FileMaker® Server 12

Publication Web personnalisée avec PHP

© 2007-2012 FileMaker, Inc. Tous droits réservés.

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

FileMaker et Bento sont des marques commerciales de FileMaker, Inc. déposées aux Etats-Unis et dans d'autres pays. Le logo du dossier et le logo Bento sont des marques 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, des sociétés, des adresses emails ou des URL 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

Préface 6

Présentation de ce guide 6

Chapitre 1
Introduction à la Publication Web personnalisée 7

A propos du moteur de publication Web 8

Traitement d'une requête du moteur de publication Web 8

Publication Web personnalisée avec PHP 9

Publication Web personnalisée avec XML 9

Comparaison entre PHP et XML 9

Raisons de choisir PHP 9

Raisons de choisir XML 10

Chapitre 2
A propos de la fonction de publication Web personnalisée avec PHP 11

Fonctions clés de la publication Web personnalisée avec PHP 11

Configuration de la publication Web personnalisée 12

Configuration requise pour la publication d'une base de données à l'aide de la Publication Web personnalisée 12

Eléments devant être employés par les utilisateurs Web pour accéder à une solution de publication Web personnalisée 12

Connexion à Internet ou à un intranet 13

Installation manuelle de l'API FileMaker pour PHP 13

Etape suivante 14

Chapitre 3
Préparation des bases de données pour la Publication Web personnalisée 15

Activation de la publication Web personnalisée avec PHP pour les bases de données 15

Création de modèles pour la publication Web personnalisée avec PHP 16

Protection de vos bases de données publiées 16

Accès à une base de données protégée 17

Publication du contenu des rubriques Conteneur sur le Web 18

Rubriques Conteneur imbriquées dans une base de données 18

Rubriques Conteneur avec fichiers référencés 18

Rubriques Conteneur avec données stockées en externe 19

Affichage des objets d'une rubrique Conteneur par les utilisateurs Web 21

Scripts FileMaker et Publication Web personnalisée 21

Astuces et considérations à propos des scripts 21

Comportement des scripts dans les solutions de Publication Web personnalisée 23

Déclencheurs de scripts et solutions de Publication Web personnalisée 24

Chapitre 4
Présentation de la Publication Web personnalisée avec PHP 25

Fonctionnement du moteur de publication Web avec les solutions PHP 25

Etapes générales de la publication Web personnalisée avec PHP 25

Chapitre 5
Utilisation de l'API FileMaker pour PHP 28
Où trouver des informations supplémentaires 28
Référence API FileMaker pour PHP 28
Didacticiel API FileMaker pour PHP 29
Exemples d'API FileMaker pour PHP 29
Utilisation de la classe FileMaker 29
Objets de classe FileMaker 29
Objets de commande FileMaker 30
Connexion à une base de données FileMaker 30
Utilisation des enregistrements 31
Création d'un enregistrement 31
Duplication d'un enregistrement 32
Edition d'un enregistrement 32
Suppression d'un enregistrement 32
Exécution des scripts FileMaker 33
Obtention de la liste des scripts disponibles 33
Exécution d'un script FileMaker 33
Exécution d'un script avant l'exécution d'une commande 33
Exécution d'un script avant le tri d'un ensemble de résultats 34
Exécution d'un script après la génération d'un ensemble de résultats 34
Ordre d'exécution du script 34
Utilisation des modèles FileMaker 35
Utilisation des tables externes 35
Création d'une liste des tables externes définies sur un modèle spécifique 35
Obtention des noms de tables externes d'un objet de résultat spécifique 36
Obtention d'informations sur les tables externes d'un modèle spécifique 36
Obtention d'informations sur une table externe spécifique 36
Obtention du nom de table d'une table externe 36
Obtention des enregistrements externes d'un enregistrement spécifique 36
Création d'un nouvel enregistrement dans une table externe 37
Suppression d'un enregistrement d'une table externe 37
Utilisation des listes de valeurs 37
Obtention des noms de toutes les listes de valeurs d'un modèle spécifique 37
Obtention d'un tableau de toutes les listes de valeurs d'un modèle spécifique 38
Obtention des valeurs pour une liste de valeurs nommée 38
Exécution de requête 39
Utilisation de la commande Rechercher tout 40
Utilisation de la commande Rechercher au moins 40
Utilisation de la commande Rechercher 40
Utilisation d'une commande Recherche composée 41
Traitement des enregistrements d'un ensemble de résultats 43
Filtrage des lignes des tables externes renvoyées par les requêtes 44
Prévalidation des commandes, des enregistrements et des rubriques 44
Prévalidation des enregistrements dans une commande 46
Prévalidation des enregistrements 46
Prévalidation des rubriques 46
Traitement des erreurs de validation 46

Gestion des erreurs 48

Chapitre 6
Stockage, test et suivi d'un site 49

Stockage d'un site de Publication Web personnalisée 49
Test d'un site de Publication Web personnalisée 50
Suivi de votre site 51
Utilisation des journaux d'erreurs et d'accès au serveur Web 51
Utilisation du journal du moteur de publication Web 52
Utilisation du journal des erreurs du module de serveur Web 54
Utilisation des journaux Tomcat 54
Résolution des problèmes sur votre site 54

Annexe A
Codes d'erreur de la Publication Web personnalisée avec PHP 55

Numéros des codes d'erreur des bases de données FileMaker 55
Numéros des codes d'erreur des composants PHP 62

Index 63

Préface

Présentation de ce guide

Ce guide part de l'hypothèse que vous avez déjà une certaine expérience dans le domaine du langage PHP, 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 Conteneur. Pour plus d'informations sur FileMaker Pro, reportez-vous à l'aide de FileMaker Pro.

Concernant la Publication Web personnalisée avec le langage PHP sous FileMaker Server, ce guide donne des informations sur :

- les éléments nécessaires au développement d'une solution de publication Web personnalisée à l'aide du langage PHP ;
- la façon de publier des bases de données à l'aide du langage PHP ;
- les éléments que les utilisateurs Web doivent employer pour accéder à une solution de publication Web personnalisée ;
- la façon d'utiliser l'API FileMaker pour PHP à partir de bases de données hébergées par FileMaker Server.

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	*Guide de démarrage de FileMaker Server* Aide FileMaker Server
Publication Web instantanée	*Guide de la Publication Web instantanée FileMaker*
Publication Web personnalisée avec PHP	*Publication Web personnalisée FileMaker Server avec PHP* (ce manuel)
Publication Web personnalisée avec XML	*Publication Web personnalisée FileMaker Server avec XML*
Installation et paramétrage des pilotes ODBC et JDBC, et utilisation d'ODBJ et de JDBC	*Guide ODBC et JDBC FileMaker*

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, FileMaker Server ou FileMaker Server Advanced. 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 vos données ne changent pas souvent 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 :

- intégrer votre base de données à un autre site Web ;
- déterminer la façon dont les utilisateurs interagissent avec les données ;
- 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. Comme vous codez les pages Web PHP vous-même, vous avez le contrôle complet de l'interface utilisateur et du mode d'interaction entre l'utilisateur et les données.

- Publication Web personnalisée avec XML : Utilisez la publication de données XML pour échanger des données FileMaker avec d'autres sites Web et applications. 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, et utiliser les données obtenues de la façon qui vous plaît.

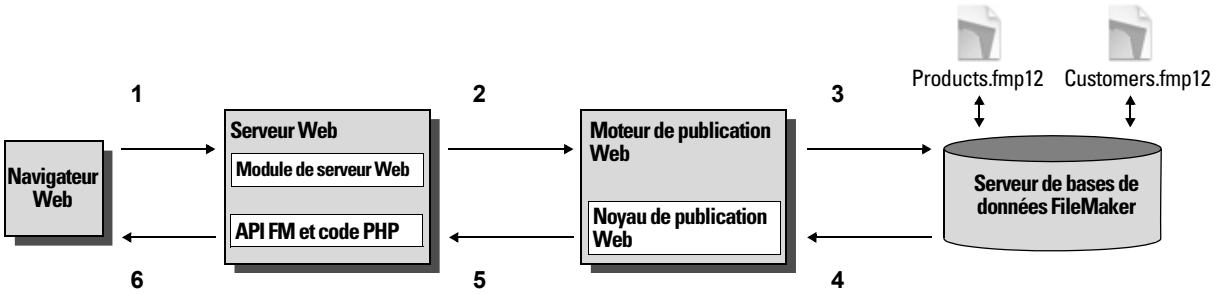
A propos du moteur de publication Web

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 : Les utilisateurs Web accèdent à votre solution de Publication Web personnalisée en cliquant sur un lien HREF ou en saisissant une URL (Uniform Resource Locator) qui indique l'adresse du serveur Web, ainsi qu'une requête qui contient la chaîne de requête FileMaker. Le moteur de publication Web renvoie les données XML spécifiées dans la requête de chaîne.

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 Web 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.

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 *Manuel de l'utilisateur 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.

Publication Web personnalisée avec XML

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 alors exporter les données XML vers d'autres applications.

Comparaison entre PHP et XML

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.

Raisons de choisir XML

- 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 est une norme 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.
- Vous pouvez utiliser FMPXMLRESULT pour accéder aux données XML en utilisant la Publication Web personnalisée et pour l'export XML à partir de bases de données FileMaker Pro.

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

Chapitre 2
A propos de la fonction de publication Web personnalisée avec PHP

La publication Web personnalisée avec PHP vous permet d'utiliser le langage de script PHP pour intégrer des données issues des bases de données FileMaker à vos modèles de page Web personnalisés. La publication Web personnalisée avec PHP fournit l'API FileMaker pour PHP, qui est une classe PHP créée par FileMaker qui accède aux bases de données hébergées par FileMaker Server. Cette classe PHP se connecte au moteur de publication Web de FileMaker Server et rend les données disponibles pour le moteur PHP du serveur Web.

Fonctions clés de la publication Web personnalisée avec PHP

- Créez des applications Web qui utilisent le langage de script PHP Open Source. Utilisez la version de PHP 5 prise en charge par FileMaker Server ou votre propre version de PHP 5. (Si vous décidez d'utiliser votre propre version de PHP, reportez-vous à la section « Installation manuelle de l'API FileMaker pour PHP », page 13.)
- Hébergez des bases de données sur FileMaker Server. FileMaker Pro n'est pas requis pour la publication Web personnalisée car FileMaker Server héberge les bases de données.
- Rédigez un code PHP permettant de créer, de supprimer, d'éditer et de dupliquer les enregistrements d'une base de données FileMaker hébergée. Votre code peut procéder à la validation des rubriques et des enregistrements avant de valider les modifications dans la base de données hébergée.
- Rédigez un code PHP permettant d'accéder aux modèles, aux tables externes, aux listes de valeurs et aux rubriques liées. 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 16.
- Rédigez un code PHP permettant d'exécuter des scripts complexes comportant plusieurs actions. FileMaker prend en charge plus de 65 actions de script dans la publication Web personnalisée. Consultez la section « Scripts FileMaker et Publication Web personnalisée », page 21.
- Rédigez un code PHP permettant d'effectuer des requêtes complexes.

Configuration de la publication Web personnalisée

Cette section décrit la configuration requise pour développer une solution de Publication Web personnalisée à l'aide de PHP, ce dont les utilisateurs Web ont besoin pour accéder à une solution de Publication Web personnalisée, ainsi que l'impact possible de l'hébergement d'une solution de publication Web sur votre serveur.

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 PHP, il vous faut :

- un déploiement FileMaker Server, comprenant trois composants ;
  - un serveur Web, Microsoft IIS (Windows) ou Apache (Mac OS) (le module de serveur Web FileMaker est installé sur le serveur Web) ;
  - le moteur de publication Web FileMaker ;
  - le serveur de bases de données FileMaker.
- PHP installé sur le serveur Web. FileMaker Server peut installer la version prise en charge de PHP 5 ou vous pouvez utiliser votre propre version. La version minimale requise de PHP sous Mac OS X v10.6 est PHP 5.3.3. La version minimale requise de PHP sous Windows est PHP 5.3.5. Pour plus d'informations sur PHP, consultez le site http://php.net. La version de PHP installée sur le serveur Web doit prendre en charge les fonctions cURL (bibliothèques d'URL du client). Pour plus d'informations sur cURL, consultez le site http://php.net/curl.

  Important Lorsque vous installez la version de PHP 5 prise en charge par FileMaker Server, elle n'apparaît pas dans l'outil Server Admin de Mac OS X ; elle n'est pas supposée le faire. Si vous utilisez l'outil Server Admin de Mac OS X pour activer PHP, vous désactivez la version de PHP 5 prise en charge par FileMaker Server et activez votre propre version de PHP.

- 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*.

Eléments devant être employés par les utilisateurs Web pour accéder à une solution de publication Web personnalisée

Pour accéder à une solution de publication Web personnalisée utilisant le langage PHP, 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.

Installation manuelle de l'API FileMaker pour PHP

Lorsque vous installez FileMaker Server, vous avez la possibilité d'installer la version de PHP prise en charge par FileMaker (PHP 5). Si un moteur PHP est déjà installé et configuré et que vous souhaitez uniquement ajouter l'API FileMaker pour PHP, installez manuellement la classe API FileMaker pour PHP pour qu'elle soit disponible dans vos scripts PHP.

Si vous n'avez pas installé la version de PHP prise en charge par FileMaker, veillez à effectuer les tâches de configuration suivantes dans votre version du moteur PHP :

- Activez le module cURL dans php.ini.
- Indiquez l'emplacement de l'API FileMaker pour PHP dans la variable include_path de php.ini.
- Si vous accédez à des bases de données contenant des dates et des heures, installez le package Pear de date. Pour plus d'informations, consultez la rubrique : http://pear.php.net/package/date/

Remarque FileMaker Server a été testé avec la version 5.3.3 de PHP pour Mac OS X v10.6 et avec la version 5.3.5 de PHP pour Windows. Pour des résultats optimums, utilisez la version de PHP appropriée.

Pour que l'API FileMaker pour PHP soit accessible dans vos scripts PHP

Lorsque vous avez installé FileMaker Server, l'API FileMaker pour PHP a été incluse sous la forme d'un fichier .zip à l'emplacement suivant :

- Pour IIS (Windows) :
  `<disque>:\Program Files\FileMaker\FileMaker Server\Web Publishing\FM_API_for_PHP_Standalone.zip`
  où `<disque>` correspond au disque sur lequel réside le composant Serveur Web de votre déploiement FileMaker Server.
- Pour Apache (Mac OS) :
  `/Bibliothèque/FileMaker Server/Web Publishing/FM_API_for_PHP_Standalone.zip`

Le fichier FM_API_for_PHP_Standalone.zip contient un fichier nommé FileMaker.php et un dossier nommé FileMaker. Décompressez ce fichier et copiez le fichier FileMaker.php ainsi que le dossier FileMaker dans l'un des emplacements suivants :

- le dossier racine du serveur Web dans lequel résident vos scripts PHP ;
  - Pour IIS (Windows) : `<disque>:\Inetpub\wwwroot` où `<disque>` est le disque sur lequel réside le composant Moteur de publication Web de votre déploiement de serveur FileMaker.
  - Pour Apache (Mac OS) : `/Bibliothèque/WebServer/Documents`
- l'un des répertoires include_path de votre installation de PHP. L'emplacement par défaut sous Mac OS X est `/usr/lib/php`.

Etape suivante

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

- 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 avec PHP pour les bases de données », page 15.
- Pour savoir comment accéder aux données des bases de données FileMaker à l'aide de l'API FileMaker pour PHP, reportez-vous au chapitre 5, « Utilisation de l'API FileMaker pour PHP ».

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 avec PHP pour les bases de données

Vous devez activer la publication Web personnalisée avec PHP dans chaque base de données que vous souhaitez publier. Sinon, les utilisateurs Web ne pourront pas utiliser la publication Web personnalisée pour accéder à la base de données, même si elle est hébergée par un serveur FileMaker 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 d'accès Accès intégral ou Gérer les autorisations étendues.
2. Attribuez le privilège d'accès étendu **fmphp** à un ou plusieurs jeux de privilèges d'accès pour activer la publication Web personnalisée avec PHP.
3. Attribuez les jeux de privilèges d'accès avec privilège d'accès étendu Publication Web personnalisée aux comptes concernés (par exemple, les comptes Admin et Invité).

   Important 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 autres qu'alphanumériques tels qu'un point d'exclamation (!) ou le symbole de pourcentage (%). Les deux-points (:) sont interdits. Pour plus d'informations sur la configuration de comptes, consultez l'aide de FileMaker Pro.
4. A l'aide de l'Admin Console de FileMaker Server, vérifiez que l'hébergement de la base de données est correctement configuré, et qu'elle est accessible au serveur FileMaker. Pour obtenir des instructions, consultez l'aide de FileMaker Server.

Remarque La publication Web personnalisée avec PHP n'utilisant pas des sessions de base de données permanentes, les références à une source de données ODBC externe dans le graphe de liens de FileMaker Pro peuvent limiter les fonctionnalités disponibles pour votre solution PHP. Si votre base de données accède aux données d'une source de données SQL externe, vous ne pourrez probablement pas mettre à jour les données d'enregistrement de la table externe.

Création de modèles pour la publication Web personnalisée avec PHP

La publication Web personnalisée avec PHP ne fournit pas un accès direct aux données d'une base de données FileMaker Pro, mais utilise les modèles définis dans les bases de données. Bien qu'il n'y ait pas d'obligation de créer un modèle unique pour la publication Web personnalisée avec PHP, la création d'un modèle spécifique pour une solution PHP peut présenter des avantages :

- Améliorez la performance en créant un modèle limité aux rubriques, aux étiquettes, aux calculs et aux tables externes que vous avez besoin d'intégrer à la solution PHP.
- Simplifiez votre code PHP en réduisant le traitement des données car les enregistrements contiennent moins de rubriques.
- Séparez le travail de conception d'interface des données afin d'adapter l'interface à l'utilisateur Web.

Protection de vos bases de données publiées

La publication Web personnalisée avec PHP vous permet de restreindre l'accès à vos bases de données publiées. Vous pouvez utiliser les méthodes suivantes :

- Demander des mots de passe pour les comptes de base de données utilisés pour la publication Web personnalisée avec PHP.
- Activer le privilège étendu Publication Web personnalisée avec PHP uniquement dans les jeux de privilèges d'accès auxquels vous souhaitez autoriser l'accès.
- Désactiver la publication Web personnalisée avec PHP pour une base de données spécifique en désélectionnant le privilège étendu fmphp pour tous les jeux de privilèges d'accès de cette base de données. Consultez l'Aide FileMaker Pro.
- Activer ou désactiver la publication Web personnalisée pour toutes les solutions de publication Web personnalisée dans le moteur de publication Web à l'aide de l'Admin Console de FileMaker Server. Reportez-vous au *Guide de démarrage de FileMaker Server* et à 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. Par exemple, indiquez 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 vos navigateurs 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.

Accès à une base de données protégée

Lorsqu'un utilisateur Web accède à une base de données en utilisant une solution PHP, le code PHP doit fournir les informations d'authentification requises pour accéder à la base de données à l'aide de l'API FileMaker pour PHP. Si le compte Invité de la base de données est désactivé ou si le privilège étendu **fmphp** n'est pas activé pour ce compte, l'API FileMaker pour PHP renvoie une erreur et votre code PHP doit fournir les informations de connexion de l'utilisateur.

Le didacticiel de l'API FileMaker pour PHP inclut un exemple montrant comment utiliser la méthode `setProperty()` pour définir le nom d'utilisateur et le mot de passe d'une base de données protégée. Consultez la section « Didacticiel API FileMaker pour PHP », page 29.

La liste suivante récapitule le processus qui se produit lors de l'utilisation de la publication Web personnalisée pour accéder à une base de données :

- Si aucun mot de passe n'a été attribué à un compte activé pour la publication Web personnalisée, la solution PHP doit fournir le nom de compte uniquement.
- Si le compte Invité est désactivé, la solution PHP doit fournir un nom de compte et un mot de passe. La solution PHP peut soit inviter l'utilisateur Web à fournir un nom de compte et un mot de passe, soit stocker le nom de compte et le mot de passe dans le code PHP. Le privilège d'accès étendu **fmphp** doit être activé pour le compte.
- Si le compte Invité est activé et que le privilège d'accès étendu **fmphp** est également activé :
  - La solution PHP n'a pas besoin d'inviter l'utilisateur Web à fournir un nom de compte et un mot de passe lors de l'ouverture d'un fichier. Tous les utilisateurs Web se connectent automatiquement avec le compte Invité et héritent des privilèges d'accès de celui-ci.
  - Les autorisations 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.
- La solution PHP peut utiliser l'action de script Reconnexion pour autoriser les utilisateurs à se connecter en utilisant un compte différent (par exemple, pour passer d'un compte Invité à un compte disposant de privilèges d'accès supplémentaires). Consultez l'aide FileMaker Pro. Toutefois, les connexions PHP n'utilisant pas des sessions de base de données permanentes, la solution PHP doit stocker le nom de compte et le mot de passe afin de les utiliser pour chaque requête suivante.

  Remarque Par défaut, les utilisateurs Web ne peuvent pas modifier le mot de passe de leur compte depuis un navigateur Web. Vous pouvez activer 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 un navigateur. Consultez l'aide FileMaker Pro.

Publication du contenu des rubriques Conteneur sur le Web

Le contenu d'une rubrique Conteneur peut être imbriqué dans la base de données, lié par référence à l'aide d'un chemin relatif ou stocké en externe.

Rubriques Conteneur imbriquées dans une base de données

Si une rubrique Conteneur stocke les fichiers proprement dits dans la base de données FileMaker, appliquez la procédure suivante pour utiliser les objets de rubrique Conteneur dans une solution PHP.

- Utilisez l'API FileMaker pour PHP afin de définir l'objet de base de données (`$fm`) à l'aide d'informations d'authentification adaptées (nom de compte et mot de passe).

```
$fm = new FileMaker();
$fm->setProperty('database', $databaseName);
$fm->setProperty('username', $userName);
$fm->setProperty('password', $passWord);
```

- Utilisez des balises HTML correctes pour indiquer le type d'objet compatible Web contenu dans la rubrique Conteneur, puis créez une chaîne d'URL représentant le chemin du fichier de l'attribut source de la balise HTML.

```
<IMG src="img.php?-url=<?php echo urlencode($record->getField('Cover
Image')); ?>">
```

- Utilisez ensuite la méthode `getContainerData()` pour récupérer l'objet de rubrique de conteneur.

```
echo $fm->getContainerData($_GET['-url']);
```

Le didacticiel de l'API FileMaker pour PHP inclut des exemples montrant comment utiliser les rubriques Conteneur. Consultez la section « Didacticiel API FileMaker pour PHP », page 29.

Remarque Le moteur de publication Web prend en charge un téléchargement progressif des fichiers audio (.mp3), des fichiers vidéo (.mov, .mp4 et .avi recommandés) et des fichiers PDF pour les conteneurs interactifs. Par exemple, un utilisateur Web peut commencer à visionner un film, même si la totalité du fichier vidéo n'est pas encore téléchargée. Pour rendre ce téléchargement progressif possible, vous devrez créer les fichiers en utilisant des options qui prennent en charge la lecture en direct ou qui sont optimisées pour un affichage sur le Web. Par exemple, créez les fichiers PDF à l'aide de l'option « Optimisé pour un affichage Web ».

Rubriques Conteneur avec fichiers référencés

Si une rubrique Conteneur stocke une référence de fichier, vous pouvez utiliser la méthode `getContainerData()` pour récupérer les objets de rubrique Conteneur à partir de la base de données dans votre code PHP ou vous pouvez utiliser la méthode `getContainerDataURL()` pour récupérer une URL entièrement qualifiée pour l'objet de rubrique Conteneur.

Vous devez également suivre cette procédure pour publier les fichiers référencés à l'aide du moteur de publication Web :

1. Stockez les fichiers d'objet Conteneur dans le dossier Web, situé dans le dossier FileMaker Pro.
2. Dans FileMaker Pro, insérez les objets dans la rubrique Conteneur et sélectionnez l'option **Stocker uniquement la référence au fichier**.

3. Copiez ou déplacez les fichiers d'objet référencé du dossier Web vers le même emplacement relatif, dans le dossier racine du logiciel de serveur.
   - Pour IIS (Windows) : `<disque>:\Inetpub\wwwroot` où `<disque>` est le disque sur lequel réside le composant Moteur de publication Web de votre déploiement de serveur FileMaker.
   - Pour Apache (Mac OS) : `/Bibliothèque/WebServer/Documents`

Remarques

- Pour les objets Conteneur stockés sous forme de référence externe, votre serveur Web doit être configuré pour prendre en charge les types MIME (Multipurpose Internet Mail Extensions) correspondant aux types de fichiers que vous souhaitez publier, par exemple des vidéos. Votre serveur Web détermine la prise en charge des types MIME actuellement enregistrés pour Internet. Le moteur de publication Web ne change pas la prise en charge MIME par le serveur Web. Pour plus d'informations, consultez la documentation de votre serveur Web.
- Toutes les séquences QuickTime sont stockées dans une rubrique Conteneur sous forme de référence.

Rubriques Conteneur avec données stockées en externe

Si une rubrique stocke des objets en externe (en d'autres termes, si vous avez sélectionné **Stocker les données de conteneur en externe** dans la boîte de dialogue Options de rubrique, votre code PHP doit utiliser la méthode `getContainerDataURL()` pour récupérer une URL entièrement qualifiée pour l'objet de rubrique Conteneur.

Utilisez l'API FileMaker pour PHP pour définir l'objet de base de données avec les informations d'authentification appropriées (nom de compte et mot de passe), puis utilisez la méthode `getContainerDataURL()`.

Exemple d'images utilisant la balise HTML img

```
$fm=new FileMaker($database, $hostspec, $user, $password);
$findCommand = $fm->newFindCommand($layout);
$findCommand->addFindCriterion('type', 'png');
$result = $findCommand->execute();
$records = $result->getRecords();
foreach ($records as $record) {
        echo $record->getField('container').'<br>';
        // Pour les images, utilisez la balise HTML img
        echo '<img src="'.$fm->
          getContainerDataURL($record->getField('container')) .'">';
        break;
        }
```

Exemple de données incorporées utilisant la balise HTML embed

```
$fm=new FileMaker($database, $hostspec, $user, $password);
$findCommand = $fm->newFindCommand($layout);
$findCommand->addFindCriterion('type', 'pdf');
$result = $findCommand->execute();
$records = $result->getRecords();
foreach ($records as $record) {
        echo $record->getField('container').'<br>';
        // Pour les vidéos et les fichiers PDF, utilisez la balise HTML embed
        //echo '<embed src="'.$fm->
          getContainerDataURL($record->getField('container')) .'">';
        break;
        }
```

Si une rubrique Conteneur stocke des objets en externe, utilisez l'assistant Téléchargement de base de données pour transférer les fichiers de base de données depuis le système de fichiers client vers FileMaker Server. L'assistant Téléchargement de base de données transfère la base de données et les objets de rubrique Conteneur vers les dossiers adéquats pour votre serveur pour l'hébergement. Consultez l'aide FileMaker Server pour plus d'informations sur l'assistant Téléchargement de base de données. Consultez l'aide FileMaker Pro pour plus d'informations sur la configuration de la rubrique pour stocker des données en externe.

Si vous téléchargez manuellement une base de données qui utilise une rubrique Conteneur avec des objets stockés en externe, vous devez exécuter ces opérations pour publier les objets de conteneurs en utilisant le moteur de publication Web.

Pour télécharger manuellement une base de données :

1. Placez le fichier de base de données à l'emplacement adéquat sur le serveur. Placez les fichiers de base de données FileMaker Pro que FileMaker Server doit ouvrir (ou leurs raccourcis (Windows) ou alias (MacOS)) dans les dossiers suivants :
   - Windows (32 bits) : `[lecteur]:\Program Files\FileMaker\FileMaker Server\Data\Databases\`
   - Windows (64 bits) : `[lecteur]:\Program Files (x86)\FileMaker\FileMaker Server\Data\Databases\`
   - Mac OS : `/Bibliothèque/FileMaker Server/Data/Databases/`

   Vous pouvez également placer les fichiers dans un dossier de bases de données supplémentaire défini en option.

2. Dans le dossier dans lequel vous avez placé la base de données, créez un dossier nommé RC_Data_FMS, s'il n'existe pas encore.

3. Dans le dossier RC_Data_FMS, créez un dossier dont le nom correspond à celui de votre base de données. Par exemple, si votre base de données s'appelle Clients, créez un dossier nommé Clients. Placez les objets stockés en externe dans le dossier que vous venez de créer.

   Remarque Lorsque les bases de données sont hébergées sur FileMaker Server, plusieurs bases de données ne peuvent en aucun cas partager un dossier commun d'objets de conteneur. Les objets Conteneur de chacune des base de données doivent se trouver dans un dossier identifié par ce nom de base de données.

4. Pour les fichiers partagés depuis Mac OS, changez les fichiers de façon à ce qu'ils appartiennent au groupe **fmsadmin**.

   Pour plus d'informations sur le téléchargement manuel de bases de données, consultez l'Aide de FileMaker Server.

Remarque Le moteur de publication Web prend en charge un téléchargement progressif des fichiers audio (.mp3), des fichiers vidéo (.mov, .mp4 et .avi recommandés) et des fichiers PDF pour les conteneurs interactifs. Par exemple, un utilisateur Web peut commencer à visionner un film, même si la totalité du fichier vidéo n'est pas encore téléchargée. Pour rendre ce téléchargement progressif possible, vous devrez créer les fichiers en utilisant des options qui prennent en charge la lecture en direct ou qui sont optimisées pour un affichage sur le Web. Par exemple, créez les fichiers PDF à l'aide de l'option « Optimisé pour un affichage Web ».

Affichage des objets d'une rubrique Conteneur par les utilisateurs Web

Lorsque vous publiez une base de données à l'aide d'un moteur de publication Web, les limites suivantes s'appliquent aux données des rubriques Conteneur :

- Les utilisateurs Web ne peuvent pas modifier ni compléter le contenu des rubriques Conteneur. Les utilisateurs Web ne peuvent pas utiliser les rubriques Conteneur pour télécharger des objets vers la base de données.
- Pour les bases de données utilisant une rubrique Conteneur dans laquelle les miniatures sont activées, le moteur de publication Web télécharge l'intégralité du fichier et pas uniquement une miniature.

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 une série de tâches. Les scripts FileMaker permettent également d'exécuter des tâches non prises en charge autrement, telles que l'utilisation de l'action de script Modifier le mot de passe pour permettre aux utilisateurs Web de modifier les mots de passe à partir d'un navigateur.

FileMaker prend en charge plus de 65 actions de script dans la publication Web personnalisée. Pour visualiser les actions de script non prises en charge, sélectionnez **Publication Web personnalisée** 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 par la publication Web personnalisée. 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 23. 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.

Gardez à l'esprit les astuces et considérations suivantes :

- Utilisez les comptes et les privilèges 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. 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 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 en limitant 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 supplémentaires pour enregistrer les 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 telles que 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 seront 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 », vous savez que l'utilisateur actuel accède à votre base de données à l'aide de la publication Web personnalisée. Pour plus d'informations sur les fonctions, consultez l'Aide de FileMaker Pro.
- Une fois vos fichiers convertis, ouvrez chaque script susceptible d'être exécuté par les utilisateurs Web et sélectionnez **Publication Web** dans la liste Afficher la compatibilité située dans la fenêtre Modifier le script pour vous assurer que le script s'exécutera correctement avec la publication Web instantanée.

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 toutes les fenêtres mais ne quitte pas le navigateur Web.
Autor. 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 rubrique "Astuces et considérations à propos des scripts », page 21. Remarque 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 ces actions de script soient prises en charge par la publication Web personnalisée, il faut éviter de les 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 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 déclencheurs de scripts peuvent être activés à la fois par des actions de script et des actions de l'utilisateur (clic sur une rubrique par exemple). 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. Par contre, si un script entraîne le déplacement du focus sur la rubrique, le déclencheur de script SurEntreeObjet est alors activé. Pour obtenir des informations sur les déclencheurs de script, consultez l'aide de FileMaker Pro.

Remarque Pour FileMaker Pro 12, la boîte de dialogue Options de fichier a été modifiée. Ainsi, pour spécifier que vous souhaitez exécuter un script à l'ouverture d'un fichier, vous devez utiliser le déclencheur de script SurOuverturePremiereFenetre. Ainsi, pour spécifier que vous souhaitez exécuter un script à la fermeture d'un fichier, vous devez utiliser le déclencheur de script SurFermetureDerniereFenetre.

Chapitre 4
Présentation de la Publication Web personnalisée avec PHP

L'API FileMaker pour PHP permet d'intégrer des données à partir de bases de données FileMaker Pro dans des solutions PHP. Ce chapitre décrit le fonctionnement de PHP avec le moteur de publication Web personnalisé de FileMaker Server. Pour plus d'informations sur l'API FileMaker pour PHP, reportez-vous au chapitre 5, « Utilisation de l'API FileMaker pour PHP ».

Fonctionnement du moteur de publication Web avec les solutions PHP

FileMaker Server se compose de trois éléments : un serveur Web, le moteur de publication Web et le serveur de bases de données. (Ces éléments peuvent être déployés sur un, deux ou trois ordinateurs. Pour plus d'informations, consultez le *Guide de démarrage de FileMaker Server*.) FileMaker Server héberge la solution PHP lorsque vous placez les fichiers PHP sur le serveur Web d'installation du moteur PHP.

- Lorsqu'un utilisateur Web ouvre une solution PHP, le serveur Web achemine la demande vers le moteur PHP, qui traite le code PHP.
- Si le code PHP contient des appels vers l'API FileMaker pour PHP, ces appels sont interprétés et envoyés sous la forme de requêtes vers le moteur de publication Web.
- Le moteur de publication Web recherche les données dans les bases de données hébergées par le serveur de bases de données.
- Le serveur de bases de données envoie les données demandées au moteur de publication Web.
- Le moteur de publication Web envoie les données au moteur PHP sur le serveur Web en réponse à l'appel d'API.
- La solution PHP traite les données et les affiche pour l'utilisateur Web.

Etapes générales de la publication Web personnalisée avec PHP

La présente section récapitule la procédure d'utilisation de la publication Web personnalisée avec PHP :

1. Vérifiez que **Activer la publication PHP** est coché dans l'Admin Console. Reportez-vous au *Guide de démarrage de FileMaker Server*.

2. Dans l'Admin Console, choisissez le volet **Bases de données** et veillez à ce que le privilège étendu **fmphp** soit activé pour la Publication Web personnalisée avec PHP et ce, pour chaque base de données FileMaker que vous publiez.

   Si nécessaire, utilisez FileMaker Pro pour activer la Publication Web personnalisée pour une base de données. Consultez la section chapitre 3, « Préparation des bases de données pour la Publication Web personnalisée ».

   Remarque Lors de la création de solutions PHP 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. Utilisez les outils de création PHP pour créer votre solution PHP, en intégrant les fonctions de l'API FileMaker à votre code PHP pour accéder à vos données FileMaker. Voir chapitre 5, « Utilisation de l'API FileMaker pour PHP ».

4. Copiez ou déplacez la structure et les fichiers de répertoire de votre site vers le dossier racine du serveur Web.
   - Pour IIS (Windows) : `<disque>:\Inetpub\wwwroot` où `<disque>` est le disque sur lequel réside le composant Moteur de publication Web de votre déploiement de serveur FileMaker.
   - Pour Apache (Mac OS) : `/Bibliothèque/WebServer/Documents`

5. Si une rubrique de base de données de conteneur stocke une référence à un fichier et non le fichier lui-même, l'objet Conteneur référencé doit alors être stocké dans le dossier FileMaker Pro Web lorsque l'enregistrement est créé ou modifié. Vous devez copier ou déplacer l'objet dans un dossier possédant le même emplacement relatif dans le dossier racine du logiciel de serveur Web.

   Consultez la section « Publication du contenu des rubriques Conteneur sur le Web », page 18.

6. Assurez-vous que les mécanismes de sécurité pour votre site ou votre programme sont mis en œuvre.

7. Testez votre site en utilisant les comptes et les privilèges d'accès définis pour les utilisateurs Web.

8. Mettez le site à la disposition des utilisateurs. L'URL saisie par l'utilisateur Web utilise le format suivant :

```
http://<serveur>/<chemin_site>
```

- `<serveur>` est l'ordinateur sur lequel se trouve FileMaker Server
- `<chemin_site>` est le chemin d'accès relatif vers la page d'accueil de votre site, déterminé par la structure de répertoire que vous avez utilisée à l'étape 4 ci-dessus.

Par exemple, si l'adresse de votre serveur Web est 192.168.123.101 et que la page d'accueil de votre site se trouve sur le serveur Web à l'emplacement c:\Inetpub\wwwroot\customers\index.php, l'utilisateur Web doit saisir l'URL suivante :

```
http://192.168.123.101/customers/index.php
```

Remarque PHP 5 utilise le codage Latin-1 (ISO-8859-1). FileMaker Server renvoie des données Unicode (UTF-8). Utilisez l'Admin Console de FileMaker Server pour spécifier le codage de caractères par défaut de votre site. Pour les sites PHP, vous pouvez spécifier UTF-8 ou ISO-8859-1 ; UTF-8 est recommandé. Spécifiez le même paramètre pour l'attribut `charset` dans la section `<HEAD>` de les fichiers PHP de votre site.

Pour plus d'informations sur le déploiement et l'utilisation d'une solution PHP, consultez la rubrique chapitre 6, « Stockage, test et suivi d'un site ».

Chapitre 5
Utilisation de l'API FileMaker pour PHP

L'API FileMaker pour PHP utilise une classe PHP, la classe FileMaker, qui fournit une interface orientée objet aux bases de données FileMaker. L'API 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 FileMaker pour PHP permet au code PHP d'effectuer le même type d'opérations que celles déjà disponibles dans les bases de données FileMaker Pro :

- créer, supprimer, modifier et dupliquer des enregistrements ;
- exécuter des requêtes ;
- procéder à la validation des rubriques et des enregistrements ;
- utiliser des modèles ;
- exécuter des scripts FileMaker ;
- afficher les tables externes et les enregistrements liés ;
- utiliser les listes de valeurs.

Ce chapitre explique comment utiliser les objets et méthodes de classe FileMaker pour ajouter ces fonctions courantes à une solution PHP. Il ne traite pas de l'API FileMaker pour PHP dans son intégralité, mais présente les objets et méthodes clés.

Où trouver des informations supplémentaires

Pour en savoir plus sur l'API FileMaker pour PHP, consultez les ressources suivantes.

Si vous disposez déjà d'un moteur PHP installé et configuré et que vous souhaitez uniquement ajouter l'API FileMaker pour PHP, reportez-vous à la section « Installation manuelle de l'API FileMaker pour PHP », page 13.

Référence API FileMaker pour PHP

Si vous avez installé l'API FileMaker pour PHP, vous pouvez trouver des informations de référence sur le composant Serveur Web de votre déploiement FileMaker Server.

- Pour IIS (Windows) :
  `<disque>:\Program Files\FileMaker\FileMaker Server\Documentation\PHP API Documentation\index.html`
  où `<disque>` correspond au disque sur lequel réside le composant Serveur Web de votre déploiement FileMaker Server.
- Pour Apache (Mac OS) : `/Bibliothèque/FileMaker Server/Documentation/PHP API Documentation/index.html`

Didacticiel API FileMaker pour PHP

Si vous avez installé l'API FileMaker pour PHP, un didacticiel est disponible sur le composant Serveur Web de votre déploiement FileMaker Server.

- Pour IIS (Windows) : `<disque>:\Program Files\FileMaker\FileMaker Server\Examples\PHP\Tutorial`
  où disque<> correspond au disque sur lequel réside le composant Serveur Web de votre déploiement FileMaker Server.
- Pour Apache (Mac OS) : `/Bibliothèque/FileMaker Server/Examples/PHP/Tutorial`

Pour héberger ces fichiers de didacticiel PHP, copiez-les dans le dossier racine du serveur Web.

Exemples d'API FileMaker pour PHP

Si vous avez installé l'API FileMaker pour PHP, vous trouverez des exemples supplémentaires sur le composant Serveur Web de votre déploiement FileMaker Server.

- Pour IIS (Windows) : `<disque>:\Program Files\FileMaker\FileMaker Server\Examples\PHP\API Examples` où `<disque>` correspond au disque sur lequel réside le composant Serveur Web de votre déploiement FileMaker Server.
- Pour Apache (Mac OS) : `/Bibliothèque/FileMaker Server/Examples/PHP/API Examples`

Pour héberger ces fichiers d'exemple d'API, copiez-les dans le dossier racine du serveur Web.

Utilisation de la classe FileMaker

Pour utiliser la classe FileMaker dans votre solution PHP, ajoutez l'instruction suivante à votre code PHP :

```
require_once ('FileMaker.php');
```

Objets de classe FileMaker

La classe FileMaker définit des objets de classe que vous pouvez utiliser pour extraire des données à partir des bases de données FileMaker Pro.

Objet de classe	Utilisez cet objet pour
Base de données FileMaker	Définir les propriétés de la base de données Vous connecter à une base de données FileMaker Pro Obtenir des informations sur l'API FileMaker pour PHP
Commande	Créer des commandes qui ajoutent, suppriment, dupliquent ou modifient des enregistrements, effectuent des requêtes et des scripts
Modèle	Manipuler des modèles de base de données
Enregistrement	Manipuler des données d'enregistrement
Rubrique	Manipuler des données de rubrique
Ensemble lié	Manipuler des enregistrements de table externe

Objet de classe	Utilisez cet objet pour
Résultat	Traiter les enregistrements renvoyés par une requête
Erreur	Vérifier si une erreur s'est produite Traiter les erreurs éventuelles

Objets de commande FileMaker

La classe FileMaker définit un objet de commande de base qui vous permet d'instancier une commande spécifique et de spécifier les paramètres de cette commande. Pour exécuter la commande, vous devez appeler la méthode `execute()`.

La classe FileMaker définit les commandes spécifiques suivantes :

- Ajouter, commande
- Recherche composée, commande
- Supprimer, commande
- Dupliquer, commande
- Edition, commande
- Commande Rechercher, commande Rechercher tout, commande Rechercher au moins
- Commande Requête, qui s'ajoute à une commande de recherche composée
- Commande Exécuter le script

Ces commandes sont décrites plus en détail dans les sections suivantes :

- « Utilisation des enregistrements », page 31
- « Exécution des scripts FileMaker », page 33
- « Exécution de requête », page 39

Connexion à une base de données FileMaker

La classe FileMaker définit un objet de base de données que vous instanciez pour vous connecter à un serveur ou à une base de données. Définissez les propriétés de l'objet avec le constructeur de classe ou en appelant la méthode `setProperty()`.

Par exemple : connexion à un serveur pour obtenir une liste des bases de données

```
$fm = new FileMaker();
$databases = $fm->listDatabases();
```

Par exemple : connexion à une base de données spécifique sur un serveur

Les propriétés du nom d'utilisateur et du mot de passe déterminent le jeu de privilèges d'accès de cette connexion.

```
$fm = new FileMaker();
$fm->setProperty('database', 'questionnaire');
$fm->setProperty('hostspec', 'http://192.168.100.110');
$fm->setProperty('username', 'web');
$fm->setProperty('username', 'web');
```

Remarque La propriété hostspec prend la valeur `http://localhost` par défaut. Si le moteur PHP est exécuté sur le même ordinateur que le composant Serveur Web du déploiement FileMaker Server, il est inutile de spécifier la propriété hostspec. Si le moteur PHP se trouve sur un autre ordinateur, utilisez la propriété hostspec pour indiquer l'emplacement du composant Serveur Web du déploiement FileMaker Server.

Utilisation des enregistrements

La classe FileMaker définit un objet d'enregistrement que vous instanciez pour manipuler les enregistrements. Une instance d'objet d'enregistrement représente un enregistrement à partir d'une base de données FileMaker Pro. Utilisez un objet d'enregistrement avec les commandes Ajouter, Supprimer, Dupliquer et Edition pour modifier les données de l'enregistrement. Les commandes Rechercher (Rechercher, Rechercher tout, Rechercher au moins et Recherche composée) renvoient un tableau d'objets d'enregistrement.

Création d'un enregistrement

Il existe deux manières de créer un enregistrement :

- Utilisez la méthode `createRecord()`, en spécifiant un nom de modèle et éventuellement un tableau de valeurs de rubriques. Vous pouvez également définir des valeurs individuellement dans le nouvel objet d'enregistrement.

  La méthode `createRecord()` ne sauvegarde pas le nouvel enregistrement dans la base de données. Pour sauvegarder l'enregistrement dans la base de données, appelez la méthode `commit()`.

  Par exemple :

  ```
  $rec = $fm->createRecord('Form View', $values);
  $result = $rec->commit();
  ```

- Utilisez la commande Ajouter. Utilisez la méthode `newAddCommand()` pour créer un objet FileMaker_Command_Add, en spécifiant le nom du modèle et un tableau de données d'enregistrement. Pour sauvegarder l'enregistrement dans la base de données, appelez la méthode `execute()`.

  Par exemple :

  ```
  $newAdd = $fm->newAddCommand('Respondent', $respondent_data);
  $result = $newAdd->execute();
  ```

Duplication d'un enregistrement

Dupliquez un enregistrement existant en utilisant la commande Dupliquer. Utilisez la méthode `newDuplicateCommand()` pour créer un objet FileMaker_Command_Duplicate, en spécifiant le nom du modèle et l'ID de l'enregistrement que vous souhaitez dupliquer. Puis, dupliquez l'enregistrement en appelant la méthode `execute()`.

Exemple

```
$newDuplicate = $fm->newDuplicateCommand('Respondent', $rec_ID);
$result = $newDuplicate->execute();
```

Edition d'un enregistrement

Il existe deux manières d'éditer un enregistrement :

- Utilisation de la commande Edition. Utilisez la méthode `newEditCommand()` pour créer un objet FileMaker_Command_Edit, en spécifiant le nom du modèle et l'ID de l'enregistrement que vous souhaitez éditer, ainsi qu'un tableau de valeurs que vous souhaitez mettre à jour. Puis, éditez l'enregistrement en appelant la méthode `execute()`.

  Par exemple :

  ```
  $newEdit = $fm->newEditCommand('Respondent', $rec_ID, $respondent_data);
  $result = $newEdit->execute();
  ```

- Utilisation d'un objet d'enregistrement. Extrayez un enregistrement à partir de la base de données, modifiez les valeurs des rubriques, puis éditez l'enregistrement en appelant la méthode `commit()`.

  Par exemple :

  ```
  $rec = $fm->getRecordById('Form View', $rec_ID);
  $rec->setField('Name', $nameEntered);
  $result = $rec->commit();
  ```

Suppression d'un enregistrement

Il existe deux manières de supprimer un enregistrement :

- Extrayez un enregistrement à partir de la base de données, puis appelez la méthode `delete()`.

  Par exemple :

  ```
  $rec = $fm->getRecordById('Form View', $rec_ID);
  $rec->delete();
  ```

- Supprimez un enregistrement existant en utilisant la commande Supprimer. Utilisez la méthode `newDeleteCommand()` pour créer un objet FileMaker_Command_Delete, en spécifiant le nom du modèle et l'ID de l'enregistrement que vous souhaitez supprimer. Puis, supprimez l'enregistrement en appelant la méthode `execute()`.

  Par exemple :

  ```
  $newDelete = $fm->newDeleteCommand('Respondent', $rec_ID);
  $result = $newDelete->execute();
  ```

Exécution des scripts FileMaker

Un script FileMaker est un ensemble nommé d'actions de script. La classe FileMaker définit plusieurs méthodes qui vous permettent d'utiliser des scripts FileMaker définis dans une base de données FileMaker Pro. Pour plus d'informations sur les actions de script compatibles Web (les actions de script qui peuvent être exécutées dans une solution Web), reportez-vous à la section « Scripts FileMaker et Publication Web personnalisée », page 21.

Obtention de la liste des scripts disponibles

Utilisez la méthode `listScripts()` pour obtenir la liste des scripts disponibles à partir de la base de données actuellement connectée. La méthode `listScripts()` renvoie un tableau de scripts pouvant être exécutés par le nom d'utilisateur et le mot de passe spécifiés lors de la définition de la connexion à la base de données. (Reportez-vous à la section « Connexion à une base de données FileMaker », page 30.)

Exemple

```
$scripts = $fm->listScripts();
```

Exécution d'un script FileMaker

Utilisez la méthode `newPerformScriptCommand()` pour créer un objet FileMaker_Command_PerformScript, spécifiant le modèle, le nom du script et les paramètres de script de votre choix. Puis, exécutez le script en appelant la méthode `execute()`.

Exemple

```
$newPerformScript = $fm->newPerformScriptCommand('Order Summary',
'ComputeTotal');
$result = $newPerformScript->execute();
```

Exécution d'un script avant l'exécution d'une commande

Utilisez la méthode `setPreCommandScript()` pour spécifier l'exécution d'un script avant l'exécution d'une commande. L'exemple suivant utilise une commande Rechercher, mais vous pouvez utiliser la méthode `setPreCommandScript()` avec n'importe quelle commande.

Exemple

```
$findCommand = $fm->newFindCommand('Students');
$findCommand->addFindCriterion('GPA', $searchValue);
$findCommand->setPreCommandScript('UpdateGPA');
$result = $findCommand->execute();
```

Exécution d'un script avant le tri d'un ensemble de résultats

Utilisez la méthode `setPreSortScript()` pour spécifier l'exécution d'un script après la génération d'un ensemble de résultats à l'aide de la commande Rechercher, mais avant le tri de cet ensemble de résultats. Pour plus d'informations, reportez-vous à la rubrique « Utilisation de la commande Rechercher », page 40.

Exemple

```
$findCommand = $fm->newFindCommand('Students');
$findCommand->setPreSortScript('RemoveExpelled');
```

Exécution d'un script après la génération d'un ensemble de résultats

Utilisez la commande `setScript()` pour spécifier l'exécution d'un script après la génération d'un ensemble de résultats à l'aide de la commande Rechercher. Pour plus d'informations, reportez-vous à la rubrique « Utilisation de la commande Rechercher », page 40.

Exemple

```
$findCommand = $fm->newFindCommand('Students');
$findCommand->setScript('myScript','param1|param2|param3');
```

Ordre d'exécution du script

Vous pouvez spécifier les méthodes `setPreCommandScript()`, `setPreSortScript()` et `setScript()` en conjonction avec les méthodes `setResultLayout()` et `addSortRule()` pour une seule commande. Voici l'ordre dans lequel FileMaker Server et le moteur de publication Web traitent ces méthodes :

1. Exécution du script spécifié dans la méthode `setPreCommandScript()`, le cas échéant.
2. Traitement de la commande elle-même, par exemple la commande Rechercher ou Supprimer l'enregistrement.
3. Exécution du script spécifié dans la méthode `setPreSortScript()`, le cas échéant.
4. Tri de l'ensemble de résultats de la commande Rechercher, si la méthode `addSortRule()` a été spécifiée.
5. Traitement de la méthode `setResultLayout()` pour passer d'un modèle à un autre, si indiqué.
6. Exécution du script spécifié dans la méthode `setScript()`, le cas échéant.
7. Renvoi de l'ensemble de résultats final de la commande Rechercher.

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

Par exemple, imaginons une commande qui supprime l'enregistrement ouvert, trie les enregistrements, puis exécute un script. Si la méthode `addSortRule()` 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 des modèles FileMaker

Un modèle est un agencement de rubriques, d'objets, d'images et d'autres éléments qui déterminent l'organisation et la présentation des informations lors de la lecture, de la prévisualisation ou de l'impression d'enregistrements par l'utilisateur. La classe FileMaker définit plusieurs méthodes qui vous permettent d'utiliser des modèles définis dans une base de données FileMaker Pro. Vous pouvez obtenir des informations sur les modèles à partir de plusieurs objets de classe FileMaker.

Avec cet objet de classe	Utilisez ces méthodes
Base de données	▪ `listLayouts()` obtient la liste des noms de modèles disponibles. ▪ `getLayout()` obtient un objet de modèle en pointant vers un nom de modèle.
Modèle	▪ `getName()` extrait le nom de modèle d'un objet de modèle spécifique. ▪ `listFields()` extrait un tableau de tous les noms de rubriques utilisés dans un modèle. ▪ `getFields()` extrait un tableau associatif contenant les noms de toutes les rubriques sous la forme de clés et les objets FileMaker_Field liés sous la forme d'un tableau de valeurs. ▪ `listValueLists()` extrait un tableau des noms de listes de valeurs. ▪ `listRelatedSets()` extrait un tableau des noms des ensembles liés. ▪ `getDatabase()` renvoie le nom de la base de données.
Enregistrement	▪ `getLayout()` renvoie l'objet de modèle lié à un enregistrement spécifique.
Rubrique	▪ `getLayout()` renvoie l'objet de modèle contenant une rubrique spécifique.
Commande	▪ `setResultLayout()` renvoie les résultats de la commande dans un modèle différent du modèle en cours.

Utilisation des tables externes

Une table externe est une table qui affiche les lignes de données d'un ou plusieurs enregistrements correspondants. La classe FileMaker définit un objet d'ensemble lié et plusieurs méthodes qui vous permettent d'utiliser des tables externes définies dans une base de données FileMaker Pro.

Un objet d'ensemble lié est un tableau d'objets d'enregistrement issus de la table externe correspondante ; chaque objet d'enregistrement correspond à une ligne de données dans la table externe.

Création d'une liste des tables externes définies sur un modèle spécifique

Pour un objet de modèle spécifique, utilisez la méthode `listRelatedSets()` pour extraire une liste des noms de tables de toutes les tables externes définies dans ce modèle.

Exemple

```
$tableNames = $currentLayout->listRelatedSets();
```

Obtention des noms de tables externes d'un objet de résultat spécifique

Pour un objet FileMaker_Result spécifique, utilisez la méthode `getRelatedSets()` pour extraire les noms de toutes les tables externes de cet enregistrement.

Exemple

```
$relatedSetsNames = $result->getRelatedSets();
```

Obtention d'informations sur les tables externes d'un modèle spécifique

Pour un objet de modèle spécifique, utilisez la méthode `getRelatedSets()` pour extraire un tableau d'objets FileMaker_RelatedSet décrivant les tables externes du modèle. Le tableau renvoyé est un tableau associatif contenant les noms des tables sous la forme d'un tableau de clés, et les objets FileMaker_RelatedSet liés sous la forme d'un tableau de valeurs.

Exemple

```
$relatedSetsArray = $currentLayout->getRelatedSets();
```

Obtention d'informations sur une table externe spécifique

Pour un objet de modèle spécifique, utilisez la méthode `getRelatedSet()` pour extraire l'objet FileMaker_RelatedSet décrivant une table externe spécifique.

Exemple

```
$relatedSet = $currentLayout->getRelatedSet('customers');
```

Obtention du nom de table d'une table externe

Pour un objet d'ensemble lié, utilisez la méthode `getName()` pour obtenir le nom de table de la table externe.

Exemple

```
$tableName = $relatedSet->getName();
```

Obtention des enregistrements externes d'un enregistrement spécifique

Pour un objet d'enregistrement spécifique, utilisez la méthode `getRelatedSet()` pour extraire un tableau des enregistrements externes liés à cet enregistrement.

Exemple

```
$relatedRecordsArray = $currentRecord->getRelatedSet('customers');
```

Création d'un nouvel enregistrement dans une table externe

Utilisez la méthode `newRelatedRecord()` pour créer un nouvel enregistrement dans l'ensemble lié spécifié, et validez cette modification dans la base de données en appelant la méthode `commit()`.

Exemple

```
//créer une nouvelle ligne de table externe dans la table externe 'customer'
$new_row = $currentRecord->newRelatedRecord('customer');

//définir les valeurs des rubriques dans la nouvelle ligne de table externe
$new_row->setField('customer::name', $newName);
$new_row->setField('customer::company', $newCompany);

$result = $new_row->commit();
```

Suppression d'un enregistrement d'une table externe

Utilisez la méthode `delete()` pour supprimer un enregistrement d'une table externe.

Exemple

```
$relatedSet = $currentRecord->getRelatedSet('customers');
/* Exécuté sur chacune des lignes de la table externe */
foreach ($relatedSet as $nextRow) {

    $nameField = $nextRow->getField('customer::name')
    if ($nameField == $badName ) {
        $result =   $newRow->delete();
    }
}
```

Utilisation des listes de valeurs

Une liste de valeurs est un ensemble de choix prédéfinis. La classe FileMaker définit plusieurs méthodes qui vous permettent d'utiliser des listes de valeurs définies dans une base de données FileMaker Pro.

Obtention des noms de toutes les listes de valeurs d'un modèle spécifique

Pour un objet de modèle spécifique, utilisez la méthode `listValueLists()` pour extraire un tableau contenant les noms des listes de valeurs.

Exemple

```
$valueListNames = $currentLayout->listValueLists();
```

Obtention d'un tableau de toutes les listes de valeurs d'un modèle spécifique

Pour un objet de modèle spécifique, utilisez la méthode `getValueListsTwoFields()` pour extraire un tableau contenant les valeurs de toutes les listes de valeurs. Le tableau renvoyé est un tableau associatif. Les clés du tableau correspondent aux noms des listes de valeurs et les valeurs du tableau sont des tableaux associatifs qui dressent la liste des noms affichés et des choix correspondants pour chaque liste de valeurs.

Exemple

```
$valueListsArray = $currentLayout->getValueListsTwoFields();
```

Remarque Bien que la méthode `getValueLists()` soit encore prise en charge dans l'API FileMaker pour PHP, elle n'est plus utilisée. Utilisez plutôt la méthode `getValueListsTwoFields()`.

Obtention des valeurs pour une liste de valeurs nommée

Pour un objet de modèle spécifique, utilisez la méthode `getValueListTwoFields()` pour extraire un tableau des choix définis dans la liste de valeurs nommée. Le tableau renvoyé est un tableau associatif. Ses clés correspondent aux valeurs affichées dans la seconde rubrique de la liste de valeurs et ses valeurs, aux valeurs stockées associées de la première rubrique.

En fonction des options sélectionnées dans la boîte de dialogue **Rubriques pour la liste de valeurs** de la base de données FileMaker, la méthode `getValueListTwoFields()` renvoie la valeur dans la première rubrique uniquement, la valeur dans la seconde rubrique uniquement ou les valeurs dans les deux rubriques d'une liste de valeurs, l'une étant la valeur stockée et l'autre, la valeur affichée.

- Si l'option **Afficher également les valeurs de la seconde rubrique** n'est pas sélectionnée, la méthode `getValueListTwoFields()` renvoie la valeur de la première rubrique de la liste de valeurs, correspondant à la fois à la valeur stockée et à la valeur affichée.
- 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, la méthode `getValueListTwoFields()` renvoie la valeur de la première rubrique sous forme de valeur stockée et la valeur de la seconde rubrique sous forme de valeur affichée.
- 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, la méthode `getValueListTwoFields()` renvoie la valeur de la première rubrique sous forme de valeur stockée et les deux valeurs provenant de la première et de la seconde rubriques, sous forme de valeur affichée.

Utilisez un itérateur avec la méthode `getValueListTwoFields()` pour rechercher la valeur affichée et la valeur stockée.

Exemple

```
$layout = $fm->getLayout('customers');
$valuearray = $layout->getValueListTwoFields("region", 4);
foreach ($valuearray as $displayValue => $value) {
    ....
}
```

Remarques

- Bien que la méthode `getValueList()` soit encore prise en charge dans l'API FileMaker pour PHP, elle n'est plus utilisée. Utilisez plutôt la méthode `getValueListTwoFields()`.
- Lorsque vous utilisez la méthode `getValueListTwoFields()`, veillez à inclure une boucle `foreach` pour faire une boucle dans le tableau associatif. Par contre, n'utilisez pas de boucle `for` qui pourrait générer des résultats inattendus.

Exécution de requête

La classe FileMaker définit quatre types d'objets de commande Rechercher :

- Commande Rechercher tout. Consultez la section « Utilisation de la commande Rechercher tout », page 40.
- Commande Rechercher au moins. Consultez la section « Utilisation de la commande Rechercher au moins », page 40.
- Commande Rechercher. Consultez la section « Utilisation de la commande Rechercher », page 40.
- Commande Recherche composée. Consultez la section « Utilisation d'une commande Recherche composée », page 41.

La classe FileMaker définit également plusieurs méthodes pouvant être utilisées pour les quatre types de commandes Rechercher :

- Utilisez la méthode `addSortRule()` pour ajouter une règle définissant la manière dont l'ensemble de résultats doit être trié. Utilisez la méthode `clearSortRules()` pour effacer toutes les règles de tri qui ont été définies.
- Utilisez la méthode `setLogicalOperator()` pour basculer entre les recherches AND logiques et les recherches OR logiques.
- Utilisez la méthode `setRange()` pour effectuer une requête visant à obtenir uniquement une partie de l'ensemble de résultats. Utilisez la méthode `getRange()` pour extraire la définition de la plage actuelle.

  L'utilisation de la méthode `setRange()` peut améliorer la performance de votre solution en réduisant le nombre d'enregistrement renvoyés par la requête Rechercher. Par exemple, si une requête Rechercher renvoie 100 enregistrements, vous pouvez répartir l'ensemble de résultats en cinq groupes de 20 enregistrements chacun au lieu de traiter les 100 enregistrements en une seule fois.
- Vous pouvez exécuter les scripts FileMaker en conjonction avec les commandes Rechercher.
  - Pour exécuter un script avant d'exécuter la commande Rechercher, utilisez la méthode `setPreCommandScript()`.
  - Pour exécuter un script avant le tri de l'ensemble de résultats, utilisez la méthode `setPreSortScript()`.
  - Pour exécuter un script après la génération d'un ensemble de résultats, mais avant le tri de cet ensemble de résultats, utilisez la méthode `setScript()`.

Utilisation de la commande Rechercher tout

Utilisez la commande Rechercher tout pour extraire tous les enregistrements d'un modèle spécifié. Utilisez la méthode `newFindAllCommand()`, en spécifiant un modèle spécifique, pour créer un objet FileMaker_Command_FindAll. Puis, exécutez la requête en appelant la méthode `execute()`.

Exemple

```
$findCommand = $fm->newFindAllCommand('Form View');
$result = $findCommand->execute;
```

Remarque Pour éviter les problèmes de surcharge de la mémoire de l'ordinateur si vous utilisez la commande Rechercher tout, indiquez par défaut un nombre maximum d'enregistrements à renvoyer par page.

Utilisation de la commande Rechercher au moins

Utilisez la commande Rechercher au moins pour extraire un enregistrement aléatoire d'un modèle spécifié. Utilisez la méthode `newFindAnyCommand()`, en spécifiant un modèle spécifique, pour créer un objet FileMaker_Command_FindAny. Puis, exécutez la requête en appelant la méthode `execute()`.

Exemple

```
$findCommand = $fm->newFindAnyCommand('Form View');
$result = $findCommand->execute;
```

Utilisation de la commande Rechercher

Utilisez la méthode `newFindCommand()`, en spécifiant un modèle spécifique, pour créer un objet FileMaker_Command_Find. Puis, exécutez la requête en appelant la méthode `execute()`.

Utilisez la méthode `addFindCriterion()` pour ajouter des critères à la requête. Utilisez la méthode `clearFindCriteria()` pour effacer tous les critères de recherche qui ont été définis.

Exemple : recherche d'un enregistrement par nom de rubrique

```
$findCommand = $fm->newFindCommand('Form View');
$findCommand->addFindCriterion('Questionnaire ID', $active_questionnaire_id);
$result = $findCommand->execute();
```

Exemple : ajout d'un ordre de tri

```
$findCommand = $fm->newFindCommand('Customer List');
$findCommand->addSortRule('Title', 1, FILEMAKER_SORT_ASCEND);
$result = $findCommand->execute();
```

Utilisation d'une commande Recherche composée

La commande Recherche composée vous permet de combiner plusieurs objets de requête en une seule commande.

Pour créer une commande Recherche composée :

- Créez un objet FileMaker_Command_CompoundFind en appelant la méthode `newCompoundFindCommand()`.
- Créez un ou plusieurs objets FileMaker_Command_FindRequest en appelant la méthode `newFindRequest()`.
- Utilisez la méthode `setOmit()` pour indiquer quels enregistrements de l'ensemble de résultats d'une requête spécifique devront être ignorés dans l'ensemble de résultats final.
- Utilisez la méthode `add()` pour ajouter les objets Requête à l'objet de commande Recherche composée.
- Exécutez la commande Recherche composée en appelant la méthode `execute()`.

Exemple : commande Recherche composée

```
// Créer l'objet de commande Recherche composée
$compoundFind = $fm->newCompoundFindCommand('Form View');

// Créer la première requête
$findreq1 = $fm->newFindRequest('Form View');

// Créer la seconde requête
$findreq2 = $fm->newFindRequest('Form View');

// Créer la troisième requête
$findreq3 = $fm->newFindRequest('Form View');

// Spécifier le critère de recherche de la première requête
$findreq1->addFindCriterion('Quantity in Stock', '<100');

// Spécifier le critère de recherche de la seconde requête
$findreq2->addFindCriterion('Quantity in Stock', '0');
$findreq2->setOmit(true);

// Spécifier le critère de recherche de la troisième requête
$findreq3->addFindCriterion('Cover Photo Credit', 'The London Morning News');
$findreq3->setOmit(true);

// Ajouter des requêtes à une commande Recherche composée
$compoundFind->add(1,$findreq1);
$compoundFind->add(2,$findreq2);
$compoundFind->add(3,$findreq3);

// Définir un ordre de tri
$compoundFind->addSortRule('Title', 1, FILEMAKER_SORT_DESCEND);

// Exécuter une commande Recherche composée
$result = $compoundFind->execute();

// Obtenir des enregistrements à partir de l'ensemble trouvé
$records = $result->getRecords();

// Indiquer le nombre d'enregistrements trouvés
echo 'Found '. count($records). " results.<br><br>";
```

Traitement des enregistrements d'un ensemble de résultats

- Extrayez un tableau contenant chaque enregistrement de l'ensemble de résultats en appelant la méthode `getRecords()`. Chaque membre du tableau est un objet FileMaker_Record, ou une instance du nom de classe défini dans l'API pour instancier les enregistrements. Le tableau peut être vide si l'ensemble de résultats ne contient aucun enregistrement.
- Obtenez la liste des noms de toutes les rubriques de l'ensemble de résultats en appelant la méthode `getFields()`. Cette méthode renvoie uniquement les noms des rubriques. Si vous avez besoin d'informations supplémentaires concernant les rubriques, utilisez l'objet de modèle lié.
- Obtenez le nombre d'enregistrements du jeu complet d'enregistrements trouvés en appelant la méthode `getFoundSetCount()`.
- Obtenez le nombre d'enregistrements du jeu d'enregistrements trouvés filtré en appelant la méthode `getFetchCount()`. Si aucun paramètre de plage n'a été spécifié pour la commande Rechercher, cette valeur est alors égale au résultat de la méthode `getFoundSetCount()`. Elle est toujours égale à la valeur de `count($response->getRecords())`.
- Pour un enregistrement spécifique, utilisez la méthode `getField()` pour renvoyer le contenu d'une rubrique sous la forme d'une chaîne.
- Pour un enregistrement spécifique, utilisez la méthode `getFieldAsTimestamp()` pour renvoyer le contenu d'une rubrique sous la forme d'un horodatage Unix (la représentation interne d'une date en PHP).
  - Si la rubrique est une rubrique Date, l'horodatage concerne la date de la rubrique à minuit.
  - Si la rubrique est une rubrique Heure, l'horodatage concerne l'heure au 1er janvier 1970.
  - Si la rubrique est une rubrique Horodatage, la valeur d'horodatage de FileMaker est directement mise en correspondance avec l'horodatage Unix.
  - Si la rubrique spécifiée n'est ni une rubrique Date ni une rubrique Heure, ou si l'horodatage généré se situe en dehors de la plage, la méthode `getFieldAsTimestamp()` renvoie un objet FileMaker_Error.
- Pour un enregistrement spécifique, utilisez la méthode `getContainerData()` pour renvoyer un objet de rubrique Conteneur sous la forme de données binaires :

```
<IMG src="img.php?-url=<?php echo urlencode($record->getField('Cover
Image')); ?>">
echo $fm->getContainerData($_GET['-url']);
```

- Pour un enregistrement spécifique, utilisez la méthode `getContainerDataURL()` pour renvoyer une URL entièrement qualifiée pour l'objet de la rubrique Conteneur :

```
// Pour les images, utilisez la balise HTML img
        echo '<img src="'.$fm->
getContainerDataURL($record->getField('container')) .'">';
// Pour les vidéos et les fichiers PDF, utilisez la balise HTML embed
        //echo '<embed src="'.$fm->
getContainerDataURL($record->getField('container')) .'">';
```

Filtrage des lignes des tables externes renvoyées par les requêtes

Dans une solution comportant de nombreux enregistrements liés, l'interrogation et le tri des enregistrements externes peuvent demander beaucoup de temps. Pour restreindre le nombre d'enregistrements affichés dans un ensemble lié, utilisez la méthode `setRelatedSetsFilters()` avec les requêtes. La méthode `setRelatedSetsFilters()` prend deux arguments :

- une valeur de filtre des ensembles liés : `layout` ou `none`.
  - Si vous spécifiez la valeur `none`, le moteur de publication Web renvoie toutes les lignes de la table externe, ainsi que les enregistrements externes qui ne sont pas prétriés.
  - Si vous spécifiez la valeur `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.
- le nombre maximal d'enregistrements externes renvoyés : un nombre entier ou `all`.
  - Cette valeur est utilisée uniquement lorsque le paramètre Barre de défilement vertical est activé dans la boîte de dialogue Table externe. Lorsque vous spécifiez un nombre entier, ce nombre de lignes est renvoyé après la ligne initiale. Lorsque vous spécifiez `all`, le moteur de publication Web renvoie tous les enregistrements liés.
  - Lorsque le paramètre Barre de défilement vertical est désactivé, le nombre de lignes de la boîte de dialogue Table externe détermine le nombre maximal d'enregistrements liés qui sont renvoyés.

Prévalidation des commandes, des enregistrements et des rubriques

La classe FileMaker vous permet de *prévalider* les données d'une rubrique dans une solution PHP sur le serveur Web avant de les valider dans la base de données.

Lorsque vous envisagez d'utiliser la prévalidation, tenez compte du nombre de valeurs saisies par l'utilisateur Web. Si l'utilisateur met à jour un petit nombre de rubriques, vous pouvez améliorer la performance en n'utilisant pas la prévalidation. Mais si l'utilisateur saisit des données dans de nombreuses rubriques, la prévalidation peut lui éviter la frustration liée au rejet d'un enregistrement par la base de données en raison d'erreurs de validation.

Avec la classe FileMaker, le moteur PHP prévalide les contraintes de rubrique suivantes :

- non vide

  Les données valides sont les chaînes de caractères non vides. Les données doivent contenir au moins un caractère.
- numérique uniquement

  Les données valides contiennent des caractères numériques uniquement.
- nombre maximal de caractères

  Les données valides contiennent au plus le nombre maximal de caractères spécifié.

- année à quatre chiffres

  Les données valides sont les chaînes de caractères représentant une date avec une année à quatre chiffres au format J/M/AAAA, où A est un nombre situé entre 1 et 12 inclus, J est un nombre situé entre 1 et 31 inclus et AAAA est un nombre à quatre chiffres situé entre 0001 et 4000 inclus. Par exemple, `30/1/3030` est une valeur d'année à quatre chiffres valide. Alors que `31/4/2012` est une valeur d'année à quatre chiffres non valide car le mois d'avril ne compte pas 31 jours. La validation de la date tient compte de la barre oblique (/), la barre oblique inversée (\) et du tiret (-) comme délimiteurs. Toutefois, la chaîne ne peut pas contenir différents délimiteurs. Par exemple, `30\1-2012` n'est pas valide.
- heure du jour

  Les données valides sont les chaînes de caractères représentant une valeur horaire sur 12 heures dans l'un des formats suivants :

  - H
  - H:M
  - H:M:S
  - H:M:S AM/PM
  - H:M AM/PM

  où H est un nombre situé entre 1 et 12 inclus ; M et S sont des nombres situés entre 1 et 60 inclus.

La prévalidation du moteur PHP prend en charge la vérification implicite des données de la rubrique en fonction du type de rubrique :

- date

  Une rubrique définie comme une rubrique Date est validée en fonction des règles de validation de l'« année à quatre chiffres », exception faite que la valeur année peut contenir de 0 à 4 chiffres (la valeur année peut être vide). Par exemple, `30/1` est une date valide même si aucune année n'est spécifiée.
- heure

  Une rubrique définie comme une rubrique Heure est validée en fonction des règles de validation de l'« heure du jour », exception faite que le composant heure (H) peut être un nombre situé entre 1 et 24 inclus pour prendre en charge les valeurs horaires sur 24 heures.
- horodatage

  Une rubrique définie comme une rubrique Horodatage est validée en fonction des règles de validation de l'« heure » du composant heure et en fonction des règles de validation de la « date » du composant date.

La classe FileMaker ne peut pas prévalider toutes les options de validation de rubrique disponibles dans FileMaker Pro. Les options de validation suivantes ne peuvent pas être prévalidées car elles dépendent de l'état de toutes les données de la base de données au moment de la validation de ces données :

- valeur unique
- valeur existante
- entre

- élément de la liste de valeurs
- contrôle par le calcul

Prévalidation des enregistrements dans une commande

Pour un objet de commande, utilisez la méthode `validate()` pour valider une rubrique ou la commande entière en fonction des règles de prévalidation mises en œuvre par le moteur PHP. Si vous transmettez l'argument facultatif du nom de la rubrique, seule cette rubrique sera prévalidée.

Lorsque la prévalidation est réussie, la méthode `validate()` renvoie TRUE. Lorsque la prévalidation échoue, la méthode `validate()` renvoie un objet FileMaker_Error_Validation contenant des détails sur ce qui n'a pas pu être validé.

Prévalidation des enregistrements

Pour un objet d'enregistrement, utilisez la méthode `validate()` pour valider une rubrique ou toutes les rubriques de l'enregistrement en fonction des règles de prévalidation mises en œuvre par le moteur PHP. Si vous transmettez l'argument facultatif du nom de la rubrique, seule cette rubrique sera prévalidée.

Lorsque la prévalidation est réussie, la méthode `validate()` renvoie TRUE. Lorsque la prévalidation échoue, la méthode `validate()` renvoie un objet FileMaker_Error_Validation contenant des détails sur ce qui n'a pas pu être validé.

Prévalidation des rubriques

Pour un objet rubrique, utilisez la méthode `validate()` pour déterminer si la valeur donnée d'une rubrique est valide.

Lorsque la prévalidation est réussie, la méthode `validate()` renvoie TRUE. Lorsque la prévalidation échoue, la méthode `validate()` renvoie un objet FileMaker_Error_Validation contenant des détails sur ce qui n'a pas pu être validé.

Traitement des erreurs de validation

Lorsque la prévalidation échoue, l'objet FileMaker_Error_Validation renvoyé contient un tableau comportant trois éléments pour chaque échec de validation :

1. L'objet de rubrique pour lequel la prévalidation a échoué

2. Une constante de validation indiquant la règle de validation qui a échoué :

1 - FILEMAKER_RULE_NOTEMPTY

2 - FILEMAKER_RULE_NUMERICONLY

3 - FILEMAKER_RULE_MAXCHARACTERS

4 - FILEMAKER_RULE_FOURDIGITYEAR

5 - FILEMAKER_RULE_TIMEOFDAY

6 - FILEMAKER_RULE_TIMESTAMP_FIELD

7 - FILEMAKER_RULE_DATE_FIELD

8 - FILEMAKER_RULE_TIME_FIELD

3. La valeur réelle entrée dans la rubrique pour laquelle la prévalidation a échoué

Vous pouvez également utiliser les méthodes suivantes avec un objet FileMaker_Error_Validation :

- Utilisez la méthode `isValidationError()` pour savoir si une erreur est une erreur de validation.
- Utilisez la méthode `numErrors()` pour obtenir le nombre de règles de validation ayant échoué.

Exemple

```
//Créer une requête Ajouter
$addrequest = $fm->newAddCommand('test', array('join' => 'added', 'maxchars' =>
'abcx', 'field' => 'something' , 'numericonly' => 'abc'));

//Valider toutes les rubriques
$result = $addrequest->validate();

//Si la méthode validate() a renvoyé des erreurs, afficher le nom de la rubrique,
le numéro d'erreur et la valeur ayant échoué.
if(FileMaker::isError($result)){
    echo 'Validation failed:'. "\n";
    $validationErrors= $result->getErrors();
    foreach ($validationErrors as $error) {
        $field =  $error[0];
        echo 'Field Name: ' .  $field->getName(). "\n";
        echo 'Error Code: ' .  $error[1] . "\n";
        echo 'Value: ' .  $error[2] . "\n";
    }
}
```

Résultat

```
Echec de la validation :
Nom : numericonly
Code d'erreur : 2
Valeur : abc
Nom : maxchars
Code d'erreur : 3
Valeur : abcx
```

Gestion des erreurs

La classe FileMaker définit l'objet FileMaker_Error pour vous permettre de gérer les erreurs qui se produisent dans une solution PHP.

Une erreur peut se produire lorsqu'une commande est exécutée. Si une erreur se produit, la commande renvoie un objet FileMaker_Error. Il est conseillé de vérifier l'erreur renvoyée lorsqu'une commande est exécutée.

Utilisez les méthodes suivantes pour en savoir plus sur l'erreur indiquée dans l'objet FileMaker_Error.

- Vérifiez si une variable est un objet FileMaker Error en appelant la méthode `isError()`.
- Obtenez le nombre d'erreurs qui se sont produites en appelant la méthode `numErrors()`.
- Extrayez un ensemble de tableaux décrivant les erreurs qui se sont produites en appelant la méthode `getErrors()`.
- Affichez un message d'erreur en appelant la méthode `getMessage()`.

Exemple

```
$result = $findCommand->execute();
if(FileMaker::isError($result)){
        echo "<p>Error: " . $result->getMessage() . "</p>";
        exit;
}
```

Pour plus d'informations sur les codes d'erreur renvoyés avec l'objet FileMaker Error, reportez-vous à la section annexe A, « Codes d'erreur de la Publication Web personnalisée avec PHP ».

Chapitre 6
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, copiez ou déplacez les fichiers nécessaires vers 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 du chapitre 3, « Préparation des bases de données pour la Publication Web personnalisée ».
2. Assurez-vous que la publication Web personnalisée avec PHP a été activée et correctement paramétrée dans l'Admin Console de FileMaker Server.

   Remarque Vous trouverez des instructions 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 fichiers de votre site vers le composant serveur Web de votre déploiement FileMaker Server.

   Copiez ou déplacez les fichiers de votre site vers le répertoire suivant sur la machine du serveur Web :

   - IIS (Windows) : `<disque>:\Inetpub\wwwroot` où `<disque>` est le disque sur lequel réside le composant Moteur de publication Web de votre déploiement de serveur FileMaker.
   - Apache (Mac OS) : `/Bibliothèque/WebServer/Documents`

5. Si ce n'est pas déjà fait, copiez ou déplacez tout objet Conteneur référencé vers l'emplacement correct sur la machine du serveur Web.
   - Si le fichier de base de données est correctement hébergé et accessible sur le composant Database Server du déploiement de FileMaker Server et si les rubriques Conteneur stockent les fichiers proprement dits dans la base de données FileMaker, toute action sur le contenu de la rubrique Conteneur est alors inutile.
   - Si une rubrique de base de données de conteneur stocke une référence à un fichier et non le fichier lui-même, l'objet Conteneur 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 Conteneur référencés dans un dossier possédant le même emplacement relatif dans le dossier racine du logiciel de serveur Web.
   - Si une rubrique Conteneur de base de données stocke l'objet Conteneur en externe, utilisez l'assistant Téléchargement de base de données pour transférer le fichier de base de données et les objets de la rubrique Conteneur depuis le système de fichiers de votre ordinateur vers FileMaker Server. Si vous téléchargez manuellement une base de données utilisant un conteneur avec des objets stockés en externe, vous devez copier ou déplacer les objets référencés dans un sous-dossier du dossier RC_Data_FMS, comme décrit dans la section « Rubriques Conteneur avec données stockées en externe », page 19.

   Consultez la section « Publication du contenu des rubriques Conteneur sur le Web », page 18.

6. Commencez à tester votre site.

Test d'un site de Publication Web personnalisée

Avant d'informer les utilisateurs de la disponibilité de votre site de publication Web personnalisée, vérifiez qu'il s'affiche 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.
- 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 21 pour obtenir des informations sur la conception de scripts adaptés au Web.

- Testez votre site avec différents systèmes d'exploitation et navigateurs Web.
- Lors de la création de solutions utilisant l'API FileMaker pour PHP, nous recommandons la conception de solutions prenant en charge les cookies. L'API FileMaker pour PHP offre des délais de réponse plus courts lorsque les cookies sont activés. Les cookies ne sont pas nécessaires à l'utilisation des fonctions Publication Web personnalisée, mais permettent la mise en cache des informations de session par le moteur de publication Web.

Remarque Si vous avez installé le serveur Web, le moteur de publication Web et le serveur de bases de données dans un déploiement sur une seule machine, vous pouvez afficher et tester votre site sans utiliser la connexion réseau. Déplacez les fichiers de votre site vers le répertoire approprié sur cette machine, puis saisissez l'URL suivante dans votre navigateur :

```
http://127.0.0.1/<chemin_site>
```

Où `<chemin_site>` est le chemin relatif vers la page d'accueil de votre site.

Suivi de votre site

Utilisez 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 du moteur de publication Web
- Journal d'erreurs du module de serveur Web
- Journaux Tomcat

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

- 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.
- Apache (Mac OS uniquement) : 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.

Remarque 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 http://www.w3.org.

Utilisation du journal du moteur de publication Web

Par défaut, le moteur de publication Web génère un fichier journal nommé wpe.log qui contient un enregistrement des erreurs de moteur de publication Web qui se sont produites, notamment les erreurs d'application, d'utilisation et système. Vous pouvez également inclure dans le moteur de publication Web des informations relatives à la publication Web personnalisée, par exemple, les requêtes XML de l'utilisateur final afin de générer des résultats de publication ou des modifications des paramètres de Publication Web personnalisée.

Etant donné que l'API FileMaker pour PHP utilise HTTP POST pour accéder au moteur de publication Web, le fichier wpe.log n'enregistre pas les détails sur les requêtes PHP. Vous pouvez utiliser le fichier wpe.log pour consulter le moment où les requêtes PHP sont émises par les utilisateurs en recherchant les requêtes XML enregistrées.

Le fichier wpe.log se trouve sur le composant Moteur de publication Web du déploiement FileMaker Server :

- IIS (Windows) :
  `<disque>:\Program Files\FileMaker\FileMaker Server\Logs\wpe.log`
  où `<disque>` correspond au disque principal servant au démarrage de votre système.
- Apache (Mac OS) : `/Bibliothèque/FileMaker Server/Logs/wpe.log`

Paramètres du journal du moteur de publication Web

Le fichier wpe.log est généré si l'option **Activer la journalisation pour la publication Web personnalisée** est activée dans l'Admin Console.

Option de journalisation activée	Information enregistrée dans wpe.log
Messages de niveau d'erreur	Toutes les erreurs de moteur de publication Web qui se sont produites, notamment les erreurs d'application, les erreurs d'utilisation et les erreurs système.
Messages et informations sur le niveau d'erreur	Toutes les erreurs décrites ci-dessus et les informations sur l'accès au moteur de publication Web. Il contient un enregistrement de toutes les requêtes XML de l'utilisateur final pour générer la sortie de publication Web personnalisée.

La configuration des **messages de niveau d'erreur** est activée par défaut. Pour plus d'informations sur le paramétrage de ces options via l'Admin Console, consultez l'aide de FileMaker Server.

Remarque Pour la publication Web personnalisée avec FileMaker Server 12, le fichier wpe.log remplace les fichiers wpc_access_log.txt et pe_application_log.txt utilisés dans les versions précédentes.

Important Au fil du temps, le fichier wpe.log peut devenir très important. Utilisez l'Admin Console pour définir la taille maximale du fichier wpe.log. Lorsque le fichier wpe.log atteint la taille maximale, le moteur de publication Web copie le fichier wpe.log dans un seul fichier de sauvegarde, wpe.log.1 et crée un nouveau fichier wpe.log. Si vous souhaitez conserver plusieurs copies de sauvegarde, vous pouvez enregistrer régulièrement une archive du fichier wpe.log.1.

Format du journal du moteur de publication Web

Le fichier wpe.log utilise le format suivant pour chaque entrée :

```
[TIMESTAMP_GMT] [WPC_HOSTNAME] [CLIENT_IP:PORT] [ACCOUNT_NAME] [MODULE_TYPE]
[SEVERITY] [FM_ERRORCODE] [RETURN_BYTES] [MESSAGE]
```

où :

- [TIMESTAMP_GMT] correspond aux date et heure de la saisie, à l'heure de Greenwich (GMT).
- [WPC_HOSTNAME] est le nom de la machine sur laquelle le moteur de publication Web est installé.
- [CLIENT_IP:PORT] est l'adresse IP et le port client d'origine de la requête XML.
- [ACCOUNT_NAME] est le nom de compte utilisé pour se connecter à la base de données FileMaker hébergée.
- [MODULE_TYPE] est soit : XML, pour la publication Web personnalisée, ou PHP, pour la publication Web personnalisée avec requête PHP.
- [SEVERITY] est soit INFO, désignant un message d'information ou ERREUR, désignant un message d'erreur.
- [FM_ERROR_CODE] est le numéro d'erreur renvoyé pour un message d'erreur. Le numéro d'erreur peut être un code erreur des bases de données FileMaker (voir « Numéros des codes d'erreur des bases de données FileMaker », page 55).

  En outre, le numéro d'erreur peut être un numéro HTTP, précédé de la chaîne « HTTP: ».
- [RETURN_BYTES] est le nombre d'octets renvoyés par la requête.
- [MESSAGE] fournit des informations complémentaires sur l'entrée du journal.

Exemples de message de journal de moteur de publication Web

Les exemples suivants affichent les types de messages pouvant être inclus dans le fichier wpe.log :

- Lorsque le moteur de publication Web démarre et s'arrête

```
2012-06-02 15:15:31 -0700 - - - - INFO - - Le moteur de publication Web de FileMaker Server a commencé.
2012-06-02 15:46:52 -0700 - - - - INFO - - Le moteur de publication Web de FileMaker Server s'est arrêté.
```

- Requêtes XML réussies ou ayant échoué

```
2012-06-02 15:21:08 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML INFO 0 3964 "/fmi/xml/fmresultset.xml?-db=Contacts&-lay=Contact_Details&-findall"
2012-06-02 15:26:31 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML ERROR 5 596 "/fmi/xml/fmresultset.xml?-db=Contacts&-layout=Contact_Details&-findall"
```

- Erreurs de création de script

```
2012-06-02 17:33:12 -0700 WPC_SERVER 192.168.100.101:0 jdoe - ERROR 4 - Erreur de création de script : 4, File: "10b_MeetingsUpload", Script: "OnOpen", Script Step: "Show Custom Dialog"
```

- Modification des paramètres de publication Web personnalisée

```
2012-06-09 10:59:49 -0700 WPC_SERVER 192.168.100.101:0 jdoe - INFO - - Le moteur de publication Web XML est activé.
```

- Erreurs système

```
2012-06-02 15:30:42 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML ERROR - - La communication a échoué
```

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 consignant toutes les erreurs liées à son utilisation. Ce fichier journal est nommé web_server_module_log.txt et se trouve sur le composant Moteur de publication Web du déploiement FileMaker Server :

- IIS (Windows) : `<disque>:\Program Files\FileMaker\FileMaker Server\Logs\web_server_module_log.txt`
  où `<disque>` correspond au disque principal servant au démarrage de votre système.
- Apache (Mac OS) : `/Bibliothèque/FileMaker Server/Logs/web_server_module_log.txt`

Utilisation des journaux Tomcat

Lorsque FileMaker Server a un problème causé par une erreur de serveur Web interne, il peut être utile d'afficher les journaux Tomcat. Les journaux Tomcat sont affectés sur le composant de serveur Web du déploiement de FileMaker Server :

- IIS (Windows) : `<disque>:\Program Files\FileMaker\FileMaker Server\Admin\admin-master-tomcat\logs/`
  où `<disque>` correspond au disque principal servant au démarrage de votre système.
- Apache (Mac OS) : `/Bibliothèque/FileMaker Server/Admin/admin-master-tomcat/logs/`

Résolution des problèmes sur votre site

Si vous rencontrez des problèmes pour l'affichage ou l'utilisation de votre site, vérifiez les points suivants :

- Les privilèges d'accès étendus dans la base de données sont configurés pour la publication Web personnalisée avec PHP et attribuées à un compte utilisateur. Consultez la section « Activation de la publication Web personnalisée avec PHP pour les bases de données », page 15.
- La base de données est hébergée et 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 serveur Web et le moteur de publication Web sont opérationnels.
- La publication PHP 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 PHP** pour ouvrir une page PHP 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.

Annexe A
Codes d'erreur de la Publication Web personnalisée avec PHP

Le moteur de publication Web prend en charge deux types de codes d'erreur pouvant survenir dans le cadre de la Publication Web personnalisée :

- les erreurs de requête sur la base de données et les données. Le moteur de publication Web génère un code d'erreur chaque fois que des données font l'objet d'une requête à partir de la base de données publiée. L'API FileMaker pour PHP renvoie ce code d'erreur sous la forme d'un objet FileMaker_Error. Consultez la section suivante, « Numéros des codes d'erreur des bases de données FileMaker ».
- les erreurs PHP. Ces erreurs sont générées et renvoyées par des composants PHP, notamment le module cURL. Consultez la section « Numéros des codes d'erreur des composants PHP », page 62.

Numéros des codes d'erreur des bases de données FileMaker

En tant que développeur de la solution de Publication Web personnalisée, il vous appartient de vérifier la valeur du code d'erreur renvoyé 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

Numéro d'erreur	Description
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
112	Fenêtre manquante
113	Fonction manquante
114	Référence de fichier manquante
115	Jeu de menus manquant
116	Objet de modèle manquant
117	Source de données manquante
118	Ce thème est manquant
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 ce mot de passe ne permet pas d'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

Numéro d'erreur	Description
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
218	Le mot de passe et le mot de passe de confirmation ne correspondent pas
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
307	La transaction ne peut pas être verrouillée en raison d'une erreur de communication avec l'hôte
308	Ce thème est utilisé par un autre utilisateur
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
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

Numéro d'erreur	Description
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	Enregistrement déjà modifié par un autre utilisateur
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	Importation de fichier impossible, ou moniteur gérant les couleurs est nécessaire pour effectuer cette opération
709	L'importation d'une séquence QuickTime a échoué
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

Numéro d'erreur	Description
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 Excel non prise en charge (convertissez le fichier au format Excel 2000 ou une version ultérieure prise en charge, puis réessayez)
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 : ▪ Nom de base de données non valide ▪ Le fichier est fermé dans FileMaker Server ▪ Permission non valide
803	Le fichier est mono-utilisateur ou bien l'hôte est introuvable
804	Le fichier ne peut pas être ouvert en lecture seule dans son état actuel
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

Numéro d'erreur	Description
825	Le fichier n'est pas autorisé à référencer le fichier protégé
826	Le chemin d'accès indiqué pour le fichier est non valide
850	Le chemin est non valide pour le système d'exploitation
851	Impossible de supprimer un fichier externe du disque
852	Impossible d'écrire dans un fichier se trouvant dans le stockage externe
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 utilisée 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
960	Paramètre non valide
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 (+, -, *,).

Numéro d'erreur	Description
1213	Cette variable a déjà été définie dans la fonction Définir
1214	MOYENNE, NOMBRE, MULTIVALUEE, OBTENIRREPETITION, 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 Statistique 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 à laquelle il est fait référence n'est pas encore mise en œuvre
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
1300	Le nom indiqué ne peut pas être utilisé
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)
1409	Erreur étendue (ODBC)
1410	Erreur étendue (ODBC)
1411	Erreur étendue (ODBC)
1412	Erreur étendue (ODBC)
1413	Erreur étendue (ODBC)
1414	L'instruction SQL est trop longue
1450	Action nécessitant une extension de privilège PHP
1451	Action nécessitant une gestion à distance du fichier actif
1501	Echec de l'authentification SMTP
1502	Connexion refusée par le serveur SMTP
1503	Erreur avec SSL
1504	Le serveur SMTP requiert le cryptage de la connexion
1505	L'authentification spécifiée n'est pas prise en charge par le serveur SMTP

Numéro d'erreur	Description
1506	L'envoi du ou des emails a échoué
1507	Impossible de se connecter au serveur SMTP
1550	Impossible de charger le module, ou module non valide
1551	Impossible d'installer le module. Impossible de supprimer un module existant ou d'écrire sur le dossier ou le disque
1626	Ce protocole n'est pas pris en charg
1627	Echec de l'authentification
1628	Une erreur s'est produite avec SSL
1629	Le temps de connexion a expiré. Le délai d'expiration est de 60 secondes
1630	Le format de l'URL est incorrect
1631	Echec de la connexion

Numéros des codes d'erreur des composants PHP

L'API FileMaker pour PHP utilise plusieurs composants PHP. Ces composants PHP peuvent renvoyer des codes d'erreur supplémentaires non listés ci-dessus.

Par exemple, si les services du Noyau de publication Web ou de FileMaker Server ne sont pas exécutés, vous pouvez recevoir l'erreur du module cURL CURLE_GOT_NOTHING (52).

Pour plus d'informations sur les codes d'erreur associés à PHP, consultez le site Web du PHP à l'adresse http://php.net.

Index

A

activation Publication Web personnalisée dans base de données 15
add(), méthode 41
addSortRule(), méthode 39
Admin Console 14, 15
adresse IP dynamique 13
adresse IP statique 13
Ajouter, commande 31
année à quatre chiffres, rubrique 45
API FileMaker pour PHP 9
  définition 9
  didacticiel 29
  exemples 29
  installation manuelle 13
  référence 28
API PHP pour la Publication Web personnalisée 9
Avantages XML 10

B

bases de données, protection après publication 16
bibliothèque d'URL du client 12

C

classe FileMaker 29
clearSortRules(), méthode 39
commit(), méthode 31
compte Invité
  activation 17
  Désactivation 17
comptes et privilèges
  activation, Publication Web personnalisée 15
  compte Invité 17
  scripts 22
configuration du serveur 12
connexion
  à FileMaker Server 30
  à une base de données FileMaker 30
Conteneur, rubriques
  accès aux données par les utilisateurs Web 21
  publication du contenu 18
  téléchargement progressif 21
contrôle par le calcul 46
createRecord(), méthode 31
création d'un enregistrement 31
cURL 12
cURL, erreurs du module 62

D

date, rubrique 45
déclencheurs 24
delete(), méthode 32, 37
didacticiel sur l'API FileMaker pour PHP 29
documentation 6
documentation électronique 6
documentation en ligne 6
documentation JDBC 6
documentation ODBC 6
documentation relative à l'installation 6
dossier Web, copie des objets de rubrique Conteneur 18
duplication d'un enregistrement 32
Dupliquer, commande 32

E

édition d'un enregistrement 32
Edition, commande 32
élément de la liste de valeurs, validation 46
enregistrements 31
ensemble de résultats 43
entre, validation 45
erreurs
  fichiers journaux au serveur Web 51
  gestion 48
  numéros des codes d'erreur des bases de données 55
exécution de requêtes 39
exemples d'API FileMaker pour PHP 29

F

fichiers journaux 50
  accès au serveur Web 51
  description 51
  Tomcat 54
  web_server_module_log.txt 54
  wpe.log 52
fichiers journaux d'accès au serveur Web, description 51
field
  année à quatre chiffres 45
  Date 45
  Heure 45
  heure du jour 45
  Horodatage 45
  nombre maximal de caractères 44
  non vide 44
  numérique uniquement 44
FileMaker Server Admin
  voir Admin Console
FileMaker Server
  documentation 6
  installation 6

G

gestion des erreurs 48
getDatabase(), méthode 35
getErrors(), méthode 48

getFetchCount(), méthode 43
getField(), méthode 43
getFieldAsTimestamp(), méthode 43
getFields(), méthode 35, 43
getFoundSetCount(), méthode 43
getLayout(), méthode 35
getMessage(), méthode 48
getName(), méthode 35, 36
getRange(), méthode 39
getRecords(), méthode 43
getRelatedSet(), méthode 36
getRelatedSets(), méthode 36
getValueListsTwoFields(), méthode 38
getValueListTwoFields(), méthode 38

H

heure du jour, rubrique 45
heure, rubrique 45
horodatage Unix 43
horodatage, rubrique 43, 45

I

informations de référence 28
installation de l'API FileMaker pour PHP 13
installation manuelle de l'API FileMaker pour PHP 13
isError(), méthode 48
isValidationError(), méthode 47

J

jeu de privilèges d'accès, Publication Web personnalisée 15
journal d'application 52
Journaux Tomcat 54

L

Latin-1, codage 27
limites des sources de données ODBC 15
listes de valeurs 37
listFields(), méthode 35
listLayouts(), méthode 35
listRelatedSets(), méthode 35
listScripts(), méthode 33
listValueLists(), méthode 35, 37

M

Mac OS X, outil Server Admin 12
méthode getContainerData() 18, 43
méthode getContainerDataURL() 19, 43
méthodes
  add() 41
  addSortRule() 39
  clearSortRules() 39
  commit() 31
  createRecord() 31
  delete() 32, 37
  getContainerData() 18, 43
  getContainerDataURL() 19, 43
  getDatabase() 35
  getErrors() 48
  getFetchCount() 43
  getField() 43
  getFieldAsTimestamp() 43
  getFields() 35, 43
  getFoundSetCount() 43
  getLayout() 35
  getMessage() 48
  getName() 35, 36
  getRange() 39
  getRecords() 43
  getRelatedSet() 36
  getRelatedSets() 36
  getValueListsTwoFields() 38
  getValueListTwoFields() 38
  isError() 48
  isValidationError() 47
  listFields() 35
  listLayouts() 35
  listRelatedSets() 35
  listScripts() 33
  listValueLists() 35, 37
  newAddCommand() 31
  newCompoundFindCommand() 41
  newDeleteCommand() 32
  newDuplicateCommand() 32
  newEditCommand() 32
  newFindAllCommand() 40
  newFindAnyCommand() 40
  newFindCommand() 40
  newFindRequest() 41
  newPerformScriptCommand() 33
  newRelatedRecord() 37
  numErrors() 47, 48
  setLogicalOperator() 39
  setOmit() 41
  setPreCommandScript() 33, 39
  setPreSortScript() 34, 39
  setProperty() 30
  setRange() 39
  setRelatedSetsFilters() 44
  setResultsLayout() 35
  setScript() 34, 39
  validate() 46
modèles 35
Modifier le mot de passe, script 17
Moteur de publication Web
  codes d'erreurs générés 55
  description 8
  journal d'application 52
  traitement d'une requête 8

Mots de passe
  aucun mot de passe de connexion 17
  définition pour la Publication Web personnalisée 15
  Modifier le mot de passe, script 17

N

newAddCommand(), méthode 31
newCompoundFindCommand(), méthode 41
newDeleteCommand(), méthode 32
newDuplicateCommand(), méthode 32
newEditCommand(), méthode 32
newFindAllCommand(), méthode 40
newFindAnyCommand(), méthode 40
newFindCommand(), méthode 40
newFindRequest(), méthode 41
newPerformScriptCommand(), méthode 33
newRelatedRecord(), méthode 37
nombre maximal de caractères, rubrique 44
noms d'utilisateur
  définition pour la Publication Web personnalisée 15
non vide, rubrique 44
numérique uniquement, rubrique 44
numéros des
  codes d'erreurs des bases de données 55
numErrors(), méthode 47, 48

O

objet d'enregistrement 31
objet d'ensemble lié 35
objet de base de données 30
objets de classe FileMaker 30
  base de données 30
  enregistrement 31
  ensemble lié 35
objets de commande FileMaker
  Ajouter 31
  Dupliquer 32
  Modifier 32
  Recherche composée, commande 41
  Rechercher au moins, commande 40
  Rechercher tout, commande 40
  Rechercher, commande 39, 40
  Supprimer 32

P

PHP
  activation dans une base de données 15
  avantages 9
  erreurs 62
  problèmes 54
  récapitulatif de la procédure de publication 25
  test d'un site Web 50
  version prise en charge 13
PHP 5 12
prévalidation 44
  année à quatre chiffres 45
  commandes 44
  Date 45
  enregistrements 46
  Heure 45
  heure du jour 45
  Horodatage 45
  nombre maximal de caractères 44
  non vide 44
  numérique uniquement 44
  rubriques 46
privilège d'accès étendu, Publication Web personnalisée 15
privilèges d'accès 17
problèmes
  Publication de sites Web personnalisée 50
protection des bases de données publiées 16
publication statique, définition 7
publication sur le Web
  codes d'erreurs des bases de données 55
  objets de rubrique Conteneur 18
  protection des bases de données 16
  utilisation de PHP 25
  vidéos QuickTime 19
Publication Web instantanée.
  définition 7
  documentation 6
Publication Web personnalisée
  activation dans le moteur de publication Web 16
  activation dans une base de données 15
  avec PHP 9
  avec XML 9
  définition 7
  privilège d'accès étendu 15
  restriction d'accès au serveur Web à l'aide des adresses IP 16
  scripts 23
  utilisation des scripts 21
  vue d'ensemble 7
publication Web personnalisée XML 9

R

recherche composée
  commande 41
  exemple 42
Rechercher au moins, commande 40
Rechercher tout, commande 40
Rechercher, commande 40
Rechercher, objets de commande 39
Reconnexion, script 17
représentation de la date 43
résolution des problèmes des sites Web 54

S

SAT
  voir Admin Console

scripts 33
  astuces et considérations 21
  comptes et privilèges 22
  dans la Publication Web personnalisée 21
  déclencheurs 24
  Modifier mot de passe 17
  Reconnexion 17
sécurité
  comptes utilisateurs et mots de passe 16
  consignes pour la protection de vos bases de données publiées 16
  documentation 9
  limitation d'accès depuis des adresses IP 16
Server Admin, outil
  voir Mac OS X, Server Admin
sessions de base de données permanentes 15, 17
sessions de base de données, permanence 15, 17
setLogicalOperator(), méthode 39
setOmit(), méthode 41
setPreCommandScript(), méthode 33, 39
setPreSortScript(), méthode 34, 39
setProperty(), méthode 30
setRange(), méthode 39
setRelatedSetsFilters(), méthode 44
setResultsLayout(), méthode 35
setScript(), méthode 34, 39
sites Web
  pages d'assistance FileMaker 6
  problèmes 54
  stockage 49
  suivi 51
  Test 50
source de données SQL externe 15
SSL (Secure Sockets Layer), cryptage 16
stockage des sites Web 49
suivi des sites Web 51
suppression d'un enregistrement 32
Supprimer, commande 32

T

Table externe, boîte de dialogue 44
tables externes 35
tant que serveur Web
  fichiers journaux 51
test des sites Web 50
tester la publication PHP 54
tests technologiques 54
traitement d'un ensemble de résultats 43
traitement d'une requête du moteur de publication Web 8

U

Unicode 27
UTF-8,codage 27
Utilisateurs Web
  utilisation de données d'une rubrique Conteneur 21
utilisation
  enregistrements 31
  listes de valeurs 37
  modèles 35
  scripts 33
  tables externes 35

V

valeur existante, validation 45
valeur unique, validation 45
validate(), méthode 46
validation 44
  année à quatre chiffres 45
  commandes 44
  Date 45
  enregistrements 46
  Heure 45
  heure du jour 45
  Horodatage 45
  nombre maximal de caractères 44
  non vide 44
  numérique uniquement 44
  rubriques 46
vidéos QuickTime, publication sur le Web 19
vue d'ensemble
  publication PHP 25
  Publication Web personnalisée 7

W

web_server_module_log.txt, fichier journal 54
wpe.log, fichier journal 52