



CONCEPTS ET MISE EN ŒUVRE

teaching sciences
for innovation

Package L^AT_EX pour l'édition de documents pédagogiques

v1.1

Emmanuel PINAULT-BIGEARD Version 1.1
e.pinault-bigeard@upsti.fr 2019/07/16

Table des matières

1	Préambule	3
2	Concepts	3
2.1	Dissociation fond/forme	3
2.2	Préambules de fichiers tex simplifiés	3
2.3	Approche modulaire	4
2.4	Personnalisation du package	4
3	Changelog	4
4	Utilisation	4
4.1	Appel du package	4
4.2	Options	4
5	Configuration et personnalisation	5
5.1	Variantes	5
5.2	Couleurs	5
5.3	Hooks	6
6	Gestion du mode prof/élève	6
6.1	Principe	6
6.2	Syntaxe des commandes	6
7	Styles et Templates	7
7.1	Terminologie	7
7.2	Style par défaut	7



7.3	Style basic	10
7.4	Style vide	12
7.5	Corrigés UPSTI	12
7.6	Modèle perso	12
8	Script de compilation rapide (pour Windows)	13
8.1	Présentation	13
8.2	Installation	13
8.3	Configuration du script	13
8.4	Organisation des dossiers	13
8.5	Upload sur un serveur FTP	14
8.6	Convention de nommage	14
9	Migration depuis le package EPB_Cours.sty	14

1 Préambule

Le package `UPSTI_Document.sty` est le package « maître » d'un ensemble de packages personnalisables et intégrables destinés à faciliter l'édition de documents pédagogiques en \LaTeX :

- `UPSTI_SI.sty` : ensemble de macros et commandes facilitant l'édition de documents en S2I ;
- `UPSTI_Typographie.sty` : outils de mise en forme du texte ;
- `UPSTI_Pedagogique.sty` : gestion des diagrammes et compétences pour les S2I en CPGE.

L'objectif est de pouvoir produire des documents basés sur l'utilisation de logiciels **libres, maintenables, portables, pérennes** et **facilement mutualisables** entre collègues, tout en permettant à chacun de conserver ses propres habitudes et environnements de travail.



Remarque

Ce document utilise lui-même le package `UPSTI_Document.sty`

2 Concepts

2.1 Dissociation fond/forme

L'idée est de penser les documents comme on pourrait penser l'articulation HTML/CSS pour des pages Web. Le langage à base de balise assigne une fonction au divers éléments du document, les styles CSS se chargent de la mise en forme. On pourrait parler « d'approche fonctionnelle » du document. Cette approche est aussi similaire à la conception d'un document Word à base de feuilles de styles détaillées.

Ainsi, lors d'un échange de documents entre collègues par exemple, on peut très rapidement (voire de manière immédiate et totalement transparente) adapter la présentation du document à ses propres habitudes.

\LaTeX est déjà un outil très puissant de ce point de vue là. Les packages présentés ici permettent de pousser un peu plus loin cette approche dans le cadre de l'enseignement des Sciences de l'Ingénieur.

2.2 Préambules de fichiers tex simplifiés

L'objectif de cette solution est aussi de simplifier l'édition des préambules des fichier `tex`. Plutôt que de systématiquement appeler un nombre plus ou moins importants de packages à chaque fois, on va uniquement appeler le package `UPSTI_Document.sty`. On affinera ensuite l'affichage en définissant un certain nombre de commandes (titre, classe, type de document, etc), comme le montre l'exemple ci-dessous :

```
1 | \documentclass[11pt]{article}
2 | \usepackage{UPSTI_Document} % Appel du package principal
3 | \newcommand{\UPSTIidClasse}{2} % PT
4 | \newcommand{\UPSTIidTypeDocument}{1} % Cours
5 | \newcommand{\UPSTItitreEnTete}{Concepts et mise en oeuvre} % Titre principal
6 | \newcommand{\UPSTInumeroVersion}{0.1} % Version
```

2.3 Approche modulaire

Comme il a été dit plus tôt, le package `UPSTI_Document.sty` est le package « maître » d'un ensemble de packages. Il permet la mise en place de mécanismes génériques (gestion de version prof/élève, mise en page, etc.). Ensuite, chacun choisit les fonctionnalités qu'il souhaite utiliser via les options d'appel du package (voir §4.2).

On peut aussi ne pas utiliser le package « maître » et n'utiliser que quelques uns des packages proposés en standalone (`UPSTI_Pedagogique.sty`, `UPSTI_SI.sty`). Dans ce cas, il faut se reporter à la documentation propre à chaque package. Chacun d'entre eux est prévu pour fonctionner seul ou pour s'intégrer à `UPSTI_Document.sty`.

2.4 Personnalisation du package

Ces packages ne sont certainement pas parfaits. Afin de faciliter les mises à jours futures, il est souhaitable de ne jamais modifier directement les fichiers `.sty`. Il a été prévu pour cela un fichier `UPSTI_Custom.sty` qui regroupera toutes les personnalisations et toutes les redéfinitions personnelles de macros (voir §5).

3 Changelog

Version 1.1 : 16/07/2019

- Correction de bugs mineurs (voir fichier source)

Version 1.0 : 17/10/2017

- Mise en ligne de la première version

4 Utilisation

4.1 Appel du package

Le package est appelé en début de document par : `\RequirePackage{UPSTI_Document}`. Il faut ensuite définir un certain nombre de commandes (voir §7) puis enfin exécuter la commande `\UPSTIbuildPage` après l'instruction `\begin{document}`.

Un fichier d'exemple `UPSTI_exemple.tex` est fourni dans le fichier zip du package.

4.2 Options

On peut aussi appeler le package en déclarant quelques options (parmi la liste présentée ci-après), de la manière suivante : `\RequirePackage[option1,option2]{UPSTI_Document}`.

- Options de mise en page :
 - ◊ `vide` : n'utilise aucun des styles prédéfinis (utile si on veut garder la main sur la présentation du document (voir §7.4) ;

- ◇ `basic` : utilise le style « basic » dont la mise en page est plus sobre que le style par défaut (voir §7.3);
- ◇ `corrigeUPSTI` : template destiné à rédiger des corrigés pour le site de l'UPSTI (voir §7.5);
- ◇ `kholle` : template spécifique aux sujets de colle (voir §7.2.3);
- ◇ `QCM` : template adapté aux sujets de QCM utilisant AMC (AutoMultipleChoice). Voir §7.2.2;
- ◇ `modelePerso` : on peut utiliser cette option si on souhaite utiliser un fichier de modèle personnel (voir §7.6);
- ◇ `margesEtroites` : à utiliser pour avoir des documents dont les marges sont plus étroites (peu fiable...).
- Options de chargement de packages :
 - ◇ `noSI` : on ne charge pas le package `UPSTI_SI` (chargé par défaut);
 - ◇ `noTypographie` : on ne charge pas le package `UPSTI_Typographie` (chargé par défaut);
 - ◇ `noPedagogique` : on ne charge pas le package `UPSTI_Pedagogique` (chargé par défaut);
 - ◇ `noCustomPackages` : on ne charge pas les packages créés par [R.Papanicola](#) (`schemabloc`, `rpcinematik`, `bodegraph`) qui sont chargés par défaut;
 - ◇ `noPackages` : on ne charge aucun package.

5 Configuration et personnalisation

Pour faciliter les éventuelles mises à jours du package, il est recommandé de ne pas modifier directement le fichier `UPSTI_Document.sty`.

Pour personnaliser et configurer le package, on va donc utiliser le fichier `UPSTI_Custom.sty`.

5.1 Variantes

On définit dans le fichier `UPSTI_Custom.sty` une présentation par défaut. On doit pour cela y préciser le chemin des images et logos utilisés par tous les documents. De même, on définira un certain nombre de variables tels que le nom du lycée, son adresse, etc...

On pourra aussi y définir plusieurs variantes, correspondant à plusieurs lycées par exemple. Cette possibilité est très intéressante quand on enseigne sur 2 lycées par exemple. Il suffira de changer une ligne pour adapter immédiatement un document à une autre classe ou à un autre lycée (par exemple : `\newcommand{\UPSTIvariante}{2}`).

Le fichier `UPSTI_Custom.sty` distribué dans le zip montre plusieurs configurations à titre d'exemple.

5.2 Couleurs

Ce package utilise un jeu de couleurs prédéfinis, afin de garantir une certaine homogénéité entre les divers documents en terme de présentation. On pourra les redéfinir dans le fichier `UPSTI_Custom.sty`. Par défaut, on utilise les couleurs suivantes :

- `UPSTIcouleurCorrige` (rouge) : Couleur utilisée pour les corrigés
- `UPSTIcustomColor1` (bleu clair) : couleur 1;
- `UPSTIcustomColor2` (rouge) : couleur 2;
- `UPSTIcustomColor3` (mauve) : couleur 3;
- `UPSTIcustomColor4` (vert) : couleur 4;
- `UPSTIcustomColor5` (orange) : couleur 5.

5.3 Hooks

Le fichier `UPSTI_Custom.sty` permet aussi d'intervenir à différents endroits dans la génération des diverses mises en page. Ce serait trop long de tout développer ici mais il faut juste savoir qu'on peut pratiquement tout personnaliser par ce biais (en éditant les commandes `\UPSTIeditCustomVars` et `\UPSTIeditCustomVarsEnd`).

6 Gestion du mode prof/élève

6.1 Principe

Ce package permet de générer plusieurs versions d'un même document, en y faisant notamment apparaître (ou pas) les corrigés. On peut aussi générer des documents « à trous ». On distingue 3 types de documents :

- Version Élève : les corrigés sont masqués. Dans le cas de textes à trous, les zones à compléter sont laissées vides.
- Version Prof : les corrigés sont visibles, affichés dans la couleur définie par `UPSTIcouleurCorrige` (rouge par défaut). Dans le cas de textes à trous, les zones à compléter sont affichées en rouge.
- Version « À publier » : se comporte comme la version élève, sauf que les zones à compléter sont affichées en noir (à utiliser lorsqu'on souhaite distribuer ou publier un cours « à trous »).

Version	Prof	Élève	À publier
<code>UPSTIidVersionDocument</code>	1	2	3
<code>UPSTIcorrection, UPSTIcorrectionEnv</code>	Rouge	Invisible	Invisible
<code>UPSTIaCompleter, UPSTIaCompleterEnv</code>	Rouge	Blanc	Noir
<code>UPSTIeleveOnly, UPSTIeleveOnlyEnv</code>	Invisible	Noir	Noir
<code>UPSTIprofOnly, UPSTIprofOnlyEnv</code>	Noir ou rouge	Invisible	Invisible
<code>UPSTIprofEleve</code>	Selon option	Selon option	Selon option

En règle général, on utilisera les commandes `UPSTIaCompleter` et `UPSTIaCompleterEnv` pour les documents de cours, et les autres pour les TD, colles et autres exercices...

6.2 Syntaxe des commandes

- `\UPSTIcorrection[opt noSpaceAfter]{contenu}` :
 - ◊ `noSpaceAfter` : si on définit cette variable à 1, alors on ne sautera pas de ligne à la fin de la correction (utile dans un tableau par exemple) ;
 - ◊ `contenu` : contenu apparaissant en noir ou rouge, selon la version du document.
- `\UPSTIaCompleter{contenu}` :
 - ◊ `contenu` : contenu invisible, en noir ou rouge, selon la version du document (pour les documents à trous).
- `\UPSTIeleveOnly{contenu}` :
 - ◊ `contenu` : texte qui ne s'affiche qu'en mode élève.
- `\UPSTIprofOnly[opt isRouge]{contenu}` :
 - ◊ `contenu` : texte qui ne s'affiche qu'en mode prof ;

- ◇ `isRouge` : si on définit cette variable à 1, le texte sera affiché en rouge en mode prof (dans ce cas, le comportement est le même que `UPSTIcorrection`, à ceci près que dans ce cas, la taille du texte n'est pas réduite).
- `\UPSTIprofEleve[opt isRouge]{contenuProf}{contenuEleve}` :
 - ◇ `contenuProf` : texte qui ne s'affiche qu'en mode prof ;
 - ◇ `contenuEleve` : texte qui ne s'affiche qu'en mode élève ;
 - ◇ `isRouge` : si on définit cette variable à 1, le texte sera affiché en rouge en mode prof.

7 Styles et Templates

7.1 Terminologie

On utilisera par la suite la terminologie suivante :

- **Template** : un template est une mise en page spécifique à un type de document. On aura notamment un template pour les cours, TD et autres documents pédagogiques, un template spécifique pour les colles, un pour les QCM, etc.
- **Style** : un style regroupe plusieurs templates. Il y a à ce jour 3 styles : le style par défaut (utilisé pour ce document par exemple), le style « basic », un peu plus sobre, et le style « vide ». Il y a aussi le style « corrigeUPSTI » qui est spécifique et ne compte qu'un seul template.
- **Modèle** : un modèle regroupe plusieurs style. Ainsi, ce package est fourni avec un modèle standard (composé de 3 styles). Il est cependant possible de ne pas utiliser ce modèle standard mais un modèle personnel (voir §7.6).

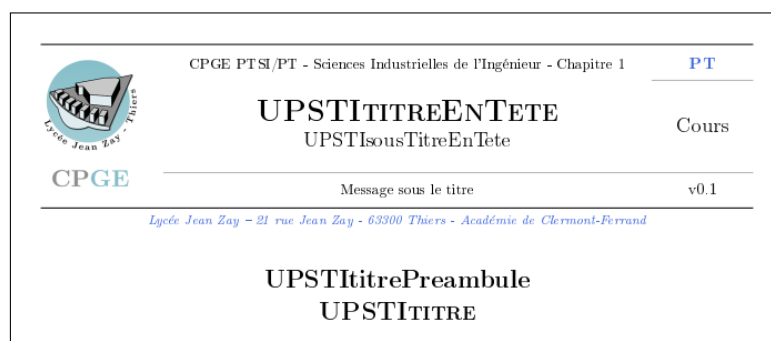
7.2 Style par défaut

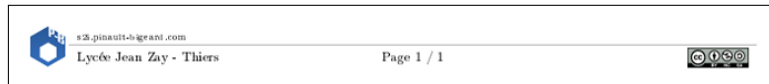


Attention

L'utilisation d'une image en pied de page provoque un léger bug : si on compile le document considéré et que le nombre de page a changé depuis la dernière compilation, le pied de page se positionne n'importe où sur la page. Pour résoudre ce problème, il suffit de recompiler une 2^e fois.

7.2.1 Template par défaut





Paramètres obligatoires

- **UPSTIidClasse** : code classe :

◇ 0 : Custom (définir manuellement UPSTIclasse dans le préambule du document)	◇ 3 : PCSI	◇ 9 : ATS
◇ 1 : PTSI	◇ 4 : PSI	◇ 10 : PC
◇ 2 : PT (défaut)	◇ 5 : MPSI	◇ 11 : Sup
	◇ 6 : MP	◇ 12 : Spé
	◇ 7 : TSI1	
	◇ 8 : TSI2	
- **UPSTIidTypeDocument** : type de document (Cours, TD, TP, etc...) :

◇ 0 : Custom (définir manuellement UPSTItypeDocument dans le préambule du document)	◇ 3 : TP	◇ 10 : Mémo
◇ 1 : Cours (défaut)	◇ 4 : Colle	◇ 11 : Dossier Technique
◇ 2 : TD	◇ 5 : DS	◇ 12 : Dossier Ressource
	◇ 6 : DM	◇ 13 : Concours Blanc
	◇ 7 : Fiche Méthode	◇ 14 : Document Réponses
	◇ 8 : Fiche Synthèse	◇ 15 : Programme de colle
	◇ 9 : Formulaire	◇ 16 : QCM
- **UPSTIidVersionDocument** : version prof ou élève (voir §6) :
 - ◇ 1 : version prof (avec le corrigé affiché) ;
 - ◇ 2 : version élève (sans le corrigé) ;
 - ◇ 3 : version à publier.
- **UPSTItitreEnTete** : titre principal dans le bandeau de la première page ;
- **UPSTInumeroVersion** : version du document.

Paramètres facultatifs

- **UPSTIauteur** : nom de l'auteur, si ce n'est pas l'auteur par défaut défini dans UPSTI_Custom.sty ;
- **UPSTIiddocumentCollegial** : si le document est fait par une équipe ;
- **UPSTIduree** : durée de l'activité ;
- **UPSTIidMatiere** : code matière (S2I, IPT, TIPE...) ;
 - ◇ 0 : Custom (il faut définir manuellement UPSTIintituleMatiere et UPSTIisigleMatiere ;
 - ◇ 1 : S2I (défaut) ;
 - ◇ 2 : IPT ;
 - ◇ 3 : TIPE.
- **UPSTImessage** : texte personnalisé dans la ligne sous le titre dans le bandeau ;
- **UPSTIinoteBasDePremierePage** : en bas à droite de la première page ;
- **UPSTInumero** : numéro du TD, Cours, DS, etc. ;
- **UPSTInumeroChapitre** : numéro du chapitre ;
- **UPSTIprogramme** : références au programme, dans la ligne sous le titre dans le bandeau ;
- **UPSTIserie** : numéro de série (pour les TP notamment) ;
- **UPSTIsource** : sources et références ;
- **UPSTIsousTitreEnTete** : sous-titre dans le bandeau de la première page ;
- **UPSTItitre** : titre de l'exercice ;
- **UPSTItitrePreambule** : sous-titre de l'exercice ;
- **UPSTIvariante** : variante du template (en fonction du lycée... voir §5).

7.2.2 Template pour les QCM AMC

Les paramètres à définir pour le template QCM sont les mêmes que pour le template par défaut.

Commandes spécifiques pour le template QCM

- `\UPSTIamcZoneIdentification[opt nbDigits][opt codeChamp]` : affiche le champ d'identification de l'étudiant (Nom + code étudiant)
 - ◇ `nbDigits` : nombre de chiffres pour le code étudiant (par défaut : 2)
 - ◇ `codeChamp` : nom du champ identification pour AMC.

7.2.3 Template pour les sujets de colle

Paramètres obligatoires :

- `UPSTIidVersionDocument` : version prof ou élève (voir §6) :
 - ◇ 1 : version prof (avec le corrigé affiché) ;
 - ◇ 2 : version élève (sans le corrigé) ;
 - ◇ 3 : version à publier.
- `UPSTItitreEnTete` : thème du sujet de colle ;
- `UPSTItitre` : titre du sujet de colle ;

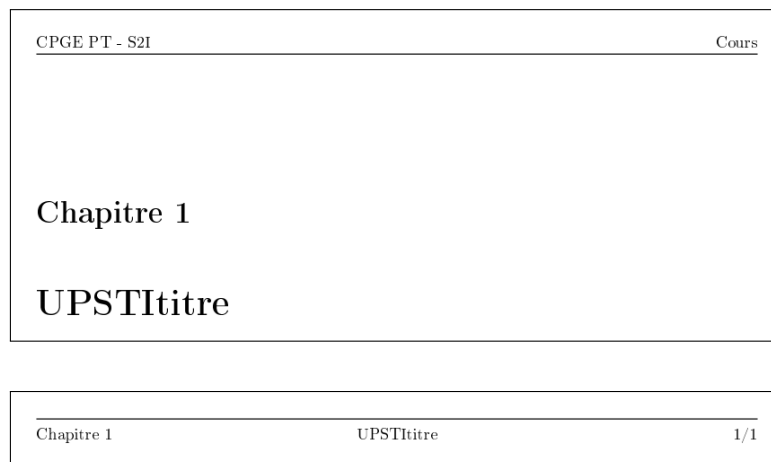
- `UPSTInumeroVersion` : version du document.

Paramètres facultatifs

- `UPSTIdocumentCollegial` : si le document est fait par une équipe ;

7.3 Style basic

7.3.1 Template par défaut



Paramètres obligatoires

- `UPSTIidClasse` : code classe :

◇ 0 : Custom (définir manuellement dans le préambule du document)	◇ 3 : PCSI	◇ 9 : ATS
◇ 1 : PTSI	◇ 4 : PSI	◇ 10 : PC
◇ 2 : PT (défaut)	◇ 5 : MPSI	◇ 11 : Sup
	◇ 6 : MP	◇ 12 : Spé
	◇ 7 : TSI1	
	◇ 8 : TSI2	
- `UPSTIidTypeDocument` : type de document (Cours, TD, TP, etc...) :

◇ 0 : Custom (définir manuellement dans le préambule du document)	◇ 3 : TP	◇ 10 : Mémo
◇ 1 : Cours (défaut)	◇ 4 : Colle	◇ 11 : Dossier Technique
◇ 2 : TD	◇ 5 : DS	◇ 12 : Dossier Ressource
	◇ 6 : DM	◇ 13 : Concours Blanc
	◇ 7 : Fiche Méthode	◇ 14 : Document Réponses
	◇ 8 : Fiche Synthèse	◇ 15 : Programme de colle
	◇ 9 : Formulaire	◇ 16 : QCM
- `UPSTIidVersionDocument` : version prof ou élève (voir §6) :
 - ◇ 1 : version prof (avec le corrigé affiché) ;
 - ◇ 2 : version élève (sans le corrigé) ;
 - ◇ 3 : version à publier.
- `UPSTItitre` : titre principal.

Paramètres facultatifs

- `UPSTIidMatiere` : code matière (S2I, IPT, TIPE...);
 - ◊ 0 : Custom (il faut définir manuellement `UPSTIintituleMatiere` et `UPSTIsigleMatiere`;
 - ◊ 1 : S2I (défaut);
 - ◊ 2 : IPT;
 - ◊ 3 : TIPE.
- `UPSTImessage` : texte personnalisé dans la ligne sous le titre dans le bandeau;
- `UPSTInumeroChapitre` : numéro du chapitre.

7.3.2 Template pour les QCM AMC

● ●

Actions mécaniques

Nom et prénom :

0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9

● ●

Actions mécaniques - 1/2

Commandes spécifiques pour le template QCM

- `\UPSTIamcZoneIdentification[opt nbDigits][opt codeChamp]` : affiche le champ d'identification de l'étudiant (Nom + code étudiant). Voir §7.2.2.

7.3.3 Template pour les sujets de colle

CPGE Colle

Colle: UPSTItitreEnTete
UPSTItitre

UPSTItitre 1/2

Paramètres obligatoires :

- `UPSTIidVersionDocument` : version prof ou élève (voir §6) :
 - ◊ 1 : version prof (avec le corrigé affiché);
 - ◊ 2 : version élève (sans le corrigé);
 - ◊ 3 : version à publier.

- `UPSTItitreEnTete` : thème du sujet de colle ;
- `UPSTItitre` : titre du sujet de colle ;

7.4 Style vide

Lorsque l'option `vide` est utilisée, les en-têtes et pieds de page sont vides. On peut utiliser cette option si on ne souhaite pas que le package `UPSTI_Document` gère la mise en page.

7.5 Corrigés UPSTI

Ce style est utilisé pour la rédaction de corrigés à utiliser au nom de l'UPSTI. On peut sans problème l'utiliser avec les options `noCustomPackages` voire même `noPackages`, si on souhaite conserver ses habitudes de rédaction des documents L^AT_EX.

Paramètres obligatoires

- `UPSTIidConcours` : code concours :
 - ◊ `0` : Custom (définir manuellement `UPSTIconcoursIntitule` et `UPSTIconcoursSigle` dans le préambule du document)
 - ◊ `1` : Banque PT
 - ◊ `2` : CCMP
 - ◊ `3` : CCS
 - ◊ `4` : e3a
 - ◊ `5` : X/ENS
 - ◊ `6` : CCP
 - ◊ `7` : ATS
- `UPSTIidVersionDocument` : version prof ou élève (voir §6) :
 - ◊ `1` : version prof (diffusée sur la partie privée du site de l'UPSTI) ;
 - ◊ `2` : version élève (avec image en filigrane et page d'avertissement).
- `UPSTIfiliere` : filière concernée (on peut indiquer plusieurs filières ex : MP, PSI, TSI) ;
- `UPSTIepreuveIntitule` : intitulé de l'épreuve (ex : Sciences Industrielles A) ;
- `UPSTIepreuveSigle` : sigle de l'épreuve (ex : SIA) ;
- `UPSTIsession` : année de l'épreuve ;
- `UPSTItitreSujet` : sujet traité.

Paramètres facultatifs

- `UPSTIauteur` : auteur(s) du corrigé.

7.6 Modèle perso

Si on souhaite utiliser ses modèles de documents personnels, ou si on souhaite modifier les styles distribués par défaut dans le package, on peut modifier le fichier `UPSTI_Modele_Perso.sty`.

Il faudra cependant garder la structure de ce fichier et bien évidemment le manipuler avec précaution !

8 Script de compilation rapide (pour Windows)

8.1 Présentation

Ce package est distribué avec le VBscript `UPSTIcompilRapide.vbs` qui peut être utilisé sous Windows et qui a plusieurs fonctions :

- permettre la compilation des fichiers `.tex` depuis l'explorateur Windows ;
- créer en une seule manipulation le fichier prof et le fichier élève si nécessaire ;
- zipper automatiquement l'ensemble des fichiers nécessaires à la compilation du document ;
- uploader les fichiers sur un serveur FTP si nécessaire.



Attention

Ce script n'est absolument pas robuste. Il fonctionne, mais il doit être utilisé avec des pincettes... Mais quand tout fonctionne, le gain de temps est important.

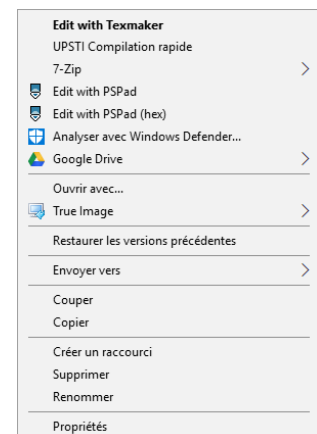
8.2 Installation

Il faut dans un premier temps copier le fichier `UPSTIcompilRapide.vbs` dans un dossier sur le disque (ex : `D:\LaTeX\`) ;

Il faut ensuite éditer le fichier `RegUPSTIcompilRapide.reg` fourni avec le package pour mettre à jour le dossier (s'il est différent de `D:\LaTeX\`).

Ensuite, on peut double-cliquer sur ce fichier pour ajouter les informations au registre. Dès lors, lorsqu'on fera un clic droit sur un fichier `.tex`, on devrait voir apparaître l'option *UPSTI Compilation Rapide* dans le menu contextuel (voir ci-contre).

Il faut ensuite installer 7zip : <http://www.7-zip.org/>.



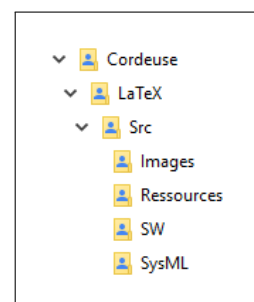
8.3 Configuration du script

Il faut éditer le fichier `UPSTIcompilRapide.vbs` avec un éditeur de texte et compléter la section « Paramètres du script ».

8.4 Organisation des dossiers

Pour que le script fonctionne, il faut respecter une certaine organisation des dossiers :

- **Niveau 0** : le dossier principal du document (`Cordeuse` dans l'exemple ci-contre). C'est ici que le script va copier les versions compilées du document ainsi que le fichier montrant le numéro de version ;
- **Niveau 1** : le fichier `.tex` sera stocké dans un sous-dossier `LaTeX` du dossier principal. On y disposera le fichier `@parametres.upsti` qui est un fichier texte qui contient un mot de 6 chiffres :
 - ◊ 1^{er} chiffre : vaut 1 si on doit compiler la version élève. 0 sinon.
 - ◊ 2^e chiffre : vaut 1 si on doit compiler la version prof. 0 sinon.



- ◇ 3^e chiffre : vaut 1 si on doit uploader un diaporama. 0 sinon.
- ◇ 4^e chiffre : vaut 1 si on souhaite créer un zip lors de la compilation. 0 sinon.
- ◇ 5^e chiffre : vaut 1 si on souhaite compiler une version « À publier » (voir §). 0 sinon.
- ◇ 6^e chiffre : vaut 1 si on souhaite que le script copie les fichiers compilés dans le dossier de niveau 0.
- **Niveau 2** : dans ce dossier, on stockera toutes les fichiers qui seront intégré au zip lors de la compilation (dont les images nécessaires à la compilation. Voir arborescence ci-dessus).

8.5 Upload sur un serveur FTP

Si on souhaite uploader les documents compilés et les sources sur un serveur FTP, il faut remplir 3 conditions :

- avoir complété les informations relatives au serveur FTP dans le fichier `UPSTIcompilRapide.vbs` ;
- avoir installé ncFTP (voir dans le zip du package) ;
- avoir disposé dans le dossier Niveau 1 un fichier texte `@dossierFTP.upsti` contenant le dossier dans lequel uploader les fichiers sur le serveur FTP.

Si le dossier contenant le fichier `.tex` ne contient pas le fichier `@dossierFTP.upsti` alors la phase d'upload est simplement ignorée par le script.

8.6 Convention de nommage

Admettons que le fichier `.tex` s'appelle `monFichier.tex`. Le script de compilation créera les fichiers suivants (selon le contenu de `@parametres.upsti`) :

- `monFichier.tex` : version élève ou à publier ;
- `monFichier-Prof.tex` : version prof (corrigé...);
- `monFichier-Eleve.tex` : s'il s'agit d'un cours (document avec les zones à compléter. Voir §).

Si on souhaite uploader aussi un diaporama, celui-ci devra s'appeler : `monFichier-Diaporama.pptx`.

9 Migration depuis le package EPB_Cours.sty

Dans le zip du package, on trouvera aussi un fichier `Migration EPB-UPSTI.py`. En l'exécutant, on peut aller choisir un fichier `.tex` utilisant les packages `EPB_Cours`, `EPB_SI` et `EPB_Pedagogique`. Le script se chargera de l'adapter au package `UPSTI_Document`.

Là encore, le script n'est pas parfait, mais fonctionnel dans la très grande majorité des cas.