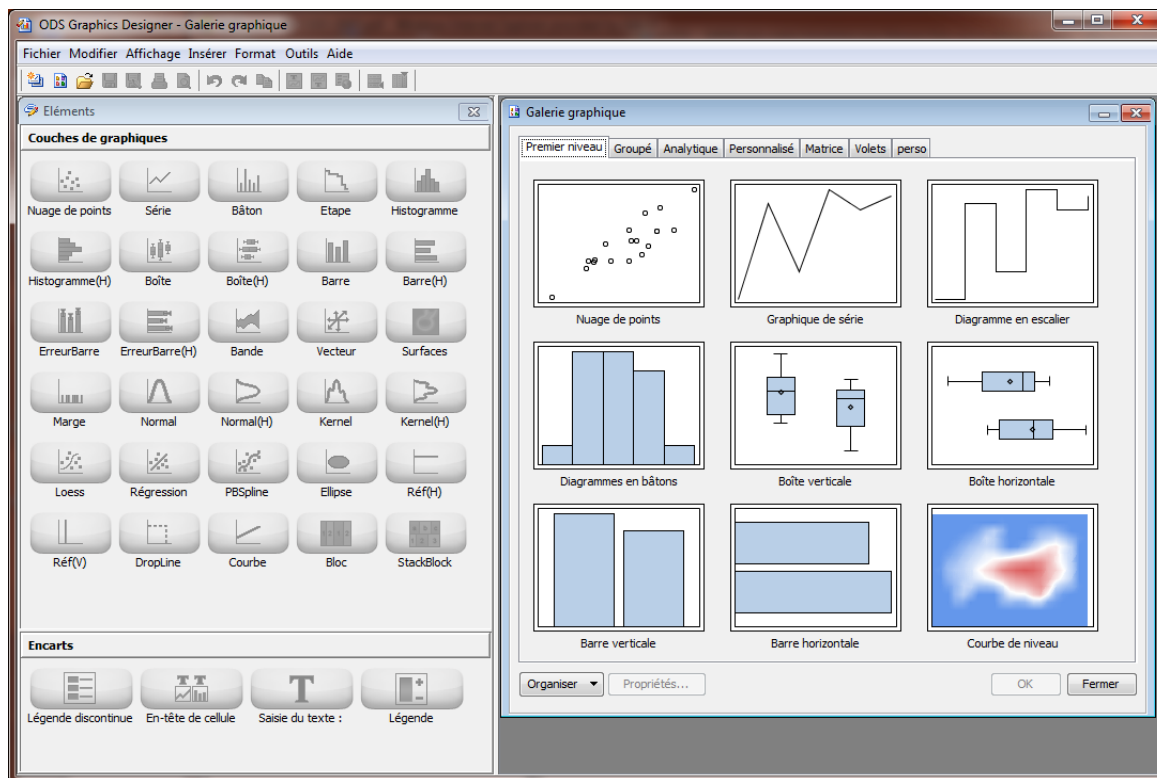
```
Proc datasets lib=mytempl nolist;  
delete templat(memtype=itemstor);  
run;
```

sachant que dans ce cas, tous les templates définis au sein de la bibliothèque spécifiée (ici mytempl) seront supprimés.

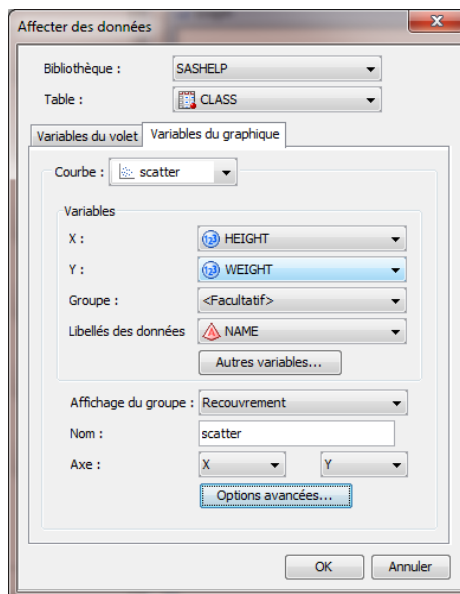
### **SAS ODS GRAPHICS DESIGNER**

Et si les graphiques pouvaient être 'dessinés' à la souris ? Et le code généré à la volée ? Cela permettrait de bénéficier des atouts du code GTL sans avoir à écrire le code soi-même.

Nous vous proposons à cet effet l'outil ODS GRAPHICS DESIGNER. Il s'ouvre depuis SAS, dans le menu Outils (tools). Tout se faisant par clics souris, nous allons maintenant illustrer la création d'un graphique via des copies d'écran.

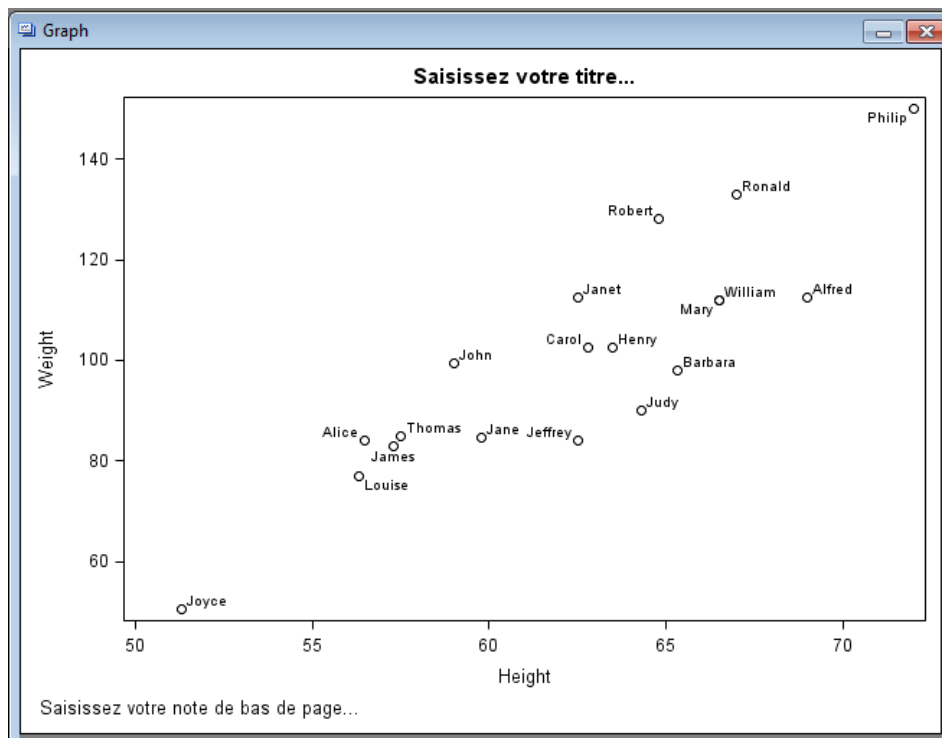


En sélectionnant le type de graphique « Nuage de points », cet écran s'ouvre :



Il permet de spécifier les variables à représenter sur les axes des abscisses et des ordonnées, et éventuellement de spécifier une variable dont le libellé sera affiché sur les points, entre autres. Pour information, la table class de la bibliothèque sashelp est utilisée par défaut.

Le graphique obtenu est celui-ci :



Le titre et la note de bas de page se renseignent par simple clic sur les zones correspondantes. Le libellé sur les variables peut être supprimé via un clic droit sur le graphique en choisissant « Affecter des données... ».

Et la galerie de graphiques à gauche de l'écran permet de superposer des éléments sur le graphique, comme une droite de régression :

**Affecter des donnéesRégression**

Bibliothèque : SASHELP

Table : CLASS

Variables du volet | Variables du graphique

☒ Adapter une courbe existante

Courbe : scatter

Variables

X : <Obligatoire>

Y : <Obligatoire>

Autres variables...

Bande modèle : ☒ CLM ☐ CLI

Nom : regression

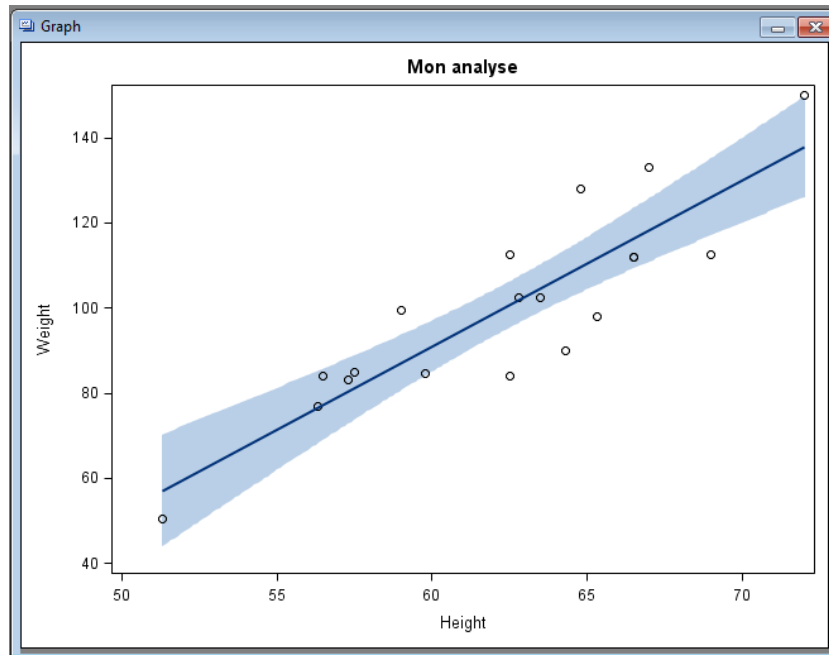
Axe : X Y

Options avancées...

OK Annuler



Ce qui donne :



Une fois le graphique finalisé, le code SAS correspondant peut être affiché en passant par le menu Affichage > Code :

```
proc template;
define statgraph sgdesign;
dynamic _HEIGHT _WEIGHT;
begingraph;
  entrytitle halign=center 'Mon analyse';
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10 columngutter=10;
  layout overlay;
    modelband 'CLM3' / name='modelband';
    scatterplot x=_HEIGHT y=_WEIGHT / name='scatter';
    regressionplot x=_HEIGHT y=_WEIGHT / name='regression' clm='CLM3';
  endlayout;
endlayout;
endgraph;
end;
run;

proc sgrender data=SASHELP.CLASS template=sgdesign;
dynamic _HEIGHT="HEIGHT" _WEIGHT="WEIGHT";
run;
```

A noter : la fenêtre « code » peut être activée à tout moment, ce qui permet de voir le code évoluer au fil des actions réalisées sur le graphique.

Le programme peut ensuite être exécuté depuis une session SAS.

## Conclusion

Toutes les fonctionnalités décrites dans cet article sont, à compter de SAS 9.3, incluses dans Base SAS (contre SAS/GRAPH auparavant), à savoir l'ODS GRAPHICS, le code GTL et les deux interfaces ODS Graphics Designer et ODS Graphics Editor.

Pour aller plus loin sur le sujet, la page de référence est [ODS Statistical Graphics](#) et voici quelques autres liens utiles :

[SAS 9.3 ODS Graphics](#) Getting Started with Business and Statistical Graphics

[SAS 9.3 Graph Template Language](#) Reference

[SAS 9.3 ODS Graphics Designer](#) User's guide

Enfin, un forum de discussion existe pour échanger avec les autres utilisateurs SAS : [SAS/Graph and ODS Graphics](#).

Karine Chrillesen  
Consultant Support Clients SAS France