# Utilisation de l’importeur/exporteur Excel pour Eclipse/Capella

## Installation

Le composant est livré sous forme d’un update site composé d’un ensemble de plugins eclipse, accompagné d’un importeur d’exemple permettant d’importer une liste de classe définies dans un fichier excel dans un ePackage d’un fichier ecore.

Il demande un prérequis : l’installation d’apache POI. Celui-ci est disponible via Clarity. Le même POI, en même version, est également utilisé par le générateur documentaire m2doc.

Les plugins du moteur doivent être installé via la commande d’installation d’éclipse ou de capella. On a donc le choix entre installer le moteur, et/ou les exemple. Même si le moteur est basé sur EMF exclusivement, les exemples peuvent être basé sur Capella comme en EMF Pur.

Contenu des plugins et fonctions

|  |  |
| --- | --- |
| org.obeonetwork.excel.importer | Moteur d’importation Excel |
| org.obeonetwork.excel.exporter | Moteur d’exportation Excel |
| org.obeonetwork.excel.importer.ui | Interface utilisateur permettant de déclencher un import dans un modèle, ainsi que l’export d’une portion de modèle dans un fichier excel. |
| org.obeonetwork.excel.exporter.wizard | Wizard d’export sans programmation, permettant d’exporter toutes les instances d’une métaclasse avec leur attributs. |
| org.obeonetwork.excel.importer.automatic | Importeur automatique permettant d’importer |
| org.obeonetwork.excel.importer.ecore.example | L’exemple d’importateur basé EMF, permettant d’importer une liste d’Eclasses dans un ePackage |
| org.obeonetwork.excel.importer.capella.example | Importeur basic basé sur capella |
| com.thalesgroup.clarity.excelimport.examples | Importeur d’exemple montrant un import de fonctions avec hiérarchie, mais également des liens sur des Requirements. |

## Importeur : Utilisation par l’utilisateur

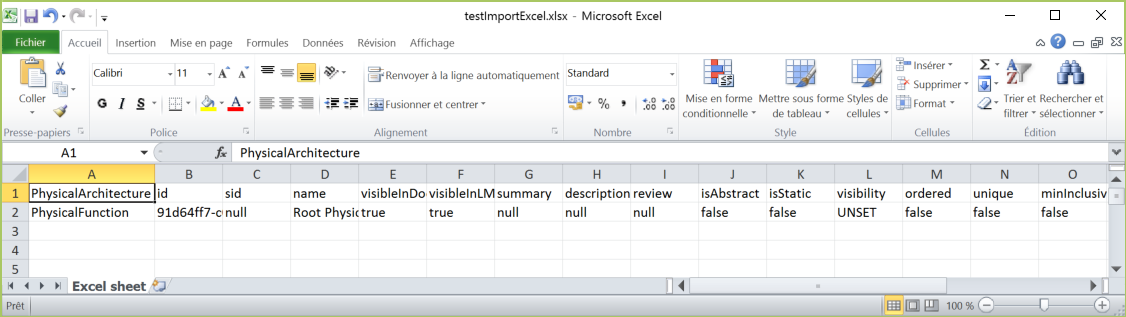
L’interface utilisateur est aujourd’hui composée d’une seule commande. Dans son explorateur, l’utilisateur sélectionne son fichier Excel et la destination d’import. Ensuite, il lui suffit de faire clic-droit et « import » dans la catégorie correspondante.

|  |  |
| --- | --- |
| Voici le fichier excel source : une liste de classe, avec pour chacune, le fait d’être abstraite, ses héritages et pour chaque classe un nom d’attribut |  |
| On lance l’importation avec la sélection du fichier Excel dans l’explorateur, et en précisant l’endroit où l’import va être réalisé, ici un ePackage. |  |
| Une boite de dialogue s’ouvre ensuite, précisant l’importeur à utiliser. Ici, on utilisera l’importeur eCore. On valide avec Ok. |  |
| Le contenu du fichier excel est donc finalement correctement importé. |  |
|  |  |

## Exporteur : Utilisation par l’utilisateur

L’exporteur excel est disponible dans le menu contextuel, à partir d’un élément EMF du modèle, on sélectionne la commande « Export ».

|  |  |
| --- | --- |
|  | L’exporteur excel est disponible dans le menu contextuel, à partir d’un élément EMF du modèle, on sélectionne la commande « Export ». |
| Une boite s’ouvre permettant de choisir l’exporteur à utiliser. Par defaut, c’est l’exporteur wizard, présenté dans ce chapitre, qui est utilisé. Mais l’utilisateur peux présenter d’autres exporteur, plus cablés et automatique, qui n’ont pas besoin d’interface utilisateurs. |  |
|  | La fenêtre suivante permet la sélection de la metaclasse. Le principe de cet exporteur, c’est que l’on va exporter tous les éléments de la metaclasse choisie située dans le sous arbre de l’objet sur lequel on a appelé la commande. |
| Sse |  |

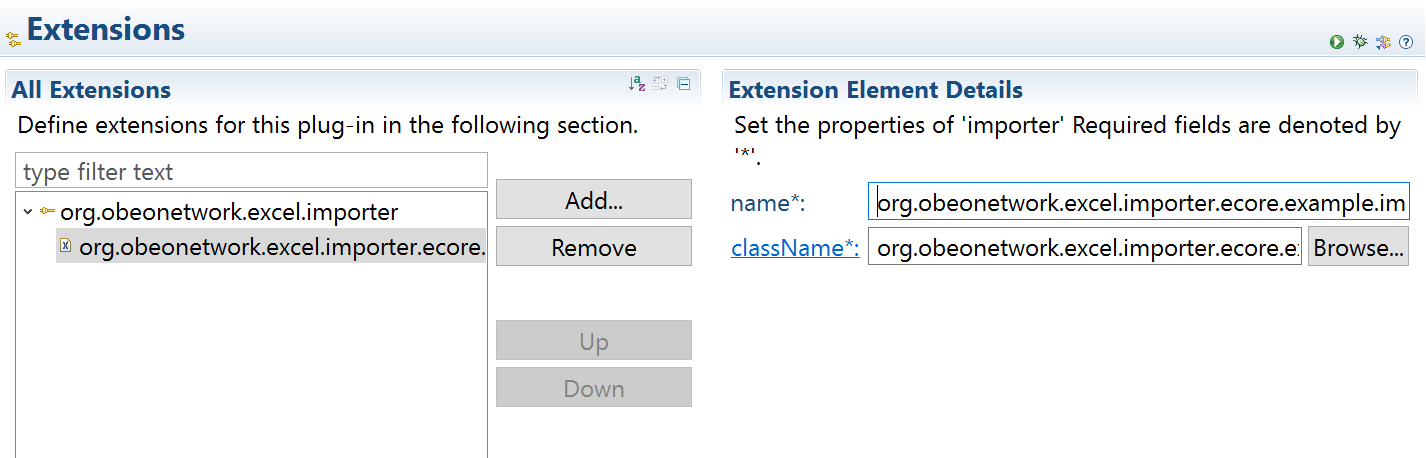


Et voici au final le fichier exporté. Une ligne par objet exporté, avec les colonnes correspondant aux différents attributs, mais aussi extensions et property values portées par l’élément.

## Création d’un nouvel Importeur

La création d’un nouvel importeur se fait en java, en implémentant un point d’extension publié par le moteur d’import. Ce chapitre va expliciter les différentes étapes pour créer un nouveau format d’importation.

Il faut déjà commencer par créer un nouveau projet de plugin, et, dans les dépendances, rajouter le plugin du moteur, org.obeonetwork.excel.importer. Ensuite, dans l’onglet extensions, il faut implémenter le point d’extension de même nom.



Dans ce point d’extension, l’attribut className est une classe du plugin implémentant l’interface org.obeonetwork.excel.importer.IExcelImporter. C’est l’implémentation de cette interface qui va permettre de préciser le comportement de l’importeur.

Le principe de l’importeur est un parcours par ligne du fichier excel, et pour chaque ligne, une action est appelée sur l’interface, en lui passant comme paramètre la liste des contenus de cellules, cellules vides incluses pour ne pas perdre les alignements de données.

Les 5 méthodes à implémenter sont :

|  |  |
| --- | --- |
| public String getName() | Retourne le nom de l’importeur, c’est lui qui est affiché dans la boite de dialogue d’import |
| public void setDestination(EObject destinationObject) | Recoit la destination EMF de l’import ; Cette méthode permet de le garder dans un attribut de la classe, mais aussi de vérifier que l’importeur est appelé sur le bon type d’objet. |
| public void computeFirstLine(List<String> lineData) | Appelée lors du parsing de la première ligne excel ayant du contenu. Cette méthode permet souvent de récupérer les informations d’abscisse, qui peuvent être variables. On peut aussi en profiter pour vérifier qu’il s’agit bien du format Excel attendu. |
| public void computeOtherLine(int lineNumber, List<String> lineData) | Méthode centrale de l’import, cette méthode est appelée pour chaque ligne du fichier en passant les données de la ligne, éventuellement avec les chaines vides pour les cellules vides, et le numéro de la ligne (commençant à 0, attention à la différence avec Excel) |
| public void importEnded() | Méthode appelé en fin d’import. C’est souvent le moment pour valider le modèle obtenu, voire de créer des liens entres les éléments importés, car c’est le moment où tout le fichier a été monté en mémoire. |

L’exemple d’import ecore a spécialement été réalisé pour aider à la compréhension. N’hésitez pas a vous y référer pour démarrer votre propre importeur.