

Projet de TPA

Réalisation d'un IDE en Python



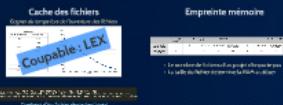
- CARREAU Alexis
- LÉCLUSE Thomas
- MAUGER Emma
- SARRAZIN Théo

Licence professionnelle
de seconde année

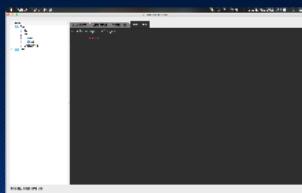
2016-2017

Optimisation & mémoire

Être rapide tout en restant efficace



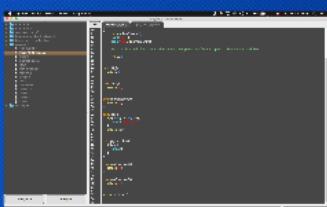
Précédemment...



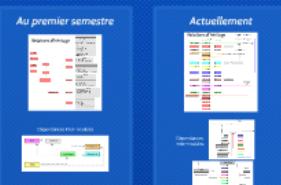
Travail réalisé au cours du semestre :

- Différents thèmes et styles pour l'éditeur et ses éléments
- Traitement des projets
- Fichiers de configuration au format XML
- Intégration d'un compilateur
- Fonctionnalités diverses de recherche et d'édition de texte
- Inspecteur d'éléments
- Une autre barre de statut pour d'autres types d'informations
- La numérotation des lignes
- Fenêtre de paramétrage des raccourcis
- Ajout de cache afin d'optimiser l'ouverture de fichiers
- Grammaire arithmétique puis grammaire python
- Ajout de l'Anglais
- Quelques bonus...

...Maintenant



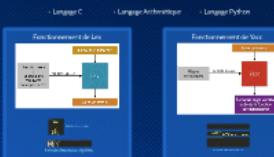
Architecture du projet



Paramétrage : choix des raccourcis et de la langue



Les grammaires



Édition

Recherche, édition du code & Inspecteur d'éléments

- Ajout de fonctionnalités typiques d'édition
- Recherche de mots
 - Sélections diverses
 - Insertion, indentation
 - Commenter une sélection

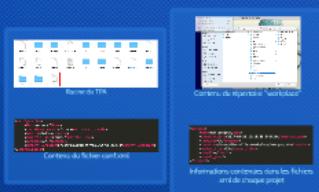
- Ajout d'un inspecteur d'éléments pour lister :
- Les variables
 - Les fonctions
 - Les structures (en C)
 - Les classes (en Python)

Intégration du compilateur

- Langage C : Utilisation de GCC
- Langage Python : Utilisation de l'interpréteur Python



Fichiers XML & gestion des projets



Projet de TPA

Réalisation d'un IDE en Python

BONUS !!!



Licence informatique
deuxième année

2016-2017

- CARREAU Alexis
- LÉCLUSE Thomas
- MAUGER Emma
- SARRAZIN Théo

Précédemment...

The screenshot shows a Mac OS X desktop environment. At the top, there's a dark blue header bar with the Apple logo, the word "Python", and menu items "Fichier" and "Projet". To the right of the header are various system icons, including battery level (79%), signal strength, and a search icon. The date and time "Mer. 30 nov. 21:50" are also visible.

The main window is titled "Cthulhu (Bleu Royal)". It has a toolbar with four tabs: "Sans nom1", "Sans nom2", "Sans nom3", and "Sans nom4". The "Sans nom4" tab is currently selected and active. The main pane displays the following C code:

```
int main ( int argc, char** argv )  
{  
    return 0;  
}
```

On the left side of the window, there's a sidebar titled "Name" which lists a project named "Test" containing several files: "3.c", "4.c", "Untitled.c", "Untitled2.c", and "Untitled66.c". Below the sidebar, a status bar at the bottom of the window says "Nouveau fichier Sans nom4".

Travail réalisé au cours du semestre :

- Différents thèmes et styles pour l'éditeur et ses éléments
- Traitement des projets
- Fichiers de configuration au format XML
- Intégration d'un compilateur
- Fonctionnalités diverses de recherche et d'édition de texte
- Inspecteur d'éléments
- Une autre barre de statut pour d'autres types d'informations
- La numérotation des lignes
- Fenêtre de paramétrage des raccourcis
- Ajout de cache afin d'optimiser l'ouverture de fichiers
- Grammaire arithmétique puis grammaire python
- Ajout de l'Anglais
- Quelques bonus...

...Maintenant

The screenshot shows the SpaghettiIDE interface on a Mac OS X desktop. The title bar reads "Python Fichier Edition Projet SpaghettiIDE". The menu bar includes "Fichier", "Edition", "Projet", and "SpaghettiIDE". The system tray shows battery level at 80%, network, and date/time "Jeu. 27 avr. 11:03".

The left sidebar displays a file tree:

- autre projet
- autre projet 1
- bleu_royal_ide_g3
- dm-programmation-fonctionnelle
- dm-système-d-exploitation
- mon projet
 - battlefield10.c
 - copie_1000_lignes.c
 - empty.h
 - exemple_code.c
 - fic1.h
 - mon projet.xml
 - rainbow.c
 - swagg.c
 - test.c
 - unamed.txt
 - Untitled.c
 - Untitled1.c
 - Untitled2.c
 - yolo.h
- statistiques

The main window shows two tabs: "exemple_code.c" (selected) and "copie_1000_lignes.c". The code editor displays the following C code:

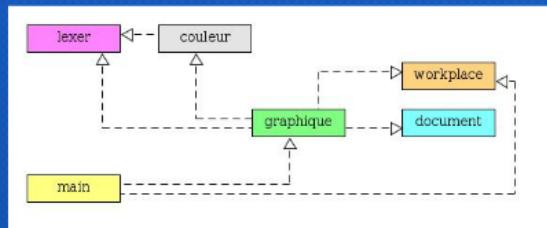
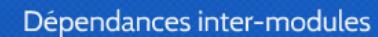
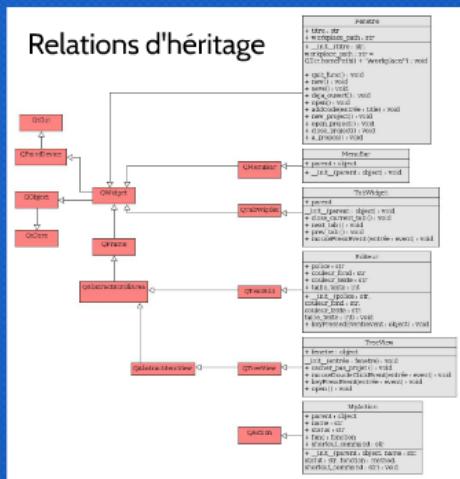
```
10 int nombreEntre = 0;
11 printf("text");
12 scanf("%d", &nombreEntre);
13
14 // Le résultat de la fonction est directement envoyé au printf et n est pas stocké dans une variable
15
16 return 0;
17 }
18
19 int ksfdg(){
20     return 0
21 }
22
23 char zbra(){
24     return "a";
25 }
26
27 char maFonction1(){
28     return "a";
29 }
30
31 bool btie(){
32     for(i = 0 ; i < 10 ; i++){
33         printf("Lugsdafuh")
34     }
35     return true;
36 }
37
38 void yfgsldkhhbfs(){
39     if(true){
40         printf();
41     }
42 }
43
44 char maFonction2(){
45     return "a";
46 }
47
48 char maFonction3(){
49     return "a";
50 }
51
52 struct maStruct{
53 }
```

At the bottom of the code editor, it says "55 lignes".

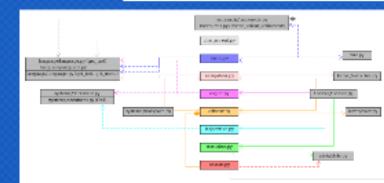
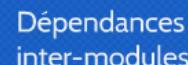
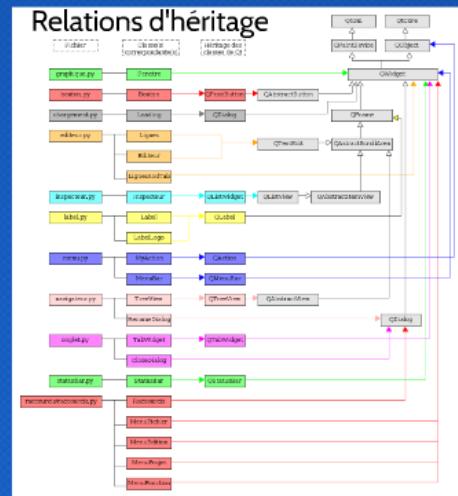
Bottom navigation buttons: "Navigateur" and "Analyse".

Architecture du projet

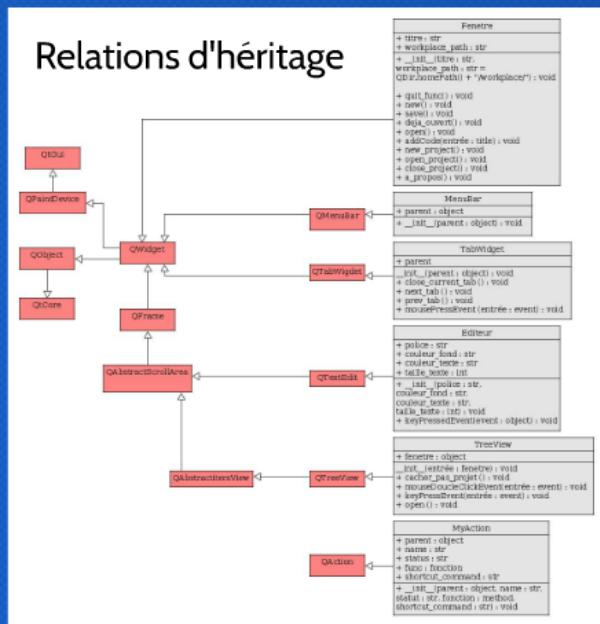
Au premier semestre



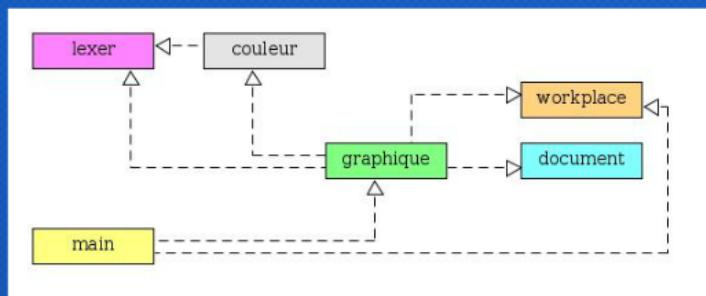
Actuellement



Au premier semestre

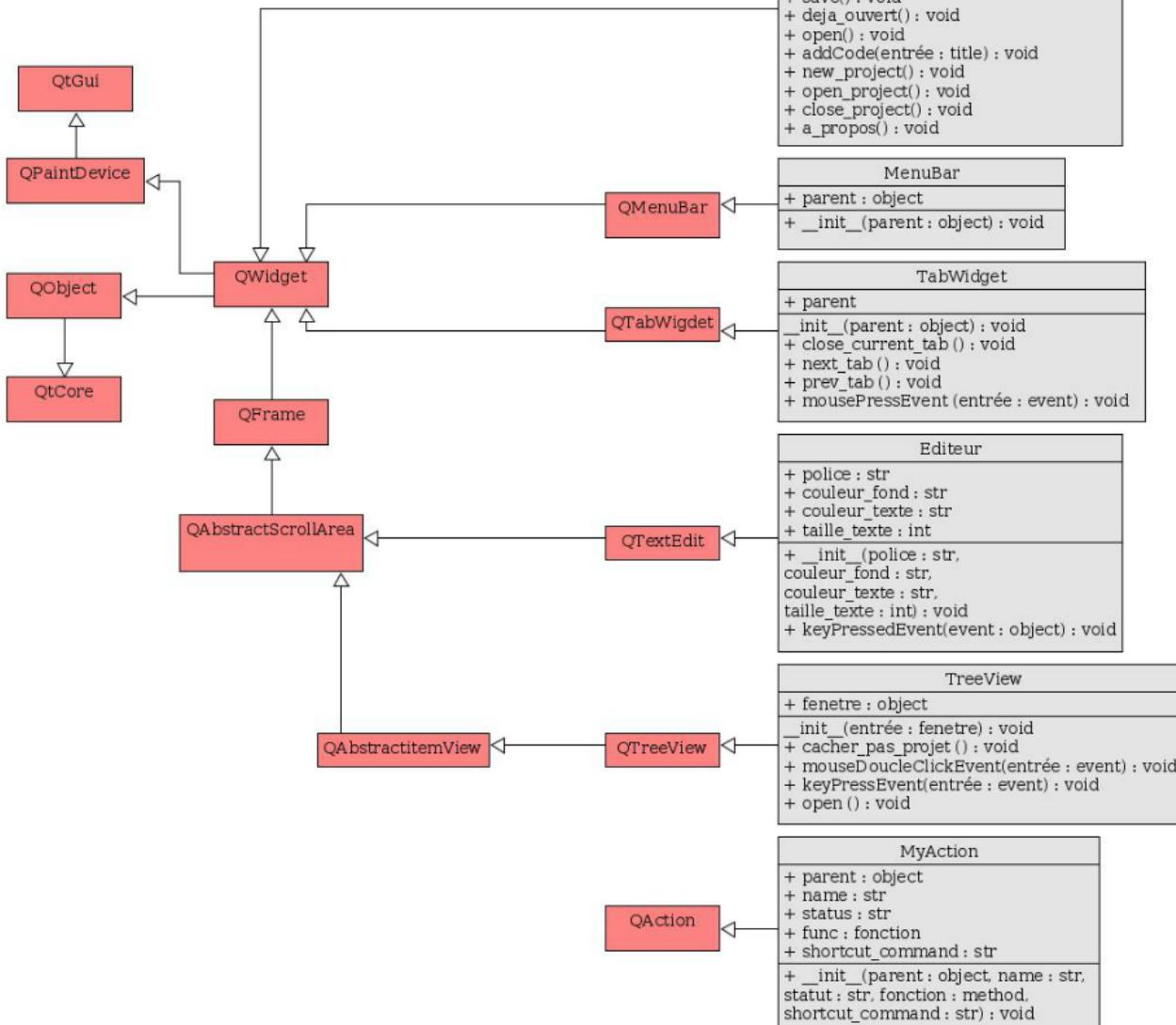


Dépendances inter-modules

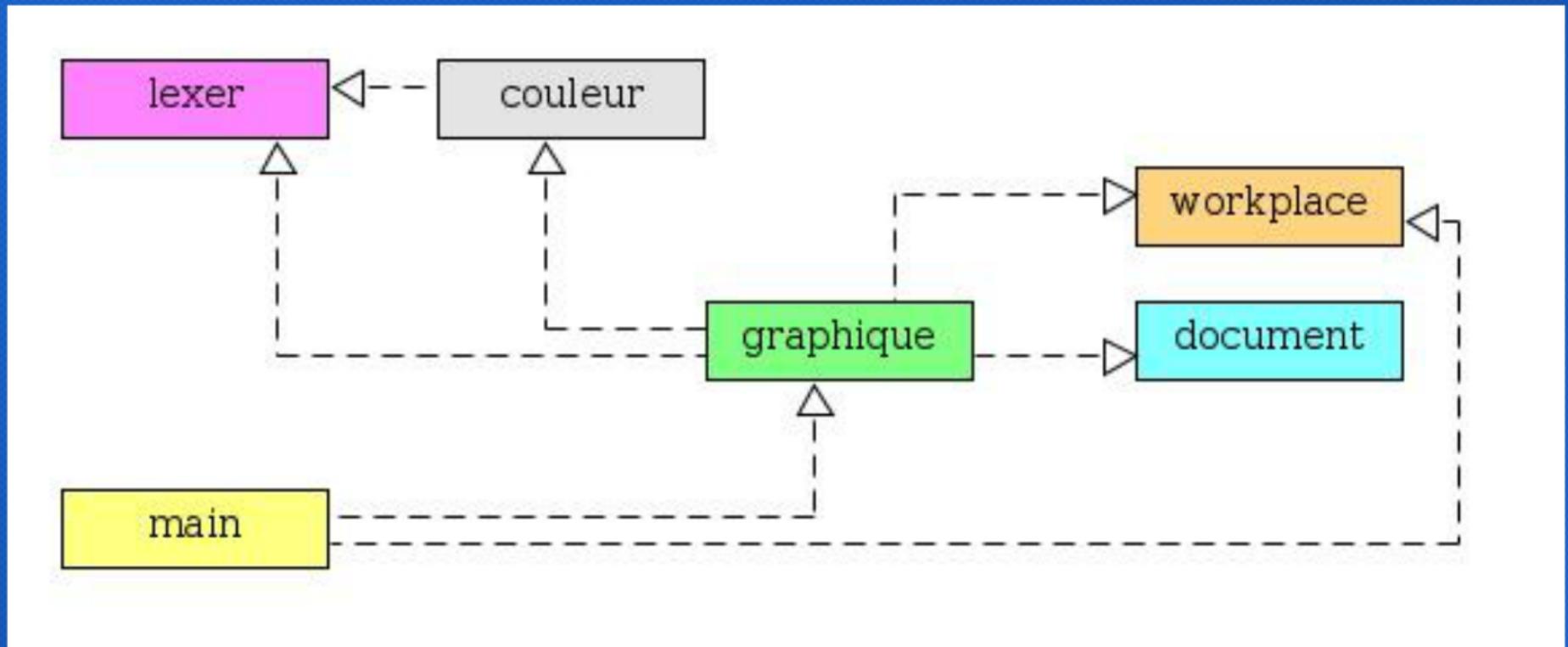


Dépendance inter-modale

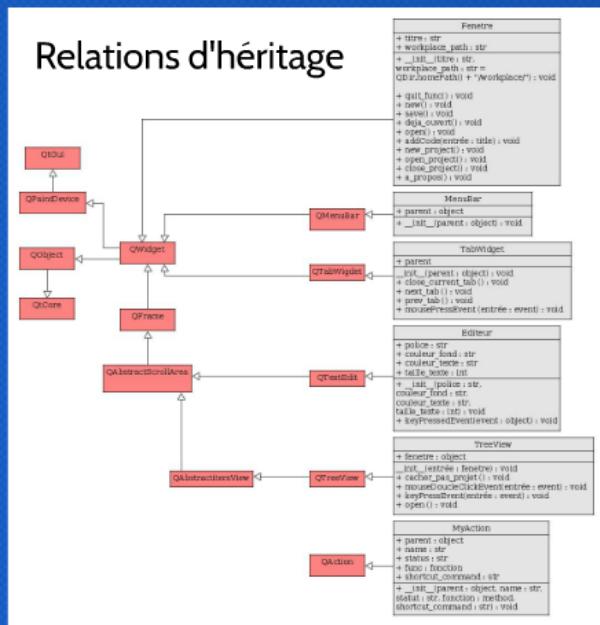
Relations d'héritage



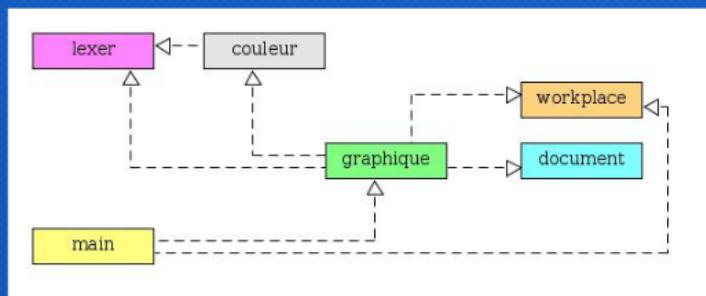
Dépendances inter-modules



Au premier semestre

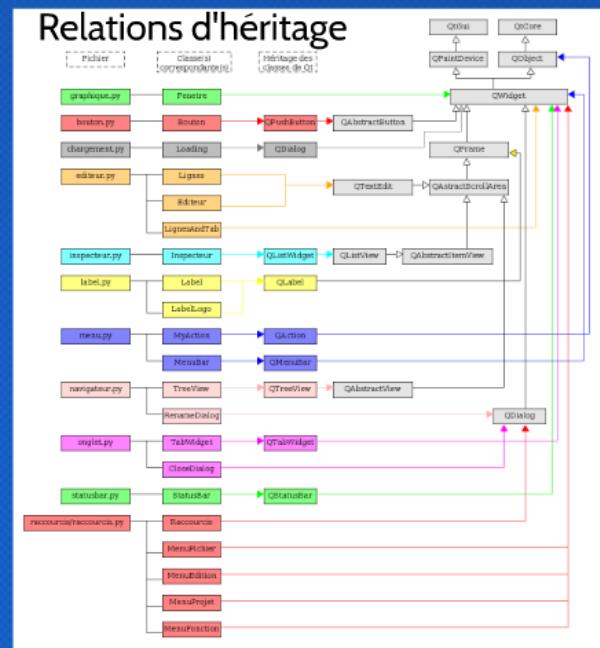


Dépendances inter-modules

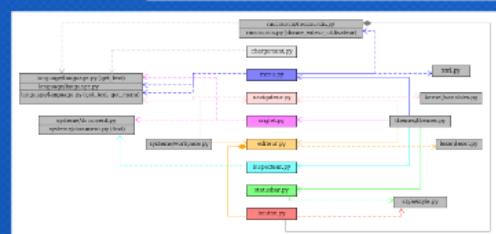
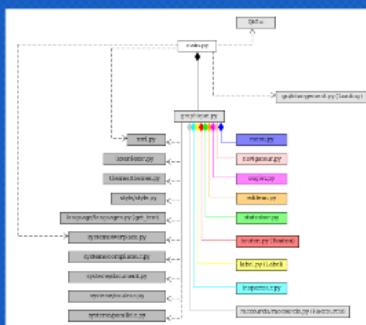


Dépendance inter-modale

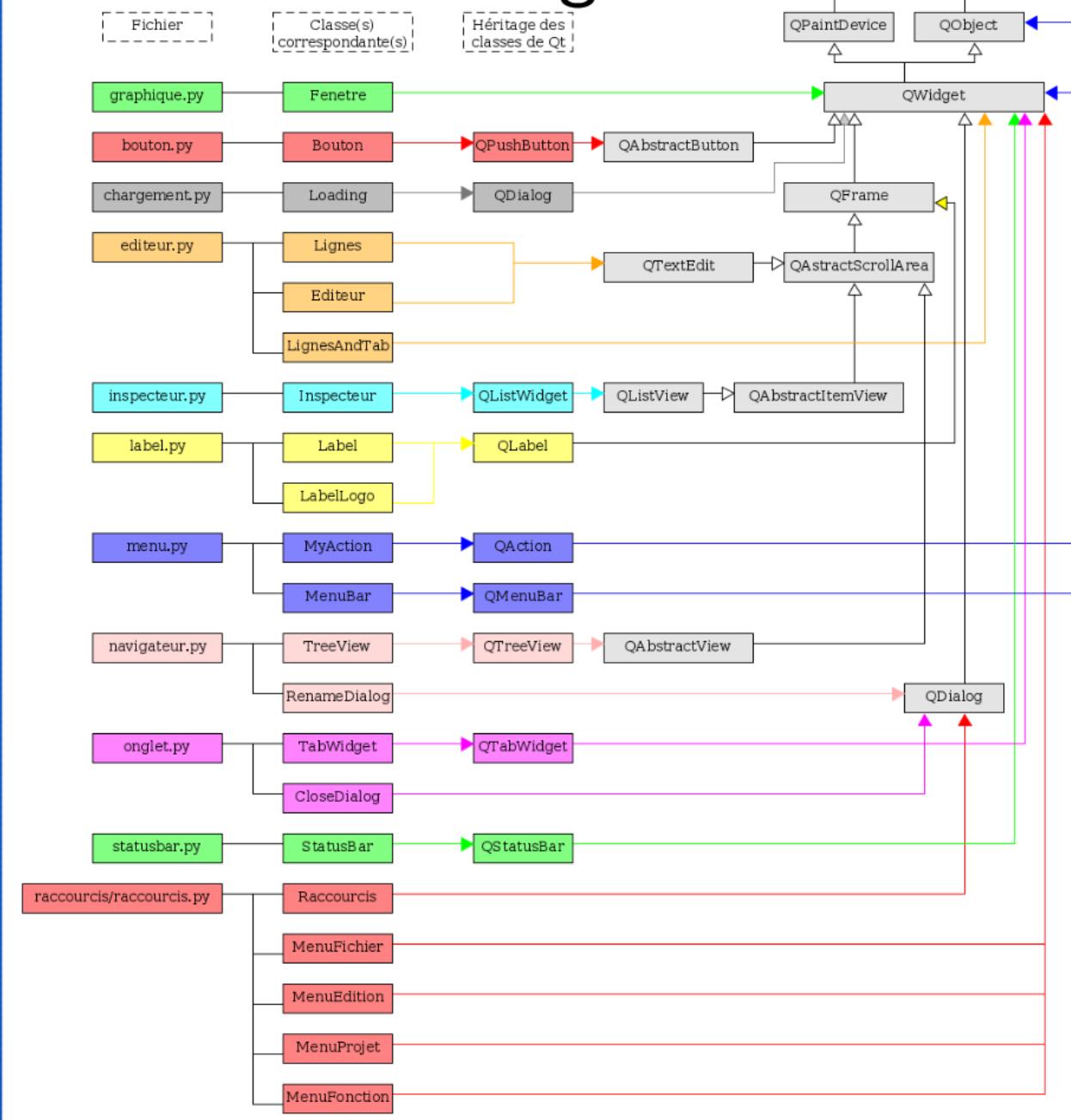
Actuellement



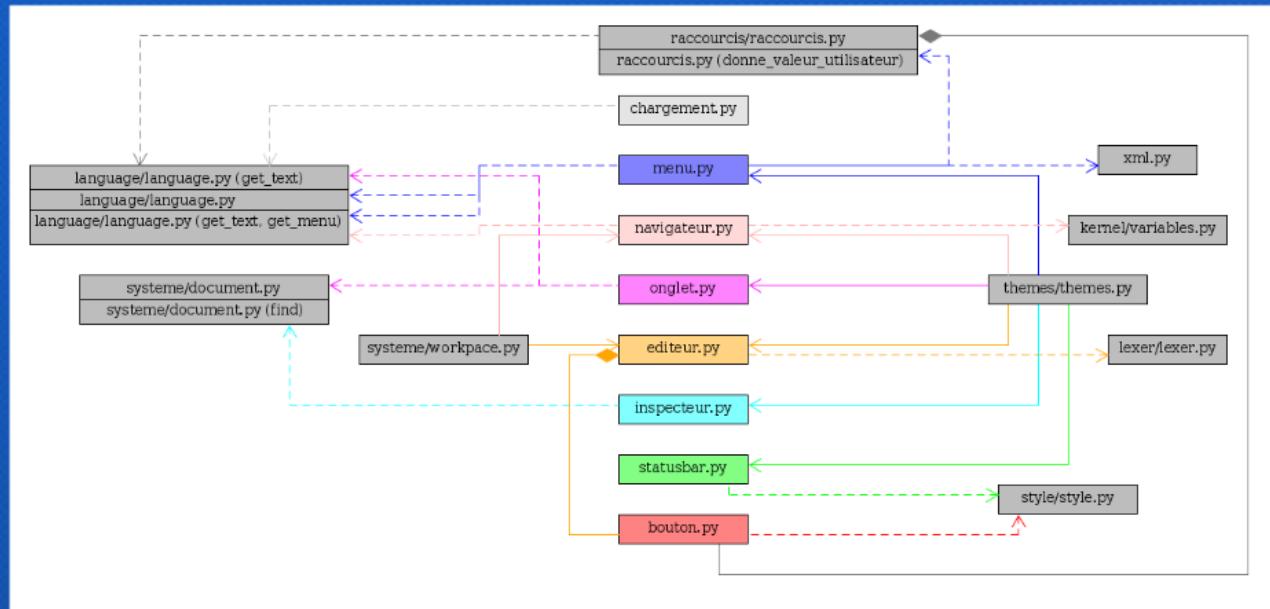
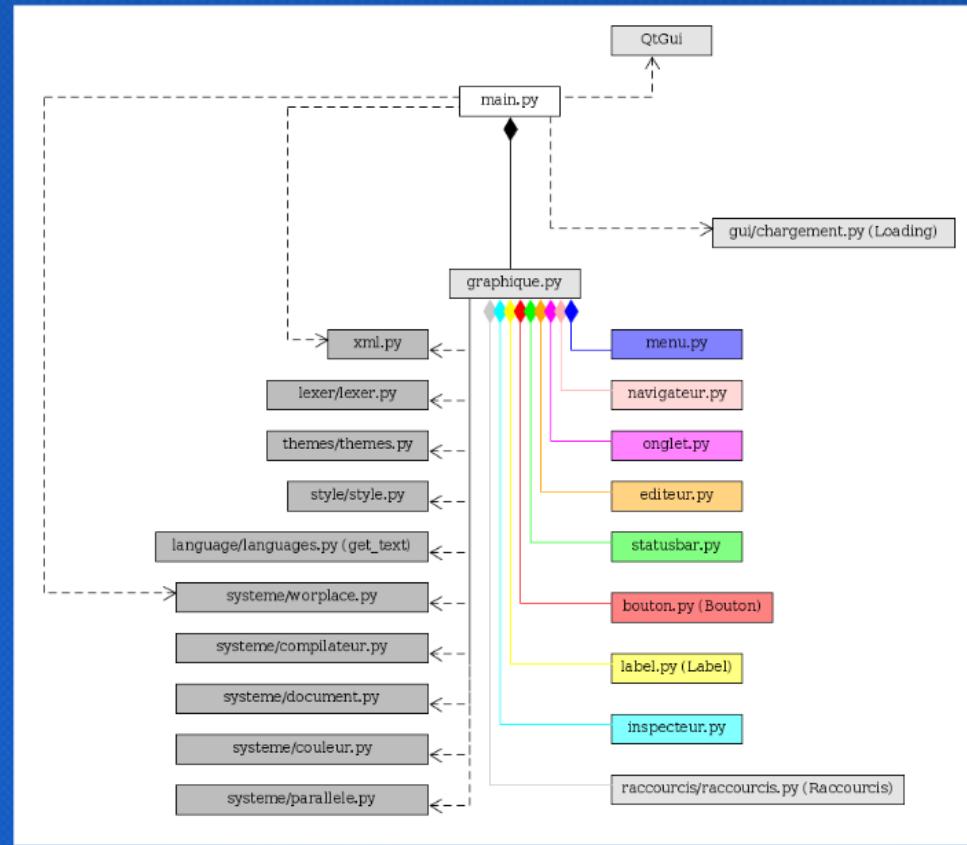
Dépendances inter-modules



Relations d'héritage

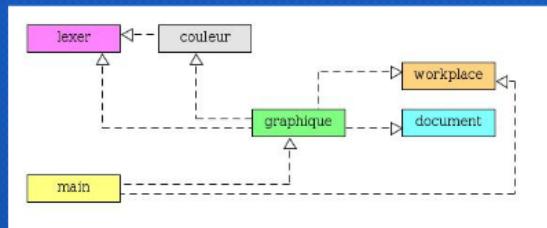
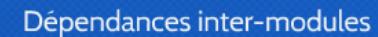
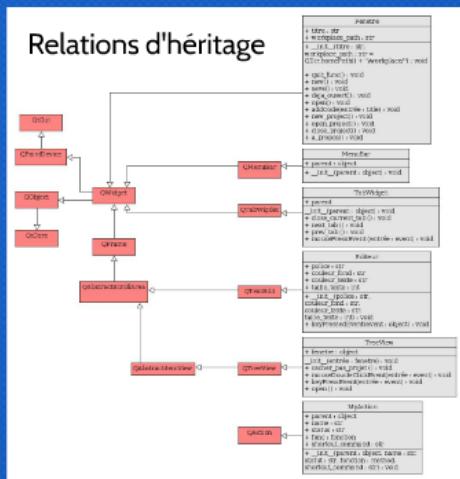


Dépendances inter-modules

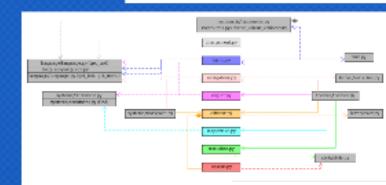
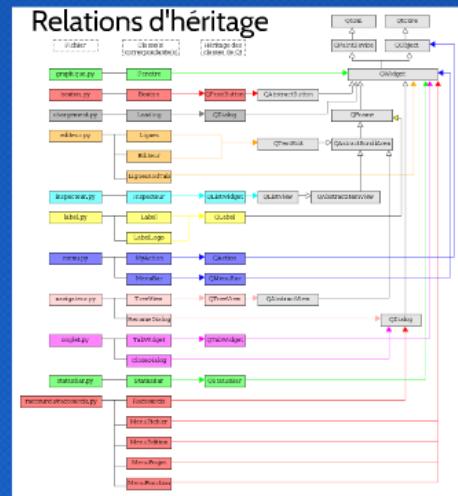


Architecture du projet

Au premier semestre

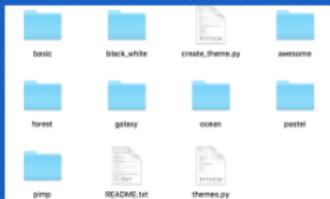


Actuellement

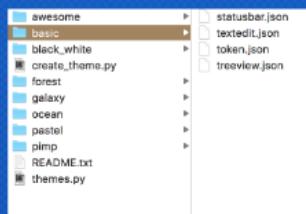


IHM (Interface Homme-Machine)

Thèmes du logiciel



Contenu du répertoire "themes/"



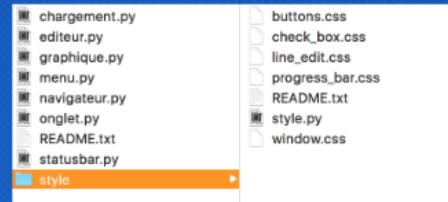
Contenu du thème "basic"

```
textedit.json
```

```
1  {
2      "text-back-color": [31, 31, 31],
3      "text-color": [255, 255, 255],
4      "tab-color": [0, 0, 0],
5      "tab-back-color": [100, 100, 100],
6      "tab-hover-back-color": [31, 31, 31],
7      "tab-hover-color": [255, 255, 255],
8      "tab-hover-border-color": [0, 0, 0]
9 }
```

Contenu du fichier textedit.json

Styles des éléments



Contenu du répertoire "style/"

Cancel

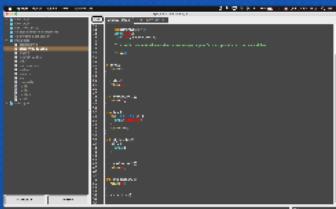
→ Bouton normal

Cancel

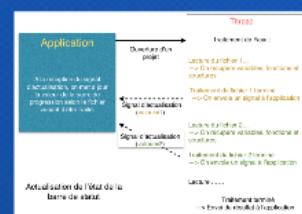
→ Bouton stylisé

Barre de statut & barre de progression

Une autre barre de statut :

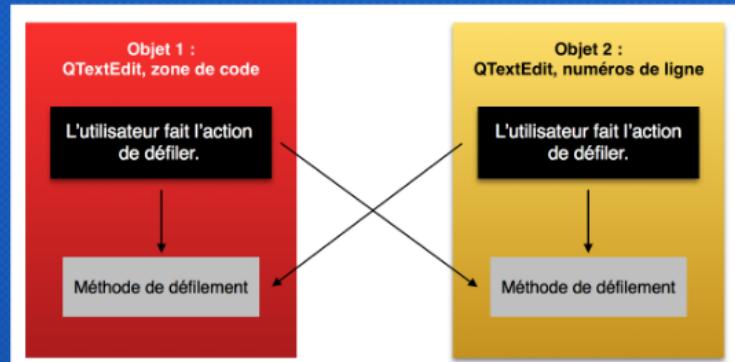


Lors du chargement d'un projet :
affichage d'une barre de progression.



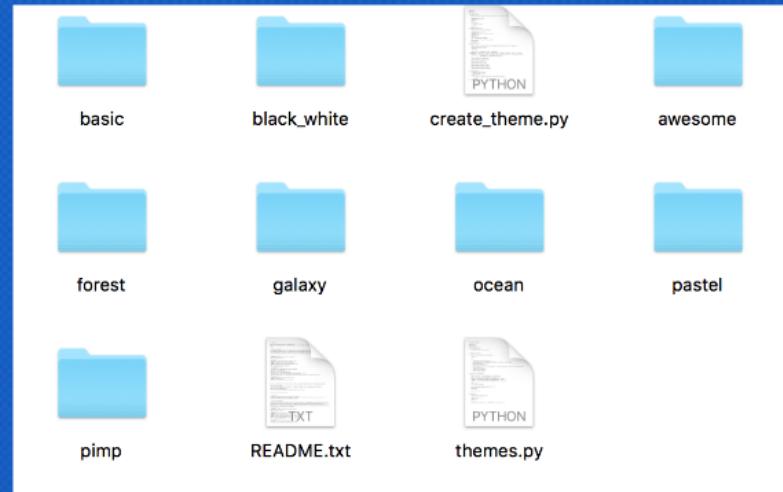
Numérotation des lignes

Problématique : Synchroniser le défilement.

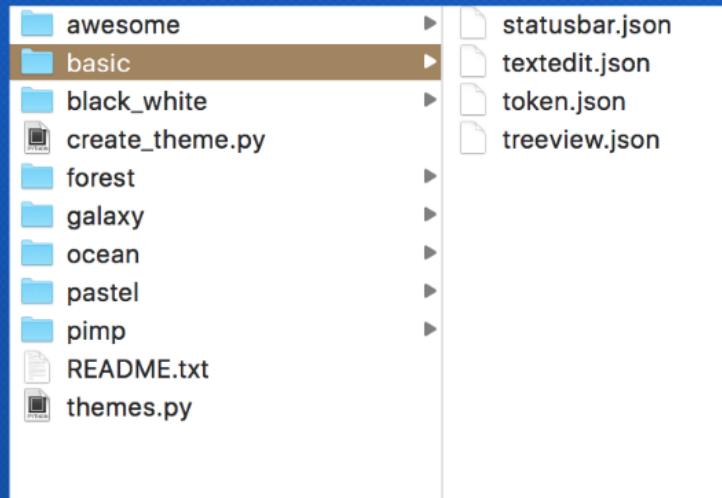


Prezi

Thèmes du logiciel



Contenu du répertoire "themes/"



Contenu du thème "basic"

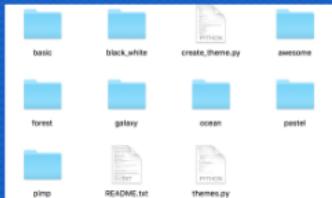
The image shows a code editor window with a dark background. It displays the 'textedit.json' file, which contains JSON configuration data for a text editor's appearance. The code is as follows:

```
1 {  
2     "text-back-color": [31, 31, 31],  
3     "text-color": [255, 255, 255],  
4     "tab-color": [0, 0, 0],  
5     "tab-back-color": [100, 100, 100],  
6     "tab-hover-back-color": [31, 31, 31],  
7     "tab-hover-color": [255, 255, 255],  
8     "tab-hover-bord-bot-color": [0, 0, 0]  
9 }
```

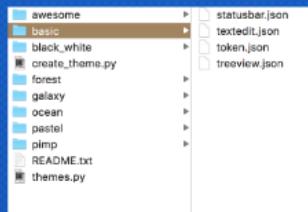
Contenu du fichier textedit.json

IHM (Interface Homme-Machine)

Thèmes du logiciel



Contenu du répertoire "themes/"



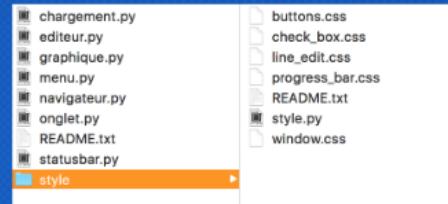
Contenu du thème "basic"

```
textedit.json
```

```
1  {
2      "text-back-color": [31, 31, 31],
3      "text-color": [255, 255, 255],
4      "tab-color": [0, 0, 0],
5      "tab-back-color": [100, 100, 100],
6      "tab-hover-back-color": [31, 31, 31],
7      "tab-hover-color": [255, 255, 255],
8      "tab-hover-border-color": [0, 0, 0]
9 }
```

Contenu du fichier textedit.json

Styles des éléments



Contenu du répertoire "style/"

Cancel

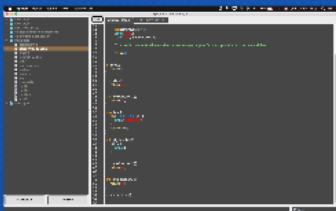
→ Bouton normal

Cancel

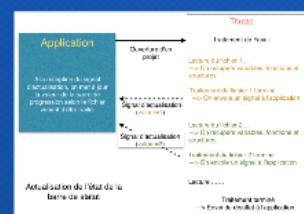
→ Bouton stylisé

Barre de statut & barre de progression

Une autre barre de statut :

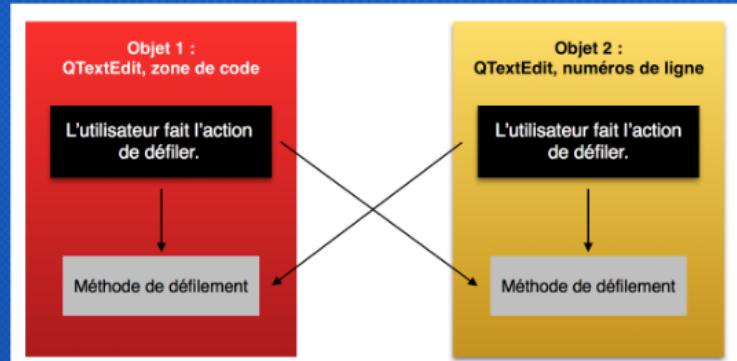


Lors du chargement d'un projet :
affichage d'une barre de progression.



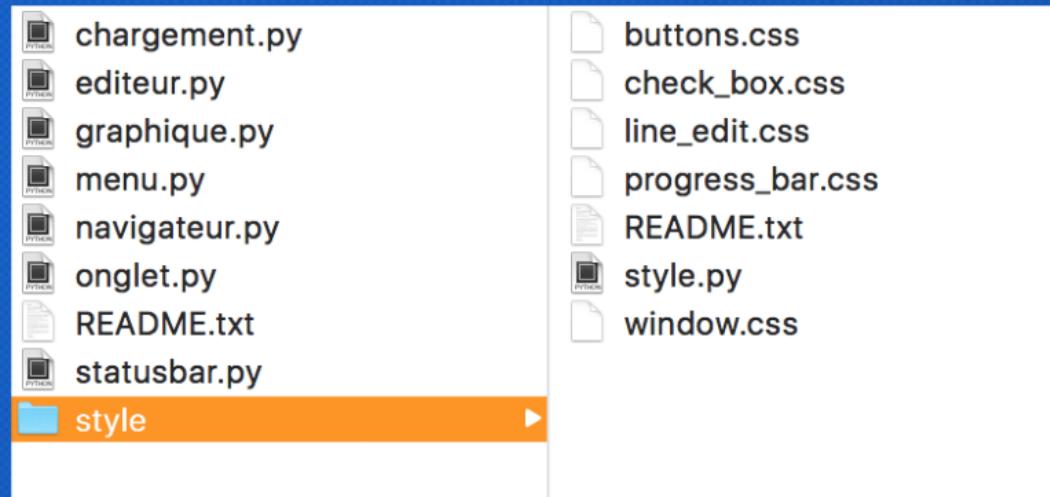
Numérotation des lignes

Problématique : Synchroniser le défilement.

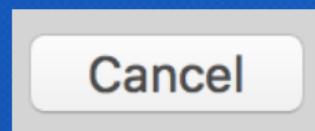


Prezi

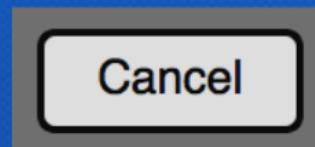
Styles des éléments



Contenu du répertoire "style/"



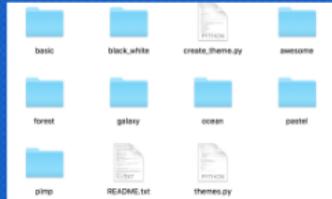
-> Bouton normal



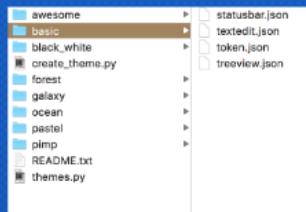
-> Bouton stylisé

IHM (Interface Homme-Machine)

Thèmes du logiciel



Contenu du répertoire "themes/"



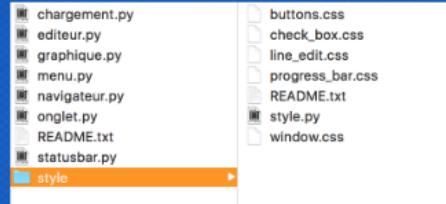
Contenu du thème "basic"

```
textedit.json
```

```
1  {
2      "text-back-color": [31, 31, 31],
3      "text-color": [255, 255, 255],
4      "tab-color": [0, 0, 0],
5      "tab-back-color": [100, 100, 100],
6      "tab-hover-back-color": [31, 31, 31],
7      "tab-hover-color": [255, 255, 255],
8      "tab-hover-border-color": [0, 0, 0]
9 }
```

Contenu du fichier textedit.json

Styles des éléments



Contenu du répertoire "style/"

Cancel

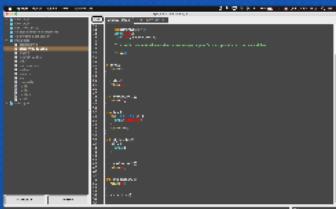
→ Bouton normal

Cancel

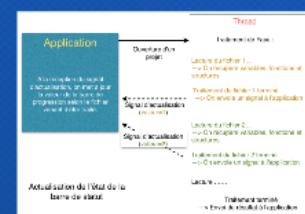
→ Bouton stylisé

Barre de statut & barre de progression

Une autre barre de statut :



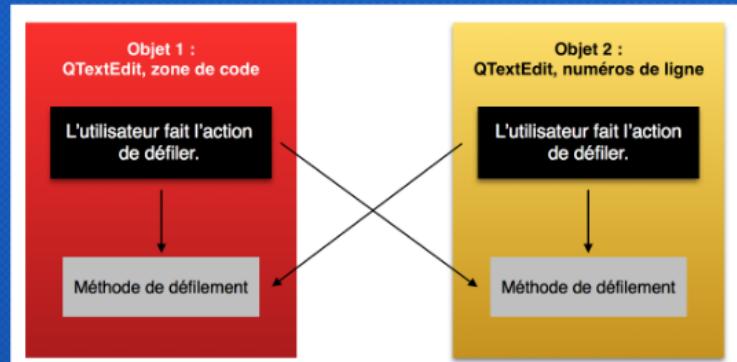
Lors du chargement d'un projet :
affichage d'une barre de progression.



- Nombre total de lignes
- Nombre total de propositions pour l'auto-complétion
- Nombre total d'occurrences trouvées lors d'une recherche

Numérotation des lignes

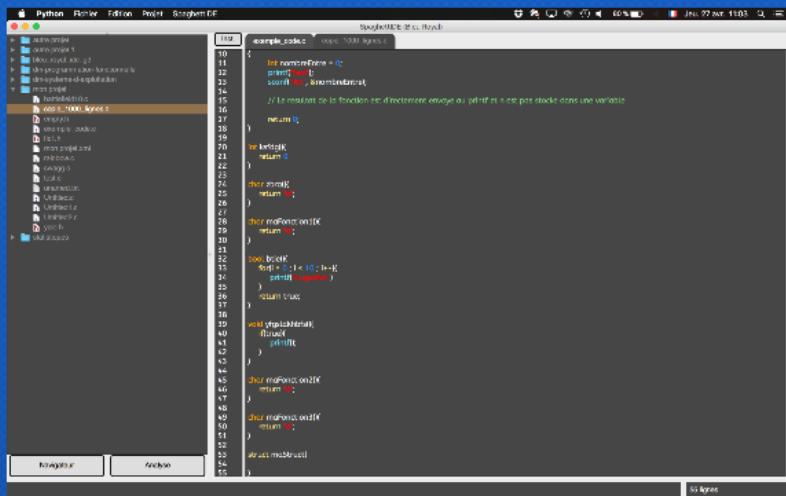
Problématique : Synchroniser le défilement.



Prezi

Barre de statut & barre de progression

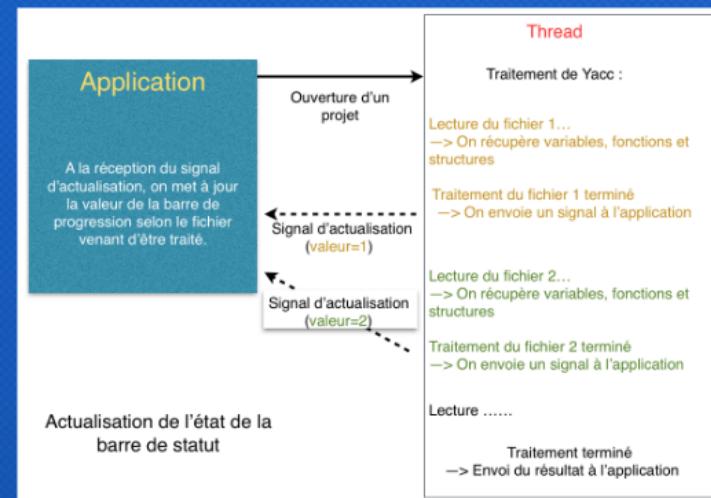
Une autre barre de statut :



A screenshot of a Mac OS X desktop environment showing a Python code editor window titled "example_code.py". The status bar at the bottom of the window displays "65 lignes". The code editor shows several lines of Python code, including imports and function definitions.

- Nombre total de lignes
- Nombre total de propositions pour l'auto-complétion
- Nombre total d'occurrences trouvées lors d'une recherche

Lors du chargement d'un projet :
affichage d'une barre de progression.



Une autre partie de statut .

Python Fichier Edition Projet SpaghettiIDE

SpaghettiIDE (Bleu Royal)

Inst. exemple_code.c copie_1000_lignes.c

```
10 int nombreEntre = 0;
11 printf("text");
12 scanf("%d", &nombreEntre);
13
14 // Le resultat de la fonction est directement envoye au printf et n est pas stocke dans une variable
15
16 return 0;
17
18
19 int ksfdg(){
20     return 0
21 }
22
23 char zbra(){
24     return "a";
25 }
26
27
28 char maFonction1(){
29     return "a";
30 }
31
32 bool btie(){
33     for(i = 0 ; i < 10 ; i++){
34         printf("lugsdfuh")
35     }
36     return true;
37 }
38
39 void yfgsldkhbfs(){
40     if(true){
41         printf();
42     }
43 }
44
45 char maFonction2(){
46     return "a";
47 }
48
49 char maFonction3(){
50     return "a";
51 }
52
53 struct maStruct{
54
55 }
```

Navigateur Analyse

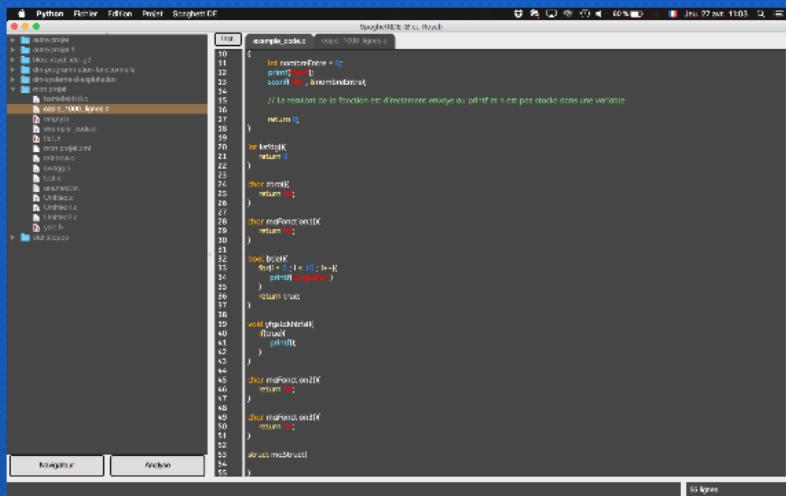
55 lignes



Prezi

Barre de statut & barre de progression

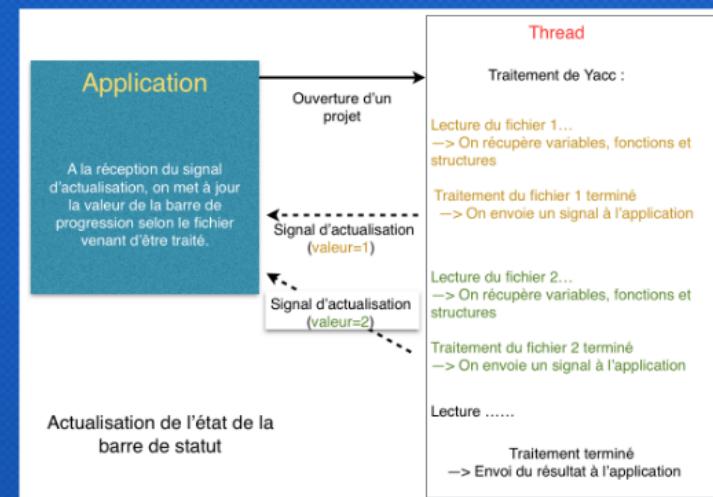
Une autre barre de statut :



A screenshot of a Python code editor window titled "example_code.py". The status bar at the bottom shows "65 lignes". The code editor displays several lines of Python code, including imports and function definitions.

- Nombre total de lignes
- Nombre total de propositions pour l'auto-complétion
- Nombre total d'occurrences trouvées lors d'une recherche

Lors du chargement d'un projet :
affichage d'une barre de progression.



Application

A la réception du signal d'actualisation, on met à jour la valeur de la barre de progression selon le fichier venant d'être traité.

Actualisation de l'état de la barre de statut

Ouverture d'un projet

Signal d'actualisation
(valeur=1)

Signal d'actualisation
(valeur=2)

Thread

Traitement de Yacc :

Lecture du fichier 1...

—> On récupère variables, fonctions et structures

Traitement du fichier 1 terminé

—> On envoie un signal à l'application

Lecture du fichier 2...

—> On récupère variables, fonctions et structures

Traitement du fichier 2 terminé

—> On envoie un signal à l'application

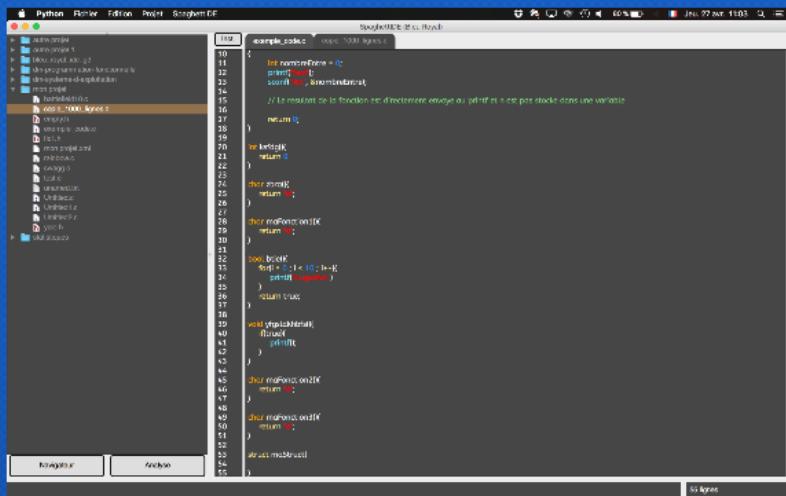
Lecture

Traitement terminé

—> Envoi du résultat à l'application

Barre de statut & barre de progression

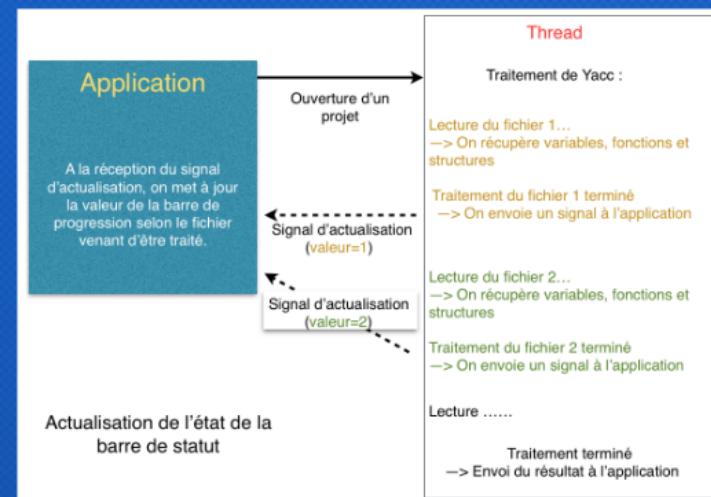
Une autre barre de statut :



A screenshot of a Python code editor window titled "example_code.py". The status bar at the bottom shows "65 lignes". The code editor displays several lines of Python code, including imports and function definitions.

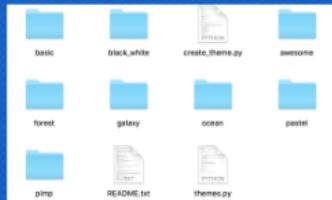
- Nombre total de lignes
- Nombre total de propositions pour l'auto-complétion
- Nombre total d'occurrences trouvées lors d'une recherche

Lors du chargement d'un projet :
affichage d'une barre de progression.

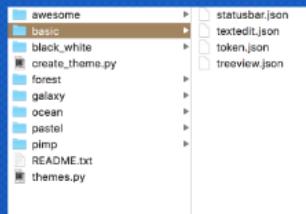


IHM (Interface Homme-Machine)

Thèmes du logiciel



Contenu du répertoire "themes/"



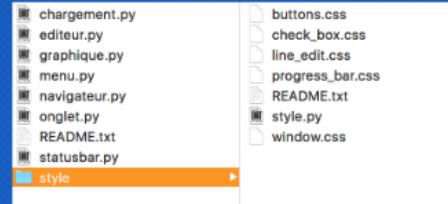
Contenu du thème "basic"

```
textedit.json
```

```
1  {
2      "text-back-color": [31, 31, 31],
3      "text-color": [255, 255, 255],
4      "tab-color": [0, 0, 0],
5      "tab-back-color": [100, 100, 100],
6      "tab-hover-back-color": [31, 31, 31],
7      "tab-hover-color": [255, 255, 255],
8      "tab-hover-border-color": [0, 0, 0]
9 }
```

Contenu du fichier textedit.json

Styles des éléments



Contenu du répertoire "style/"

Cancel

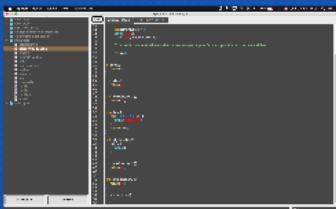
→ Bouton normal

Cancel

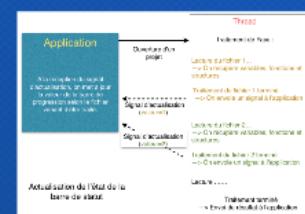
→ Bouton stylisé

Barre de statut & barre de progression

Une autre barre de statut :

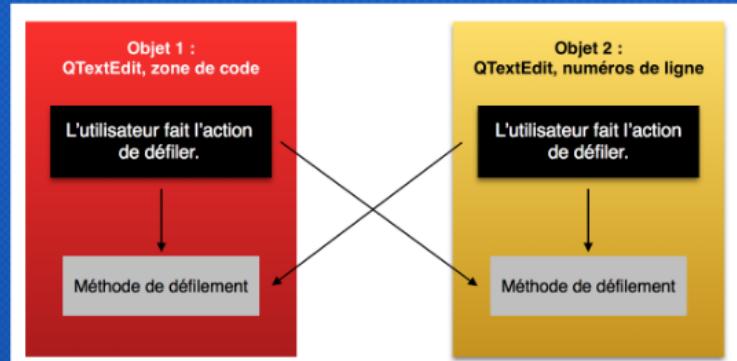


Lors du chargement d'un projet :
affichage d'une barre de progression.



Numérotation des lignes

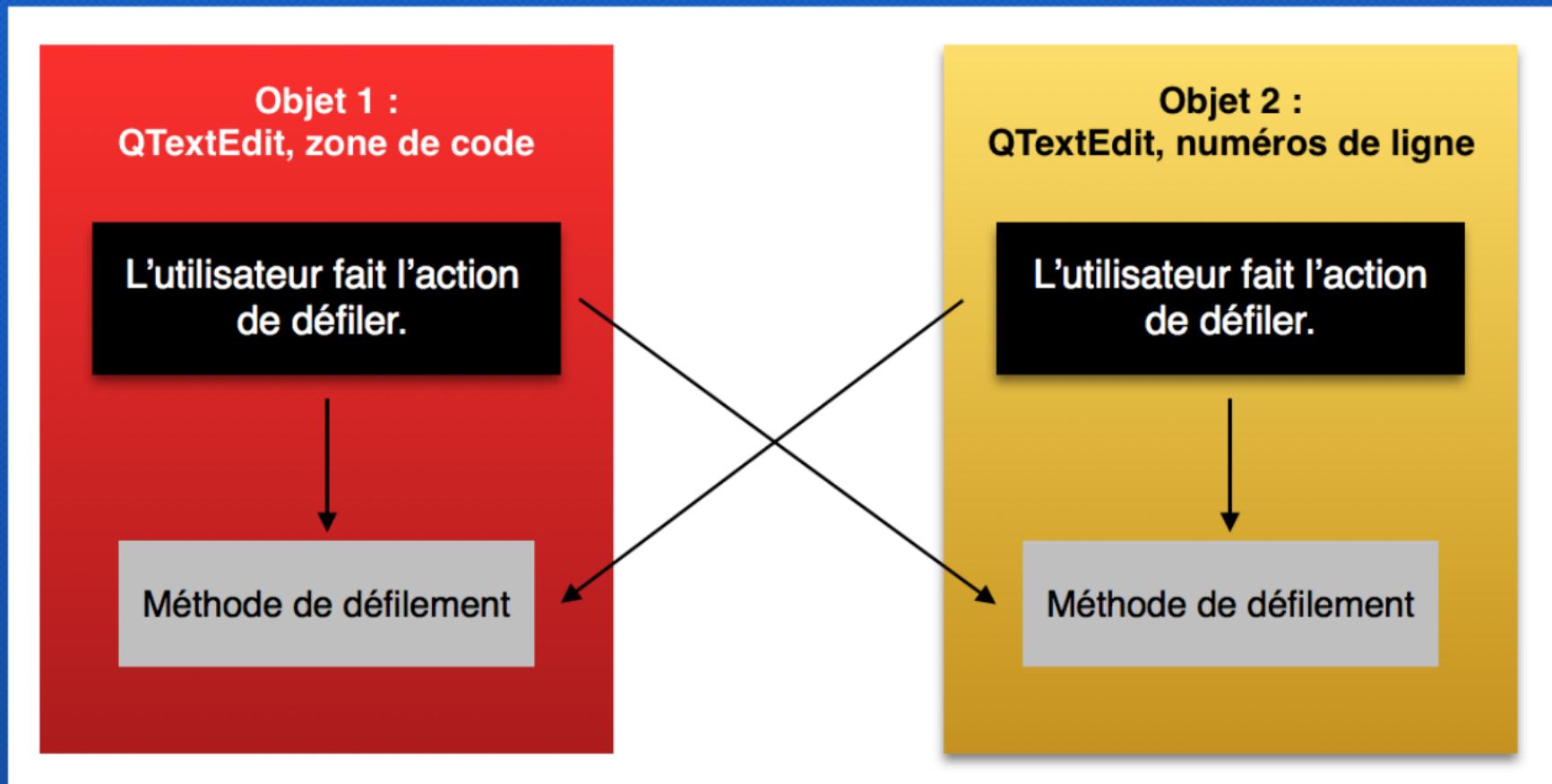
Problématique : Synchroniser le défilement.



Prezi

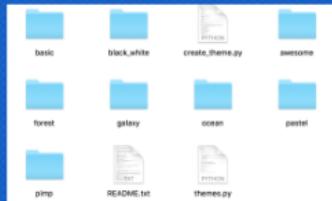
Numérotation des lignes

Problématique : Synchroniser le défilement.

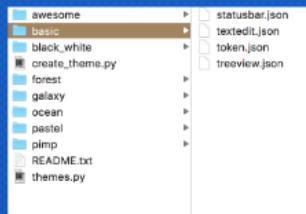


IHM (Interface Homme-Machine)

Thèmes du logiciel



Contenu du répertoire "themes/"



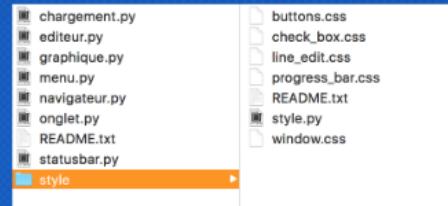
Contenu du thème "basic"

```
textedit.json
```

```
1  {
2      "text-back-color": [31, 31, 31],
3      "text-color": [255, 255, 255],
4      "tab-color": [0, 0, 0],
5      "tab-back-color": [100, 100, 100],
6      "tab-hover-back-color": [31, 31, 31],
7      "tab-hover-color": [255, 255, 255],
8      "tab-hover-border-color": [0, 0, 0]
9 }
```

Contenu du fichier textedit.json

Styles des éléments



Contenu du répertoire "style/"

Cancel

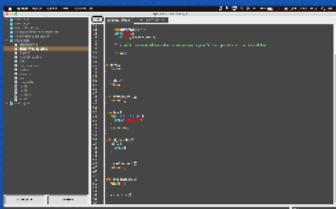
→ Bouton normal

Cancel

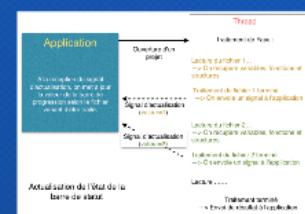
→ Bouton stylisé

Barre de statut & barre de progression

Une autre barre de statut :



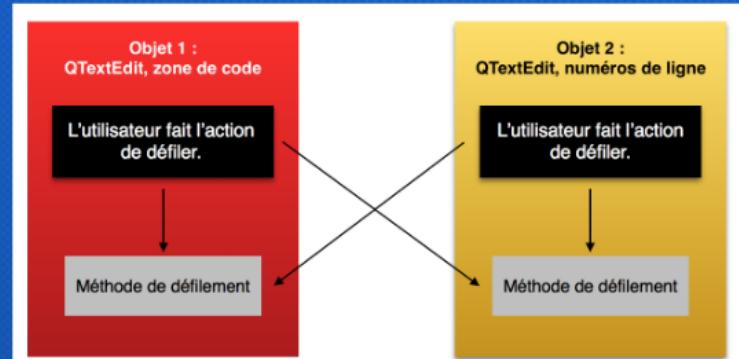
Lors du chargement d'un projet :
affichage d'une barre de progression.



- Nombre total de lignes
- Nombre total de propositions pour l'auto-complétion
- Nombre total d'occurrences trouvées lors d'une recherche

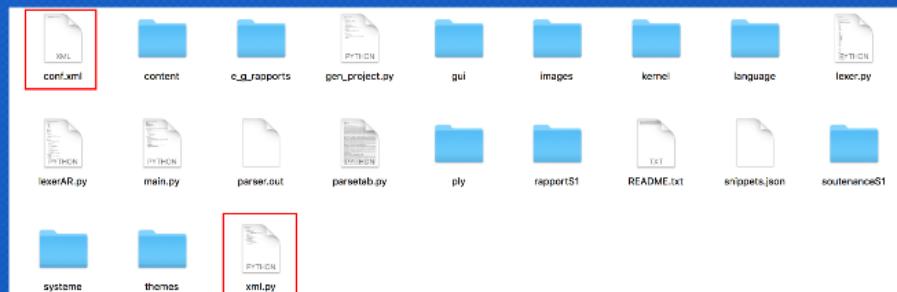
Numérotation des lignes

Problématique : Synchroniser le défilement.

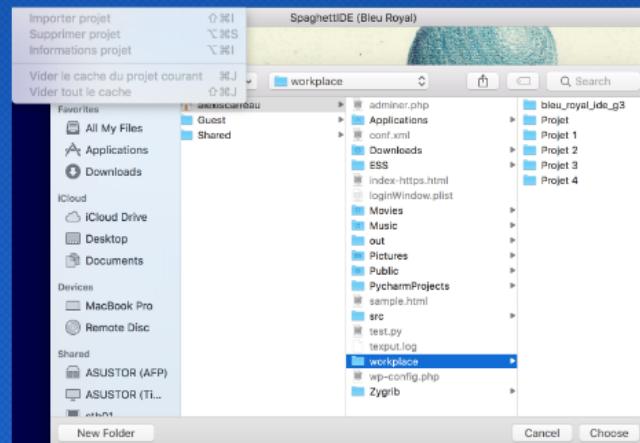


Prezi

Fichiers XML & gestion des projets



Racine du TPA



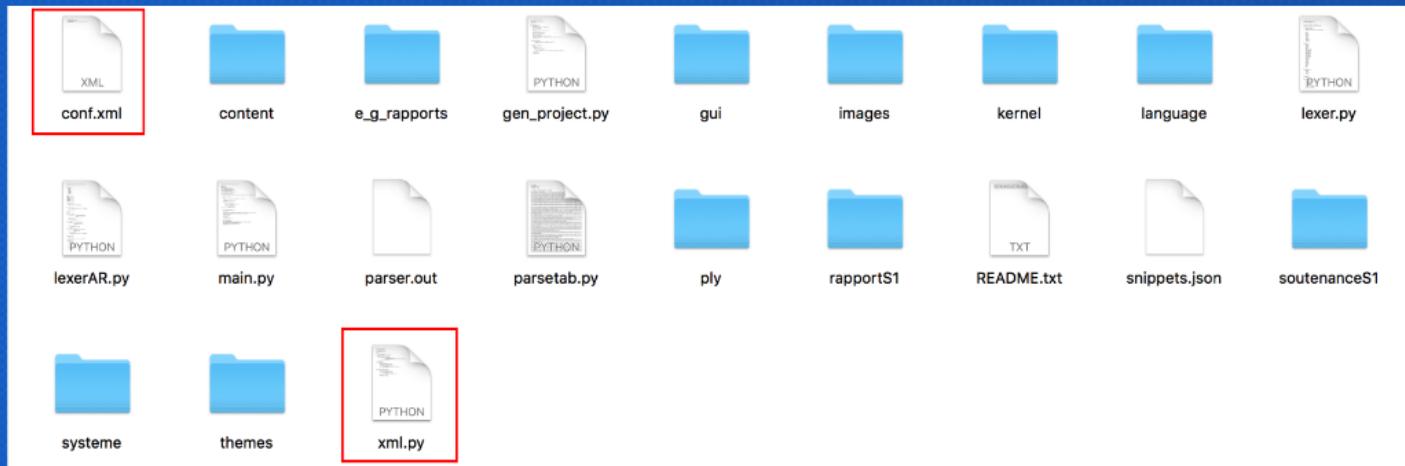
Contenu du répertoire "workplace"

```
<configuration>
  <theme>ocean</theme>
  <assistance_vocale>False</assistance_vocale>
  <loading>False</loading>
  <numerote_lines>True</numerote_lines>
  <language>en</language>
  <current_workplace>/Users/alexiscarreau/workplace/</current_workplace>
</configuration>
```

Contenu du fichier conf.xml

```
<project>
  <name>mon projet</name>
  <creation_date>2017-03-20 12:18:36.648419</creation_date>
  <language>C</language>
  <location>/Users/Sharkiller/workplace/mon projet</location>
  <number_files>13</number_files>
  <compil></compil>
  <compil_json></compil_json>
</project>
```

Informations contenues dans les fichiers .xml de chaque projet

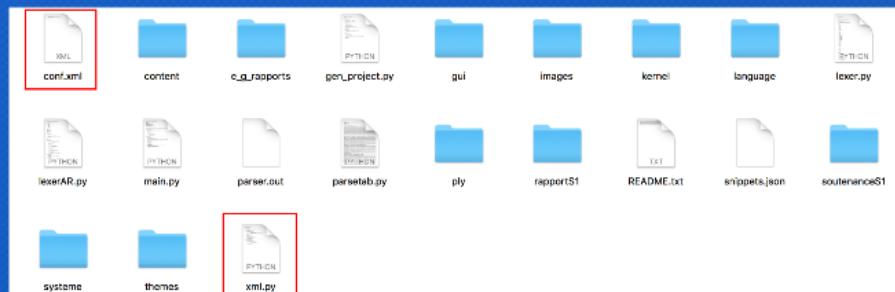


Racine du TPA

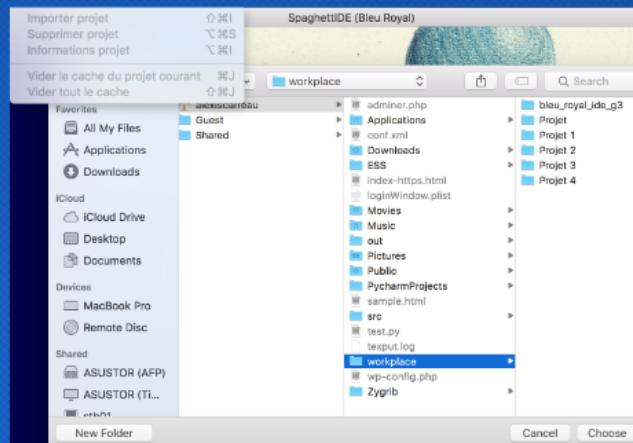
```
<configuration>
    <theme>ocean</theme>
    <assistance_vocale>False</assistance_vocale>
    <loading>False</loading>
    <numerote_lines>True</numerote_lines>
    <language>en</language>
    <current_workplace>/Users/alexiscarreau/workplace/</current_workplace>
</configuration>
```

Contenu du fichier conf.xml

Fichiers XML & gestion des projets



Racine du TPA



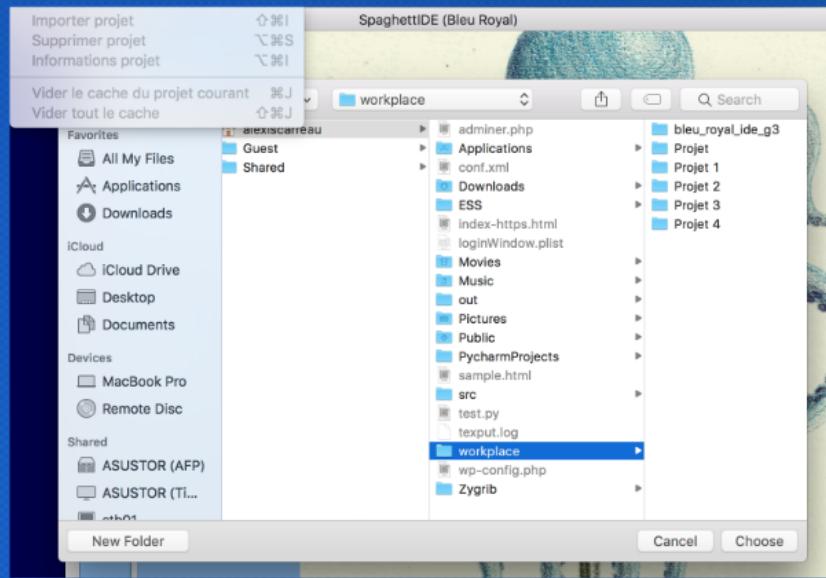
Contenu du répertoire "workplace"

```
<configuration>
  <theme>ocean</theme>
  <assistance_vocale>False</assistance_vocale>
  <loading>False</loading>
  <numerote_lines>True</numerote_lines>
  <language>en</language>
  <current_workplace>/Users/alexiscarreau/workplace/</current_workplace>
</configuration>
```

Contenu du fichier conf.xml

```
<project>
  <name>mon projet</name>
  <creation_date>2017-03-20 12:18:36.648419</creation_date>
  <language>C</language>
  <location>/Users/Sharkiller/workplace/mon projet</location>
  <number_files>13</number_files>
  <compil></compil>
  <compil_json></compil_json>
</project>
```

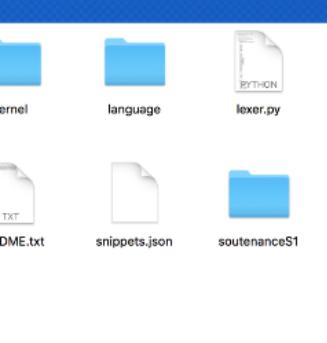
Informations contenues dans les fichiers .xml de chaque projet



Contenu du répertoire "workplace"

```
<project>
  <name>mon projet</name>
  <creation_date>2017-03-20 12:18:36.648419</creation_date>
  <language>C</language>
  <location>/Users/Sharkiller/workplace/mon projet</location>
  <number_files>13</number_files>
  <compil></compil>
  <compil_json></compil_json>
</project>
```

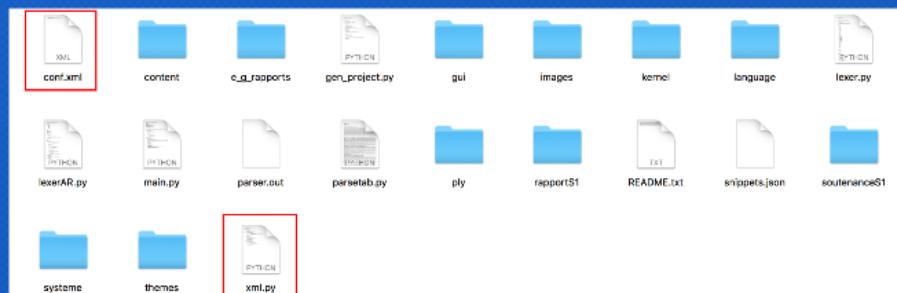
Informations contenues dans les fichiers .xml de chaque projet



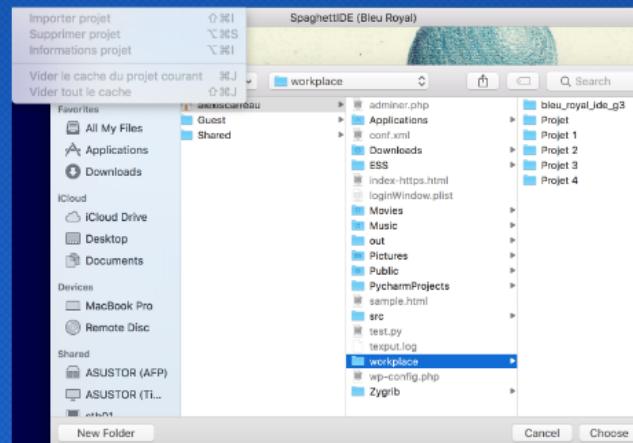
e/</current_workplace>

.xml

Fichiers XML & gestion des projets



Racine du TPA



Contenu du répertoire "workplace"

```
<configuration>
  <theme>ocean</theme>
  <assistance_vocale>False</assistance_vocale>
  <loading>False</loading>
  <numerote_lines>True</numerote_lines>
  <language>en</language>
  <current_workplace>/Users/alexiscarreau/workplace/</current_workplace>
</configuration>
```

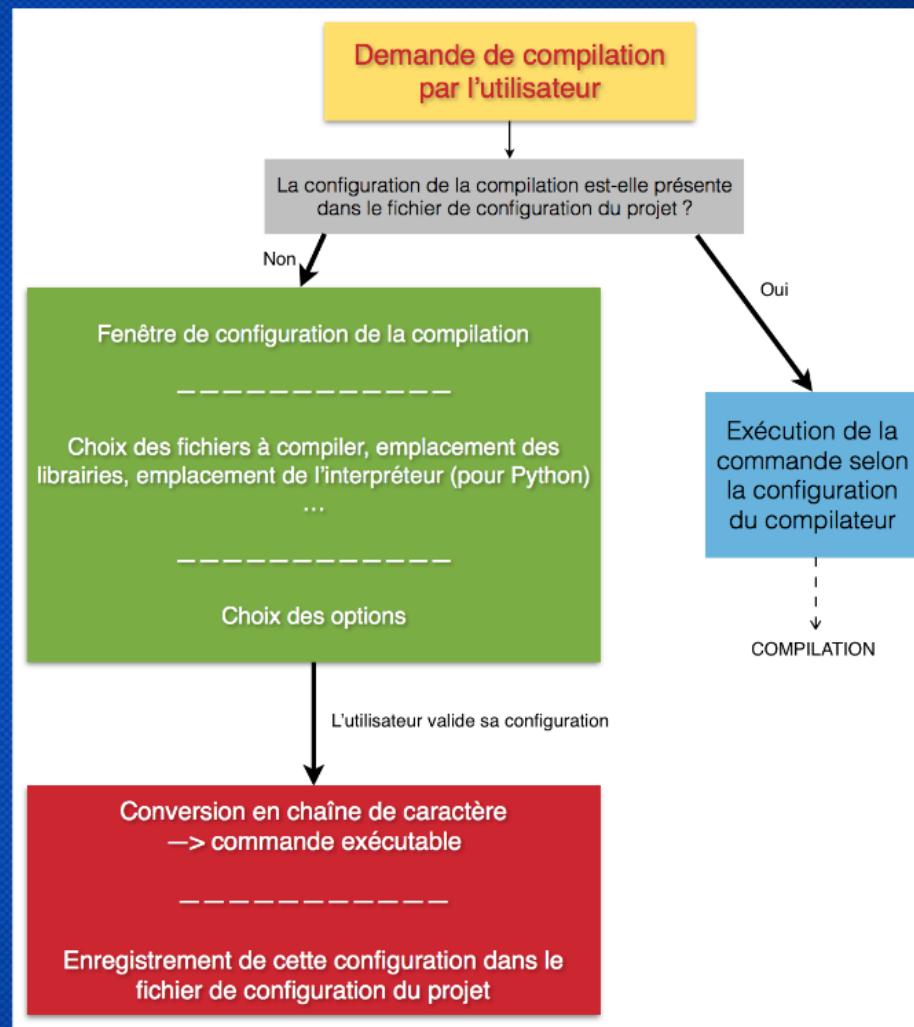
Contenu du fichier conf.xml

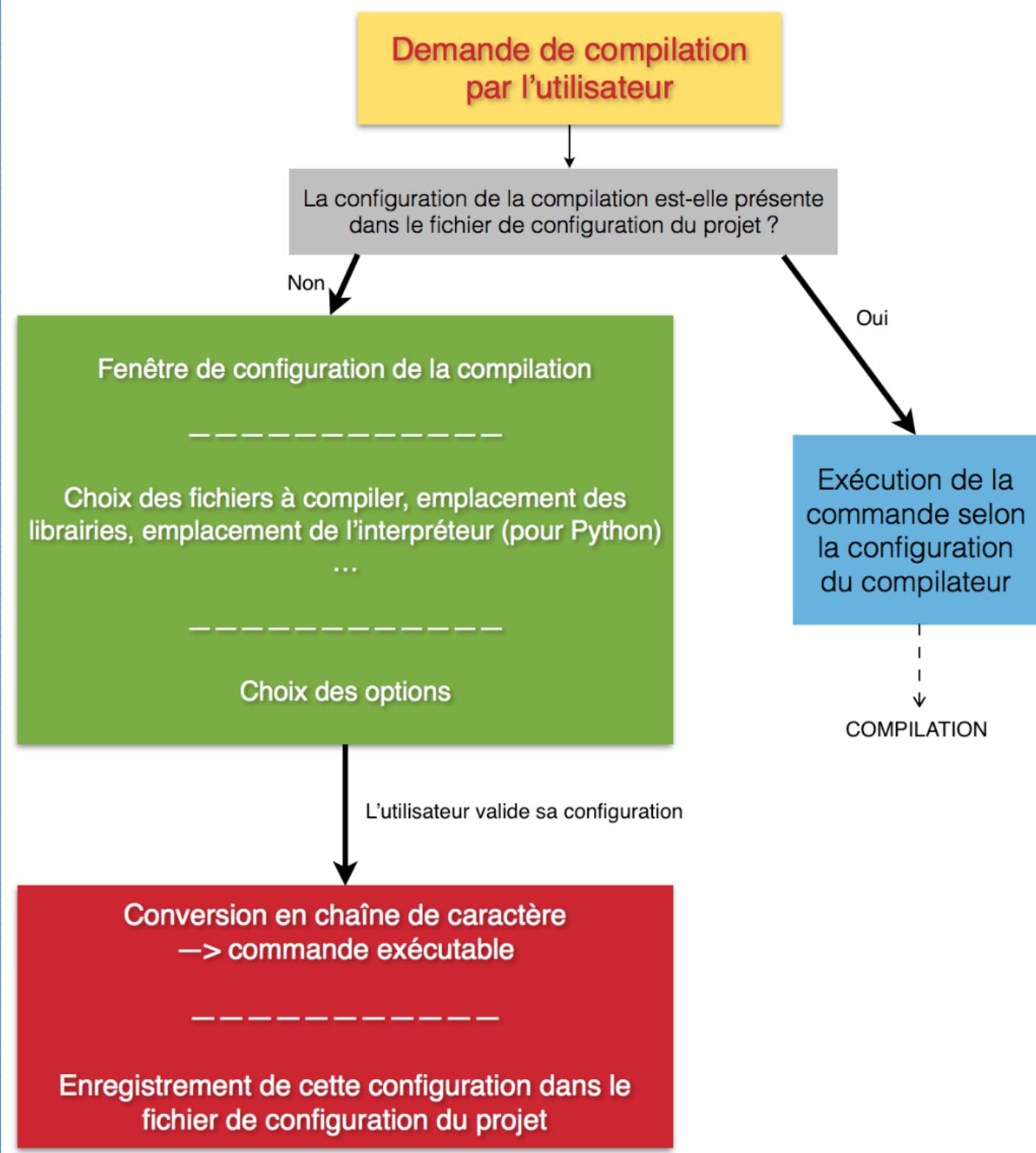
```
<project>
  <name>mon projet</name>
  <creation_date>2017-03-20 12:18:36.648419</creation_date>
  <language>C</language>
  <location>/Users/Sharkiller/workplace/mon projet</location>
  <number_files>13</number_files>
  <compil></compil>
  <compil_json></compil_json>
</project>
```

Informations contenues dans les fichiers .xml de chaque projet

Intégration du compilateur

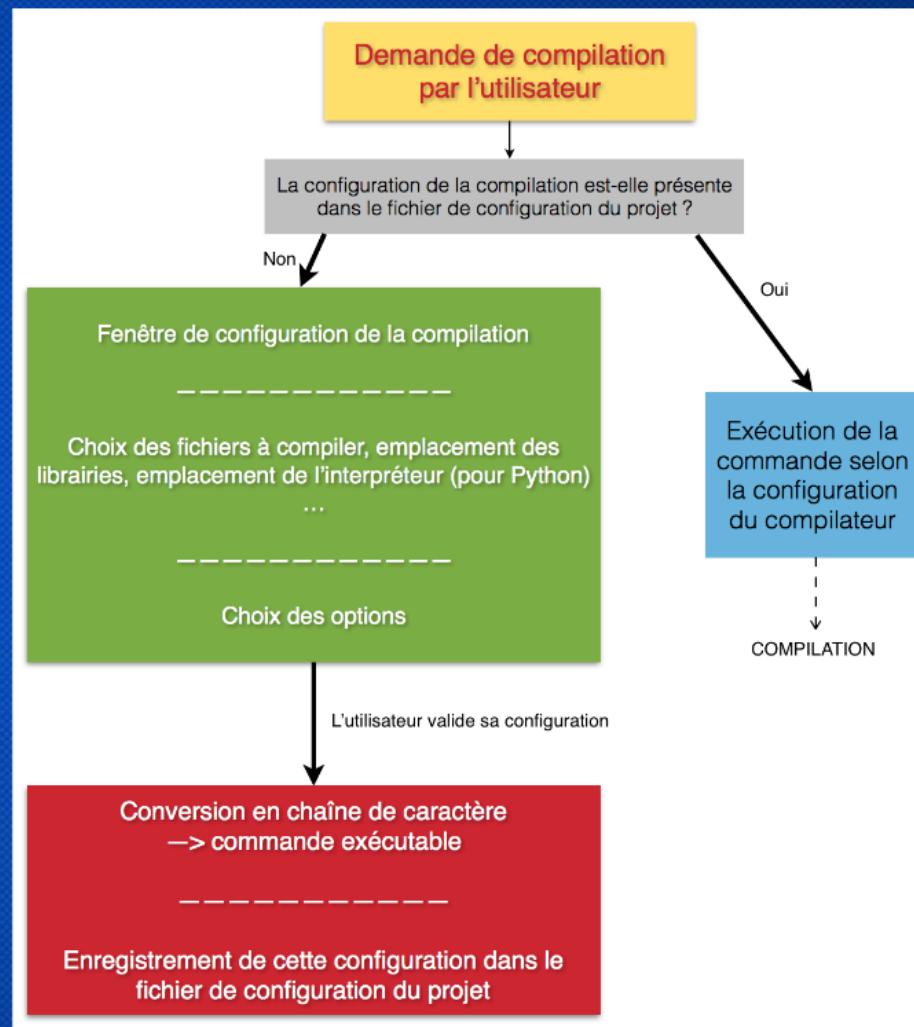
- *Langage C : Utilisation de GCC*
- *Langage Python : Utilisation de l'interpréteur Python*





Intégration du compilateur

- *Langage C : Utilisation de GCC*
- *Langage Python : Utilisation de l'interpréteur Python*



Édition

Recherche, édition du code & Inspecteur d'éléments

Ajout de fonctionnalités typiques d'édition :

- Recherche de mots
- Sélections diverses
- Insertion, indentation
- Commenter une sélection

Ajout d'un inspecteur d'éléments pour lister :

- Les variables
- Les fonctions
- Les structures (en C)
- Les classes (en Python)

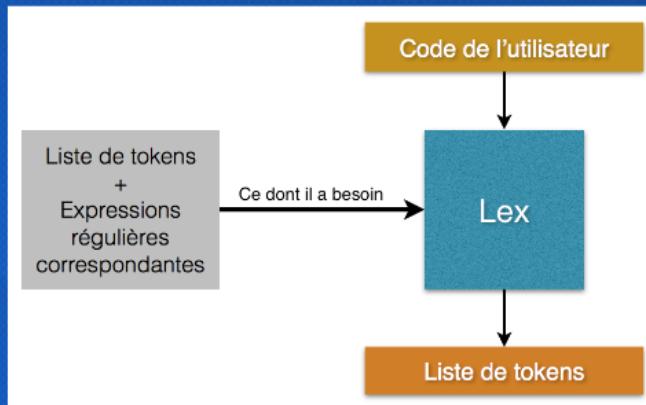
Les grammaires

- Langage C

- Langage Arithmétique

- Langage Python

Fonctionnement de Lex



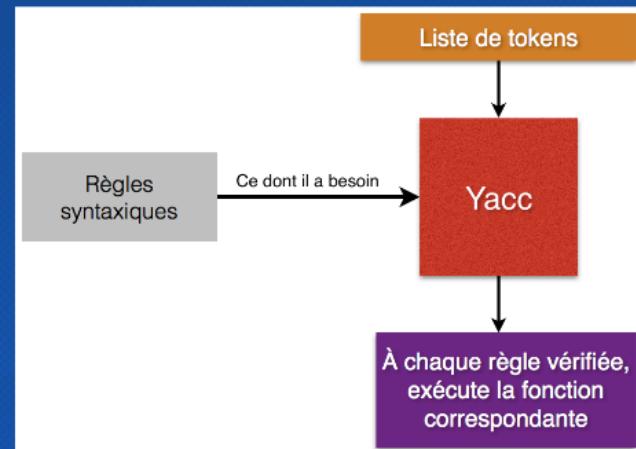
Exemple de liste de tokens

```
tokens = I  
NUMBER  
PLUS  
MINUS  
MULTIPLY  
DIVIDE  
LEFT_PAREN  
RIGHT_PAREN  
ASSIGN  
EQUAL  
NOT_EQ  
NULL_ASSIGN
```

Exemple d'expression régulières

```
I: [0-9]+([.][0-9]+)?|[.][0-9]+  
NUMBER: [-+]?[0-9]+([.][0-9]+)?|[.][0-9]+  
PLUS: '+'  
MINUS: '-'  
MULTIPLY: '*'  
DIVIDE: '/'  
LEFT_PAREN: '('  
RIGHT_PAREN: ')'  
ASSIGN: '='  
EQUAL: '=='  
NOT_EQ: '!='  
NULL_ASSIGN: ':='
```

Fonctionnement de Yacc

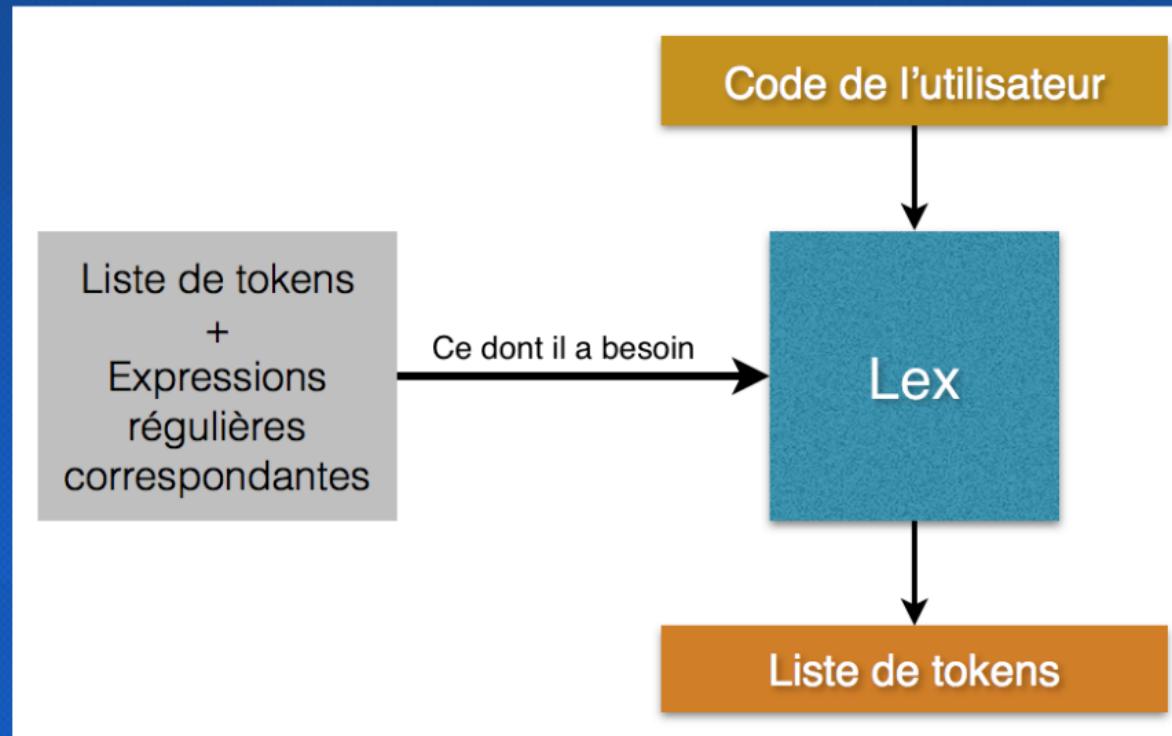


Exemples de règles syntaxiques pour Yacc

```
%token NUMBER ID  
%left '+' '-'  
%left '*' '/'  
  
%%  
expr : term '+' term {plus($1,$3)}  
| term '-' term {minus($1,$3)}  
| term '*' term {multi($1,$3)}  
| term '/' term {divide($1,$3)}  
| ID '=' term {assign($1,$3)}  
| ID {id($1)}  
  
term : NUMBER  
| ID  
| '(' expr ')' {par($2)}
```

Exemples de règles syntaxiques pour Yacc

Fonctionnement de Lex



```
tokens = [
    "IDENTIFIER",
    "CONSTANT",
    "STRING_LITERAL",
    "ELLIPSIS",
    "RIGHT_ASSIGN",
    "LEFT_ASSIGN",
    "ADD_ASSIGN",
    "SUB_ASSIGN",
    "MUL_ASSIGN",
    "DIV_ASSIGN"]
```

Exemple de liste de tokens

```
t_STRING_LITERAL = r"([A-Za-z_]\w*(\.\w*)|\\"")|([A-Za-z_]\w*(\.\w*\.\w*)|\\"")"
t_ELLIPSIS = r"\.\.\."
t_ADD_ASSIGN = r"\+="
t_SUB_ASSIGN = r"\-="
t_MUL_ASSIGN = r"\*="
t_DIV_ASSIGN = r"\/=
```

Exemple d'expression régulières

```
tokens = [
    "IDENTIFIER",
    "CONSTANT",
    "STRING_LITERAL",
    "ELLIPSIS",
    "RIGHT_ASSIGN",
    "LEFT_ASSIGN",
    "ADD_ASSIGN",
    "SUB_ASSIGN",
    "MUL_ASSIGN",
    "DTV_ASSIGN"]
```

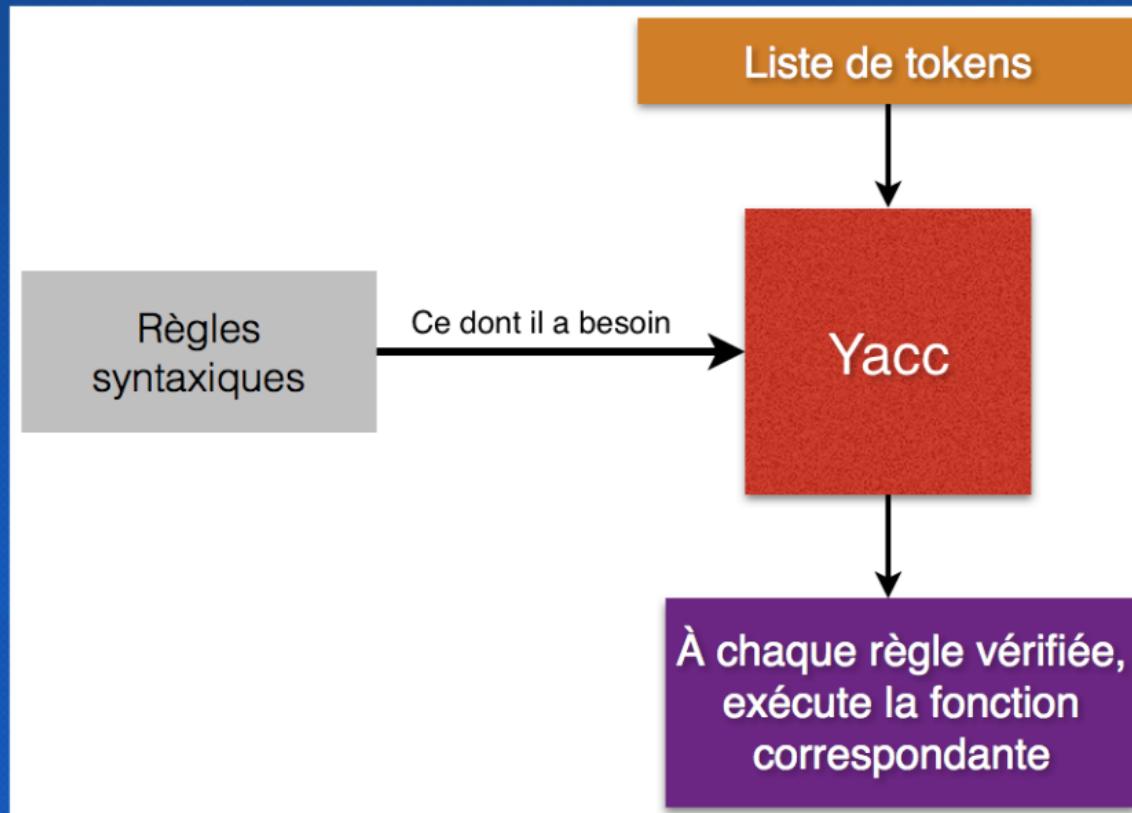
Exemple de liste de tokens

```
t_STRING_LITERAL = r"([A-Za-z_]?\"(\. | [\^\"])*)\" | ([A-Za-z_]?'(\. | [\^\'])*)'"
t_ELLIPSIS = r"\.\.\."
t_ADD_ASSIGN = r"\+="
t_SUB_ASSIGN = r"\-="
t_MUL_ASSIGN = r"\*="
+ DTV_ASSIGN = r"\/-"
```

Exemple d'expression régulières



Fonctionnement de Yacc



```
def p_multiplicative_expression(p):
    '''multiplicative_expression : cast_expression
                               | multiplicative_expression TIMES cast_expression
                               | multiplicative_expression DIVIDE cast_expression
                               | multiplicative_expression MOD cast_expression'''
```



```
def p_cast_expression(p):
    '''cast_expression : unary_expression
                      | L_BRACKET type_name R_BRACKET cast_expression'''
```



```
def p_unary_expression(p):
    '''unary_expression : postfix_expression
                        | INC_OP unary_expression
                        | DEC_OP unary_expression
                        | unary_operator cast_expression
                        | SIZEOF unary_expression
                        | SIZEOF L_BRACKET type_name R_BRACKET'''
```

Exemples de règles syntaxiques pour Yacc

```
def p_multiplicative_expression(p):
    '''multiplicative_expression : cast_expression
                                | multiplicative_expression TIMES cast_expression
                                | multiplicative_expression DIVIDE cast_expression
                                | multiplicative_expression MOD cast_expression'''
```

```
def p_cast_expression(p):
    '''cast_expression : unary_expression
                        | L_BRACKET type_name R_BRACKET cast_expression'''
```

```
def p_unary_expression(p):
    '''unary_expression : postfix_expression
                        | INC_OP unary_expression
                        | DEC_OP unary_expression
                        | unary_operator cast_expression
                        | SIZEOF unary_expression
                        | SIZEOF L_BRACKET type_name R_BRACKET'''
```

Exemples de règles syntaxiques pour Yacc

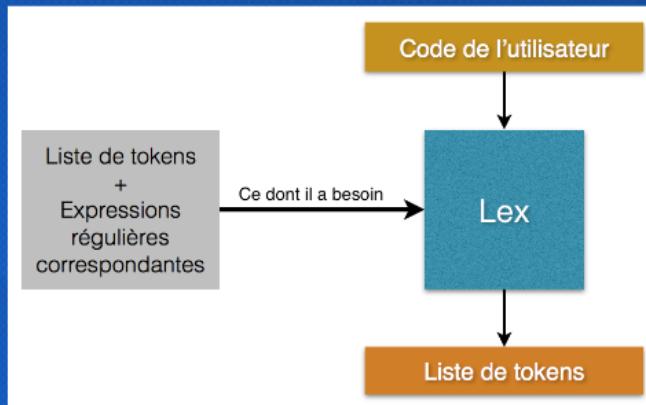
Les grammaires

- Langage C

- Langage Arithmétique

- Langage Python

Fonctionnement de Lex



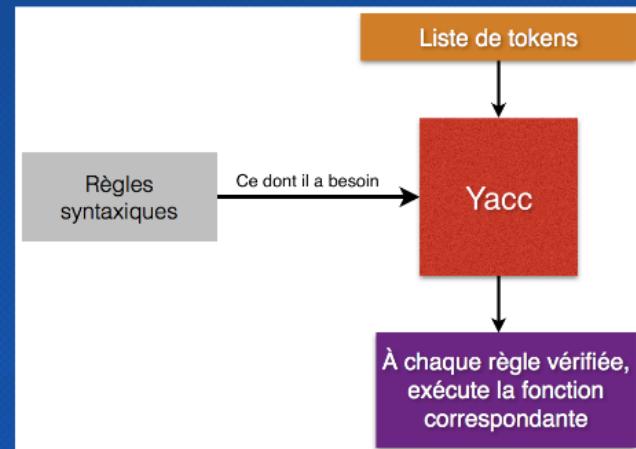
Exemple de liste de tokens

```
tokens = I  
NUMBER  
PLUS  
MINUS  
MULTIPLY  
DIVIDE  
LEFT_PAREN  
RIGHT_PAREN  
ASSIGN  
EQUAL  
NOT_EQ  
NULL_ASSIGN
```

Exemple d'expression régulières

```
I: [0-9]+([.][0-9]+)?|[.][0-9]+  
NUMBER: [-+]?[0-9]+([.][0-9]+)?|[.][0-9]+  
PLUS: '+'  
MINUS: '-'  
MULTIPLY: '*'  
DIVIDE: '/'  
LEFT_PAREN: '('  
RIGHT_PAREN: ')'  
ASSIGN: '='  
EQUAL: '=='  
NOT_EQ: '!='  
NULL_ASSIGN: ':='
```

Fonctionnement de Yacc



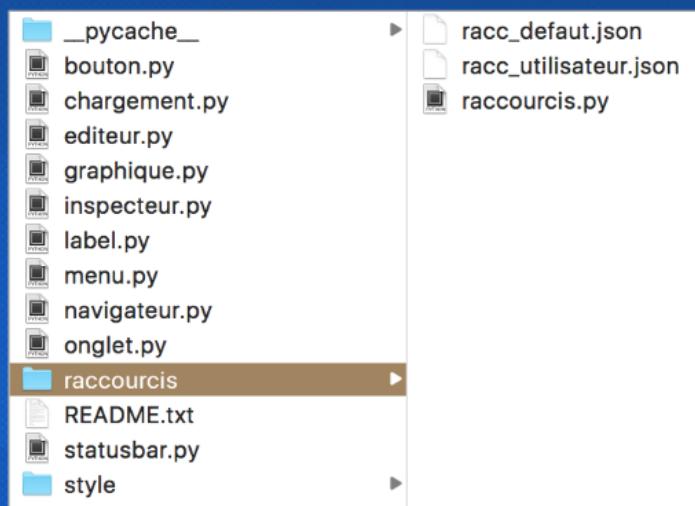
Exemples de règles syntaxiques pour Yacc

```
%token NUMBER ID  
%left '+' '-'  
%left '*' '/'  
  
%%  
expr : term '+' term {plus($1,$3)}  
     | term '-' term {minus($1,$3)}  
     | term '*' term {multi($1,$3)}  
     | term '/' term {divide($1,$3)}  
     | ID '=' term {assign($1,$3)}  
     | ID;  
  
term : NUMBER;  
     | ID;  
     | '(' expr ')';  
  
plus : term '+' term {add($1,$3)};  
minus : term '-' term {sub($1,$3)};  
multi : term '*' term {mult($1,$3)};  
divide : term '/' term {div($1,$3)};  
assign : ID '=' term {set($1,$3)};
```

Exemples de règles syntaxiques pour Yacc

Paramétrage : choix des raccourcis et de la langue

Le choix de ses raccourcis

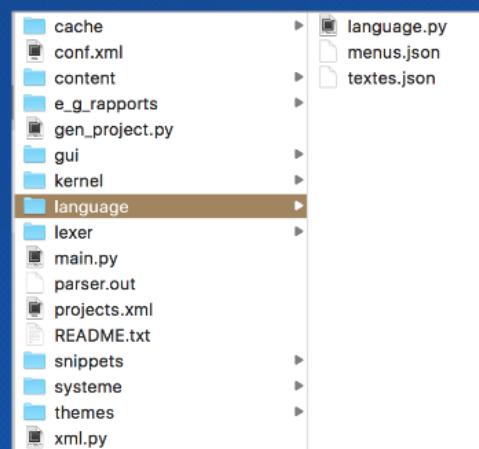


Contenu du répertoire "raccourcis/"

```
{  
    "Fichier": {  
        "Nouveau Fichier": "Ctrl+N",  
        "Nouveau Projet": "Ctrl+M",  
        "Ouvrir": "Ctrl+O",  
        "Sauvegarder": "Ctrl+S",  
    },  
}
```

Contenu du fichier "racc_utilisateur.json"

Le changement de langue



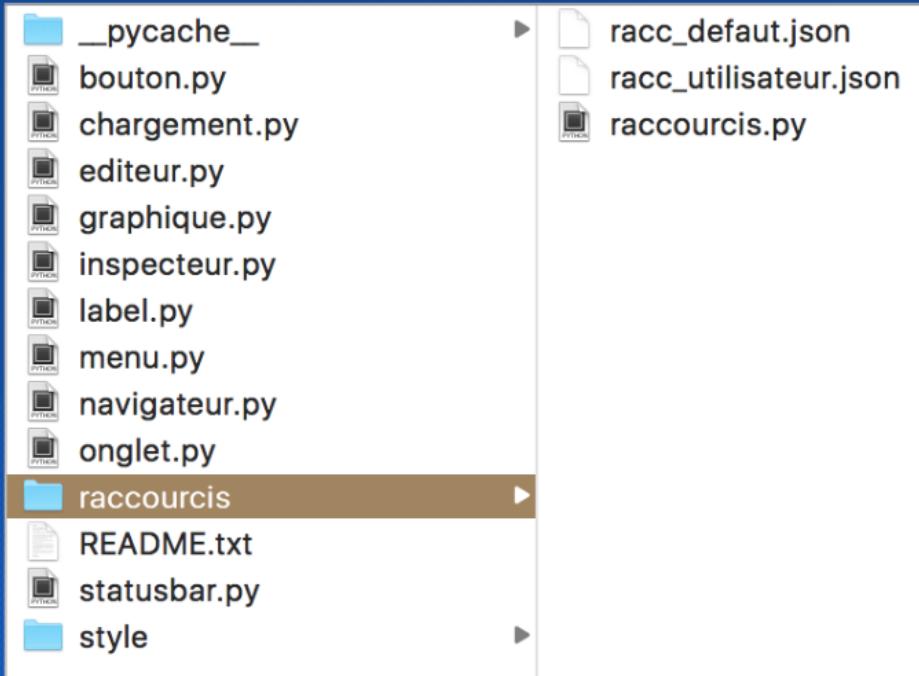
Contenu du répertoire "language/"

```
"status_fic_analys": {  
    "fr": "Le fichier courant a bien été analysé.",  
    "en": "Current file has been successfully processed."  
},  
  
"status_fic_closed": {  
    "fr": "Le fichier sélectionné a bien été fermé.",  
    "en": "Current file has been successfully closed."  
},
```

Contenu du fichier "textes.json"

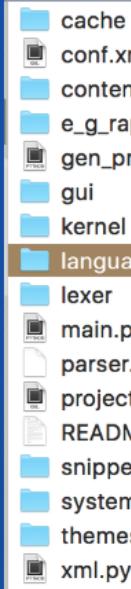


Le choix de ses raccourcis



Contenu du répertoire "raccourcis/"

Contenu du fichier "racc_utilisateur.json"



Contenu de

```
"status_fic_analyst":  
    "fr": "Le fichier d'analyse est actuellement en cours de chargement.",  
    "en": "The analysis file is currently being loaded.",  
},  
  
"status_fic_cleaned":  
    "fr": "Le fichier a été nettoyé et est maintenant prêt à être analysé.",  
    "en": "The file has been cleaned and is now ready to be analyzed.",  
},
```

Conte

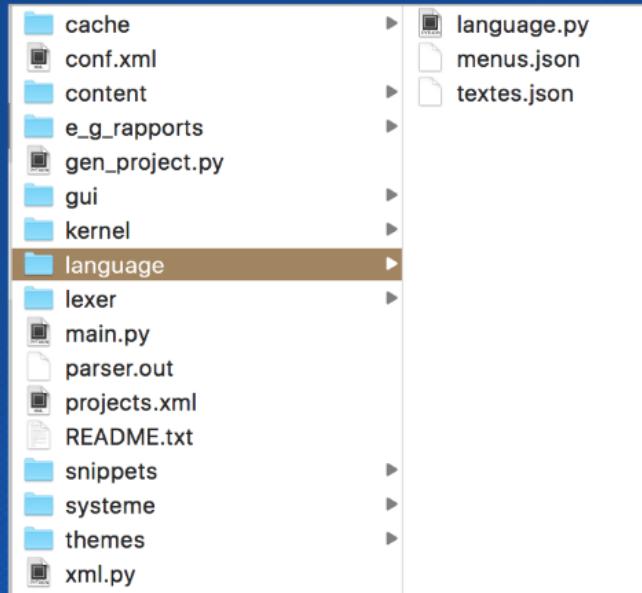
accourcis

_defaut.json
_utilisateur.json
courcis.py

accourcis/"

trl+N",
rl+M",
S",

Le changement de langue



Contenu du répertoire "language/"

```
"status_fic_analys": {  
    "fr": "Le fichier courant a bien été analysé.",  
    "en": "Current file has been successfully processed."  
},  
  
"status_fic_closed": {  
    "fr": "Le fichier sélectionné a bien été fermé.",  
    "en": "Current file has been successfully closed."  
},
```

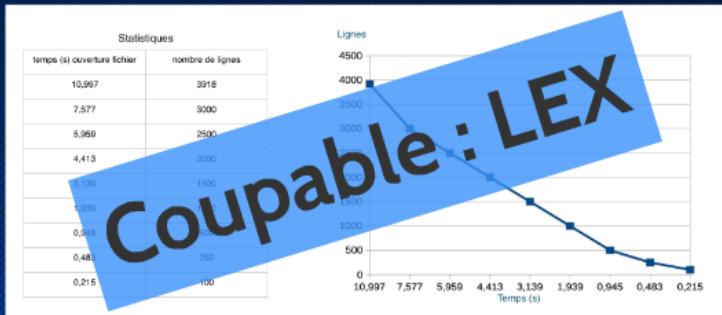
Contenu du fichier "textes.json"

Optimisation & mémoire

Être rapide tout en restant efficace

Cache des fichiers

Gagner du temps lors de l'ouverture des fichiers



Contenu d'un fichier de cache (.json)

Empreinte mémoire

	Sans projet ouvert			Avec un projet ouvert contenant 13 fichiers			Avec un projet ouvert contenant 214 fichiers		
Nombres de fichiers ouverts	0	0	1 (55 lignes)	1 (1000 lignes)	10 (moins de 10 lignes chacun)	0	1 (42 lignes)	5 (75 lignes en moyenne)	5 (entre 150 et 750 lignes)
Consommation RAM (Mo)	72	77	78	95	78	78	78	80	121

- Le nombre de fichiers d'un projet n'importe pas
 - La taille du fichier détermine la RAM à utiliser

Cache des fichiers

Gagner du temps lors de l'ouverture des fichiers



{

```
"\tint monNombre;": [["INT", "int"], ["IDENTIFIER", "monNombre"], ["SEMICOLON", ";"]],  
"\treturn 2*a;": [["RETURN", "return"], ["CONSTANT", "2"], ["TIMES", "*"], ["IDENTIFIER", "a"], ["SEMICOLON", ";"]],  
"\tprintf(\"Le double de %d est : %d \\n\", monNombre, fois2(monNombre), truc);": [["IDENTIFIER", "printf"], ["L_BRACKET"]]
```

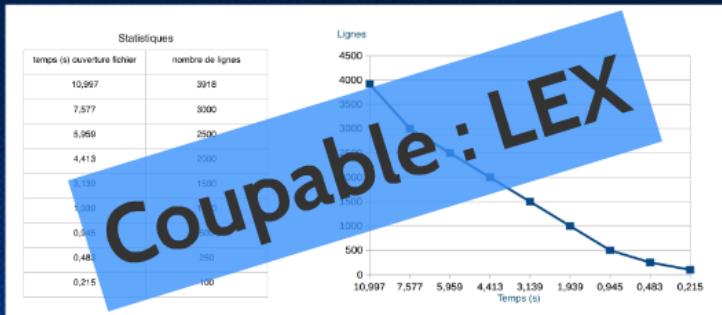
Contenu d'un fichier de cache (.json)

Optimisation & mémoire

Être rapide tout en restant efficace

Cache des fichiers

Gagner du temps lors de l'ouverture des fichiers



```
"\"tint monNombre;": [{"INT": "int"}, [{"IDENTIFIER": "monNombre"}, {"SEMICOLON": ";"}, {"INT": "int"}], {"RETURN_2x": [{"RETURN": "return"}, [{"CONSTANT": "2"}, {"TIMES": "*"}, {"IDENTIFIER": "a"}, {"SEMICOLON": ";"}, {"PRINT": "printf"}], [{"PRINT": "printf"}, {"L_BRACKET": "["}, {"LITERAL": "\"Le double de \", id: \"\\n\", monNombre, fois2(monNombre), true}, {"R_BRACKET": "]"}]}], {"PRINT": "printf"}]
```

Contenu d'un fichier de cache (.json)

Empreinte mémoire

	Sans projet ouvert			Avec un projet ouvert contenant 13 fichiers			Avec un projet ouvert contenant 214 fichiers		
Nombres de fichiers ouverts	0	0	1 (55 lignes)	1 (1000 lignes)	10 (moins de 10 lignes chacun)	0	1 (42 lignes)	5 (75 lignes en moyenne)	5 (entre 150 et 750 lignes)
Consommation RAM (Mo)	72	77	78	95	78	78	78	80	121

- Le nombre de fichiers d'un projet n'importe pas
 - La taille du fichier détermine la RAM à utiliser

Empreinte mémoire

	Sans projet ouvert	Avec un projet ouvert contenant 13 fichiers				Avec un projet ouvert contenant 214 fichiers			
Nombres de fichiers ouverts	0	0	1 (55 lignes)	1 (1000 lignes)	10 (moins de 10 lignes chacun)	0	1 (42 lignes)	5 (75 lignes en moyenne)	5 (entre 150 et 750 lignes)
Consommation RAM (Mo)	72	77	78	95	78	78	78	80	121

- Le nombre de fichiers d'un projet n'importe pas
- La taille du fichier détermine la RAM à utiliser

Projet de TPA

Réalisation d'un IDE en Python

BONUS !!!



Licence informatique
deuxième année

2016-2017

- CARREAU Alexis
- LÉCLUSE Thomas
- MAUGER Emma
- SARRAZIN Théo