

# ParaView - Démo/TP

- 1) Prise en main
- 2) Le pipeline et les filtres
- 3) Explorer, sélectionner les données
- 4) Animations temporelles et spatiales
- 5) Automatisation avec les macros et les scripts python
- 6) Connections distantes (3me partie du cours)

# Pour démarrer

- **Copier en local les jeux de données** qui sont sur froggy dans le répertoire `/scratch/PROJECTS/pr-formation-ced-2017/visu/data/`. Par exemple, à partir d'un répertoire où on veut placer les données :

```
$ scp -r monlogin@froggy.ciment:/scratch/PROJECTS/  
pr-formation-ced-2017/visu/data/ .
```

Ou: récupérer le jeu de données sur le projet Demos de gitlab

- **Lancer Paraview** en ligne de commande sur le poste local :

```
$ paraview
```

- **Lancer pvpython** en ligne de commande sur le poste local :

```
$ pvpython
```

Note : utiliser `exit()` pour quitter python.

# Prise en main - I

- Ouvrir Paraview, menu File/Open, ouvrir le fichier bluntfin.vts. Regarder le nombre de format de fichier lus (Files of type)
- Dans le panneau 'Properties', section 'properties', cocher toutes les variables puis cliquer sur 'Apply'
- Dans le panneau 'Information', retrouver des informations sur le dataset
- Dans les raccourcis des options d'affichage :

modifier les valeurs des 2 listes déroulantes : la variable à afficher (Solid Color, Density, ...) ou le type de représentation (Outline, Wireframe, Points, Surface, ...)

- Tester les mouvements de la caméra (voir tableau sur le slide suivant)
- Pour aller plus loin : afficher et modifier le centre de rotation des données



# Les mouvements de la caméra

Reset : 



	<b>souris mac</b>	<b>souris molette</b>	<b>trackpad mac</b>
<b>Zoom</b>	molette/bton droit	molette/bton droit	molette / cmd click
<b>Tourner autour</b>	bton gauche	bton gauche	click
<b>Pivoter</b>	maj + bton gauche	maj + bton gauche	maj + click
<b>Translation</b>	maj + bton droit	click molette/maj + bton droit	maj + click 2 doigts

# Prise en main - La Color Map



La **Color Map** sert à traduire des valeurs scalaires en couleur et transparence via une fonction de transfert. C'est l'objet **Lookup Table** de VTK.

L'effet visuel est très dépendant des valeurs choisies pour cette lookup table

- Choisir comme mode de représentation 'surface', avec par exemple, la densité
- Utiliser le raccourci  pour afficher l'éditeur de la color map.
- Tester d'autres jeux de couleur préexistant : 
- Activer/désactiver la transparence («enable opacity »)
- Tester la modification de la fonction de transfert pour les couleurs et l'opacité.
- Diminuer le nombre de couleur (à 10 par ex)



(Des bonnes explications - en anglais - ici <https://blog.kitware.com/using-the-color-map-editor-in-paraview-the-basics/>)

# Prise en main - Organisation des écrans






Les panneaux de contrôle, accessibles via le menu « View »

- Le panneau **properties** : Bien distinguer les 3 parties (on peut les replier): Properties, Display, View. Retrouver les raccourcis utilisés en I. Afficher les axes.
- Afficher/enlever des panneaux. Par exemple, memory inspecteur, color map éditeur, ...

Les « layouts » : Représentation des données, sous différentes formes, appelées 'Views' comme pour les panneaux de contrôle —> Attention à la confusion.

- Ajouter une vue via un nouvel onglet : cliquer sur le +  et choisir la vues 'SpreadSheet View' (la dernière) - ou la Slice View
- Ajouter une vue, sur le même écran : utiliser (  ) et ajouter une Render view.
- En + : Lier les caméras des rendus d'un layout : Click bouton droit souris sur une des 2 images, 'Link Camera' puis cliquer sur la deuxième image.


# Filtres : Slice, Contour (isosurface)

- Ouvrir le fichier disk\_out\_ref.ex2 - Chercher la température et la pression. Display mode : surface + temp
- Cliquer sur le filtre 'Slice' dans les raccourcis (  ) puis sur 'Apply'
- Afficher/Masquer des éléments du pipeline en cliquant sur l'oeil 
- Sélectionner 'disk\_out\_ref', puis cliquer sur le filtre 'Contour'  .  
Properties panel : Contour By --> Temp ; isosurface = 400. Apply.
- Revenir au panneau 'Properties', et choisir plusieurs valeurs d'isosurface ('Add a range of values'  ; cocher l'option compute scalars)
- Sélectionner 'Contour I' dans le pipeline browser, et appliquer le filtre 'Clip' . Désactiver l'option 'Show Plane' 
- En + : tester avec le jeu de couleur black body radiation. Tester le rendu volumique avec ces couleurs.

# Filtres - Stream tracer

# Glyph

Toujours à partir du fichier source disk\_out\_ref.ex2, utiliser le filtre stream tracer pour obtenir un rendu similaire aux 2 slides suivants.

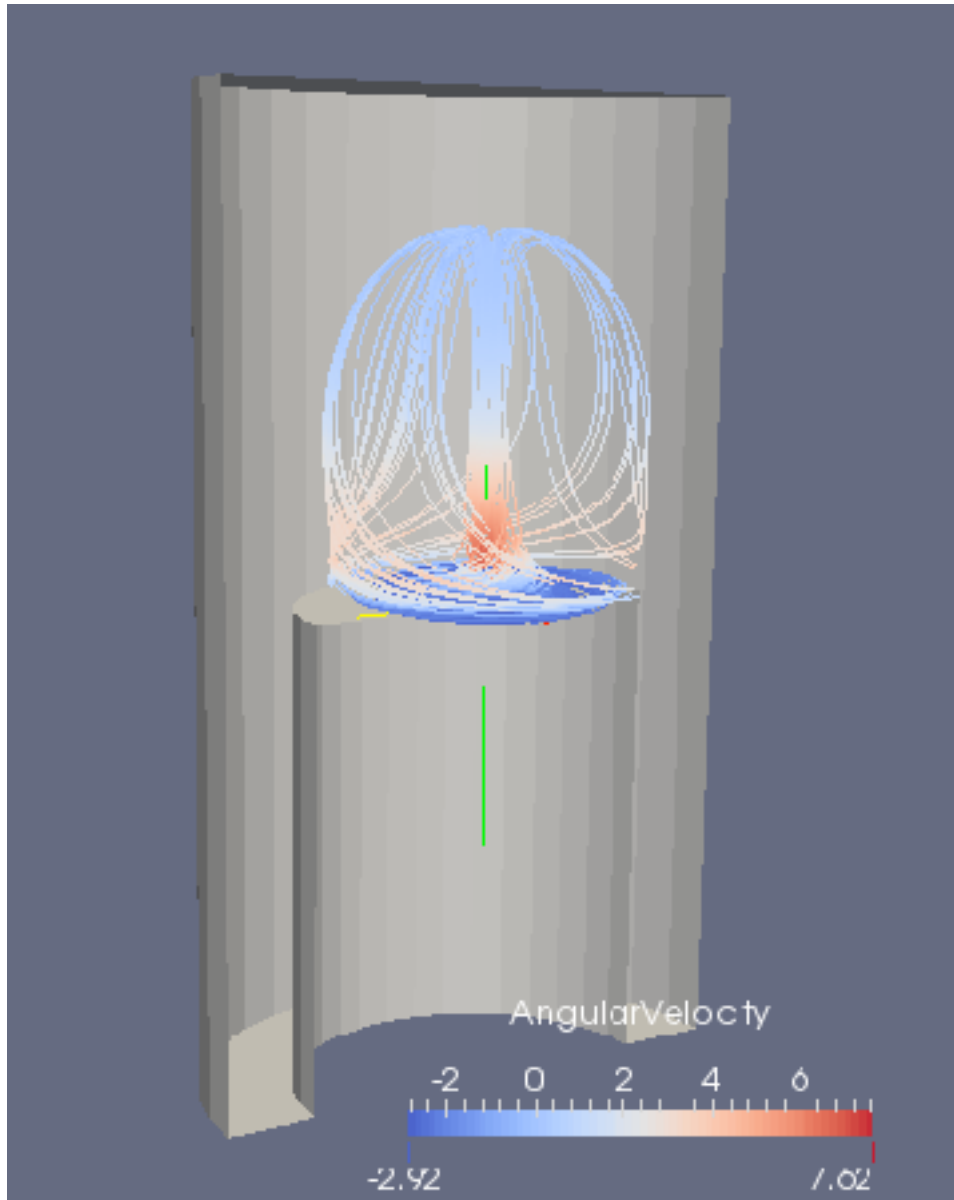
Le stream tracer  trace des lignes de champs à partir d'un nuage de points (seeds, dans les propriétés), groupés autour d'un point source ou d'une ligne. Vous pouvez modifier les paramètres suivants pour test : les coordonnées du point source, le nombre de point sources, utiliser une ligne source à la place d'un point

Pour l'utilisation de **Glyph**  : dans propriétés, choisir le vector V, et comme Glyph Type le cône

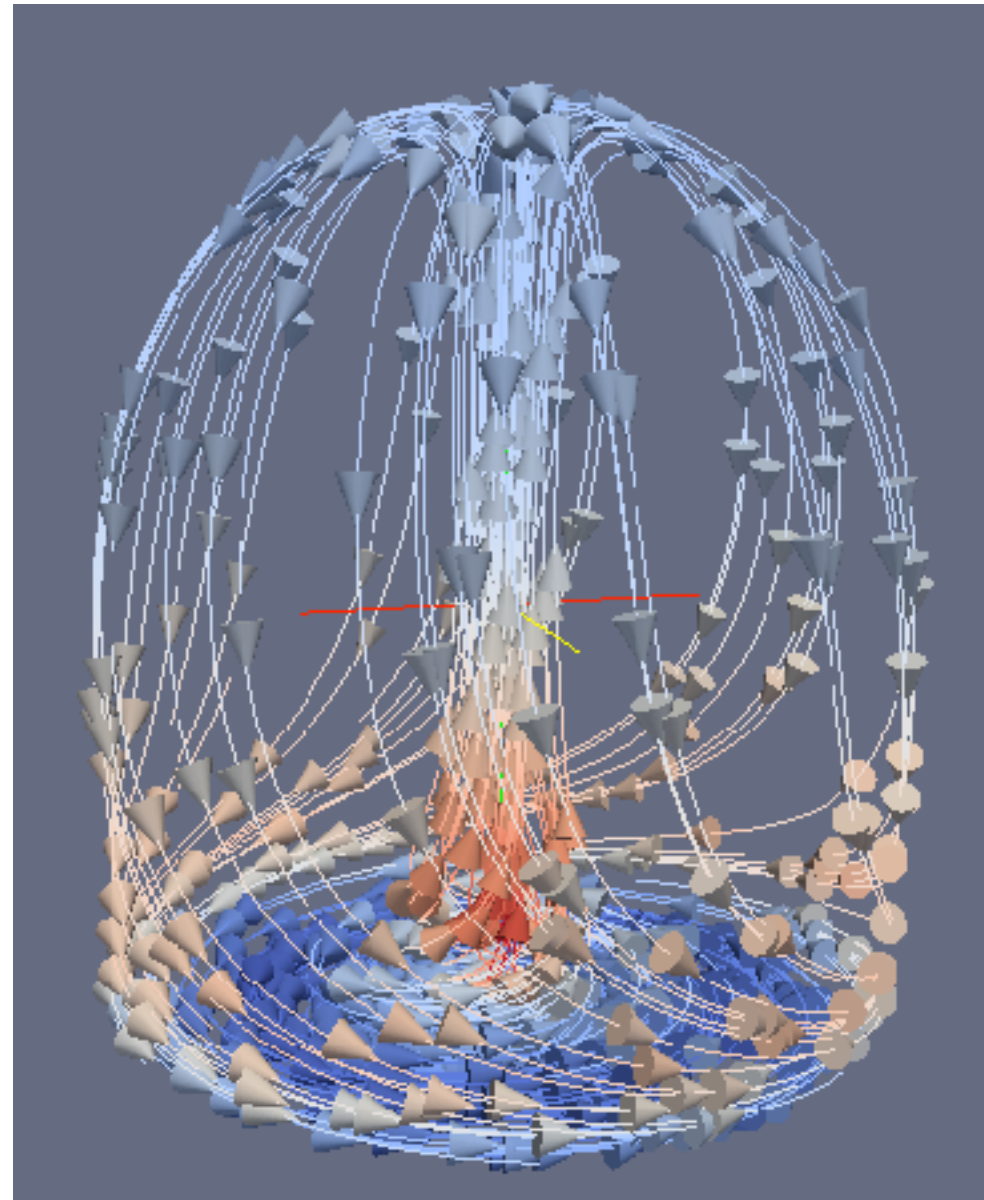
Pour le filtre 'mask point' : essayer 10 comme ratio par ex



# Filtre - Stream Tracer

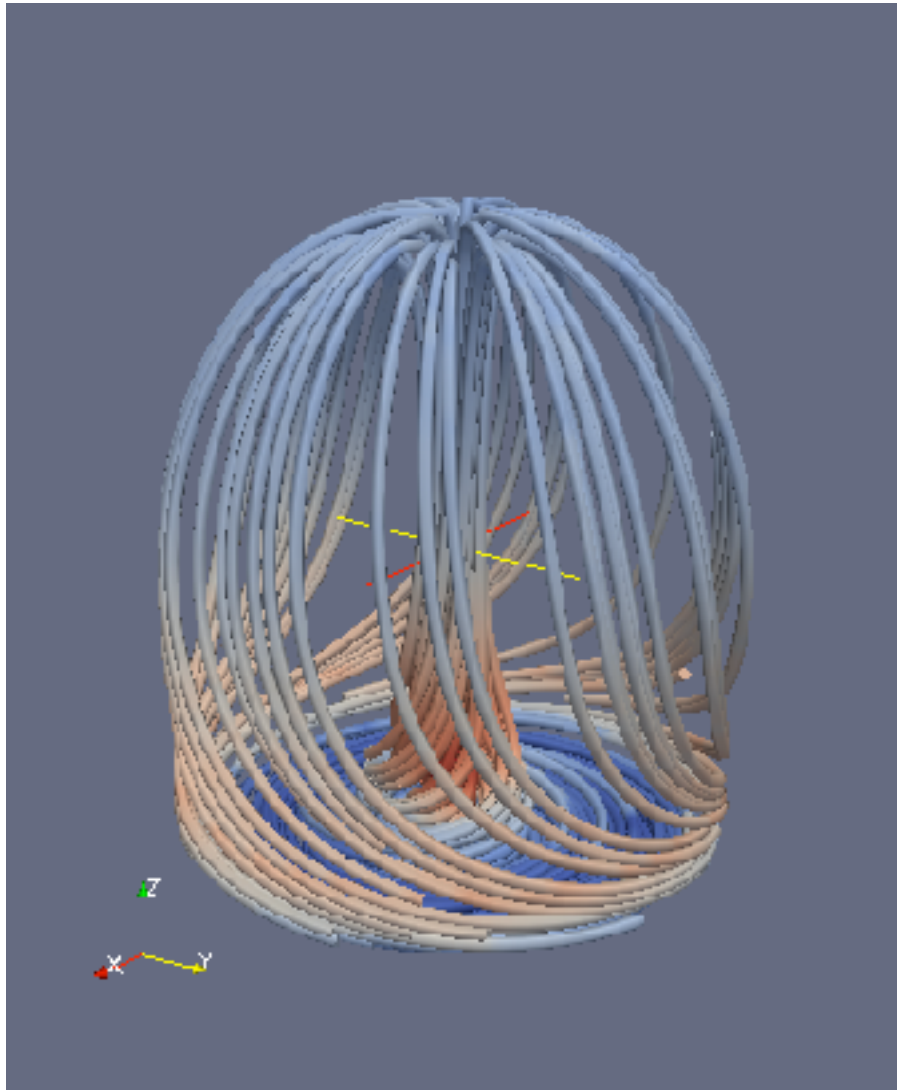


Stream Tracer  
+ Extract Surface  
+ Clip

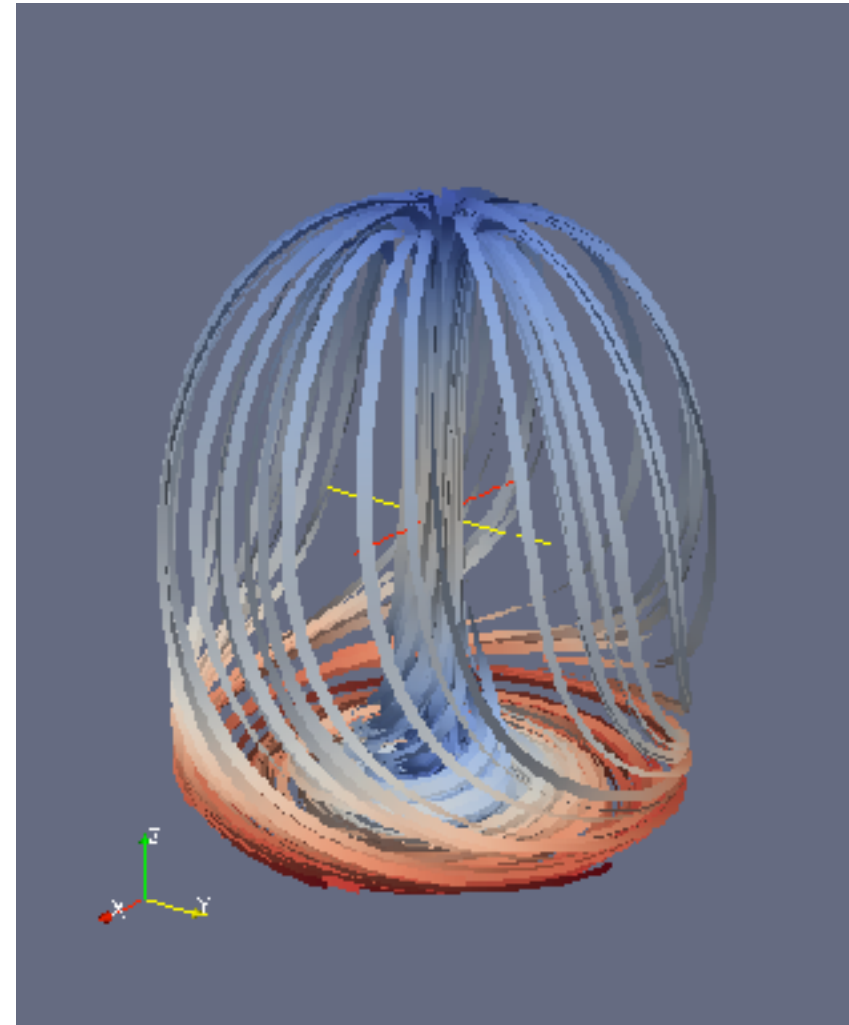


Stream Tracer  
+ Mask Point  
+ Glyph

# Exemples de filtres

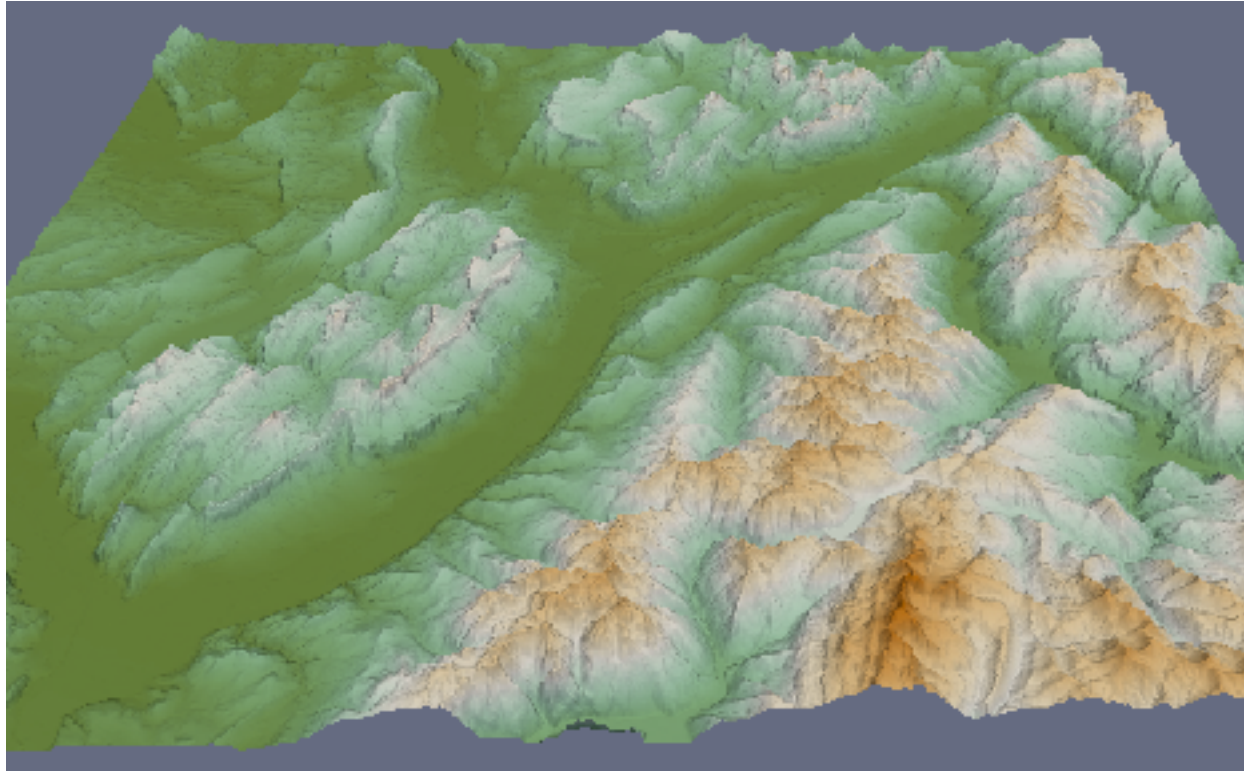


Stream Tracer  
+ Tube



Stream Tracer  
+ Ribbon

# Exemples de filtres

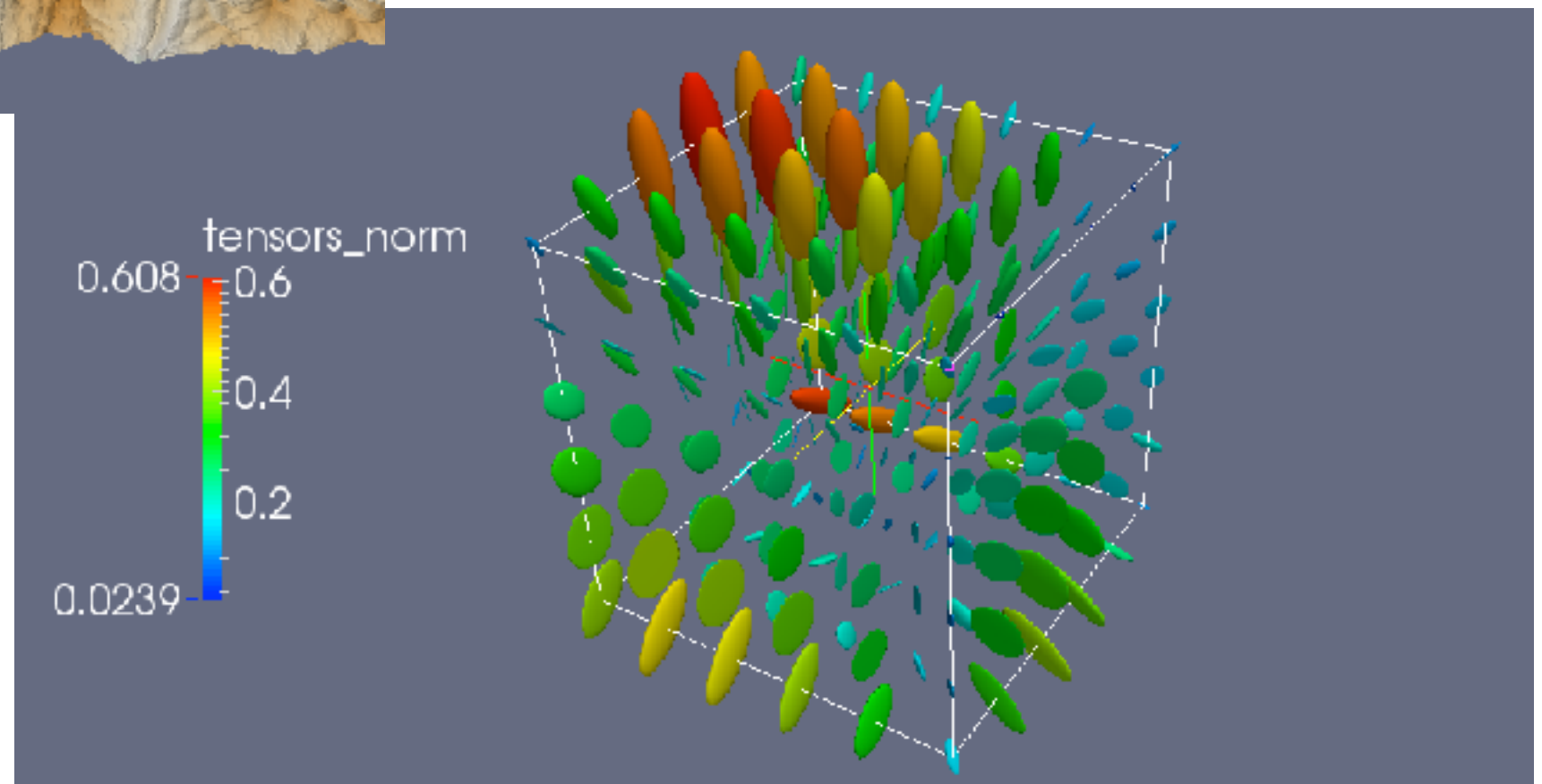


## Warp by scalar



Données initiales : image 2D  
avec niveau de gris fonction  
de l'élévation

## Tenseur

Source des données : Rheolef



# Filtres - Autres

- **Calculator**  : Utiliser ce filtre pour visualiser la valeur de la normale du vecteur vitesse ( indication : Le champ de la calculette doit contenir  $V_X^2 + V_Y^2 + V_Z^2$  ; par défaut la valeur résultat est stockées dans le champ 'result')
- Essayer le filtre Plot over Line 

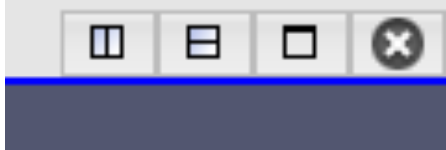
# Les filtres - quelques remarques



- Menu Filter/Alphabetical --> Liste de tous les filtres. La fonction de recherche est pratique.
- On trouve les filtres principaux dans la barre d'outil



- Un filtre peut changer la nature d'un dataset (image data --> polygonal mesh). Ca peut être le but recherché (ex : clean to grid)
- Un filtre s'applique généralement à certains types de dataset.
- Certains filtres dupliquent les données (En général, quand la nature du dataset change). --> Attention à la mémoire !
- Le paramétrage des filtres peut être complexe
- [http://www.paraview.org/Wiki/ParaView/Users\\_Guide/List\\_of\\_filters](http://www.paraview.org/Wiki/ParaView/Users_Guide/List_of_filters) (183)

# Exploration, sélection des données

- Repartir du fichier dist\_out\_ref.ex2, affichage : Wireframe
- Accès aux valeurs : ajouter une vue via le + à côté de 'Layout #1' ou un des boutons  et sélectionner 'Spreadsheet view'

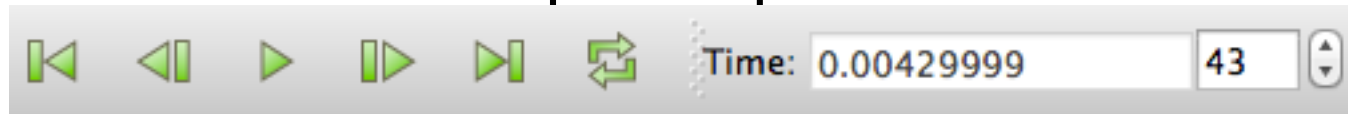
-  Sélection des données via la vue 3D. Dans la spreadsheet view, activer l'option 'Show only selected elements' Tester différentes sélection dans la vue 3D et leurs impact sur la spreadsheet view 

- Utiliser une requête pour sélectionner des données : **menu Edit/Find Data**. Exemple : temp compris entre 360 et 400
- Pour afficher des données à propos d'une sélection (ex, valeur de la température à un point) : utiliser le 'Selection Display Inspector' (menu 'View')



# Animations et films

- Ouvrir le fichier can.ex2
- Tester l'animation temporelle par défaut en utilisant les raccourcis:



- Ouvrir l'Animation View : Menu 'View'/'Animation View'
- Dans la list-box en bas sélectionner 'Caméra', puis cliquer sur le + pour ajouter une animation. Laisser les autres options par défaut (ou pas). Décocher l'animation 'Time Keeper - Time', et relancer l'animation.
- Pour enregistrer une animation : menu 'File' / 'Save Animation'
- Autre type d'animation : Variation de contour, déplacement d'objet le long de ligne de champ, ...

# Automatiser des actions dans l'interface graphique

## 1. **State files**

- Sauvegarder une visualisation (= pipeline) : File/Save state.
- Effacer le pipeline, puis recharger la visu avec File/ 'Load state' (Ligne cde : `paraview --state=my-state.pvsm`)

## 2. **Macros** (= script python)

- Enregistrer une série d'actions sous forme de macro (script python) :
  - Menu Tools / Start trace
  - Exécuter des actions (ajouter un filtre par ex)
  - Tools / Stop trace. Sauvegarder le script qui s'affiche. (File / save as ...)
- Effacer les actions effectuées
- Utiliser le menu Macro pour ajouter la macro (add new macro) puis lancer la macro



# Script python - Génération automatique d'image

**pvpython** : console python avec les environnements VTK et Paraview disponibles. Permet d'utiliser des scripts plutôt que l'interface graphique de paraview.

On peut s'inspirer du code généré par les macros pour construire les scripts :

- Générer une macro qui ouvre un fichier, crée un filtre, exporte une image (Export d'une image : Menu 'File' / 'Save Screenshot')
- Modifier éventuellement le nom de l'image de sortie dans la macro
- Lancer `pvpython myMacro.py` pour générer l'image sans passer par l'interface graphique

Note : Suivant la version de paraview, pour avoir le même background avec `pvpython` qu'avec l'interface graphique il faut ajouter les lignes suivantes au script :

```
# lights
RenderView1.LightSwitch=0 # turns off the headlight
RenderView1.UseLight=1 # turns on the ambient lighting
# background
view=GetActiveView()
view.Background = [0.32,0.34,0.43]
```

# Paraview - Ressources

- Télécharger le code et/ou des jeux de données : <http://www.paraview.org/download/>
- Télécharger le tutoriel et ses données : [http://www.paraview.org/Wiki/The\\_ParaView\\_Tutorial](http://www.paraview.org/Wiki/The_ParaView_Tutorial)
- wiki : <http://www.paraview.org/Wiki/ParaView>
- Paraview python : [http://www.paraview.org/Wiki/ParaView/Python\\_Scripting](http://www.paraview.org/Wiki/ParaView/Python_Scripting) ou <http://www.paraview.org/ParaView3/Doc/Nightly/www/py-doc/index.html>
- User guide - payant depuis la dernière version. Table des matières de l'ancienne version : [http://www.paraview.org/Wiki/index.php?title=ParaView/Users\\_Guide/Table\\_Of\\_Contents&oldid=51626](http://www.paraview.org/Wiki/index.php?title=ParaView/Users_Guide/Table_Of_Contents&oldid=51626)
- Programmable filter (Permet d'insérer du code python coté serveur pour modifier les données) : [http://www.paraview.org/Wiki/Python\\_Programmable\\_Filter](http://www.paraview.org/Wiki/Python_Programmable_Filter)