**Architecture**

Notre programme est divisé en différents modules qui sont les suivants :

**Parser**

Le parser se charge de lire les arguments qui sont données en entrée. Pour ce faire il utilise la fonction *getopt\_long\_only.*

Le parser permet de lire et de traiter les options suivantes :

-**name**

Cette option permet de spécifier un nom exact pour les fichiers à rechercher.  
La syntaxe est **-name exactName**, où *exactName* est le nom exact du fichier.

Exemple : *-name source.c*

-**name\_contain**

Cette option permet de spécifier une sous-chaine qui se trouve dans le nom des fichiers à rechercher.

La syntaxe est **-name\_contain partOfName**, où *partOfName* est une sous-chaine de caractère qui se trouve dans le fichier à rechercher.

Exemple : -name\_containt source

**-size**

Cette option permet de spécifier la taille des fichiers à rechercher.

La syntaxe est -**size** [+/-/RIEN]taille[unite] où taille est la taille du fichier de l’ordre de l’unité. L’unité peut être b(byte), k(kilobyte), m(megabyte) ou g(gigabyte). Si aucune unité n’est spécifiée, l’unité par défaut sera le byte. Le flag [+/-/RIEN] permet de spécifier si la taille du fichier doit être inférieure (-), supérieure(+) ou égale(RIEN) à la taille donnée.

Exemple : -size -2g

**-date**

Cette option permet de spécifier la date du fichier. Celle-ci peut être la date d’accès, de création ou de modification.

La syntaxe est **-date [+/-/RIEN]date[c/m/u],** où date est la date du ficher sous le format YYYY-MM-JJ. Le flag [+/-/RIEN] permet de spécifier si la date du fichier doit être inférieure (-), supérieure(+) ou égale(RIEN) à la date donnée. Le flag c/m/u permet de spécifier s’il s’agit de la date d’accès(u), de modification(m) ou de création(c). Si aucun de ces trois flags n’est spécifié, le flag de la date d’accès(u) sera sélectionné.

Exemple : -date -2016-05-12c

**-owner**

Cette option permet de spécifier le propriétaire du fichier. Celui-ci peut être un utilisateur ou un groupe.

La syntaxe est **-owner [RIEN/!]number[u/g],** ou *number* est l’id du propriétaire. Le flag ! permet de choisir sa négation. Le flag u/g permet de spécifier s’il s’agit d’un utilisateur(u) ou d’un groupe(g).

Exemple : *-owner 14299u*

**-perm**

Cette option permet de spécifier les droits du ficher.

La syntaxe est **-perm *permission***, ou *permission* sont les droits sous le format rwxrwxrwx. Chaque droit peut être remplacé par un n (n) s’il n’est pas présent. Le droit peut aussi être remplacé par un astérisque (\*) s’il ne doit pas être pris en compte. L’ordre des droits est le suivant : [USER\_READ/ USER\_WRITE/USER\_EXEC][GROUP:R/W/X][OTHER:R/W/X].

Exemple : *-perm rwxrnn\*\*\**

Tous les arguments peuvent être utilisé avec leur négation en utilisant l’argument **-not**.

Les arguments peuvent également être combinées avec les options « et » (**-and**) et/ou « ou » (**-or**).

La notation polonaise doit être utilisée.

Exemple :

**-name\_contain .mkv -name\_contain .mp4 -or -size +10g -and**

Ce qui correspond à (name\_contain .mkv ou name\_contains .mp4) et (size +10g).

Les arguments sont vérifiés. Si un argument ne peut pas être exploité correctement il est ignoré. Ceux qui sont vérifiées et donc valides sont ajouté dans la structure de donnée suivante :

C:\Users\matth\AppData\Local\Microsoft\Windows\INetCacheContent.Word\diagram.png

Cette structure de donnée est générique est utilisée pour tous les arguments de la même manière.

Le type contient le type de paramètre (and, or, not, name, size, date, etc..).  
L’opération contient le modificateur pour l’opération (Valeur exacte, +, -, not).  
Le flag contient des informations supplémentaires en fonction du type. La spécification de la date (création, modification, utilisation) se trouve ici par exemple.  
Le String contient la valeur de l’argument. Celle-ci est directement utilisable par le filtre en fonction des autres données (type, opération et flag).

Cette structure est donnée au filtre qui l’exploitera.