

Le bon usage de **struct** comme type concret

Mettre le modèle de **struct** dans l'interface permet de déclarer des variables de ce type et d'accéder/modifier librement les champs.

Il est pertinent d'exploiter **struct** pour des types concrets de **bas-niveau** pour lesquels les risques d'utilisations incorrectes sont faibles et l'interface est stable.

Par exemple, pour le projet, on définira un module tools avec des types concret permettant de travailler avec des point, vecteur, segment, cercle, etc.. Ici un type **S2d** est défini ; il permet de représenter aussi bien un point qu'un vecteur dans le plan 2D.

```
#include <iostream>
#include "tools.h"
using namespace std;
int main(void)
{
    S2d centre={0.,0.};
    centre.x = 1.;
    centre.y = 1.;
    ...
}
```

prog.cc

```
struct S2d ;
{
    double x;
    double y;
};
+ autres struct éventuels
+ prototypes de fonctions
```

tools.h

```
#include <tools.h>

// Définition des fonctions
```

tools.cc