



## Comment écrire du contenu

### 1. Conteneur (l'élément Content)

Nous listons ici les différents types de **Content**. Les Content sont posés dans les Sections des documents.

#### Exemple 1: Content non typé

FormPublisher est une suite ouverte de génération de **formulaires** et d'applications libres qui permet de créer et de mettre en ligne des guichets électroniques ou tout processus expert à base de formulaires intelligents. Les applications générées par FormPublisher s'intègrent facilement dans une architecture ouverte de services en ligne. La notion de document intelligent proposée par J-Way permet à un rédacteur non informaticien de créer un **formulaire** de manière intuitive ; FormPublisher transforme ce document en une application didactique qui se comporte comme un véritable système expert d'interview et de rédaction au service de l'utilisateur. Il est particulièrement adapté pour les applications de type e-gov (e-gouvernement).

#### Exemple 2: Content de type *example*

FormPublisher permet de créer facilement des **SmartForms** "formulaires intelligents indépendants de toute architecture spécifique et déployables sans contrainte". A l'aide d'un **SmartForm**, les utilisateurs (citoyens, clients, employés) peuvent facilement remplir les formulaires les plus complexes et produire les documents Pdf associés à ces formulaires. Les **données** recueillies peuvent être utilisées afin de nourrir une base de données ou de s'intégrer dans une gestion de processus (Alfresco ou autres) en s'appuyant sur les extensions fournies par J-Way ou d'autres contributeurs.

#### Exemple 3: Content de type *remark*

FormPublisher fournit tout ce dont une **organisation** a besoin pour créer rapidement des services en ligne efficaces, rentables et conformes aux normes d'accessibilité. Notre solution ouverte se caractérise par sa puissance, sa simplicité de mise en œuvre et son excellent rapport qualité-coût qui est sans commune mesure avec les solutions traditionnelles tant du monde propriétaire que du monde Open Sources. FormPublisher génère des applications Open Sources tout en proposant une garantie "éditeur" sur le processus qui permet de les construire et de les utiliser en ligne.

#### Exemple 4: Content de type *important*

Si vous avez besoin d'une assistance, vous pouvez contacter l'équipe support par mail : [support@jway.lu](mailto:support@jway.lu)

#### Exemple 5: Content de type *définition*

Ce type de Content permet d'afficher les définitions du glossaire.

#### Exemple 6: Content de type *warning*

Le lien sélectionné est manquant. Pour nous signaler cette anomalie, veuillez nous contacter en recopiant l'**URL** de la page en question.

Veuillez nous excuser pour ce désagrément.

#### Exemple 7: Content de type *questions*

Nom

Prénom

### 2. Paragraphes et styles

#### Exemple 1: Paragraphe par défaut (HorizontalAlignment = left, Spacing = normal)

FormPublisher est une suite ouverte de génération de **formulaires** et d'applications libres qui permet de créer et de mettre en ligne des guichets électroniques ou tout processus expert à base de formulaires intelligents. Les applications générées par FormPublisher s'intègrent facilement dans une architecture ouverte de services en ligne. [...]

FormPublisher permet de créer facilement des **SmartForms** "formulaires intelligents indépendants de toute architecture spécifique et déployables sans contrainte". A l'aide d'un **SmartForm**, les utilisateurs (citoyens, clients, employés) peuvent facilement remplir les formulaires les plus complexes et produire les documents Pdf associés à ces formulaires.[...]

#### Exemple 2: Paragraphes HorizontalAlignment = right, Spacing = small.

FormPublisher est une suite ouverte de génération de formulaires et d'applications libres qui permet de créer et de mettre en ligne des guichets électroniques ou tout processus expert à base de formulaires intelligents. Les applications générées par FormPublisher s'intègrent facilement dans une architecture ouverte de services en ligne. [...]

FormPublisher permet de créer facilement des **SmartForms** "formulaires intelligents indépendants de toute architecture spécifique et déployables sans contrainte". A l'aide d'un **SmartForm**, les utilisateurs (citoyens, clients, employés) peuvent facilement remplir les formulaires les plus complexes et produire les documents Pdf associés à ces formulaires.[...]

#### Exemple 3: Paragraphe HorizontalAlignment = center, Spacing = large.

FormPublisher est une suite ouverte de génération de formulaires et d'applications libres qui permet de créer et de mettre en ligne des guichets électroniques ou tout processus expert à base de formulaires intelligents. Les applications générées par FormPublisher s'intègrent facilement dans une architecture ouverte de services en ligne. [...]

FormPublisher est une suite ouverte de génération de formulaires et d'applications libres qui permet de créer et de mettre en ligne des guichets électroniques ou tout processus expert à base de formulaires intelligents. Les applications générées par FormPublisher s'intègrent facilement dans une architecture ouverte de services en ligne. [...]

#### Exemple 4: Paragraphe HorizontalAlignment = justify, Spacing = x-large.

FormPublisher est une suite ouverte de génération de formulaires et d'applications libres qui permet de créer et de mettre en ligne des guichets électroniques ou tout processus expert à base de formulaires intelligents. Les applications générées par FormPublisher s'intègrent facilement dans une architecture ouverte de services en ligne. [...]

FormPublisher permet de créer facilement des SmartForms "formulaires intelligents indépendants de toute architecture spécifique et déployables sans contrainte". A l'aide d'un SmartForm, les utilisateurs (citoyens, clients, employés) peuvent facilement remplir les formulaires les plus complexes et produire les documents Pdf associés à ces formulaires.[...]

#### Les styles :

1. Le style **bold**
2. le style *italic*
3. le style emphasis
4. le style underline
5. le style ~~strikethrough~~
6. le style smaller
7. le style larger
8. le style <sup>exponent</sup>
9. le style <sub>indice</sub>
10. le style **red**

### 3. Listes

#### 👉 Exemple de List number :

1. Un [modèle](#) pour créer les [formulaires](#) et processus d'interview ;
2. Un modèle pour créer les documents informatifs ;
3. Un modèle pour créer les parties communes à plusieurs documents ;
4. Un modèle expert pour créer tout type d'application.

#### 👉 Exemple de List bullet :

- Un modèle pour créer les formulaires et processus d'interview ;
- Un modèle pour créer les documents informatifs ;
- Un modèle pour créer les parties communes à plusieurs documents ;
- Un modèle expert pour créer tout type d'application.

#### 👉 Exemple de List bullet2 contenant une sous-liste dash :

- **En amont, l'expression de la connaissance** : J-Publisher™ s'appuie sur des documents [Xml](#) pour produire automatiquement une application (*publication multimédia, dialogue en ligne ou rédaction automatique*) . [...]
- **Au coeur du processus la génération d'application** :
  - de publication html pour diffuser en ligne des documents, ouvrages et fascicules ;
  - de publication papier de documents, ouvrages et fascicules ;
  - d'instanciation documentaire pour la construire automatiquement des documents personnalisés ;
  - d'interview dynamique pour transformer tout document en [formulaire](#) interactif.
  - de connexion pour intégrer l'application générée dans un service plus large.[...]
- **En aval, le déploiement de l'application générée** : J-Publisher™ produit une application normée librement déployable sans contrainte.[..]

**Exemple de List *indent* :**

- Un modèle pour créer les formulaires et processus d'interview ;
- Un modèle pour créer les documents informatifs ;
- Un modèle pour créer les parties communes à plusieurs documents ;
- Un modèle expert pour créer tout type d'application.

**Exemple de List *letter* :**

- a. Un modèle pour créer les formulaires et processus d'interview ;
- b. Un modèle pour créer les documents informatifs ;
- c. Un modèle pour créer les parties communes à plusieurs documents ;
- d. Un modèle expert pour créer tout type d'application.

**Une List *roman* :**

- I. Un modèle pour créer les formulaires et processus d'interview ;
- II. Un modèle pour créer les documents informatifs ;
- III. Un modèle pour créer les parties communes à plusieurs documents ;
- IV. Un modèle expert pour créer tout type d'application.

## 4. Tableaux

Dans FormPublisher, les tableaux sont créés en utilisant l'élément "Table". Nous définissons et expliquons dans cette section l'utilisation des éléments qui interviennent dans la création des tableaux.

### 4.1. L'élément "Table"

L'élément "Table" est utilisé pour définir un tableau.

**L'élément "ColumnDefinitions":**

Les largeurs des colonnes du tableau sont définies avec l'attribut "ColumnDefinitions". Le nombre de colonnes correspond au nombre de ColumnDefintion à définir. [Cette fenêtre](#) montre un exemple de configuration du "ColumnDefinitions".

**Définir l'alignement d'un tableau:**

L'attribut "HorizontalAlignement" permet de définir l'alignement du tableau.

Pour mettre en évidence des différents alignements de l'élément "Table", nous avons défini ci-dessous trois tableaux de largeur (Width = 50 %) et d'"HorizontalAlignement" différents.

Table HorizontalAlignment = left

Bonjour	Moien
---------	-------

Table HorizontalAlignment = center

Good morning	Guten Morgen
--------------	--------------

Table HorizontalAlignment = right

Buenos días	Buon giorno
-------------	-------------

**Définir les bordures d'un tableau:**

L'attribut "StyleName" permet de spécifier le type de bordures du tableau. Nous récapitulons ci-dessous tous les types de bordures de tableaux.

StyleName = borderOnTable

Text	Text	Text
Text	Text	Text
Text	Text	Text

StyleName = borderOnCells

Text	Text	Text
Text	Text	Text
Text	Text	Text

StyleName = oddEven

Text	Text	Text
Text	Text	Text
Text	Text	Text

StyleName = border

Text	Text	Text
Text	Text	Text

Text	Text	Text
------	------	------

StyleName = noBorder

StyleName = dataQuery

Noms	Prénoms	Participe	Noms	Prénoms	Participe
<input type="text"/>	<input type="text"/>	<input type="radio"/> Oui <input type="radio"/> Non	<input type="text"/>	<input type="text"/>	<input type="radio"/> Oui <input type="radio"/> Non
<input type="text"/>	<input type="text"/>	<input type="radio"/> Oui <input type="radio"/> Non	<input type="text"/>	<input type="text"/>	<input type="radio"/> Oui <input type="radio"/> Non

## 4.2. Définir les bordures et alignements de l'élément Td

Comme dans le cas du Table et du TableBody, l'attribut StyleName du "Td" permet de définir la bordure de la cellule. L'alignement vertical du contenu du "Td" est défini avec l'attribut VerticalAlignment.

Pour une meilleure mise en évidence des styles de cellules, les bordures du "Table" et du TableBody " définis ci-dessous sont de type "noBorder".

### Tableau récapitulatif des bordures du Td ou Th (Th StyleName = noBorder)

Td StyleName = border	Td StyleName = noBorder
-----------------------	-------------------------

Td StyleName = border
-----------------------

Td StyleName = noBorder
-------------------------

### Tableau récapitulatif des bordures du Td ou Th (Th StyleName = border)

Td StyleName = border	Td StyleName = noBorder
-----------------------	-------------------------

Td StyleName = emphasis
-------------------------

Td StyleName = emphasis
-------------------------

Td StyleName = emphasis
-------------------------

### Tableau récapitulatif des alignements d'un Td ou Th

Alignement vertical = top
---------------------------

Alignement vertical = bottom
------------------------------

Alignement vertical = middle
------------------------------

Alignement vertical = baseline
--------------------------------

### 4.3. Exemples de tableaux

#### Exemple 1:

Le tableau ci-dessous est un "Table" (StyleName = border), de quatre colonnes. L'attribut ColumnDefinitions permet de définir les largeurs des colonnes "ColumnDefinitions = 15%, 35%, 20%, 30%". Le tableau est d'une largeur (Width = 100%).

Le TableBody a quatre lignes. Pour la description des cellules du TableBody, confère le tableau.

Ce tableau illustre un exemple de cellules s'étendant sur plusieurs colonnes (exemple de *colspan*) et d'une cellule étendue sur plusieurs lignes (exemple de *rowspan*) . L'alignement du contenu des colonnes peut varier en utilisant l'attribut VerticalAlignment. La largeur de chaque colonne est défini dans ColumnDefintion.

Col1 : 15%	Col2 : 35%	Col3 : 20%	Col4 : 30%
Une cellule Td étendue sur 4 colonnes (Colspan = 4) Alignement vertical centré (VerticalAlignment = top)			
Text	Td StyleName = border	Text	Td étendu sur 2 lignes (Rowspan = 2)
Text	Text	Text	
Td sur 4 colonnes et Paragraph HorizontalAlignment = center			

#### Exemple 2:

Table (StyleName = borderOnTable). Le tableau a trois colonnes de dimensions égales. La largeur du tableau est 80%. Tableau aligné au centre.

Nous avons défini un Caption après le Table.

colonne 1	colonne 2	colonne 3
Td sur 3 lignes et StyleName emphasis	l1c2:	l1c3
	l2c2:	l2c3
	l3c2:	l3c3

**Table n°2 (ceci est la légende du tableau placée en bas)**

### Exemple 3:

Le tableau suivant est une Table (StyleName = borderOnCells) de largeur 90%, et alignée à droite  
La légende est placée en haut du tableau.

#### Légende du tableau <Caption> placé en haut

Cellule Th Colspan='4' Paragraph HorizontalAlignment="center")

Colonne 1	Colonne 2	Colonne 3	Colonne 4
Text	Td sur 3 colonnes		
Text	Text	Text	Text
Text	Td sur 3 colonnes		
Text	Text	Text	Text
Text	Td sur 2 lignes et 2 colonnes (Rowspan = 2 et Colspan = 2)		Text
Text			Text

### Exemple 4:

Le tableau ci-dessous est une Table (StyleName = oddEven) de 70% de large, alignée à gauche

Ce tableau illustre un exemple de "Table" sans "TableHeader" et le "StyleName" des cellules de la première colonne sont de type "emphasis".

Text	Text	Text	Text
Text	Text	Text	Text
Text	Text	Text	Text
Text	Text	Text	Text

## 5. Liens

FormPublisher implémente plusieurs types de liens. Les liens peuvent différer en fonction de leur source et du mode d'affichage. Ci-dessous, nous les illustrons avec quelques exemples.

### Fonction de l'emplacement:

Les attributs Url, EmbeddedRessource, DocumentId permettent de définir l'emplacement de la source du lien.

- Lien vers un site externe (url): [site jway](#)
- Lien vers un fichier interne: [fichier "publication" de cette application](#)
- Lien vers un document interne: [document](#)

### Fonction du type de média en sortie:

Le lien peut générer trois type de média en sortie: Le HTML, le PDF, le WORD. La spécification du média désiré en sortie est fait par usage de l'attribut "MediaType".

#### 1- Liens avec attribut MediaType = HTML

**Mode d'affichage:** l'attribut Layout permet de définir si le lien s'affichera dans une fenêtre print, dans la même fenêtre, ou dans une fenêtre externe.

**Mode de sortie:** FormPublisher dispose de trois types de modes de sortie (OutputMode).

L'attribut OutputMode permet de choisir le mode de sortie de la fenêtre. Le tableau ci-dessous liste les layouts disponibles pour chaque OutputMode.

OutputMode = static	OutputMode = interview	OutputMode = report
1. moreInfo	1. normal	1. external
2. external	2. external	2. print
3. print	3. simpleSubmit	
4. keyword		

a) Exemple OutputMode = interview

- Lien vers ["formulaire"](#) valeur de "Layout" est normal
- Lien vers une fenêtre [externe](#) valeur de "Layout" est external
- Lien vers une fenêtre de type [simpleSubmit](#)

b) Exemple OutputMode = static

- Lien vers une fenêtre [moreInfo en html](#)
- Lien vers une fenêtre de type [print en html](#)
- Lien vers une fenêtre de type [external en html](#)
- Lien vers une fenêtre [keyword en html](#)

2- Liens avec attribut MediaType = PDF

OutputMode = report	OutputMode = static
1. normal	1. moreInfo
2. external	2. external
3. simpleSubmit	3. print
	4. keyword

a) Exemple OutputMode = report

- Lien vers une fenêtre [normale en pdf](#)
- Lien vers une fenêtre [external en pdf](#)
- Lien vers une fenêtre [simpleSubmit en pdf](#)

b) Exemple OutputMode = static

- Lien vers une fenêtre de type [moreInfo en pdf](#)
- Lien vers une fenêtre [external en pdf](#)
- Lien vers une fenêtre [print en pdf](#)
- Lien vers une fenêtre [keyword en pdf](#)

3- Liens avec attribut MediaType = word

OutputMode = report
1. letter
2. report

a) Exemple OutputMode = report

- Lien vers une fenêtre [letter en word](#)
- Lien vers une fenêtre [report en word](#)

b) Exemple OutputMode = static

- Lien vers une fenêtre de type [letter en word](#)
- Lien vers une fenêtre [report en word](#)

## 6. Notes

FormPublisher permet de faire les notes de bas de page ça c'est connu. Il propose aussi la possibilité de faire référence aux notes de bas de page déjà définies.

L'élément "Note" a un attribut "Id". Définir une note devant être plus tard référencée consiste à attribuer à son attribut "Id" une valeur.

Référencer une note consiste à insérer un élément "NoteReference" auquel est attribué à son attribut "NotelId" la valeur de l'attribut "Id" de la note à référencée.

Les exemples suivants illustrent les deux cas d'utilisation.

Exemple d'une note de bas de page (popup en html):<sup>(1)</sup> ("noteJway" est posée comme valeur de l'attribut "Id" de cette Note)

(1) Jway 2009

Exemple d'une référence vers une note de bas de page:<sup>(1)</sup> ("noteJway" est posée comme valeur de l'attribut "Notelid" de cette Note)

## 7. Objets multimédia

### Exemple d'utilisation :

Ci- dessous, nous avons une présentation en flash. Pour ajouter cette présentation dans notre document, un élément "Object" a été utilisé.

Cette [image](#) montre comment l'objet a été défini.

---