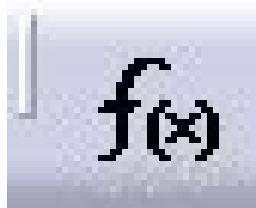


CV5-Les formules dans CatiaV5, ça sert à quoi ?

CV5

- 22h19



Sommaire [[Cacher](#)]

1

- 1.1 [Les formules dans CATIA V5](#)
- 1.2 [Attribution d'une formule à un paramètre](#)
- 1.3 [Voyons maintenant comment lier deux paramètres avec des formules dans CatiaV5](#)
- 1.4 [Conclusion sur les formules dans CatiaV5](#)

Nous avons vu dans [l'article précédent](#) que Catia V5 créait des paramètres à chaque fois que l'on créait une entité 3D.

Nous avons vu également comment créer un paramètre dit "utilisateur" c'est à dire créé par nos soins et non automatiquement par Catia.

En allant plus loin que l'utilisation d'un "simple" [modèle paramétrique](#) (c'est à dire modifiable grâce à un historique de paramètres) nous allons voir maintenant comment lier les paramètres de façon à donner une intelligence à notre conception.

Les formules dans CATIA V5

Attribution d'une formule à un paramètre

Et oui, car c'est bien là que l'on veut en venir... Créer des interactions entre les différents paramètres.

Nous aurons ainsi des paramètres pilotant et des paramètres pilotés.

Exemple:

Vous dessinez une échelle avec 4 barreaux équi-répartis sur sa longueur. Si vous voulez éviter de déplacer manuellement les barreaux à chaque fois que vous modifiez la longueur de l'échelle, vous utilisez une formule du type:

DistanceEntre2Barreaux = LongueurEchelle / (NombreBarreaux + 1)

Si vous avez pris soin de créer un paramètre **LongueurEchelle** (de type longueur) et un autre **NombreBarreaux** (de type Entier) vous pourrez alors obtenir une valeur calculée en mm de

DistanceEntre2Barreaux (du type Longueur bien sûr).

Ainsi, les paramètres pilotant seront **LongueurEchelle** et

NombreBarreaux et le paramètre piloté

DistanceEntre2Barreaux.

A ce stade, on peut d'ailleurs très bien ne rien dessiner et se servir de Catia comme d'une calculatrice en créant des variables (de type longueur, angle, pression ...) qui ne sont rien de plus que les paramètres utilisateurs.

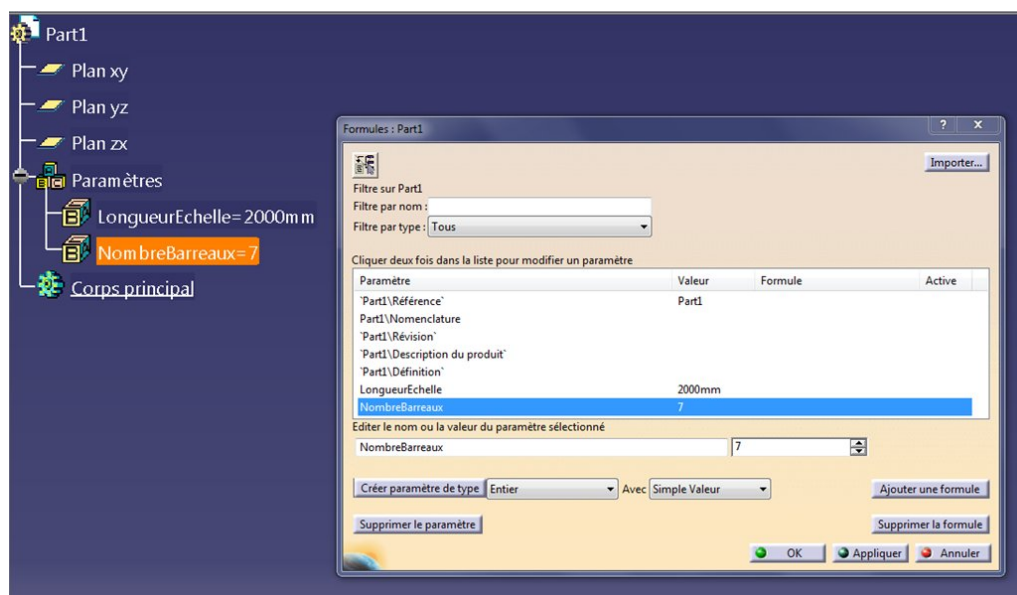
En images dans Catia:

Je crée d'abord mes deux paramètres d'entrée (les pilotant). En

prenant soin de bien choisir le type (longueur pour l'un et entier pour l'autre) avant de cliquer sur créer le paramètre.

Précision: la liste des types de paramètre est très longue et ceux qui vous intéressent se trouvent en tête de liste mais celle-ci ne s'affiche pas en montrant les premiers éléments car l'ascenseur n'est pas tout à fait en haut.

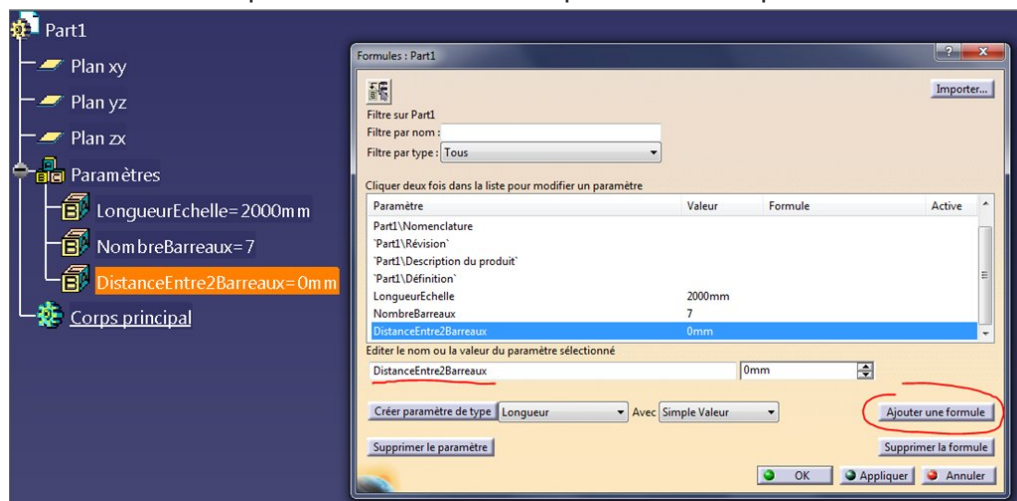
A chaque création les paramètres semblent bel et bien créés dans l'arbre. Il n'en est rien. Appuyer aussitôt sur "Appliquer" car si vous en créez 6 ou 8 et que vous cliquez malheureusement sur annuler, ceux-ci ne seront pas enregistrés dans la base et disparaîtront de l'arbre.



Il nous reste à créer le paramètre "intelligent" qui aura une valeur calculée à partir des deux autres (la distance entre chaque barreau):

On commence par créer un paramètre du type souhaité (ici longueur). On le renomme "**DistanceEntre2Barreaux**"

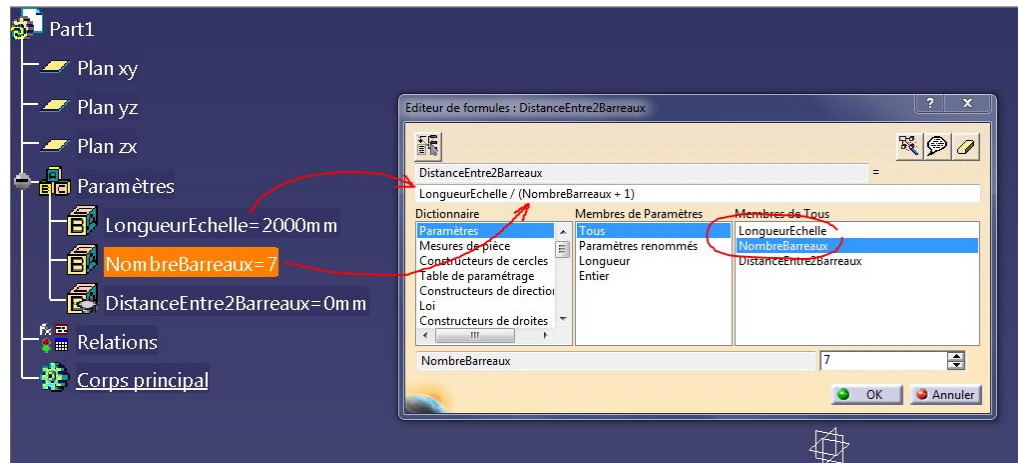
On vérifie bien qu'il est en édition (et pas un autre paramètre)



A ce stade, on pourrait simplement lui attribuer une valeur manuellement en remplacement de "0 mm". Mais il y a mieux à faire.

On clique alors sur "ajouter une formule".

Une nouvelle fenêtre s'offre à nous, c'est **l'éditeur de formules**.



En haut de cette fenêtre se trouve une égalité. On ne la voit pas très bien mais il y a un champ grisé (celui qui porte le nom du paramètre recevant la formule) et au bout du champ grisé il y a un signe égal vous invitant à donner le deuxième terme de cette égalité dans le champ vide qui se trouve juste en dessous.

C'est ici que nous allons écrire notre formule.

Inutile de partir bille en tête en recopiant soigneusement les noms des paramètres.

Non, ici, il suffit de cliquer une fois sur un paramètre (dans l'arbre par exemple) pour que celui-ci s'inscrive dans ce champ.

Nous avons juste à ajouter le signe "/", les parenthèses et notre "+1" pour que l'équation soit complète.

Une fois inscrits dans ce champ, vous pouvez vérifier la valeur de chaque variable en cliquant dessus. Dans cet exemple, le curseur est quelque part sur "NombreBarreaux" et sa valeur s'affiche en bas à droite ici 7 qui est sans unité car c'est un entier.

L'équation ne sera évaluée que lorsque l'on aura validé par Ok.

Si il n'y a pas d'erreur de syntaxe, on retourne dans l'outil "Formules" et on a une valeur de 250 mm pour ce nouveau paramètre.

Vous remarquerez que contrairement aux autres paramètres, celui-ci n'est pas modifiable directement. Le champ valeur est grisé et il existe sur la ligne la formule que nous venons de rentrer.

Si vous voulez la modifier, il faut d'ailleurs double cliquer sur la

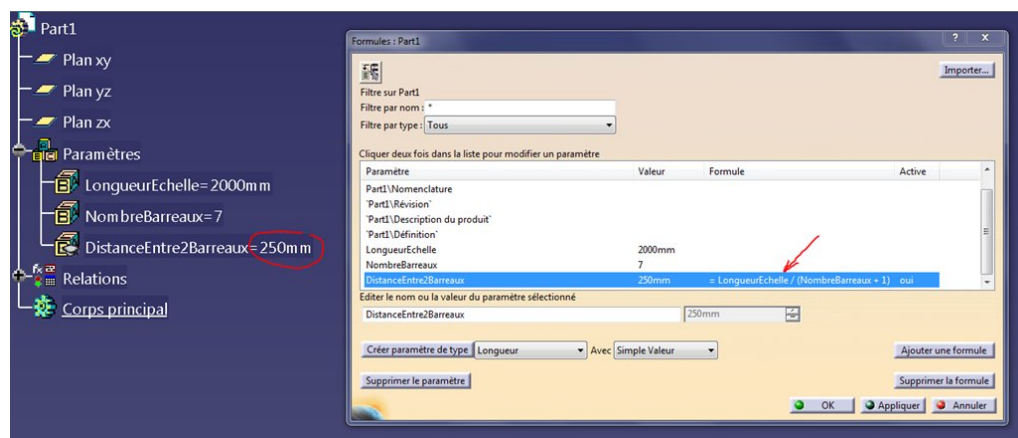
formule (flèche rouge sur l'image) afin de retrouver l'éditeur de formule.

Vous remarquerez aussi que dans l'arbre, le paramètre en question est graphiquement affublé du petit chapeau "Knowledge". Les autres n'en ont pas. C'est qu'il n'est pas modifiable. Sa valeur est issue d'un calcul (il porte une formule).

Un truc marrant et utile. Bien qu'il y ait déjà une formule, il est toutefois possible de cliquer sur le bouton "Ajouter Formule". Cela permet, non pas de modifier la formule précédente, mais d'en créer une nouvelle qui pourra devenir active à la place de l'autre (rendue inactive).

Un paramètre peut donc avoir plusieurs formules mais une seule peut être activée.

C'est pratique pour faire des essais sans rien supprimer.



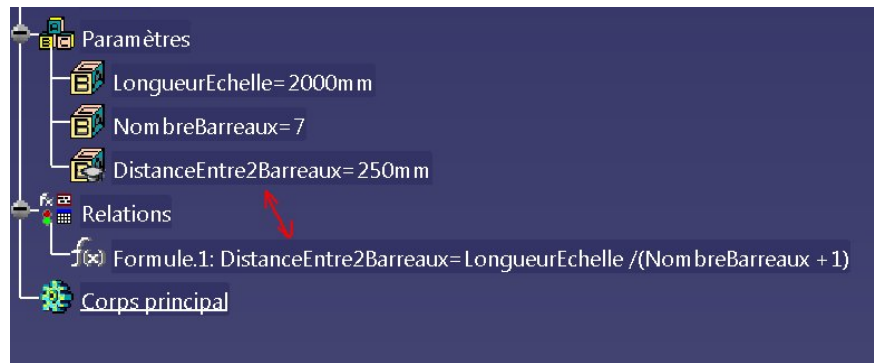
Ok, On sort de l'outil Formule.

Vous pouvez maintenant jouer en modifiant les deux paramètres d'entrée pour voir la nouvelle valeur s'afficher sur le troisième (pour que cela soit plus parlant, pensez à activer l'Update automatique, c'est à dire le bouton mise à jour manuelle non activée comme ceci...).



Vous remarquerez au passage que sous "Relations" nous retrouvons notre formule.





On se sert ici de Catia comme d'une simple calculatrice sans qu'il existe la moindre géométrie.

L'intérêt bien sûr est de pouvoir agir sur les paramètres liés aux entités géométriques elles-mêmes.

C'est ce que nous allons voir maintenant.

Voyons maintenant comment lier deux paramètres avec des formules dans CatiaV5

Nous allons maintenant affubler d'une formule un paramètre lié à une entité 3D.

Il y a trois façons majeures de réaliser cette action:

- Soit on édite le paramètre seul en double cliquant dessus puis par un bouton droit on choisit "Editer formule"
- Soit on édite l'entité 3D qui utilise ce paramètre et sur le champ valeur on fait un clic droit et on choisit "Editer formule"
- Soit on utilise l'outil "Formules"

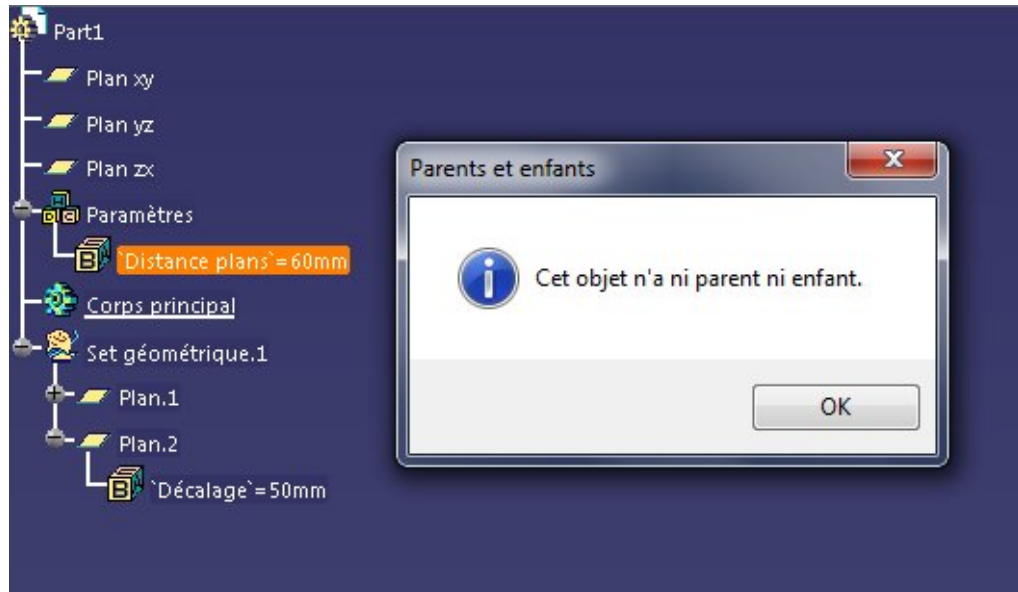
Utilisons la dernière pour être plus clair.

et reprenons les deux plans de [l'article précédent](#).

Puisque nous venions de créer un paramètre utilisateur nommé "Distance plans", je vous propose de le rendre pilotant en imposant sa valeur au paramètre de décalage du "plan.2".

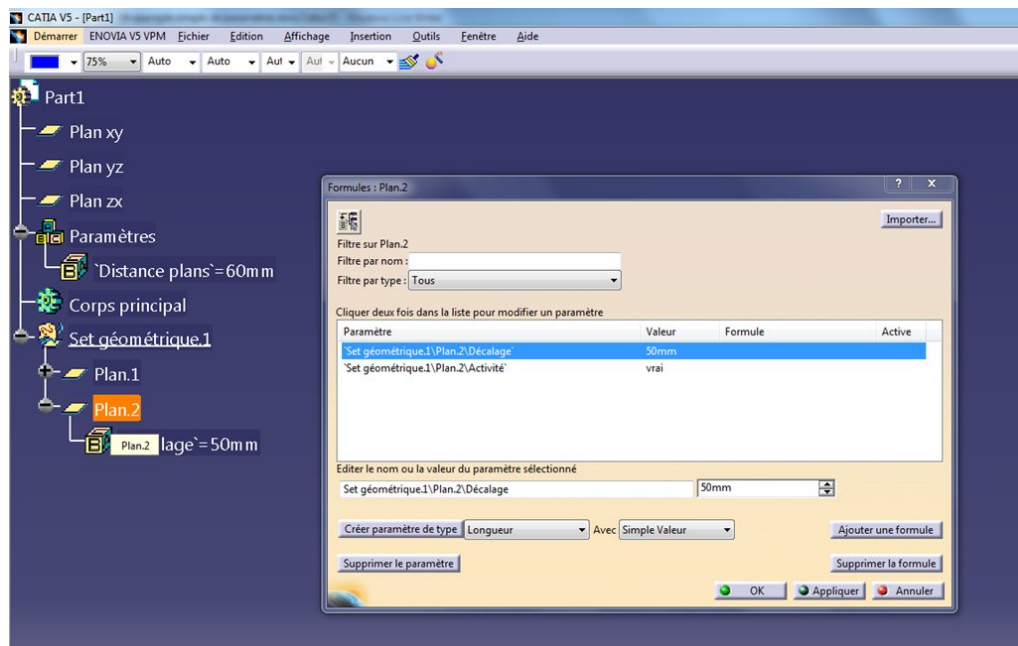
Vous pouvez déjà vérifier, si vous le souhaitez, que notre nouveau

paramètre n'a pas d'enfant, ni de parent d'ailleurs (Bouton droit sur le paramètre puis "parent/enfant").



Retournons dans l'outil "Formules" et sélectionnons le "plan.2" (dans l'arbre, c'est plus facile).

Le tri se fait et deux lignes s'offrent à nous. Ce sont les deux seuls paramètres concernant le "Plan.2".



Pourquoi ai je commencé par sélectionner "Plan.2" et non pas notre nouveau paramètre ?

Parce que c'est bien "plan.2" qui va être modifié. C'est lui qui va être impacté (piloté) par l'autre paramètre, le pilotant (paramètre utilisateur).

L'erreur fréquente au début et de faire cette confusion entre le

piloté et le pilotant et d'imposer la formule au paramètre pilotant.

On sélectionne la ligne correspondant au décalage du "plan.2" dont la valeur est actuellement de 50mm et on actionne le bouton **"Ajouter une formule"**.

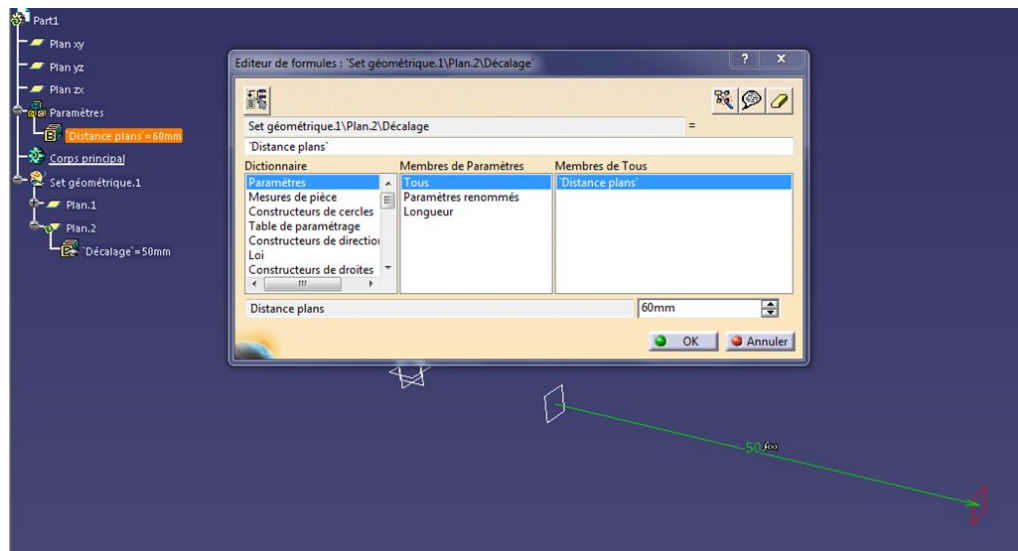
Puis on sélectionne le paramètre utilisateur ("Distance plans" dont la valeur est 60mm) comme "valeur source" pour le paramètre de décalage du "plan.2".

On a :

Set géométrique.1\Plan.2\Décalage = `Distance plans`

Nous voyons sur l'image qu'à ce stade, la valeur de décalage (en vert dans le 3D) est toujours de 50mm mais qu'il y a maintenant un petit Fx accolé.

On voit également qu'il y a le petit chapeau sur le paramètre de décalage du plan et que ce plan attend une mise à jour.



Validons par OK.

Nous retournons dans l'outil "Formule".

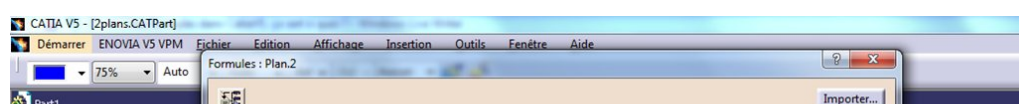
La formule est maintenant affichée sur la ligne du paramètre de décalage du plan 2.

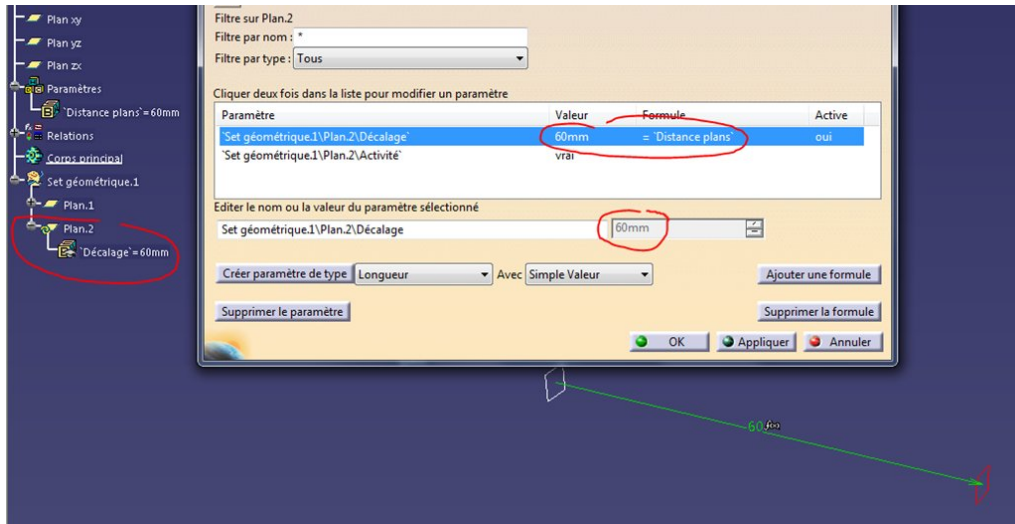
La valeur de ce paramètre est maintenant évaluée (60 mm c'est à dire la même que le paramètre utilisateur).

Mais le plan attend toujours la mise à jour et reste à 50mm.

Rappelez vous que si vous annulez à ce stade, vous perdez vos modifications.

On valide par Ok

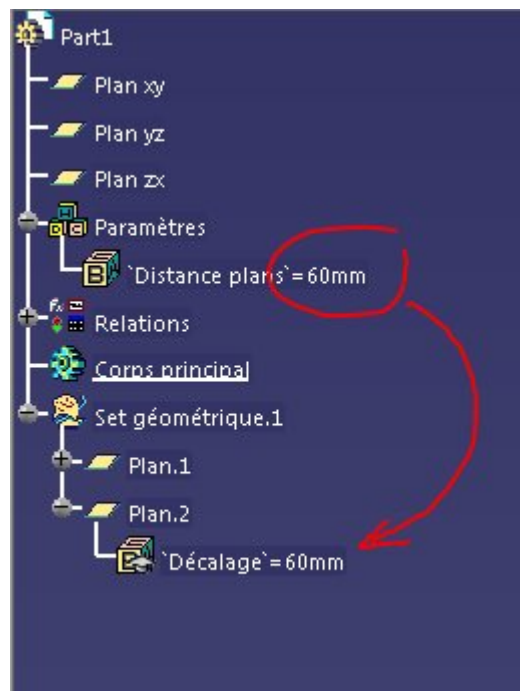




Si l'update automatique est actif, le plan se déplace de 10mm (de 50 à 60mm), sinon un ctrl+U fera l'affaire.

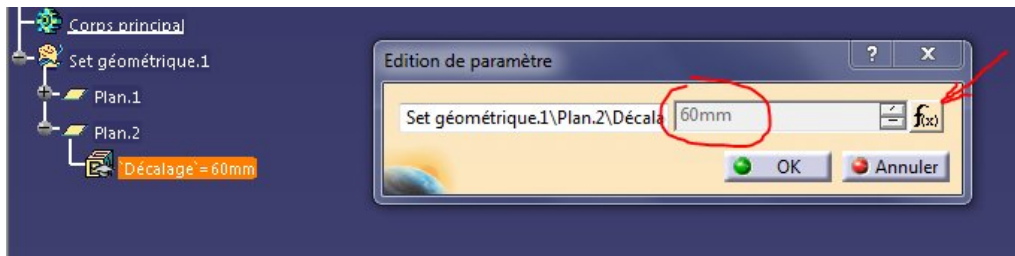
Maintenant, pour piloter le décalage du "plan.2" vous devez modifier le paramètre utilisateur "Distance plans".

A la validation cela aura donc pour effet de modifier la valeur du décalage du "plan.2"



Vous verrez que si vous voulez modifier directement le paramètre du "plan.2", cela sera impossible (le champs est grisé) car il est piloté par une formule.

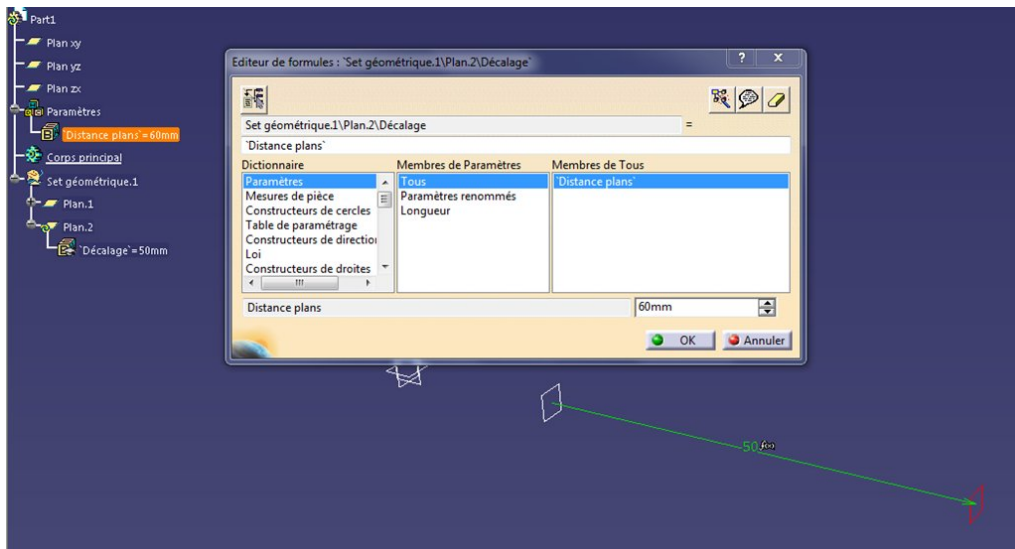




Vous pouvez alors cliquer sur le bouton Fx en bout de ce champ et retrouver l'éditeur de formules.

Il suffit alors de mettre le curseur sur le paramètre "Distance plans" dans le champs de la formule puis de modifier sa valeur en bas à droite. Cela aura pour effet de modifier la valeur du paramètre utilisateur (ici en orange) et par conséquent de modifier le décalage du "plan.2".

Il est tout de même plus aisé de modifier directement le paramètre utilisateur dans l'arbre.



Nous venons de créer un système bien pratique, dans lequel les paramètres pilotant notre géométrie sont bien rangés et nommés en tête de notre arbre.

Il est bien plus commode maintenant de modifier les grandes lignes de notre pièce (par exemple longueur, nombre de trous, épaisseur de paroi) sans à avoir à chercher dans l'arbre les paramètres auquel ils correspondent.

Cela devient vraiment intéressant, encore faut-il que cette cascade de pilotages ait un sens vis à vis de la géométrie créée.

En effet, nous pourrions très bien imaginer qu'un jeu de paramètres fasse en sorte qu'une partie d'une pièce soit finalement trop petite par rapport à une opération de perçage et que cela conduise à une erreur de mise à jour.

Une des grandes difficultés est donc de créer un modèle intelligent, certes, mais réellement robuste face à des fluctuations de valeurs sur les paramètres d'entrée.

Conclusion sur les formules dans CatiaV5

Nous avons vu que nous pouvions créer des paramètres utilisateur et créer entre eux des relations par des formules.

Nous avons vu aussi que nous pouvions lier un paramètre intrinsèque à une entité 3D avec un paramètre utilisateur explicite pour le piloter plus facilement depuis le début de l'arbre.

Tous les paramètres accessibles à travers l'éditeur de formules peuvent être couplés par des formules à la condition que les unités soient respectées, exemples:

- $\text{mm} = \text{mm}$ (les longueurs)
- $\text{N.m} = \text{N} * \text{m}$ (des moments)
- $\text{m}_s2 = \text{m} / \text{s}^{**2}$ (des accélérations)

Enfin, attention à ne pas tenter de piloter un élément par un de ses descendants. Ceci s'appelle une "**boucle de mise à jours**" et c'est un piège bien pénible pour les débutants. Cela fera l'objet d'un prochain article.

J'espère que cela vous a aidé, alors n'hésitez pas à le partager sur les réseaux sociaux.

Les commentaires et questions au sujet des formules dans CatiaV5 sont les bienvenus.

.

 No Tag

PREVIOUS POST

NEXT POST

No responses yet

Laisser un commentaire

Votre adresse e-mail ne sera pas publiée. Les champs obligatoires sont indiqués avec *

Commentaire *

Nom *

E-mail *

Site web

Enregistrer mon nom, mon e-mail et mon site dans le navigateur pour mon prochain commentaire.

LAISSER UN COMMENTAIRE

Search



Articles récents

Rhino #4 – GrassHopper ou Python, sélection d'un élément par son nom

Rhino #2 – l'indispensable manipulateur (Gumball) de Rhinocéros

Rhino #1 – C'est quoi Rhinocéros 3D ?

Rhino #3 – Une cafetière Bialetti sur Rhinocéros – Part 1

Commentaires récents

Akrim dans CV5-Utilisez-vous les lois ?

PSX59 dans Quel logiciel 3D pour mon modèle?

STEFANOVIC dans Quel logiciel 3D pour mon modèle?

Vince PSX dans CV5 – Comment faire un moletage partiel en 2 étapes par copie optimisée ?

Vince PSX dans Tutoriel débutant – Premier assemblage

Archives

juin 2019

mai 2019

avril 2019

mars 2019

février 2019

janvier 2019

décembre 2018

novembre 2018

octobre 2018

septembre 2018

août 2018

juillet 2018

juin 2018

mai 2018

avril 2018

mars 2018

février 2018

janvier 2018

décembre 2017

novembre 2017

octobre 2017

septembre 2017

août 2017

juillet 2017
juin 2017
mai 2017
avril 2017
mars 2017
février 2017
janvier 2017
décembre 2016
novembre 2016
octobre 2016
septembre 2016
août 2016
juillet 2016
juin 2016
mai 2016
avril 2016
mars 2016
février 2016
janvier 2016
décembre 2015
novembre 2015
octobre 2015
septembre 2015
août 2015
juillet 2015
juin 2015

Catégories

Calcul
CV5
Evolve
Fusion 360
Non classé
Python

[Rhino](#)

[Usinage](#)

[Vb.Net](#)

[ZW3D](#)

Méta

[Connexion](#)

[Flux des publications](#)

[Flux des commentaires](#)

[Site de WordPress-FR](#)