













4D Write Pro

-  Présentation
-  Définir une zone 4D Write Pro
-  Stocker les documents 4D Write Pro dans des champs objet 4D
-  Utiliser une zone 4D Write Pro
-  Imprimer des documents 4D Write Pro
-  Filtrer les expressions contenues dans un document 4D Write Pro
-  Import de documents 4D Write
-  Utiliser les actions standard 4D Write Pro
-  Manipuler des images
-  4D Write Pro - Langage
-  Liste des nouveautés
-  Liste alphabétique des commandes

Présentation

Vue d'ensemble

4D Write Pro offre aux utilisateurs 4D un outil de traitement de texte avancé, entièrement intégré aux bases de données 4D. Avec 4D Write Pro, vous pouvez écrire des emails préformatés ou des lettres personnalisées contenant des images, du texte formaté et des références de variables. Vous pouvez également créer dynamiquement des factures ou des états contenant du texte formaté et des images.

Les fonctions majeures de 4D Write Pro sont :

- Compatibilité 4D Write : un objet 4D Write Pro peut convertir et ouvrir des anciens documents 4D Write en conservant la quasi-totalité de leurs propriétés spécifiques.
- Traitement de texte : un objet 4D Write Pro inclus dans un formulaire donne accès à toutes les fonctions standard de traitement de texte, c'est-à-dire la manipulation de paragraphes et de styles, l'insertion d'images, l'import et l'export de texte, et bien plus encore.
- Intégration dans la base de données :
 - un objet 4D Write Pro peut contenir des parties variables qui seront automatiquement remplies par des données provenant de la base ou calculées par 4D.
 - les documents 4D Write Pro peuvent être stockés dans les champs de la base de données ou sur disque.

Installation et activation

A la différence de 4D Write, 4D Write Pro n'est pas un plug-in mais est intégré à 4D lui-même, ce qui le rend plus facile à gérer et à déployer. Aucune installation supplémentaire n'est requise, vous pouvez directement ajouter des zones 4D Write Pro dans vos formulaires et gérer les variables 4D Write Pro dans vos applications 4D.

Notez cependant que 4D Write Pro utilise la même licence que 4D Write. Cette licence doit être installée dans votre application pour que la fonctionnalité soit activée.

Compatibilité : Sous Windows, les fonctionnalités de 4D Write Pro s'appuient sur Direct2D. Avec des machines sous Windows 7 ou Windows Server 2008, assurez-vous que le composant *Platform Update for Windows* a bien été installé, afin de bénéficier de la version requise de Direct2D.

A propos de ce manuel

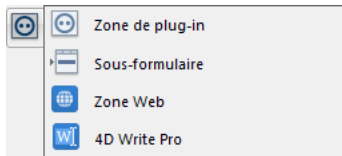
Ce manuel est le *Guide de référence de 4D Write Pro*. Il décrit toutes les fonctionnalités de 4D Write Pro, y compris l'interface utilisateur et les commandes de langage.

Notez que les objets 4D Write Pro peuvent être gérés par des commandes de langage dédiées (cf. "**4D Write Pro - Langage**") ainsi que des commandes d'autres thèmes ("**Objets (Formulaires)**" et "**Texte multistyle**"), documentés dans le manuel *Langage de 4D*.

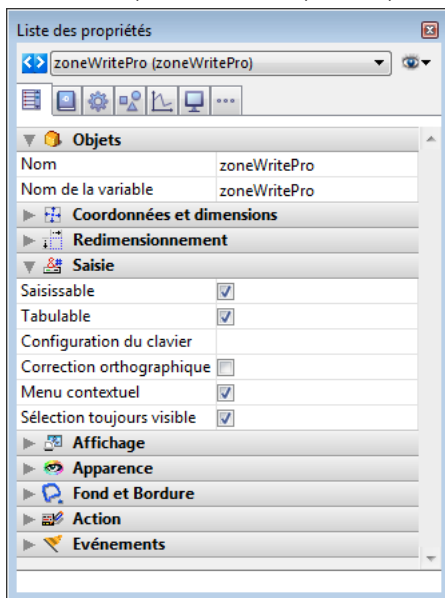
■ Définir une zone 4D Write Pro

Création de la zone

Dans 4D, les documents 4D Write Pro sont affichés et modifiés manuellement dans un objet de formulaire appelé **4D Write Pro**. Cet objet est sélectionnable dans le dernier groupe d'outils (zone de plug-in, zone Web, etc.) de la barre d'objets de l'éditeur de formulaires :



Une zone de formulaire 4D Write Pro est configurée via les options standard de la Liste des propriétés, telles que **nom** et **nom de variable**, **coordonnées**, **saisie**, **affichage**, **apparence** et/ou **événements**.



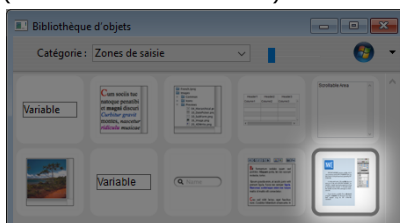
La propriété **Nom de variable** peut être utilisée dans le langage comme référence de la zone 4D Write Pro. Notez bien que cette variable doit être du type *objet* (pour plus d'informations, reportez-vous à la commande **C_OBJET**).

Les propriétés du thème "Saisie" contrôlent les fonctions élémentaires de saisie de texte :

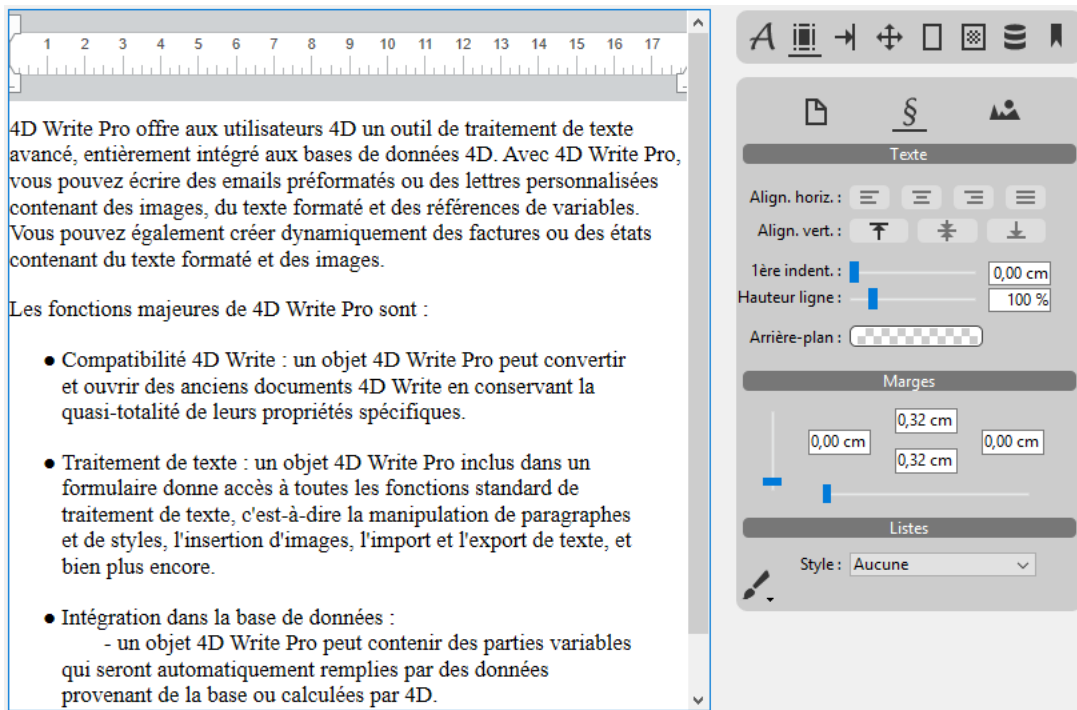
- **Saisissable** : vous permet de verrouiller/déverrouiller la zone afin d'autoriser ou d'interdire la saisie.
- **Correction orthographique** : disponible pour les zones 4D Write Pro
- **Menu contextuel** : vous permet d'activer/désactiver le menu contextuel en mode Application (voir la section **Utiliser une zone 4D Write Pro**)
- **Sélection toujours visible** : contrôle la sélection de texte comme dans les zones de texte standard.

Utiliser le Widget 4D Write Pro de la Bibliothèque d'objets

Il est possible de créer directement une zone 4D Write Pro préconfigurée via l'objet **4D Write Pro** de la Bibliothèque d'objets (thème "Zones de saisie") :



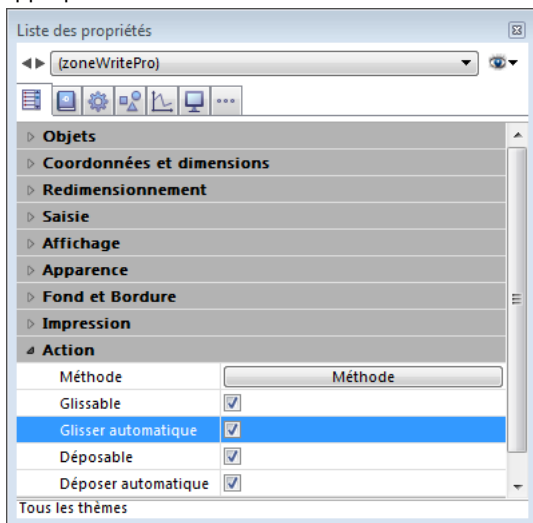
La zone ainsi créée est automatiquement dotée d'un panneau de contrôle personnalisable permettant de gérer tous les attributs de la zone (polices, couleurs, styles...) :



Pour plus d'informations, reportez-vous à la section **Zone 4D Write Pro**.

Configuration du glisser-déposer

Pour configurer les fonctionnalités de glisser-déposer dans vos zones 4D Write Pro, vous devez sélectionner les options appropriées dans le thème "Action" de la Liste des propriétés :



Les zones 4D Write Pro prennent en charge deux modes de glisser-déposer :

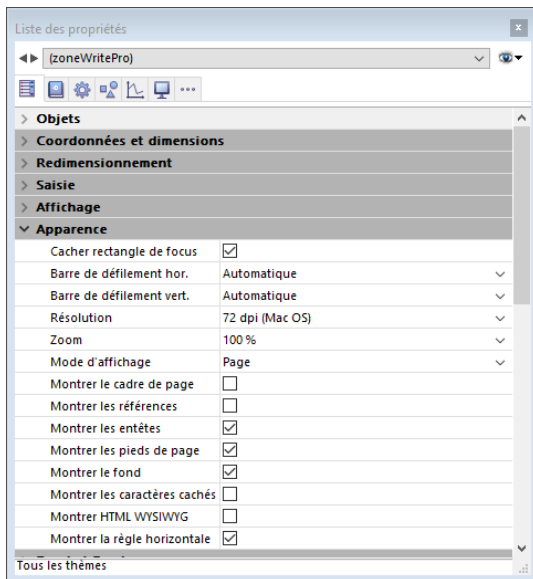
- **Mode personnalisé** : lorsque seules les options "Glissable" et "Déposable" sont cochées. Dans ce mode, vous pouvez sélectionner du texte et commencer à le déplacer. La méthode objet est alors appelée avec l'événement `Sur début glisser`, et vous pouvez alors définir l'action de déposer via du code personnalisé.
- **Mode automatique** : lorsque toutes les options sont cochées ("Glissable", "Déposable", "Glisser automatique" et "Déposer automatique"). Dans ce mode, Vous pouvez automatiquement déplacer ou copier (en maintenant la touche **Alt/Option**) le texte sélectionné. L'événement `Sur début glisser` n'est pas généré.

Note : Sélectionner uniquement les options "Glisser automatique" et "Déposer automatique" n'aura pas d'effet sur la zone 4D Write Pro.

Configurer les propriétés d'affichage

Les propriétés de vue des documents 4D Write Pro sont accessibles dans la Liste des propriétés pour les zones 4D Write Pro afin de vous permettre de définir la façon dont un document sera visualisé par défaut dans cette zone. Ces propriétés définissent, par exemple, si les documents 4D Write Pro doivent être affichés en vue "impression" ou en vue "Web". Vous pouvez définir différentes vues pour un document 4D Write Pro dans le même formulaire.

Les propriétés de vue des documents sont gérées via des options spécifiques dans le thème **Apparence** de la Liste des propriétés pour les objets de formulaire 4D Write Pro :



- **Résolution** : Définit la résolution écran pour le contenu de la zone 4D Write Pro. Par défaut, elle est fixée à **72 dpi (Mac OS)**, qui est la résolution standard des formulaires 4D sur toutes les plates-formes. Définir une valeur de dpi fixe permet d'obtenir un rendu de document identique sur les plates-formes Mac OS et Windows. L'option Automatique adapte la résolution à la plate-forme courante, ce qui signifie que le rendu du document sera différent entre les plates-formes Mac OS et Windows.
- **Zoom** : Définit le pourcentage de zoom pour l'affichage du contenu de la zone 4D Write Pro area. Par défaut, la valeur est 100%.
- **Mode d'affichage** : Définit le type de vue à utiliser pour l'affichage du document 4D Write Pro dans la zone du formulaire. Trois modes sont disponibles :
 - **Page** (défaut) : vue la plus complète, incluant le cadre de page, l'orientation, les marges, les sauts de page, les entêtes et les pieds de page, etc. Pour plus d'informations, veuillez vous reporter au paragraphe **Fonctionnalités de l'affichage en page**.
 - **Bouillon** : mode brouillon avec des propriétés de document basiques.
 - **Inclus** : mode adapté aux zones incluses ; les marges, en-têtes, pieds de pages, cadre de page, etc. ne sont pas affichés. Ce mode permet également d'obtenir une vue "Web" (en l'associant à la résolution **96 dpi** et à l'option **Montrer HTML WYSIWYG**).

Note : La propriété **Mode d'affichage** est utilisée uniquement pour le rendu écran. Pour l'impression, des règles de rendu spécifiques sont automatiquement mises en oeuvre (cf. **Imprimer des documents 4D Write Pro**).
- **Montrer le cadre de page** : Affiche/masque les limites des pages lorsque le mode d'affichage est "Page". Par défaut, le cadre est masqué.
- **Montrer les références** : Affiche en tant que *références* toutes les expressions 4D insérées dans le document. Lorsque cette option est désélectionnée (défaut), les expressions 4D sont affichées en tant que *valeurs*. Lorsque vous insérez un champ ou une expression 4D, 4D Write Pro calcule et affiche sa valeur courante. Si vous souhaitez savoir quel champ ou quelle expression est utilisé(e), cochez cette option. Les références des champs ou des expressions sont alors affichées dans votre document avec un fond gris. Par exemple, vous avez inséré la date du jour avec un format, la date est affichée par défaut :
 29 juillet 2016
 Si vous cochez l'option **Montrer les références**, la référence est alors affichée :
 Chaîne(Date du jour;Interne date long)
Note : Les expressions 4D peuvent être insérées à l'aide de la commande **ST INSERER EXPRESSION**.
- **Montrer les entêtes / pieds de page** : Affiche/masque les en-têtes et les pieds de page lorsque le mode d'affichage est "Page" (affichés par défaut). Pour plus d'informations sur les en-têtes et les pieds de page, veuillez vous reporter à la section .
- **Montrer le fond** : Affiche/masque les images de fond ainsi que les couleurs de fond (affichées par défaut).
- **Montrer les caractères cachés** : Affiche/masque les caractères invisibles (cachés par défaut).
- **Montrer HTML WYSIWYG** : Active/désactive la vue WYSIWYG HTML, dans laquelle les attributs 4D Write Pro avancés qui ne sont pas compatibles avec tous les navigateurs sont supprimés (désactivé par défaut).
- **Montrer la règle horizontale** : Affiche/masque la règle dans le document (affichée par défaut). Pour plus d'informations sur les règles dans 4D Write Pro, reportez-vous à la section **Utiliser les règles**.

Notes de compatibilité :

- Pour les documents 4D Write Pro créés avec des versions antérieures à 4D v16, les valeurs par défaut de ces propriétés sont utilisées, à l'exception de la propriété **Résolution**, qui a pour valeur **Automatique** dans ce cas.
- La règle horizontale est affichée dans les bases de données créées à compter de 4D v16 R2 ; pour les bases converties depuis des versions précédentes, cette propriété n'est pas sélectionnée par défaut.

■ Stocker les documents 4D Write Pro dans des champs objet 4D

Vous pouvez stocker automatiquement vos documents 4D Write Pro dans le fichier de données de 4D. Si vous avez créé une zone 4D Write Pro dans un formulaire ainsi qu'un champ de type Objet pour stocker le contenu de la zone, le texte saisi dans la zone est automatiquement sauvegardé dans les données au moment de la validation de l'enregistrement. Vous pouvez alors utiliser la commande **CHERCHER PAR ATTRIBUT** afin de sélectionner des enregistrements en fonction de la valeur de leurs attributs internes. Vous pouvez également ajouter des attributs personnalisés aux zones 4D Write Pro et les utiliser pour des recherches.

Cette section décrit les fonctionnalités suivantes :

- Associer un champ objet 4D à une zone 4D Write Pro dans un formulaire.
- Fixer, lire et rechercher des attributs personnalisés dans les documents 4D Write Pro, à l'aide des commandes standard **OB FIXER**, **OB Lire** et **CHERCHER PAR ATTRIBUT**.

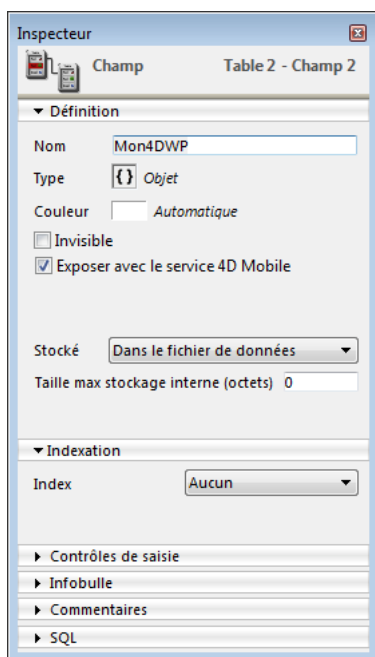
Associer un champ objet 4D à une zone 4D Write Pro

Pour associer une zone 4D Write Pro à un champ 4D de type Objet, il vous suffit de référencer le champ dans la propriété "Nom de variable" de la zone.

Créer le champ objet en structure

Dans la structure de votre base de données, tout champ objet 4D peut être utilisé pour stocker des documents 4D Write Pro. Vous devez simplement définir, en fonction de vos besoins, ses propriétés standard :

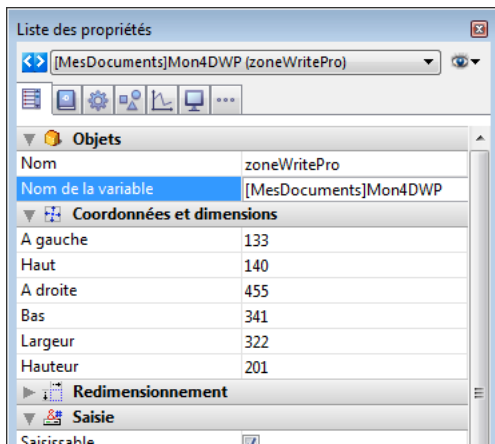
- le nom du champ,
- ses attributs, tels que "Exposer avec le service 4D Mobile" et son index,
- son option de stockage (voir à ce sujet le paragraphe **Stockage externe des données**).



Affecter le champ objet à la zone 4D Write Pro

Une fois que le champ objet destiné à stocker vos documents 4D Write Pro est défini, il vous suffit de le référencer dans le formulaire contenant la zone. Vous pouvez utiliser un formulaire table ou un formulaire projet.

Dans l'éditeur de formulaires, saisissez le nom du champ, en utilisant la notation standard "[Table]Champ", dans la zone **Nom de la variable** de la Liste des propriétés pour la zone 4D Write Pro :



Votre zone 4D Write Pro est alors associée au champ, ce qui vous donne l'assurance que son contenu sera automatiquement sauvegardé avec chaque enregistrement. A noter que si vous n'utilisez pas les boutons à action automatique de 4D, vous devrez programmer vous-même la sauvegarde de la zone, à l'aide des commandes 4D.

Utiliser des attributs personnalisés

Lorsque les zones 4D Write Pro sont stockées dans des champs de type Objet, vous pouvez écrire et lire des attributs personnalisés dans les documents 4D Write Pro, comme par exemple le nom de l'auteur, la catégorie du document, ou toute information supplémentaire qui vous serait utile. Vous pouvez effectuer des recherches parmi les attributs personnalisés afin de sélectionner des enregistrements en fonction de critères spécifiques.

Les attributs personnalisés sont exportés avec les commandes **WP EXPORTER DOCUMENT** et **WP EXPORTER VARIABLE**. Ils sont également exportés lorsque vous convertissez un champ objet 4D Write Pro en JSON à l'aide de la commande **JSON Stringify** (en plus des attributs principaux de document de 4D Write Pro).

Pour écrire ou lire des attributs personnalisés, vous pouvez utiliser la notation objet ou les commandes **OB Lire** et **OB FIXER**. Par exemple, dans la méthode du formulaire, vous pouvez écrire :

```
Si(Eventement formulaire=Sur validation)
[MesDocuments]Mon4DWP["monatt_Dernière modif par"]:=Utilisateur courant
[MesDocuments]Mon4DWP.monatt_Catégorie:=Memo
[MesDocuments]Mon4DWP:=[MesDocuments]Mon4DWP //enregistrer la modification
Fin de si
```

ou :

```
Si(Eventement formulaire=Sur validation)
OB FIXER([MesDocuments]Mon4DWP;"monatt_Dernière modif par";Utilisateur courant)
OB FIXER([MesDocuments]Mon4DWP;"monatt_Catégorie";"Memo")
Fin de si
```

Vous pouvez bien entendu lire les attributs personnalisés des documents :

```
vAttrib:=[MesDocuments]Mon4DWP.monatt_Catégorie
```

ou :

```
vAttrib:=OB Lire([MesDocuments]Mon4DWP;"monatt_Catégorie")
```

Si vous avez stocké des attributs personnalisés avec les documents 4D Write Pro dans votre fichier de données, vous pouvez effectuer des recherches sur ces attributs afin de créer des sélections d'enregistrements contenant les valeurs recherchées. Exemple :

```
CHERCHER PAR ATTRIBUT([MesDocuments];[MesDocuments]Mon4DWP;"monatt_Catégorie";="Memo")
//sélectionne tous les enregistrements de la table MesDocuments dont l'attribut personnalisé "monatt_Catégorie" contient la valeur
"Memo"
//dans le champ objet Mon4DWP (associé à une zone 4D Write Pro)
```

Note sur les noms des attributs personnalisés : Comme les attributs personnalisés partagent le même espace de nommage que les attributs internes des documents 4D Write Pro, nous recommandons fortement l'utilisation de préfixes lorsque vous définissez les noms de vos attributs, afin d'éviter tout conflit entre les attributs internes et personnalisés. Les noms sans préfixe sont réservés aux attributs internes de 4D Write Pro. En revanche, la définition des préfixes est libre (nous utilisons par exemple le préfixe "monatt_" dans les exemples ci-dessus).

Note : A compter de 4D v15 R4, les attributs internes de 4D Write Pro sont également accessibles par programmation via les commandes standard **CHERCHER PAR ATTRIBUT**, **OB Lire** et **OB FIXER**, mais aussi **WP FIXER ATTRIBUTS**, **WP LIRE ATTRIBUTS** et **WP REINITIALISER ATTRIBUTS**. Pour plus d'information, veuillez vous reporter à la section **Attributs 4D Write Pro**.

Utiliser une zone 4D Write Pro

Gestion des documents dans les zones 4D Write Pro

Dans les applications 4D, les documents 4D Write Pro sont créés, importés et/ou exportés via des commandes dédiées placées dans le thème **4D Write Pro** (**WP EXPORTER DOCUMENT**, **WP EXPORTER VARIABLE**, **WP Importer document**, **WP Nouveau**).

Vous pouvez également associer une zone 4D Write Pro à un champ objet de la base dans un formulaire. De cette manière, chaque document 4D Write Pro est automatiquement sauvegardé avec l'enregistrement et est conservé dans les données de la base (cf. section **Stocker les documents 4D Write Pro dans des champs objet 4D**).

Format de document .4wp

Vous pouvez sauvegarder des documents 4D Write Pro sur disque et les rouvrir sans aucune perte de données grâce au format natif **.4wp**.

Le format **.4wp** est constitué d'un dossier zip dont le nom est le titre du document, et contenant du texte HTML et des images :

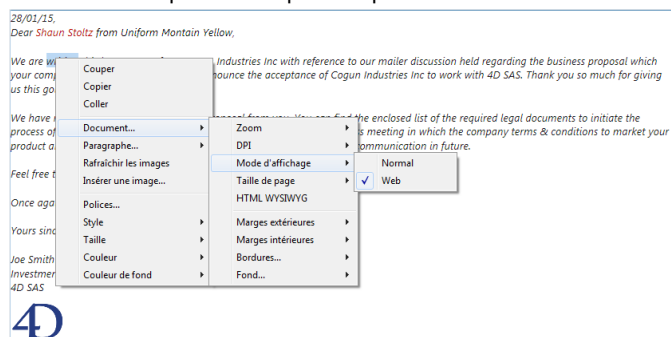
- le texte HTML combine du HTML standard et des expressions 4D (non interprétées) ainsi que des balises 4D spécifiques,
- les images sont stockées dans un dossier du même nom que le document, situé à côté du fichier HTML.

Comme les documents .4wp sont basés sur du HTML, ils peuvent être importés et ouverts dans toute application tierce qui prend en charge le format HTML.

Note : Le format interne des documents 4D Write Pro est du HTML étendu propriétaire, compatible HTML5/XHTML5, mais utilisant son propre sous-ensemble d'attributs et de balises HTML/CSS. Par conséquent, seuls les documents HTML exportés par 4D Write Pro peuvent être ouverts par 4D Write Pro sans risque de perte d'informations. Importer des documents HTML qui ont été créés par une source externe peut provoquer des erreurs.

Interface utilisateur

Lorsque la propriété **Menu contextuel** est cochée pour une zone 4D Write Pro (voir **Définir une zone 4D Write Pro**), un menu contextuel complet est disponible pour les utilisateurs en mode Application :



Ce menu donne accès à l'ensemble des fonctions de 4D Write Pro.

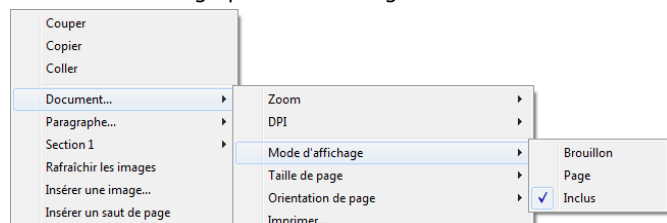
Sélectionner le mode d'affichage

4D Write Pro propose trois modes d'affichage pour les documents :

- **Brouillon** : Mode brouillon avec des propriétés basiques
- **Page** (défaut) : Mode "vue impression"
- **Inclus** : Mode adapté aux zones incluses dans les formulaires ; dans ce mode, les marges, pieds de page, colonnes, entêtes, cadres, etc, ne sont pas affichés.

Ce mode peut également être utilisé pour obtenir un affichage de type Web (si vous avez également sélectionné la résolution 96 dpi et l'option HTML WYSIWYG).

Le mode d'affichage peut être configuré via le menu contextuel de la zone :



Note : Le mode d'affichage n'est pas stocké avec le document.

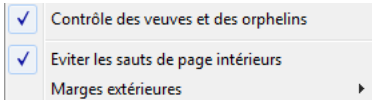
Pour les zones incluses dans les formulaires 4D, le mode d'affichage peut également être défini par défaut à l'aide de la Liste des propriétés. Dans ce cas, le mode d'affichage est une propriété de l'objet de formulaire 4D View Pro (pour plus d'informations, veuillez vous reporter au paragraphe **Configurer les propriétés d'affichage**).

Fonctionnalités de l'affichage en page

Lorsqu'un document est en mode d'affichage **Page**, les propriétés de document suivantes sont apparentes pour l'utilisateur :

- les contours des pages, qui représentent les limites d'impression
- la largeur et la hauteur des pages (21x29,7 cm par défaut)
- l'orientation des pages (portrait par défaut)
- les marges (2,5 cm par défaut)

De plus, des propriétés de paragraphe supplémentaires sont disponibles dans le sous-menu **Paragraphe...** :



- **Contrôle des veuves et des orphelins** : Lorsque cette option est cochée pour un paragraphe, 4D Write Pro n'autorise pas les *veuves* (dernière ligne d'un paragraphe isolée en haut d'une page) ni les *orphelines* (première ligne d'un paragraphe isolée en bas d'une page) dans le document. Dans le premier cas, la ligne précédente du paragraphe est passée en haut de la page afin que deux lignes soient affichées. Dans le second cas, la ligne isolée est passée sur la page suivante.
- **Eviter les sauts de page intérieurs** : Lorsque cette option est cochée pour un paragraphe, 4D Write Pro empêche que le paragraphe soit divisé sur deux pages.

Vous pouvez également utiliser des commandes relatives à la gestion des pages telles que **Insérer un saut de page/colonne/section, Document.../Taille de page** ou **Document.../Orientation de page**.

Note : Lorsque le mode d'affichage d'un document est **Inclus** ou **Brouillon**, les propriétés de page peuvent être définies, même si leur effet n'est pas visible. En mode Brouillon, les effets des propriétés de paragraphe suivantes sont toutefois visibles :

- Limites des hauteurs de pages (des lignes sont dessinées)
- Colonnes
- Propriété "Eviter les sauts de pages intérieurs"
- Propriété "Contrôle des veuves et des orphelins".

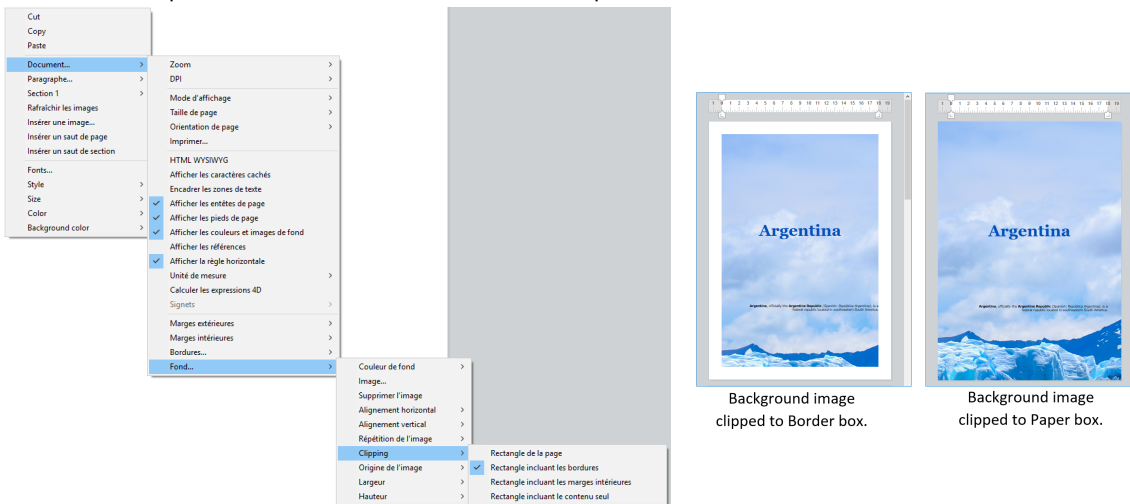
Fond

L'arrière-plan (le fond) des documents et des éléments de documents de 4D Write Pro (tableaux, paragraphes, sections, en-têtes/pieds de pages, etc.) peut être défini à l'aide des attributs suivants :

- couleurs
- bordures
- images
- positionnement horizontal, vertical et origine
- rectangle d'application
- répétition

Ces attributs peuvent être définis par programmation pour le fond des éléments individuels dans la page et/ou de l'ensemble du document via la commande **WP FIXER ATTRIBUTS** ou les **Utiliser les actions standard 4D Write Pro**. Pour obtenir la liste complète des attributs de gestion des arrière-plans ainsi que les éléments auxquels ils peuvent s'appliquer, reportez-vous à la section **Attributs 4D Write Pro**.

Les utilisateurs peuvent modifier les attributs d'arrière-plan à l'aide du menu contextuel de la zone :



Pour un exemple d'ajout d'image d'arrière-plan en taille réelle, [téléchargez la base How Do I \(HDI\) dédiée](#).

Gestion des en-têtes, pieds de pages et sections

Les documents 4D Write Pro peuvent contenir des en-têtes et des pieds de page. Les en-têtes et les pieds de page sont liés aux sections.

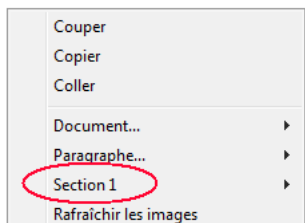
Une section est une partie d'un document qui est définie par un intervalle de pages, et qui peut comporter des attributs de pagination et communs qui lui sont propres. Un document peut contenir de une à *N* sections, *N* étant le nombre total de pages. Chaque page ne peut appartenir qu'à une seule section.

Vous pouvez définir un ensemble d'en-têtes et de pieds de page pour chaque section.

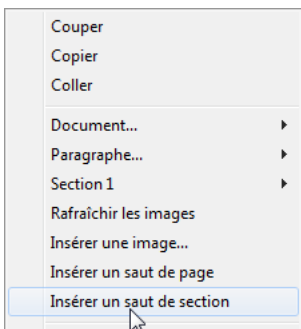
Définir une section

Une section est un sous-ensemble de pages contiguës d'un document 4D Write Pro. Un document peut contenir une ou plusieurs sections, chaque section pouvant elle-même contenir un nombre variable de pages, depuis une page unique jusqu'au nombre total de pages du document. Une section peut contenir une seule colonne ou jusqu'à 20 colonnes par page.

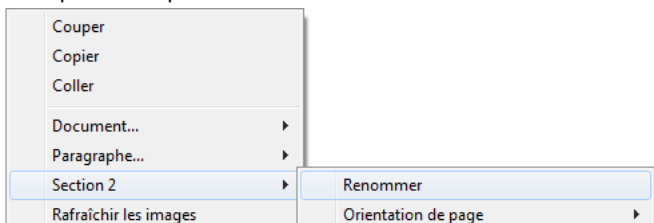
Par défaut, un document contient une seule section, nommée **Section 1**. Le menu contextuel de 4D Write Pro affiche ce numéro de section dans toutes les pages du document :



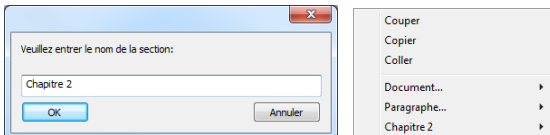
Vous créez une nouvelle section en ajoutant un saut de section dans le texte :



Lorsqu'un ou plusieurs sauts de section ont été ajoutés, le menu contextuel affiche un numéro incrémenté pour chaque section. Vous pouvez cependant renommer les sections :



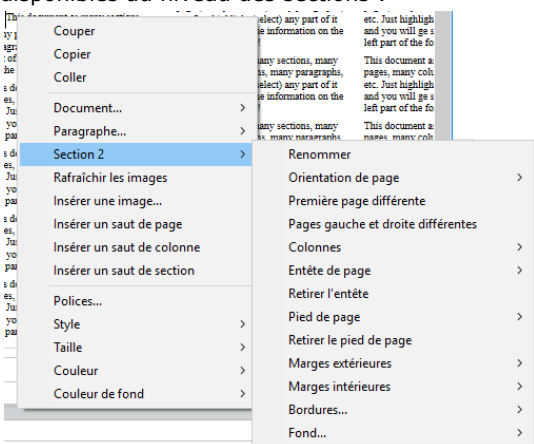
Le nom que vous saisissez est alors utilisé comme nom de section pour le document :



Notez que si vous avez défini une première page différente ou des pages droite/gauche différentes pour une section, le type de page est également affiché dans le menu (voir ci-dessous).

Attributs de section

Les sections héritent des attributs du document. Cependant, les attributs de documents communs, y compris les en-têtes et pieds de page, peuvent être modifiés séparément pour chaque section. Le menu contextuel affiche les propriétés et attributs disponibles au niveau des sections :



- **Orientation de page** : permet de définir une orientation de page spécifique (portrait ou paysage) par section
- **Première page différente** : permet de définir des attributs spécifiques pour la première page de la section ; cette fonction peut être utilisée pour créer des pages de garde, par exemple. Lorsque cette option est cochée, la première page de la section est considérée elle-même comme une sous-section et peut disposer de ses propres attributs.

Chapitre 2 (Première page) ▶

- **Pages gauche et droite différentes** : permet de définir des attributs différents pour les pages gauches et droites de la section. Lorsque cette option est cochée, les pages gauches et droites de la section sont considérées comme des sous-sections et peuvent disposer de leurs propres attributs.

Chapitre 2 (Page droite) ▶

- **Colonnes**: permet de définir le nombre et les propriétés des colonnes pour la section. Ces options sont détaillées ci-dessous.
- **Entête et Pied de page** : ces options vous permettent de définir des en-têtes et pieds de page spécifiques. Elles sont détaillées ci-dessous.
- **Marges extérieures et intérieures / Bordures / Fond** : ces attributs peuvent être définis séparément pour chaque section. Pour plus d'informations sur ces attributs, veuillez vous reporter à la page **Attributs 4D Write Pro**.

Insérer des en-têtes et des pieds de page

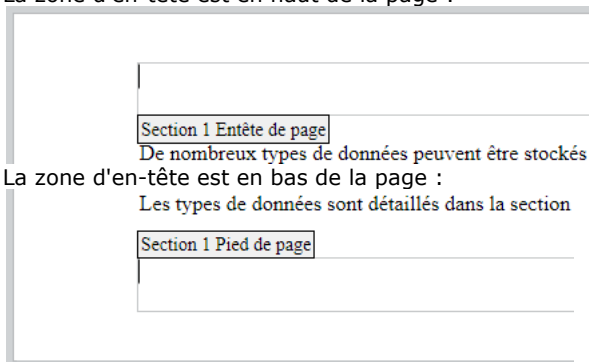
Chaque section peut comporter un en-tête et un pied de page spécifiques. Les en-têtes et les pieds de page sont visibles uniquement lorsque le mode d'affichage du document est **Page**.

A l'intérieur d'une section, vous pouvez définir jusqu'à trois en-têtes et pieds de page différents, en fonction des options activées :

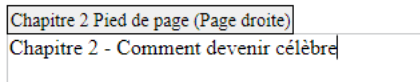
- première page,
- page(s) gauche(s),
- page(s) droite(s),

Pour créer un en-tête ou un pied de page :

1. Assurez-vous que le document est en mode d'affichage **Page**.
2. Double-cliquez dans la zone d'en-tête ou de pied de la section et de la page souhaitées afin de passer en édition.
 - La zone d'en-tête est en haut de la page :



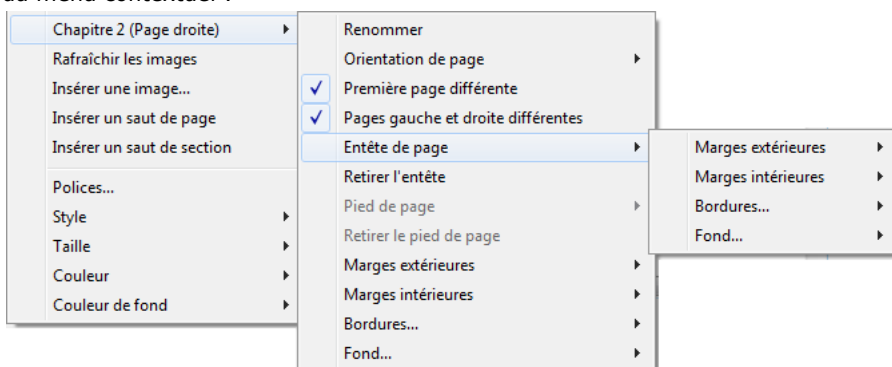
Vous pouvez ajouter tout contenu statique, qui sera automatiquement répété sur chaque page de la section (à l'exception de la première page, si l'option est activée).



Vous pouvez ajouter du contenu dynamique tel que le numéro de page ou le nombre de pages, à l'aide de la commande **ST INSERER EXPRESSION** (pour plus d'informations, veuillez vous reporter au paragraphe **Insérer des expressions de page et de document**).

Note : Vous pouvez également gérer les en-têtes et pieds de page par programmation à l'aide de commandes spécifiques telles que **WP Lire entete** et **WP Lire pied**.

Une fois qu'un en-tête ou un pied de page a été défini pour une section, vous pouvez configurer ses attributs communs à l'aide du menu contextuel :



Pour plus d'informations sur les attributs **Marges extérieures**, **Marges intérieures**, **Bordures** et **Fond**, veuillez vous reporter à la section **Attributs 4D Write Pro**.

Vous pouvez supprimer l'ensemble de la définition d'un en-tête ou d'un pied de page (contenu et attributs) en sélectionnant la commande **Retirer l'entête** ou **Retirer le pied de page** dans le menu contextuel.

Compatibilité

4D Write Pro prend en charge les en-têtes et les pieds de page des documents convertis depuis le plug-in 4D Write.

Les expressions et propriétés suivantes sont également prises en charge et converties lorsqu'elles étaient présentes dans les en-têtes et pieds de page des documents du plug-in 4D Write :

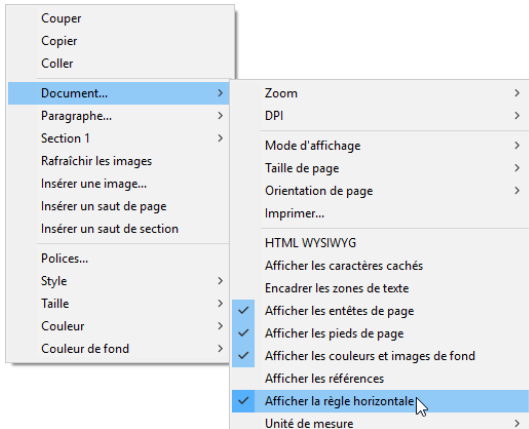
- variables numéros et nombre de pages
- première page différente
- pages gauche/droite différentes

Utiliser les règles

Les règles sont disponibles dans tous les modes d'affichage de 4D Write Pro et ont les caractéristiques suivantes :

- Graduations en cm, mm, pouces ou pt en fonction de l'unité courante définie dans le document 4D Write Pro. Vous pouvez changer l'unité de mesure du document à l'aide du menu contextuel ou via l'attribut [wk.layout.unit](#).
- Symbole de retrait première ligne
- Symbole de marge de paragraphe gauche
- Symbole de marge de paragraphe droite
- Tabulations affichées sur le bord inférieur de la règle
- Zones de couleur représentant les marges de page gauche et droite

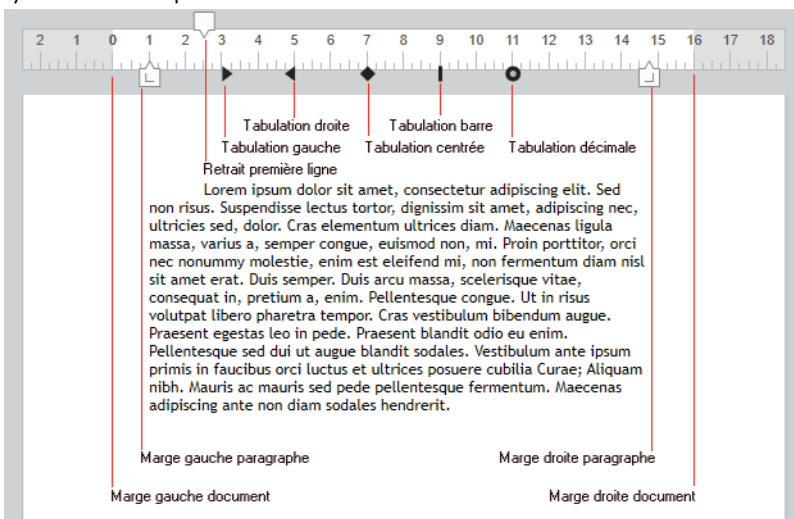
Vous pouvez afficher ou masquer la règle en sélectionnant ou en désélectionnant la ligne **Afficher la règle horizontale** dans le menu contextuel de la zone 4D Write Pro :



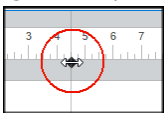
Note : Une option des propriétés de la zone permet également de définir l'affichage par défaut de la règle (cf. section [Configurer les propriétés d'affichage](#)).

Ajuster les marges et le retrait

Vous pouvez modifier les marges, l'indentation (le retrait) des premières lignes et les positions des tabulations en déplaçant les symboles correspondants à l'aide de la souris :



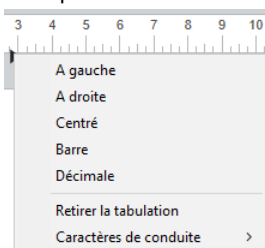
Lorsque vous survolez l'un de ces symboles avec la souris, le curseur est modifié pour indiquer qu'il peut être déplacé, et une ligne de repère verticale apparaît lorsque vous le déplacez :



Lorsque plusieurs paragraphes sont sélectionnés, le déplacement d'un symbole de marge ou d'indentation est appliqué à tous les paragraphes sélectionnés. Si vous maintenez la touche **Majuscule** enfoncée pendant le déplacement, les intervalles existants entre les marges sont conservés dans les paragraphes sélectionnés.

Gestion des tabulations

Vous pouvez utiliser le menu contextuel de la règle pour créer, modifier ou supprimer des tabulations :



Pour créer une tabulation, cliquez sur la règle avec le bouton droit de la souris et choisissez son type depuis le menu contextuel ; un clic simple avec le bouton gauche de la souris crée automatiquement une tabulation gauche par défaut. Vous pouvez également effectuer un clic droit sur une tabulation existante pour modifier son type dans le menu contextuel.

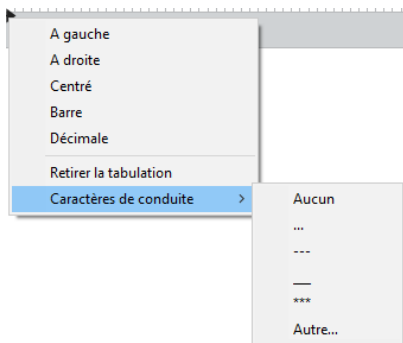
Note : Le libellé **Retirer la tabulation** est disponible en cas de clic droit sur une tabulation existante. Vous pouvez également supprimer des tabulations en les faisant glisser hors de la zone de la règle.

Définir les caractères de conduite

Vous pouvez définir les caractères qui précèdent les tabulations (caractères de conduite) en sélectionnant un caractère prédéfini ou en saisissant le caractère à utiliser. Les caractères prédéfinis sont :

- Aucun (aucun caractère n'est affiché - *défaut*)
- (points)
- --- (tirets)
- ___ (traits de soulignement)
- *** (astérisques)

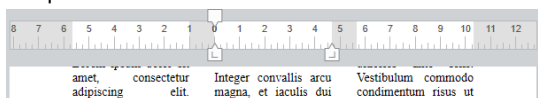
Les caractères de conduite apparaissent toujours avant la tabulation et suivent la direction du texte (gauche à droite ou droite à gauche). Ils peuvent être définis soit par programmation avec les commandes **WP FIXER ATTRIBUTS**, **WP LIRE ATTRIBUTS** ou **WP REINITIALISER ATTRIBUTS** et le sélecteur `wk tab stop leadings`, soit via le menu contextuel de la règle :



L'option **Autre..** affiche une boîte de dialogue permettant de saisir un caractère de conduite personnalisé.

Règles multi-colonnes

Lorsque deux ou plusieurs colonnes sont définies pour le document ou la section, la règle affiche une zone spécifique pour chaque colonne :



Note : La fonctionnalité multi-colonnes n'est pas disponible en mode d'affichage **Inclus**.

Événement Sur après modification

Un événement Sur après modification est généré dans l'objet de formulaire 4D Write Pro à chaque déplacement d'une tabulation ou d'une marge, ainsi qu'à chaque ajout ou suppression d'une tabulation, que l'action ait été effectuée via le menu contextuel ou par glisser-déposer.

Gestion des colonnes

4D Write Pro vous permet de créer des documents comportant plusieurs colonnes. Les colonnes sont chaînées depuis la colonne la plus à gauche jusqu'à la colonne la plus à droite. Autrement dit, lorsque vous saisissez du texte, le flux de texte commencera par remplir la colonne de gauche puis continuera dans la colonne à sa droite, puis ainsi de suite jusqu'à ce qu'il atteigne la fin de la page. Une fois que le bas de la page est atteint, le flux de texte est dirigé sur la page suivante. Pour que vous puissiez contrôler ce fonctionnement, 4D Write Pro permet d'insérer des sauts de colonnes.



Les colonnes peuvent être définies au niveau du document (elles sont alors affichées dans l'ensemble du document) et/ou au niveau de la section (chaque section peut avoir sa propre configuration de colonne).

Note : Les colonnes sont prises en charge uniquement dans les modes d'affichage **Page** et **Brouillon** (elles ne sont pas affichées en mode **Inclus** et ne sont pas exportées par **WP EXPORTER DOCUMENT** au format `wk page web complète`).

Les colonnes peuvent être définies via :

- le sous-menu **Colonnes** du menu contextuel de la zone 4D Write Pro,
- les attributs 4D Write Pro (voir **Attributs 4D Write Pro**),
- les actions standard 4D Write Pro (voir **Utiliser les actions standard 4D Write Pro**).

Vous pouvez définir ou lire les propriétés et actions suivantes pour les colonnes :

Propriétés	Description	Attributs de Document	Actions standard
Nombre de colonnes	Vous pouvez définir jusqu'à 20 colonnes par document/section	wk column count	columnCount
Espacement de colonne	Espacement entre les colonnes en pts, pouces ou cm. A noter que toutes les colonnes ont la même largeur. La largeur de chaque colonne est calculée automatiquement par 4D Write Pro en fonction du nombre de colonnes, de la largeur de la page et de l'espacement	wk column spacing	columnSpacing
Largeur de colonne	(attribut en lecture seule) Largeur actuelle de chaque colonne, i.e. largeur calculée	wk column width	-
Style, couleur et épaisseur du séparateur de colonne	Vous pouvez ajouter un séparateur vertical (une ligne décorative) entre les colonnes. Ces options vous permettent de définir le style, la couleur et l'épaisseur de la ligne. <div style="display: flex; justify-content: space-between; font-size: small;"> <div style="width: 30%;"> <p> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam eget vehicula lorem. Fusce vehicula dictum nunc ut pulvinar. Sed ornare risus ligula, sit amet ullamcorper nisi tempor a. Donec elementum sapien quis auctor rhoncus. Cras eget facilisis odio. Proin</p> </div> <div style="width: 30%;"> <p> mauris magna, ut ultricies ipsum pulvinar vel. Praesent id volutpat lorem, a tempor est. Suspendisse tristique sapien sapien, vitae malesuada augue vehicula in. Sed tempus enim a bibendum rhoncus. Morbi semper ipsum et gravida lobortis. Pellentesque sed sem tempor nulla laoreet</p> </div> <div style="width: 30%; text-align: center;"> <p>Pour supprimer le séparateur vertical, choisissez Aucun comme style.</p> </div> </div>	wk column rule style , wk column rule color , wk column rule width	columnRuleStyle , columnRuleColor , columnRuleWidth
Insérer saut	Insérer un saut de colonne	wk column break , voir aussi WP INSERER RUPTURE	insertColumnBreak
Menu Colonnes	Créer un sous-menu Colonnes	-	columns

Utiliser des tableaux

Les documents 4D Write Pro peuvent contenir des tableaux. Les tableaux 4D Write Pro sont créés et construits par programmation, mais leur contenu peut être modifié et géré par l'utilisateur. Plusieurs attributs de tableau 4D Write Pro peuvent être définis, notamment la hauteur de ligne, l'alignement, les marges, le style de texte, la couleur, ou encore les bordures.

Name	Date	Value
Alpha	03/04/2017	867
Bravo	05/04/2017	579
Charlie	02/06/2017	501
Delta	06/06/2017	354
Echo	09/04/2017	463
Foxtrot	31/05/2017	912
Golf	17/03/2017	756
Hotel	11/04/2017	437
India	20/04/2017	64
Juliett	05/04/2017	684
Total		5617

Créer un tableau

Pour créer un tableau 4D Write Pro, vous devez appeler la commande **WP Insérer tableau**. Vous pouvez ensuite ajouter des lignes à l'aide de la commande **WP Tableau ajouter ligne**.

Note : L'utilisateur peut créer un tableau par copier-coller d'une plage de cellules :

Golf	17/03/2017	756
Hotel	11/04/2017	437
India	20/04/2017	64
Juliett	05/04/2017	684
Total		5617

Golf	17/03/2017
Hotel	11/04/2017

Modifier les tableaux

Le contenu des cellules peut être ajouté par programmation à l'aide de la commande **WP Tableau ajouter ligne**.

Une fois qu'un tableau est créé, le contenu des cellules peut également être modifié à l'exécution par l'utilisateur. Il peut cliquer dans les cellules, puis sélectionner, modifier, copier/coller ou supprimer le texte et les images comme dans tout paragraphe. Il peut naviguer parmi les cellules en appuyant sur la touche **Tabulation (Maj+Tabulation)** pour se déplacer dans la direction inverse).

Notez que la largeur des cellules est fixe : lorsque l'utilisateur saisit du texte ou colle une image, la hauteur de la ligne est automatiquement ajustée si nécessaire et le texte passe automatiquement à la ligne :

Charlie	20/05/2017	844
Delta is the fourth letter of the most known spelling alphabet	06/05/2017	188
Echo	29/05/2017	579

Les utilisateurs peuvent également sélectionner des colonnes, des lignes ou des cellules et appliquer divers attributs relatifs au style de texte, aux couleurs, alignements etc., via le pop up menu intégré ou une interface personnalisée. 4D Write Pro fournit plusieurs commandes permettant de sélectionner toutes les parties d'un tableau :

- **WP Tableau lire lignes** pour obtenir une plage de lignes
- **WP Tableau lire colonnes** pour obtenir une plage de colonnes(*)
- **WP Tableau lire cellules** pour obtenir une plage de cellules

(*) Les colonnes n'ont pas d'équivalent en html. Dans 4D Write Pro, une plage de colonne est en réalité une plage de cellules, ce qui signifie que les colonnes ont la même priorité que les cellules (cf. ci-dessous).

Une fois que vous avez sélectionné une plage, vous pouvez lui appliquer un attribut à l'aide de la commande **WP FIXER ATTRIBUTS**. A l'intérieur des cellules, les attributs sont appliqués aux paragraphes, caractères ou images, suivant leur contenu. Par exemple, vous pouvez définir la hauteur, la taille de police, la bordure, la marge intérieure etc. d'un tableau ou des cellules d'un tableau (pour plus d'informations, veuillez vous référer à la section **Attributs 4D Write Pro**).

Lorsque des attributs différents sont appliqués à des éléments concurrents d'un tableau, l'ordre de priorité suivant est utilisé pour le rendu :

1. Le tableau est dessiné en premier
2. Les lignes sont ensuite dessinées (écrasant les attributs du tableau)
3. Les cellules/colonnes sont ensuite dessinées (écrasant les attributs des lignes).

■ Imprimer des documents 4D Write Pro

Les documents 4D Write Pro peuvent être imprimés de deux manières :

- en tant que parties de formulaires 4D
- en tant que documents indépendants

Impression de documents dans des formulaires 4D

Vous pouvez imprimer des objets 4D Write Pro inclus dans des formulaires 4D de tout type (projet, table, entrée ou sortie), à l'aide des commandes d'impression standard de 4D telles que **IMPRIMER SELECTION** ou **IMPRIMER ENREGISTREMENT**.

L'option standard **Impression taille variable** est également prise en charge(*) pour les zones 4D Write Pro, vous permettant de gérer les variations de taille durant l'impression. Lorsque cette option est cochée, la marge (extérieure et intérieure) et la bordure hautes ne sont appliquées qu'à la première page. La marge (extérieure et intérieure) et la bordure basses ne sont appliquées qu'à la dernière page. Les propriétés de pagination du document sont ignorées : le contrôle des veuves et des orphelins est désactivé, et les sauts de page ne sont pas appliqués (ces propriétés sont utilisées uniquement pour le rendu des pages à l'écran ou l'impression autonome du document). Lorsque l'option **Impression taille variable** est sélectionnée, seuls les objets situés au-dessus de la zone dans le formulaire sont imprimés. Pour plus d'informations sur cette option, reportez-vous à la section "**Impression taille variable**" dans le manuel Mode Développement.

(*) Les commandes **Imprimer objet** et **Imprimer ligne** ne sont pas compatibles avec cette option.

Mode d'affichage et impression

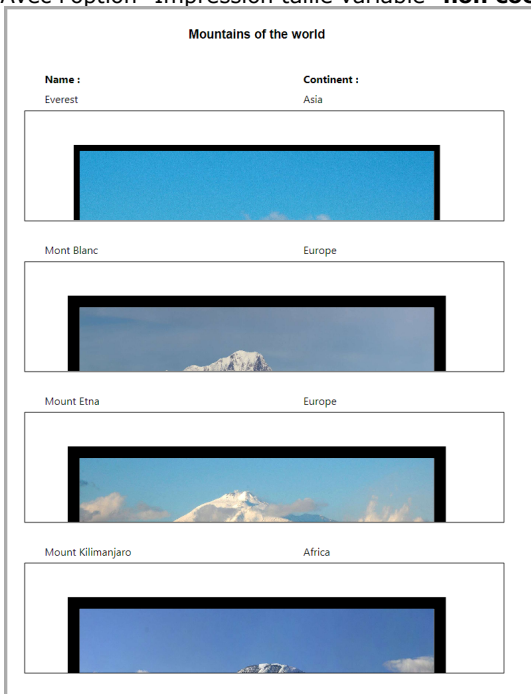
Quel que soit le **Mode d'affichage** défini pour la zone 4D Write Pro (cf. **Configurer les propriétés d'affichage**), elle est toujours imprimée en mode **Inclus** lorsque vous utilisez une commande d'impression 4D telle que **Imprimer ligne**. Dans ce contexte, les propriétés d'Apparence ne sont pas prises en charge pour les objets de formulaire 4D Write Pro : Mode d'affichage (toujours "Inclus"), Montrer les entêtes, Montrer les pieds de page, Montrer le cadre de page (toujours "non") et Montrer les caractères cachés (toujours "non").

Exemple

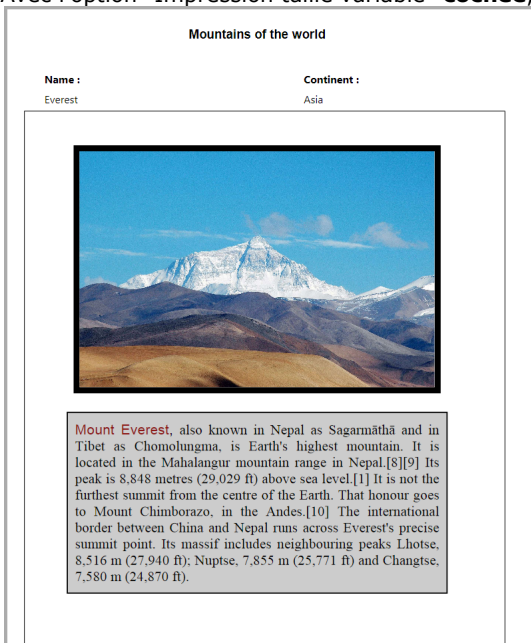
L'exemple suivant illustre l'effet de l'option **Impression taille variable** sur une zone 4D Write Pro incluse dans le formulaire de sortie par défaut. Le code suivant est exécuté :

```
TOUT SELECTIONNER([Movies])
TRIER([Movies]Title)
IMPRIMER SELECTION([Movies])
```

- Avec l'option "Impression taille variable" **non cochée**, vous obtenez le résultat suivant :



- Avec l'option "Impression taille variable" **cochée**, vous obtenez le résultat suivant :



(Source des texte d'exemple : wikipedia)

Impression de documents indépendants

A partir de 4D v15 R5, 4D Write Pro inclut des fonctionnalités d'impression vous permettant d'imprimer des documents 4D Write Pro indépendants et de contrôler les options standard d'impression comme le format, l'orientation ou les numéros de page.

Commandes 4D Write Pro

Essentiellement, deux commandes gèrent les fonctions d'impression avec 4D Write Pro : **WP IMPRIMER** et **WP UTILISER PARAMETRES IMPRESSION**.

- **WP IMPRIMER** lance une tâche d'impression pour le document 4D Write Pro ou ajoute le document dans la tâche d'impression courante.
- **WP UTILISER PARAMETRES IMPRESSION** modifie les options d'impression de la page courante, sur la base des attributs du document 4D Write Pro pour la taille de page et l'orientation.

Notes de configuration :

- Les commandes d'impression de 4D Write Pro reposent sur la nouvelle architecture interne conçue pour les versions 64 bits de 4D. Elles sont toutefois prises en charge dans les versions 32 bits, mais en-dehors des tâches d'impression démarrées avec **OUVRIR TACHE IMPRESSION** (cf. **WP IMPRIMER**).
- Sur des machines avec Windows 7 ou Windows Server 2008 R2, vérifiez que le *Platform Update for Windows 7* a bien été installé pour que les fonctionnalités d'impression soient prises en charge.

Commandes 4D standard

Les commandes 4D suivantes prennent en charge les fonctionnalités d'impression de 4D Write Pro :

- **FIXER OPTION IMPRESSION** et **LIRE OPTION IMPRESSION** : Toutes les options sont prises en charge pour les documents 4D Write Pro imprimés avec **WP IMPRIMER**. Pour Option papier et Option orientation, il sera généralement préférable d'appeler **WP UTILISER PARAMETRES IMPRESSION** de façon à facilement synchroniser ces attributs avec les options du document 4D Write Pro. L'option Option intervalle de page (15) vous permet d'imprimer de la page x à la page y uniquement.
- **PARAMETRES IMPRESSION** : permet de fixer les paramètres d'impression pour l'imprimante courante ; si la commande **WP IMPRIMER** est appelée par la suite, elle utilisera les paramètres d'impression modifiés si certains ont été modifiés au moyen des dialogues de paramétrage d'impression (excepté pour les options de marges qui prennent toujours en compte les options du document 4D Write Pro).
- **OUVRIR TACHE IMPRESSION** et **FERMER TACHE IMPRESSION**: **WP IMPRIMER** peut être appelée entre **OUVRIR TACHE IMPRESSION** et **FERMER TACHE IMPRESSION** de façon à insérer un ou plusieurs document(s) 4D Write Pro dans une tâche d'impression.

Filtrer les expressions contenues dans un document 4D Write Pro

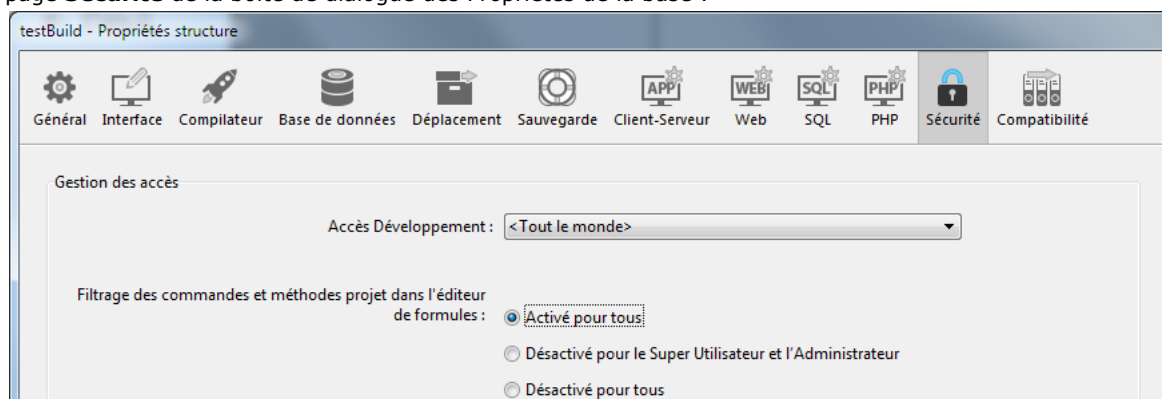
Présentation

Les documents 4D Write Pro peuvent contenir des références à des expressions 4D dynamiques telles que des variables ou des champs, mais aussi des formules, des méthodes projet ou des commandes du langage. Ces références sont évaluées lorsque les documents sont affichés ou imprimés. Pour des raisons de sécurité, l'évaluation des expressions dynamiques doit être contrôlée par le développeur pour éviter qu'une expression inappropriée soit utilisée et génère des modifications inattendues ou des effets de bord dans la base de données. Ce principe prévient par exemple l'exécution de commandes telles que **SUPPRIMER SELECTION** ou de méthodes telles que "SupprimerClient".

Note : Les expressions peuvent être insérées à l'aide de la commande **ST INSERER EXPRESSION** ou par la modification des documents 4D Write Pro avec un éditeur HTML.

Support de la fonctionnalité standard de filtrage de 4D

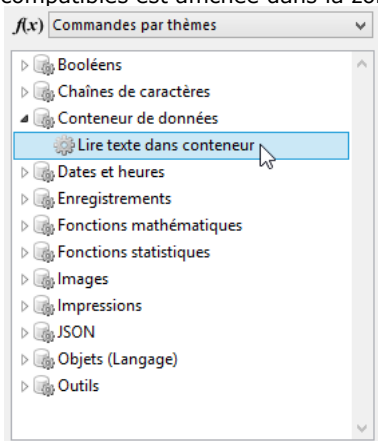
A partir de 4D v16, l'évaluation des commandes et méthodes projet de 4D insérées dans les documents 4D Write Pro est désormais sous le contrôle de l'option de filtrage de 4D, définie au niveau de la base de données. Cette option se trouve dans la page **Sécurité** de la boîte de dialogue des Propriétés de la base :



Par défaut, dans 4D, la valeur de cette option est **Activé pour tous**, ce qui signifie que les commandes et méthodes doivent être explicitement autorisées ; dans le cas contraire, des messages d'erreur sont retournés. Vous pouvez désactiver ce filtrage partiellement (pour le Super Utilisateur et l'Administrateur) ou pour tous les utilisateurs. Cette option est utilisée dans toute la base et contrôle toutes les évaluations de formule utilisateur. Pour plus d'informations sur cette option, veuillez vous référer au manuel *Mode Développement*.

Lorsque cette option est activée :

- les utilisateurs peuvent appeler uniquement des commandes "compatibles" avec les formules. La liste des commandes compatibles est affichée dans la zone de droite de l'éditeur de Formule :



- les utilisateurs ne peuvent exécuter aucune méthode projet. Les méthodes projet que vous souhaitez autoriser dans les documents 4D Write Pro doivent être explicitement déclarées en utilisant la commande **FIXER METHODES AUTORISEES**.

Si une commande ou une méthode 4D "interdite" est trouvée durant l'évaluation d'une expression dans un document 4D Write Pro, la valeur est remplacée par "#command_5_command_5 Error _command_5 48".

Exemple

Vous insérez l'expression suivante dans votre document 4D Write Pro :

```
ST INSERER EXPRESSION(*;"WriteProArea";"Gender")
```

Par défaut, si l'option de sécurité est cochée, la méthode **Gender** ne sera pas évaluée :

Dear ## Error # 48 Smith,

Si vous exécutez le code suivant :

```
TABLEAU TEXTE(aTallow;1)
aTallow{1}:= "Gender"
FIXER METHODES AUTORISEES(aTallow)
```

La méthode **Gender** retournera une valeur :

Dear Mr. Smith,

Compatibilité

Ce filtrage n'était pas activé pour les documents 4D Write Pro dans les versions précédentes. Si vos documents 4D Write Pro référençaient des méthodes 4D, elles ne seront plus évaluées correctement une fois les documents convertis en 4D v16 ou plus. Des messages "#command_5_command_5 Error _command_5 48" seront affichés à la place. Dans ce cas, vous devrez ajouter les méthodes dans la liste des méthodes autorisées en utilisant la commande **FIXER METHODES AUTORISEES**.

Import de documents 4D Write

Une des principales fonctions du nouvel objet 4D Write Pro est sa capacité à importer et à convertir les documents 4D Write existants. Ce principe vous permet de migrer les applications qui s'appuient actuellement sur le plug-in 4D Write.

Notes de compatibilité :

- Seuls les documents 4D Write de dernière génération ("4D Write v7") sont pris en charge.
- Le copier-coller d'un document 4D Write vers une zone 4D Write Pro n'est pour le moment pas pris en charge. L'importation d'un document 4D Write peut être uniquement effectuée via les commandes du langage de 4D Write Pro.

Comment importer un document 4D Write ?

Les objets 4D Write Pro permettent d'importer des documents 4D Write de deux manières :

- Pour les fichiers 4D Write stockés sur disque, vous pouvez utiliser la commande **WP Importer document**,
- Pour les fichiers 4D Write stockés dans des champs BLOB, vous pouvez utiliser la commande **WP Nouveau**.

Pour plus d'informations, veuillez vous reporter aux descriptions de ces commandes.

Quelles propriétés 4D Write sont importées ?

Afin de faciliter votre migration du plug-in 4D Write vers 4D Write Pro, nous voulons conserver autant de fonctions 4D Write que possible dans les objets 4D Write Pro.

Ce paragraphe liste les propriétés du plug-in 4D Write qui sont actuellement récupérées dans une zone 4D Write Pro après un import effectué à l'aide de la commande **WP Importer document** ou **WP Nouveau**.

Notez que quelques différences de rendu pourront apparaître, qui ne seront pas nécessairement considérées comme des bogues. Par exemple, des différences peuvent être liées à la police par défaut utilisée par 4D Write Pro pour les puces, ou aux conversions des caractères soulignés.

Document info

4D Write plug-in	4D Write Pro
Date & heure de création	Disponible
Date & heure de modification	Disponible
Verrouillé	Non disponible (utiliser la propriété d'objet lecture seule)
Titre	Disponible
Sujet	Disponible (texte brut uniquement)
Auteur	Disponible
Société	Disponible
Commentaire	Disponible

Paramètres de vue

4D Write plug-in	4D Write Pro
Mode affichage	Non importé (utiliser Document/Mode d'affichage dans le menu contextuel)
Règles	Non importé (utiliser propriété d'objet)
Cadres texte	Non importé (utiliser propriété d'objet)
En-têtes	Non importé (utiliser propriété d'objet)
Pieds de page	Non importé (utiliser propriété d'objet)
En-tête première page	Non disponible
Pied de page première page	Non disponible
Images	Non disponible
Barres défil horiz	Non importé (utiliser propriété d'objet Barre de défilement horiz)
Barres défil vert	Non importé (utiliser propriété d'objet Barre de défilement vert)
Caractères invisibles	Non importé (utiliser propriété d'objet)
Références	Non importé (utiliser ST FIXER OPTIONS)
Séparateur de colonne	Disponible (à partir de 4D v17)
Séparateur horiz	Non disponible
Séparateur vert	Non disponible
Wysiwyg	Non disponible
Zoom	Non importé (utiliser Document/Zoom dans le menu contextuel)

Paramètres de document

4D Write plug-in	4D Write Pro
Unité	Non importé (utiliser Document/Unité de mesure dans le menu contextuel)
Langage	Non disponible
Nombre de colonnes	Disponible (à partir de 4D v17)
Espacement colonnes	Disponible (à partir de 4D v17)
Veuves & orphelins	Disponible
Tabu défaut	Disponible
Tabu début	Disponible
Couleur lien	Non disponible
Couleur lien visité	Non disponible

Paramètres pagination document

4D Write plug-in	4D Write Pro
Largeur page	Disponible
Hauteur page	Disponible
Numéro première page	Disponible (à partir de 4D v16)
Première page en-tête pied différents	Disponible (à partir de 4D v16)
Pages en-têtes pieds paires impaires différents	Disponible (à partir de 4D v16)
Reliure	Disponible (à partir de 4D v16)
Pages opposées	Disponible (à partir de 4D v16)
Marges page	Disponible
En-tête marge haut	Disponible (à partir de 4D v16)
En-tête marge bas	Disponible (à partir de 4D v16)
Pied de page marge haut	Disponible (à partir de 4D v16)
Pied de page marge bas	Disponible (à partir de 4D v16)
Première page marge haut	Disponible (à partir de 4D v16)
Première page marge bas	Disponible (à partir de 4D v16)
En-tête première page marge haut	Disponible (à partir de 4D v16)
En-tête première page marge bas	Disponible (à partir de 4D v16)
Pied de page première page marge haut	Disponible (à partir de 4D v16)
Pied de page première page marge bas	Disponible (à partir de 4D v16)
Première page droite	Disponible (à partir de 4D v16)

Paramètre impression document

4D Write plug-in	4D Write Pro
Papier	Non disponible
Orientation	Non disponible
Largeur	Non disponible
Hauteur	Non disponible
Marges utilisateur	Non disponible
Echelle	Non disponible
Résolution X	Non disponible
Résolution Y	Non disponible

Images

Note de compatibilité : Sous Windows, les images au format PICT Mac OS insérées dans le document 4D Write sont importées mais ne peuvent pas être restituées dans 4D Write Pro (32 bits et 64 bits). Si vous importez un document contenant ce type d'images, il sera nécessaire de les convertir. Gardez à l'esprit que le format PICT est obsolète et ne doit plus être utilisé (cf. [Images au format PICT](#)).

4D Write plug-in	4D Write Pro
X (gauche)	(& position :absolute) (uniquement images en page)
Y (haut)	(& position :absolute) (uniquement images en page)
Largeur	Disponible
Hauteur	Disponible
Numéro de page	Disponible (à partir de 4D v16 R6)
Devant	Disponible (à partir de 4D v16 R6)
Pas première page	Disponible (à partir de 4D v16 R6)
Mode d'affichage (rognée, etc.)	Disponible
Est une expression	Disponible pour images dans le texte (à partir de 4D v16 R5)
Conserver la taille	Non importé (automatique dans 4D Write Pro)

Propriétés de caractères

4D Write plug-in

Italique
Gras
Barré
Souligné
Ombré
Exposant ou indice
Majuscules ou petites majuscules
Famille de police
Taille de police
Couleur texte
Couleur fond du texte
Couleur soulignement
Couleur barré
Couleur ombré
Propriété utilisateur
Correction orthographique (grammaire et syntaxe)
Aspect liens
Feuille de style

4D Write Pro (span)

Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Disponible
Non disponible
Non disponible
Non disponible
Non importé (styles importés mais feuilles de style caractères non disponibles)

Propriétés paragraphes

4D Write plug-in

4D Write Pro

Justification	Disponible
Interligne	Disponible
Puces	Disponible
Marge gauche	Disponible
Marge droite	Disponible
Retrait alinéa	Disponible
Style encadrement	Disponible
Couleur encadrement	Disponible
Couleur fond encadrement	Disponible
Encadrement gauche	Disponible
Encadrement droit	Disponible
Encadrement haut et inter haut	Disponible
Encadrement bas et inter bas	Disponible
Espace encadrement	Disponible
Feuille de style	Disponible
Tabulations	Disponible

Liens hypertexte

4D Write plug-in

4D Write Pro

Lien URL	Disponible (*)
Lien méthode 4D	Non disponible
Lien ouvrir document	Disponible (converti en url de fichier local)

(*) A compter de 4D v16 R4, les liens URL sont convertis en attributs modifiables. Pour plus d'informations, veuillez vous reporter au paragraphe [Gestion des URLs](#).

Expressions 4D

4D Write plug-in

4D Write Pro

Expression 4D	Disponible
Date & Heure	Disponible
Expression HTML	Non disponible
Expression RTF	Non disponible

Données texte

4D Write plug-in

4D Write Pro

Données texte principal	Disponible
Données texte en-tête	Disponible
Données texte pied de page	Disponible

Utiliser les actions standard 4D Write Pro

Vue d'ensemble

L'interface utilisateur des zones 4D Write Pro peut être gérée grâce à un grand nombre d'actions standard. Les actions standard peuvent être assignées :

- aux commandes de menu, disponibles via une **barre de menu** ou la commande **Pop up menu dynamique**,
- aux éléments de listes, pop-ups/listes déroulantes, pop-up menus hiérarchiques,
- aux boutons, cases à cocher,
- ou exécutées avec la commande **APPELER ACTION**.

Lorsqu'elles sont assignées à des objets d'interface, les actions standard gèrent les activations/désactivations de l'objet en fonction du contexte. Pour plus d'informations, référez-vous à la section **Actions standard**.

Deux catégories d'actions sont disponibles pour les zones 4D Write Pro :

- des actions spécifiques à 4D Write Pro, qui ne peuvent être utilisées qu'avec les zones 4D Write Pro,
- des **Autres actions** plus génériques de gestion des polices, expressions, du correcteur orthographique et des opérations d'édition, qui peuvent être utilisées avec les zones 4D Write Pro et les autres zones 4D. Ces actions sont détaillées dans la section **Actions standard**.

Actions 4D Write Pro

Les actions standard ci-dessous sont disponibles pour les zones 4D Write Pro :

Notes :

- Les actions 4D Write Pro n'ont pas de constantes correspondantes.
- Les actions qui affichent un menu/une liste automatique peuvent uniquement être attachées aux commandes de menus ou aux objets pop-ups/liste déroulantes et pop-up menus hiérarchiques (voir *).
- Les cases à cocher et cases à cocher 3D doivent être associées à des actions de statut, telles que "section/differentFirstPage" ou "visibleHorizontalRuler". L'option "Trois états" est prise en charge avec les cases à cocher standard uniquement.
- Un statut d'action de section est toujours égal au statut courant de la section sélectionnée (statut hérité si l'attribut est hérité de la section parente ou de l'attribut par défaut) car il doit refléter le statut de la section sélectionnée (le statut qui est actuellement appliqué à la section). Cependant, modifier un attribut de section avec une action standard surchargera uniquement l'attribut de la section sélectionnée.

(*) **Sous-menu** : Lorsqu'elles sont utilisées sans paramètres et associées à une commande de menu, un objet pop-up/Liste déroulante ou pop-up menu hiérarchique, ces actions affichent un sous-menu automatique. Par exemple, si vous assignez l'action "backgroundColor" à une commande de menu, la sélection de cette commande de menu à l'exécution affichera automatiquement les commandes du sous-menu "couleur de fond". Si vous assignez l'action "zoom" à un pop-up menu hiérarchique, il contiendra automatiquement une liste de valeurs de zoom prédéfinies. A noter que ces actions ne peuvent pas être assignées à des boutons.

(**) **showDialog** : Ajouter la chaîne **/showDialog** au nom de l'action permet d'afficher la boîte de dialogue standard associée à l'action. Par exemple, vous pouvez utiliser "paragraph/styleSheet/showDialog" pour ouvrir le dialogue permettant de saisir le nom d'une nouvelle feuille de style.

Action	Document	Paragraphe	Image	Section	En-tête	Pied	Sous-menu(*)	showDialog(**)
anchorHorizontalAlign			X				X	
anchorLayout			X				X	
anchorOrigin			X				X	
anchorPage			X				X	
anchorSection			X				X	
anchorVerticalAlign			X				X	
avoidPageBreakInsideEnabled		X						
background	X	X	X	X	X	X	X	
backgroundColor	X	X	X	X	X	X	X	X
backgroundClip	X	X	X	X	X	X	X	
backgroundImage	X	X	X	X	X	X	X	X
backgroundOrigin	X	X	X	X	X	X	X	
backgroundPositionH	X	X	X	X	X	X	X	
backgroundPositionV	X	X	X	X	X	X	X	
backgroundRepeat	X	X	X	X	X	X	X	
backgroundSizeH	X	X	X	X	X	X	X	
backgroundSizeV	X	X	X	X	X	X	X	
bookmark							X	
borderCollapse		X						
borderColor	X	X	X	X	X	X	X	X
borderRadius	X	X	X	X	X	X	X	
borders	X	X	X	X	X	X	X	
borderStyle	X	X	X	X	X	X	X	
borderWidth	X	X	X	X	X	X	X	
columnCount	X			X			X	
columnRuleColor	X			X			X	X
columnRuleStyle	X			X			X	
columnRuleWidth	X			X			X	
columns	X			X			X	
columnSpacing	X			X			X	
deleteStyleSheet		X	X				X	
direction		X					X	
doc	X						X	
dpi							X	
fontStyleWrite							X	
fontSubscript		X						
fontSuperscript		X						
footer							X	
footer/remove						X		
header							X	
header/remove					X			
htmlWYSIWIGEnabled	X							
image			X				X	
image/height			X				X	
image/verticalAlign			X				X	
insertColumnBreak	X			X				
insertImage	X							
insertPageBreak	X							
insertSectionBreak	X							
lineHeight		X					X	
listStartNumber							X	
listStyleImage								X
listStyleType		X					X	
margin	X	X	X	X	X	X	X	
merge		X						
minHeight		X	X				X	
minWidth		X	X				X	
moveToFront			X					
moveToBack			X					
newLineStyleSheet		X					X	
padding	X	X	X	X	X	X	X	
pageMode							X	
pageOrientation				X			X	
pageSize							X	
paragraph		X					X	
print	X							

refreshImages			X						
section					X			X	
section/differentFirstPage					X				
section/differentLeftRightPages					X				
section/name/showDialog					X				X
section/reset					X				
split		X							
styleSheet		X		X				X	X
styleSheets		X		X				X	
textAlign								X	
textIndent								X	
textLinethrough								X	
textLinethroughColor								X	X
textLinethroughStyle								X	
textShadow								X	
textShadowColor								X	X
textShadowOffset								X	
textTransform								X	
textUnderline								X	
textUnderlineColor								X	X
textUnderlineStyle								X	
updateStyleSheet		X		X				X	
userUnit								X	
verticalAlign		X						X	
visibleBackground	X	X		X	X	X	X		
view								X	
visibleFooters	X								
visibleHeaders	X								
visibleHiddenChars	X								
visibleHorizontalRuler	X								
visiblePageFrames	X								
widowAndOrphanControlEnabled	X								
width		X		X				X	
zoom	X							X	

anchorHorizontalAlign

Syntaxe : `anchorHorizontalAlign?value={left | center | right}`

Définit l'alignement horizontal de l'image, relatif à **anchorOrigin** pour les images en page ou à la zone de format pour les images en mode inclus. Cette action réinitialise à 0 le décalage horizontal. (non activée pour les images en ligne).

anchorLayout

Syntaxe : `anchorLayout?value={inline | front | behind}`

Définit le type de position d'ancre pour une image. Peut être utilisée pour transformer une image en ligne en image ancrée devant ou derrière le texte. Note : Si une image ancrée en transformée en image en ligne, elle est insérée au début du texte sélectionné.

anchorOrigin

Syntaxe : `anchorOrigin?value={paper-box | header-box | footer-box}`

Définit la zone relative à la position absolue d'une image ancrée. Si une image est relative à la zone d'en-tête ou de pied (header-box/footer-box) qui n'est pas visible sur une page, l'image n'est pas affichée. Cette action remet à zéro les valeurs de décalage horizontal et vertical. Mode page uniquement, non activée pour les images en ligne.

anchorPage

Syntaxe : `anchorPage?value={all | current | currentSubSection}`

Définit la ou les page(s) où l'image sélectionnée est affichée. Mode page uniquement, non activée pour les images en ligne.

anchorSection

Syntaxe : `anchorSection?value={all | current}`

Définit la ou les section(s) où l'image sélectionnée est affichée. Mode page uniquement, non activée pour les images en ligne.

anchorVerticalAlign

Syntaxe : `anchorVerticalAlign?value={top | center | bottom}`

Définit l'alignement vertical de l'image, relatif à **anchorOrigin** pour les images en page ou à la zone de format pour les images en mode inclus. Cette action réinitialise à 0 le décalage vertical (non activée pour les images en ligne).

avoidPageBreakInsideEnabled

Syntaxe : avoidPageBreakInsideEnabled
Active/Désactive le saut de page dans les paragraphes.

background

Syntaxe : {doc | paragraph | image | section | header | footer}/background
Pour les commandes de menu uniquement. Sous-menu par défaut pour tous les attributs de fond.

backgroundClip

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundClip?value={paper-box | border-box | padding-box | content-box}
Modifie la zone de découpe du fond de la cible (par défaut, la cible est le paragraphe). Paper-box est disponible uniquement avec doc et section.

backgroundColor

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundColor?value={<Css_color> | transparent}
Modifie l'arrière-plan de la cible.

backgroundImage

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundImage?value=none
Efface l'image d'arrière-plan de la cible (par défaut, la cible est le paragraphe).

backgroundOrigin

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundOrigin?value={paper-box | border-box | padding-box | content-box}
Modifie l'origine de l'image d'arrière-plan de la cible (la cible par défaut est le paragraphe). Paper-box est disponible uniquement avec doc et section.

backgroundPositionH

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundPositionH?value={left | right | center}
Modifie l'alignement horizontal de l'image d'arrière-plan de la cible (la cible par défaut est le paragraphe).
Ex: *paragraph/backgroundPositionH?value=left*

backgroundPositionV

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundPositionV?value={top | bottom | center}
Modifie l'alignement vertical de l'image d'arrière-plan de la cible (la cible par défaut est le paragraphe).
Ex: *paragraph/backgroundPositionV?value=top*

backgroundRepeat

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundRepeat?value={no-repeat | repeat | repeat-x | repeat-y}
Modifie le mode de répétition de l'image d'arrière-plan de la cible (la cible par défaut est le paragraphe).

backgroundSizeH

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundSizeH?value={<Css_length> | <Css_percentage> | auto | cover | contain}
Modifie la largeur de l'image d'arrière-plan de la cible (la cible par défaut est le paragraphe).
Ex: *paragraph/backgroundSizeH?value=100%*. *section/backgroundSizeH?value=cover*

backgroundSizeV

Syntaxe : {doc | paragraph | image | section | header | footer}/backgroundSizeV?value={<Css_length> | <Css_percentage> | auto}
Modifie la hauteur de l'image d'arrière-plan de la cible (la cible par défaut est le paragraphe).
Ex: *paragraph/backgroundSizeV?value=50%*. Ex: *section/backgroundSizeV?value=40pt*

bookmark

Syntaxe : bookmark?index=<number>
Sélectionne le Nième signet.
Ex: *bookmark?index=2 // sélectionne le deuxième signet*

borderCollapse

Syntaxe : borderCollapse
Fusionne les bordures de paragraphes et les marges intérieures (padding).

borderColor

Syntaxe : {doc | paragraph | image | section | header | footer}/{inside | outside}/{borderColor | borderColorLeft | borderColorRight | borderColorTop | borderColorBottom | borderColorLeftRight | borderColorTopBottom}?value=<Css_color>

Modifie la couleur de la bordure de la cible (la cible par défaut est le paragraphe).
Ex: *paragraph/borderColorLeft?value=green*

borderRadius

Syntaxe : {doc | paragraph | image | section | header | footer}/borderRadius?value=<Css_length>
Change le rayon pour les coins arrondis de la cible (par défaut la cible est le paragraphe).
Ex: *paragraph/borderRadius?value=4pt*

borders

Syntaxe : {doc | paragraph | image | section | header | footer}/{inside | outside}/borders
Sous-menu par défaut pour les bordures de la cible.

borderStyle

Syntaxe : {doc | paragraph | image | section | header | footer}/{inside | outside}/{borderStyle | borderStyleLeft | borderStyleRight | borderStyleTop | borderStyleBottom | borderStyleLeftRight | borderStyleTopBottom}?value={none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset}
Modifie le style de la bordure de la cible (par défaut la cible est le paragraphe).
Ex: *paragraph/borderStyleLeft?value=double*. Ex: *outside/borderStyle?value=solid //Définit en style solid la bordure extérieure du paragraphe sélectionné*. Ex: *inside/borderStyle?value=none //Supprime les bordures intérieures des paragraphes sélectionnés*

borderWidth

Syntaxe : {doc | paragraph | image | section | header | footer}/{inside | outside}/{borderWidth | borderWidthLeft | borderWidthRight | borderWidthTop | borderWidthBottom | borderWidthLeftRight | borderWidthTopBottom}?value=<Css_length>
Modifie la largeur de la bordure de la cible (par défaut la cible est le paragraphe).
Ex: *paragraph/borderWidthLeft?value=4pt*

columnCount

Syntaxe : {section}/columnCount?value={1<=nombre<=20}
Nombre de colonnes dans le document et/ou la section courant(e).
Ex: *section/columnCount?value=3*

columnRuleColor

Syntaxe : {section}/columnRuleColor?value={CSS color}
Couleur de la ligne de séparation verticale entre les colonnes du document ou de la section.
Ex : *columnRuleColor?value="#command_5FFFFFFF"*

columnRuleStyle

Syntaxe : {section}/columnRuleStyle?value={none | dotted | dashed | solid | double | groove | ridge | inset | outset}
Style de la ligne de séparation verticale entre les colonnes du document ou de la section.
Ex: *columnRuleStyle?value="solid"*

columnRuleWidth

Syntaxe : {section}/columnRuleWidth?value={CSS length}
Largeur de la ligne de séparation verticale entre les colonnes du document ou de la section
Ex: *columnRuleWidth?value="2pt"*

columns

Syntaxe : {section}/columns

Pour les commandes de menu uniquement. Crée automatiquement un sous-menu **Colonnes** complet avec tous les sous-menus d'action de colonne (hormis **insertColumnBreak**)

columnSpacing

Syntaxe : {section}/columnSpacing?value={CSS length}
Espacement entre deux colonnes dans le document et/ou la section courant(e).
Ex: *columnSpacing?value="1cm"*

deleteStyleSheet

Syntaxe : {paragraph | image}/deleteStyleSheet?index={1<= number <= number of target style sheets}
Supprime la Nième feuille de style parmi les feuilles de style stockées pour la cible sélectionnée (par défaut la cible est le paragraphe). Le nom de l'élément de menu est remplacé, au moment de l'exécution, par le nom de la feuille de style sur le point d'être supprimée.

direction

Syntaxe : direction?value={ltr | rtl}
Sens de l'écriture du paragraphe (ltr : left to right - rtl : right to left)

doc

Syntaxe : doc

Sous-menu par défaut pour la disposition du corps du document (attributs du mode inclus ou attributs par défaut des sections - mais pour la marge réservée au mode inclus seulement)

dpi

Syntaxe : dpi?value={auto | 72 <= integer <= 72*4}

Modifie le dpi de la vue courante (indépendant de l'attribut de document [wk_dpi](#) utilisé en interne pour la conversion pixels<->points).

Ex: *dpi?value=72*

fontStyleWrite

Syntaxe : fontStyleWrite

Pour les commandes de menu uniquement. Affiche le sous-menu Style de police par défaut de 4D Write Pro

fontSubscript

Syntaxe : fontSubscript

Bascule l'attribut de police indice.

fontSuperscript

Syntaxe : fontSuperscript

Bascule l'attribut de police exposant.

footer

Syntaxe : footer

Affiche le sous-menu Pied de page.

footer/remove

Syntaxe : footer/remove

Supprime le pied de page sélectionné.

header

Syntaxe : header

Affiche le sous-menu En-tête.

header/remove

Syntaxe : header/remove

Supprime l'en-tête sélectionné.

htmlWYSIWIGEnabled

Syntaxe : htmlWYSIWIGEnabled

Bascule du mode html au mode wysiwyg.

image

Syntaxe : image

Affiche le sous-menu Image.

image/height

Syntaxe : image/height?value={<Css_length> | auto}

Hauteur de l'image.

Ex: *image/height?value=50pt*. Pour image/width, voir *width*.

image/verticalAlign

Syntaxe : image/verticalAlign?value={top | middle | bottom | baseline | super | sub}

Alignement vertical de l'image.

Ex: *image/verticalAlign?value=super*

insertImage

Syntaxe : insertImage

Ouvre une boîte de dialogue de sélection d'image et insère l'image sélectionnée (le cas échéant) en tant que caractère dans la zone.

insertColumnBreak

Syntaxe: insertColumnBreak

Inserts a column break at the selection.

insertPageBreak

Syntaxe : insertPageBreak

Insère un saut de page à l'endroit sélectionné.

insertSectionBreak

Syntaxe : insertSectionBreak

Insère un saut de section à l'endroit sélectionné.

lineHeight

Syntaxe : lineHeight?value={<Css_length> | <Css_percentage>}

Hauteur de ligne du paragraphe.

Ex: *lineHeight?value=120%*

listStartNumber

Syntaxe : listStartNumber?value={<number> (longint) | auto}

Numéro de départ de la liste.

Ex: *listStartNumber?value=10*

listStyleImage

Syntaxe : listStyleImage/showDialog

Ouvre un dialogue d'ouverture de fichier pour choisir une image à afficher en tant que puce pour une liste ; définit également un élément de liste sous forme de disque – style à utiliser si l'image n'est pas trouvée.

listStyleType

Syntaxe : listStyleType?value={none | disc | circle | square | hollow-square | diamond | club | decimal | decimal-leading-zero | lower-latin | lower-roman | upper-latin | upper-roman | lower-greek | decimal-greek | armenian | georgian | hebrew | hiragana | katakana | cjk-ideographic}

Type de style de liste de paragraphes.

margin

Syntaxe : {doc | paragraph | image | section | header | footer/}{inside | outside/}{margin | marginLeft | marginRight | marginTop | marginBottom | marginLeftRight | marginTopBottom}?value={<Css_length> | auto}

Modifie la marge de la cible (la cible par défaut est le paragraphe).

Ex: *margin?value=4pt //sets all paragraph margins to 4pt.*

Ex: *outside/margin?value=4pt //fixe la marge extérieure du paragraphe à 4pt.*

Ex: *doc/marginLeft?value=1cm //fixe la marge gauche à 1 cm pour le document utilisé en mode inclus.*

Ex: *section/marginLeft?value=1cm //fixe la marge gauche à 1 cm pour les pages de la première section sélectionnée*

merge

Syntaxe : merge

Fusionne les paragraphes.

minHeight

Syntaxe : {paragraph | image}/minHeight?value=<Css_length>

Hauteur minimum de la cible (la cible par défaut est le paragraphe).

Ex: *paragraph/minHeight?value=50pt*

minWidth

Syntaxe : {paragraph | image}/minWidth?value=<Css_length>

Largeur minimum de la cible (la cible par défaut est le paragraphe).

Ex: *paragraph/minWidth?value=50pt*

moveToBack

Syntaxe : moveToBack

Place l'image à l'arrière plan. (non activée pour les images en ligne)

moveToFront

Syntaxe : moveToFront

Place l'image au premier plan. (non activée pour les images en ligne)

newLineStyleSheet

Syntaxe : newLineStyleSheet?index={1<=number<=number of paragraph stylesheets | newLineStyleSheet?value=auto}

Les éléments du paragraphe sélectionné utiliseront la Nième feuille de style pour les nouveaux paragraphes créés par Retour Chariot ou lors d'un fractionnement de paragraphes ; si value=auto, les nouveaux paragraphes utiliseront la même feuille de style (fonctionnement par défaut). (le libellé de la ligne de menu est remplacé, à l'exécution, par le nom de la feuille de style).

padding

Syntaxe : {doc | paragraph | image | section | header | footer/}{inside | outside/}{padding | paddingLeft | paddingRight | paddingTop | paddingBottom | paddingLeftRight | paddingTopBottom }?value=<Css_length>
Modifie le padding de la cible (la cible par défaut est le paragraphe)

pageMode

Syntaxe : pageMode?value={embedded | page | draft}
Change le mode d'affichage. Ex: *pageMode?value=page*

pageOrientation

Syntaxe : {section/}pageOrientation?value={landscape | portrait}
Modifie le document ou l'orientation de la page de la première section sélectionnée (si la section n'est pas spécifiée, elle s'appliquera à toutes les sections, sinon à la première section sélectionnée).
Ex: *pageOrientation?value=portrait //change l'orientation en Portrait pour tout le document.*
Ex: *section/pageOrientation?value=landscape //change l'orientation en Paysage pour la première section sélectionnée*

pageSize

Syntaxe : pageSize?index=<number>
Modifie la taille de la page du document : <number> est la Nième taille de page dans la liste des tailles de page disponibles dans l'imprimante courante.
Ex: *pageSize?index=1*

paragraph

Syntaxe : paragraph
Sous-menu par défaut pour le paragraphe.

print

Syntaxe : print
Imprime le document en mode Page avec les paramètres de vue courants de la zone 4D Write Pro qui a le focus.

refreshImages

Syntaxe : refreshImages
Recharge les images du réseau pour les images avec URLs.

section

Syntaxe : section
Sous-menu par défaut de disposition de la page pour la première section sélectionnée

section/differentFirstPage

Syntaxe : section/differentFirstPage
La section a ou non une première page différente

section/differentLeftRightPages

Syntaxe : section/differentLeftRightPages
La section a ou non des pages droites et gauches différentes

section/name/showDialog

Syntaxe : section/name/showDialog
Affiche un dialogue permettant d'entrer le nom de la première section sélectionnée

section/reset

Syntaxe : section/reset
Réinitialise tous les attributs de la première section sélectionnée pour les remplacer par les attributs par défaut - qui sont hérités des attributs du document pour l'arrière-plan, la bordure et les marges intérieures (même pour le mode inclus) et de 2.5 cm pour les marges des pages. Les propriétés de colonnes de la première section de la sélection sont également réinitialisées (seules les propriétés de section par défaut sont appliquées).

split

Syntaxe : split
Fractionne les paragraphes.

styleSheet

Syntaxe : {paragraph | image}/styleSheet?value={1<=number<=number of target style sheets}
Applique la Nième feuille de style aux éléments sélectionnés de la cible. Avec la boîte de dialogue, la nouvelle feuille de style utilisera les attributs courants de la première cible sélectionnée (par défaut la cible est le paragraphe).
Ex: *styleSheet?value=1 //applique la première feuille de style de paragraphes aux paragraphes sélectionnés et la définit comme feuille de style.*

styleSheets

Syntaxe : {paragraph | image}/styleSheets

Sous-menu par défaut pour la modification des feuilles de style de la cible

textAlign

Syntaxe : textAlign?value={left | right | center | justify | initial}

Alignement du texte du paragraphe (initial=utilise right pour la direction rtl ou left pour la direction ltr)

textIndent

Syntaxe : textIndent?value=<Css_length> (peut être négative)

Indentation de la première ligne du paragraphe.

Ex: *textIndent?value=2cm*

textLinethrough

Syntaxe : textLinethrough

Pour les commandes de menu uniquement. Sous-menu Texte barré (avec style et couleur).

textLinethroughColor

Syntaxe : textLinethroughColor?value={<Css_color> | currentColor}

Ex: *textLinethroughColor?value=red*

textLinethroughStyle

Syntaxe : textLinethroughStyle?value={none | solid | dotted | dashed | double | semi-transparent | word}

Ex: *textLinethroughStyle?value=dotted*

textShadow

Syntaxe : textShadow

Pour les commandes de menu uniquement. Sous-menu par défaut pour le texte ombré (avec couleur et décalage)

textShadowColor

Syntaxe : textShadowColor?value={<Css_color> | none}

Modifie la couleur d'ombre des caractères.

Ex: *textShadowColor?value=green*

textShadowOffset

Syntaxe : textShadowOffset?value=<Css_length>

Définit le décalage de l'ombré. Seule une valeur en points est possible.

Ex: *textShadowOffset?value=2pt*

textTransform

Syntaxe : textTransform?value={none | capitalize | uppercase | lowercase | small-uppercase}

Formate le texte.

Ex: *textTransform?value=capitalize*

textUnderline

Syntaxe : textUnderline

Pour les commandes de menu uniquement. Sous-menu Souligné (avec style et couleur)

textUnderlineColor

Syntaxe : textUnderlineColor?value={<Css_color> | currentColor}

Ex: *textUnderlineColor?value=red*

textUnderlineStyle

Syntaxe : textUnderlineStyle?value={none | solid | dotted | dashed | double | semi-transparent | word}

Ex: *textUnderlineStyle?value=dotted*

updateStyleSheet

Syntaxe : {paragraph | image}/updateStyleSheet

Mise à jour de la première feuille de style de la cible sélectionnée (la cible par défaut est le paragraphe)

userUnit

Syntaxe : userUnit?value={cm | mm | in | pt}

Modifie l'unité de mise en page du document (sera répercutée sur les règles)

verticalAlign

Syntaxe : `verticalAlign?value={top | middle | bottom}`

Alignement vertical du paragraphe.

Ex: `verticalAlign?value=top`. L'effet dépend de la valeur **minHeight** du paragraphe.

view

Syntaxe : `view`

Pour les commandes de menu uniquement. Sous-menu par défaut pour les paramétrages de vue

visibleBackground

Syntaxe : `visibleBackground`

Affiche/cache l'arrière-plan

visibleFooters

Syntaxe : `visibleFooters`

Affiche/cache les pieds de page

visibleHeaders

Syntaxe : `visibleHeaders`

Affiche/cache les en-têtes.

visibleHiddenChars

Syntaxe : `visibleHiddenChars`

Affiche/cache les caractères cachés.

visibleHorizontalRuler

Syntaxe : `visibleHorizontalRuler`

Affiche/cache la règle horizontale.

visiblePageFrames

Syntaxe : `visiblePageFrames`

Affiche/cache les cadres des pages.

widowAndOrphanControlEnabled

Syntaxe : `widowAndOrphanControlEnabled`

Bascule du mode de contrôle des veuves et orphelines.

width

Syntaxe : `{paragraph | image}/width?value=<Css_length> | auto`

Largeur de la cible (uniquement pour le paragraphe et l'image - la cible par défaut est le paragraphe).

Ex: `image/width?value=50pt`

zoom

Syntaxe : `zoom?value={25% <= percentage in Css <= 400%}`

Modification du zoom du document.

Ex: `zoom?value=120%`

Autres actions

De nombreuses autres actions standard proposées avec les formulaires 4D peuvent être utilisées dans les zones 4D Write Pro :

- **Actions d'édition**, comme les actions Copier/Coller.
- Actions sur la **Police**, telles que **fontBold** ou **fontSize**.
- Actions sur les **Expressions dynamiques**, permettant de gérer l'insertion d'expressions.
- Actions du **Correcteur orthographique**.

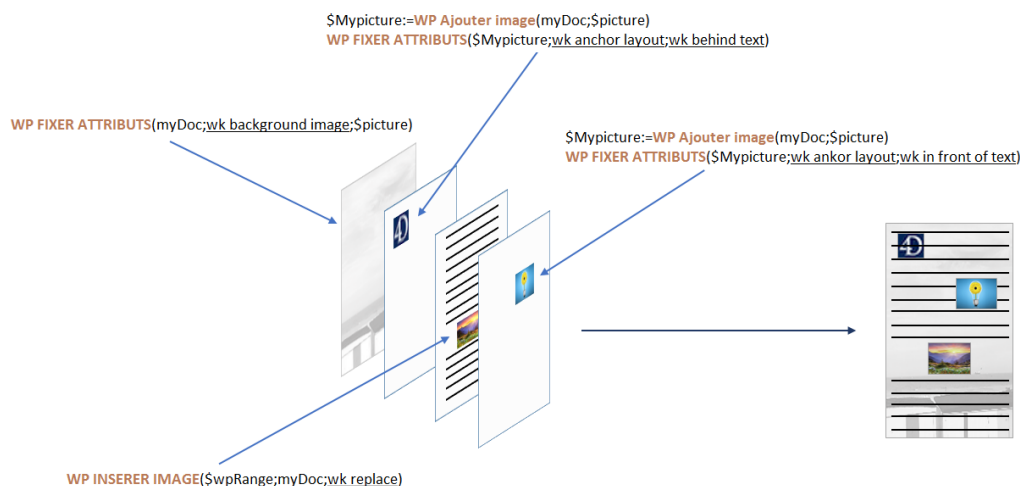
Manipuler des images

Ajouter des images

Ajouter des images dans un document 4D Write Pro peut se faire de multiples façons et dépend de vos besoins :

- pour ajouter une **image d'arrière-plan**, utilisez l'attribut `wk_background_image` ou `wk_background_image_url` avec la commande **WP FIXER ATTRIBUTS**
- pour ajouter une **image en ligne**, *i.e.* insérée dans le texte comme un caractère, utilisez la commande **WP INSERER IMAGE** ou la commande **ST INSERER EXPRESSION**
- pour ajouter une **image ancrée** dans la page (derrière ou devant le texte), utilisez la commande **WP Ajouter image** et fixez l'attribut `wk_anchor_layout`.

La façon dont vous ajoutez une image détermine la couche où elle est positionnée, comme le montre le diagramme ci-dessous :



Images en arrière-plan

Des images peuvent être mises en place en tant qu'arrière-plan de documents 4D Write Pro ou de partie du document (tableau, paragraphe, sections, en-têtes/pieds, etc.).

Voici un exemple montrant deux façons différentes d'utiliser une image comme arrière-plan d'un document :



Les images d'arrière-plan peuvent être placées soit par programmation, soit via le menu contextuel. Pour plus d'informations, voir la section **Fond** dans l'article **Utiliser une zone 4D Write Pro**.

Positionnement et affichage des images ancrées

Les images ancrées sont ajoutées avec une position absolue, devant/derrière le texte, et ancrées à la page ou à des parties spécifiques d'un document (*i.e.*, en-tête, pied de page, section). La définition d'une position absolue pour une image se fait grâce aux commandes **WP Ajouter image** et **WP FIXER ATTRIBUTS**.

Les positions d'images ancrées peuvent être modifiées avec les attributs spécifiques et / ou les actions standard suivants :

Propriété (constante)	Action standard
wk_anchor_layout	anchorLayout
wk_anchor_horizontal_offset	
wk_anchor_horizontal_align	anchorHorizontalAlign
wk_anchor_vertical_offset	
wk_anchor_vertical_align	anchorVerticalAlign
wk_anchor_origin	anchorOrigin
wk_anchor_page	anchorPage
wk_anchor_section	anchorSection
	moveToBack
	moveToFront

Les images ancrées sont affichées uniquement en mode Page. Elles ne sont pas affichées si :

- elles sont ancrées à un en-tête ou un pied de page qui n'est pas visible ;
- le mode d'affichage est le mode Brouillon ;
- elles sont centrées ou ancrées aux sections, et l'option **Afficher HTML WYSIWYG** est cochée ;
- l'option "Afficher l'arrière-plan" n'est pas sélectionnée.

Propriétés des images

Toutes les images ont des propriétés (attributs) telles que la hauteur, la largeur, les bordures, etc., qui peuvent être lues ou fixées via le langage 4D Write Pro (**WP LIRE ATTRIBUTS** et **WP FIXER ATTRIBUTS**) ou les actions standard.

- La liste complète des propriétés utilisables pour les images est disponible sur la page **Attributs 4D Write Pro**.
- La section **Image** contient les attributs spécifiques aux images seules.
- La page **Utiliser les actions standard 4D Write Pro** liste aussi les propriétés image disponibles.

Référence d'image ou URL d'image

Vous pouvez travailler avec des références d'images (variable, champ ou expression image) ou des URLs d'images (texte contenant l'adresse locale ou réseau de l'image).

Deux ensembles d'attributs vous permettent de définir si vous voulez fixer ou lire une référence ou un URL d'image :

- [wk_image](#), [wk_background_image](#), [wk_list_style_image](#) : pour lire ou fixer des références d'images.
Lorsque vous lisez une image à l'aide d'un de ces attributs, vous recevez une image 4D quelle que soit la manière dont l'image a été initialement définie.
- [wk_image_url](#), [wk_background_image_url](#), [wk_list_style_image_url](#) : pour lire ou fixer des URLs d'images.
Lorsque vous lisez une image à l'aide d'un de ces attributs, vous recevez un texte. Si l'image a été définie via une référence (par exemple une variable image), vous recevez un URI local suivi de l'image encodée en Base64.

Exemple :

```
$range:=WP Lire selection(WPArea) //lire l'image sélectionnée par l'utilisateur
$range:=WP Creer plage images($range) //creer une plage
$url:="http://doc.4d.com/image/logo/poweredby4D_web.png"

WP FIXER ATTRIBUTS($range;wk_image;$url) //définir une image via un URL

//lire l'image
C_IMAGE(vPictureGet)
WP LIRE ATTRIBUTS($range;wk_image;vPictureGet) //vPictureGet contient une image
C_TEXTE(vPictureURLGet)
WP LIRE ATTRIBUTS($range;wk_image_url;vPictureURLGet) //vPictureURLGet=$url
```

Récupérer des images

Les commandes ci-dessous peuvent être utilisées pour récupérer des images :

- **WP Creer plage images** - s'applique uniquement aux images en ligne
- **WP Lire selection** - s'applique uniquement aux images sélectionnées par l'utilisateur















































Actions souris / clavier

Les images peuvent être manipulées via la souris ou le clavier. Les actions disponibles incluent :

- **Sélectionner**
 - Les images en avant-plan peuvent être sélectionnées en cliquant dessus ;
 - Les images d'arrière-plan peuvent être sélectionnées avec CTRL+ clic (Windows) ou COMMAND+clic (Mac) ;
 - Une seule image avec une position absolue peut être sélectionnée à la fois : la sélection multiple n'est pas possible.
- **Déplacer**
 - Les images peuvent être déplacées et redimensionnées à l'aide de la souris. (Les images en ligne ne peuvent pas être déplacées en dehors du document).
- **Insérer**

- Si des images avec position absolue sont collées en mode Brouillon, elles sont affichées en ligne au début de la sélection courante au clavier.
- **Couper/Coller/Copier**
 - Les images peuvent être copiées dans le presse-papiers, ou coupées/collées, lorsqu'elles sont sélectionnées.
 - Coller une image avec une position absolue en mode Page, collera l'image dans la première page sélectionnée et l'ancrera à cette page, quels que soient les paramétrages d'ancrage de la page d'origine ; les images collées conserveront leurs décalages (offsets), sauf si elles sont collées dans un mode différent de celui d'origine (par exemple, si l'image est copiée en mode Page et collée en mode Inclus, les décalages sont réinitialisés à 0).
- **Supprimer** - Les images peuvent être supprimées lorsqu'elles sont sélectionnées avec :
 - la touche SUPPRIMER, ou
 - le raccourci-clavier pour l'action Effacer

4D Write Pro - Langage

-  A propos des objets 4D Write Pro
-  Utiliser les commandes du thème Objets (Formulaires)
-  Utiliser les commandes du thème Texte multistyle
-  Accéder au contenu des documents par programmation
-  WP Ajouter image
-  WP Créer page
-  WP Créer page images
-  WP Créer page paragraphes
-  WP Créer page tableaux Nouveauté 17.0
-  WP CREER SIGNET
-  WP EXPORTER DOCUMENT
-  WP EXPORTER VARIABLE
-  WP FIXER ATTRIBUTS
-  WP FIXER CADRE
-  WP FIXER TEXTE Nouveauté 17.0
-  WP Importer document
-  WP IMPRIMER
-  WP INSERER DOCUMENT
-  WP INSERER IMAGE
-  WP INSERER RUPTURE Modifié 17.0
-  WP Insérer tableau
-  WP LIRE ATTRIBUTS
-  WP Lire cadre
-  WP Lire corps
-  WP Lire element par ID Nouveauté 17.0
-  WP Lire elements Nouveauté 17.0
-  WP Lire entete
-  WP Lire pied
-  WP Lire page signet
-  WP Lire position Nouveauté 17.0
-  WP Lire selection
-  WP LIRE SIGNETS
-  WP Lire texte Nouveauté 17.0
-  WP Nombre de pages
-  WP Nouveau
-  WP REINITIALISER ATTRIBUTS
-  WP SELECTIONNER
-  WP Style pris en charge
-  WP SUPPRIMER SIGNET
-  WP Tableau ajouter ligne
-  WP Tableau lire cellules Modifié 17.0
-  WP Tableau lire colonnes Modifié 17.0
-  WP Tableau lire lignes Modifié 17.0
-  WP UTILISER PARAMETRES IMPRESSION
-  Constantes 4D Write Pro
-  Attributs 4D Write Pro

A propos des objets 4D Write Pro

Objet 4D Write Pro

Dans les applications 4D, les documents 4D Write Pro sont stockés dans des variables de type *objet* (pour plus d'informations, reportez-vous à la commande **C_OBJET**).

Une fois référencé en mémoire, un objet 4D Write Pro peut être :

- affiché et modifié dans un formulaire 4D via un objet de formulaire de type "zone 4D Write Pro" ; il suffit simplement que la zone de formulaire ait le même nom que la variable objet (cf. **Définir une zone 4D Write Pro**).
- manipulé par programmation à l'aide de commandes de différents thèmes (voir ci-dessous).
- exporté vers un document HTML sur disque à l'aide de la commande **WP EXPORTER DOCUMENT**.

Commandes de manipulation des objets 4D Write Pro

Vous pouvez gérer les objets 4D Write Pro à l'aide de commandes de plusieurs thèmes :

- Les commandes dédiées du thème **4D Write Pro** ; ces commandes sont décrites dans ce chapitre.
- Les commandes **Objets (Formulaires)** pour modifier le formatage et les propriétés des objets 4D Write Pro ; pour plus d'informations, veuillez vous référer à la section **Utiliser les commandes du thème Objets (Formulaires)**.
- Les commandes **Texte multistyle** pour modifier le contenu des objets 4D Write Pro ; pour plus d'informations, veuillez vous référer à la section **Utiliser les commandes du thème Texte multistyle**.
- Les commandes **Objets (Langage)** pour travailler avec les attributs d'objet 4D Write Pro ; pour plus d'informations, veuillez vous référer à la section **Stocker les documents 4D Write Pro dans des champs objet 4D**.

Utiliser les commandes du thème Objets (Formulaires)

Les commandes suivantes prennent en charge les objets de formulaire 4D Write Pro :

Commande	Commentaire
OBJET DUPLIQUER	
OBJET Lire correction orthographique/OBJET FIXER CORRECTION ORTHOGRAPHIQUE	
OBJET Lire style bordure/OBJET FIXER STYLE BORDURE	
OBJET Lire menu contextuel/OBJET FIXER MENU CONTEXTUEL	
OBJET LIRE COORDONNEES/OBJET FIXER COORDONNEES	
OBJET Lire source donnees/OBJET FIXER SOURCE DONNEES	
OBJET LIRE OPTIONS GLISSER DEPOSER/OBJET FIXER OPTIONS GLISSER DEPOSER	
OBJET Lire activation/OBJET FIXER ACTIVATION	
OBJET Lire saisissable/OBJET FIXER SAISSISSABLE	
OBJET LIRE EVENEMENTS/OBJET FIXER EVENEMENTS	
OBJET Lire rectangle focus invisible/OBJET FIXER RECTANGLE FOCUS INVISIBLE	
OBJET Lire police/OBJET FIXER POLICE	Appliqué à la sélection (le cas échéant)
OBJET Lire taille police/OBJET FIXER TAILLE POLICE	Appliqué à la sélection (le cas échéant)
OBJET Lire style police/OBJET FIXER STYLE POLICE	Appliqué à la sélection (le cas échéant)
OBJET Lire alignement horizontal/OBJET FIXER ALIGNEMENT HORIZONTAL	Appliqué à la sélection (le cas échéant). Prise en charge de la constante wk_justify pour les zones 4D Write Pro
OBJET LIRE REDIMENSIONNEMENT/OBJET FIXER REDIMENSIONNEMENT	
OBJET FIXER COULEUR	Appliqué à la sélection (le cas échéant)
OBJET LIRE COULEURS RVB/OBJET FIXER COULEURS RVB	Appliqué à la sélection (le cas échéant)
OBJET Lire type	
OBJET Lire alignement vertical/OBJET FIXER ALIGNEMENT VERTICAL	Alignement vertical des paragraphes : a un effet lorsque la hauteur du paragraphe est supérieure à la hauteur du texte du paragraphe
OBJET Lire visible/OBJET FIXER VISIBLE	
OBJET Est un texte style	Retourne Vrai
OBJET DEPLACER	
OBJET LIRE DEFILEMENT/OBJET FIXER DEFILEMENT	
OBJET LIRE TAILLE SOUS FORMULAIRE	
OBJET Lire nom	
OBJET Lire pointeur	

Les commandes OBJET non listées ci-dessus ne sont pas applicables aux zones 4D Write Pro.

Exemple

Vous voulez centrer l'alignement horizontal d'une zone 4D Write Pro :

```
Au cas ou
:(Evenement formulaire=Sur clic)
  OBJET FIXER ALIGNEMENT HORIZONTAL(*;"myObject";Aligné au centre)
  hAlignLeft:=0 //stocker la propriété pour affichage ultérieur
  hAlignRight:=0
Fin de cas
```

Utiliser les commandes du thème Texte multistyle

Les commandes suivantes prennent en charge les zones 4D Write Pro. Vous pouvez les utiliser soit avec des noms d'objets de formulaire 4D Write Pro (chaîne), soit avec des références de document 4D Write Pro (variable de type *Objet*) :

Commande	Commentaire
ST CALCULER EXPRESSIONS	
ST FIGER EXPRESSIONS	
ST LIRE ATTRIBUTS/ST FIXER ATTRIBUTS	
ST Lire type contenu	Un type supplémentaire (6) a été ajouté pour le contenu de type image
ST Lire expression / ST INSERER EXPRESSION	A partir de 4D v16, les expressions sont filtrées par défaut dans les documents 4D Write Pro et doivent être explicitement autorisées. Pour plus d'information, voir la section Filtrer les expressions contenues dans un document 4D Write Pro . A compter de 4D v16 R5, vous pouvez insérer des expressions image (voir ci-dessous).
ST LIRE OPTIONS/ST FIXER OPTIONS	
ST Lire texte brut/ST FIXER TEXTE BRUT	A partir de 4D v17, vous pouvez lire/écrire du texte brut dans les documents 4D Write Pro à l'aide des commandes WP Lire texte et WP FIXER TEXTE .
ST Lire texte / ST FIXER TEXTE	
ST LIRE URL / ST INSERER URL	A compter de 4D v16 R4, vous pouvez insérer et gérer directement des URLs éditables dans n'importe quelle page 4D Write Pro (y compris de type image ou cellule) via les commandes WP FIXER ATTRIBUTS / WP LIRE ATTRIBUTS et l'attribut <u>wk link url</u> .

Exemple

Vous voulez remplacer la sélection dans une zone 4D Write Pro par le contenu d'une variable :

```
C_TEXTE(nomComplet)

Au cas ou
:(Evenement formulaire=Sur clic)
  ST INSERER EXPRESSION(myArea;"nomComplet";ST Début sélection;ST Fin sélection)
Fin de cas
```

Insérer des expressions de page et de document

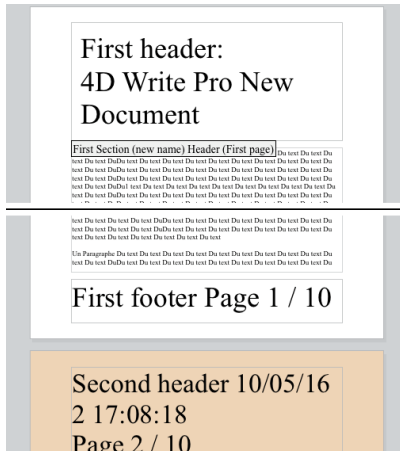
Vous pouvez insérer des expressions spéciales relatives aux attributs de document ou aux attributs de page en utilisant la commande **ST INSERER EXPRESSION**.

Expression	Disponible dans	Type	Description
\$wp_title	toutes les parties du document	Texte	Titre défini dans l'attribut <u>wk title</u>
\$wp_author	toutes les parties du document	Texte	Auteur défini dans l'attribut <u>wk author</u>
\$wp_subject	toutes les parties du document	Texte	Sujet défini dans l'attribut <u>wk subject</u>
\$wp_company	toutes les parties du document	Texte	Société définie dans l'attribut <u>wk company</u> .
\$wp_notes	toutes les parties du document	Texte	Notes définies dans l'attribut <u>wk notes</u>
\$wp_dateCreation	toutes les parties du document	Date	Date de création définie dans l'attribut <u>wk date creation</u>
\$wp_dateModified	toutes les parties du document	Date	Date de modification définie dans l'attribut <u>wk date modified</u>
\$wp_pageNumber	En-tête & Pied de page - si ailleurs renvoie une erreur	Entier long	Numéro de page tel que défini soit à partir du début du document (fonctionnement par défaut) soit à partir du début de la section
\$wp_pageCount	En-tête & Pied de page - si ailleurs renvoie une erreur	LongInt	Nombre total de pages

Pour insérer une expression, vérifiez bien que le curseur se situe dans la zone appropriée (en-tête, pied ou corps du document) et appelez la commande **ST INSERER EXPRESSION**. Par exemple, pour insérer le numéro de page dans la zone de pied sélectionnée :

```
ST INSERER EXPRESSION(*;"4DWParea";"$wp_pageNumber")
```

La présentation de document ci-dessous peut être définie par exemple :



Insérer des expressions image

Vous pouvez insérer des expressions 4D qui retournent des images dans vos zones 4D Write Pro à l'aide de la commande **ST INSERER EXPRESSION**. Les expressions peuvent être des variables, des champs ou des méthodes 4D(*).

(*)N'oubliez pas qu'appeler une méthode 4D sous forme d'expression requiert que la méthode soit explicitement autorisée au préalable (voir **Filtrer les expressions contenues dans un document 4D Write Pro**).

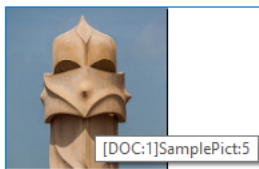
Exemples :

```
//Insertion d'une variable image
ST INSERER EXPRESSION(*;"WParea";"vPicture";ST Début sélection;ST Fin sélection)

//Insertion d'un champ
ST INSERER EXPRESSION(*;"WParea";"[DOC]SamplePict";ST Début sélection;ST Fin sélection)

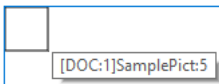
//Insertion d'une méthode 4D
ST INSERER EXPRESSION(*;"WParea";"M_ComputeChart";ST Début sélection;ST Fin sélection)
```

Vous pouvez visualiser la référence de l'expression dans l'info-bulle de l'image :



Tous les attributs d'images peuvent être appliqués aux expressions image (l'attribut wk image peut être lu uniquement). Notez cependant que puisque les images ont des attributs spécifiques, 4D Write Pro doit évaluer l'expression au moins une fois afin de détecter que son résultat est une image et donc la gérer comme une expression image. Cela signifie que lorsqu'une expression image est insérée avec **ST INSERER EXPRESSION**, la commande **ST CALCULER EXPRESSIONS** doit être appelée avant toute définition d'attribut image.

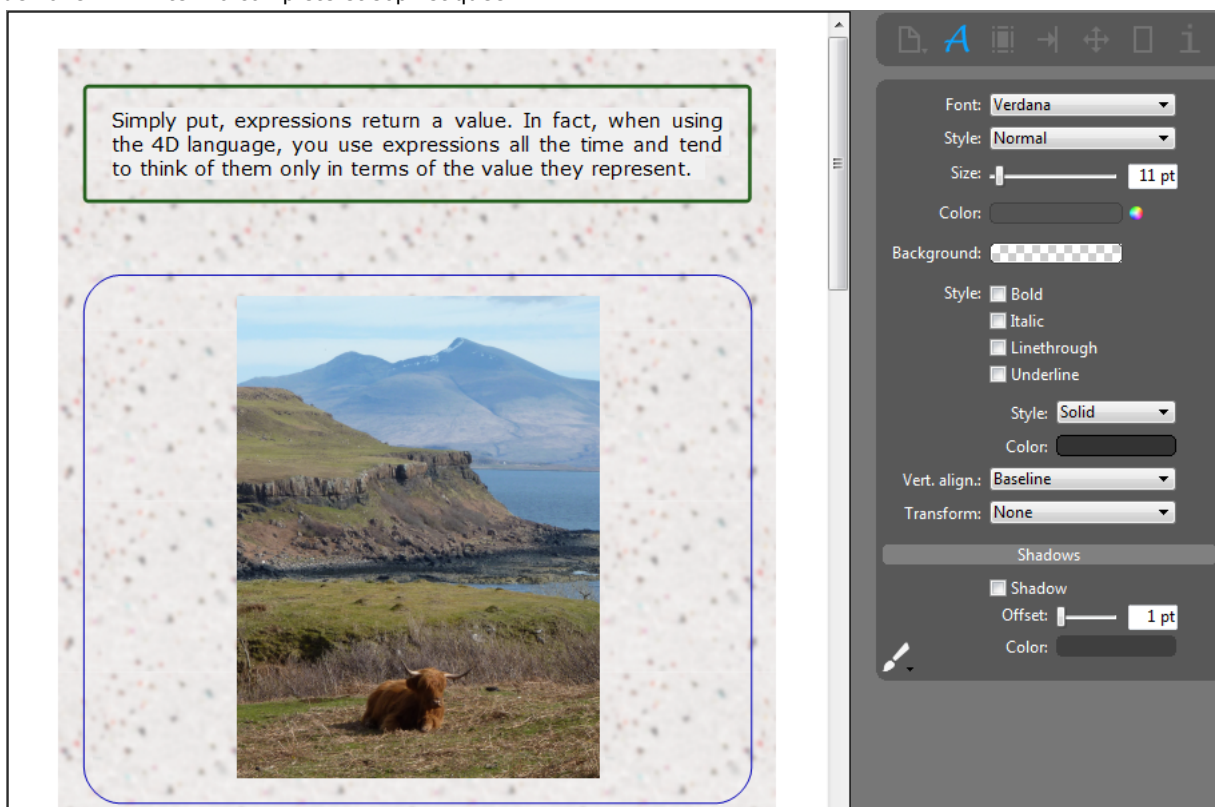
Si l'image résultant de l'expression ne peut pas être évaluée ou chargée, 4D Write Pro affiche le cadre noir par défaut indiquant un url indéfini ou non trouvé :



Accéder au contenu des documents par programmation

4D Write Pro propose un ensemble complet de commandes de langage vous permettant de modifier les attributs de texte et d'image de vos documents. Grâce à ces fonctionnalités, les développeurs 4D peuvent concevoir leurs propres interfaces (à l'aide de boutons, menus, cases à cocher, etc.) pour les documents 4D Write Pro. Les commandes peuvent s'appliquer à tout le document ou uniquement à certaines parties, basées soit sur la sélection utilisateur, soit sur des sélections créées par programmation. Les attributs disponibles incluent les unités du document, les marges internes et externes, les arrière-plans, les propriétés des paragraphes, couleurs, polices, styles, ainsi que les propriétés des images.

Par exemple, l'objet de la bibliothèque **Zone 4D Write Pro** s'appuie sur toutes ces fonctionnalités pour proposer une interface de zone 4D Write Pro complète et sophistiquée :



Gestion des pages

4D Write Pro permet de sélectionner et de manipuler par programmation les contenus des documents. Comme le contenu sélectionné peut inclure du texte, des images, des tableaux, etc, et également des balises (invisibles) de formatage, 4D Write Pro travaille avec des objets appelés *pages*.

Une page est un objet qui représente une portion de document 4D Write Pro. Elle peut être utilisée pour désigner les éléments à sélectionner ou manipuler des attributs sur une partie du document (à l'aide des commandes **WP LIRE ATTRIBUTS** et **WP FIXER ATTRIBUTS**).

Il existe différents types de pages. Vous pouvez connaître le type d'une page à l'aide de l'attribut `wk_type` (en lecture seule). Chaque page contient plusieurs attributs privés qui la définissent :

Constante	Valeur	Comment
wk end	end	(Attribut de page en lecture seule) Position fin de page. Valeur : entier long
wk id	id	ID interne de l'élément. Valeur : Chaîne
wk owner	owner	(Attribut en lecture seule) Possesseur de la page. Valeur : objet
wk start	start	(Attribut de page en lecture seule) Position début de page. Valeur : entier long
		(Attribut de page en lecture seule) Type d'objet 4D Write Pro. Valeurs possibles :
		<ul style="list-style-type: none">• <code>wk_type_default</code> : Page sans type défini• <code>wk_type_paragraph</code> : Page de type paragraphe• <code>wk_type_image</code> : Image (ancrée ou en ligne)• <code>wk_type_image_anchored</code> : Image ancrée• <code>wk_type_image_inline</code> : Image en ligne• <code>wk_type_container</code> : En-tête ou pied, par exemple• <code>wk_type_table</code> : Référence de tableau• <code>wk_type_table_row</code> : Référence de ligne de tableau• <code>wk_type_table_cell</code> : Référence de cellule de tableau• <code>wk_type_table_column</code> : Référence de colonne de tableau (uniquement pour page colonne)• <code>wk_type_body</code> : Référence de corps (body)
wk type	type	

Note : Les plages liées aux tableaux (plages de cellules, de colonnes et de lignes) contiennent des attributs spécifiques (cf. paragraphe ci-dessous).

Plusieurs commandes permettent de définir des plages :

- **WP Créer plage**(*wpZone ; débutObjPlage ; finObjPlage*) -> *objPlage* : retourne une nouvelle plage correspondant aux limites que vous avez passées en paramètres.
- **WP Lire sélection**({*; } *wpZone*) -> *objPlage* : retourne une nouvelle plage correspondant à la sélection utilisateur courante.
- **WP Créer plage images**(*objPlage*) -> *objPlage* : retourne une nouvelle plage contenant uniquement les images.
- **WP Créer plage paragraphes**(*objPlage*) -> *objPlage* : retourne une nouvelle plage contenant uniquement les paragraphes.

Vous pouvez obtenir des informations sur l'emplacement d'une plage dans un document (numéro de page, numéro de colonne...) à l'aide de la commande **WP Lire position**.

Gestion des tableaux

Les tableaux 4D Write Pro sont créés et construits par programmation à l'aide des commandes suivantes :

- **WP Insérer tableau** pour insérer un tableau et obtenir sa référence
- **WP Tableau ajouter ligne** pour ajouter des lignes au tableau et obtenir la plage correspondante
- **WP Tableau lire lignes** pour obtenir une plage de lignes
- **WP Tableau lire colonnes** pour obtenir une plage de colonnes(*)
- **WP Tableau lire cellules** pour obtenir une plage de cellules

(*) Les colonnes n'ont pas d'équivalent en html. Dans 4D Write Pro, une plage de colonne est en réalité une plage de cellules.

Les plages de lignes, colonnes et cellules disposent d'attributs privés spécifiques permettant de les définir :

Constante	Valeur	Comment
		(Disponible pour les tableaux, les documents les sections) Nombre de colonnes. Type de valeur : Entier long
wk column count	columnCount	<ul style="list-style-type: none">• Pour un tableau : attribut en lecture seulement• Pour un document ou une section : attribut en lecture/écriture. Valeur par défaut=1 (colonne unique). Valeur maximum=20
wk first column	firstColumn	(Attribut en lecture seule) Numéro de la première colonne. Valeur : Entier long
wk first row	firstRow	(Attribut en lecture seule) Numéro de la première ligne du tableau. Valeur : Entier long
wk row count	rowCount	(Attribut en lecture seule) Nombre total de lignes. Valeur : Entier long
wk table ID	tableID	(Attribut en lecture seule) ID du tableau parent. Valeur : Chaîne

Gestion des signets

4D Write Pro vous permet de créer et de manipuler des références dynamiques sur des parties de vos documents, appelées **signets**. Un signet est une référence nommée associée à une plage spécifique dans un document 4D Write Pro.

Les signets sont dynamiques, ce qui signifie que si l'utilisateur déplace, ajoute ou supprime du texte appartenant au signet, la plage associée sera automatiquement mise à jour et le signet continuera de référencer le même contenu dans le document. Par exemple :

- Vous créez un signet nommé "MyBM" qui référence la plage de texte "Hello world" à la page 20 de votre document.
- Puis vous insérez 50 pages au début du document.
- Vous pouvez toujours accéder automatiquement au même contenu "Hello world", maintenant à la page 70 du document, par l'intermédiaire du signet "MyBM".

Un document peut contenir un nombre illimité de signets. Plusieurs signets peuvent référencer la même plage, et les plages des signets peuvent être entrelacées. Cependant, chaque nom de signet doit être unique dans le document. Les signets ne sont pas importés lorsque la commande **WP INSERER DOCUMENT** est utilisée (les signets dans le document de destination ne peuvent pas être écrasés).

Une fois créé, un signet est stocké dans le document. Il est enregistré avec le document, et peut être manipulé par différentes commandes.

Les signets peuvent être utilisés pour référencer des parties d'un document modèle (*template*). Ces parties peuvent alors être automatiquement assemblées avec des données de la base afin de générer des documents finaux tels que des catalogues ou des factures.

Plusieurs commandes vous permettent de créer, supprimer et utiliser les signets :

- **WP CREER SIGNET** pour créer un signet à partir d'une plage,
- **WP LIRE SIGNETS** pour récupérer tous les signets définis dans le document,
- **WP Lire plage signet** pour récupérer une plage depuis un signet existant,
- **WP SUPPRIMER SIGNET** pour supprimer un signet.

Gestion des attributs

Les commandes suivantes permettent de lire et de fixer tout attribut dans le document :

- **WP FIXER ATTRIBUTS**(*objPlage | wpDoc ; nomAttrib ; valeurAttrib { ; nomAttrib2 ; valeurAttrib2 ; ... ; nomAttribN ; valeurAttribN }*) : fixe une ou plusieurs paires attribut/valeur dans le document ou la plage.

- **WP LIRE ATTRIBUTS**(*objPlage* | *wpDoc* ; *nomAttrib* ; *valeurAttrib* { ; *nomAttrib2* ; *valeurAttrib2* ; ... ; *nomAttribN* ; *valeurAttribN* }) : lit la valeur courante d'un ou plusieurs attribut(s) dans le document ou la page.
- **WP REINITIALISER ATTRIBUTS**(*objPlage* ; *nomAttrib* { ; *nomAttrib2* ; ... ; *nomAttribN* }) : réinitialise la valeurs d'attribut(s) dans le document ou la page.

Les attributs sont détaillés dans la section **Attributs 4D Write Pro**.

Gestion des polices

Une commande vous permet d'obtenir des informations sur la prise en charge des styles dans une plage :

- **WP Style pris en charge**(*objPlage* ; *wpStylePolice*) -> *vrai ou faux* : vous permet de savoir si une plage prend en charge un style de police spécifique (utile pour concevoir une interface).

Gestion des URLs

4D Write Pro vous permet d'associer un lien hypertexte à n'importe quelle plage de votre document (texte, paragraphe, image, colonne, cellule de tableau...). Par exemple, vous pouvez définir un lien pour une plage de type image ; une fois le document 4D Write Pro exporté en HTML, les utilisateurs pourront cliquer sur l'image et ouvrir la page Web qui correspond à l'adresse du lien.

Note : Les liens hypertexte peuvent être activés directement depuis les documents 4D Write Pro via **Ctrl+clac** (Windows) ou **Commande+clac** (macOS).

Les liens hypertexte sont gérés par 4D Write Pro en tant qu'attributs de plage, et sont définis ou lus à l'aide des commandes **WP FIXER ATTRIBUTS** et **WP REINITIALISER ATTRIBUTS**, associées à la constante `wk link url`. Par exemple, si vous voulez transformer en URL le texte sélectionné par l'utilisateur :

Please refer to the [4D documentation](#)
for more information

Vous pouvez écrire :

```
$range:=WP Lire selection(*;"WParea")
WP FIXER ATTRIBUTS($range;wk link url;"http://doc.4d.com")
```

Please refer to the [4D documentation](#)
for more information

Note : 4D Write Pro ne traduit pas les caractères spéciaux des URLs, s'il y en a. Vous devez vous charger de l'encodage des caractères spéciaux. Si l'URL n'est pas valide, une erreur est générée.

Les liens insérés adoptent le format par défaut des liens hypertexte (couleur bleue et souligné), sauf si un style de texte est déjà appliqué à la plage.

Pour le traitement des URLs dans vos documents, 4D Write Pro applique les règles suivantes :

- Si vous associez un lien à une plage qui contient plusieurs paragraphes, plusieurs liens individuels seront créés.
- Si vous associez un lien à une plage qui contient déjà un ou plusieurs lien(s) :
 - si un lien existant est entièrement inclus dans la plage, il est supprimé et le nouveau lien est appliqué à la plage
 - si un lien est partiellement sélectionné (intersection avec la plage), il est réduit de manière à ce que la partie extérieure conserve le lien précédent et que le nouveau lien soit appliqué à la partie intérieure.
- Lors de la lecture de liens avec la commande **WP LIRE ATTRIBUTS**, les cas suivants peuvent se produire :
 - la plage contient un seul lien : la commande retourne la chaîne de l'URL
 - la plage contient plusieurs liens : la commande retourne la chaîne du premier URL
 - la plage ne contient aucun lien : la commande retourne une chaîne vide.
- Pour supprimer un lien dans une plage, vous pouvez appeler :

```
WP REINITIALISER ATTRIBUTS($range;wk link url)
```

ou bien :

```
WP FIXER ATTRIBUTS($range;wk link url;"")
```

WP Ajouter image

WP Ajouter image (wpDoc ; image) -> Résultat

Paramètre	Type	Description
wpDoc	Objet	→ Document 4D Write Pro
image	Image, Texte	→ Image (champ ou variable) ou chemin d'accès de l'image (texte)
Résultat	Objet	↪ Objet référençant l'image

Description

La commande **WP Ajouter image** ancre *image* à un emplacement fixe dans *wpDoc* et retourne sa référence. La référence retournée peut alors être passée à la commande **WP FIXER ATTRIBUTS** pour déplacer *image* à n'importe quel endroit dans *wpDoc* (page, section, en-tête, pied, etc.) avec une couche définie, taille, etc.

Dans *wpDoc*, passez le nom de l'objet document 4D Write Pro.

Dans *image*, vous pouvez passer :

- soit une image 4D,
- ou une chaîne contenant un chemin d'accès à un fichier image stocké sur le disque, exprimé en utilisant la syntaxe système. Si vous utilisez une chaîne, vous pouvez passer soit le chemin d'accès complet, ou le chemin relatif au fichier de structure. Vous pouvez aussi passer un nom de fichier, dans ce cas, le fichier doit se trouver à côté du fichier de structure de la base. Si vous passez un nom de fichier, vous devez indiquer son extension.

Note : Tous les formats d'image supportés par 4D peuvent être utilisés (voir la section **Introduction aux images**). Vous pouvez obtenir la liste des formats image disponibles en utilisant la commande **LISTE CODECS IMAGES**. Si l'image encapsule plusieurs formats (codecs), 4D Write Pro garde seulement un format pour l'affichage et un format pour l'impression (s'ils sont différents) dans le document ; les "meilleurs" formats sont automatiquement sélectionnés.

Par défaut, l'image ajoutée est :

- intégrée derrière le texte,
- affichée dans le coin supérieur gauche du papier,
- affichée sur toutes les pages.

L'emplacement, la couche (en ligne, devant/derrrière le texte), la visibilité, et toutes les propriétés de *image* peuvent être modifiées à l'aide de la commande **WP FIXER ATTRIBUTS**, ou via des actions standard (voir **Utiliser les actions standard 4D Write Pro**).

Note : La commande **WP Lire selection** retourne un objet *référence image* si une image ancrée est sélectionnée, et un *objet plage* si une image en ligne est sélectionnée. Vous pouvez vérifier si l'objet sélectionné est un objet image avec l'attribut wk type. Si *value* = 2, alors c'est un objet image, sinon c'est un objet plage (*value* = 0).

Exemple 1

Vous souhaitez ajouter une image avec les paramétrages par défaut, en utilisant un chemin d'accès.

```
C_OBJET($obPict)
$obPict:=WP Ajouter image(myDoc;"C:\\Users\\John.Doe\\Pictures\\Sunrise.jpg")
```

Le résultat est :



Exemple 2

Vous voulez ajouter une image retournée, centrée et ancrée au papier :

```
C_OBJET($obImage)
$obImage:=WP Ajouter image(myDoc;"C:\\Users\\John.Doe\\Pictures\\Saved Pictures\\Sunrise.jpg")
WP FIXER ATTRIBUTS($obImage;wk anchor origin;wk header box)
WP FIXER ATTRIBUTS($obImage;wk anchor horizontal align;wk center)
WP FIXER ATTRIBUTS($obImage;wk anchor vertical align;wk center)
WP FIXER ATTRIBUTS($obImage;wk width;"650px";wk height;"120px")
```

Le résultat est :



WP Creer plage

WP Creer plage (objCible ; débutPlage ; finPlage) -> Résultat

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément ou document 4D Write Pro
débutPlage	Entier long	→ Position du début de la plage dans la zone
finPlage	Entier long	→ Position de la fin de la plage dans la zone
Résultat	Objet	↻ Nouvel objet plage

Compatibilité

La commande **WP Creer plage** était nommée **WP Lire plage** dans les versions précédentes de 4D Write Pro. Elle a été renommée pour plus de clarté.

Description

La commande **WP Creer plage** retourne une nouvelle plage (*objPlage*) contenant les caractères situés entre *débutPlage* et *finPlage* dans l'objet *objCible*.

Un **objet plage 4D Write Pro** peut être utilisé pour manipuler des attributs sur une partie du document (notamment à l'aide des commandes **WP LIRE ATTRIBUTS** et **WP FIXER ATTRIBUTS**). Pour plus d'informations, veuillez vous reporter au paragraphe **Gestion des plages**.

Dans *objCible*, vous pouvez passer :

- un objet plage 4D Write Pro, ou
- une référence d'élément (en-tête / pied / corps / tableau / ligne / image en ligne / paragraphe), ou
- un document 4D Write Pro.

Si un objet non valide est passé dans *objCible*, une plage *objPlage* vide est retournée.

Dans *débutPlage* et *finPlage*, passez des valeurs correspondant à la position du premier et du dernier caractère à sélectionner pour la plage. Vous pouvez passer *wk_start_text* dans *débutPlage* pour désigner le début du document et *wk_end_text* dans *finPlage* pour désigner la fin du document. N'oubliez pas qu'un document 4D Write Pro ne contient pas seulement du texte visible mais également des caractères de formatage et des balises qui sont inclus dans la plage.

Si vous avez passé une référence d'élément tableau, ligne, paragraphe, image en ligne, body, en-tête ou pied dans *objCible*, les valeurs *débutPlage* et *finPlage* seront relatives aux indices de début et de fin de la plage de l'élément. Par exemple, si un en-tête est passé, *wk_start_text* désigne le caractère de début de l'en-tête et *wk_end_text* le dernier caractère de l'en-tête ; si une référence d'élément tableau est passée, la commande retourne la plage de texte relative au tableau référencé.

Note : Si vous avez passé une référence d'élément dans *objCible*, la plage retournée est relative au contenu de l'élément parent. L'objet plage conserve une référence interne à l'objet parent (en-tête, pied, corps, tableau, etc).

Exemple

Vous souhaitez sélectionner une plage de 12 caractères à partir du début d'un champ 4D Write Pro. Le champ est affiché dans un objet de formulaire :

Description

The DIALOG command presents the form form to the user. This command is often used to get information from the user through the use of variables, or to present information to the user, such as options for performing an operation. It is common to display the form inside a modal window created with the Open window command.

Si vous exécutez :

```
$range2:=WP Creer plage([SAMPLE]WP;wk start text;12)  
WP SELECTIONNER(*;"WParea";$range2)
```

... le résultat est le suivant :

Description

The DIALOG command presents the form form to the user. This command is often used to get information from the user through the use of variables, or to present information to the user, such as options for performing an operation. It is common to display the form inside a modal window created with the Open window command.

⚙️ WP Creer plage images

WP Creer plage images (objCible) -> Résultat

Paramètre	Type		Description
objCible	Objet	➔	Plage ou référence d'élément
Résultat	Objet	↩	Plage contenant uniquement les images

Compatibilité

La commande **WP Creer plage images** était nommée **WP Lire images** dans les versions précédentes de 4D Write Pro. Elle a été renommée pour plus de clarté.

Description

La commande **WP Creer plage images** retourne un objet plage qui référence uniquement les images contenues dans *objCible* que vous avez passé en paramètre. L'objet plage d'images retourné peut être utilisé avec **WP LIRE ATTRIBUTS** et **WP FIXER ATTRIBUTS** afin de manipuler les attributs d'images uniquement.

Dans *objCible*, vous pouvez passer :

- un objet plage 4D Write Pro, ou
- une référence d'élément (en-tête / pied / corps / tableau / ligne / image en ligne).

Exemple

Vous souhaitez modifier la couleur de bordure des images uniquement :

```
$oPicts:=WP Creer plage images($oSelection)  
WP FIXER ATTRIBUTS($oPicts;wk border color;"blue")
```

⚙️ WP Creer plage paragraphes

WP Creer plage paragraphes (objCible) -> Résultat

Paramètre	Type	Description
objCible	Objet →	Plage ou référence d'élément
Résultat	Objet ↻	Plage définissant uniquement les paragraphes

Compatibilité

La commande **WP Creer plage paragraphes** était nommée **WP Lire paragraphes** dans les versions précédentes de 4D Write Pro. Elle a été renommée pour plus de clarté.

Description

La commande **WP Creer plage paragraphes** retourne un objet plage qui référence uniquement les paragraphes contenus dans *objCible* que vous avez passé en paramètre. L'objet plage de paragraphes retourné peut être utilisé avec **WP LIRE ATTRIBUTS** et **WP FIXER ATTRIBUTS** afin de manipuler les attributs de paragraphes uniquement.

Dans *objCible*, vous pouvez passer :

- une plage 4D Write Pro, ou
- une référence d'élément (en-tête / pied / corps / tableau / ligne / image en ligne)

Exemple

Vous voulez définir la marge intérieure pour les paragraphes uniquement :

```
$oParagraphs:=WP Creer plage paragraphes($oSelection)  
WP FIXER ATTRIBUTS($oParagraphs;wk padding;20)
```


⚙ WP Creer plage tableaux

WP Creer plage tableaux (objCible) -> Résultat

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément d'un document 4D Write Pro
Résultat	Objet	↩ Objet plage contenant uniquement les tableaux

Description

La commande **WP Creer plage tableaux** retourne un objet plage qui n'adresse que les tableaux contenus dans l'*objCible* que vous passez en paramètre. Elle retournera une plage de wk type table, même si *objCible* passé en paramètre ne contient pas de tableau. L'objet plage tableaux retourné peut être utilisé par **WP LIRE ATTRIBUTS** et **WP FIXER ATTRIBUTS** pour gérer les attributs à appliquer aux tableaux et à leur contenu.

Dans le paramètre *objCible*, vous pouvez passer :

- une plage, ou
- une référence d'élément (corps / en-tête / pied / paragraphe), ou
- un document 4D Write Pro

Exemple

Vous souhaitez modifier l'apparence des tableaux contenus dans la plage tableau :

```
tableRange:=WP Creer plage tableaux(myDoc)
//attributs tableaux
WP FIXER ATTRIBUTS(tableRange;wk border style;wk solid;wk border color;"Blue";wk border width;"3px")
WP FIXER ATTRIBUTS(tableRange;wk table align;wk center)
//attributs texte
WP FIXER ATTRIBUTS(tableRange;wk font size;12)
WP FIXER ATTRIBUTS(tableRange;wk text transform;wk capitalize)
WP FIXER ATTRIBUTS(tableRange;wk font bold;wk true)
```

WP CREER SIGNET

WP CREER SIGNET (objCible ; nomSignet)

Paramètre	Type		Description
objCible	Objet	→	Plage ou référence d'élément
nomSignet	Chaîne	→	Nom du signet à créer

Description

La commande **WP CREER SIGNET** crée un nouveau signet nommé *nomSignet* basé sur *objPlage* du document parent.

Les signets sont des références nommées associées à des plages ou à des références d'éléments, vous permettant de désigner et de réutiliser des parties spécifiques du document, par exemple pour les besoins liés à la génération de modèles. Pour plus d'informations, veuillez vous reporter à la section **Gestion des signets**.

Dans *objCible*, vous pouvez passer :

- une plage 4D Write Pro, ou
- une référence d'élément (corps / tableau / ligne / image en ligne / paragraphe)

Note : Si vous passez une référence d'élément dans *objCible*, le signet contiendra uniquement l'élément défini.

Dans *nomSignet*, passez le nom du nouveau signet. Un nom de signet doit être conforme aux règles de nommage HTML/CSS, i.e. il doit uniquement contenir des caractères alphanumériques (les caractères invalides, tels que les espaces, sont automatiquement supprimés). Chaque nom de signet doit être unique dans un document. Si un signet de même nom existe déjà dans le document, il est écrasé.

Vous pouvez créer autant de signets que vous voulez dans un même document. Plusieurs signets peuvent être créés avec la même plage. Une fois créé, un signet est automatiquement stocké dans le document parent et est sauvegardé en même temps que le document lui-même.

Exemple 1

Vous souhaitez créer un nouveau signet référençant le texte couramment sélectionné dans le document. Vous pouvez écrire :

```
C_OBJET($range)
$range:=WP Lire selection(*;"WPDocument")
WP CREER SIGNET($range;"my_bookmark")
```

Exemple 2

Vous souhaitez renommer un signet existant. Pour cela, vous devez créer un nouveau signet basé sur la même plage puis supprimer l'ancien :

```
C_TEXTE($bookmarkOldName)
C_TEXTE($bookmarkNewName)
C_ENTIER LONG($p)
C_OBJET($wpRange)

$bookmarkOldName:="MyBookmark"
$bookmarkNewName:="MyNewBookmark"

TABLEAU TEXTE($_bookmarks;0)
WP LIRE SIGNETS(WParea;$_bookmarks)

$p:=Chercher dans tableau($_bookmarks;$bookmarkOldName)
Si($p>0)
    $wpRange:=WP Lire plage signet(WParea;$bookmarkOldName)
    WP SUPPRIMER SIGNET(WParea;$bookmarkOldName)
    WP CREER SIGNET($wpRange;$bookmarkNewName)
Fin de si
```

WP EXPORTER DOCUMENT

WP EXPORTER DOCUMENT (*docWP* ; *cheminFichier* {; *format* {; *options*} })

Paramètre	Type	Description
<i>docWP</i>	Objet	Variable 4D Write Pro
<i>cheminFichier</i>	Chaîne	Chemin du fichier d'export
<i>format</i>	Entier long	Format de sortie du document
<i>options</i>	Entier long	Options d'export

Description

La commande **WP EXPORTER DOCUMENT** exporte l'objet 4D Write Pro *docWP* dans un document sur disque défini par le paramètre *cheminFichier* ainsi que des paramètres optionnels.

Dans *docWP*, passez l'objet 4D Write Pro que vous voulez exporter.

Dans *cheminFichier*, passez le chemin de destination et le nom du document à exporter. Si vous passez uniquement un nom de fichier, il sera enregistré au même niveau que le fichier de structure de 4D. Par défaut, si vous omettez le paramètre *format*, la commande utilise l'extension du fichier pour déterminer le format d'export du document.

Vous pouvez également passer une constante du thème **Constantes 4D Write Pro** dans le paramètre *format*. Dans ce cas, 4D ajoute l'extension appropriée au nom de fichier si nécessaire. Les formats suivants sont pris en charge :

Constante	Type	Valeur	Comment
wk 4wp	Entier long	4	Le document 4D Write Pro est sauvegardé dans un format d'archive natif (HTML <i>zippé</i> avec images stockées dans un dossier séparé). Les expressions 4D ne sont pas calculées et les balises 4D spécifiques sont incluses. Ce format est particulièrement approprié pour la sauvegarde et l'archivage de documents 4D Write Pro sur disque sans aucune perte d'informations.
wk page web complète	Entier long	2	Extension .htm ou .html. Le document est sauvegardé au format HTML standard et ses ressources sont sauvegardées séparément. Les balises 4D spécifiques sont supprimées et les expressions sont calculées. Ce format est particulièrement adapté à l'affichage d'un document 4D Write Pro dans un navigateur Web.

Notes :

- "Balises 4D spécifiques" désigne le XHTML 4D avec *namespace* 4D et styles CSS 4D.
- Les expressions peuvent être figées à tout moment à l'aide de la commande **ST FIGER EXPRESSIONS**.
- Pour plus d'informations sur le format de document 4D Write Pro, veuillez vous reporter au paragraphe **Format de document .4wp**.

Vous pouvez passer dans le paramètre *options* des options de configuration de l'export. Vous devez passer une valeur de type *entier long* pour définir le style du code HTML. Les constantes suivantes sont disponibles :

Constante	Type	Valeur	Comment
wk html debug	Entier long	1	Code HTML formaté ("pretty print"), facilitant le débogage
wk normal	Entier long	0	Code HTML standard

- **Option HTML debug désactivée (défaut) :**

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }p,li { white-space:pre-wrap;margin:0pt;padding:0pt;font-family:Times New Roman }p.Normal,li.Normal { text-align:left }p_p1,li_p1 { font-family:Arial;font-size:18pt;color:#1D1D1D }img_img1 { width:51pt;height:51pt }</style></head><body><p class="Normal_p1"><span>23/12/14</span></p><p class="Normal_p1">Dear <span style="color:#BE0E00">Shaun Stoltz</span><span style="color:#800062"><span style="color:#000000">from
```

- **Option HTML debug activée :**

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }
```

Exemple 1

Vous voulez exporter le contenu de l'objet 4D Write Pro *myArea* dans un document dans le dossier de votre base. Vous pouvez activer l'option debug à l'aide d'un bouton 'pprint' :

```
C_TEXTE($filePath)
$filePath:=Dossier 4D(Dossier base)+"Export fichiers"+Séparateur dossier+"WriteProExport.html"
Si(pprint=0) //si l'option debug est désactivée
  WP EXPORTER DOCUMENT(myArea;$filePath;wk web page complète;wk normal)
Sinon
  WP EXPORTER DOCUMENT(myArea;$filePath;wk web page complète;wk html debug)
Fin de si
```

Exemple 2

Vous souhaitez exporter le contenu de l'objet 4D Write Pro *myArea* dans le format .4wp :

C_TEXTE(\$path)

C_ENTIER LONG(\$docRef)

Au cas ou

:(**Evenement formulaire**=Sur clic)

\$path:=**Dossier 4D**(Dossier base)+"Export"+Séparateur dossier

\$path:=**Selectionner document**(\$path;".4wp";" title";Saisie nom de fichier)

Si(\$path#command_5"")

WP EXPORTER DOCUMENT(myArea;document;wk 4wp;wk normal)

Sinon

ALERTE("An error occurred.")

Fin de si

Fin de cas

WP EXPORTER VARIABLE

WP EXPORTER VARIABLE (docWP ; destination ; format { ; options })

Paramètre	Type	Description
docWP	Objet	→ Variable 4D Write Pro
destination	Variable texte	← Variable devant recevoir le contenu exporté
format	Entier long	→ Format de sortie de la variable
options	Entier long	→ Options d'export

Description

La commande **WP EXPORTER VARIABLE** exporte l'objet 4D Write Pro *docWP* dans la variable 4D *destination* avec le *format* spécifié.

Dans *docWP*, passez l'objet 4D Write Pro que vous voulez exporter.

Dans *destination*, passez la variable que vous voulez remplir avec l'objet 4D Write Pro exporté. Le type de la variable dépend du format d'export défini dans le paramètre *format* :

- si vous passez le format natif .4wp, la variable sera de type Blob,
- si vous passez un format html, la variable sera de type Texte.

Dans le paramètre *format*, passez une constante du thème **Constantes 4D Write Pro** définissant le format d'export que vous voulez utiliser. Chaque format est lié à un usage spécifique. Les formats suivants sont pris en charge :

Constante	Type	Valeur	Comment
wk 4wp	Entier long	4	Le document 4D Write Pro est sauvegardé dans un format d'archive natif (HTML <i>zippé</i> avec images stockées dans un dossier séparé). Les expressions 4D ne sont pas calculées et les balises 4D spécifiques sont incluses. Ce format est particulièrement approprié pour la sauvegarde et l'archivage de documents 4D Write Pro sur disque sans aucune perte d'informations.
wk mime html	Entier long	1	Le document 4D Write Pro est sauvegardé au format MIME HTML avec les documents html et les images embarqués en tant que parties MIME (encodées en base64). Les expressions sont calculées et les balises 4D spécifiques sont supprimées. Ce format est particulièrement adapté à l'envoi de mails au format HTML à l'aide de la commande SMTP_QuickSend .
wk page web html 4D	Entier long	3	Le document 4D Write Pro est sauvegardé au format HTML et inclut les balises 4D spécifiques ; chaque expression est insérée sous forme d'espace insécable. Comme ce format est sans perte, il est approprié pour le stockage dans un champ texte.

Notes :

- "Balises 4D spécifiques" désigne le XHTML 4D avec *namespace* 4D et styles CSS 4D.
- Les expressions peuvent être figées à tout moment à l'aide de la commande **ST FIGER EXPRESSIONS**.
- Pour plus d'informations sur le format de document 4D Write Pro, veuillez vous reporter au paragraphe **Format de document .4wp**.

Dans le paramètre *options*, passez des options de configuration de l'export. Vous pouvez passer soit :

- un *entier long* définissant le style du code HTML ; les constantes suivantes sont disponibles :

Constante	Type	Valeur	Comment
wk html debug	Entier long	1	Code HTML formaté ("pretty print"), facilitant le débogage
wk normal	Entier long	0	Code HTML standard

o Option HTML debug désactivée (défaut) :

```
<html xmlns="http://www.w3.org/1999/xhtml"> <head> <title> New 4D Write Pro Document</title> <style type="text/css"> body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }p,li { white-space:pre-wrap;margin:0pt;padding:0pt;font-family:'Times New Roman' }p.Normal,li.Normal { text-align:left }p._p1,li._p1 { font-family:'Arial';font-size:18pt;color:#1D1D1D }img._img1 { width:51pt;height:51pt }</style> </head> <body> <p class="Normal _p1"> <span>23/12/14</span>,</p> <p class="Normal _p1"> Dear <span style="color:#BE0E00"> Shaun Stoltz</span> <span style="color:#800062"> <span style="color:#000000"> from
```

o Option HTML debug activée :

```
<html xmlns="http://www.w3.org/1999/xhtml"> <head> <title> New 4D Write Pro Document</title> <style type="text/css"> body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }
```

- soit une *chaîne*. La propriété suivante est prise en charge (uniquement lorsque le format `wk mime html` est utilisé):
 - o Nom de domaine CID : Nom de domaine hôte qui sera ajouté aux URLs CID générés, avec un '@' comme séparateur. Par exemple, si vous passez "gmail.com", "123@gmail.com" sera inséré si le CID unique est 123. Par défaut, si ce paramètre est omis, seul le CID unique est inséré (accepté par la plupart des serveurs de messagerie).

Exemple

Vous voulez envoyer un mail contenant du texte multistyle, des références 4D et des images. Vous pouvez utiliser une zone 4D Write Pro exportée au format MIME HTML et l'envoyer en utilisant 4D Internet Commands :

```
C_ENTIER LONG($smtpid_1;$err_1;$smtpOption_1;$smtpPort_1)
C_TEXTE($str;$emailBody_t;$smtpHost_t;$emailTo_t;$emailFrom_t;$smtpPass_t)
```

```
//export zone dans le format approprié
```

```
WP_EXPORTER VARIABLE(myWPArea;$str;wk mime.html)
```

```
$emailTo_t:="johnsmith@4d.com"
```

```
$emailFrom_t:"testWritePro@gmail.com"
```

```
$emailBody_t:=$str
```

```
$smtpHost_t:"smtp.gmail.com"
```

```
$smtpOption_1:=9
```

```
$smtpPort_1:=465
```

```
$smtpPass_t:"QRN_on_bretzelburg"
```

```
$err_1:=SMTP_QuickSend($smtpHost_t;$emailFrom_t;$emailTo_t;$emailTitle_t;\
$emailBody_t;$smtpOption_1;$smtpPort_1;$smtpUser_t;$smtpPass_t)
```

```
Si(($err_1=0))
```

```
ALERTE("Message envoyé à "+emailTo_t)
```

```
Sinon
```

```
ALERTE("Erreur dans les paramètres, veuillez réessayer.")
```

```
Fin de si
```

WP FIXER ATTRIBUTS

WP FIXER ATTRIBUTS (objCible ; nomAttribut ; valeurAttribut { ; nomAttribut2 ; valeurAttribut2 ; ... ; nomAttributN ; valeurAttributN })

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément ou document 4D Write Pro
nomAttribut	Texte	→ Nom d'attribut dont vous souhaitez modifier la valeur
valeurAttribut	Texte, Numérique, Tableau entier long, Tableau réel, Image, Date	→ Nouvelle valeur de l'attribut

Description

La commande **WP FIXER ATTRIBUTS** vous permet de fixer la valeur d'un ou plusieurs attribut(s) dans une plage, un élément référencé ou un document 4D Write Pro. Cette commande donne accès à tous les types d'attributs internes de 4D Write Pro : caractère, paragraphe, tableau, document ou image.

Dans *objCible*, vous pouvez passer :

- un objet plage 4D Write Pro, ou
- une référence d'élément (en-tête / pied / corps / tableau / ligne / image en ligne ou ancrée / paragraphe), ou
- un document 4D Write Pro.

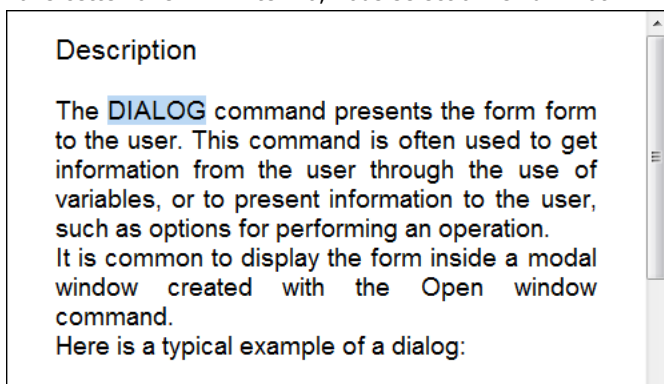
Dans *nomAttribut*, passez le nom de l'attribut à modifier pour la cible et dans *valeurAttribut*, passez la valeur à définir. Pour une liste complète des attributs que vous pouvez passer dans *nomAttribut* ainsi que les valeurs que vous pouvez passer dans *valeurAttribut*, veuillez vous référer à la section **Attributs 4D Write Pro**.

Vous pouvez passer autant de paires *nomAttribut* / *valeurAttribut* que vous voulez.

Note : Si vous devez définir plusieurs attributs pour la même cible, il est plus optimisé d'effectuer un appel unique à **WP FIXER ATTRIBUTS** avec toutes les paires attributs/valeurs, que d'appeler plusieurs fois **WP FIXER ATTRIBUTS**.

Exemple 1

Dans cette zone 4D Write Pro, vous sélectionnez un mot :



Si vous exécutez le code suivant :

```
$range:=WP Lire selection(*;"WParea") //récupérer la plage sélectionnée

// fixer le décalage d'ombre en pt pour la sélection
WP FIXER ATTRIBUTS($range;wk text shadow offset;1)
//définir la marge intérieure du paragraphe
WP FIXER ATTRIBUTS($range;wk padding;1)
//définir une bordure de 10 pt
WP FIXER ATTRIBUTS($range;wk border style;wk solid;wk border width;10)
//définir les couleurs de bordure
WP FIXER ATTRIBUTS($range;wk border color;"blue";wk border color bottom;"#command_500FA9A";wk border color
right;"#command_500FA9A")
```

Vous obtenez le résultat suivant :

Description

The **DIALOG** command presents the form form to the user. This command is often used to get information from the user through the use of variables, or to present information to the user, such as options for performing an operation.

It is common to display the form inside a modal window created with the Open window command.

Here is a typical example of a dialog:

Exemple 2

Cet exemple illustre le rôle des constantes `wk inside` et `wk outside` :

```
$wpRange:=WP Lire selection(writeProdoc)
WP FIXER ATTRIBUTS($wpRange;wk border style+wk inside;wk dotted)
WP FIXER ATTRIBUTS($wpRange;wk border style+wk outside;wk solid)
WP FIXER ATTRIBUTS($wpRange;wk border color+wk outside;"#command_500FA9A")
```

En supposant que tout le contenu était sélectionné, le résultat est :

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed non risus. Suspendisse lectus tortor, dignissim sit amet, adipiscing nec, ultricies sed, dolor.

.....

Cras elementum ultrices diam. Maecenas ligula massa, varius a, semper congue, euismod non, mi. Proin porttitor, orci nec nonummy molestie, enim est eleifend mi, non fermentum diam nisl sit amet erat. Duis semper.

.....

Duis arcu massa, scelerisque vitae, consequat in, pretium a, enim. Pellentesque congue. Ut in risus volutpat libero pharetra tempor. Cras vestibulum bibendum augue. Praesent egestas leo in pede. Praesent blandit odio eu enim.

.....

Pellentesque sed dui ut augue blandit sodales. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam nibh. Mauris ac mauris sed pede pellentesque fermentum. Maecenas adipiscing ante non diam sodales hendrerit.

Exemple 3

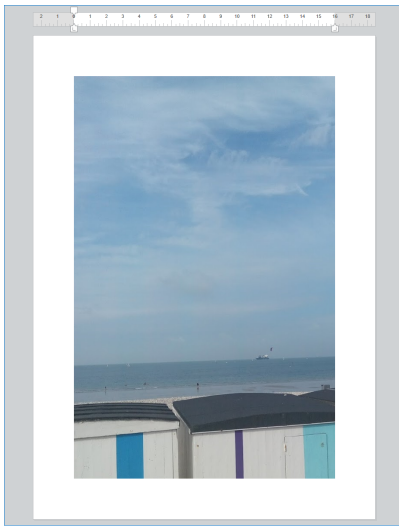
Vous souhaitez définir une image de fond pour le document :

```
C_OBJET(WParea)
WParea:=WP Nouveau

LIRE FICHIER IMAGE("C:\\Pictures\\boats.jpg";$picture)

WP FIXER ATTRIBUTS(WParea;wk background image;$picture)
```

Le résultat est le suivant :



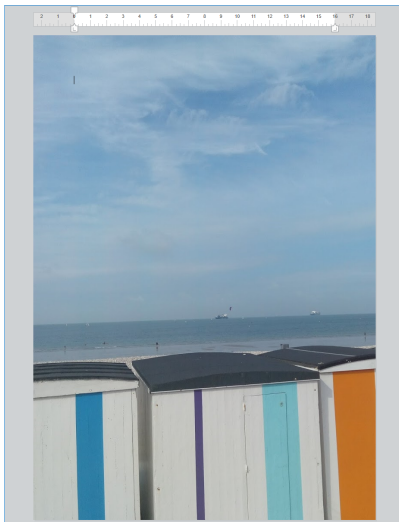
Vous souhaitez définir une image de fond qui recouvre la totalité de la zone d'impression :

```
C_OBJET(WParea)
WParea:=WP Nouveau

LIRE FICHIER IMAGE("C:\\Pictures\\boats.jpg";$picture)

WP FIXER ATTRIBUTS(WParea;wk background image;$picture)
WP FIXER ATTRIBUTS(WParea;wk background clip;wk paper box)
WP FIXER ATTRIBUTS(WParea;wk background origin;wk paper box)
```

Le résultat est le suivant :



Note : La valeur wk paper box peut être appliquée uniquement aux documents ou aux sections.

Exemple 4

Vous voulez définir des tabulations à intervalles variés et utiliser différents caractères de conduite :

```
C_OBJET(WParea)

TABLEAU TEXTE(_position;0)
TABLEAU ENTIER LONG(_type;0)
TABLEAU TEXTE(_leadings;0)

AJOUTER A TABLEAU(_position;"3cm")
AJOUTER A TABLEAU(_type;wk left)
AJOUTER A TABLEAU(_leadings;".")

AJOUTER A TABLEAU(_position;"7.5cm")
AJOUTER A TABLEAU(_type;wk right)
AJOUTER A TABLEAU(_leadings;".")

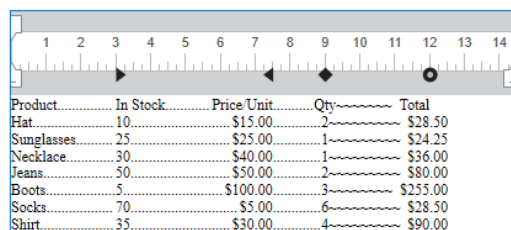
AJOUTER A TABLEAU(_position;"9cm")
AJOUTER A TABLEAU(_type;wk center)
AJOUTER A TABLEAU(_leadings;".")
```

AJOUTER A TABLEAU(_position;"12cm")
AJOUTER A TABLEAU(_type;wk decimal)
AJOUTER A TABLEAU(_leadings;"~")

AJOUTER A TABLEAU(_position;"2cm")
AJOUTER A TABLEAU(_type;wk left)
AJOUTER A TABLEAU(_leadings;"")

\$range:=WP Lire selection(WParea)
WP FIXER ATTRIBUTS(\$range;wk tab stop offsets;_position)
WP FIXER ATTRIBUTS(\$range;wk tab stop types;_type)
WP FIXER ATTRIBUTS(\$range;wk tab stop leadings;_leadings)

Le résultat est du type :



Product	In Stock	Price Unit	Qty	Total
Hat	10	\$15.00	2	\$28.50
Sunglasses	25	\$25.00	1	\$24.25
Necklace	30	\$40.00	1	\$36.00
Jeans	50	\$50.00	2	\$80.00
Boots	5	\$100.00	3	\$255.00
Socks	70	\$5.00	6	\$28.50
Shirt	35	\$30.00	4	\$90.00

WP FIXER CADRE ({ * ; } zoneWP ; sélecteurCadre)

Paramètre	Type	Description
*	Opérateur	⇒ Si spécifié, zoneWP est un nom d'objet (chaîne). Si omis, zoneWP est un champ ou une variable
zoneWP	Chaîne	⇒ Nom d'objet de formulaire (si * spécifié) ou champ ou variable objet 4D Write Pro (si * omis)
sélecteurCadre	Entier long	⇒ Cadre dans lequel placer le curseur

Description

La commande **WP FIXER CADRE** place le curseur dans le cadre désigné par le paramètre *sélecteurCadre* de la zone 4D Write Pro *wpZone*.

Si vous passez le paramètre optionnel ***, vous indiquez que le paramètre *zoneWP* est un nom d'objet (une chaîne). Si vous ne passez pas le paramètre, vous indiquez que le paramètre *zoneWP* est un champ ou une variable objet.

Vous pouvez passer dans *sélecteurCadre* une des constantes suivantes du thème **Constantes 4D Write Pro** :

Constante	Type	Valeur	Comment
wk body	Entier long	0	Zone de corps (<i>body</i>) de la section
wk current page footer	Entier long	2	
wk current page header	Entier long	1	
wk current section default footer	Entier long	10	Pied de page par défaut de la section (disponible uniquement s'il n'y a pas de pied de sous-section active).
wk current section default header	Entier long	9	En-tête par défaut de la section (disponible uniquement s'il n'y a pas d'en-tête de sous-section active).
wk current section first footer	Entier long	4	Pied de la première page de la section
wk current section first header	Entier long	3	En-tête de la première page de la section
wk current section left footer	Entier long	6	Pied de la (des) page(s) gauche(s) de la section
wk current section left header	Entier long	5	En-tête de la (des) page(s) gauche(s) de la section
wk current section right footer	Entier long	8	Pied de la (des) page(s) droite(s) de la section
wk current section right header	Entier long	7	En-tête de la (des) page(s) droite(s) de la section

- Si *wk current page header* ou *wk current page footer* est passée dans *sélecteurCadre*, **WP FIXER CADRE** donnera le focus au premier en-tête ou au pied de la page sélectionnée, quel que soit le type de sous-section (si elle ne contient ni en-tête ni pied de page, la commande ne fera rien).
- Si une constante du type *wk current section[...]* est passée dans *sélecteurCadre*, **WP FIXER CADRE** donnera le focus à la sous-section en-tête ou pied spécifiée de la première section sélectionnée (si elle ne contient ni en-tête ni pied de page, la commande ne fera rien).
- **WP FIXER CADRE** ne fera rien si le cadre *sélecteurCadre* n'existe pas pour la section sélectionnée (par exemple, *wk current section left header* lorsqu'il n'y a pas de distinction entre les pages gauches et droites dans *zoneWP*).

Exemple

Vous souhaitez placer le curseur dans le pied de page gauche de la section courante :

```
WP FIXER CADRE(*;"WPArea";wk current section left footer)
```

WP FIXER TEXTE

WP FIXER TEXTE (*objPlage* ; *nouveauTexte* ; *positionTexte* { ; *miseAJourPlage* })

Paramètre	Type	Description
<i>objPlage</i>	Objet	→ Plage 4D Write
<i>nouveauTexte</i>	Texte	→ Texte à insérer
<i>positionTexte</i>	Numérique	→ Position de l'insertion (remplace, avant, après)
<i>miseAJourPlage</i>	Numérique	→ Inclut ou exclut le contenu inséré dans la plage

Description

La commande **WP FIXER TEXTE** insère le texte passé dans le paramètre *nouveauTexte* en texte brut dans un document 4D Write Pro, à la position spécifiée par les paramètres *objPlage* and *positionTexte*.

Dans le paramètre *objPlage*, passez un objet plage pour désigner la position où le texte doit être inséré. La plage peut être un emplacement explicite dans le document 4D Write Pro ou peut être basée sur une sélection de l'utilisateur.

Le paramètre *nouveauTexte* contient le texte à insérer dans *objPlage*. **WP FIXER TEXTE** insère du texte brut seulement, tel qu'il est écrit, mais sans aucune balise de style.

Dans le paramètre *positionTexte*, l'emplacement du texte inséré peut être précisée en passant l'un des sélecteurs suivants :

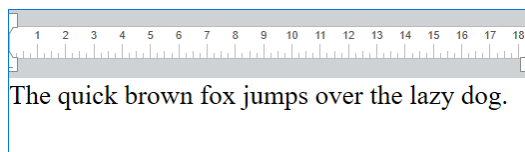
Constante	Comment
<i>wk append</i>	Insère le contenu à la fin de la plage
<i>wk prepend</i>	Le contenu est inséré au début de la plage
<i>wk replace</i>	Remplace le contenu de la plage

Dans le paramètre optionnel *miseAJourPlage*, vous pouvez passer une des constantes suivantes pour préciser si le texte inséré doit être inclus ou non dans la plage résultante :

Constante	Comment
<i>wk exclude from range</i>	Le contenu inséré n'est pas inclus dans la plage mise à jour
<i>wk include in range</i>	Le contenu inséré est inclus dans la plage mise à jour (défaut)

Exemple

Vous souhaitez ajouter du texte à ce document :



Avant le texte existant :

```
$range:=WP Creer plage(myDoc;wk start text;wk end text)
WP FIXER TEXTE($range;" Hello world!";wk prepend)
// écrit " Hello world! The quick brown fox jumps over the lazy dog."
```

Après le texte existant :

```
$range:=WP Creer plage(myDoc;wk start text;wk end text)
WP FIXER TEXTE($range;" Hello world!";wk append)
// écrit "The quick brown fox jumps over the lazy dog. Hello world!"
```

Remplace le texte existant :

```
$range:=WP Creer plage(myDoc;wk start text;wk end text)
WP FIXER TEXTE($range;" Hello world!";wk replace)
// écrit " Hello world!"
```

WP Importer document

WP Importer document (cheminFichier) -> Résultat

Paramètre	Type		Description
cheminFichier	Chaîne	→	Chemin du document 4D Write (.4w7 ou .4wt) ou du document 4D Write Pro (.4wp)
Résultat	Objet	↪	Objet 4D Write Pro

Description

La commande **WP Importer document** convertit un document 4D Write Pro ou 4D Write existant (extension .4wp, .4w7 ou .4wt) en un nouvel objet 4D Write Pro.

Dans le paramètre *cheminFichier*, vous passez le chemin d'accès d'un document stocké sur disque. Les types de documents suivants sont pris en charge :

- ancien document 4D Write (.4w7 ou .4wt). Pour une liste détaillée des fonctions 4D Write qui sont actuellement prises en charge dans les objets 4D Write Pro, veuillez vous référer à la section **Import de documents 4D Write**.
- document au format 4D Write Pro (.4wp). Pour plus d'informations sur le format de document 4D Write Pro, veuillez vous reporter au paragraphe **Format de document .4wp**.

Vous devez passer un chemin d'accès complet, sauf si le document est situé au même niveau que le fichier de structure, auquel cas vous pouvez passer uniquement son nom.

Après exécution, la commande retourne l'objet 4D Write Pro résultant de la conversion.

Note : Si vous voulez importer un document stocké dans un champ BLOB 4D, utilisez plutôt la commande **WP Nouveau**.

Une erreur est retournée si le paramètre *cheminFichier* est invalide, si le fichier est manquant, ou si le format du fichier n'est pas pris en charge.

Exemple

```
C_OBJET(WPDoc)
WPDoc:=WP Importer document("C:\\documents\\4DWriteDocs\\Letter.4w7")
```

WP IMPRIMER (docWP {; modeImpr})

Paramètre	Type	Description
docWP	Objet	→ Document 4D Write Pro
modeImpr	Entier long	→ Mode d'impression pour le document 4D Write Pro : 0 (défaut)=Mode 4D Write Pro, 1=Mode HTML WYSIWYG

Description

La commande **WP IMPRIMER** lance une tâche d'impression pour le document 4D Write Pro désigné par *docWP*, ou ajoute le document dans la tâche d'impression courante si elle est appelée entre les commandes **OUVRIR TACHE IMPRESSION** et **FERMER TACHE IMPRESSION** (uniquement en version 64 bits de 4D, cf. ci-dessous). **WP IMPRIMER** utilise les paramètres d'impression définis par les commandes 4D **PARAMETRES IMPRESSION** ou **FIXER OPTION IMPRESSION**, excepté pour les marges des pages qui sont toujours définies avec les options d'impression du document 4D Write Pro. **WP IMPRIMER** utilise les options courantes de mise en page (comme la taille de la page et l'orientation), ou celles du document si **WP UTILISER PARAMETRES IMPRESSION** a été appelée au préalable.

Le paramètre optionnel *modeImpr* permet l'utilisation de la vue WYSIWYG HTML pour les sorties impression. Vous pouvez utiliser l'une des constantes suivantes, placées dans le thème "4D Write Pro" :

Constante	Type	Valeur	Comment
wk 4D Write Pro layout	Entier long	0	Format 4D Write Pro standard, pouvant inclure des attributs de style spécifiques
wk html wysiwyg	Entier long	1	Dans ce format, les éventuels attributs avancés de 4D Write Pro non disponibles dans tous les navigateurs sont supprimés (multi-colonnes, double interlignes...)

Si *modeImpr* est omis, le format standard 4D Write Pro (0) est utilisé.

Note : Lorsqu'ils sont imprimés avec la commande **WP IMPRIMER**, les documents 4D Write Pro sont toujours imprimés en mode Page, indépendamment de la valeur de la propriété **Mode d'affichage** pour la zone (cf. **Configurer les propriétés d'affichage**).

Note pour les versions 32 bits

La commande **WP IMPRIMER** est prise en charge dans les versions 32 bits de 4D mais avec la limitation suivante : elle ne peut pas être appelée à l'intérieur d'une tâche d'impression démarrée avec la commande **OUVRIR TACHE IMPRESSION**. Si la commande est appelée dans une tâche d'impression sur une version 32 bits de 4D, une erreur est retournée.

Exemple

Vous souhaitez imprimer un document 4D Write Pro en format standard ou wysiwyg HTML en fonction de la valeur d'une variable :

```
// impression en mode wysiwyg HTML ou 4D Write Pro standard
Si(rb_htmlwysiwyg=1)
  WP IMPRIMER(writeProDoc;wk_html_wysiwyg)
Sinon
  WP IMPRIMER(writeProDoc;wk 4D Write Pro layout)
Fin de si
```

WP INSERER DOCUMENT

WP INSERER DOCUMENT (objPlage ; docWP ; mode {; miseAJourPlage})

Paramètre	Type		Description
objPlage	Objet	→	Plage 4D Write Pro
docWP	Objet	→	Document 4D Write Pro à insérer
mode	Entier long	→	Mode d'insertion
miseAJourPlage	Entier long	→	Mode de mise à jour de la plage de sélection

Description

La commande **WP INSERER DOCUMENT** insère le document *docWP* dans la plage *objPlage* selon le *mode* d'insertion spécifié et le paramètre *miseAJourPlage*.

Dans *objPlage*, passez une plage 4D Write Pro valide.

Le document inséré *docWP* peut être n'importe quel document 4D Write Pro, créé avec la commande **WP Nouveau** ou **WP Importer document**. Seuls les éléments du corps du texte sont insérés (sections et signets dans la plage de destination sont préservés). De plus, les éléments sont copiés, ainsi *docWP* peut être ré-utilisé plusieurs fois.

Dans le paramètre *mode*, passez une ou une combinaison des constantes suivantes du thème **Constantes 4D Write Pro**, pour indiquer le mode d'insertion à utiliser pour le document dans la plage de sélection *objPlage* du document de destination :

Constante	Type	Valeur	Comment
wk append	Entier long	2	Insère le contenu à la fin de la plage
wk prepend	Entier long	1	Le contenu est inséré au début de la plage
wk replace	Entier long	0	Remplace le contenu de la plage

Vous pouvez combiner une des constantes ci-dessus avec les options d'insertion suivantes :

Constante	Type	Valeur	Comment
wk freeze expressions	Entier long	64	Figurer les expressions lors de l'insertion
wk keep paragraph styles	Entier long	128	Conservé les styles du paragraphe de destination

Dans le paramètre optionnel *miseAJourPlage*, vous pouvez passer une des constantes suivantes pour spécifier si le contenu inséré est inclus ou pas dans la plage mise à jour :

Constante	Type	Valeur	Comment
wk exclude from range	Entier long	1	Le contenu inséré n'est pas inclus dans la plage mise à jour
wk include in range	Entier long	0	Le contenu inséré est inclus dans la plage mise à jour (défaut)

Si vous ne passez pas le paramètre *miseAJourPlage*, par défaut, le contenu inséré sera inclus dans la plage de sélection résultante.

Exemple 1

Vous souhaitez remplacer le contenu d'un document par le texte sélectionné d'un autre document :

```
$tempRange:=WP Lire selection(WPTemplate) //on lit la sélection utilisateur dans le document WPTemplate
$doctoCopy:=WP Nouveau($tempRange) //on crée un nouveau document à partir de WPTemplate
WP INSERER DOCUMENT(WPDoc;$doctoCopy;wk replace) //on remplace le contenu de WPDoc par le contenu du nouveau document
```

Exemple 2

Vous avez défini un document modèle (*template*) comportant différentes parties préformatées, chacune d'elles étant stockée en tant que signet. Lorsque vous produisez un document final à partir du *template*, vous pouvez traiter chaque signet en tant que nouveau document et l'insérer dans le document final.

```
TABLEAU TEXTE($_BookmarkNames;0)
WP LIRE SIGNETS([TEMPLATES]WP;$_BookmarkNames) //lire les signets du template
$targetRange:=WP Nouveau //créer un nouveau document vide (sera le document final)

$p:=Chercher dans tableau($_BookmarkNames;"Main_Header") //traiter l'entête principal nommé "Main_Header"
Si($p>0)
  $Range:=WP Lire plage signet(WParea;$_BookmarkNames{$p}) //sélectionner la plage
  $RangeDoc:=WP Nouveau($Range) //créer un nouveau document depuis la plage
  WP INSERER DOCUMENT($targetRange;$RangeDoc;wk append+wk freeze expressions) //wk append=après remplacement, $buildRange
est égal à la fin du texte remplacé
Fin de si
```

WP INSERER IMAGE

WP INSERER IMAGE (objPlage ; image ; mode { ; miseAJourPlage})

Paramètre	Type	Description
objPlage	Objet	→ Plage 4D Write Pro
image	Image	→ Champ ou variable Image, ou chemin d'accès à une image sur le disque
mode	Entier long	→ Mode d'insertion
miseAJourPlage	Entier long	→ Mode de mise à jour de la plage de sélection

Description

La commande **WP INSERER IMAGE** insère *image* dans la plage de sélection *objPlage* selon le *mode* d'insertion spécifié. L'image sera insérée en tant que caractère dans la plage de sélection *objPlage*.

Dans *objPlage*, passez une plage 4D Write Pro valide.

Dans *image*, vous pouvez passer :

- soit un champ image ou une variable image 4D,
- soit une chaîne contenant un chemin d'accès à un fichier image stocké sur disque, en utilisant la syntaxe système. Si vous utilisez une chaîne, vous pouvez passer soit un chemin d'accès complet, soit un chemin d'accès en relatif par rapport au fichier de structure de la base. Vous pouvez aussi passer un nom de fichier, dans ce cas le fichier doit être situé à côté du fichier de structure de la base. Si vous passez un nom de fichier, vous devez indiquer son extension.

Tous les formats d'image pris en charge par 4D peuvent être utilisés (voir la section **Introduction aux images**). Vous pouvez obtenir la liste des formats d'image disponibles en utilisant la commande **LISTE CODECS IMAGES**. Si l'*image* encapsule plusieurs formats (codecs), 4D Write Pro utilise un seul format pour l'affichage et un autre pour l'impression (s'il est différent) ; les formats les mieux adaptés sont automatiquement sélectionnés.

Dans le paramètre *mode*, passez une des constantes suivantes pour indiquer le mode d'insertion à utiliser pour insérer l'image dans le document :

Constante	Type	Valeur	Comment
wk append	Entier long	2	Insère le contenu à la fin de la plage
wk prepend	Entier long	1	Le contenu est inséré au début de la plage
wk replace	Entier long	0	Remplace le contenu de la plage

Dans le paramètre optionnel *miseAJourPlage*, vous pouvez passer une des constantes suivantes pour préciser si l'image est incluse ou non dans la plage de sélection résultante :

Constante	Type	Valeur	Comment
wk exclude from range	Entier long	1	Le contenu inséré n'est pas inclus dans la plage mise à jour
wk include in range	Entier long	0	Le contenu inséré est inclus dans la plage mise à jour (défaut)

Si vous ne passez pas le paramètre *miseAJourPlage*, par défaut l'image est incluse dans la plage mise à jour.

Exemple

Dans l'exemple suivant, un utilisateur choisit une image qu'il souhaite insérer dans une plage de sélection de l'objet 4D Write Pro, et obtiendra une alerte si l'image ne peut pas être insérée :

```
C_OBJET($wpRange)
$wpRange:=WP Lire selection([Courrier]ZoneWP)

C_BOOLEEN($fail)
$fail:=Faux

//on demande à l'utilisateur de choisir, sur le disque,l'image qu'il souhaite insérer
$imgRef:=Ouvrir document("")
//si l'utilisateur n'annule pas
Si(OK=1)
    //si le format du fichier image est pris en charge
    Si(Est un fichier image(document))
        // on insère l'image
        WP INSERER IMAGE($wpRange;document;wk replace)
    Sinon
        $fail:=Vrai
    Fin de si
Sinon
    $fail:=Vrai
Fin de si
//si l'insertion a échoué, on informe l'utilisateur
Si($fail)
    ALERTE("l'insertion de l'image a échoué")
Fin de si
```


WP INSERER RUPTURE

WP INSERER RUPTURE (objPlage ; typeRupture ; mode {; miseAJourPlage})

Paramètre	Type		Description
objPlage	Objet	→	Plage 4D Write Pro
typeRupture	Entier long	→	Type de rupture à insérer
mode	Entier long	→	Mode d'insertion
miseAJourPlage	Entier long	→	Mode de mise à jour de la plage de sélection

Description

La commande **WP INSERER RUPTURE** insère une nouvelle rupture de type *typeRupture* dans la plage de sélection *objPlage* selon le mode d'insertion *mode* et le paramètre *miseAJourPlage*.

Dans *objPlage*, passez une plage.

Dans *typeRupture*, passez une des constantes suivantes, du thème **Constantes 4D Write Pro**, pour définir le type de rupture à insérer :

Constante	Type	Valeur	Comment
wk column break	Entier long	3	Saut de colonne : insère un saut de colonne
wk line break	Entier long	0	Saut de ligne (dans le même paragraphe)
wk page break	Entier long	2	Saut de page : définit une nouvelle page
wk section break	Entier long	1	Saut de section: définit une nouvelle section

Dans le paramètre *mode*, passez une constante pour indiquer le mode d'insertion à utiliser pour la rupture dans la plage de sélection *objPlage* :

Constante	Type	Valeur	Comment
wk append	Entier long	2	Insère le contenu à la fin de la plage
wk prepend	Entier long	1	Le contenu est inséré au début de la plage
wk replace	Entier long	0	Remplace le contenu de la plage

Dans le paramètre optionnel *miseAJourPlage*, vous pouvez passer une des constantes suivantes, pour spécifier si le contenu est inclus dans la sélection résultante ou pas :

Constante	Type	Valeur	Comment
wk exclude from range	Entier long	1	Le contenu inséré n'est pas inclus dans la plage mise à jour
wk include in range	Entier long	0	Le contenu inséré est inclus dans la plage mise à jour (défaut)

Si vous ne passez pas le paramètre *miseAJourPlage*, par défaut, le contenu inséré sera inclus dans la plage mise à jour.

Exemple

En construisant vos factures, vous souhaitez insérer un saut de page à chaque facture, excepté sur la dernière page :

```
$nblInvoices:=Enregistrements trouves([INVOICE])
Boucle($;1;$nblInvoices)
... //traitement des factures
Si($j#command_5$nblInvoices) //on insère un saut de page, excepté sur la dernière
    WP INSERER RUPTURE($buildRange;wk.page.break;wk.append;wk.exclude.from.range)
Fin de si
Fin de boucle
```

WP Insérer tableau

WP Insérer tableau (objPlage ; mode {; miseAJourPlage {; nbCols {; nbLignes}} }) -> Résultat

Paramètre	Type		Description
objPlage	Objet	→	Plage 4D Write Pro
mode	Entier long	→	Mode d'insertion
miseAJourPlage	Entier long	→	Mode de mise à jour de la plage de sélection
nbCols	Entier long	→	Nombre de colonnes à créer dans le tableau
nbLignes	Entier long	→	Nombre de lignes à créer dans le tableau
Résultat	Objet	→	Objet listant les propriétés du tableau

Description

La commande **WP Insérer tableau** insère un nouveau tableau dans la plage *objPlage* en fonction du *mode* d'insertion spécifié et retourne le nouvel objet correspondant. Optionnellement, vous pouvez passer le paramètre *miseAJourPlage* ainsi qu'un nombre prédéfini de colonnes et de lignes dans *nbCols* et *nbLignes*.

Dans *objPlage*, passez une plage standard 4D Write Pro.

Note : L'insertion d'un tableau dans un autre tableau n'est pas possible (une erreur est retournée).

Dans le paramètre *mode*, passez une des constantes suivantes pour indiquer le mode d'insertion à utiliser pour insérer le tableau dans le document :

Constante	Type	Valeur	Comment
wk append	Entier long	2	Insère le contenu à la fin de la plage
wk prepend	Entier long	1	Le contenu est inséré au début de la plage
wk replace	Entier long	0	Remplace le contenu de la plage

Dans le paramètre optionnel *miseAJourPlage*, vous pouvez passer une des constantes suivantes pour spécifier si le tableau inséré fait partie ou pas de la plage :

Constante	Type	Valeur	Comment
wk exclude from range	Entier long	1	Le contenu inséré n'est pas inclus dans la plage mise à jour
wk include in range	Entier long	0	Le contenu inséré est inclus dans la plage mise à jour (défaut)

Si vous ne passez pas le paramètre *miseAJourPlage*, par défaut, le tableau inséré est inclus dans la plage.

Vous pouvez passer dans *nbCols* et *nbLignes* le nombre de colonnes et de lignes du tableau inséré. Par défaut, si *nbCols* et *nbLignes* sont omis, un nouveau tableau avec 0 colonne et 0 ligne est créé. Vous pouvez alors ajouter ces éléments en utilisant la commande **WP Tableau ajouter ligne**.

La commande retourne le nouveau tableau en tant qu'objet, que vous pouvez passer ensuite à une autre commande tableau 4D Write Pro. Le tableau est créé avec police, bordure, et attributs couleur par défaut, que vous pouvez personnaliser en utilisant la commande **WP FIXER ATTRIBUTS**.

Exemple

Dans un document vide, vous souhaitez ajouter un nouveau tableau avec 3 colonnes et 2 lignes :

```
C_OBJET($WPRange;$WPTable)
$WPRange:=WP Créer plage(WParea;wk start text;wk end text)
$WPTable:=WP Insérer tableau($WPRange;wk append;wk exclude from range;3;2)
```

Résultat :

⚙ WP LIRE ATTRIBUTS

WP LIRE ATTRIBUTS (objCible ; nomAttribut ; valeurAttribut { ; nomAttribut2 ; valeurAttribut2 ; ... ; nomAttributN ; valeurAttributN })

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément ou document 4D Write Pro
nomAttribut	Texte	→ Nom d'attribut dont vous souhaitez lire la valeur
valeurAttribut	Texte, Numérique, Tableau entier long, Tableau réel, Image, Date	← Valeur courante de l'attribut

Description

La commande **WP LIRE ATTRIBUTS** retourne la valeur courante d'attribut(s) dans une plage, une référence d'élément ou un document 4D Write Pro. Cette commande donne accès à tous les types d'attributs internes de 4D Write Pro : caractère, paragraphe, document, tableau ou image.

Dans le paramètre *objCible*, vous pouvez passer :

- un objet plage 4D Write Pro, ou
- une référence d'élément (en-tête / pied / corps / tableau / ligne / image en ligne ou ancrée), ou
- un document 4D Write Pro

Pour une liste complète des attributs que vous pouvez passer dans *nomAttribut* ainsi que leurs valeurs respectives retournées dans *valeurAttribut*, veuillez vous référer à la section **Attributs 4D Write Pro**.

S'il y a des valeurs différentes pour un même attribut dans l'élément passé en paramètre, la commande retourne :

- pour les valeurs numériques, wk mixed
- pour un tableau, un tableau vide (tabulations, couleur si *valeurAttribut* est défini en tant que tableau), avec une exception pour l'attribut wk text shadow offset pour lequel la valeur du tableau contiendra toujours 2 éléments qui peuvent être fixés séparément à wk mixed si soit le décalage horizontal soit le décalage vertical (ou les deux) sont mélangés.
- pour les valeurs chaîne, une chaîne vide
- pour les valeurs image, une image vide

Exemple

Vous souhaitez obtenir la couleur de fond de la zone sélectionnée :

```
$range:=WP Lire selection(*;"WParea")  
WP LIRE ATTRIBUTS($range;wk background color;$bcol)
```

WP Lire cadre

WP Lire cadre ({ * ; } zoneWP) -> Résultat

Paramètre	Type	Description
*	Opérateur	→ Si spécifié, zoneWP est un nom d'objet (chaîne). Si omis, zoneWP est un champ ou une variable
zoneWP	Chaîne	→ Nom d'objet de formulaire (si * spécifié) ou champ ou variable objet 4D Write Pro (si * omis)
Résultat	Entier long	↪ Cadre dans lequel se trouve le curseur

Description

La commande **WP Lire cadre** retourne le cadre de la zone 4D Write Pro *zoneWP* dans lequel se trouve actuellement le curseur. Si vous passez le paramètre optionnel *, vous indiquez que le paramètre *zoneWP* est un nom d'objet (une chaîne). Si vous ne passez pas le paramètre, vous indiquez que le paramètre *zoneWP* est un champ ou une variable objet.

La valeur retournée peut être comparée à l'une des constantes suivantes du thème **Constantes 4D Write Pro** :

Constante	Type	Valeur	Comment
wk body	Entier long	0	Zone de corps (<i>body</i>) de la section
wk current section default footer	Entier long	10	Pied de page par défaut de la section (disponible uniquement s'il n'y a pas de pied de sous-section active).
wk current section default header	Entier long	9	En-tête par défaut de la section (disponible uniquement s'il n'y a pas d'en-tête de sous-section active).
wk current section first footer	Entier long	4	Pied de la première page de la section
wk current section first header	Entier long	3	En-tête de la première page de la section
wk current section left footer	Entier long	6	Pied de la (des) page(s) gauche(s) de la section
wk current section left header	Entier long	5	En-tête de la (des) page(s) gauche(s) de la section
wk current section right footer	Entier long	8	Pied de la (des) page(s) droite(s) de la section
wk current section right header	Entier long	7	En-tête de la (des) page(s) droite(s) de la section

Exemple

Vous souhaitez vous assurer que l'utilisateur a placé le curseur dans un cadre d'en-tête ou de pied de page :

```
C_ENTIER LONG($frameGet)
$frameGet:=WP Lire cadre(*;"WParea")
Si($frameGet=wk body)
  ALERTE("Veuillez sélectionner une zone d'en-tête ou de pied de page.")
Fin de si
```

⚙️ WP Lire corps

WP Lire corps (docWP) -> Résultat

Paramètre	Type		Description
docWP	Objet	→	Document 4D Write Pro
Résultat	Objet	↩	Référence de corps de 4D Write Pro

Description

La commande **WP Lire corps** retourne une référence vers la partie corps (*body*) du document 4D Write Pro *docWP*. Cette référence peut ensuite être passée aux commandes :

- **WP SELECTIONNER** pour sélectionner le corps du document.
- **WP Creer plage** pour définir le corps en tant que plage.
- **WP FIXER ATTRIBUTS** et **WP LIRE ATTRIBUTS** pour gérer les attributs du corps du document.

Exemple

Vous voulez copier le corps d'un document 4D Write Pro dans un autre document 4D Write Pro :

```
C_OBJET($bodySource;$rangeSource;$tempoc;$bodyTarget;$rangeTarget)
```

```
$bodySource:=WP Lire corps([TEMPLATES]WPtemplate)
```

```
$rangeSource:=WP Creer plage($bodySource;wk start text;wk end text)
```

```
$tempoc:=WP Nouveau($rangeSource)
```

```
$bodyTarget:=WP Lire corps([Docs]MyWPDoc)
```

```
$rangeTarget:=WP Creer plage($bodyTarget;wk start text;wk end text)
```

```
WP INSERER DOCUMENT($rangeTarget;$tempoc;wk replace)
```

⚙️ WP Lire element par ID

WP Lire element par ID (docWP ; ID) -> Résultat

Paramètre	Type		Description
docWP	Objet	→	Document 4D Write Pro
ID	Chaîne	→	Identifiant de l'élément à retrouver
Résultat	Objet	↩	Référence de l'élément 4D Write Pro

Description

La commande **WP Lire element par ID** retourne un objet contenant une référence à l'élément de *docWP* passé dans le paramètre *ID*.

Dans le paramètre *docWP*, passez une référence de document 4D Write Pro.

Passez l'attribut ID de l'élément à retrouver dans le paramètre *ID*. L'élément peut être à l'intérieur du document, comme une image, un paragraphe, un tableau, une ligne, etc., ou le document lui-même. Si aucun élément dans le document ne possède l'ID spécifié, une valeur **null** est retournée.

Exemple 1

Pour obtenir une référence d'une image avec l'ID "img1" :

```
C_OBJET(oblImage)
oblImage:=WP Lire element par ID(myDoc;"img1")
```

Exemple 2

Vous souhaitez retrouver un élément tableau et le modifier en l'entourant d'une large bordure violette :

```
C_OBJET($element)
$element:=WP Lire element par ID(myDoc;"Table1")
WP FIXER ATTRIBUTS($element;wk border style;wk solid)
WP FIXER ATTRIBUTS($element;wk border width;"4px")
WP FIXER ATTRIBUTS($element;wk border color;"purple")
```

Exemple 3

Vous souhaitez retrouver un élément et modifier son ID :

```
$colTable:=WP Creer plage tableaux([INFO]Sample)
$elements:=WP Lire elements($colTable)
$elem:=WP Lire element par ID([INFO]Sample;"Paris")
Si($elem#command_5Null)
  $elem.ID:="Lyon"
Fin de si
```

WP Lire elements

WP Lire elements (objCible {; typeElement}) -> Résultat

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément ou document 4D Write Pro
typeElement	Entier long	→ Choix du type d'éléments à retrouver
Résultat	Collection	↻ Collection contenant des références d'éléments

Description

La commande **WP Lire elements** retourne une collection d'objets contenant des références aux paragraphes, images, tableaux et/ou lignes de tableaux.

Dans le paramètre *objCible*, vous pouvez passer :

- une plage, ou
- une référence à un élément (tableau / ligne / paragraphe / corps / en-tête / pied), ou
- un document 4D Write Pro

Si une plage ou une référence typée est passée, la commande renverra une collection contenant uniquement les éléments du type correspondant (sauf si vous spécifiez un type grâce au paramètre *typeElement*, voir ci-dessous). Sinon, la commande retourne une collection contenant tous les éléments disponibles dans *objCible*.

Vous pouvez passer le paramètre optionnel *typeElement* pour spécifier les types d'éléments à renvoyer dans la collection. Les types suivants sont disponibles :

Constante	Valeur	Comment
wk type default	0	Type de plage par défaut
wk type image	2	Images ancrées et/ou en ligne
wk type image anchored	200	Référence d'image ancrée pour objets documents
wk type image inline	100	Référence d'image en ligne pour objets documents
wk type paragraph	1	Plage de type paragraphe
wk type table	4	Référence de tableau
wk type table row	5	Référence de ligne de tableau

Note :

- Si vous passez le sélecteur wk type default, la commande renvoie une collection non filtrée.
- Si vous passez le sélecteur wk type paragraph, wk type image ou wk type table, la collection retournée contiendra uniquement les éléments du type indiqué, même s'ils diffèrent du type spécifié dans le paramètre *typeElement*.
- Les éléments de la collection retournée sont classés en fonction de leur ordre d'apparition dans *objCible*, sauf lors de la transmission d'une référence de document 4D Write Pro. Dans ce cas, les éléments seront classés par leur ID.
- Si une plage indiquée dans *objCible* contient un élément qui n'est pas entièrement contenu dans la plage, l'élément entier sera inclus dans les résultats. Par exemple, si la plage inclut une partie d'un tableau (mais pas tout le tableau), la référence de l'élément tableau entier est renvoyée, en revanche seules les références de cellules (paragraphes) comprises dans la plage sont incluses.

Exemple 1

Vous voulez récupérer une collection de tous les paragraphes dans un document :

```
C_COLLECTION($docElements)
$docElements:=WP Lire elements($myDoc;wk type paragraph)
//retourne une collection contenant seulement les éléments paragraphes
```

Exemple 2

Vous souhaitez alterner l'alignement et la couleur des paragraphes dans un document :

```
C_COLLECTION($col)
C_OBJET($obj)
C_OBJET($body)
C_ENTIER LONG($i;$n)
$col:=Creer collection
$body:=WP Lire corps(myDoc)
$col:=WP Lire elements($body;wk type paragraph)
$i:=0
$n:=$col.length
Boucle($i;0;$n-1)
  $obj:=$col[$i]
  WP REINITIALISER ATTRIBUTS($obj)
```


WP FIXER ATTRIBUTS(\$obj;wk font bold;wk false)

WP FIXER ATTRIBUTS(\$obj;wk font italic;wk false)

Si(\$i%2=0)

WP FIXER ATTRIBUTS(\$obj;wk text align;wk left)

WP FIXER ATTRIBUTS(\$obj;wk font bold;wk true)

WP FIXER ATTRIBUTS(\$obj;wk text color;"#command_5804040")

WP FIXER ATTRIBUTS(\$obj;wk margin right;"5cm")

WP FIXER ATTRIBUTS(\$obj;wk margin left;"1cm")

Sinon

WP FIXER ATTRIBUTS(\$obj;wk text align;wk right)

WP FIXER ATTRIBUTS(\$obj;wk font italic;wk true)

WP FIXER ATTRIBUTS(\$obj;wk text color;"#command_5404020")

WP FIXER ATTRIBUTS(\$obj;wk margin left;"5cm")

WP FIXER ATTRIBUTS(\$obj;wk margin right;"1cm")

Fin de si

Fin de boucle

WP Lire entete

WP Lire entete (objPlage) | (docWP ; indexSection {; typeSousSection}) -> Résultat

Paramètre	Type	Description
objPlage	Objet	→ Plage 4D Write Pro
docWP	Objet	→ Document 4D Write Pro
indexSection	Entier long	→ Numéro de la section
typeSousSection	Entier long	→ Type de sous-section (wk first page, wk left page ou wk right page)
Résultat	Objet	→ Référence d'en-tête 4D Write Pro

Description

La commande **WP Lire entete** retourne une référence vers l'en-tête associé à la plage *objPlage* ou à la section *indexSection* du document 4D Write Pro *docWP*.

- Avec la première syntaxe (utilisation du paramètre *objPlage*), la commande retourne la référence de l'en-tête de la première page de la plage. Si une sous-section est active sur la page, la commande retourne l'en-tête de la sous-section active, elle retourne ainsi toujours l'en-tête visible.
- Avec la seconde syntaxe (utilisation du paramètre *docWP*), la commande retourne la référence de l'en-tête de la section *indexSection* spécifiée (la valeur de *indexSection* doit être ≥ 1). Si le paramètre *typeSousSection* est passé, la commande retourne la référence de l'en-tête associé au type de sous-section spécifié. Vous pouvez utiliser une des constantes suivantes dans le paramètre *typeSousSection* :

Constante	Type	Valeur
wk first page	Entier long	1
wk left page	Entier long	2
wk right page	Entier long	3

Si le *typeSousSection* n'est pas spécifié, l'en-tête de section principal est retourné. Dans ce cas, si une sous-section est active, l'objet en-tête de section principal est *undefined* (l'en-tête de section principal est disponible uniquement s'il n'y a pas de sous-section wk first page ou wk left page active).

Exemple

Vous souhaitez copier l'en-tête de la première page d'un document 4D Write Pro vers les en-têtes de toutes les pages d'un autre document 4D Write Pro :

```
C_OBJET($headerSource;$rangeSource;$tempoc;$headerTarget;$rangeTarget)
```

```
$headerSource:=WP Lire entete([TEMPLATES]WPtemplate;1;wk first page) //récupérer l'en-tête de la 1re page de la section 1
```

```
$rangeSource:=WP Creer plage($headerSource;wk start text;wk end text)
```

```
$tempoc:=WP Nouveau($rangeSource)
```

```
$headerTarget:=WP Lire entete([Docs]MyWPDoc;1) //toutes les pages de la section 1
```

```
$rangeTarget:=WP Creer plage($headerTarget;wk start text;wk end text)
```

```
WP INSERER DOCUMENT($rangeTarget;$tempoc;wk replace)
```

WP Lire pied

WP Lire pied (objPlage) | (docWP ; indexSection { ; typeSousSection }) -> Résultat

Paramètre	Type	Description
objPlage	Objet	→ Plage 4D Write Pro
docWP	Objet	→ Document 4D Write Pro
indexSection	Entier long	→ Numéro de la section
typeSousSection	Entier long	→ Type de sous-section (wk first page, wk left page ou wk right page)
Résultat	Objet	→ Référence de pied de page 4D Write Pro

Description

La commande **WP Lire pied** retourne une référence vers le pied de page associé à la plage *objPlage* ou à la section *indexSection* du document 4D Write Pro *docWP*.

- Avec la première syntaxe (utilisation du paramètre *objPlage*), la commande retourne la référence du pied de la première page de la plage. Si une sous-section est active sur la page, la commande retourne le pied de page de la sous-section active, elle retourne ainsi toujours le pied de page visible.
- Avec la seconde syntaxe (utilisation du paramètre *docWP*), la commande retourne la référence du pied de la section *indexSection* spécifiée (la valeur de *indexSection* doit être ≥ 1).
Si le paramètre *typeSousSection* est passé, la commande retourne la référence du pied associé au type de sous-section spécifié. Vous pouvez utiliser une des constantes suivantes dans le paramètre *typeSousSection* :

Constante	Type	Valeur
wk first page	Entier long	1
wk left page	Entier long	2
wk right page	Entier long	3

Si le *typeSousSection* n'est pas spécifié, le pied de section principal est retourné. Dans ce cas, si une sous-section est active, l'objet pied de section principal est *undefined* (le pied de section principal est disponible uniquement s'il n'y a pas de sous-section wk first page ou wk left page active).

Exemple

Vous voulez copier le pied de la première page d'un document 4D Write Pro vers les pieds de toutes les pages d'un autre document 4D Write Pro :

```
C_OBJET($footerSource;$rangeSource;$tempoc;$footerTarget;$rangeTarget)
```

```
$footerSource:=WP Lire pied([TEMPLATES]WPtemplate;1;wk first page) //récupérer le pied de la 1re page de la section 1
```

```
$rangeSource:=WP Creer plage($footerSource;wk start text;wk end text)
```

```
$tempoc:=WP Nouveau($rangeSource)
```

```
$footerTarget:=WP Lire pied([Docs]MyWPDoc;1) //toutes les pages de la section 1
```

```
$rangeTarget:=WP Creer plage($footerTarget;wk start text;wk end text)
```

```
WP INSERER DOCUMENT($rangeTarget;$tempoc;wk replace)
```

⚙️ WP Lire plage signet

WP Lire plage signet (docWP ; nomSignet) -> Résultat

Paramètre	Type		Description
docWP	Objet	→	Document 4D Write Pro
nomSignet	Texte	→	Nom du signet dont vous souhaitez récupérer la plage
Résultat	Objet	↩	Plage du signet

Description

La commande **WP Lire plage signet** retourne un objet de type plage (*objPlage*) contenant la plage associée au signet nommé *nomSignet* dans le document *docWP*.

Si le signet *nomSignet* n'existe pas dans *docWP*, un objet plage *objPlage* vide est retourné.

Exemple

Vous souhaitez faire apparaître la plage associée au signet "MyBookmark" dans votre document :

```
C_OBJET($wpRange)
$wpRange:=WP Lire plage signet(WParea;"MyBookmark")
WP SELECTIONNER(WParea;$wpRange)
```

WP Lire position

WP Lire position (objCible {; presentation}) -> Résultat

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément ou document 4D Write Pro
presentation	Entier long	→ Mise en page de document 4D Write Pro utilisée pour la présentation : 0 (par défaut) = mise en page 4D Write Pro, 1 = HTML WYSIWYG
Résultat	Objet	→ Informations sur la position

Description

La commande **WP Lire position** retourne un objet décrivant la position courante de *objCible*.

Dans le paramètre *objCible* vous pouvez passer :

- une plage, ou
- une référence d'élément (tableau / ligne / paragraphe / corps / en-tête / pied / image), ou
- un document 4D Write Pro

La commande retourne des informations sur la position dans un objet avec les propriétés suivantes :

Nom de la propriété	Type	Description
section	Nombre	Numéro de la première section qui croise l' <i>objCible</i>
page	Nombre	Numéro de la première page qui croise l' <i>objCible</i>
column	Nombre	Index de la première colonne qui croise l' <i>objCible</i> (base 1)
line	Nombre	Index de la première ligne qui croise l' <i>objCible</i> (par rapport à la colonne -ou la page si aucune colonne- base 1)
position	Nombre	Position dans la ligne du premier caractère de l' <i>objCible</i> (base 1)

Si *objCible* est une plage vide, **WP Lire position** retourne des informations sur la position du début de la plage.

Si *objCible* référence un texte en en-tête ou en pied, **WP Lire position** retourne des informations sur la position où l'en-tête et le pied de page sont affichés dans la première section/page.

Si *objCible* référence une image ancrée (qui peut être répliquée sur plusieurs pages), **WP Lire position** renvoie des informations sur la position où l'image est ancrée dans la première section/page.

Le paramètre *presentation* peut être utilisé pour définir du wysiwyg HTML pour l'évaluation de l'*objCible*. Vous pouvez passer une des constantes suivantes du thème "4D Write Pro" :

Constante	Type	Valeur	Comment
wk 4D Write Pro layout	Entier long	0	Format 4D Write Pro standard, pouvant inclure des attributs de style spécifiques
wk html wysiwyg	Entier long	1	Dans ce format, les éventuels attributs avancés de 4D Write Pro non disponibles dans tous les navigateurs sont supprimés (multi-colonnes, double interlignes...)

Si *presentation* est omis, la présentation 4D Write Pro (0) est utilisée par défaut.

Les mêmes paramétrages de présentation que pour la commande **WP IMPRIMER** sont utilisés avec **WP Lire position** :

- en-têtes : visible
- pieds : visible
- expressions : calculées et affichées
- mode page : page ou brouillon (vue wysiwyg HTML telle que définie par la commande)

Si *objCible* est affiché dans une zone formulaire 4D Write Pro, assurez-vous que les paramètres d'affichage de la zone correspondent aux paramètres d'affichage de la commande afin d'obtenir des informations cohérentes. Notez également que si ces paramètres sont différents dans la zone, 4D Write Pro devra "cloner" le document pour calculer la présentation à chaque appel de la commande, ce qui peut prendre un certain temps. Pour des raisons de performances, il est recommandé dans ce cas de créer le document hors ligne et de le copier dans la zone sur le formulaire uniquement lorsque la génération est terminée.

Exemple 1

Vous souhaitez connaître le numéro de page d'une plage :

```
C_OBJET($range;$rangePosition)
$range:=WP Lire selection(WParea) //plage de la sélection de l'utilisateur
$rangePosition:=WP Lire position($range;wk 4D Write Pro layout)
ALERTE("Votre sélection se trouve en page "+Chaine($rangePosition.page))
```

Exemple 2

Vous voulez appliquer la couleur bleue pour le texte de la première et de la dernière page d'un document :

```
C_ENTIER LONG($nbPages)
C_OBJET($body)
C_COLLECTION($_paragraphs)
```

```

$nbPages:=WP Nombre de pages([DOC]Sample)
$body:=WP Lire corps([DOC]Sample)
$_paragraphs:=WP Lire elements($body;wk type paragraph)
Pour chaque($paragraph;$_paragraphs)
  $info:=WP Lire position($paragraph)
  Si($info.page=1)|($info.page=$nbPages) // pour la première et la dernière page, couleur du texte en bleu
    WP FIXER ATTRIBUTS($paragraph;wk text color;"blue")
  Sinon //pour les paragraphes des autres pages, le texte est en noir
    WP FIXER ATTRIBUTS($paragraph;wk text color;"black")
  Fin de si
Fin de chaque

```

Exemple 3

Lors de la création d'un document, vous souhaitez éviter les fractionnements de paragraphes :

```

C_OBJET(WParea;$body)
C_COLLECTION($_paragraphs)
WParea:=WP Nouveau

$body:=WP Lire corps([DOC]Sample)
$_paragraphs:=WP Lire elements($body;wk type paragraph)

Pour chaque($paragraph;$_paragraphs)
  $insert:=WP Nouveau($paragraph)

  //mémorise la plage courante et le numéro de page
  $rangeBefore:=WP Creer plage(WParea;wk end text;wk end text)
  $info:=WP Lire position($rangeBefore)
  $memoPage:=$info.page

  //insère les contenus, sans toucher à la plage
  WP INSERER DOCUMENT($rangeBefore;$insert;wk append;wk exclude from range)

  //vérifie la position après l'insertion
  $rangeAfter:=WP Creer plage(WParea;wk end text;wk end text)
  $info:=WP Lire position($rangeAfter)

  Si($memoPage#command_5$info.page) // si la page a changé, insertion d'un saut de page
    WP INSERER RUPTURE($rangeBefore;wk page break;wk replace)
  fin de si
Fin de chaque

```

⚙ WP Lire selection

WP Lire selection ({* ;} zoneWP) -> Résultat

Paramètre	Type	Description
*	Opérateur	→ Si spécifié, zoneWP est un nom d'objet (chaîne). Si omis, zoneWP est un champ ou une variable
zoneWP	Chaîne	→ Nom d'objet de formulaire (si * spécifié) ou champ ou variable objet 4D Write Pro (si * omis)
Résultat	Objet	↪ Nouvel objet plage ou image

Description

La commande **WP Lire selection** retourne une nouvelle plage de sélection ou une nouvelle image basée sur la sélection courante dans la zone 4D Write Pro *zoneWP*.

Si vous passez le paramètre optionnel *, vous indiquez que le paramètre *zoneWP* est un nom d'objet (une chaîne). Si vous ne passez pas le paramètre, vous indiquez que le paramètre *zoneWP* est un champ ou une variable objet. Si le paramètre *zoneWP* ne désigne pas une zone 4D Write Pro valide, un objet vide est retourné.

Note : Cette commande ne peut être utilisée que si *zoneWP* est associée à un objet de formulaire (i.e. elle est affichée dans le formulaire).

Si une image ancrée (image à emplacement fixe) est sélectionnée dans *wpArea*, **WP Lire selection** retourne un objet contenant une référence d'image et non un objet plage (pour plus d'informations, reportez-vous à la description de la commande **WP Ajouter image**). Sinon, **WP Lire selection** retourne un nouvel objet de type *objPlage* (pour plus d'informations, reportez-vous au paragraphe **Gestion des plages**).

Exemple

Vous souhaitez récupérer le texte sélectionné dans une zone 4D Write Pro :

```
$range:=WP Lire selection(*;"WParea")
```

WP LIRE SIGNETS

WP LIRE SIGNETS (docWP ; tabNomsSignets)

Paramètre	Type		Description
docWP	Objet	⇒	Document 4D Write Pro
tabNomsSignets	Tableau texte	⇐	Tableau des noms de signets

Description

La commande **WP LIRE SIGNETS** retourne un tableau contenant le nom de tous les signets définis dans le document *docWP*. Après l'exécution de la commande, le tableau *tabNomSignets* est rempli avec tous les noms des signets du document. Dans le tableau, les noms sont triés en fonction de leur position dans le document. Si plusieurs signets débutent à la même position, ils sont triés par ordre alphabétique.

Exemple

Vous souhaitez connaître le nombre de signets définis dans votre document :

```
TABLEAU TEXTE($_bookmarks;0)
WP LIRE SIGNETS(WParea;$_bookmarks)
ALERTE("Le document contient "+Taille tableau($_bookmarks)+" signets.")
```


WP Lire texte

WP Lire texte (objCible {; expressions}) -> Résultat

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément ou document 4D Write Pro
expressions	Numérique	→ Instructions pour l'évaluation de l'expression
Résultat	Texte	↻ Texte de objCible

Description

La commande **WP Lire texte** récupère le contenu textuel d'un document 4D Write Pro dans l'*objCible* spécifié.

WP Lire texte renvoie uniquement du texte brut. Tous les styles appliqués (gras, italique, etc.) seront ignorés.

Dans le paramètre *objCible*, passez un objet contenant :

- une plage, ou
- une référence d'élément (tableau / paragraphe / corps / en-tête / pied), ou
- un document 4D Write Pro

Le paramètre optionnel *expressions* fournit des instructions pour l'évaluation des expressions 4D dans l'*objCible*. Les options sont :

Constante	Comment
wk expressions as source	La chaîne d'origine des références des expressions 4D est retournée.
wk expressions as space	Chaque référence est retournée en tant qu'espace insécable.
wk expressions as value	Les références d'expressions 4D sont retournées dans leur forme évaluée (option par défaut).

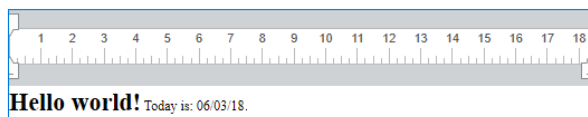
Important : Seul le texte brut avec [wk expressions as space](#) est cohérent avec les index de plage de texte utilisés par les commandes WP ou ST, car ces commandes présument toujours que la taille de l'expression = 1 caractère.

Notes :

- Si *objCible* contient des images, elles sont ignorées et renvoyées sous forme de caractères espaces (" ").
- Si *objCible* contient des tableaux, le contenu de chaque cellule est traité comme des paragraphes individuels et renvoyé en tant que texte séparé par des tabulations. Les lignes sont séparées par des retours chariot.

Exemple

Pour retrouver le texte dans ce document :



Vous pouvez écrire :

```
$range:=WP Creer plage(myDoc2;wk start text;wk end text)
vText:=WP Lire texte($range;wk expressions as value) // retourne "Hello world! Today is: 06/03/18."
vText2:=WP Lire texte($range;wk expressions as source) //retourne "Hello world! Today is: date du jour.
vText3:=WP Lire texte($range;wk expressions as space) //retourne "Hello world! Today is: ."
```

⚙️ WP Nombre de pages

WP Nombre de pages (docWP) -> Résultat

Paramètre	Type		Description
docWP	Objet	➔	Document 4D Write Pro
Résultat	Entier long	➡	Nombre de pages dans le document

Description

La commande **WP Nombre de pages** retourne le nombre total de pages du document 4D Write Pro *docWP*.
Les mêmes paramètres d'affichage que la commande **WP IMPRIMER** sont utilisés avec **WP Nombre de pages** :

- en-têtes : visibles
- pieds : visibles
- expressions : calculées et affichées
- mode d'affichage : page ou brouillon

Si *docWP* est affiché dans une zone de formulaire 4D Write Pro, assurez-vous que les paramètres d'affichage de la zone correspondent à ceux de la commande afin d'obtenir des informations cohérentes. Notez également que si ces paramètres sont différents dans la zone, 4D Write Pro devra "cloner" le document afin de calculer son format à chaque appel de la commande, ce qui peut prendre du temps. Pour des raisons de performances, il est recommandé dans ce cas de construire le document hors-écran et de le copier dans la zone de formulaire une fois les calculs terminés.

Exemple

Vous souhaitez connaître le nombre total de pages des documents 4D Write Pro stockés dans le champ "Manual" de votre sélection courante. Vous pouvez écrire :

```
C_ENTIER LONG($pageCount)
C_ENTIER LONG($totalCount)
DEBUT SELECTION([Items])
Tant que(Non(Fin de selection([Items])))
    $pageCount:=WP Nombre de pages([Items]Manual)
    $totalCount:=$totalCount+$pageCount
ENREGISTREMENT SUIVANT([Items])
Fin tant que
ALERTE("Nombre total de pages des manuels de la sélection : "+Chaine($totalCount))
```

WP Nouveau

WP Nouveau {{ source }} -> Résultat

Paramètre	Type	Description
source	Chaîne, BLOB, Objet	→ Chaîne : Source HTML 4D, BLOB : Blob document 4D Write (.4w7/.4wt) ou document 4D Write Pro (.4wp), Objet : Objet plage ou référence d'élément 4D Write Pro
Résultat	Objet	↪ Objet 4D Write Pro

Description

La commande **WP Nouveau** crée et retourne un nouvel objet 4D Write Pro.

Par défaut, si vous omettez le paramètre *source*, la commande retourne un objet 4D Write Pro vide.

Vous pouvez également utiliser le paramètre *source*, auquel cas le nouvel objet 4D Write Pro sera prérempli avec le contenu de *source*. Vous pouvez passer :

- un paramètre *chaîne* : dans ce cas, vous passez une source de type HTML 4D, c'est-à-dire un texte exporté par **WP EXPORTER VARIABLE** avec l'option [wk web_page_html 4D](#). Ce texte peut contenir des références (balises et expressions 4D) et des images embarquées.
- un paramètre *blob* : dans ce cas, vous passez soit :
 - un document au format 4D Write Pro (.4wp) stocké dans un BLOB. Pour plus d'informations sur le format de document 4D Write Pro, veuillez vous reporter au paragraphe **Format de document .4wp**.
 - une ancienne zone 4D Write chargée dans un BLOB (les Blobs contenant des documents .4w7 ou .4wt sont acceptés). Pour une liste détaillée des fonctions 4D Write qui sont actuellement prises en charge dans les objets 4D Write Pro, veuillez vous référer à la section **Importer des documents 4D Write**. Si vous souhaitez importer un document 4D Write (.4w7 ou .4wt) stocké sur disque, utilisez plutôt la commande **WP Importer document**.
- un paramètre *objet* : dans ce cas, vous passez soit :
 - une plage ; dans ce cas, **WP Nouveau** retournera un nouveau document créé à partir de cette plage spécifique. A noter que si la plage n'est pas égale à la totalité du document, seule la première section est exportée et les signets, s'il y en a, ne sont pas exportés.
 - une référence d'élément (tableau / ligne / paragraphe / image en ligne / en-tête / pied de page) ; dans ce cas, **WP Nouveau** retournera un nouveau document dont le corps (*body*) sera constitué du contenu de l'élément (pour une ligne de tableau, le document contiendra un élément tableau constitué de la ligne).

L'objet retourné est un document complet pouvant être passé à la commande **WP INSERER DOCUMENT**, par exemple.

Exemple 1

Vous voulez créer un objet 4D Write Pro vide :

```
myWPObject:=WP Nouveau
```

Exemple 2

Vous voulez créer un objet 4D Write Pro contenant une simple référence d'expression 4D :

```
C_TEXTE(myText)
myText:="Nous sommes le "
ST INSERER EXPRESSION(myText;"chaîne(date du jour;Système date long)";ST Fin texte)
myWPA:=WP Nouveau(myText)
```

Exemple 3

Vous voulez initialiser votre zone 4D Write pro avec un template précédemment créé :

```
//Exporter un template depuis une zone existante
C_TEXTE(wpTemplate)
WP EXPORTER VARIABLE(myWPArea;wpTemplate;wk_page_web_html 4D)

// Utiliser le template pour une nouvelle zone
C_OBJET(myNewWPA)
myNewWPA:=WP Nouveau(wpTemplate)
```

Exemple 4

Vous voulez importer dans une zone 4D Write Pro un document 4D Write stocké dans un champ de l'enregistrement courant :

```
C_OBJET(wpArea)
wpArea:=WP Nouveau([Templates]Reference_)
```

Exemple 5

Vous avez défini un document modèle (*template*) comportant différentes parties préformatées, chacune d'elles étant stockée en tant que signet. Lorsque vous produisez un document final à partir du *template*, vous pouvez traiter chaque signet en tant que nouveau document et l'insérer dans le document final.

```
TABLEAU TEXTE($_BookmarkNames;0)
WP LIRE SIGNETS([TEMPLATES]WP;$_BookmarkNames) //lire les signets du template
$targetRange:=WP Nouveau //créer un nouveau document vide (sera le document final)

$p:=Chercher dans tableau($_BookmarkNames;"Main_Header") //traiter l'entête principal nommé "Main_Header"
Si($p>0)
  $Range:=WP Lire plage signet(WParea;$_BookmarkNames{$p}) //sélectionner la plage
  $RangeDoc:=WP Nouveau($Range) //créer un nouveau document depuis la plage
  WP INSERER DOCUMENT($targetRange;$RangeDoc;wk append+wk freeze expressions) //wk append=après remplacement, $buildRange
  est égal à la fin du texte remplacé
Fin de si
```

WP REINITIALISER ATTRIBUTS

WP REINITIALISER ATTRIBUTS (objCible ; nomAttribut { ; nomAttribut2 ; ... ; nomAttributN })

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément ou document 4D Write Pro
nomAttribut	Chaîne	→ Nom(s) d'attribut(s) à réinitialiser

Description

La commande **WP REINITIALISER ATTRIBUTS** vous permet de réinitialiser la valeur d'un ou plusieurs attributs dans la plage, la référence d'élément ou le document 4D Write Pro passé(e) en paramètre. Cette commande peut traiter tout type d'attribut 4D Write Pro : caractère, paragraphe, document, table ou image.

Dans le paramètre *objCible*, vous pouvez passer :

- un objet plage 4D Write Pro, ou
- une référence d'élément (en-tête / pied / corps / tableau / ligne / image en ligne ou ancrée), ou
- un document 4D Write Pro.

Lorsqu'une valeur d'attribut est supprimée à l'aide de la commande **WP REINITIALISER ATTRIBUTS**, elle est réinitialisée à sa valeur par défaut. Les valeurs par défaut sont listées dans la section **Attributs 4D Write Pro**.

Si l'attribut à réinitialiser n'était pas défini dans l'élément passé en paramètre, la commande ne fait rien.

Exemple

Vous souhaitez supprimer plusieurs attributs de la sélection suivante :

Vous pouvez exécuter :

```
$range:=WP Lire selection(*;"WParea")
WP REINITIALISER ATTRIBUTS($range;wk padding)
WP REINITIALISER ATTRIBUTS($range;wk background color)
WP REINITIALISER ATTRIBUTS($range;wk text underline style)
WP REINITIALISER ATTRIBUTS($range;wk margin)
WP REINITIALISER ATTRIBUTS($range;wk border style)
```

Le document résultant est alors :

WP SELECTIONNER

WP SELECTIONNER ({ * ; } zoneWP { ; objCible } | { ; débutPlage ; finPlage })

Paramètre	Type	Description
*	Opérateur	→ Si spécifié, zoneWP est un nom d'objet (chaîne). Si omis, zoneWP est un champ ou une variable
zoneWP	Chaîne, Objet	→ Nom d'objet de formulaire (si * spécifié) ou champ ou variable objet 4D Write Pro (si * omis)
objCible	Objet	→ Plage ou référence d'élément à utiliser pour créer une sélection
débutPlage	Entier long	→ Position du début de la plage
finPlage	Entier long	→ Position de la fin de la plage

Description

La commande **WP SELECTIONNER** crée une nouvelle sélection dans la zone de 4D Write Pro *zoneWP*, basée sur l'*objCible* ou la plage définie par *débutPlage* et *finPlage*.

Si vous passez le paramètre optionnel *, vous indiquez que le paramètre *zoneWP* est un nom d'objet (une chaîne). Si vous ne passez pas le paramètre, vous indiquez que le paramètre *zoneWP* est un champ ou une variable objet. Si le paramètre *zoneWP* ne désigne pas une zone 4D Write Pro valide, la commande ne fait rien.

Note : Cette commande ne peut être utilisée que si *zoneWP* est associée à un objet de formulaire (i.e. elle est affichée dans le formulaire).

Pour définir la sélection, vous pouvez passer soit un objet existant dans *objCible*, soit des bornes de sélection dans *débutPlage* / *finPlage* :

- **Première syntaxe :** `WP SELECTIONNER({* ; } zoneWP; objCible)`
objCible peut être :
 - soit une plage 4D Write Pro
 - soit une référence d'élément (tableau / ligne / paragraphe / image en ligne ou ancrée / en-tête / corps / pied de page).
- **Seconde syntaxe :** `WP SELECTIONNER({* ; } zoneWP; débutPlage ; finPlage)`
Passez dans *débutPlage* et *finPlage* des valeurs correspondant à la position du premier et du dernier caractère à sélectionner dans le document. Vous pouvez passer wk start text dans *débutPlage* pour désigner le début du document et wk end text dans *finPlage* pour désigner la fin du document. N'oubliez pas qu'un document 4D Write Pro ne contient pas seulement du texte visible mais également des caractères de formatage et des balises qui sont inclus dans la plage.

Exemple

Le code suivant :

```
$range2:=WP Creer plage([SAMPLE]WP;wk start text;12)
WP SELECTIONNER(*;"WParea";$range2)
```

... aura le même résultat que :

```
WP SELECTIONNER(*;"WParea";wk start text;12)
```

WP Style pris en charge

WP Style pris en charge (objCible ; stylePoliceWP) -> Résultat

Paramètre	Type	Description
objCible	Objet	→ Plage ou référence d'élément
stylePoliceWP	Entier long	→ Constante de style de police : wk font bold, wk font italic, wk text underline style, wk text linethrough style
Résultat	Booléen	→ Vrai si le style est pris en charge dans tout ou partie de objCible, Faux sinon

Description

La commande **WP Style pris en charge** retourne Vrai si le style *stylePoliceWP* est pris en charge par au moins une partie du texte inclus dans *objCible*.

Dans *objCible*, vous pouvez passer :

- une plage, ou
- une référence d'élément (tableau / ligne / paragraphe / image en ligne / en-tête / corps / pied de page).

Dans *stylePoliceWP*, vous pouvez passer une des constantes de style suivantes, placées dans le thème "4D Write Pro" :

Constante	Type	Valeur	Comment
wk font bold	Chaîne	fontBold	<p>Définit la graisse du texte (dépend des styles de police disponibles). Valeurs possibles :</p> <ul style="list-style-type: none">• <u>wk true</u> pour passer en style gras les caractères sélectionnés ; avec la commande WP LIRE ATTRIBUTS, <u>wk true</u> est retournée si au moins un des caractères sélectionnés prend en charge le style de police gras.• <u>wk false</u> (défaut) pour supprimer le style gras des caractères sélectionnés s'il y en a ; avec la commande WP LIRE ATTRIBUTS, <u>wk false</u> est retournée si aucun des caractères sélectionnés ne prend en charge le style de police gras.
wk font italic	Chaîne	fontItalic	<p>Définit le style italique du texte (dépend des styles de police disponibles). Valeurs possibles :</p> <ul style="list-style-type: none">• <u>wk true</u> pour passer en style italique ou oblique les caractères sélectionnés ; avec la commande WP LIRE ATTRIBUTS, <u>wk true</u> est retournée si au moins un des caractères sélectionnés prend en charge le style de police italique ou oblique.• <u>wk false</u> (défaut) pour supprimer le style italique ou oblique des caractères sélectionnés s'il y en a ; avec la commande WP LIRE ATTRIBUTS, <u>wk false</u> est retournée si aucun des caractères sélectionnés ne prend en charge le style de police italique ou oblique.
wk text linethrough style	Chaîne	textLinethroughStyle	<p>Définit le style du texte barré. Valeurs possibles :</p> <ul style="list-style-type: none">• <u>wk none</u> (défaut) : pas d'effet texte barré• <u>wk solid</u> : dessine une ligne pleine sur le texte sélectionné• <u>wk dotted</u> : dessine une ligne en pointillés sur le texte sélectionné• <u>wk dashed</u> : dessine une ligne en tirets sur le texte sélectionné• <u>wk double</u> : dessine une ligne double sur le texte sélectionné• <u>wk semi transparent</u> : atténue la ligne sur le texte sélectionné. Peut être combiné à un autre style de ligne.• <u>wk word</u> : dessine une ligne sur les mots uniquement (exclut les blancs). Peut être combiné à un autre style de ligne.
wk text underline style	Chaîne	textUnderlineStyle	<p>Définit le style du trait de soulignement. Valeurs possibles :</p> <ul style="list-style-type: none">• <u>wk none</u> (défaut) : pas de soulignement• <u>wk solid</u> : dessine un trait plein• <u>wk dotted</u> : dessine un trait en pointillés• <u>wk dashed</u> : dessine un trait en tirets• <u>wk double</u> : dessine un trait double• <u>wk semi transparent</u> : atténue le trait. Peut être combiné à un autre style de ligne.• <u>wk word</u> : dessine le trait sur les mots uniquement (exclut les blancs). Peut être combiné à un autre style de trait.

Typiquement, cette commande permet aux développeurs d'implémenter des objets d'interface personnalisés, comme un pop-up menu proposant des options de style en fonction du texte sélectionné.

WP SUPPRIMER SIGNET

WP SUPPRIMER SIGNET (docWP ; nomSignet)

Paramètre	Type		Description
docWP	Objet	→	Document 4D Write Pro
nomSignet	Chaîne	→	Nom du signet à supprimer

Description

la commande **WP SUPPRIMER SIGNET** supprime de *docWP* le signet nommé *nomSignet*.
Si le signet *nomSignet* n'existe pas dans *docWP*, la commande ne fait rien.

Exemple

Vous souhaitez renommer un signet existant. Pour cela, vous devez créer un nouveau signet basé sur la même plage puis supprimer l'ancien :

```
C_TEXTE($bookmarkOldName)
C_TEXTE($bookmarkNewName)
C_ENTIER LONG($p)
C_OBJET($wpRange)

$bookmarkOldName:="MyBookmark"
$bookmarkNewName:="MyNewBookmark"

TABLEAU TEXTE($_bookmarks;0)
WP LIRE SIGNETS(WParea;$_bookmarks)

$p:=Chercher dans tableau($_bookmarks;$bookmarkOldName)
Si($p>0)
    $wpRange:=WP Lire plage signet(WParea;$bookmarkOldName)
    WP SUPPRIMER SIGNET(WParea;$bookmarkOldName)
    WP CREER SIGNET($wpRange;$bookmarkNewName)
Fin de si
```


WP Tableau ajouter ligne

WP Tableau ajouter ligne (refTableau ; valeur {; valeur2 ; ... ; valeurN}) -> Résultat

Paramètre	Type	Description
refTableau	Objet	→ Référence de tableau
valeur	Texte, Numérique, Heure, Date, Image	→ Valeur(s) à fixer dans la ligne
Résultat	Objet	→ Objet plage ligne

Description

La commande **WP Tableau ajouter ligne** ajoute une ligne au tableau *refTableau*, le remplit avec les *valeur(s)* et retourne l'objet correspondant à la plage de la ligne.

La commande ajoute autant de cellules dans la ligne qu'il y a de valeurs dans le paramètre *valeur*. Vous pouvez ajouter n'importe quel nombre de valeurs de différents types ; l'alignement par défaut dans la cellule dépend du type de la valeur :

- texte : aligné à gauche
- image : centrée
- autres type (nombres, date, et heure) : alignés à droite.

Note : Les valeurs de type Tableau ne sont pas supportées.

La commande retourne la nouvelle ligne en tant qu'objet plage de ligne.

Exemple 1

Vous souhaitez créer un tableau vide et y ajouter plusieurs lignes de taille différente. Vous pouvez écrire :

```
C_OBJET($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3)
$wpRange:=WP Creer plage(WParea;wk start text;wk end text)
$wpTable:=WP Insérer tableau($wpRange;wk append)
$wpRow1:=WP Tableau ajouter ligne($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Tableau ajouter ligne($wpTable;"John";"Richmond";40)
$wpRow3:=WP Tableau ajouter ligne($wpTable;"Mary";"Trenton";18;"New!")
```

Paul	Smith	25	
John	Richmond	40	
Mary	Trenton	18	New!

Exemple 2

Dans une facturation, vous souhaitez créer un tableau automatiquement rempli avec la sélection les lignes facture :

```
C_OBJET($wpTable;$wpRange)

$wpRange:=WP Creer plage(4DWPArea;wk start text;wk end text)

$wpTable:=WP Insérer tableau($wpRange;wk append) //création du tableau

// on ajoute la ligne d'en-tête
$row:=WP Tableau ajouter ligne($wpTable;"Name";"Quantity";"Unit Price";"Discount Rate";"Total")
WP FIXER ATTRIBUTS($row;wk font bold;wk true;wk text align;wk center)

//on utilise simplement un appliquer à sélection
APPLIQUER A SELECTION([INVOICE_LINES];WP Tableau ajouter ligne($wpTable;[INVOICE_LINES]ProductName;[INVOICE_LINES]Quantity;
[INVOICE_LINES]ProductUnitPrice;[INVOICE_LINES]DiscountRate;[INVOICE_LINES]Total))

//on ajoute une ligne pour le pied
$row:=WP Tableau ajouter ligne($wpTable;"Total";"Somme([INVOICE_LINES]Quantity);"";"Somme([INVOICE_LINES]Total))

//on formate le tableau
$range:=WP Tableau lire colonne($wpTable;1;5)
WP FIXER ATTRIBUTS($range;wk width;"80pt")
WP FIXER ATTRIBUTS($wpTable;wk font size;10)
```

Name	Quantity	Unit Price	Discount Rate	Total
Stylo Bille Granger Bright Yellow	7	20	10	126
Stylo Multifonction Voldermont Green	4	5	4	19,2
Stylo Bille Dumbledore Bright Red	4	5	7	18,6
Stylo Bille Rogue Green	3	35	7	97,65
Stylo Plume Granger Yellow	1	30	6	28,2
Stylo Multifonction Voldermont Red	3	15	0	45
Total:	22			334,65

WP Tableau lire cellules

WP Tableau lire cellules (refTableau ; debutCol ; debutLigne { ; nbCols { ; nbLignes} }) -> Résultat

Paramètre	Type	Description
refTableau	Objet	→ Référence de tableau
debutCol	Entier long	→ Position de la colonne de la première cellule
debutLigne	Entier long	→ Position de la ligne de la première cellule
nbCols	Entier long	→ Nombre de colonnes à lire
nbLignes	Entier long	→ Nombre de lignes à lire
Résultat	Objet	→ Nouvelle plage contenant les cellules

Description

La commande **WP Tableau lire cellules** retourne un nouvel objet contenant la sélection des cellules de *refTableau*, cellules définies par *debutCol*, *debutLigne*, *nbCols* et *nbLignes*.

Les paramètres *debutCol* et *debutLigne* déterminent les coordonnées en haut à gauche de la première cellule à lire.

Optionnellement, vous pouvez passer les paramètres *nbCols* et *nbLignes* pour préciser respectivement le nombre de colonnes et le nombre de lignes de la plage de cellules. Si les deux paramètres *nbCols* et *nbLignes* sont omis, la plage retournée contient une seule cellule.

Si *debutCol* plus *nbCols* ou *debutLigne* plus *nbLignes* excède le nombre de colonnes/lignes dans *refTableau*, ou si *debutCol* ou *debutLigne* est plus grand que le nombre de colonnes/lignes dans *refTableau*, la plage retournée contient le maximum de colonnes/lignes.

Exemple 1

Vous voulez modifier les attributs d'une plage de cellules :

```
C_OBJET($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3;$wpRow4;$wpRow5;$cells)
$wpRange:=WP Creer plage(WParea;wk start text;wk end text)
```

```
$wpTable:=WP Insérer tableau($wpRange;wk append)
$wpRow1:=WP Tableau ajouter ligne($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Tableau ajouter ligne($wpTable;"John";"Richmond";40)
$wpRow3:=WP Tableau ajouter ligne($wpTable;"Mary";"Trenton";18)
$wpRow4:=WP Tableau ajouter ligne(($wpTable;"Christopher";"Lee";53)
$wpRow5:=WP Tableau ajouter ligne(($wpTable;"Henry";"Cartier";42)
```

```
$cells:=WP Tableau lire cellules($wpTable;2;2;2;2)
WP FIXER ATTRIBUTS($cells;wk background color;0x00E0E0E0)
WP FIXER ATTRIBUTS($cells;wk text color;"blue")
WP FIXER ATTRIBUTS($cells;wk border color;0x00E000E0)
WP FIXER ATTRIBUTS($cells;wk text align;wk center)
```

Résultat :

Paul	Smith	25
John	Richmond	40
Mary	Trenton	18
Christopher	Lee	53
Henry	Cartier	42

Exemple 2

Obtenir une plage de cellules à partir de la cellule "5x7" et jusqu'à la fin de la ligne :

```
$cellRange:=WP Tableau lire cellules(tableRef;5;7;MAXLONG;1)
```

Obtenir une plage de cellules à partir de la cellule "5x7" et jusqu'à la fin de la colonne :

```
$cellRange:=WP Tableau lire cellules(tableRef;5;7;1;MAXLONG)
```

Obtenir une plage de cellules à partir de la cellule "5x7" et jusqu'à la fin du tableau :

```
$cellRange:=WP Tableau lire cellules(tableRef;5;7;MAXLONG;MAXLONG)
```

WP Tableau lire colonnes

WP Tableau lire colonnes (refTableau ; debutCol {; nbCols}) -> Résultat

Paramètre	Type		Description
refTableau	Objet	→	Référence de tableau
debutCol	Entier long	→	Position de la première colonne
nbCols	Entier long	→	Nombre de colonnes à lire
Résultat	Objet	→	Nouvelle plage contenant les colonnes sélectionnées

Description

La commande **WP Tableau lire colonnes** retourne un nouvel objet contenant la sélection des colonnes de *refTableau*, colonnes définies par *debutCol* et *nbCols*.

Le paramètre *debutCol* pointe sur la première colonne du tableau à retourner, et le paramètre optionnel *nbCols* précise combien de colonnes doivent être retournées. Si *nbCols* est omis, la seule colonne indiquée par *debutCol* est retournée.

Si *debutCol* plus *nbCols* excède le nombre de colonnes dans *refTableau*, ou si *debutCol* est plus grand que le nombre de colonnes dans *refTableau*, la plage retournée contient le maximum de colonnes.

Exemple 1

Vous souhaitez modifier des attributs des deux dernières colonnes du tableau :

```
C_OBJET($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3;$wpRow4;$wpRow5;$cols)
$wpRange:=WP Créer plage(WParea;wk start text;wk end text)
$wpTable:=WP Insérer tableau($wpRange;wk append)
$wpRow1:=WP Tableau ajouter ligne($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Tableau ajouter ligne($wpTable;"John";"Richmond";40)
$wpRow3:=WP Tableau ajouter ligne($wpTable;"Mary";"Trenton";18)
$wpRow4:=WP Tableau ajouter ligne($wpTable;"Christopher";"Lee";53)
$wpRow5:=WP Tableau ajouter ligne($wpTable;"Henry";"Cartier";42)

$cols:=WP Tableau lire colonnes($wpTable;2;2)
WP FIXER ATTRIBUTS($cols;wk background color;0x00E0E0E0;wk text color;"red";wk height;"36pt";wk text align;wk center)
```

Résultat :

Paul	Smith	25
John	Richmond	40
Mary	Trenton	18
Christopher	Lee	53
Henry	Cartier	42

Exemple 2

Vous souhaitez obtenir une plage de colonnes à compter de la 10e jusqu'à la dernière :

```
WP Tableau lire colonnes(tableRef;10;MAXLONG)
```

WP Tableau lire lignes

WP Tableau lire lignes (refTableau ; debutLigne {; nbLignes}) -> Résultat

Paramètre	Type		Description
refTableau	Objet	→	Référence de tableau
debutLigne	Entier long	→	Position de la première ligne
nbLignes	Entier long	→	Nombre de lignes à lire
Résultat	Objet	→	Nouvelle plage contenant les lignes sélectionnées

Description

La commande **WP Tableau lire lignes** retourne un nouvel objet contenant la sélection de lignes définie par *debutLigne* et *nbLignes* dans *refTableau*.

Le paramètre *debutLigne* pointe sur la première ligne du tableau à lire, et le paramètre optionnel *nbLignes* précise le nombre de lignes à lire. Si le paramètre *nbLignes* est omis, seule la ligne *debutLigne* est retournée.

Si *debutLigne* plus *nbLignes* excède le nombre de lignes dans *refTableau*, ou si *debutLigne* est plus grand que le nombre de lignes dans *refTableau*, la plage retournée contient le maximum de lignes.

Exemple 1

Vous souhaitez fixer une couleur d'arrière-plan spécifique pour les deux premières lignes d'un tableau, et modifier la bordure de la troisième ligne :

```
C_OBJET($wpTable;$wpRange;$wpRow1;$wpRow2;$wpRow3;$wpRow4;$wpRow5;$rows;$rows2)
$wpRange:=WP Creer plage(WParea;wk start text;wk end text)
```

```
$wpTable:=WP Insérer tableau($wpRange;wk append)
$wpRow1:=WP Tableau ajouter ligne($wpTable;"Paul";"Smith";25)
$wpRow2:=WP Tableau ajouter ligne($wpTable;"John";"Richmond";40)
$wpRow3:=WP Tableau ajouter ligne($wpTable;"Mary";"Trenton";18)
$wpRow4:=WP Tableau ajouter ligne($wpTable;"Christopher";"Lee";53)
$wpRow5:=WP Tableau ajouter ligne($wpTable;"Henry";"Cartier";42)
```

```
$rows:=WP Tableau lire lignes($wpTable;1;2)
WP FIXER ATTRIBUTS($rows;wk background color;0x00E0F0FF)
$rows2:=WP Tableau lire lignes($wpTable;3)
WP FIXER ATTRIBUTS($rows2;wk border style;wk solid)
WP FIXER ATTRIBUTS($rows2;wk border width;4)
```

Paul	Smith	25
John	Richmond	40
Mary	Trenton	18
Christopher	Lee	53
Henry	Cartier	42

Exemple 2

Vous souhaitez obtenir une plage de lignes à compter de la 10e jusqu'à la dernière :

```
WP Tableau lire lignes(tableRef;10;MAXLONG)
```

⚙️ WP UTILISER PARAMETRES IMPRESSION

WP UTILISER PARAMETRES IMPRESSION (docWP)

Paramètre	Type	Description
docWP	Objet	Document 4D Write Pro

Description

La commande **WP UTILISER PARAMETRES IMPRESSION** modifie les options d'impression de la page courante, sur la base des attributs du document 4D Write Pro pour la taille de page et l'orientation. Cette commande doit être appelée juste avant **WP IMPRIMER** de façon à synchroniser les paramètres courants d'impression de la page avec les options de page du document 4D Write Pro.

D'autres options peuvent être définies avec la commande 4D **PARAMETRES IMPRESSION**. Les options d'impression sont définies pour toute la session 4D.

Exemple

Avant d'imprimer un document, vous souhaitez réinitialiser sa taille de page et son orientation avec les valeurs stockées dans le document :

```
WP UTILISER PARAMETRES IMPRESSION(writeProDoc)
```


Constante	Type	Valeur	Comment
wk 4D Write Pro layout	Entier long	0	Format 4D Write Pro standard, pouvant inclure des attributs de style spécifiques
wk 4wp	Entier long	4	Le document 4D Write Pro est sauvegardé dans un format d'archive natif (HTML <i>zip</i> avec images stockées dans un dossier séparé). Les expressions 4D ne sont pas calculées et les balises 4D spécifiques sont incluses. Ce format est particulièrement approprié pour la sauvegarde et l'archivage de documents 4D Write Pro sur disque sans aucune perte d'informations.
wk anchor all	Entier long	-1	Valeur d'ancre d'image pour wk anchor section et wk anchor page
wk anchor embedded	Entier long	-2	Valeur d'ancre d'image pour wk anchor section
wk anchor first page	Entier long	-2	Valeur d'ancre d'image pour wk anchor page
wk anchor horizontal align	Chaîne	anchorHorizontalAlign	Définit l'alignement horizontal d'une image relativement à l'origine (voir wk anchor origin). Valeurs possibles : <ul style="list-style-type: none"> wk left - alignement à gauche wk center - alignement au centre (<i>non compatible HTML, image non affichée sur le web</i>) wk right - alignement à droite
wk anchor horizontal offset	Chaîne	anchorHorizontalOffset	Définit le décalage horizontal d'une image exprimé en dimension CSS de type chaîne ou entier long (cm ou pt ou pixel) dans l'unité wk layout unit . Valeurs possibles : <ul style="list-style-type: none"> Limite gauche ou droite de l'image relativement à wk anchor horizontal align Limite gauche ou droite du corps (body) en mode inclus (si wk anchor section = wk anchor embedded)
			Valeur par défaut = 0.
			Définit la position relative d'une image par rapport au texte ou à la page. Valeurs possibles :
wk anchor layout	Chaîne	anchorLayout	<ul style="list-style-type: none"> wk behind text - l'image est ancrée derrière le texte wk in front of text - l'image est ancrée devant le texte wk inline with text - l'image est insérée dans le texte (position par défaut pour les images insérées avec WP INSERER IMAGE). Cet attribut est en lecture seule (les images dans le texte ne peuvent pas être converties en images ancrées par programmation).
wk anchor left page	Entier long	-3	Valeur d'ancre d'image pour wk anchor page
			Définit si l'image est ancrée dans la page, l'en-tête ou le pied. Valeurs possibles :
wk anchor origin	Chaîne	anchorOrigin	<ul style="list-style-type: none"> wk paper box (défaut) - l'image est ancrée en bordure de page wk header box - l'image est ancrée dans l'en-tête du document. Si l'en-tête n'est pas visible, l'image n'est pas affichée. wk footer box - l'image est ancrée dans le pied du document. Si le pied n'est pas visible, l'image n'est pas affichée.
			Ce sélecteur est ignoré en mode inclus.
			Définit le numéro de page ou le type de page où l'image est ancrée. Valeurs possibles :
wk anchor page	Chaîne	anchorPage	<ul style="list-style-type: none"> wk anchor all - l'image est ancrée dans toutes les pages de la ou des section(s) définie(s) par wk anchor section wk anchor embedded - l'image est ancrée au corps (body) du document en mode inclus (l'image est visible uniquement en mode inclus) wk anchor first page - l'image est ancrée dans la première page de la ou des section(s) définie(s) par wk anchor section wk anchor left page - l'image est ancrée dans la page gauche de la ou des section(s) définie(s) par wk anchor section wk anchor right page - l'image est ancrée dans la page droite de la ou des section(s) définie(s) par wk anchor section valeur numérique (entier long >= 0) - indique le numéro de la page dans laquelle l'image est ancrée. Dans ce cas, wk anchor section = wk anchor embedded. L'ancrage de section est ignoré si une image est ancrée dans une seule page.
			Note : Les images en mode Page ne sont pas affichées dans les navigateurs.
wk anchor right page	Entier long	-4	Valeur d'ancre d'image pour wk anchor page

Constante	Type	Valeur	Comment
wk anchor section	Chaîne	anchorSection	<p>Définit le numéro de section ou le type de section où l'image est ancrée. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk anchor all</u> (par défaut) - l'image est ancrée dans toutes les sections du document (l'image est visible uniquement en mode page) • <u>wk anchor embedded</u> - l'image est ancrée au corps (body) du document en mode inclus (l'image est visible uniquement en mode inclus). • valeur numérique (entier long >= 1) - indique le numéro de la section dans laquelle l'image est ancrée (l'image est visible uniquement en mode page). <p>Note : Les images en mode Page ne sont pas affichées dans les navigateurs.</p> <p>Définit l'alignement vertical d'une image relativement à l'origine (voir <u>wk anchor origin</u>). Valeurs possibles :</p>
wk anchor vertical align	Chaîne	anchorVerticalAlign	<ul style="list-style-type: none"> • <u>wk top</u> - alignement en haut • <u>wk center</u> - alignement au centre (<i>non compatible HTML, image non affichée sur le web</i>) • <u>wk bottom</u> - alignement en bas <p>Définit le décalage vertical d'une image exprimé en dimension CSS de type chaîne ou nombre (cm ou pt ou pixel) dans l'unité <u>wk layout unit</u>. Valeurs possibles :</p>
wk anchor vertical offset	Chaîne	anchorVerticalOffset	<ul style="list-style-type: none"> • Haut, centre ou bas de la page (voir <u>wk anchor horizontal align</u>) • Haut, centre ou bas du corps (body) en mode inclus (si <u>wk anchor section</u> = <u>wk anchor embedded</u>) <p>Valeur par défaut = 0.</p>
wk append	Entier long	2	Insère le contenu à la fin de la page
wk armenian	Entier long	19	Style de numérotation arménien standard utilisé (valeur pour <u>wk list style type</u>)
wk author	Chaîne	author	Indique le nom de l'auteur du document (chaîne)
wk auto	Entier long	0	La valeur de la propriété (constante) à laquelle cette constante est appliquée est ajustée automatiquement en fonction du contenu ou du contexte de l'élément
wk background clip	Chaîne	backgroundClip	<p>Définit la zone de fond. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk border box</u> (défaut) : le fond est jusqu'au bord extérieur de la bordure • <u>wk content box</u> : le fond est à l'intérieur de la zone "content" • <u>wk padding box</u> : le fond est jusqu'au bord extérieur du padding (ou jusqu'au bord interne de la bordure, le cas échéant) • <u>wk paper box</u> : le fond est jusqu'au bord du document ou de la section <p>Définit la couleur de fond d'un élément. Valeurs possibles :</p>
wk background color	Chaîne	backgroundColor	<ul style="list-style-type: none"> • une couleur CSS ("command_5010101" ou "#command_5FFFFFF" ou "red"). • une valeur de couleur 4D (entier long, voir commande) • un tableau entier long contenant un élément pour chaque composante R, G, B (0-255) <p>La valeur par défaut pour les documents est "#command_5FFFFFF" et <u>wk transparent</u> ou "transparent" pour les paragraphes et les images.</p>

Constante	Type	Valeur	Comment
wk background height	Chaîne	backgroundHeight	<p>Définit la dimension verticale d'une image de fond. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk auto (défaut) : utilise la hauteur courante de l'image de fond • wk contain : redimensionne l'image à sa hauteur maximale de manière qu'elle "tienne" entièrement dans la zone "content" tout en préservant ses proportions. Cette option modifie également la valeur de l'autre attribut de dimension. • wk cover : redimensionne l'image à sa hauteur maximale de manière à ce que le fond soit entièrement recouvert par l'image de fond tout en préservant ses proportions. Certaines parties de l'image peuvent être tronquées. Cette option modifie également la valeur de l'autre attribut de dimension • Taille définie : définit la largeur de l'image via une valeur de type réel ou chaîne : <ul style="list-style-type: none"> ◦ Réel : Taille en wk layout unit. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. Il est possible de passer une valeur relative (pourcentage %). <p>Définit une référence d'image à utiliser en fond. Peut être toute expression ou variable image 4D valide.</p>
wk background image	Chaîne	backgroundImage	<ul style="list-style-type: none"> • Valeur retournée (WP LIRE ATTRIBUTS) : si l'image avait été définie via un URL réseau, l'image cible est retournée si elle a déjà été chargée, sinon une image vide est retournée. <p>Utilisez wk background image url si vous souhaitez gérer des images via des URLs ou des URIs locaux.</p> <p>Image de fond définie via un URL (chaîne). Peut être un URL réseau ou un data URI, absolu ou relatif au fichier de structure.</p>
wk background image url	Chaîne	backgroundImageURL	<ul style="list-style-type: none"> • Valeur retournée (WP LIRE ATTRIBUTS) : URL réseau ou URI de donnée. Cette valeur peut ne pas être égale à l'URL initial pour une image non référencée par un URL réseau (seuls les URLs réseau sont conservés). Pour les URLs de fichiers locaux, le flux d'image (stream) lui-même est conservé dans le document et donc l'URL retourné est un URI de données avec le flux d'image encodé en base64. <p>Utilisez wk background image pour gérer des images de fond via des expressions image.</p>
wk background origin	Chaîne	backgroundOrigin	<p>Définit la position de l'image de fond. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk padding box (défaut) : l'image de fond débute au rectangle de padding (ou à l'intérieur de la bordure) • wk border box : l'image de fond débute au rectangle de la bordure (bord extérieur) • wk content box : l'image de fond au rectangle "content" • wk paper box : l'image de fond débute au bord du document ou de la section
wk background position horizontal	Chaîne	backgroundPositionHorizontal	<p>Définit la position horizontale de départ d'une image de fond. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk left (défaut) : l'image de fond débute horizontalement à gauche de l'élément • wk center : l'image de fond débute horizontalement au centre de l'élément • wk right : l'image de fond débute horizontalement à droite de l'élément
wk background position vertical	Chaîne	backgroundPositionVertical	<p>Définit la position verticale de départ d'une image de fond. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk top (défaut) : l'image de fond débute verticalement en haut de l'élément • wk middle : l'image de fond débute verticalement au centre de l'élément • wk bottom : l'image de fond débute verticalement à droite de l'élément

Constante	Type	Valeur	Comment
wk background repeat	Chaîne	backgroundRepeat	<p>Définit si et comment une image de fond est répétée. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk repeat</u> (défaut) : l'image de fond est répétée verticalement et horizontalement • <u>wk no repeat</u> : l'image de fond n'est pas répétée • <u>wk repeat x</u> : l'image de fond est répétée horizontalement uniquement • <u>wk repeat y</u> : l'image de fond est répétée verticalement uniquement <p>Définit la dimension horizontale d'une image de fond. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk auto</u> (défaut) : utilise la largeur courante de l'image de fond • <u>wk contain</u> : redimensionne l'image à sa largeur maximale de manière qu'elle "tienne" entièrement dans la zone "content" tout en préservant ses proportions. Cette option modifie également la valeur de l'autre attribut de dimension. • <u>wk cover</u> : redimensionne l'image de manière à ce que le fond soit entièrement recouvert par l'image de fond tout en préservant ses proportions. Certaines parties de l'image peuvent être tronquées. Cette option modifie également la valeur de l'autre attribut de dimension • Taille définie : définit la largeur de l'image via une valeur de type réel ou chaîne : <ul style="list-style-type: none"> ◦ Réel : Taille en <u>wk layout unit</u>. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. Il est possible de passer une valeur relative (pourcentage %).
wk background width	Chaîne	backgroundWidth	
wk bar	Entier long	4	Insère une barre verticale à l'emplacement de la tabulation (valeur pour <u>wk tab stop types</u>)
wk baseline	Entier long	4	Aligne la base d'un élément avec la base de l'élément parent (valeur pour <u>wk vertical align</u>)
wk behind text	Entier long	0	Valeur pour <u>wk anchor layout</u>
wk body	Entier long	0	Zone de corps (<i>body</i>) de la section
wk border box	Entier long	0	Le fond est découpé par la zone de bordure (valeur pour <u>wk background clip</u>)
wk border color	Chaîne	borderColor	<p>Définit la couleur pour les quatre bordures. Valeurs possibles :</p> <ul style="list-style-type: none"> • une couleur CSS ("command_5010101" ou "command_5FFFFFF" ou "red"). • une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) • un tableau entier long contenant un élément pour chaque composante R, G, B (0-255) <p>La valeur par défaut est "command_5000000" (si valeur chaîne). S'il y a plusieurs couleurs, WP LIRE ATTRIBUTS retourne une chaîne vide.</p>
wk border color bottom	Chaîne	borderColorBottom	<p>Définit la couleur pour la bordure basse. Valeurs possibles :</p> <ul style="list-style-type: none"> • une couleur CSS ("command_5010101" ou "command_5FFFFFF" ou "red"). La valeur par défaut est "command_5000000" • une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) • un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)
wk border color left	Chaîne	borderColorLeft	<p>Définit la couleur pour la bordure gauche. Valeurs possibles :</p> <ul style="list-style-type: none"> • une couleur CSS ("command_5010101" ou "command_5FFFFFF" ou "red"). La valeur par défaut est "command_5000000" • une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) • un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)
wk border color right	Chaîne	borderColorRight	<p>Définit la couleur pour la bordure droite. Valeurs possibles :</p> <ul style="list-style-type: none"> • une couleur CSS ("command_5010101" ou "command_5FFFFFF" ou "red"). La valeur par défaut est "command_5000000" • une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) • un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

Constante	Type	Valeur	Comment
wk border color top	Chaîne	borderColorTop	<p>Définit la couleur pour la bordure haute. Valeurs possibles :</p> <ul style="list-style-type: none"> • une couleur CSS ("command_5010101" ou "command_5FFFFFF" ou "red"). La valeur par défaut est "command_5000000" • une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) • un tableau entier long contenant un élément pour chaque composante R, G, B (0-255) <p>Définit l'arrondi des angles de la bordure. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk none</u> (défaut) : la bordure n'a pas d'angles arrondis • Valeur d'angle exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : Angle en <u>wk layout unit</u>. ◦ Chaîne : chaîne CSS avec valeur et unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres.
wk border radius	Chaîne	borderRadius	<p>Définit le style des quatre bordures. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk none</u> (défaut) : pas de bordure • <u>wk hidden</u> : équivaut à <u>wk none</u>, sauf en résolution de conflit de bordure • <u>wk solid</u> : bordure pleine • <u>wk dotted</u> : bordure en pointillés • <u>wk dashed</u> : bordure en tirets • <u>wk double</u> : bordure double • <u>wk groove</u> : bordure relief 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk ridge</u> : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure) • <u>wk inset</u> : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk outset</u> : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure) <p>Définit le style de la bordure basse. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk none</u> (défaut) : pas de bordure basse • <u>wk hidden</u> : équivaut à <u>wk none</u>, sauf en résolution de conflit de bordure • <u>wk solid</u> : bordure pleine • <u>wk dotted</u> : bordure en pointillés • <u>wk dashed</u> : bordure en tirets • <u>wk double</u> : bordure double • <u>wk groove</u> : bordure relief 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk ridge</u> : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure) • <u>wk inset</u> : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk outset</u> : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)
wk border style	Chaîne	borderStyle	<p>Définit le style de la bordure gauche. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk none</u> (défaut) : pas de bordure gauche • <u>wk hidden</u> : équivaut à <u>wk none</u>, sauf en résolution de conflit de bordure • <u>wk solid</u> : bordure pleine • <u>wk dotted</u> : bordure en pointillés • <u>wk dashed</u> : bordure en tirets • <u>wk double</u> : bordure double • <u>wk groove</u> : bordure relief 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk ridge</u> : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure) • <u>wk inset</u> : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk outset</u> : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)
wk border style bottom	Chaîne	borderStyleBottom	<p>Définit le style de la bordure droite. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk none</u> (défaut) : pas de bordure droite • <u>wk hidden</u> : équivaut à <u>wk none</u>, sauf en résolution de conflit de bordure • <u>wk solid</u> : bordure pleine • <u>wk dotted</u> : bordure en pointillés • <u>wk dashed</u> : bordure en tirets • <u>wk double</u> : bordure double • <u>wk groove</u> : bordure relief 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk ridge</u> : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure) • <u>wk inset</u> : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk outset</u> : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)
wk border style left	Chaîne	borderStyleLeft	<p>Définit le style de la bordure droite. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk none</u> (défaut) : pas de bordure droite • <u>wk hidden</u> : équivaut à <u>wk none</u>, sauf en résolution de conflit de bordure • <u>wk solid</u> : bordure pleine • <u>wk dotted</u> : bordure en pointillés • <u>wk dashed</u> : bordure en tirets • <u>wk double</u> : bordure double • <u>wk groove</u> : bordure relief 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk ridge</u> : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure) • <u>wk inset</u> : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure) • <u>wk outset</u> : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)

Constante	Type	Valeur	Comment
wk border style right	Chaîne	borderStyleRight	<p>Définit le style de la bordure droite. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk none (défaut) : pas de bordure droite • wk hidden : équivaut à wk none, sauf en résolution de conflit de bordure • wk solid : bordure pleine • wk dotted : bordure en pointillés • wk dashed : bordure en tirets • wk double : bordure double • wk groove : bordure relief 3D (l'effet réel dépend de la couleur de la bordure) • wk ridge : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure) • wk inset : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure) • wk outset : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure) <p>Définit le style de la bordure haute. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk none (défaut) : pas de bordure haute • wk hidden : équivaut à wk none, sauf en résolution de conflit de bordure • wk solid : bordure pleine • wk dotted : bordure en pointillés • wk dashed : bordure en tirets • wk double : bordure double • wk groove : bordure relief 3D (l'effet réel dépend de la couleur de la bordure) • wk ridge : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure) • wk inset : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure) • wk outset : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure) <p>Définit la largeur des quatre bordures. Vous devez spécifier le style de la bordure avant sa largeur. Valeurs possibles :</p> <ul style="list-style-type: none"> • Largeur exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : largeur en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres • Valeur par défaut : 2pt <p>Définit la largeur de la bordure basse. Valeurs possibles :</p> <ul style="list-style-type: none"> • Largeur exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : Largeur en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres • Valeur par défaut : 2pt <p>Définit la largeur de la bordure gauche. Valeurs possibles :</p> <ul style="list-style-type: none"> • Largeur exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : Largeur en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres • Valeur par défaut : 2pt <p>Définit la largeur de la bordure droite. Valeurs possibles :</p> <ul style="list-style-type: none"> • Largeur exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : Largeur en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres • Valeur par défaut : 2pt <p>Définit la largeur de la bordure haute. Valeurs possibles :</p> <ul style="list-style-type: none"> • Largeur exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : Largeur en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres • Valeur par défaut : 2pt
wk border style top	Chaîne	borderStyleTop	
wk border width	Chaîne	borderWidth	
wk border width bottom	Chaîne	borderWidthBottom	
wk border width left	Chaîne	borderWidthLeft	
wk border width right	Chaîne	borderWidthRight	
wk border width top	Chaîne	borderWidthTop	
wk bottom	Entier long	1	Définit la position d'une image de fond (valeur pour wk background position v) ou le bas d'un élément aligné avec l'élément le plus bas de la ligne (valeur pour wk vertical align)

Constante	Type	Valeur	Comment
wk capitalize	Entier long	1	Met en majuscule le premier caractère de chaque mot (valeur pour wk text transform)
wk cell count	Chaîne	cellCount	Nombre total de cellules dans la ligne. Type de valeur : Entier long (valeur pour wk type table row)
wk center	Entier long	2	Centre le texte ou l'image (valeur pour wk background position h , wk text align et/ou wk tab stop types)
wk circle	Entier long	11	Glyphe disque vide utilisé (valeur pour wk list style type)
wk cjk ideographic	Entier long	24	Nombres idéographiques utilisés (valeur pour wk list style type)
wk club	Entier long	27	Glyphe barre utilisé (valeur pour wk list style type)
wk column break	Entier long	3	Saut de colonne : insère un saut de colonne (Disponible pour les tableaux, les documents les sections) Nombre de colonnes. Type de valeur : Entier long
wk column count	Chaîne	columnCount	<ul style="list-style-type: none"> Pour un tableau : attribut en lecture seulement Pour un document ou une section : attribut en lecture/écriture. Valeur par défaut=1 (colonne unique). Valeur maximum=20
wk column rule color	Chaîne	columnRuleColor	Couleur du séparateur (trait) de colonne vertical. Valeurs possibles : <ul style="list-style-type: none"> une couleur CSS ("#command_5010101" ou "#command_5FFFFFFF" ou "red"). Valeur par défaut : "#command_5000000" (black) une valeur entier long de couleur 4D (voir commande OBJET FIXER COULEUR) un tableau entier long contenant un élément pour chaque composante R, V, B (0-255)
wk column rule style	Chaîne	columnRuleStyle	Style du séparateur (trait) de colonne vertical. Valeurs possibles : <ul style="list-style-type: none"> wk none (défaut) : pas de séparateur wk hidden : équivaut à wk none, sauf en résolution de conflit wk solid : ligne pleine wk dotted : ligne en pointillés wk dashed : ligne en tirets wk double : ligne double wk groove : ligne relief 3D (l'effet réel dépend de la couleur de la ligne) wk ridge : ligne relief 3D inversé (l'effet réel dépend de la couleur de la ligne) wk inset : ligne biseau 3D (l'effet réel dépend de la couleur de la ligne)
wk column rule width	Chaîne	columnRuleWidth	Largeur du séparateur de colonne vertical. Valeurs possibles : <ul style="list-style-type: none"> Réel : largeur en wk layout unit. Chaîne : Chaîne CSS avec valeur et unité concaténées (e.g.: "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres). Valeur par défaut="2.5pt"
wk column spacing	Chaîne	columnSpacing	(Documents ou sections uniquement) Espacement entre deux colonnes. Valeurs possibles : <ul style="list-style-type: none"> Réel : largeur en wk layout unit Chaîne : Chaîne CSS avec valeur et unité concaténées (e.g.: "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres). Valeur par défaut="12pt"
wk column width	Chaîne	columnWidth	(Documents ou sections uniquement) Attribut en lecture seulement. Largeur courante de chaque colonne, i.e. largeur calculée basée sur la largeur de la page, les marges de la page, le nombre de colonnes et l'espacement entre colonnes. Pour le document, le calcul utilise la largeur de la section par défaut, et peut donc être différent de la largeur réelle des colonnes de section(s) si certains attributs sont surchargés dans une section. Valeurs possibles : <ul style="list-style-type: none"> Réel : largeur en wk layout unit. Chaîne : Chaîne CSS avec valeur et unité concaténées (e.g.: "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres).
wk company	Chaîne	company	Définit une entreprise associée au document (chaîne)
wk contain	Entier long	-1	Redimensionne l'image à sa plus grande taille de manière à ce que sa largeur et sa hauteur correspondent à l'intérieur de la zone "content" (valeur pour wk background size h et/ou wk background size v)

Constante	Type	Valeur	Comment
wk content box	Entier long	2	Le fond est découpé par la zone "content" (valeur pour wk background clip) ou l'image de fond débute à l'angle supérieur gauche de la zone "content" (valeur pour wk background origin)
wk cover	Entier long	-2	Redimensionne l'image de fond à la taille lui permettant de recouvrir entièrement la zone de fond (valeur pour wk background size h et/ou wk background size v)
wk current page footer	Entier long	2	
wk current page header	Entier long	1	
wk current section default footer	Entier long	10	Pied de page par défaut de la section (disponible uniquement s'il n'y a pas de pied de sous-section active).
wk current section default header	Entier long	9	En-tête par défaut de la section (disponible uniquement s'il n'y a pas d'en-tête de sous-section active).
wk current section first footer	Entier long	4	Pied de la première page de la section
wk current section first header	Entier long	3	En-tête de la première page de la section
wk current section left footer	Entier long	6	Pied de la (des) page(s) gauche(s) de la section
wk current section left header	Entier long	5	En-tête de la (des) page(s) gauche(s) de la section
wk current section right footer	Entier long	8	Pied de la (des) page(s) droite(s) de la section
wk current section right header	Entier long	7	En-tête de la (des) page(s) droite(s) de la section
wk custom	Entier long	29	Marqueur personnalisé utilisé (valeur pour wk list style type)
wk dashed	Entier long	3	Tirets (valeur pour wk text linethrough style et/ou wk text underline style)
wk date creation	Chaîne	dateCreation	Retourne la date de création du document (date). Cette valeur est en lecture seulement et ne peut pas être fixée.
wk date modified	Chaîne	dateModified	Retourne la date de dernière modification du document (date). Cette valeur est en lecture seulement et ne peut pas être fixée.
wk decimal	Entier long	3	Alignement décimal (valeur pour wk tab stop types) ou nombres utilisés (valeur pour wk list style type)
wk decimal greek	Entier long	28	Nombres grecs utilisés (valeur pour wk list style type)
wk decimal leading zero	Entier long	13	Nombres décimaux avec zéros initiaux utilisés (valeur pour wk list style type)
wk default	Entier long	-1	Utiliser la valeur par défaut de la propriété (constante).
wk diamond	Entier long	26	Glyphe "carreau" utilisé (valeur pour wk list style type)
wk direction	Chaîne	direction	Définit le sens du paragraphe. Valeurs possibles : <ul style="list-style-type: none"> • wk left to right (défaut) • wk right to left
wk disc	Entier long	10	Glyphe disque plein utilisé (valeur pour wk list style type)
wk dotted	Entier long	2	Ligne ou bordure en pointillés (valeur pour wk border style , wk text linethrough style et/ou wk text underline style)
wk double	Entier long	4	Ligne ou bordure double (valeur pour wk border style , wk text linethrough style et/ou wk text underline style)
wk dpi	Chaîne	dpi	DPI utilisés pour la conversion interne pixels <->points (entier). Toujours 96 (lecture seulement) Ne pas confondre cet attribut interne avec l'action standard "dpi" qui permet de modifier ou de lire les DPI de la vue courante du document.
wk end	Chaîne	end	(Attribut de plage en lecture seule) Position fin de plage. Valeur : entier long
wk end text	Entier long	0	Définit la fin du document en tant que fin de la plage de texte

Constante	Type	Valeur	Comment
wk exclude from range	Entier long	1	Le contenu inséré n'est pas inclus dans la plage mise à jour
wk expressions as source	Entier long	1	La chaîne d'origine des références des expressions 4D est retournée.
wk expressions as space	Entier long	2	Chaque référence est retournée en tant qu'espace insécable.
wk expressions as value	Entier long	0	Les références d'expressions 4D sont retournées dans leur forme évaluée (option par défaut).
wk false	Entier long	0	
wk first column	Chaîne	firstColumn	(Attribut en lecture seule) Numéro de la première colonne. Valeur : Entier long
wk first page	Entier long	1	
wk first row	Chaîne	firstRow	(Attribut en lecture seule) Numéro de la première ligne du tableau. Valeur : Entier long
wk font	Chaîne	font	Définit le nom de police complet avec les styles, comme retourné par la commande WP LIRE ATTRIBUTS . Si vous passez un nom invalide, la commande ne fait rien. Valeur par défaut : "Times New Roman". Définit la graisse du texte (dépend des styles de police disponibles). Valeurs possibles :
wk font bold	Chaîne	fontBold	<ul style="list-style-type: none"> • <u>wk true</u> pour passer en style gras les caractères sélectionnés ; avec la commande WP LIRE ATTRIBUTS, <u>wk true</u> est retournée si au moins un des caractères sélectionnés prend en charge le style de police gras. • <u>wk false</u> (défaut) pour supprimer le style gras des caractères sélectionnés s'il y en a ; avec la commande WP LIRE ATTRIBUTS, <u>wk false</u> est retournée si aucun des caractères sélectionnés ne prend en charge le style de police gras.
wk font family	Chaîne	fontFamily	Spécifie le nom de famille de police tel que défini par <u>wk font</u> . Valeur par défaut : "Times New Roman". Une chaîne vide est retournée par la commande WP LIRE ATTRIBUTS si les caractères sélectionnés contiennent des propriétés de famille de police différentes. Définit le style italique du texte (dépend des styles de police disponibles). Valeurs possibles :
wk font italic	Chaîne	fontItalic	<ul style="list-style-type: none"> • <u>wk true</u> pour passer en style italique ou oblique les caractères sélectionnés ; avec la commande WP LIRE ATTRIBUTS, <u>wk true</u> est retournée si au moins un des caractères sélectionnés prend en charge le style de police italique ou oblique. • <u>wk false</u> (défaut) pour supprimer le style italique ou oblique des caractères sélectionnés s'il y en a ; avec la commande WP LIRE ATTRIBUTS, <u>wk false</u> est retournée si aucun des caractères sélectionnés ne prend en charge le style de police italique ou oblique.
wk font size	Chaîne	fontSize	Définit la taille de police du texte. Valeurs possibles (en points uniquement) : <ul style="list-style-type: none"> • Réel (défaut = 12) • Chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points.
wk footer box	Entier long	1	Valeur d'ancre d'image pour <u>wk anchor origin</u>
wk freeze expressions	Entier long	64	Figurer les expressions lors de l'insertion
wk georgian	Entier long	20	Numéros géorgiens traditionnels utilisés (valeur pour <u>wk list style type</u>)
wk groove	Entier long	6	Bordure relief 3D utilisée (valeur pour <u>wk border style</u>)
wk header box	Entier long	0	Valeur d'ancre d'image pour <u>wk anchor origin</u>
wk hebrew	Entier long	21	Nombres hébreux traditionnels utilisés (valeur pour <u>wk list style type</u>)

Constante	Type	Valeur	Comment
wk height	Chaîne	height	<p>Définit la hauteur de l'élément. La propriété <i>height</i> n'inclut pas le padding, les bordures ou les marges ; elle définit la hauteur de la zone située à l'intérieur du padding, de la bordure et de la marge de l'élément. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk auto (défaut) : la hauteur est basée sur le contenu de l'élément • Taille définie : taille exprimée par une valeur de type réel ou chaîne : <ul style="list-style-type: none"> ◦ Réel : Taille en wk layout unit. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. <p>L'attribut wk height est écrasé par l'attribut wk min height (si défini).</p>
wk hidden	Entier long	5	Pas de bordure utilisée. Equivaut à aucune (pas de bordure) excepté qu'il prend la précedence sur toutes les autres bordures en conflit (valeur pour wk border style)
wk hiragana	Entier long	22	Numéros hiragana traditionnels utilisés (valeur pour wk list style type)
wk hollow square	Entier long	25	Glyphe carré vide utilisé (valeur pour wk list style type)
wk html debug	Entier long	1	Code HTML formaté ("pretty print"), facilitant le débogage
wk html wysiwyg	Entier long	1	Dans ce format, les éventuels attributs avancés de 4D Write Pro non disponibles dans tous les navigateurs sont supprimés (multi-colonnes, double interlignes...)
wk id	Chaîne	id	ID interne de l'élément. Valeur : Chaîne Définit une référence d'image. Peut être toute expression ou variable image 4D valide.
wk image	Chaîne	image	<ul style="list-style-type: none"> • Valeur retournée (WP LIRE ATTRIBUTS) : si l'image avait été définie via un URL réseau, l'image cible est retournée si elle a déjà été chargée, sinon une image vide est retournée. <p>Utilisez wk image url si vous souhaitez gérer des images via des URLs ou des URIs locaux.</p>
wk image alternate text	Chaîne	imageAlternateText	Définit un texte alternatif à l'image, si elle ne peut pas être affichée.
wk image url	Chaîne	imageURL	<p>Image définie via un URL (chaîne). Peut être un URL réseau ou un data URI, absolu ou relatif au fichier de structure.</p> <ul style="list-style-type: none"> • Valeur retournée (WP LIRE ATTRIBUTS) : URL réseau ou URI de donnée). Cette valeur peut ne pas être égale à l'URL initial pour une image non référencée par un URL réseau (seuls les URLs réseau sont conservés). Pour les URLs de fichiers locaux, le flux d'image (stream) lui-même est conservé dans le document et donc l'URL retourné est un URI de données avec le flux d'image encodé en base64. <p>Utilisez wk image pour gérer des images via des expressions image.</p>
wk in front of text	Entier long	1	Valeur pour wk anchor layout
wk include in range	Entier long	0	Le contenu inséré est inclus dans la plage mise à jour (défaut)
wk inherit style from paragraph	Entier long	32	Le contenu inséré hérite des styles de caractères par défaut du paragraphe.
wk inline with text	Entier long	-1	Valeur pour wk anchor layout
wk inset	Entier long	8	Bordure biseau 3D utilisée (valeur pour wk border style)
wk inside	Chaîne	Inside	Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété inter-paragraphes correspondante (et pas à l'extérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande WP FIXER ATTRIBUTS .
wk justify	Entier long	5	Alignement justifié. Disponible uniquement pour les zones 4D Write Pro.
wk katakana	Entier long	23	Numéros katakana traditionnels utilisés (valeur pour wk list style type)
wk keep paragraph styles	Entier long	128	Conserver les styles du paragraphe de destination

Constante	Type	Valeur	Comment
			<p>Définit l'unité de dimension par défaut du document lorsqu'une valeur numérique est écrite ou lue. Désigne l'unité de la règle et des attributs de dimension tels que <u>wk width</u>, à l'exception de <u>wk font size</u>, <u>wk border width</u> (et ses variantes), <u>wk border radius</u> et <u>wk text shadow offset</u> pour lesquels l'unité de valeur numérique est toujours le point.</p> <p>Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk unit cm</u> (défaut) : centimètres • <u>wk unit pt</u> : points • <u>wk unit px</u> : pixels • <u>wk unit percent</u> (seulement pour <u>wk line height</u> et <u>wk background size h / wk background size v</u>) • <u>wk unit mm</u> : millimètres • <u>wk unit inch</u> : pouces <p>Note : Lorsqu'une unité non prise en charge par la règle est sélectionnée via cet attribut (i.e. <u>wk unit px</u> ou <u>wk unit percent</u>), la règle utilise alors l'unité cm.</p>
wk layout unit	Chaîne	userUnit	
wk left	Entier long	0	Aligne le texte ou la tabulation à gauche (valeur pour <u>wk text align</u> ou <u>wk tab stop types</u>) ou définit la position de départ d'une image de fond (valeur pour <u>wk background position h</u>)
wk left page	Entier long	2	
wk left to right	Entier long	0	Sens du texte/de l'écriture de gauche à droite (valeur pour <u>wk direction</u>)
wk line break	Entier long	0	Saut de ligne (dans le même paragraphe)
			Définit l'espacement entre les lignes. Valeurs possibles :
wk line height	Chaîne	lineHeight	<ul style="list-style-type: none"> • <u>wk normal</u> (défaut) : utiliser une valeur basée sur la taille du texte. • Hauteur exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : hauteur en <u>wk layout unit</u>. ◦ Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Il est possible de passer une valeur relative (pourcentage %). <p>Lien hypertexte associé à la plage. Valeurs possibles :</p>
wk link url	Chaîne	linkUrl	<ul style="list-style-type: none"> • url absolu, par exemple "http://www.4d.com/" • lien relatif, par exemple "/test/page.html" (le lien est relatif au fichier de structure de la base) • chaîne vide = pas de lien
wk list auto	Entier long	2147483647	Restaure/applique les valeurs de style de liste automatiques
wk list font	Chaîne	listFont	Spécifie un nom complet de police, comme retourné par la commande LISTE STYLES POLICE , pour afficher le marqueur de l'élément de liste (et non le texte du paragraphe). Si le système ne reconnaît pas le nom de police, il gère la substitution. Si vous passez un nom de police invalide, la commande ne fait rien. Valeur par défaut : "Times".
wk list font family	Chaîne	listFontFamily	Spécifie un nom de famille de police tel que défini par <u>wk list font</u> , utilisé pour afficher le marqueur de l'élément de liste (et non le texte du paragraphe). Valeur par défaut : "Times".
wk list start number	Chaîne	listStartNumber	Définit la valeur de départ d'une liste ordonnée. Valeurs possibles : <ul style="list-style-type: none"> • <u>wk auto</u> (défaut) : la valeur de départ dépend des éléments de liste précédents s'il y en a. • valeur entière : valeur de départ
			Marqueur d'élément de liste pour les paragraphes gauche à droite. Si défini, remplace le marqueur d'élément de la liste.
wk list string format LTR	Chaîne	listStringFormatLtr	<ul style="list-style-type: none"> • Pour les listes non ordonnées : chaîne utilisée en tant que marqueur d'élément de liste (généralement un seul caractère, par exemple "-") • Pour les listes ordonnées : chaîne contenant le caractère "#command_5". "#command_5" est un caractère d'emplacement pour le nombre ou la lettre calculé(e). Par défaut "#command_5.", donc par exemple si le numéro d'élément courant de la liste est 15 et que le type de style de liste est décimal, la chaîne du marqueur d'élément de liste sera "15."

Constante	Type	Valeur	Comment
wk list string format RTL	Chaîne	listStringFormatRtl	<p>Marqueur d'élément de liste pour les paragraphes droite à gauche. Si défini, remplace le marqueur d'élément de la liste.</p> <ul style="list-style-type: none"> Pour les listes non ordonnées : chaîne utilisée en tant que marqueur d'élément de liste (généralement un seul caractère, par exemple "-") Pour les listes ordonnées : chaîne contenant le caractère "#command_5". "#command_5" est un caractère d'emplacement pour le nombre ou la lettre calculé(e). Par défaut "#command_5.", donc par exemple si le numéro d'élément courant de la liste est 15 et que le type de style de liste est décimal, la chaîne du marqueur d'élément de liste sera "15." <p>Définit une référence d'image comme marqueur d'élément de liste dans une liste non ordonnée. Valeurs possibles :</p> <ul style="list-style-type: none"> <u>wk none</u> (défaut) : le marqueur d'élément de liste n'est pas une image toute expression ou variable image 4D valide. Valeur retournée (WP LIRE ATTRIBUTS) : si l'image avait été définie via un URL réseau, l'image cible est retournée si elle a déjà été chargée, sinon une image vide est retournée. <p>Utilisez <u>wk list style image url</u> si vous souhaitez gérer des images via des URLs ou des URIs locaux.</p> <p>Définit la hauteur d'une image utilisée comme marqueur d'élément de liste. Valeurs possibles :</p> <ul style="list-style-type: none"> <u>wk auto</u> (défaut) : utilise la hauteur courante de l'image Taille définie : valeur de type réel ou chaîne <ul style="list-style-type: none"> Réel : Taille en <u>wk layout unit</u>. Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. <p>Image à utiliser comme marqueur d'élément de liste dans une liste non ordonnée. Valeurs possibles :</p> <ul style="list-style-type: none"> <u>wk none</u> (défaut) : le marqueur d'élément de liste n'est pas une image un URL réseau ou un data URI, absolu ou relatif au fichier de structure
wk list style image	Chaîne	listStyleImage	
wk list style image height	Chaîne	listStyleImageHeight	
wk list style image url	Chaîne	imageURL	<ul style="list-style-type: none"> Valeur retournée (WP LIRE ATTRIBUTS) : URL réseau ou URI de donnée). Cette valeur peut ne pas être égale à l'URL initial pour une image non référencée par un URL réseau (seuls les URLs réseau sont conservés). Pour les URLs de fichiers locaux, le flux d'image (stream) lui-même est conservé dans le document et donc l'URL retourné est un URI de données avec le flux d'image encodé en base64. <p>Utilisez <u>wk list style image</u> pour gérer des images de marqueurs d'éléments via des expressions.</p>

Constante	Type	Valeur	Comment
			<p>Spécifie le type de marqueur d'élément d'une liste ordonnée ou non ordonnée. Valeurs possibles :</p> <ul style="list-style-type: none"> • wk disc (défaut) • wk circle • wk square • wk decimal : 1 2 3 • wk decimal leading zero : 01 02 03 • wk lower latin : a b c • wk lower roman : i ii iii iv • wk upper latin : A B C • wk upper roman : I II III IV • wk lower greek : alpha, beta, gamma, etc. • wk armenian • wk georgian • wk hebrew • wk hiragana • wk katakana • wk cjk ideographic • wk hollow square • wk diamond • wk club • wk decimal greek • wk custom : par défaut, liste non ordonnée avec "-" comme marqueur d'élément de liste ; il s'agit d'un style pratique permettant de personnaliser un marqueur d'élément de liste avec wk list string format LTR ou wk list string format RTL sans devoir modifier les marqueurs de liste standard • wk none
wk list style type	Chaîne	listStyleType	
wk lower greek	Entier long	18	Minuscules Grec classique utilisées (valeur pour wk list style type)
wk lower latin	Entier long	14	Lettres ASCII minuscules utilisées (valeur pour wk list style type)
wk lower roman	Entier long	15	Minuscules numériques Roman utilisées (valeur pour wk list style type)
wk lowercase	Entier long	2	Passer tous les caractères en minuscules (valeur pour wk text transform)
wk margin	Chaîne	margin	<p>Définit la taille de toutes les marges de l'élément. Valeurs possibles :</p> <ul style="list-style-type: none"> • Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : taille en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres • wk none (défaut) : pas de marge spécifique <p>Définit la taille de la marge basse de l'élément. Valeurs possibles :</p>
wk margin bottom	Chaîne	marginBottom	<ul style="list-style-type: none"> • Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : taille en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres • wk none (défaut) : pas de marge spécifique <p>Définit la taille de la marge gauche de l'élément. Valeurs possibles :</p>
wk margin left	Chaîne	marginLeft	<ul style="list-style-type: none"> • Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : taille en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres • wk none (défaut) : pas de marge spécifique <p>Définit la taille de la marge droite de l'élément. Valeurs possibles :</p>
wk margin right	Chaîne	marginRight	<ul style="list-style-type: none"> • Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : taille en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres • wk none (défaut) : pas de marge spécifique <p>Définit la taille de la marge haute de l'élément. Valeurs possibles :</p>
wk margin top	Chaîne	marginTop	<ul style="list-style-type: none"> • Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : taille en wk layout unit. ◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres • wk none (défaut) : pas de marge spécifique
wk middle	Entier long	2	Définit la position de départ d'une image de fond (valeur pour wk background position v) ou place l'élément au milieu de l'élément parent (valeur pour wk vertical align)

Constante	Type	Valeur	Comment
wk mime html	Entier long	1	<p>Le document 4D Write Pro est sauvegardé au format MIME HTML avec les documents html et les images embarqués en tant que parties MIME (encodées en base64). Les expressions sont calculées et les balises 4D spécifiques sont supprimées. Ce format est particulièrement adapté à l'envoi de mails au format HTML à l'aide de la commande .</p> <p>Définit la hauteur minimum de l'élément. La valeur de la propriété <u>wk height</u> ne peut pas devenir plus petite que <u>wk_min_height</u>. Valeurs possibles :</p> <ul style="list-style-type: none"> • <u>wk_auto</u> (défaut) : la hauteur minimum est basée sur le contenu de l'élément • Taille définie : taille exprimée par une valeur de type réel ou chaîne : <ul style="list-style-type: none"> ◦ Réel : Taille en <u>wk layout unit</u>. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. <p>L'attribut <u>wk_min_height</u> écrase la valeur de l'attribut <u>wk height</u>.</p> <p>Note : Cet attribut n'est pas pris en charge par les colonnes, les lignes et les cellules.</p> <p>Définit la largeur minimum de l'élément. La valeur de la propriété <u>wk width</u> ne peut pas devenir plus petite que <u>wk_min_width</u>. Valeurs possibles :</p>
wk min height	Chaîne	minHeight	<ul style="list-style-type: none"> • <u>wk_auto</u> (défaut) : la largeur minimum est basée sur le contenu de l'élément • Taille définie : taille exprimée par une valeur de type réel ou chaîne : <ul style="list-style-type: none"> ◦ Réel : Taille en <u>wk layout unit</u>. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. <p>L'attribut <u>wk_min_width</u> écrase la valeur de l'attribut <u>wk width</u>.</p>
wk min width	Chaîne	minWidth	<ul style="list-style-type: none"> • <u>wk_auto</u> (défaut) : la largeur minimum est basée sur le contenu de l'élément • Taille définie : taille exprimée par une valeur de type réel ou chaîne : <ul style="list-style-type: none"> ◦ Réel : Taille en <u>wk layout unit</u>. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. <p>L'attribut <u>wk_min_width</u> écrase la valeur de l'attribut <u>wk width</u>.</p>
wk mixed	Entier long	-2147483648	<p>Retourné lorsqu'il y a différentes valeurs pour un attribut dans la page ou le document</p> <p>Définit la feuille de style à utiliser en cas d'ajout d'une nouvelle ligne dans le paragraphe. Valeurs possibles :</p>
wk new line style sheet	Chaîne	newLineStyleSheet	<ul style="list-style-type: none"> • Nom de feuille de style existante • <u>wk_none</u> (défaut)
wk no repeat	Entier long	3	Image de fond non répétée (valeur pour <u>wk_background_repeat</u>)
wk none	Entier long	0	
wk normal	Entier long	0	Code HTML standard
wk notes	Chaîne	notes	Définit des commentaires dans le document (chaîne).
wk outset	Entier long	9	Bordure biseau 3D inversé utilisée (valeur pour <u>wk_border_style</u>)
wk outside	Chaîne	Outside	<p>Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété de paragraphe externe correspondante (et pas à l'intérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande WP FIXER ATTRIBUTS.</p> <p>(Attribut en lecture seule) Possesseur de la page. Valeur : objet</p> <p>Définit la taille de la marge intérieure (<i>padding</i>) pour tous les côtés de l'élément. Valeurs possibles :</p>
wk owner	Chaîne	owner	
wk padding	Chaîne	padding	<ul style="list-style-type: none"> • Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : Taille en <u>wk layout unit</u>. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. • <u>wk_none</u> (défaut) : pas de marge intérieure spécifique <p>Définit la taille de la marge intérieure (<i>padding</i>) pour le bas de l'élément. Valeurs possibles :</p>
wk padding bottom	Chaîne	paddingBottom	<ul style="list-style-type: none"> • Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> ◦ Numérique : Taille en <u>wk layout unit</u>. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. • <u>wk_none</u> (défaut) : pas de marge intérieure spécifique
wk padding box	Entier long	1	Le fond est découpé par la zone de marge intérieure (valeur pour <u>wk_background_clip</u>) ou l'image de fond démarre à l'angle supérieur gauche du bord de la marge intérieure (valeur pour <u>wk_background_origin</u>)

Constante	Type	Valeur	Comment
			Définit la taille de la marge intérieure (<i>padding</i>) pour la gauche de l'élément. Valeurs possibles :
wk padding left	Chaîne	paddingLeft	<ul style="list-style-type: none"> Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> Numérique : Taille en wk layout unit. Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. wk none (défaut) : pas de marge intérieure spécifique
			Définit la taille de la marge intérieure (<i>padding</i>) pour la droite de l'élément. Valeurs possibles :
wk padding right	Chaîne	paddingRight	<ul style="list-style-type: none"> Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> Numérique : Taille en wk layout unit. Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. wk none (défaut) : pas de marge intérieure spécifique
			Définit la taille de la marge intérieure (<i>padding</i>) pour le haut de l'élément. Valeurs possibles :
wk padding top	Chaîne	paddingTop	<ul style="list-style-type: none"> Taille exprimée par une valeur de type numérique ou chaîne : <ul style="list-style-type: none"> Numérique : Taille en wk layout unit. Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. wk none (défaut) : pas de marge intérieure spécifique
wk page break	Entier long	2	Saut de page : définit une nouvelle page
			Page height (in page mode) expressed using a real or string value (Read-only with section or sub-section). Possible values:
wk page height	Chaîne	pageHeight	<ul style="list-style-type: none"> Real: Height in wk layout unit. String: CSS string with value and unit concatenated (e.g.: "12pt" for 12 points, or "1.5cm" for 1.5 centimeters). Supported units: pt,cm,mm, inches.
wk page web complète	Entier long	2	Extension .htm ou .html. Le document est sauvegardé au format HTML standard et ses ressources sont sauvegardées séparément. Les balises 4D spécifiques sont supprimées et les expressions sont calculées. Ce format est particulièrement adapté à l'affichage d'un document 4D Write Pro dans un navigateur Web.
wk page web html 4D	Entier long	3	Le document 4D Write Pro est sauvegardé au format HTML et inclut les balises 4D spécifiques ; chaque expression est insérée sous forme d'espace insécable. Comme ce format est sans perte, il est approprié pour le stockage dans un champ texte.
wk paper box	Entier long	3	Le fond est étendu jusqu'au bord extérieur (valeur pour wk background clip) ou l'image de fond démarre au bord extérieur (valeur pour wk background origin). Documents et sections uniquement.
wk prepend	Entier long	1	Le contenu est inséré au début de la page
wk repeat	Entier long	0	L'image de fond est répétée verticalement et horizontalement (valeur pour wk background repeat)
wk repeat x	Entier long	1	L'image de fond est répétée horizontalement uniquement (valeur pour wk background repeat)
wk repeat y	Entier long	2	L'image de fond est répétée verticalement uniquement (valeur pour wk background repeat)
wk replace	Entier long	0	Remplace le contenu de la page
wk ridge	Entier long	7	Bordure 3D relief inversé utilisée (valeur pour wk border style)
wk right	Entier long	1	Aligne le texte ou la tabulation à droite (valeur pour wk text align ou wk tab stop types) ou définit la position de départ d'une image de fond (valeur pour wk background position h)
wk right page	Entier long	3	
wk right to left	Entier long	1	Sens du texte/de l'écriture de droite à gauche (valeur pour wk direction)
wk row count	Chaîne	rowCount	(Attribut en lecture seule) Nombre total de lignes. Valeur : Entier long
wk section break	Entier long	1	Saut de section: définit une nouvelle section
wk semi transparent	Entier long	5	Ligne semi-transparente utilisée (valeur pour wk text linethrough style et/ou wk text underline style)
wk small uppercase	Entier long	4	Passer tous les caractères en petites majuscules (valeur pour wk text transform)
wk solid	Entier long	1	Ligne ou bordure pleine utilisée (valeur pour wk border style , wk text linethrough style et/ou wk text underline style)

Constante	Type	Valeur	Comment
wk square	Entier long	12	Marqueur carré utilisé (valeur pour wk_list_style_type)
wk start	Chaîne	start	(Attribut de plage en lecture seule) Position début de plage. Valeur : entier long
wk start text	Entier long	1	Définit le début du document en tant que début de la plage de texte
wk style sheet	Chaîne	styleSheet	Définit la feuille de style courante pour le ou les élément(s) sélectionné(s). Valeurs possibles : <ul style="list-style-type: none"> Nom de feuille de style existante wk_none (défaut)
wk subject	Chaîne	subject	Définit le sujet du document (chaîne)
wk subscript	Entier long	6	Alignement en indice (valeur pour wk_vertical_align)
wk superscript	Entier long	5	Alignement en exposant (valeur pour wk_vertical_align)
wk tab stop leadings	Chaîne	tabStopLeadings	Définit les caractères de conduite pour les tabulations dans les plages de paragraphes. Valeurs possibles : <ul style="list-style-type: none"> Tableau texte Liste ordonnée de caractères de conduite pour chaque position de tabulation, à partir de la marge gauche. Le caractère défini par la dernière valeur est répété pour chaque tabulation supplémentaire saisie dans le paragraphe. Texte Un seul caractère est utilisé comme caractère de conduite. Si un texte contenant plusieurs caractères est passé, le premier caractère est utilisé. Propriété synchronisée avec les tableaux wk_tab_stop_offsets et wk_tab_stop_types . Si les tableaux n'ont pas la même taille (et ne sont pas vides), le dernier caractère du tableau est utilisé comme caractère par défaut pour les tabulations. Si les tableaux sont vides, le caractère de conduite None est utilisé. <p>Note : Le dernier caractère utilisé est toujours le caractère par défaut pour les tabulations.</p> Définit les tabulations pour le paragraphe. Valeurs possibles : <ul style="list-style-type: none"> Tableau de valeurs de tabulations : une liste ordonnée de valeurs absolues, à partir de la marge gauche. La dernière position de tabulation définie est répétée pour chaque nouvelle tabulation saisie dans le paragraphe. Si la position de tabulation est supérieure à la largeur du paragraphe, le texte passe à la ligne suivante et débute à la première tabulation. Si une valeur du tableau est plus petite que la précédente, elle est ignorée. Valeur fixe (défaut 35.45pt) : position par défaut pour tout le paragraphe. La commande WP LIRE ATTRIBUTS retourne la dernière position (qui est la position relative par défaut pour les positions définies au-delà de la dernière position absolue). <p>Note : Vous ne pouvez pas utiliser de tableau et de valeurs fixes dans le même appel pour différents attributs.</p> Les valeurs sont exprimées à l'aide de chaînes CSS (défaut) ou de réels dans wk_layout_unit . Valeur maximum : 10000pt. Si des tableaux sont utilisés, cette propriété est synchronisée avec les tableaux wk_tab_stop_leadings et wk_tab_stop_types .
wk tab stop offsets	Chaîne	tabStopOffsets	Définit le type de tabulation pour le paragraphe. Valeurs possibles : <ul style="list-style-type: none"> tableau de valeurs de types de tabulations (si les tabulations ont été définies via un tableau) wk_left (défaut) : le texte est décalé vers la droite à partir de la tabulation wk_right : le texte est décalé vers la gauche à partir de la tabulation jusqu'à ce que l'espace de la tabulation soit rempli. wk_center : le texte est centré sur la tabulation wk_decimal : le texte avant le point décimal est décalé vers la gauche, le texte après le point décimal est décalé vers la droite wk_bar : affiche une barre verticale à l'emplacement spécifié. Si des tableaux sont utilisés, cette propriété est synchronisée avec les tableaux wk_tab_stop_leadings et wk_tab_stop_offsets .
wk tab stop types	Chaîne	tabStopTypes	

Constante	Type	Valeur	Comment
wk table align	Chaîne	tableAlign	Définit l'alignement horizontal pour un élément de tableau. Valeurs possibles : <ul style="list-style-type: none"> wk left (défaut) wk right wk center
wk table ID	Chaîne	tableID	(Attribut en lecture seule) ID du tableau parent. Valeur : Chaîne Définit l'alignement horizontal du texte dans le paragraphe. Valeurs possibles :
wk text align	Chaîne	textAlign	<ul style="list-style-type: none"> wk left (défaut) wk right wk justify wk center
wk text color	Chaîne	color	Définit la couleur du texte. Valeurs possibles : <ul style="list-style-type: none"> une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000" une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)
wk text indent	Chaîne	textIndent	Définit l'indentation de la première ligne du paragraphe. Valeurs possibles : <ul style="list-style-type: none"> Réel : Taille en wk layout unit. Défaut : 0 Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt.
wk text linethrough color	Chaîne	textLinethroughColor	Définit la couleur de la ligne du texte barré. Valeurs possibles : <ul style="list-style-type: none"> une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000" une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)
wk text linethrough style	Chaîne	textLinethroughStyle	Par défaut, "currentColor" si chaîne, ou wk default si entier long. Définit le style du texte barré. Valeurs possibles : <ul style="list-style-type: none"> wk none (défaut) : pas d'effet texte barré wk solid : dessine une ligne pleine sur le texte sélectionné wk dotted : dessine une ligne en pointillés sur le texte sélectionné wk dashed : dessine une ligne en tirets sur le texte sélectionné wk double : dessine une ligne double sur le texte sélectionné wk semi transparent : atténue la ligne sur le texte sélectionné. Peut être combiné à un autre style de ligne. wk word : dessine une ligne sur les mots uniquement (exclut les blancs). Peut être combiné à un autre style de ligne.
wk text shadow color	Chaîne	textShadowColor	Définit la couleur d'ombre du texte sélectionné. Valeurs possibles : <ul style="list-style-type: none"> une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) un tableau entier long contenant un élément pour chaque composante R, G, B (0-255) wk transparent (défaut)
wk text shadow offset	Chaîne	textShadowOffset	Définit le décalage pour l'effet ombré. Valeurs possibles : <ul style="list-style-type: none"> Taille exprimée en points. Valeur par défaut : 1pt
wk text transform	Chaîne	textTransform	Définit la casse des lettres dans le texte. Valeurs possibles : <ul style="list-style-type: none"> wk capitalize : les lettres initiales sont passées en capitales wk lowercase : les lettres sont passées en minuscules wk uppercase : les lettres sont passées en majuscules wk small uppercase : les lettres sont passées en petites majuscules wk none (défaut) : pas de transformation

Constante	Type	Valeur	Comment
wk text underline color	Chaîne	textUnderlineColor	<p>Définit la couleur du trait de soulignement. Valeurs possibles :</p> <ul style="list-style-type: none"> une couleur CSS ("#command_5010101" ou "#command_5FFFFFF" ou "red"). La valeur par défaut est "#command_5000000" une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR) un tableau entier long contenant un élément pour chaque composante R, G, B (0-255) <p>Par défaut, "currentColor" si chaîne, ou <u>wk default</u> si entier long.</p>
wk text underline style	Chaîne	textUnderlineStyle	<p>Définit le style du trait de soulignement. Valeurs possibles :</p> <ul style="list-style-type: none"> <u>wk none</u> (défaut) : pas de soulignement <u>wk solid</u> : dessine un trait plein <u>wk dotted</u> : dessine un trait en pointillés <u>wk dashed</u> : dessine un trait en tirets <u>wk double</u> : dessine un trait double <u>wk semi transparent</u> : atténue le trait. Peut être combiné à un autre style de ligne. <u>wk word</u> : dessine le trait sur les mots uniquement (exclut les blancs). Peut être combiné à un autre style de trait.
wk title	Chaîne	title	Définit le titre du document (chaîne). Par défaut "New 4D Write Pro Document"
wk top	Entier long	0	Définit la position d'une image de fond (valeur pour <u>wk background position y</u>) ou aligne le haut de l'élément avec le haut de l'élément le plus haut de la ligne (valeur pour <u>wk vertical align</u>)
wk transparent	Entier long	-1	Définit une couleur transparente (valeur pour <u>wk background color</u> ou <u>wk text shadow color</u>)
wk true	Entier long	1	
wk type	Chaîne	type	<p>(Attribut de plage en lecture seule) Type d'objet 4D Write Pro. Valeurs possibles :</p> <ul style="list-style-type: none"> <u>wk type default</u> : Plage sans type défini <u>wk type paragraph</u> : Plage de type paragraphe <u>wk type image</u> : Image (ancrée ou en ligne) <u>wk type image anchored</u> : Image ancrée <u>wk type image inline</u> : Image en ligne <u>wk type container</u> : En-tête ou pied, par exemple <u>wk type table</u> : Référence de tableau <u>wk type table row</u> : Référence de ligne de tableau <u>wk type table cell</u> : Référence de cellule de tableau <u>wk type table column</u> : Référence de colonne de tableau (uniquement pour plage colonne) <u>wk type body</u> : Référence de corps (body)
wk type body	Entier long	8	Référence de corps (body) (valeur pour <u>wk type</u>)
wk type container	Entier long	3	En-tête ou pied, par exemple (valeur pour <u>wk type</u>)
wk type default	Entier long	0	Type de plage par défaut
wk type image	Entier long	2	Images ancrées et/ou en ligne
wk type image anchored	Entier long	200	Référence d'image ancrée pour objets documents
wk type image inline	Entier long	100	Référence d'image en ligne pour objets documents
wk type paragraph	Entier long	1	Plage de type paragraphe
wk type table	Entier long	4	Référence de tableau
wk type table cell	Entier long	6	Référence de cellule de tableau (valeur pour <u>wk type</u>)
wk type table column	Entier long	7	Référence de colonne - uniquement pour plage de colonne (valeur pour <u>wk type</u>)
wk type table row	Entier long	5	Référence de ligne de tableau
wk unit cm	Chaîne	cm	Unité définie en centimètres (valeur pour <u>wk layout unit</u>)
wk unit inch	Chaîne	in	Unité définie en pouces (valeur pour <u>wk layout unit</u>)
wk unit mm	Chaîne	mm	Unité définie en millimètres (valeur pour <u>wk layout unit</u>)

Constante	Type	Valeur	Comment
wk unit percent	Chaîne	%	Unité définie en pourcentages (valeur pour <u>wk layout unit</u>)
wk unit pt	Chaîne	pt	Unité définie en points (valeur pour <u>wk layout unit</u>)
wk unit px	Chaîne	px	Unité définie en pixels (valeur pour <u>wk layout unit</u>)
wk upper latin	Entier long	16	Lettres ASCII majuscules utilisées (valeur pour <u>wk list style type</u>)
wk upper roman	Entier long	17	Lettres Roman numériques majuscules utilisées (valeur pour <u>wk list style type</u>)
wk uppercase	Entier long	3	Passer tous les caractères en majuscules (valeur pour <u>wk text transform</u>)
wk value unit not percentage	Entier long	-100000	Retourné dans la commande WP LIRE ATTRIBUTS lorsque la valeur d'unité courante n'est pas les pourcentages et que vous avez passé une variable chaîne (résultat invalide).
wk value unit percentage	Entier long	-100001	Retourné dans la commande WP LIRE ATTRIBUTS lorsque la valeur d'unité courante est les pourcentages et que vous avez passé une variable numérique (résultat invalide).
wk version	Chaîne	version	Retourne le numéro de version 4DWP interne du document (réel). Ce numéro peut uniquement être lu par la commande WP LIRE ATTRIBUTS , il ne peut pas être fixé. Définit l'alignement vertical d'un élément. Peut être utilisé avec les caractères, les paragraphes, les images, les tableaux, les lignes de tableaux et les colonnes/cellules de tableaux. Valeurs possibles : <ul style="list-style-type: none"> • <u>wk baseline</u> (défaut) : aligne la base de l'élément avec la base de l'élément parent • <u>wk top</u> : aligne le haut de l'élément avec le haut de l'élément le plus grand sur la ligne • <u>wk bottom</u> : aligne le bas de l'élément avec le bas de l'élément le plus bas sur la ligne • <u>wk middle</u> : aligne l'élément au milieu de l'élément parent • <u>wk superscript</u> : aligne l'élément comme s'il était en exposant • <u>wk subscript</u> : aligne l'élément comme s'il était en indice <p>Pour les caractères, <u>wk top</u> et <u>wk bottom</u> ont le même effet que <u>wk baseline</u>. Pour les paragraphes, <u>wk baseline</u>, <u>wk superscript</u> et <u>wk subscript</u> ont le même effet que <u>wk top</u>. Pour les tableaux, lignes de tableaux et colonnes/cellules de tableaux, seules les valeurs <u>wk top</u>, <u>wk bottom</u> et <u>wk middle</u> sont prises en charge. Définit la largeur de l'élément. Valeurs possibles : <ul style="list-style-type: none"> • <u>wk auto</u> (défaut) : la largeur est basée sur le contenu de l'élément • Taille définie : taille exprimée par une valeur de type réel ou chaîne : <ul style="list-style-type: none"> ◦ Réel : Taille en <u>wk layout unit</u>. ◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. L'attribut <u>wk width</u> est écrasé par l'attribut <u>wk min width</u> (si défini).</p>
wk vertical align	Chaîne	verticalAlign	
wk width	Chaîne	width	
wk word	Entier long	6	Les lignes sont associées aux mots uniquement (exclusion des espaces) (valeur pour <u>wk text linethrough style</u> et/ou <u>wk text underline style</u>)

☰ Attributs 4D Write Pro

Les attributs 4D Write Pro vous permettent de contrôler tous les aspects graphiques du texte et des images contenus dans vos documents. Ces attributs sont gérés par les commandes suivantes :

- **WP FIXER ATTRIBUTS**
- **WP LIRE ATTRIBUTS**
- **WP REINITIALISER ATTRIBUTS**

Note : Vous pouvez également manipuler les attributs des zones 4D Write Pro via la notation objet ou les commandes 4D génériques :

- Notation objet(*) - par exemple, vous pouvez utiliser l'une ou l'autre des instructions suivantes :

```
$bcol:=$range[wk background color]
$bcol:=$range.backgroundColor //same results
```

- Commandes **OB FIXER** et **OB Lire**, par exemple :

```
$bcol:=OB Lire($range;wk background color)
```

(*) La notation objet doit être activée dans la base de données. Pour plus d'informations, reportez-vous à la section **Utiliser la notation objet**.

Fond

Les attributs de fond (*background*) permettent de définir des effets d'arrière-plan dans vos documents. Ces attributs s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
X	X		X	X	X	X	X

Constante Comment

Définit la zone de fond. Valeurs possibles :

- `wk background clip`
 - `wk border box` (défaut) : le fond est jusqu'au bord extérieur de la bordure
 - `wk content box` : le fond est à l'intérieur de la zone "content"
 - `wk padding box` : le fond est jusqu'au bord extérieur du padding (ou jusqu'au bord interne de la bordure, le cas échéant)
 - `wk paper box` : le fond est jusqu'au bord du document ou de la section

Définit la couleur de fond d'un élément. Valeurs possibles :

- `wk background color`
 - une couleur CSS ("`#command_5010101`" ou "`#command_5FFFFFF`" ou "red").
 - une valeur de couleur 4D (entier long, voir commande **OBJET FIXER COULEUR**)
 - un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

La valeur par défaut pour les documents est "`#command_5FFFFFF`" et `wk transparent` ou "transparent" pour les paragraphes et les images.

Définit la dimension verticale d'une image de fond. Valeurs possibles :

- `wk background height`
 - `wk auto` (défaut) : utilise la hauteur courante de l'image de fond
 - `wk contain` : redimensionne l'image à sa hauteur maximale de manière qu'elle "tienne" entièrement dans la zone "content" tout en préservant ses proportions. Cette option modifie également la valeur de l'autre attribut de dimension.
 - `wk cover` : redimensionne l'image à sa hauteur maximale de manière à ce que le fond soit entièrement recouvert par l'image 4D tout en préservant ses proportions. Certaines parties de l'image peuvent être tronquées. Cette option modifie également la valeur de l'autre attribut de dimension
 - Taille définie : définit la largeur de l'image via une valeur de type réel ou chaîne :
 - Réel : Taille en `wk layout unit`.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. Il est possible de passer une valeur relative (pourcentage %).

Définit une référence d'image à utiliser en fond. Peut être toute expression ou variable image 4D valide.

- `wk background image`
 - Valeur retournée (**WP LIRE ATTRIBUTS**) : si l'image avait été définie via un URL réseau, l'image cible est retournée si elle a déjà été chargée, sinon une image vide est retournée.

Utilisez `wk background image url` si vous souhaitez gérer des images via des URLs ou des URIs locaux.

Image de fond définie via un URL (chaîne). Peut être un URL réseau ou un data URI, absolu ou relatif au fichier de structure.

- `wk background image url`
 - Valeur retournée (**WP LIRE ATTRIBUTS**) : URL réseau ou URI de donnée. Cette valeur peut ne pas être égale à l'URL initial pour une image non référencée par un URL réseau (seuls les URLs réseau sont conservés). Pour les URLs de fichiers locaux, le flux d'image (stream) lui-même est conservé dans le document et donc l'URL retourné est un URI de données avec le flux d'image encodé en base64.

Utilisez `wk background image` pour gérer des images de fond via des expressions image.

Définit la position de l'image de fond. Valeurs possibles :

- `wk background origin`
 - `wk padding box` (défaut) : l'image de fond débute au rectangle de padding (ou à l'intérieur de la bordure)
 - `wk border box` : l'image de fond débute au rectangle de la bordure (bord extérieur)
 - `wk content box` : l'image de fond au rectangle "content"
 - `wk paper box` : l'image de fond débute au bord du document ou de la section

Définit la position horizontale de départ d'une image de fond. Valeurs possibles :

- `wk background position horizontal`
 - `wk left` (défaut) : l'image de fond débute horizontalement à gauche de l'élément
 - `wk center` : l'image de fond débute horizontalement au centre de l'élément
 - `wk right` : l'image de fond débute horizontalement à droite de l'élément

Définit la position verticale de départ d'une image de fond. Valeurs possibles :

- `wk background position vertical`
 - `wk top` (défaut) : l'image de fond débute verticalement en haut de l'élément
 - `wk middle` : l'image de fond débute verticalement au centre de l'élément
 - `wk bottom` : l'image de fond débute verticalement à droite de l'élément

Définit si et comment une image de fond est répétée. Valeurs possibles :

- `wk background repeat`
 - `wk repeat` (défaut) : l'image de fond est répétée verticalement et horizontalement
 - `wk no repeat` : l'image de fond n'est pas répétée
 - `wk repeat x` : l'image de fond est répétée horizontalement uniquement
 - `wk repeat y` : l'image de fond est répétée verticalement uniquement

Constante **Comment**

Définit la dimension horizontale d'une image de fond. Valeurs possibles :

wk
background
width

- wk auto (défaut) : utilise la largeur courante de l'image de fond
- wk contain : redimensionne l'image à sa largeur maximale de manière qu'elle "tienne" entièrement dans la zone "content" tout en préservant ses proportions. Cette option modifie également la valeur de l'autre attribut de dimension.
- wk cover : redimensionne l'image de manière à ce que le fond soit entièrement recouvert par l'image de fond tout en préservant ses proportions. Certaines parties de l'image peuvent être tronquées. Cette option modifie également la valeur de l'autre attribut de dimension
- Taille définie : définit la largeur de l'image via une valeur de type réel ou chaîne :
 - Réel : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt. Il est possible de passer une valeur relative (pourcentage %).

Bordures

Les attributs de bordure permettent de définir le style, la largeur et la couleur des bordures. Ces attributs s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
X	X		X	X	X	X	X

Constante Comment

Définit la couleur pour les quatre bordures. Valeurs possibles :

- une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red").
- une valeur de couleur 4D (entier long, voir commande **OBJET FIXER COULEUR**)
- un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

wk border color

La valeur par défaut est "command_5000000" (si valeur chaîne). S'il y a plusieurs couleurs, **WP LIRE ATTRIBUTS** retourne une chaîne vide.

Définit la couleur pour la bordure basse. Valeurs possibles :

- une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000"
- une valeur de couleur 4D (entier long, voir commande **OBJET FIXER COULEUR**)
- un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

wk border color bottom

Définit la couleur pour la bordure gauche. Valeurs possibles :

- une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000"
- une valeur de couleur 4D (entier long, voir commande **OBJET FIXER COULEUR**)
- un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

wk border color left

Définit la couleur pour la bordure droite. Valeurs possibles :

- une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000"
- une valeur de couleur 4D (entier long, voir commande **OBJET FIXER COULEUR**)
- un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

wk border color right

Définit la couleur pour la bordure haute. Valeurs possibles :

- une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000"
- une valeur de couleur 4D (entier long, voir commande **OBJET FIXER COULEUR**)
- un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

wk border color top

Définit l'arrondi des angles de la bordure. Valeurs possibles :

- wk none (défaut) : la bordure n'a pas d'angles arrondis
- Valeur d'angle exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Angle en wk layout unit.
 - Chaîne : chaîne CSS avec valeur et unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres.

wk border radius

Définit le style des quatre bordures. Valeurs possibles :

- wk none (défaut) : pas de bordure
- wk hidden : équivaut à wk none, sauf en résolution de conflit de bordure
- wk solid : bordure pleine
- wk dotted : bordure en pointillés
- wk dashed : bordure en tirets
- wk double : bordure double
- wk groove : bordure relief 3D (l'effet réel dépend de la couleur de la bordure)
- wk ridge : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure)
- wk inset : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure)
- wk outset : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)

wk border style

Définit le style de la bordure basse. Valeurs possibles :

- wk none (défaut) : pas de bordure basse
- wk hidden : équivaut à wk none, sauf en résolution de conflit de bordure
- wk solid : bordure pleine
- wk dotted : bordure en pointillés
- wk dashed : bordure en tirets
- wk double : bordure double
- wk groove : bordure relief 3D (l'effet réel dépend de la couleur de la bordure)
- wk ridge : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure)
- wk inset : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure)
- wk outset : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)

wk border style bottom

Constante Comment

Définit le style de la bordure gauche. Valeurs possibles :

- wk none (défaut) : pas de bordure gauche
- wk hidden : équivaut à wk none, sauf en résolution de conflit de bordure
- wk solid : bordure pleine
- wk dotted : bordure en pointillés
- wk dashed : bordure en tirets
- wk double : bordure double
- wk groove : bordure relief 3D (l'effet réel dépend de la couleur de la bordure)
- wk ridge : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure)
- wk inset : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure)
- wk outset : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)

wk border style left

Définit le style de la bordure droite. Valeurs possibles :

- wk none (défaut) : pas de bordure droite
- wk hidden : équivaut à wk none, sauf en résolution de conflit de bordure
- wk solid : bordure pleine
- wk dotted : bordure en pointillés
- wk dashed : bordure en tirets
- wk double : bordure double
- wk groove : bordure relief 3D (l'effet réel dépend de la couleur de la bordure)
- wk ridge : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure)
- wk inset : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure)
- wk outset : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)

wk border style right

Définit le style de la bordure haute. Valeurs possibles :

- wk none (défaut) : pas de bordure haute
- wk hidden : équivaut à wk none, sauf en résolution de conflit de bordure
- wk solid : bordure pleine
- wk dotted : bordure en pointillés
- wk dashed : bordure en tirets
- wk double : bordure double
- wk groove : bordure relief 3D (l'effet réel dépend de la couleur de la bordure)
- wk ridge : bordure relief 3D inversé (l'effet réel dépend de la couleur de la bordure)
- wk inset : bordure biseau 3D (l'effet réel dépend de la couleur de la bordure)
- wk outset : bordure biseau 3D inversé (l'effet réel dépend de la couleur de la bordure)

wk border style top

Définit la largeur des quatre bordures. Vous devez spécifier le style de la bordure avant sa largeur. Valeurs possibles :

- Largeur exprimée par une valeur de type numérique ou chaîne :
 - Numérique : largeur en wk layout unit.
 - Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres
- Valeur par défaut : 2pt

wk border width

Définit la largeur de la bordure basse. Valeurs possibles :

- Largeur exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Largeur en wk layout unit.
 - Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres
- Valeur par défaut : 2pt

wk border width bottom

Définit la largeur de la bordure gauche. Valeurs possibles :

- Largeur exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Largeur en wk layout unit.
 - Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres
- Valeur par défaut : 2pt

wk border width left

Définit la largeur de la bordure droite. Valeurs possibles :

- Largeur exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Largeur en wk layout unit.
 - Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres
- Valeur par défaut : 2pt

wk border width right

Définit la largeur de la bordure haute. Valeurs possibles :

- Largeur exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Largeur en wk layout unit.
 - Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres
- Valeur par défaut : 2pt

wk border width top

Constante **Comment**

wk inside	Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété inter-paragraphes correspondante (et pas à l'extérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande WP FIXER ATTRIBUTS .
wk outside	Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété de paragraphe externe correspondante (et pas à l'intérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande WP FIXER ATTRIBUTS .

Document

Les attributs de document sont utilisés pour fixer ou lire des meta informations relatives au document lui-même, telles que le sujet, le nom de l'auteur et de sa société, ou encore des commentaires. Ces attributs s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
X							X

Constante	Comment
wk author	Indique le nom de l'auteur du document (chaîne)
wk column count	(Attribut de plage en lecture seule) Nombre de colonnes de la plage. Valeur : Entier long
wk column rule color	Couleur du séparateur (trait) de colonne vertical. Valeurs possibles : <ul style="list-style-type: none"> • une couleur CSS ("#command_5010101" ou "#command_5FFFFFF" ou "red"). Valeur par défaut : "#command_5000000" (black) • une valeur entier long de couleur 4D (voir commande OBJET FIXER COULEUR) • un tableau entier long contenant un élément pour chaque composante R, V, B (0-255)
wk column rule style	Style du séparateur (trait) de colonne vertical. Valeurs possibles : <ul style="list-style-type: none"> • wk none (défaut) : pas de séparateur • wk hidden : équivaut à wk none, sauf en résolution de conflit • wk solid : ligne pleine • wk dotted : ligne en pointillés • wk dashed : ligne en tirets • wk double : ligne double • wk groove : ligne relief 3D (l'effet réel dépend de la couleur de la ligne) • wk ridge : ligne relief 3D inversé (l'effet réel dépend de la couleur de la ligne) • wk inset : ligne biseau 3D (l'effet réel dépend de la couleur de la ligne)
wk column rule width	Largeur du séparateur de colonne vertical. Valeurs possibles : <ul style="list-style-type: none"> • Réel : largeur en wk layout unit. • Chaîne : Chaîne CSS avec valeur et unité concaténées (e.g.: "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres). Valeur par défaut="2.5pt"
wk column spacing	(Documents ou sections uniquement) Espacement entre deux colonnes. Valeurs possibles : <ul style="list-style-type: none"> • Réel : largeur en wk layout unit • Chaîne : Chaîne CSS avec valeur et unité concaténées (e.g.: "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres). Valeur par défaut="12pt"
wk column width	(Documents ou sections uniquement) Attribut en lecture seulement. Largeur courante de chaque colonne, i.e. largeur calculée basée sur la largeur de la page, les marges de la page, le nombre de colonnes et l'espacement entre colonnes. Pour le document, le calcul utilise la largeur de la section par défaut, et peut donc être différent de la largeur réelle des colonnes de section(s) si certains attributs sont surchargés dans une section. Valeurs possibles : <ul style="list-style-type: none"> • Réel : largeur en wk layout unit. • Chaîne : Chaîne CSS avec valeur et unité concaténées (e.g.: "12pt" pour 12 points ou "1.5cm" pour 1,5 centimètres).
wk company	Définit une entreprise associée au document (chaîne)
wk date creation	Retourne la date de création du document (date). Cette valeur est en lecture seulement et ne peut pas être fixée.
wk date modified	Retourne la date de dernière modification du document (date). Cette valeur est en lecture seulement et ne peut pas être fixée.
wk dpi	DPI utilisés pour la conversion interne pixels <->points (entier). Toujours 96 (lecture seulement) Ne pas confondre cet attribut interne avec l'action standard "dpi" qui permet de modifier ou de lire les DPI de la vue courante du document.
wk layout unit	Définit l'unité de dimension par défaut du document lorsqu'une valeur numérique est écrite ou lue. Désigne l'unité de la règle et des attributs de dimension tels que wk width , à l'exception de wk font size , wk border width (et ses variantes), wk border radius et wk text shadow offset pour lesquels l'unité de valeur numérique est toujours le point. Valeurs possibles : <ul style="list-style-type: none"> • wk unit cm (défaut) : centimètres • wk unit pt : points • wk unit px : pixels • wk unit percent (seulement pour wk line height et wk background size h / wk background size v) • wk unit mm : millimètres • wk unit inch : pouces <p>Note : Lorsqu'une unité non prise en charge par la règle est sélectionnée via cet attribut (i.e. wk unit px ou wk unit percent), la règle utilise alors l'unité cm.</p>
wk notes	Définit des commentaires dans le document (chaîne).
wk subject	Définit le sujet du document (chaîne)
wk title	Définit le titre du document (chaîne). Par défaut "New 4D Write Pro Document"
wk version	Retourne le numéro de version 4DWP interne du document (réel). Ce numéro peut uniquement être lu par la commande WP LIRE ATTRIBUTS , il ne peut pas être fixé.

Police et texte

Ces attributs définissent la famille de police ainsi que la taille et le style du texte. Ils s'appliquent aux éléments suivants :

Documents	Paragraphe	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
	X			X*	X*	X*	X*

*Appliqué aux caractères des paragraphes dans les éléments

Constante Comment

wk font	Définit le nom de police complet avec les styles, comme retourné par la commande LISTE STYLES POLICE . Si vous passez un nom invalide, la commande ne fait rien. Valeur par défaut : "Times New Roman". Définit la graisse du texte (dépend des styles de police disponibles). Valeurs possibles :
wk font bold	<ul style="list-style-type: none">• <u>wk true</u> pour passer en style gras les caractères sélectionnés ; avec la commande WP LIRE ATTRIBUTS, <u>wk true</u> est retournée si au moins un des caractères sélectionnés prend en charge le style de police gras.• <u>wk false</u> (défaut) pour supprimer le style gras des caractères sélectionnés s'il y en a ; avec la commande WP LIRE ATTRIBUTS, <u>wk false</u> est retournée si aucun des caractères sélectionnés ne prend en charge le style de police gras.
wk font family	Spécifie le nom de famille de police tel que défini par <u>wk font</u> . Valeur par défaut : "Times New Roman". Une chaîne vide est retournée par la commande WP LIRE ATTRIBUTS si les caractères sélectionnés contiennent des propriétés de famille de police différentes. Définit le style italique du texte (dépend des styles de police disponibles). Valeurs possibles :
wk font italic	<ul style="list-style-type: none">• <u>wk true</u> pour passer en style italique ou oblique les caractères sélectionnés ; avec la commande WP LIRE ATTRIBUTS, <u>wk true</u> est retournée si au moins un des caractères sélectionnés prend en charge le style de police italique ou oblique.• <u>wk false</u> (défaut) pour supprimer le style italique ou oblique des caractères sélectionnés s'il y en a ; avec la commande WP LIRE ATTRIBUTS, <u>wk false</u> est retournée si aucun des caractères sélectionnés ne prend en charge le style de police italique ou oblique.
wk font size	Définit la taille de police du texte. Valeurs possibles (en points uniquement) : <ul style="list-style-type: none">• Réel (défaut = 12)• Chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points.
wk text color	Définit la couleur du texte. Valeurs possibles : <ul style="list-style-type: none">• une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000"• une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR)• un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)
wk text linethrough color	Définit la couleur de la ligne du texte barré. Valeurs possibles : <ul style="list-style-type: none">• une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000"• une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR)• un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)
wk text linethrough style	Par défaut, "currentColor" si chaîne, ou <u>wk default</u> si entier long. Définit le style du texte barré. Valeurs possibles : <ul style="list-style-type: none">• <u>wk none</u> (défaut) : pas d'effet texte barré• <u>wk solid</u> : dessine une ligne pleine sur le texte sélectionné• <u>wk dotted</u> : dessine une ligne en pointillés sur le texte sélectionné• <u>wk dashed</u> : dessine une ligne en tirets sur le texte sélectionné• <u>wk double</u> : dessine une ligne double sur le texte sélectionné• <u>wk semi transparent</u> : atténue la ligne sur le texte sélectionné. Peut être combiné à un autre style de ligne.• <u>wk word</u> : dessine une ligne sur les mots uniquement (exclut les blancs). Peut être combiné à un autre style de ligne.
wk text shadow color	Définit la couleur d'ombre du texte sélectionné. Valeurs possibles : <ul style="list-style-type: none">• une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red").• une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR)• un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)• <u>wk transparent</u> (défaut)
wk text shadow offset	Définit le décalage pour l'effet ombré. Valeurs possibles : <ul style="list-style-type: none">• Taille exprimée en points. Valeur par défaut : 1pt
wk text transform	Définit la casse des lettres dans le texte. Valeurs possibles : <ul style="list-style-type: none">• <u>wk capitalize</u> : les lettres initiales sont passées en capitales• <u>wk lowercase</u> : les lettres sont passées en minuscules• <u>wk uppercase</u> : les lettres sont passées en majuscules• <u>wk small uppercase</u> : les lettres sont passées en petites majuscules• <u>wk none</u> (défaut) : pas de transformation
wk text underline color	Définit la couleur du trait de soulignement. Valeurs possibles : <ul style="list-style-type: none">• une couleur CSS ("command_5010101" ou "command_5FFFFFFF" ou "red"). La valeur par défaut est "command_5000000"• une valeur de couleur 4D (entier long, voir commande OBJET FIXER COULEUR)• un tableau entier long contenant un élément pour chaque composante R, G, B (0-255)

Par défaut, "currentColor" si chaîne, ou wk default si entier long.

Constante Comment

Définit le style du trait de soulignement. Valeurs possibles :

- wk none (défaut) : pas de soulignement
- wk solid : dessine un trait plein
- wk dotted : dessine un trait en pointillés
- wk dashed : dessine un trait en tirets
- wk double : dessine un trait double
- wk semi transparent : atténue le trait. Peut être combiné à un autre style de ligne.
- wk word : dessine le trait sur les mots uniquement (exclut les blancs). Peut être combiné à un autre style de trait.

wk text underline style

Définit l'alignement vertical d'un élément. Peut être utilisé avec les caractères, les paragraphes et les images. Valeurs possibles :

- wk baseline (défaut) : aligne la base de l'élément avec la base de l'élément parent
- wk top : aligne le haut de l'élément avec le haut de l'élément le plus grand sur la ligne
- wk bottom : aligne le bas de l'élément avec le bas de l'élément le plus bas sur la ligne
- wk middle : aligne l'élément au milieu de l'élément parent
- wk superscript : aligne l'élément comme s'il était en exposant
- wk subscript : aligne l'élément comme s'il était en indice

wk vertical align

Pour les caractères, wk top et wk bottom ont le même effet que wk baseline.

Pour les paragraphes, wk baseline, wk superscript et wk subscript ont le même effet que wk top.

Hauteur/Largeur

Les attributs hauteur/largeur permettent de gérer les dimensions des paragraphes et des images. Ces attributs s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
	X		X		X	X*	X**

*Appliqué aux cellules

**Lecture seulement pour les en-têtes et les pieds de page (fixés automatiquement)

Constante Comment

Définit la hauteur de l'élément. La propriété *height* n'inclut pas le padding, les bordures ou les marges ; elle définit la hauteur de la zone située à l'intérieur du padding, de la bordure et de la marge de l'élément. Valeurs possibles :

- wk auto (défaut) : la hauteur est basée sur le contenu de l'élément
- Taille définie : taille exprimée par une valeur de type réel ou chaîne :
 - Réel : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt.

wk height

L'attribut wk height est écrasé par l'attribut wk min height (si défini).

Définit la hauteur minimum de l'élément. La valeur de la propriété wk height ne peut pas devenir plus petite que wk min height. Valeurs possibles :

- wk auto (défaut) : la hauteur minimum est basée sur le contenu de l'élément
- Taille définie : taille exprimée par une valeur de type réel ou chaîne :
 - Réel : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt.

wk min height

L'attribut wk min height écrase la valeur de l'attribut wk height.

Note : Cet attribut n'est pas pris en charge par les colonnes, les lignes et les cellules.

Définit la largeur minimum de l'élément. La valeur de la propriété wk width ne peut pas devenir plus petite que wk min width. Valeurs possibles :

- wk auto (défaut) : la largeur minimum est basée sur le contenu de l'élément
- Taille définie : taille exprimée par une valeur de type réel ou chaîne :
 - Réel : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt.

wk min width

L'attribut wk min width écrase la valeur de l'attribut wk width.

Définit la largeur de l'élément. Valeurs possibles :

- wk auto (défaut) : la largeur est basée sur le contenu de l'élément
- Taille définie : taille exprimée par une valeur de type réel ou chaîne :
 - Réel : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt.

wk width

L'attribut wk width est écrasé par l'attribut wk min width (si défini).

Image

Les attributs Image permettent de gérer les images insérées dans la zone ou ajoutées dans la page. Ils s'appliquent aux éléments suivants :

Documents	Paragraphe	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
			X	X*	X*	X*	

*Appliqué aux images dans les cellules (images insérées dans le texte uniquement)

Rappel : Comme indiqué dans la section **Manipuler des images**, 4D Write Pro prend en charge deux types d'images :

- les images insérées dans le texte via les commandes **WP INSERER IMAGE** ou **ST INSERER EXPRESSION**
- les images ancrées dans la page via la commande **WP Ajouter image**.

Les attributs suivants sont disponibles aussi bien pour les images insérées que pour les images ancrées :

Constante Comment

Définit une référence d'image. Peut être toute expression ou variable image 4D valide.

wk image • Valeur retournée (**WP LIRE ATTRIBUTS**) : si l'image avait été définie via un URL réseau, l'image cible est retournée si elle a déjà été chargée, sinon une image vide est retournée.

Utilisez wk image url si vous souhaitez gérer des images via des URLs ou des URIs locaux.

wk image
alternate
text

Définit un texte alternatif à l'image, si elle ne peut pas être affichée.

Image définie via un URL (chaîne). Peut être un URL réseau ou un data URI, absolu ou relatif au fichier de structure.

wk image url • Valeur retournée (**WP LIRE ATTRIBUTS**) : URL réseau ou URI de donnée). Cette valeur peut ne pas être égale à l'URL initial pour une image non référencée par un URL réseau (seuls les URLs réseau sont conservés). Pour les URLs de fichiers locaux, le flux d'image (stream) lui-même est conservé dans le document et donc l'URL retourné est un URI de données avec le flux d'image encodé en base64.

Utilisez wk image pour gérer des images via des expressions image.

wk owner

(Attribut en lecture seule) Possesseur de la plage. Valeur : objet
(Attribut de plage en lecture seule) Type d'objet 4D Write Pro. Valeurs possibles :

- wk type default : Plage sans type défini
- wk type paragraph : Plage de type paragraphe
- wk type image : Image (ancrée ou en ligne)
- wk type image anchored : Image ancrée
- wk type image inline : Image en ligne
- wk type container : En-tête ou pied, par exemple
- wk type table : Référence de tableau
- wk type table row : Référence de ligne de tableau
- wk type table cell : Référence de cellule de tableau
- wk type table column : Référence de colonne de tableau (uniquement pour plage colonne)
- wk type body : Référence de corps (body)

wk type

Les attributs suivants sont disponibles pour les images insérées uniquement :

Constante Comment

wk end (Attribut de plage en lecture seule) Position fin de plage. Valeur : entier long

wk start (Attribut de plage en lecture seule) Position début de plage. Valeur : entier long

Définit l'alignement vertical d'un élément. Peut être utilisé avec les caractères, les paragraphes et les images. Valeurs possibles :

- wk vertical align
- wk baseline (défaut) : aligne la base de l'élément avec la base de l'élément parent
 - wk top : aligne le haut de l'élément avec le haut de l'élément le plus grand sur la ligne
 - wk bottom : aligne le bas de l'élément avec le bas de l'élément le plus bas sur la ligne
 - wk middle : aligne l'élément au milieu de l'élément parent
 - wk superscript : aligne l'élément comme s'il était en exposant
 - wk subscript : aligne l'élément comme s'il était en indice

Pour les caractères, wk top et wk bottom ont le même effet que wk baseline.

Pour les paragraphes, wk baseline, wk superscript et wk subscript ont le même effet que wk top.

Les attributs suivants sont disponibles pour les images ancrées uniquement :

Constante Comment

wk anchor horizontal align	<p>Définit l'alignement horizontal d'une image relativement à l'origine (voir wk anchor origin). Valeurs possibles :</p> <ul style="list-style-type: none">• wk left - alignement à gauche• wk center - alignement au centre (<i>non compatible HTML, image non affichée sur le web</i>)• wk right - alignement à droite
wk anchor horizontal offset	<p>Définit le décalage horizontal d'une image exprimé en dimension CSS de type chaîne ou entier long (cm ou pt ou pixel) dans l'unité wk layout unit. Valeurs possibles :</p> <ul style="list-style-type: none">• Limite gauche ou droite de l'image relativement à wk anchor horizontal align• Limite gauche ou droite du corps (body) en mode inclus (si wk anchor section = wk anchor embedded) <p>Valeur par défaut = 0.</p> <p>Définit la position relative d'une image par rapport au texte ou à la page. Valeurs possibles :</p> <ul style="list-style-type: none">• wk behind text - l'image est ancrée derrière le texte• wk in front of text - l'image est ancrée devant le texte• wk inline with text - l'image est insérée dans le texte (position par défaut pour les images insérées avec WP INSERER IMAGE). Cet attribut est en lecture seule (les images dans le texte ne peuvent pas être converties en images ancrées par programmation).
wk anchor layout	<p>Définit si l'image est ancrée dans la page, l'en-tête ou le pied. Valeurs possibles :</p> <ul style="list-style-type: none">• wk paper box (défaut) - l'image est ancrée en bordure de page• wk header box - l'image est ancrée dans l'en-tête du document. Si l'en-tête n'est pas visible, l'image n'est pas affichée.• wk footer box - l'image est ancrée dans le pied du document. Si le pied n'est pas visible, l'image n'est pas affichée. <p>Ce sélecteur est ignoré en mode inclus.</p> <p>Définit le numéro de page ou le type de page où l'image est ancrée. Valeurs possibles :</p> <ul style="list-style-type: none">• wk anchor all - l'image est ancrée dans toutes les pages de la ou des section(s) définie(s) par wk anchor section• wk anchor embedded - l'image est ancrée au corps (body) du document en mode inclus (l'image est visible uniquement en mode inclus)• wk anchor first page - l'image est ancrée dans la première page de la ou des section(s) définie(s) par wk anchor section• wk anchor left page - l'image est ancrée dans la page gauche de la ou des section(s) définie(s) par wk anchor section• wk anchor right page - l'image est ancrée dans la page droite de la ou des section(s) définie(s) par wk anchor section• valeur numérique (entier long >= 0) - indique le numéro de la page dans laquelle l'image est ancrée. Dans ce cas, wk anchor section = wk anchor embedded. L'ancrage de section est ignoré si une image est ancrée dans une seule page. <p>Note : Les images en mode Page ne sont pas affichées dans les navigateurs.</p>
wk anchor page	<p>Définit le numéro de section ou le type de section où l'image est ancrée. Valeurs possibles :</p> <ul style="list-style-type: none">• wk anchor all (par défaut) - l'image est ancrée dans toutes les sections du document (l'image est visible uniquement en mode page)• wk anchor embedded - l'image est ancrée au corps (body) du document en mode inclus (l'image est visible uniquement en mode inclus).• valeur numérique (entier long >= 1) - indique le numéro de la section dans laquelle l'image est ancrée (l'image est visible uniquement en mode page). <p>Note : Les images en mode Page ne sont pas affichées dans les navigateurs.</p>
wk anchor section	<p>Définit l'alignement vertical d'une image relativement à l'origine (voir wk anchor origin). Valeurs possibles :</p> <ul style="list-style-type: none">• wk top - alignement en haut• wk center - alignement au centre (<i>non compatible HTML, image non affichée sur le web</i>)• wk bottom - alignement en bas
wk anchor vertical align	<p>Définit le décalage vertical d'une image exprimé en dimension CSS de type chaîne ou nombre (cm ou pt ou pixel) dans l'unité wk layout unit. Valeurs possibles :</p> <ul style="list-style-type: none">• Haut, centre ou bas de la page (voir wk anchor horizontal align)• Haut, centre ou bas du corps (body) en mode inclus (si wk anchor section = wk anchor embedded) <p>Valeur par défaut = 0.</p>
wk anchor vertical offset	<p>Définit le décalage vertical d'une image exprimé en dimension CSS de type chaîne ou nombre (cm ou pt ou pixel) dans l'unité wk layout unit. Valeurs possibles :</p> <ul style="list-style-type: none">• Haut, centre ou bas de la page (voir wk anchor horizontal align)• Haut, centre ou bas du corps (body) en mode inclus (si wk anchor section = wk anchor embedded) <p>Valeur par défaut = 0.</p>
wk id	<p>(Attribut en lecture seule) ID interne de l'élément. Valeur : Chaîne</p>

Liens

Les attributs des liens sont utilisés pour définir ou lire des URLs dans les pages. Ces attributs s'appliquent aux éléments suivants :

Documents	Paragraphe	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
	X	X	X*	X	X	X	

*Images insérées dans le texte uniquement

Constante **Comment**

Lien hypertexte associé à la plage. Valeurs possibles :

- wk link url
- url absolu, par exemple "http://www.4d.com/"
 - lien relatif, par exemple "/test/page.html" (le lien est relatif au fichier de structure de la base)
 - chaîne vide = pas de lien

Listes

4D Write Pro prend en charge deux principaux types de listes :

- les listes non ordonnées : listes aussi appelées "listes à puces"
- les listes ordonnées : listes dont les "marqueurs" d'éléments sont des chiffres ou des lettres et sont incrémentés

Les attributs de liste permettent de configurer la police ainsi que les marqueurs des éléments. Ces attributs s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
	X			X*	X*	X*	

*Appliqués aux paragraphes dans les cellules

Constante Comment

wk list font	Spécifie un nom complet de police, comme retourné par la commande LISTE STYLES POLICE , pour afficher le marqueur de l'élément de liste (et non le texte du paragraphe). Si le système ne reconnaît pas le nom de police, il gère la substitution. Si vous passez un nom de police invalide, la commande ne fait rien. Valeur par défaut : "Times".
wk list font family	Spécifie un nom de famille de police tel que défini par <u>wk list font</u> , utilisé pour afficher le marqueur de l'élément de liste (et non le texte du paragraphe). Valeur par défaut : "Times".
wk list start number	Définit la valeur de départ d'une liste ordonnée. Valeurs possibles : <ul style="list-style-type: none">• <u>wk auto</u> (défaut) : la valeur de départ dépend des éléments de liste précédents s'il y en a.• valeur entière : valeur de départ
	Marqueur d'élément de liste pour les paragraphes gauche à droite. Si défini, remplace le marqueur d'élément de la liste.
wk list string format LTR	<ul style="list-style-type: none">• Pour les listes non ordonnées : chaîne utilisée en tant que marqueur d'élément de liste (généralement un seul caractère, par exemple"-")• Pour les listes ordonnées : chaîne contenant le caractère "#command_5". "#command_5" est un caractère d'emplacement pour le nombre ou la lettre calculé(e). Par défaut "#command_5.", donc par exemple si le numéro d'élément courant de la liste est 15 et que le type de style de liste est décimal, la chaîne du marqueur d'élément de liste sera "15."
	Marqueur d'élément de liste pour les paragraphes droite à gauche. Si défini, remplace le marqueur d'élément de la liste.
wk list string format RTL	<ul style="list-style-type: none">• Pour les listes non ordonnées : chaîne utilisée en tant que marqueur d'élément de liste (généralement un seul caractère, par exemple"-")• Pour les listes ordonnées : chaîne contenant le caractère "#command_5". "#command_5" est un caractère d'emplacement pour le nombre ou la lettre calculé(e). Par défaut "#command_5.", donc par exemple si le numéro d'élément courant de la liste est 15 et que le type de style de liste est décimal, la chaîne du marqueur d'élément de liste sera "15."
	Définit une référence d'image comme marqueur d'élément de liste dans une liste non ordonnée. Valeurs possibles :
wk list style image	<ul style="list-style-type: none">• <u>wk none</u> (défaut) : le marqueur d'élément de liste n'est pas une image• toute expression ou variable image 4D valide.• Valeur retournée (WP LIRE ATTRIBUTS) : si l'image avait été définie via un URL réseau, l'image cible est retournée si elle a déjà été chargée, sinon une image vide est retournée.
	Utilisez <u>wk list style image url</u> si vous souhaitez gérer des images via des URLs ou des URIs locaux.
	Définit la hauteur d'une image utilisée comme marqueur d'élément de liste. Valeurs possibles :
wk list style image height	<ul style="list-style-type: none">• <u>wk auto</u> (défaut) : utilise la hauteur courante de l'image• Taille définie : valeur de type réel ou chaîne<ul style="list-style-type: none">◦ Réel : Taille en <u>wk layout unit</u>.◦ Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt.
	Image à utiliser comme marqueur d'élément de liste dans une liste non ordonnée. Valeurs possibles :
wk list style image url	<ul style="list-style-type: none">• <u>wk none</u> (défaut) : le marqueur d'élément de liste n'est pas une image• un URL réseau ou un data URI, absolu ou relatif au fichier de structure• Valeur retournée (WP LIRE ATTRIBUTS) : URL réseau ou URI de donnée). Cette valeur peut ne pas être égale à l'URL initial pour une image non référencée par un URL réseau (seuls les URLs réseau sont conservés). Pour les URLs de fichiers locaux, le flux d'image (stream) lui-même est conservé dans le document et donc l'URL retourné est un URI de données avec le flux d'image encodé en base64.
	Utilisez <u>wk list style image</u> pour gérer des images de marqueurs d'éléments via des expressions.

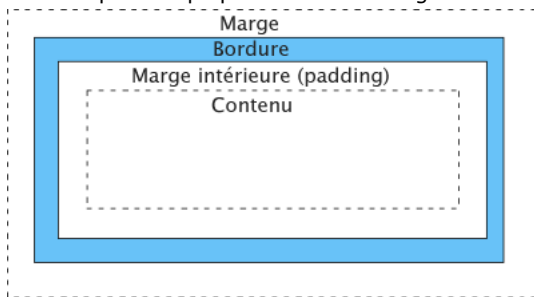
Constante Comment

Spécifie le type de marqueur d'élément d'une liste ordonnée ou non ordonnée. Valeurs possibles :

- [wk disc](#) (défaut)
 - [wk circle](#)
 - [wk square](#)
 - [wk decimal](#) : 1 2 3
 - [wk decimal leading zero](#) : 01 02 03
 - [wk lower latin](#) : a b c
 - [wk lower roman](#) : i ii iii iv
 - [wk upper latin](#) : A B C
 - [wk upper roman](#) : I II III IV
 - [wk lower greek](#) : alpha, beta, gamma, etc.
 - [wk armenian](#)
 - [wk georgian](#)
 - [wk hebrew](#)
 - [wk hiragana](#)
 - [wk katakana](#)
 - [wk cjk ideographic](#)
 - [wk hollow square](#)
 - [wk diamond](#)
 - [wk club](#)
 - [wk decimal greek](#)
 - [wk custom](#) : par défaut, liste non ordonnée avec "-" comme marqueur d'élément de liste ; il s'agit d'un style pratique permettant de personnaliser un marqueur d'élément de liste avec [wk list string format LTR](#) ou [wk list string format RTL](#) sans devoir modifier les marqueurs de liste standard
 - [wk none](#)
- wk list
style type

Marges

Les marges sont les zones situées au-delà de la bordure d'un élément. Elles sont transparentes. Le schéma suivant illustre les diverses parties qui peuvent être configurées :



Les attributs de marge s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
X	X		X	X			X

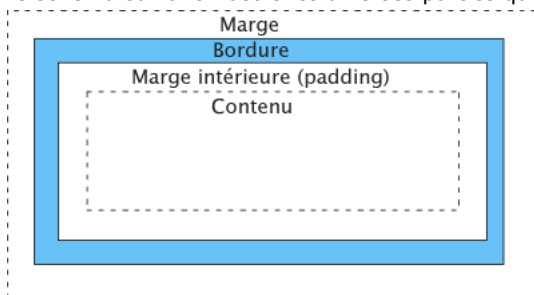
Constante Comment

wk inside	<p>Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété inter-paragraphes correspondante (et pas à l'extérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande WP FIXER ATTRIBUTS.</p> <p>Définit la taille de toutes les marges de l'élément. Valeurs possibles :</p> <ul style="list-style-type: none">• Taille exprimée par une valeur de type numérique ou chaîne :<ul style="list-style-type: none">◦ Numérique : taille en <u>wk layout unit</u>.◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres• <u>wk none</u> (défaut) : pas de marge spécifique
wk margin	<p>Définit la taille de la marge basse de l'élément. Valeurs possibles :</p> <ul style="list-style-type: none">• Taille exprimée par une valeur de type numérique ou chaîne :<ul style="list-style-type: none">◦ Numérique : taille en <u>wk layout unit</u>.◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres• <u>wk none</u> (défaut) : pas de marge spécifique
wk margin bottom	<p>Définit la taille de la marge gauche de l'élément. Valeurs possibles :</p> <ul style="list-style-type: none">• Taille exprimée par une valeur de type numérique ou chaîne :<ul style="list-style-type: none">◦ Numérique : taille en <u>wk layout unit</u>.◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres• <u>wk none</u> (défaut) : pas de marge spécifique
wk margin left	<p>Définit la taille de la marge droite de l'élément. Valeurs possibles :</p> <ul style="list-style-type: none">• Taille exprimée par une valeur de type numérique ou chaîne :<ul style="list-style-type: none">◦ Numérique : taille en <u>wk layout unit</u>.◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres• <u>wk none</u> (défaut) : pas de marge spécifique
wk margin right	<p>Définit la taille de la marge haute de l'élément. Valeurs possibles :</p> <ul style="list-style-type: none">• Taille exprimée par une valeur de type numérique ou chaîne :<ul style="list-style-type: none">◦ Numérique : taille en <u>wk layout unit</u>.◦ Chaîne : chaîne CSS avec la valeur et l'unité concaténées. Ex. : 12pt pour 12 points ou 1.5cm pour 1,5 centimètres• <u>wk none</u> (défaut) : pas de marge spécifique
wk margin top	<p>Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété de paragraphe externe correspondante (et pas à l'intérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande WP FIXER ATTRIBUTS.</p>
wk outside	

Marges intérieures (padding)

Les marges intérieures (*Padding*) sont l'espace blanc entre le contenu de l'élément et sa bordure. Les marges intérieures sont affectées par la couleur de fond de l'élément.

Le schéma suivant illustre les diverses parties qui peuvent être configurées :



Les attributs de marge intérieure s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
X	X		X	X*	X*	X	X

*Appliqués aux cellules

Constante Comment

wk inside Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété inter-paragraphes correspondante (et pas à l'extérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande **WP FIXER ATTRIBUTS**.

wk outside Lorsque la zone sélectionnée contient plusieurs paragraphes, définit que l'attribut doit être appliqué uniquement sur la propriété de paragraphe externe correspondante (et pas à l'intérieur). Peut être utilisé avec les attributs de bordures, padding ou marges, et doit être ajouté à l'attribut spécifié. Voir exemple 2 de la commande **WP FIXER ATTRIBUTS**.

Définit la taille de la marge intérieure (*padding*) pour tous les côtés de l'élément. Valeurs possibles :

- Taille exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres.
- wk none (défaut) : pas de marge intérieure spécifique

Définit la taille de la marge intérieure (*padding*) pour le bas de l'élément. Valeurs possibles :

- Taille exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres.
- wk none (défaut) : pas de marge intérieure spécifique

Définit la taille de la marge intérieure (*padding*) pour la gauche de l'élément. Valeurs possibles :

- Taille exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres.
- wk none (défaut) : pas de marge intérieure spécifique

Définit la taille de la marge intérieure (*padding*) pour la droite de l'élément. Valeurs possibles :

- Taille exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres.
- wk none (défaut) : pas de marge intérieure spécifique

Définit la taille de la marge intérieure (*padding*) pour le haut de l'élément. Valeurs possibles :

- Taille exprimée par une valeur de type numérique ou chaîne :
 - Numérique : Taille en wk layout unit.
 - Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres.
- wk none (défaut) : pas de marge intérieure spécifique

Paragraphes

Les attributs de paragraphe permettent de définir les propriétés d'organisation du texte dans les paragraphes. Ils s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
	X			X*	X*	X*	X*

*Appliqués aux paragraphes dans les cellules

Constante Comment

	Définit le sens du paragraphe. Valeurs possibles :
wk direction	<ul style="list-style-type: none">• <u>wk left to right</u> (défaut)• <u>wk right to left</u>
	Définit l'espacement entre les lignes. Valeurs possibles :
wk line height	<ul style="list-style-type: none">• <u>wk normal</u> (défaut) : utiliser une valeur basée sur la taille du texte.• Hauteur exprimée par une valeur de type numérique ou chaîne :<ul style="list-style-type: none">◦ Numérique : hauteur en <u>wk layout unit</u>.◦ Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Il est possible de passer une valeur relative (pourcentage %).
	Définit les caractères de conduite pour les tabulations dans les plages de paragraphes. Valeurs possibles :
wk tab stop leadings	<ul style="list-style-type: none">• Tableau texte Liste ordonnée de caractères de conduite pour chaque position de tabulation, à partir de la marge gauche. Le caractère défini par la dernière valeur est répété pour chaque tabulation supplémentaire saisie dans le paragraphe.• Texte Un seul caractère est utilisé comme caractère de conduite. Si un texte contenant plusieurs caractères est passé, le premier caractère est utilisé.
	Propriété synchronisée avec les tableaux <u>wk tab stop offsets</u> et <u>wk tab stop types</u> . Si les tableaux n'ont pas la même taille (et ne sont pas vides), le dernier caractère du tableau est utilisé comme caractère par défaut pour les tabulations. Si les tableaux sont vides, le caractère de conduite None est utilisé.
	Note : Le dernier caractère utilisé est toujours le caractère par défaut pour les tabulations.
	Définit les tabulations pour le paragraphe. Valeurs possibles :
wk tab stop offsets	<ul style="list-style-type: none">• Tableau de valeurs de tabulations : une liste ordonnée de valeurs absolues, à partir de la marge gauche. La dernière position de tabulation définie est répétée pour chaque nouvelle tabulation saisie dans le paragraphe. Si la position de tabulation est supérieure à la largeur du paragraphe, le texte passe à la ligne suivante et débute à la première tabulation. Si une valeur du tableau est plus petite que la précédente, elle est ignorée.• Valeur fixe (défaut 35.45pt) : position par défaut pour tout le paragraphe. La commande WP LIRE ATTRIBUTS retourne la dernière position (qui est la position relative par défaut pour les positions définies au-delà de la dernière position absolue).
	Note : Vous ne pouvez pas utiliser de tableau et de valeurs fixes dans le même appel pour différents attributs. Les valeurs sont exprimées à l'aide de chaînes CSS (défaut) ou de réels dans <u>wk layout unit</u> . Valeur maximum : 10000pt.
	Si des tableaux sont utilisés, cette propriété est synchronisée avec les tableaux <u>wk tab stop leadings</u> et <u>wk tab stop types</u> .
	Définit le type de tabulation pour le paragraphe. Valeurs possibles :
wk tab stop types	<ul style="list-style-type: none">• tableau de valeurs de types de tabulations (si les tabulations ont été définies via un tableau)• <u>wk left</u> (défaut) : le texte est décalé vers la droite à partir de la tabulation• <u>wk right</u> : le texte est décalé vers la gauche à partir de la tabulation jusqu'à ce que l'espace de la tabulation soit rempli.• <u>wk center</u> : le texte est centré sur la tabulation• <u>wk decimal</u> : le texte avant le point décimal est décalé vers la gauche, le texte après le point décimal est décalé vers la droite• <u>wk bar</u> : affiche une barre verticale à l'emplacement spécifié.
	Si des tableaux sont utilisés, cette propriété est synchronisée avec les tableaux <u>wk tab stop leadings</u> et <u>wk tab stop offsets</u> .
	Définit l'alignement horizontal du texte dans le paragraphe. Valeurs possibles :
wk text align	<ul style="list-style-type: none">• <u>wk left</u> (défaut)• <u>wk right</u>• <u>wk justify</u>• <u>wk center</u>
	Définit l'indentation de la première ligne du paragraphe. Valeurs possibles :
wk text indent	<ul style="list-style-type: none">• Réel : Taille en wk layout unit. Défaut : 0• Chaîne : Chaîne CSS avec valeur et unité concaténées. Ex. : 12pt pour 12 points, ou 1.5cm pour 1,5 centimètres. Valeur minimum : 0pt, valeur maximum : 10000pt.

Constante Comment

Définit l'alignement vertical d'un élément. Peut être utilisé avec les caractères, les paragraphes et les images. Valeurs possibles :

- | | |
|-------------------|---|
| wk vertical align | <ul style="list-style-type: none">• wk baseline (défaut) : aligne la base de l'élément avec la base de l'élément parent• wk top : aligne le haut de l'élément avec le haut de l'élément le plus grand sur la ligne• wk bottom : aligne le bas de l'élément avec le bas de l'élément le plus bas sur la ligne• wk middle : aligne l'élément au milieu de l'élément parent• wk superscript : aligne l'élément comme s'il était en exposant• wk subscript : aligne l'élément comme s'il était en indice |
|-------------------|---|

Pour les caractères, [wk top](#) et [wk bottom](#) ont le même effet que [wk baseline](#).

Pour les paragraphes, [wk baseline](#), [wk superscript](#) et [wk subscript](#) ont le même effet que [wk top](#).

Feuilles de style

Les attributs de feuilles de style permettent de gérer les feuilles de style associées au contenu. Ils s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
	X		X**				X*

*Lecture seulement pour les en-têtes et les pieds de page, gérés par 4D Write Pro (il y a une feuille de style par en-tête ou pied de section)

**Images insérées dans le texte uniquement

Constante

Comment

Définit la feuille de style à utiliser en cas d'ajout d'une nouvelle ligne dans le paragraphe. Valeurs possibles :

- | | |
|-------------------------|--|
| wk new line style sheet | <ul style="list-style-type: none">• Nom de feuille de style existante• wk none (défaut) |
|-------------------------|--|

Définit la feuille de style courante pour le ou les élément(s) sélectionné(s). Valeurs possibles :

- | | |
|----------------|--|
| wk style sheet | <ul style="list-style-type: none">• Nom de feuille de style existante• wk none (défaut) |
|----------------|--|

Plages

Les attributs de plage sont en lecture seulement (sauf [wk id](#)) et permettent d'obtenir les propriétés des plages. Les attributs de plages sont applicables uniquement aux objets de type plage.

Constante

Comment

- | | |
|----------|---|
| wk end | (Attribut de plage en lecture seule) Position fin de plage. Valeur : entier long |
| wk id | ID interne de l'élément. Valeur : Chaîne |
| wk owner | (Attribut en lecture seule) Possesseur de la plage. Valeur : objet |
| wk start | (Attribut de plage en lecture seule) Position début de plage. Valeur : entier long
(Attribut de plage en lecture seule) Type d'objet 4D Write Pro. Valeurs possibles : |
| wk type | <ul style="list-style-type: none">• wk type default : Plage sans type défini• wk type paragraph : Plage de type paragraphe• wk type image : Image (ancrée ou en ligne)• wk type image anchored : Image ancrée• wk type image inline : Image en ligne• wk type container : En-tête ou pied, par exemple• wk type table : Référence de tableau• wk type table row : Référence de ligne de tableau• wk type table cell : Référence de cellule de tableau• wk type table column : Référence de colonne de tableau (uniquement pour plage colonne)• wk type body : Référence de corps (body) |

Tableaux

Les attributs de tableaux sont spécifiques aux tableaux. Ils s'appliquent aux éléments suivants :

Documents	Paragraphes	Caractères	Images	Tableaux	Lignes	Colonnes/Cellules	En-têtes/Pieds/Corps
				X			

Constante	Comment
wk cell count	Nombre total de cellules dans la ligne. Type de valeur : Entier long (valeur pour wk_type_table_row)
wk column count	(Attribut de plage en lecture seule) Nombre de colonnes de la plage. Valeur : Entier long
wk first column	(Attribut en lecture seule) Numéro de la première colonne. Valeur : Entier long
wk first row	(Attribut en lecture seule) Numéro de la première ligne du tableau. Valeur : Entier long
wk row count	(Attribut en lecture seule) Nombre total de lignes. Valeur : Entier long
	Définit l'alignement horizontal pour un élément de tableau. Valeurs possibles :
wk table align	<ul style="list-style-type: none"> • wk_left (défaut) • wk_right • wk_center
wk table ID	(Attribut en lecture seule) ID du tableau parent. Valeur : Chaîne

Lorsque des attributs différents sont appliqués à des éléments d'un tableau, un ordre de priorité est appliqué pour le rendu :

1. Le tableau est dessiné en premier
2. Les lignes sont ensuite dessinées (remplaçant les attributs du tableau)
3. Les cellules/colonnes sont ensuite dessinées (remplaçant les attributs des lignes).