

Utilisation VTAM

Principe général d'utilisation :

VTAM (Variantes de Textes Avec Marqueurs) permet de générer des variantes d'un texte balisé avec des marqueurs. Avant de l'utiliser le logiciel il faut correctement baliser le fichier.

Différents marqueurs :

- Marqueurs « blocs » :

- Simples :

```
// if MARQUEUR START DELETE
Contenu du bloc
// MARQUEUR STOP DELETE
```

- Else /Elif :

```
// if MARQUEUR_1 START DELETE
Contenu du bloc if
// elif MARQUEUR_2 START DELETE
Contenu du bloc elif
// else START DELETE
Contenu du bloc else
// MARQUEUR STOP DELETE
```

Remarque : le if en début de bloc et le rappel du nom du marqueur en fin sont optionnels

- Marqueurs « inline » :

```
Début <(MARQUEUR)>contenu <(/ MARQUEUR)>fin
```

Pour supprimer ou garder le contenu des marqueurs il suffira alors d'utiliser xc_filter ou l'interface graphique

Regroupements de marqueurs

Lors du traitement de grands fichiers utilisant de nombreux marqueurs il peut parfois être utile d'utiliser des regroupements de marqueurs que l'on appellera variantes dans la suite. Pour définir une variante il faut l'ajouter dans un fichier variante (*.var).

Exemple de définition d'une variante :

```
@VERSION_PROF :
  @PROPRIETES
  - AIDE
  + DEMO1
  + DEMO2
  + Correction
  @METHODES
  - DEMO_PARTIELLE1
  - DEMO_PARTIELLE2

@METHODES :
  + methode1
  + methode2

@PROPRIETES :
  + PROPRIETE1
  + PROPRIETE2
  + PROPRIETE3
  + PROPRIETE4
```

Le – signifie enlever, + signifie conserver et @ permet d'appeler une autre variante

Utilisation de dossier

Il est possible, si besoin, de traiter en même temps plusieurs fichiers à la fois. Pour cela il suffit de les placer dans un même dossier avec si nécessaire les fichiers de cohérence et de variantes. De plus si les fichiers n'ont pas la même extension, il faut créer un fichier « *.list » où écrire la liste des fichiers à traiter (un nom de fichier avec l'extension par ligne).

VTAM avec xc_filter.py

Le programme xc_filter.py permet d'engendrer des variantes avec des lignes de commandes.

Le modèle de la ligne de commande est le suivant :

```
./xcfilter.py [-h] [-m CHECK_FILE.marker] [-v VARIANT_FILE.var] [-f] [-i] [-dir]  
[TEMPLATE_FILE] [OUTPUT_FILE] +|- [MARKER_1] |@[VARIANT_1] ... +|- [MARKER_N] |@[VARIANT_N]
```

Différentes options

- -h : affiche l'aide à cette commande
- -m : permet de préciser le fichier de cohérence
- -v : permet de préciser le fichier des variantes
- -f : si ajouté le programme ignorera la vérification de cohérence
- -i : si ajouté le programme ignorera la déduction des marqueurs
- -dir : à ajouter pour traiter des dossiers à la place
- TEMPLATE_FILE : fichier source
- OUTPUT_FILE : fichier sortie

Exemples d'utilisation

Pour clarifier le fonctionnement voici plusieurs exemples fait avec les fichiers compris dans Cours_modeles :

Choix de marqueurs:

Par exemple pour produire le résultat ci-dessous avec modele_td1.txt il faut saisir :

```
python .\vtam\xcfilter.py -m .\legacy_examples\Cours_modeles\cours.marker .\legacy_examples  
\Cours_modeles\modele_td1.txt .\legacy_examples\td_corrige.txt +methode2 -AIDE +Correction  
+PROPRIETE2 -DEMO2
```

Consigne exercice

Correction via méthode 1

Remarque quant au manque d'efficacité de la méthode 1

Correction via méthode 2

Exercice 2:

Enoncer et démontrer la propriété 2 du cours

Démonstration de la propriété 2

Utilisation de variantes

On peut obtenir exactement le même résultat que précédemment en utilisant les variantes :

```
python .\vtam\xcfilter.py -m .\legacy_examples\Cours_modeles\cours.marker -v .\legacy_examples\Cours_modeles\cours.var .\legacy_examples\Cours_modeles\modele_td1.txt .\legacy_examples\td_corrige.txt -DEMO2 '@VERSION_PROF'
```

Utilisation d'un dossier

On peut aussi traiter un dossier complet avec les mêmes marqueurs en faisant :

```
python .\vtam\xcfilter.py -dir .\legacy_examples\Cours_modeles .\legacy_examples\Cours\ -DEMO2 '@VERSION_PROF'
```

On obtient alors aussi dans le dossier de sortie les autres fichiers transformés et un fichier result.txt qui contient la liste des marqueurs appliqués :

```
MARQUEURS UTILISES
-DEMO2
+PROPRIETE1
+PROPRIETE2
+PROPRIETE3
+PROPRIETE4
-AIDE
+DEMO1
+DEMO3
+DEMO4
+Correction
+methode1
+methode2
-DEMO_PARTIELLE1
-DEMO_PARTIELLE2
-DEMO_PARTIELLE3
-DEMO_PARTIELLE4
```

Erreur de cohérence :

```
python .\vtam\xcfilter.py -m .\legacy_examples\Cours_modeles\cours.marker .\legacy_examples\Cours_modeles\modele_cours.txt .\legacy_examples\cour_simple.txt -methode1 +methode2
```

Cette entrée produira l'erreur suivante :

```
Verifying that the set of markers is consistent with .\legacy_examples\Cours_modeles\cours.marker

- OK : The .\legacy_examples\Cours_modeles\cours.marker file is satisfiable by itself.
- ERROR : The .\legacy_examples\Cours_modeles\cours.marker file is NOT satisfiable (UNSAT) with the given arguments.

Aborting ...
```

Cette erreur est levée car la présence methode2 est censée impliquer celle de methode1 d'après cours.marker. Pour éviter cette erreur saisissez une entrée en accord avec le fichier de cohérence ou forcez la génération avec -f

VTAM avec une interface graphique

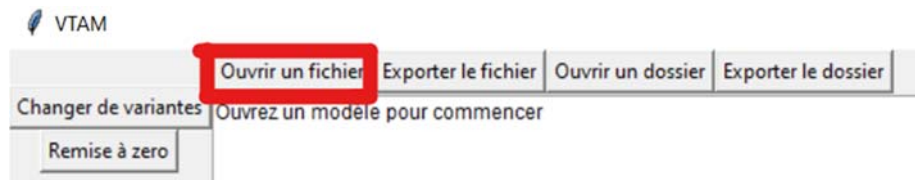
Il est aussi possible d'utiliser l'interface graphique pour générer les variantes.

Comment utiliser l'interface graphique ?

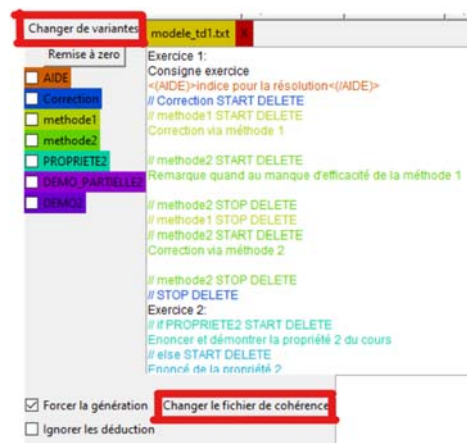
Ouvrir un fichier

1 – Lancer l'interface graphique en exécutant interfaceGraphique.py depuis n'importe quel terminal

2 – Cliquer sur « Ouvrir un fichier »



3 – Choisir le fichier dans la fenêtre qui vient de s'ouvrir



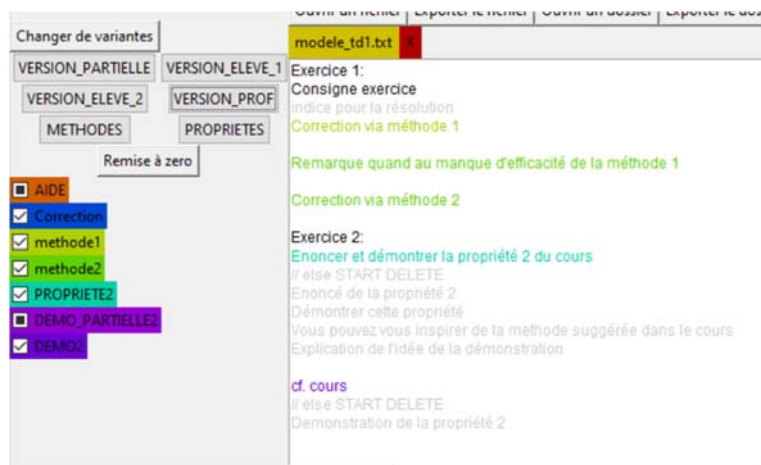
A ce moment-là le fichier doit apparaître dans cadre blanc central et les marqueurs de ce fichier doivent apparaître sur la gauche de la fenêtre

On peut alors choisir si nécessaire les fichiers de variantes et de cohérence en utilisant les boutons entourés ci-contre.

Pour forcer ou non la génération et pour prendre en compte les déductions il suffit de cocher ou non les cases du coin inférieur gauche de la fenêtre.

Engendrer une version

1 – Choisir les marqueurs et les variantes



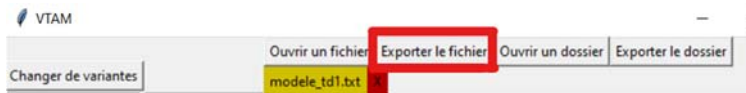
Une case vide équivaut à un marqueur non positionné et une case noire un marqueur supprimé.

Pour remettre un marqueur à l'état non positionné il faut appuyer sur « remise à zéro », par contre cela remettra tous les marqueurs à zéro.

Quand on sélectionne une variante on ne modifie pas les marqueurs déjà cochés.

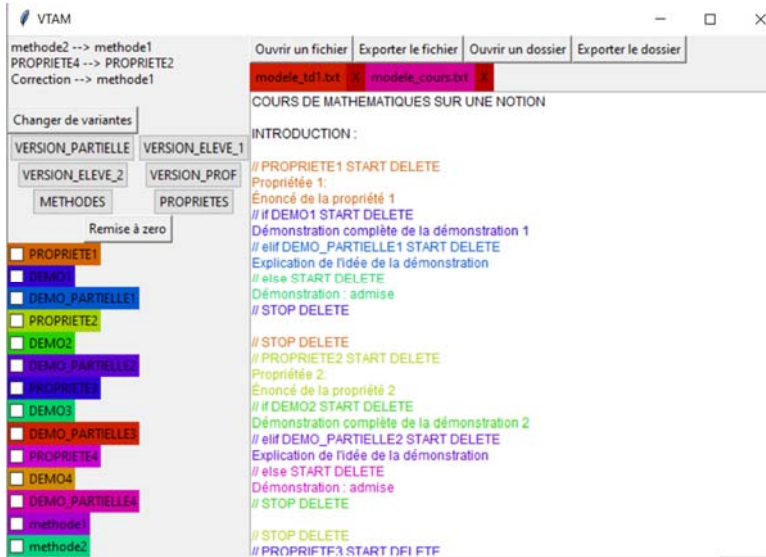
Le texte grisé n'apparaîtra pas dans le résultat il est juste présent pour savoir ce qui a été enlevé.

2 – Exporter la version générée en cliquant sur « Exporter le fichier »



Utiliser un dossier

Suivre les mêmes étapes que pour un fichier mais utiliser « ouvrir un dossier » et « exporter le dossier ».



Il suffit alors de cliquer sur le nom d'un fichier pour l'afficher