

# FileMaker® Server 11

Publication Web personnalisée  
avec XML et XSLT



© 2007-2010 FileMaker, Inc. Tous droits réservés.

FileMaker, Inc.  
5201 Patrick Henry Drive  
Santa Clara, Californie 95054

FileMaker est une marque de FileMaker, Inc., déposée aux Etats-Unis et dans d'autres pays. Le logo en forme de dossier est une marque de FileMaker, Inc. Toutes les autres marques sont la propriété de leurs détenteurs respectifs.

La documentation de FileMaker est protégée par la législation sur les droits d'auteur. Vous n'êtes pas autorisé à créer des copies supplémentaires ni à distribuer cette documentation sans l'accord écrit de FileMaker. Vous devez posséder une copie sous licence valide de FileMaker pour utiliser cette documentation.

Toutes les personnes, sociétés, adresses email et URL citées dans les exemples sont fictives et toute ressemblance avec des personnes et des sociétés existantes ne serait que pure coïncidence. La liste des auteurs est disponible dans les documents Remerciements fournis avec ce logiciel. Les produits tiers et les adresses URL sont mentionnés à titre indicatif uniquement, et non pas à titre de recommandation. FileMaker, Inc. se dégage de toute responsabilité concernant les performances de ces produits.

Pour plus de détails, consultez notre site Web à l'adresse <http://www.filemaker.fr>.

Edition : 01

# ***Table des matières***

Présentation de ce guide	9
--------------------------	---

## **Chapitre 1**

### ***Introduction à la Publication Web personnalisée***

A propos du moteur de publication Web	12
Traitement d'une requête du moteur de publication Web	12
Publication Web personnalisée avec PHP	13
Publication Web personnalisée avec XML et XSLT	13
Comparaison de PHP avec XML et XSLT	14
Raisons de choisir PHP	14
Raisons de choisir XML et XSLT	14

## **Chapitre 2**

### ***A propos de la publication Web personnalisée avec XML et XSLT***

Création de sites Web dynamiques avec le moteur de publication Web	17
A propos de la fonction de publication Web personnalisée utilisant le langage XML	17
A propos de la fonction de publication Web personnalisée utilisant le langage XSLT	18
A propos du développement des feuilles de style XSLT	18
Principales fonctionnalités de la fonction de publication Web personnalisée utilisant les langages XML et XSLT	18
Configuration pour la publication Web	19
Configuration requise pour la publication d'une base de données à l'aide de la Publication Web personnalisée	19
Éléments à utiliser par les utilisateurs Web pour accéder à une solution de publication Web personnalisée	20
Connexion à Internet ou à un intranet	20
Étape suivante	20

## **Chapitre 3**

### ***Préparation des bases de données pour la Publication Web personnalisée***

Activation de la Publication Web personnalisée dans une base de données	21
Accès à une base de données protégée	22
Protection de vos bases de données publiées	22
Prise en charge du serveur Web pour les types de média Internet (MIME)	23
A propos de la publication du contenu des rubriques Multimédia sur le Web	23
Publication d'objets de rubrique Multimédia stockés dans une base de données	23
Publication d'objets de rubrique Multimédia stockés comme référence externe	23
Affichage des données d'une rubrique Multimédia par les utilisateurs Web	24

Scripts FileMaker et Publication Web personnalisée	24
Astuces et considérations à propos des scripts	25
Comportement des scripts dans les solutions de Publication Web personnalisée	26
Déclencheurs de scripts et solutions de Publication Web personnalisée	27

## Chapitre 4

### **Introduction à la Publication Web personnalisée avec XSLT**

A propos de feuilles de style XSLT FileMaker	29
Exemples d'utilisation des feuilles de style XSLT	29
Utilisation de la Publication Web personnalisée avec XSLT	30
Mode de génération des pages par le moteur de publication Web en fonction des données XML et des feuilles de style XSLT	30
Étapes générales pour l'utilisation de la Publication Web personnalisée avec XSLT	31
Utilisation de l'outil FileMaker XSLT Site Assistant pour générer des feuilles de style FileMaker XSLT	32
Avant d'utiliser XSLT Site Assistant	32
Démarrage de XSLT Site Assistant	33
Utilisation de XSLT Site Assistant	33
A propos des feuilles de style générées par XSLT Site Assistant	34
Utilisation des feuilles de style FileMaker XSLT dans un programme ou un site Web	34
Résolution des problèmes sur les feuilles de style XSLT	35

## Chapitre 5

### **Accès aux données XML avec le moteur de publication Web**

Utilisation de la Publication Web personnalisée avec XML	37
Différences entre le moteur de publication Web et les options d'importation/exportation FileMaker Pro XML	37
Génération de données XML par le moteur de publication Web à partir d'une requête	38
Processus général d'accès aux données XML à partir du moteur de publication Web	40
A propos de la syntaxe d'URL pour les données XML et les objets de type Multimédia	41
A propos de la syntaxe d'URL pour les données XML	41
A propos de la syntaxe d'URL pour les objets FileMaker de type Multimédia dans les solutions XML	42
A propos du codage de texte URL	42
Accès aux données XML via le moteur de publication Web	43
A propos des espaces de nom pour le langage XML de FileMaker	43
A propos des codes d'erreurs des bases de données FileMaker	44
Récupération des définitions de type de document pour les grammaires FileMaker	44
Utilisation de la grammaire fmresultset	44
Description des éléments de la grammaire fmresultset	45
Exemple de données XML dans la grammaire fmresultset	47

Utilisation des grammaires FileMaker XML	48
Description des éléments de la grammaire FMPXMLRESULT	48
Exemple de données XML dans la grammaire FMPXMLRESULT	49
Description des éléments dans la grammaire FMPXMLLAYOUT	49
Exemple de données XML dans la grammaire FMPXMLLAYOUT	51
A propos des données UTF-8	52
Utilisation de chaînes de requête FileMaker pour rechercher des données XML	52
Passage d'un modèle à un autre pour obtenir une réponse XML	54
Comprendre le traitement d'une requête XML	55
Utilisation du traitement des feuilles de style côté serveur et côté client	55
Dépannage de l'accès aux documents XML	56

## Chapitre 6

### **Conception de feuilles de style FileMaker XSLT**

Utilisation des feuilles de style XSLT avec le moteur de publication Web	57
A propos de la base de référence des fonctions d'extension FileMaker XSLT	58
A propos des solutions de démarrage FileMaker XSLT	58
A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT	59
A propos de la syntaxe d'URL relative aux objets multimédias dans les solutions XSLT	60
Utilisation de chaînes de requête dans les feuilles de style FileMaker XSLT	60
Spécification d'une grammaire XML pour une feuille de style FileMaker XSLT	61
A propos des espaces de nom et des préfixes dans les feuilles de style FileMaker XSLT	62
Utilisation de commandes et de paramètres de requête définis de façon statique	62
Définition du codage du texte dans les requêtes	63
Définition d'une méthode de sortie et du codage correspondant	64
A propos du codage des feuilles de style XSLT	65
Traitement des requêtes XSLT n'adressant pas de requêtes à FileMaker Server	65
Utilisation du paramètre –token pour transmettre des informations entre plusieurs feuilles de style	66
Utilisation des fonctions d'extension et des paramètres FileMaker XSLT	66
A propos des paramètres XSLT propres à FileMaker définis par le moteur de publication Web	67
Accès aux demandes d'informations dans une requête	67
Obtention d'informations client	68
Utilisation du paramètre URI de base du moteur de publication Web	68
Utilisation du paramètre URI de base authentifié	69
Chargement de documents supplémentaires	69
Utilisation des informations de modèle d'une base de données dans une feuille de style	70
Utilisation de la fonction de mise du contenu en mémoire tampon	70
Utilisation de sessions du moteur de publication Web pour stocker des informations d'une requête à l'autre	71
Utilisation des fonctions d'extension de session	72
Envoi d'emails à partir du moteur de publication Web	73

Utilisation des fonctions d'en-tête	75
Utilisation des fonctions d'extension de cookie	76
Utilisation des fonctions d'extension de manipulation de chaînes	77
Comparaison de chaînes au moyen d'expressions régulières Perl 5	78
Recherche de valeurs dans une rubrique formatée sous forme de case à cocher	78
Utilisation des fonctions d'extension de date, d'heure et de jour	79
Vérification de l'état d'erreur des fonctions d'extension	83
Utilisation de la fonction de journalisation	83
Utilisation du traitement des langages de script côté serveur	83
Définition d'une fonction d'extension	84
Exemple de fonction d'extension	84

## Chapitre 7

### **Stockage, test et suivi d'un site**

Stockage d'un site de Publication Web personnalisée	87
Test d'un site de Publication Web personnalisée	88
Exemples de feuilles de style permettant de tester les sorties XML	89
Suivi de votre site	90
Utilisation des journaux d'erreurs et d'accès au serveur Web	90
Utilisation du journal d'application du moteur de publication Web	90
Utilisation du journal des erreurs du module de serveur Web	91
Utilisation des journaux d'accès internes du noyau de publication Web	91

## Annexe A

### **Noms corrects utilisés dans les chaînes de requête**

A propos des commandes et des paramètres de requête	93
Consignes d'utilisation des commandes et des paramètres de requête	94
A propos du document Référence des chaînes de requête FileMaker	95
A propos de la syntaxe d'un nom de rubrique entièrement qualifié	95
Utilisation de commandes de requêtes avec des rubriques de table externe	95
A propos de la syntaxe de spécification d'une rubrique de type Global	97
Références des commandes de requête	98
Commande de requête –dbnames (noms des bases de données)	98
Commande de requête –delete (supprimer l'enregistrement)	98
Commande de requête –dup (dupliquer l'enregistrement)	98
Commande de requête –edit (modifier l'enregistrement)	99
Commandes de requête –find, –findall, or –findany (rechercher les enregistrements)	99
–findquery (recherche composée), commande de requête	100
Commande de requête –layoutnames (noms des modèles)	100
Commande de requête –new (nouvel enregistrement)	100
–process (traiter les feuilles de style XSLT)	101
Commande de requête –scriptnames (noms des scripts)	101
Commande de requête –view (affichage des informations de modèle)	101

Références de paramètre de requête	102
Paramètre de requête –db (nom de base de données)	102
–delete.related (suppression d’enregistrements externes), paramètre de requête	102
Paramètre de requête –encoding (codage de la requête XSLT)	102
Paramètre de requête –field (nom de rubrique Multimédia)	102
Paramètre de requête fieldname (nom de rubrique non-multimédia)	103
Paramètre de requête fieldname.op (opérateur de comparaison)	104
Paramètre de requête –grammar (grammaire pour feuilles de style XSLT)	105
Paramètre de requête –lay (modèle)	105
Paramètre de requête –lay.response (changer de modèle pour la réponse)	105
Paramètre de requête –lop (opérateur logique)	105
Paramètres de requête –max (nombre maximal d’enregistrements)	106
Paramètre de requête –modid (numéro d’identification de la modification)	106
–query (requête de recherche composée), commande de requête	106
Paramètre de requête –recid (ID d’enregistrement)	107
–relatedsets.filter (filtrage d’enregistrements de table externe), paramètre de requête	108
–relatedsets.max (limitation d’enregistrements de table externe), paramètre de requête	108
Paramètre de requête –script (Script)	108
Paramètre de requête –script.param (transfert du paramètre au script)	109
Paramètre de requête –script.prefind (script avant une recherche)	109
Paramètre de requête –script.prefind.param (transfert du paramètre au script avant recherche)	110
Paramètre de requête –script.presort (script avant un tri)	110
Paramètre de requête –script.presort.param (transfert du paramètre au script avant recherche)	110
Paramètre de requête –skip (enregistrements ignorés)	111
Paramètre de requête –sortfield (rubrique de tri)	111
Paramètre de requête –sortorder (ordre de tri)	111
Paramètre de requête –stylehref (Style href)	112
Paramètre de requête –styletype (type de style)	112
Paramètre de requête –token.[string] (transfert des valeurs d’une feuille de style XSLT à une autre)	113

## Annexe B

### **Codes d’erreur de la Publication Web personnalisée**

Numéros des codes d’erreur des bases de données FileMaker :	115
Numéros des codes d’erreur du moteur de publication Web	122
Numéros des codes d’erreur des fonctions d’extension XSLT de FileMaker	123

<b>Index</b>	125
--------------	-----





# Préface

## Présentation de ce guide

Ce guide part de l'hypothèse que vous avez déjà une certaine expérience dans le domaine des langages XML et XSLT, du développement de sites Web et de l'utilisation de FileMaker® Pro pour la création de bases de données. Vous devez comprendre les bases de la conception des bases de données avec FileMaker Pro, ainsi que les concepts de rubriques, de liens, de modèles, de tables externes et d'objets multimédias. Concernant la Publication Web personnalisée avec les langages XML et XSLT sous FileMaker Server, ce guide donne des informations sur :

- les éléments nécessaires au développement d'une solution de publication Web personnalisée à l'aide des langages XML ou XSLT ;
- la façon de publier des bases de données à l'aide des langages XML ou XSLT ;
- les éléments que les utilisateurs Web doivent employer pour accéder à une solution de publication Web personnalisée ;
- la façon d'obtenir des données XML à partir de bases de données hébergées par FileMaker Server ;
- la façon de développer des feuilles de style FileMaker XSLT ;

**Important** Vous pouvez télécharger la version PDF de la documentation de FileMaker à l'adresse suivante : <http://www.filemaker.fr/support/product/documentation.html>. Toutes les mises à jour de ce document sont également disponibles sur le site Web.

La documentation de FileMaker Server inclut les informations suivantes :

Pour plus d'informations sur	Consultez le document
Installation et configuration de FileMaker Server	<i>Guide de démarrage de FileMaker Server</i> <i>Aide FileMaker Server</i>
Publication Web instantanée	<i>Guide de la Publication Web instantanée FileMaker</i>
Publication Web personnalisée avec PHP	<i>Publication Web personnalisée de FileMaker Server avec PHP</i>
Utilisation de PHP Site Assistant	<i>Aide de PHP Site Assistant</i>
Publication Web personnalisée avec XML et XSLT	<i>Publication Web personnalisée FileMaker Server avec XML et XSLT</i> (ce manuel)
Utilisation de XSLT Site Assistant	<i>Aide de XSLT Site Assistant</i>
Installation et paramétrage des pilotes ODBC et JDBC, et utilisation d'ODBJ et de JDBC	<i>Guide ODBC et JDBC FileMaker</i>
Explique comment FileMaker Server Auto Update peut télécharger le plug-in le plus récent sur les ordinateurs clients de la base de données FileMaker Pro.	<i>Guide FileMaker de la mise à jour des plug-ins</i>



# Chapitre 1

## ***Introduction à la Publication Web personnalisée***

FileMaker Server vous permet de publier vos bases de données FileMaker Server sur Internet ou sur un intranet. Vous pouvez procéder de différentes manières :

**Publication Web instantanée :** Avec la Publication Web instantanée, vous avez la possibilité de publier rapidement et simplement votre base de données sur le Web. Aucune modification de vos fichiers de base de données ou installation de logiciels supplémentaires n'est nécessaire. Tout utilisateur d'Internet possédant un navigateur Web compatible et disposant d'un accès à Internet ou à un intranet pourra se connecter à votre base de données pour consulter, éditer, trier ou chercher des enregistrements, à condition que vous lui donniez un privilège d'accès.

Avec la Publication Web instantanée, l'ordinateur hôte doit travailler sous FileMaker Pro ou FileMaker Server. L'interface utilisateur ressemble à la version bureau de l'application FileMaker Pro. Les pages web et les formulaires sur lesquels l'utilisateur agit dépendent des modèles et des affichages définis dans la base de données FileMaker Pro. Pour plus d'informations, consultez le Guide de la publication Web instantanée FileMaker.

**Publication statique :** Si vous ne modifiez pas souvent vos données ou que vous ne souhaitez pas que les utilisateurs puissent se connecter directement à votre base de données, vous pouvez faire appel à la publication statique. Avec cette dernière, vous exportez les données d'une base de données FileMaker Pro pour créer une page Web, que vous pouvez ensuite personnaliser à l'aide du langage HTML. Ainsi, la page n'est pas modifiée lorsque des données de la base le sont et les utilisateurs ne se connectent pas directement à votre base. (Avec la fonction de Publication Web instantanée, les données sont mises à jour dans la fenêtre du navigateur à chaque fois que ce dernier transmet une requête à FileMaker Server). Pour plus d'informations, consultez le *Guide de la publication Web instantanée FileMaker*.

**Publication Web personnalisée :** Si vous souhaitez disposer d'un plus grand choix dans la définition de la présentation d'une base de données publiée et des fonctions qu'elle propose, utilisez les technologies de Publication Web personnalisée de FileMaker Server. Pour FileMaker Server, qui héberge les bases de données publiées, il n'est pas nécessaire d'installer ni d'exécuter FileMaker Pro pour que la Publication Web personnalisée soit disponible.

La Publication Web personnalisée vous permet :

- d'intégrer votre base de données à un autre site Web ;
- de déterminer la façon dont les utilisateurs interagissent avec les données ;
- de contrôler l'affichage des données dans les navigateurs Web.

FileMaker Server propose deux technologies de Publication Web personnalisée :

- Publication Web personnalisée avec PHP : Utilisez l'API FileMaker pour PHP, qui propose une interface PHP orientée objet pour les bases de données FileMaker Pro, ce qui permet d'intégrer vos données FileMaker dans une application web PHP. Vous pouvez employer PHP Site Assistant pour générer un site Web PHP complet, ou coder les pages web en PHP vous-même.

- Publication Web personnalisée avec XML et XSLT :
  - Utilisez la publication de données XML pour échanger des données FileMaker avec d'autres sites Web et applications.
  - Utilisez les feuilles de style XSLT traitées par le serveur pour intégrer des données FileMaker dans d'autres sites Web et dans d'autres applications personnalisées ou de type Middleware. Vous pouvez utiliser XSLT Site Assistant pour générer des feuilles de style XSLT, ou coder vos feuilles de style vous-même.

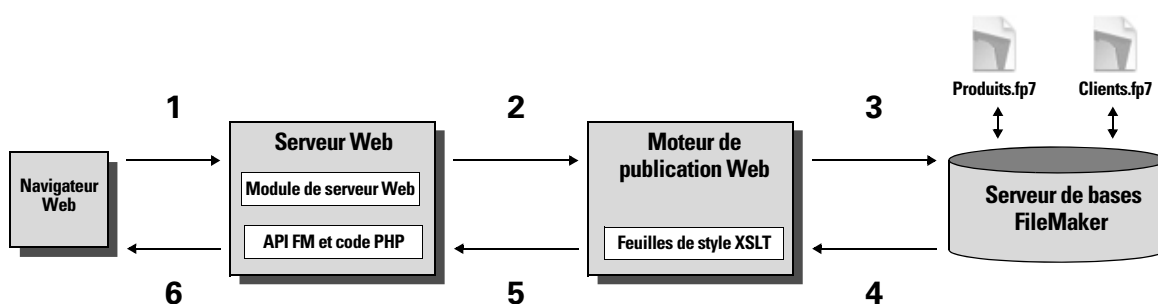
## A propos du moteur de publication Web

Pour prendre en charge la Publication Web instantanée ou la Publication Web personnalisée, FileMaker Server utilise un jeu de composants logiciels appelé *Moteur de publication Web FileMaker Server*. Le moteur de publication Web gère les interactions entre le navigateur Web de l'utilisateur, votre serveur Web et FileMaker Server.

**Publication Web personnalisée avec XML et XSLT :** Le moteur de publication Web fonctionne comme un processeur XSLT, il génère des sorties au format HTML, XML ou texte (vCards par exemple) sur le serveur Web, qui génère à son tour des sorties vers le navigateur Web. Les utilisateurs Web accèdent à votre solution de Publication Web personnalisée en cliquant sur un lien HREF ou en saisissant un URL (Uniform Resource Locator) qui indique l'adresse du serveur Web, ainsi qu'une requête qui contient la chaîne de requête FileMaker. L'URL peut accéder à des données XML ou référencer une feuille de style XSLT. Le moteur de publication Web renvoie soit les données XML indiquées dans la requête sur la chaîne de requête, soit les résultats de la feuille de style XSLT référencée.

**Publication Web personnalisée avec PHP :** Lorsqu'un utilisateur Web accède à votre solution de Publication Web personnalisée, PHP, sur FileMaker Server, se connecte avec le moteur de publication Web et répond via l'API de FileMaker pour PHP.

Utilisation du moteur de publication Web de FileMaker Server pour la Publication Web personnalisée



## Traitement d'une requête du moteur de publication Web

1. Une requête est envoyée par un navigateur ou une application au serveur web.
2. Le serveur achemine la requête par le module serveur Web de FileMaker jusqu'au moteur de publication Web.
3. Le moteur de publication Web recherche les données dans la base de données hébergée par le serveur de bases de données.

4. FileMaker Server envoie les données FileMaker demandées au moteur de publication Web.
5. Le moteur de publication Web convertit les données FileMaker pour répondre à la requête.
  - Pour les requêtes PHP, le moteur de publication Web répond à la requête API.
  - Pour les requêtes XML, le moteur de publication Web envoie directement les données XML au serveur Web.
  - Pour les requêtes XSLT, le moteur de publication Web utilise une feuille de style XSLT pour formater ou transformer les données XML, puis génère une sortie sous forme de pages HTML, de document XML ou de texte qu'il envoie au serveur Web.
6. Le serveur Web envoie le résultat au navigateur web ou au programme demandeur.

**Important** La sécurité est un aspect important de la publication de données sur le Web. Consultez les règles de sécurité dans le *Guide de la sécurité FileMaker Pro*, disponible au format PDF à l'adresse <http://www.filemaker.fr/support/product/documentation.html>.

## Publication Web personnalisée avec PHP

L'API FileMaker pour PHP propose une interface PHP orientée objet aux bases de données FileMaker. L'API de FileMaker pour PHP permet l'accès aux données et logiques stockées dans une base de données FileMaker Pro et leur publication sur le Web, ainsi que leur export vers d'autres applications. L'API gère également les commandes de recherche complexes et composées permettant l'extraction et le filtrage des informations stockées dans des bases de données FileMaker Pro.

Conçu à l'origine comme un langage de programmation procédural, PHP est devenu un langage de développement Web orienté objet. PHP propose des fonctionnalités de langage de programmation permettant de construire n'importe quel type de logique dans une page de site. Par exemple, vous pouvez utiliser les constructions logiques conditionnelles pour contrôler la génération de page, l'acheminement des données ou un déroulement d'opérations. PHP permet également l'administration du site et la sécurité.

En outre, vous pouvez employer FileMaker PHP Site Assistant pour créer du code PHP contenant toutes les conditions préalables et fonctions nécessaires pour accéder correctement aux informations de la base de données FileMaker Pro. PHP Site Assistant génère un site web de plusieurs pages qui permet aux utilisateurs Web d'interroger une base de données, d'afficher une liste d'enregistrements, de parcourir, trier, ajouter, modifier, dupliquer ou supprimer des enregistrements, et aussi d'afficher un rapport récapitulatif. Les développeurs FileMaker connaissant un peu PHP peuvent employer PHP Site Assistant pour générer un site PHP complet. Les développeurs PHP connaissant un peu FileMaker peuvent employer PHP Site Assistant pour comprendre l'API FileMaker pour les objets et méthodes PHP.

## Publication Web personnalisée avec XML et XSLT

La Publication Web personnalisée avec XML vous permet d'envoyer des demandes de requêtes à une base de données FileMaker Pro hébergée par FileMaker Server, et d'afficher, modifier ou manipuler les résultats. L'utilisation d'une requête HTTP avec les commandes et les paramètres appropriés vous aide à récupérer les données FileMaker sous la forme d'un document XML. Vous pouvez ensuite exporter ces données XML vers d'autres applications ou décider d'y appliquer une feuille de style XSLT.

La Publication Web personnalisée avec XSLT vous permet de transformer, filtrer ou formater des données XML pour les navigateurs Web ou autres applications. Vous pouvez effectuer les actions suivantes :

- utiliser une feuille de style XSLT pour transformer les données développées dans une grammaire XML FileMaker en données développées dans une autre grammaire XML, et les utiliser dans d'autres applications ou bases de données ;
- filtrer les données contrôlant les rubriques de base de données publiées par la feuille de style ;
- formater des données présentées dans une page Web et le contrôle de l'interaction entre l'utilisateur Web et les données.

Le moteur de publication Web utilise vos feuilles de style pour obtenir des données d'une base de données FileMaker, à chaque fois qu'un utilisateur Web envoie une requête HTTP et une adresse URL qui fait référence à l'une de vos feuilles de style XSLT. Le moteur de publication Web utilise une feuille de style pour convertir et formater les données XML, puis génère la page HTML correspondante manipulable par l'utilisateur Web.

En outre, vous pouvez employer FileMaker XSLT Site Assistant pour créer des feuilles de style XSLT de base. Ces feuilles servent de point de départ à une publication Web personnalisée utilisant le langage XSLT. XSLT Site Assistant génère des feuilles de style pour des pages permettant d'effectuer des recherches dans la base de données, de parcourir les enregistrements un par un, de lister, ajouter, modifier, dupliquer ou supprimer des enregistrements, et aussi d'afficher un rapport récapitulatif.

## Comparaison de PHP avec XML et XSLT

La section suivante propose quelques grandes lignes permettant de déterminer la meilleure solution pour votre site.

### Raisons de choisir PHP

- PHP est un langage de script procédural orienté objet plus puissant, mais son apprentissage est relativement simple. Des nombreuses ressources sont disponibles pour la formation, le développement et l'assistance.
- L'API de FileMaker pour PHP permet l'accès aux données et logiques stockées dans une base de données FileMaker Pro et leur publication sur le Web, ainsi que leur export vers d'autres applications.
- PHP permet d'utiliser des logiques conditionnelles pour contrôler la construction de la page ou les flux.
- PHP propose des fonctionnalités de langage de programmation permettant de construire n'importe quel type de logique dans une page de site.
- PHP est l'un des langages de script les plus populaires du web.
- PHP est un langage libre ; il est disponible sur <http://php.net>.
- PHP permet l'accès à une grande variété de modules tiers que vous pouvez intégrer dans vos solutions.

**Remarque** Pour plus d'informations sur la publication Web personnalisée avec PHP, consultez la section *Publication Web personnalisée FileMaker Server avec PHP*.

### Raisons de choisir XML et XSLT

- La syntaxe des paramètres d'interrogation XML de FileMaker est conçue pour l'interaction avec les bases de données, ce qui simplifie le développement de solutions.
- XML et XSLT sont des normes W3C.

- XML est un format lisible par les hommes et les machines. Il gère Unicode, ce qui permet la communication des données dans n'importe quelle langue écrite.
- XML est bien adapté à la présentation d'enregistrements, de listes et de données structurées en liste.
- XSLT permet de transformer les sorties XML en documents de texte structurés tels que RSS, RTF, vCard.
- Vous pouvez employer XSLT pour transformer une sortie XML d'une grammaire dans une autre.
- Les modèles facilitent l'application d'un formatage conditionnel aux données variables.
- Vous pouvez employer les feuilles de style de type FMPXMLRESULT pour la Publication Web personnalisée et pour l'export XML à partir de bases de données FileMaker Pro.
- FileMaker Server gère le traitement des feuilles de style FileMaker XSLT, ce qui empêche l'accès non autorisé aux données, qui serait sinon possible avec les feuilles de style côté client.





# Chapitre 2

## ***A propos de la publication Web personnalisée avec XML et XSLT***

### **Création de sites Web dynamiques avec le moteur de publication Web**

Le moteur de publication Web dote FileMaker Server de la Publication Web personnalisée en utilisant la publication de données XML et les feuilles de style XSLT traitées par le serveur. La Publication Web personnalisée offre divers avantages :

- **Personnalisation** : Vous pouvez déterminer comment les utilisateurs Web interagissent avec les données FileMaker et comment les données s'affichent dans les navigateurs Web.
- **Echange de données** : L'utilisation du format XML de FileMaker vous permet d'échanger des données FileMaker avec d'autres sites Web et d'autres applications.
- **Intégration des données** : L'utilisation des feuilles de style FileMaker XSLT vous permet d'intégrer des données FileMaker dans d'autres sites Web et dans d'autres applications personnalisées ou de type Middleware. Vous pouvez faire en sorte que les données s'affichent comme si elles provenaient d'un autre site Web au lieu d'afficher un modèle FileMaker complet dans le navigateur Web.
- **Sécurité** : l'administrateur FileMaker Server peut activer ou désactiver individuellement les fonctions de publication Web instantanée, de publication Web XML ou de publication Web XSLT pour toutes les bases de données hébergées sur le serveur. En tant que propriétaire de la base de données FileMaker, vous pouvez contrôler l'accès des utilisateurs aux fonctions de publication Web instantanée, de publication Web XML ou de publication Web XSLT pour chaque base de données.
- **Feuilles de style côté serveur** : le traitement des feuilles de style XSLT côté serveur empêche la consultation non autorisée des données contenues dans les bases de données confidentielles, consultation qui serait sinon possible avec les feuilles de style côté client.
- **Contrôle et filtrage des données publiées** : en utilisant les feuilles de style XSLT, vous contrôlez et vous filtrez les données et le type des données contenues dans la base de données que vous souhaitez publier tout en empêchant l'utilisation non autorisée de la base de données. Vous pouvez également masquer les métadonnées, comme par exemple les noms de la base de données et de la rubrique.
- **Utilisation de normes ouvertes** : les outils, les ressources et le personnel compétent auxquels vous avez accès pour les solutions de publication Web personnalisée sont plus importants. Si vous connaissez déjà les langages XML et XSLT standard, vous pouvez vous lancer immédiatement dans le développement de solutions. Vous devez néanmoins connaître la syntaxe d'URL et les paramètres de requête à employer pour utiliser la Publication Web personnalisée avec XML.

### **A propos de la fonction de publication Web personnalisée utilisant le langage XML**

La publication Web personnalisée avec XML vous permet de récupérer des données dans les bases de données FileMaker et de les utiliser facilement dans d'autres formats d'arrivée. L'utilisation d'une requête HTTP avec les commandes et les paramètres appropriés vous aide à récupérer les données FileMaker sous la forme d'un document XML. Vous pouvez ensuite utiliser ces données XML dans d'autres applications ou décider d'y appliquer une feuille de style XSLT. Consultez la section chapitre 5, « Accès aux données XML avec le moteur de publication Web ».

## A propos de la fonction de publication Web personnalisée utilisant le langage XSLT

La fonction de publication Web personnalisée utilisant le langage XSLT vous permet de transformer, de filtrer ou de formater des données XML à utiliser dans un navigateur Web ou dans d'autres applications. Faites appel à une feuille de style XSLT pour transformer les données développées dans une grammaire XML FileMaker en données développées dans une autre grammaire XML, et utilisez-les dans une autre application ou dans une autre base de données. Vous pouvez filtrer les données en contrôlant les rubriques des bases de données à publier sur la feuille de style. Vous pouvez également formater la présentation des données sur une page Web, de même que contrôler la façon dont l'utilisateur Web interagit avec les données. Consultez la section chapitre 4, « Introduction à la Publication Web personnalisée avec XSLT ».

Le moteur de publication Web utilise vos feuilles de style pour obtenir des données d'une base de données FileMaker de façon dynamique, à chaque fois qu'un utilisateur Web envoie une requête HTTP et une adresse URL qui fait référence à l'une de vos feuilles de style XSLT. Le moteur de publication Web utilise une feuille de style pour convertir et formater les données XML, puis génère la page HTML correspondante manipulable par l'utilisateur Web.

Pour obtenir des informations supplémentaires sur l'utilisation de la publication Web personnalisée de FileMaker Server avec XML et XSLT, rendez-vous sur le site <http://www.filemaker.fr/support/product/documentation.html>.

## A propos du développement des feuilles de style XSLT

FileMaker Server propose un outil de développement des feuilles de style XSLT. FileMaker XSLT Site Assistant est une application utilisée pour créer des feuilles de style XSLT de base. Ces feuilles servent de point de départ à une publication Web personnalisée utilisant le langage XSLT. L'outil XSLT Site Assistant vous aide à mieux appréhender la structure des feuilles de style FileMaker XSLT. N'hésitez pas à utiliser vos propres outils de composition de feuilles de style XSLT pour modifier les feuilles de style, si nécessaire. Consultez la section « Utilisation de l'outil FileMaker XSLT Site Assistant pour générer des feuilles de style FileMaker XSLT », page 32.

**Remarque** FileMaker Server prend en charge XSLT 1.0 selon les normes définies par le World Wide Web Consortium. Les outils de création XSLT que vous utilisez doivent produire des résultats conformes à la norme XSLT 1.0.

## Principales fonctionnalités de la fonction de publication Web personnalisée utilisant les langages XML et XSLT

La Publication Web personnalisée FileMaker Server avec XML et XSLT fournit plusieurs nouvelles fonctionnalités importantes :

- Les bases de données sont hébergées sur FileMaker Server et l'application FileMaker Pro ne doit pas forcément être lancée.
- Vous pouvez faire appel au traitement des feuilles de style XSLT côté serveur, qui se révèle plus sécurisé que le traitement des feuilles de style côté client.
- Vous pouvez utiliser le traitement JavaScript côté serveur dans les feuilles de style XSLT. Pour plus d'informations, consultez la section « Utilisation du traitement des langages de script côté serveur », page 83.

- Vous pouvez prévenir l'utilisation non autorisée des commandes et des paramètres de requête avec votre feuille de style XSLT FileMaker en définissant de manière statique les commandes, les paramètres et les valeurs des requêtes à utiliser au moment de rechercher des données XML. Consultez la section « Utilisation de commandes et de paramètres de requête définis de façon statique », page 62.
- Comme dans FileMaker Pro, l'accès aux données, aux modèles et aux rubriques est basé sur les paramètres de compte d'utilisateur définis dans les privilèges d'accès de la base de données. Le moteur de publication Web présente également plusieurs autres améliorations en matière de sécurité. Consultez la section « Protection de vos bases de données publiées », page 22.
- Les utilisateurs Web peuvent exécuter des scripts complexes contenant plusieurs actions. FileMaker prend en charge près de 70 actions de script dans la Publication Web personnalisée. Consultez la section « Scripts FileMaker et Publication Web personnalisée », page 24.
- Vous pouvez transmettre une valeur de paramètre à un script FileMaker. Pour plus d'informations, consultez les sections « Paramètre de requête –script.param (transfert du paramètre au script) », page 109, « Paramètre de requête –script.prefind.param (transfert du paramètre au script avant recherche) », page 110 et « Paramètre de requête –script.presort.param (transfert du paramètre au script avant recherche) », page 110.
- La grammaire XML `fmresultset` vous permet d'accéder aux rubriques par leur nom et de manipuler les données `relatedset` (table externe).
- L'utilisation des fonctions de session dans une feuille de style XSLT vous permet de stocker des informations et des transactions relatives à l'utilisateur Web dans des sessions gérées sur le serveur.
- Pour accéder aux données d'une base de données, vous devez indiquer un modèle. Consultez la section annexe A, « Noms corrects utilisés dans les chaînes de requête ».
- Chaque utilisateur Web possède une valeur de rubrique de type Global unique qui reste valide tant que la session est active. Pour plus d'informations générales sur les rubriques de type Global, consultez l'Aide de FileMaker Pro. Pour plus d'informations sur l'utilisation des rubriques de type Global avec la Publication Web personnalisée, consultez la section « A propos de la syntaxe de spécification d'une rubrique de type Global », page 97.

## Configuration pour la publication Web

### Configuration requise pour la publication d'une base de données à l'aide de la Publication Web personnalisée

Pour publier des bases de données à l'aide de la fonction de publication Web personnalisée avec XML et XSLT, il vous faut :

- un déploiement FileMaker Server comportant ;
  - un serveur Web, Microsoft IIS (Windows) ou Apache (Mac OS X) ;
  - le serveur de bases de données FileMaker, avec option Publication Web personnalisée activée ;
  - le moteur de publication Web installé et configuré ;
- une ou plusieurs bases de données FileMaker Pro hébergées par FileMaker Server ;
- l'adresse IP ou le nom de domaine de l'hôte exécutant le serveur Web.

- Un navigateur Web et un accès au serveur Web pour développer et tester la solution de publication Web personnalisée.

Pour plus d'informations, consultez le *Guide de démarrage de FileMaker Server*.

## Éléments à utiliser par les utilisateurs Web pour accéder à une solution de publication Web personnalisée

Pour accéder à une solution de publication Web personnalisée utilisant les langages XML ou XSLT, les utilisateurs Web doivent posséder :

- un navigateur Web ;
- un accès à Internet ou à un intranet, ainsi qu'un accès au serveur Web ;
- l'adresse IP ou le nom de domaine de l'hôte exécutant le serveur Web.

Si la base de données est protégée par un mot de passe, les utilisateurs Web doivent également saisir un nom d'utilisateur et un mot de passe pour accéder au compte de la base de données.

## Connexion à Internet ou à un intranet

Lorsque vous publiez des bases de données sur Internet ou sur un intranet, l'ordinateur hôte doit exécuter FileMaker Server et les bases de données que vous souhaitez partager doivent être hébergées et accessibles. En outre :

- Publiez votre base de données sur un ordinateur possédant une connexion permanente à Internet ou à un intranet. En effet, si la connexion n'est pas permanente, les utilisateurs Web doivent attendre que l'hôte se connecte à Internet ou à un intranet pour accéder à la base de données.
- L'ordinateur hôte du serveur Web, partie intégrante du déploiement FileMaker Server, doit posséder une adresse IP *statique* (permanente) dédiée ou un nom de domaine. Si vous vous connectez à Internet par le biais d'un fournisseur d'accès Internet (FAI), votre adresse IP est généralement *attribuée de façon dynamique* (elle est donc différente à chaque fois que vous vous connectez). Le cas échéant, les utilisateurs Web auront plus de difficultés à localiser une base de données. Si vous ne savez pas de quel type d'accès vous disposez, contactez votre fournisseur d'accès à Internet ou l'administrateur du réseau.

## Étape suivante

Voici quelques suggestions pour commencer à développer des solutions de publication Web personnalisée :

- Si ce n'est pas déjà fait, utilisez l'Admin Console de FileMaker Server pour activer la publication Web personnalisée. Reportez-vous à l'aide de FileMaker Server et au *Guide de démarrage de FileMaker Server*.
- Dans FileMaker Pro, ouvrez les bases de données FileMaker à publier et assurez-vous que chacune d'entre elles dispose du ou des privilèges d'accès étendus pour la Publication Web personnalisée. Consultez la section « Activation de la Publication Web personnalisée dans une base de données », page 21.
- Pour savoir comment accéder aux données dans les bases de données FileMaker en utilisant le langage XML, consultez le chapitre 5, « Accès aux données XML avec le moteur de publication Web ».
- Pour savoir comment vous lancer dans le développement de feuilles de style XSLT FileMaker, consultez le chapitre 4, « Introduction à la Publication Web personnalisée avec XSLT ».

# **Chapitre 3**

## ***Préparation des bases de données pour la Publication Web personnalisée***

Avant d'utiliser la Publication Web personnalisée avec une base de données, vous devez préparer la base de données et la protéger contre les accès non autorisés.

### **Activation de la Publication Web personnalisée dans une base de données**

Vous devez activer la Publication Web personnalisée dans chaque base de données que vous souhaitez publier. Vous pouvez activer individuellement la fonction de Publication Web personnalisée utilisant le langage XML, la fonction de publication Web personnalisée utilisant le langage XSLT, ou bien les deux technologies dans chaque base de données. Si vous n'activez aucune de ces technologies dans la base de données, les utilisateurs Web ne pourront pas utiliser la Publication Web personnalisée pour accéder à la base de données, même si celle-ci est hébergée par un serveur FileMaker Server configuré pour prendre en charge un moteur de publication Web.

Pour activer la Publication Web personnalisée pour une base de données :

1. Dans FileMaker Pro, ouvrez la base de données que vous souhaitez publier en utilisant un compte bénéficiant du jeu de privilèges Accès intégral. Une autre possibilité consiste à ouvrir la base de données en utilisant un compte bénéficiant des privilèges d'accès Gérer les autorisations étendues.
2. Affectez l'un de ces privilèges étendus ou les deux à l'un ou plusieurs des jeux de privilèges :
  - Pour autoriser la fonction de publication Web personnalisée utilisant le langage XML, le mot-clé est le suivant : `fmxml`
  - Pour autoriser la fonction de publication Web personnalisée utilisant le langage XSLT, le mot-clé est le suivant : `fmxslt`

Depuis la version 8 de FileMaker Pro, les mots-clés `fmxml` et `fmxslt` sont définis pour vous dans l'onglet Privilèges étendus.

3. Affectez le ou les jeux de privilèges d'accès comprenant les privilèges d'accès étendus pour la Publication Web personnalisée à un ou plusieurs comptes, ou au compte Admin ou Invité.

**Remarque** Au moment de définir les noms des comptes et les mots de passe relatifs aux solutions de publication Web personnalisée, utilisez les caractères ASCII imprimables, par exemple a-z, A-Z et 0-9. Pour sécuriser davantage encore les noms des comptes et les mots de passe, intégrez des caractères de ponctuation du type « ! » et « % » mais n'utilisez pas les deux points. Pour obtenir des informations sur la configuration de comptes, consultez l'aide FileMaker Pro.

## Accès à une base de données protégée

Lorsqu'ils utilisent une solution de publication Web personnalisée pour accéder à une base de données, les utilisateurs Web peuvent être invités à spécifier leurs informations de compte. Si le compte Invité de la base de données est désactivé ou ne possède aucun jeu de privilèges d'accès activé comprenant un privilège d'accès étendu pour la Publication Web personnalisée, le moteur de publication Web utilise une authentification de base HTTP pour demander aux utilisateurs Web de s'authentifier. Le navigateur de l'utilisateur Web affiche la boîte de dialogue d'authentification de base HTTP pour permettre à l'utilisateur de saisir le nom d'utilisateur et le mot de passe d'un compte bénéficiant d'un privilège d'accès étendu pour la Publication Web personnalisée.

La liste suivante résume le processus qui se produit lorsqu'un utilisateur web utilise une solution de Publication Web personnalisée pour accéder à une base de données :

- Si vous n'avez pas défini de mot de passe pour un compte, les utilisateurs Web devront uniquement spécifier le nom de compte.
- Si le compte Invité est désactivé, les utilisateurs sont invités à donner un nom de compte et un mot de passe lorsqu'ils accèdent à une base de données. Un privilège d'accès étendu pour la Publication Web personnalisée doit être activé pour ce compte.
- Si le compte Invité est activé et bénéficie d'un jeu de privilèges comprenant un privilège d'accès étendu pour la Publication Web personnalisée, tous les utilisateurs Web peuvent automatiquement ouvrir la base de données avec les privilèges d'accès du compte Invité. Si le privilège étendu de Publication Web personnalisée est attribué au compte Invité :
  - Les utilisateurs Web n'ont pas besoin de spécifier de nom de compte ni de mot de passe lorsqu'ils ouvrent un fichier.
  - Tous les utilisateurs Web se connectent automatiquement avec le compte Invité et héritent des privilèges d'accès de celui-ci. Vous pouvez permettre aux utilisateurs de changer de compte de connexion via leur navigateur Web grâce à la commande de script Reconnexion (par exemple, pour basculer du compte Invité à un autre compte, jouissant de privilèges plus étendus).
  - Les privilèges par défaut des comptes Invité fournissent un accès en lecture seule. Vous pouvez modifier les privilèges par défaut de ce compte, notamment les privilèges étendus. Consultez l'aide FileMaker Pro.

**Remarque** Par défaut, les utilisateurs Web ne peuvent pas modifier le mot de passe de leur compte depuis un navigateur Web. Vous pouvez intégrer cette fonction dans une base de données grâce à l'action de script Modifier le mot de passe, qui permet aux utilisateurs Web de changer leur mot de passe depuis leur navigateur. Consultez l'aide FileMaker Pro.

## Protection de vos bases de données publiées

La fonction de publication Web personnalisée utilisant les langages XML ou XSLT vous permet de limiter l'accès à vos bases de données publiées.

- Affectez des mots de passe aux comptes de base de données utilisés pour la Publication Web personnalisée.
- Activez uniquement la fonction de publication Web personnalisée utilisant les langages XML ou XSLT dans les jeux de privilèges d'accès des comptes que vous souhaitez autoriser à accéder à vos bases de données publiées.

- Pour activer ou désactiver une technologie de Publication Web personnalisée pour une base de données individuelle, définissez le privilège d'accès étendu.
- Pour activer ou désactiver une technologie de Publication Web personnalisée pour toutes les solutions de Publication Web personnalisée dans le moteur de publication Web, utilisez l'Admin Console de FileMaker Server. Consultez l'Aide de FileMaker Server.
- Configurez votre serveur Web de façon à restreindre les adresses IP ayant accès à vos bases de données via le moteur de publication Web. Vous pouvez par exemple indiquer que seuls les utilisateurs Web utilisant l'adresse IP 192.168.100.101 sont autorisés à accéder à vos bases de données. Pour plus d'informations sur la restriction d'adresses IP, consultez la documentation de votre serveur Web.
- Utilisez le cryptage Secure Socket Layer (SSL) pour les communications entre votre serveur Web et les navigateurs des utilisateurs Web. Le cryptage SSL convertit les informations échangées entre les serveurs et les clients en informations incompréhensibles, grâce à l'utilisation de formules mathématiques appelées algorithmes de cryptage. Ces algorithmes sont utilisés pour retransformer les informations en données lisibles via des clés de cryptage. Pour plus d'informations sur l'activation et la configuration de SSL, consultez la documentation de votre serveur Web.

Pour plus d'informations sur la protection de votre base de données, reportez-vous au *Guide de l'utilisateur de FileMaker Pro*, disponible au format PDF à l'adresse <http://www.filemaker.fr/support/product/documentation.html>.

## Prise en charge du serveur Web pour les types de média Internet (MIME)

Votre serveur Web détermine la prise en charge des types MIME (Multipurpose Internet Mail Extensions) actuellement enregistrés pour Internet. Le moteur de publication Web ne change pas la prise en charge MIME du serveur Web. Pour plus d'informations, consultez la documentation de votre serveur Web.

## A propos de la publication du contenu des rubriques Multimédia sur le Web

Le contenu d'une rubrique Multimédia, tel qu'un fichier image, peut être stocké au sein d'une base de données FileMaker ou sous la forme d'une référence externe par le biais d'un chemin d'accès relatif.

**Remarque** Le moteur de publication Web ne prend pas en charge la lecture en direct de fichiers vidéo. Les utilisateurs Web doivent télécharger l'intégralité d'un fichier vidéo avant de pouvoir le visualiser.

## Publication d'objets de rubrique Multimédia stockés dans une base de données

Si une rubrique Multimédia stocke les fichiers eux-mêmes dans la base de données FileMaker, toute action sur le contenu de la rubrique Multimédia est alors inutile si le fichier de base de données est correctement hébergé et facile d'accès sur FileMaker Server. Consultez les rubriques « A propos de la syntaxe d'URL pour les objets FileMaker de type Multimédia dans les solutions XML », page 42, et « A propos de la syntaxe d'URL relative aux objets multimédias dans les solutions XSLT », page 60.



## Publication d'objets de rubrique Multimédia stockés comme référence externe

Si une rubrique multimédia stocke des références externes en lieu et place des fichiers eux-mêmes, procédez ainsi pour publier les objets de rubrique multimédia.

**Remarque** Toutes les séquences QuickTime sont stockées dans une rubrique Multimédia sous forme de référence.

Pour publier des objets de rubrique Multimédia stockés comme une référence externe :

1. Stockez les fichiers d'objet Multimédia dans le sous-dossier **Web** du dossier FileMaker Pro.
2. Dans FileMaker Pro, insérez les objets dans la rubrique Multimédia et sélectionnez l'option **Stocker uniquement la référence au fichier**.
3. Copiez ou déplacez les fichiers d'objet référencé du sous-dossier **Web** vers le même emplacement relatif, dans le dossier racine du logiciel de serveur.
  - Pour IIS, déplacez les fichiers vers : `<lecteur>\inetpub\wwwroot`
  - Pour Apache, déplacez les fichiers vers : `/Bibliothèque/WebServer/Documents`

**Remarque** Pour les objets Multimédia stockés sous forme de référence externe, votre serveur Web doit être configuré pour prendre en charge les types MIME correspondant aux types de fichiers que vous souhaitez publier, par exemple des vidéos. Pour plus d'informations, consultez la documentation de votre serveur Web.

## Affichage des données d'une rubrique Multimédia par les utilisateurs Web

Lorsque vous publiez une base de données sur le Web à l'aide du moteur de publication Web, les utilisateurs Web peuvent travailler avec les données des rubriques Multimédia de façon limitée, comme suit :

- Les utilisateurs Web ne peuvent pas écouter les sons ou afficher les objets OLE dans une rubrique Multimédia. Une image apparaît à leur place.
- Les utilisateurs Web ne peuvent pas modifier ni compléter le contenu des rubriques Multimédia. Les utilisateurs Web ne peuvent pas utiliser une rubrique Multimédia pour télécharger des données vers la base de données.
- Si votre base de données contient des images qui ne sont pas au format GIF ou JPEG, le moteur de publication Web crée une image JPEG temporaire quand les données de l'image sont requises par un navigateur Web.

## Scripts FileMaker et Publication Web personnalisée

La fonction ScriptMaker de FileMaker Pro peut automatiser les tâches fréquentes ou combiner plusieurs actions. Lorsque les scripts FileMaker sont utilisés avec la Publication Web personnalisée, ils permettent aux utilisateurs Web d'exécuter plusieurs tâches ou une série de tâches.

FileMaker prend en charge plus de 75 actions de script dans la publication Web personnalisée. Les utilisateurs Web peuvent exécuter toute une série de tâches automatisées pendant que vous utilisez des scripts dans une chaîne de requête pour une adresse URL ou dans une instruction de traitement `<?xslt-cwp-query?>`, dans une feuille de style XSLT. Pour visualiser les actions de script non prises en charge, sélectionnez **Publication Web** dans la liste **Afficher la compatibilité** de la fenêtre **Modifier le script** de FileMaker Pro. Les actions de script grisées ne sont pas prises en charge sur le Web. Pour obtenir des informations sur la création de scripts, consultez l'aide de FileMaker Pro.



## Astuces et considérations à propos des scripts

Bien qu'un grand nombre d'actions de script fonctionnent de manière identique sur le Web, plusieurs fonctionnent différemment. Consultez la section « Comportement des scripts dans les solutions de Publication Web personnalisée », page 26. Avant de partager votre base de données, testez tous les scripts qui seront exécutés depuis un navigateur Web. Veillez à vous connecter avec différents comptes d'utilisateur, afin de vous assurer qu'ils fonctionnent comme prévu pour tous les clients. Consultez le fichier journal de l'application moteur de publication Web (`pe_application_log.txt`) pour rechercher les éventuelles erreurs liées aux scripts ; pour plus d'informations, consultez la section « Utilisation du journal d'application du moteur de publication Web », page 90.

Gardez à l'esprit les astuces et considérations suivantes :

- Utilisez les comptes et les privilèges d'accès pour restreindre l'ensemble des scripts pouvant être exécutés par un utilisateur Web. Vérifiez que les scripts contiennent exclusivement des actions compatibles avec le Web et donnent uniquement accès aux scripts pouvant être utilisés depuis un navigateur Web.
- Pensez aux effets secondaires des scripts qui exécutent une combinaison d'actions contrôlées par des privilèges d'accès. Par exemple, si un script comporte une action visant à supprimer des enregistrements et qu'un utilisateur Web ne se connecte pas avec un compte permettant la suppression d'enregistrements, le script n'exécutera pas l'action Supprimer des enregistrements. L'exécution du script peut néanmoins se poursuivre et mener à des résultats inattendus.
- Dans la fenêtre de Modifier le script de ScriptMaker, sélectionnez **Exécuter le script avec tous les privilèges d'accès** pour permettre aux scripts d'effectuer des actions auxquelles les utilisateurs individuels n'ont normalement pas accès. Par exemple, vous pouvez empêcher les utilisateurs de supprimer des enregistrements avec leurs comptes et leurs privilèges d'accès, tout en leur permettant d'exécuter un script visant à supprimer certains types d'enregistrements dans les conditions définies au sein du script.
- Si vos scripts comportent des actions non prises en charge, par exemple des actions incompatibles avec le Web, utilisez l'action de script **Autoriser annulation utilisateur** pour déterminer le traitement des actions suivantes.
  - Si l'option d'action de script **Autoriser annulation utilisateur** est activée, les actions de script non prises en charge empêchent la poursuite du script.
  - Si l'option d'action de script **Autoriser annulation utilisateur** est désactivée, les actions de script non prises en charge sont ignorées et l'exécution du script se poursuit.
  - Si cette action de script n'est pas incluse, les scripts sont exécutés comme si la fonctionnalité était activée, c'est-à-dire que les actions de script non prises en charge interrompent l'exécution du script.
- Certains scripts fonctionnant avec une action d'un client FileMaker Pro peuvent nécessiter une action **Valider enreg./requêtes** supplémentaire pour permettre l'enregistrement des données sur l'hôte. Comme les utilisateurs Web ne bénéficient pas d'une connexion directe à l'hôte, ils ne sont pas informés des modifications de données. Par exemple, des fonctions comme les listes de valeurs conditionnelles ne présentent pas la même réactivité pour les utilisateurs Web parce que les données doivent être enregistrées sur l'hôte avant que leurs effets soient visibles dans la rubrique Liste de valeurs.

- Tout script modifiant des données doit inclure l'action Valider enreg/requêtes, car toutes les modifications de données ne sont pas visibles dans le navigateur tant que les données ne sont pas sauvegardées ou « soumises » au serveur. Ceci inclut plusieurs actions de script telles que Couper, Copier, Coller, etc. De nombreuses actions de script uniques doivent être converties en script pour inclure l'action Valider enreg/requêtes. Lors de la conception de scripts destinés à être exécutés depuis un navigateur Web, incorporez l'action Valider enreg./requêtes en fin de script pour garantir l'enregistrement de toutes les modifications.
- Pour créer des scripts conditionnels basés sur le type de client, utilisez la fonction Obtenir(VersionApplication). Si la valeur renvoyée comprend « moteur de publication Web 7.0v1 », vous savez que l'utilisateur actuel accède à votre base de données à l'aide de la publication Web personnalisée. Consultez l'Aide de FileMaker Pro pour de plus amples informations sur les fonctions.
- Si vous utilisez un script dans une feuille de style XSLT qui définit ou modifie un état, vous devez utiliser l'Admin Console de FileMaker Server pour activer l'option Sessions de base de données XSLT du moteur de publication Web. Dans le cas contraire, les états ne sont pas conservés entre chaque requête. Consultez l'Aide de FileMaker Server.

## Comportement des scripts dans les solutions de Publication Web personnalisée

Les actions de script suivantes ne fonctionnent pas comme dans FileMaker Pro quand elles sont utilisées sur le Web. Pour obtenir des informations sur toutes les actions de script, consultez l'aide de FileMaker Pro.

Action de script	Comportement dans les solutions de Publication Web personnalisée
Exécuter script	Les scripts ne peuvent pas s'exécuter dans d'autres fichiers, sauf quand ces fichiers sont hébergés par FileMaker Server et que la Publication Web personnalisée est activée dans les autres fichiers.
Quitter application	Déconnecte les utilisateurs Web, ferme les fenêtres mais ne quitte pas le navigateur Web.
Autoriser annulation utilisateur	Détermine comment les actions de script non prises en charge sont gérées. Activez cette option pour empêcher la poursuite des scripts, ou désactivez-la pour ignorer les actions non prises en charge. Pour plus d'informations, consultez la section « Astuces et considérations à propos des scripts », page 25. Les utilisateurs Web ne peuvent pas annuler les scripts de Publication Web personnalisée mais cette option permet aux actions de script non prises en charge d'empêcher la poursuite du script.
Gestion erreurs	Toujours activé avec la Publication Web personnalisée. Les utilisateurs Web ne peuvent pas annuler les scripts de Publication Web personnalisée.
Suspendre/reprendre script	Bien que cette action de script soit prise en charge par la Publication Web personnalisée, il faut éviter de l'employer. Lorsqu'une action de script est suspendue, l'exécution du script l'est également. Seul un script contenant l'action de script Reprendre script permet de procéder à une reprise. Si l'exécution du script reste suspendue jusqu'à expiration de la session, le script n'est pas terminé.
Trier les enregistrements	Avec l'action de script Trier enregistrements, vous devez enregistrer un ordre de tri à exécuter dans la Publication Web personnalisée.
Ouvrir URL	Cette action de script n'a aucun effet dans une solution de Publication Web personnalisée.
Activer rubrique	Vous ne pouvez pas utiliser l'action de script Activer rubrique pour activer une rubrique donnée dans le navigateur Web, mais vous pouvez l'employer conjointement à d'autres actions de script pour effectuer des opérations. Par exemple, vous pouvez activer une rubrique, copier son contenu, activer une autre rubrique et y coller la valeur. Pour visualiser l'effet dans le navigateur, assurez-vous de sauvegarder l'enregistrement avec l'action de script Valider enregistrement.
Valider enreg/requêtes	Soumet les enregistrements à la base de données.

## Déclencheurs de scripts et solutions de Publication Web personnalisée

Dans FileMaker Pro, les actions de scripts et les actions des utilisateurs (le fait de cliquer dans une rubrique par exemple) peuvent activer les déclencheurs de script. Cependant, dans la Publication Web personnalisée, les scripts sont les seuls à pouvoir activer les déclencheurs de script. Par exemple, si un utilisateur de la Publication Web instantanée clique dans une rubrique possédant un déclencheur de script SurEntreeObjet, le déclencheur n'est pas activé. Toutefois, si un script provoque le déplacement dans la rubrique, alors le déclencheur de script SurEntreeObjet est activé. Pour plus d'informations sur les déclencheurs de scripts, consultez l'aide FileMaker Pro.



# Chapitre 4

## ***Introduction à la Publication Web personnalisée avec XSLT***

Vous pouvez utiliser des feuilles de style FileMaker XSLT pour convertir, filtrer ou formater des données XML destinées à être manipulées dans un navigateur Web ou dans d'autres programmes et applications. Ce chapitre présente les feuilles de style FileMaker XSLT, ainsi qu'un outil qui vous aidera à débiter dans la création de feuilles de style XSLT — FileMaker XSLT Site Assistant. Pour plus d'informations concernant la structure des feuilles de style FileMaker XSLT, consultez le chapitre 6, « Conception de feuilles de style FileMaker XSLT ».

### **A propos de feuilles de style XSLT FileMaker**

Les feuilles de style FileMaker XSLT vous permettent d'effectuer :

- le filtrage des données FileMaker contrôlant les rubriques de base de données publiées par la feuille de style ;
- le masquage des métadonnées, telles que les noms de base de données et de rubrique ;
- le formatage des données présentées dans une page Web et le contrôle de l'interaction entre l'utilisateur Web et les données ;
- la sortie des données au format HTML ou texte, tel que vCards ou valeurs séparées par des virgules (CSV) ;
- la conversion des données utilisant la grammaire FileMaker XML en une autre grammaire XML destinée à une autre base de données ou application, telle que le format polices vectorielles modulaires (SVG) ;
- l'intégration d'un sous-ensemble quelconque de données FileMaker à d'autres sites Web et à d'autres applications middleware et personnalisées potentiellement très différentes de la base de données FileMaker ;
- la modification des noms de rubrique publiés pour empêcher l'utilisation non autorisée des informations concernant la structure des bases de données.

**Remarque** La Publication Web personnalisée avec XSLT pour FileMaker Server repose sur la recommandation W3C pour XSLT 1.0. Pour plus d'informations concernant XSLT 1.0, consultez le site [www.w3.org](http://www.w3.org). Les fonctions d'extension FileMaker XSLT offrent des fonctionnalités supplémentaires telles que la gestion de sessions, l'envoi d'emails et l'accès aux cookies et aux en-têtes. Pour plus d'informations, consultez la section « Utilisation des fonctions d'extension et des paramètres FileMaker XSLT », page 66. Le moteur de publication Web ne prend pas en charge les XSL Formatting Objects (XSL-FO).

### **Exemples d'utilisation des feuilles de style XSLT**

Voici quelques-unes des nombreuses possibilités d'utilisation de feuilles de style FileMaker XSLT :

- Dans une page Web, vous pouvez insérer une table contenant un sous-ensemble des données d'une base de données FileMaker à l'intention des utilisateurs Web. Cette table peut, par exemple, contenir le nom et l'adresse de personnes, mais non leur numéro de téléphone. Pour éviter les accès non autorisés, la page Web peut présenter des étiquettes génériques pour les données (telles que Name) au lieu des véritables noms de rubrique dans la base de données FileMaker, tels que first\_name.
- Vous pouvez créer une application ou une page Web intégrant les données d'une table externe FileMaker à des informations issues d'autres sources de données.

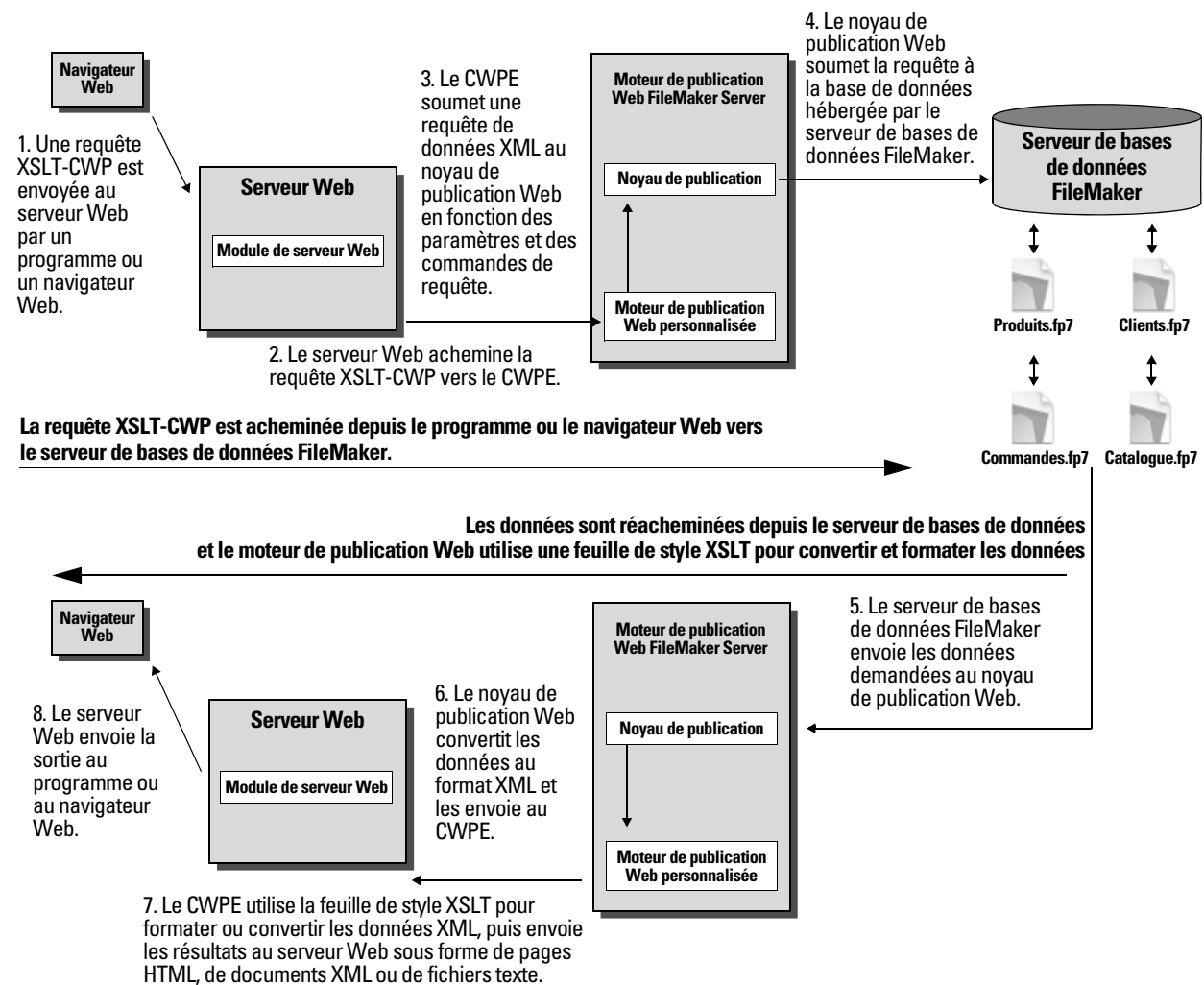
- Vous pouvez ajouter sur une page Web un bouton créant une vCard à partir des coordonnées d'une personne dans une base de données FileMaker.
- Vous pouvez convertir les données XML d'une base de données FileMaker en une grammaire XML reconnue par un tableur ou par une application de base de données.

## Utilisation de la Publication Web personnalisée avec XSLT

Si vous connaissez les langages XML et XSLT standard, vous pouvez commencer à utiliser le moteur de publication Web immédiatement après avoir pris connaissance des détails spécifiques à la publication XML et XSLT avec FileMaker, concernant notamment l'utilisation des fonctions d'extension et des commandes et paramètres de requête XSLT de FileMaker. XSLT Site Assistant est un outil qui vous aide à créer des feuilles de style et vous enseigne la manière dont elles sont construites. Vous pouvez ensuite utiliser vos outils de composition XML et XSLT favoris pour améliorer l'aspect de vos feuilles de style.

## Mode de génération des pages par le moteur de publication Web en fonction des données XML et des feuilles de style XSLT

Après l'envoi d'une requête de Publication Web personnalisée XSLT (XSLT-CWP) au serveur Web, le moteur de publication Web interroge la base de données FileMaker en fonction des commandes et des paramètres de requête définis dans la feuille de style et dans l'URL, puis sort les données conformément aux instructions de la feuille de style XSLT.



## Etapes générales pour l'utilisation de la Publication Web personnalisée avec XSLT

La présente section récapitule la procédure d'utilisation de la Publication Web personnalisée avec XSLT :

1. Vérifiez si la publication XSLT est activée dans l'Admin Console. Consultez l'Aide de FileMaker Server.
2. A l'aide de FileMaker Pro, ouvrez chaque base de données FileMaker que vous publiez et assurez-vous que le privilège d'accès étendu `fmxml` des bases de données est activé pour la Publication Web personnalisée avec XSLT. Consultez la section « Activation de la Publication Web personnalisée dans une base de données », page 21.

**Remarque** Lors de la création de feuilles de style destinées à un utilisateur final, prenez soin d'utiliser des jeux de privilèges d'accès équivalents dans la base de données FileMaker. Dans le cas contraire, vous pouvez avoir accès à des modèles et des fonctions de la base de données FileMaker qui ne seront pas accessibles à l'utilisateur final, ce qui risque d'entraîner des problèmes de fonctionnement.

3. Créez des feuilles de style XSLT comportant des fonctions d'extension, des commandes de requête et des paramètres de requête XSLT propres à FileMaker pour formater ou convertir les données XML d'une base de données FileMaker.

Vous pouvez utiliser l'outil FileMaker XSLT Site Assistant pour créer une ou plusieurs feuilles de style XSLT de base comme point de départ pour votre site. Consultez la section « Utilisation de l'outil FileMaker XSLT Site Assistant pour générer des feuilles de style FileMaker XSLT ».

Vous avez également la possibilité d'utiliser vos propres outils de composition XSLT ou éditeurs de texte pour modifier les feuilles de style XSLT à votre convenance ou pour en créer de nouvelles. Consultez la section chapitre 6, « Conception de feuilles de style FileMaker XSLT ».

4. Copiez ou placez les feuilles de style XSLT dans le sous-dossier `xslt-template-files`, situé dans le dossier Publication Web de FileMaker Server, sur l'ordinateur hôte sur lequel le moteur de publication Web est installé.

Vous pouvez également placer les feuilles de style dans un dossier provisoire ou dans le sous-dossier `xslt-template-files`.

5. Installez tous les fichiers statiques sur le serveur Web. Consultez la section « Utilisation des feuilles de style FileMaker XSLT dans un programme ou un site Web », page 34.
6. Créez ou modifiez un programme ou un site Web utilisant les feuilles de style XSLT.

Par exemple, vous pouvez utiliser pour votre site Web une page statique telle qu'`index.html`, qui réoriente automatiquement les utilisateurs Web vers une feuille de style XSLT ou qui comporte un lien vers cette dernière.

7. Assurez-vous que les mécanismes de sécurité pour votre site ou votre programme sont mis en œuvre.
8. Testez le site ou le programme avec les feuilles de style XSLT en utilisant les comptes et les privilèges d'accès définis pour les utilisateurs Web.
9. Mettez le site ou le programme à la disposition des utilisateurs.

## Utilisation de l'outil FileMaker XSLT Site Assistant pour générer des feuilles de style FileMaker XSLT

FileMaker XSLT Site Assistant est une application utilisée pour créer des feuilles de style XSLT de base. Ces dernières servent de point de départ pour la publication Web personnalisée avec XSLT. L'outil XSLT Site Assistant vous aide à mieux appréhender la structure des feuilles de style FileMaker XSLT. Le cas échéant, n'hésitez pas à utiliser vos propres outils de composition de feuilles de style XSLT ou éditeurs de texte pour modifier les feuilles de style. Vous ne pouvez pas utiliser XSLT Site Assistant pour modifier ni mettre à jour les feuilles de style existantes ; en revanche, vous pouvez l'employer pour générer les premières feuilles de style d'un site complet ou une feuille de style destinée à ajouter des fonctionnalités de base (comme la suppression des enregistrements) à un site existant.

Vous pouvez utiliser l'outil XSLT Site Assistant pour générer des feuilles de style XSLT pour tous les types de pages utilisables avec les bases de données FileMaker via la Publication Web personnalisée. Selon les options que vous choisissez dans XSLT Site Assistant, vous pouvez créer un site permettant aux utilisateurs :

- de consulter un seul enregistrement à la fois ;
- d'afficher une liste de tous les enregistrements de la base de données ;
- d'effectuer des recherches dans la base de données et de visualiser les résultats dans une liste ;
- de trier les enregistrements ;
- d'ajouter des enregistrements ;
- de modifier et de dupliquer les enregistrements ;
- de supprimer des enregistrements ;
- d'afficher un rapport récapitulatif.

Vous pouvez également générer une page d'accueil facultative liée aux autres feuilles de style XSLT générées.

Le moteur de publication Web utilise chacune de vos feuilles de style pour obtenir des données d'une base de données FileMaker de façon dynamique, chaque fois qu'un utilisateur Web envoie une requête HTTP et une adresse URL faisant référence à l'une de vos feuilles de style XSLT. Le moteur de publication Web utilise une feuille de style pour convertir et formater les données XML, puis génère la page HTML correspondante manipulable par l'utilisateur Web.

**Remarque** Les feuilles de style de XSLT Site Assistant convertissent les données FileMaker XML en pages HTML en fonction de la grammaire XML `fmresultset`, qui rend les feuilles de style incompatibles avec d'autres utilisations des données XML, telles qu'une exportation XML dans FileMaker Pro.

### Avant d'utiliser XSLT Site Assistant

Avant d'utiliser XSLT Site Assistant pour générer des feuilles de style XSLT pour une base de données :

- Définissez le privilège d'accès étendu sur `fmxml` dans la base de données. Lors de l'exécution de XSLT Site Assistant, utilisez les jeux de privilèges d'accès similaires à ceux que vous attribuez aux utilisateurs Web. Consultez la section « Activation de la Publication Web personnalisée dans une base de données », page 21.
- Ouvrez et hébergez la base de données sur le composant Serveur de bases de données de FileMaker Server. Consultez l'Aide de FileMaker Server.
- Assurez-vous que le composant serveur Web du déploiement FileMaker Server est en cours d'exécution.



- Assurez-vous que le composant moteur de publication Web du déploiement FileMaker Server est en cours d'exécution.
- Activez la publication XSLT dans le moteur de publication Web pour l'utilisation et le test des feuilles de style XSLT. Consultez l'Aide de FileMaker Server.

## Démarrage de XSLT Site Assistant

**Remarque** Pour utiliser XSLT Site Assistant, vous devez avoir préalablement installé la version 5 ou 6 de Java Runtime Environment.

Pour démarrer XSLT Site Assistant

1. Ouvrez un navigateur à la page Outils de publication Web FileMaker Server.

Allez à l'URL suivant :

`http://<serveur>:16000/tools`

Où `<serveur>` est la machine sur laquelle se trouve FileMaker Server.

2. Cliquez sur Outils PHP Site Assistant et XSLT Site Assistant pour accéder à la page Outils de publication Web FileMaker Server.

3. Cliquez sur Lancer XSLT Site Assistant.

FileMaker Server installe les fichiers JAR nécessaires sur votre machine locale. Une barre de progression s'affiche tant que la procédure n'est pas terminée.

4. (Option) Après l'installation des fichiers, vous pouvez choisir d'installer ou non une icône pour XSLT Site Assistant sur votre bureau. Cliquez sur OK pour installer l'icône.

Vous pouvez maintenant commencer à utiliser XSLT Site Assistant.

## Utilisation de XSLT Site Assistant

Pour obtenir des informations et des procédures détaillées concernant l'utilisation de XSLT Site Assistant, consultez l'aide de XSLT Site Assistant. Pour plus d'informations sur l'utilisation des feuilles de style générées par XSLT Site Assistant, consultez la section « Utilisation des feuilles de style FileMaker XSLT dans un programme ou un site Web », page 34.

**Important** Lors de l'utilisation de XSLT Site Assistant, si vous choisissez une base de données comportant plusieurs tables, prenez soin de sélectionner des modèles associés à la même table ; dans le cas contraire, le site généré produira des résultats inattendus. Par exemple, supposons qu'une base de données contienne une table Products et une table Customers. Lorsque vous sélectionnez les modèles pour une page de recherche, une page de modification d'enregistrements et une page d'ajout d'enregistrements, assurez-vous que les modèles sont tous associés à la même table.

## A propos des feuilles de style générées par XSLT Site Assistant

Les feuilles de style XSLT générées par XSLT Site Assistant intègrent plusieurs paramètres, éléments et instructions de traitement propres à FileMaker. Voici quelques exemples des éléments inclus :

- L’instruction de traitement `<?xslt-cwp-query params="query string-fragment"?>` spécifie la grammaire XML à utiliser et définit de façon statique le nom de la base de données que vous avez choisie dans XSLT Site Assistant. Consultez la section « Utilisation de commandes et de paramètres de requête définis de façon statique », page 62.
- L’élément `<xsl:param name="request-query"/>` permet d’accéder aux informations de requête dans une requête ou des données de formulaire HTML. Par exemple, cet élément peut être utilisé dans les feuilles de style XSLT Site Assistant afin d’accéder aux informations de la requête en cours pour déterminer l’emplacement dans un jeu d’enregistrements trouvé et créer des liens vers les enregistrements précédent et suivant. Consultez la section « Accès aux demandes d’informations dans une requête », page 67.
- L’élément `<xsl:param name="authenticated-xml-base-uri"/>`, parfois non inclus, permet d’accéder à l’identificateur URI (Uniform Resource Identifier) de base authentifié dans une requête lorsque des données XML supplémentaires sont requises dans la requête. Consultez la section « Utilisation du paramètre URI de base authentifié », page 69.

XSLT Site Assistant génère également la feuille de style `utilities.xsl` pour définir les erreurs et les modèles XSLT courants appelés par plusieurs feuilles de style Site Assistant.

Pour plus d’informations sur d’autres parties des feuilles de style XSLT Site Assistant, consultez le chapitre 6, « Conception de feuilles de style FileMaker XSLT ».

## Utilisation des feuilles de style FileMaker XSLT dans un programme ou un site Web

Que vous ayez utilisé XSLT Site Assistant pour générer des feuilles de style XSLT, ou que vous ayez créé intégralement vos propres feuilles de style, la procédure d’utilisation de ces dernières dans un programme ou un site Web avec le moteur de publication Web est la même.

Pour utiliser les feuilles de style FileMaker XSLT dans un programme ou un site Web :

1. Copiez ou placez les feuilles de style XSLT dans le sous-dossier `xslt-template-files`, situé dans le dossier Publication Web de FileMaker Server, sur l’ordinateur hôte sur lequel le moteur de publication Web est installé. Vous pouvez également placer les feuilles de style dans un dossier provisoire ou dans le sous-dossier `xslt-template-files`.
2. Si vos feuilles de style XSLT font référence à des fichiers statiques, tels que des fichiers HTML ou des images statiques, placez ces fichiers dans leur arborescence de dossiers d’origine sous le dossier racine du serveur Web. Assurez-vous que le chemin d’accès relatif est préservé.

Par exemple, supposons qu’une feuille de style XSLT fasse référence à un fichier image appelé `logo.jpg` à l’aide de la balise HTML ``. Le fichier `logo.jpg` doit être installé dans l’emplacement suivant du serveur Web :

```
<dossier racine>/fmi/xsl/logo.jpg
```

3. Si une rubrique de base de données Multimédia stocke une référence à un fichier et non le fichier lui-même, l'objet multimédia référencé doit alors être stocké dans le dossier FileMaker Pro Web lorsque l'enregistrement est créé ou modifié, puis doit être copié ou déplacé dans un dossier possédant le même emplacement relatif dans le dossier racine du logiciel de serveur Web. Consultez la section « A propos de la publication du contenu des rubriques Multimédia sur le Web », page 23.

**Remarque** Si les rubriques Multimédia stockent les fichiers proprement dits dans la base de données FileMaker, toute action sur le contenu de la rubrique Multimédia est alors inutile si le fichier de base de données est correctement hébergé et aisément accessible sur FileMaker Server.

4. Pour effectuer une requête de traitement de feuille de style XSLT, utilisez la syntaxe d'URL suivante :  
`<scheme>://<host>[:<port>]/fmi/xsl/<folder>/<stylesheet>.xsl[?<query string>]`

Consultez la section « A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT », page 59.

**Remarque** Dans le cas des sites Web, il est recommandé d'inclure une feuille de style XSLT comme page d'accueil accessible aux utilisateurs sans requérir la saisie d'une chaîne de requête. L'outil XSLT Site Assistant peut créer un fichier `home.xsl` ne nécessitant aucune chaîne de requête grâce à l'utilisation de l'instruction de traitement `<?xslt-cwp-query?>`. Par exemple, si vous avez copié vos feuilles de style (y compris une feuille de style `home.xsl`) dans le sous-dossier `my_templates` du dossier `xslt-template-files`, vous pouvez utiliser l'adresse URL suivante pour effectuer une requête de traitement des feuilles de style :  
[http://192.168.123.101/fmi/xsl/my\\_templates/home.xsl](http://192.168.123.101/fmi/xsl/my_templates/home.xsl)

**Important** Le moteur de publication Web ne permet pas aux utilisateurs Web d'afficher la source des feuilles de style XSLT installées dans le dossier `xslt-template-files`. Lorsque les utilisateurs Web envoient une requête pour traiter une feuille de style, le moteur de publication Web n'envoie au programme ou au navigateur Web que le résultat de la conversion de la feuille de style.

## Résolution des problèmes sur les feuilles de style XSLT

Si vous rencontrez des problèmes lors de l'utilisation de feuilles de style XSLT, vérifiez que :

- Les privilèges d'accès étendus dans la base de données sont configurés pour la Publication Web personnalisée avec XSLT et attribués à un compte utilisateur. Consultez la section « Activation de la Publication Web personnalisée dans une base de données », page 21.
- La base de données est hébergée et ouverte sur le composant Serveur de bases de données de FileMaker Server. Consultez l'Aide de FileMaker Server.
- Le nom de compte et le mot de passe que vous utilisez éventuellement pour la base de données sont corrects.
- Le composant serveur Web du déploiement FileMaker Server est en cours d'exécution.
- Le composant moteur de publication Web du déploiement FileMaker Server est en cours d'exécution.
- La publication XSLT est activée dans le moteur de publication Web.
  - Ouvrez la page Tests technologiques FileMaker Server dans un navigateur :  
<http://<serveur>:16000/test>  
 où <serveur> est la machine sur laquelle se trouve FileMaker Server.
  - Cliquez sur le lien Tester la publication Web personnalisée XSLT pour ouvrir une page XSLT accédant à la base de données de test FMServer\_Sample.

Pour plus d'informations, consultez le *Guide de démarrage de FileMaker Server* et l'aide de FileMaker Server.



# Chapitre 5

## ***Accès aux données XML avec le moteur de publication Web***

Vous pouvez utiliser et mettre à jour des données FileMaker au format XML (Extensible Markup Language) à l'aide du moteur de publication Web. De même que HTML est devenu le langage d'affichage standard pour communiquer sur le Web, XML est devenu le langage standard pour l'échange de données structurées. Un grand nombre d'utilisateurs, d'organisations et d'entreprises utilisent le langage XML pour transférer des informations sur les produits, des transactions, des informations sur le stock, ainsi que d'autres données commerciales.

### **Utilisation de la Publication Web personnalisée avec XML**

Si vous connaissez déjà le langage XML standard, vous pouvez commencer immédiatement à utiliser le moteur de publication Web. Vous devez néanmoins apprendre la syntaxe d'URL et connaître les paramètres de requête à employer pour la publication Web personnalisée avec XML.

Grâce aux requêtes URL HTTP avec les commandes et les paramètres de requête de FileMaker, vous pouvez interroger une base de données hébergée par FileMaker Server et télécharger les données obtenues au format XML. Par exemple, vous pouvez interroger une base de données pour obtenir tous les enregistrements relatifs à un code postal donné, puis employer les données XML obtenues comme bon vous semble.

Vous pouvez également utiliser les feuilles de style XSLT du moteur de publication Web côté serveur pour filtrer les données XML, les reformater au format HTML ou texte, par exemple vCards, ou les transformer à l'aide d'autres grammaires XML, par exemple le format SVG (polices vectorielles modulaires). Consultez les rubriques chapitre 4, « Introduction à la Publication Web personnalisée avec XSLT » et chapitre 6, « Conception de feuilles de style FileMaker XSLT ».

Pour obtenir davantage d'informations d'ordre général sur XML, d'autres exemples d'utilisation du langage XML, ainsi que des liens vers des ressources XML, consultez le site Web de FileMaker à l'adresse [www.filemaker.fr](http://www.filemaker.fr).

**Remarque** Le moteur de publication Web génère des données XML correctement formatées et conformes à la spécification XML 1.0. Pour plus de détails sur les conditions requises pour le formatage des données XML, reportez-vous aux spécifications XML disponibles à l'adresse [www.w3.org](http://www.w3.org).

### **Différences entre le moteur de publication Web et les options d'importation/exportation FileMaker Pro XML**

Le moteur de publication Web et FileMaker Pro permettent tous les deux d'utiliser des données XML avec des bases de données FileMaker. Cependant, il existe des différences importantes entre ces deux méthodes :

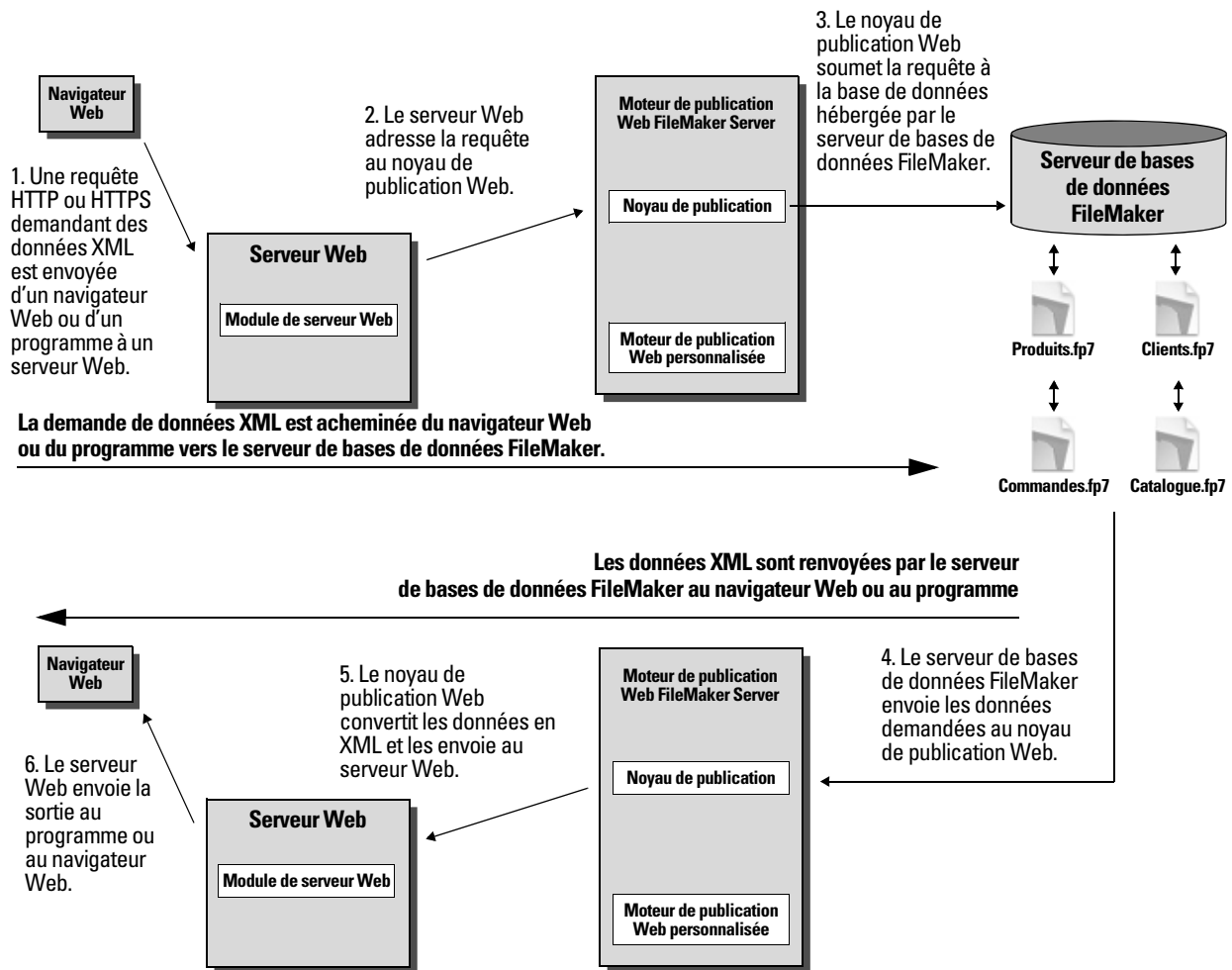
- Pour accéder aux données XML et à la fonction de publication Web XSLT, le moteur de publication Web prend en charge les grammaires fmresultset, FMPXMLRESULT et FMPXMLLAYOUT. Pour l'importation XML, FileMaker Pro utilise la grammaire FMPXMLRESULT, alors que pour l'exportation, FileMaker Pro utilise la grammaire FMPXMLRESULT ou FMPDSORESULT. Consultez la section « Accès aux données XML via le moteur de publication Web », page 43.
- Pour accéder aux données XML à l'aide du moteur de publication Web, utilisez une chaîne de requête du moteur de publication Web dans une URL. Pour importer et exporter des données XML avec FileMaker Pro, utilisez les commandes ou les scripts du menu FileMaker Pro.

- Le moteur de publication Web est basé sur le serveur et peut être installé sur le même hôte que FileMaker Server ou sur un hôte différent. L'importation et l'exportation XML de FileMaker Pro se font à partir d'un ordinateur de bureau.
- Vous pouvez accéder de façon dynamique aux données XML à partir des bases de données FileMaker en utilisant des requêtes URL avec le moteur de publication Web. La fonction d'exportation XML de FileMaker Pro génère un fichier de données XML spécifié à l'avance.
- Les opérations sur les données XML via un moteur de publication Web sont interactives. L'importation et l'exportation XML de FileMaker Pro se font selon un processus par lot.
- Contrairement à FileMaker Pro, le moteur de publication Web peut accéder aux données XML à partir d'une table externe FileMaker.
- Contrairement à FileMaker Pro, le moteur de publication Web peut accéder aux données d'une rubrique de type Multimédia.
- Contrairement à FileMaker Pro, le moteur de publication Web fournit un accès en temps réel aux données FileMaker via HTTP ou HTTPS.

**Remarque** Pour en savoir plus sur l'utilisation de FileMaker Pro pour effectuer des importations et des exportations au format XML, consultez l'aide FileMaker Pro.

### **Génération de données XML par le moteur de publication Web à partir d'une requête**

Une fois qu'une demande de données XML a été envoyée au serveur Web, le moteur de publication Web interroge la base de données FileMaker et renvoie les données sous forme de document XML.



## Processus général d'accès aux données XML à partir du moteur de publication Web

Voici un aperçu du processus consistant à employer le moteur de publication Web pour accéder aux données XML dans une base de données FileMaker.

1. Vérifiez si la publication XML est activée dans l'Admin Console de FileMaker Server. Consultez l'Aide de FileMaker Server.
2. Dans FileMaker Pro, ouvrez les bases de données FileMaker à publier et assurez-vous que pour chacune d'elles, le privilège fmxml est activé pour la fonction de publication Web personnalisée utilisant XML. Consultez la section « Activation de la Publication Web personnalisée dans une base de données », page 21.

Pour accéder aux données XML dans une table externe, définissez l'affichage de modèle de base de données sur **Afficher sous forme de formulaire** ou sur **Afficher sous forme de liste**. Si un utilisateur ou un script modifie l'affichage du modèle de base de données et le définit sur **Afficher sous forme de tableau**, seul le premier enregistrement (la première rangée de la table externe) est accessible sous forme de données XML.

Les données XML sortent dans un ordre qui correspond à l'ordre dans lequel les objets de la rubrique ont été ajoutés au modèle. Si vous souhaitez que l'ordre des données XML corresponde à l'ordre dans lequel les rubriques apparaissent à l'écran (du haut vers le bas et de la gauche vers la droite), sélectionnez toutes les rubriques, regroupez-les, puis dégrouperez-les. Cette procédure réinitialise l'ordre du modèle pour le faire correspondre à l'ordre de l'écran.

3. Envoyez une requête HTTP ou HTTPS sous la forme d'une URL qui spécifie la grammaire XML FileMaker, une commande de requête, puis un ou plusieurs paramètres de requête FileMaker au moteur de publication Web, par l'intermédiaire d'un formulaire HTML, d'un lien HREF ou d'un script dans votre programme ou votre page Web. Vous pouvez également entrer l'URL dans un navigateur Web. Pour plus d'informations sur la spécification d'une URL, consultez la section « A propos de la syntaxe d'URL pour les données XML et les objets de type Multimédia ». Pour plus d'informations sur les commandes de requête et sur les paramètres, consultez les sections « Utilisation de chaînes de requête FileMaker pour rechercher des données XML », page 52, et annexe A, « Noms corrects utilisés dans les chaînes de requête ».
4. Le moteur de publication Web utilise la grammaire spécifiée dans l'URL et génère les données XML contenant les résultats de votre requête, par exemple un ensemble d'enregistrements de la base de données, puis les renvoie à votre programme ou à votre navigateur Web.
5. S'il est équipé d'un analyseur XML, le navigateur Web affiche les données, ou le programme utilise les données de la façon que vous avez indiquée.

Si vous spécifiez une feuille de style côté client, l'analyseur du navigateur Web applique également les instructions de la feuille de style. Consultez la section « Utilisation du traitement des feuilles de style côté serveur et côté client », page 55.



## A propos de la syntaxe d'URL pour les données XML et les objets de type Multimédia

Cette section décrit la syntaxe d'URL à utiliser au niveau du moteur de publication Web pour accéder aux données XML et aux objets de type Multimédia à partir de bases de données FileMaker. Les feuilles de style XSLT n'emploient pas la même syntaxe que les feuilles de style XML. Consultez les rubriques « A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT », page 59 et « A propos de la syntaxe d'URL relative aux objets multimédias dans les solutions XSLT », page 60.

### A propos de la syntaxe d'URL pour les données XML

La syntaxe d'URL à employer pour utiliser le moteur de publication Web et accéder aux données XML à partir de bases de données FileMaker est la suivante :

`<scheme>://<host>[:<port>]/fmi/xml/<xml_grammar>.xml[?<query string>]`

où :

- `<scheme>` peut correspondre au protocole HTTP ou HTTPS.
- `<host>` correspond à l'adresse IP ou au nom de domaine de l'hôte sur lequel le serveur Web est installé.
- `<port>` est facultatif et spécifie le port utilisé par le serveur Web. Si aucun port n'est précisé, le port par défaut du protocole est utilisé (port 80 pour HTTP ou port 443 pour HTTPS).
- `<xml_grammar>` est le nom de la grammaire XML FileMaker. Les valeurs possibles sont `fmresultset.xml`, `FMPXMLRESULT.xml`, `FMPXMLLAYOUT.xml` ou `FMPDSORESET.xml`. Voir « Utilisation de la grammaire `fmresultset` », page 44 et « Utilisation des grammaires FileMaker XML », page 48.
- `<query string>` associe une commande de requête à un ou plusieurs paramètres de requête pour la publication XML FileMaker. (La commande `-dbnames` ne requiert aucun paramètre.) Consultez les rubriques « Utilisation de chaînes de requête FileMaker pour rechercher des données XML », page 52, et annexe A, « Noms corrects utilisés dans les chaînes de requête ».

**Remarque** La syntaxe d'URL, notamment les noms de commande et de paramètre de requête, applique la distinction majuscules-minuscules, à l'exception des parties de la chaîne de requête. La majorité des URL sont écrites en minuscules, à l'exception des trois noms de grammaire qui sont en majuscules : `FMPXMLRESULT`, `FMPXMLLAYOUT` et `FMPDSORESET`. Pour plus d'informations sur les règles de distinction majuscules-minuscules applicables à la chaîne de requête, consultez la section « Consignes d'utilisation des commandes et des paramètres de requête », page 94.

Voici deux exemples d'URL permettant d'accéder aux données XML via le moteur de publication Web :

`http://server.company.com/fmi/xml/fmresultset.xml?-db=products&-lay=sales&-findall`

`http://192.168.123.101/fmi/xml/FMPXMLRESULT.xml?-db=products&-lay=sales&-findall`

## A propos de la syntaxe d'URL pour les objets FileMaker de type Multimédia dans les solutions XML

Dans un document XML généré pour une solution XML, la syntaxe à employer pour faire référence à un objet de type Multimédia est différente pour les rubriques de type Multimédia qui stockent l'objet concerné dans la base de données de celle pour les rubriques de type Multimédia qui stockent une référence à l'objet.

- Si une rubrique de type Multimédia stocke l'objet lui-même dans la base de données, les <données> de la rubrique de type Multimédia utilisent la syntaxe d'URL suivante pour faire référence à l'objet :

```
<data>/fmi/xml/cnt/data.<extension>?<query string></data>
```

où <extension> est l'extension du nom de fichier qui identifie le type d'objet, par exemple .jpg.

L'extension du nom de fichier définit le type MIME qui permet au navigateur Web d'identifier correctement les données de type Multimédia. Pour plus d'informations concernant <query string>, consultez la section « A propos de la syntaxe d'URL pour les données XML ».

Par exemple :

```
<data>/fmi/xml/cnt/data.jpg?-db=products&-lay=sales&-field=product_image(1)&-recid=2</data>
```

**Remarque** Dans le document XML généré pour une rubrique Multimédia, la valeur du paramètre de requête -field est un nom de rubrique qualifié complet. Le nombre entre parenthèses indique le nombre de répétitions pour la rubrique Multimédia. Il est généré à la fois pour les rubriques multivaluées et pour les rubriques non multivaluées. Consultez la section « A propos de la syntaxe d'un nom de rubrique entièrement qualifié », page 95.

Pour extraire les données Multimédia de la base de données, utilisez la syntaxe suivante :

```
<scheme>://<host>[:<port>]/fmi/xml/cnt/data.<extension>?<query string>
```

Pour plus d'informations concernant <scheme>, <host> ou <port>, consultez la section « A propos de la syntaxe d'URL pour les données XML ».

Par exemple :

```
http://www.company.com/fmi/xml/cnt/data.jpg?-db=products&-lay=sales&-field=product_image(1)&-recid=2
```

- Si une rubrique Multimédia stocke une référence de fichier au lieu d'un objet proprement dit, l'élément <data> de la rubrique contient un chemin d'accès relatif faisant référence à l'objet. Par exemple :

```
<data>/images/logo.jpg</data>
```

**Remarque** L'objet multimédia référencé doit être stocké dans le dossier FileMaker Pro Web lorsque l'enregistrement est créé ou modifié, puis doit être copié ou déplacé dans un dossier possédant le même emplacement relatif dans le dossier racine du logiciel de serveur Web. Consultez la section « A propos de la publication du contenu des rubriques Multimédia sur le Web », page 23.

- Si une rubrique Multimédia est vide, l'élément <data> de la rubrique est également vide.

**Remarque** La syntaxe des objets de type Multimédia utilisant le langage XML est différente de la syntaxe des objets de type Multimédia utilisant le langage XSLT. Consultez la section « A propos de la syntaxe d'URL relative aux objets multimédias dans les solutions XSLT », page 60.

## A propos du codage de texte URL

Les URL utilisées pour accéder aux données XML et aux objets de type Multimédia doivent absolument être codées au format UTF-8 (Unicode Transformation Format 8 bits). Consultez la section « A propos des données UTF-8 », page 52.

Par exemple, pour définir la valeur de la rubrique info sur fiancée, vous pouvez utiliser l'URL suivante :

```
http://server.company.com/fmi/xml/fmresultset.xml?-db=members&-lay=relationships&-recid=2
&info= fianc%C3%A9e&-edit
```

Dans cet exemple d'URL, %C3%A9 est la représentation codée au format UTF-8 du caractère é.

Pour plus d'informations sur le codage de texte URL, consultez les spécifications de l'URL disponible à l'adresse [www.w3.org](http://www.w3.org).

## Accès aux données XML via le moteur de publication Web

Pour accéder aux données XML via le moteur de publication Web, utilisez une URL qui spécifie le nom de la grammaire FileMaker à employer, une commande de requête FileMaker, puis un ou plusieurs paramètres de requête FileMaker. À partir de votre base de données, le moteur de publication Web génère des données XML formatées à l'aide de l'un des types de grammaire suivants :

- **fmresultset** : c'est la grammaire que nous recommandons pour le moteur de publication Web. Elle est flexible et particulièrement adaptée à la création de feuilles de style XSLT avec un accès par nom simplifié aux rubriques et une manipulation plus souple des données **relatedset** (table externe). Cette grammaire est également plus étroitement liée à la terminologie et aux fonctionnalités de FileMaker, notamment aux options de stockage de type Global et à l'identification des rubriques de type Statistique et Calcul. Vous pouvez utiliser cette grammaire pour accéder aux données XML et aux feuilles de style XSLT. Pour faciliter la publication Web, cette grammaire se veut plus prolixe que la grammaire **FMPXMLRESULT**. Consultez la section « Utilisation de la grammaire fmresultset », page 44.
- **FMPXMLRESULT et FMPXMLLAYOUT** : vous pouvez employer les grammaires **FMPXMLRESULT** et **FMPXMLLAYOUT** avec le moteur de publication Web pour accéder aux données XML et aux feuilles de style XSLT. Pour utiliser une feuille de style dédiée à l'exportation XML et à la publication Web personnalisée, utilisez la grammaire **FMPXMLRESULT**. Pour accéder aux listes de valeurs et aux informations d'affichage des rubriques dans des modèles, utilisez la grammaire **FMPXMLLAYOUT**. Consultez la section « Utilisation des grammaires FileMaker XML », page 48.
- **FMPDSORESET** : La grammaire **FMPDSORESET**, qui est prise en charge dans FileMaker Pro pour l'exportation XML, n'est plus utilisée pour accéder aux données XML via le moteur de publication Web. La grammaire **FMPDSORESET** n'est pas prise en charge dans les feuilles de style XSLT. Pour plus d'informations sur la grammaire **FMPDSORESET**, consultez l'aide FileMaker Pro.

En fonction de la grammaire spécifiée dans la requête de type URL, le moteur de publication Web génère un document XML utilisant l'une des grammaires disponibles. Chaque document XML comporte une déclaration d'espace de nom XML par défaut pour la grammaire. Consultez la section « À propos des espaces de nom pour le langage XML de FileMaker ». Pour afficher et manipuler des données FileMaker au format XML, utilisez l'une de ces grammaires dans votre document ou dans votre page Web.

**Remarque** Les données XML générées par le moteur de publication Web sont codées au format UTF-8 (Unicode Transformation Format 8). Consultez la section « À propos des données UTF-8 », page 52.

## À propos des espaces de nom pour le langage XML de FileMaker

Des espaces de nom XML uniques aident à distinguer les balises XML de l'application à laquelle ils sont destinés. Par exemple, si vos documents XML contiennent deux éléments **<DATABASE>** destinés respectivement à des données XML FileMaker et à des données XML Oracle, les espaces de nom permettent d'identifier les éléments **<DATABASE>** pour ces deux logiciels.

Le moteur de publication Web génère un espace de nom par défaut pour chaque grammaire.

Pour cette grammaire	Cet espace de nom par défaut est généré
fmresultset	xmlns="http://www.filemaker.com/xml/fmresultset"
FMPXMLRESULT	xmlns="http://www.filemaker.com/ fmpxmlresult"
FMPXMLLAYOUT	xmlns="http://www.filemaker.com/fmpxmllayout"

## A propos des codes d'erreurs des bases de données FileMaker

Accessoirement, le moteur de publication Web renvoie un code d'erreur au début de chaque document XML répertoriant une erreur dans l'exécution de la commande de requête la plus récemment exécutée. La valeur zéro (0) indique qu'il n'y a pas d'erreur.

Pour cette grammaire	La syntaxe utilisée est
fmresultset	<error code="0"></error>
FMPXMLRESULT	<ERRORCODE>0</ERRORCODE>
FMPDSORESET	<ERRORCODE>0</ERRORCODE>

L'élément de code d'erreur dans le document XML indique des erreurs relatives à la base de données et aux chaînes de requête. D'autres types d'erreurs peuvent également survenir dans les feuilles de style XSLT, dans la mesure où elles sont gérées différemment. Consultez la section annexe B, « Codes d'erreur de la Publication Web personnalisée ».

## Récupération des définitions de type de document pour les grammaires FileMaker

Utilisez une requête HTTP pour récupérer les définitions de type de document (DTD) pour les grammaires FileMaker.

Pour cette grammaire	Utilisez la requête HTTP suivante
fmresultset	http://<host>[:<port>]/fmi/xml/fmresultset.dtd
FMPXMLRESULT	http://<host>[:<port>]/fmi/xml/FMPXMLRESULT.dtd
FMPXMLLAYOUT	http://<host>[:<port>]/fmi/xml/FMPXMLLAYOUT.dtd
FMPDSORESET	http://<host>[:<port>]/fmi/xml/FMPDSORESET.dtd?-db=<database>&-lay=<layout>

## Utilisation de la grammaire fmresultset

Les noms des éléments XML de cette grammaire utilisent la terminologie FileMaker et le stockage des rubriques est séparé des types de rubrique. La grammaire offre également la possibilité d'identifier les rubriques de type Statistique, Calcul et Global.

Pour utiliser la grammaire fmresultset, indiquez le nom de la grammaire fmresultset dans l'URL qui recherche le document XML à partir du moteur de publication Web :

fmresultset.xml

Par exemple :

http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family&-findall

**Remarque** Veillez à utiliser des minuscules pour spécifier la grammaire fmresultset.

Le moteur de publication Web génère alors un document XML utilisant la grammaire `fmresultset`. Dans le document XML, le moteur de publication Web fera référence à la définition du type de document pour la grammaire `fmresultset` dans l'instruction `<!DOCTYPE>` située sur la deuxième ligne du document, immédiatement après l'instruction `<?xml...?>`. L'instruction `<!DOCTYPE>` indique l'URL à employer pour télécharger la DTD relative à la grammaire `fmresultset`.

## Description des éléments de la grammaire `fmresultset`

La grammaire `fmresultset` se compose en priorité des éléments `<datasource>`, `<metadata>` et `<resultset>`.

### `<datasource>`, élément

Dans la grammaire `fmresultset`, l'élément `<datasource>` contient les attributs `table`, `layout`, `date-format`, `time-format`, `timestamp-format`, `total-count` et `database`.

- L'attribut `date-format` de l'élément `<datasource>` indique le format des dates dans le document XML :

`MM/jj/aaaa`

où :

- `MM` est la valeur à 2 chiffres relative au mois (de 01 à 12, 01 correspondant au mois de janvier et 12, au mois de décembre)
- `dd` est la valeur à 2 chiffres relative au jour du mois (de 00 à 31)
- `yyyy` est la valeur à 4 chiffres relative à l'année
- L'attribut `time-format` de l'élément `<datasource>` indique le format des heures dans le document XML :

`HH:mm:ss`

où :

- `HH` est la valeur à 2 chiffres pour les heures (de 00 à 23 pour le format 24 heures)
- `mm` est la valeur à 2 chiffres pour les minutes (de 00 à 59)
- `ss` est la valeur à 2 chiffres pour les secondes (de 00 à 59)
- L'attribut `timestamp-format` de l'élément `<datasource>` associe les formats de date et d'heure dans un horodatage :

`MM/jj/aaaa HH:mm:ss`

### `<metadata>`, élément

L'élément `<metadata>` de la grammaire `fmresultset` contient un ou plusieurs éléments `<field-definition>` et `<relatedset-definition>`, chacun contenant des attributs pour l'une des rubriques de cet ensemble de résultats.

L'attribut `<field-definition>` définit :

- si la rubrique est de type `auto-enter` (yes ou no)
- si la rubrique est de type `four-digit-year` (yes ou no)
- si la rubrique est de type `global` (yes ou no)
- le nombre maximal de valeurs multivaluées (attribut `max-repeat`)
- le nombre maximal de caractères autorisé (attribut `max-characters`)
- si la rubrique est de type `not-empty` (yes ou no)
- si la rubrique est réservée aux données numériques (yes ou no)

- le format du résultat (text, number, date, time, timestamp ou container)
- si la rubrique est de type time-of-day (yes ou no)
- le type (normal, calculation ou summary)
- et le nom de la rubrique (entièrement qualifié si nécessaire)

L'élément `<relatedset-definition>` correspond à une table externe. Chaque rubrique correspondante dans une table externe est représentée par l'élément `<field-definition>` intégré à l'élément `<relatedset-definition>`. Si une table externe possède plusieurs rubriques correspondantes, les définitions des rubriques correspondantes sont regroupées dans un même élément `<relatedset-definition>`.

### **`<resultset>`, élément**

L'élément `<resultset>` contient les éléments `<record>` renvoyés comme étant le résultat d'une requête et d'un attribut pour le nombre total d'enregistrements trouvés. Chaque élément `<record>` contient les données de la rubrique pour l'un des enregistrements dans l'ensemble des résultats, y compris les attributs `mod-id` et `record-id` de l'enregistrement, et l'élément `<data>` contenant les données pour l'une des rubriques de l'enregistrement.

Chaque enregistrement d'une table externe est représenté par un élément `<record>` intégré à l'élément `<relatedset>`. L'attribut de comptage de l'élément `<relatedset>` indique le nombre d'enregistrements de la table externe, et l'attribut de table spécifie la table associée avec la table externe.

## Exemple de données XML dans la grammaire fmresultset

L'exemple ci-dessous représente les données XML générées à l'aide de la grammaire fmresultset.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE fmresultset PUBLIC "-//FMI//DTD fmresultset//EN" ""http://localhost:16014/fmi/xml/fmresultset.dtd">
<fmresultset xmlns="http://www.filemaker.com/xml/fmresultset" version="1.0">
  <error code="0" />
  <product build="12/31/2012" name="FileMaker Web Publishing Engine" version="0.0.0.0" />
  <datasource database="art" date-format="MM/dd/yyyy" layout="web3" table="art" time-format="HH:mm:ss" timestamp-
format="MM/dd/yyyy HH:mm:ss" total-count="12" />
  <metadata>
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="Title" not-empty="no" numeric-
only="no" result="text" time-of-day="no" type="normal" />
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="Artist" not-empty="no"
numeric-only="no" result="text" time-of-day="no" type="normal" />
    <relatedset-definition table="artlocations">
      <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="artlocations::Location" not-
empty="no" numeric-only="no" result="text" time-of-day="no" type="normal" />
      <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="artlocations::Date" not-
empty="no" numeric-only="no" result="date" time-of-day="no" type="normal" />
    </relatedset-definition>
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="Style" not-empty="no"
numeric-only="no" result="text" time-of-day="no" type="normal" />
    <field-definition auto-enter="no" four-digit-year="no" global="no" max-repeat="1" name="length" not-empty="no"
numeric-only="no" result="number" time-of-day="no" type="calculation" />
  </metadata>
  <resultset count="1" fetch-size="1">
    <record mod-id="6" record-id="14">
      <field name="Title">
        <data>Spring in Giverny 3</data>
      </field>
      <field name="Artist">
        <data>Claude Monet</data>
      </field>
      <relatedset count="0" table="artlocations" />
      <field name="Style">
        <data />
      </field>
      <field name="length">
        <data>19</data>
      </field>
    </record>
  </resultset>
</fmresultset>
```

## Utilisation des grammaires FileMaker XML

Les autres grammaires XML FileMaker contiennent des informations au sujet des types de rubrique, des listes de valeur et des modèles. FMPXMLRESULT est une grammaire fonctionnellement équivalente à fmresultset. Pour accéder aux listes de valeurs et aux informations d’affichage des rubriques dans des modèles, utilisez la grammaire FMPXMMLAYOUT. Les grammaires FMPXMLRESULT et FMPXMMLAYOUT sont plus compactes pour les échanges de données.

Pour utiliser la grammaire FMPXMLRESULT, indiquez le nom de la grammaire dans l’URL qui recherche le document XML à partir du moteur de publication Web :

FMPXMLRESULT.xml

Par exemple :

<http://192.168.123.101/fmi/xml/FMPXMLRESULT.xml?-db=employees&-lay=family&-findall>

Pour utiliser la grammaire FMPXMMLAYOUT, indiquez le nom de la grammaire avec la commande de requête –view dans l’URL qui recherche le document XML à partir du moteur de publication Web :

FMPXMMLAYOUT.xml

Par exemple :

<http://192.168.123.101/fmi/xml/FMPXMMLAYOUT.xml?-db=employees&-lay=family&-view>

**Remarque** Veillez à entrer le nom des grammaires FMPXMLRESULT et FMPXMMLAYOUT en majuscules.

Dans le document XML généré, le moteur de publication Web référence la définition du type de document de la grammaire dans l’instruction <!DOCTYPE> située sur la deuxième ligne du document, immédiatement après l’instruction <?xml...?>. L’instruction <!DOCTYPE> indique l’URL à employer pour télécharger la DTD relative à la grammaire.

## Description des éléments de la grammaire FMPXMLRESULT

Dans la grammaire FMPXMLRESULT, l’élément <DATABASE> contient les attributs NAME, RECORDS, DATEFORMAT, LAYOUT et TIMEFORMAT.

L’attribut DATEFORMAT de l’élément <DATABASE> permet de déterminer le format des dates dans le document XML. L’attribut TIMEFORMAT de l’élément <DATABASE> permet de déterminer le format des heures dans le document XML. Les formats date et heure des grammaires FMPXMLRESULT et fmresultset sont identiques. Consultez les tableaux de la rubrique « Description des éléments de la grammaire fmresultset », page 45.

L’élément <METADATA> de la grammaire FMPXMLRESULT comporte un ou plusieurs éléments <FIELD>, contenant chacun des informations pour l’une des rubriques/colonnes de résultats, notamment le nom de la rubrique définie dans la base de données, le type de rubrique, la tolérance du choix Yes ou No pour les rubriques vides (attribut EMPTYOK), ainsi que le nombre maximal de rubriques multivaluées (attribut MAXREPEAT). Les valeurs autorisées pour les types de rubriques sont les suivantes : TEXT, NUMBER, DATE, TIME, TIMESTAMP et CONTAINER.

L’élément <RESULTSET> contient tous les éléments <ROW> renvoyés comme étant le résultat d’une requête et d’un attribut pour le nombre total d’enregistrements trouvés. Chaque élément <ROW> contient les données de la rubrique/colonne pour l’une des rangées de l’ensemble des résultats. Ces données incluent RECORDID et MODID pour la rangée (voir « Paramètre de requête –modid (numéro d’identification de la modification) », page 106) et l’élément <COL>. L’élément <COL> contient les données pour une rubrique/colonne de la rangée dans laquelle les éléments <DATA> représentent l’une des valeurs d’une rubrique multivaluée ou d’une rubrique de table externe.



## Exemple de données XML dans la grammaire FMPXMLRESULT

L'exemple ci-dessous représente les données XML générées à l'aide de la grammaire FMPXMLRESULT.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLRESULT PUBLIC "-//FMI//DTD FMPXMLRESULT//EN" ""http://localhost:16014/fmi/xml/
FMPXMLRESULT.dtd">
<FMPXMLRESULT xmlns="http://www.filemaker.com/fmpxmlresult">
  <ERRORCODE>0</ERRORCODE>
  <PRODUCT BUILD="12/31/2012" NAME="FileMaker Web Publishing Engine" VERSION="0.0.0.0" />
  <DATABASE DATEFORMAT="MM/dd/yyyy" LAYOUT="web" NAME="art" RECORDS="12" TIMEFORMAT="HH:mm:ss" />
  <METADATA>
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Title" TYPE="TEXT" />
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Artist" TYPE="TEXT" />
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Image" TYPE="CONTAINER" />
  </METADATA>
  <RESULTSET FOUND="1">
    <ROW MODID="6" RECORDID="15">
      <COL>
        <DATA>Spring in Giverny 4</DATA>
      </COL>
      <COL>
        <DATA>Claude Monet</DATA>
      </COL>
      <COL>
        <DATA>/fmi/xml/cnt/data.jpg?-db=art&-lay=web&-recid=15&-field=Image(1)</DATA>
      </COL>
    </ROW>
  </RESULTSET>
</FMPXMLRESULT>
```

L'ordre des éléments <COL> correspond à l'ordre des éléments <FIELD> dans l'élément <METADATA> : par exemple, si les rubriques Title et Artist sont répertoriées dans l'élément <METADATA>, Village Market puis Camille Pissarro apparaissent dans le même ordre dans les éléments <RESULTSET> et <ROW>.

## Description des éléments dans la grammaire FMPXMLLAYOUT

Dans la grammaire FMPXMLLAYOUT, l'élément <LAYOUT> contient le nom du modèle, le nom de la base de données et les éléments <FIELD> de chaque rubrique figurant dans le modèle correspondant de la base de données. Chaque élément <FIELD> décrit le type de style de la rubrique et contient l'attribut VALUELIST pour toute liste de valeurs associée à la rubrique.

L'élément <VALUELISTS> contient un ou plusieurs éléments <VALUELIST> pour chaque liste de valeurs répertoriée dans le modèle, chacun de ces éléments comportant le nom de la liste de valeurs et un élément <VALUE> pour chaque valeur de la liste.

En fonction des options sélectionnées dans la boîte de dialogue Rubriques pour la liste de valeurs de la base de données FileMaker, l'élément <VALUE> contient un attribut DISPLAY qui contient la valeur de la première rubrique uniquement, la valeur de la seconde rubrique uniquement ou les deux rubriques d'une liste de valeurs. Supposons que la première rubrique d'une liste de valeurs stocke le numéro d'ID d'un style

d'art (par exemple 100) et que la seconde rubrique affiche le nom associé au style (par exemple Impressionism). Voici un résumé du contenu de l'attribut DISPLAY lorsque les différentes combinaisons d'options sont sélectionnées dans la boîte de dialogue Rubriques pour la liste de valeurs :

- Si l'option **Afficher également les valeurs de la seconde rubrique** n'est pas sélectionnée, l'attribut DISPLAY contient pour valeur dans la première rubrique, d'une liste de valeurs uniquement. Dans l'exemple de données XML suivant, l'attribut DISPLAY contient uniquement le numéro d'ID du style d'art :

```
<VALUELISTS>
  <VALUELIST NAME="style">
    <VALUE DISPLAY="100">100</VALUE>
    <VALUE DISPLAY="101">101</VALUE>
    <VALUE DISPLAY="102">102</VALUE>
  </VALUELIST>
</VALUELISTS>
```

- Si les options **Afficher également les valeurs de la seconde rubrique** et **Afficher uniquement les valeurs de la seconde rubrique** sont toutes les deux sélectionnées, l'attribut DISPLAY contient la valeur dans la seconde rubrique uniquement. Dans l'exemple de données XML suivant, l'attribut DISPLAY contient uniquement le nom du style d'art :

```
<VALUELISTS>
  <VALUELIST NAME="style">
    <VALUE DISPLAY="Impressionism">100</VALUE>
    <VALUE DISPLAY="Cubism">101</VALUE>
    <VALUE DISPLAY="Abstract">102</VALUE>
  </VALUELIST>
</VALUELISTS>
```

- Si l'option **Afficher également les valeurs de la seconde rubrique** est sélectionnée et que l'option **Afficher uniquement les valeurs de la seconde rubrique** ne l'est pas, l'attribut DISPLAY contient les valeurs, dans les deux rubriques, d'une liste de valeurs. Dans l'exemple de données XML suivant, l'attribut DISPLAY contient le numéro d'ID du style d'art, de même que son nom :

```
<VALUELISTS>
  <VALUELIST NAME="style">
    <VALUE DISPLAY="100 Impressionism">100</VALUE>
    <VALUE DISPLAY="101 Cubism">101</VALUE>
    <VALUE DISPLAY="102 Abstract">102</VALUE>
  </VALUELIST>
</VALUELISTS>
```

Pour les rubriques de date, d'heure et d'horodatage, les données des listes de valeurs sont formatées à l'aide du format fm pour ce type de rubrique. Les formats fm sont MM/jj/aaaa pour la date, HH:mm:ss pour l'heure et MM/jj/aaaa HH:mm:ss pour l'horodatage. Consultez la section « Utilisation des fonctions d'extension de date, d'heure et de jour », page 79. Par exemple, si une liste de valeur birthdays est utilisée pour un menu local dans une rubrique birthdate d'un modèle et que la rubrique birthdate est de type date, alors les valeurs résultantes pour cette liste de valeurs seront toutes au format de date fm.

**Remarque** Si deux rubriques possédant des types de rubrique différents dans un modèle partagent la même liste de valeurs, le type de la première rubrique détermine le format des données de la liste de valeurs.

## Exemple de données XML dans la grammaire FMPXMMLAYOUT

L'exemple ci-dessous représente les données XML générées à l'aide de la grammaire FMPXMMLAYOUT.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMMLAYOUT PUBLIC "-//FMI//DTD FMPXMMLAYOUT//EN" ""http://localhost:16014/fmi/xml/
FMPXMMLAYOUT.dtd">
<FMPXMMLAYOUT xmlns="http://www.filemaker.com/fmpxmmlayout">
  <ERRORCODE>0</ERRORCODE>
  <PRODUCT BUILD="12/31/2012" NAME="FileMaker Web Publishing Engine" VERSION="0.0.0.0" />
  <LAYOUT DATABASE="art" NAME="web2">
    <FIELD NAME="Title">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="Artist">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="Image">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="artlocations::Location">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="artlocations::Date">
      <STYLE TYPE="EDITTEXT" VALUELIST="" />
    </FIELD>
    <FIELD NAME="Style">
      <STYLE TYPE="POPUPMENU" VALUELIST="style"/>
    </FIELD>
  </LAYOUT>
  <VALUELISTS>
    <VALUELIST NAME="style">
      <VALUE DISPLAY="Impressionism">100</VALUE>
      <VALUE DISPLAY="Cubism">101</VALUE>
      <VALUE DISPLAY="Abstract">102</VALUE>
    </VALUELIST>
  </VALUELISTS>
</FMPXMMLAYOUT>
```

## A propos des données UTF-8

Les données XML générées par le moteur de publication Web sont codées au format UTF-8 (Unicode Transformation Format 8 bits). Ce format compresse les données du format Unicode standard 16 bits au format 8 bits pour les caractères ASCII. Les analyseurs XML sont nécessaires à la prise en charge des codages Unicode et UTF-8.

Le codage au format UTF-8 comprend les représentations directes de la plupart des caractères utilisés en anglais, avec les valeurs 0 à 127 pour le jeu de caractères ASCII standard, et fournit des codages multioctets pour les caractères Unicode dont les valeurs sont supérieures.

**Remarque** Veillez à utiliser un navigateur Web ou un éditeur de texte prenant en charge les fichiers UTF-8.

Le format du codage UTF-8 présente les caractéristiques suivantes :

- Tous les caractères ASCII correspondent à des caractères UTF-8 d'un octet. Une chaîne ASCII valide est une chaîne UTF-8 valide.
- Tout caractère non ASCII (tout caractère dont le bit de poids fort est défini) fait partie d'un caractère multioctet.
- Le premier octet d'un caractère au format UTF-8 indique le nombre d'octets supplémentaires dans le caractère en question.
- Le premier octet d'un caractère multioctet est facilement identifiable par rapport aux octets suivants. Il est ainsi facile de repérer le début d'un caractère quel que soit son emplacement dans un flux de données.
- La conversion entre le format UTF-8 et le format Unicode ne pose aucun problème.
- Le codage au format UTF-8 est relativement compact. Dans le cas d'un texte contenant un pourcentage élevé de caractères ASCII, ce type de codage est plus compact que le codage au format Unicode. Dans le pire des cas, une chaîne UTF-8 n'est que 50 % plus grande que la chaîne Unicode équivalente.

## Utilisation de chaînes de requête FileMaker pour rechercher des données XML

Pour rechercher des données XML dans une base de données FileMaker, utilisez les commandes de requête et les paramètres dans une chaîne de requête. Par exemple, utilisez la commande de requête `–findall` dans la chaîne de requête suivante d'une URL pour rechercher une liste de tous les produits d'une base de données FileMaker dont le nom est `products`.

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=products–lay=sales&–findall`

Une chaîne de requête doit contenir uniquement une seule commande de requête, du type `–new`. La plupart des commandes de requête exigent également plusieurs paramètres de requête concordants dans la chaîne de requête. Par exemple, toutes les commandes de requête, à l'exception de `–dbnames`, ont besoin du paramètre `–db`, qui spécifie la base de données sur laquelle effectuer la recherche.

Vous pouvez également employer des commandes et des paramètres de requête dans une URL ou dans une instruction de traitement `<?xslt-cwp-query?>` de la feuille de style XSLT FileMaker. Consultez la section chapitre 6, « Conception de feuilles de style FileMaker XSLT ».

Cette section contient un récapitulatif des commandes et des paramètres de requête de FileMaker. Pour plus d'informations sur leur utilisation dans une chaîne de requête, consultez l'annexe A, « Noms corrects utilisés dans les chaînes de requête ».

**Remarque** Le moteur de publication Web prend également en charge une commande de requête supplémentaire (`-process`), ainsi que trois paramètres de requête qui ne doivent être utilisés qu'avec les feuilles de style XSLT FileMaker. Consultez la section « Utilisation de chaînes de requête dans les feuilles de style FileMaker XSLT », page 60.

Utilisez cette commande de requête	Pour exécuter cette commande
<code>-dbnames</code>	Extraire le nom de toutes les bases de données hébergées et partagées sur le Web.
<code>-delete</code>	Supprimer l'enregistrement.
<code>-dup</code>	Dupliquer l'enregistrement.
<code>-edit</code>	Modifier un enregistrement.
<code>-find</code>	Rechercher des enregistrements.
<code>-findall</code>	Rechercher tous les enregistrements.
<code>-findany</code>	Rechercher un enregistrement aléatoire.
<code>-findquery</code>	Accomplir une requête de recherche complexe ou composée.
<code>-layoutnames</code>	Extraire le nom de tous les modèles disponibles correspondant à une base de données spécifique, hébergée et partagée sur le Web.
<code>-new</code>	Ajouter un nouvel enregistrement.
<code>-scriptnames</code>	Extraire le nom de tous les scripts disponibles correspondant à une base de données spécifique, hébergée et partagée sur le Web.
<code>-view</code>	Extraire les informations du modèle d'une base de données si la grammaire <code>FMPXMLLAYOUT</code> est spécifiée. Extraire la section de <code>&lt;metadata&gt;</code> du document XML et un ensemble d'enregistrements vides si la grammaire <code>fmresultset</code> ou <code>FMPXMLRESULT</code> est spécifiée.

Utilisez ce paramètre de requête	Avec ces commandes de requête
<code>-db</code> (nom de la base de données)	Obligatoire pour toutes les commandes de requête, à l'exception de <code>-dbnames</code> et de <code>-process</code> (requêtes XSLT uniquement)
<code>-delta.relaté</code>	Facultatif pour <code>-edit</code>
<code>-field</code>	Obligatoire pour indiquer une rubrique dans une URL pour les requêtes de type Multimédia. Consultez la section « A propos de la syntaxe d'URL pour les objets FileMaker de type Multimédia dans les solutions XML », page 42.
<code>fieldname</code>	Un nom de rubrique au moins est obligatoire avec <code>-edit</code> . Facultatif pour <code>-find</code> . Consultez la section « Paramètre de requête <code>fieldname</code> (nom de rubrique non-multimédia) », page 103.
<code>fieldname.op</code> (opérateur)	Facultatif pour <code>-find</code>
<code>-lay</code> (nom du modèle)	Obligatoire pour toutes les commandes de requête, à l'exception de <code>-dbnames</code> , <code>-layoutnames</code> , <code>-scriptnames</code> et <code>-process</code> (requêtes XSLT uniquement)
<code>-lay.response</code> (passer d'un modèle à un modèle pour obtenir une réponse XML)	Obligatoire pour toutes les commandes de requête, à l'exception de <code>-dbnames</code> , <code>-layoutnames</code> , <code>-scriptnames</code> et <code>-process</code> (requêtes XSLT uniquement)
<code>-lop</code> (opérateur logique)	Facultatif pour <code>-find</code>
<code>-max</code> (nombre maximal d'enregistrements)	Facultatif pour <code>-find</code> , <code>-findall</code>
<code>-modid</code> (ID de modification)	Facultatif pour <code>-edit</code>

Utilisez ce paramètre de requête	Avec ces commandes de requête
<code>-query</code>	Nécessaire avec les requêtes de recherche composée <code>-findquery</code> .
<code>-recid</code> (ID de l'enregistrement)	Obligatoire pour <code>-edit</code> , <code>-delete</code> , <code>-dup</code> . Facultatif pour <code>-find</code>
<code>-relatedsets.filter</code>	Facultatif avec <code>-find</code> , <code>-edit</code> , <code>-new</code> , <code>-dup</code> , et <code>-findquery</code> .
<code>-relatedsets.max</code>	Facultatif avec <code>-find</code> , <code>-edit</code> , <code>-new</code> , <code>-dup</code> , et <code>-findquery</code> .
<code>-script</code> (exécution du script)	Facultatif pour <code>-find</code> , <code>-findall</code> , <code>-findany</code> , <code>-new</code> , <code>-edit</code> , <code>-delete</code> , <code>-dup</code> , <code>-view</code>
<code>-script.param</code> (transmettre une valeur de paramètre au script spécifié par <code>-script</code> )	Facultatif pour <code>-script</code>
<code>-script.prefind</code> (exécution du script avant <code>-find</code> , <code>-findany</code> et <code>-findall</code> )	Facultatif pour <code>-find</code> , <code>-findany</code> , <code>-findall</code>
<code>-script.prefind.param</code> (transmettre une valeur de paramètre au script spécifié par <code>-script.prefind</code> )	Facultatif pour <code>-script.prefind</code>
<code>-script.presort</code> (exécuter le script avant le tri)	Facultatif pour <code>-find</code> , <code>-findall</code>
<code>-script.presort.param</code> (transmettre une valeur de paramètre au script spécifié par <code>-script.presort</code> )	Facultatif pour <code>-script.presort</code>
<code>-skip</code> (ignorer les enregistrements)	Facultatif pour <code>-find</code> , <code>-findall</code>
<code>-sortfield.[1-9]</code> (trier la rubrique)	Facultatif pour <code>-find</code> , <code>-findall</code>
<code>-sortorder.[1-9]</code> (trier l'ordre)	Facultatif pour <code>-find</code> , <code>-findall</code>
<code>-stylehref</code> (feuille de style HREF)	Facultatif avec toutes les commandes de requête (indiquer une feuille de style URL pour <code>-styletype</code> )
<code>-styletype</code> (type de feuille de style)	Facultatif avec toutes les commandes de requête (pour indiquer une feuille de style côté client)

## Passage d'un modèle à un autre pour obtenir une réponse XML

Le paramètre de requête `-lay` indique le modèle à utiliser pour extraire des données XML. Souvent, le même modèle se révèle adapté au traitement des données répertoriées dans la requête. Dans certains cas, vous devrez rechercher les données qui utilisent un modèle contenant des rubriques qui, pour des raisons de sécurité, n'existent dans aucun autre modèle servant à l'affichage des résultats. (Pour rechercher des données dans une rubrique, il faut que celle-ci soit placée dans le modèle indiqué dans la requête XML.)

Pour définir un modèle destiné à l'affichage des réponses XML qui soit différent du modèle utilisé pour le traitement de la requête XML, utilisez le paramètre de requête facultatif `-lay.response`.

Par exemple, la requête suivante recherche les valeurs supérieures à 100 000 dans la rubrique **Salary** du modèle **Budget**. Les données obtenues s'affichent au moyen d'un modèle **ExecList**, qui n'inclut pas la rubrique **Salary**.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=Budget&Salary=100000&Salary.op=gt&-find
&-lay.response=ExecList
```

## Comprendre le traitement d'une requête XML

Plusieurs paramètres de requête interviennent dans le traitement d'une requête XML et dans la génération d'un document XML.

Voici l'ordre dans lequel FileMaker Server et le moteur de publication Web traitent une requête XML :

1. Traitez le paramètre de requête `-lay`.
2. Définissez les valeurs des rubriques de type Global dans la requête (la partie `.global=` d'une URL).
3. Traitez le paramètre de requête `-script.prefind` si indiqué.
4. Traitez les commandes de requêtes du type `-find` ou `-new`.
5. Traitez le paramètre de requête `-script.presort`, si indiqué.
6. Triez les données obtenues, si un tri a été indiqué.
7. Traitez le paramètre de requête `-lay.response` pour passer d'un modèle à un autre, si indiqué.
8. Traitez le paramètre de requête `-script`, si indiqué.
9. Générez le document XML.

Si l'une des étapes décrites ci-dessus génère un code d'erreur, le traitement de la requête s'arrête ; les étapes suivantes ne sont alors pas exécutées. Toutefois, toutes les étapes antérieures dans la requête restent exécutées.

Par exemple, imaginons une requête qui supprime l'enregistrement ouvert, trie les enregistrements, puis exécute un script. Si le paramètre `-sortfield` pointe vers une rubrique qui n'existe pas, la requête supprime l'enregistrement ouvert et renvoie le code d'erreur 102 (Rubrique manquante), mais n'exécute cependant pas le script.

## Utilisation du traitement des feuilles de style côté serveur et côté client

Le moteur de publication Web prend en charge le traitement d'une feuille de style XSLT côté serveur et vous permet également d'utiliser un paramètre de requête qui indique le traitement d'une feuille de style côté client.

Il est important de comprendre les différences entre les deux façons de traiter les feuilles de style, ainsi que les implications du traitement côté client en termes de sécurité. Le traitement côté serveur est mieux sécurisé que le traitement côté client car le premier interdit l'accès du Web aux données XML non filtrées. Dans le cas du traitement côté serveur, les données sont présentées sous la forme choisie par le propriétaire des données ou l'auteur de la feuille de style XSLT. Le traitement côté serveur masque de la vue des utilisateurs Web les noms des bases de données, les noms des rubriques, ainsi que d'autres détails relatifs à la mise en œuvre. Le traitement côté serveur peut également être utilisé pour spécifier des paramètres de requête définis de manière statique. Ceux-ci empêchent l'utilisation de commandes et de paramètres de requête non autorisés, comme par exemple les noms des bases de données. Consultez les rubriques chapitre 4, « Introduction à la Publication Web personnalisée avec XSLT » et chapitre 6, « Conception de feuilles de style FileMaker XSLT ».

Si votre solution exige un traitement des feuilles de style côté client, vous pouvez utiliser le moteur de publication Web pour générer une instruction de traitement de feuille de style XML avec chacune des grammaires, en intégrant les paramètres `-styletype` et `-stylehref` dans la requête de la chaîne de requête FileMaker. Vous pouvez utiliser des feuilles de style en cascade (CSS) ou des feuilles de style XSLT pour l'affichage du document XML.

- Le paramètre `–styletype` permet de définir la valeur de l'attribut `type` (`type=text/css` ou `type=text/xml`).
- Le paramètre `–stylehref` est utilisé pour définir la valeur de l'attribut `HREF` qui indique la position de la feuille de style à l'aide d'un chemin d'accès absolu. Par exemple : `href=/mystylesheet.css` ou `href=/stylesheets/mystylesheet.xml`. Le nom de la feuille de style peut être un nom quelconque, mais il doit comporter l'extension `.css` ou `.xml`.

Exemple de chaîne de requête FileMaker qui génère un traitement de feuille de style côté client :

`http://localhost/fmi/xml/fmresultset.xml?–db=products–lay=sales&–findall&–styletype=text/xml&–stylehref=/mystylesheet.xml`

**Remarque** Dans cet exemple, le `/` dans `–stylehref=/document.xml` est utilisé en raison du fait que la feuille de style est située au niveau du dossier racine du logiciel de serveur Web. Utilisez une URL pour la feuille de style possédant un chemin d'accès absolu pour indiquer son emplacement sur le serveur Web. La feuille de style peut également se trouver sur un autre serveur Web.

D'après cette requête, le moteur de publication Web inclut les instructions de traitement suivantes dans le document XML :

```
<?xml–stylesheet type="text/xml" href="/mystylesheet.xml"?>
```

Copiez ou placez la feuille de style à traiter côté client sur le serveur Web, à l'emplacement indiqué par le chemin d'accès absolu dans l'URL de l'attribut `HREF`.

**Important** Ne placez en aucun cas de feuilles de style dans le dossier `xslt-template-files` en vue d'un traitement côté client. En effet, ce dossier n'est utilisé que pour le traitement côté serveur des feuilles de style XSLT. Consultez la section « Utilisation des feuilles de style FileMaker XSLT dans un programme ou un site Web », page 34.

**Remarque** Certains navigateurs Web ne prennent pas en charge le traitement côté client. Pour plus d'informations, consultez la documentation de votre navigateur Web.

## Dépannage de l'accès aux documents XML

Si vous rencontrez des difficultés pour accéder aux documents XML avec le moteur de publication Web, vérifiez les points suivants :

- Les privilèges étendus contenus dans la base de données sont définis pour une publication Web personnalisée pour XML et affectés à un compte utilisateur. Consultez la section « Activation de la Publication Web personnalisée dans une base de données », page 21.
- La base de données est hébergée sur le composant Serveur de bases de données de FileMaker Server, et elle est ouverte par FileMaker Server. Consultez l'Aide de FileMaker Server.
- Le nom de compte et le mot de passe que vous utilisez éventuellement pour la base de données sont corrects.
- Le composant serveur Web du déploiement FileMaker Server est en cours d'exécution.
- Le composant moteur de publication Web du déploiement FileMaker Server est en cours d'exécution.
- La publication XML est activée dans le moteur de publication Web. Consultez l'Aide de FileMaker Server.



# Chapitre 6

## ***Conception de feuilles de style FileMaker XSLT***

Le présent chapitre fournit des informations sur la structure des feuilles de style FileMaker XSLT et sur l'utilisation des fonctions d'extension FileMaker XSLT.

### **Utilisation des feuilles de style XSLT avec le moteur de publication Web**

Lors de la conception et de l'utilisation de feuilles de style XSLT pour effectuer des requêtes de données FileMaker XML par l'intermédiaire du moteur de publication Web, prenez en compte les points énoncés ci-après.

- Pour utiliser une feuille de style XSLT avec le moteur de publication Web, vous devez indiquer le nom de la feuille de style dans une URL. Si vous ne spécifiez pas de nom ou que le moteur de publication Web ne parvient pas à localiser ni à analyser la feuille de style, le moteur de publication Web affiche une page d'erreur. Consultez la section « A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT », page 59.
- Le nom du fichier de feuille de style et le nom du dossier de stockage doivent présenter un codage URL au format UTF-8. Si votre feuille de style doit être compatible avec les anciens navigateurs Web, limitez les noms aux caractères ASCII.
- Vous devez indiquer la grammaire FileMaker XML à utiliser sous la forme d'un paramètre de requête dans l'URL ou d'un paramètre de requête défini de façon statique dans l'instruction de traitement `<?xslt-cwp-query?>`. Si vous ne spécifiez aucune grammaire XML, le moteur de publication Web affiche une erreur. Consultez la section « Spécification d'une grammaire XML pour une feuille de style FileMaker XSLT », page 61.
- Les paramètres de requête identifiant les données FileMaker XML sur lesquelles vous souhaitez faire porter les requêtes peuvent être spécifiés soit dans l'URL, soit sous la forme d'un paramètre de requête défini de façon statique dans l'instruction de traitement `<?xslt-cwp-query?>`. Consultez les rubriques « A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT », page 59 et « Utilisation de commandes et de paramètres de requête définis de façon statique », page 62.
- Vous pouvez éventuellement spécifier le codage de texte d'une requête XSLT en utilisant le paramètre de requête `-encoding`. Si vous n'indiquez aucun codage, le moteur de publication Web utilise son paramètre de codage de texte par défaut pour les requêtes. Consultez la section « Définition du codage du texte dans les requêtes », page 63.
- Si vous le souhaitez, vous pouvez définir une méthode de sortie via l'attribut de méthode de l'élément `<xsl:output>`. Si vous n'indiquez aucune méthode de sortie, le moteur de publication Web utilise le format de sortie HTML. Vous pouvez également spécifier le codage de la page de sortie au moyen de l'attribut de codage de l'élément `<xsl:output>`. Si vous n'indiquez aucun codage, le moteur de publication Web utilise le paramètre de codage de texte par défaut pour les pages de sortie. Consultez la section « Définition d'une méthode de sortie et du codage correspondant », page 64.
- Vous pouvez éventuellement préciser le codage de texte des emails envoyés par le moteur de publication Web par l'intermédiaire d'un paramètre de fonction pour la fonction d'extension `fmxslt:send_email()`. Consultez la section « Envoi d'emails à partir du moteur de publication Web », page 73.

Pour concevoir une requête, le moteur de publication Web commence par utiliser les paramètres et les commandes de requête éventuellement définis de façon statique dans l'instruction de traitement facultative `<?xslt-cwp-query?>`. La commande de requête et les paramètres de requête statiques constituent la requête de base. L'instruction de traitement `<?xslt-cwp-query?>` n'est pas requise dans une feuille de style, mais sa requête de base prévaut sur tout autre paramètre ou toute autre commande de requête correspondants spécifiés dans la chaîne de requête URL. Le moteur de publication Web ajoute ensuite à la requête de base dans la chaîne de requête URL une commande de requête ou des paramètres de requête supplémentaires *non* définis dans l'instruction de traitement `<?xslt-cwp-query?>`. Le moteur de publication Web utilise cette requête pour obtenir les données FileMaker XML et les renvoyer à votre programme ou à votre navigateur Web selon la méthode de sortie que vous avez choisie ou au format HTML.

## A propos de la base de référence des fonctions d'extension FileMaker XSLT

Cette version intègre une base de données FileMaker appelée XSLT Reference.fp7 contenant de brèves descriptions et des exemples de chacune des fonctions d'extension FileMaker XSLT. Cette base de données de référence des fonctions se trouve dans le répertoire suivant de toutes les machines de votre déploiement FileMaker Server (maîtres ou subordonnées).

### Mac OS

/Library/FileMaker Server/Example/XSLT

### Windows

<lecteur>:\Program Files\FileMaker\FileMaker Server\Examples\XSLT

où <lecteur> correspond au lecteur principal servant au démarrage de votre système.

## A propos des solutions de démarrage FileMaker XSLT

Cette version intègre une solution de démarrage FileMaker XSLT proposant un exemple de ce qu'il est possible de réaliser avec les solutions XSLT. Cette solution de démarrage XSLT se trouve dans le répertoire suivant de toutes les machines de votre déploiement FileMaker Server (maîtres ou subordonnées).

### Mac OS

/Library/FileMaker Server/Example/XSLT/Starter Solution

### Windows

<lecteur>:\Program Files\FileMaker\FileMaker Server\Examples\XSLT\Starter Solution

où <lecteur> correspond au lecteur principal servant au démarrage de votre système.

## A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT

La syntaxe d'URL pour l'utilisation des feuilles de style FileMaker XSLT avec le moteur de publication Web est la suivante :

`<scheme>://<host>[:<port>]/fmi/xsl/ [<path>]/<stylesheet.xml>[?<query string>]`

où :

- `<scheme>` peut correspondre au protocole HTTP ou HTTPS.
- `<host>` correspond à l'adresse IP ou au nom de domaine de l'hôte sur lequel le serveur Web est installé.
- `<port>` est facultatif et spécifie le port utilisé par le serveur Web. Si aucun port n'est précisé, le port par défaut du protocole est utilisé (port 80 pour HTTP ou port 443 pour HTTPS).
- `<path>` est facultatif et spécifie le ou les dossiers présents dans le répertoire `xslt-template-files` dans lequel se situe la feuille de style XSLT.
- `<stylesheet.xml>` est le nom de fichier de la feuille de style XSLT.
- `<query string>` peut correspondre à la combinaison d'une commande de requête et d'un ou de plusieurs paramètres de requête pour la Publication Web personnalisée avec XSLT. Consultez les rubriques « Utilisation de chaînes de requête dans les feuilles de style FileMaker XSLT », page 60, et annexe A, « Noms corrects utilisés dans les chaînes de requête ». Si la feuille de style spécifiée intègre une instruction de traitement `<?xslt-cwp-query?>`, la commande et les paramètres de requête statiques définis prévalent sur tout autre paramètre ou toute autre commande de requête correspondants, spécifiés dans la chaîne de requête URL. Consultez la section « Utilisation de commandes et de paramètres de requête définis de façon statique », page 62.

**Remarque** La syntaxe d'URL, notamment les noms de commande et de paramètre de requête, applique la distinction majuscules-minuscules, à l'exception des parties de la chaîne de requête. La majorité des adresses URL sont en minuscules, à l'exception des noms de grammaire en majuscules `FMPXMLRESULT` et `FMPXMLLAYOUT`. Pour plus d'informations sur les règles de distinction majuscules-minuscules applicables à la chaîne de requête, consultez la section « Consignes d'utilisation des commandes et des paramètres de requête », page 94.

Voici un exemple d'URL utilisant une feuille de style FileMaker XSLT avec le moteur de publication Web :

`http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?-grammar=fmresultset&-db=mydatabase  
&-lay=mylayout&-findall`

## A propos de la syntaxe d'URL relative aux objets multimédias dans les solutions XSLT

Dans un document XML généré pour une solution XSLT, la syntaxe permettant de faire référence à un objet multimédia n'est pas la même pour les rubriques Multimédia stockant l'objet proprement dit dans la base de données que pour les rubriques Multimédia stockant une référence à l'objet.

- Si une rubrique Multimédia stocke l'objet lui-même dans la base de données, l'élément `<data>` de la rubrique utilise la syntaxe d'URL ci-après pour faire référence à l'objet :

```
<data>/fmi/xsl/cnt/data.<extension>?<query string></data>
```

où `<extension>` est l'extension du nom de fichier qui identifie le type d'objet, par exemple `.jpg` ou `.mov`. Pour plus d'informations concernant `<query string>`, consultez la section « A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT ».

Par exemple :

```
<data>/fmi/xsl/cnt/data.jpg?-db=products&-lay=sales&-field=product_image(1)&-recid=2</data>
```

**Remarque** Dans le document XML généré pour une rubrique Multimédia, la valeur du paramètre de requête `-field` est un nom de rubrique qualifié complet. Le nombre entre parenthèses indique le nombre de répétitions pour la rubrique Multimédia. Il est généré à la fois pour les rubriques multivaluées et pour les rubriques non multivaluées. Consultez la section « A propos de la syntaxe d'un nom de rubrique entièrement qualifié », page 95.

Pour extraire les données Multimédia de la base de données, utilisez la syntaxe suivante :

```
<scheme>://<host>[:<port>]/fmi/xsl/cnt/data.<extension>?<query string>
```

Pour plus d'informations concernant `<scheme>`, `<host>` ou `<port>`, consultez la section « A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT ».

Par exemple :

```
http://www.company.com/fmi/xsl/cnt/data.jpg?-db=products&-lay=sales&-field=product_image(1)&-recid=2
```

- Si une rubrique Multimédia stocke une référence de fichier au lieu d'un objet proprement dit, l'élément `<data>` de la rubrique contient un chemin d'accès relatif faisant référence à l'objet. Par exemple, si le fichier `logo.jpg` se trouvait dans le sous-dossier **Web** du dossier **FileMaker Pro**, l'élément `<data>` de la rubrique Multimédia est le suivant :

```
<data>/images/logo.jpg</data>
```

**Remarque** L'objet multimédia référencé doit être stocké dans le dossier **FileMaker Pro Web** lorsque l'enregistrement est créé ou modifié, puis doit être copié ou déplacé dans un dossier possédant le même emplacement relatif dans le dossier racine du logiciel de serveur Web. Consultez la section « A propos de la publication du contenu des rubriques Multimédia sur le Web », page 23.

- Si une rubrique Multimédia est vide, l'élément `<data>` de la rubrique est également vide.

## Utilisation de chaînes de requête dans les feuilles de style FileMaker XSLT

Lors de l'utilisation d'une chaîne de requête dans une URL ou dans l'instruction de traitement `<?xslt-cwp-query?>` d'une feuille de style FileMaker XSLT, vous pouvez inclure un paramètre et une commande de requête définis quels qu'ils soient pour requérir des données XML d'une base de données FileMaker. Consultez la section « Utilisation de chaînes de requête FileMaker pour rechercher des données XML », page 52.

Vous pouvez également utiliser les commandes et les paramètres de requête ci-après définis pour une utilisation exclusive avec des feuilles de style FileMaker XSLT.

**Utilisez ce nom de commande ou de paramètre de requête XSLT**

	Pour	Commentaire
<code>-grammar3</code>	Spécifier la grammaire XML pour les requêtes XSLT-CWP ou pour les feuilles de style XSLT. Consultez la section « Spécification d'une grammaire XML pour une feuille de style FileMaker XSLT ».	Ce paramètre de requête est requis dans toutes les requêtes XSLT.
<code>-encoding</code>	Indiquer le codage du texte pour une requête. Consultez la section « Définition du codage du texte dans les requêtes », page 63.	Ce paramètre de requête est facultatif dans toutes les requêtes XSLT.
<code>-process</code>	Traiter une feuille de style sans effectuer de requêtes de données. Consultez la section « Traitement des requêtes XSLT n'adressant pas de requêtes à FileMaker Server », page 65.	Cette commande de requête requiert le paramètre de requête <code>-grammar</code> .
<code>-token</code>	Transmettre des valeurs entre des pages sans utiliser de sessions ni de cookies. Consultez la section « Utilisation du paramètre <code>-token</code> pour transmettre des informations entre plusieurs feuilles de style », page 66.	Ce paramètre de requête est facultatif dans toutes les requêtes XSLT.

## Spécification d'une grammaire XML pour une feuille de style FileMaker XSLT

La grammaire XML recommandée pour la Publication Web personnalisée avec XSLT est la grammaire `fmresultset`, conçue pour s'adapter aisément à XSLT. Consultez la section « Utilisation de la grammaire `fmresultset` », page 44. Vous pouvez également utiliser les grammaires antérieures `FMPXMLRESULT` ou `FMPXMLLAYOUT`. Pour accéder aux listes de valeurs et aux informations d'affichage des rubriques dans des modèles, utilisez la grammaire `FMPXMLLAYOUT`. Consultez la section « Utilisation des grammaires FileMaker XML », page 48. Vous ne pouvez pas utiliser la grammaire `FMPDSORESET` pour la Publication Web personnalisée avec XSLT.

Pour indiquer la grammaire applicable à une feuille de style FileMaker XSLT, utilisez le paramètre de requête `-grammar` dans une URL ou sous la forme d'un paramètre de requête défini de façon statique dans l'instruction de traitement `<?xslt-cwp-query?>`.

Par exemple, dans un URL :

```
http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?-grammar=fmresultset&-db=mydatabase
&-lay=mylayout&-findall
```

Par exemple, dans une instruction de traitement :

```
<?xslt-cwp-query params="-grammar=fmresultset&-db=mydatabase&-lay=mylayout&-findall"?>
```

**Important** Si vous ne spécifiez pas de grammaire XML pour une feuille de style FileMaker XSLT, le message d'erreur `QUERY -ER0001` s'affiche. Consultez la section annexe B, « Codes d'erreur de la Publication Web personnalisée ».

## A propos des espaces de nom et des préfixes dans les feuilles de style FileMaker XSLT

L'utilisation d'espaces de nom XSLT uniques permet d'identifier les balises XSLT par l'application pour laquelle elles ont été conçues. Dans l'élément `<xsl:stylesheet>` situé au début de toutes les feuilles de style FileMaker XSLT, déclarez les espaces de nom pour les fonctions d'extension FileMaker XSLT et pour les grammaires spécifiques que vous utilisez dans la feuille de style.

Si vous utilisez	Déclarez l'espace de nom	Utilisez le préfixe
Grammaire XML fmresultset	<code>xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"</code>	fmrs
Grammaire FMPXMLRESULT	<code>xmlns:fmp="http://www.filemaker.com/fmpxmlresult"</code>	fmp
Grammaire FMPXMLLAYOUT	<code>xmlns:fml="http://www.filemaker.com/fmpxmllayout"</code>	fml
Pour la grammaire de requête XML	<code>xmlns:fmq="http://www.filemaker.com/xml/query"</code>	fmq
Pour les fonctions d'extension FileMaker XSLT	<code>xmlns:fmxls="xalan://com.fmi.xslt.ExtensionFunctions"</code>	fmxls

Vous devez également déclarer l'espace de nom ci-après car il est requis dans chaque feuille de style FileMaker XSLT :

```
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
```

Voici quelques exemples de déclarations d'espaces de nom :

```
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"
  xmlns:fml="http://www.filemaker.com/fmpxmllayout"
  xmlns:fmq="http://www.filemaker.com/xml/query"
  xmlns:fmxls="xalan://com.fmi.xslt.ExtensionFunctions"
  exclude-result-prefixes="xsl fmrs fmq fml fmxls">
```

## Utilisation de commandes et de paramètres de requête définis de façon statique

Vous pouvez empêcher l'utilisation non autorisée de commandes et de paramètres de requête avec votre feuille de style FileMaker XSLT en définissant de façon statique les commandes et les paramètres de requête que vous souhaitez utiliser pour les requêtes de données XML. Si des commandes et des paramètres de requête sont définis de façon statique dans une feuille de style, ils prévalent sur tout autre paramètre ou commande de requête correspondant qu'un client tente de spécifier dans la chaîne de requête URL.

Les feuilles de style générées par XSLT Site Assistant utilisent les commandes et paramètres de requêtes définis de façon statique. L'utilisation des paramètres et des commandes de requête statiques avec FileMaker est recommandée car il s'agit du meilleur moyen d'améliorer la sécurité de votre solution.

Pour définir les commandes et les paramètres de requête de façon statique, utilisez l'instruction de traitement suivante au début de votre feuille de style FileMaker XSLT :

```
<?xslt-cwp-query params="query string-fragment"?>
```

où :

query string-fragment est une chaîne contenant les paires nom-valeur au format suivant :

name=value&name2=value2....

où :

name est une chaîne correspondant au nom d'une commande de requête, d'un paramètre de requête ou d'une rubrique de base de données.

value est une valeur de chaîne de longueur arbitraire. Pour les paramètres de requête et les noms de rubrique, utilisez la valeur spécifique que vous souhaitez définir, telle que `-db=products`. Pour les commandes de requête, n'indiquez pas le signe = ni même une valeur après le nom de la commande, par exemple `-findall`. Consultez la section annexe A, « Noms corrects utilisés dans les chaînes de requête ».

Les chaînes utilisées dans le fragment doivent présenter un codage URL. Consultez la section « A propos du codage de texte URL », page 42. Vous devez utiliser le codage de caractère spécifié par l'attribut de codage dans la balise `<xsl:output>`. Si aucun codage n'est spécifié, le moteur de publication Web applique son codage par défaut.

Le séparateur entre deux paires nom-valeur doit correspondre au caractère Et commercial (&).

Par exemple, supposons que vous utilisiez l'instruction de traitement suivante dans la feuille de style `my_stylesheet.xml` :

```
<?xslt-cwp-query params="-db=products&-lay=sales&-grammar=fmresultset&productname=the%20item&-find"?">
```

Cet exemple d'instruction de traitement oblige toutes les requêtes concernant la feuille de style `my_stylesheet.xml` à utiliser la grammaire `fmresultset` avec la base de données `products` et le modèle `sales`, puis à exécuter une requête `-find` avec la rubrique `productname` configurée sur la valeur `the%20item`.

Si un client effectue la requête ci-après à l'aide de la feuille de style `my_stylesheet.xml` :

```
http://server.company.com/fmi/xsl/my_stylesheet.xml?-lay=revenue&city=London&-edit
```

le moteur de publication Web traite la requête XML suivante :

```
http://server.company.com/fmi/xml/fmresultset.xml?-db=products&-lay=sales&productname=the%20item&city=London&-find
```

La commande et les paramètres de requête définis de façon statique prévalent sur le paramètre de requête `-lay=revenue` et sur la commande de requête `-edit` fournis par le client. La rubrique `city` n'étant pas définie de façon statique dans l'instruction de traitement, le moteur de publication Web insère dans la requête XML la valeur `London` pour cette rubrique.

## Définition du codage du texte dans les requêtes

Le moteur de publication Web effectue les étapes ci-après dans l'ordre indiqué jusqu'à ce qu'il détermine le codage d'une requête XSLT :

1. Le moteur vérifie si l'attribut de jeu de caractères est défini dans l'en-tête de requête `Content-Type`.
2. Le moteur vérifie si vous avez spécifié un codage avec le paramètre de requête `-encoding`. Vous pouvez indiquer ce paramètre dans une URL ou sous la forme d'un paramètre de requête défini de façon statique dans l'instruction de traitement `<?xslt-cwp-query?>`. La valeur du paramètre `-encoding` désigne le codage appliqué au reste des paramètres de la requête. Les valeurs correctes pour ce paramètre sont répertoriées dans le tableau ci-après. Par exemple :

```
http://192.168.123.101/fmi/xsl/template/my_stylesheet.xml?-db=products-lay=sales&-grammar=fmresultset
&-encoding=Shift_JIS&-findall
```



3. Le moteur de publication Web utilise le paramétrage en cours de son option de codage de texte par défaut requêtes et pages de sortie. Lorsque le moteur de publication Web est installé pour la première fois, le codage de texte par défaut initial pour les requêtes correspond à UTF-8. Vous pouvez modifier ce paramétrage à l'aide de l'Admin Console. Consultez l'Aide de FileMaker Server.

Une fois que le moteur de publication Web a déterminé le codage, il applique ce dernier et aucune autre opération n'est effectuée pour définir le codage. Par exemple, si l'attribut de jeu de caractères est défini dans l'en-tête de requête Content-Type, le moteur de publication Web n'utilise pas la valeur du paramètre de requête `–encoding`.

Le codage de texte spécifié par l'intermédiaire de l'une des méthodes ci-dessus doit appliquer l'un des codages suivants :

Codage	Description
US-ASCII	Jeu de caractères ASCII de base habituellement utilisé pour les emails en anglais au format texte brut.
ISO-8859-1	Jeu de caractères Latin 1 habituellement utilisé pour les pages Web et les emails contenant des caractères latins et nécessitant des caractères ASCII en majuscules.
ISO-8859-15	Jeu de caractères Latin 9, quasiment identique au jeu de caractères Latin 1, complété par l'ajout du symbole Euro €.
ISO-2022-JP	Codage japonais ISO habituellement utilisé pour les emails en japonais.
Shift_JIS	Codage japonais habituellement utilisé pour les pages Web en japonais.
UTF-8	Codage Unicode Format 8 bits. L'utilisation du codage UTF-8 pour les emails et les pages Web devient de plus en plus fréquente car les principaux navigateurs et clients de messagerie prennent désormais ce codage en charge. Le codage UTF-8 reconnaissant l'intégralité de la gamme des caractères Unicode, il prend en charge des pages dans n'importe quelle langue.

### Remarques

- Lorsque le moteur de publication Web est installé pour la première fois, le codage de texte par défaut initial pour les pages de sortie correspond à UTF-8. Consultez la section « Définition d'une méthode de sortie et du codage correspondant ». Dans le cas des emails, le codage de texte par défaut initialement utilisé par le moteur de publication Web est ISO-8859-1. Vous pouvez modifier ce paramétrage à l'aide de l'Admin Console.
- Vous pouvez également configurer le codage des emails au moyen de la fonction d'extension `fmxml:send_email(String smtpFields, String body, String encoding)`. Consultez la section « Envoi d'emails à partir du moteur de publication Web », page 73.

## Définition d'une méthode de sortie et du codage correspondant

Vous pouvez définir une méthode de sortie ainsi que le codage des pages de sortie en utilisant les attributs de méthode et de codage de l'élément `<xsl:output>`. Ces deux attributs sont facultatifs.

L'attribut de méthode spécifie le type de sortie, à savoir html, text ou xml. Aucun autre type de méthode n'est pris en charge. Si vous n'indiquez pas de méthode, le moteur de publication Web utilise le format html.

L'attribut de codage désigne le codage des pages de sortie. Vous pouvez spécifier l'un des codages répertoriés dans le tableau de la section précédente. Si vous n'indiquez aucun codage, le moteur de publication Web utilise son paramètre de codage de texte par défaut pour les pages de sortie.

Par exemple :

```
<xsl:output method="html" encoding="ISO-8859-1"/>
```

Si vous n'utilisez pas l'élément `<xsl:output>` dans une feuille de style, le moteur de publication Web sort les pages HTML en utilisant le paramètre de codage de texte par défaut en vigueur pour les pages de sortie.



## A propos du codage des feuilles de style XSLT

Outre le codage des requêtes et des pages de sortie, le codage de vos feuilles de style XSLT doit être spécifié dans l'attribut de codage de la déclaration XML au début de la feuille de style. Vous pouvez utiliser l'un des codages de texte répertoriés dans le tableau de la , page 64.

Par exemple, la déclaration ci-après spécifie le codage de feuille de style UTF-8 :

```
<?xml version="1.0" encoding="UTF-8"?>
```

Si vous ne précisez pas le codage des feuilles de style, le moteur de publication Web applique le codage UTF-8.

## Traitement des requêtes XSLT n'adressant pas de requêtes à FileMaker Server

Vous pouvez utiliser la commande de requête `–process` pour traiter les requêtes XSLT ne portant pas sur les données de la base de données ou dans le cas où votre feuille de style ne requiert pas d'informations propres à la base, telles que des enregistrements, des noms de rubrique ou des noms de modèle. L'utilisation de la commande `–process` dans ces types de situations contribue à réduire la charge de travail de FileMaker Server.

Par exemple, vous pouvez utiliser la commande `–process` pour :

- charger une feuille de style générant une page statique, si aucune information de base de données n'est requise ;
- charger une feuille de style créant un enregistrement, si la feuille de style ne requiert aucune information de base de données ou de modèle, telle qu'une liste de valeurs ;
- utiliser une fonction d'extension telle que `fmxml:send_email()` ne requérant aucune donnée de la base de données ;
- accéder aux informations stockées dans une session si aucune information de base de données n'est requise.

La commande `–process` renvoie un document XML contenant des informations produit relatives au moteur de publication Web.

Le seul paramètre requis pour la commande `–process` est le paramètre `–grammar`, et vous devez utiliser les grammaires `fmresultset` ou `FMPXMLRESULT`.

Par exemple :

```
http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–grammar=fmresultset&–process
```

## Utilisation du paramètre –token pour transmettre des informations entre plusieurs feuilles de style

Vous pouvez utiliser le paramètre de requête –token dans une URL ou sous la forme d’une commande de requête définie de façon statique pour transmettre toute information définie par l’utilisateur -entre différentes feuilles de style sans utiliser de sessions ni de cookies. Le paramètre de requête –token est facultatif avec toutes les commandes de requête.

La valeur de ce paramètre définie par l’utilisateur peut correspondre à n’importe quelle chaîne de caractères présentant un codage URL. Par exemple :

```
http://192.168.123.101/fmi/xsl/template/my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset
&-token.D100=Pending&-findall
```

Consultez la section « Paramètre de requête –token.[string] (transfert des valeurs d’une feuille de style XSLT à une autre) », page 113.

**Important** N’utilisez pas le paramètre de requête –token pour transmettre des données confidentielles.

Pour récupérer la valeur du paramètre de requête –token, utilisez l’instruction `<xsl:param name="request-query" />`. Consultez la section « Accès aux demandes d’informations dans une requête », page 67.

## Utilisation des fonctions d’extension et des paramètres FileMaker XSLT

Les fonctions d’extension FileMaker XSLT sont définies dans l’espace de nom `fmxls`. Veillez à inclure une déclaration de l’espace de nom `fmxls` dans l’élément `<xsl:stylesheet>` au début d’une feuille de style XSLT. Consultez la section « A propos des espaces de nom et des préfixes dans les feuilles de style FileMaker XSLT », page 62.

Les fonctions d’extension FileMaker XSLT ont été conçues pour pouvoir être utilisées dans une feuille de style XSLT sous la forme d’un appel de fonction dans une instruction XPath. Les instructions XPath sont utilisées comme valeurs de l’attribut de sélection et de l’attribut de test dans de nombreux éléments XSLT.

Par exemple, supposons que vous souhaitiez vérifier l’en-tête User-Agent pour déterminer le navigateur utilisé. Pour ce faire, vous pouvez utiliser une variable contenant la valeur de l’en-tête User-Agent :

```
<xsl:variable name="user-agent" select="fmxls:get_header('User-Agent')"/>
```

Dans le cas des fonctions d’extension renvoyant une valeur, la valeur est retournée dans le type XSLT spécifié. La plupart des fonctions renvoient des chaînes, mais d’autres renvoient un ensemble de nœuds pouvant être traversés.

**Remarque** Cette section décrit les fonctions d’extension et les paramètres FileMaker XSLT et comporte quelques exemples. Pour obtenir d’autres exemples de chaque fonction, consultez la base de référence des fonctions d’extension FileMaker XSLT. Consultez la section « A propos de la base de référence des fonctions d’extension FileMaker XSLT », page 58.

## A propos des paramètres XSLT propres à FileMaker définis par le moteur de publication Web

Lors du traitement d'une requête, le moteur de publication Web définit dynamiquement les valeurs des paramètres XSLT propres à FileMaker ci-après. Vous pouvez utiliser les valeurs de ces paramètres dans votre feuille de style au moyen de l'élément `<xsl:param>`.

Paramètre XSLT propre à FileMaker	Pour plus d'informations, consultez la section
<code>&lt;xsl:param name="request-query"/&gt;</code>	« Accès aux demandes d'informations dans une requête » disponible dans la section suivante.
<code>&lt;xsl:param name="client-ip"/&gt;</code> <code>&lt;xsl:param name="client-user-name"/&gt;</code> <code>&lt;xsl:param name="client-password"/&gt;</code>	« Obtention d'informations client », page 68.
<code>&lt;xsl:param name="xml-base-uri"/&gt;</code>	« Utilisation du paramètre URI de base du moteur de publication Web », page 68.
<code>&lt;xsl:param name="authenticated-xml-base-uri"&gt;</code>	« Utilisation du paramètre URI de base authentifié », page 69.

## Accès aux demandes d'informations dans une requête

Vous pouvez utiliser un paramètre FileMaker XSLT pour accéder aux informations de requête dans une requête sous la forme de données URL ou HTML. Par exemple, vous avez la possibilité d'accéder aux informations de requête en cours pour déterminer l'emplacement actuel dans un jeu d'enregistrements trouvés, puis de créer des liens vers l'enregistrement précédent et suivant.

Le paramètre FileMaker XSLT ci-après permet d'accéder à la totalité des commandes de requête et des paramètres de requête utilisés pour les requêtes de données FileMaker XML par l'intermédiaire du moteur de publication Web :

```
<xsl:param name="request-query"/>
```

A l'exception des noms de rubrique, le moteur de publication Web renvoie tous les noms de commande et de paramètre de requête en minuscules. La mise en majuscules des noms de rubrique est conservée.

Un fragment de document XML est chargé dans le paramètre `-query` de la requête dans la grammaire suivante :

```
<!DOCTYPE query [
  <!ELEMENT query (parameter)*>
    <!ATTLIST query action CDATA #REQUIRED>
    <!ELEMENT parameter (#PCDATA)>
    <!ATTLIST parameter name CDATA #REQUIRED>
]
```

**Remarque** Les informations de requête sont définies dans l'espace de nom `fmq="http://www.filemaker.com/xml/query"`. Veillez à inclure une déclaration de l'espace de nom `fmq` dans l'élément `<xsl:stylesheet>` au début de votre feuille de style XSLT. Consultez la section « A propos des espaces de nom et des préfixes dans les feuilles de style FileMaker XSLT », page 62.

Par exemple, supposons que vous souhaitiez accéder aux commandes et aux paramètres de requête dans la requête suivante :

```
http://192.168.123.101/fmi/xsl/my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset&-token.1=abc123&-findall
```

Si vous insérez l'instruction `<xsl:param name="request-query" />` avant la section de modèle, le moteur de publication Web stocke le fragment de document XML suivant dans ce paramètre :

```
<query action="my_stylesheet.xml" xmlns="http://www.filemaker.com/xml/query">
  <parameter name="-db">products</parameter>
  <parameter name="-lay">sales</parameter>
  <parameter name="-grammar">fmresultset</parameter>
  <parameter name="-token.1">abc123</parameter>
  <parameter name="-findall"></parameter>
</query>
```

Vous pouvez ensuite utiliser le paramètre de requête `-query` pour accéder à la valeur d'un paramètre `-token` transmise dans une URL au moyen d'une expression XPath. Par exemple :

```
$request-query/fmq:query/fmq:parameter[@name = '-token.1']
```

## Obtention d'informations client

Les paramètres FileMaker XSLT ci-après vous permettent d'obtenir des informations sur les clients Web (adresse IP, nom d'utilisateur et mot de passe) à partir du moteur de publication Web :

```
<xsl:param name="client-ip"/>
<xsl:param name="client-user-name"/>
<xsl:param name="client-password">
```

Insérez ces instructions de paramètre dans votre feuille de style XSLT avant le début de l'élément `<xsl:template>`.

Ces paramètres fournissent les informations d'authentification de l'utilisateur Web au moment où une feuille de style est programmée pour charger des documents XML supplémentaires protégés par mot de passe. Consultez la section « Chargement de documents supplémentaires », page 69. L'utilisateur Web doit indiquer le nom d'utilisateur et le mot de passe initialement définis au moyen de la boîte de dialogue d'authentification de base HTTP. Consultez la section « Accès à une base de données protégée », page 22.

Pour plus d'informations et d'exemples d'utilisation de ces trois paramètres FileMaker XSLT, consultez la base de référence des fonctions d'extension FileMaker XSLT.

## Utilisation du paramètre URI de base du moteur de publication Web

Le moteur de publication Web définit le paramètre URI (Uniform Resource Identifier) de base comme étant l'hôte et le port sur lesquels le moteur de publication Web est installé. L'identificateur URI de base permet de résoudre les requêtes de données XML à partir de bases de données FileMaker en fonction de l'hôte du moteur de publication Web.

Pour accéder au paramètre URI de base du moteur de publication Web, insérez l'instruction ci-après dans votre feuille de style XSLT, avant le début de l'élément `<xsl:template>` :

```
<xsl:param name="xml-base-uri"/>
```

Vous pouvez ensuite utiliser le paramètre URI de base pour la feuille de style active par l'intermédiaire de la variable `$xml-base-uri` pour chaque nouvelle requête de données FileMaker XML. Par exemple, utilisez le paramètre URI de base dans la requête de données XML supplémentaires ci-après :

```
<xsl:variable name="layout_information" select="document(concat($xml-base-uri,'fmi/xml/FMPXMMLLAYOUT.xml?
-db=products&-lay=sales&-view'))" />
```

## Utilisation du paramètre URI de base authentifié

Le paramètre `authenticated-xml-base-uri` associe les fonctionnalités des paramètres `client-user-name` et `client-password` au paramètre `xml-base-uri` :

```
<xsl:param name="authenticated-xml-base-uri"/>
```

Utilisez ce paramètre pour charger un autre document XML protégé par mot de passe et requérant les mêmes nom d'utilisateur et mot de passe que ceux spécifiés dans la requête d'origine en cours de traitement. Pour obtenir un exemple, consultez la section « Chargement de documents supplémentaires ».

Insérez cette instruction de paramètre dans votre feuille de style XSLT avant le début de l'élément `<xsl:template>`.

Si les valeurs des paramètres `client-user-name` et `client-password` ne sont pas vides, le paramètre `authenticated-xml-base-uri` prend la valeur suivante :

```
http://username:password@hostname:port
```

Si les valeurs des paramètres `client-user-name` et `client-password` sont vides, le paramètre `authenticated-xml-base-uri` prend la même valeur que le paramètre `xml-base-uri`.

## Chargement de documents supplémentaires

Pour charger un document XML supplémentaire pendant le traitement d'une feuille de style XSLT, utilisez la fonction XSLT `standard document()` avec un identificateur URI pour le document XML. La fonction `document()` renvoie les données XML requises sous la forme d'un ensemble de nœuds pouvant être stockés dans un élément `<xsl:variable>`.

Pour charger un document XML contenant des données issues d'une base de données FileMaker, utilisez la fonction `document()` avec les commandes et les paramètres de requête FileMaker. Par exemple :

```
<xsl:variable name="other-data" select="document(concat($xml-base-uri,'/fmi/xml/FMPXMMLAYOUT.xml?
-db=products&-lay=sales&-view'))" />
```

Pour charger un autre document XML protégé par mot de passe et requérant les mêmes nom d'utilisateur et mot de passe que ceux spécifiés dans la requête d'origine en cours de traitement, utilisez le paramètre `authenticated-xml-base-uri`. Ce paramètre spécifie les nom d'utilisateur et mot de passe dans l'identificateur URI transmis à la fonction `document()`.

Par exemple :

```
<xsl:variable name="other-data" select="document(concat($authenticated-xml-base-uri,
'/fmi/xml/FMPXMMLAYOUT.xml?-db=products&-lay=sales&-view'))"/>
```

Pour charger un document XML protégé par mot de passe et requérant un nom d'utilisateur et un mot de passe différents de ceux spécifiés dans la requête parente, utilisez la syntaxe ci-après pour spécifier le nom d'utilisateur et le mot de passe dans l'identificateur URI transmis à la fonction `document()` :

```
http://username:password@hostname/path?querystring
```

Pour charger un document XML non basé sur une base de données FileMaker, utilisez la fonction `document()` sans les commandes ni les paramètres de requête FileMaker. Par exemple :

```
<xsl:variable name="other-data" select="document('http://server.company.com/data.xml')" />
```

Si vous utilisez la fonction `document()` avec un URL relatif, le moteur de publication Web tente de charger le document XML à partir du système de fichiers local à l'emplacement relatif de la feuille de style. Par exemple, supposons qu'une feuille de style située dans le sous-dossier `mystylesheets` du dossier `xslt-template-files` contienne la fonction `document()` ci-après avec une URL relative :

```
<xsl:variable name="mydoc" select="document('mystylesheets/mydoc.xml')" />
```

Le moteur de publication Web tentera de charger le document `mydoc.xml` depuis le sous-dossier `mystylesheetsdu` du dossier `xslt-template-files` dans le système de fichiers local.

**Remarque** Lorsque vous utilisez le paramètre URI de base du moteur de publication Web, ce dernier prend uniquement en charge HTTP. Lorsque vous chargez un document à partir d'un serveur externe, le moteur de publication Web prend en charge HTTP et HTTPS.

## Utilisation des informations de modèle d'une base de données dans une feuille de style

Vous pouvez intégrer les informations de modèle d'une base de données FileMaker dans une feuille de style en demandant ces informations au moyen de la grammaire FMPXMMLAYOUT, puis en les chargeant dans une variable par l'intermédiaire de la fonction XSLT `document()` :

```
<xsl:variable name="layout" select="document(concat($xml-base-uri,'fmi/xml/FMPXMMLAYOUT.xml?–view'))" />
```

Par exemple, supposons que vous souhaitiez créer un menu déroulant pour une rubrique appelée **Color** contenant des valeurs issues d'une liste de valeurs à deux rubriques intitulée **shirts**, définie dans un modèle de base de données FileMaker. Supposons que la première rubrique de la liste de valeurs à deux rubriques stocke le numéro d'ID de la couleur (par exemple 100) et que la seconde rubrique stocke le nom de la couleur (par exemple Light Green). Voici comment utiliser la fonction `document()` pour charger les informations du modèle dans une variable XSLT, en même temps que l'attribut `DISPLAY` pour afficher la valeur de la seconde rubrique dans une liste de valeurs à deux rubriques :

```
<xsl:variable name="layout" select="document(concat($xml-base-uri,'fmi/xml/FMPXMMLAYOUT.xml?–db=products
&–lay=sales&–view'))" />
<select size="1">
  <xsl:attribute name="name">Color</xsl:attribute>
  <option value="">Select One...</option>
  <xsl:for-each select="$layout/fmi:FMPXMMLAYOUT/fmi:VALUELISTS/fmi:VALUELIST[@NAME = 'shirts']/
fmi:VALUE">
    <option>
      <xsl:attribute name="value"><xsl:value-of select="."/></xsl:attribute>
      <xsl:value-of select="@DISPLAY"/>
    </option>
  </xsl:for-each>
</select>
```

## Utilisation de la fonction de mise du contenu en mémoire tampon

Lorsque la fonction de mise du contenu en mémoire tampon est désactivée, le moteur de publication Web retourne les résultats d'une conversion XSLT directement au client. La fonction de mise du contenu en mémoire tampon est toujours désactivée, à moins que vous ne l'activiez explicitement. Si vous activez cette fonction, le moteur de publication Web stocke le contenu converti jusqu'à la fin de la totalité de la conversion.

La fonction de mise du contenu en mémoire tampon est requise pour les feuilles de style XSLT manipulant des en-têtes. Les en-têtes étant écrits avant le corps de la réponse, ce dernier doit être stocké en mémoire tampon pour que les informations d'en-tête ajoutées puissent être incluses.

Il existe quatre fonctions d'extension FileMaker nécessitant la mise en mémoire tampon des résultats de conversion XSLT :

- `fmxslt:create_session()` : Consultez la section « Utilisation des fonctions d'extension de session », page 72.

- `fmxml:set_header()` : Consultez la section « Utilisation des fonctions d'en-tête », page 75.
- `fmxml:set_status_code()` : Consultez la section « Utilisation des fonctions d'en-tête », page 75.
- `fmxml:set_cookie()` : Consultez la section « Utilisation des fonctions d'extension de cookie », page 76.

Pour que ces fonctions d'extension FileMaker fonctionnent correctement, vous devez insérer l'instruction de traitement XSLT ci-après dans le document de premier niveau pour la requête :

```
<?xslt-cwp-buffer buffer-content="true"?>
```

**Important** Si vous utilisez une feuille de style de base contenant une autre feuille de style, la feuille de base doit inclure l'instruction de traitement `<?xslt-cwp-buffer?>`. Cette instruction est ignorée si elle est utilisée dans une feuille de style incluse dans une autre feuille.

L'utilisation de cette instruction de traitement pour mettre la réponse en mémoire tampon offre l'avantage de permettre au moteur de publication Web de déterminer la longueur de la réponse et de définir l'en-tête Content-Length dans cette dernière. La mise en mémoire tampon de la réponse peut réduire les performances du moteur de publication Web.

## Utilisation de sessions du moteur de publication Web pour stocker des informations d'une requête à l'autre

Vous pouvez utiliser les sessions côté serveur du moteur de publication Web pour suivre et stocker tout type d'informations entre deux requêtes. Les sessions vous permettent de créer une application Web pouvant conserver des états en utilisant des éléments d'information persistants arbitraires entre les requêtes. Par exemple, des informations client utilisateur entrées sur la première page d'un formulaire peuvent être stockées dans une session, puis réutilisées pour remplir les valeurs d'une autre page du formulaire.

Par défaut, le moteur de publication Web utilise un cookie pour stocker l'ID de la session. Pour prendre en charge les clients n'autorisant pas les cookies, vous pouvez utiliser la fonction `fmxml:session_encode_url()` pour ajouter l'ID de session à l'URL. Pour garantir une parfaite compatibilité dans tous les types de situations, il est recommandé de coder toutes les URL inscrites sur la page avec la fonction `fmxml:session_encode_url()`. Cette fonction insère dans votre URL un paramètre précédé d'un point-virgule intitulé `jsessionid`, correspondant à l'identificateur de la session associée d'un client donné.

Par exemple, au lieu d'insérer sur une page le lien ci-après :

```
<a href="my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset&-findall">hyperlinked text</a>
```

vous devez coder tous les liens d'une page de la façon suivante :

```
<a href="{fmxml:session_encode_url('my_stylesheet.xml?-db=products&-lay=sales&-grammar=fmresultset&-findall')}">hyperlinked text</a>
```

Si le client n'autorise pas les cookies, la page inclut :

```
<a href="my_stylesheet.xml;jsessionid=<session id>?-db=products&-lay=sales&-grammar=fmresultset&-findall">hyperlinked text</a>
```

Si le moteur de publication Web détecte que le client autorise les cookies, la fonction `fmxml:session_encode_url()` stocke l'ID de session dans un cookie et non dans l'URL.

**Remarque** Les informations de session ne sont pas conservées après le redémarrage du moteur de publication Web.



## Utilisation des fonctions d'extension de session

Utilisez les fonctions d'extension de session ci-après pour manipuler les variables de session. Un objet de session peut stocker une chaîne, un nombre, une valeur booléenne ou un ensemble de nœuds. L'utilisation d'un ensemble de nœuds vous permet de créer une structure de données en XML, puis de la stocker entre les requêtes dans l'objet de session.

Fonction d'extension de session	Résultat	Description
fmxls:session_exists(String session-name)	booléen	Recherche l'existence d'une session portant le nom spécifié.
fmxls:create_session(String session-name)	booléen	Crée une session portant le nom indiqué et présentant le délai d'expiration par défaut défini au moyen de l'Admin Console. Consultez l'Aide de FileMaker Server. <b>Remarque</b> Cette fonction requiert l'instruction de traitement <code>&lt;?xslt-cwp-buffer?&gt;</code> . Consultez la section « Utilisation de la fonction de mise du contenu en mémoire tampon », page 70.
fmxls:invalidate_session(String session-name)	booléen	Déclenche l'arrivée à expiration immédiate de la session.
fmxls:set_session_timeout(String session-name, Number timeout)	booléen	Définit le délai d'expiration de la session en secondes. Le délai d'expiration par défaut des sessions est défini au moyen de l'Admin Console.
fmxls:session_encode_url(String url)	chaîne	Code une URL avec l'ID de la session si le client ne prend pas en charge les cookies ; dans le cas contraire, cette fonction renvoie une URL d'entrée.
fmxls:set_session_object(String session-name, String name, Object value)	objet XSLT (nombre, chaîne, valeur booléenne ou ensemble de -nœuds)	Stocke un objet XSLT (nombre, chaîne, valeur booléenne ou ensemble de nœuds) sous une session, réutilisable par la suite au moyen de la fonction fmxls:get_session_object(). Cette fonction renvoie également l'objet précédemment stocké sous le nom d'objet de session spécifié. Si aucun objet n'a été stocké sous ce nom, la fonction renvoie un objet Null. <b>Remarque</b> La fonction d'extension set_session_object() ne permet de stocker que des valeurs de type chaîne ; elle interprète tous les objets reçus comme des chaînes. Si la conversion de l'objet en chaîne est impossible, aucune valeur n'est alors stockée dans la session et le code d'erreur de la fonction d'extension est défini sur 10100 (erreur de session inconnue). Si vous tentez de définir un objet de session en utilisant une chaîne nulle ou vide, vous obtiendrez également le code d'erreur 10100 (erreur de session inconnue). Pour effacer une variable de session, supprimez-la de la session en utilisant la fonction remove_session_object().
fmxls:get_session_object(String session-name, String name)	objet XSLT	Récupère un objet XSLT à partir de la session.
fmxls:remove_session_object(String session-name, String name)	objet XSLT	Renvoie puis supprime un objet XSLT provenant de la session.



Voici un exemple de création d'une session et de stockage d'une couleur favorite dans la session :

```
<xsl:variable name="session">
  <xsl:choose>
    <xsl:when test="not (fmxs:session_exists(string($session-name)))">
      <xsl:value-of select="fmxs:create_session(string($session-name))"/>
    </xsl:when>
    <xsl:otherwise>true</xsl:otherwise>
  </xsl:choose>
</xsl:variable>
<xsl:variable name="favorite-color" select="fmxs:set_session_object(string($session-name), 'favorite-color', string($color))"/>
```

### Important

- Pour vous assurer que les utilisateurs sont déconnectés d'une base de données après avoir terminé une session, utilisez la fonction `fmxs:invalidate_session()` pour déclencher l'arrivée à expiration immédiate de la session.
- Si vous utilisez des rubriques de type Global ou un script définissant ou modifiant un état, vous devez utiliser l'Admin Console pour activer l'option Sessions de base de données XSLT pour le moteur de publication Web. Dans le cas contraire, les valeurs des rubriques Global et les états ne seront pas conservés entre les requêtes. Consultez l'Aide de FileMaker Server.
- Si vous accédez à un autre fichier de base de données à l'aide des sessions du moteur de publication Web, les valeurs des rubriques de type Global ne sont pas préservées. Le moteur de publication Web ferme le premier fichier avant d'ouvrir le second. Une autre possibilité consiste pour vous à accéder aux données depuis le second fichier de base de données en utilisant un modèle dans le premier fichier de base de données.

## Envoi d'emails à partir du moteur de publication Web

Vous pouvez utiliser le moteur de publication Web pour générer des emails, ce qui se révèle d'une grande utilité pour les solutions Web personnalisées. Pour envoyer un email à partir du moteur de publication Web, utilisez l'une des trois fonctions d'extension `fmxs:send_email()` ci-après dans une feuille de style XSLT. Ces fonctions vous permettent d'envoyer un ou plusieurs messages distincts. Les fonctions `fmxs:send_email()` étant contenues dans la feuille de style XSLT du moteur de publication Web au niveau du serveur, un client ne peut pas envoyer de emails non autorisés à l'aide du moteur de publication Web.

Fonction d'extension de messagerie électronique	Résultat	Description
<code>fmxs:send_email(String smtpFields, String corps)</code>	booléen	Envoie un email en texte brut d'une longueur quelconque à partir du moteur de publication Web en utilisant le codage de texte par défaut de ce dernier pour les emails
<code>fmxs:send_email(String smtpFields, String body, String encoding)</code>	booléen	Envoie un email en texte brut d'une longueur quelconque en utilisant l'un des codages de texte suivants : US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift_JIS, UTF-8. Pour plus d'informations sur ces codages, consultez la section « Définition du codage du texte dans les requêtes », page 63.
<code>fmxs:send_email(String smtpFields, String xsltFile, Node xml, boolean includeImages)</code>	booléen	Envoie un email HTML en utilisant le codage spécifié par l'attribut de codage de l'élément <code>&lt;xsl:output&gt;</code> dans la feuille de style. Si l'attribut de codage n'est pas inclus dans l'élément <code>&lt;xsl:output&gt;</code> , le moteur de publication Web utilise son codage de texte par défaut pour les emails.

## Remarques

- Dans chacune des trois formes de la fonction `fmxml:send_email()`, le paramètre `smtpFields` est une chaîne présentant un codage URL d'une longueur quelconque contenant les informations d'adresse et d'objet au format ci-après, reposant sur RFC 2368, le protocole URL `mailto` :

`username@host?name1=value1&name2=value2...`

où `username@host` désigne un destinataire. Les paires nom/valeur peuvent être spécifiées dans n'importe quel ordre et sont définies de la façon suivante :

- `from=username@host` (ne doit apparaître qu'une seule fois). La rubrique `from` doit être spécifiée.
- `to=username@host`. Utilisez cette paire nom/valeur pour indiquer des destinataires supplémentaires.
- `reply-to=username@host` (ne peut apparaître qu'une seule fois)
- `cc=username@host`
- `bcc=username@host`
- `subject=string` (ne peut apparaître qu'une seule fois)

Si les rubriques `from`, `reply-to` ou `subject` sont spécifiées plusieurs fois, l'email n'est pas envoyé, la valeur `false()` est renvoyée par la fonction et le code d'état d'erreur approprié est défini.

- Le moteur de publication Web vérifie la syntaxe de toutes les adresses électroniques indiquées. Ces dernières doivent être spécifiées sous la forme suivante :

`user@host.tld` ou `quoted identifier entre guillemets<user@host.tld>`

où `tld` désigne un quelconque domaine de premier niveau tel que `com` ou `net`. Si l'une des rubriques contient une adresse électronique incorrecte, l'email n'est pas envoyé et le code d'état d'erreur approprié est défini.

- Les différentes valeurs du paramètre `smtpFields`, telles que l'objet, doivent correspondre à une chaîne présentant un codage URL. Par exemple, le caractère `&` doit être spécifié sous la forme `&amp;` et les espaces vides doivent être indiqués sous la forme `%20`. La totalité de la chaîne correspondant au paramètre `smtpFields` doit présenter un codage XML. (Consultez l'exemple situé à la fin de cette section.)
- Pour chacune de ces fonctions, la valeur `true()` est renvoyée si l'email est correctement envoyé ; dans le cas contraire, la valeur `false()` est renvoyée.
- Pour les emails en français, le moteur de publication Web utilise le codage de texte par défaut initial ISO-8859-1. Vous pouvez modifier ce paramétrage au moyen de l'Admin Console. Consultez l'Aide de FileMaker Server.
- La fonction `fmxml:send_email(String smtpFields, String xsltFile, Node xml, boolean includelimages)` envoie un email constitué de données XML qui est traité par la feuille de style d'email que vous spécifiez dans cette fonction.
  - Pour le paramètre `xsltFile`, indiquez le nom de la feuille de style d'email en entrant une URL relative pour le fichier principal de feuille de style de traitement correspondant à la requête.
  - Pour le paramètre `xml`, indiquez le nœud parent des données XML que vous souhaitez utiliser avec la feuille de style d'email. Pour envoyer un email en utilisant les données XML affichées dans le navigateur, fournissez simplement l'expression XPath pour la racine du document : `"/`". Si vous souhaitez utiliser un autre document XML, commencez par charger ce dernier avec la fonction `document()`, puis transmettez le document dans la fonction `fmxml:send_email()`.

- Pour le paramètre `includelImages`, indiquez la valeur booléenne `true()` pour que le moteur de publication Web contienne toutes les images spécifiées en tant que pièces jointes dans le marquage HTML de l'email. Ce paramètre inclut aussi bien les images de base de données FileMaker que les images non associées à des bases de données provenant d'autres emplacements. Le moteur de publication Web change les URL des images pour faire référence aux pièces jointes. L'utilisation de fichiers image volumineux ou nombreux peut ralentir les performances. Si vous indiquez la valeur `false ()` pour ce paramètre, le moteur de publication Web ne change pas les URL des images. Si les URL sont absolues, le client de messagerie tentera de charger les images à partir du serveur Web.

Voici un exemple d'utilisation de la fonction `fmxls:send_email(String smtpFieldsd, String xsltFile, Node xml, boolean includelImages)` dans une instruction XPath, par exemple dans l'élément `<xsl:if>` :

```
fmxls:send_email('tom_jones@company.com?subject=project%20status&from=john_smith@company.com
&cc=jane_doe@company.com','my_mail_template.xml',/, true())
```

Pour plus d'informations sur le paramétrage du moteur de publication Web pour la connexion à un serveur SMTP, consultez l'aide de FileMaker Server.

## Utilisation des fonctions d'en-tête

Vous pouvez utiliser la fonction `fmxls:get_header()` pour lire les informations des en-têtes de réponse et de requête HTTP, ainsi que la fonction `fmxls:set_header()` pour écrire des informations dans les en-têtes. Ces fonctions sont utiles si le client peut utiliser les informations d'en-tête pour récupérer des informations du serveur Web ou que vous avez besoin de définir un en-tête HTTP pour d'autres raisons.

Fonction d'extension d'en-tête	Résultat	Description
<code>fmxls:get_header(String name)</code>	chaîne	Renvoie la valeur d'en-tête spécifiée
<code>fmxls:set_header(String name, String value)</code>	vide	Définit la valeur d'en-tête spécifiée
<code>fmxls:set_status_code(Number status-code)</code>	vide	Définit le code d'état HTTP

### Remarques

- Le nom utilisé dans les fonctions `fmxls:get_header()` et `fmxls:set_header()`, ainsi que la valeur indiquée dans la fonction `fmxls:set_header()`, correspondent à une chaîne d'une longueur quelconque.
- Les fonctions `fmxls:set_header()` et `fmxls:set_status_code()` requièrent l'instruction de traitement `<?xslt-cwp-buffer?>`. Consultez la section « Utilisation de la fonction de mise du contenu en mémoire tampon », page 70.

L'exemple qui suit illustre comment définir la valeur de l'en-tête. Supposons que vous utilisiez une feuille de style pour sortir une vCard. Lorsqu'un navigateur tente de charger la page de feuille de style, il existe un risque que ce navigateur interprète le fichier .xsl comme une feuille de style et non comme une vCard. Si vous utilisez l'en-tête intitulé `Content-Disposition`, vous pouvez indiquer qu'il existe une pièce jointe portant l'extension .vcf.

```
<xsl:value-of select="fmxls:set_header('Content-Disposition','attachment;filename=test.vcf')"/>
```

## Utilisation des fonctions d'extension de cookie

Les fonctions d'extension de cookie vous permettent de récupérer ou de définir des cookies stockés dans le navigateur Web du client.

Fonction d'extension de cookie	Résultat	Description
<code>fmxls: get_cookie(String name)</code>	ensemble de nœuds	Renvoie l'ensemble de nœuds COOKIE contenant le nom de cookie spécifié.
<code>fmxls: get_cookies()</code>	ensemble de nœuds	Renvoie l'ensemble de nœuds COOKIE avec tous les cookies fournis par le client.
<code>fmxls: set_cookie(String name, String value)</code>	vide	Stocke le cookie spécifié dans le navigateur du client avec la valeur indiquée.
<code>fmxls: set_cookie(String name, String value, Number expires, String path, String domain)</code>	vide	Stocke le cookie spécifié dans le navigateur du client avec toutes les valeurs disponibles pour un cookie. Le paramètre Expires indique le nombre de secondes avant l'arrivée à expiration du cookie.

### Remarques

- Les fonctions `fmxls: get_cookie()` et `fmxls: get_cookies()` renvoient un ensemble de nœuds présentés dans la structure suivante :
 

```
<!ELEMENT cookies (cookie)*>
      <!ATTLIST cookie xmlns CDATA #FIXED "http://www.filemaker.com/xml/cookie">
      <!ELEMENT cookie (#PCDATA)>
      <!ATTLIST cookie name CDATA #REQUIRED>
```
- L'espace de nom XML pour l'ensemble de nœuds de cookies est "http://www.filemaker.com/xml/cookie". Vous devez déclarer l'espace de nom et indiquez un préfixe pour l'espace de nom.
- Toutes les valeurs de paramètre indiquées dans les fonctions `fmxls: set_cookie` doivent être valides ; dans le cas contraire, le navigateur Web ignorera les requêtes `fmxls: set_cookie`.
- Dans toutes les fonctions de cookie, la longueur des paramètres de chaîne est illimitée.
- Les deux formes de la fonction `fmxls: set_cookie()` requièrent l'instruction de traitement `<?xslt-cwp-buffer?>`. Consultez la section « Utilisation de la fonction de mise du contenu en mémoire tampon », page 70.

### Exemple : get\_cookie

L'exemple suivant récupère un cookie appelé `preferences` ainsi que sa valeur :

```
<xsl:variable name="pref_cookie" select="fmxls: get_cookie('preferences')"/>
<xsl:value-of select="concat('Cookie Name = ', $pref_cookie/fmc:cookies/fmc:cookie/@name)"/> <br/>
<xsl:value-of select="concat('Cookie Value = ', $pref_cookie/fmc:cookies/fmc:cookie)"/>
```

### Exemple : set\_cookie

Voici un exemple de définition d'un cookie avec toutes les valeurs :

```
<xsl:variable name="storing_cookie" select="fmxls: set_cookie ('text1', 'text2', 1800, 'my_text', 'my.company.com')"/>
```

## Utilisation des fonctions d'extension de manipulation de chaînes

Les fonctions d'extension de manipulation de chaînes vous permettent de modifier le codage de chaînes de n'importe quelle longueur.

Fonction d'extension de manipulation de chaînes	Résultat	Description
fmxsit:break_encode(String value)	chaîne	<p>Renvoie une chaîne présentant un codage de césures HTML. Les caractères tels que &amp; (Et commercial) sont remplacés par &amp;amp;. Les caractères de début de nouvelle ligne tels que les sauts de ligne et les retours chariot sont remplacés par &lt;br/&gt;. Cette fonction ne fonctionne que si l'attribut disable-output-escaping des éléments &lt;xsl:value-of&gt; et &lt;xsl:text&gt; est défini sur yes (disable-output-escaping="yes").</p> <p><b>Remarque</b> Pour inclure un saut de ligne ou un retour chariot dans la chaîne à laquelle s'applique la fonction fmxsit:break_encode(), vous devez utiliser les caractères d'échappement ci-après dans la chaîne : "&amp;#10;" (pour un saut de ligne) ou "&amp;#13;" (pour un retour chariot). Vous ne pouvez pas inclure de saut de ligne ni de retour chariot dans la chaîne en appuyant sur la touche Retour de votre éditeur de texte.</p>
fmxsit:html_encode(String value)	chaîne	Renvoie une chaîne présentant un codage HTML ; les caractères tels que & (Et commercial) sont remplacés par &amp;.
fmxsit:url_encode(String value)	chaîne	Renvoie une chaîne présentant un codage URL. Le codage de l'URL est utilisé pour transmettre les caractères sur Internet, particulièrement les URL. Par exemple, le & (Et commercial) codé en URL correspond à %26. Si un caractère réservé est utilisé dans votre href, utilisez cette fonction pour coder votre chaîne en URL.
fmxsit:url_encode(String value, String encoding)	chaîne	<p>Renvoie une chaîne à codage URL utilisant le codage de caractères que vous spécifiez pour le paramètre <b>codage</b>, à savoir : US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift_JIS ou UTF-8.</p> <p>Utilisez cette fonction lorsque vous savez qu'un serveur Web attendra un autre codage de caractères que celui utilisé dans votre feuille de style active. Par exemple, supposons que la page d'accueil de votre site Web soit affichée en UTF-8, mais que les utilisateurs puissent cliquer sur un lien pour accéder à une page en japonais. Si la requête inclut des caractères japonais et que les pages en japonais utilisent le codage Shift_JIS, il est préférable de coder la chaîne en Shift_JIS.</p>
fmxsit:url_decode(String value)	chaîne	Renvoie une chaîne ayant fait l'objet d'un décodage URL à partir d'une chaîne URL précédemment codée.
fmxsit:url_decode(String value, String encoding)	chaîne	<p>Renvoie une chaîne ayant fait l'objet d'un décodage URL et utilisant le codage de caractères que vous spécifiez pour le paramètre <b>codage</b>, à savoir : US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift_JIS ou UTF-8.</p> <p>Utilisez cette fonction lorsque vous devez spécifier le codage de caractères utilisé dans une chaîne codée en URL afin de décoder correctement cette chaîne. Par exemple, même si votre site Web utilise le codage ISO-8859-1, il est possible que des utilisateurs soumettent un formulaire utilisant un autre codage de caractères.</p>

## Comparaison de chaînes au moyen d'expressions régulières Perl 5

La fonction d'extension `fmxml:regex_contains()` vous permet de comparer des chaînes au moyen d'expressions régulières Perl 5. Une comparaison par expressions régulières est une fonction de correspondance de texte avancée vous permettant de déterminer si une chaîne correspond à un schéma spécifié. La syntaxe de cette fonction est la suivante :

```
fmxml:regex_contains(String input, String pattern)
```

où `input` est une chaîne et `pattern` une expression régulière Perl 5. Pour plus d'informations sur la syntaxe des expressions régulières Perl 5, consultez le site [www.perldoc.com](http://www.perldoc.com). La fonction `fmxml:regex_contains()` renvoie une valeur booléenne.

Utilisez cette fonction si vous avez besoin d'une fonction de manipulation de chaînes plus élaborée que celle fournie par le langage XSLT standard. Par exemple, vous pouvez déterminer si une valeur de rubrique contient un numéro de téléphone ou une adresse électronique valide en comparant la chaîne à une expression régulière Perl 5.

Voici un exemple d'utilisation de cette fonction pour déterminer si une valeur de rubrique contient des adresses électroniques correctement structurées :

```
<xsl:variable name="email" select="'foo@bar.com'"/>
<xsl:if test="fmxml:regex_contains($email, '^\\w+([\\w-\\.])*\\w+((-[\\w+])|(\\w*))\\.\\[a-z]{2,3}$')">Valid Email</xsl:if>
```

Si le moteur de publication Web ne parvient pas à analyser le schéma, l'état d'erreur est configuré sur le code d'erreur 10311. Consultez la section « Numéros des codes d'erreur des fonctions d'extension XSLT de FileMaker », page 123.

## Recherche de valeurs dans une rubrique formatée sous forme de case à cocher

La fonction d'extension ci-après vous permet de déterminer si une valeur spécifique appartenant à une liste de valeurs de case à cocher est stockée dans une rubrique de la base de données FileMaker :

```
fmxml:contains_checkbox_value(String valueString, String valueListEntry)
```

où `valueString` est une expression XPath spécifiant la rubrique et `valueListEntry` indique la valeur recherchée.

Si la valeur spécifiée est stockée dans la rubrique, cette fonction booléenne renvoie la valeur `true()`. Dans le cas contraire, elle retourne la valeur `false()`. Vous pouvez utiliser cette fonction pour déterminer si l'attribut `checked` d'un formulaire HTML doit ou non afficher une case à cocher comme étant activée.

Par exemple, supposons que l'une des rubriques d'un modèle de base de données FileMaker comporte les options de case à cocher suivantes :

- ☐ Rouge
- ☐ Bleu
- ☐ Vert
- ☐ Petit
- ☐ Moyen
- ☐ Grand

Si un utilisateur ne sélectionne que la valeur Rouge, la rubrique doit contenir la chaîne Rouge. Pour déterminer si la rubrique contient la chaîne Bleu, vous pourriez utiliser l'appel de fonction suivant :

```
fmxml:contains_checkbox_value(<field value node>,'Blue')
```

où `<field value node>` est l'expression XPath pour l'élément `<data>` correspondant à la rubrique de type case à cocher. Dans cet exemple, la fonction renverrait la valeur `false`.

Une application courante de cette fonction consiste à afficher la liste des valeurs de case à cocher dans une page Web et à activer les cases à cocher de la page Web qui sont sélectionnées dans la base de données. Par exemple, les instructions HTML et XSLT ci-après créent un ensemble de cases à cocher pour une rubrique intitulée **style** en utilisant une liste de valeurs appelée **color\_size** :

```
<xsl:variable name="field-value" select="fmrs:field[@name='style']/fmrs:data" />
<xsl:for-each select="$valueLists[@NAME = 'color_size']/fml:VALUE">
  <input type="checkbox">
    <xsl:attribute name="name">style</xsl:attribute>
    <xsl:attribute name="value"><xsl:value-of select="."/></xsl:attribute>
    <xsl:if test="fmxls:contains_checkbox_value($field-value,.)">
      <xsl:attribute name="checked">checked</xsl:attribute>
    </xsl:if>
  </input><xsl:value-of select="."/><br/>
</xsl:for-each>
```

Les instructions HTML et XSLT de cet exemple afficheraient les cases à cocher ci-après sur une page Web, avec les valeurs **Red** et **Medium** sélectionnées :

[x] Rouge  
 [ ] Bleu  
 [ ] Vert  
 [ ] Petit  
 [x] Moyen  
 [ ] Grand

## Utilisation des fonctions d'extension de date, d'heure et de jour

Vous disposez de fonctions d'extension vous permettant de récupérer la date, l'heure ou le jour en cours et de comparer deux dates, deux heures ou deux jours.

Les fonctions dans la table suivante utilisent les formats fm sans tenir compte des paramètres régionaux. Les formats fm sont MM/jj/aaaa pour la date, HH:mm:ss pour l'heure et MM/jj/aaaa HH:mm:ss pour l'horodatage.

Pour réarranger les valeurs des résultats dans un format différent ou préférentiel, utilisez les fonctions de calcul ou JavaScript.

Fonction d'extension de date, d'heure ou de jour	Résultat	Description
fmxls:get_date()	chaîne	Renvoie la date du jour au format fm.
fmxls:get_date(String format)	chaîne	Renvoie la date du jour au format spécifié. Entrez la chaîne short, long ou fm pour le paramètre format.
fmxls:get_time()	chaîne	Renvoie l'heure du jour au format fm.
fmxls:get_time(String format)	chaîne	Renvoie l'heure en cours au format spécifié. Entrez la chaîne short, long ou fm pour le paramètre format.
fmxls:get_day()	chaîne	Renvoie le jour en cours au format court.
fmxls:get_day(String format)	chaîne	Renvoie le jour en cours au format spécifié. Entrez la chaîne short ou long pour le paramètre format.
fmxls:get_fm_date_format()	chaîne	Renvoie la chaîne de formatage pour le format de date fm : « MM/jj/aaaa »



Fonction d'extension de date, d'heure ou de jour	Résultat	Description
fmxls: get_short_date_format()	chaîne	Renvoie la chaîne de formatage pour le format de date court : « MM/jj/aa »
fmxls: get_long_date_format()	chaîne	Renvoie la chaîne de formatage pour le format de date long : « MMMM j aaaa »
fmxls: get_fm_time_format()	chaîne	Renvoie la chaîne de formatage pour le format d'heure fm : « HH:mm:ss »
fmxls: get_fm_timestamp_format()	chaîne	Renvoie la chaîne de formatage pour le format d'horodatage fm : « MM/jj/aaaa HH:mm:ss »
fmxls: get_short_time_format()	chaîne	Renvoie la chaîne de formatage pour le format d'heure court : « h:mm a »
fmxls: get_long_time_format()	chaîne	Renvoie la chaîne de formatage pour le format d'heure long : « HH:mm:ss z »
fmxls: get_short_day_format()	chaîne	Renvoie la chaîne de formatage pour le format de jour court : « EEE »
fmxls: get_long_day_format()	chaîne	Renvoie la chaîne de formatage pour le format de jour long : « EEEE »
fmxls: compare_date(String date1, String date2)	nombre	Cette fonction compare deux valeurs de date. Elle renvoie une valeur négative si <b>date1</b> est antérieure à <b>date2</b> . Elle renvoie une valeur positive si <b>date1</b> est postérieure à <b>date2</b> . Elle renvoie la valeur 0 si <b>date1</b> est identique à <b>date2</b> . Les deux dates doivent être spécifiées au format de date fm.
fmxls: compare_time(String time1, String time2)	nombre	Cette fonction compare deux valeurs d'heure. Elle renvoie une valeur négative si <b>time1</b> est antérieure à <b>time2</b> . Elle renvoie une valeur positive si <b>time1</b> est postérieure à <b>time2</b> . Elle renvoie la valeur 0 si <b>time1</b> est identique à <b>time2</b> . Les deux heures doivent être spécifiées au format d'heure fm.
fmxls: compare_day(String day1, String day2)	nombre	Cette fonction compare deux valeurs de jour. Elle renvoie une valeur négative si <b>day1</b> est antérieur à <b>day2</b> . Elle renvoie une valeur positive si <b>day1</b> est postérieur à <b>day2</b> . Elle renvoie la valeur 0 si <b>day1</b> est identique à <b>day2</b> . Les deux jours doivent être spécifiés au format de jour court.

Les fonctions du tableau ci-après utilisent des chaînes de formatage de date personnalisées spécifiant un format de date et d'heure. Consultez la section « Détails concernant les chaînes de format de date et d'heure ».

Fonction d'extension de date, d'heure ou de jour	Résultat	Description
fmxls: get_datetime(String dateFormat)	chaîne	Renvoie la date et l'heure en cours en utilisant les chaînes de format de date et d'heure.
fmxls: convert_datetime(String oldFormat, String newFormat, String date)	chaîne	Convertit la date spécifiée dans l'ancien format ( <b>oldFormat</b> ) indiqué en chaîne utilisant le nouveau format ( <b>newFormat</b> ) défini. Les chaînes <b>oldFormat</b> et <b>newFormat</b> doivent être spécifiées en utilisant les chaînes de format de date et d'heure.
fmxls: compare_datetime(String dateFormat1, String dateFormat2, String date1, String date2)	nombre	Cette fonction compare <b>date1</b> à <b>date2</b> en décodant ces dates selon leurs formats de date respectifs. Elle renvoie une valeur négative si <b>date1</b> est antérieure à <b>date2</b> . Elle renvoie une valeur positive si <b>date1</b> est postérieure à <b>date2</b> . Elle renvoie la valeur 0 si <b>date1</b> est identique à <b>date2</b> . Les chaînes <b>dateFormat1</b> et <b>dateFormat2</b> doivent être spécifiées en utilisant les chaînes de format de date et d'heure.



## Détails concernant les chaînes de format de date et d'heure

Les formats de date et d'heure sont spécifiés par des chaînes de schéma de date et d'heure. Dans les chaînes de schéma de date et d'heure, les lettres non indiquées entre guillemets de A à Z et de a à z sont interprétées comme des lettres de schéma représentant les composants d'une chaîne de date ou d'heure.

Les lettres de schéma définies sont les suivantes (tous les autres caractères de A à Z et de a à z sont réservés) :

Lettre	Composant de date ou d'heure	PowerPoint	Exemples
G	Indicateur de l'ère	Texte	ap. J-C.
y	Année	Année	1996; 96
M	Mois de l'année	Mois	Juillet ; Juil. 07
w	Semaine de l'année	Nombre	27
W	Semaine du mois	Nombre	2
D	Jour de l'année	Nombre	189
d	Jour du mois	Nombre	10
F	Jour de la semaine dans le mois	Nombre	2
C	Jour de la semaine	Texte	mercredi ; mer
a	Indicateur am/pm	Texte	PM
A	Heure du jour (0-23)	Nombre	0
k	Heure du jour (1-24)	Nombre	24
K	Heure au format am/pm (0-11)	Nombre	0
h	Heure au format am/pm (1-12)	Nombre	12
m	Minutes	Nombre	30
s	Secondes	Nombre	55
S	Millisecondes	Nombre	978
z	Fuseau horaire	Fuseau horaire général	Heure avancée du Pacifique ; PDT ; TMG-08:00
Z	Fuseau horaire	Fuseau horaire RFC 822	-0800

Les lettres de schéma sont généralement répétées car leur nombre détermine la présentation exacte :

- **Texte** : Pour le formatage, si le nombre de lettres de schéma est au moins égal à quatre, la forme complète est utilisée ; dans le cas contraire, la forme courte ou abrégée est appliquée si elle existe. Pour l'analyse, les deux formes sont acceptées, indépendamment du nombre de lettres de schéma.
- **Nombre** : Pour le formatage, le nombre de lettres de schéma est le nombre de chiffres minimal, et les nombres plus courts sont remplis avec des zéros pour atteindre ce nombre. Pour l'analyse, le nombre de lettres de schéma est ignoré, à moins qu'il ne soit requis pour séparer deux rubriques adjacentes.

- **Année** : Pour le formatage, si le nombre de lettres de schéma est égal à deux, l'année est tronquée après deux chiffres ; dans le cas contraire, elle est interprétée comme une valeur.

Pour l'analyse, si le nombre de lettres de schéma est supérieur à 2, l'année est interprétée de façon littérale, quel que soit le nombre de chiffres. Ainsi, si vous utilisez le schéma « MM/jj/aaaa », l'analyse de la chaîne « 01/11/12 » produit le résultat 11 janvier, 12 ap. J.-C.

Pour l'analyse avec le schéma d'année abrégé (« y » ou « yy »), l'année abrégée doit être interprétée par rapport au siècle en ajustant les dates comme étant comprises dans les 80 années qui précèdent et les 20 années qui succèdent à la date de création de l'instance de format de date. Par exemple, si vous utilisez un schéma MM/jj/aa et une instance de format de date créée le 1er janvier 1997, l'analyse de la chaîne « 01/11/12 » produit le résultat 11 janvier 2012, tandis que la chaîne « 05/04/64 » serait interprétée comme correspondant au 4 mai 1964. Pendant l'analyse, seules les chaînes composées d'exactly deux chiffres sont analysées dans le siècle par défaut. Toute autre chaîne numérique, telle qu'une chaîne d'un seul chiffre, de trois chiffres ou davantage, ou de deux chiffres n'incluant pas que des chiffres (par exemple, « -1 »), est interprétée de façon littérale. Ainsi, l'analyse des chaînes « 01/02/3 » ou « 01/02/003 » utilisant le même schéma produit le résultat 2 janvier 3 ap. J.-C. De la même façon, la chaîne « 01/02/-3 » est analysée comme correspondant au 2 janvier 4 av. J.-C.

- **Mois** : Si le nombre de lettres de schéma est supérieur ou égal à trois, le mois est interprété comme du texte ; dans le cas contraire, il est interprété comme un nombre.
- **Fuseau horaire général** : Les fuseaux horaires sont interprétés comme du texte s'ils comportent des noms. Pour les fuseaux horaires représentant une valeur de correction TMG, la syntaxe utilisée est la suivante :
  - GMTOffsetTimeZone. *TMG Signe Heures:Minutes*
  - Signe. + ou -
  - Heures. *Chiffre* ou *Chiffre Chiffre*
  - Minutes. *Chiffre Chiffre*
  - Chiffre. L'une des valeurs suivantes : 0 1 2 3 4 5 6 7 8 9

Le paramètre *Heures* doit être compris entre 0 et 23, et le paramètre *Minutes* entre 00 et 59. Le format est indépendant de la langue utilisée et les chiffres doivent être extraits du bloc Latin de base du standard Unicode.

Pour l'analyse, les fuseaux horaires RFC 822 sont également acceptés.

- **Fuseau horaire RFC 822** : Pour le formatage, le format de fuseau horaire RFC 822 à 4 chiffres est utilisé :
  - RFC822TimeZone. *Signe HeuresDeuxChiffres Minutes*
  - TwoDigitHours. *Chiffre Chiffre*

Le paramètre *TwoDigitHours* doit être compris entre 00 et 23. Les autres définitions sont considérées comme des fuseaux horaires généraux.

Pour l'analyse, les fuseaux horaires généraux sont également acceptés.

Les exemples ci-après présentent la façon dont les schémas de date et d'heure sont interprétés aux Etats-Unis. Les date et heure utilisées sont : 04-07-2001 12:08:56 heure locale des Etats-Unis, fuseau horaire Heure du Pacifique.

Schéma de date et d'heure	Résultat
"yyyy.MM.dd G 'à' HH:mm:ss z"	2001.07.04 ap. J.-C. à 12:08:56 PDT
"EEE, MMM d, ' 'yy"	Mer, 4 jul, '01
"HH:mm"	12:08 PM
"hh 'o' 'clock' a, zzzz"	12 heures PM, heure avancée du Pacifique (PDT)
"K:mm a, z"	0:08 PM, PDT
"yyyyy.MMMMM.dd GGG hh:mm aaa"	02001.juillet.04 ap. J.-C. 12:08 PM
"EEE, d MMM yyyy HH:mm:ss Z"	Mer, 4 juil. 2001 12:08:56 -0700
"yyMMddHHmmssZ"	010704120856-0700

Copyright 2003 Sun Microsystems, Inc. Reproduit avec autorisation.

## Vérification de l'état d'erreur des fonctions d'extension

Vous pouvez utiliser la fonction d'extension ci-après dans une feuille de style XSLT pour rechercher l'état d'erreur de la dernière fonction d'extension FileMaker XSLT utilisée et pour gérer les erreurs survenant pendant le traitement de vos pages :

`fmxls:check_error_status()`

Lorsque vous utilisez la fonction `fmxls:check_error_status()`, le moteur de publication Web renvoie la valeur de code d'erreur active de la dernière fonction invoquée sous la forme d'un nombre, puis redéfinit l'état d'erreur sur 0 (Pas d'erreur). Pour plus d'informations sur les valeurs de code d'erreur, consultez la section « Numéros des codes d'erreur des fonctions d'extension XSLT de FileMaker », page 123.

## Utilisation de la fonction de journalisation

L'élément XSLT standard `<xsl:message>` vous permet de consigner les entrées de journal dans le fichier journal d'applications du moteur de publication Web. Consultez la section « Utilisation du journal d'application du moteur de publication Web », page 90.

## Utilisation du traitement des langages de script côté serveur

Le transformateur XSLT intégré au moteur de publication Web prend en charge le traitement des langages de script côté serveur. En conséquence, vous pouvez utiliser JavaScript pour développer des fonctions ayant vos propres extensions et qui pourront être directement appelées depuis une feuille de style XSLT.

Deux bibliothèques Java sont installées pour activer cette fonctionnalité :

- `bsf.jar` : cette bibliothèque permet au transformateur XSLT de se connecter aux langages de script.
- `js.jsr` : cette bibliothèque est une mise en œuvre JavaScript complète provenant du projet Mozilla.

Grâce à ces bibliothèques, vous pouvez créer des fonctions ayant vos propres extensions au sein du code de votre feuille de style XSLT. Ces fonctions d'extension peuvent mettre en œuvre n'importe quelle logique de script et sont à préférer à XSLT et à Xpath pour réaliser des fonctions logiques.

Vous trouverez des informations plus détaillées sur la prise en charge des extensions du transformateur XSLT sur le site Web des extensions Apache Xalan à l'adresse suivante :

<http://xml.apache.org/xalan-j/extensions.html>

## Définition d'une fonction d'extension

Pour définir une fonction d'extension au sein de votre feuille de style :

### 1. Définissez l'espace de nom de l'extension.

Ajoutez l'espace de nom `xalan` pour informer le transformateur XSLT de prendre en charge les composants d'extension, en fournissant le nom de votre propre espace de nom de fonction d'extension. L'exemple suivant utilise `fmp-ex` comme préfixe de l'espace nom de la fonction d'extension.

```
<xsl:stylesheet version="1.0"
xmlns:xsl=http://www.w3.org/1999/XSL/Transform
xmlns:xalan=http://xml.apache.org/xslt
xmlns:fmp-ex="ext1"
exclude-result-prefixes="xsl xalan fmp-ex">
```

### 2. Définissez le composant d'extension et les fonctions d'extension avec le code qui met réellement en œuvre votre fonction d'extension.

```
<xalan:component prefix="fmp-ex" functions="getValueColor">
<xalan:script lang="javascript">
    function getValueColor(value) {
        if (value > 0)
            return ("009900");
        else
            return ("CC0000");
    }
</xalan:script>
</xalan:component>
```

Cet exemple renvoie une valeur couleur en fonction d'une valeur d'entrée. Si la valeur d'entrée est supérieure à 0, la couleur renvoyée est vert ("`#009900`") ; dans le cas contraire, si la valeur est inférieure à 0, la couleur renvoyée est rouge ("`#CC0000`").

**Remarque** L'élément `<xalan:component>` doit être un enfant de l'élément `<xsl:stylesheet>`.

### 3. Utilisez la fonction d'extension au sein de la feuille de style.

Les exemples suivants montrent comment appeler une fonction d'extension utilisant une instruction Xpath.

Ce premier exemple devrait définir la couleur de la police sur vert ("`#009900`").

```
<font color="{fmp-ex:getValueColor(50)}">The value is 50</font>
```

Ce second exemple devrait définir la couleur de la police sur rouge ("`#CC0000`").

```
<font color="{fmp-ex:getValueColor(-500)}">The value is -500</font>
```

## Exemple de fonction d'extension

La fonction JavaScript simple utilisée dans le processus ci-dessus aurait pu être mise en œuvre en utilisant une instruction `<xsl:choose>`. Cependant, la vraie force de l'utilisation d'une extension de script est que vous pouvez créer une fonction qui ne peut pas être mise en œuvre dans XSLT ou Xpath seul.

Par exemple, supposons que vous souhaitez créer un site portail intranet pour votre entreprise et que vous voulez y inclure des informations sur les cours boursiers actuels. Il existe des listes de cours en XML mais ne sont généralement accessibles qu'au moyen d'une licence commerciale. Toutefois, vous pouvez télécharger les données boursières depuis le site Web de Yahoo!, dans un document au format CSV (Comma Separated Values). La fonction `document()` XPath peut importer le contenu provenant de sources XML mais vous devez pour cela convertir le contenu CSV en XML. Une solution consiste à utiliser JavaScript pour télécharger les informations sur les cours boursiers au format CSV, à analyser le fichier, puis à extraire les données.

Cette URL montre la syntaxe de récupération d'un cours boursier depuis le site Web de Yahoo!, sous forme de fichier CSV :

`http://quote.yahoo.com/d/quotes.csv?s=<ticker>&f=l1gh&e=.csv`

où `<ticker>` représente le code de l'action dont vous tentez de récupérer les données.

Les données renvoyées se présentent sous la forme de trois nombres séparés par des virgules, par exemple :

`31.79,31.17,32.12`

où la première valeur correspond au dernier cours d'échange, la seconde valeur, au cours le plus bas, et enfin la troisième valeur, le cours le plus haut.

L'exemple ci-dessous montre une fonction d'extension XSLT JavaScript qui récupère un cours boursier depuis le site Web de Yahoo! et le rend accessible via une fonction Xpath :

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet
  exclude-result-prefixes="xsl fmxslt fmrs xalan fmp-ex"
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"
  xmlns:fmxslt="xalan://com.fmi.xslt.ExtensionFunctions"
  xmlns:xalan="http://xml.apache.org/xslt"
  xmlns:fmp-ex="ext1"
>

<?xslt-cwp-query params="-grammar=fmresultset&-process" ?>
<xsl:output method="html"/>
<xalan:component prefix="fmp-ex" functions="include get_quote" >
<xalan:script lang="javascript">
  function include(url) {
    var dest = new java.net.URL(url);
    var dis = new java.io.DataInputStream(dest.openStream());
    var res = "";
    while ((line = dis.readLine()) != null)
    {
```

```

        res += line + java.lang.System.getProperty("line.separator");
    }
    dis.close();
    return res;
}

function get_quote(ticker) {
    url = "http://quote.yahoo.com/d/quotes.csv?s=" +
        "+ticker+"&f=l1gh&e=.csv";
    csv_file = include(url);
    var str_tokenizer = new java.util.StringTokenizer(csv_file, ',');
    // the first token is the last trade price
    var last = str_tokenizer.nextToken();
    return last;
}
</xalan:script>
</xalan:component>

<xsl:template match="/fmrs:fmresultset">
    <html>
        <body>
            <font size="2" face="verdana, arial">
                Apple Stock Price: <xsl:value-of select="fmp-ex:get_quote('AAPL')"/>
            </font>
        </body>
    </html>
</xsl:template>

</xsl:stylesheet>

```

Lorsque le moteur de publication Web traite cette feuille de style, il effectue une requête auprès du site Web de Yahoo! pour obtenir les informations relatives au cours boursier. La fonction `get_quote()` analyse les données du cours et renvoie les données à la feuille de style. Le résultat transformé s'affiche dans le navigateur.

# Chapitre 7

## *Stockage, test et suivi d'un site*

Ce chapitre propose des instructions pour stocker et tester un site en publication Web personnalisée avant son déploiement dans un environnement de production. Vous y trouverez également des instructions relatives à l'utilisation des fichiers journaux permettant d'assurer le suivi du site en phase de test ou après le déploiement.

### Stockage d'un site de Publication Web personnalisée

Avant de pouvoir tester correctement votre site, vous devez copier ou déplacer les fichiers nécessaires à leur emplacement correct sur le(s) serveur(s) de stockage.

Pour stocker votre site et le préparer pour les tests :

1. Suivez toutes les étapes exposées dans chapitre 3, « Préparation des bases de données pour la Publication Web personnalisée ».
2. Assurez-vous que XSLT et XML ont été activés et correctement paramétrés dans l'Admin Console de FileMaker Server.

**Remarque** Vous trouverez des instructions complémentaires dans l'aide de FileMaker Server.

3. Vérifiez le bon fonctionnement du serveur Web et du moteur de publication Web.
4. Copiez ou déplacez les feuilles de style XSLT sur la machine où se trouve le moteur de publication Web.

Copiez ou déplacez les feuilles de style XSLT dans le répertoire suivant de la machine où se trouve le moteur de publication Web :

- Apache (Mac OS) : /Library/FileMaker Server/Web Publishing/xslt-template-files
  - IIS (Windows) : <disque>:\Program Files\FileMaker\FileMaker Server\Web Publishing\xslt-template-files
- où <lecteur> correspond au lecteur principal servant au démarrage de votre système.

**Remarque** Vous pouvez également placer les feuilles de style dans un dossier provisoire ou dans le sous-dossier xslt-template-files.

5. Copiez ou déplacez les éventuels objets multimédia référencés sur la machine contenant le serveur Web.

Si une rubrique de base de données Multimédia stocke une référence à un fichier et non le fichier lui-même, l'objet multimédia référencé doit alors être stocké dans le dossier FileMaker Pro Web lorsque l'enregistrement est créé ou modifié. Pour stocker votre site, vous devez copier ou déplacer les objets multimédia référencés dans un dossier possédant le même emplacement relatif dans le dossier racine du logiciel de serveur Web.

**Remarque** Si le fichier de base de données est correctement hébergé accessible sur FileMaker Server et si les rubriques Multimédia stockent les fichiers proprement dits dans la base de données FileMaker, toute action sur le contenu de la rubrique Multimédia est alors inutile.

6. Utilisez la syntaxe URL suivante pour effectuer une requête ou traiter une feuille de style XSLT et générer la page HTML associée :

```
<scheme>://<host>[:<port>]/fmi/xsl/<path>/<stylesheet>.xsl[?<query string>]
```

où :

- <scheme> correspond au protocole HTTP ou HTTPS.
- <hôte> correspond à l'adresse IP ou au nom de domaine de l'ordinateur hôte sur lequel le serveur Web est installé.
- <port> est facultatif et spécifie le port d'écoute du serveur Web. Si aucun port n'est spécifié, le port par défaut du protocole est utilisé (port 80 pour HTTP ou port 443 pour HTTPS).
- <path> est facultatif et spécifie le ou les dossiers présents dans le répertoire xslt-template-files dans lequel se situe la feuille de style XSLT.
- <stylesheet> correspond au nom de la feuille de style avec l'extension .xsl.
- <query string> peut correspondre à la combinaison d'une commande de requête et d'un ou de plusieurs paramètres de requête pour la Publication Web personnalisée avec XSLT.

Si la feuille de style spécifiée intègre une instruction de traitement `<?xslt-cwp-query ?>`, la commande de requête statique et les paramètres prennent le pas sur les commandes de requête ou les paramètres correspondants dans la chaîne de requête URL. Si vous référencez une feuille de style `home.xsl` par le biais de XSLT Site Assistant, vous n'avez pas besoin d'intégrer une chaîne de requête. Consultez la section annexe A, « Noms corrects utilisés dans les chaînes de requête » pour plus d'informations sur les chaînes de requête.

La syntaxe URL est sensible à la casse, notamment les noms des commandes de requête et les paramètres, à l'exception des parties de la chaîne de requête. Les adresses URL sont, en majorité, en minuscules. Par exemple, si vous avez copié vos feuilles de style (y compris une feuille de style `home.xsl`) dans le sous-dossier `my_templates` du dossier `xslt-template-files`, vous pouvez utiliser l'adresse URL suivante pour effectuer une requête de traitement des feuilles de style :

```
http://192.168.123.101/fmi/xsl/my_templates/home.xsl
```

**Remarque** Le moteur de publication Web ne permet pas aux utilisateurs Web d'afficher la source des feuilles de style XSLT installées dans le dossier `xslt-template-files`. Lorsqu'un utilisateur Web envoie une requête pour traiter une feuille de style, le moteur de publication Web envoie uniquement au navigateur Web les pages HTML résultant des feuilles de style de XSLT Site Assistant.

## Test d'un site de Publication Web personnalisée

Avant d'informer les utilisateurs de la disponibilité de votre site de Publication Web personnalisée, vérifiez s'il se présente et fonctionne comme prévu.

- Testez les fonctions telles que la recherche, l'ajout, la suppression et le tri d'enregistrements avec différents jeux de comptes et de privilèges d'accès.
- Vérifiez si les jeux de privilèges fonctionnent comme prévu en vous connectant à différents comptes. Assurez-vous que des utilisateurs non autorisés ne peuvent pas accéder à vos données ou les modifier.
- Contrôlez tous les scripts pour vérifier que leur effet est conforme aux attentes. Consultez la section « Scripts FileMaker et Publication Web personnalisée », page 24 pour obtenir des informations sur la conception de scripts adaptés au Web.



- Testez votre site avec différents systèmes d'exploitation et navigateurs Web.

**Remarque** Si vous ne possédez pas de connexion réseau et si vous avez installé FileMaker Server en choisissant un déploiement sur une seule machine — le serveur Web, le moteur de publication Web et le serveur de bases de données se trouvant sur un même ordinateur — vous pouvez tester votre site de Publication Web personnalisée en utilisant `http://localhost/` ou `http://127.0.0.1/` dans l'URL. Pour plus d'informations sur la syntaxe d'URL, consultez les sections « A propos de la syntaxe d'URL pour les données XML et les objets de type Multimédia », page 41, et « A propos de la syntaxe d'URL des feuilles de style FileMaker XSLT », page 59.

## Exemples de feuilles de style permettant de tester les sorties XML

Voici deux exemples de feuilles de style XSLT utiles pour tester le résultat XML.

- L'exemple de feuille de style suivant renvoie les données XML requises sans aucune transformation. Cette feuille de style est utile pour afficher les données XML réelles utilisées par le moteur de publication Web.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset">
  <xsl:output method="xml"/>
  <xsl:template match="/">
    <xsl:copy-of select="."/>
  </xsl:template>
</xsl:stylesheet>
```

- Pour le débogage d'une feuille de style, vous pouvez utiliser l'exemple suivant d'une balise HTML `<textarea>` pour afficher le document source XML accessible via la feuille de style, dans une zone de texte défilante. Sur la même page, vous pouvez comparer les résultats XSLT transformés aux documents XML source, avant la transformation.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fmrs="http://www.filemaker.com/xml/fmresultset">
  <xsl:output method="html"/>
<html>
  <body>
    <xsl:template match="/fmrs:fmresultset">
      <textarea rows="20" cols="100">
        <xsl:copy-of select="."/>
      </textarea><br/>
    </xsl:template>
  </body>
</html>
</xsl:stylesheet>
```

## Suivi de votre site

Vous pouvez utiliser les types de fichiers de consignation suivants pour surveiller votre site de Publication Web personnalisée et réunir des informations au sujet des utilisateurs Web qui visitent votre site :

- Journaux d'erreurs et d'accès au serveur Web
- Journal d'application du moteur de publication Web
- Journal d'erreurs du module de serveur Web
- Journaux d'accès internes du noyau de publication Web

### Utilisation des journaux d'erreurs et d'accès au serveur Web

Apache (Mac OS) : Le serveur Web Apache génère un fichier journal des accès et un fichier journal des erreurs. Le fichier journal des accès Apache, qui répond par défaut au format standard de fichier journal du W3C, consigne toutes les requêtes HTTP reçues par le serveur Web. Le journal des erreurs Apache consigne les problèmes survenus dans le cadre du traitement des requêtes HTTP. Pour plus d'informations sur ces fichiers journaux, consultez la documentation du serveur Web Apache.

IIS (Windows) : Le serveur Web Microsoft IIS génère un fichier journal des accès, mais affiche les erreurs dans l'Observateur d'événements de Windows au lieu de les consigner dans un fichier journal. Le fichier journal des accès, qui répond par défaut au format étendu de fichier journal du W3C, consigne toutes les requêtes HTTP reçues par le serveur Web. Vous pouvez aussi utiliser le format standard de fichier journal du W3C pour le journal des accès. Pour plus d'informations, consultez la documentation du serveur Web Microsoft IIS.

Pour obtenir des informations sur les formats standard et étendu de fichier journal du W3C, consultez le site Web du World Wide Web Consortium, à l'adresse [www.w3.org](http://www.w3.org).

### Utilisation du journal d'application du moteur de publication Web

Par défaut, le moteur de publication Web génère un fichier journal d'application, consignait des informations sur les erreurs, les scripts et les connexions d'utilisateurs du moteur de publication Web.

- Les informations du journal des erreurs décrivent toute erreur inhabituelle survenue dans le moteur de publication Web. Les erreurs habituelles signalées à l'utilisateur Web, telles que Base de données fermée, ne sont pas consignées.
- Les informations du journal des scripts décrivent toute erreur générée quand des utilisateurs Web exécutent des scripts. Les instructions de script ignorées en raison de leur incompatibilité avec le Web y sont par exemple recensées.
- Le journal de connexion des utilisateurs contient des messages générés par l'élément XSLT `<xsl:message>` dans les feuilles de style XSLT. Dès qu'un utilisateur Web accède à votre feuille de style XSLT, les informations que vous avez incluses dans un élément `<xsl:message>` sont consignées dans le fichier journal d'application. Consultez la section chapitre 6, « Conception de feuilles de style FileMaker XSLT ».

Le journal d'application se nomme `pe_application_log.txt` et se situe dans le sous-dossier Logs du dossier FileMaker Server sur l'hôte du moteur de publication Web.

Le fichier `pe_application_log.txt` est généré si l'une des options de consignations suivante est activée dans le moteur de publication Web :

Option de consignation activée	Informations enregistrées dans <code>pe_application_log.txt</code>
Journalisation des erreurs	Toute erreur inhabituelle survenue dans le moteur de publication Web. Les erreurs habituelles signalées à l'utilisateur Web, telles que Base de données fermée, ne sont pas consignées.
Journalisation des scripts	Toute erreur générée quand des utilisateurs Web exécutent des scripts. Les instructions de script ignorées en raison de leur incompatibilité avec le Web y sont par exemple recensées.
Journalisation des utilisateurs	Messages générés lorsque les utilisateurs Web accèdent à votre solution de publication Web personnalisée.

Ces options de journalisation, sont toutes trois activées par défaut. Pour plus d'informations sur le paramétrage de ces options via l'Admin Console, consultez l'aide de FileMaker Server.

**Remarque** Les entrées du journal d'application ne sont pas supprimées automatiquement et le fichier peut devenir très volumineux avec le temps. Pour économiser de l'espace sur le disque dur de l'ordinateur hôte, archivez régulièrement le fichier journal d'application.

## Utilisation du journal des erreurs du module de serveur Web

Si le serveur Web n'est pas en mesure de se connecter au moteur de publication Web, le module de serveur Web génère un fichier journal consignait toutes les erreurs liées à son utilisation. Ce fichier se nomme `web_server_module_log.txt` et se situe dans le sous-dossier Logs du dossier FileMaker Server, sur l'hôte du serveur Web.

## Utilisation des journaux d'accès internes du noyau de publication Web

Le composant logiciel noyau de publication Web du moteur de publication Web génère deux fichiers journaux d'accès internes consignait tous les accès du noyau de publication Web.

- Le journal d'accès `wpc_access_log.txt` recense toutes les requêtes d'utilisateurs finaux visant à générer des données XML et à utiliser la fonction de publication Web instantanée de FileMaker Server. Ces requêtes sont directement orientées vers le noyau de publication Web par le serveur Web.
- Le journal d'accès `pe_internal_access_log.txt` recense toutes les requêtes XML internes que le composant logiciel XSLT-CWP du moteur de publication Web effectue durant le traitement des requêtes XSLT. Ces requêtes sont redirigées en interne au sein du moteur de publication Web, depuis le composant logiciel XSLT-CWP vers le noyau de publication Web.

Ces fichiers journaux se trouvent dans le dossier Logs du dossier FileMaker Server de l'hôte moteur de publication Web.

Les journaux d'accès interne sont générés si l'option de Journalisation des accès est activée dans le moteur de publication Web. Le paramètre par défaut est activé. Pour plus d'informations sur le paramétrage de l'option Journalisation des accès de l'Admin Console, consultez l'aide de FileMaker Server.



# Annexe A

## *Noms corrects utilisés dans les chaînes de requête*

Cette annexe décrit les noms valides des commandes et des paramètres de requête que vous pouvez utiliser dans une chaîne de requête lorsque vous accédez à des données FileMaker à l'aide du moteur de publication Web.

### A propos des commandes et des paramètres de requête

Voici une liste complète des noms de commandes de requête et des noms de paramètres de requête :

Noms des commandes de requête	Noms des paramètres de requête
–dbnames (Voir, page 98.)	–db (Voir, page 102.)
–delete (Voir, page 98.)	–encoding (XSLT uniquement) (Voir, page 102.)
–dup (Voir, page 98.)	–field (Voir, page 102.)
–edit (Voir, page 99.)	fieldname (Voir, page 103.)
–find, –findall, –findany (Voir, page 99.)	fieldname.op (Voir, page 104.)
–findquery (Voir, page 100.)	–grammar (XSLT uniquement) (Voir, page 105.)
–layoutnames (Voir, page 100.)	–lay (Voir, page 105.)
–new (Voir, page 100.)	–lay.response (Voir, page 105.)
–process (XSLT uniquement) (Voir, page 101.)	–lop (Voir, page 105.)
–scriptnames (Voir, page 101.)	–max (Voir, page 106.)
–view (Voir, page 101.)	–modid (Voir, page 106.)
	–query (Voir, page 106.)
	–recid (Voir, page 107.)
	–relatedsets.filter (Voir, page 108.)
	–relatedsets.max (Voir, page 108.)
	–script (Voir, page 108.)
	–script.param (Voir, page 109.)
	–script.prefind (Voir, page 109.)
	–script.prefind.param (Voir, page 110.)
	–script.presort (Voir, page 110.)
	–script.presort.param (Voir, page 110.)
	–skip (Voir, page 111.)
	–sortfield.[1-9] (Voir, page 111.)
	–sortorder.[1-9] (Voir, page 111.)
	–stylehref (Voir, page 112.)
	–styletype (Voir, page 112.)
	–token.[string] (XSLT uniquement) (Voir, page 113.)

**Important** Le paramètre –lay servant à spécifier un modèle est nécessaire avec toutes les commandes de requête, à l'exception de –dbnames, –layoutnames, –scriptnames et –process (requêtes XSLT uniquement).

## Consignes d'utilisation des commandes et des paramètres de requête

Lorsque vous utilisez des commandes et des paramètres de requête dans une chaîne de requête, gardez les consignes suivantes à l'esprit :

- Une chaîne de requête doit contenir une seule commande de requête, ni plus, ni moins. Une chaîne de requête peut par exemple contenir `–new` pour ajouter un nouvel enregistrement, mais `–new` et `–edit` ne peuvent être contenus dans la même chaîne de requête.
- La majorité des commandes de requête nécessitent divers paramètres de requêtes correspondants dans la chaîne de requête. Par exemple, toutes les commandes de requête, à l'exception de `–dbnames` et `–process` ont besoin du paramètre `–db`, qui spécifie la base de données sur laquelle effectuer la recherche. Consultez le tableau des paramètres requis dans la section « Utilisation de chaînes de requête FileMaker pour rechercher des données XML », page 52.
- Pour les paramètres de requête et les noms de rubrique, indiquez la valeur particulière que vous souhaitez utiliser, par exemple `–db=employees`. Pour les commandes de requête, n'indiquez pas le signe `=` ou une valeur après le nom de la commande, par exemple `–findall`.
- Les noms de commandes et paramètres de requête doivent être indiqués en minuscules, par exemple `–delete` ou `–lay`.
- Les noms de bases de données, de modèles et de rubriques utilisés dans les chaînes de requête ne tiennent pas compte de la casse, vous pouvez par exemple indiquer `–lay=mylayout` pour spécifier `MyLayout`.

**Remarque** Les noms de rubrique et de base de données utilisés dans les instructions XSLT en dehors des chaînes de requête sont sensibles à la casse et doivent donc correspondre aux noms exacts employés dans la base de données. Par exemple, dans l'instruction suivante :

```
<xsl:value-of select="fms:field[@name='LastName']"/>
```

la référence de rubrique `LastName` doit correspondre exactement au nom de la rubrique `LastName` dans la base de données.

- Les noms de rubriques peuvent contenir des points, sauf dans les cas suivants :
  - Le point ne peut pas être suivi d'un chiffre. Par exemple, `myfield.9` est un nom de rubrique invalide.
  - Le point ne peut pas être suivi de la chaîne de texte `op` (les deux lettres « `op` »). Par exemple, `myfield.op` est un nom de rubrique invalide.
  - Le point ne peut pas être suivi de la chaîne de texte `global` (le mot « `global` »). Par exemple, `myfield.global` est un nom de fichier invalide.

Les noms de rubrique contenant l'une de ces exceptions ne sont pas accessibles via XML ou XSLT lorsque vous utilisez une requête HTTP. Ces constructions sont destinées aux ID d'enregistrements, comme décrit dans la section « A propos de la syntaxe d'un nom de rubrique entièrement qualifié, » ci-dessous.

- Pour la commande `–find`, la valeur d'une rubrique ne tient pas compte de la casse. Par exemple, vous pouvez aussi bien écrire `Field1=Blue` que `Field1=blue`. Pour les commandes `–new` and `–edit`, la casse que vous utilisez dans la valeur d'une rubrique est conservée et stockée dans la base de données exactement comme vous la spécifiez dans la chaîne de requête. Par exemple, `LastName=Doe`.

## A propos du document Référence des chaînes de requête FileMaker

Cette version comprend une base de données FileMaker appelée *Query Strings Reference.fp7*. Elle contient de courtes descriptions et des exemples de toutes les commandes et de tous les paramètres de requête de FileMaker. Elle se trouve dans le répertoire suivant de toutes les machines de votre déploiement FileMaker Server (maîtres ou subordonnées) :

Mac :

/Library/FileMaker Server/Examples/XSLT

Windows :

<lecteur>:\Program Files\FileMaker\FileMaker Server\Examples\XSLT

Où : <lecteur> correspond au lecteur principal servant au démarrage de votre système.

## A propos de la syntaxe d'un nom de rubrique entièrement qualifié

Un nom de rubrique entièrement qualifié identifie une instance précise d'une rubrique. Dans la mesure où les rubriques possédant des noms communs peuvent être basées sur différentes tables, vous devez utiliser des noms entièrement qualifiés, dans certains cas, pour éviter les erreurs.

La syntaxe permettant de spécifier un nom de rubrique entièrement qualifié est la suivante :

table-name::field-name(repetition-number).record-id

où :

- **table-name** est le nom de la table qui contient la rubrique. Le nom de la table est uniquement requis si la rubrique ne se trouve pas dans la table sous-jacente du modèle spécifié dans la chaîne de requête.
- **field-name(repetition-number)**, qui est une valeur spécifique dans une rubrique multivaluée, est uniquement requis pour les rubriques multivaluées. Le numéro de valeur débute à 1. Par exemple, **field-name(2)** fait référence à la seconde valeur dans la rubrique multivaluée. Si vous n'indiquez pas de numéro de valeur pour une rubrique multivaluée, la première valeur de celui-ci est utilisée. Le numéro de valeur est requis pour les commandes de requête **–new** and **–edit** impliquant des rubriques multivaluées, mais n'est pas requis pour la commande **–find**.
- **record-id** est l'ID de l'enregistrement, qui est uniquement requis si vous utilisez une chaîne de requête pour ajouter ou modifier des enregistrements dans des rubriques de table externe. Consultez les sections « Ajout d'enregistrements à une table externe, » et « Modification d'enregistrements dans une table externe. ». Le paramètre **record-id** est requis pour les commandes de requête **–new** et **–edit** impliquant des rubriques de table externe, mais n'est pas requis pour la commande **–find**.

**Remarque** Pour être accessibles, les rubriques doivent être placées dans le modèle que vous spécifiez dans la chaîne de requête.

## Utilisation de commandes de requêtes avec des rubriques de table externe

Les sections suivantes décrivent le fonctionnement des commandes de requêtes avec des rubriques de table externe.

## Ajout d'enregistrements à une table externe

Pour ajouter un nouvel enregistrement à une table externe en même temps qu'un enregistrement parent, utilisez la commande de requête `–new` et effectuez les opérations suivantes dans une chaîne de requête :

- Utilisez le nom de rubrique entièrement qualifié pour la rubrique de table externe correspondante.
- Spécifiez 0 en guise d'ID d'enregistrement après le nom de la rubrique de table externe liée.
- Spécifiez au moins une des rubriques pour l'enregistrement parent, avant de spécifier la rubrique de table externe liée.
- Spécifiez les données pour la rubrique source (rubrique clé) dans l'enregistrement parent.

Par exemple, l'URL suivant ajoute simultanément un nouvel enregistrement Employee parent pour John Doe, et un nouvel enregistrement lié pour Jane dans la table externe. Le nom de la table liée est Dependents et le nom de la rubrique liée dans la table externe est Names. La rubrique source ID stocke un numéro d'identification de type Employee.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&FirstName=John&LastName=Doe
&ID=9756&Dependents::Names.0=Jane&–new
```

**Remarque** Vous ne pouvez ajouter qu'un enregistrement correspondant à une table externe par requête.

## Modification d'enregistrements dans une table externe

Pour modifier un ou plusieurs enregistrements dans une table externe, utilisez la commande `–edit` et un ID d'enregistrement pour spécifier l'enregistrement parent qui contient les enregistrements de table externe que vous souhaitez modifier. Spécifiez l'enregistrement de table externe spécifique à modifier en utilisant son ID d'enregistrement dans un nom de rubrique entièrement qualifié. Vous pouvez déterminer un ID d'enregistrement à partir de l'attribut d'ID d'enregistrement de l'élément `<record>` dans l'élément `<relatedset>` au sein des données XML. Consultez la section « Utilisation de la grammaire fmresultset », page 44.

Par exemple, l'URL suivant modifie un enregistrement dans une table externe, où l'enregistrement parent porte l'ID d'enregistrement 1001. Dependents est le nom de la table liée, Names est le nom de la rubrique liée dans la table externe et le chiffre 2 dans Names.2 est l'ID d'enregistrement d'un enregistrement de table externe.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–recid=1001
&Dependents::Names.2=Kevin&–edit
```

Voici un exemple illustrant comment utiliser une seule requête pour modifier plusieurs enregistrements de table externe par le biais de l'enregistrement parent :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–recid=1001
&Dependents::Names.2=Kevin&Dependents::Names.5=Susan&–edit
```

Vous pouvez également utiliser la commande `–edit` et spécifier 0 en guise d'ID d'enregistrement de table externe pour ajouter un nouvel enregistrement lié dans la table externe pour un enregistrement parent existant. Par exemple :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–recid=1001
&Dependents::Names.0=Timothy&–edit
```



## Suppression d'enregistrements dans une table externe

Pour supprimer des enregistrements de table externe, utilisez le paramètre `–delete.related` avec la commande `–edit` au lieu d'employer la commande `–delete`.

Par exemple, l'URL suivant supprime l'enregistrement 1001 de la table `employés`:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–recid=1001&–delete
```

En revanche, l'URL suivant supprime l'enregistrement de table externe dont l'ID est 3 de la table liée nommée `Dependents`, et dont l'ID d'enregistrement parent est 1001.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–recid=1001
&–delete.related=Dependents.3&–edit
```

Pour plus d'informations, consultez la section « `–delete.related` (suppression d'enregistrements externes), paramètre de requête », page 102.

## Requêtes de rubriques de table externe

Dans une solution comportant de nombreux enregistrements liés, l'interrogation et le tri des enregistrements externes peuvent demander beaucoup de temps. Pour limiter le nombre d'enregistrements et de lignes à afficher dans un ensemble lié, utilisez les paramètres `–relatedsets.filter` et `–relatedsets.max` avec les requêtes de recherche. Pour plus d'informations, consultez les sections « `–relatedsets.filter` (filtrage d'enregistrements de table externe), paramètre de requête », page 108 et « `–relatedsets.max` (limitation d'enregistrements de table externe), paramètre de requête », page 108.

## A propos de la syntaxe de spécification d'une rubrique de type Global

La syntaxe de spécification d'une rubrique de type Global est la suivante :

```
table-name::field-name(repetition-number).global
```

où `global` identifie une rubrique utilisant le stockage global. Pour obtenir des informations à propos de `nom-table` et `nom-rubrique(numéro-valeur)`, consultez la section « A propos de la syntaxe d'un nom de rubrique entièrement qualifié », page 95. Pour plus d'informations sur les rubriques de type Global, consultez l'aide FileMaker Pro.

Vous devez utiliser la syntaxe `.global` pour identifier une rubrique de type Global dans une chaîne de requête. Le moteur de publication Web définit les valeurs de paramètres pour les rubriques de type Global avant d'exécuter la commande de requête ou de définir toute autre valeur de paramètre dans la chaîne de requête. Pour les requêtes XML directes et les requêtes effectuées par le biais de feuilles de style XSLT n'utilisant pas de sessions, les valeurs globales expirent immédiatement après l'exécution de la requête. Pour les requêtes effectuées par l'intermédiaire d'une feuille de style XSLT utilisant des sessions, les valeurs globales persistent pendant la durée de la session définie dans la feuille de style ou jusqu'à ce qu'elles soient à nouveau modifiées par une autre requête.

Si vous n'utilisez pas la syntaxe `.global` pour identifier une rubrique de type Global dans une chaîne de requête, le moteur de publication Web traite la rubrique de type Global avec le reste de la chaîne de requête, sans définir au préalable la valeur de la rubrique de type Global.

Par exemple :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments
&Country.global=USA&-recid=1&-edit
```

**Important** Si vous utilisez une rubrique de type Global dans une feuille de style XSLT, vous devez utiliser l'Admin Console pour activer l'option **Sessions de base de données XSLT** pour le moteur de publication Web. Sinon, les valeurs des rubriques de type Global ne seront pas conservées d'une requête à une autre. Consultez l'Aide de FileMaker Server.

## Références des commandes de requête

Cette section contient des informations à propos des commandes de requête disponibles pour les requêtes XML et XSLT.

**Remarque** Seules les requêtes XSLT nécessitent le paramètre de requête `-grammar` pour toutes les commandes de requête ci-après.

### Commande de requête `-dbnames` (noms des bases de données)

Extrait le nom de toutes les bases de données hébergées par FileMaker Server et activées pour la Publication Web personnalisée avec XML ou XSLT.

**Paramètres de requête indispensables :** (aucun)

**Exemple :**

Pour extraire les noms de bases de données :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-dbnames
```

### Commande de requête `-delete` (supprimer l'enregistrement)

Supprime l'enregistrement indiqué par le paramètre `-recid`.

**Paramètres de requête indispensables :** `-db`, `-lay`, `-recid`

**Paramètres de requête facultatifs :** `-script`

**Exemple :**

Pour supprimer un enregistrement :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-recid=4&-delete
```

### Commande de requête `-dup` (dupliquer l'enregistrement)

Duplique l'enregistrement spécifié par `-recid`.

**Paramètres de requête indispensables :** `-db`, `-lay`, `-recid`

**Paramètres de requête facultatifs :** `-script`

**Exemple :**

Pour dupliquer l'enregistrement spécifié :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-recid=14&-dup
```

## Commande de requête –edit (modifier l’enregistrement)

Met à jour l’enregistrement indiqué par le paramètre –recid et renseigne les rubriques avec le contenu de n’importe quelle paire nom rubrique/valeur. Le paramètre –recid indique le nom de l’enregistrement qui doit être modifié.

**Paramètres de requête indispensables :** –db, –lay, –recid et un ou plusieurs noms de rubriques

**Paramètres de requête facultatifs :** –modid, –script

**Remarque** Pour plus d’informations sur la modification d’enregistrements dans une table externe, consultez la section « Modification d’enregistrements dans une table externe », page 96.

### Exemple :

Pour modifier un enregistrement :

<http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–recid=13&Country=USA&–edit>

## Commandes de requête –find, –findall, or –findany (rechercher les enregistrements)

Soumet une requête utilisant des critères de recherche spécifiques.

**Paramètres de requête indispensables :** –db, –lay

**Paramètres de requête facultatifs :** –recid, –lop, –op, –max, –skip, –sortorder, –sortfield, –script, –script.prefind, –script.presort, field name

### Exemples :

Pour chercher un enregistrement par son nom de rubrique :

<http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&Country=USA&–find>

**Remarque** Le fait de spécifier un nom de rubrique à plusieurs reprises dans une même requête n’est pas pris en charge ; FileMaker Server analyse toutes les valeurs mais utilise uniquement la dernière valeur analysée.

Pour chercher un enregistrement par son ID d’enregistrement :

<http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–recid=427&–find>

Pour chercher tous les enregistrements de la base de données, utilisez –findall :

<http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–findall>

Pour chercher un enregistrement de façon aléatoire, utilisez –findany :

<http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=family&–findany>

## **–findquery (recherche composée), commande de requête**

Soumet une requête de recherche avec des critères de recherche multiples ainsi que des exceptions.

**Paramètres de requête indispensables :** –db, –lay, –query

**Paramètres de requête facultatifs :** –max, –skip, –sortorder, –sortfield, –script, –script.prefind, –script.presort

### **Exemple :**

Rechercher les enregistrements des chats ou des chiens qui ne s'appellent pas « Fluffy ».

```
http://host/fmi/xml/fmresultset.xml?–db=vetclicnic&–lay=animals&–query=(q1);(q2);!(q3)
&–q1=typeofanimal&–q1.value=Cat&–q2=typeofanimal&–q2.value=Dog&–q3=name&–q3.value=Fluffy&–findquery
```

## **Utilisation de la commande –findquery pour les recherches composées**

Une instruction –findquery se compose de quatre parties, dans l'ordre suivant :

- Le paramètre de requête –query
- Les déclarations de demande de requête, composées de l'identificateur de requête et des opérations de demande.
- Le champ de recherche et les définitions de valeur pour chaque identificateur de requête.
- La commande –findquery, à la fin de l'instruction complète.

Pour plus d'informations sur l'utilisation du paramètre –query, consultez la section « –query (requête de recherche composée), commande de requête », page 106.

## **Commande de requête –layoutnames (noms des modèles)**

Extrait le nom de tous les modèles disponibles pour une base de données spécifique, hébergée par FileMaker Server et activée pour la Publication Web personnalisée avec XML ou XSLT.

**Paramètres de requête indispensables :** –db

### **Exemple :**

Pour extraire les noms des modèles disponibles :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–layoutnames
```

## **Commande de requête –new (nouvel enregistrement)**

Crée un nouvel enregistrement et renseigne cet enregistrement avec le contenu de n'importe quelle paire nom/valeur d'une rubrique.

**Paramètres de requête indispensables :** –db, –lay

**Paramètres de requête facultatifs :** un ou plusieurs noms de rubriques, –script

**Remarque** Pour plus d'informations sur la manière d'inclure de nouvelles données pour une table externe, consultez la section « Ajout d'enregistrements à une table externe », page 96.

### **Exemple :**

Pour ajouter un nouvel enregistrement :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&Country=Australia&–new
```

## **–process (traiter les feuilles de style XSLT)**

Traite une feuille de style XSLT sans rechercher de données dans la base de données. Cette commande de requête peut uniquement être utilisée avec des feuilles de style XSLT.

**Paramètre de requête indispensable :** –grammar. Vous devez utiliser les grammaires fmresultset ou FMPXMLRESULT.

### **Exemple :**

`http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–grammar=fmresultset&–process`

Consultez la section « Traitement des requêtes XSLT n’adressant pas de requêtes à FileMaker Server », page 65.

## **Commande de requête –scriptnames (noms des scripts)**

Extrait le nom de tous les scripts disponibles pour une base de données spécifique, hébergée par FileMaker Server et activée pour la Publication Web personnalisée avec XML ou XSLT.

**Paramètres de requête indispensables :** –db

### **Exemple :**

Pour extraire les noms de tous les scripts :

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–scriptnames`

## **Commande de requête –view (affichage des informations de modèle)**

Si la grammaire FMPXMLLAYOUT est spécifiée, cette commande extrait des informations de modèle d’une base de données et les affiche dans la grammaire FMPXMLLAYOUT. Si une grammaire de données (fmresultset or FMPXMLRESULT) est spécifiée, la commande extrait la section de métadonnées d’un document XML et un recordset vide.

**Paramètres de requête indispensables :** –db, –lay

**Paramètres de requête facultatifs :** –script

### **Exemples :**

Pour extraire des informations de modèle :

`http://192.168.123.101/fmi/xml/FMPXMLLAYOUT.xml?–db=employees&–lay=departments&–view`

Pour extraire des informations de métadonnées :

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–view`

## Références de paramètre de requête

Cette section contient des informations à propos des paramètres de requête disponibles pour les requêtes XML et XSLT. Pour obtenir des informations sur les paramètres uniquement disponibles pour les requêtes XSLT, consultez la section « Utilisation de chaînes de requête dans les feuilles de style FileMaker XSLT », page 60.

### Paramètre de requête **–db (nom de base de données)**

Indique la base de données à laquelle la commande de requête s'applique.

**Valeur :** nom de la base de données sans l'extension de nom de fichier, si elle existe.

**Remarque** Lorsque vous indiquez le nom de la base de données pour le paramètre **–db** dans les chaînes de requête, n'incluez pas l'extension du nom de fichier. Le nom de fichier réel de la base de données peut éventuellement comprendre une extension, mais les extensions ne sont pas autorisées comme valeur pour le paramètre **–db**.

**Obligatoire pour :** toutes les commandes de requête, à l'exception de **–dbnames** et **–process**

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–findall`

### **–delete.related (suppression d'enregistrements externes), paramètre de requête**

Supprime un enregistrement dans une rubrique de table externe.

**Facultatif pour :** la commande de requête **–edit**

**Nécessite :** un nom de table liée et un identificateur d'enregistrement

**Exemple :**

L'exemple suivant supprime l'enregistrement de table externe dont l'ID est 20 de la table liée nommée `jobtable`, et dont l'ID d'enregistrement parent est 7.

`http://host/fmi/xml/fmresultset.xml?–db=career&–lay=applications&–recid=7&–delete.related=jobtable.20&–edit`

### Paramètre de requête **–encoding (codage de la requête XSLT)**

Spécifie le codage de texte pour une requête XSLT. Cette commande de requête peut uniquement être utilisée avec la Publication Web personnalisée pour les requêtes XSLT.

**Valeur :** US-ASCII, ISO-8859-1, ISO-8859-15, ISO-2022-JP, Shift\_JIS ou UTF-8

**Facultatif pour :** toutes les commandes de requête d'une requête XSLT.

**Exemple :**

`http://192.168.123.101/fmi/xsl/my_template/my_stylesheet.xml?–db=employees&–lay=departments  
&–grammar=fmresultset&–encoding=Shift_JIS&–findall`

Consultez la section « Définition du codage du texte dans les requêtes », page 63.

### Paramètre de requête **–field (nom de rubrique Multimédia)**

Indique le nom d'une rubrique Multimédia

**Obligatoire pour :** les requêtes de données dans une rubrique Multimédia.

Consultez les rubriques « A propos de la syntaxe d'URL pour les objets FileMaker de type Multimédia dans les solutions XML », page 42, et « A propos de la syntaxe d'URL relative aux objets multimédias dans les solutions XSLT », page 60.

### Paramètre de requête **fieldname** (nom de rubrique non-multimédia)

Les noms de rubrique sont utilisés pour contrôler les critères de la commande de requête `–find` ou pour modifier le contenu d'un enregistrement. Quand vous devez spécifier une valeur pour une rubrique non-multimédia pour une commande ou un paramètre de requête, utilisez le nom de rubrique sans le trait d'union (`–`) en guise de partie nom de la paire nom/valeur.

**Nom :** nom de la rubrique dans la base de données FileMaker. Si la rubrique ne se trouve pas dans la table sous-jacente du modèle spécifié dans la chaîne de requête, le nom de rubrique doit être entièrement qualifié. Les noms de rubriques peuvent contenir des points, sauf dans les cas suivants :

- Le point ne peut pas être suivi d'un chiffre. Par exemple, `myfield.9` est un nom de rubrique invalide.
- Le point ne peut pas être suivi de la chaîne de texte `op` (les deux lettres « `op` »). Par exemple, `myfield.op` est un nom de rubrique invalide.
- Le point ne peut pas être suivi de la chaîne de texte `global` (le mot « `global` »). Par exemple, `myfield.global` est un nom de fichier invalide.

Les noms de rubrique contenant l'une de ces exceptions ne sont pas accessibles via XML ou XSLT lorsque vous utilisez une requête HTTP. Ces constructions sont destinées aux ID d'enregistrements, comme décrit dans la section « A propos de la syntaxe d'un nom de rubrique entièrement qualifié », page 95.

**Valeur :** pour les commandes de requête `–new` and `–edit`, indiquez la valeur que vous souhaitez stocker dans la rubrique de l'enregistrement actuel. Pour les commandes de requête `–find`, indiquez la valeur que vous souhaitez rechercher dans la rubrique. Lorsque vous indiquez la valeur d'une rubrique de type date, heure ou horodatage, spécifiez-la en utilisant le format `fm`. Les formats `fm` sont `MM/jj/aaaa` pour la date, `HH:mm:ss` pour l'heure et `MM/jj/aaaa HH:mm:ss` pour l'horodatage.

**Obligatoire pour :** la commande de requête `–edit`

**Facultatif pour :** les commandes de requête `–new` et `–find`

**Exemple :**

```
http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–op=eq&FirstName=Sam
&–max=1&–find
```

**Remarque** Le fait de spécifier un nom de rubrique à plusieurs reprises dans une même requête n'est pas pris en charge ; FileMaker Server analyse toutes les valeurs mais utilise uniquement la dernière valeur analysée.

## Paramètre de requête **fieldname.op** (opérateur de comparaison)

Indique l'opérateur de comparaison à appliquer au nom de rubrique précédant l'opérateur. Les opérateurs de comparaison sont utilisés avec la commande de requête `–find`.

**Valeur :** l'opérateur que vous souhaitez utiliser. L'opérateur par défaut est `begins with`. Les opérateurs valides sont les suivants :

Mot-clé	Opérateur FileMaker Pro équivalent
eq	= word
cn	*word*
bw	word*
ew	*word
gt	> word
gte	>= word
lt	< word
lte	<= word
neq	omit, word

### Facultatif pour : la commande de requête `–find`

**Nécessite :** un nom de rubrique et une valeur.

La syntaxe de spécification d'un opérateur de comparaison est la suivante :

`table-name::field-name=value&table-name::field-name.op=op-symbol`

où :

- `table-name` est la table contenant la rubrique et est uniquement requise si la rubrique n'est pas dans la table source du modèle spécifié dans la chaîne de requête.
- `op-symbol` est l'un des mots-clés du tableau précédent, tel que `cn`.

### Exemple :

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&name=Tim&name.op=cn&–find`

Vous pouvez utiliser n'importe quel opérateur de recherche FileMaker Pro, en spécifiant le mot-clé `bw`. Par exemple, pour chercher une plage de valeurs à l'aide de l'opérateur de plage (...), spécifiez le mot-clé `bw` et placez les caractères ... devant le critère de recherche.

### Exemple :

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&IDnum=915...925&IDnum.op=bw&–find`

Pour plus d'informations sur les opérateurs pouvant être utilisés pour rechercher du texte, consultez l'aide FileMaker Pro.



### Paramètre de requête –grammar (grammaire pour feuilles de style XSLT)

Indique la grammaire à utiliser pour une feuille de style XSLT. Cette commande de requête peut uniquement être utilisée avec la Publication Web personnalisée pour les requêtes XSLT.

**Valeur :** fmresultset, FMPXMLRESULT ou FMPXMLLAYOUT

**Obligatoire pour :** toutes les requêtes XSLT.

**Exemple :**

`http://192.168.123.101/fmi/xsl/my_template/my_stylesheetsheet.xsl?–grammar=fmresultset&–db=mydatabase  
&–lay=mylayout&–findall`

Consultez la section « Spécification d’une grammaire XML pour une feuille de style FileMaker XSLT », page 61.

### Paramètre de requête –lay (modèle)

Indique le modèle de base de données que vous souhaitez utiliser.

**Valeur :** nom du modèle.

**Obligatoire pour :** Toutes les commandes de requête à l’exception de –dbnames, –layoutnames, –scriptnames et –process (requêtes XSLT uniquement).

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–view`

### Paramètre de requête –lay.response (changer de modèle pour la réponse)

Indique que FileMaker Server devrait utiliser le modèle spécifié par le paramètre –lay lors du traitement d’une requête et passer au modèle spécifié par le paramètre –lay.response pour le traitement de la réponse XML.

Si vous n’incluez pas le paramètre –lay.response, FileMaker Server utilise le modèle spécifié par le paramètre –lay pour le traitement de la requête et de la réponse.

Vous pouvez utiliser le paramètre –lay.response pour des requêtes XML ou dans des requêtes de feuille de style XSLT.

**Valeur :** nom du modèle.

**Facultatif pour :** toutes les commandes de requête à l’exception de –dbnames, –layoutnames, –scriptnames et –process (requêtes XSLT uniquement)

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=Budget&Salary=100000&Salary.op=gt&–find  
&–lay.response=ExecList`

### Paramètre de requête –lop (opérateur logique)

Indique comment les critères de recherche de la commande de requête –find sont combinés en recherche and ou or.

**Valeur :** and ou or (qui doivent être indiqués en minuscules). Si le paramètre de requête –lop n’est pas inclus, la commande de requête –find utilise la valeur and.

**Facultatif pour :** la commande de requête –find

**Remarque** Non pris en charge par la commande de requête –findquery.

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&Last+Name=Smith  
&Birthdate=2/5/1972&–lop=and&–find`

## Paramètres de requête –max (nombre maximal d’enregistrements)

Indique le nombre maximum d’enregistrements que vous souhaitez obtenir.

**Valeur :** un nombre, ou la valeur all pour obtenir tous les enregistrements. La valeur all doit être spécifiée en minuscules. Si –max n’est pas spécifié, tous les enregistrements sont renvoyés.

**Facultatif pour :** les commandes de requête –find, –findall, et –findquery

**Exemples :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–max=10&–findall`

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–max=all&–findall`

## Paramètre de requête –modid (numéro d’identification de la modification)

L’ID de modification est un compteur incrémentiel indiquant la version actuelle d’un enregistrement.

En spécifiant un ID de modification lorsque vous utilisez une commande de requête –edit, vous pouvez vous assurer que vous éditez la version actuelle d’un enregistrement. Si l’ID de modification que vous spécifiez ne correspond pas à la valeur d’ID de modification actuelle dans la base de données, la commande de requête –edit n’est pas autorisée et un code d’erreur est renvoyé.

**Valeur :** un identifiant (ID) de modification unique, indiquant la dernière version d’un enregistrement d’une base FileMaker.

**Facultatif pour :** la commande de requête –edit

**Nécessite :** le paramètre –recid

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–recid=22&–modid=6  
&last_name=Jones&–edit`

## –query (requête de recherche composée), commande de requête

Indique les noms de requête et les critères de recherche d’une requête de recherche composée. Consultez la section « –findquery (recherche composée), commande de requête », page 100.

**Valeur :** Une expression de requête.

**Obligatoire pour :** la commande de requête –findquery

La syntaxe d’une requête de recherche composée est :

`–query=<request-declarations><request-definitions>&–findquery`

Où :

`<request-declarations>` représente au moins deux déclarations de requête.

- Chaque déclaration de requête se compose d’un ou plusieurs identificateur(s) de requête, séparé(s) par des virgules et mis entre parenthèses. Un identificateur de requête est représenté par la lettre « q » suivie d’un nombre. Par exemple : q1
- Entourées de parenthèses, les requêtes multiples agissent comme un opérateur logique ET, ce qui réduit le résultat de la recherche. Par exemple, (q1), (q2) renvoie des enregistrements correspondant à q1 et q2.

- Comme avec FileMaker Pro, chaque requête peut être une recherche ou une exclusion. Une requête de recherche ajoute les enregistrements correspondants au jeu d'enregistrements trouvés ; une exclusion les en retire. Par défaut, les requêtes sont des recherches. Pour une requête d'exclusion, placer un point d'exclamation ( ! ) devant la parenthèse ouvrante.

Par exemple : (q1);!(q2)

Dans cet exemple, q1 est une requête de recherche ; q2 est une requête d'exclusion car elle est précédée d'un point d'exclamation.

- Les requêtes sont séparées par des points-virgules. Les requêtes multiples agissent comme un opérateur logique OU, ce qui élargit le résultat de la recherche. Par exemple, (q1);(q2) renvoie des enregistrements correspondant à q1 ou q2. Les requêtes d'exclusion n'agissent pas comme un opérateur logique OU car elles excluent des enregistrements du résultat de la recherche.
- Les requêtes sont exécutées dans l'ordre indiqué ; le jeu d'enregistrements trouvés inclut les résultats de la totalité de la requête de recherche composée.

<request-definitions> est une définition de requête correspondant à chaque déclaration. Chaque définition de requête se compose d'une rubrique de recherche et d'une définition de valeur. Un signe moins (–) indique le début de la définition de requête.

Syntaxe :

–<query-id>=<fieldname>&–<query-id>.value=<value>

Par exemple :

–q1=typeofanimal&–q1.value=Cat

–q2=name&–q2.value=Fluffy

### Exemple :

Rechercher les enregistrements des chats gris qui ne s'appellent pas « Fluffy ».

[http://host/fmi/xml/fmresultset.xml?–db=petclinic&–lay=Patients&–query=\(q1,q2\);!\(q3\)&–q1=typeofanimal&–q1.value=Cat&–q2=color&–q2.value=Gray&–q3=name&–q3.value=Fluffy&–findquery](http://host/fmi/xml/fmresultset.xml?–db=petclinic&–lay=Patients&–query=(q1,q2);!(q3)&–q1=typeofanimal&–q1.value=Cat&–q2=color&–q2.value=Gray&–q3=name&–q3.value=Fluffy&–findquery)

## Paramètre de requête –recid (ID d'enregistrement)

Indique l'enregistrement que vous souhaitez traiter. Utilisé principalement dans les commandes de requête –edit et –delete. Utilisé par la commande –view pour extraire des données de liste de valeurs liées dans la grammaire FMPXMLLAYOUT.

**Valeur :** un ID d'enregistrement qui est l'identifiant unique d'un enregistrement de la base de données FileMaker.

**Obligatoire pour :** les commandes de requête –edit, –delete et –dup

**Facultatif pour :** les commandes –find et –view

### Exemple 1 :

<http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–recid=22&–delete>

### Exemple 2 :

<http://localhost/fmi/xml/FMPXMLLAYOUT.xml?–db=test&–lay=empty&–view&–recid=9>

**–relatedsets.filter (filtrage d'enregistrements de table externe), paramètre de requête**

Indique s'il convient de filtrer les lignes renvoyées pour les requêtes utilisant des rubriques de table externe.

**Valeur :** layout ou none

- La valeur par défaut est none si ce paramètre n'est pas précisé.
- Si la requête indique layout, les paramètres indiqués dans la boîte de dialogue Table externe de FileMaker Pro sont respectés. Les enregistrements sont triés sur la base du tri défini dans la boîte de dialogue Table externe, le jeu d'enregistrements filtré commençant par la ligne initiale indiquée.
- Si le paramètre Barre de défilement vertical est activé dans la boîte de dialogue Table externe, vous pouvez employer l'option –relatedsets.max pour indiquer le nombre maximum de lignes à renvoyer en réponse à la requête.
- Si le paramètre Barre de défilement vertical est désactivé, alors le nombre de lignes de la boîte de dialogue Table externe de FileMaker Pro détermine le nombre de lignes à afficher.
- Si –relatedsets.filter est réglé sur none, le moteur de publication Web renvoie toutes les lignes de la table externe, ainsi que les enregistrements externes qui ne sont pas prétriés.

**Facultatif pour :** –find, –edit, –new, –dup et –findquery.

**Exemples :**

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&–relatedsets.filter=none&–findany
```

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&relatedsets.filter=layout
&–relatedsets.max=all&–findany
```

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&–relatedsets.filter=layout
&–relatedsets.max=10&–findany
```

**–relatedsets.max (limitation d'enregistrements de table externe), paramètre de requête**

Indique le nombre maximum de lignes à renvoyer dans les résultats de cette requête.

**Valeur :** un nombre entier, ou all.

- Si la requête indique un nombre entier, le moteur de publication Web renvoie ce nombre de lignes après la ligne initiale.
- Si la requête indique all, le moteur de publication Web renvoie tous les enregistrements liés.
- Si la requête ne précise pas le paramètre –relatedsets.max, le nombre de colonnes est déterminé par la valeur du paramètre –relatedsets.filter. Consultez la section « –relatedsets.filter (filtrage d'enregistrements de table externe), paramètre de requête », page 108.

**Facultatif pour :** –find, –edit, –new, –dup et –findquery.

**Exemples :**

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&relatedsets.filter=layout
&–relatedsets.max=all&–findany
```

```
http://localhost/fmi/xml/fmresultset.xml?–db=FMPHP_Sample&–lay=English&–relatedsets.filter=layout
&–relatedsets.max=10&–findany
```

**Paramètre de requête –script (Script)**

Indique le script FileMaker à lancer quand la commande de requête et le tri ont été exécutés. Consultez la section « Comprendre le traitement d'une requête XML », page 55.

**Valeur :** Nom du script.

**Facultatif pour :** toutes les commandes de requête, à l'exception de `-dbnames`, `-layoutnames`, `-process` et `-scriptnames`

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script=myscript&-findall`

### Paramètre de requête `-script.param` (transfert du paramètre au script)

Transmet un paramètre au script FileMaker spécifié par `-script`

**Valeur :** Un simple paramètre texte.

- Pour transmettre plusieurs paramètres, vous pouvez créer une chaîne délimitant les paramètres et faire en sorte que votre script analyse chacun des paramètres. Par exemple, transmettez `param1|param2|param3` en tant que liste, en utilisant le caractère `|` de codage URL de la façon suivante :  
`param1%7Cparam2%7Cparam3`
- Pour traiter le paramètre texte en tant que valeur qui ne soit pas du texte, votre script peut convertir la valeur texte. Par exemple, pour convertir la valeur texte en nombre, votre script doit intégrer :  
`ObtenirNombre(Obtenir(ParamètreScript))`
- Si votre requête contient `-script.param` sans `-script`, alors `-script.param` est ignoré.
- Si votre requête contient plusieurs `-script.param`, alors le moteur de publication Web utilise la dernière valeur analysée.

**Facultatif pour :** `-script`

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script=myscript  
&-script.param=Smith%7CChatterjee%7CSu&-findall`

### Paramètre de requête `-script.prefind` (script avant une recherche)

Spécifie le script FileMaker à exécuter avant la recherche et le tri d'enregistrements (le cas échéant) pendant le traitement d'une commande de requête `-find`.

**Valeur :** Nom du script.

**Facultatif pour :** toutes les commandes de requête, à l'exception de `-dbnames`, `-layoutnames`, `-process` et `-scriptnames`

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-script.prefind=myscript&-findall`

## Paramètre de requête –script.prefind.param (transfert du paramètre au script avant recherche)

Transfère un paramètre au script FileMaker spécifié par –script.prefind

**Valeur :** Un simple paramètre texte.

- Pour transmettre plusieurs paramètres, vous pouvez créer une chaîne délimitant les paramètres et faire en sorte que votre script analyse chacun des paramètres. Par exemple, transmettez param1|param2|param3 en tant que liste, en utilisant le caractère | de codage URL de la façon suivante :  
param1%7Cparam2%7Cparam3
- Pour traiter le paramètre texte en tant que valeur qui ne soit pas du texte, votre script peut convertir la valeur texte. Par exemple, pour convertir la valeur texte en nombre, votre script doit intégrer :  
ObtenirNombre(Obtenir(ParamètreScript))
- Si votre requête contient –script.prefind.param sans –script.prefind, alors –script.prefind.param est ignoré.
- Si votre requête contient plusieurs –script.prefind.param, alors le moteur de publication Web utilise la dernière valeur analysée.

**Facultatif pour :** –script.prefind

**Exemple :**

http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–script.prefind=myscript  
&–script.prefind.param=payroll&–findall

## Paramètre de requête –script.presort (script avant un tri)

Spécifie le script FileMaker à exécuter après l'extraction d'enregistrements (le cas échéant) et avant leur tri, pendant le traitement d'une commande de requête –find.

**Facultatif pour :** toutes les commandes de requête, à l'exception de –dbnames, –layoutnames, –process et –scriptnames

**Exemple :**

http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–script.presort=myscript  
&–sortfield.1=dept&–sortfield.2=rating&–findall

## Paramètre de requête –script.presort.param (transfert du paramètre au script avant recherche)

Transfère un paramètre au script FileMaker spécifié par –script.presort

**Valeur :** Un simple paramètre texte.

- Pour transmettre plusieurs paramètres, vous pouvez créer une chaîne délimitant les paramètres et faire en sorte que votre script analyse chacun des paramètres. Par exemple, transmettez param1|param2|param3 en tant que liste, en utilisant le caractère | de codage URL de la façon suivante :  
param1%7Cparam2%7Cparam3
- Pour traiter le paramètre texte en tant que valeur qui ne soit pas du texte, votre script peut convertir la valeur texte. Par exemple, pour convertir la valeur texte en nombre, votre script doit intégrer :  
ObtenirNombre(Obtenir(ParamètreScript))
- Si votre requête contient –script.presort.param sans –script.presort, alors –script.presort.param est ignoré.
- Si votre requête contient plusieurs –script.presort.param, alors le moteur de publication Web utilise la dernière valeur analysée.

**Facultatif pour :** `–script.presort`

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–script.presort=myscript  
&–script.presort.param=18%7C65&–sortfield.1=dept&–sortfield.2=rating&–findall`

### Paramètre de requête `–skip` (enregistrements ignorés)

Indique le nombre d'enregistrements à ignorer dans l'ensemble trouvé.

**Valeur :** un nombre. Si la valeur est supérieure au nombre d'enregistrements contenus dans l'ensemble trouvé, aucun enregistrement n'est affiché. La valeur par défaut est 0.

**Facultatif pour :** la commande de requête `–find`

Dans l'exemple suivant, les 10 premiers enregistrements de l'ensemble trouvé sont ignorés et les enregistrements 11 à 15 sont affichés.

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–skip=10&–max=5&–findall`

### Paramètre de requête `–sortfield` (rubrique de tri)

Spécifie la rubrique à utiliser pour le tri.

**Valeur :** nom de rubrique.

**Facultatif pour :** les commandes de requête `–find` ou `–findall`

Le paramètre de requête `–sortfield` peut être utilisé plusieurs fois pour effectuer des tris de rubriques multiples. La syntaxe permettant de spécifier la priorité des rubriques de tri est la suivante :

`–sortfield.precedence-number=fully-qualified-field-name`

où `precedence-number` dans le paramètre de requête `–sortfield.precedence-number` est un nombre (débutant à 1) qui indique la priorité à utiliser dans le cas de rubriques à tri multiple.

Dans l'exemple suivant, la rubrique `dept` est triée en premier, puis vient le tour de la rubrique `rating`. Les deux rubriques sont triées par ordre croissant car le paramètre de requête `–sortorder` n'est pas spécifié.

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=performance&–sortfield.1=dept  
&–sortfield.2=rating&–findall`

### Paramètre de requête `–sortorder` (ordre de tri)

Indique l'ordre d'un tri.

**Valeur :** l'ordre de tri. Les ordres de tri suivants sont autorisés, quand `<value-list-name>` est un nom de liste de valeurs tel que `Personnalisé` :

Mot-clé	Opérateur FileMaker Pro équivalent
<code>ascend</code>	Tri a à z, –10 à 10
<code>descend</code>	Tri z à a, 10 à –10
<code>&lt;value-list-name&gt;</code>	Tri à l'aide de la liste des valeurs spécifiée, associée à la rubrique dans le modèle.

**Facultatif pour : les commandes de requête –find ou –findall****Nécessite :** le paramètre de requête –sortfield

Le paramètre de requête –sortorder peut être utilisé avec le paramètre de requête –sortfield pour spécifier l'ordre de tri des rubriques de tri multiple. La syntaxe permettant de spécifier l'ordre de tri d'une rubrique de tri est la suivante :

–sortorder.precedence-number=sort-method

où :

- precedence-number dans le paramètre –sortorder.precedence-number est un nombre situé entre 1 et 9 indiquant le paramètre de requête –sortfield auquel le paramètre de requête –sortorder s'applique.
- sort-method est un des mots-clés du tableau précédent permettant de spécifier l'ordre de tri, par exemple ascend.

Dans l'exemple suivant, l'ordre de tri de la rubrique ayant la priorité la plus élevée (dept) est ascend, tandis que l'ordre de tri de la rubrique de priorité secondaire (rating) est descend. La valeur 2 affectée à precedence-number dans –sortorder.2 indique que le paramètre de requête –sortorder.2=descend s'applique au paramètre de requête –sortfield.2=rating.

**Exemple :**

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=performance&–sortfield.1=dept  
&–sortorder.1=ascend&–sortfield.2=rating&–sortorder.2=descend&–findall`

**Remarque** Si un paramètre de requête –sortorder n'est pas spécifié pour une rubrique de tri, le tri ascendant est utilisé par défaut.

**Paramètre de requête –stylehref (Style href)**

Génère une instruction de traitement de feuille de style XML au sein du document de sortie, en définissant la valeur de l'attribut href (href=/mystylesheet.css ou href=/stylesheets/mystylesheet.xml), pour vous permettre d'utiliser des feuilles de style en cascade (CSS) ou XSLT côté client avec votre document XML. La valeur du paramètre –stylehref doit être un chemin d'accès absolu. Le nom de la feuille de style peut être un nom quelconque, mais il doit comporter l'extension .css ou .xml. Consultez la section « Utilisation du traitement des feuilles de style côté serveur et côté client », page 55. Ce paramètre est utilisé en même temps que le paramètre –styletype.

**Facultatif pour :** toutes les commandes de requête.

**Nécessite :** le paramètre –styletype

**Exemple** (suppose que mystylesheet.xml se trouve dans le dossier racine du logiciel de serveur Web) :

`http://192.168.123.101/fmi/xml/fmresultset.xml?–db=employees&–lay=departments&–styletype=text/xml  
&–stylehref=/mystylesheet.xml&–findall`

**Paramètre de requête –styletype (type de style)**

Génère une instruction de traitement de feuille de style XML au sein du document de sortie, en définissant la valeur de l'attribut type (type=text/css ou type=text/xml), pour vous permettre d'utiliser des feuilles de style en cascade (CSS) ou XSLT côté client avec votre document XML. Consultez la section « Utilisation du traitement des feuilles de style côté serveur et côté client », page 55. Ce paramètre est utilisé en même temps que le paramètre –stylehref.

**Facultatif pour :** toutes les commandes de requête.

**Nécessite :** le paramètre –stylehref



**Exemple** (suppose que mystylesheet.css se trouve dans le dossier racine du logiciel de serveur Web) :

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments&-styletype=text/css
&-stylehref=/mystylesheet.css&-findall
```

### Paramètre de requête **-token.[string]** (transfert des valeurs d'une feuille de style XSLT à une autre)

Transfère toute information définie par l'utilisateur entre des feuilles de style XSLT sans utiliser de sessions ou de cookies. Ce paramètre de requête peut uniquement être utilisé avec la Publication Web personnalisée pour les requêtes XSLT.

**string dans -token.[string] correspond à :** une chaîne alphanumérique d'une longueur quelconque, à l'exception des espaces, comprenant des chiffres de 0 à 9, des lettres minuscules de a à z ou des lettres majuscules de A à Z.

**Valeur du paramètre défini par l'utilisateur :** toute chaîne de caractères codée à la façon d'une URL.

**Facultatif pour :** toutes les requêtes XSLT.

**Exemple :**

```
http://192.168.123.101/fmi/xsl/template/my_stylesheet.xsl?-db=employees&-lay=departments
&-grammar=fmresultset&-token.D100=Active&-findall
```

Consultez la section « Utilisation du paramètre -token pour transmettre des informations entre plusieurs feuilles de style », page 66.



# Annexe B

## ***Codes d'erreur de la Publication Web personnalisée***

Le moteur de publication Web prend en charge trois types de codes d'erreur pouvant survenir dans le cadre de la Publication Web personnalisée :

- **Erreurs de base de données et de chaîne de requête.** Le moteur de publication Web génère un code d'erreur pour une base de données publiée chaque fois qu'une requête de données XML à lieu. Consultez la section suivante, « Numéros des codes d'erreur des bases de données FileMaker : ».
- **Erreurs du moteur de publication Web.** Quand le moteur de publication Web est en mode Développement, il génère une page d'erreur spécifique à chaque fois qu'une erreur se produit au sein du moteur de publication Web lui-même. En mode Production, un message texte général est affiché. Consultez le document « Numéros des codes d'erreur du moteur de publication Web », page 122.
- **Erreurs de fonction d'extension XSLT de FileMaker.** Vous pouvez utiliser la fonction d'extension `fmxml:check_error_status()` au sein d'une feuille de style XSLT pour vérifier l'état d'erreur des fonctions d'extension quand elles sont appelées. Consultez le document « Numéros des codes d'erreur des fonctions d'extension XSLT de FileMaker », page 123.

La liste mise à jour des codes d'erreur est disponible sur la Base de connaissances FileMaker (<http://www.filemaker.com/kb/>).

### **Numéros des codes d'erreur des bases de données FileMaker :**

Le moteur de publication Web génère un code d'erreur pour les bases de données publiées au format XML lorsque des données sont requises. Ce type de valeur de code d'erreur s'insère au début du document XML dans l'élément `<error code>` pour la grammaire `fmresultset`, ou dans l'élément `<ERRORCODE>` pour les grammaires `FMPXMLRESULT` ou `FMPDSORESULT`. Un code d'erreur égal à 0 indique qu'aucune erreur ne s'est produite.

Voici un exemple de code d'erreur de base de données dans la grammaire `fmresultset` :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE fmresultset PUBLIC "-//FMI//DTD fmresultset//EN" "fmi/xml/fmresultset.dtd">
<fmresultset xmlns="http://www.filemaker.com/xml/fmresultset" version="1.0">
  <error code="0"></error>
```

Voici un exemple de code d'erreur de base de données dans la grammaire `FMPXMLRESULT` :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLRESULT PUBLIC "-//FMI//DTD FMPXMLRESULT//EN" "fmi/xml/FMPXMLRESULT.dtd">
<fmpxmlresult xmlns="http://www.filemaker.com/fmpxmlresult">
  <ERRORCODE>0</ERRORCODE>
```

En tant que développeur de la solution de Publication Web personnalisée, il vous appartient de vérifier la valeur des éléments <error code> ou <ERRORCODE> et d'agir en conséquence. Le moteur de publication Web ne gère pas les erreurs de base de données.

Numéro d'erreur	Description
-1	Erreur inconnue
0	Pas d'erreur
1	L'utilisateur a annulé l'action
2	Erreur de mémoire
3	Commande non disponible (par exemple, système d'exploitation incorrect, mode incorrect, etc.)
4	Commande inconnue
5	Commande incorrecte (par exemple, un calcul n'a pas été indiqué pour une action de script Définir rubrique)
6	Fichier accessible en lecture seule
7	Mémoire insuffisante
8	Résultat vide
9	Privilèges d'accès insuffisants
10	Données requises manquantes
11	Nom incorrect
12	Ce nom existe déjà
13	Le fichier ou l'objet est déjà utilisé
14	En dehors de l'intervalle
15	Division par zéro impossible
16	Echec de l'opération, réessayez (par exemple une requête utilisateur)
17	Echec de la tentative de conversion d'un jeu de caractères étranger en UTF-16
18	Le client doit fournir les données du compte pour poursuivre
19	La chaîne contient des caractères autres que A-Z, a-z, 0-9 (ASCII)
20	Commande ou opération annulée par un script déclenché
100	Fichier manquant
101	Enregistrement manquant
102	Rubrique manquante
103	Lien manquant
104	Script manquant
105	Modèle manquant
106	Table manquante
107	Index manquant
108	Liste de valeurs manquante
109	Jeu de privilèges d'accès manquant
110	Tables liées manquantes
111	Valeur de rubrique incorrecte

Numéro d'erreur	Description
112	Fenêtre manquante
113	Fonction manquante
114	Référence de fichier manquante
115	Le jeu de menus est manquant
116	L'objet de modèle est manquant
117	La source de données est manquante
130	Les fichiers sont endommagés ou manquants et doivent être réinstallés
131	Les fichiers des modules de langue sont manquants (par exemple les modèles)
200	L'accès à l'enregistrement est interdit
201	Impossible de modifier la rubrique
202	L'accès à la rubrique est interdit
203	Aucun enregistrement à imprimer, ou bien un mot de passe est nécessaire pour imprimer
204	Aucun accès aux rubriques dans l'ordre de tri
205	Impossible de créer de nouveaux enregistrements ; l'importation peut remplacer les données existantes
206	Impossible de modifier le mot de passe ou bien le fichier ne peut pas être modifié
207	Impossible d'accéder au schéma de base de données ou bien le fichier ne peut pas être modifié
208	Nombre de caractères insuffisant dans le mot de passe
209	Le nouveau mot de passe doit être différent du mot de passe existant
210	Compte utilisateur inactif
211	Mot de passe expiré
212	Compte utilisateur et/ou mot de passe incorrect. Réessayez
213	Le compte utilisateur et/ou le mot de passe n'existe pas
214	Tentatives de connexion trop nombreuses
215	Impossible de dupliquer les droits Admin
216	Impossible de dupliquer un compte Invité
217	L'utilisateur ne dispose pas de droits suffisants pour modifier le compte administrateur
300	Fichier verrouillé ou en cours d'utilisation
301	L'enregistrement est déjà utilisé par un autre utilisateur
302	La table est déjà utilisée par un autre utilisateur
303	Le schéma de base de données est déjà utilisé par un autre utilisateur
304	Le modèle est déjà utilisé par un autre utilisateur
306	L'ID de modification d'enregistrement ne correspond pas
400	Les critères de recherche sont vides
401	Aucun enregistrement ne correspond à cette requête
402	Ceci n'est pas une rubrique clé pour une référence externe
403	Le nombre maximal d'enregistrements indiqué est atteint pour la version de démonstration FileMaker Pro
404	Ordre de tri incorrect

Numéro d'erreur	Description
405	Le nombre d'enregistrements indiqué dépasse le nombre d'enregistrements pouvant être ignorés
406	Les critères de remplacement et de renumérotation sont incorrects
407	Une ou les deux rubriques clés manquent (lien incorrect)
408	Le type de la rubrique indiquée est incorrect pour ce type d'opération
409	Ordre d'importation incorrect
410	Ordre d'exportation incorrect
412	Version de FileMaker Pro incorrecte pour récupérer le fichier
413	Le type de la rubrique indiquée est incorrect
414	Résultat impossible à afficher sur le modèle
415	Un ou plusieurs enregistrements liés requis ne sont pas disponibles
416	Une clé primaire est requise dans la table source de données
417	La base de données n'est pas une source de données prise en charge
500	La date indiquée ne correspond pas aux options d'entrée définies
501	L'heure indiquée ne correspond pas aux options d'entrée définies
502	Le numéro indiqué ne correspond pas aux options d'entrée définies
503	La valeur de la rubrique n'est pas comprise dans la fourchette de valeurs définies
504	La valeur de la rubrique ne correspond pas à la valeur définie dans les options d'entrée
505	La valeur de la rubrique ne correspond à aucune donnée existante contenue dans le fichier de base de données des options d'entrée
506	La valeur de la rubrique n'est pas incluse dans la liste des valeurs définies
507	La valeur de la rubrique n'est pas conforme aux critères de contrôle de calcul des options d'entrée
508	Valeur incorrecte saisie en mode Recherche
509	La rubrique requiert une valeur correcte
510	La valeur liée est vide ou non disponible
511	La valeur de la rubrique dépasse le nombre maximal de caractères autorisés
512	L'enregistrement a déjà été modifié par un autre utilisateur
513	Pour créer un enregistrement, l'une de ses rubriques doit comporter une valeur
600	Une erreur d'impression s'est produite
601	L'en-tête et le pied de page combinés sont plus longs qu'une page
602	Le corps de la page ne tient pas sur une page pour la disposition en colonnes active
603	Connexion d'imprimante perdue
700	Le fichier ne possède pas le bon format d'importation
706	Le fichier EPSF est dépourvu d'images de prévisualisation
707	Le traducteur graphique est introuvable
708	Impossible d'importer le fichier, ou bien un moniteur gérant les couleurs est nécessaire pour effectuer cette opération
709	L'importation d'une séquence QuickTime a échoué

Numéro d'erreur	Description
710	Impossible de mettre à jour la référence du fichier QuickTime car le fichier de base de données est accessible en lecture seule
711	Le traducteur d'importation est introuvable
714	Les privilèges d'accès associés à votre mot de passe ne vous permettent pas d'effectuer cette opération
715	Feuille de calcul ou plage nommée Excel spécifiée manquante
716	Une requête SQL utilisant les instructions DELETE, INSERT ou UPDATE n'est pas autorisée dans l'importation ODBC
717	Les informations XML/XSL sont insuffisantes pour procéder à l'importation ou à l'exportation
718	Erreur lors de l'analyse du fichier XML (à partir de Xerces)
719	Erreur lors de la transformation du fichier XML en fichier XSL (à partir de Xalan)
720	Erreur lors de l'exportation ; le format souhaité ne prend pas en charge les rubriques multivaluées
721	Une erreur inconnue s'est produite dans l'analyseur ou le transformateur
722	Impossible d'importer des données dans un fichier dépourvu de rubriques
723	Vous ne disposez pas des droits nécessaires pour ajouter des enregistrements ou les modifier dans la table cible
724	Vous ne disposez pas des droits nécessaires pour ajouter des enregistrements à la table cible
725	Vous ne disposez pas des droits nécessaires pour modifier les enregistrements dans la table cible
726	Le fichier d'importation contient davantage d'enregistrements que la table cible. Les enregistrements n'ont pas tous été importés
727	La table cible contient davantage d'enregistrements que le fichier d'importation. Les enregistrements n'ont pas tous été mis à jour
729	Erreur lors de l'importation. Des enregistrements n'ont pas été importés
730	Version d'Excel non prise en charge. Convertir le fichier en Excel 7.0 (Excel 95), 97, 2000, XP ou au format 2007, puis réessayer.
731	Le fichier servant à l'importation ne contient aucune donnée
732	Impossible d'insérer ce fichier car il contient d'autres fichiers
733	Une table ne peut être importée en elle-même
734	Ce type de fichier n'a pu être affiché sous la forme d'une image
735	Ce type de fichier n'a pu être affiché sous la forme d'une image. Il sera inséré et affiché sous forme de fichier
736	Il y a trop de données à exporter dans ce format. Les données seront tronquées
737	La table Bento que vous tentez d'importer est manquante.
800	Impossible de créer le fichier sur le disque
801	Impossible de créer un fichier temporaire sur le disque
802	Impossible d'ouvrir le fichier Cette erreur peut être causée par l'un des éléments suivants : <ul style="list-style-type: none"> <li>■ Nom de base de données non valide</li> <li>■ Le fichier est fermé dans FileMaker Server</li> <li>■ Permission non valide</li> </ul>
803	Le fichier est mono-utilisateur ou bien l'hôte est introuvable
804	Le fichier ne peut être ouvert en lecture seule dans son état actuel

Numéro d'erreur	Description
805	Le fichier est endommagé
806	Utilisez la commande Récupérer. Le fichier ne peut être ouvert avec cette version de FileMaker Pro
807	Le fichier n'est pas un fichier FileMaker Pro ou est sérieusement endommagé
808	Impossible d'ouvrir le fichier car les privilèges d'accès sont endommagés
809	Le disque/volume est saturé
810	Le disque/volume est verrouillé
811	Un fichier temporaire ne peut être ouvert en tant que fichier FileMaker Pro
813	Erreur de synchronisation d'enregistrements sur le réseau
814	Impossible d'ouvrir le fichier, car le nombre maximal de fichiers ouverts est atteint
815	Impossible d'ouvrir le fichier de référence externe
816	Impossible de convertir le fichier
817	Impossible d'ouvrir le fichier, car il n'appartient pas à cette solution
819	Impossible d'enregistrer une copie locale d'un fichier distant
820	Fichier en cours de fermeture
821	Déconnexion forcée par l'hôte
822	Fichiers FMI introuvables ; réinstallez les fichiers manquants
823	Impossible de définir le fichier en mode mono-utilisateur, des invités sont connectés
824	Le fichier est endommagé ou n'est pas un fichier FileMaker
825	Le fichier n'est pas autorisé à référencer le fichier protégé
900	Erreur générale de vérification orthographique
901	Le dictionnaire principal n'est pas installé
902	Impossible de lancer le système d'aide
903	Cette commande ne peut pas être utilisée dans un fichier partagé
904	Cette commande ne peut être utilisée que dans un fichier se trouvant sur le serveur FileMaker Server
905	Aucune rubrique active sélectionnée ; la commande ne peut être utilisée que si une rubrique est active
906	Le fichier en cours n'est pas partagé ; la commande ne peut être employée que si le fichier est partagé
920	Impossible d'initialiser le correcteur orthographique
921	Impossible de charger le dictionnaire de l'utilisateur pour modification
922	Dictionnaire de l'utilisateur introuvable
923	Dictionnaire de l'utilisateur en lecture seule
951	Une erreur inattendue s'est produite
954	Grammaire XML non prise en charge
955	Aucun nom de base de données
956	Le nombre maximal de sessions de base de données a été dépassé
957	Commande en conflit
958	Paramètre manquant dans la requête
959	La technologie de Publication Web personnalisée est désactivée



Numéro d'erreur	Description
1200	Erreur de calcul générique
1201	Paramètres trop peu nombreux dans la fonction
1202	Paramètres trop nombreux dans la fonction
1203	Fin de calcul non conforme
1204	Nombre, chaîne, nom de rubrique ou parenthèse ouvrante « ( » nécessaire
1205	Commentaire non terminé par "*/"
1206	La chaîne doit se terminer par un guillemet
1207	Parenthèses non équilibrées
1208	Opérateur manquant, fonction introuvable ou parenthèse ouvrante « ( » non nécessaire
1209	Nom (nom de rubrique ou de modèle) manquant
1210	Fonction plug-in déjà enregistrée
1211	L'utilisation de listes n'est pas autorisée dans cette fonction
1212	Entrez ici un opérateur (+, -, *, ).
1213	Cette variable a déjà été définie dans la fonction Définir
1214	MOYENNE, NOMBRE, MULTIVALUEE, NOMBREVALEURS, MAX, MIN, NPV, ECARTECH, SOMME et RECAPITULATIF : le programme a rencontré une expression requérant une seule rubrique
1215	Ce paramètre est incorrect pour la fonction Obtenir
1216	Seules les rubriques Récapitulatif sont autorisées comme premier argument dans RECAPITULATIF
1217	Rubrique de regroupement incorrecte
1218	Evaluation du nombre impossible
1219	Impossible d'utiliser une rubrique dans sa propre formule
1220	Le type de la rubrique doit être de type normal ou calculé
1221	Le type de données doit être un nombre, une date, une heure ou un horodatage
1222	Impossible d'enregistrer le calcul
1223	La fonction auquel il est fait référence n'est pas encore mise en place
1224	La fonction à laquelle il est fait référence n'existe pas
1225	La fonction à laquelle il est fait référence n'est pas prise en charge dans ce contexte
1400	L'initialisation du pilote ODBC a échoué ; assurez-vous que les pilotes ODBC sont correctement installés.
1401	Echec de l'allocation d'environnement (ODBC)
1402	Echec de la libération d'environnement (ODBC)
1403	Echec de la déconnexion (ODBC)
1404	Echec de l'allocation de connexion (ODBC)
1405	Echec de la libération de connexion (ODBC)
1406	Echec de la vérification de SQL API (ODBC)
1407	Echec de l'allocation d'instruction (ODBC)
1408	Erreur étendue (ODBC)

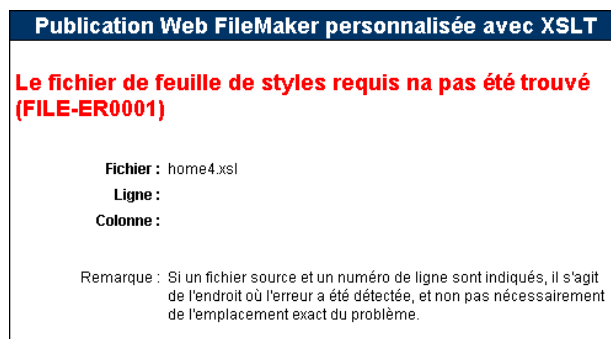
Numéro d'erreur	Description
1409	Erreur étendue (ODBC)
1410	Erreur étendue (ODBC)
1411	Erreur étendue (ODBC)
1412	Erreur étendue (ODBC)
1413	Erreur étendue (ODBC)
1450	Action nécessitant une extension de privilège PHP
1451	Action nécessitant que le fichier actif soit distant
1501	Echec de l'authentification SMTP
1502	Refus de connexion par le serveur SMTP
1503	Erreur avec SSL
1504	Le serveur SMTP exige un cryptage de la connexion.
1505	L'authentification spécifiée n'est pas prise en charge par le serveur SMTP.
1506	L'envoi du ou des emails a échoué.
1507	Connexion impossible au serveur SMTP

## Numéros des codes d'erreur du moteur de publication Web

Quand le moteur de publication Web est en mode Développement, il génère une page d'erreur spécifique quand une erreur se produit au sein du moteur de publication Web lui-même. Ce type d'erreur peut être la conséquence de diverses causes, par exemple si le moteur de publication Web est incapable :

- de trouver un fichier de feuille de style requis ou imbriqué (via <xsl:include>) ;
- d'analyser un fichier de feuille de style requis ou imbriqué en raison d'une erreur XML dans le fichier ;
- de générer une feuille de style à partir du fichier en raison d'une erreur XSLT ou XPath dans le fichier ;
- de traiter la requête parce que la grammaire XML n'est pas spécifiée convenablement dans le CGI ;
- de communiquer avec le noyau de publication Web pour extraire XML.

Lorsque le moteur de publication Web fonctionne en mode Développement, la page d'erreur de ce type d'erreur contient un message d'erreur et un numéro d'erreur entre parenthèses. Par exemple :



Exemple de page d'erreur quand le moteur de publication Web est en mode Développement.

Voici une liste partielle des valeurs de codes d'erreur du moteur de publication Web :

Valeur du code d'erreur	Description
QUERY-ER0001	Aucune grammaire XML n'a été indiquée dans le paramètre de requête <code>–grammar</code>
QUERY-ER0002	xxx n'est pas une grammaire XML valide pour le XSLT FileMaker
FILE-ER0001	Le fichier de feuille de style requis n'a pas été trouvé
FILE-ER0002	Le fichier requis n'a pas été trouvé
UNKNOWN	Une erreur imprévue s'est produite
MCS-000 à MCS-600	Une erreur imprévue s'est produite
MCS-601	La ressource x n'a pas pu être chargée car il n'existe pas de prise en charge pour les ressources de type : x
MCS-602	L'URL x n'a pas pu être résolue
MCS-603	La requête HTTP pour x a rendu une erreur de type x
MCS-604	La ressource x n'a pas pu être chargée à cause d'une erreur imprévue
MCS-605	La ressource x n'a pas pu être chargée à cause d'un type de contenu non valide
MCS-606	La ressource x n'a pas pu être chargée à cause d'une erreur XML dans le document
MCS-607	La ressource x n'a pas pu être chargée à cause d'un problème d'authentification
MCS-700	Une erreur imprévue s'est produite
MCS-800	Une erreur imprévue s'est produite

Si le moteur de publication Web est en mode Production, le message de texte général par défaut suivant s'affiche sur la page d'erreur `pe_server_error.html` pour les erreurs du moteur de publication Web :

Une erreur inattendue s'est produite durant l'utilisation de la Publication Web personnalisée de FileMaker avec XSLT.

Le fichier `pe_server_error.html` par défaut contient le message texte précédent en six langues.

En tant que développeur, vous pouvez modifier le texte de la page d'erreur `pe_server_error.html`, si nécessaire, pour votre solution. Le fichier `pe_server_error.html` se trouve dans le dossier `cwpe`, à l'intérieur du dossier `publishing-engine`, sur l'hôte où vous avez installé le moteur de publication Web.

Pour plus d'informations sur la manière de paramétrer le moteur de publication Web en mode Développement ou en mode Production, consultez l'aide de FileMaker Server.

## Numéros des codes d'erreur des fonctions d'extension XSLT de FileMaker

La fonction d'extension `fmxls:check_error_status()` (consultez la section « Vérification de l'état d'erreur des fonctions d'extension », page 83), renvoie l'une des erreurs du tableau suivant :

Valeur du code d'erreur	Description
-1	Erreur inconnue
0	Pas d'erreur
<b>Erreurs générales</b>	
10000	Nom d'en-tête incorrect
10001	Code de statut http incorrect

**Valeur du code d'erreur Description**

<b>Erreurs de session</b>	
10100	Erreur de session inconnue
10101	Le nom de session requis est déjà utilisé
10102	Accès à la session impossible – peut-être qu'elle n'existe pas
10103	La session est arrivée à expiration
10104	L'objet de session spécifié n'existe pas
<b>Erreurs de messagerie</b>	
10200	Erreur de messagerie inconnue
10201	Erreur de formatage de message
10202	Erreur de champs SMTP de message
10203	Erreur dans le champ A du message
10204	Erreur dans le champ De du message
10205	Erreur dans le champ CC du message
10206	Erreur dans le champ CCI du message
10207	Erreur dans le champ Objet du message
10208	Erreur dans le champ Répondre à du message
10209	Erreur dans le corps du message
10210	Erreur de messagerie récurrente – tentative d'appel send_email() au sein d'une feuille de style d'email XSLT
10211	Erreur d'authentification SMTP – échec de la connexion ou mauvais type d'authentification fourni
10212	Utilisation incorrecte de fonction – tentative d'appel de set_header(), set_status_code() ou set_cookie() au sein d'une feuille de style d'email XSLT
10213	Le serveur SMTP n'est pas valide ou ne fonctionne pas
<b>Erreurs de mise en forme</b>	
10300	Erreur de mise en forme inconnue
10301	Format de date ou d'heure incorrect
10302	Format de date incorrect
10303	Format d'heure incorrect
10304	Format de jour incorrect
10305	Mise en forme incorrecte de chaîne date/heure
10306	Mise en forme incorrecte de chaîne date
10307	Mise en forme incorrecte de chaîne heure
10308	Mise en forme incorrecte de chaîne jour
10309	Codage de texte non pris en charge
10310	Codage d'URL incorrect
10311	Erreur de schéma d'expression régulière

# Index

## A

- activation Publication Web personnalisée dans base de données 21
- Admin Console FileMaker Server 40
- Admin Console pour le moteur de publication Web 40
- API FileMaker pour PHP 13
  - définition 13
- API PHP pour la Publication Web personnalisée 13
- attribut de méthode, élément <xsl:output> 64
- authentification de base, Web 22, 68
- authentification des utilisateurs Web 22
  - ce mot de passe 69
  - de base 68
- auto-enter, attribut 45
- avantages de XML et XSLT 14

## B

- bases de données, protection après publication 22
- break\_encode(), fonction d'extension 77

## C

- Caractères ASCII, dans les documents XML 52
- Caractères Unicode 52
- cases à cocher, recherche de valeurs 78
- chaînes
  - comparaison, expressions régulières Perl 78
  - utilisation des fonctions d'extension de manipulation de chaînes 77
- chaînes de format, date et heure 81
- chaînes de requête 52, 60, 93
  - ajout d'enregistrements à des tables externes 96
  - commandes et paramètres 52, 60, 93
  - consignes pour 94
  - définition statique, feuilles de style XSLT 62
  - données XML 52, 93
  - feuilles de style XSLT, utilisation dans 60
  - modification d'enregistrements dans des tables externes 96
  - nom de rubrique entièrement qualifié, syntaxe 95
  - référence des chaînes de requête 95
  - rubriques de type Global, syntaxe 97
- chaînes de requête statiques, feuilles de style XSLT 62
- chargement de documents supplémentaires 69

- check\_error\_status(), fonction d'extension 83, 123
- codage
  - données XML 43, 52
  - encoding, paramètre de requête 63, 102
  - feuilles de style XSLT 65
  - requêtes 64
  - sortie via l'élément <xsl:output> 64
  - URL 42, 71
  - utilisation des fonctions d'extension de manipulation de chaînes 77
- codage de texte
  - données XML générées 43
  - encoding, paramètre de requête 63, 102
  - paramètres de codage 64
  - paramètres initiaux par défaut 64
  - paramètres par défaut des requêtes et des pages de sortie 64
  - requêtes XSLT 63
  - URL 42, 71
  - utilisation des fonctions d'extension de manipulation de chaînes 77
- codage de texte URL 42
- codes d'erreurs des bases de données 44
- commande de requête de recherche composée 100
- commandes. *Voir* chaînes requête
- comparaison de chaînes 78
- compare\_date(), fonction d'extension 80
- compare\_datetime(), fonction d'extension 80
- compare\_day(), fonction d'extension 80
- compare\_time(), fonction d'extension 80
- compte Invité
  - activation 22
  - avec la Publication Web personnalisée 22
  - désactivation 22
- comptes et privilèges
  - activation, Publication Web personnalisée 21
  - compte Invité 22
  - scripts 25
- configuration de la publication Web personnalisée 19
- Console d'administration pour le moteur de publication Web 31
- contains\_checkbox\_value(), fonction d'extension 78
- conversion de données avec les feuilles de style 29
- convert\_datetime(), fonction d'extension 80

## cookies

- fonctions d'extension, utilisation 76
- stockage de l'ID de session 71

courriers électroniques. *Voir* emails

create\_session(), fonction d'extension 72

création d'un nouvel enregistrement 100

## D

<datasource>, élément 45

Date, chaînes de format 81

Date, utilisation des fonctions d'extension 79

–db, paramètre de requête 102

–dbnames, commande de requête 98

déclencheurs 27

définition des fonctions d'extension 83

définitions de type de document (DTD) 44, 48

–delete, commande de requête 98

demandes d'informations, accès dans une requête 67

document(), fonction 69

documentation 9

documentation électronique 9

documentation en ligne 9

documentation JDBC 9

documentation ODBC 9

documentation relative à l'installation 9

documents, chargement au moyen de la fonction

document() 69

dossier Logs 90

dossier Web, copie des objets de rubrique

Multimédia 24

–dup, commande de requête 98

## E

–edit, commande de requête 99

éléments

- code d'erreur de la base de données 44

- dans grammaire fmresultset 45

- dans la grammaire FMPXMLLAYOUT 49

- dans la grammaire FMPXMLRESULT 48

emails

- fonctions d'extension 73

- paramètre de codage initial par défaut 64

–encoding, paramètre de requête 102

erreurs

- à propos des codes d'erreur 115

- éléments des codes d'erreur de la base de

- données 44

- fichiers journaux au serveur Web 90

- moteur de publication Web, numéros des codes

- d'erreur 122

- numéros des codes d'erreur des bases de

- données 115

- numéros des codes d'erreur des fonctions

- d'extension 123

- pe\_application\_log.txt, fichier journal 90

- pe\_server\_error.html, page d'erreur 123

- vérification de l'état d'erreur des fonctions

- d'extension 83, 123

<error code> et <ERRORCODE>, éléments 115

espaces de nom

- XML 43

- XSLT 62

état, enregistrement dans les sessions 71, 72

Exemples de

- grammaire FMPXMLLAYOUT générée 51

- grammaire FMPXMLRESULT générée 49

- grammaire fmresultset, génération 47

exportation de données XML 37

expressions régulières Perl, comparaison de

chaînes 78

Extensible Markup Language (XML). *Voir* XML

## F

fest

- sites Web 88

- sorties XML 89

feuilles de style

- à propos de 29

- cases à cocher, recherche de valeurs 78

- chaînes de format de date et d'heure 81

- chaînes de requête 60

- codage 65

- comparaison de chaînes, expressions régulières

- Perl 78

- conception 57

- consignes de conception 57

- cookie, fonctions d'extension 76

- côté client 55

- côté serveur 29, 57

- création avec XSLT Site Assistant 32

- date, heure et jour, fonctions d'extension 79

- emails, envoi 73

- état d'erreur des fonctions d'extension,

- vérification 83
- exemples d'utilisation 29
- fest 88, 89
- fonction de session, utilisation 71, 72
- fonctions d'en-tête, utilisation 75
  - grammar, paramètre pour 61
- instructions pour le traitement d'une feuille de style XML 55
- manipulation de chaînes, fonction d'extension 77
- mise du contenu en mémoire tampon, utilisation 70
- utilisation dans un programme ou un site Web 34
- utilisation des informations de modèle 70
- XSLT, description 29
- feuilles de style côté client 40, 55
- Feuilles de style XSLT côté serveur 29
- feuilles de style XSLT côté serveur 57
- Fichiers journaux
  - journalisation via l'élément <xsl:message> 83
  - web\_server\_module\_log.txt 91
- fichiers journaux 88, 91
  - accès au serveur Web 90
  - description 90
  - pe\_application\_log.txt 90
  - pe\_internal\_access\_log.txt 91
  - <xsl:stylesheet>, élément 90
- fichiers journaux d'accès au serveur Web, description 90
- field name, paramètre de requête (non-multimédia) 103
  - field, paramètre de requête (Multimédia) 102
- <field-definition>, élément 45
  - fieldname.op, paramètre de requête 104
- FileMaker Pro, différences avec le moteur de publication Web 37
- FileMaker Server
  - documentation 9
  - installation 9
- FileMaker Server Admin
  - voir Admin Console 23
- FileMaker Site Assistant. *Voir* XSLT Site Assistant
- FileMaker, paramètres spécifiques pour XSLT 67
- Filtrage de données avec les feuilles de style 29
- filtrage des lignes des rubriques de table externe 108
  - find, commande de requête 99
  - findall, commande de requête 99
  - findany, commande de requête 99
  - findquery, commande de requête 100
- FMPDSORESULT, grammaire
  - comparée aux autres grammaires 43
- FMPXMMLAYOUT, grammaire 37, 49–51
  - comparée aux autres grammaires 43
- FMPXMLRESULT, grammaire 37, 48–49
  - comparée aux autres grammaires 43
- fmresultset, grammaire 37, 44–47
  - comparée aux autres grammaires 43
- fmxml, mot-clé pour l'activation de la publication XML 21, 40
- fmxslt, fonctions d'extension
  - fmxslt:break\_encode(), fonction 77
  - fmxslt:check\_error\_status(), fonction 83, 123
  - fmxslt:compare\_date(), fonction 80
  - fmxslt:compare\_datetime(), fonction 80
  - fmxslt:compare\_day(), fonction 80
  - fmxslt:compare\_time(), fonction 80
  - fmxslt:contains\_checkbox\_value(), fonction 78
  - fmxslt:convert\_datetime(), fonction 80
  - fmxslt:create\_session(), fonction 72
  - fmxslt:get\_cookie(), fonction 76
  - fmxslt:get\_cookies(), fonction 76
  - fmxslt:get\_date(), fonction 79
  - fmxslt:get\_datetime(), fonction 80
  - fmxslt:get\_day(), fonction 79
  - fmxslt:get\_fm\_date\_format() fonction 79
  - fmxslt:get\_fm\_time\_format() fonction 80
  - fmxslt:get\_fm\_timestamp\_format() fonction 80
  - fmxslt:get\_header(), fonction 75
  - fmxslt:get\_long\_date\_format(), fonction 80
  - fmxslt:get\_long\_day\_format(), fonction 80
  - fmxslt:get\_long\_time\_format(), fonction 80
  - fmxslt:get\_session\_object(), fonction 72
  - fmxslt:get\_short\_date\_format(), fonction 80
  - fmxslt:get\_short\_day\_format(), fonction 80
  - fmxslt:get\_short\_time\_format(), fonction 80
  - fmxslt:get\_time(), fonction 79
  - fmxslt:html\_encode(), fonction 77
  - fmxslt:invalidate\_session(), fonction 72, 73
  - fmxslt:regex\_contains(), fonction 78
  - fmxslt:remove\_session\_object(), fonction 72
  - fmxslt:send\_email(), fonctions 73
  - fmxslt:session\_encode\_url(), fonction 71, 72
  - fmxslt:session\_exists(), fonction 72
  - fmxslt:set\_cookie(), fonction 76
  - fmxslt:set\_header(), fonction 75
  - fmxslt:set\_session\_object(), fonction 72
  - fmxslt:set\_session\_timeout(), fonction 72
  - fmxslt:set\_status\_code(), fonction 75
  - fmxslt:url\_decode(), fonction 77
  - fmxslt:url\_encode(), fonction 77
- fonctions d'en-tête, utilisation 75
- fonctions d'extension de session, utilisation dans les feuilles de style 71, 72

fonctions d'extension pour FileMaker XSLT

*Voir aussi* fonctions d'extension fmxslt

formatage de données avec les feuilles de style 29

four-digit-year, attribut 45

## G

génération d'une page statique 65

get\_cookie(), fonction d'extension 76

get\_cookies(), fonction d'extension 76

get\_date(), fonction d'extension 79

get\_datetime(), fonction d'extension 80

get\_day(), fonction d'extension 79

get\_fm\_date\_format() extension function 79

get\_fm\_time\_format() extension function 80

get\_fm\_timestamp\_format() extension function 80

get\_header(), fonction d'extension 75

get\_long\_date\_format(), fonction d'extension 80

get\_long\_day\_format(), fonction d'extension 80

get\_long\_time\_format(), fonction d'extension 80

get\_session\_object(), fonction d'extension 72

get\_short\_date\_format(), fonction d'extension 80

get\_short\_day\_format(), fonction d'extension 80

get\_short\_time\_format(), fonction d'extension 80

get\_time(), fonction d'extension 79

GIF, publication des fichiers sur le Web 24

global, attribut 45

grammaires pour XML, description 43

Grammaires recommandées pour XSLT 61

–grammar, paramètre de requête 61, 105

## H

Heure, chaînes de format 81

Heure, utilisation des fonctions d'extension 79

HTML

formulaires de requêtes XML 40

reformatage de données XML en 37

html\_encode(), fonction d'extension 77

## I

importation de données XML 37

Informations client, obtention au moyen de paramètres XSLT 68

informations de modèle, utilisation dans une feuille de style 70

Informations sur la documentation 9, 20

Intégration de données avec les feuilles de style 29

invalidate\_session(), fonction d'extension 72, 73

ISO-2022-JP, codage 64

ISO-8859-1, codage 64

ISO-8859-15, codage 64

## J

JavaScript

définition des fonctions d'extension 83

jeu de privilèges d'accès, Publication Web personnalisée 21

Jour, utilisation des fonctions d'extension 79

journal d'application 83, 90

JPEG, publication des fichiers sur le Web 24

jsessionid, paramètre 71

## L

–lay, paramètre de requête 54, 105

–lay.response, paramètre de requête 54, 105

–layoutnames, commande de requête 100

limitation des lignes des rubriques de table externe 108

–lop, paramètre de requête 105

## M

masquage de métadonnées avec les feuilles de style 29

–max, paramètre de requête 106

max-characters, attribut 45

max-repeat, attribut 45

<metadata>, élément 45

métadonnées, masquage avec les feuilles de style 29

MIME (Multipurpose Internet Mail Extensions), types 23

mise du contenu en mémoire tampon, utilisation 70

mise en mémoire tampon, utilisation dans une feuille de style 70

mode Développement, moteur de publication Web 122

mode Production, moteur de publication Web 123

modèles de base de données disponibles 100

modèles, passage de l'un à l'autre pour une réponse XML 54

–modid, paramètre de requête 106

Modifier le mot de passe, script 22

mot-clé fmxslt pour l'activation de la publication XSLT 21, 31



- moteur de publication Web
  - Admin Console 40
  - avantages 17
  - codes d'erreurs générés 115
  - Console d'administration 31
  - description 12
  - généré de documents XML 40
  - généré de pages, XSLT 30
  - généré données XML 38
  - journal d'application 90
  - mode Développement 122
  - mode Production 123
  - traitement d'une requête 12
- Moteur de publication Web personnalisée (CWPE) 30, 38
- mots de passe
  - accès aux documents XML 69
  - aucun mot de passe de connexion 22
  - authentification de base, Web 22, 68
  - définition pour la Publication Web personnalisée 21
  - Modifier le mot de passe, script 22
- mots-clés pour l'activation de la Publication Web personnalisée 21, 31, 40
- Multimédia, rubriques
  - accès aux données par les utilisateurs Web 24
  - publication du contenu 23, 35
  - syntaxe d'URL pour l'accès aux objets dans les solutions XSLT 60
  - syntaxe de l'URL pour accéder aux solutions XML 42

## N

- name, attribut 46
- navigateurs Web
  - réception des sorties 12
  - rôle dans requêtes XML 38
  - rôle, requêtes XSLT-CWP 30
- new, commande de requête 100
- nom de rubrique entièrement qualifié, syntaxe 95
- noms d'utilisateur
  - accès aux documents XML 69
  - authentification de base, Web 22, 68
  - définition pour la Publication Web personnalisée 21
- noms de rubrique, syntaxe entièrement qualifiée 95
- not-empty, attribut 45
- nouvelles fonctionnalités de la Publication Web personnalisée 18

- noyau de publication Web
  - illustration 30, 38
  - journaux d'accès internes 91
- numeric-only, attribut 45
- numéros des
  - codes d'erreur des fonctions d'extension 123
  - codes d'erreurs des bases de données 115
  - codes d'erreurs du moteur de publication Web 122

## O

- opérateurs de comparaison des rubriques 104
- opérateurs, comparaison 104
- ordre de traitement de requête XML 55
- outils pour XSLT, description 18, 32

## P

- pages de sortie
  - codage, définition 64
  - méthode de sortie, définition 64
  - paramètre de codage initial par défaut 64
  - <xsl:output>, élément 64
- paramètre de requête de recherche composée 106
- Paramètre URI de base 68
- Paramètre URI de base authentifié 69
- paramètres XSLT propres à FileMaker 67
- paramètres. *Voir* chaînes requête
- Passage d'informations d'une feuille de style à une autre 66
- passage d'un modèle à un autre pour une réponse XML 54
- PDF 9
- pe\_application\_log.txt, fichier journal 90
- pe\_internal\_access\_log.txt, fichier journal 91
- pe\_server\_error.html, page d'erreur 123
- PHP
  - avantages 14
  - résolution des problèmes 35
- Polices vectorielles modulaires (SVG), transformation de données XML en 37
- présentation
  - Publication Web personnalisée 11
- présentation de procédure
  - accès aux données XML 40
  - publication XSLT 31
- privilège d'accès étendu, Publication Web personnalisée 21
- privilèges d'accès 22

- process, commande de requête 65, 101
- protection des bases de données publiées 22
- publication statique, définition 11
- publication sur le Web
  - avec XSLT 18, 31, 57
  - codes d'erreurs des bases de données 115
  - configuration de 19
  - connexion à Internet ou à un intranet 20
  - objets de rubrique Multimédia 23, 35
  - protection des bases de données 22
  - utilisation du langage XML 17, 40
  - vidéos QuickTime 23
- Publication Web instantanée
  - définition 11
  - documentation 9
- Publication Web personnalisée
  - accès, Web 22, 68
  - activation dans le moteur de publication Web 23
  - activation dans une base de données 21
  - avec PHP 13
  - avec XML 13
  - avec XSLT 13, 30, 57
  - compte Invité 22
  - configuration de 19
  - définition 11
  - description 17, 18
  - nouvelles fonctionnalités 18
  - présentation 11
  - privilège d'accès étendu 21
  - restriction d'accès au serveur Web à l'aide des adresses IP 23
  - scripts 26
  - utilisant une adresse IP statique 20
  - utilisation des scripts 24
  - utilisation du langage XML 37
- publication Web personnalisée XML 13
- publication Web personnalisée XSLT 13

## Q

- query, paramètre de requête 106

## R

- récapitulatif de procédure
  - accès aux données XML 40
  - publication XSLT 31
- recid, paramètre de requête 107
- Reconnexion, script 22
- récupération des informations de modèle 101
- récupération des noms de scripts disponibles 101
- récupération des noms des modèles 100

- référence des chaînes de requête 95
- regex\_contains(), fonction d'extension 78
- <relatedset-definition>, élément 46
- relatedsets.filter, paramètre de requête 97, 108
- relatedsets.max, paramètre de requête 108
- remove\_session\_object(), fonction d'extension 72
- réponse XML
  - passage d'un modèle à un autre 54
- requête XML
  - spécification du modèle 54
- requêtes de rubriques de table externe 97, 108
- requêtes pour des données XML 40
- résolution des problèmes
  - accès aux documents XML 56
  - feuilles de style XSLT 35
  - Publication de sites Web personnalisée 88
- result, attribut 46
- <resultset>, élément 46
- rubriques de type Global
  - sessions de base de données, activation 73, 98
  - syntaxe 97
  - utilisation avec des sessions 73, 98

## S

### SAT

- voir Admin Console 23
- script, paramètre de requête 108
- script.param, paramètre de requête 109
- script.prefind, paramètre de requête 109
- script.prefind.param, paramètre de requête 110
- script.presort, paramètre de requête 110
- script.presort.param, paramètre de requête 110
- scriptnames, commande de requête 101
- scripts
  - astuces et considérations 25
  - comptes et privilèges 25
  - dans la Publication Web personnalisée 24
  - déclencheurs 27
  - Modifier le mot de passe 22
  - pour des requêtes XML 40
  - Reconnexion 22
  - sessions de base de données, activation 73
- scripts disponibles 101
- sécurité
  - chaînes de requête statiques, utilisation 62
  - comptes utilisateurs et mots de passe 23
  - consignes pour la protection de vos bases de

- données publiées 22
- documentation 13
- limitation d'accès depuis des adresses IP 23
- send\_email(), fonctions d'extension 73
- serveur Web
  - fichiers journaux 90
  - MIME, prise en charge des types 23
  - rôle dans requêtes XML 38
  - rôle, requêtes XSLT-CWP 30
- session\_encode\_url(), fonction d'extension 71
- session\_exists(), fonction d'extension 72
- sessions de base de données, activation 73, 98
- set\_cookie(), fonction d'extension 76
- set\_header(), fonction d'extension 75
- set\_session\_object(), fonction d'extension 72
- set\_session\_timeout(), fonction d'extension 72
- set\_status\_code(), fonction d'extension 75
- Shift\_JIS, codage 64
- Site Assistant
  - description 18
- Site Assistant. *Voir* XSLT Site Assistant
- sites Web
  - création à l'aide du moteur de publication Web 17
  - pages d'assistance FileMaker 9
  - suivi 90
  - test 88
- skip, paramètre de requête 111
- sortfield, paramètre de requête 111
- Sortie de données avec les feuilles de style 29
- sortorder, paramètre de requête 111
- spécification de la grammaire XML 61
- spécification du modèle au moment d'effectuer une requête sur les données XML 54
- SSL (Secure Sockets Layer), cryptage 23
- stockage d'informations dans les sessions 71, 72
- stylehref, paramètre de requête 112
- styletype, paramètre de requête 112
- suivi des sites Web 90
- suppression d'enregistrements dans une table externe 97
- syntaxe d'URL
  - feuilles de style XSLT 59
  - objets de type Multimédia dans les solutions XML 42
  - objets multimédia dans les solutions XSLT 60
  - requêtes XML 41

## T

- tables externes
  - ajout d'enregistrements 96
  - modèle 108
  - modification d'enregistrements 96
  - suppression d'enregistrements 97
  - tri d'enregistrements 108
- tester la publication PHP 35
- tests technologiques 35
- time-of-day, attribut 46
- token, paramètre de requête 66, 113
- traitement d'une requête du moteur de publication Web 12
- traitement des feuilles de style XSLT 101
- type, attribut 46

## U

- url\_decode(), fonction d'extension 77
- url\_encode(), fonction d'extension 77
- US-ASCII, codage 64
- User-Agent, vérification de l'en-tête 66
- UTF-8 (Unicode Transformation Format 8 bits)
  - format 42, 52
  - paramètre de codage 64
- utilisateurs Web
  - accès aux bases de données protégées 22, 68, 69
  - configuration pour l'accès aux solutions de publication Web personnalisée 20
  - utilisation de données d'une rubrique Multimédia 24

## V

- Valeurs, recherche dans les cases à cocher 78
- vCards, reformatage de données XML en 37
- vidéos QuickTime, publication sur le Web 23
- view, commande de requête 101

## W

- web\_server\_module\_log.txt, fichier journal 91
- wpc\_access\_log.txt, fichier 91

**X****XML**

- activation dans une base de données 21
- analyseurs 40, 52
- chaînes de requête 52, 93
- codage au format UTF-8 43, 52
- codage de texte URL 42
- définitions de type de document (DTD) 44, 45, 48
- dépannage de l'accès aux documents XML 56
- description 37
- espaces de nom 43
- filtrage des données 37
- FMPXMLLAYOUT, grammaire 49
- FMPXMLRESULT, grammaire 48
- fmresultset, grammaire 45
  - <datasource>, élément 45
  - <field-definition>, élément 45
  - <metadata>, élément 45
  - <relatedset-definition>, élément 46
  - <resultset>, élément 46
- génération données XML à partir de requête 38
- grammaires, description 43
- instructions pour le traitement d'une feuille de style XML 55
- ordre de traitement de requête 55
- récapitulatif des étapes d'accès aux données XML 40
- recherche de données 40
- spécifications XML 1.0 37
- utilisation de feuilles de style côté client 55
- XPath, instructions 66
- <xsl:stylesheet>, élément 62, 66, 67, 89, 83
- <xsl:output>, élément 64
- <xsl:param name="authenticated-xml-base-uri"/>, paramètre 69
- <xsl:param name="client-ip"/> paramètre 68
- <xsl:param name="client-password"/>, paramètre 68
- <xsl:param name="client-user-name"/>, paramètre 68
- <xsl:param name="request-query"/>, paramètre 67
- <xsl:param name="xml-base-uri"/>, paramètre 68
- <xsl:param>, élément 67
- <xsl:template>, élément 68, 69, 89
- <xsl:variable>, élément 69

**XSLT**

- activation dans une base de données 21
- cases à cocher, recherche de valeurs 78
- chaînes de format de date et d'heure 81
- chaînes de requête 60
- comparaison de chaînes, expressions régulières Perl 78
- conception de feuilles de style 57
- cookie, fonctions d'extension 76
- date, heure et jour, fonctions d'extension 79
- description 29
- emails, envoi 73
- espaces de nom 62
- état d'erreur des fonctions d'extension, vérification 83
- exemples de feuilles de style 29
- extensions JavaScript 83
- feuilles de style côté serveur 29, 57
- fonctions d'en-tête, utilisation 75
- fonctions d'extension pour FileMaker 66
- génération de pages, XSLT 30
- grammar, paramètre 61
- informations de modèle, utilisation 70
- manipulation de chaînes, fonction d'extension 77
- mise du contenu en mémoire tampon, utilisation 70
- paramètres XSLT propres à FileMaker 67
- récapitulatif de la procédure de publication 31
- référence des chaînes de requête 95
- requêtes XSLT-CWP 30
- résolution des problèmes sur les feuilles de style 35
- spécification XSLT 1.0 29
- utilisation des feuilles de style dans un programme ou un site Web 34
- XSLT Site Assistant, utilisation 32
- xslt-template-files, dossier 31, 34, 70
- XSLT Site Assistant
  - démarrage 33
  - description 32
  - feuilles de style générées, description 34
  - procédure préalable à l'utilisation 32
  - utilisation 33
- <?xslt-cwp-buffer buffer-content="true"?>, instruction de traitement 71
- <?xslt-cwp-query?>, instruction de traitement 58, 62
- xslt-template-files, dossier 31, 34, 70