

OpenJUMP – Jeux d'attributs

édition 1 (2019-09-16)

Ce document décrit l'extension SetAttributes (Jeux d'attributs) pour OpenJUMP.

History

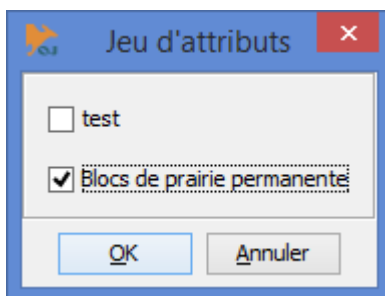
- **version 0.8 [2019-09]** ajoute la possibilité de regrouper plusieurs jeux d'attributs dans un menu déroulant. Cette nouvelle fonctionnalité offre un moyen simple de saisir des attributs de type énuméré.

- **version 0.7.1 [2015-06]** ajoute le support du undo/redo

1 - Installation

Le plugin s'installe en déposant le fichier SetAttributes-x.y.jar dans le dossier lib/ext
Il donne accès à un nouveau menu :

Extensions > Jeux d'attributs



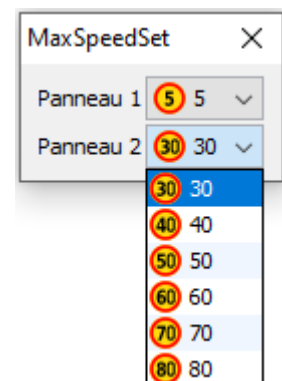
Cette commande ouvre une boîte de dialogue qui contient une case à cocher par jeu d'attributs disponible.

Le plugin « jeux d'attributs » présente dans un premier temps **une case à cocher pour chaque fichier de définition trouvé**. Cocher les cases correspondant aux palettes de jeux d'attributs que l'on veut ouvrir, puis valider.

Un jeu d'attribut est défini par un fichier xml et se présente à l'écran sous la forme d'une boîte de dialogue contenant des boutons :



ou des menus déroulants :



Note : le chargement des jeux d'attributs se fait à l'exécution du plugin (Extensions > Jeux d'attributs) et non au démarrage d'OpenJUMP. Il est donc possible de modifier la définition d'une palette et de recharger celle-ci en passant par le menu sans avoir à relancer OpenJUMP.

2 – Emplacement des fichiers de définition

A l'exécution du plugin, les fichiers de définition des « jeux d'attributs » sont recherchés aux emplacements suivants :

`%OpenJUMP%/lib/ext/set_attributes` (où %OpenJUMP % représente le répertoire d'installation)

`~/.OpenJUMP/set_attributes` (où ~ représente le répertoire utilisateur)

Ils sont recherchés dans cet ordre, si bien qu'un jeu d'attributs situé dans le répertoire utilisateur et nommé de la même façon qu'un jeu d'attribut situé dans le répertoire d'installation remplacera ce dernier.

3 – Fichier de définition (référence)

Une palette de jeux d'attributs est défini dans un fichier xml qui contient les éléments et attributs suivants (fichier minimaliste contenant 2 boutons) :

```
<?xml version="1.0" encoding="UTF-8"?>
<setAttributesToolbox title="Titre boîte de la dialogue">
  <button text="A">
    <attribute name="Nom" value="a" />
    <attribute name="Remarque" value="modifié" />
  </button>
  <button text="B">
    <attribute name="Nom" value="b" />
    <attribute name="Remarque" value="modifié" />
  </button>
</setAttributesToolbox>
```

La version 0.8 ajoute la possibilité de définir une liste déroulante proposant différents jeux d'attributs.

En voici un exemple minimaliste présentant les mêmes jeux d'attributs que précédemment mais dans un menu déroulant :

```
<?xml version="1.0" encoding="UTF-8"?>
<setAttributesToolbox title="Titre boîte de la dialogue">
  <combo text="titre du menu déroulant">
    <button text="A">
      <attribute name="Nom" value="a" />
      <attribute name="Remarque" value="modifié" />
    </button>
    <button text="B">
      <attribute name="Nom" value="b" />
      <attribute name="Remarque" value="modifié" />
    </button>
  </combo>
</setAttributesToolbox>
```

Chacun des éléments **setAttributesToolbox**, **button** et **attribute** supporte aussi des options

facultatives décrites ci-dessous.

setAttributesToolbox

Cet élément représente la boîte de dialogue. Il peut contenir plusieurs éléments **button** ainsi que les attributs suivants :

title	titre de la boîte de dialogue (ex. title="Largeurs de routes")
maxCol	nombre maximum de boutons dans une ligne avant de passer à la ligne suivante (ex. maxCol="10"). Par défaut, la palette fabrique des lignes de 6 boutons.
iconWidth	définit la largeur des icones (ex. iconWidth="24")
iconHeight	définit la hauteur des icones (ex. iconHeight="24"). Si les boutons contiennent des icones, on préférera généralement une hauteur égale à la largeur. Si les boutons contiennent du texte, on préfère généralement une largeur plus importante permettant de visualiser la totalité du texte.

combo

Cet élément représente un menu déroulant. Il offre plusieurs choix, chaque choix étant l'équivalent d'un « button » (cf ci-dessous).

L'élément combo ne supporte qu'un seul attribut « text » permettant de donner un titre au menu déroulant.

Si on veut se servir du menu déroulant pour permettre la saisie d'un seul attribut de type énuméré, on mettra par exemple le nom de l'attribut dans cet attribut text.

text	Titre du menu déroulant venant se placer à gauche du menu déroulant.
------	--

button

Cet élément représente un bouton. Il peut contenir plusieurs éléments **attribute** ainsi que les attributs suivants

icon	Chemin de l'icone à afficher dans le bouton. Ce chemin est relatif à la position du fichier de définition xml (ex. icon="images/train.png"). Il supporte les icones au format png et jpg mais pas celles au format ico.
text	Texte à afficher dans le bouton quand aucune icône n'est définie (ou trouvée).
atomic	Si cet attribut vaut true, les champs de l'objet ne seront pas modifiés si l'un des champs défini par le bouton n'est pas trouvé (ou si l'une des valeurs définies par le bouton n'est pas compatible avec l'attribut à remplir) Si cet attribut vaut false, les champs de l'objet qui peuvent être modifiés sont modifiés et ceux qui ne peuvent pas l'être à cause d'un problème de schéma ne le sont pas.
layer	Permet de restreindre les couches sur lesquelles on souhaite agir. On peut indiquer le nom de la couche à modifier (ex. layer="route") ou toutes les couches (layer="*") ou encore une famille de couches (layer="route*"). Le nom de la couche ou des couches est sensible à la casse.
tooltip	Bulle d'aide à afficher quand on survole le bouton

attribute

Cet élément représente un attribut à changer. Il contient les attributs suivants

name	Nom de l'attribut à changer (obligatoire!)
value	Nouvelle valeur (obligatoire). Il n'est pas possible, pour l'instant, d'indiquer qu'on peut vider un champ. Le transtypage se fait en fonction du type de l'attribut cible.
prerequisite	Cet attribut permet d'indiquer un pré-requis sans lequel l'attribut ne sera pas modifié. Il peut prendre les valeurs suivantes : <ul style="list-style-type: none">- isNull : la valeur actuelle ne sera changée que si elle est nulle- isNullOrEmpty : la valeur actuelle ne sera changée que si elle est nulle ou vide- =someText : la valeur actuelle ne sera changée que si elle vaut « someText »- #someText : la valeur actuelle ne sera changée que si elle est différente de « someText »

4 – TODO

~~Pour l'instant, le remplissage d'attributs avec jeux d'attributs n'est pas « annulable »~~

La modification d'attributs via jeux d'attributs est annulable.

~~Permettre de vider un attribut en remplaçant la valeur par null.~~

Un attribut peut être vidé (=null) simplement en omettant l'attribut « value »

Permettre de remplir un attribut avec une expression telle que la date courante ou le nom d'utilisateur