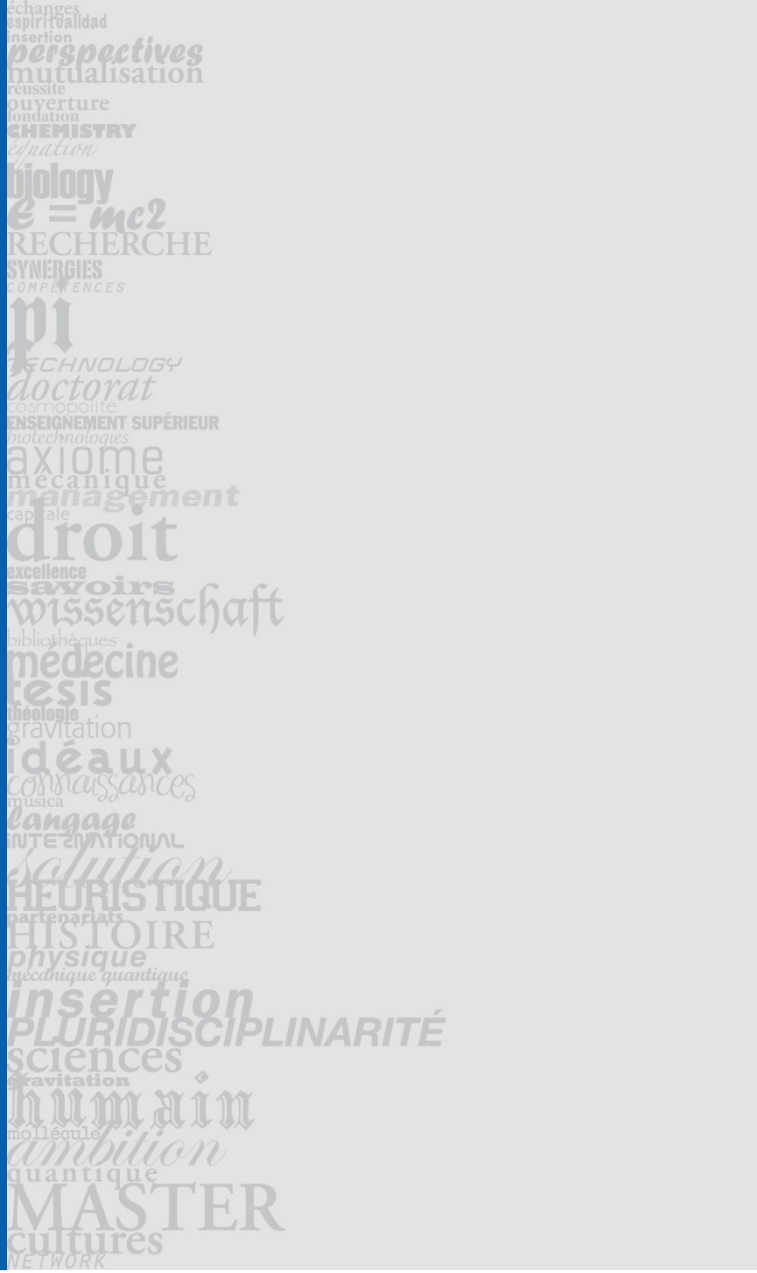
Du Python qui ne manque pas d'air

Mohcine El Barhbarh, Bernard Barrois
Romaric David, Yannick Hoarau



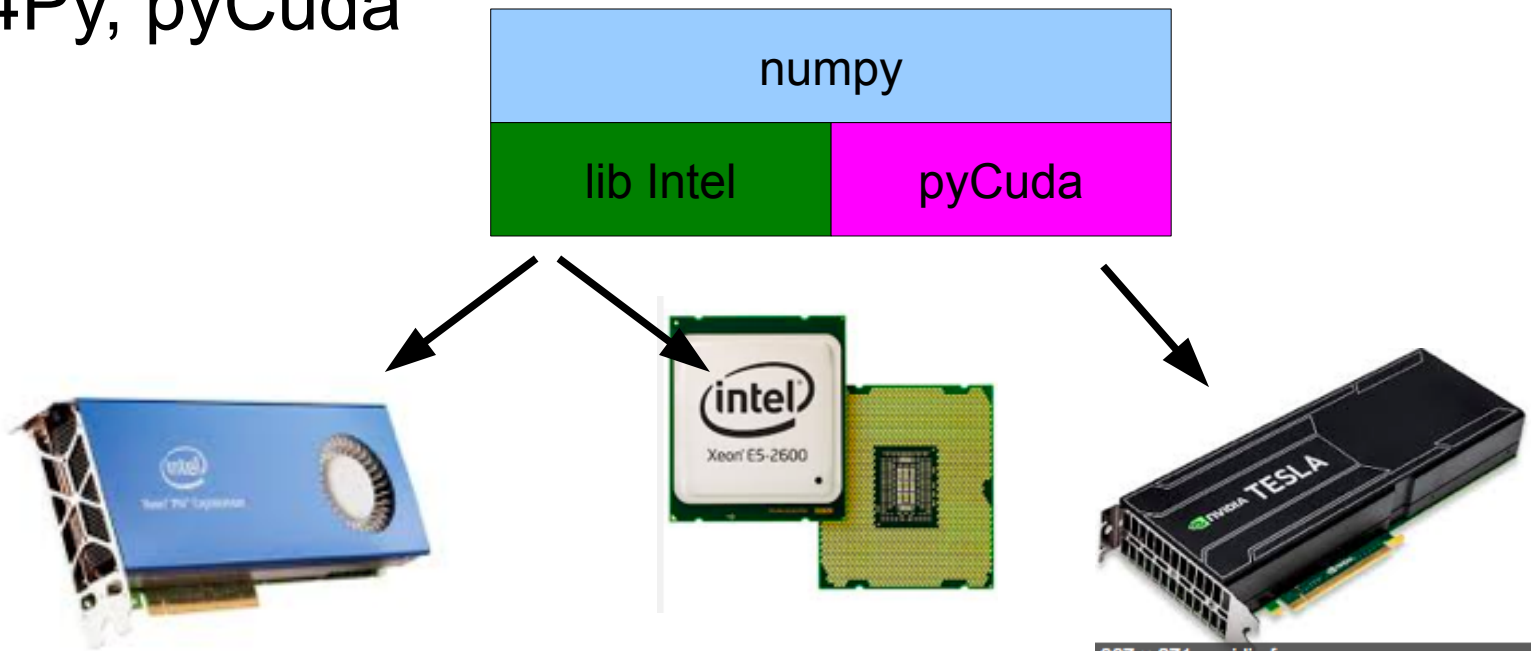
iCUBE
PSA PEUGEOT CITROËN
UNIVERSITÉ DE STRASBOURG
EcoInfo
Groupe CALCUL

Plan

- Du python et du HPC
- Simulation de flux d'air
- Todo
- Conclusion

Du python et du HPC

- Centre de calcul de l'Université de Strasbourg
- http://hpc.unistra.fr
- 300 serveurs, 4000 coeurs, 400 TO de stockage
- Nous utilisons python pour :
  - La programmation d'applications performantes : numpy, mpi4Py, pyCuda



UNIVERSITÉ DE STRASBOURG

Du python et du HPC

- Pour notre Datacentre : le relevé de mesures sur notre chaîne de distribution d'électricité
- La formation (Parallélisme, I/O, Visualisation...)

- Intérêt pour python depuis 2006
  - Point d'intérêt remonté par nos utilisateurs
  - Formations python dès lors

Plan

- Du python et du HPC
- Simulation de flux d'air
- Todo
- Conclusion

Pourquoi ?

- En 2011, lors de la conception de notre Datacenter, nous avons lancé des études CFD sur la salle dans le but de produire des belles images
- Le travail s'est fait en collaboration avec Yannick Hoarau du laboratoire Icube
  - OpenFoam
  - Simulation très précise ⇒ temps de calcul longs
- Nous avons relancé un travail comparable par le truchement d'Ecoinfo, suite à un contact initié par Bernard Barrois (PSA)

PSA PEUGEOT CITROËN

EcoInfo

UNIVERSITÉ DE STRASBOURG

Pourquoi ?

- Pouvoir étudier les flux d'air et la thermique dans différents scénarios d'aménagement de salle machine...
- … sans reprendre toutes les fonctionnalités de logiciels comme 6Sigma
- et sur la base de logiciels libres
- L'équipe du projet :
  - Mohcine El Barhbarh : stagiaire, validation du modèle
  - Bernard Barrois : PSA, développement du modèle
  - Romaric David : Développement du pré-processeur
  - Yannick Hoarau, ICUBE, Unistra, validation du modèle et pilotage scientifique

UNIVERSITÉ DE STRASBOURG

Comment ?

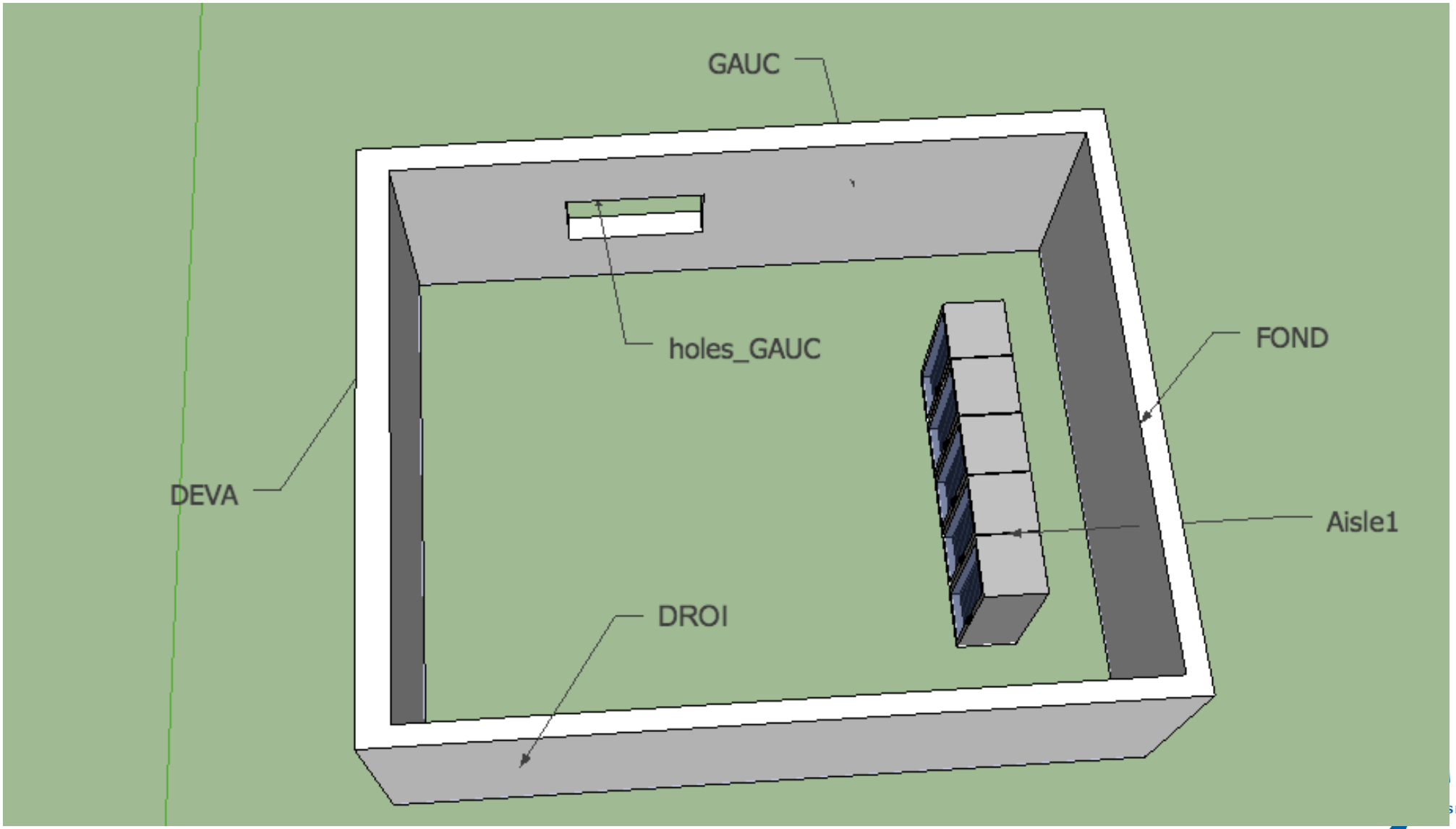
- Simulation : FreeFem++
  - http://www.freefem.org/ff++/index.htm
  - Langage dédié pour la simulation par éléments finis
  - Maillage automatique à partir d'une géométrie
  - IDE disponible sous plusieurs plateformes (Freefem-cs)
- Pour chaque salle machine, il faut écrire un programme FreeFem++ différent :

```
// 4 holes in the floor
border d31(t=0,.6)   {x=12.8+t; y=3.;   label=3;};      // 7,3 => 7.6,3
border d32(t=0,4.)   {x=13.4;   y=3.+t; label=3;};      // 7.6,3 => 7.6,6
border d33(t=0,.6)   {x=13.4-t; y=7.;   label=3;};      // 7.6,6 => 7,6
border d34(t=0,4.)   {x=12.8;   y=7.-t; label=3;};      // 7,6 => 7,3
// wall of the back of the room, contains a hole
mesh thmur0 = buildmesh (  a11(10*n) + a12(5*n) + a13(10*n) + a14(5*n)
                         +b01(-3*ntrou)  + b02(-2*ntrou)  + b03(-3*ntrou)  + b04(-2*ntrou));// mur
```

Comment ?

- Le code FreeFem prend en charge l'aéraulique et la thermique de la salle
- Ce code est devenu trop compliqué à écrire à la main, nous avons écrit un pré-processeur python qui utilise en entrée une description de salle en format texte
- Ce fichier texte ressemble à un fichier .ini de Windows : sections + clef=valeurs
- Le fichier décrit : une salle machine contient des murs et des allées. Les allées à leur tour contiennent des racks

La pièce



La pièce

Structure générale de la description de la pièce :

```
[Constants]
# Some constants used by the simulator
[Room]
name = Room1
# Dims : H x W x L.
dims = 5x10x20
holes_GAUC = dims:1,3 pos:3,16 speed:1.7 temp:-50 puffing:0
holes_PLAN = dims:0.6,4 pos:5,3 speed:-1.2 temp:17 puffing:1
             dims:0.6,4 pos:6,3 speed:-1.2 temp:17 puffing:1
contains = aisle1 aisle2 aisle3 aisle4

[aisle1]
contains = Rack1
# Which wall is facing the servers front-side ?
orientation = DEVA
```

Noms des murs (pré-défini)

Ce mur a 2 entrées d'air

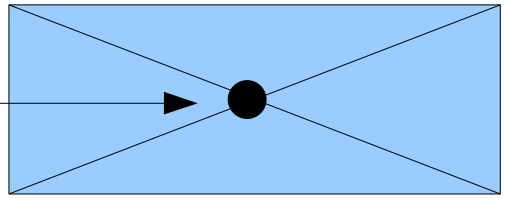
La pièce

```
[Rack1]
# Dimensions. At most 1 digit after comma
hauteur = 2
largeur = 4
profondeur = 1.2
# Position of the center of the rack on the floor
position_milieu = x_largeur:5 y_longueur:4.3

# Delta t in the rack (temp_out - temp_in)
temp_in = 23
temp_out = 33
power_it = 6000

[Rack2]
# and so on...
```

1 grand rack, plutôt un méta-rack



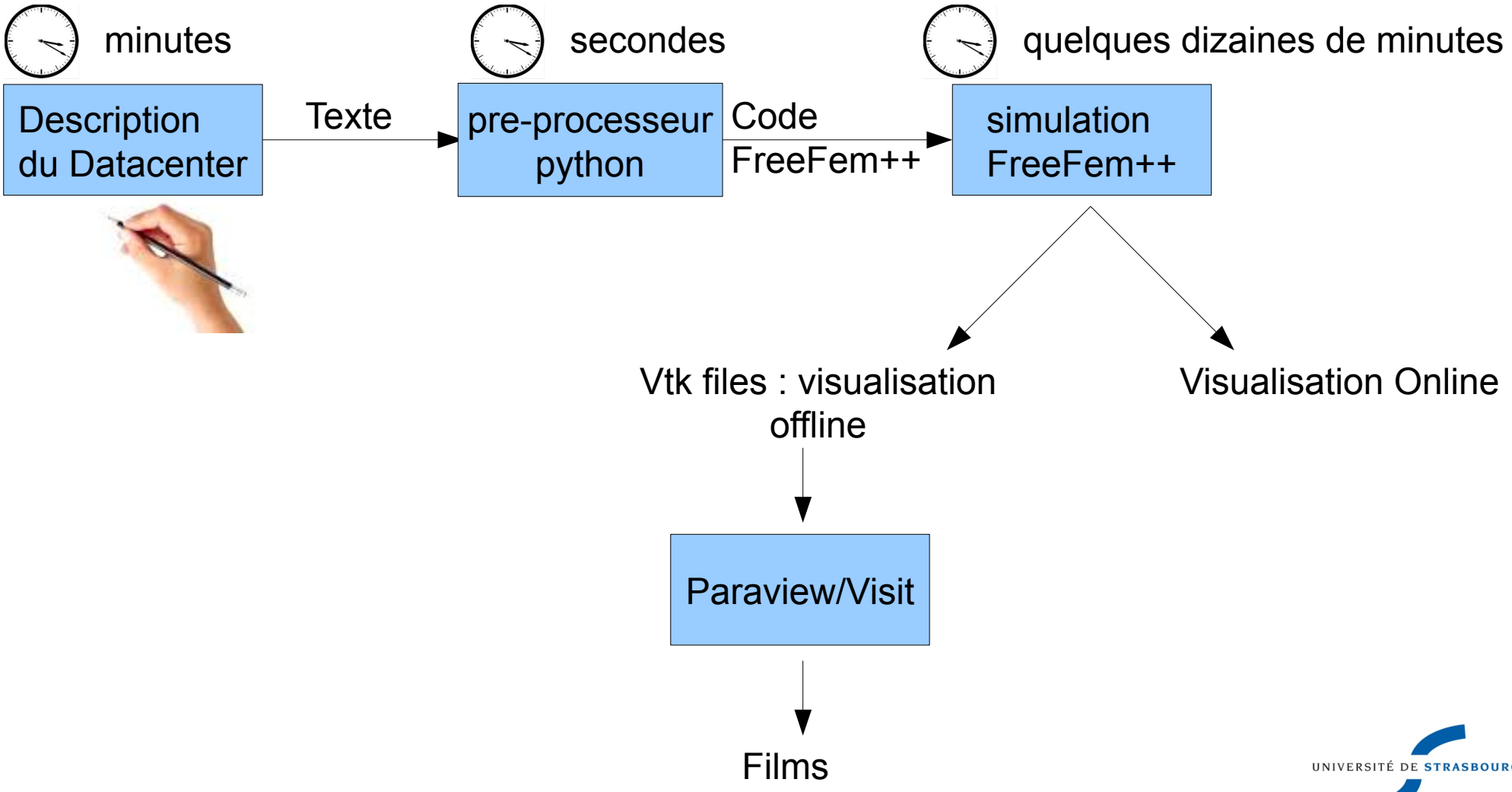
Le pré-processeur

- Le pré-processeur en Python
  - utilise le module `configparser`
  - effectue des substitutions de chaines de caractères dans des templates de texte non complètement instanciés
  - calcule l'ensemble des éléments géométriques de la pièce à partir de la description fournie
  - utilise un grand nombre de dictionnaires (indexation par chaîne de caractères, pratique comme table des symboles)
  - n'effectue pas de vérification de cohérence des dimensions (= peut générer un code FreeFem qui fait 

)

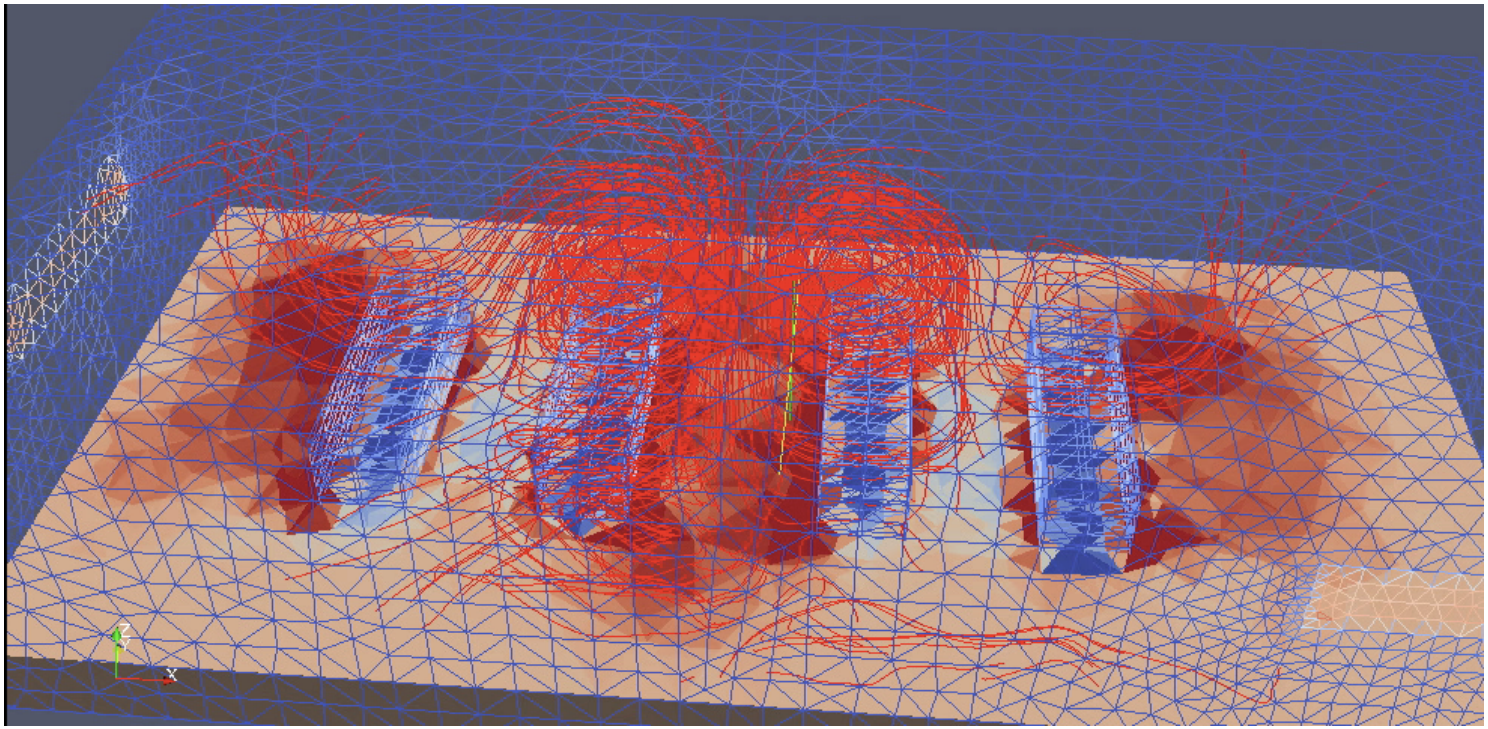
Logiciels : workflow

- L'enchaînement des outils est le suivant :



Logiciels : visualisation avec paraview

- Datacenter avec 4 allées
- L'air froid vient du sol, et l'air chaud est aspiré par 2 trous dans le mur



Plan

- Du python et du HPC
- Simulation de flux d'air
- Todo
- Conclusion

Logiciel : Todo

- Pour l'instant il faut décrire à la main la salle machine
- Un petit logiciel de conception serait le bienvenu
  - En python ? Quels modules ?
  - Parsing de la sortie d'un autre logiciel ?
- Vérification de la cohérence des dimensions indiquées :
  - Dessin à postériori de la salle. En propre ? Sketchup ?
  - Indications des conflits possibles
- Ajout d'objets dans le pré-processeur : gaînes, poteaux, ...

Logiciel : Todo

- Nettoyage et factorisation du code
- Modularisation :
  - création de code FreeFem avec uniquement l'aéraulique par exemple
  - Génération de plusieurs codes FreeFem avec différentes finesses de maillage...
  - Que peut-on faire avec l'API python de FreeFem ?

Plan

- Du python et du HPC
- Simulation de flux d'air
- Todo
- Conclusion

Conclusion

- Modèle validé scientifiquement
- Testé par PSA et présenté durant formation Datacentres
- À ce jour, logiciel téléchargeable à l'adresse http://hpc-web.u-strasbg.fr/Jelly
- Testeurs bienvenus !

Conclusion

- Équipe
  - Mohcine El Barhbarh
  - Bernard Barrois
  - Romaric David
  - Yannick Hoarau

Bernard

PSA PEUGEOT CITROËN

Mohcine

EcoInfo

Yannick

ICUBE

UNIVERSITÉ DE STRASBOURG