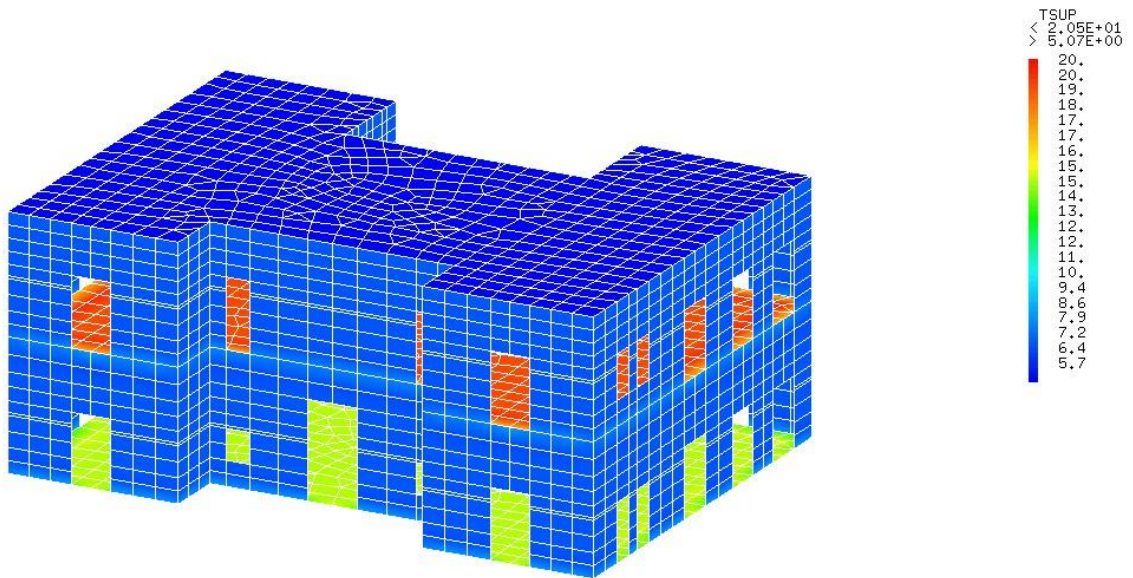


Très bon travail :  
Thermique transitoire avec prise en compte multi-couche + méthode de maillage originale et générale.  
18/20

**CASTEM**  
**ETUDE THERMIQUE DU DIDAM**



DUPORT Nicolas - GARATE Amélie – VIDEGAIN Maider  
ISA 5 - H&E

## Table des matières

INTRODUCTION.....	3
<i>Localisation du bâtiment</i> .....	3
<i>Valeur patrimoniale</i> .....	3
MODELISATION 3D DU DIDAM .....	4
INTEGRATION DES ECHANGES THERMIQUES .....	7
CALCUL INSTANTANE.....	10
RESOLUTION EN FONCTION DU TEMPS.....	11
ANALYSE, INTERPRETATION ET SENSIBILITE DES RESULTATS .....	16
CONCLUSION .....	18

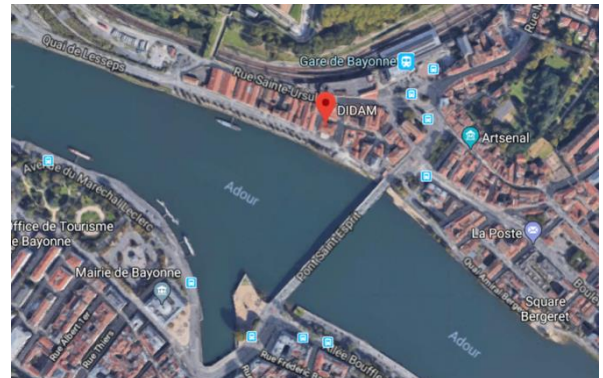
## INTRODUCTION

---

### Localisation du bâtiment

Le DIDAM est situé sur la rive droite de l'Adour. Avec la façade principale orientée Sud, les utilisateurs peuvent profiter d'une vue directe sur le vieux Bayonne. Le bâtiment est proche de la gare et voit les rails longer sa façade nord.

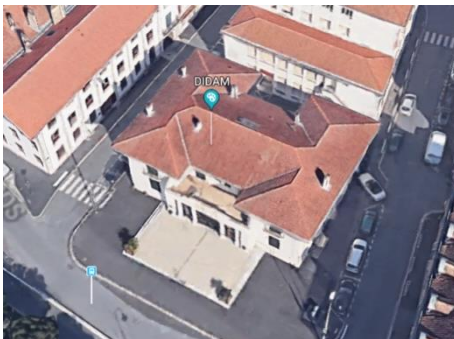
Le quartier de la gare est un quartier aujourd'hui peu actif, c'est pour cela qu'il est le sujet d'une rénovation globale.



### Valeur patrimoniale

Ancien bâtiment de l'Inscription Maritime et de la Bourse du travail, situé sur les bords de l'Adour, il a été réalisé en 1933 par les frères Gomez (Louis Gomez, 1876-1940, architecte D.P.L.G. et Benjamin Gomez, 1885-1959, architecte et décorateur). Cette commande de la Ville de Bayonne était un manifeste du courant Art déco.

Aujourd'hui propriété de l'Agglomération Côte Basque - Adour, il accueille, en complémentarité de temps et d'espace, les expositions culturelles, patrimoniales ou d'art contemporain proposées par la Ville et l'atelier du projet d'urbanisation de la rive droite de l'Adour, mis en place par l'Agglomération.



Nous allons dans un premier temps décrire le programme qui nous a permis d'étudier le bâtiment du DIDAM. Ensuite nous analyserons les résultats et nous reviendrons sur les points clés du programme. Nous finirons par un bilan de ce projet qui fut très intéressant de par l'interopérabilité qu'il existe entre la méthode aux éléments finis et le logiciel Cast3m.

## MODELISATION 3D DU DIDAM

---

OPTION DIMENSION 3 ELEM TRI3 TRAC OPEN  
DENSITE 1;

z0=0.0;  
z1=0.9;  
z2=1.1;  
z3=0.1;  
z4=0.4;  
z5=1.0;  
z6=1.2;  
z7=0.9;  
z8=1.1;  
z9=0.1;  
z10=0.4;  
z11=2.0;

vec1=0.0 0.0 z1;  
vec2=0.0 0.0 z2;  
vec3=0.0 0.0 z3;  
vec4=0.0 0.0 z4;  
vec5=0.0 0.0 z5;  
vec6=0.0 0.0 z6;  
vec7=0.0 0.0 z7;  
vec8=0.0 0.0 z8;  
vec9=0.0 0.0 z9;  
vec10=0.0 0.0 z10;  
vec11=0.0 0.0 z11;

ProgX=PROG 0 2.62 4.22 2 \* 7. 7.67 8.62 10.99  
13.01 15.43 16.33 17. 17.  
19.78 21.38 12 \* 24. 21.38 19.78 8 \* 17.  
16.33 13.61 13.36 10.64 10.39 7.67 \* 7. 4.22 2.62  
9 \* 0.0.;

ProgY=PROG 0 0 0 0 2.47 2.47 2.47 2.47 2.47 2.47  
2.47 2.47 0 0 0  
0 1.88 2.78 3.38 4.28 7.12 8.72 10.82 12.42 13.96  
15.56 16.97  
16.97 16.97 16.97 16.24 15.79 15.13 14.68 14  
13.55 12.97  
12.97 12.97 12.97 12.97 12.97 12.97 12.97 13.55  
14 14.68 15.13 15.79 16.24 16.97 16.97 16.97  
16.97 15.39 14.19 11.19 9.59 7.22 5.62 3.27 1.67  
0.;

progz=prog z1 Z2 Z3 z4 z5 z6 z7 z8 z9 z10 z11;

### → Géométrie du modèle

**OPTION** : sert à préciser les options générales du calcul.

**DIME** : dimension de l'espace.

**ELEM** : élément à fabriquer.

**DENSITE** : sert à définir, par défaut, la taille locale de la maille s'appuyant sur les points à construire.

### → Génération des différents vecteurs pour chaque hauteur

Définition des hauteurs caractéristiques **z** du DIDAM. Ces hauteurs représentent chaque altitude des allèges et des linteaux de chaque menuiserie du bâtiment.

On crée les vecteurs **VEC** qui nous permettent d'avoir les différentes hauteurs en fonction de la façade du DIDAM : allèges, menuiseries et portes.

La constitution de ces vecteurs offre une méthode itérative pour construire la géométrie du bâtiment. Cependant, nous verrons plus loin une autre méthode plus fine pour réaliser la modélisation.

### → Création de listes

**PROG** => création de listes pour les X, Y et Z.

Chaque point du bâtiment est répertorié selon les trois axes X,Y et Z.

La correspondance entre ces trois coordonnées se fera par la suite. L'assemblage se fera par « panneaux ».

```
ltyp=lect 1 2 1 1 1 3 1 4 1 3 1 1 1 2 1 1 6 1 6 1 2 1
2 1 2 1 1 2 1 1 5
1 5 1 5 1 1 3 1 3 1 3 1 1 5 1 5 1 5 1 1 2 1 1 2 1 2 1
2 1 2 1;
```

```
nz=dime progz;
```

```
tabtyp=table;
tabtyp.1=lect 1 PAS 1 1 1;
tabtyp.2=lect 0 0 0 0 1 1 0 0 0 0 1;
tabtyp.3=lect 1 0 1 1 1 1 0 0 0 0 1;
tabtyp.4=lect 0 0 0 0 0 1 1 1 1 1 1;
tabtyp.5=lect 1 0 0 1 1 1 1 0 0 1 1;
tabtyp.6=lect 0 0 0 1 1 1 1 0 0 0 1;
```

```
nel=dime ltyp;
```

```
repete bou1 nel;
```

```
x1=extr ProgX &bou1;
y1=extr ProgY &bou1;
x2=extr ProgX (&bou1 + 1);
y2=extr ProgY (&bou1 + 1);
typ1=extr ltyp &bou1;
P1=x1 y1 z0;
P2=x2 y2 z0;
d1=d P1 P2;
dtyp=d1;
```

```
ltyp1=tabtyp.typ1;
```

```
prims=vrai;
```

```
repete bou2 (nz);
vecz=0. 0. (extr progz &bou2);
si ((extr ltyp1 &bou2) ega 0);
dtyp=dtyp plus vecz;
sinon;
styp=dtyp trans vecz dini 0.5 dfin 0.5;
dtyp=styp cote 3;
```

```
si (prims);
stypt=styp;
prims=faux;
sinon;
stypt=stypt et styp;
fini;
fin bou2;
```

**lect** → Liste de nombres. Chaque numéro correspond à un type de panneau vertical. Les panneaux sont créés en fonction de l'altimétrie des ouvertures.

**Dime** → fournit le nombre d'éléments contenus dans progz à nz.

**lect** → création de liste de nombres.

On crée ici une table représentant chaque type de panneaux. Le 0 correspond à une surface vide (ouverture) et le 1 correspond à une surface pleine. La table comprend désormais 6 types de panneaux.

**Dime** → fournit la dimension de l'objet ltyp à nel.

**repete** → permet de répéter *nel* fois l'exécution de l'ensemble d'instructions *bou1*.

**extr** → permet d'extraire le nième terme de la liste ProgX, ProgY et ltyp, qui correspond à la nième itération de bou1. **&bou1 + 1** → permet de prendre le terme de la liste qui correspond à l'itération i+1 de bou1.

On crée P1 et P2, deux points sur le niveau du sol.

**d** : droite → tracer une droite entre les points P1 et P2

On rentre dans ltyp1 le type de panneaux sur lequel on va travailler dans la suite de la boucle.

→ On initialise *prims* qui sera une condition dans la suite de la boucle.

→ On crée une deuxième boucle (à l'intérieur de bou1) sur le nombre d'éléments de nz.

On extrait la hauteur z de progz en fonction de l'itération de bou2. Ensuite, si dans le tabtyp on tombe sur zero, on réalise une translation de la droite qu'on avait tracé au sol entre P1 et P2 (opérateur *plus*). Sinon, l'opérateur *trans* permet de créer une surface entre la droite du sol et la hauteur z.

« dini ... dfin ... » permet de modifier la dimension du maillage sur les surfaces que l'on est en train de créer.

→ La condition *prims* permet d'ajouter les surfaces que l'on crée dans la boucle bou2 au fur et à mesure.

```

si (&bou1 ega 1);
  stot=stypt;dtot=d1;
sinon;
  stot=stot et stypt;
  dtot=dtot et d1;
fin si;
elim dtot;
tracer dtot;
tracer stot;
fin bou1;

```

```

stot1=stot;
P=(6 8.48 4.5);
Xmil=12.;
X=coord 1 stot;
Pg= X point infega Xmil;
Sg= stot1 elem appuye strictement Pg;
Sg=Sg orienter 'POINT' (6 8.48 4.5) INVE;
Sd=diff stot Sg;
*trac Sg;
*trac Sd;

```

```

Smac=Sd et Sg coul rouge;
elim smac 1.e-4;
*trac Smac;

```

```

cont1b=dtot;
cont1d=dtot plus (0. 0. 0.);
dall1=surf cont1d plane ORIEN (0. 0. 1.);
vec12=(0. 0. (z1 + z2 + z3 + z4 + z5 + z6));
vec13=(0.
(z1+z2+z3+z4+z5+z6+z7+z8+z9+z10+z11));
cont2d dall2=cont1d dall1 plus vec12;
cont2b=cont1b plus vec12;
cont3d dall3=cont1d dall1 plus vec13;
cont3b=cont1b plus vec13;
elim smac cont1b;
elim smac cont2b;
elim smac cont3b;

```

```

Sisol=Smac plus (0. 0. 0.) coul bleu;
*trac Sisol;

```

```

dalltot=dall1 et dall2 et dall3;
tout1=Smac et dalltot;
tout2=Sisol et dalltot;
*trac tout1 face;
*trac tout2 face;

```

→ On assemble dans cette condition tous les panneaux entre eux (surface et contour).

La fonction *elim* permet d'éliminer les nœuds en double (sur les panneaux et sur le contour).

→ Cette partie permet d'orienter les faces vers l'intérieur du volume.

Le principe est de construire seulement la moitié du volume et ensuite d'effectuer une symétrie. La symétrie permet de garder les orientations.

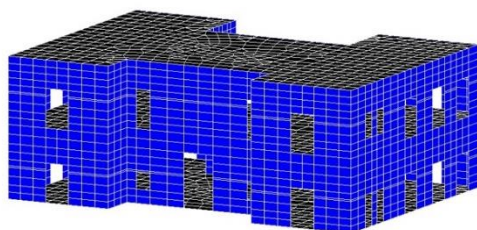
→ Lorsque l'on regroupe les deux côtés, il faut éliminer les nœuds qui sont en double. Smac est finalement la surface extérieure du bâtiment. Elle représente la couche de maçonnerie.

→ On crée maintenant les planchers.

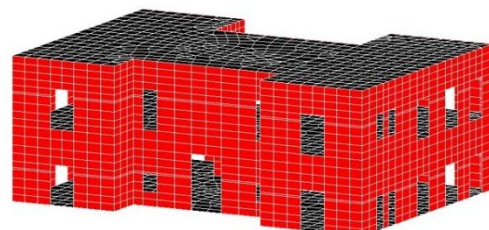
0. Dall1 représente le plancher bas. Dall2 le plancher intermédiaire et Dall3 le plancher haut. Il est important dans cette étape de créer un contour pour les planchers et un contour pour les murs. On vient ensuite supprimer les nœuds en double pour que la relation de température que nous imposerons plus tard permette une transmission de chaleur correcte.

→ Sisol correspond à la couche d'isolant.

→ On peut maintenant constituer notre bâtiment avec les deux couches des murs périphériques ainsi que les trois planchers.



*Smac*



*Sisol*

## INTEGRATION DES ECHANGES THERMIQUES

lambmur=0.952;  
 lambisol=0.833;  
 \*lambisol=0.039;  
 lambpl1=1.75;  
 lambpl2=0.140;  
 lambpl3=0.048;

CPBET=1008;  
 RHOBET=2150.;  
 \*briquettes  
 \*CPISOL=1000;  
 \*RHOISOL=1850;  
 \*isolants  
 RHOISOL=35.;  
 CPISOL=1030.;  
 RHOPL2=500.;  
 CPPL2=2500.;  
 RHOPL3=170;  
 CPPL3=1290;

eppl2=0.15;  
 eppl1=0.15;  
 eppl3=0.263;  
 epmac=0.15;  
 \*episol=0.05;  
 episol=0.10;

Hint=9.09;  
 Hext=16.67;  
 Hpl1sup=5.89;  
 Hpl1inf=11.1;  
 Hpl2inf=11.1;  
 Hpl2sup=5.89;  
 Hpl3inf=11.1;  
 Hpl3sup=20;  
 Tint=19.;  
 Text=5.;  
 Tss=10.;

```
MOD1=MODE Smac THERMIQUE CONDUCTION
COQ4;
MOD2=MODE Sisol THERMIQUE CONDUCTION
COQ4;
MAT1=MATE MOD1 'K' lambmur 'C' CPBET 'RHO'
RHOBET EPAI epmac;
MAT2=MATE MOD2 'K' lambisol 'C' CPISOL 'RHO'
RHOISOL EPAI episol;
MOCONV1=MODE Smac THERMIQUE
CONVECTION 'SUPERIEURE' COQ4;
MOCONV2=MODE Sisol THERMIQUE
CONVECTION 'INFERIEURE' COQ4;
MACONV1=MATE MOCONV1 'H' Hext;
MACONV2=MATE MOCONV2 'H' Hint;
```

➔ Définition des paramètres des matériaux.

Conductivité thermique en  $m^2K/W$ .

Capacité thermique massique en  $J/kg.K$ .

Masse volumique en  $kg/m^3$ .

Ces deux paramètres serviront à la résolution en fonction du temps. La procédure PASAPAS requiert des paramètres qui permettront de prendre en compte l'inertie des matériaux.

Epaisseur des éléments en mètre.

Coefficients de convection intérieure et extérieure : ils permettent de quantifier un transfert de chaleur réalisé par un phénomène de convection au sein d'un fluide en mouvement exprimés en  $W/m^2.K$ .

Il a été choisi d'appliquer un coefficient convectif sur chaque face de chaque paroi (contact extérieur et intérieur).

Températures initiales intérieures et extérieures.

On définit des valeurs de températures arbitrairement en degré Celsius.

➔ L'opérateur **MODE** (MODELISER) permet d'associer à un maillage un modèle de comportement du matériau. Dans notre cas d'étude, il faut préciser que l'on modélise en THERMIQUE, la conduction étant le comportement par défaut.

➔ Le phénomène de convection entre milieu externe et paroi est un comportement spécifique et doit être mentionné par l'extension **CONVECTION**.

➔ COQ4 permet de définir le maillage comme étant des éléments coques à 4 nœuds. La conduction est appliquée aux éléments COQ4.

➔ L'opérateur **MATE** (MATERIAU) crée un champ de propriétés matérielles. On rattache à cet opérateur les propriétés de chaque matériau définies précédemment à sa surface.

```

MODPL1=MODE dall1 thermique conduction COQ3
COQ4;
MODPL2=MODE dall2 thermique conduction COQ3
COQ4;
MODPL3=MODE dall3 thermique conduction COQ3
COQ4;
MATPL1=MATE MODPL1 'K' lambpl1 'C' CPBET
'RHO' RHOBET EPAI eppl1;
MATPL2=MATE MODPL2 'K' lambpl2 'C' CPPL2
'RHO' RHOPL2 EPAI eppl2;
MATPL3=MATE MODPL3 'K' lambpl3 'C' CPPL3
'RHO' RHOPL3 EPAI eppl3;
MOCOPL1=MODE dall1 THERMIQUE
CONVECTION 'INFERIEURE' COQ3 COQ4;
MOCOPL11=MODE dall1 THERMIQUE
CONVECTION 'SUPERIEURE' COQ3 COQ4;
MOCOPL2=MODE dall2 THERMIQUE
CONVECTION 'INFERIEURE' COQ3 COQ4;
MOCOPL22=MODE dall2 THERMIQUE
CONVECTION 'SUPERIEURE' COQ3 COQ4;
MOCOPL3=MODE dall3 THERMIQUE
CONVECTION 'INFERIEURE' COQ3 COQ4;
MOCOPL33=MODE dall3 THERMIQUE
CONVECTION 'SUPERIEURE' COQ3 COQ4;
MACOPL1=MATE MOCOPL1 'H' Hpl1inf;
MACOPL11=MATE MOCOPL11 'H' Hpl1sup;
MACOPL2=MATE MOCOPL2 'H' Hpl2inf;
MACOPL22=MATE MOCOPL22 'H' Hpl2sup;
MACOPL3=MATE MOCOPL3 'H' Hpl3inf;
MACOPL33=MATE MOCOPL33 'H' Hpl3sup;

MOTOTPL=MODPL1 ET MODPL2 ET MODPL3 ET
MOCOPL1 ET MOCOPL11
ET MOCOPL2 ET MOCOPL22 ET MOCOPL3 ET
MOCOPL33;
MATOTPL=MATPL1 ET MATPL2 ET MATPL3 ET
MACOPL1 ET MACOPL11
ET MACOPL2 ET MACOPL22 ET MACOPL3 ET
MACOPL33;

MOTOT=MOD1 ET MOD2 ET MOCONV1 ET
MOCONV2 ET MOTOTPL;
MATOT=MAT1 ET MAT2 ET MACONV1 ET
MACONV2 ET MATOTPL;
COND1=COND MOTOT MATOT;

FLEXT=CONVECTION MOCONV1 MACONV1 'TSUP'
Text;
FLINT=CONVECTION MOCONV2 MACONV2 'TINF'
Tint;
FLPL1=CONVECTION MOCOPL1 MACOPL1 'TINF'
Tss;
FLPL11=CONVECTION MOCOPL11 MACOPL11
'TSUP' Tint;
FLPL2=CONVECTION MOCOPL2 MACOPL2 'TINF'
Tint;
FLPL22=CONVECTION MOCOPL22 MACOPL22
'TSUP' Tint;

```

→ On réalise la même opération pour les planchers.

→ Concernant les planchers, il faut définir des conditions de convection sur chaque face de chaque plancher. Les valeurs de convection ne sont pas les mêmes en fonction de l'orientation et du sens du flux.

→ On assemble les modèles des planchers entre eux. On fait de même pour la définition des matériaux des planchers.

→ On assemble ici les modèles des murs et des planchers.

→ On fixe une relation de conductivité entre les modèles et les matériaux.

→ Dans cette partie, les échanges thermiques par convection sont définis comme un flux entre la face supérieure ou inférieure d'une paroi et l'air avec lequel cette dernière est en contact.



```
FLPL3=CONVECTION MOCOPL3 MACOPL3 'TINF'
Tint;
FLPL33=CONVECTION MOCOPL33 MACOPL33
'TSUP' Text;
```

```
FLT=FLEXT ET FLINT ET FLPL1 ET FLPL11 ET
FLPL2
ET FLPL22 ET FLPL3 ET FLPL33;
```

```
rel1=rela 'T' cont1b - 'T' cont1d;
rel2=rela 'T' cont2b - 'T' cont2d;
rel3=rela 'T' cont3b - 'T' cont3d;
REL4=RELA Smac TINF - Sisol TSUP;
```

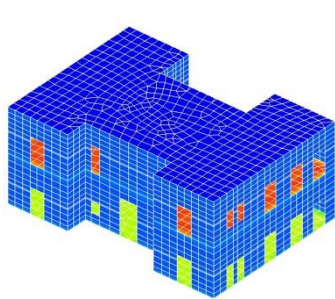
```
CONDT=COND1 ET REL1 ET REL2 ET REL3 ET
REL4;
TEMP1 = RESOU CONDT FLT;
```

```
trac TEMP1 dalltot;
trac TEMP1 tout1;
trac TEMP1 tout2;
```

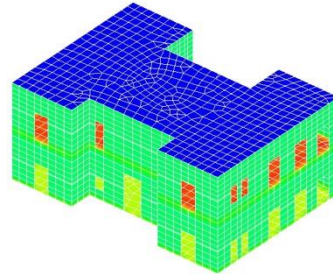
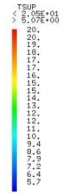
- ➔ Toutes les relations de flux sont rassemblées pour faciliter l'écriture du calcul thermique.
- ➔ Ces relations permettent de fixer la température de la couche intermédiaire 'T' entre deux contours.
- ➔ Une des relations sinon la plus importante du programme est celle-ci car elle permet de faire le lien entre la surface extérieure Smac et la surface intérieure Sisol des murs périphériques de notre bâtiment.
- ➔ On rassemble toutes les conditions de relation.
- ➔ C'est à cette étape que nous réalisons le calcul thermique.
- ➔ L'opérateur RESOU construit une solution du système linéaire dont les inconnues sont les températures.
- ➔ Tracé des solutions.
- ➔ On applique les températures sur le champ point défini auparavant.

# CALCUL INSTANTANE

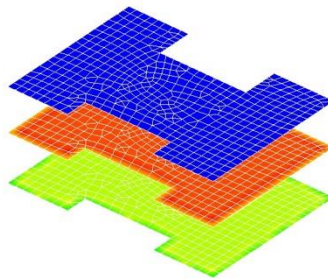
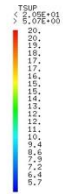
---



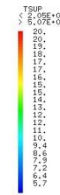
*Smac*



*Sisol*



*Plancher bas, plancher intermédiaire et plancher haut*



## RESOLUTION EN FONCTION DU TEMPS

```
OPTI ACQU 10 ACQU 'temps.txt';
ACQU PROGT*LISTREEL 169;
OPTI ACQU 10 ACQU 'temperature.txt';
ACQU PROGTP*LISTREEL 169;
```

```
EVOLT=EVOL MANU PROGT PROGTP;
```

```
n1=dime progt;
```

```
flintm=manu chpo sisol 1 'TINF' 1. nature
diffus;
flextm=manu chpo smac 1 'TSUP' 1. nature
diffus;
```

```
flpl1inf=manu chpo dall1 1 'TINF' 1. nature
diffus;
flpl1sup=manu chpo dall1 1 'TSUP' 1. nature
diffus;
```

```
flpl2inf=manu chpo dall2 1 'TINF' 1. nature
diffus;
flpl2sup=manu chpo dall2 1 'TSUP' 1. nature
diffus;
```

```
flpl3inf=manu chpo dall3 1 'TINF' 1. nature
diffus;
flpl3sup=manu chpo dall3 1 'TSUP' 1. nature
diffus;
```

```
flint= flintm et flpl1sup et flpl2inf et flpl2sup et
flpl3inf;
flex= flextm et flpl3sup;
```

```
flss=flpl1inf;
```

```
tfin=extr progt n1;
prog1=prog 0. tfin;
prog2=prog tint tint;
EVOL2=EVOL MANU PROG1 PROG2;
prog3=prog tss tss;
evol3=evol manu prog1 prog3;
```

```
CHAR2=CHAR flex evol 'TECO';
CHAR1=CHAR flint EVOL2 'TECO';
CHAR3=CHAR FLSS EVOL3 'TECO';
```

- L'opérateur ACQU permet d'extraire des données, des réels en l'occurrence, d'une liste d'un fichier .txt. Les réels sont ensuite rangés dans un prog (liste).
  - Les deux fichiers comprennent d'une part les températures et d'autre part le temps associé qui proviennent des DJU (données journalières unifiées).
  - On crée une évolution avec l'option MANU qui permet de définir une fonction entre deux listes de réels. Cette évolution permettra de faire varier la température extérieure.
- 
- Comme pour le calcul en instantané, on applique une condition de flux sur chaque face de chaque paroi.
  - L'opérateur MANUEL avec le mot clé CHPO construit un champ par point. En l'occurrence, on construit un champ par point à partir du maillage que l'on a défini dans le premier chapitre.
  - Sur chaque face, on précise la couche supérieure ou inférieure sur laquelle on applique notre champ point.
  - Le mot clé NATURE permet d'attribuer la nature du champ : **indéterminé, diffus** ou **discret**.
- 
- La fonction *evol manu* crée une fonction entre deux listes de réels.
  - Ici, on crée deux évolutions : une pour bloquer la température intérieure du début à la fin du calcul et une autre qui fait la même chose mais pour la température du sous sol.
- 
- L'opérateur CHARG construit un objet de type chargement de sous type FORCE contenant la description spatiale et temporelle du chargement.
  - La composante fl... est la description spatiale du chargement comme précisé ci-dessus (champ point).

```
T0=TEMP1;
TAB1=TABLE;
TAB1.TEMPERATURES=TABLE;
TAB1.TEMPERATURES.0=Temp1;
TAB1.BLOCAGES_THERMIQUES=REL1 ET REL2
ET REL3 ET REL4;
TAB1.MODELE=MOTOT;
TAB1.CARACTERISTIQUES=MATOT;
TAB1.CHARGEMENT=char1 et char2 et CHAR3;
TAB1.TEMPS_CALCULES=progt;
```

```
PASAPAS TAB1;
```

```
*opti sauv 'didam3.sauv';
*sauv;
*opti rest 'didam3.sauv';rest;
```

```
N1=DIME (TAB1 . TEMPS_CALCULES) - 1 ;
REPETER BOU1 N1;
*TRAC TAB1.TEMPERATURES . &BOU1 smac ;
Fin BOU1;
```

```
n1=dime tab1.temperatures;
progint=prog;
progext=prog;
progt=prog;
progpl1e=prog;
progpl1i=prog;
progpl3e=prog;
progpl3i=prog;
progpl2i=prog;
progpl2e=prog;
```

- La composante evol... est la description temporelle du chargement. C'est une fonction contenant en abscisse les temps et en ordonnée les valeurs de température.
- Puisque on utilisera une procédure PASAPAS, le mot TECO est obligatoire. Elle mentionne les températures extérieures qui sont fonctions de la convection appliquée.
- On crée la table qui permettra de lancer la procédure de calcul PASAPAS.
- On crée dans un premier temps la table température et on applique comme première valeur Temp1 qui est le résultat obtenu dans la première partie du projet : calcul en instantané.
- On rentre ensuite les conditions de blocages thermiques, les modèles thermiques, les caractéristiques des matériaux, les chargements et finalement les temps sur lesquels on souhaite réaliser le calcul.
- Cette procédure permet de réaliser un calcul linéaire et non linéaire en tenant compte de la convection, de la conduction et du rayonnement.
- En entrée, la table TAB1 sert à définir les options et les paramètres du calcul.
- En sortie, TAB1 permet de retrouver les résultats. Ceux-ci sont mis dans des tables dont les indices sont des entiers correspondants aux numéros de sauvegarde (0 : temps initial).
- Cette option permet de sauvegarder le calcul. Tout ce qu'il y a avant le opti sauv sera sauvegardé dans un fichier que l'on peut ouvrir dans une autre fenêtre. Cette option est très intéressante lorsque les calculs sont longs et qu'on veut seulement changer des fonctionnalités d'aspect ou afficher des résultats.
- Cette boucle affiche les résultats de la procédure sur la surface extérieure de notre bâtiment. Elle permet de s'assurer que la procédure ait bien fonctionné.
- On initialise plusieurs listes vides dans un premier temps. Ces listes seront remplies par la suite avec les différentes valeurs de températures contenues dans la table TAB1 et qui seront appliquées aux maillages de notre bâtiment.

```

repetier bou1 n1;

i=&bou1 - 1;

t1=tab1.temps.i;
tparoi=(redu tab1.temperatures.i smac) exco
tsup ;
tpint=(redu tab1.temperatures.i sisol) exco tinf
;
tpl1i=(redu tab1.temperatures.i dall1) exco tinf;
tpl1e=(redu tab1.temperatures.i dall1) exco
tsup;
tpl3i=(redu tab1.temperatures.i dall3) exco tinf;
tpl3e=(redu tab1.temperatures.i dall3) exco
tsup;
tpl2i=(redu tab1.temperatures.i dall2) exco tinf;
tpl2e=(redu tab1.temperatures.i dall2) exco
tsup;

```

```
text=ipol evolt t1;
```

```

FLext=HEXT*(tparoi - text);
FLINT=HINT*(TPINT - TINT);
Flpl1i=Hpl1sup*(tpl1i - tint);
Flpl1e=Hpl1inf*(tpl1e - tss);
Flpl3i=Hpl3inf*(tpl3i - tint);
Flpl3e=Hpl3sup*(tpl3e - text);
Flpl2i=Hpl2inf*(tpl2i - tint);
Flpl2e=Hpl2sup*(tpl2e - tint);

```

```

chamext=chan cham flex mod1;
CHAMINT=chan cham flint mod2;
champl1e=chan cham flpl1e modpl1;
champl1i=chan cham flpl1i modpl1;
champl3e=chan cham flpl3e modpl3;
champl3i=chan cham flpl3i modpl3;
champl2e=chan cham flpl2e modpl2;
champl2i=chan cham flpl2i modpl2;

```

- Tout ce qui est compris dans la boucle *bou1* sera répété pour chaque valeur de températures contenues dans le fichier que nous avons introduit précédemment.
  - Cette boucle réduit dans un premier temps le champ par points *tab1.temperatures* aux points du maillage de chaque paroi.
  - L'opérateur EXCO crée à partir d'un modèle champ point (la réduction que l'on vient de voir), un champ de même type (*tsup* ou *tinf*) en extrayant plusieurs composantes données.
- 
- On réalise ici une interpolation linéaire qui calcule un champ point à partir de la table *tab1* et du temps *evolt*. Finalement, cette interpolation linéaire calcule la température entre les points dont on connaît déjà la valeur des températures.
- 
- On détermine dans cette partie la valeur du flux entrant ou sortant de chaque paroi.
  - $F=H*\delta T$
- 
- Pour pouvoir calculer le flux traversant les parois, il faut intégrer ce flux sur l'ensemble de chaque surface.
  - Pour cela, on transforme les champs par points en champ intégral.
  - En présence du mot clé CHAM, l'opérateur CHAN crée un champ par éléments à partir d'un champ par points appuyé sur un maillage ou un modèle dans notre cas. Nous n'avons pas indiqué ici le lieu où l'on souhaite que le champ soit appuyé. Par défaut, ce dernier s'appuie sur les nœuds.
  - Il est à préciser que, dans le cas d'un champ de température défini sur des coques avec intégration dans l'épaisseur, la température en chaque point est calculée en fonction de *T* (couche intermédiaire), *TSUP* (couche supérieure) et *TINF* (couche inférieure) suivant une variation parabolique de l'épaisseur.

```

fprese=intg chamext mod1;
fresi=intg chamint mod2;
frpl1e=intg champl1e modpl1;
frpl1i=intg champl1i modpl1;
frpl3e=intg champl3e modpl3;
frpl3i=intg champl3i modpl3;
frpl2e=intg champl2e modpl2;
frpl2i=intg champl2i modpl2;

progext = inse progext &bou1 fprese;
progeint = inse progeint &bou1 fresi;
progpl1e=inse progpl1e &bou1 frpl1e;
progpl1i=inse progpl1i &bou1 frpl1i;
progpl3e=inse progpl3e &bou1 frpl3e;
progpl3i=inse progpl3i &bou1 frpl3i;
progpl2e=inse progpl2e &bou1 frpl2e;
progpl2i=inse progpl2i &bou1 frpl2i;
progt=inse progt &bou1 t1;

fin bou1;

```

```

evolxt=evol 'BLEU' manu progt progext;
evolint=evol 'ROUG' manu progt progeint ;
evolpl1e=evol 'VIOL' manu progt progpl1e;
evolpl1i=evol 'VIOL' manu progt progpl1i ;
evolpl3e=evol 'VERT' manu progt progpl3e;
evolpl3i=evol 'VERT' manu progt progpl3i ;
evolpl2e=evol 'GRIS' manu progt progpl2e;
evolpl2i=evol 'GRIS' manu progt progpl2i ;

```

```

TAB2 = Table;
TAB2.1= 'Tirr';
TAB2.2= 'Tirr';
TAB2.3= 'Tirr';
TAB2.4= 'Tirr';
TAB2.TITRE=Table;
TAB2.TITRE.1 = MOT 'Murs extérieurs';
TAB2.TITRE.2 = MOT 'Plancher bas';
TAB2.TITRE.3 = MOT 'Plancher étage';
TAB2.TITRE.4 = MOT 'Plancher haut';
Titre 'Evolution du flux en fonction du temps';

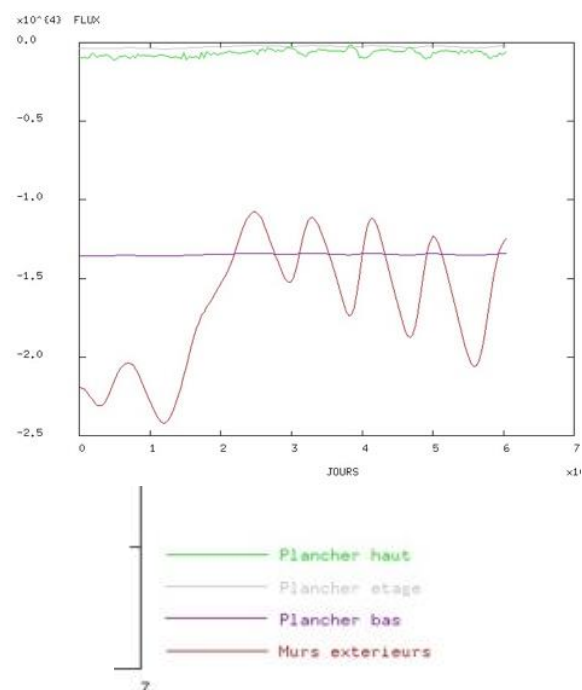
```

```

dess (evolint et evolpl1i et evolpl2i et evolpl3i)
'LEGE' TAB2
'TITX' Jours 'TITY' Flux 'POSY' EXCE;
*dess (evolxt et evolpl1e et evolpl2e et
evolpl3e);

```

- On réalise l'intégration du champ sur chacun des éléments. Puisque nous avons des éléments coques, le domaine d'intégration est la surface de la coque.
- On obtient finalement un résultat de type nombre.
- On insère les flux que l'on vient de calculer dans les listes que nous avons initialisées précédemment.
- La boucle bou1 s'arrête ici.
- Ces évolutions permettent de créer une fonction entre la liste de temps et la liste des valeurs de flux de chaque paroi.
- A noter que ces évolutions peuvent être représentées sous forme de graphe (voir ci-dessous).
- Cette table *tab2* ajoute une légende et un titre au graphe représentant les flux.



```

flutoe= intg evolext;
flutoi= intg evolint;
flutpl1e= intg evolpl1e;
flutpl1i= intg evolpl1i;
flutpl3e= intg evolpl3e;
flutpl3i= intg evolpl3i;
flutpl2e= intg evolpl2e;
flutpl2i= intg evolpl2i;

flutoti=flutpl1i + flutpl2i + flutpl3i + flutoi;
flutote=flutpl1e + flutpl2e + flutpl3e + flutoe;

list flutoti;
list flutpl1i;
list flutpl2i;
list flutpl3i;
list flutoi;

lcou1=prog 4 pas 0.25 14;
opti trac psc ftra 'actuel.ps';
ms=60;
hs=60*ms;
js=24*hs;
repete bou1 n1;
i=&bou1-1;
t1=tab1.temps.i;
t1=t1/hs;
nj=(t1/24);
nj=entier nj+1;
t2=t1/24 - (nj-1)*24;
list nj;
list t2;
titre "Temperature peau externe a ' nj 'jours et '
t2 ' heures";
trac (exco tab1.temperatures.i 'TSUP') (smac et
dall3 et dall2) LCOU1;
fin bou1;

```

- On réalise une intégration des évolutions pour pouvoir calculer la valeur du flux.
- On demande d'afficher avec l'opérateur LIST la valeur des flux.
- Les flux sont donnés en J.s. Il est plus intéressant de les convertir en kWh pour une meilleure compréhension des résultats.
- Ce programme place dans un fichier .ps toutes les images du bâtiment avec les températures affectées aux parois à chaque itération de temps.
- La boucle bou1 permet d'afficher à chaque itération de temps, le jour et l'heure à laquelle on affiche les valeurs de température de la peau externe.
- La fonction lcou1 permet de fixer l'échelle de couleur de représentation des températures.
- Ce fichier .ps peut être transformé en .gif pour un meilleur rendu visuel des résultats.

**ANALYSE, INTERPRETATION ET SENSIBILITE DES RESULTATS**

Les résultats du calcul thermique dépendent de plusieurs paramètres :

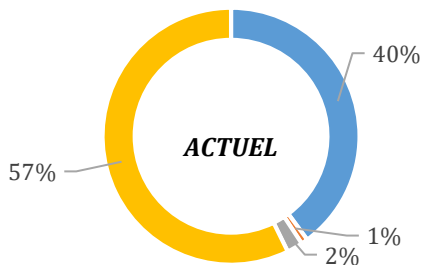
- Les **caractéristiques intrinsèques des matériaux** ;
- La géométrie et plus précisément le **maillage** du bâtiment peut jouer un rôle dans l'expression des résultats ;
- Les **coefficients de convection** ont un rôle prépondérant dans le calcul. Ils doivent donc être déterminés réglementairement.

Comme nous l'avons vu dans la description du programme, nous avons modélisé deux couches pour les murs périphériques. Une représentant la maçonnerie et l'autre la couche d'isolant. Dans un premier temps, nous avons calculé le bâtiment tel qu'il est aujourd'hui, c'est à dire avec des *blocs de béton* et un *parement en brique de terre cuite*. Ensuite, nous avons changé cette couche d'isolant en mettant *10cm de laine minérale type laine de roche*.

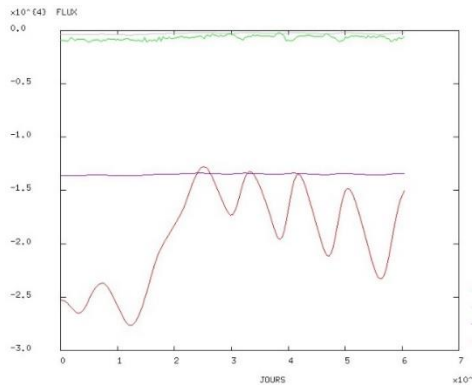
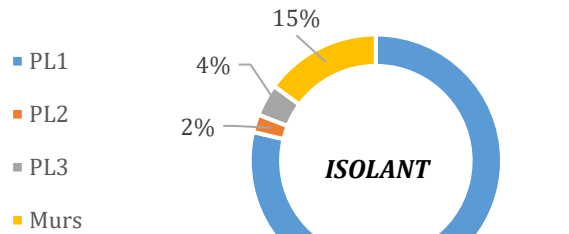
ACTUEL		
	J	Kwh
<i>Flux tot</i>	2,04E+10	5,67E+03
<i>PL1</i>	8,15E+09	2,26E+03
<i>PL2</i>	1,70E+08	4,72E+01
<i>PL3</i>	4,09E+08	1,14E+02
<i>Murs</i>	1,17E+10	3,25E+03

ISOLATION DES MURS		
	J	Kwh
<i>Flux tot</i>	1,04E+10	2,90E+03
<i>PL1</i>	8,22E+09	2,28E+03
<i>PL2</i>	2,40E+08	6,67E+01
<i>PL3</i>	4,30E+08	1,19E+02
<i>Murs</i>	1,56E+09	4,33E+02

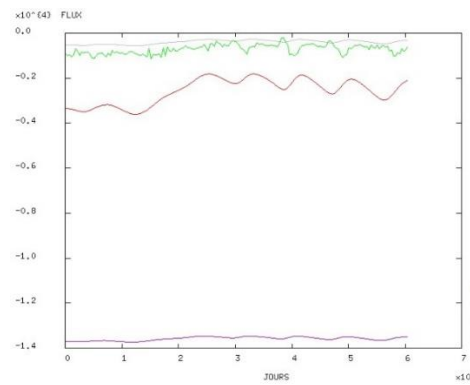
**RÉPARTITION DU FLUX**



**RÉPARTITION DU FLUX**



— Plancher haut  
— Plancher bas  
— Plancher tage



— Plancher bas  
— Murs extérieurs



On diminue l'énergie nécessaire pour chauffer le bâtiment par deux en mettant une isolation contre les murs périphériques. La toiture étant déjà isolée les plus grandes déperditions proviennent du **plancher bas** où seule une dalle de béton armé de 15cm fait office d'isolation et de structure.

Pour aller plus loin, il serait intéressant de modéliser une couche *d'isolant sous le plancher bas* du bâtiment. Il faudrait pour cela, réaliser la même procédure que pour les murs en modélisant deux couches avec des relations de conditions thermiques entre les deux. Il faudra veiller à éliminer les nœuds en double et vérifier l'orientation des surfaces pour qu'elles soient orientées dans la même direction.

Regardons maintenant l'incidence du **maillage** sur les résultats. Il est possible de différencier l'affinage du maillage entre les parois verticales et les planchers. C'est ce que nous avons fait pour voir s'il y avait une réelle incidence sur les résultats et si les déperditions seraient plus marquées dans la visualisation des résultats.

VALEURS DU FLUX EN FONCTION DU MAILLAGE DES MURS PERIPHERIQUES SUR 7 JOURS					
kWh	flutot	flupl1	flupl2	flupl3	flumur
<b>Actuel</b>					
Densité 1	5 666,67	2 263,89	47,50	113,61	3 250,00
0.5/0.5	5 666,67	2 261,11	43,33	113,33	3 250,00
0.3/0.3	5 666,67	2 261,11	40,00	113,33	3 250,00
kWh	flutot	flupl1	flupl2	flupl3	flumur
<b>Isolation des murs</b>					
Densité 1	5 222,22	2 263,89	50,56	114,44	2 805,56
0.5/0.5	5 222,22	2 263,89	46,11	114,44	2 805,56
0.3/0.3	5 222,22	2 263,89	42,50	114,44	2 805,56

Nous pouvons voir que seules les déperditions dues au *plancher intermédiaire* (ponts thermiques) changent en fonction du maillage. **Plus le maillage est réduit, plus les déperditions diminuent.** La puissance de notre ordinateur ne nous permet pas de réaliser le calcul avec un maillage plus fin que 0.3/0.3.

Aussi, nous pourrions effectuer la même étude sur une plus longue durée (1 mois voire une année) pour corroborer nos résultats. Cependant, là encore nous atteignons la **limite de puissance de nos machines informatiques.**

Les *menuiseries* n'ont pas été prises en compte dans cette étude. Il nous semble que si l'on voudrait étudier les ponts thermiques et les flux dus aux ouvertures, il faudrait mailler la surface des ouvertures et mettre des relations de blocages thermiques entre les différents maillages. Cela augmenterait la complexité du programme ainsi que le temps de calcul.

## CONCLUSION

---

Ce projet d'introduction aux éléments finis avec le logiciel Cast3m nous a permis de nous familiariser avec la **méthode aux éléments finis** et de mettre en pratique la théorie vue en cours magistral.

La méthode des éléments finis est utilisée pour résoudre numériquement des équations aux dérivées partielles. Concrètement, cela permet de **calculer numériquement le comportement d'objets** à condition qu'ils soient *continus* et *décrits par une équation aux dérivées partielles linéaires*.

Cast3m est un logiciel de simulation numérique développé par le Département de Modélisation des Systèmes et Structures. Il utilise la méthode des éléments finis pour résoudre différents types de problèmes scientifiques (mécaniques ou thermiques).

Concernant notre projet, plusieurs éléments, que nous pourrions réutiliser ultérieurement pour nos futurs projets, sont à mettre en lumière :

- La méthode utilisée pour *modéliser la géométrie du bâtiment* : la table « tabtyp » qui différencie les valeurs nulles pour des ouvertures des valeurs unitaires pour remplir les surfaces (méthode binaire) ;
- Modéliser *deux couches* pour la maçonnerie et l'isolant : les relations de blocages de températures sont indispensables au calcul et permettent d'avoir un **lien entre intérieur et extérieur** ;
- Utiliser des *éléments coques* pour modéliser les différentes surfaces. Une étude consisterait à mener la même étude mais avec d'autres types d'éléments et comparer ensuite les résultats ;
- Le calcul en fonction du *temps* et des *DJU* (variation de la température extérieure) permet de résoudre les équations thermiques sur n'importe quel pas de temps que l'on souhaite. Cela serait intéressant de comparer l'énergie nécessaire au refroidissement du bâtiment l'été à l'énergie nécessaire au chauffage l'hiver. Cependant, les calculs sont conséquents et peuvent prendre beaucoup de temps en fonction de la précision souhaitée ;
- Les résultats obtenus peuvent être comparés à ceux obtenus par simulation thermique dynamique. Seules les *déperditions surfaciques* sont comparables. La STD prend aussi en compte les déperditions par renouvellement d'air.

