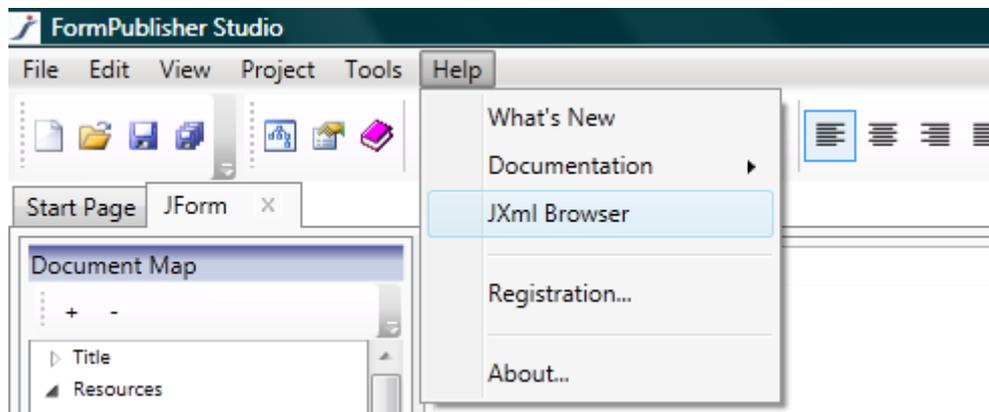




Éléments et attributs JForm

1. Introduction

Pour visualiser aisément les éléments JForm et leurs imbrications, utilisez le Jxml Browser :



Les éléments [JForm](#), [IncludeDocument](#) et [Menu](#) sont les socles de construction de documents FormPublisher Jxml. Ce sont les éléments racine au sens XML.

Sur les socles [JForm](#) et [IncludeDocument](#), les éléments [Section](#), [Content](#), [List](#), [Table](#), [QuestionSet](#) ainsi que [Paragraph](#) forment la charpente d'un document.

Les [Resources](#) d'un document contiennent des [métadonnées](#) : [MetaField](#) et [MetaContent](#).

Afin de rendre les interfaces visuelles et les documents plus agréables aux yeux des utilisateurs, il est possible d'agrémenter les contenus bruts avec [Image](#), [Description](#), [Style](#), [Title](#), [Subtitle](#), [ShortTitle](#), [Caption](#), [Note](#), etc.

Les éléments [Question](#) (avec [TextBox](#), [ListBox](#), [ComboBox](#), [PasswordBox](#), [MemoBox](#), [CheckBox](#), [RadioBox](#), [AttachmentBox](#), [UploadBox](#)) sont le pivot des interviews avec [Label](#), [Suffix](#), [Option](#) et [OptionData](#).

Par [WebService](#) les applications générées s'intègrent avec d'autres systèmes.

Jxml Browser

Elements

- JForm
- IncludeDocument
- Menu
- Alias
- Caption
- CheckBox
- ColumnDefinition
- ColumnDefinitions
- ComboBox
- Content
- Control
- Customizer
- Data
- DataIterator
- Description
- Hyperlink
- Image
- Include
- Index
- IteratorControlButton
- Label
- List

Diagram

JForm: Possible Parents

Children of:

- JForm
 - Title
 - Subtitle
 - ShortTitle
 - Text
 - Resources
 - Content

ShortTitle (#PCDATA)

JForm ShortTitle

2. Questions

Ces éléments JForm sont les fondations des interviews.

2.1. AttachmentBox

2.1.1. Description

AttachmentBox est un champ de saisie à placer dans une [Question](#) pour pointer une pièce jointe annexée au document.

2.1.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.2. CheckBox

2.2.1. Description

CheckBox est un champ de saisie de type "case à cocher" à placer dans une [Question](#).

2.2.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **DefaultValue** contient un nom de [Variable](#) dont la valeur initialise la saisie. Par défaut, les [champs](#) de saisie sont vides.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.3. ComboBox

2.3.1. Description

ComboBox est un champ de saisie de type "liste à déplier" à placer dans une [Question](#).

2.3.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **DefaultValue** contient un nom de [Variable](#) dont la valeur initialise la saisie. Par défaut, les [champs](#) de saisie sont vides.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.4. Control

2.4.1. Description

Control permet d'effectuer des tests de contrôle de saisie (prix positif, adresse email valide, etc). Il se place sur les **champs** de saisie [TextBox](#), [PasswordBox](#), [MemoBox](#), [UploadBox](#) dans les [Question](#).

2.4.2. Attributs

Type est le contrôle de saisie à appliquer, à choisir dans la liste des contrôles disponibles.

ErrorType prend la valeur "error" ou la valeur "warning" :

- error : si le contrôle de saisie est en échec, le champ contrôlé est en erreur.
- warning : si le contrôle de saisie est en échec, le champ contrôlé est en avertissement.

2.5. Label

Toute [Question](#) peut commencer par un Label avant les **champs** de saisie.

2.6. ListBox

2.6.1. Description

ListBox est un champ de saisie de type "liste proposant plusieurs réponses" à placer dans une [Question](#).

2.6.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **DefaultValue** contient un nom de [Variable](#) dont la valeur initialise la saisie. Par défaut, les **champs** de saisie sont vides.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.7. MemoBox

2.7.1. Description

MemoBox est un champ de saisie de type "espace multiligne à remplir en prose" à placer dans une [Question](#).

2.7.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **DefaultValue** contient un nom de [Variable](#) dont la valeur initialise la saisie. Par défaut, les **champs** de saisie sont vides.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.8. Option

2.8.1. Description

Le contenu de l'élément Option est affiché à l'utilisateur (via [ComboBox](#), [ListBox](#) ou [RadioBox](#)) pour faire un choix parmi plusieurs options.

2.8.2. Attributs

L'attribut Value renferme le codage technique de l'Option retenue.

Par exemple, 3 Option contiennent "Madame", "Mademoiselle" et "Monsieur" tandis que leur 3 Value contiennent 11, 22 et 33 (codage numérique plus facilement traité par le système aval et indépendant de la langue).

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

2.9. OptionData

L'élément OptionData produit dynamiquement une liste d'[Option](#).

2.10. PasswordBox

2.10.1. Description

PasswordBox est un champ de saisie de type "mot de passe" (saisie remplacée à l'écran pour protection contre les regards indiscrets) à placer dans une [Question](#).

2.10.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.11. Question

2.11.1. Description

Question est le pivot des documents interactifs (interviews web). C'est dans Question que sont placés les [champs](#) de saisie ([TextBox](#), [ListBox](#), [ComboBox](#), [PasswordBox](#), [MemoBox](#), [CheckBox](#), [RadioBox](#), [AttachmentBox](#), [UploadBox](#)) qui captent les informations dynamiques des [SmartForm](#).

Question se place en groupe dans [QuestionSet](#) ou à part dans [Paragraph](#).

Les [données](#) reçues portent le nom de l'attribut "Name" des "box". Celui-ci doit respecter des règles de syntaxe précises :

1. le premier [caractère](#) peut être : "_", "a-z", "A-Z", (accents non autorisés !)
2. les caractères suivant peuvent être des chiffres, des lettres (accents non autorisés !)
3. le caractère [(crochet ouvrant) est autorisé à condition qu'il soit suivi de chiffres ou d'une expression \$(NomVar), puis de]
4. le caractère] (crochet fermant) est autorisé à condition de suivre la règle 3, que ce soit le dernier caractère de la chaîne ou qu'il soit suivi du caractère | en respectant la règle 5
5. le caractère | est autorisé à condition que le terme précédent et le terme suivant suivent les règles 1, 2, 3 et 4

2.11.2. Attributs

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

2.12. QuestionSet

2.12.1. Description

QuestionSet est un conteneur pour présenter des [Question](#) groupées sous forme de tableau.
Un QuestionSet peut contenir un Title.

2.12.2. Attributs

L'attribut **OutputMode** détermine la visibilité de l'élément selon le mode d'appel du document : "interview" (élément présent seulement dans les [formulaire](#)s interactifs avec l'utilisateur, absent des rapports), "report" (élément présent seulement dans les documents produits à la volée, pas dans les interviews) ou "all" (toujours présent). Il vaut "all" par défaut.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

2.13. RadioBox

2.13.1. Description

RadioBox est un champ de saisie de type "bouton radio à cocher" à placer dans une [Question](#).

2.13.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **DefaultValue** contient un nom de [Variable](#) dont la valeur initialise la saisie. Par défaut, les [champs](#) de saisie sont vides.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.14. Suffix

Le Suffix sert à placer du texte après les [champs](#) de saisie dans une [Question](#).

2.15. SignatureBox

2.15.1. Description

SignatureBox est un champ de saisie à placer dans une [Question](#) pour insérer une signature.

2.15.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

Description sert à décrire la signature attendue.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.16. TextBox

2.16.1. Description

TextBox est un champ de saisie ordinaire à placer dans une [Question](#).

2.16.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **DefaultValue** contient un nom de [Variable](#) dont la valeur initialise la saisie. Par défaut, les [champs](#) de saisie sont vides.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

2.17. UploadBox

2.17.1. Description

UploadBox est un champ de saisie à placer dans une [Question](#) pour joindre une image légère à afficher durant l'interview. Pour les pièces jointes annexées au [formulaire](#), voir [AttachmentBox](#).

2.17.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut **RefreshOnExit** contient une [expression](#) qui détermine si l'écran en cours doit être rafraîchi au moindre passage dans le champ de saisie. Par défaut, seule la navigation ordinaire (map, boutons, etc) provoque des rafraîchissements d'écran.

L'attribut **IsRequired** contient une [expression](#) qui détermine si la saisie est obligatoire ou non. Par défaut, la saisie est facultative.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

3. Organisation du document

Ces éléments JForm structurent le document.

3.1. JForm

3.1.1. Description

JForm est le socle principal de document, pour les interviews comme pour les rapports.

JForm doit être posé à la racine d'un document. Il ne peut y avoir qu'un unique JForm par document (racine [XML](#)).

3.1.2. Attributs

| Attribut | Type | Description |
|------------|--------|--|
| Nature | String | form (formulaire interactif) ou classic (document statique) |
| Language | String | code ISO 2 lettres : fr, en, lu, etc. |
| Version | String | |
| DocumentId | String | Identifiant de document lié au nom du fichier xml |

Le Studio remplit automatiquement l'attribut DocumentId pour garantir son unicité dans le projet. Cet [identifiant](#) sert à pointer un document avec l'élément [Hyperlink](#).

3.2. IncludeDocument

3.2.1. Description

IncludeDocument est le socle de réutilisation : il sert à construire des sous-documents destinés à être incorporés à d'autres documents via [Include](#). Grâce à IncludeDocument, il est possible de constituer des bibliothèques de composants réutilisables ("shared includes folder").

IncludeDocument doit être posé à la racine d'un document. Il ne peut y avoir qu'un unique IncludeDocument par document (racine [XML](#)).

3.2.2. Attributs

| Attribut | Type | Description |
|------------|--------|--|
| Language | String | code ISO 2 lettres : fr, en, lu, etc. |
| Version | String | |
| DocumentId | String | Identifiant de document lié au nom du fichier jxml |

Le Studio remplit automatiquement l'attribut DocumentId pour garantir son unicité dans le projet. Cet [identifiant](#) sert à pointer l'IncludeDocument avec l'élément [Include](#).

3.2.3. Exemple

Voir comment partager et [réutiliser des parties récurrentes](#) de documents.

3.3. Menu

3.3.1. Description

[Menu](#) offre une possibilité d'aiguillage visuel dans une hiérarchie de documents. Chaque entrée de [Menuitem](#) comporte un [Hyperlink](#).

Menu doit être posé à la racine d'un document. Il ne peut y avoir qu'un unique Menu par document (racine [XML](#)).

3.3.2. Attributs

| Attribut | Type | Description |
|------------|--------|--|
| Nature | String | menu |
| Language | String | code ISO 2 lettres : fr, en, lu, etc. |
| Version | String | |
| DocumentId | String | Identifiant de document lié au nom du fichier jxml |

Le Studio remplit automatiquement l'attribut DocumentId pour garantir son unicité dans le projet.

3.4. Menuitem

Menuitem sert à la constitution hiérarchique de Menu. Il contient un [Title](#), un [Hyperlink](#) ou d'autres [Menuitem](#).

3.5. MetaContent

3.5.1. Description

Un MetaContent est similaire à un [Content](#) ordinaire.

3.5.2. Attributs

L'attribut Type définit son type et prend ses valeurs parmi : contact, legal, presentation, validation.

3.6. MetaField

L'attribut **Name** est l'[identifiant](#) de l'élément.

3.7. MetaFields

MetaFields regroupe plusieurs [MetaField](#) sous un intitulé de classe à des fins de rangement.

L'attribut **Name** est l'[identifiant](#) de l'élément.

3.8. Content

3.8.1. Description

L'élément Content est le conteneur standard pouvant grouper en séquence plusieurs sous éléments (par exemple : des [Paragraph](#), des [List](#), des tableaux [Table](#)).

3.8.2. Attributs

L'attribut **OutputMode** détermine la visibilité de l'élément selon le mode d'appel du document : "interview" (élément présent seulement dans les [formulaires](#) interactifs avec l'utilisateur, absent des rapports), "report" (élément présent seulement dans les documents produits à la volée, pas dans les interviews) ou "all" (toujours présent). Il vaut "all" par défaut.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

3.9. Customizer

3.9.1. Description

Un Customizer transforme à la volée le contenu d'un [IncludeDocument](#) lors de son appel par un [Include](#).

3.9.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément.

L'attribut (textuel) Value sera substitué à chaque occurrence de l'attribut Name dans le document appelé.

3.9.3. Exemple

Voir comment partager et [réutiliser des parties récurrentes](#) de documents.

3.10. Include

3.10.1. Description

C'est par Include qu'est incorporé dans le flux courant un [IncludeDocument](#). Lors du processus, des adaptations peuvent avoir lieu via des [Customizer](#).

3.10.2. Attributs

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

3.10.3. Exemple

Voir comment partager et [réutiliser des parties récurrentes](#) de documents.

3.11. Resources

Les [Resources](#) d'un document contiennent des [métadonnées](#) : [MetaField](#) et [MetaContent](#). Soit directement, soit via des [Include](#).

3.12. Section

3.12.1. Description

L'élément Section sert à découper un document ou une Section en parties plus petites. Il constitue le granule équivalent à une page pour les interviews en [HTML](#) (étapes de saisie) et à un niveau hiérarchique (sous chapitre, numérotation) en [PDF](#). A l'intérieur d'une Section (ou sous section), les éléments sont groupés par [Content](#).

3.12.2. Attributs

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

L'attribut **OutputMode** détermine la visibilité de l'élément selon le mode d'appel du document : "interview" (élément présent seulement dans les [formulaires](#) interactifs avec l'utilisateur, absent des rapports), "report" (élément présent seulement dans les documents produits à la volée, pas dans les interviews) ou "all" (toujours présent). Il vaut "all" par défaut.

4. Traitement de texte

Ces éléments JForm portent le texte mis en forme dans les documents.

4.1. Alias

L'élément Alias est utilisé pour les glossaires.

L'attribut ParagraphId...

4.2. Caption

L'élément Caption sert à légender les tableaux [Table](#).

4.3. ColumnDefinition

ColumnDefinition permet de spécifier les caractéristiques d'une colonne d'un tableau [Table](#) dans son [ColumnDefinitions](#).

4.4. ColumnDefinitions

ColumnDefinitions contient un [ColumnDefinition](#) pour chaque colonne du tableau [Table](#).

4.5. Description

L'élément description sert à fournir une explication brève, à la manière d'une bulle d'aide.

4.6. Hyperlink

4.6.1. Description

L'élément Hyperlink sert à placer des hyperliens dans les documents.

4.6.2. Attributs

L'attribut DocumentId sert à pointer le document cible. Les attributs SectionId et ParagraphId peuvent permettre de pointer plus précisément à l'intérieur d'un document.

L'attribut MediaType détermine le format de publication de la cible : html (par défaut), pdf ou word.

L'attribut Layout permet de sélectionner un gabarit de restitution ([template](#)) alternatif.

L'attribut Url s'utilise seul pour pointer un lien web en dehors de la publication.

L'attribut OutputMode vaut "static" par défaut (lien vers un document figé). Il peut prendre les valeurs dynamiques "report" (production de document à la volée) ou "interview" ([formulaire](#) interactif avec l'utilisateur).

4.7. Image

4.7.1. Description

Une Image est un élément graphique comme un dessin, une capture d'écran, une photographie, un diagramme, un pictogramme, etc.

4.7.2. Attributs

Une image externe est pointée par l'attribut Url tandis qu'une image interne à la publication est pointée par l'attribut EmbeddedResource.

4.8. Index

Index comporte au moins un [Alias](#).

4.9. List

4.9.1. Description

L'élément List est un conteneur pour afficher successivement plusieurs choses en liste (énumérations). Ces séquences ordonnées de [ListItem](#) sont éventuellement numérotées.

Avant les [ListItem](#), une List peut afficher un [Title](#).

4.9.2. Attributs

MarkerStyle détermine l'aspect de la liste (puces, numérotation).

L'attribut **OutputMode** détermine la visibilité de l'élément selon le mode d'appel du document : "interview" (élément présent seulement dans les [formulaire](#)s interactifs avec l'utilisateur, absent des rapports), "report" (élément présent seulement dans les documents produits à la volée, pas dans les interviews) ou "all" (toujours présent). Il vaut "all" par défaut.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

4.10. ListItem

4.10.1. Description

ListItem est l'élément d'une énumération [List](#). Il ne se pose que dans [List](#) et nulle part ailleurs. Ses seuls attributs sont IsReadOnly et IsVisible.

4.10.2. Attributs

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

4.11. Note

4.11.1. Description

Une Note permet de placer du contenu court (précisions, aide, etc) en dehors du flux principal affiché. Par exemple en popup dans une vue web ([HTML](#)) et en bas de page dans une vue papier ([PDF](#)). Une Note doit rester concise, quelques petits [Paragraph](#) simples au maximum, voire un seul.

4.11.2. Attributs

L'attribut Id identifie les Note d'un document. Ces Id sont pointés par [NoteReference](#).

4.11.3. Remarque concernant FormPublisher v1.0.0 à v2.0.0

Suite à une [limitation du processeur FO->PDF Apache FOp](#) embarqué par FormPublisher (avant la v2.0.1), les Note n'étaient pas supportées sur les exports [PDF](#) dans les tableaux [Table](#), les listes [List](#) et les [QuestionSet](#). Pour que les Note apparaissent correctement dans les exports [PDF](#), il est possible :

- d'utiliser FormPublisher >=2.0.1 (qui embarque un FOp corrigé plus récent),
- de placer les Notes en dehors des Table, List, QuestionSet,
- de changer le processeur [FO->PDF](#) embarqué appelé,
- d'externaliser la production [PDF](#) à un service tiers.

4.12. NoteReference

L'élément NoteReference est vide (pas de sous-élément). Son attribut Noteld référence une [Note](#).

4.13. Object

Object est **réservé** à une utilisation avancée de FormPublisher, en collaboration étroite avec J-Way.

4.14. Paragraph

4.14.1. Description

L'élément Paragraph sert à contenir la masse principale de [texte](#) ("hors labels" de `i18nlabels.xml`) des documents et les [Question](#) hors [QuestionSet](#). La forme des textes contenus est modulée par les [Style](#) et autres [Image](#). Les textes peuvent offrir des [Hyperlink](#).

4.14.2. Attributs

L'attribut **OutputMode** détermine la visibilité de l'élément selon le mode d'appel du document : "interview" (élément présent seulement dans les [formulaire](#)s interactifs avec l'utilisateur, absent des rapports), "report" (élément présent seulement dans les documents produits à la volée, pas dans les interviews) ou "all" (toujours présent). Il vaut "all" par défaut.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

4.15. Parameter

L'attribut **Name** est l'[identifiant](#) de l'élément.

4.16. ShortTitle

Les ShortTitle est un substitut, sans mise en forme possible, au [Title](#).

4.17. Style

4.17.1. Description

Style sert à la mise en forme des textes (italique, gras, souligné, etc).

4.17.2. Attributs

L'attribut Type détermine le style parmi : bold, emphasis, exponent, indice, italic, larger, note, red, smaller, strikethrough, underline.

4.18. Subtitle

L'élément Subtitle peut être adjoint au [Title](#) d'un document.

4.19. Table

4.19.1. Description

Table définit des tableaux, à base d'un [TableHeader](#) et d'au moins un [TableBody](#).

4.19.2. Attributs

L'attribut **OutputMode** détermine la visibilité de l'élément selon le mode d'appel du document : "interview" (élément présent seulement dans les [formulaire](#)s interactifs avec l'utilisateur, absent des rapports), "report" (élément présent seulement dans les documents produits à la volée, pas dans les interviews) ou "all" (toujours présent). Il vaut "all" par défaut.

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

4.20. TableBody

4.20.1. Description

Un tableau [Table](#) peut contenir plusieurs sections TableBody. TableBody contient des "lignes" [Tr](#).

4.20.2. Attributs

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

4.21. TableHeader

Un tableau [Table](#) doit contenir un TableHeader pour légender ses colonnes par une ligne de titres.

4.22. Td

Td pose une cellule de [Table](#) (une colonne dans une ligne [Tr](#)).

4.23. Th

Th pose une cellule d'entête de colonne de [Table](#) (une colonne dans une ligne [Tr](#))

4.24. Title

L'élément Title donne le titre du document, d'un [QuestionSet](#) ou d'une [List](#).

4.25. Tr

4.25.1. Description

Tr pose une ligne de [Table](#) dans un [TableBody](#).

4.25.2. Attributs

L'attribut **IsReadOnly** contient une [expression](#) qui détermine si l'élément peut être modifié ou non. Par défaut l'élément est modifiable.

L'attribut **IsVisible** contient une [expression](#) qui détermine si l'élément doit être affiché ou non. Par défaut l'élément est affiché.

4.26. Text

Text n'est pas un véritable élément [XML](#), il est présenté ici comme un pseudo-élément ; il contient simplement le texte qui sera affiché (prose de caractères). Il se trouve dans les [Paragraph](#) mais aussi dans presque tous les [éléments JForm](#) : [Label](#), [Suffix](#), [Title](#), [Hyperlink](#), [Caption](#), etc.

5. Utilisation des données

Ces éléments JForm traitent les [données](#) des [formulaires](#).

5.1. Data

Data évalue une [expression](#) pour en afficher le résultat. L'élément Data est vide. Son seul attribut est Expression.

Le plus simplement, cette expression n'est qu'un appel à une [information](#) contenue dans le datastore du [formulaire](#) en cours de saisie.

Pour évaluer une [expression](#) sans l'afficher, utiliser [Variable](#).

5.2. Datalterator

5.2.1. Description

Le Datalterator est le mécanisme puissant apportant aux [SmartForms](#) la flexibilité d'itérer des éléments dont le nombre n'est pas connu à l'avance. Par exemple un groupe de personnes (saisie d'une fiche par personne) ou un ensemble d'activités (saisie d'une fiche d'inscription pour chaque activité) ou encore des lignes de commande ([affichage](#) d'un récapitulatif pour chaque article dans un bon de commande).

Le Datalterator peut être posé sur [Section](#), [Content](#) et [TableBody](#).

L'élément Datalterator est vide (pas de sous-élément).

5.2.2. Attributs

L'attribut **Datapath** fixe le nom de l'ensemble d'enregistrements.

L'attribut **InternalName** ("item" par défaut) sert de préfixe pointant l'enregistrement courant dans l'itération locale.

L'attribut **Counter** définit un nom de variable de zéro à N-1.

L'attribut **MinItemToPrint** fixe le nombre minimum d'éléments à imprimer.

L'attribut **MinItemToDisplay** fixe le nombre minimum d'éléments à afficher.

L'attribut **MinItemAllowed** fixe le nombre minimum d'éléments requis.

L'attribut **MaxItemAllowed** fixe le nombre maximum d'éléments permis.

5.3. IteratorControlButton

Dans le cadre d'usage d'un [DataIterator](#), un **IteratorControlButton** permet d'ajouter/supprimer un élément dans le groupe de taille variable.

5.4. OnErrorAction

L'élément **OnErrorAction** permet de définir un traitement d'anomalie.

5.5. OnErrorMessage

L'élément **OnErrorMessage** définit un message de traitement d'anomalie.

5.6. Variable

5.6.1. Description

Comme une [Data](#), une **Variable** provoque l'évaluation d'une [expression](#) via l'attribut **Expression**. La valeur obtenue reste accessible via l'attribut **Name** dans d'autres [expressions](#).

A la différence de [Data](#), **Variable** n'affiche rien.

5.6.2. Attributs

L'attribut **Name** est l'[identifiant](#) de l'élément. Il doit respecter les règles de syntaxes suivantes :

1. le premier [caractère](#) peut être : "_", "a-z", "A-Z", (accents non autorisés !)
2. les caractères suivant peuvent être des chiffres, des lettres (accents non autorisés !)
3. le caractère [(crochet ouvrant) est autorisé à condition qu'il soit suivi de chiffres ou d'une expression \$(NomVar), puis de]
4. le caractère] (crochet fermant) est autorisé à condition de suivre la règle 3, que ce soit le dernier caractère de la chaîne ou qu'il soit suivi du caractère | en respectant la règle 5
5. le caractère | est autorisé à condition que le terme précédent et le terme suivant suivent les règles 1, 2, 3 et 4

L'attribut **DataType** détermine le type de [donnée](#) : nombre, caractères...

L'attribut **Expression** contient l'[expression](#) à évaluer.

L'attribut **IsEnabled** contient une [expression](#) qui détermine si l'élément est activé ou non. Par défaut l'élément est activé.

L'attribut **Submit** permet d'[enregistrer](#) la **Variable** dans le datastore :

- **Submit true** : la **Variable** apparaît dans le datastore comme un champ de [données](#) ordinaire.
- **Submit false** : la **Variable** reste volatile et n'est pas stockée dans le datastore.

5.7. WebService

5.7.1. Description

Par **WebService**, l'application générée peut faire appel à un service automatique externe publiant une interface WSDL, lui envoyer des paramètres ([InputMapping](#)) et recevoir les résultats ([OutputMapping](#)).

5.7.2. Attributs

Id est réservé pour usage interne.

WsdL pointe l'offre du fournisseur du **WebService** exprimée au format standard WSDL, **WebService Definition Language**.

Service, **Port**, **Operation** : sélectionnent l'action parmi l'offre du **WebService**.

StoreOutputInDatastore : true ou false, true par défaut. Si cette valeur est à true les [données](#) extraites de la réponse du webservice seront intégrées dans le datastore draftData. Si cette valeur est à false, ces données ne seront exploitées que dans l'[affichage](#) de la page contenant l'appel du webservice.

CallOnInputChangeOnly : true ou false, true par défaut. Si cette valeur est à true, l'appel du webservice n'aura lieu que si les données servant à appeler le webservice ont changé depuis le dernier appel. Si cette valeur est false, l'appel aura toujours lieu.

5.7.3. Exemples

Comment [appeler un Webservice simple](#) depuis une application FormPublisher.

Comment appeler un Webservice à réponse itérative depuis une application FormPublisher.

5.8. Instructions

JForm étant un dialecte [XML](#), des "Processing Instructions" (PI) peuvent être placées dans les documents. Les auteurs peuvent poser 4 PI particulières directement avec le Studio ([If](#), [IfExist](#), [Loop](#) et [Foreach](#)) et d'autres PI ([ProcessingInstruction](#)).

5.9. If

If est la [PI](#) pour poser une condition d'exécution : If [expression](#) CEI (sinon CELA).

5.10. IfExist

IfExist est la [PI](#) pour poser une condition d'exécution sur l'existence d'une [Variable](#) technique.

5.11. Loop

Loop est la [PI](#) pour répéter une portion de document jusqu'à satisfaction d'une condition.

5.12. Foreach

Foreach est la [PI](#) pour itérer un traitement sur un ensemble, c'est à dire pour opérer sur chacun de ses éléments.

5.13. ProcessingInstruction

ProcessingInstruction permet de poser une [PI](#) autre que [If](#), [IfExist](#), [Loop](#) et [Foreach](#).

5.14. Button

Dans le cadre d'usage d'un [Paragraph](#), un Button permet d'assurer un rafraîchissement de la page courante sur demande.
