

Liste des nouveautés AsciiDoctor version 4.1.0 au 10/09/2024

- Mettre la documentation à jour pour la prise en charge d'un logo via l'attribut utilisateur : `_user_cover_logo_filename` suivi du nom de l'image. Si le logo est global, son image doit être placée dans le répertoire des images au niveau `config/images/nom_image.png` et la configuration ci-dessous doit être faite dans le fichier `config/attributes/global_application_attributes.adoc`

```
:_user_cover_logo_filename: none
```

Préciser qu'il est possible d'écraser le logo défini au niveau global par le logo local en écrivant la même configuration mais dans le fichier `books/un_theme/dossier_du_livre/attributes/local_application_attributes.adoc` sans oublier d'activer l'attribut `_use_local_application_attributes` dans le fichier `main.adoc` du livre puis en plaçant l'image dans le répertoire `images` du livre, soit `books/un_theme/dossier_du_livre/images/nom_image_logo.png`

Si un logo global est défini et que l'on n'en veut pas au niveau local, il faut spécifier la valeur `none` dans le fichier `books/un_theme/dossier_du_livre/attributes/local_application_attributes.adoc` :

```
:_user_cover_logo_filename: none
```

- possibilité de désactiver localement l'image de la page de garde définie au niveau global (même principe que pour le logo mais avec l'attribut `_user_cover_image_filename`)
- possibilité d'afficher en mode portrait ou paysage via l'attribut `_user_pdf_layout` et les valeurs `portrait` ou `landscape`
- possibilité de spécifier une image d'icône de logo qui peut être utilisable depuis un fichier de thème. Pour définir l'image d'icône de logo (au niveau global ou local), il faut spécifier l'attribut suivant :

```
:_user_logo_icon_filename: icone_max_height_30px.png
```

Puis dans le fichier de thème, l'attribut `_logo_icon` peut être utilisé pour cibler l'image d'icône de logo :

```
header:
  recto:
    left:
      content: image: {_logo_icon}[]
```

Grâce à AsciiDocPro, que l'image d'icône de logo soit placée dans le dossier des images "globales"

ou dans celui du livre, l'image est retrouvée. De plus, si aucune image d'icône de logo n'est spécifiée, tout continue de fonctionner normalement. Vous n'avez pas besoin de supprimer la macro d'image, AsciiDocPro charge une image transparente de 1px de côté.

Lorsqu'une image d'icône de logo a été définie au niveau globale et qu'il n'en faut aucune au niveau local, alors il faut spécifier la valeur `none` à l'attribut au niveau local, donc d'un livre:

```
:_user_logo_icon_filename: none
```

NOTE Une taille d'icône peut être spécifiée telle que : `image:{_logo_icon}[width=30]`

Cependant, si vous "désactivez" l'icône localement ou qu'aucun icône de logo n'est défini, l'image "vide" va occuper 30px et se voir (même si elle est transparente)

- ajout de la possibilité d'utiliser des fichiers de thèmes organisés dans des sous-répertoires (à la condition qu'ils soient créés dans le dossier `config/themes`). Le fichier de thème appelé depuis `global_application_attribute` ou `local_application_attributes` doit être placé à la racine du dossier de thème (ceci afin de garantir la comptabilité avec AsciiDocPro. Si cette règle n'est pas respectée, le fichier de thème ne sera pas chargé et le fichier de thème "défaut" prendra la relève. Par contre, les différents "composants" du fichier de thème peuvent être placés dans un sous-dossier, dans plusieurs sous-dossiers, etc. Il faut pour cela utiliser la clé `extends` et liste les fichiers à inclure en veillant à leur ordre d'ajout.
- nouveau bloc à utiliser pour lister les ressources que les étudiants doivent récupérer.

Le label utilisé dans ce bloc peut être personnalisé globalement ou localement via l'attribut `_user_bloc_resources_student_label`

La liste des ressources doit respecter un format précis. Chaque item de niveau 1 dans la liste correspond au nom de la ressource, sa description, etc. mais pas à son emplacement. Les sous-items de niveau 2 listent la ou les éléments (fichier, cours, url, etc.) relatif à la description faite dans l'item supérieur. Ainsi, plusieurs ressources peuvent être listées avec pour chaque ressource un ou plusieurs éléments :

Voici un exemple de déclaration d'un bloc de ressources

```
[.resourcesstudent]
****
{ _bloc_resources_student_label }

* (une première ressource) nom de la ressource et ou explication de la ressource et ou description etc.
** lien1 vers la ressource (ce peut être un lien externe, une url par exemple)
** lien2 vers la ressource ([.warning]#si une ressource interne est pointée, celle-ci doit être placée dans le dossier 'assets' du fichier de chapitre ou d'exercice#)
** lien3 vers un autre chapitre du projet (le lien ne sera pas fonctionnel mais il aura le mérite d'indiquer le chapitre à utiliser) ou un renvoi sans lien vers le chapitre concerné ou encore le livre concerné ou une correction
```

```
* (une deuxième ressource) nom, description, explication pour une autre ressource
** lien vers un fichier, une annexe, etc.
```

```
//end ressourcesstudent
****
```

IMPORTANT

Respecter le pattern d'un bloc de ressources vous permet de partager votre support avec les mêmes règles de conception d'un utilisateur à l'autre. De plus, AsciiDocProManager va identifier les ressources en fonction de ce pattern.

- nouveau bloc à utiliser pour lister les ressources du professeur.

Le label utilisé dans ce bloc peut être personnalisé globalement ou localement via l'attribut `_user_bloc_ressources_teacher_label`

Voici un exemple de déclaration d'un bloc de ressources du professeur

```
[.resourcesteacher]
****
*{_bloc_ressources_teacher_label}*

* fichier path/to/folder/support_correction.sql
* fichier path/to/folder2/annexe.pdf

//end resources
****
```

Pour ne pas afficher les éléments destinés au professeur, il est conseillé de l'envelopper dans un test permettant de contrôler l'affichage ou le masquage du bloc via le nouvel attribut `_show_teacher` :

NOTE

```
ifeval::[_show_teacher] == 1]
//ici un contenu qui ne sera affiché que pour le professeur (en
fonction de la configuration de la valeur de l'attribut _show_teacher
//end _show_teacher
endif::[]
```

- ajout de nouveaux attributs utilisateurs:
 - `_user_bloc_ressources_student_label` permet de personnaliser le label utilisé dans le bloc de ressources de l'étudiant. La valeur par défaut est "Ressources à récupérer".
 - `_user_bloc__teacher_label` permet de personnaliser le label utilisé dans le bloc de ressources du professeur. La valeur par défaut est "Ressources pour le professeur".
 - `_user_show_teacher` permet de contrôler l'affichage de tout élément enveloppé par une structure d'évaluation de la valeur de cet attribut. La valeur par défaut est 1 ce qui indique

que le contenu est affiché par défaut (tous les éléments sont affichés par défaut dans le framework AsciiDoc)

Exemple d'utilisation pour afficher les ressources qui ne concernent que le professeur :

```
ifeval::[_show_teacher] == 1]
[.resourcesteacher]
****
{ _bloc_ressources_teacher_label}

//ici les resources

//end resourcesteacher
****
//end _show_teacher
endif::[]
```

- ajout d'un nouvel attribut de métadonnée `_container_path` dont la valeur correspond au chemin du dossier qui contient le fichier courant (un exercice ou un chapitre). La valeur de cet attribut est affichée **sous les métadonnées affichées sous chaque titre de chapitre** d'un livre. L'affichage de cette valeur est conditionnée à la valeur de l'attribut `_show_teacher`.

Voici un exemple d'utilisation de l'attribut de métadonnée `_container_path` :

```
:_author:
:_duration:
:_duration_of_correction:
:_version_number: 1.0.0
:_version_date: 03/09/2024 à 16:54
:_container_path: chapters\theme_A\sujet_7\exercices\live_template_directive_if ①
:_chapter:
[[chapitre_ceci_est_un_titre_de_chapitre]]
= Ceci est un titre de chapitre
Unresolved directive in mises_a_jour.adoc - include:../../../run_app.adoc[]
```

- ① chemin du dossier parent du fichier courant main.adoc (ici un dossier d'exercice). Le chemin est donné depuis la racine du projet AsciiDocPro.

TIP

La valeur de `_container_path` est utile pour le professeur qui doit faire une mise à jour d'un chapitre car il dispose du nom du dossier qui le contient dans le pdf (généré avec la valeur d'attribut `1` pour `_show_teacher`).

Pour retrouver rapidement le dossier conteneur, depuis l'IDE, faire un double appui sur la touche `kbd:[Maj]` et coller le nom du dossier parent situé en fin du chemin. Cela permet de retrouver rapidement le dossier conteneur.