

# CV5- L'homogénéité des formules ou la conformité des magnitudes

## 📁 CV5

👤 - 🕒 13h14

Peut être vous êtes vous déjà lancé dans la paramétrie sous catia ? Dans ce cas, je vous félicite car c'est là un des plus grands intérêts de ce logiciel.

Votre objectif était sûrement de retranscrire une équation simple ou complexe dans laquelle il fallait mêler des variables de types différents (des mm et des N.m par exemple).

Il est alors fort possible que vous vous soyez arrêté sur le problème suivant...

### Sommaire [[Cacher](#)]

#### 1 La non homogénéité d'une formule (non conformité des magnitudes)

##### 1.0.1 Pourquoi ?

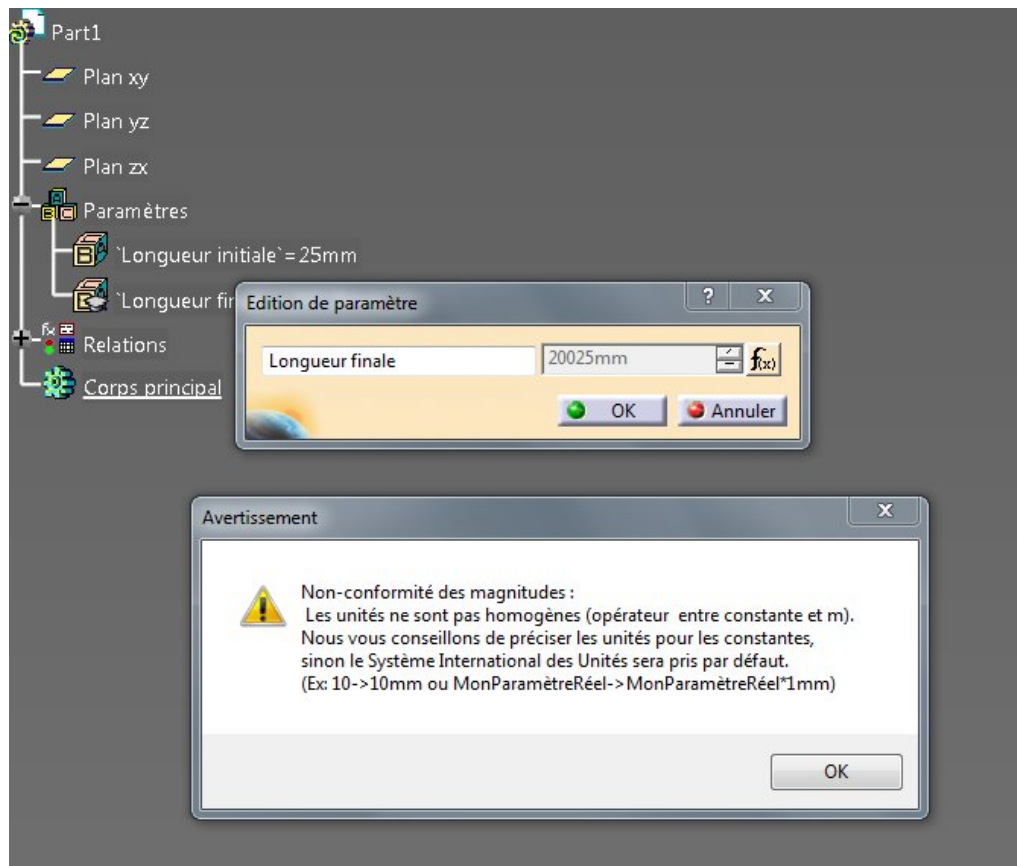
##### 1.0.2 Un autre exemple, la pression

##### 1.0.3 Changer le type ou forcer le type, c'est possible !

##### 1.0.3.1 Imposer un type à un réel

##### 1.0.3.2 Supprimer le type

# La non homogénéité d'une formule (non conformité des magnitudes)



Prenons cet exemple simple:

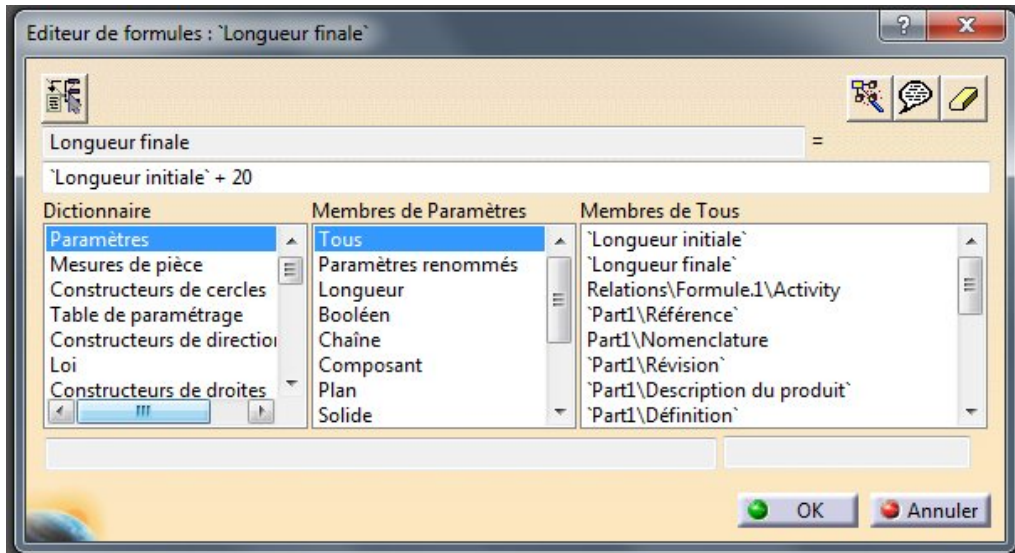
Créons un paramètre de type longueur nommé "Longueur initiale" avec une valeur de 25mm.

Puis créons un autre paramètre de type longueur nommé "longueur finale".

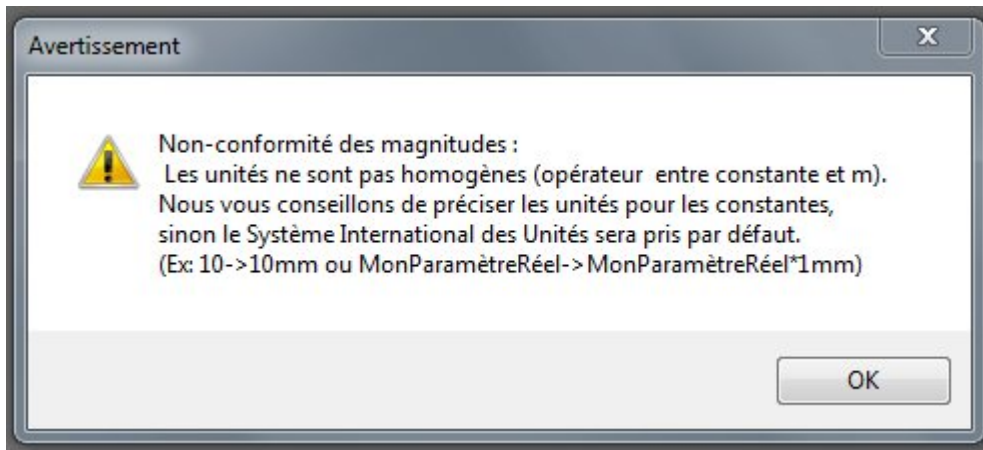
Sur ce deuxième paramètre ajoutons une formule dans laquelle vous voulez simplement ajouter 20mm à "longueur initiale".

On écrit alors cette formule qui est claire dans notre tête (on veut ajouter 20mm !)

'Longueur finale' = 'Longueur initiale' + 20



mais que cette stupide machine n'accepte pas.

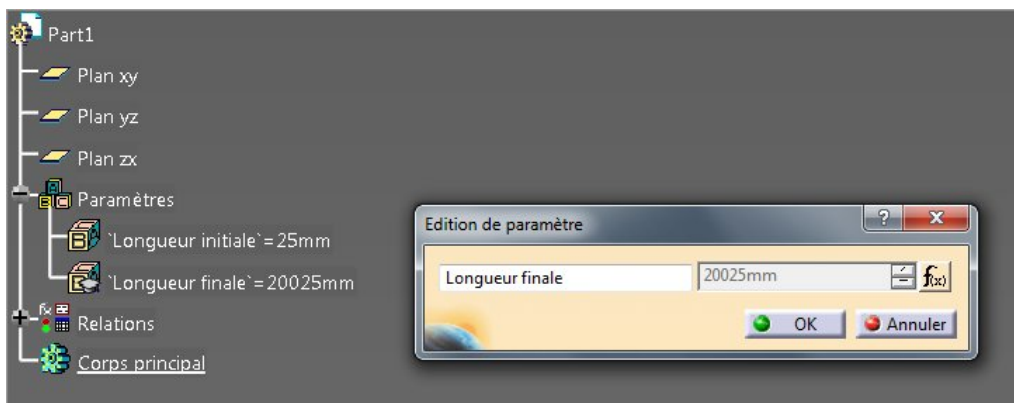


Dans la plus part des cas, Catia nous propose de désactiver la formule ou bien de revenir à l'éditeur de formule pour faire une correction.

Ici, c'est plus pernicieux car il nous donne tout de même un résultat en mm mais pas celui que nous attendions.

Nous espérions 45mm et nous obtenons 20025mm.

Le système ne vous fera d'ailleurs plus la remarque, alors attention restons attentif.



# Pourquoi ?

En fait, c'est Catia qui a raison.

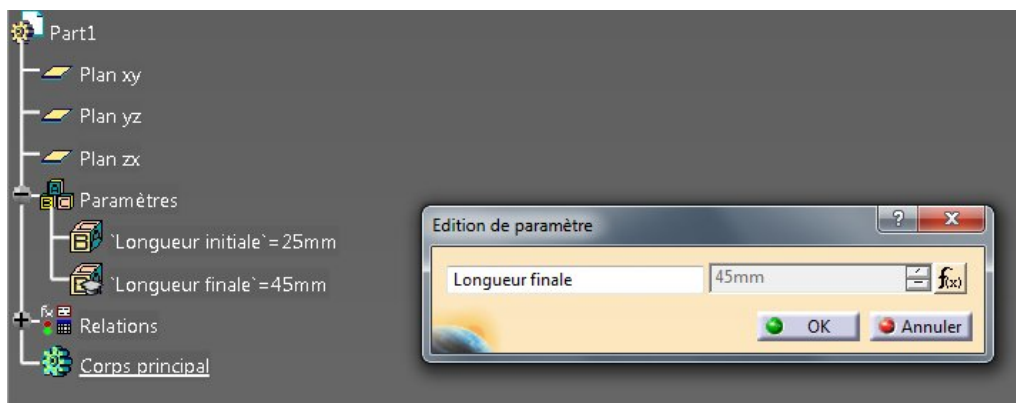
Il faut veiller au fait que la **combinaison** des différents types (longueur, angle, vitesse...) contenus dans l'équation (variables et constantes subissant toutes sortes d'opérations) conduise au **même type** que le paramètre portant l'équation.

Ici le paramètre "Longueur finale" est du type Longueur (on attend une valeur de résultat en mm) or dans notre formule nous avons une addition de deux types **différents**: Longueur initiale (**des mm**) et 20 (un entier = **sans dimension**).

**Ceci n'a pas de sens.** On ne peut additionner des mm avec autre chose que des mm.

Corrigeons la formule dans ce sens:

'Longueur finale' = 'Longueur initiale' + 20mm



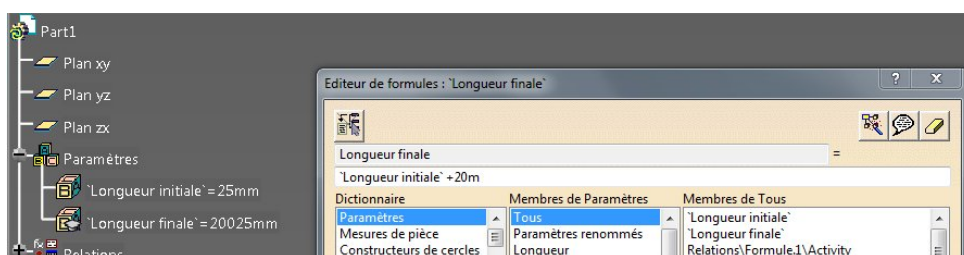
qui nous donne le résultat attendu soit 45 mm !

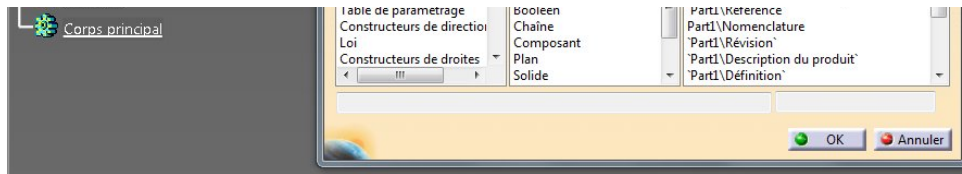
Nous avons simplement **imposé** un **type longueur** à la valeur 20 en précisant qu'il s'agissait de **mm**.

L'addition est donc juste mm+mm de même dimension qui conduit à une longueur (ici en mm).

Et si nous avons mis 20 m au lieu de 20mm ?

Essayons,





Nous obtenons 20025 mm !

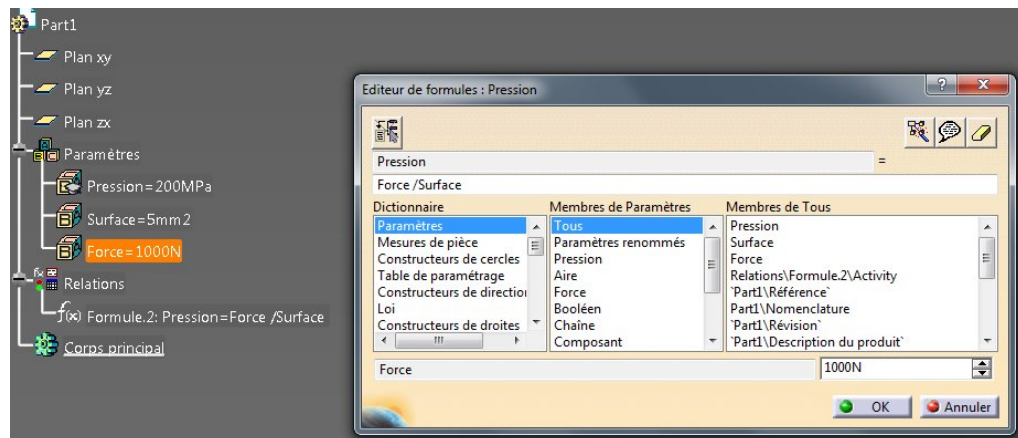
Oui 20m c'est bien 20000mm plus les 25mm...

20025 cela vous rappelle le problème de départ, non ?

Il y a deux conclusions à tirer de cela:

- Puisqu'il n'y a pas d'erreur, Catia accepte de mélanger des m et des mm. C'est pratique)
- Par défaut, Catia nous alerte mais cherche à convertir le nombre sans unité dans le type du paramètre résultant (ici une longueur) mais maladroitement en prenant l'unité de base le mètre d'où le résultat 20025 mm.

## Un autre exemple, la pression



Pression = Force / Surface

Pour être homogène dans ce cas il faut que le paramètre résultant (Pression) soit bien du type pression (au moment de la création) et que les deux autres soient bien du type "force" et "aire".

Oui mais les unités me direz vous ?

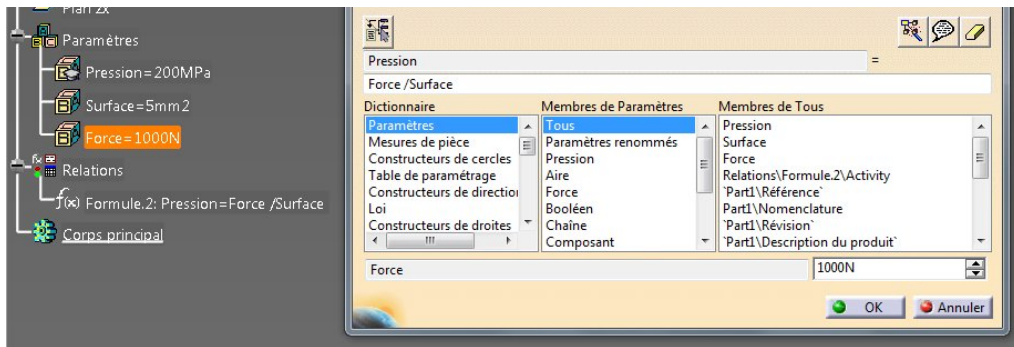
Je renforce ce que je viens de dire au dessus.

C'est vrai, si vous faites le calcul à la main, il faut faire attention aux unités en ayant par exemple une pression en Méga Pascal (Mpa), une force en Newton (N) et une surface en millimètre carré (mm<sup>2</sup>).

MPa = N / mm<sup>2</sup>



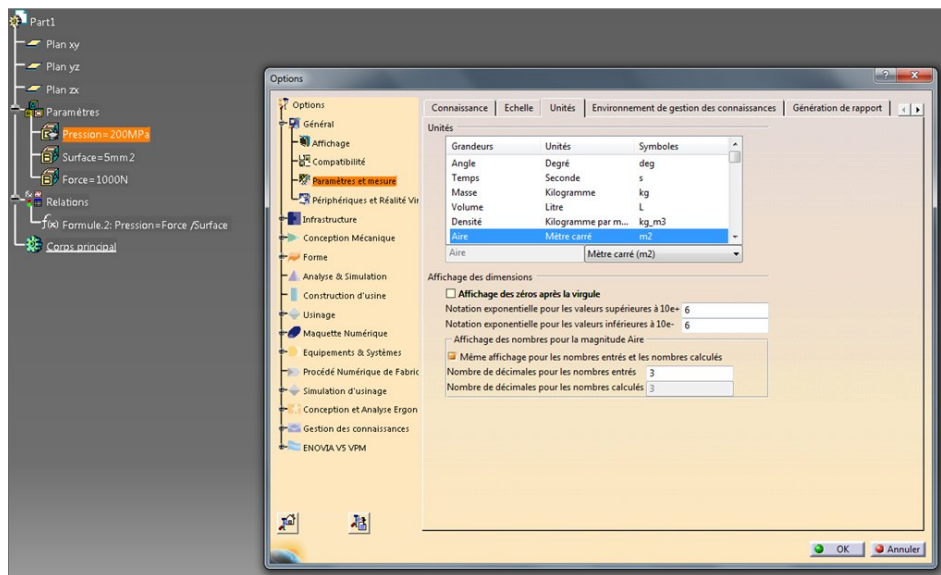




Cela fonctionne parfaitement bien.

Si l'on change d'unité pour l'un d'entre eux, cela fonctionne t'il toujours ?

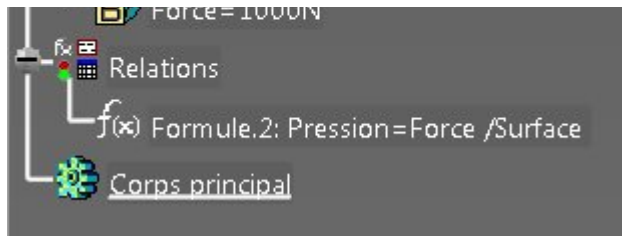
Allons dans Outil/options puis dans paramètres et mesure et dans l'onglet unité et changeons par exemple l'unité de surface en m<sup>2</sup> (mm<sup>2</sup> devient m<sup>2</sup>).



Regardons l'arbre... (un update manuel si nécessaire) ... la pression n'a pas changé (toujours 200MPa).

La surface elle, a vu sa valeur se convertir à la nouvelle unité sans que nous ayons à intervenir.





Ceci est donc très pratique. Nous n'avons pas à nous soucier de l'unité utilisée du moment que le type est respecté. Une aire (ou une surface comme vous voulez) quelle soit exprimée en mm<sup>2</sup>, en m<sup>2</sup>, en square inch ou en km<sup>2</sup> est toujours du même type et Catia fera la conversion pour vous.

Que se passe t'il maintenant si l'on se trompe dans la formule en écrivant une grosse ânerie:

Pression = Surface / Force !???

Faites l'essai !

Cela conduit à notre fameux message d'erreur "**Non conformité des magnitudes**" !

Normal puisque Catia manipule bien plus que des valeurs, il manipule des dimensions (les types) et comme nous venons de le voir une pression dont l'unité de base est le Pascal (1N/m<sup>2</sup>) est bien un force divisée par une surface et non l'inverse.

### **Conseil:**

Si vous ne vous en sortez pas avec votre formule je peux vous donner deux astuces:

- Faites sur le papier [l'équation aux dimensions](#) de votre formule et vérifiez que chaque entrée soit bien du bon type. Cela doit fonctionner
- Passez par plusieurs étapes en créant des paramètres intermédiaires. Si vous avez une formule constituée de plusieurs membres (au numérateur et au dénominateur) fractionnez la en créant un paramètre numérateur et un autre dénominateur puis faites un dernier paramètre utilisant ces deux derniers (une simple division).

## Changer le type ou forcer le type, c'est possible !

## Imposer un type à un réel

Oui, on peut par exemple dans une formule imposer un type souhaité à un paramètre de type réel (c'est à dire sans unité).

Comment:

Soit "ParamReel" le paramètre de type réel de valeur 58.3 (pourquoi pas) -> donc sans unité (on est d'accord ?)

Dans une formule je peux l'utiliser comme si il était un facteur d'une valeur unitaire d'un autre type.

Exemple:

**"ParamReel" \* 1deg** ce qui veut dire **58.3 \* 1deg** ce qui donne **58.3deg** (un angle !)

58.3 est donc un facteur multiplicateur de l'angle unitaire.

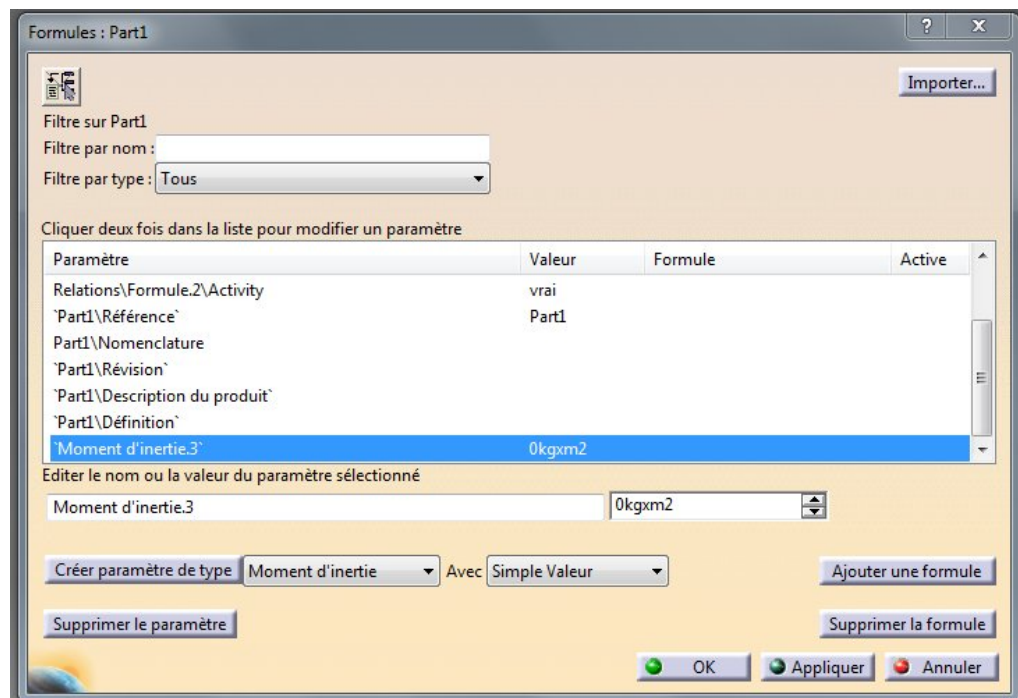
Cela fonctionne avec n'importe quel autre type, encore faut il les connaître:

**s** pour seconde, **mm** pour millimètre, **m\_s2** pour une accélération, etc...

Pour savoir comment écrire l'unité, vous pouvez créer un paramètre que vous détruirez dans la foulée.

Comment écrit on l'unité d'un moment d'inertie ?

On crée le paramètre en choisissant le type moment d'inertie



Ca s'écrit **kgxm2** !

Je le note et je détruis le paramètre.

Dans ma formule, je peux mettre **"ParamReel" \* 1kgxm2** si ça



m'arrange.

Bon, ça ne sert pas souvent mais c'est bon à savoir.

Par contre ce qui sert un peu plus c'est cela.

## Supprimer le type

Un peut dans le même esprit, il suffit de diviser par la même unité.

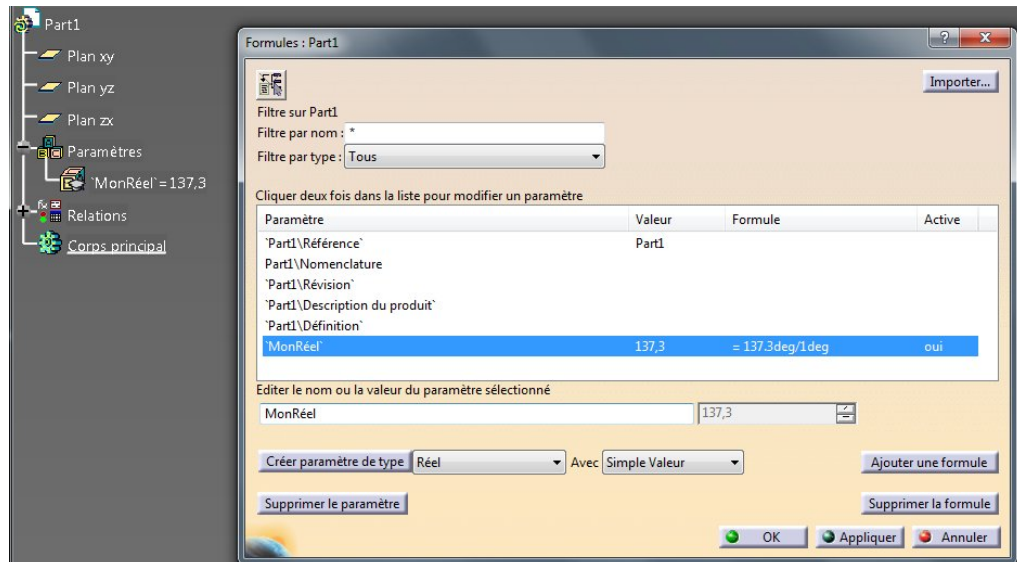
Exemple:

Je veux **recupérer la valeur** (donc un réel) de cet angle **137.3deg**

Je crée un paramètre de type réel et lui colle une formule du type:

MonRéal = 137.3deg / 1deg

L'équation aux dimensions dit Angle/angle -> cela s'annule, il n'y a plus de dimension et Catia est content.



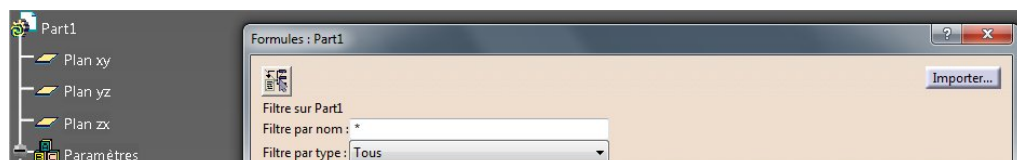
## Changeons alors le type

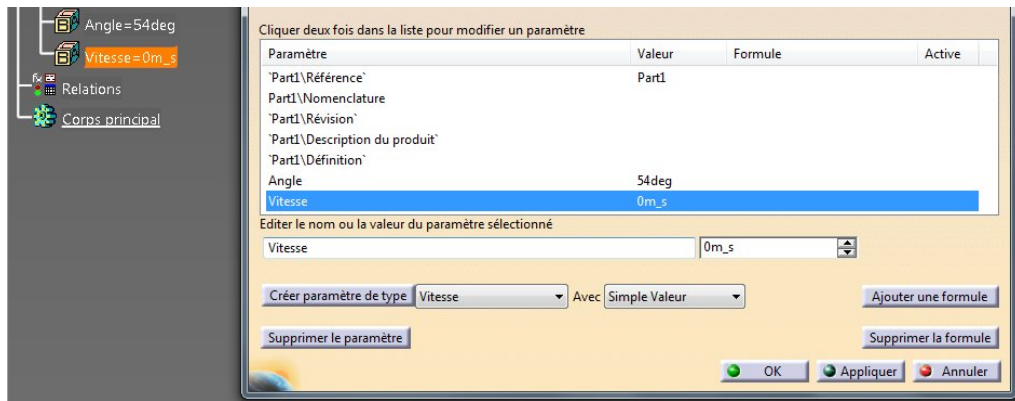
C'est simplement la combinaison des deux premiers cas.

On supprime le type et on impose un autre type.

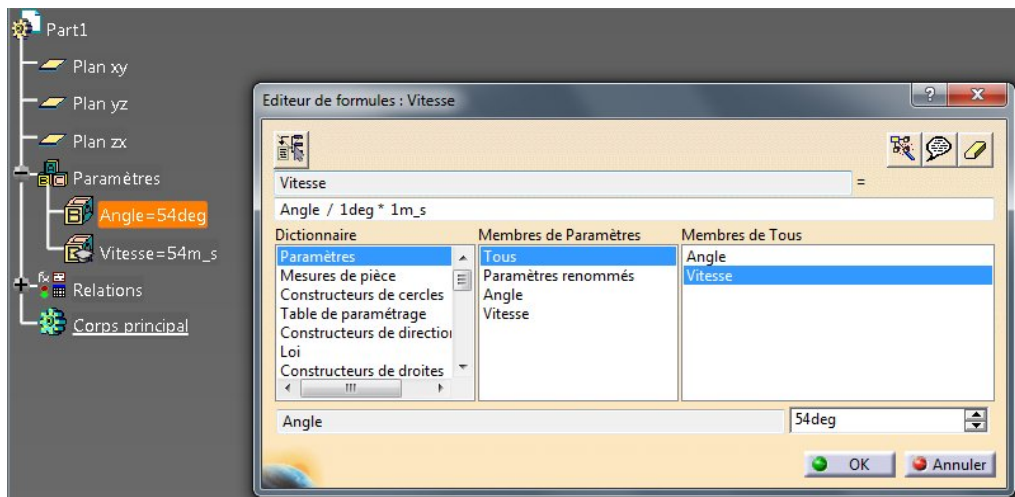
Je pars par exemple d'un paramètre angulaire et je veux récupérer sa valeur et en faire un vitesse.

- Je pars d'un paramètre angle
- Je crée un paramètre vitesse





- J'ajoute une formule à mon paramètre vitesse  
 première étape, je divise par 1deg pour obtenir une valeur sans dimension  
 deuxième étape, je multiplie par 1m\_s pour obtenir une vitesse  
 Résultat un paramètre de vitesse dont la valeur est liée à celle de l'angle.



Voilà pour les explications.

Si cet article vous a aidé à mieux comprendre Catia, j'en suis heureux alors n'hésitez pas à le partager.

Et si vous souhaitez plus de précision à ce sujet, laissez moi un commentaire en bas de page.

[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](#)