

# ADOBE® COLDFUSION® BUILDER™ ユーザーガイド

© 2010 Adobe Systems Incorporated. All rights reserved.

Adobe® ColdFusion® Builder™ ユーザーガイド

This user guide is protected under copyright law, furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

This user guide is licensed for use under the terms of the Creative Commons Attribution Non-Commercial 3.0 License. This License allows users to copy, distribute, and transmit the user guide for noncommercial purposes only so long as (1) proper attribution to Adobe is given as the owner of the user guide; and (2) any reuse or distribution of the user guide contains a notice that use of the user guide is governed by these terms. The best way to provide notice is to include the following link. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/>

Adobe, the Adobe logo, Adobe AIR, ColdFusion, ColdFusion Builder, and Flash Builder are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Java is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. Macintosh is a trademark of Apple Inc., registered in the United States and other countries. Windows is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries. All other marks are the property of their respective owners.

Updated Information/Additional Third Party Code Information available at <http://www.adobe.com/go/thirdparty>.

Portions include software under the following terms:

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

This product contains either BSAFE and/or TIPEM software by RSA Data Security, Inc.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Notice to U.S. Government End Users. The Software and Documentation are “Commercial Items,” as that term is defined at 48 C.F.R. §2.101, consisting of “Commercial Computer Software” and “Commercial Computer Software Documentation,” as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

# 目次

## 第 1 章：はじめに

ColdFusion および ColdFusion Builder のマニュアル	1
ColdFusion および ColdFusion Builder の学習用リソース	2

## 第 2 章：ColdFusion Builder について

ColdFusion Builder の設定	3
ColdFusion Builder のライセンス認証	3
ColdFusion Builder の更新	4
サードパーティ Eclipse プラグインのインストール	4
ColdFusion Builder のヘルプシステム	4
ColdFusion Builder の概要	7

## 第 3 章：ColdFusion Builder ワークベンチの基本

ワークベンチについて	10
ワークベンチのメニュー、ツールバー、およびキーボードショートカット	11
ColdFusion Builder パースペクティブについて	16
ColdFusion Builder 開発パースペクティブ	16
ColdFusion デバッグパースペクティブ	25
ワークベンチへのビューの追加	31
ColdFusion Builder のエディタ	31
パースペクティブのカスタマイズ	31

## 第 4 章：ColdFusion Builder でのコード編集

CFML エディタ	33
コードアシスト	34
コードの色付け	39
SQL エディタ	40
コードの折りたたみと展開	41
シンタックスのチェックと強調表示	42
コードリファクタリング	43
タグウィザード	45
入力設定	45
CFML エディタのキーボードショートカット	46

## 第 5 章：サーバーの管理

Web サーバーの用語について	47
ColdFusion サーバーの作成	48
[サーバー]ビューの使用	54
RDS サーバー設定のインポート	55

**第 6 章：プロジェクトの管理**

プロジェクトについて .....	56
ColdFusion プロジェクトの作成 .....	56
プロジェクトのプロパティおよびサーバーの設定 .....	57
ColdFusion ページ、インターフェイス、およびコンポーネントの追加 .....	58
URL 接頭辞の設定 .....	59
プロジェクトのインポート、エクスポート、および削除 .....	60
プロジェクトおよびファイルの除外 .....	61
ワークスペース外のリソースへのリンク .....	61
FTP および Secure FTP 接続によるプロジェクトのデプロイ .....	62
Ajax ライブラリのインポート .....	63
Flash Builder プロジェクトの使用 .....	63
AIR アプリケーションの開発 .....	63

**第 7 章：アプリケーションのデバッグ**

ColdFusion デバッガの使用 .....	66
アプリケーションのデバッグ .....	68
リモートアプリケーションのデバッグ .....	72

**第 8 章：ColdFusion Builder 拡張機能**

拡張機能について .....	73
拡張機能の開発 .....	73
拡張機能のユーザーインターフェイスの作成 .....	83
ハンドラからのコールバックコマンドの指定 .....	85
[ 拡張機能 ] ビューの使用 .....	88
Adobe Extension Builder を使用した拡張機能の作成とパッケージ化 .....	89

**第 9 章：拡張機能の使用**

Adobe CFC Generator .....	91
AS Class Generator .....	93
Model-Glue Assistant .....	96

# 第 1 章：はじめに

Adobe® ColdFusion® Builder™ は、Adobe® ColdFusion® アプリケーションを開発するための IDE (Integrated Development Environment: 統合開発環境) です。

ColdFusion Builder の使用方法を解説する資料は数多く提供されています。背景となる知識や、ColdFusion Builder の利用目的に応じて、適切なマニュアル資料に目を通すようにしてください。

## ColdFusion を初めて使用する場合

- ❖ ColdFusion Builder を使用する前に ColdFusion の基本知識を習得しておく、後で非常に役に立ちます。最初に、ColdFusion Administrator から参照できる『ファーストステップ』マニュアルに目を通しておくことをお勧めします。

## ColdFusion での開発経験がある場合

- ColdFusion Builder は Eclipse™ (オープンソース IDE) をベースに開発されているため、ColdFusion Builder を効率的に使用するには、Eclipse 固有の用語や概念を理解しておく必要があります。ただし、今までに Eclipse ベースの環境を使用した経験がある場合、ColdFusion Builder ワークベンチの操作方法はそれらの環境とほとんど変わりありません。詳細については、10 ページの「[ColdFusion Builder ワークベンチの基本](#)」を参照してください。
- 今までテキストエディタを使用してコードを開発していた場合、高機能な ColdFusion Builder を使用すると、コードの開発がはるかに容易になります。詳細については、3 ページの「[ColdFusion Builder について](#)」を参照してください。
- ColdFusion Builder の機能の概要については、7 ページの「[ColdFusion Builder の概要](#)」を参照してください。
- ColdFusion Builder を使用して効率的に ColdFusion アプリケーションを開発するには、『CFML リファレンス』と『ColdFusion アプリケーションの開発』を参照してください。『CFML リファレンス』では、CFML 言語要素について詳しく説明しています。この資料は、ColdFusion Builder のヘルプから参照できます。詳細については、6 ページの「[CFML リファレンスヘルプの表示](#)」を参照してください。
- ColdFusion および ColdFusion Builder に関連して入手できるマニュアルについては、1 ページの「[ColdFusion および ColdFusion Builder のマニュアル](#)」を参照してください。

## ColdFusion および ColdFusion Builder のマニュアル

ColdFusion および ColdFusion Builder のマニュアルは、統合ヘルプシステムを使用して ColdFusion Builder 内で参照できます。詳細については、4 ページの「[ColdFusion Builder のヘルプシステム](#)」を参照してください。

オンラインマニュアルを表示するには、ColdFusion ヘルプ & サポートページ ([www.adobe.com/go/learn\\_cfu\\_support\\_jp](http://www.adobe.com/go/learn_cfu_support_jp)) にアクセスしてください。

これらのマニュアルでは、新しいコメントを追加したり、既存のコメントを参照することもできます。

## Adobe ColdFusion のマニュアル

ColdFusion のマニュアルセットには、次のマニュアルが含まれています。

マニュアル	説明
『Adobe® ColdFusion® 9 インストール』	Windows、Macintosh、Solaris、Linux、および AIX 環境でのシステムインストールおよび基本設定について説明します。
『Adobe ColdFusion® 9 設定と管理』	サーバー設定の管理、データソースと Web サーバーの設定、セキュリティの管理、ColdFusion アプリケーションのデプロイなど、ColdFusion の管理方法について説明します。 また、キャッシュの設定、CFX タグの設定、ColdFusion サーバーモニタを使用したサーバーアクティビティの監視などについても説明しています。
『Adobe® ColdFusion® 9 アプリケーションの開発』	ダイナミック Web アプリケーションの開発方法について説明します。 このガイドでは、CFML プログラミング言語と ColdFusion の機能 (ColdFusion Web サービス、ColdFusion ポートレット、ColdFusion ORM、AJAX サポート、Flex および AIR との統合など) の使用方法と、他社の製品およびテクノロジー (Microsoft Office、OpenOffice、SharePoint など) との統合に関する詳細情報が記載されています。
『Adobe® ColdFusion® 9 CFML リファレンス』	すべての ColdFusion タグ、関数、変数に関する説明、シンタックス、使用方法、コード例が含まれています。

## Adobe ColdFusion Builder のマニュアル

ColdFusion Builder のマニュアルセットには、次のマニュアルが含まれています。

マニュアル	説明
『Adobe® ColdFusion® Builder インストール』	Windows プラットフォームおよび Macintosh プラットフォームにおいて、ColdFusion Builder をスタンドアロン設定でインストールする方法、または既存の Eclipse ワークベンチのプラグインとしてインストールする方法について説明します。
『ColdFusion® Builder ユーザーガイド』	ColdFusion アプリケーションを開発するための IDE (Integrated Development Environment: 統合開発環境) として ColdFusion Builder を使用方法について説明します。

## ColdFusion および ColdFusion Builder の学習用リソース

ColdFusion および ColdFusion Builder の学習用に、次のリソースがオンラインで提供されています。

リソース	リンク
ColdFusion ヘルプ & サポート	<a href="http://www.adobe.com/go/learn_cfu_support_jp">www.adobe.com/go/learn_cfu_support_jp</a>
ColdFusion デベロッパーセンター	<a href="http://www.adobe.com/go/learn_cfu_cfdevcenter_jp">www.adobe.com/go/learn_cfu_cfdevcenter_jp</a>
Adobe ColdFusion オンラインフォーラム	<a href="http://www.adobe.com/go/learn_cfu_forums_jp">www.adobe.com/go/learn_cfu_forums_jp</a>

## 第 2 章 : ColdFusion Builder について

Adobe® ColdFusion® Builder™ は、オープンソース IDE (Integrated Development Environment: 統合開発環境) である Eclipse™ をベースに開発されました。IDE は、開発者が必要とするツールの大部分を 1 つにまとめたアプリケーションです。ColdFusion Builder には、コードの編集と検証、ファイルとプロジェクトの管理、およびデバッグ用のツールが用意されています。ColdFusion Builder のプラグインアーキテクチャを利用すると、必要に応じて機能を拡張できます。

ColdFusion Builder は Microsoft® Windows® プラットフォームおよび Apple® Macintosh プラットフォームで動作します。ColdFusion Builder のインストール設定オプションでは、スタンドアロン設定で ColdFusion Builder をインストールするか、既存の Eclipse ワークベンチインストール内にプラグインセットとして ColdFusion Builder をインストールするかを選択できます。詳細については、『Adobe ColdFusion Builder インストール』を参照してください。

### ColdFusion Builder の設定

ColdFusion Builder インストーラには、次の 2 つの設定オプションがあります。

**スタンドアロン設定の ColdFusion Builder IDE インストール** Eclipse 3.5.1 をベースに構築されたスタンドアロン IDE (Integrated Development Environment: 統合開発環境) として ColdFusion Builder をインストールします。

**Eclipse IDE 内の ColdFusion Builder プラグイン** 既存の Eclipse 環境または Adobe® Flash® Builder™ 環境内のプラグインとして ColdFusion Builder をインストールします。既に Eclipse ワークベンチを使用していて、Eclipse プラグインのツールキットに ColdFusion Builder プラグインを追加したいユーザーに適した設定です。

**注意:** プラグイン設定で ColdFusion Builder をインストールするには、コンピュータに Eclipse 3.4.2 以上がインストールされている必要があります。

どちらの設定を使用すればよいか分からない場合は、次のガイドラインに従ってください。

- 既に Eclipse 3.4.2 以上がインストールされている場合は、プラグイン設定を選択してください。
- まだ Eclipse 3.4.2 以上がインストールされていない場合は、スタンドアロン設定を選択してください。この設定では他の Eclipse プラグインもインストールできるので、将来の開発作業の幅が広がります。

どちらの設定も機能的には変わりません。ただし、プラグイン設定では、メニューの名前や、メニューコマンドへのアクセス方法が少し異なる場合があります。

### ColdFusion Builder のライセンス認証

ColdFusion Builder を起動すると、製品シリアル番号の入力を求めるメッセージが表示されます。有効な製品シリアル番号を入力すると、インターネット接続が最初に検出されたときに、ライセンス認証がバックグラウンドで実行されます。詳細については、Adobe 製品ライセンス認証センター ([www.adobe.com/go/activation\\_jp](http://www.adobe.com/go/activation_jp)) にアクセスしてください。

ColdFusion Builder の起動時に製品シリアル番号を入力しない場合は、ColdFusion Builder の体験版を 60 日間使用できます。体験版の有効期限が切れた後も ColdFusion Builder を引き続き使用するためには、ライセンスを購入する必要があります。

### ColdFusion Builder ライセンスの管理

シングルライセンスユーザーの場合は、最大 2 台のコンピュータで ColdFusion Builder を使用できます。最初の 2 台のコンピュータ以外のコンピュータで ColdFusion Builder のライセンスを使用するには、2 台のコンピュータのいずれかの ColdFusion Builder のライセンス認証を解除します。その後、別のコンピュータでライセンスを使用できます。

ColdFusion Builder のライセンス認証を解除するには、[ ヘルプ ]-[Adobe ColdFusion Builder のライセンス認証解除] を選択します。

**注意：** ColdFusion Builder をアンインストールする場合は、まずライセンス認証を解除してから、ColdFusion Builder をアンインストールすることをお勧めします。

## ColdFusion Builder の更新

ColdFusion Builder の更新には、セキュリティに対する変更や製品の機能が含まれる場合があります。更新を定期的に確認して、入手可能になればインストールすることをお勧めします。

- 1 Adobe ColdFusion Builder で、[ ヘルプ ]-[Adobe ColdFusion Builder の更新] を選択します。新しい更新が入手可能であれば、更新がダウンロードされてインストールされます。
- 2 更新を有効にするには、ColdFusion Builder を再起動 (スタンドアローン設定の場合) するか、Eclipse を再起動 (プラグイン設定の場合) します。

## サードパーティ Eclipse プラグインのインストール

ColdFusion Builder は、開発環境の拡張やカスタマイズを可能にするサードパーティ製 Eclipse プラグインとの統合をサポートしています。サードパーティ製 Eclipse プラグインは、Windows および Mac OS X プラットフォームにインストールできます。

- 1 Adobe ColdFusion Builder で、[ ヘルプ ]-[新規ソフトウェアをインストール] を選択します。
- 2 [追加] をクリックします。
- 3 インストールするプラグインの名前と URL を入力します。たとえば、Subclipse バージョン 1.0 をインストールするには、[http://subclipse.tigris.org/update\\_1.0.x](http://subclipse.tigris.org/update_1.0.x) という URL を入力します。
- 4 インストールするプラグインを選択します。
- 5 [次へ] をクリックし、インストールウィザードの指示に従います。
- 6 [終了] をクリックします。

**注意：** プラグインが正しくインストールされていることを確認するには、Galileo 更新サイトの URL <http://download.eclipse.org/releases/galileo> (または同等の Eclipse 更新サイトの URL) を追加します。

## ColdFusion Builder のヘルプシステム

ColdFusion Builder で作業する際には、ワークベンチのユーザーインターフェイス要素 (ビューとダイアログボックス) に関する状況依存型ヘルプと、コード要素に関する言語リファレンスヘルプを表示できます。

## Adobe Community Help Client

Adobe CHC (Community Help Client) は、ColdFusion Builder のヘルプコンテンツを表示する AIR アプリケーションです。ヘルプコンテンツは Web から直接提供されます。CHC を初めて開く場合は、インターネット接続が必要です。その後、ヘルプコンテンツをデスクトップにダウンロードすると、オフラインで参照または検索できます。

ColdFusion Builder のヘルプシステムを開いて表示するには、[ ヘルプ ]-[ColdFusion Builder ヘルプ] を選択します。

### 状況依存型ヘルプの使用

- 1 ワークベンチで、ビューやダイアログボックスなどのユーザーインターフェイス要素を選択します。
  - 2 Windows の場合は、F1 キーを押すと、関連する [ ヘルプ ] ページがコミュニティヘルプクライアントで開きます。  
Mac OS で状況依存型ヘルプを使用するには、ダイナミックヘルプコマンドへのキーボードショートカットを設定します。ダイナミックヘルプコマンドへのキーボードショートカットを指定するには、次のようにします。
- 1 [Adobe ColdFusion Builder]-[ 環境設定 ] を選択します。
  - 2 ツリービューで、[ 一般 ]-[ キー ] を選択します。
  - 3 ダイナミックヘルプコマンドを選択します。
  - 4 設定するキーの組み合わせを押します。たとえば、Ctrl + Shift + 1 と入力するには、Ctrl キーと Shift キーを押した状態で 1 キーを押します。キーの間のプラス記号 (+) は、キーを連続して押す必要があることを示しています。

### Community Help Client の設定

CHC の設定を調整すると、ヘルプコンテンツの表示を制御できます。

- 1 CHC で、[ 編集 ]-[ 環境設定 ] を選択します。
- 2 [ 環境設定 ] ダイアログボックスで、必要に応じて次のオプションを設定します。

#### 一般設定

- CHC のテーマとして、明るいテーマや暗いテーマを選択できます。テーマは CHC の外観を決定する形式設定です。たとえば、暗いテーマを適用すると、CHC で暗い背景とテーマ固有のフォントが使用されます。
- デフォルトでは、CHC は AIR アプリケーションとして起動されます。ただし、Web ブラウザ内で CHC を起動することもできます。その場合は、[ ブラウザーでヘルプを起動します ] を選択します。

#### ダウンロードの環境設定

オフラインで利用する製品とヘルプパッケージを選択できます。

#### アップデーターの設定

CHC では、AIR アプリケーションを再インストールしなくても、コンテンツと機能を更新できます。CHC を開くとき、または閉じるときに、新しい更新に関するメッセージを表示するには、[ アップデートが提供されたら通知する ] を選択します。更新を手動で検索してインストールすることもできます。

#### ローカルコンテンツ

デスクトップにダウンロードしたヘルプパッケージを手動で削除できます。

### 検索とフィルタのオプション

CHC では、複数のヘルプパッケージや検索場所にまたがる検索を実行できます。

[ 検索オプション ] をクリックすると、次の検索場所が表示されます。

**ローカルヘルプ** オフラインで表示するためにデスクトップにダウンロードしたヘルプパッケージを検索します。[ フィルター結果 ] をクリックすると、検索結果がヘルプパッケージ内の特定のブックに絞り込まれます。

**Adobe.com** デベロッパーセンターの記事、チュートリアル、ナレッジベースの記事など、adobe.com で入手できるすべてのオンラインリソースから信頼性の高い最新のリファレンスコンテンツを検索します。

**コミュニティヘルプ** adobe.com で入手できるすべてのリソースに加え、Adobe コミュニティのエキスパートから投稿された実用的なコンテンツを検索します。

**コード検索** 固有のコード検索機能を使用して、adobe.com で入手可能なコードスニペットを検索できます。

## ダイナミックナビゲーション

CHC には、関連するコンテンツを検索するために、検索結果に基づくダイナミックナビゲーション機能が用意されています。関連するコンテンツはすべて、[ 関連トピック ] タブに表示されます。

## フィードバックと評価

有効な Adobe ID を持っている場合は、ヘルプコンテンツに対するコメントの追加、評価、寄稿などを行えます。Adobe ID をお持ちでない場合は、<https://www.adobe.com/jp/membership> で登録することで、ID を取得できます。

CHC の左下隅にある [ フィードバックと評価 ] をクリックし、フィードバックを記入します。フィードバックは、ヒントなどの形式で記入することも、関連するオンラインリソースへのリンクを示すこともできます。

## Eclipse のヘルプシステム

Eclipse のヘルプシステムには、コード要素に関する CFML リファレンスヘルプが表示されます。また、Eclipse のヘルプシステムには、インストールされているサードパーティプラグインや、ColdFusion Builder に付属するプラグインのヘルプも表示されます。

Eclipse のヘルプシステムを開いて表示するには、[ ヘルプ ]-[ ヘルプ目次 ] を選択します。

## CFML リファレンスヘルプの表示

ColdFusion Builder には『CFML リファレンス』が組み込まれているので、CFML タグや関数のリファレンスヘルプをすばやく参照できます。

- CFML タグや関数のヘルプトピックを開くには、次のようにします。

- 1 CFML エディタで、ビルトインタグや関数名を強調表示するか、その場所にマウスポインタを置いて、CFML タグ要素または関数を選択します。
- 2 Windows の場合は F1 キー、Mac OS の場合は相当するキーボードショートカットを押します。選択したタグに関するヘルプページへのリンクが、ドッキングされたヘルプウィンドウに表示されます。  
**注意 :** Mac OS の場合は、Command + Shift + / キーを押して CFML タグまたは関数のヘルプを表示すると、ドッキングされたヘルプウィンドウに全般的なエディタのヘルプが表示されます。タグまたは関数に固有のヘルプを表示するには、CFML エディタで選択したタグまたは関数名を再度クリックします。
- 3 ヘルプページリンクをクリックします。選択したタグまたは関数に関するヘルプページが表示されます。

## ヘルプの設定

ヘルプの設定を調整すると、Eclipse ヘルプシステムでのヘルプの表示方法を制御できます。

- 1 ColdFusion Builder で [ ウィンドウ ]-[ 設定 ] を選択します。

2 [設定] ダイアログボックスの左側にツリービュー構造が表示されます。[ヘルプ] を選択します。

3 必要に応じて、次のオプションを設定します。

**[外部ブラウザを使用]** 外部の Web ブラウザにヘルプを表示できます。デフォルトでは、IDE に組み込まれているブラウザにヘルプが表示されます。Web ブラウザを選択するには、[外部ブラウザを使用] チェックボックスを選択し、[Web ブラウザー] リンクを選択します。

**[ウィンドウコンテキストヘルプを開く]** ウィンドウを開く状況依存型ヘルプリンクの表示方法を決定します。デフォルトでは、状況依存型ヘルプリンクは、ダイナミックヘルプビューに表示されます。ダイナミックヘルプビューは、他のビューと同様に、現在のパースペクティブにドッキングされます。状況依存型ヘルプリンクを情報ポップアップ (ツールヒントに類似したもの) に表示するには、[情報ポップアップ内] を選択します。

**[ダイアログコンテキストヘルプを開く]** ダイアログボックスを開く状況依存型ヘルプリンクの表示方法を決定します。デフォルトでは、ダイアログボックスにヘルプが表示されます。状況依存型ヘルプリンクを情報ポップアップ (ツールヒントに類似したもの) に表示するには、[情報ポップアップ内] を選択します。

**[ヘルプビュードキュメントを開く]** ダイナミックヘルプのリンクから開かれたドキュメントをどこに表示するかを決定します。デフォルトでは、[インプレース] が選択されており、ヘルプドキュメントはダイナミックヘルプビューウィンドウに表示されます。ヘルプドキュメントを ColdFusion Builder IDE の編集領域で開くには、[インプレース] を選択します。

## ダイナミックヘルプの使用

ダイナミックヘルプは現在のパースペクティブにドッキングされ、関連するビューとダイアログボックスについてのトピックが表示されます。

❖ [ヘルプ]-[ダイナミック・ヘルプ] を選択します。

デフォルトのキーボードショートカットバインディングを変更するには、[ウィンドウ]-[設定]-[一般]-[キー] を選択し、[ダイナミックヘルプ] コマンドに割り当てられたバインディングを変更します。キーボードショートカットを変更する方法の詳細については、15 ページの「[ショートカットキー](#)」を参照してください。

# ColdFusion Builder の概要

ColdFusion Builder を使用すると、次の作業を行うための機能をすべて備えた IDE で ColdFusion アプリケーションを開発できます。

**注意:** 作業を開始する前に、ColdFusion Builder ワークベンチとその機能について理解しておいてください。詳細については、10 ページの「[ColdFusion Builder ワークベンチの基本](#)」を参照してください。

- **ColdFusion プロジェクトの作成:** プロジェクトウィザードを使用してプロジェクトを作成し、そのプロジェクトで使用するローカルサーバーまたはリモートサーバーを設定できます。詳細については、56 ページの「[ColdFusion プロジェクトの作成](#)」を参照してください。

- **サーバーの追加と管理:** ColdFusion Builder にはサーバー管理機能が組み込まれており、ColdFusion Builder 内で ColdFusion サーバー、ColdFusion Administrator、およびサーバーモニタを管理できます。

JRun サーバーと JRun 以外のサーバーの両方を設定できます。アプリケーションを効率的に管理するために、ColdFusion Builder には仮想ホスト、仮想ディレクトリ、URL 接頭辞などの機能が用意されています。

これらの機能の使用方法与サーバー管理の詳細については、47 ページの「[サーバーの管理](#)」を参照してください。

- **FTP 接続によるリモートプロジェクトのデプロイ:** ColdFusion Builder では、ローカルサーバーとリモートサーバーの両方を管理できます。リモートサーバーへの接続には、RDS (Remote Data Services) 接続または FTP (File Transfer

Protocol: ファイル転送プロトコル) 接続を使用できます。FTP 接続を介したリモートプロジェクトのデプロイと同期の詳細については、62 ページの「[FTP および Secure FTP 接続によるプロジェクトのデプロイ](#)」を参照してください。

- **ColdFusion ページ、コンポーネント、およびインターフェイスの作成**: ColdFusion Builder には、ColdFusion ページ、ColdFusion コンポーネント (CFC)、インターフェイスなどのリソースを作成するためのウィザードが用意されています。これらのリソースはプロジェクトに関連付けることができます。詳細については、58 ページの「[ColdFusion ページ、インターフェイス、およびコンポーネントの追加](#)」を参照してください。

- **CFML エディタを使用した ColdFusion アプリケーションのソースコードの記述と編集**: ColdFusion Builder には、CFML、HTML、JavaScript、および CSS のコードを編集するためのコードエディタが用意されています。編集するコードのタイプに応じて、適切なエディタが開きます。詳細については、31 ページの「[ColdFusion Builder のエディタ](#)」を参照してください。

これらのエディタには、コードの色付け、コードアシスト、[アウトライン] ビューなど、コードのナビゲーションを容易にするための機能が数多く含まれています。

CFML エディタには、コード補完、コードリファクタリング、効率的なコードナビゲーションなどの機能が用意されています。ColdFusion Builder では、さまざまな色やフォントを使用してエディタ内のコードを表示できます。CFML エディタの機能の詳細については、33 ページの「[ColdFusion Builder でのコード編集](#)」を参照してください。

- **SQL エディタを使用した SQL ステートメントの作成と編集**: CFML エディタには SQL エディタが組み込まれており、SQL ステートメントを作成または編集できます。サポートされている SQL ステートメントのコードの色付けを設定することもできます。SQL エディタの詳細については、40 ページの「[SQL エディタ](#)」を参照してください。
- **CFML 辞書の使用と作成**: ColdFusion Builder には、CFML コードの入力を支援する CFML 辞書が組み込まれています。CFML 辞書は、ColdFusion バージョン 7、8、および 9 でサポートされています。独自のカスタム CFML 辞書を作成することもできます。辞書の使用と作成の詳細については、38 ページの「[CFML 辞書](#)」を参照してください。
- **ColdFusion Builder 開発ビューの使用**: ColdFusion Builder には、アプリケーションの開発を容易にするさまざまなビューがあります。

[RDS ファイルビュー] では、ローカルサーバーとリモートサーバーのファイルシステムにアクセスして、ファイルを検索できます。[RDS データビュー] では、ローカルまたはリモートのデータソースにアクセスして、データを検索できます。詳細については、18 ページの「[RDS ファイルビューおよび RDS データビュー](#)」を参照してください。

また、RDS クエリービューアを使用して、ローカルデータサーバーまたはリモートデータサーバー上のデータを照会することもできます。詳細については、19 ページの「[RDS クエリービューア](#)」を参照してください。

[サービスブラウザ] ビューでは、サーバーの Web ルート内にある CFC とそのメソッドを参照できます。ローカルとリモート両方の CFC および Web サービスにアクセスできます。詳細については、22 ページの「[\[サービスブラウザ\] ビュー](#)」を参照してください。

[スニペット] ビューと [アウトライン] ビューを使用すると、コードの再利用や簡素化が容易になります。詳細については、23 ページの「[\[スニペット\] ビュー](#)」および 17 ページの「[\[アウトライン\] ビュー](#)」を参照してください。

すべての開発ビューの詳細については、16 ページの「[ColdFusion Builder 開発パースペクティブ](#)」を参照してください。

- **ColdFusion debugger を使用したアプリケーションのデバッグ**: ColdFusion Builder には、アプリケーションをデバッグできる [デバッグ] ビューや [ブレークポイント] ビューなどのデバッグビューを備えた統合デバッガが用意されています。これらのビューの詳細については、25 ページの「[ColdFusion デバッガパースペクティブ](#)」を参照してください。

コードに対するブレークポイントの追加、関数のステップインやステップオーバー、式の検査と評価を行えます。ローカルとリモートの両方のサーバーにあるファイルをデバッグできます。詳細については、66 ページの「[アプリケーションのデバッグ](#)」を参照してください。

[問題] ビュー、[TailView] ビュー、および [コンソール] ビューでは、シンタックスエラー、サーバーエラー、およびコンパイルエラーが検出されて表示されます。

- **ColdFusion Builder 拡張機能の使用と開発** : ColdFusion Builder 拡張機能を管理するには、[ 拡張機能 ] ビューを使用します。

コードを生成したり、ユーザーインターフェイスを設計したり、データベースに対する基本的な CRUD ( 作成、読み込み、更新、削除 ) 操作を実行できます。CFML エディタでのファイルのオープン、開いたファイルへのテキストの挿入などのカスタムアクションを実行する拡張機能も開発できます。拡張機能の作成の詳細については、73 ページの「[ColdFusion Builder 拡張機能](#)」を参照してください。

ColdFusion Builder には、拡張機能の作成とパッケージ化に必要な手順をガイドする Adobe Extension Builder が用意されています。詳細については、89 ページの「[Adobe Extension Builder を使用した拡張機能の作成とパッケージ化](#)」を参照してください。

ColdFusion Builder には次の拡張機能が付属しています。

- Adobe CFC Generator
- ActionScript Class Generator
- Model-Glue Assistant

これらの拡張機能のインストールと使用については、91 ページの「[拡張機能の使用](#)」を参照してください。

## 第3章：ColdFusion Builder ワークベンチの基本

ColdFusion Builder は Eclipse ベースの開発環境であり、これを利用すると Adobe ColdFusion アプリケーションを開発して ColdFusion サーバーで実行できます。ColdFusion Builder では、コーディングツールやデバッグツールを利用して ColdFusion アプリケーションを開発できます。

### ワークベンチについて

ColdFusion Builder ワークベンチは、Adobe ColdFusion アプリケーションの開発に必要な機能をすべて備えた環境です。ColdFusion Builder IDE の基本機能の多くは、Eclipse から受け継いでいます。ColdFusion Builder プラグインをインストールすると、ColdFusion アプリケーションの作成に必要な機能を追加できます。また、これらのプラグインには、IDE のユーザーインターフェイスを変更するためのツールもあり、アプリケーションのビルドをサポートする主要機能が提供されます。

**ワークベンチ** ワークベンチは ColdFusion Builder の開発環境です。ワークベンチには、パースペクティブ、エディターおよびビューという 3 つの主要要素があります。アプリケーション開発プロセスの各種段階で 3 つの要素をさまざまに組み合わせて使用します。ワークベンチには、アプリケーションの開発に必要なツールがすべて用意されています。

**パースペクティブ** パースペクティブは、ワークベンチにあるビュー、エディタ、メニュー、およびツールバーのグループです。これは、基本的には特定の種類の作業に役立つ特別な作業環境です。たとえば、ColdFusion Builder には 2 つのパースペクティブがあります。アプリケーションの開発には ColdFusion Builder 開発パースペクティブを使用し、アプリケーションのデバッグには ColdFusion デバッグパースペクティブを使用します。

パースペクティブの詳細については、16 ページの「[ColdFusion Builder パースペクティブについて](#)」を参照してください。

**エディタ** エディタではさまざまな種類のファイルを編集できます。使用できるエディタは、インストールされている Eclipse プラグインの数と種類によって異なります。ColdFusion Builder には、CFML、HTML、JavaScript、および CSS (Cascading Style Sheets: カスケードスタイルシート) のコードを記述するためのエディタが含まれています。ColdFusion Builder でのコード編集の詳細については、31 ページの「[ColdFusion Builder のエディタ](#)」および 33 ページの「[ColdFusion Builder でのコード編集](#)」を参照してください。

**ビュー** ビューは一般的にエディタをサポートします。たとえば、CFML を編集するときには、ColdFusion Builder 開発パースペクティブに [アウトライン] ビューと [スニペット] ビューも表示されます。これらのビューは、ColdFusion アプリケーションの開発をサポートするため、CFML ファイルが編集用に開かれる则表示されます。

ビューの中にはワークベンチ自体の主要な機能をサポートするものがあります。たとえば、ファイルエクスプローラビューでは、ワークベンチ内のファイルおよびフォルダを管理できます。[RDS データビュー] および [RDS ファイルビュー] には、リモートサーバーとローカルサーバーのデータソース、ファイル、およびディレクトリが表示されます。

ビューという用語は、Adobe Dreamweaver® やその他の Adobe 開発ツールで使用されるパネルという用語と同義です。

ColdFusion Builder 開発パースペクティブのビューの詳細については、16 ページの「[ColdFusion Builder 開発パースペクティブ](#)」を参照してください。

ColdFusion デバッグパースペクティブのビューの詳細については、25 ページの「[ColdFusion デバッグパースペクティブ](#)」を参照してください。

**ワークスペース** ワークベンチと混同しないようにしてください。ワークスペースとは、ファイルシステム内の特定の領域です。ワークスペースには、アプリケーションプロジェクトの構成要素となるリソース (ファイルとフォルダ) が格納されます。1 つのワークスペースに複数のプロジェクトを格納することもできます。一度に使用できるワークスペースは 1 つだけ

ですが、ColdFusion Builder を起動するたびに異なるワークスペースを選択できます。詳細については、56 ページの「プロジェクトの管理」を参照してください。

**リソース** リソースという一般用語は、ワークスペース内のプロジェクトに含まれるファイルとフォルダを意味します。詳細については、58 ページの「ColdFusion ページ、インターフェイス、およびコンポーネントの追加」 および 59 ページの「その他のファイルの追加」を参照してください。

**プロジェクト** アプリケーションを構成するリソースはすべてプロジェクト内にあります。最初にプロジェクトを作成しないと、ColdFusion Builder でアプリケーションを構築できません。詳細については、56 ページの「プロジェクトの管理」を参照してください。

## ワークベンチのメニュー、ツールバー、およびキーボードショートカット

ワークベンチのコマンドにアクセスするには、メニューバー、右クリックのコンテキストメニュー、ツールバー、およびキーボードショートカットを使用します。

### ワークベンチのメニュー

ColdFusion Builder ワークベンチには次のメインメニューがあります。

#### [ファイル] メニュー

[ファイル] メニューでは、ワークベンチリソースの作成、保存、エディタの終了、印刷、インポート、エクスポート、およびワークベンチの終了を行います。

メニューコマンド	説明
[新規]	リソースを作成します。ColdFusion コンポーネント、インターフェイス、またはページを作成する前に、それらのリソースを保存するためのプロジェクトを作成しておく必要があります。
[ファイルを開く]	編集するファイルを開きます。ワークスペースに含まれないファイルを開くこともできます。
[閉じる]	アクティブなエディタを閉じます。ファイルを閉じる前に、変更を保存するかどうかを確認するメッセージが表示されます。
[すべて閉じる]	開いているエディタをすべて閉じます。ファイルを閉じる前に、変更を保存するかどうかを確認するメッセージが表示されます。
[保管]	アクティブなエディタの内容を保存します。
[別名保管]	アクティブなエディタの内容を別のファイル名で他の場所に保存できます。
[すべて保管]	開いているすべてのエディタの内容を保存します。
[前回保管した状態に戻す]	アクティブなエディタの内容を、前回保存した内容に戻します。
[移動]	現在選択されているリソースを別のプロジェクトに移動します。
[名前変更]	選択されているリソースの名前を変更します。
[更新]	リソースにファイルシステムの現在の内容を反映します。

メニューコマンド	説明
[行区切り文字の変換]	開発プラットフォームまたはデブレイ先プラットフォームのオペレーティングシステムに合わせて行区切り文字を変換します。 <ul style="list-style-type: none"> <li>• Windows (デフォルト)</li> <li>• UNIX</li> <li>• MacOS 9</li> </ul>
[印刷]	アクティブなエディタの内容を印刷します。
[ワークスペースの切り替え]	別のワークスペースに切り替えるための [ワークスペース・ランチャー] を開きます。ワークベンチは、別のワークスペースに切り替えるたびに再起動されます。
[再開]	ColdFusion Builder を再起動します。
[インポート]	インポートウィザードを使用して、ワークベンチにリソースをインポートします。
[エクスポート]	エクスポートウィザードを使用して、ワークベンチからリソースをエクスポートします。
[プロパティ]	選択されているリソースのプロパティダイアログボックスを表示します。プロパティダイアログボックスには、リソースのパスと最終変更日が表示されます。また、プロジェクトのリソースが親のエンコードや行区切り文字を継承しているか、特定の値に設定されているかに関する情報も表示されます。
最近使ったファイルのリスト	最近ワークベンチでアクセスしたファイルのリストです。これらのファイルを開くには、[ファイル]メニューでファイル名をクリックします。このリストに表示するファイルの数は、[設定]ダイアログボックスで設定できます。
[終了]	開いているリソースを閉じて、ワークベンチを終了します。

## [ナビゲート]メニュー

[ナビゲート]メニューでは、ワークベンチ内のリソースを探して、該当する場所に移動できます。

メニューコマンド	説明
[次へジャンプ]	選択したリソースがルートに来るように、リソースの階層内を移動します。つまり、プロジェクト内のフォルダを選択してから [次へジャンプ] を選択すると、選択したフォルダ (およびフォルダ内のアーティファクト) のみが [ナビゲーター] ビューに表示されます。  このコマンドは、複雑な階層構造を持つ大規模なプロジェクト内を移動するときに便利です。
[ジャンプ]	特定のリソースにジャンプするか、前に表示されていた階層に戻ります。  たとえば、特定のリソースを選択して [次へジャンプ] コマンドを選択した後に、[ジャンプ]-[戻る] を選択するとします。[戻る] コマンドを実行すると、[次へジャンプ] コマンドの実行前と同一の階層が表示されます。
[宣言を開く]	選択したコード要素の宣言を開きます。
[クイック・アウトライン]	[アウトライン] ビューの簡易版を表示します。ページのコード構造が階層状になってポップアップメニューに表示されます。
[リソースを開く]	[リソースを開く] ダイアログボックスを使用して、ワークスペース内のリソースを選択し、エディタで開きます。
[表示]	別のビューで現在選択されているリソースを探して選択できます。エディタがアクティブになっている場合は、これらのコマンドを使用して、別のビューで現在編集されているリソースを選択できます。
[次へ]	アクティブなビューのリストまたはテーブル内で次の項目に移動します。
[前へ]	アクティブなビューのリストまたはテーブル内で前の項目に移動します。
[最後の編集位置]	アクティブなエディタ内で最後に編集した位置に移動します。

メニューコマンド	説明
[ 指定行へジャンプ ]	アクティブなエディタ内の特定の行に移動します。
[ 戻る ]	以前に開いたファイルにエディタのフォーカスを移動します。
[ 進む ]	以前のファイルからエディタのフォーカスを戻します。

## [ プロジェクト ] メニュー

[ プロジェクト ] メニューでは、ワークベンチのプロジェクトを管理します。

メニューコマンド	説明
[ プロジェクトを開く ]	現在選択されているプロジェクトを開きます。このコマンドは、現在選択されているプロジェクトが閉じられている場合にのみ使用できます。
[ プロジェクトを閉じる ]	現在開いているプロジェクトを閉じます。
[ プロパティ ]	プロジェクトのプロパティダイアログボックスを表示します。

## [ ウィンドウ ] メニュー

[ ウィンドウ ] メニューでは、ワークベンチの各種ビュー、パースペクティブ、およびアクションの表示、非表示、管理を行います。

メニューコマンド	説明
[ 新規ウィンドウ ]	現在開いているパースペクティブと同じパースペクティブで、新しいワークベンチウィンドウを開きます。
[ 新規エディター ]	現在アクティブなエディタと同じタイプのエディタを開きます。
[ パースペクティブを開く ]	ワークベンチ内に新しいパースペクティブを開きます。
[ ビューの表示 ]	ワークベンチに表示するビューを選択できます。
[ パースペクティブのカスタマイズ ]	現在選択されているパースペクティブをカスタマイズできます。
[ パースペクティブの別名保管 ]	現在選択されているパースペクティブを保存し、カスタマイズした独自のパースペクティブを作成します。
[ パースペクティブのリセット ]	カスタマイズしたパースペクティブを元の状態に戻します。
[ パースペクティブを閉じる ]	現在開いているパースペクティブを閉じます。
[ すべてのパースペクティブを閉じる ]	ワークベンチで開いているパースペクティブをすべて閉じます。
[ ナビゲーション ]	ワークベンチのビュー、パースペクティブ、エディタを切り替えるためのショートカットがあります。
[ 設定 ]	ワークベンチを使用するための設定を指定できます。ワークベンチ本体だけでなく、ワークベンチに含まれるビューやエディタの外観に関連するさまざまな設定項目があります。

## ツールバー

### ワークベンチツールバー

ワークベンチツールバーには、頻繁に使用される重要なコマンドのボタンがあります。これらのコマンドは、ColdFusion Builder の各種メニューからも使用できます。



ワークベンチツールバー

ワークベンチツールバーには、次のボタンが ( 左から右の順に ) 表示されます。

ボタン / コマンド	説明
[新規]	作成できるプロジェクトとドキュメントのすべてのタイプを表示するポップアップメニューが表示されます。
[保管]	エディタで開かれていて現在選択されているドキュメントを保存します。
[印刷]	エディタで開かれていて現在選択されているドキュメントを印刷します。
[新規 ColdFusion プロジェクト]	ColdFusion プロジェクトを作成するためのウィザードを開きます。
[CFC を開く]	[ナビゲーター] ビューに表示されているプロジェクトで使用できるすべての CFC を表示します。
[デバッグ]	プロジェクトで現在開かれているファイルを使用してデバッグセッションを開始します。付帯しているポップアップメニューから、プロジェクト内の他のアプリケーションファイルを選択することもできます。
[実行]	プロジェクトで現在開かれているファイルを実行します。付帯しているポップアップメニューから、プロジェクト内の他のアプリケーションファイルを選択することもできます。
[外部ツール]	カスタム起動設定を選択します。
[検索]	プロジェクト内のファイルのテキスト文字列およびファイル名のパターンを検索します。
[次の注釈]	先の位置にあるコード注釈を選択して移動できます。
[前の注釈]	前の位置にあるコード注釈を選択して移動できます。
[最後の編集位置]	現在開いているファイルで最後に編集したコード要素に移動します。
[戻る]	以前に開いたファイルにエディタのフォーカスを移動します。
[進む]	以前のファイルからエディタのフォーカスを戻します。

## CFML エディタのツールバー

CFML エディタのユーザーインターフェイスには、使用頻度の高いコマンドをすばやく実行するためのボタンがあります。これらのコマンドは、ColdFusion Builder の各種メニューからも使用できます。このツールバーには使用頻度の高いコード要素へのショートカットとして機能するボタンもあり、それらのボタンを使用できるかどうかは CFML エディタで入力したコードの内容に依存します。



CFML エディタのツールバー

ボタン / コマンド	説明
[TagWizard を開く]	タグウィザードを開きます。 タグウィザードの詳細については、45 ページの「 <a href="#">タグウィザード</a> 」を参照してください。
[cfabort の挿入]	cfabort タグを挿入します。 このコマンドを使用できるかどうかはスクリプトコードの内容に依存します。
[cfdump の挿入]	cfdump タグを挿入します。このコマンドを使用できるかどうかはスクリプトコードの内容に依存します。

ボタン / コマンド	説明
[cfoutput で囲む]	選択したコードを cfoutput タグで囲みます。 このコマンドを使用できるかどうかは、CFML エディタで入力したスクリプトコードの内容に依存します。
[cfset の挿入]	cfset タグを挿入します。
[cfscript ブロックの挿入]	cfscript コードブロックを挿入します。
[CF Try/Catch で囲む]	内容に応じて、選択したコードを cftry または cfcatch タグで囲みます。 このコマンドを使用できるかどうかはスクリプトコードの内容に依存します。
[## で囲む]	選択したコードを # マークで囲みます。
[CF コメント <!-- で囲む / 解除]	選択したコードをコメント化するか、選択したコードのコメント化を解除します。
[cfscript コメント /* */ で囲む / 解除]	選択したコードを /* */ マークで囲むか、選択したコードから /* */ マークを外します。
[一重引用符で囲む]	選択したコードを一重引用符で囲みます。
[二重引用符で囲む]	選択したコードを二重引用符で囲みます。
[大文字へ]	選択したコードのテキストを大文字に変更します。
[小文字へ]	選択したコードのテキストを小文字に変更します。

## ショートカットキー

ColdFusion Builder のメニューから実行できる機能の大部分は、キーボードショートカットからも実行できます。キーボードショートカットのリストを表示するには、次のようにします。

❖ [ヘルプ]-[キー・アシスト] を選択します。

### キーボードショートカットの変更

既存のキーボードショートカットを変更するには、次のようにします。

- 1 ColdFusion Builder で [ウィンドウ]-[設定] を選択します。
- 2 [設定] ダイアログボックスの左側にツリービュー構造が表示されます。
- 3 ツリービュー構造で、[一般]-[キー] を選択します。
- 4 コマンドとキーのバインディングの中から、変更する項目を選択します。たとえば、[エディターをアクティブにする] コマンドは F12 キーに関連付けられています。
- 5 設定するキーの組み合わせを押します。たとえば、「Ctrl+Shift+F」と入力するには、Ctrl と Shift を押しながら F を押します。キーの間のプラス記号 (+) は、キーを順番に押す必要があることを表します。
- 6 デフォルトのキーバインディングのコンテキストを変更する場合は、[場合] ドロップダウンリストを使用します。キーバインディングのコンテキストに応じて、キーボードショートカットがアクティブになる状況が決定されます。
- 7 同じキーボードショートカットに別のキーバインディングを追加するには、コマンドを選択して [コマンドのコピー] をクリックします。別のキーバインディングの組み合わせを設定できます。
- 8 コマンドに関連付けられているキーバインディングをデフォルトの設定に戻すには、変更したコマンドを選択し、[コマンドの復元] をクリックします。
- 9 コマンドに関連付けられているキーバインディングを削除するには、コマンドを選択し、[コマンドのアンバインド] をクリックします。
- 10 [適用] または [OK] をクリックします。

## ColdFusion Builder パースペクティブについて

パースペクティブとは、作業に必要なエディタとビューを1つにまとめたものを指します。ColdFusion Builder には、ColdFusion Builder 開発パースペクティブと ColdFusion Builder デバッグパースペクティブという2つのパースペクティブがあります。

### パースペクティブの表示と切り替え

特定のパースペクティブに関連付けられているファイルを開くと、そのパースペクティブが自動的に開きます。つまり、パースペクティブは、実行しようとしているタスクに合わせて自動的に変化します。たとえば、ColdFusion プロジェクトを作成する場合は、ワークベンチに開発パースペクティブが表示されます。同様に、デバッグセッションを始めるときには、ColdFusion Builder がデバッグパースペクティブに切り替わります。

デフォルトでは、同じウィンドウ内に複数のパースペクティブが開きます。パースペクティブを新しいウィンドウで開くには、次のようにします。

- 1 [ウィンドウ]-[設定] を選択します。
- 2 ツリービュー構造で、[一般]-[パースペクティブ] を選択します。
- 3 [新規パースペクティブを開く方法] で、[同じウィンドウ] を選択します。

手でパースペクティブを切り替えるには、次のいずれかの操作を行います。

- メインメニューで [ウィンドウ]-[パースペクティブを開く]-[その他] を選択します。
- ワークベンチのメインツールバーにあるパースペクティブバーを使用します。

Eclipse には定義済みのパースペクティブが数多く用意されています。したがって、プラグイン設定の ColdFusion Builder を使用している場合は、他のパースペクティブも使用できます。

Eclipse パースペクティブを表示するには、次のいずれかの操作を行います。

- メインメニューで [ウィンドウ]-[パースペクティブを開く]-[その他] を選択します。
- ワークベンチのメインツールバーにあるパースペクティブバーで  をクリックし、[その他] を選択します。

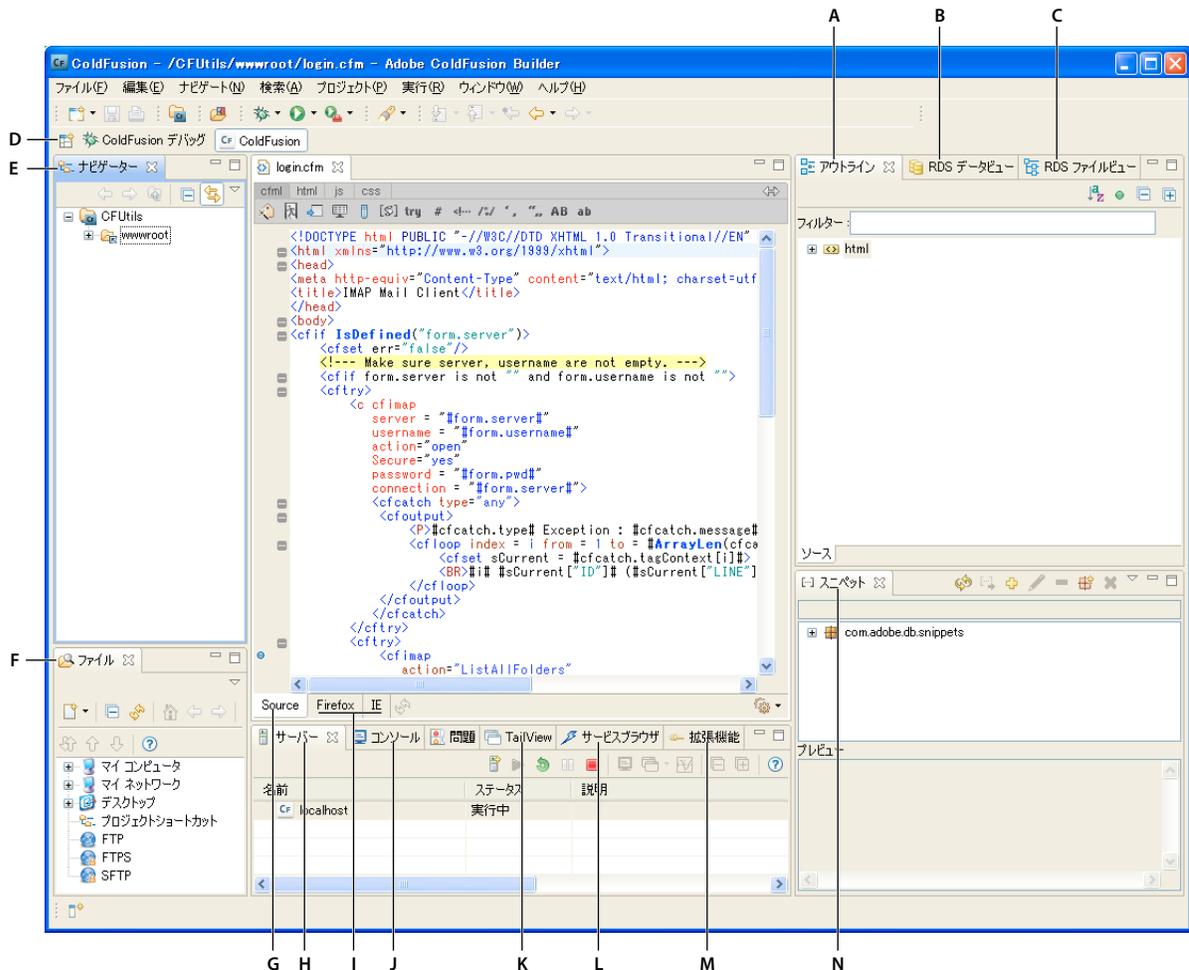
### デフォルトのパースペクティブの設定

- 1 [ウィンドウ]-[設定] を選択します。
- 2 ツリービュー構造で、[一般]-[パースペクティブ] を選択します。
- 3 [選択可能なパースペクティブ] でデフォルトとして設定するパースペクティブを選択し、[デフォルトにする] をクリックします。
- 4 [OK] をクリックします。

デフォルトに設定したパースペクティブの名前の後には「(デフォルト)」と表示されます。

## ColdFusion Builder 開発パースペクティブ

ColdFusion Builder 開発パースペクティブは、ColdFusion Builder で ColdFusion アプリケーションを作成、編集、設定、または実行するときに使用します。さらに、ローカルやリモートの ColdFusion サーバーの設定と管理もできます。



ColdFusion パースペクティブ

A. [アウトライン]ビュー B. RDS データビュー C. RDS ファイルビュー D. パースペクティブバー E. [ナビゲーター]ビュー F. [ファイル]ビュー G. [ソース]ビュー H. [サーバー]ビュー I. デフォルトブラウザビュー J. [コンソール]ビュー K. [TailView]ビュー L. サービスブラウザー M. [拡張機能]ビュー N. [スニペット]ビュー

ColdFusion Builder 開発パースペクティブには次のビューがあります。

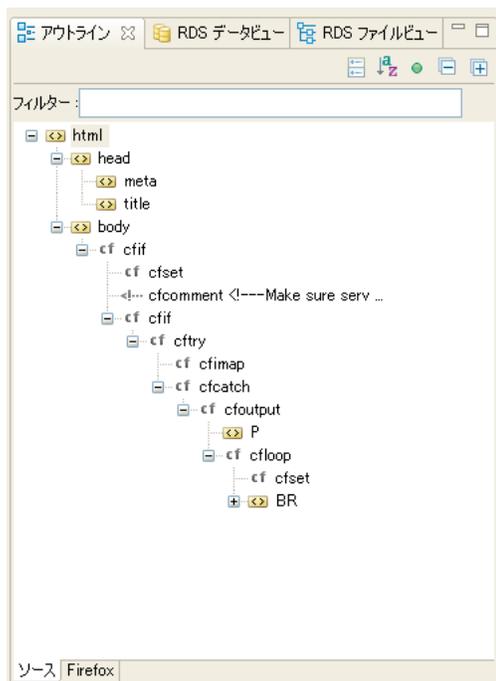
## [アウトライン]ビュー

[アウトライン]ビューには、エディタで現在開いているファイル内の要素の階層が表示されます。たとえば、CFC ファイル内の関数や HTML ファイル内のタグが表示されます。

[アウトライン]ビューでは、CFML、HTML、JavaScript、および CSS ページの構造の調査とナビゲートを行います。複数のコード要素を含むページでは、[アウトライン]ビューを使用して、ページのコード構造を階層形式で表示できます。

[アウトライン]ビューの要素は、アルファベット順、またはページ内の要素定義の順に並べ替えることができます。このビューでは、すべてのコード要素を表示するか、特定のコード要素のみを表示するかを選択できます。

- [アウトライン]ビューには、CFML コードの構造が表示されます。たとえば、[アウトライン]ビューの各アイテムは CFML タグを表す場合があります。



[アウトライン] ビュー

- コード全体をスクロールせずに、エディタ内の要素に直接移動するには、[アウトライン]ビューで要素をダブルクリックします。[アウトライン]ビューでアイテムを選択すると、そのアイテムがエディタ内で強調表示されるので、コードのナビゲートがより簡単になります。
- [アウトライン]ビュー内のタグや要素を検索するには、フィルタを使用します。[フィルタ]フィールドに、検索するタグまたは要素の名前を入力します。一致する文字列が[アウトライン]ビューに表示されます。

[アウトライン]ビューで表示する CFML タグを指定するには、[ColdFusion]-[エディタプロファイル]-[エディタ]-[アウトライン]を選択します。必要に応じて、タグを選択、追加、または削除します。

## RDS ファイルビューおよび RDS データビュー

Remote Data Services (RDS) を使用するには、ColdFusion サーバーのインストール時に RDS を有効にします。ColdFusion Builder には、リモートサーバー上のファイルやデータソースにアクセスするためのビューが用意されています。

RDS ファイルビューには、リモートサーバーおよびローカルサーバーのファイルとディレクトリが表示されます。RDS データビューには、リモートサーバーで設定されたデータソースが表示されます。

ColdFusion Builder で追加した ColdFusion サーバーインスタンスは自動的に RDS ファイルビューおよび RDS データビューからアクセス可能になります。

### RDS ファイルビューでのリモートサーバーの設定

RDS ファイルビューを使用してリモートサーバーを手動で設定し、そのサーバー上のファイルにアクセスするには、次のようにします。

- 1 ColdFusion Builder パースペクティブの右上隅にある [RDS ファイルビュー] をクリックします。
- 2 RDS ファイルビュー内で右クリックし、[RDS 構成] を選択します。
- 3 [新規] をクリックします。

- 4 リモートサーバーの情報 ( ホスト名、ポート番号、ユーザー名、パスワードなど ) を指定します。
- 5 [ パスワードを要求 ] を選択します。
- 6 [ 接続のテスト ] をクリックして、RDS 構成が正しいことを確認します。[OK] をクリックします。
- 7 RDS ファイルビューからリモートサーバーを選択します。
- 8 リモートサーバー上のファイルとディレクトリを表示するためのパスワードを指定します。

#### RDS データビューでのリモートサーバーの設定

RDS データビューを使用してリモートサーバーを設定し、リモートサーバー上のデータソースにアクセスするには、次のようにします。

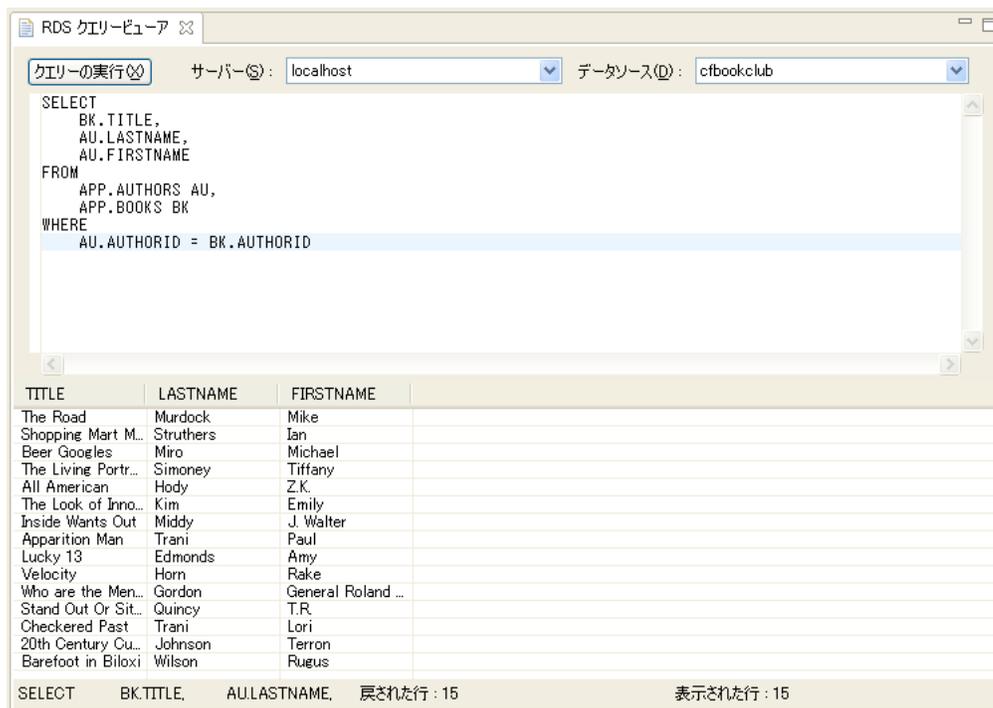
- 1 [RDS データビュー] をクリックします。
- 2 RDS データビュー内を右クリックし、[RDS 構成] を選択します。
- 3 [新規] をクリックします。
- 4 サーバーの情報 ( ホスト名、ポート番号、リモートサーバーを使用するためのユーザー名、パスワードなど ) を指定します。
- 5 [パスワードを要求] を選択します。
- 6 [接続のテスト] をクリックして、RDS 構成が正しいことを確認します。[OK] をクリックします。
- 7 新しく設定したサーバーを RDS データビューで展開して、データソースを表示します。

#### RDS クエリービューア

RDS クエリービューアでは、選択したデータソースに対するクエリーを作成して実行できます。

RDS クエリービューアは [RDS データビュー] のツールバーから起動できます。RDS クエリービューアを使用してクエリーを作成または実行するには、次のようにします。

- 1 RDS データビューで  をクリックして、[RDS クエリービューア] 領域を開きます。
- 2 空白の領域にクエリーを入力します。  
[RDS データビュー] ビューから [RDS クエリービューア] にテーブルと列をドラッグしてクエリーを作成することもできます。
- 3 ドロップダウンリストからサーバーとデータソースを選択します。
- 4 [クエリーの実行] をクリックします。[RDS クエリービューア] 領域にクエリーの結果が表示されます。



RDS クエリービューア

[RDS データビュー] ビューのテーブルを右クリックして、[テーブルの内容の表示] を選択します。RDS クエリービューアでは、選択したテーブルにあるすべてのレコードを表示できます。

## [ナビゲーター] ビュー

[ナビゲーター] ビューには ColdFusion プロジェクトやその他のプロジェクトが表示され、プロジェクトのファイルやフォルダの間をナビゲートできます。

## [ファイル] ビュー

[ファイル] ビューには ColdFusion Builder がインストールされているローカルコンピュータ上のファイルとディレクトリが表示され、新しいファイルを作成できます。

ColdFusion Builder の外部でファイルを編集した場合は、そのファイルを ColdFusion Builder で自動的に更新できます。これを行うには、[ウィンドウ]-[設定]-[一般]-[ワークスペース] を選択し、[自動的に更新] を選択します。

また、FTP サイトを追加したり、FTP サーバー上のファイルを参照することもできます。ColdFusion プロジェクトを FTP 接続に関連付ければ、ファイルのアップロード、ダウンロード、および同期を簡単に行えます。詳細については、62 ページの「[FTP および Secure FTP 接続によるプロジェクトのデプロイ](#)」を参照してください。

## [サーバー] ビュー

ColdFusion サーバーの詳細 (サーバー名、ステータス、説明、サーバータイプ、サーバーホスト、ポートなど) が表示されます。[サーバー] ビューでは次の作業を行えます。

- ColdFusion サーバーの追加と削除
- サーバーの起動と停止

- サーバーモニタの起動
- ColdFusion Server Administrator の起動
- サーバーログを開く

詳細については、47 ページの「[サーバーの管理](#)」を参照してください。

## [ソース]ビューとデフォルトブラウザビュー

[ソース]ビューには、ファイルのソースコードが表示されます。デフォルトブラウザビューには、HTML または CFML ページの出力が表示されます。

## [コンソール]ビュー

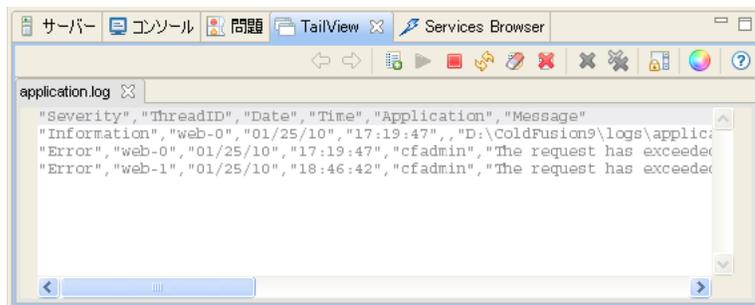
[コンソール]ビューには、ColdFusion サーバーの状態と、ColdFusion サーバーからのメッセージが表示されます。

## [TailView]ビュー

[TailView]ビューには、サーバーログとワークスペースログが表示され、ログを簡単にナビゲートできます。ログを自動的に表示するには、RDS を構成して有効にする必要があります。

サーバーを選択した後は、Server Manager のボタンバーにあるショートカットからログを開くことができます。ログを選択すると、[TailView]ビューにログの内容が表示されます。デフォルトでは、[TailView]ビューに例外ログが表示されません。[サーバー]ビューを使用している場合は、ビュー内を右クリックして [ログを開く] を選択すると、例外ログが開きます。

ログを手動で開くには、[TailView]ビューで右クリックして、[追加]を選択します。コンピュータのファイルシステムにあるログファイルを参照して選択します。



[TailView]ビュー

[TailView]ビューのツールバーには、(左から右の順に)次のボタンがあります。これらのボタンを使用すると、複数のログを個別のタブで管理できます。



[TailView]ツールバー

ボタン/コマンド	説明
[左に移動]	サーバーログまたはワークスペースログを左に移動します。
[右に移動]	サーバーログまたはワークスペースログを右に移動します。
[ログを追加します]	コンピュータのファイルシステムでログファイルを開きます。

ボタン/コマンド	説明
[ ログファイルの監視を開始します ]	ログファイルの監視を開始します。
[ ログファイルの監視を停止します ]	ログファイルの監視を停止します。
[ ログファイルを再ロードします ]	ログファイルを再ロードします。
[ 表示されているログファイルをクリアします ]	[TailView] ビューに表示されているログファイルを削除します。ただし、コンピュータのファイルシステムからは削除されません。
[ ディスク上のログファイルを消去 ]	コンピュータのファイルシステムからログファイルを削除します。
[ ログタブを削除します ]	コンピュータのファイルシステムからログファイルを削除します。
[ すべてのログタブを削除します ]	[TailView] ビューからすべてのログタブを削除します。
[ スクロールロック ]	[ コンソール ] ビューがスクロールしないように固定します。
[ 色の設定 ]	[TailView] ビューにログファイルを表示するときの色を設定します。

## [ サービスブラウザ ] ビュー

[ サービスブラウザ ] ビューには、Server Manager で追加されたサーバーが一覧表示され、CFC (ColdFusion Component: ColdFusion コンポーネント)、および Web サービスとそのメソッドを参照できます。

### ColdFusion コンポーネントの表示

[ サービスブラウザ ] ビューには、次のコンポーネントが一覧表示されます。

- ColdFusion コンポーネントブラウザに一覧表示されるコンポーネント。ColdFusion コンポーネントブラウザの場所は、"<ColdFusion のルートディレクトリ>¥wwwroot¥CFIDE¥componentutils¥componentdoc.cfm" です。
- ColdFusion Administrator のマッピングページで指定されたディレクトリに配置されているコンポーネント
- ColdFusion Administrator のカスタムタグのパスページで指定されたディレクトリに配置されているコンポーネント

[ サービスブラウザ ] ビューのツールバーには、( 左から右の順に ) 次のボタンがあります。これらのボタンを使用すると、さまざまなコンポーネントを表示できます。CFC 内の関数がリモート関数、パブリック関数、プライベート関数のいずれであるかに応じて、一覧に表示する CFC を絞り込むことができます。



[ サービスブラウザ ] ツールバー

ボタン/コマンド	説明
[ システム CFC の表示 ]	サーバーの Web ルート内にある CFC を一覧表示します。
[ リモートの表示 ]	リモート関数のみを含む CFC を一覧表示します。
[ Public の表示 ]	パブリック関数のみを含む CFC を一覧表示します。
[ パッケージの表示 ]	
[ Private の表示 ]	プライベート関数のみを含む CFC を一覧表示します。

### Web サービスの管理

[ サービスブラウザ ] ビューでは、Web サービスの一覧に対して WSDL URL の追加や削除を行うことで、Web サービスの一覧を管理できます。

- Web サービスの一覧を表示するには、[ サービスブラウザ ] ビューで  をクリックします。

- 新しい Web サービスを一覧に追加するには、 をクリックし、有効な WSDL URL を入力して [OK] をクリックします。
- Web サービスを削除するには、該当する Web サービスを選択して  をクリックします。

ColdFusion ファイルを編集する際には、[ サービスブラウザ ] ビューを使用して、Web サービスの実行や Web サービスオブジェクトの作成を行う CFML コードを生成できます。同様に、ActionScript ファイルを編集する際にも、サービスブラウザを使用して ActionScript を生成できます。

Web サービスの実行または作成を行うには、次のようにします。

- 1 コードを挿入する位置にポインタを配置します。
- 2 Web サービスの一覧を表示します。
- 3 Web サービスまたは Web サービス内のメソッドを強調表示し、右クリックして次のいずれかを選択します。
  - Web サービスを挿入する場合は [CFOBJECT の挿入]
  - Web サービスを作成する場合は [CreateObject の挿入]

## [ 拡張機能 ] ビュー

[ 拡張機能 ] ビューでは次の作業を行えます。

- 88 ページの「[拡張機能のインストールとアンインストール](#)」
- 89 ページの「[拡張機能のインポートとリロード](#)」

[ 拡張機能 ] ビューには、拡張機能名やインストールされている拡張機能の説明などの詳細も表示されます。

[ 拡張機能 ] ビューのツールバーには ( 左から右の順に ) 次のボタンがあります。これらのボタンを使用すると、[ 拡張機能 ] ビューで拡張機能を管理できます。



[ 拡張機能 ] ビューのツールバー

ボタン/コマンド	説明
[ アーカイブファイルから拡張機能をインストール ]	インストールするアーカイブファイルを選択して、拡張機能のインストールウィザードを開きます。
[ フォルダから拡張機能をインポート ]	拡張機能ファイルを含むフォルダを選択します。
[ 選択した拡張機能をアンインストール ]	選択した拡張機能をアンインストールします。
[ 選択した拡張機能の詳細を表示 ]	選択した拡張機能の詳細を表示します。

## [ スニペット ] ビュー

コードスニペットとは、ファイルに挿入して何度でも再利用できるテキストであり、これを作成することにより、同じテキストを毎回入力する必要がなくなります。コードスニペット、変数、およびプレーンテキストを作成して保存するには、[ColdFusion スニペット] ビューを使用します。コードスニペットは、既存のドキュメントまたは新規のドキュメントに追加できます。

[ スニペット ] ビューのツールバーには ( 左から右の順に ) 次のボタンがあります。これらのボタンを使用すると、[ スニペット ] ビューでコードスニペットを管理できます。



コードスニペットツールバー

ボタン/コマンド	説明
[スニペットビューの更新]	[スニペット]ビューの表示を更新します。
[選択されているスニペットをドキュメントに挿入]	選択されているスニペットをドキュメントに挿入します。
[新規スニペットの作成]	コードスニペットを作成します。
[選択されているスニペットの編集]	選択されているコードスニペットを編集できます。
[選択されているスニペットの削除]	選択されているコードスニペットを削除します。
[スニペットパッケージの新規作成]	コードスニペットを追加するためのスニペットパッケージを作成します。
[選択されているスニペットパッケージの削除]	選択されているスニペットパッケージを削除します (パッケージが空の場合のみ)。スニペットを含んでいるスニペットパッケージを削除することはできません。

### コードスニペットの作成

- 1 コードスニペット領域で  をクリックします。
- 2 [新規スニペット] ダイアログボックスで、スニペット名、コードの開始ブロック、および終了ブロックを指定します。
- 3 [OK] をクリックするとコードスニペットが作成されます。

### コードスニペットの挿入

- 1 挿入ポイントにカーソルを移動します。
- 2 コードスニペット領域でコードスニペットを選択します。
- 3 コードスニペットツールバーで  をクリックします。

エディタでコードを選択してスニペットを挿入すると、選択したコードの開始ブロックと終了ブロックの間にスニペットがラップされます。エディタでコードを選択せずにスニペットを挿入すると、現在のカーソルの位置にスニペットが挿入されます。カーレットとは、次の文字が出現する位置を示す CFML エディタ内のマーカーです。

コードスニペット領域では既存のコードスニペットやコードスニペットパッケージ全体を編集したり削除したりすることもできます。

### トリガテキストを使用したスニペットの挿入

トリガテキストを使用して CFML エディタにスニペットを挿入することもできます。たとえば、スニペットを作成し、そのスニペットに "abc" というトリガテキストを設定します。このスニペットをエディタに挿入するには、「abc」と入力し、Ctrl + J キー (Windows の場合) または Command + J キー (Mac の場合) を押します。テキスト abc がスニペットに置換されます。

### スニペットを保存するパスの指定

スニペットは、スニペットのディレクトリに XML ファイルとして保存されます。デフォルトでは、スニペットはプロジェクトワークスペースの `workspace/.metadata/snippets` ディレクトリに格納されます。別のディレクトリを指定するには、次のようにします。

- 1 [ウィンドウ] メニューから [設定] を選択します。
- 2 [設定] ダイアログボックスのツリービュー構造で、[ColdFusion]-[スニペット] を選択します。
- 3 スニペットを保存するディレクトリのパスを指定します。

### システム定義スニペット変数

スニペット変数の大文字と小文字は区別されます。次のシステム定義変数が用意されています。

変数	説明
\$\$ {DATE}	4桁の年号を含むシステム日付。例: 12/01/2009
\$\$ {TIME}	12時間表記のシステム時刻。例: 15:15:05 PM
\$\$ {DATETIME}	システム日時。例: 12/01/2009 15:15:05 PM
\$\$ {DAYOFWEEK}	完全な曜日名。例: Monday
\$\$ {CURRENTFILE}	現在開いているファイル名。例: application.cfm
\$\$ {CURRENTFOLDER}	現在開いているフォルダの完全修飾パス。例: C:\workspace\myproject
\$\$ {CURRENTPATH}	現在開いているファイルの完全修飾パス。例: C:\workspace\myproject\application.cfm
\$\$ {CURRENTPRJPATH}	現在開いているプロジェクトのプロジェクト名。例: myproject
\$\$ {USERNAME}	現在のユーザーの名前。
\$\$ {MONTHNUMBER}	月 (01 ~ 12)。
\$\$ {DAYOFMONTH}	日 (01 ~ 31)。
\$\$ {DAYOFWEEKNUMBER}	1 ~ 7の数値で表した曜日。日曜日が週の始まりになります。したがって、月曜日は2になります。
\$\$ {DATETIME24}	24時間表記の日時。例: 12/01/2009 15:30:00
\$\$ {YEAR}	4桁の数値で表した現在の年。例: 2009
\$\$ {YEAR2DIGIT}	2桁の数値で表した現在の年。例: 09

たとえば、現在の日付を挿入するには、次のシステム定義変数を使用します。

```
Date: $$ {DATE}
```

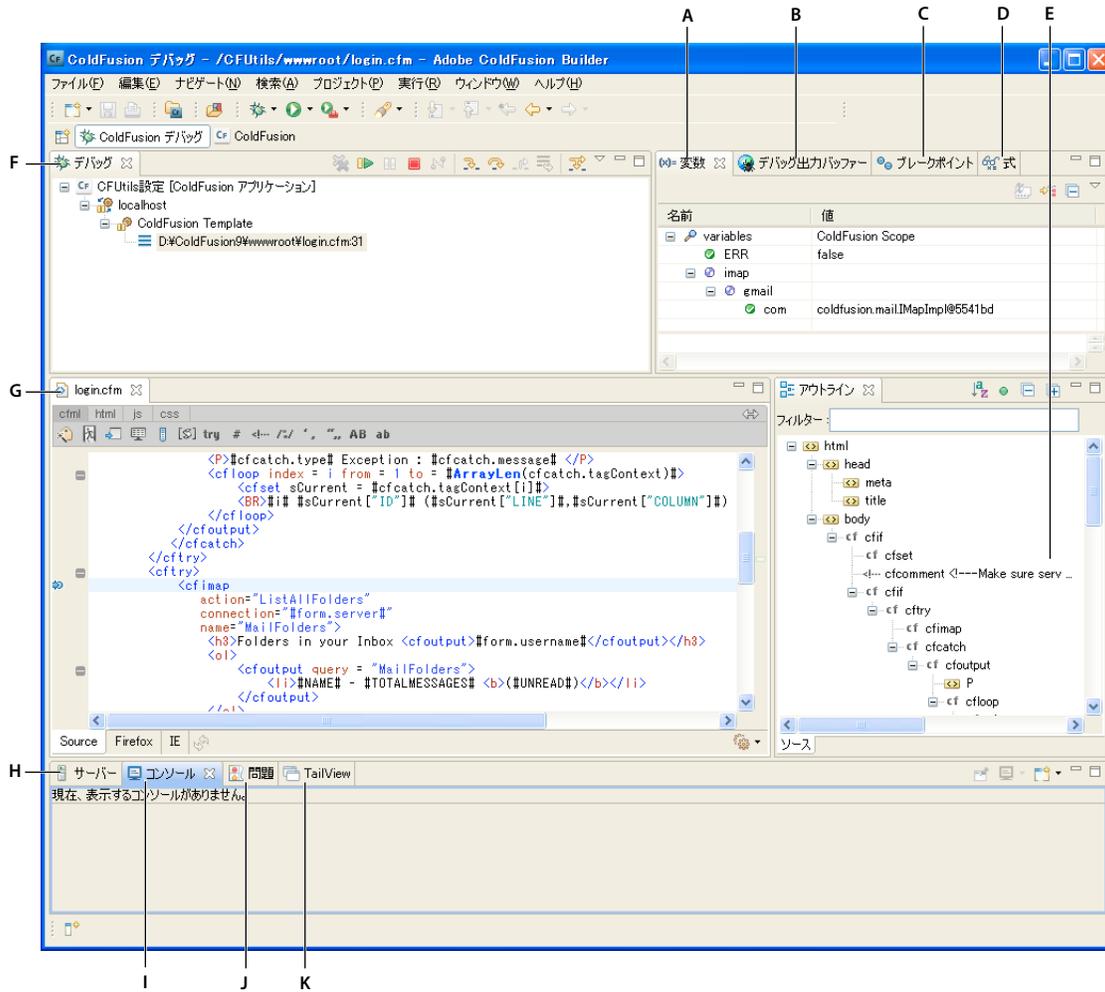
#### ユーザー定義スニペット変数

ユーザー定義変数では、変数名とデフォルト値の入力を求められます。たとえば、次のユーザー定義変数では、作成者名の入力を求められます。

```
Author name: $$ {Author}
```

## ColdFusion デバッグパースペクティブ

ColdFusion Debugging デバッグパースペクティブには、ColdFusion アプリケーションをデバッグするためのツールがあります。また、コードにブレークポイントを追加したり、コード内の関数にステップインまたはステップオーバーしたり、式を評価するためのさまざまなビューが用意されています。エディタとデバッグツールを組み合わせると、修正が必要なコード行を検索して強調表示することができます。



ColdFusion デバッグパースペクティブ

A. [変数] ビュー B. [デバッグ出力バッファ] ビュー C. [ブレイクポイント] ビュー D. [式] ビュー E. [アウトライン] ビュー F. [デバッグ] ビュー G. 編集ビュー H. [サーバー] ビュー I. [コンソール] ビュー J. [問題] ビュー K. [TailView] ビュー

ColdFusion デバッグパースペクティブには次のビューがあります。

## [変数] ビュー

[変数] ビューのツールバーには、(左から右の順に) 次のボタンが付いています。これらのボタンを使用すると、現在の変数 (変数スコープも含む) を表示できます。



[変数] ビューのツールバー

ボタン/コマンド	説明
[ 型名の表示 ]	変数の型を表示します。
[ 論理構造の表示 ]	このコマンドは、ColdFusion Builder ではサポートされていません。
[ すべてを折りたたむ ]	ビュー内の情報を折りたたんで、変数の型のみを表示します。

## [ デバッグ出力バッファ ] ビュー

[ デバッグ出力バッファ ] には次の 2 のペインがあります。

**[ ブラウザ ]** アプリケーションの実行中にブラウザに出力される内容を表示します。デバッグするページの URL を指定します。

**注意:** [ ホーム ] ボタンをクリックすると、[ 設定 ] ダイアログボックス ([ ウィンドウ ]-[ 設定 ]-[ ColdFusion ]-[ デバッグ の設定 ]) でホームページの URL として指定した URL が表示されます。デバッグの設定を編集する方法については、67 ページの「[ColdFusion Builder でのデバッグ設定の指定](#)」を参照してください。

**[ サーバー出力バッファ ]** デバッグの出力を 2 つのビュー ( ソースビューと HTML ビュー ) に表示します。

[ サーバー出力バッファ ] ペインには、デバッグがブレークポイントで一時停止したときのみ出力が表示されます。表示される内容は、一時停止したブレークポイントに到達するまでの出力のみです。

**注意:** [ サーバー出力バッファ ] には、外部ブラウザで実行されたページの出力も表示されます。

## [ ブレークポイント ] ビュー

[ ブレークポイント ] ビューのツールバーには、( 左から右の順に ) 次のボタンがあります。これらのボタンを使用するとデバッグセッション中にブレークポイントを管理できます。

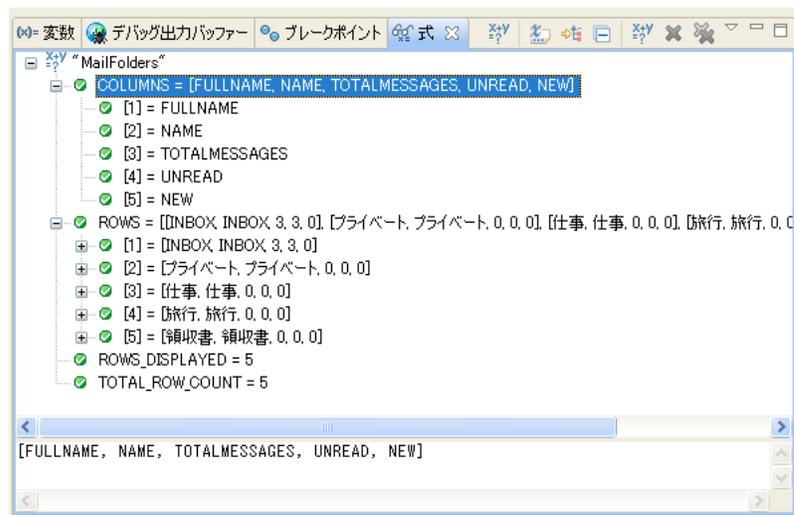


[ ブレークポイント ] ビューのツールバー

ボタン/コマンド	説明
[ 選択されたブレークポイントを削除 ]	選択されたブレークポイントを削除します。
[ すべてのブレークポイントを削除 ]	すべてのブレークポイントを削除します。
[ 選択したターゲットでサポートされるブレークポイントを表示 ]	現在デバッグ中のブレークポイントを表示します。
[ ブレークポイント用のファイルヘジャンプ ]	選択されたブレークポイントが設定されているファイルに移動します。
[ すべてのブレークポイントをスキップ ]	すべてのブレークポイントを無視します。
[ すべてを展開 ]	ビュー内のブレークポイント情報をすべて展開します。
[ すべてを折りたたむ ]	ビュー内のブレークポイント情報をすべて折りたたみます。
[ デバッグ・ビューにリンク ]	アプリケーションの実行が停止されたときに、選択されたブレークポイントを [ デバッグ ] ビューで強調表示します。
[ Java 例外ブレークポイントの追加 ]	選択されたブレークポイントに達したときに返す Java 例外を指定できます。
[ メニュー ]	[ ブレークポイント ] ビューに表示する情報の種類を指定できます。

## [ 式 ] ビュー

[ 式 ] ビューは、変数や関数を使用して式を作成し、式の検査、評価、監視を行うときに使用します。また、[ 変数 ] ビューで選択した変数の評価と監視を行うこともできます。



[式] ビュー

[式] ビューのツールバーには、(左から右の順に) 次のボタンがあります。これらのボタンを使用すると、式の作成、評価、監視を行えます。



[式] ビューのツールバー

ボタン/コマンド	説明
[型名の表示]	変数の型を表示します。
[論理構造の表示]	ビューに論理構造を表示します。
[すべてを折りたたむ]	ビュー内の式をすべて折りたたみます。
[新規監視式を作成]	監視式を追加します。
[選択された式の削除]	選択された変数または監視式を削除します。
[すべての式を削除]	[式] ビューに表示されている変数と監視式をすべて削除します。

## [アウトライン] ビュー

[アウトライン] ビューには、現在のソースファイルの内容がアウトライン形式で表示されます。詳細については、17 ページの「[\[アウトライン\] ビュー](#)」を参照してください。

## [デバッグ] ビュー

[デバッグ] ビューには、各デバッグセッションの結果が表示されます。[デバッグ] ビューには、ブレークポイントでページの実行が中断したとき、またはコードにステップインまたはステップオーバーしたときのスタックトレースが表示されません。

[デバッグ] ツールバーには、(左から右の順に) 次のボタンがあります。



[デバッグ] ツールバー

ボタン/コマンド	説明
[再開]	デバッグセッションを再開します。
[中断]	デバッグセッションを一時停止します。
[終了]	デバッグセッションを停止します。
[切断]	リモートでデバッグしている場合に、選択したデバッグターゲットからデバッグを切断します。
[終了したすべての起動を削除]	終了したすべてのデバッグターゲットをビューから消去します。
[ステップイン]	インクルードされたコード、UDF、CFCを含めて、コードを行単位で実行します。
[ステップオーバー]	インクルードされたコード、UDF、CFCを除いて、コードを行単位で実行します。
[ステップ・リターン]	インクルードされたコード、UDF、CFCの呼び出し元ページに戻ります。
[フレームにドロップ]	このコマンドは、ColdFusion Builder ではサポートされていません。
[ステップ・フィルター/ステップ・デバッグの使用]	すべてのステップ関数でステップフィルタを適用します。

## 編集ビュー

編集ビューにはソースのタブが並んで表示されます。各タブは、開いているソースファイルに対応します。

## [サーバー]ビュー

[サーバー]ビューは、ColdFusion Builder からサーバーの起動、停止、管理を行う場合に使用します。また、[サーバー]ビューからは、ColdFusion サーバーモニタと ColdFusion Server Administrator も起動できます。



[サーバー]ビューのツールバー

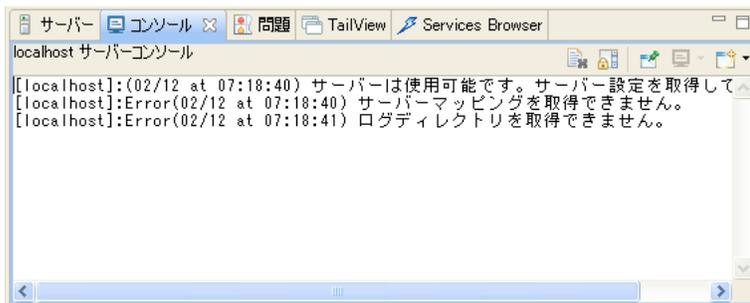
[サーバー]ビューのツールバーには、(左から右の順に)次のボタンがあります。これらのボタンを使用すると、サーバーを管理できます。

ボタン/コマンド	説明
[サーバーの追加]	サーバーを追加します。
[サーバーを起動]	サーバーを起動します。
[サーバーを再起動]	サーバーを停止して起動します。
[サーバーを一時停止]	サーバーを一時停止します。
[サーバーを停止]	サーバーを停止します。
[コンソール/シェルを開く]	新しいコンソールを開き、他のサーバーを選択するためのポップアップメニューを表示します。
[ログを開く]	[TailView] ビューでサーバーログを開きます。

サーバーの管理方法と [サーバー]ビューの使用方法については、54 ページの「[\[サーバー\]ビューの使用](#)」を参照してください。

## [コンソール]ビュー

[コンソール]ビューには、CFML コードに含まれる trace ステートメントの出力と、デバッグ自体からのフィードバック (ステータス、警告、エラーなど) が表示されます。



[コンソール]ビュー

[コンソール]ビューのツールバーには、(左から右の順に) 次のボタンがあります。これらのボタンを使用すると、[コンソール]ビューで trace ステートメントを表示して管理できます。

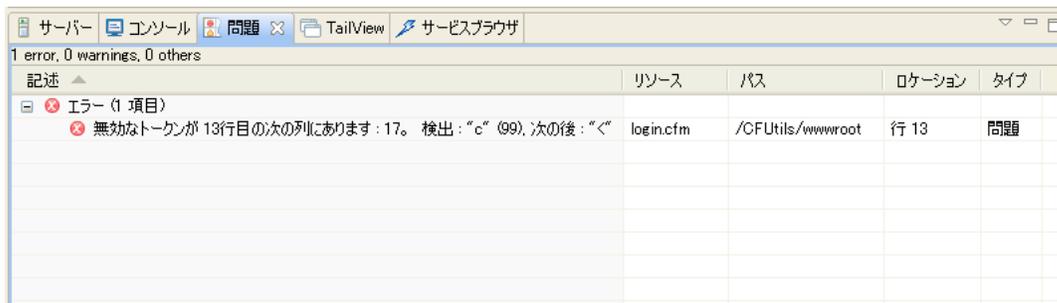


[コンソール]ツールバー

ボタン/コマンド	説明
[コンソールのクリア]	[コンソール]ビューの内容をすべて消去します。
[スクロールロック]	[コンソール]ビューがスクロールしないように固定します。
[コンソールのピン留め]	他のプロセスが選択されたときに、コンソールの内容が更新されないようにします。
[選択されたコンソールの表示]	選択されたコンソールを表示します。
[コンソールを開く]	新しいコンソールを開き、他のコンソールビューを選択するためのポップアップメニューを表示します。

## [問題]ビュー

コードを入力するときに、ColdFusion Builder コンパイラによってシンタックスエラーや他のコンパイルエラーが検出されると、[問題]ビューにそれらのエラーが表示されます。[問題]ビューには、ColdFusion Builder で開かれている CFML ファイルに関連するエラーのみが表示されます。



[問題]ビュー

## [TailView] ビュー

[TailView] ビューには、サーバーログが表示されます。詳細については、21 ページの「[TailView] ビュー」を参照してください。

## ワークベンチへのビューの追加

ワークベンチには、ColdFusion Builder のデフォルトの開発パースペクティブとデバッグパースペクティブに関連付けられているビューの他にも、さまざまなビューを追加できます。これらのビューをワークベンチに追加すると、アプリケーション開発プロセスの効率化を図ることができます。

❖ ワークベンチにビューを追加するには、[ ウィンドウ ]-[ ビューの表示 ]-[ その他 ]-[ 一般 ] を選択します。

これらオプションのビューはタイプで分類され、ワークベンチの異なる機能または特定の Eclipse プラグインに関連付けられます。

[ タスク ]、[ ブックマーク ]、[ 検索 ] などのワークベンチビューは、ColdFusion Builder でのアプリケーション開発に役立つビューです。

## ColdFusion Builder のエディタ

エディタはリソースタイプに関連付けられているため、ワークベンチでリソースを開くと、該当するエディタが開かれます。ワークベンチは、アプリケーションを開発するためのドキュメント中心型（およびプロジェクト中心型）の環境です。

CFML エディタは、CFML ファイルを編集するときに使用します。詳細については、33 ページの「[ColdFusion Builder でのコード編集](#)」を参照してください。

HTML エディタは HTML コードを編集するときに、JavaScript エディタは JavaScript コードを編集するときにそれぞれ使用します。CSS エディタではカスケードリングスタイルシートを表示して編集することができ、アプリケーションの視覚的要素にスタイルを適用できます。

[ カーソルにリンク ] 機能を使用すると、現在編集しているコードのタイプに応じてエディタを切り替えることができます。これらのエディタには、コードの色付け、コードアシスト、[ アウトライン ] ビューなど、コードのナビゲーションを容易にし、コードの有効性を維持するための機能が数多く含まれています。

CFML、HTML、JavaScript、CSS コードを入力する際には、コードの完成を支援するためのヒントが表示されます。この機能をコードアシストといいます。詳細については、34 ページの「[コードアシスト](#)」を参照してください。

## パースペクティブのカスタマイズ

パースペクティブのレイアウトはカスタマイズできます。つまり、パースペクティブ内で表示するビューとエディタは設定可能です。たとえば、特定のパースペクティブでは [ タスク ] ビューを表示し、別のパースペクティブでは表示しないように設定できます。また、パースペクティブに関連する次の項目も設定できます。

- [ ファイル ]-[ 新規 ] のサブメニュー項目
- [ ウィンドウ ]-[ パースペクティブ ]-[ その他 ] のサブメニュー項目
- [ ウィンドウ ]-[ ビューの表示 ]-[ その他 ] のサブメニュー項目
- ツールバーとメインメニューに表示するアクションセット（ボタンとオプション）

**注意：**プラグイン設定の ColdFusion Builder では、メニューの名前がやや異なります。

## カスタマイズされたパースペクティブの作成

- 1 変更するパースペクティブを開きます。
- 2 [ウィンドウ]-[パースペクティブのカスタマイズ] を選択します。
- 3 カスタマイズされたパースペクティブに追加する項目または削除する項目の種類に応じて、[ショートカット] タブまたは [コマンド] タブの [コマンドグループの可用性] タブを選択します。
- 4 選択したパースペクティブのメニューおよびツールバーに表示する要素のチェックボックスをオンにします。
- 5 [OK] をクリックします。
- 6 [ウィンドウ]-[パースペクティブの別名保管] を選択します。
- 7 [パースペクティブの別名保管] ダイアログボックスで、変更したパースペクティブの名前を入力し、[OK] をクリックします。

**注意:** 変更したパースペクティブを保存すると、変更したパースペクティブの名前が [ウィンドウ]-[パースペクティブを開く]-[その他] メニューに追加されます。

## カスタマイズされたパースペクティブの削除

削除可能なパースペクティブは、独自に作成したもののみです。ワークベンチに最初から付属するパースペクティブは削除できません。

- 1 ColdFusion Builder で [ウィンドウ]-[設定] を選択します。
- 2 [設定] ダイアログボックスの左側にツリービュー構造が表示されます。
- 3 ツリービュー構造で、[一般]-[パースペクティブ] を選択します。
- 4 [選択可能なパースペクティブ] で、削除するパースペクティブを選択します。
- 5 [削除] をクリックし、確認ダイアログで [はい] をクリックします。
- 6 [OK] をクリックします。

## カスタマイズされたパースペクティブのリセット

変更済みのパースペクティブは、初期のレイアウトに戻すことができます。

- 1 ColdFusion Builder で [ウィンドウ]-[設定] を選択します。
- 2 [設定] ダイアログボックスの左側にツリービュー構造が表示されます。
- 3 ツリービュー構造で、[一般]-[パースペクティブ] を選択します。
- 4 [選択可能なパースペクティブ] で、リセットするパースペクティブを選択します。
- 5 [リセット] をクリックし、[OK] をクリックします。

## 第4章：ColdFusion Builder でのコード編集

### CFML エディタ

ColdFusion Builder には、CFM ファイル、CFC ファイル、HTML ファイル、JavaScript ファイル、および CSS ファイルに対応する多彩なコード編集機能を備えた CFML エディタが組み込まれています。CFML エディタは、コード補完や効率的なコードナビゲーションなどの機能を提供することで、コードの記述を支援します。CFML エディタでは、さまざまな色やフォントを使用してワークスペース内のコードを表示できます。

開発環境をカスタマイズするには、エディタのプロファイルと設定を指定します。

### エディタプロファイル

エディタプロファイルを使用すると、次のエディタ設定を1つのプロファイルにグループ化して保存できます。

- コードアシスト
- カラー
- キー
- アウトライン
- シンタックスチェック
- 入力

エディタプロファイルは、開発のニーズに合わせて複数のエディタ設定を使い分ける場合に便利です。たとえば、コードアシスト、コードの色付け、キーボードショートカットのなどの設定を選択し、1つのエディタプロファイルに保存できます。エディタプロファイルを選択することで、これらすべての設定を同時に適用できます。

エディタプロファイルを作成するには、次のようにします。

- 1 保存する個々のエディタ設定を指定します。
- 2 [ウィンドウ]メニューから[設定]を選択します。
- 3 ツリービューで、[ColdFusion]-[エディタプロファイル]を選択し、[プロファイルの新規作成]をクリックします。
- 4 プロファイルの名前を入力します。設定した現在のエディタ設定値が、このプロファイルに保存されます。

デフォルト設定を変更し、変更した設定を新しいプロファイルとして保存することもできます。プロファイルはインポートまたはエクスポートすることもできます。

ColdFusion Builder にはデフォルトで3つのエディタプロファイルが用意されています。

- [デフォルト] - ColdFusion Builder のデフォルトのエディタ設定を適用します。
- [Dreamweaver] - Dreamweaver エディタ設定に類似したエディタ設定を適用します。
- [CFEclipse] - CFEclipse エディタ設定に類似したエディタ設定を適用します。

エディタプロファイルを選択するには、[エディタプロファイル]ダイアログボックスの[アクティブなプロファイル]ドロップダウンリストをクリックし、プロファイルを選択します。

## エディタ設定

- 1 [ウィンドウ]メニューから[設定]を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]を選択します。

作成する CFM ファイルおよび CFC ファイルのデフォルトのファイル名を指定できます。ファイル内の末尾の空白を削除するには、[ファイルを保存する前にスペースを削除]を選択します。

また、すべての CFM ファイルおよび CFC ファイルにデフォルトで挿入する任意のテキスト (ボイラープレートコード) も指定できます。たとえば、自分が作成したすべての CFC ファイルに著作権情報を追加する必要があるとします。その場合は、[新規ファイルの設定]の[CFC]タブで、著作権のテキストを指定できます。

設定可能なエディタ設定には、コードアシスト、コードの色付け、キーボードショートカット、アウトラインビュー、シンタックスチェック、入力の設定があります。

### ColdFusion Builder のパフォーマンスの最適化

設定で選択するオプションは、ColdFusion Builder のパフォーマンスに影響を与えます。プリセットのエディタ設定を適用すると、エディタのパフォーマンスを改善できます。プリセットの設定を適用すると、現在の設定は完全に失われます。現在の設定を保存するには、設定をエディタプロファイルとして保存します。

プリセットのエディタ設定を適用するには、[エディタ設定を最適化]をクリックします。次の設定が適用されます。

#### コードアシスト

次のオプションが選択解除されます。

- 入力時にコードアシストを自動表示
- 入力時に SQL コードアシストを自動表示
- 候補が 1 つだけの場合は自動的に挿入
- 終了タグを自動挿入

次のオプションが選択されます。

- 選択項目を挿入した後に空白を追加
- 等号 (=) を自動挿入
- 属性を自動的に引用符で囲む

#### 入力

- [自動挿入の有効化]が選択されます。
- [次の文字に対応する文字を自動挿入]のオプションがすべて選択解除されます。
- [タグの自動終了]が[なし]に設定されます。

## コードアシスト

**注意:** コードアシストはコンテンツアシストとも呼ばれ、これらの用語は同じ意味で使用されています。

コードアシストは、コードの完成を支援する機能です。入力するコードに応じて、コードの完成を支援するヒントが表示されます。CFML エディタでコードを入力する際には、コードアシストにより、有効な CFML タグ、パラメータ、属性のリストが表示されます。これらの項目は、ポップアップメニューに表示されます。CFML コード内に HTML、JavaScript、または CSS コンテンツを記述する場合は、コードアシストにより、それらのコードを完成させるためのヒントも表示されます。コード完成ヒントを CFML エディタに挿入するには、ダブルクリックするか Enter キーを押します。

コードアシストは、スクリプトベースのシンタックスでも使用できます。たとえば、スクリプトシンタックスの関数やコンポーネントのコードヒントも表示されます。ColdFusion プロジェクトに Ajax ライブラリをインポートするときは、JavaScript コードに関してもコードアシストを使用できます。

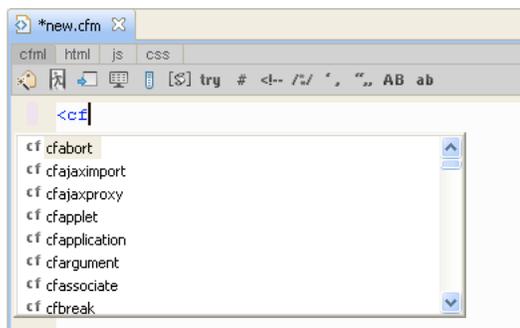
## コードアシストの使用

コードヒントは、フレームワークまたは言語 (CFML、HTML、JavaScript、および CSS) によって現在のコード式を完成させるためのオプションが提示されるときに毎回表示されます。たとえば、CFML タグの内容を入力する場合は、そのタグで指定できるすべての属性のリストが表示されます。

- 1 CFML ページでは、次のように入力することで CFML タグの入力を開始します。

```
<cf
```

関連するコードヒントが次のように表示されます。



コードヒント

- 2 コードヒントのリスト内を移動するには、上向き矢印 (↑) キーまたは下向き矢印 (↓) キーを使用します。
- 3 コードヒントを選択し、Enter キーを押します。選択したタグがエディタに追加されます。
- 4 コードの入力を続けると、追加のコードヒントが表示されます。コード行の入力中に Ctrl + スペースバーを押して、コードヒントを表示することもできます。

CFML エディタのスマートタグアシスト機能では、各タグで既に入力されているタグ属性が確認されます。既に入力されているタグ属性は、ヒントとして提示される属性のリストには表示されません。

## CFM ページに関するコードアシスト

- CFM ファイルで <cfinvoke> タグを使用するときに、メソッドとコンポーネントのリストが表示されます。
- 「<cf> と入力して Ctrl + スペースバーを押すと、事前定義された属性と値のドロップダウンリストが表示されます。
- createobject() または <cfobject> タグを使用してロードできるコンポーネント (CFC) のリストが表示されます。
- CFC 内で作成されたメソッドが表示されます。これは CFM 内で作成されたコンポーネントオブジェクトを使用して呼び出すことができます。すべての拡張済み CFC ファイルのメソッドのリストも表示されます。
- すべてのビルトイン関数とユーザー定義関数のドロップダウンリストが表示されます。

- 構造体、配列、クエリーなどの変数のリストが表示されます。これらの変数は、ページ内で宣言されています。これらの変数は、属性値のコードアシストとしても表示されます。
- <cfquery> や queryNew() を使用して作成されたすべてのクエリーが表示されます。レコードセットを表示するには、<cfoutput query=""> と入力して Enter キーを押します。または、<クエリー名.> を使用します (つまり、最後にドットがついたクエリー名を使用します)。
- <cfinclude> タグを使用して現在の CFM ページに取り込まれた、別の CFM ページの関数、変数、タグ、クエリーも表示されます。
- CFM タグの属性の入力値として指定するファイルを参照して選択できます。入力値としてファイルを指定する必要がある CFM タグには、<cfinclude>、<cfimage>、<cfdirectory>、<cfpdf>、<cffile>、<cfzip>、<cfspreadsheet>、<cfcollection> などがあります。

## CFC ページに関するコードアシスト

- createobject() を使用してロードできるコンポーネント (CFC) のリストが表示されます。
- <cfobject> タグを使用してロードできるコンポーネント (CFC) のリストが表示されます。
- CFC 内で作成されたメソッドが表示されます。これは CFM 内で作成されたコンポーネントオブジェクトを使用して呼び出すことができます。

例:

- <cfset obj1 = createobject("component", "c1")>  
<cfset x = obj1.method1()>
- <cfset obj2 = createobject("component", "c2").init()>
- <cfset obj3 = new "c3"().method3()>
- <cfinvoke component="c4" method="method4">
- すべての拡張済み CFC ファイルのメソッドのリストが表示されます。"extends" および "implements" キーワード (属性) に関しては、使用可能なコンポーネントとインターフェイスが表示されます。
- 現在の CFC ファイル内で拡張できるコンポーネントのリストが表示されます。

**注意:** コードアシストは、コンポーネントプロパティに追加されたコメントに対してはサポートされません。たとえば、コードアシストは、数値型の accountID プロパティに追加された `/** @ -- code assist is not supported here --*/` のようなコメントに対してはサポートされません。

- implement キーワードとともに CFC で使用できるインターフェイスのリストが表示されます。
- CFM タグや <cfif fileexists()> などの関数の属性の入力値として、ファイルを参照して選択できます。

入力値としてファイルを指定する必要がある CFM タグには、<cfinclude>、<cfimage>、<cfdirectory>、<cfpdf>、<cffile>、<cfzip>、<cfspreadsheet>、<cfcollection> などがあります。

## スコープ付き変数に関するコードアシスト (変数マッピングの使用)

多くの ColdFusion フレームワークでは、アプリケーションの起動時に CFC が作成され、これらの CFC がスコープ変数として保存されます。このような CFC に関するコードアシストを利用するには、ColdFusion Builder がコンテンツとデータ型 (CFC の完全修飾名) を識別できるようにする必要があります。

スコープ付き変数に格納された CFC に関連するコードアシストを有効にするには、変数名とそれに対応する CFC のタイプに関するプロジェクトレベルのマッピングを定義します。ColdFusion Builder では、このマッピング情報に基づいて、マッピングされた変数の CFC メソッドのリストが表示されます。ただし、そのデータ型のオブジェクトは作成されません。たとえば、mycfc1 を com.adobe.mycfcs.cfc1 にマッピングするとします。mycfc1 と入力して Ctrl + スペースバーを押すと、com.adobe.mycfcs.cfc1 から利用できるすべてのメソッドが表示されます。

変数マッピングを設定するには、[プロパティ] ダイアログボックスで、マッピングする変数として mycfc1 を指定し、マッピング対象の値として com.adobe.mycfcs.cfc1 を指定します。詳細については、57 ページの「[変数マッピングの設定](#)」を参照してください。

## ColdFusion ORM に関するコードアシスト

ORM (Object Relational Mapping: オブジェクトリレーショナルマッピング) は、オブジェクトモデルを使用してマッピング方針を定義できるプログラミング手法です。ORM を使用すると、リレーショナルデータベースのデータの格納および取得を行います。

ColdFusion Builder では、次の ColdFusion ORM 設定に関してコードアシストが提供されます。

- entityLoad、entityNew、および entityFindByPK 関数内のエンティティ名。
- entityLoad、entityNew、および entityFindByPK 関数から返されるメソッドとプロパティ。
- エンティティ名とメソッドのコードハイパーリンク。
- application.cfc 内のデータソース。<cfset this.datasource = > など。
- cfcomponent タグのテーブル属性。

ColdFusion ORM の使用方法の詳細については、『ColdFusion アプリケーションの開発』の ColdFusion ORM を参照してください。

## "Application.cfm" ファイルおよび "Application.cfc" ファイルに関するコードアシスト

コードアシストは、"Application.cfm" ファイルと "Application.cfc" ファイルに対してもサポートされています。

"Application.cfm" はインクルード対象のファイルと見なされ、そのファイル内にあるすべての変数と関数に対してコードアシストがサポートされます。

"Application.cfc" の場合、コードアシストは、onApplicationStart で宣言されているスコープ付き変数に対してのみサポートされます。

## CFML エディタのコードアシストの設定

1 [ウィンドウ] メニューから [設定] を選択します。

2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]-[コードアシスト] を選択します。

- ここでは、コードの入力を支援するコードアシスト辞書のバージョンを選択できます。サポートされている辞書のバージョンは、ColdFusion 9、ColdFusion 8、および ColdFusion MX 7 です。
- [入力時にコードアシストを自動表示] チェックボックスを選択すると、入力時にコードアシストが自動的に表示されます。

**注意：**サイズの大きいファイルを編集するときは、[入力時にコードアシストを自動表示] チェックボックスを選択解除すると、エディタのパフォーマンスが向上します。その場合でも、コード行の入力中に Ctrl + スペースバーを押すか、任意に設定したキーボードショートカットキーを押すことで、コードヒントを表示できます。

- 自動的に属性を引用符で囲むように設定したり、終了タグを選択することもできます。

## CFC および UDF のコードハイパーリンク

Ctrl キーを押しながらマウスポインタを合わせると、コンポーネント名および UDF がハイパーリンクに変わります。このハイパーリンクをクリックすると、対応するコードが ColdFusion Builder で表示されます。コードハイパーリンクは次の文字列で使用できます。

- UDF: local、included、および cfc.udfName
- `<cfinclude template="">` で指定されているテンプレート
- `createobject()`、`<cfobject>`、`<cfinvoke>`、`new` キーワード、および `extends` 属性で指定されている CFC

## サーバーマッピングを使用した CFC 名の解決

CFC のサーバーマッピングを定義するには、ColdFusion Administrator を使用します。サーバーマッピングを定義すると、`extend` キーワードまたは `implement` キーワードを使用するときに、コードアシストで CFC が表示されるようになります。

ColdFusion サーバーマッピングでは、`cfobject` タグと `cfinvoke` タグ、または `createObject` や `new` のような関数でページにアクセスして、ドキュメントルート以外にある ColdFusion コンポーネントを検索できます。

これらのタグで、マッピングの論理パスから始まるパスを指定すると、マッピングのディレクトリパスを使用して CFC が検索されます。CFC のサーバーマッピングは ColdFusion Administrator を使用して定義できます。マッピングを定義すると、`extend` または `implement` キーワード、および `component` または `method` 属性を使用するときに、コードアシストで CFC が表示されるようになります。コードアシストで提示される CFC は、それぞれの完全修飾名を使用して解決されます。

サーバーから CFC 名を解決するためのマッピングを作成すると、サーバー設定がキャッシュに保存され、サーバーが稼動していないときでも使用できるようになります。サーバー設定は、ColdFusion Builder からサーバーを起動または更新する際に収集されます。

### サーバー設定の指定

- 1 [ ウィンドウ ] メニューから [ 設定 ] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[サーバーの設定] を選択します。
- 3 [サーバー設定の構築] チェックボックスを選択し、サーバー設定の収集動作に関するチェックボックスを適切に選択します。たとえば、ColdFusion Builder を起動するたびにサーバー設定を収集する場合は、[ColdFusion Builder が開始しました] チェックボックスを選択します。

## CFML 辞書

ColdFusion Builder には、CFML コードの入力を支援する辞書が組み込まれています。CFML 辞書は、ライブラリに含まれる各タグと関数に関する情報が記述された XML ファイルです。たとえば、CF9 辞書ファイル (cf9.xml) には、CF9 で使用可能なすべてのタグと関数に関する情報が含まれています。

ColdFusion Builder は、ColdFusion 9、ColdFusion 8、および ColdFusion MX7 の各バージョンの辞書をサポートしています。

辞書のバージョンを選択するには、次のようにします。

- 1 [ ウィンドウ ] メニューから [ 設定 ] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]-[コードアシスト] を選択します。
- 3 [コードアシスト辞書のバージョン] ドロップダウンリストを選択し、コードの入力を支援する辞書のバージョンを選択します。
- 4 カスタム辞書を含むすべての辞書をリロードするには、[辞書のリロード] をクリックします。

## カスタム CFML 辞書の作成

カスタム CFML 辞書 (XML ファイル) を作成し、ColdFusion Builder 内のコードアシストやタグウィザードでその辞書を使用することができます。

- 1 ColdFusion Builder のインストール先で、次の場所に移動します。

```
¥plugins¥com.adobe.ide.coldfusion.dictionary_XXX¥dictionary
```

- 2 Dictionary ディレクトリ内に "Custom" というフォルダを作成します。
- 3 カスタムタグとカスタム関数に関する情報を含む XML ファイルを作成します。

次の例に示すように、XML ファイルで、各タグに関する情報を <tag></tag> 要素内に記述し、すべての tag 要素を <dictionary></dictionary> 要素で囲みます。

```
<dictionary>
  <tags>
    !--
    cfabort
    showError = "error_message"
    -->
    <tag endtagrequired="false" name="cfabort" single="true"
xmlstyle="false">
<parameter name="showerror" required="false" type="String">
</parameter>
</tag>
</tags>
</dictionary>
```

カスタム CFML 辞書を作成するには、参考として cf9.xml ファイルまたは cf8.xml ファイルを参照してください。

## コードの色付け

ColdFusion Builder では、シンタックスに含まれる CFML タグ名、属性、属性値、キーワード、コメントなどの要素を、別々の色で強調表示できます。コードの色付けは、スクリプトスタイルのコーディングでもサポートされています。カラー設定をカスタマイズすると、Eclipse エディタのデフォルトのカラー設定を上書きできます。

### CFML エディタのカラー設定

- 1 [ウィンドウ] メニューから [設定] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]-[色] を選択します。
  - カラー設定は、COL ファイルとしてインポートおよびエクスポートできます。
  - トークンのカラー設定を指定できます。トークンには、言語設定、エラー、識別子、キーワード、リテラル、区切り記号などがあります。
  - コードの特定の範囲のカラー設定を指定できます。コード要素の始まりから終わりまでの範囲を指定し、その範囲を識別するための色を定義します。

### SQL エディタのカラー設定

- 1 [ウィンドウ] メニューから [設定] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]-[色] を選択します。[トークン] で、[SQL] にスクロールし、コメント、キーワード、およびテキストのカラー設定を選択します。

## SQL エディタ

CFML エディタには SQL エディタが組み込まれており、SQL ステートメントを編集、作成、または実行できます。

SQL エディタを使用するには、ColdFusion Builder でサーバーを設定し、そのサーバーが稼動している必要があります。SQL エディタはオフラインデータベースをサポートしていません。

SQL エディタは、次の SQL ステートメントに対して、コードアシストとコードの色付けをサポートします。

- Select
- Insert
- Update
- Delete

**注意：**cfquery タグ内にある SQL ステートメントの場合、コードアシストは CFML エディタ自体で使用できます。コード補完のヒントを表示するために SQL エディタを開く必要はありません。

### SQL エディタの使用

- 1 次のいずれかの操作を行います。
  - CFML エディタを右クリックし、[SQL エディタ] を選択します。
  - キーボードショートカット Ctrl + Alt + S キー (Windows の場合) または Command\_Alt + S キー (Mac OS の場合) を使用します。
- 2 SQL エディタで、[サーバー] ドロップダウンリストからサーバーを選択し、[データソース] ドロップダウンリストからデータベースを選択します。
- 3 SQL ステートメントを入力し、次の操作を行います。
  - [クエリーの実行] をクリックすると、SQL ステートメントの結果が [クエリ結果] タブに表示されます。SQL エディタで実行できるのは、SELECT ステートメントのみです。
  - [OK] をクリックすると、CFML エディタの現在のキャレット位置に SQL ステートメントが挿入されます。キャレットとは、次の文字が出現する位置を示す CFML エディタ内のマーカーです。
- 4 コードブロックを選択し、Shift + Enter キーを押して、SQL エディタから CFML エディタに SQL コードブロックを直接コピーすることもできます。
- 5 CFML エディタで SQL コードを編集するには、コードブロックを選択し、右クリックして [SQL エディタ] を選択します。選択したコードブロックが SQL エディタに表示されます。

基本的な SQL 構文と SQL ステートメントの作成の詳細については、『ColdFusion アプリケーションの開発』の SQL の使用を参照してください。

### SQL コードアシスト

SQL コードアシストは、SQL ステートメント内のデータベーステーブル名とフィールド名に関してのみ使用できます。

ColdFusion Builder の SQL コードアシスト機能はデフォルトでオンになっているので、SQL ステートメントに関するコードヒントは CFML エディタで自動的に表示されます。SQL コードアシストの自動表示をオフにするには、次のようにします。

- 1 [ウィンドウ] メニューから [設定] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]-[コードアシスト] を選択します。
- 3 [入力時に SQL コードアシストを自動表示] を選択解除します。

SQL コードアシストの自動表示がオフになっている場合でも、次のようにすると SQL ステートメント内でコードヒントを表示できます。

- 1 データベーステーブル名またはフィールド名を表示するには、SQL エディタに SQL ステートメントを入力し、Ctrl + スペースバーを押します。
- 2 上向き矢印 (↑) キーまたは下向き矢印 (↓) キーを使用して、テーブル名またはフィールド名のリスト内を移動します。
- 3 テーブル名またはフィールド名を選択し、Enter キーを押します。選択したテーブルまたはフィールドが SQL ステートメントに追加されます。

ColdFusion Builder には、<cfoutput> タグの query 属性のコードアシストが用意されています。たとえば、<cfoutput query=""> と入力して、二重引用符 ("" ) の間にキャレットを置いて、Ctrl + スペースバーを押すとします。そのファイルやインクルードファイルで作成されたクエリーのリストが表示されます。

式で使用されるクエリー名にも、コードアシストが用意されています。

たとえば、次のようなクエリーを作成したとします。

```
<cfquery datasource="dsn1" name="q1">
Select id, firstName, lastName from employee
</cfquery>
```

<cfset name = q1.> という式を入力して、Ctrl + スペースバーを押すと、その SQL クエリーで選択された列が ColdFusion Builder に表示されます。

## SQL コードの色付け

CFML エディタで SQL コードの色付けがサポートされるのは、cfquery タグ内のみです。cfquery タグ内の SQL キーワード、コメント、およびテキストの色付けを指定できます。色付けは、次のタイプの SQL コメントに対してのみサポートされます。

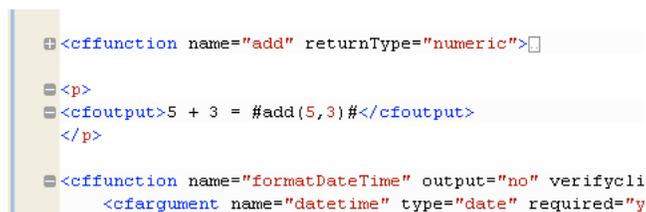
- /\* \*/
- --

SQL コードの色付けの設定については、39 ページの「[SQL エディタのカラー設定](#)」を参照してください。

## コードの折りたたみと展開

CFML エディタには、コード内をナビゲートするためにショートカットが数多く用意されています。これらのショートカットには、コードブロックの折りたたみと展開、コード定義のソースのオープン、タイプの参照とオープンなどがあります。複数のコードブロックを折りたたんだり展開することで、複雑なコードドキュメントのナビゲート、表示、管理が容易になります。ColdFusion Builder では、複数行にまたがるコードステートメントを折りたたんだり展開することを、コードの折りたたみまたは展開と呼びます。この機能は、CFM、CFC、HTML ファイルで使用できます。

- 1 CFML エディタで、左余白にある折りたたみアイコン (-) または展開アイコン (+) をクリックします。



```
⊖ <cffunction name="add" returnType="numeric">⊞
⊖ <p>
⊖ <cfoutput>5 + 3 = #add(5,3) #</cfoutput>
⊖ </p>
⊖ <cffunction name="formatDateTime" output="no" verifycli
    <cfargument name="datetime" type="date" required="y
```

コードブロックを折りたたむと、最初のコード行を除くすべての行が非表示になります。

```
<cffunction name="add" returnType="numeric">
  <cfargument name="x" type="numeric">
  <cfargument name="y" type="numeric">
  <cfreturn x+y>
</cffunction>

<cffunction name="formatDateTime" output="no" verifycli
  <cfargument name="datetime" type="date" required="y
  .....
```

展開アイコン (+) の上にマウスポインタを置くと、現在折りたたまれているコードがツールヒント内に表示されます。

```
<cffunction name="add" returnType="numeric">
  <cfargument name="x" type="numeric">
  <cfargument name="y" type="numeric">
  <cfreturn x+y>
</cffunction>

<p>
<cfoutput>5 + 3 = #add(5,3)#</cfoutput>
</p>

<cffunction name="formatDateTime" output="no" verifycli
  <cfargument name="datetime" type="date" required="y
```

コードブロックを展開すると、コードが見えるようになります。

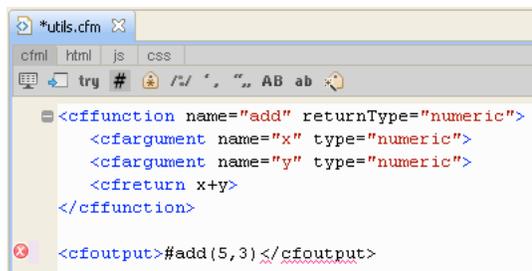
- デフォルトでは、コードのすべての行が展開されます。最初のコード行を除くすべての行を折りたたむには、エディタの左余白を右クリックします。次に [ 折りたたみ ] - [ すべてを折りたたむ ] を選択します。

## シンタックスのチェックと強調表示

CFM ファイル内のタグまたはコードブラケットを選択すると、そのタグまたはブラケットに対応するペアが自動的に強調表示されます。

有効な CFML コードとして認識されないコードを入力すると、次のような方法で通知されます。

- 次のように、該当するコード行の横に × 印が表示されます。

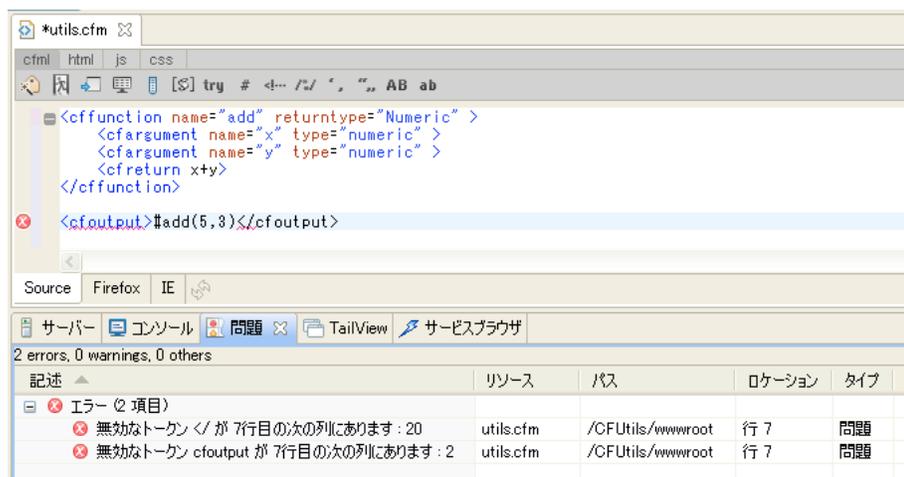


```
*utils.cfm
cfml html js css
try # /% ' " ,, AB ab
<cffunction name="add" returnType="numeric">
  <cfargument name="x" type="numeric">
  <cfargument name="y" type="numeric">
  <cfreturn x+y>
</cffunction>
<cfoutput>#add(5,3)</cfoutput>
```

強調表示されたシンタックスの問題

× 印にマウスポインタを合わせると、エラーの詳細が表示されます。

- 次のように、[ 問題 ] ビューにエラーが一覧表示されます。



[問題] ビューに表示されたシンタックスの問題

各メッセージには、エラーの簡単な説明、エラーが発生したファイルとフォルダ、ファイル内の行番号が含まれます。エラーをダブルクリックすると、エディタでファイルが開かれ、該当するコード行が強調表示されます。

**注意:** 表示される情報は、現在開かれている CFM ファイルのシンタックスエラーのみです。CFM ファイルを閉じると、そのファイルのエラーは表示されなくなります。

エラーの性質と重大度によっては、エラーを訂正するまでアプリケーションを適切に実行できない可能性があります。

### シンタックスチェックの設定

- 1 [ウィンドウ] メニューから [設定] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]-[シンタックスチェック] を選択します。
- 3 シンタックスチェックは、デフォルトでオンになっています。シンタックスチェックをオフにするには、[シンタックスチェックの有効化] を選択解除します。
- 4 ファイルを開くときまたは保存するときのみシンタックスエラーを表示するには、[ファイル保存時のみシンタックスエラーを表示] を選択します。ファイルの編集中は、ファイル内のすべてのエラーが [問題] ビューから削除されます。

## コードリファクタリング

コードリファクタリングとは、全体的な結果を変更せずにプログラムのソースコードを改善するプロセスです。一般に、コードリファクタリングを行うと、コードの可読性と保守性が向上します。

ColdFusion Builder では、CFC、CFM、UDF の名前の変更、検索、プレビューなど、さまざまなリファクタリング手法がプロジェクトレベルおよびワークスペースレベルでサポートされています。

### CFC ファイル名または CFM ファイル名のリファクタリング

CFC または CFM の名前を変更する場合は、その CFM または CFC へのすべての参照を含め、CFM または CFC の有効なインスタンスをすべてリファクタリングできます。

- 1 [ナビゲーター] ペインでファイルを右クリックします。
- 2 [リファクタリング]-[名前変更] を選択します。

- 3 新しい名前を入力し、変更結果をプレビューします。変更結果のプレビュー時には、選択されたインスタンスを確認して、名前を変更したくないインスタンスの選択を解除できます。
- 4 ワークスペース内のすべてのプロジェクトにわたって CFC または CFM の名前を変更するには、[ワークスペース内のすべてのプロジェクトを更新] を選択します。

**注意:** HTTP または Web サーバーの URL に含まれる CFC または CFM の名前を変更するには、サーバーが登録済みで稼動している必要があります。

リファクタリングを取り消すには、キーボードショートカット Ctrl + Z キー (Windows の場合) または Command + Z (Mac の場合) を使用します。

## CFC または CFM ファイル内の UDF のリファクタリング

- 1 CFM または CFC ファイルをエディタで開きます。
- 2 関数名をポイントし、右クリックして [リファクタリング]-[名前変更] を選択します。
- 3 UDF 名を指定して、[OK] をクリックします。

**注意:** 現在、関数名の変更は、メイン関数宣言内でのみサポートされており、関数呼び出しから関数名を変更することはできません。

### メモおよび注意事項

- 固有の Eclipse エディタには [Move] オプションを使用してファイルを移動する機能がありますが、CFM および CFC ファイルの場所は変更できません。
- Eclipse ではフォルダ名を変更するオプションがありますが、CFML エディタではフォルダ名の変更をサポートしていません。

## メソッドのリファクタリング

変数マッピングを作成することで、アプリケーション変数用に作成されたメソッドをリファクタリングできます。

たとえば、次のように、CFC オブジェクトに対してアプリケーション変数が定義されているとします。

```
<cfset Application.appVar = createObject("component", "abc.AppCFM.a")>
```

また、次のように呼び出される CFC 関数 Method() があるとします。

```
<cfset Application.appVar.Method()>
```

Method() 関数をリファクタリングするには、次のようにアプリケーション変数 Application.appVar の変数マッピングを作成します。

- 1 [ナビゲーター] ビューでプロジェクトを右クリックし、[プロパティ] を選択します。
- 2 [プロパティ] ダイアログボックスで [ColdFusion 変数マッピング] を選択します。
- 3 [新規] をクリックし、[マッピングの追加] ダイアログボックスで次のように入力します。
  - 変数名: Application.appVar
  - マッピング対象: abc.AppCFM.a

## 参照検索

CFC、CFM、UDF の検索は、プロジェクトレベルまたはワークスペースレベルで行えます。

- 1 [ナビゲーター] ペインでプロジェクトを右クリックします。

- 2 [参照] をクリックして、[プロジェクト] または [ワークスペース] を選択します。ColdFusion Builder の [検索] ウィンドウに結果が表示されます。

参照はプロジェクトレベルまたはワークスペースレベルで検索できるだけでなく、CFML コードエディタでの作業中に検索することもできます。ファイル参照を検索するには、CFML エディタ内を右クリックし、[参照]-[ファイル] を選択します。CFML エディタ内からプロジェクトレベルまたはワークスペースレベルで参照を検索することもできます。

## タグウィザード

タグウィザードを使用すると、CFML に精通していなくても、タグや属性を簡単に追加できます。ColdFusion Builder でタグウィザードを使用するには、次のようにします。

- 1 [CFML] タブが開いていない場合は、[ソース] 領域で [CFML] タブをクリックします。
- 2 CFML ツールバーのタグウィザードアイコンをクリックします。Ctrl + T キーを押してタグウィザードを開くこともできます。
- 3 入力するタグがわからない場合は、タグウィザードのダイアログボックスを使用してタグを検索します。
- 4 タグ属性の値を指定します。タグウィザードには、各種のアクション属性に固有の値も表示されます。
- 5 [OK] をクリックすると、CFML ページにタグが追加されます。

**注意:** タグウィザードは状況依存型のウィザードです。タグの上にカーソルを移動して Ctrl + T キーを押すとタグの編集が可能になり、そのタグ用のタグウィザードが表示され、タグ属性を指定または変更できます。

## 入力設定

CFML タグを入力する際には、入力設定をカスタマイズできます。次のような設定項目を指定できます。

- 終了タグおよび一致する文字 ( 引用符や中括弧など ) の自動挿入
- Enter キーを押したときの子タグの自動挿入
- 対応するタグが変更されたときの自動変更
- 属性を引用符で囲むときの一重引用符および二重引用符の自動提示

### CFML エディタの入力設定

デフォルトでは、自動挿入が有効になっています。一重引用符、二重引用符、中括弧、および「#」記号に対応する文字を自動挿入できます。

自動挿入をオフにするには、次のようにします。

- 1 [ウィンドウ] メニューから [設定] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[エディタプロファイル]-[エディタ]-[入力] を選択します。
- 3 [自動挿入の有効化] を選択解除します。

## CFML エディタのキーボードショートカット

CFML エディタでは、デフォルトのキーボードショートカットを変更して、独自のキーボードショートカットを指定できます。デフォルトでキーボードショートカットでは、Eclipse のキーバインディングが使用されます。

### CFML エディタのキーボードショートカットの設定

キーバインディングを変更するには、次のようにします。

- 1 [ ウィンドウ ] メニューから [ 設定 ] を選択します。
- 2 ツリービュー構造で、[ ColdFusion ] - [ エディタプロファイル ] - [ エディタ ] - [ キー ] を選択します。
- 3 変更するキーバインディングを選択します。
- 4 [ キーバインディングの編集 ] で、キーバインディングをクリックして強調表示します。
- 5 設定するキーの組み合わせを押します。たとえば、Ctrl + Shift + 1 と入力するには、Ctrl キーと Shift キーを押した状態で 1 キーを押します。キーの間のプラス記号 (+) は、キーを連続して押す必要があることを示しています。

## 第 5 章：サーバーの管理

ColdFusion Builder では、ColdFusion サーバーインスタンスを作成し、ローカルサーバーまたはリモートサーバー上に存在するデータソースを管理できます。ColdFusion の組み込み Web サーバーを使用してサーバーインスタンスを作成することも、Microsoft Internet Information Server (IIS) や Apache HTTP Server などの外部 Web サーバーを使用することも可能です。

ColdFusion Builder の包括的なサーバー管理機能を使用すると、1 つの [サーバー] ビューから複数のサーバーを管理し、ColdFusion Administrator やサーバーモニタにアクセスできます。[サーバー] ビューの使用の詳細については、54 ページの「[\[サーバー\] ビューの使用](#)」を参照してください。

### Web サーバーの用語について

サーバーをセットアップして管理する前に、以下のトピックに目を通して、マニュアルで使用されている Web サーバーの用語と概念について理解しておいてください。

Web サーバーの詳細と、Web サーバーを ColdFusion で使用できるように設定する方法については、『ColdFusion 設定と管理』の Web サーバーの管理を参照してください。

#### ドキュメントルート

ドキュメントルートという用語は、Web サーバーから Web ページを提供するときに使用されるファイルシステムディレクトリを指します。この用語はサーバーの種類によって異なる場合がありますが、大部分の Web サーバーには同じ概念が当てはまります。

#### URL 接頭辞

URL 接頭辞は、ローカルファイルシステムのリソースを URL にマップします。

ColdFusion Builder では、Web ルートまたはドキュメントルートの外にあるプロジェクトをプレビューまたはデバッグする場合に、URL 接頭辞を使用します。

URL 接頭辞は、サーバーの作成時に指定することも、既存サーバーの設定を編集して指定することもできます。詳細については、49 ページの「[URL 接頭辞](#)」を参照してください。

既存のプロジェクトまたはフォルダに対する URL 接頭辞を指定することもできます。詳細については、59 ページの「[URL 接頭辞の設定](#)」を参照してください。

#### 使用例

"Project1" という名前のプロジェクトがあります。Project1 は、server1 に対して設定されています。server1 のドキュメントルートは C:¥server1¥MyDocs にあり、URL は http://www.example1.com です。Project1 の中には "xyz" という名前のリンクフォルダがあります。フォルダ "xyz" は server2 のドキュメントルート C:¥server2¥MyDocs にリンクされており、このフォルダにアクセスするための URL は http://www.example2.com です。ColdFusion Builder で "xyz" リンクフォルダ内のファイルをすべてプレビューする必要があるとします。

この場合、"xyz" リンクフォルダ内のファイルをプレビューするには、URL 接頭辞を作成します。URL 接頭辞を作成するには、次の情報を指定します。

- 絶対パス : C:¥server2¥MyDocs¥

- "xyz" フォルダにアクセスするための URL : <http://www.example2.com>

## 仮想ホスト

仮想ホストという用語は、単一の Web サーバー上で複数の Web サイト (ドメイン名) をホストする方法を指します。複数の Web サイトは、識別可能なホスト名によって区別されます。たとえば、単一の IP アドレスで [www.example1.com](http://www.example1.com)、[www.example2.com](http://www.example2.com)、および [www.example3.com](http://www.example3.com) という Web サイトを運用できます。

ColdFusion Builder の場合は、単一の ColdFusion サーバーを使用して、IIS や Apache などの外部 Web サーバー上の仮想ホストとして設定された複数の Web サイトを運用できます。

Web サーバーで仮想ホストを設定する方法については、各 Web サーバーのマニュアルを参照してください。

プロジェクトを仮想ホストに関連付けると、ColdFusion Builder の機能 (プレビュー、デバッグ、コンテンツアシスト、拡張機能の作成など) が、プロジェクト内のすべてのフォルダとサブフォルダに適用されます。

## 仮想ディレクトリ

仮想ディレクトリという用語は、ドキュメントルートに物理的には含まれていないが、サーバー URL を通じてアクセス可能なフォルダを指します。仮想ディレクトリを作成するには、フォルダのパスのエイリアスを URL で指定します。エイリアス名は、フォルダ内のリソースにアクセスするために使用されます。

Web サイト ([www.example.com](http://www.example.com)) のドキュメントルートが `c:\xyz\docs` であり、Web サイトにコンテンツを提供するフォルダが `d:\abc\content` にあるとします。その場合は、このフォルダのエイリアスを `content` という名前で定義します。これにより、<http://www.example.com/content/> という URL を使用して、Web サイトにアクセスできるようになります。

# ColdFusion サーバーの作成

ColdFusion Builder ワークスペース内に作成したプロジェクトやアプリケーションをテストするには、ColdFusion サーバーをセットアップします。この ColdFusion サーバーは、本稼動サーバーにデプロイする前にアプリケーションをテストして実行するための開発サーバーとして使用できます。ローカルとリモート両方の ColdFusion サーバーを作成できます。

## ローカルサーバーの作成

[サーバー] ビューで、次のいずれかの操作を行います。

- 右クリックして [サーバーの追加] を選択します。
-  をクリックします。

新規 ColdFusion サーバーセットアップウィザードで次の情報を入力します。

### 一般設定

- [サーバー名]: ColdFusion サーバーの名前。
- [説明]: (オプション) サーバーの説明。
- [アプリケーションサーバー]: ドロップダウンリストを選択して [JRun] を選択するか、[その他] を選択して Jrun 以外のサーバーを設定します。

**注意:** ColdFusion Builder 内で JRun 以外のサーバーを起動、停止、または再起動することはできません。

- [ホスト名]: ColdFusion サーバーホストの名前。たとえば、localhost や 127.0.0.1 のように入力します。

- [ ローカル ] を選択します。

## その他の設定

- [WebServer ポート]: 設定する ColdFusion サーバーインスタンスのポート番号を指定します。ColdFusion サーバーのデフォルトポート番号は 8500 です。
- コンテキストルート: (JRun サーバーを使用する J2EE 設定のみに適用) コンテキストルートを入力します。J2EE 環境では、1つのサーバーインスタンス上で、独立した複数の Web アプリケーションを実行できます。したがって、サーバー上で実行される J2EE Web アプリケーションのルートは、それぞれ固有のベース URL に設定されます。これが、コンテキストルート (またはコンテキストパス) と呼ばれるものです。
- [アプリケーションサーバー名]: (JRun サーバーを使用する J2EE 設定のみに適用) ColdFusion をデプロイする JRun サーバーの名前。
- [RDS ユーザー名]: (オプション) RDS を使用している場合は、RDS ユーザー名を指定します。
- [RDS パスワード]: (オプション) RDS パスワードを指定します。

**注意:** RDS パスワードの設定は ColdFusion Administrator で行います。同じく ColdFusion Administrator で管理される ColdFusion Administrator パスワードと RDS パスワードを混同しないでください。

- ColdFusion Builder で SSL のサポートを有効にするには、[SSL の有効化] を選択します。Server Manager に登録されているサーバーとの通信には SSL を使用できます。
- ColdFusion Builder を起動または終了するたびに ColdFusion サーバーが自動的に起動または停止されるようにするには、[自動起動] および [自動停止] を選択します。

[次へ] をクリックします。

## ローカルサーバーの設定

[ローカルサーバーの設定] タブを選択し、必要に応じて次のローカルサーバー設定を指定します。

- [サーバー名]: (JRun サーバーのみに適用) ColdFusion サーバーのホームディレクトリを参照して選択します。たとえば、スタンドアロンの ColdFusion サーバーインスタンスの場合は C:\ColdFusion9 です。マルチサーバーインスタンスの場合、サーバーホームはアプリケーションサーバーのホームディレクトリです。たとえば、JRun-J2EE またはマルチサーバー設定の場合は C:\JRun4 です。
- [ドキュメントルート]: Web ルートの場所を参照して選択します。ColdFusion を IIS などの Web サーバーとともに設定する場合は、Web サーバーのドキュメントルート (c:\inetpub\wwwroot など) を選択します。  
この設定は、ColdFusion Builder でプレビュー、デバッグ、および CFC 名の解決を行うために必要です。
- [バージョン]: (Windows で動作する JRun サーバーのみに適用) [バージョン] ドロップダウンリストから ColdFusion サーバーのバージョンを選択します。
- [Windows サービス]: (Windows で動作する JRun サーバーのみに適用) [Windows サービス] オプションは、スタンドアロン設定およびマルチサーバー設定の場合にのみ使用でき、J2EE 設定では使用できません。Windows サービスを使用して ColdFusion サーバーの起動と停止を行う場合は、[Windows のサービスを使用してサーバーを起動または停止する] を選択します。

## URL 接頭辞

(オプション)

**注意:** 必ずしも、サーバーの作成時に URL を指定する必要はありません。サーバーを作成した後でも、[サーバー] ビューでサーバー設定を編集することで、URL 接頭辞を指定できます。既存のプロジェクトまたはフォルダに対する URL 接頭辞を指定することもできます。詳細については、59 ページの「[URL 接頭辞の設定](#)」を参照してください。

[URL 接頭辞] タブを選択して、次の情報を入力します。

- 1 [ローカルパス]: ローカルファイルシステムリソースのパスを参照するか、手動で入力します。
- 2 [URL 接頭辞]: URL 接頭辞を入力します。
- 3 [追加] をクリックします。

## 仮想ホストの設定

(オプション)

**注意:** ColdFusion Builder で仮想ホストまたは仮想ディレクトリを指定する場合は、設定済みの Web サーバーで、対応する設定を指定する必要があります。ColdFusion Builder ではこれらの設定は検証されません。そのため、Web サーバーの設定が ColdFusion Builder で指定した内容と異なっても、ColdFusion Builder はエラーを生成しません。

仮想ホストを設定するには、[仮想ホストの設定] タブを選択して、次のようにします。

- 1 [新規] をクリックし、[仮想ホストの設定] セクションで次の情報を入力します。
  - a [名前]: 仮想ホストの名前を指定します。(vh1 など)  
**注意:** 任意の名前を指定できます。必ずしも Web サーバーで指定した名前でなくてもかまいません。
  - b [ホスト名]: IIS または Apache Web サーバーの設定で指定したとおりの仮想ホスト名。(www.example1.com など)  
ColdFusion Builder で仮想ホストを作成した場合、その仮想ホストでは <サーバー名>-<仮想ホスト名> というネーミング規則が使用されます。プロジェクトプロパティの [サーバー] ドロップダウンリストには、このネーミング規則を使用した仮想ホスト名が表示されます。たとえば、ColdFusion サーバー (localhost) で "vh1" という名前の仮想ホストを作成した場合、仮想ホストを識別するために ColdFusion Builder で使用されるネーミング規則は、"localhost-vh1" です。
  - c [ポート]: Web サーバーの仮想ホストに割り当てられたポート。
  - d [タイプ]: ドロップダウンリストから [HTTP] または [HTTPS] を選択します。
  - e [ドキュメントルート]: 仮想ホストのホームディレクトリの場所を参照するか、手動で入力します。  
たとえば、Web サイト www.example.com が Apache Web サーバー上の "C:¥abc" ディレクトリにマップされている場合は、ホームディレクトリとして「C:¥abc」と入力します。
- 2 (オプション) 仮想ディレクトリを作成するには、[仮想ディレクトリ] をクリックして次の情報を入力します。
  - a [エイリアス]: フォルダパスのエイリアスを指定します。
  - b [場所]: エイリアスを指定するフォルダパスを参照するか、手動で入力します。  
たとえば、Web サイト (www.example.com) のドキュメントルートが c:¥abc であり、d:¥xyz¥images でアクセス可能なフォルダ内のイメージを使用する必要があるとします。その場合は、フォルダパス "d:¥xyz¥images" に対して "images" という名前のエイリアスを定義します。仮想ディレクトリの詳細については、48 ページの「[仮想ディレクトリ](#)」を参照してください。
  - c [追加] をクリックします。
  - d [OK] をクリックすると、仮想ディレクトリが [仮想ディレクトリの設定] テーブルに追加されます。
- 3 [適用] をクリックします。仮想ホストが [仮想ホスト] テーブルに追加されます。

仮想ホストの設定を変更するには、[仮想ホスト] テーブルから仮想ホストを選択し、設定を変更して、[更新] をクリックします。

仮想ホストの詳細と、ColdFusion Builder での有用性については、48 ページの「[仮想ホスト](#)」を参照してください。

## 拡張機能のインストール

ColdFusion Builder に付属する拡張機能をインストールするには、[ 拡張機能のインストール ] を選択します。

- 1 ColdFusion Web ルートの場所を参照して選択します。
- 2 拡張機能をインストールする Web ルート内の場所を参照します。拡張機能は、選択した場所の "Extensions" ディレクトリにインストールされます。

[ 終了 ] をクリックすると、ローカルの ColdFusion サーバーインスタンスが作成されます。これらの拡張機能の使用方法については、91 ページの「[拡張機能の使用](#)」を参照してください。

## リモートサーバーの作成

ColdFusion Builder でリモートサーバーを作成する前に、次の作業を行います。

### リモート ColdFusion サーバーでの Admin サーバーインスタンスの実行

リモートサーバーのバージョンと設定に応じて、次の作業を行います。

#### スタンドアロン設定のリモートサーバー (バージョン 7.0.2、8.0.1)

- 1 "AdminServerComponents.zip" ファイルを "{CFHome}" に解凍します。  
次のファイルが "{ColdFusion ホームディレクトリ }/runtime/bin" にコピーされます。
  - adminstart.bat
  - admin\_jvm.config
  - adminstart.sh

Admin Server インスタンスが "{ColdFusion ホームディレクトリ }/runtime/servers" にコピーされます。

- 2 "{ColdFusion ホームディレクトリ }/runtime/bin" に移動し、"adminstart.bat" (Windows の場合) または "adminstart.sh" (Mac OS の場合) を実行します。

#### マルチサーバー設定または J2EE 設定のリモートサーバー (バージョン 7.0.2、8.0.1)

- 1 "AdminServerComponents.zip" ファイルから、次のファイルを "{JRun ホームディレクトリ }/bin" に解凍します。
  - admin\_jvm.config
  - "adminstart.bat" (Windows の場合) または "adminstart.sh" (Mac OS の場合)
- 2 "{JRun ホームディレクトリ }/bin" に移動し、"adminstart.bat" (Windows の場合) または "adminstart.sh" (Mac OS の場合) を実行します。

#### スタンドアロン設定、マルチサーバー設定、J2EE 設定のリモートサーバー (バージョン 9)

リモートサーバーがバージョン 9 の場合は、Admin Server コンポーネントがデフォルトで含まれています。

- スタンドアロンサーバー設定の場合は、"{ColdFusion ホームディレクトリ }/runtime/bin" に移動して、"adminstart" スクリプトファイルを実行します。
- マルチサーバー設定または J2EE サーバー設定の場合は、"{JRun ホームディレクトリ }/bin" に移動して、"adminstart" スクリプトファイルを実行します。

### リモート ColdFusion サーバーのセキュリティプロパティの更新

- 1 "{ColdFusion ホームディレクトリ }/runtime/lib/security.properties" に移動します。

- 2 jrun.subnet.restriction および jrun.trusted.hosts の値を、ColdFusion Builder がインストールされているコンピュータの IP アドレスに変更します。また、IP アドレスの値としてアスタリスクワイルドカード (\*) を使用すると、制約なしでサーバーを起動および停止できるようになります。

## ColdFusion Builder でのリモートサーバー設定の指定

[サーバー] ビューで、次のいずれかの操作を行います。

- 右クリックして [サーバーの追加] を選択します。
-  をクリックします。

新規 ColdFusion サーバーセットアップウィザードで次の情報を入力します。

### 一般設定

- [サーバー名]: ColdFusion サーバーの名前。
- [説明]: (オプション) サーバーの説明。
- [アプリケーションサーバー]: ドロップダウンリストを選択して [JRun] を選択するか、[その他] を選択して Jrun 以外のサーバーを設定します。

**注意:** ColdFusion Builder 内で JRun 以外のサーバーを起動、停止、または再起動することはできません。

- [ホスト名]: リモートサーバーホストの名前。
- [リモート] を選択します。

**注意:** localhost または 127.0.0.1 以外のホスト名を入力した場合、[リモート] は自動的に選択されます。

### その他の設定

- [WebServer ポート]: 設定する ColdFusion サーバーインスタンスのポート番号を指定します。ColdFusion サーバーのデフォルトポート番号は 8500 です。
- コンテキストルート: (JRun サーバーを使用する J2EE 設定のみに適用) コンテキストルートを入力します。  
J2EE 環境では、1 つのサーバーインスタンス上で、独立した複数の Web アプリケーションを実行できます。したがって、サーバー上で実行される J2EE Web アプリケーションのルートは、それぞれ固有のベース URL に設定されます。これが、コンテキストルート (またはコンテキストパス) と呼ばれるものです。
- [アプリケーションサーバー名]: (JRun サーバーを使用する J2EE 設定のみに適用) ColdFusion をデプロイする JRun サーバーの名前。
- [RDS ユーザー名]: (オプション) RDS を使用している場合は、RDS ユーザー名を指定します。
- [RDS パスワード]: (オプション) RDS パスワードを指定します。

**注意:** RDS パスワードの設定は ColdFusion Administrator で行います。同じく ColdFusion Administrator で管理される ColdFusion Administrator パスワードと RDS パスワードを混同しないでください。

- ColdFusion Builder で SSL のサポートを有効にするには、[SSL の有効化] を選択します。Server Manager に登録されているサーバーとの通信には SSL を使用できます。
- ColdFusion Builder を起動するたびに ColdFusion サーバーを自動的に起動させるには、[自動起動] を選択します。

**注意:** リモートサーバーの場合、[自動停止] は使用できません。

[次へ] をクリックします。

## リモートサーバーの設定

1 [ネーミングポート]: リモートサーバーで実行されている administrator サーバーインスタンスのネーミングポートを指定します。デフォルトのネーミングポート値は 2910 です。ネーミングポート値は {servers}/admin/SERVER-INF/jndi.properties で指定されています。プロパティ java.naming.provider.url のポート値は、ネーミングポートまたは jndiport です。

2 [ユーザー名]: "jrun-users.xml" ファイルに記載されている jmc ユーザー名を指定します。

3 [パスワード]: jmc パスワードを指定します。

4 [ドキュメントルート]: Web ルートの場所を参照して選択します。ColdFusion を IIS などの Web サーバーとともに設定する場合は、Web サーバーのドキュメントルート (c:\inetpub\wwwroot など) を選択します。

5 [マッピング] タブを選択し、次の情報を入力します。マッピングの情報は、リモートサーバーのファイルをプレビューするときやデバッグするときで使用されます。

- [ローカルパス]: リモート ColdFusion サーバー上でプロジェクトまたはフォルダを探すときに ColdFusion Builder で使用するパス。
- [リモートパス]: (URL 接頭辞を指定する場合はオプション) リモート ColdFusion サーバー上のプロジェクトのパス。
- [URL 接頭辞]: (リモートパスを指定する場合はオプション) URL 接頭辞を入力します。URL 接頭辞の詳細と、URL 接頭辞の使用例については、47 ページの「[URL 接頭辞](#)」を参照してください。

**注意:** 必ずしも、サーバーの作成時に URL を指定する必要はありません。サーバーを作成した後でも、[サーバー] ビューでサーバー設定を編集することで、URL 接頭辞を指定できます。既存のプロジェクトまたはフォルダに対する URL 接頭辞を指定することもできます。詳細については、59 ページの「[URL 接頭辞の設定](#)」を参照してください。

- [追加] をクリックしてこのマッピングを追加します。

6 (オプション) 仮想ホストを設定するには、[仮想ホストの設定] タブを選択して、次のようにします。

**注意:** ColdFusion Builder で仮想ホストまたは仮想ディレクトリを指定する場合は、設定済みの Web サーバーで、対応する設定を指定する必要があります。ColdFusion Builder ではこれらの設定は検証されません。そのため、Web サーバーの設定が ColdFusion Builder で指定した内容と異なっても、ColdFusion Builder はエラーを生成しません。

- [新規] をクリックし、[仮想ホストの設定] セクションで次の情報を入力します。

1 [名前]: 仮想ホストの名前を指定します。(vh1 など)

**注意:** 任意の名前を指定できます。必ずしも Web サーバーで指定した名前でなくてもかまいません。

2 [ホスト名]: IIS または Apache Web サーバーの設定で指定したとおりの仮想ホスト名。(www.example1.com など)

ColdFusion Builder で仮想ホストを作成した場合、その仮想ホストでは <サーバー名>-<仮想ホスト名> というネーミング規則が使用されます。プロジェクトプロパティの [サーバー] ドロップダウンリストには、このネーミング規則を使用した仮想ホスト名が表示されます。たとえば、ColdFusion サーバー (localhost) で "vh1" という名前の仮想ホストを作成した場合、仮想ホストを識別するために ColdFusion Builder で使用されるネーミング規則は、"localhost-vh1" です。

3 [ポート]: Web サーバーの仮想ホストに割り当てられたポート。

4 [タイプ]: ドロップダウンリストから [HTTP] または [HTTPS] を選択します。

5 [ドキュメントルート]: 仮想ホストのホームディレクトリの場所を参照するか、手動で入力します。

たとえば、Web サイト www.example.com が Apache Web サーバー上の "C:\abc" ディレクトリにマップされている場合は、ホームディレクトリとして「C:\abc」と入力します。

- (オプション) 仮想ディレクトリを作成するには、[ 仮想ディレクトリ ] をクリックして次の情報を入力します。

- 1 [ エイリアス ] : フォルダパスのエイリアスを指定します。
- 2 [ 場所 ] : エイリアスを指定するフォルダパスを参照するか、手動で入力します。

たとえば、Web サイト ([www.example.com](http://www.example.com)) のドキュメントルートが `c:\abc` であり、`d:\xyz\images` でアクセス可能なフォルダ内のイメージを使用する必要があるとします。その場合は、フォルダパス "`d:\xyz\images`" に対して "images" という名前のエイリアスを定義します。仮想ディレクトリの詳細については、48 ページの「[仮想ディレクトリ](#)」を参照してください。

- 3 [ 追加 ] をクリックします。
- 4 [ OK ] をクリックすると、仮想ディレクトリが [ 仮想ディレクトリの設定 ] テーブルに追加されます。

- [ 適用 ] をクリックします。仮想ホストが [ 仮想ホスト ] テーブルに追加されます。

仮想ホストの設定を変更するには、[ 仮想ホスト ] テーブルから仮想ホストを選択し、設定を変更して、[ 更新 ] をクリックします。

仮想ホストの詳細と、ColdFusion Builder での有用性については、48 ページの「[仮想ホスト](#)」を参照してください。

[ 次へ ] をクリックします。

### 拡張機能のインストール

ColdFusion Builder に付属する拡張機能をインストールするには、[ 拡張機能のインストール ] を選択します。

- 1 ColdFusion Web ルートの場所を参照して選択します。
- 2 リモート ColdFusion サーバー上の ColdFusion Web ルートの場所を参照して選択します。
- 3 拡張機能をインストールする Web ルート内の場所を参照します。拡張機能は、選択した場所の "Extensions" ディレクトリにインストールされます。

[ 終了 ] をクリックすると、リモート ColdFusion サーバーインスタンスが追加されます。これらの拡張機能の使用方法については、91 ページの「[拡張機能の使用](#)」を参照してください。

リモートサーバーへの接続に成功すると、ColdFusion Builder に表示されるサーバーステータスが [ 起動中 ] に変わります。リモートサーバーへの接続に成功しなかった場合は、サーバーステータスが [ 不明 ] に変わります。エラーの詳細については、30 ページの「[\[ コンソール \] ビュー](#)」を参照してください。

## [ サーバー ] ビューの使用

[ サーバー ] ビューは、ColdFusion Builder からサーバーの起動、停止、再起動を行う場合に使用します。また、[ サーバー ] ビューからは、ColdFusion サーバーモニタと ColdFusion Server Administrator も起動できます。

[ サーバー ] ビューがワークベンチに表示されていない場合は、[ ウィンドウ ]-[ ビューの表示 ]-[ その他 ] を選択して、[ サーバー ] ビューを追加します。[ ビューの表示 ] ダイアログボックスで、[ ColdFusion ]-[ サーバー ] ビューを選択します。

### サーバーの起動、停止、再起動、削除

- 1 [ サーバー ] ビューでサーバーを右クリックします。
- 2 [ サーバーを起動 ]、[ サーバーを停止 ]、[ サーバーを再起動 ]、[ サーバーを削除 ] のいずれかを選択します。[ サーバーを再起動 ] を選択すると、サーバーが停止した後に起動されます。

## ColdFusion サーバーモニタの起動

- 1 [サーバー] ビュー内を右クリックします。
- 2 [サーバーモニタの起動] を選択して ColdFusion Administrator を開きます。
- 3 パスワードを指定します。
- 4 [OK] をクリックしてサーバーモニタを表示します。

## ColdFusion Server Administrator の起動

- 1 [サーバー] ビュー内を右クリックします。
- 2 [ColdFusion Administrator を起動] を選択します。
- 3 ユーザー名とパスワードを指定します。
- 4 [OK] をクリックして、ColdFusion Administrator を表示します。

## RDS サーバー設定のインポート

RDS サーバーが既に設定されている場合は、RDS サーバーの設定を直接インポートして、RDS サーバーを [サーバー] ビューに追加できます。

- 1 [サーバー] ビューを右クリックして、[サーバーの追加] を選択します。
- 2 [RDS サーバーから設定をインポート] を選択し、[サーバー] ビューに追加する RDS サーバーをドロップダウンリストから選択します。
- 3 [OK] をクリックします。
- 4 RDS 設定を変更するには、[ColdFusion サーバーセットアップの変更] ダイアログボックスで変更を指定して、[終了] をクリックします。

これで、サーバーをプロジェクトに関連付けて、プロジェクトのデバッグ、プレビュー、テストを行えるようになります。

## 第6章：プロジェクトの管理

### プロジェクトについて

プロジェクトには、ColdFusion アプリケーションの開発に使用される ColdFusion コンポーネント、インターフェイス、HTML ページ、CFML ページなどのリソースが格納されています。

各プロジェクトはデフォルトのワークスペースに保存されます。ワークスペースには、プロジェクトと他のメタデータが保存されます。推奨されるワークスペースの場所は、ColdFusion のドキュメントルートです。

プロジェクトを ColdFusion サーバーに関連付けることができます。そのようにすれば、最終的なデプロイの前にプロジェクトをテストできます。

**注意：**プロジェクトに含まれるファイルを Web ブラウザでプレビューするには、プロジェクトの場所が ColdFusion のドキュメントルートに設定されていることを確認します。

ColdFusion Builder で CFML を開発するには、最初に ColdFusion プロジェクトを作成します。ColdFusion Builder でプロジェクトを作成する方法については、56 ページの「[ColdFusion プロジェクトの作成](#)」を参照してください。

### ColdFusion プロジェクトの作成

プロジェクトビルダーウィザードの案内に従って、ColdFusion プロジェクトを作成します。

#### プロジェクト情報の入力

- 1 [ナビゲーター] 領域内を右クリックし、[新規]-[ColdFusion プロジェクト] をクリックします。
- 2 プロジェクトビルダーウィザードでプロジェクト名を指定します。
- 3 デフォルトのプロジェクトの場所を変更するには、[デフォルトの場所を使用] を選択解除します。
- 4 [次へ] をクリックして ColdFusion サーバーの詳細を指定します。

#### サーバーの詳細の入力

- 1 [サーバー] ポップアップメニューからサーバーを選択します。まだ ColdFusion サーバーが設定されていない場合は、[サーバーの追加] をクリックしてサーバーを追加します。ColdFusion サーバーをセットアップする方法については、48 ページの「[ColdFusion サーバーの作成](#)」を参照してください。

**注意：**プロジェクトがサーバーの Web ルートにある場合は、[サンプル URL] ボックスにサーバーの URL が自動的に入力されます。たとえば、<http://127.0.0.1:8500/eval> のように入力されます (127.0.0.1 はサーバーホスト、8500 はポート番号、eval はプロジェクト名です)。

- 2 コンピュータにインストールされている Web ブラウザを選択して、外部 Web ブラウザを使用するようにプレビュー設定を指定します。デフォルトでは、内部ブラウザが選択されています。
- 3 [次へ] をクリックします。

## 既存のソースの追加

- 1 この手順では、次の操作を行えます。
  - 既存のリソースフォルダをプロジェクトにリンクする。
  - Model Glue などの設定済みアプリケーションを現在のプロジェクトで選択する。
- 2 [追加] をクリックして、プロジェクトにリンクするフォルダを選択します。
- 3 [終了] をクリックして、新しい ColdFusion プロジェクトを作成します。

リソースをリンクする方法については、61 ページの「[ワークスペース外のリソースへのリンク](#)」を参照してください。

## プロジェクトのプロパティおよびサーバーの設定

ColdFusion Builder の [プロパティ] ダイアログボックスを使用すると、プロジェクトおよびサーバーのプロパティを設定できます。

このダイアログボックスを開くには、[ナビゲーター] ビューでプロジェクトを右クリックして [プロパティ] を選択します。

### プロジェクトのプロパティの設定

- ❖ [プロパティ] ダイアログボックスの左ペインで [ColdFusion プロジェクト] を選択します。ここで、外部プロジェクトをリンクとして追加および削除できます。

### 変数マッピングの設定

編集中のファイルやインクルードファイルで定義されていないコンポーネントの引数にコードアシストを用意するには、変数マッピングを使用します。

- 1 [プロパティ] ダイアログボックスの左側にツリービュー構造が表示されます。[ColdFusion 変数マッピング] を選択します。
- 2 [新規] をクリックします。
- 3 [変数名] フィールドで、マッピングする変数の名前 (たとえば application.cfc1) を指定します。
- 4 [マッピング対象] フィールドに、プロジェクト変数の完全修飾名 (たとえば com.adobe.mycfcs.cfc1) を入力します。

### サーバー設定の指定

- [プロパティ] ダイアログボックスの左ペインで [ColdFusion サーバーの設定] を選択すると、[ColdFusion サーバーの設定] ページが開きます。
- ColdFusion サーバーのインスタンスを作成するには、[サーバー] ポップアップメニューを使用します。

設定済みのサーバーインスタンスがある場合は、そのインスタンスを ColdFusion プロジェクトに割り当てることができます。サーバーインスタンスを作成する方法については、48 ページの「[ColdFusion サーバーの作成](#)」を参照してください。

## ColdFusion ページ、インターフェイス、およびコンポーネントの追加

ColdFusion Builder には、ColdFusion ページ、インターフェイス、およびコンポーネントを作成するためのウィザードが用意されています。

### ColdFusion ページの作成

- 1 [ナビゲーター] で ColdFusion プロジェクトを右クリックします。
- 2 [新規]-[ColdFusion ページ] をクリックします。
- 3 新規 ColdFusion ページウィザードで、ColdFusion (CFM) ページの名前を指定します。
- 4 [終了] をクリックします。

### ColdFusion コンポーネントの作成

- 1 [ナビゲーター] で ColdFusion プロジェクトを右クリックします。
- 2 [新規]-[ColdFusion コンポーネント] をクリックします。
- 3 新規 ColdFusion コンポーネントウィザードで、次の項目を指定します。
  - コンポーネント名
  - コンポーネントを識別するためのヒント (オプション)
  - 拡張するコンポーネント
  - 実装するインターフェイス
- 4 出力の型を true または false に設定します。
- 5 [次へ] をクリックします。
- 6 [追加] をクリックして、コンポーネントのプロパティの詳細 (プロパティ名、表示名、ヒント、デフォルト値など) を指定します。
- 7 [タイプ] ドロップダウンリストからプロパティの型を選択するか入力します。
- 8 (オプション) [getter の追加] を選択し、[アクセス] ドロップダウンリストからアクセスタイプを選択します。
- 9 (オプション) [setter の追加] を選択し、[アクセス] ドロップダウンリストからアクセスタイプを選択します。
- 10 [OK] をクリックして、新規 ColdFusion コンポーネントウィザードに戻ります。
- 11 ColdFusion コンポーネントに関数を追加するには、[関数の追加] をクリックします。
- 12 新規関数ウィザードで、関数名、表示名、ヒント、アクセスタイプ、戻り値の型、役割、出力の型などの関数の詳細を指定します。
- 13 [OK] をクリックします。
- 14 選択した関数に引数を追加するには、[引数の追加] をクリックします。
- 15 [新規関数引数] ダイアログボックスで、引数名、表示名、ヒント、引数の型、および引数のデフォルト値を指定します。

**注意:** 選択できる引数の型は、ColdFusion インターフェイスの引数の型と同じです。
- 16 [OK] をクリックします。
- 17 [終了] をクリックして ColdFusion コンポーネント (CFC ファイル) を作成します。

**注意:** [アウトライン] ビューを使用すると、ColdFusion ページまたは ColdFusion コンポーネントに追加した関数やタグを参照できます。

## ColdFusion インターフェイスの作成

- 1 [ナビゲーター] で ColdFusion プロジェクトを右クリックします。
- 2 [新規]-[ColdFusion インターフェイス] をクリックします。
- 3 新規 ColdFusion インターフェイスウィザードで、インターフェイス名、ヒント、および表示名を指定します。
- 4 別のインターフェイスを拡張するには、CFC の名前を参照または指定します。
- 5 関数を追加するには、[関数の追加] をクリックし、関数名、関数のヒント、戻り値の型、役割、出力 (true または false) などの詳細を指定します。
- 6 選択した関数の引数を指定するには、[引数の追加] をクリックします。[タイプ] ドロップダウンリストから引数の型を選択するか入力します。
- 7 [終了] をクリックして ColdFusion インターフェイスを作成します。

## その他のファイルの追加

ColdFusion プロジェクトには次のファイルも追加できます。

- .html
- .js
- .css
- .lxr (レクサーファイル)
- .col (色付けファイル)
- .sdoc (ScriptDoc ファイル)
- 表題なしの上記ファイル

また、汎用的なテキストファイル、フォルダ、プロジェクトも作成できます。これらのファイルをプロジェクトに追加する手順は、ColdFusion ページや ColdFusion コンポーネントを追加する手順とほぼ同じです。これらのファイルをプロジェクトに追加するには、次のようにします。

- 1 プロジェクトを右クリックします。
- 2 [新規]-[その他] を選択します。
- 3 [プロジェクトファイル] を展開してファイル名を指定するか、[表題なしのファイル] を展開して表題なしのファイルを作成します。
- 4 作成するファイルを選択します。
  - 表題なしのファイルを作成する場合は、[終了] をクリックします。
  - [プロジェクトファイル] を選択した場合は、[次へ] をクリックしてファイル名を指定します。[終了] をクリックします。

## URL 接頭辞の設定

- 1 [ナビゲーター] ビューで、URL 接頭辞を指定するプロジェクトまたはフォルダを選択します。

- 2 選択したプロジェクトまたはフォルダを右クリックし、[URL 接頭辞の設定] を選択します。
- 3 選択したリソースの URL を入力します。

## プロジェクトのインポート、エクスポート、および削除

### プロジェクトのインポート

ColdFusion Builder には、ColdFusion プロジェクトだけでなく、ColdFusion 以外のプロジェクトもインポートできます。

- 1 [ナビゲーター] ビュー内を右クリックし、[インポート] を選択します。
- 2 インポートウィザードで、[ColdFusion]-[既存プロジェクトのインポート] を選択します。
- 3 [ColdFusion プロジェクトのインポート] ダイアログボックスで、プロジェクトを参照して選択します。すべてのプロジェクトのリストが表示されます。
- 4 [更新] をクリックするとプロジェクトのリストが更新されます。
- 5 同じ場所にある ColdFusion 以外のプロジェクトや ColdFusion Builder 以外のプロジェクトもすべて表示するには、[すべてのプロジェクトを表示] チェックボックスを選択します。
- 6 プレビュー、編集、デバッグなど ColdFusion Builder 固有の機能を ColdFusion Builder 以外のプロジェクトに適用するには、[ColdFusion Nature を非 ColdFusion プロジェクトに追加] を選択します。
- 7 適切なプロジェクトを選択して、[終了] をクリックします。

**注意:** プロジェクトのインポートでは、サーバーの詳細、起動設定、ブラウザの詳細、アプリケーションの詳細、リンク先フォルダなどのプロジェクトプロパティが保持されます。

### プロジェクトのエクスポート

ColdFusion Builder プロジェクトまたは ColdFusion Builder 以外のプロジェクトから指定の場所にファイルとフォルダをエクスポートできます。ただし、ColdFusion Builder プロジェクトの場合は、サーバー設定や環境設定など、ColdFusion Builder 固有の機能はエクスポートされません。

- 1 [ナビゲーター] ビュー内を右クリックし、[エクスポート] を選択します。
- 2 エクスポートウィザードで、[ColdFusion]-[エクスポート] を選択します。
- 3 プロジェクトのエクスポートダイアログボックスで、プロジェクトをエクスポートする場所を参照して選択します。
- 4 [すべてを選択] をクリックして、プロジェクト内のファイルとフォルダをすべてエクスポートするか、[タイプをフィルター] をクリックして、エクスポートするファイルを指定します。
- 5 [終了] をクリックすると、指定した場所にプロジェクトがエクスポートされます。

### プロジェクトの削除

プロジェクトを削除する場合は、現在のワークスペースからプロジェクトを削除します。同時にコンピュータのファイルシステムからプロジェクトを削除することもできます。

- 1 [ナビゲーター] ビュー内を右クリックし、[削除] を選択します。
- 2 ワークスペースとファイルシステムの両方からプロジェクトを削除するには、[ディスク上からプロジェクト・コンテンツを削除] を選択します。ファイルシステムからプロジェクトを完全に削除した後に、元に戻すことはできません。

## プロジェクトおよびファイルの除外

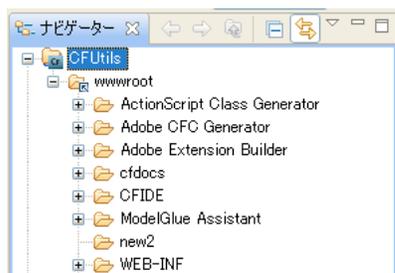
ColdFusion Builder では、特定のファイル、ファイルタイプ、およびフォルダを、同期、アップロード、ダウンロード操作の対象から除外する（隠す）ことができます。この機能は、同期やアップロードの対象から除外するファイルを大量に含むディレクトリがある場合に便利です。

ファイルやフォルダを除外する方法には、次の 2 つがあります。

- [ナビゲーター] ビューでファイルまたはフォルダを右クリックして、[同期]-[このファイルタイプを除外] を選択します。ファイルタイプの除外を解除するには、ファイルまたはフォルダを右クリックして、[除外]-[このファイルタイプの除外を解除] を選択します。
- 次の手順に従い、ワイルドカードや正規表現を使用して特定のファイルタイプを除外するように設定します。
  - 1 [ナビゲーター] ビューでファイルまたはフォルダを右クリックして、[同期]-[拡張除外設定] を選択します。
  - 2 [設定] ダイアログボックスで (+) アイコンをクリックします。
  - 3 [ファイル/フォルダーを無視] ダイアログボックスで、ファイル名またはフォルダ名を指定します。
  - 4 [OK] をクリックします。除外されるファイルのリストに、指定したファイルまたはフォルダが表示されます。

## ワークスペース外のリソースへのリンク

プロジェクトやワークスペースの外部にあるリソースにもリンクできます。この機能は、共用の場所にあるリソースを使用する場合に役に立ちます。リンク先のリソース、またはリンク先のリソースを含んでいるフォルダは、次のように表示されます。



ワークスペース外のリソースへのリンク

- 1 ColdFusion Builder の [ナビゲーター] ビューで、リンク対象のリソースを追加するプロジェクトを右クリックします。
- 2 リンクするリソースを選択します。たとえば、フォルダにリンクする場合は、[新規]-[フォルダー] を選択します。
- 3 リンクするリソースの名前を入力します。ここで名前を入力しなかった場合は、リソースのデフォルト名が割り当てられます。
- 4 [拡張] をクリックします。
- 5 [ファイル・システム内のフォルダーにリンク] を選択します。リソースの場所を参照して選択するか、手動で入力します。
- 6 [終了] をクリックして、リソースをプロジェクトにリンクします。

## パス変数を使用したリソースへのリンク

リソースにリンクするときは、リソースが保存されている場所の絶対パスを指定する代わりに、パス変数を定義することもできます。パス変数を定義した後に、リンクするリソースのパスを設定できます。

- 1 ColdFusion Builder の [ナビゲーター] ビューで、リンク対象のリソースを追加するプロジェクトを右クリックします。
- 2 リンクするリソースを選択します。たとえば、フォルダにリンクする場合は、[新規]-[フォルダー] を選択します。
- 3 リンクするリソースの名前を入力します。ここで名前を入力しなかった場合は、リソースのデフォルト名が割り当てられます。
- 4 [拡張] をクリックします。
- 5 [ファイル・システム内のフォルダーにリンク] を選択します。リソースの場所を参照して選択するか、手動で入力します。
- 6 [変数] をクリックします。
- 7 既存のパス変数を選択するか、[新規] をクリックしてパス変数を作成します。  
**注意:** 既存のリソース変数のリストは、メインメニューの [ウィンドウ]-[設定] を選択してから、[一般]-[ワークスペース]-[リンクされたリソース] を選択して表示することもできます。[リンクされたリソース] ダイアログボックスでは、リンクされたリソース変数の編集と作成を行えます。
- 8 パス変数を作成するには、リソースの名前を入力し、リソースの場所を参照して選択するか、絶対パスを入力します。[OK] をクリックして、パス変数を追加します。
- 9 [終了] をクリックして、リソースをプロジェクトにリンクします。

## FTP および Secure FTP 接続によるプロジェクトのデプロイ

ColdFusion Builder では、ファイル転送プロトコル (FTP) を使用して ColdFusion プロジェクトを Web サーバーにデプロイできます。

また、ColdFusion Builder は、Secure File Transfer Protocol (SFTP) 接続および File Transfer Protocol Secure (FTPS) 接続を使用したファイル転送もサポートしています。SFTP および FTPS は、セキュリティ機能を備えた FTP であり、リモートサーバーとの間で送受信されるファイルが暗号化されます。

### ColdFusion Builder での FTP 接続または Secure FTP 接続の作成

- 1 [ナビゲーター] ビューでプロジェクトを右クリックします。
- 2 [同期]-[同期接続を新規作成] をクリックします。
- 3 接続の名前を指定します。
- 4 [ローカル] ドロップダウンリストを選択し、ファイルまたはプロジェクトが含まれる場所を選択して、デプロイする ColdFusion プロジェクトまたはファイルを選択します。
- 5 リモートサーバーを選択し、[新規接続] ダイアログボックスで次の情報を入力します。
  - a [サイト名]: 新しいリモートサイトの名前。
  - b 接続タイプを選択します。
  - c [サーバー]: 接続先サイトの URL (admin.myWebSite.com など)。サイトの IP アドレスを入力することもできます。
  - d [ユーザー名] / [パスワード]: サーバーのユーザー名およびパスワード。

- e [公開鍵認証を使用]: このオプションは、接続タイプとして SFTP を選択した場合にのみ使用できます。公開鍵認証では、コンピュータ上の鍵ペアを選択して、公開鍵をサーバーにコピーできます。
  - f [リモートパス]: リモートサイトのルートの下にあるサブフォルダに接続する場合は、フォルダパスを指定します。
  - g [拡張オプション] をクリックして、次の項目を指定します。
    - [ポート]: 22 番以外のポートを使用する場合は、そのポートを指定します。
    - FTP 接続セッションをバックグラウンドで実行する場合は、[パッシブモードを使用] を選択します。
    - チーム環境で作業している場合は、ファイルをアップロードすると、他のユーザーが加えた変更が上書きされてしまう可能性があります。そのような場合は、予防策として [サーバーとクライアントの時差を自動的に計算] を選択してください。
- 6 [OK] をクリックして FTP 接続を追加します。

#### プロジェクト、ファイル、フォルダのアップロード、ダウンロード、または同期

- 1 [ナビゲーター] ビューでプロジェクトまたはファイルを右クリックし、[同期]-[アップロード] または [ダウンロード] を選択します。
- 2 [サイト接続を選択] ダイアログボックスで接続を選択します。
- 3 [OK] をクリックします。

## Ajax ライブラリのインポート

- 1 [ナビゲーター] ビューで、Ajax ライブラリをインポートするプロジェクトを右クリックし、[インポート] をクリックします。
- 2 インポートウィザードで、[ColdFusion]-[Ajax ライブラリインポートウィザード] を選択し、[次へ] をクリックします。
- 3 インポートする Ajax ライブラリを選択し、[終了] をクリックします。

## Flash Builder プロジェクトの使用

ColdFusion Builder で Flash Builder プロジェクトを使用するには、Flash Builder プロジェクトを ColdFusion Builder にインポートします。その後、[ナビゲーター] ビューで Flash Builder プロジェクトを選択して ColdFusion Builder の機能を適用し、右クリックして [CF Nature の適用] を選択します。

ColdFusion Builder の機能を Flash Builder プロジェクトに適用すると、ColdFusion Builder 内から Flash Builder プロジェクトのプレビュー、編集、デバッグを行えるようになります。

## AIR アプリケーションの開発

Adobe ColdFusion Builder には、Adobe® AIR® プロジェクトの作成、Adobe AIR アプリケーションのデバッグ、パッケージ化、および電子署名の追加を行うためのツールが用意されています。

### AIR プロジェクトの作成

- 1 [ファイル]-[新規]-[プロジェクト] を選択します。

- 2 新規プロジェクトウィザードで、[HTML プロジェクト]-[Adobe AIR プロジェクト] を選択し、[次へ] をクリックします。
- 3 プロジェクト名と、プロジェクト内のファイルの場所を入力します。
- 4 サンドボックスアプリケーションコードを含む HTML スタートページを指定します。アプリケーション以外のサンドボックスコードを選択することもできます。[次へ] をクリックします。
- 5 アプリケーション XML ファイルに関する次の情報を指定し、[次へ] をクリックします。
  - ID: アプリケーションの識別文字列として使用する一意のアプリケーション ID。
  - 名前: アプリケーションの名前。
  - ファイル名: ファイルの名前とアプリケーションがインストールされるフォルダ。
  - バージョン: アプリケーションのバージョン。
  - アイコン: アプリケーションを表すアイコンファイル。アイコンを指定しない場合は、オペレーティングシステムのデフォルトアイコンが使用されます。
- 6 ウィンドウのスタイル、寸法、サイズ変更オプションを指定し、[次へ] をクリックします。
- 7 プロジェクトにインポートする AIR フレームワークを選択し、[次へ] をクリックします。
- 8 インポートする Ajax JavaScript ライブラリを選択し、[終了] をクリックして AIR プロジェクトを作成します。

## AIR アプリケーションの実行とデバッグ

- 1 [ナビゲーター] ビューから、アプリケーションのソースファイル (Application.XML) を開きます。
- 2 メインツールバーの [実行] をクリックして AIR アプリケーションを起動します。
- 3 アプリケーションをデバッグするには、ワークベンチツールバーで  をクリックします。

アプリケーションが ADL アプリケーション (AIR Debugger Launcher) 内で起動して実行されます。ColdFusion デバッガによってブレークポイントまたは実行時エラーが検出され、他の ColdFusion アプリケーションと同様にアプリケーションをデバッグできます。詳細については、66 ページの「[アプリケーションのデバッグ](#)」を参照してください。

## AIR アプリケーションのパッケージ化と電子署名の追加

アプリケーションが完成して配布する準備ができたなら、AIR ファイルにパッケージ化します。パッケージ化の手順は次のとおりです。

- 1 [ナビゲーター] ビューで、パッケージ化する AIR プロジェクトを右クリックし、[エクスポート] をクリックします。
- 2 エクスポートウィザードで、[Adobe AIR]-[Adobe AIR パッケージ] を選択し、[次へ] をクリックします。
- 3 AIR プロジェクトおよびアプリケーション記述ファイルを選択します。
- 4 デフォルトの AIR SDK を選択するか、別の Adobe AIR SDK を設定し、[次へ] をクリックします。別の AIR SDK を設定するには、[Adobe AIR SDK を設定] をクリックし、AIR SDK を選択します。また、AIR SDK の追加、編集、削除も可能です。
- 5 既存の電子証明書を選択するか、自己署名証明書を作成して、AIR アプリケーションに電子署名を追加できます。AIR アプリケーションに電子署名を追加すると、アプリケーションが信頼できる証明書で署名されていることがユーザーに保証され、発行者 ID が表示されます。
  - a [エクスポートされた AIR アプリケーションに電子署名を追加] を選択します。
  - b 既存の電子証明書を選択するか、[Configure Certificates] をクリックして別の電子証明書を選択します。[追加] をクリックして自己署名証明書を作成します。
  - c 電子証明書のパスワードを指定します。

- d [TimeStamp AIR Package] を選択した場合は、インストールパッケージに署名する際に、ADT (AIR Developer Tool) がタイムスタンプ局と自動的に通信して時刻を確認します。タイムスタンプ情報は AIR ファイルに記録されません。確認済みのタイムスタンプを含む AIR ファイルは、いつでもインストールできます。
- e AIR アプリケーションを新しい証明書に移行するには、[Migrate AIR Application] を選択します。その場合は、新しい証明書と古い旧証明書の両方を使用して AIR ファイルに署名します。移行前の証明書と、そのパスワードを指定します。

電子署名なしで AIR アプリケーションをパッケージ化することもできます。その場合は、[エクスポートされた AIR アプリケーションに電子署名を追加] を選択解除します。電子署名なしで AIR アプリケーションをパッケージ化すると、中間 AIR ファイル (.airi) が作成されます。中間 AIR ファイルをデプロイまたはインストールすることはできません。このファイルは一般に、(開発者によって) テスト目的で使用され、AIR ADT コマンドラインツールで起動できます。

- 6 [次へ] をクリックして、エクスポートする AIR ファイルから除外するファイルが必要に応じて選択します。デフォルトでは、すべてのファイルがパッケージに含まれます。
- 7 [終了] をクリックして AIR ファイルを生成します。

## 第7章：アプリケーションのデバッグ

デバッグでは、アプリケーションを調べてトラブルシューティングできます。デバッグ中には、コード内の特定のポイントでアプリケーションを停止するかどうかを制御できます。また、重要な変数を監視したり、コードをテストすることもできます。デバッグでは、アプリケーションの起動方法を制御するための設定が使用されます。アプリケーションをデバッグするときは、デバッグ版のアプリケーションファイルを実行します。

### ColdFusion デバッガの使用

ColdFusion デバッガを使用する前に、次の作業を行ってください。

#### デバッガを使用するための ColdFusion の設定

デバッガを使用する前に、次のことを確認します。

- デバッグ対象のプロジェクト、またはデバッグ対象のファイルを含むプロジェクトにサーバーが関連付けられているかどうかを確認します。
  - 1 [ナビゲーター]ビューでプロジェクトを右クリックし、[プロパティ]を選択します。
  - 2 [プロパティ]ダイアログボックスで[ColdFusion サーバーの設定]を選択します。
  - 3 [サーバーの選択]で、サーバーが選択されていることを確認します。サーバーが選択されていない場合は、[サーバー]ドロップダウンリストからサーバーを選択するか、[サーバーの追加]を選択して新しいサーバーを設定します。
- ColdFusion サーバーで RDS を有効にし、ColdFusion Builder で RDS の設定情報が正しく指定されていることを確認します。
- ColdFusion Administrator でデバッグが有効になっていることを確認します。
  - 1 ColdFusion Administrator で、[デバッグとロギング]-[デバッガの設定]を選択します。
  - 2 [ラインデバッグの許可]を選択します。
  - 3 デバッグに使用するポートを指定します。デフォルト値は 5005 です。
  - 4 同時デバッグセッションの最大数を指定します。デフォルト値は 5 です。
  - 5 [変更の送信]をクリックします。
  - 6 リクエストのタイムアウト時間を長くするには、次のようにします。
    - a [サーバーの設定]-[設定]を選択します。
    - b [リクエストタイムアウト(秒)]を選択して、必要なタイムアウト値を入力します。たとえば、300 のように入力します。
    - c [変更の送信]をクリックします。
  - 7 デバッガサーバーは、手順 3 で指定したもとは別のポートで、ColdFusion Builder からのコマンドをリスンします。デフォルトでは、デバッガサーバーが使用するポートは、サーバーの起動時に使用可能なポートの中からランダムに選択されます。ColdFusion (およびデバッガサーバー) がファイアウォールの背後にある場合、デバッガがリスンするランダムなポートがファイアウォールによりブロックされるため、この動作は問題になる可能性があります。

この問題を回避するには、デバッガサーバーが使用するポート番号を固定的に指定し、このポートの使用をファイアウォールで許可します。デバッガサーバーのポート番号を固定的に設定するには、ColdFusion Administrator の [Java と JVM] ページ (または J2EE アプリケーションサーバーの適切なセクション) で、次の JVM 引数を指定します。portNumber の部分は、実際に使用するポート番号に置き換えてください。

```
-DDEBUGGER_SERVER_PORT=portNumber
```

- 8 ColdFusion を再起動します。J2EE 設定の ColdFusion を実行している場合は、指定のデバッグポートを使用してデバッグモードでサーバーを再起動します。

## J2EE 設定の ColdFusion におけるデバッグのセットアップ

- 1 サーバー設定の ColdFusion を実行していない場合は、実行するアプリケーションサーバーの設定ファイルまたはスタートアップスクリプトで Java デバッグパラメータを指定します。パラメータは次のようになります。

```
-Xdebug -Xrunjdw:transport=dt_socket,server=y,suspend=n,address=<port_number>
```

ColdFusion Administrator の [ デバッガの設定 ] ページで指定したのと同じポート番号を指定する必要があります。

サーバー設定の ColdFusion を実行している場合は、ColdFusion Administrator の [ デバッガの設定 ] ページを使用すると、ColdFusion によってこれらのデバッグパラメータが "jvm.config" ファイルに書き込まれます。

- 2 サーバー設定の ColdFusion を実行しておらず、アプリケーションサーバーが JRE 1.6 で実行されていない場合は、アプリケーションサーバーで実行している JDK バージョンの "tools.jar" ファイルを、ColdFusion の "¥lib" フォルダにコピーします。たとえば、JRE 1.4 で JRun を実行している場合は、JDK 1.4 の "tools.jar" ファイルを ColdFusion の "¥lib" フォルダにコピーする必要があります。
- 3 サーバーバージョンの ColdFusion を実行しており、JRE 1.6 以外のバージョンの JRE を "jvm.config" ファイルで指定する場合は、"jvm.config" ファイルで指定されている JDK バージョンの "tools.jar" ファイルを ColdFusion の "¥lib" フォルダにコピーします。

## マルチサーバー設定の ColdFusion におけるデバッグのセットアップ

マルチサーバー設定で実行している ColdFusion アプリケーションをデバッグするには、次のコマンドを使用してコマンドラインから ColdFusion サーバーを開始します。

```
jrun -config <path_to_jvm_config> -start <server_name>
```

## ColdFusion Builder でのデバッグ設定の指定

- 1 ColdFusion Builder で [ ウィンドウ ]-[ 設定 ] を選択します。
- 2 ツリービュー構造で、[ColdFusion]-[ デバッグの設定 ] を選択します。
- 3 ホームボタンをクリックしたときにデバッガの [ デバッグ出力バッファー ] に表示するホームページの URL を指定します。
- 4 デバッグ可能なファイルタイプの拡張子と、デバッガに認識させるデバuggスコープを指定します。サイズの大きなファイルをデバッグする際のパフォーマンスを向上させるために、情報を収集する必要がないスコープはすべて選択解除してください。
- 5 ColdFusion エラーが発生した行でデバッガを停止する場合は、[CFML ランタイム例外発生時にブレーク] を選択します。
- 6 警告ダイアログボックスを表示せずに [TailView] ビューでサーバーログを確認する場合は、[ 警告ダイアログボックスを表示せず Eclipse エラーログに例外を記録 ] を選択します。

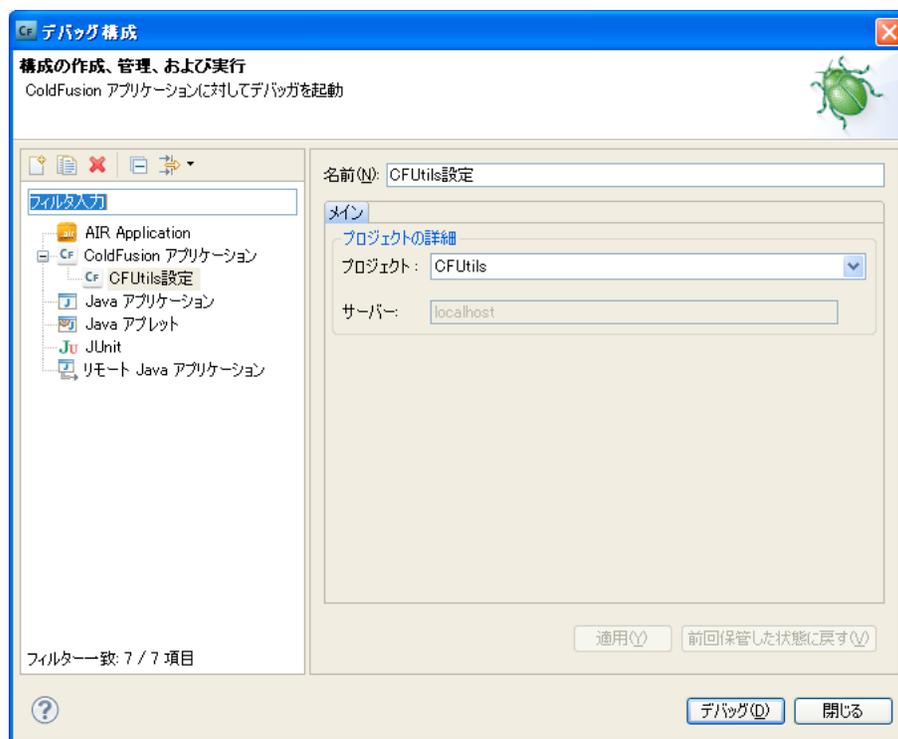
## アプリケーションのデバッグ

ColdFusion Administrator でデバッグを有効にし、ColdFusion Builder でデバッグを設定すると、ColdFusion Builder でプロジェクトをデバッグできるようになります。

### 起動設定の作成と編集

ColdFusion Builder でアプリケーションのプロジェクトを初めてデバッグすると、そのプロジェクトに固有の起動設定が自動的に作成されます。起動設定では、プロジェクト名 (デバッグ対象のプロジェクトに基づく名前)、メインのアプリケーションファイル、およびアプリケーションをデバッグするためのパスが自動的に定義されます。

起動設定の管理は [ 構成の作成、管理、および実行 ] ダイアログボックスで行います。



デバッグ設定

ColdFusion Builder によって作成されたデフォルトの起動設定は編集できます。

- 1 デバッグするプロジェクトを [ ナビゲーター ] ビューで選択します。
- 2 起動設定には、次のいずれかの方法でアクセスできます。
  - [ 実行 ]-[ デバッグ ] または [ 実行 ]-[ デバッグの構成 ] を選択します。
  - ワークベンチツールバーで  を選択します。
  - プロジェクトを右クリックし、[ デバッグ ]-[ デバッグの構成 ] を選択します。
- 3 [ デバッグの構成 ] ダイアログボックスで、編集する起動設定を選択します。  
起動設定を新規に作成することもできます。また、既存の設定に基づいて起動設定を作成することもできます。
- 4 必要に応じて、起動設定の内容を変更するための [ パースペクティブ ] リンクをクリックします。

## アプリケーションの実行

ブレークポイントを設定する前に、デバッグするアプリケーションを実行します。デバッグする前にアプリケーションを実行するとアプリケーションがコンパイルされるため、デバッグ時のパフォーマンスが向上します。

- 1 実行するプロジェクトを [ナビゲーター] ビューで選択します。
- 2 [実行構成] ダイアログボックスで、実行設定を選択または変更します。このダイアログボックスは、次のいずれかの方法で表示できます。
  - [実行]-[実行構成] を選択します。
  - プロジェクトを右クリックし、[実行]-[実行構成] を選択します。
- 3 必要に応じて、実行するか変更する設定を選択します。
- 4 [実行] をクリックします。

コンピュータのデフォルトブラウザでアプリケーションが実行されます。別の外部ブラウザを指定するには、次のようになります。

- (Windows の場合) プロジェクトを右クリックして、[プロパティ] を選択します。[ColdFusion サーバーの設定] に移動して、コンピュータにインストールされている Web ブラウザを選択します。
- (Mac の場合) [環境設定] ダイアログボックスで、[一般]-[Web ブラウザ] を選択して、コンピュータにインストールされている Web ブラウザを選択します。

## ブレークポイントの設定と削除

ブレークポイントを使用すると、アプリケーションの実行の制御ができます。これにより、コードの詳細を検査して、アプリケーションをデバッグできます。ブレークポイントを追加するにはコードエディタを使用します。追加したブレークポイントは [ブレークポイント] ビューで管理できます。ブレークポイントは、コードを記述するときや、デバッグするときにも設定できます。

[ブレークポイント] ビューでブレークポイントを管理する方法については、27 ページの「[\[ブレークポイント\] ビュー](#)」を参照してください。

## ColdFusion ブレークポイント

ColdFusion デバッガの ColdFusion ブレークポイントには、次の 4 つの状態があります。

**使用可能で有効** この状態は、ブレークポイントが有効な位置にあることを示します。ブレークポイントは、CFML エディタの左マージンにべた塗りの青い円として表示されます。このブレークポイントに到達するとコードの実行が停止します。

**未解決** ColdFusion のメモリにロードされたページにはブレークポイントが設定されます。そのページを変更した後に実行していない場合は、サーバー上のページとソースの間で同期が失われます。この状況では、ブレークポイントを設定する行が有効であるかどうかを ColdFusion が判断できない場合があります。このタイプのブレークポイントは疑問符 (?) で表されます。

**無効** ColdFusion Builder で編集されている CFML がメモリ内の CFML と同一であり、無効な行にブレークポイントが設定されていると ColdFusion で判断された場合、そのブレークポイントは赤い X 印で表されます。

**使用不可** この状態は、ブレークポイントが使用不可であることを示します。

## コードエディタでのブレークポイントの設定

- 1 ブレークポイントを設定するコードを含んでいる ColdFusion Builder プロジェクトを開きます。
- 2 ブレークポイントを設定するコード行を探し、次のいずれかの操作を行います。
  - エディタの左端にあるマーカーバーをダブルクリックします。

- [実行]-[ブレークポイントの切り替え]または[実行]-[行ブレークポイントの切り替え]を選択します。

**注意:** ColdFusion Builder では、メソッドブレークポイントの切り替えと監視ポイントの切り替えはサポートされていません。

- 3 CFML ファイル内にブレークポイントを設定すると、特定の場所でページの実行を停止できます。特定の行にブレークポイントを設定すると、その行の直前で CFML の実行が停止します。たとえば、次の CFML ページの 3 行目にブレークポイントを設定した場合、<cfset myName = "Wilson"> の前で実行が停止します。

```
<cfset yourName = "Tuckerman">
<cfoutput>Your name is #yourName#.</cfoutput>
<cfset myName = "Wilson"
```

ブレークポイントが設定された行の頭に青のドットが表示されます。

ColdFusion Builder デバッグパースペクティブの [ブレークポイント] ビューで、現在のプロジェクトに設定されているブレークポイントのリストを確認することもできます。

## コードエディタでのすべてのブレークポイントのスキップ

コード内にブレークポイントを設定した後で、デバッグ時に、それらのブレークポイントをすべて無視することもできます。

- 1 ブレークポイントを設定したコードを含んでいる ColdFusion Builder プロジェクトを開きます。
- 2 [ブレークポイント] ビューのツールバーで、[すべてのブレークポイントをスキップ]を選択します。

## コードエディタでのブレークポイントの削除

❖ エディタの左端にあるマーカーバーで、既存のブレークポイントをダブルクリックします。

選択したブレークポイントが、マーカーバーと、ColdFusion Builder デバッグパースペクティブの [ブレークポイント] ビューから削除されます。

ファイルからすべてのブレークポイントを削除するには、メインツールバーのメニューから [実行]-[すべてのブレークポイントを除去]を選択します。

## デバッグセッションの開始

- 1 [ナビゲーター] ビューで、デバッグするプロジェクトまたはファイルを選択します。
- 2 次のいずれかの方法で、デバッグセッションを開始できます。
  - [実行]-[デバッグ]を選択します。
  - ワークベンチツールバーで  をクリックします。
  - プロジェクトを右クリックし、[デバッグ]-[ColdFusion アプリケーション]を選択します。

デバッグの起動設定が自動的に作成されて起動されます。

**注意:** ページのデバッグ中にそのページを閲覧しようとしたり更新しようとしたりすると、デバッグが予期しない動作をする可能性があります。

## デバッグセッションの管理

[デバッグ] ビューを使用すると、アプリケーションのデバッグの制御、アプリケーションの一時停止、再開、終了、コードへのステップイン、ステップオーバーなどを行えます。

ColdFusion デバッグパースペクティブの各種ビューの詳細については、25 ページの「[ColdFusion デバッグパースペクティブ](#)」を参照してください。

## 行単位でのコードの実行

[ ステップイン ]、[ ステップオーバー ]、[ ステップリターン ] ボタンを使用して、行単位で CFML アプリケーションを実行することができます。

ステップを正しく機能させるには、コンパイル済みクラスのキャッシュをクリアします。これを行うには、以前のバージョンの ColdFusion でコンパイルした CFML ページをすべて再コンパイルします。

サイズの大きいファイルでは、ステップやブレイクポイントの動作が遅くなる場合があります。パフォーマンスを上げるには、次のようにします。

- 1 ColdFusion Builder で [ ウィンドウ ]-[ 設定 ] を選択します。
- 2 ツリービュー構造で、[ ColdFusion ]-[ デバッグの設定 ] を選択します。
- 3 情報を収集する必要がないスコープを選択解除します。

### ステップイン

ステップインは、UDF、CFC、カスタムタグ、およびプロジェクトに含まれるファイルに対して使用します。

cfset タグなどの CFML タグに対してはステップインを使用しないでください。ステップインを使用すると、ステップオーバーよりもパフォーマンスに負荷がかかります。

関数、タグ、ファイルにステップインする場合は、開いているプロジェクトのいずれかでファイルが表示されている必要があります。ステップイン対象のファイルは、開いているプロジェクト内に存在する必要があります。

### ステップオーバー

ステップオーバーは、プロジェクトに含まれるファイル (UDF や CFC など) をスキップして CFML アプリケーションの処理を進めるときに使用します。

### ステップリターン

ステップリターンは、プロジェクトに含まれるファイル (UDF や CFC など) から呼び出し元のページに戻るときに使用します。

## 変数の状態の確認

CFML コードを実行するときには、[ 変数 ] ビューに変数の値とスコープを表示できます。[ 変数 ] ビューに表示されるのは、[ 設定 ] ダイアログボックスで選択したスコープに属している変数のみです。[ 変数 ] ビューの使用方法の詳細については、26 ページの「[ 変数 ] ビュー」を参照してください。

## 監視式

別の関数にステップインしたときに、スコープから外れる可能性がある重要な変数を監視するときは、監視式を使用すると便利です。独自の式を作成して、監視や評価を行うことができます。デバッグセッション中に、式を変更することもできます。

[ 式 ] ビューでは、次の操作を行うことができます。

- 監視式を作成するには、右クリックメニューから [ 監視式を追加 ] を選択します。[ 監視式を追加 ] ダイアログボックスが開いたら、式を入力できます。
- 追加した監視式を無視するには、式を右クリックし、[ 使用不可にする ] を選択します。
- 監視式を編集するには、式を右クリックし、[ 監視式を編集 ] を選択します。その後、式を変更できるようになります。

[ 式 ] ビューの使用方法の詳細については、27 ページの「[ 式 ] ビュー」を参照してください。

## リモートアプリケーションのデバッグ

ColdFusion をリモートサーバーで実行している場合は、リモートサーバー接続を設定し、ColdFusion と ColdFusion Builder の間のマッピングを指定します。マッピングを指定すると、ColdFusion Builder と ColdFusion の間で同じプロジェクトまたはファイルのコピーが処理されるようになります。

- 1 リモートサーバーの設定については、51 ページの「[リモートサーバーの作成](#)」を参照してください。必要に応じて、RDS 設定情報を入力します。
- 2 [サーバー] ビューでサーバーを右クリックし、[サーバーの編集] を選択します。サーバーウィザードの [マッピング] 画面で、次のマッピング情報を入力します。
  - ローカルパス: リモート ColdFusion サーバー上でプロジェクトまたはフォルダを探すときに ColdFusion Builder で使用するパス。
  - リモートパス: リモート ColdFusion サーバー上にあるプロジェクトまたはフォルダのパス。

**注意:** リモート ColdFusion サーバーと ColdFusion Builder の間では、複数のマッピングを指定できます。

たとえば、"D:¥MyCoolApp" を参照する ColdFusion プロジェクト内のファイルを編集していて、それに対応するファイルがリモートサーバーの "D:¥Shared¥websites¥MyCoolSite" にあるとします。この場合は、ローカルパスとして "D:¥MyCool App" を指定し、リモートパスとして "D:¥Shared¥websites¥MyCoolSite" を指定してマッピングを作成します。

ColdFusion サーバーにファイルをデプロイするときは "W:¥websites¥MyCoolSite¥" にファイルをコピーします。ColdFusion サーバーはこのディレクトリを "D:¥Shared¥websites¥MyCoolSite" として認識します。

ColdFusion Builder のマッピングで、ColdFusion Builder ディレクトリは "D:¥MyCoolApp" であり、サーバーは "D:¥Shared¥websites¥MyCoolSite" であると指定します。これにより、ColdFusion Builder は、ファイルパス (D:¥MyCoolApp¥index.cfm) を、ColdFusion サーバーが認識するパス (D:¥Shared¥websites¥MyCoolSite¥index.cfm) に変換します。

クライアントとサーバー間の通信の詳細を確認するには、ColdFusion Administrator の JVM 引数に次を追加します。

```
-DDEBUGGER_TRACE=true
```

- 3 [ナビゲーター] ビューで、デバッグするプロジェクトまたはファイルを選択し、ワークベンチツールバーで  をクリックします。

## 第 8 章：ColdFusion Builder 拡張機能

### 拡張機能について

ColdFusion Builder IDE (Integrated Development Environment: 統合開発環境) の機能は、各種の ColdFusion フレームワークをサポートしたり、コード生成に必要な条件を満たすように拡張することができます。ColdFusion Builder 拡張機能を開発すると、コードを生成したり、動的なユーザーインターフェイスを設計したり、データベースに対する基本的な CRUD (作成、読み込み、更新、削除) 操作を実行できます。ColdFusion Builder 拡張機能は、ColdFusion Builder IDE にコンテキストメニューを追加し、それらのメニューに対するイベントを処理する構造化コンポーネントです。

ColdFusion 開発者は、独自の ColdFusion Builder 拡張機能を開発することも、既存の拡張機能をインストールすることもできます。

独自の ColdFusion Builder 拡張機能を開発するには、設定ファイルおよびハンドラファイルを作成します。コンテキストメニューとイベントは、設定ファイル (IDE\_Config.xml) で定義します。詳細については、73 ページの「[拡張機能の開発](#)」を参照してください。

ColdFusion Builder では、ユーザーインターフェイスを作成するなど、さまざまなレベルで IDE を拡張できます。詳細については、83 ページの「[拡張機能のユーザーインターフェイスの作成](#)」を参照してください。

ColdFusion Builder に出荷時から付属する拡張機能には、Adobe CFC Generator、Model-Glue Assistant、および AS Class Generator があります。これらの拡張機能はいつでもインストールして、ColdFusion Builder に統合できます。詳細については、91 ページの「[拡張機能の使用](#)」を参照してください。

ColdFusion Builder には、拡張機能の作成とパッケージ化に必要な手順をガイドする Adobe Extension Builder が用意されています。詳細については、89 ページの「[Adobe Extension Builder を使用した拡張機能の作成とパッケージ化](#)」を参照してください。

### 拡張機能の開発

ColdFusion Builder 拡張機能を開発するには、次の要素を作成します。

- 設定ファイル (IDE\_config.xml)
- ハンドラファイル (CFM ファイル)

これらの要素を作成するには、手動でコードを記述するか、Adobe Extension Builder ウィザードを使用します。Adobe Extension Builder ウィザードの使用法の詳細については、89 ページの「[Adobe Extension Builder を使用した拡張機能の作成とパッケージ化](#)」を参照してください。

### 設定ファイル

設定ファイル (IDE\_config.xml) の作成は、ColdFusion Builder 拡張機能を開発する上で重要な手順となります。設定ファイルの要素の定義はすべて、application タグ内で行います。

#### メタデータ要素の指定

拡張機能を作成し、拡張機能の情報 (名前、作成者、バージョン、説明など) を指定するには、メタデータ要素を使用します。

設定ファイルでアプリケーションメタデータを指定するには、次の要素を使用します。

要素	説明
name	拡張機能の名前。
author	作成者の名前。
version	ファイルのバージョン。
email	電子メールアドレスを指定します。
description	アプリケーションの簡単な説明。  この説明はプレーンテキストで記述できます。また、アプリケーションの説明が記載された HTML ファイルのパスを指定することもできます。HTML ファイルのパスを指定する場合は、拡張機能内の "Install" ディレクトリにその HTML ファイルを保存します。
license	拡張機能をインストールするときに表示されるライセンス契約の条項。  ライセンス契約の条項はプレーンテキストで表示できます。また、ライセンス契約の条項が記載された HTML ファイルのパスを指定することもできます。HTML ファイルのパスを指定する場合は、拡張機能内の "Install" ディレクトリにその HTML ファイルを保存します。

**例**

```
<application>
  <name>ORM CFC Generator</name>
  <author>Adobe</author>
  <version>1.0</version>
  <email>user@xyz.com</email>
  <description>ORM CFC code Generator</description>
  <license>license.html</license>
</application>
```

**ColdFusion Builder 拡張機能のインストールウィザードへのページの追加**

設定ファイルを定義する際には、ColdFusion Builder 拡張機能のインストールウィザードに画面を追加するコードを指定できます。これらの画面を使用すると、ユーザーに入力を求めることができます。通常、インストール後の設定タスクを実行するには、ユーザーの入力が必要です。

入力の詳細を指定するには input タグを使用します。input タグを指定する方法については、78 ページの「[入力型の指定](#)」を参照してください。これらの入力詳細に基づき、wizardtag の handlerid 属性で指定されているハンドラが呼び出されます。また、wizard タグの height 属性と width 属性を使用して、インストールウィザードの高さと幅を指定することもできます。このウィザードの各ページのタイトルを指定するには、page タグの title 属性を使用します。

**シンタックス**

```
<application>
<name>Name of the ColdFusion Builder extension</name>
<install>
<wizard height="" width="" handlerid="handlerID" >
<page title="Wizard page title" >
<input name="Input" ... />
<input name="Input" ... />
</page>
</wizard>
</install>
</application>
```

**例**

次の例では、拡張機能のインストールウィザードによる拡張機能のインストールが終了した後に、指定された入力詳細に基づいてハンドラ ID postinstallhandler が呼び出されます。

```

<application>
<name>ORM CFC Generator</name>
<install>
<wizard height="" width="" handlerid="postinstallhandler" >
<page title="Install settings" >
<input name="Mapping" ... />
<input name="Datasource" ... />
</page>
</wizard>
</install>
</application>

```

## IDE の拡張

ColdFusion Builder IDE は次のレベルで拡張できます。

- コンテキストメニューの追加: コンテキストメニューは次のビューに追加できます。
  - リソースナビゲーター
  - RDS データビュー
  - [アウトライン] ビュー
  - CFML エディタ
- ワークスペースイベントの処理: 現時点でサポートされているイベントは `onprojectcreate` のみです。

CRUD またはコード生成操作を実行するコンテキストメニューおよびイベントのハンドラを指定できます。

## ハンドラの指定

ColdFusion Builder は、CFM ハンドラをサポートしています。ColdFusion Builder の場合、ハンドラとはイベントやアクションに対する応答として実行されるコードを含んだファイルのことです。ハンドラの指定は `handlers` タグ内で行います。ハンドラをイベントまたはアクションに関連付けるには、`handlerid` 属性を使用します。ハンドラの詳細はすべて `handlers` タグ内で指定します。イベントおよびアクションを指定する方法の詳細については、76 ページの「[イベントの指定](#)」および 76 ページの「[コンテキストメニューの指定](#)」を参照してください。ハンドラファイルはすべて、"Handlers" フォルダに保存する必要があります。

## シンタックス

```

<handlers>
  <handler id ="cfm" type="cfm" filename="filename" />
</handlers>

```

属性	説明
id	ハンドラ ID
type	ハンドラタイプを指定します。 指定できるハンドラタイプは "CFM" です。
filename	CFM ファイルの名前

**注意:** 属性値として使用できる文字は英数字のみです。特殊文字は指定できません。

**例**

```
<handlers>
  <handler id="cfcgenerator" type="CFM"
    filename="ormCFCGenerator.cfm" />
  <handler id="gridgenerator" type="CFM"
    filename="cfgridGenerator.cfm" />
</handlers>
```

**イベントの指定**

イベントを指定するには `events` タグを使用します。現在 ColdFusion Builder でサポートされているイベントは、プロジェクト作成に関するイベントのみです。サポートされているイベントのタイプは `onprojectcreate` です。`onprojectcreate` イベントのハンドラ ID は必須です。イベントが発生すると、ハンドラ ID に関連付けられているハンドラが呼び出されます。

`onprojectcreate` イベントが指定されているアプリケーションは、56 ページの「[ColdFusion プロジェクトの作成](#)」にあるアプリケーションのリストにすべて記載されています。リストに記載されているアプリケーションのいずれかをユーザーが選択すると、そのアプリケーションに関連付けられているハンドラが呼び出されます。`onprojectcreate` イベントは、特定の ColdFusion Builder 拡張機能の基本的なプロジェクト構造を作成するときに役立ちます。

**シンタックス**

```
<events>
  <event
    type="onprojectcreate"
    handlerid="handler id" />
</events>
```

属性	説明
type	ハンドラを実行するイベントを指定します。 指定できるイベントタイプは <code>type="onprojectcreate"</code> です。
handlerid	渡すハンドラ ID を指定します。

**注意：**属性値として使用できる文字は英数字のみです。特殊文字は指定できません。

**例**

```
<events>
  <event type="onprojectcreate"
    handlerid="projectCreationHandler" />
</events>
```

**コンテキストメニューの指定**

コンテキストメニューを指定するには `menucontributions` タグを使用します。コンテキストメニューとは、右クリックイベントで表示されるポップアップメニューのことです。

各種ビューのメニューコントリビューションを指定するには、`contribution` タグを使用します。メニューを表示するビュー (ターゲットビュー) を指定するには、`contribution` タグ内で `target` 属性を使用します。追加するメニューを指定するには、`contribution` タグ内で `menu` タグを使用します。メニューアイテムを指定するには、`menu` タグ内で `action` タグを使用します。

## シンタックス

```
<menucontributions>
  <contribution target="rdsview|projectview|outlineview">
    <menu name="name of the menu item">
      <action name="action name">
        </action>
      </menu>
    </contribution>
  </menucontributions>
```

### contribution

属性	説明
target	ターゲットビューとして指定できる値は次のいずれかになります。 <ul style="list-style-type: none"> <li>• rdsview</li> <li>• projectview</li> <li>• outlineview</li> <li>• editor</li> </ul>

**注意：**属性値として使用できる文字は英数字のみです。特殊文字は指定できません。

### menu

属性	説明
name	メニューの名前を指定します。

**注意：**属性値として使用できる文字は英数字のみです。特殊文字は指定できません。

### action

属性	説明
name	アクションの名前を指定します。
handlerid	関連付けられているハンドラの ID を指定します。
showresponse	ハンドラが受信した HTML レスポンスをユーザーに表示するかどうかに応じて、次のブール値を指定します。 <ul style="list-style-type: none"> <li>• yes</li> <li>• no</li> </ul> デフォルト値は false です。

**注意：**属性値として使用できる文字は英数字のみです。特殊文字は指定できません。

### メニューフィルタの指定

メニューフィルタを使用すると、メニューやメニューアイテムをどこに表示するかを簡単に制御できます。メニューフィルタを指定するには、`filters` タグを使用します。`filters` タグ内では、さまざまなタイプのフィルタを指定できます。メニューやメニューアイテムをどこに表示するかを指定するには、`type` 属性と `pattern` 属性を使用します。指定したフィルタのいずれかに一致すると、指定したメニューまたはメニューアイテムが表示されます。

フィルタはメニューとアクションの両方に対して指定できます。menu タグ内で filter タグを指定すると、そのフィルタによってメニューの表示が制御されます。たとえば、次のコードでは、ユーザーが "modelglue.xml" ファイルを右クリックした場合にのみ、ORM Code Generator のメニューが表示されます。

```
<menu name="ORM Code Generator">
<filters>
<filter type="file" pattern="modelGlue.xml" />
</filters>
</menu>
```

### [ナビゲーター] ビューに対するフィルタの指定

[ナビゲーター] ビューのプロジェクト、フォルダ、またはファイルに対してフィルタを指定できます。

#### シンタックス

```
<filter type="folder|project|file"
pattern="regular expression to match folder, project, or filename" />
```

#### 例

フォルダのコンテキストでのみ ORM Code Generator のメニューが表示されるようにするには、次のようなコードを使用します。

```
<menu name="ORM Code Generator">
<filters>
<filter type="folder" />
</filters>
</menu>
```

### [アウトライン] ビューに対するフィルタの指定

[アウトライン] ビューのさまざまなノードタイプに対してフィルタを指定できます。指定したノード名がフィルタとして機能します。[アウトライン] ビューでは、ファイルに対してもフィルタを指定できます。CFML エディタでフィルタと一致するファイルを開くと、コントリビュートされたメニューが [アウトライン] ビューに表示されます。

#### シンタックス

```
<menu name="menu name">
<filters>
<filter type="node name" />
</filters>
</menu>
```

#### 例

[アウトライン] ビューの CFfunction ノードでのみ ORM Code Generator のメニューが表示されるようにするには、次のようなコードを使用します。

```
<menu name="ORM Code Generator">
<filters>
<filter type="cffunction" />
</filters>
</menu>
```

### 入力型の指定

ハンドラを呼び出す前に input タグを使用すると、ユーザーに入力を求めることができます。ユーザーの入力は、関連付けられているハンドラによって処理されます。アクションごとに input タグを指定し、input タグを action タグ内に含める必要があります。

入力ダイアログの高さと幅を制御するには、dialog タグ内で input タグを指定します。たとえば、次のようなコードを指定できます。

```
<dialog height="400" width="600" title="title" image="path to the image file">
<input name="Mapping" ... />
<input name="Datasource" ... />
</dialog>
```

属性	説明
height	ダイアログボックスの高さを指定します。
width	ダイアログボックスの幅を指定します。
title	ダイアログボックスのタイトルを指定します。
image	タイトルバーに表示されるイメージのパスを指定します。ここでは、"Extension" フォルダからの相対パスを指定する必要があります。

### シンタックス

```
<action name="action name">
  <input name="input variable name"
    label="label for the input dialog box"
    tooltip="tool tip"
    type="dir|string|boolean|file|password|list"/>
</action>
```

属性	説明
name	入力変数の名前
label	入力ダイアログボックスのラベル
tooltip	ユーザーがマウスを入力ダイアログボックスの上に移動したときに表示するツールヒント。
type	入力変数として受け入れられるデータ型は次のいずれかになります。 <ul style="list-style-type: none"> <li>• dir</li> <li>• string</li> <li>• Boolean</li> <li>• file</li> <li>• password</li> <li>• list</li> <li>• projectdir</li> <li>• projectfile</li> </ul>
required	必須の入力フィールドである場合に指定します。 入力フィールドを必須フィールドとして指定した場合、ユーザーがその必須フィールドに値を入力するまで、[OK] ボタンは有効になりません。
pattern	ユーザー入力の検証に使用する正規表現を指定します。 検証エラーの場合に表示するエラーメッセージも指定できます。エラーメッセージを指定するには、errormessage 属性を使用します。
errormessage	特定のパターンで検証が失敗したときに表示するエラーメッセージ。

属性	説明
helpmessage	特定の入力フィールドに関してダイアログボックスのタイトル領域に表示するヘルプヒント。
default	特定の入力型のデフォルト値を指定します。入力型が Boolean の場合、デフォルト値は指定できません。リストの場合、デフォルト値はあらかじめ選択されています。
checked	チェックボックスフィールドをデフォルトでオンにするかオフにするかを指定するブール値。 <ul style="list-style-type: none"> <li>• true - チェックボックスをオンにします (これがデフォルト値です)。</li> <li>• false - チェックボックスをオフにします。</li> </ul>

**注意：**属性値として使用できる文字は英数字のみです。特殊文字は指定できません。

各データ型は、次の表に示す入力型に対応しています。各入力型のシンタックスも示します。

データ型	入力型	シンタックス
dir	ディレクトリ選択フィールド	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="dir"/>
string	テキストフィールド	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="string"/>
boolean	チェックボックス	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="boolean"/>
file	ファイル選択フィールド	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="file"/>
password	パスワードフィールド	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="password"/>
list	リストフィールド	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="list"> <option value=" オプション 1"> <option value=" オプション 2"> </input>
projectdir	プロジェクトディレクトリ選択フィールド	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="projectdir"/>
projectfile	プロジェクトファイル選択フィールド	<input name=" 入力変数名 " label=" ラベル " tooltip=" ツールヒント " type="projectfile"/>

**例**

```

<menucontributions >
  <contribution target="rdsview" >
    <menu name="ORM Code Generator">
      <action name="Generate ORM CFC" handlerid="cfm1" >
        <input name="Location" label="Enter location"
        tooltip="Location where generated CFCs will be stored" type="dir"/>
        <input name="generateAppCFC" label="Generate
        Application CFC" tooltip="Generate Application CFC along with ORM CFC"
        type="boolean"/>
        <input name="generateView" label="Generate View"
        tooltip="Generate View template along with ORM CFC" type="boolean"/>
      </action>
      <action name="Generate Ajax Grid" handlerid="cfm2" >
        <input name="Location" label="Enter location"
        tooltip="Location where generated View will be stored" type="dir"/>
      </action>
    </menu>
  </contribution>
  <contribution target="projectview" >
    <menu name="ORM Code Generator">
      <action name="Generate ORM CFC" handlerid="cfm1" />
      <action name="Generate Ajax Grid" handlerid="cfm2" />
    </menu>
  </contribution>
  <contribution target="outlineview" >
    <menu name="ORM Code Generator">
      <action name="Generate ORM CFC" handlerid="cfm1" />
      <action name="Generate Ajax Grid" handlerid="cfm2" />
    </menu>
  </contribution>
</menucontributions>

```

**キーワードのサポート**

ColdFusion Builder では、入力ダイアログボックスにデフォルト値を追加するためのキーワードがサポートされています。これらのキーワードは実行時に、実際の値に置き換えられます。

次の表に、サポートされているキーワードおよび対応する実際の値を示します。

キーワード	実際の値
projectlocation	プロジェクトの場所の絶対パス
projectname	プロジェクトの名前
serverhome	サーバーのインストール場所の絶対パス
wwwroot	サーバーの Web ルート

たとえば、projectname キーワードは次のように指定できます。

```
<input name="projectname" label="projectname" default="{projectname}" type="dir"/>
```

拡張機能のインポート、リロード、パッケージ化、およびデバッグの方法については、88 ページの「[\[ 拡張機能 \] ビューの使用](#)」を参照してください。

## ColdFusion Builder とハンドラの通信について

ハンドラと ColdFusion Builder IDE の間の通信には XML が使用されます。イベントが発生すると、ColdFusion Builder からそのイベントに関連付けられているハンドラへ、イベントの詳細が XML 形式で送信されます。この XML は `ideEventInfo` という名前の FORM スコープ変数としてハンドラに送信されます。FORM スコープ変数から XML を取得するには、ハンドラで `cfparam` タグを使用します。

### シンタックス

```
<cfparam name="ideeventinfo">
```

または

```
<cfset ideData= form.ideEventInfo >
```

### XML の構造について

ここでは実際の XML コードを例に取り、XML の構造について説明します。

コードを見る前に、次のポイントに注意してください。

- ハンドラに送信される情報はすべて `event` タグ内にあります。
- イベントおよびアクションの情報はすべて `ide` タグ内にあります。
- ユーザー入力はすべて `user` タグ内にあります。

次の表で、各種の XML コードについて説明します。XML コードの内容は、イベントによって異なります。

コード	コンテキスト	説明
<pre>&lt;rdsview &gt; &lt;database name=""&gt; &lt;table name=""&gt; &lt;fields&gt; &lt;field name="" type="" length="" nullallowed="" primarykey="" /&gt; &lt;/fields&gt; &lt;/table&gt; &lt;/database&gt; &lt;/rdsview&gt;</pre>	rdsview	RDS ビューのメニューアイテムをクリックすると、関連付けられているハンドラにイベントの詳細が送信されます。
<pre>&lt;projectview projectname="" projectlocation=""&gt; &lt;resource name="" path="" type="file/folder/project" /&gt; &lt;/projectview&gt;</pre>	projectview	プロジェクトビューのメニューアイテムをクリックすると、関連付けられているハンドラにイベントの詳細が送信されます。
<pre>&lt;outlineview projectname="" projectlocation=""&gt; &lt;source filename="" path=""&gt; &lt;node type="function/other"&gt; &lt;function name="" returntype=""&gt; &lt;argument name="" type="" /&gt; &lt;/function&gt; &lt;/node&gt; &lt;/source&gt; &lt;/outlineview&gt;</pre>	outlineview	[アウトライン] ビューのメニューアイテムをクリックすると、関連付けられているハンドラにイベントの詳細が送信されます。
<pre>&lt;eventinfo projectname="" projectlocation="" eventtype=""&gt; &lt;resource name="" path="" type="file/folder/project" /&gt; &lt;/eventinfo&gt;</pre>	プロジェクト作成時	指定したイベントが発生すると、関連付けられているハンドラにイベントの詳細が送信されません。
<pre>&lt;user&gt; &lt;input name="" value="" /&gt; &lt;input name="" value="" /&gt; &lt;input name="" value="" /&gt; &lt;/user&gt;</pre>	入力ダイアログボックス	ユーザー入力はすべて <code>user</code> タグ内で指定します。 <code>user</code> タグ内の各 <code>input</code> タグで、具体的なユーザー入力を指定します。入力名は、"IDE_Config.xml" ファイルで指定されている入力名と同じです。
<pre>&lt;user&gt; &lt;page index=""&gt; &lt;input name="" value="" /&gt; &lt;/page&gt; &lt;/user&gt;</pre>	インストールウィザード	<code>user</code> タグ内で指定されたすべてのユーザー入力が、インストールウィザードに関連付けられているハンドラに送信されます。個々のユーザー入力は <code>user</code> タグ内の各 <code>input</code> タグに格納されます。入力名は、"IDE_Config.xml" ファイルで指定されている入力名と同じです。

## 拡張機能のユーザーインターフェイスの作成

### 入力ダイアログボックスの作成

入力ダイアログボックスは、設定ファイル (IDE\_Config.xml) と CFM ページのいずれかを使用して作成できます。

**注意:** XML レスポンスを使用してユーザーインターフェイスを作成する場合は、該当するコードで cfsetting showdebugoutput="no" を指定してデバッグ出力を無効にしてください。

#### 設定ファイルを使用する場合

設定ファイル (IDE\_config.xml) を作成し、必要な項目を指定することによって入力ダイアログボックスを作成します。詳細については、73 ページの「[設定ファイル](#)」を参照してください。

たとえば、次のようなコードを指定できます。

```
<action name="action name">
<dialog>
<input name="input variable name" label="label for the input dialog"
tooltip="tool tip" type="dir"/>
</dialog>
</action>
```

#### CFM ページを使用する場合

- 次のように、"IDE\_config.xml" ファイルで、Action タグの showResponse 属性を yes に設定します。

```
<action name="generate CFC" handlerid="cfcgenerator" showResponse="yes" />
```

- CFM ハンドラ側で次の作業を行います。

- 1 次のように、cfheader タグの content-type の値を text/xml に設定します。

```
<cfheader name="Content-Type" value="text/xml">
```

- 2 次のような XML を作成して入力ダイアログボックスを作成します。

```
<cfoutput>
<response>
<ide>
<dialog >
<input name="location" Label="Enter Location" type="dir" />
</dialog>
</ide>
</response>
</cfoutput>
```

### HTML ユーザーインターフェイスの作成

HTML ユーザーインターフェイスは、HTML レスポンスと XML レスポンスのいずれかを使用して作成できます。

#### HTML レスポンスを使用する場合

次のように、"IDE\_config.xml" ファイルで、Action タグの showResponse 属性を yes に設定します。

```
<action name="generate CFC" handlerid="cfcgenerator" showResponse="yes" />
```

CFM ページの HTML コンテンツがダイアログボックスに表示されます。HTML コンテンツでは、Ajax と Javascript はサポートされません。HTML コンテンツでリンクを使用する場合は、相対 URL ではなく絶対 URL を使用する必要があります。

## XML レスポンスを使用する場合

CFM ハンドラ側で次の作業を行います。

- 1 次のように、`cfheader` タグの `content-type` の値を `text/xml` に設定します。

```
<cfheader name="Content-Type" value="text/xml">
```

- 2 次のような XML を作成して HTML ユーザーインターフェイスを作成します。

```
<cfoutput>
<response showresponse="true">
<ide url="http://localhost:8500/local/Dynamic%20UI%20Test/handlers/main.swf" >
<dialog width="455" height="470" />
</ide>
</response>
</cfoutput>
```

指定した URL の HTML コンテンツがダイアログボックスに表示されます。

次のように、XML レスポンスを使用しても HTML ユーザーインターフェイスを表示できます。

```
<cfoutput>
<response showresponse="true">
<ide >
<dialog width="100" height="400" />
<body>
<![CDATA[
Any HTML content
]]>
</body>
</ide>
</response>
</cfoutput>
```

**注意：** 拡張機能では、CSS および JavaScript の相対パスはサポートされていません。CDATA セクションで指定されている Ajax コードおよび JavaScript コードは動作しません。CDATA セクションで URL を指定する場合は、絶対 URL を使用する必要があります。

## XML レスポンスの構造について

### Response

XML レスポンスは、次の属性を持つ `response` タグ内で定義します。

属性	説明
<code>showresponse</code>	レスポンスをユーザーに表示するかどうかに応じて、次のブール値を指定します。 <ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul>
<code>status</code>	成功またはエラーのダイアログボックスを表示します。 ステータス (成功かエラーか) に応じて、対応するアイコンがダイアログボックスに表示されます。IDE タグで <code>message</code> 属性が指定されている場合のみ適用されます。 この属性は、IDE タグ内で <code>message</code> 属性が指定されている場合のみ適用されます。

## IDE

IDE タグには次の属性があります。

属性	説明
message	成功またはエラーのダイアログボックスに、指定されたメッセージを表示します。メッセージをエラーメッセージとして表示するかどうかを指定するには、response タグの status="success/error" 属性を使用します。
url	ユーザーに表示するページの URL。
handlerfile	ユーザーがメッセージダイアログボックスで [OK] をクリックしたときに呼び出すハンドラファイルのパス。  この属性は、XML レスポンスから入力ダイアログボックスが作成される場合にのみ適用されます。

## Body

body タグは、HTML コンテンツを指定する場合に使用します。body タグ内の HTML コンテンツは、CDATA セクション内で指定する必要があります。

## Dialog

dialog タグは、入力ダイアログボックスを作成する場合に使用します。このタグには次の属性があります。

属性	説明
height	ダイアログボックスの高さを指定します。
width	ダイアログボックスの幅を指定します。
title	ダイアログボックスのタイトルを指定します。
image	タイトルバーに表示されるイメージのパスを指定します。
dialogclosehandler	ユーザーがメッセージダイアログボックスで [閉じる] または [キャンセル] をクリックしたときに呼び出すハンドラファイルのパス。  この属性は、XML レスポンスから入力ダイアログボックスが作成される場合にのみ適用されます。

入力ダイアログの高さと幅を制御するには、dialog タグ内で input タグを使用します。

属性	説明
height	ダイアログボックスの高さを指定します。
width	ダイアログボックスの幅を指定します。
title	ダイアログボックスのタイトルを指定します。
image	タイトルバーに表示されるイメージのパスを指定します。ここでは、"Extension" フォルダからの相対パスを指定する必要があります。

## ハンドラからのコールバックコマンドの指定

ハンドラでは、ColdFusion Builder に対して次のコマンドを実行するように要求できます。

- refreshproject
- refreshfolder

- refreshfile
- inserttext
- openfile

**注意:** filename と foldername の値では、大文字と小文字が区別されます。

### refreshproject

refreshproject コマンドは、指定されたプロジェクトを ColdFusion Builder で更新します。

例

```
<response>
  <ide>
    <commands>
      <command type="refreshproject">
        <params>
          <param key="projectname" value="[valid project name/location]" />
        </params>
      </command>
    </commands>
  </ide>
</response>
```

### refreshfolder

refreshfolder コマンドは、指定されたフォルダを ColdFusion Builder で更新します。

### refreshfile

refreshfile コマンドは、絶対パスまたは相対パスを送信することにより、指定されたファイルを ColdFusion Builder で更新します。相対パスを指定する場合は、プロジェクト名を指定します。

例:

```
<response>
  <ide>
    <commands>
      <command type="refreshfile">
        <params>
          <param key="filename" value="[valid file name/location]" />
          <param key="projectname" value="[valid project name/location]" />
        </params>
      </command>
    </commands>
  </ide>
</response>
```

### inserttext

inserttext コマンドは、指定されたテキストをアクティブなエディタに挿入します。このコマンドを実行すると、エディタで選択されているテキストが置換されます。

例:

```
<response>
  <ide>
    <commands>
      <command type="inserttext">
        <params>
          <param key="text" >
            <![CDATA[ Any text to be inserted in editor ]]>
          </param>
        </params>
      </command>
    </commands>
  </ide>
</response>
```

また、inserttext コマンドは、テキストの折り返しと置換に使用することもできます。

たとえば、starttext と endtext の間で選択したテキストを折り返すには、次のようなコードを使用します。

```
<response>
  <ide>
    <commands>
      <command type="inserttext">
        <params>
          <param key="insertmode" value="wrap" />
          <param key="starttext" >
            <![CDATA[ Any text to be inserted in editor ]]>
          </param>
          <param key="endtext" >
            <![CDATA[ Any text to be inserted in editor ]]>
          </param>
        </params>
      </command>
    </commands>
  </ide>
</response>
```

### openfile

openfile コマンドは、絶対パスまたは相対パスを送信することにより、指定されたファイルを ColdFusion Builder で開いて編集できるようにします。相対パスを指定する場合は、プロジェクト名を指定します。

ハンドラでは、レスポンスごとに、ColdFusion に対して複数のコマンドを実行するように要求することもできます。複数のコマンドを実行する場合、ColdFusion Builder は、受信した順序どおりにコマンドを実行します。

コマンドを実行するためにハンドラから ColdFusion Builder に送信されるレスポンスの XML 形式を次に示します。

```
<response>
  <ide>
    <commands>
      <command type="[command type]">
        <params>
          <param key="" value="" />
        </params>
      </command>
    </commands>
  </ide>
</response>
```

例:

```
<response>
  <ide>
    <commands>
      <command type="openfile">
        <params>
          <param key="filename" value="[valid file name/location]" />
          <param key="projectname" value="[valid project name/location]" />
        </params>
      </command>
    </commands>
  </ide>
</response>
```

## [ 拡張機能 ] ビューの使用

[ 拡張機能 ] ビューでは、拡張機能のインストール、アンインストール、インポート、リロードを行えます。

[ 拡張機能 ] ビューがまだワークベンチに表示されていない場合、[ 拡張機能 ] ビューを追加するには、[ ウィンドウ ]-[ ビューの表示 ]-[ その他 ] を選択します。次に、[ ビューの表示 ] ダイアログボックスで、[ ColdFusion ]-[ 拡張機能 ] ビューを選択します。

### 拡張機能のインストールとアンインストール

1 [ 拡張機能 ] ビューで  をクリックし、インストールするアーカイブファイルを選択します。

**注意：** ColdFusion Builder に付属する拡張機能をインストールする場合は、ColdFusion Builder のインストール先の "Extensions" ディレクトリにアーカイブファイルが保存されています。

2 拡張機能のインストールウィザードに従ってインストール作業を進めます。

3 サーバーを選択します。サーバーがまだ 1 台も設定されていない場合は、[ サーバーの追加 ] をクリックして新しいサーバーを追加します。サーバーを追加する方法については、48 ページの「[ColdFusion サーバーの作成](#)」を参照してください。

複数のサーバーに同じ拡張機能をインストールすることもできます。たとえば、CF9 ローカルサーバーと CF8 ローカルサーバーを選択できます。ただし、アクティブにできるサーバーは、拡張機能によって使用されるサーバーのみです。

4 サーバーを選択するときパスが自動的に挿入されない場合は、ColdFusion の Web ルートのパスを入力します。たとえば、C:\ColdFusion9\wwwroot のように入力します。

5 拡張機能のインストール先となる Web ルート内のフォルダを選択します。アーカイブファイルは、このインストール先に解凍されます。

6 [ インストール ] をクリックします。

インストールされている拡張機能の簡単な説明を表示するには、拡張機能の名前をクリックします。複数のサーバーに同じ拡張機能をインストールした場合、アクティブにできるサーバーは 1 つのみです。これを行うには、アクティブにするサーバーを [ アクティブサーバー ] ドロップダウンリストから選択します。

拡張機能をアンインストールするには、[ 拡張機能 ] ビューで拡張機能を選択し、 をクリックします。拡張機能は、IDE からのみアンインストールすることも、IDE とコンピュータのファイルシステムから同時に削除することもできます。ファイルシステムから拡張機能を完全に削除した後に、元に戻すことはできません。

**注意：** 複数のサーバーにインストールした拡張機能をアンインストールする場合は、各サーバーで同じアンインストール手順を繰り返します。

## 拡張機能のインポートとリロード

[ 拡張機能 ] ビューでは、拡張機能のインポートとリロードを行えます。

- 既存のディレクトリから ColdFusion Builder に拡張機能をインポートできます。[ 拡張機能 ] ビューで  をクリックし、拡張機能ファイルが保存されているディレクトリを選択します。

**注意：**既存の拡張機能を使用して新しい拡張機能を開発するときは、拡張機能をインポートすると便利です。

- インストール済みの拡張機能の設定ファイルを変更した場合は、拡張機能をリロードします。設定ファイルの変更は、拡張機能をリロードしないと反映されません。[ 拡張機能 ] ビューで  をクリックすると、インストール済みの拡張機能がすべてリロードされます。

## 拡張機能のデバッグとパッケージ化

設定ファイルから拡張機能をロードする際にエラーが発生した場合は、Eclipse ログにエラーが記録されます。エラーをデバッグするには、[ ウィンドウ ]-[ ビューの表示 ]-[ その他 ]-[ 一般 ]-[ エラーログ ] を選択して、ColdFusion Builder エラーログを表示します。ColdFusion Builder のエラーログが [ エラーログ ] ビューに表示されます。

拡張機能をパッケージ化するには、次のコンテンツを ZIP ファイルに追加します。

- 設定ファイル (IDE\_config.xml)
- すべてのハンドラ CFM ファイルを含むハンドラディレクトリ

**注意：**設定ファイルとハンドラディレクトリは、ZIP ファイルのルートの直下に追加してください。

## Adobe Extension Builder を使用した拡張機能の作成とパッケージ化

Adobe Extension Builder ウィザードは、拡張機能の作成とパッケージ化の手順を案内します。このウィザードでは、設定ファイル (IDE\_Config.xml) とハンドラファイル (CFM) の作成、メタデータ要素の指定、コンテキストメニューの追加、ユーザーインターフェイスの作成を行います。

**注意：**Adobe Extension Builder ウィザードは、ColdFusion サーバーの設定時に [ 拡張機能のインストール ] を選択した場合にのみインストールされます。インストール済みの拡張機能のリストは、[ 拡張機能 ] ビューで確認できます。

- 1 [ ナビゲーター ] ビューで、拡張機能を作成するプロジェクトを右クリックし、[ Adobe Extension Builder ]-[ New ] を選択します。
- 2 拡張機能を保存する場所を参照して選択するか、手動で入力します。拡張機能と同じ名前のフォルダに拡張機能ファイルを保存するには、[ Create Folder With Extension Name ] を選択します。
- 3 [ Extension Details ] タブを選択して、拡張機能の名前と詳細を入力します。拡張機能のインストール時に表示するライセンス契約のテキストを指定することもできます。

### メニューコントリビューションの指定

コンテキストメニューを指定するには、[ Menu contributions ] タブを選択して [ Add Menu ] をクリックします。コンテキストメニューとは、右クリックアクションで表示されるポップアップメニューのことです。

- メニューの名前と、メニューを表示するターゲットビューを入力します。ターゲットビューとして指定できるのは、ナビゲータビュー、プロジェクトビュー、RDS ビュー、アウトラインビュー、CFML エディタのいずれかです。[ Save Menu ] をクリックします。

- メニューアイテムを指定するには、[Add Action] をクリックします。
  - 1 アクションにはハンドラを関連付けることができます。ハンドラは、サーバーから取得したコード生成操作を表示します。ハンドラ ID とハンドラ名を指定して [Show Response] を選択し、サーバーのレスポンスを表示します。[保存] をクリックします。
  - 2 ハンドラが呼び出される前に、ユーザー入力を取得できます。これを行うには、[Add Input] をクリックします。
  - 3 入力変数名、入力ダイアログボックスのラベル、ツールヒント、および入力変数タイプを指定します。
  - 4 ユーザー入力の検証に使用するパターンと、検証の失敗時に表示するエラーメッセージを指定することもできます。
  - 5 入力フィールドを必須フィールドとして指定するには、[Required] を選択します。デフォルトでチェックボックスをオンにするには、[Checked] を選択します。[Save] をクリックします。
- メニューの表示を制御するためのフィルタを指定するには、[Add Filter] をクリックします。フィルタは、プロジェクトレベル、フォルダレベル、およびファイルレベルで指定できます。

### ハンドラの指定

インストール済みのハンドラに関する詳細を表示するには、[Handlers] タブを選択します。

- 1 新しいハンドラを指定するには、[Add Handler] をクリックし、必要な詳細を入力します。
- 2 ハンドラを呼び出す前にユーザー入力を取得するには、[Generates Response] を選択します。
- 3 入力ダイアログボックスのタイトルを指定し、高さと幅をピクセル単位で指定します。[Save Handler] をクリックします。

### 拡張機能のインストールウィザードの作成

拡張機能のインストール手順を案内するインストールウィザードを作成するには、[Install Wizards] タブを選択します。

- 1 ウィザードの高さと幅をピクセル単位で指定します。ウィザードにハンドラを関連付けることもできます。[Save] をクリックします。
- 2 ウィザードのタイトルを指定して、[Save] をクリックします。
- 3 入力変数名、入力ダイアログボックスのラベル、ツールヒント、および入力変数タイプを指定します。ユーザー入力の検証に使用するパターンと、検証の失敗時に表示するエラーメッセージを指定することもできます。

### 拡張機能のパッケージ化

- 1 [ナビゲーター] ビューで、"ide\_config.xml" ファイルと、すべてのハンドラ CFM ファイルを含む "Handler" フォルダを右クリックし、[Adobe Extension Builder]-[Package] を選択します。これらのファイルが ZIP ファイルに追加されます。
- 2 ZIP ファイルを保存する場所を参照して選択します。

## 第9章：拡張機能の使用

サーバーの作成時に [ 拡張機能のインストール ] オプションを選択した場合は、ColdFusion Builder とともに次の拡張機能が自動的にインストールされます。詳細については、48 ページの「[ColdFusion サーバーの作成](#)」を参照してください。

- Adobe CFC Generator
- ActionScript Class Generator
- Model-Glue Assistant

ここでは、これらの拡張機能の使用方法について説明します。拡張機能の開発方法については説明しません。

**注意：**拡張機能のユーザーインターフェイスが、コンピュータのデフォルトブラウザに表示されます。Mozilla をブラウザとして指定する場合は、[ 環境設定 ] ダイアログボックスに移動して、[ColdFusion]-[ 拡張機能 ] を選択します。[ 拡張機能に Mozilla ブラウザを使用します ] を選択します。

### Adobe CFC Generator

Adobe CFC Generator ジェネレータは、データベーステーブルから ORM CFC (ColdFusion コンポーネント) を生成します。Adobe CFC Generator 拡張機能を使用すると、従来の CFC または ORM CFC を一連のテーブルから作成できます。

ORM (Object Relational Mapping: オブジェクトリレーショナルマッピング) は、オブジェクトモデルを使用してリレーショナルデータベースのデータを格納および取得するためのマッピング方針を定義するプログラミング手法です。Adobe CFC Generator では、関係のマッピングを定義し、結合条件を指定し、CFC のサービスの生成できます。ORM の詳細については、『ColdFusion アプリケーションの開発』の ColdFusion ORM を参照してください。

### Adobe CFC Generator のインストール

サーバーの作成時に [ 拡張機能のインストール ] オプションを選択した場合は、ColdFusion Builder とともに Adobe CFC Generator 拡張機能がインストールされます。

サーバーの作成時に拡張機能をインストールしなかった場合は、88 ページの「[拡張機能のインストールとアンインストール](#)」の説明に従って拡張機能をインストールできます。Adobe CFC Generator 拡張機能のアーカイブファイル (AdobeCfcGenerator.zip) は、ColdFusion Builder に出荷時から付属しています。このアーカイブファイルは、ColdFusion Builder のインストール先の "Extensions" ディレクトリにあります。

### Adobe CFC Generator の使用

- 1 ColdFusion Builder パースペクティブで、[RDS データビュー] を選択します。
- 2 RDS ユーザー名とパスワードの設定が既に完了している場合は、その RDS ユーザー名とパスワードを指定します。  
**注意：**RDS パスワードの設定は ColdFusion Administrator で行います。同じく ColdFusion Administrator で管理される ColdFusion Administrator パスワードと RDS パスワードを混同しないでください。
- 3 設定したサーバーで利用可能なデータソースのリストが表示されます。
- 4 CFC を生成するデータソースおよびテーブルを選択します。Shift キーを押しながらテーブル名を選択すると、一度に複数のテーブルを選択できます。
- 5 選択したテーブルを右クリックし、[Adobe CFC Generator] を選択して、次のいずれかのオプションを選択します。

**注意：** [Adobe CFC Generator] オプションは、テーブル名を右クリックした場合にのみ表示されます。データソース自体またはデータソース内のレコードをクリックしても、これらのオプションは表示されません。

**Create CFC** ColdFusion コンポーネントを生成します。

**Create ORM CFC** ORM CFC を生成してデータベースにマッピングします。

## Create CFC

1 選択したデータソース内のテーブル (APP.EMPLOYEES テーブルなど) を右クリックし、[Adobe CFC Generator]-[Create CFC] を選択します。

2 [Adobe CFC Generator] ダイアログボックスで次の操作を行います。

- 生成された CFC を保存する場所を指定します。
- [Generate Service] を選択すると、メソッドが自動的に生成されます。

追加の機能を利用できるように CFC のソースコードを変更することもできます。

## Create ORM CFC

1 選択したデータソース内のテーブル (APP.EMPLOYEES テーブルなど) を右クリックし、[Adobe CFC Generator]-[Create ORM CFC] を選択します。

2 生成された CFC を保存する場所を指定します。

3 [Adobe CFC Generator] ダイアログボックスのダイナミックデータグリッドに、選択したデータベーステーブルとフィールドが表示されます。このグリッドではデータの編集やソートが可能です。

**注意：** このデータグリッドは Ajax のリッチユーザーインターフェイスであるため、ロードに時間がかかる場合があります。

ORM CFC は、次のようにカスタマイズできます。

- CFC 名の変更: 自動生成された CFC 名は、[CFC Name] フィールドで編集できます。
- プロパティ名の変更: プロパティ名は、[Property Name] フィールドで編集できます。
- プロパティタイプの変更: プロパティタイプを編集するには、[Property Type] 列に表示されるプロパティタイプをクリックし、ドロップダウンリストからプロパティタイプを選択します。
- プライマリキーの指定: 必要な列で [True] または [False] を選択することにより、プライマリキーの設定を編集できます。
- それぞれの列名の横にあるチェックボックスをオンまたはオフにすることにより、CFC で列を生成するか除外するかを選択できます。
- データのソート: データグリッドに表示されているテーブルと列をソートできます。ソートの方向は、昇順または降順です。テーブル名をソートするには、[Table] 列をクリックして、ソート順を選択します。列データをソートするには、列見出しをクリックして、ソート順を選択します。

4 基本的な ORM CFC を生成するには、[Generate Code] リンクをクリックします。

ORM CFC の関係および結合条件を定義するには、次のようにします。

- a [Relationships] データグリッドで [Insert] をクリックします。新しい行が挿入されます。
- b [Relation Name] フィールドをクリックして、関係の名前を入力します。たとえば、"Art" テーブルと "Artists" テーブルの関係を定義できます。
- c [Target Table] ドロップダウンリストからターゲットテーブルを選択します。
- d [Multiplicity] ドロップダウンリストを選択して、関係タイプを指定します。

- e 必要に応じてリンクテーブルを指定します。
  - f [Save] をクリックします。
  - g 関係を定義するときは、結合条件を定義します。指定した関係の結合条件を定義するには、手順 3 で選択したターゲットテーブルをクリックします。
  - h [Join Conditions] データグリッドで [Insert] をクリックします。
  - i [Source] ドロップダウンリストからソースフィールドを選択します。
  - j [Target] ドロップダウンリストからターゲットフィールドを選択します。
  - k [Save] をクリックします。
  - l [Generate Code] リンクをクリックします。
- 5 [ナビゲーター] ビューで、CFC を含むフォルダを右クリックし、[更新] を選択します。追加の機能を利用できるように CFC のソースコードを変更することもできます。

## AS Class Generator

AS Class Generator は ActionScript クラスを生成します。この ActionScript クラスでは、オフラインメタデータを含めることも除外することもできます。オフラインメタデータは、ColdFusion 9 の ColdFusion-AIR オフライン統合機能の一部です。ActionScript クラスは CFC から生成することも、データベーステーブルから直接生成することもできます。

### AS Class Generator のインストール

サーバーの作成時に [拡張機能のインストール] オプションを選択した場合は、ColdFusion Builder とともに AS Class Generator 拡張機能がインストールされます。

サーバーの作成時に AS Class Generator 拡張機能をインストールしなかった場合は、AS Class Generator 拡張機能のアーカイブファイル (ASClassGenerator.zip) を使用して後からインストールできます。AS Class Generator 拡張機能のアーカイブファイルは、ColdFusion Builder に出荷時から付属し、ColdFusion Builder のインストール先の "Extensions" ディレクトリ内にあります。拡張機能のインストールの詳細については、88 ページの「[拡張機能のインストールとアンインストール](#)」を参照してください。

### AS Class Generator の使用

AS Class Generator アプリケーションは次のレベルで使用できます。

- [RDS データビュー] ビュー
- プロジェクトナビゲータービュー

AS クラスの生成は、次のシナリオでサポートされています。

- データベーステーブルから生成
- CFC から生成

AS クラスは、ORM CFC または従来の CFC から生成できます。

#### データベーステーブルからの AS クラスの生成

- 1 ColdFusion Builder パースペクティブで、[RDS データビュー] を選択します。
- 2 RDS ユーザー名とパスワードの設定が既に完了している場合は、その RDS ユーザー名とパスワードを指定します。

**注意:** RDS パスワードの設定は ColdFusion Administrator で行います。同じく ColdFusion Administrator で管理される ColdFusion Administrator パスワードと RDS パスワードを混同しないでください。

- 3 設定したサーバーで利用可能なデータソースのリストが表示されます。
- 4 データソースを選択します。
- 5 データソース内のテーブルを右クリックし、[Generate AS Class from DB Table]-[Generate AS Class] を選択します。

**注意:** [Generate AS Class from DB Table] オプションは、テーブル名を右クリックした場合にのみ表示されます。データソース自体またはデータソース内のレコードをクリックしても、これらのオプションは表示されません。

- 6 [Generate AS Class] ダイアログボックスで、次の詳細を入力します。

- ActionScript クラスファイルの保存先を指定または選択します。
- 次のメタデータタグを含めるには、[Include AS Metadata Tags] チェックボックスを選択します。

[RemoteClass] ColdFusion サーバー上の ColdFusion CFC または Java オブジェクトを、生成される ActionScript クラスにマッピングします。

[Entity] 生成される ActionScript クラスを、AIR SQLite データベース内のテーブルとして永続的に保存する必要があることを示します。このタグは ColdFusion-AIR オフライン統合タグです。

[Id] ActionScript クラスを AIR SQLite データベース内のテーブルとして永続的に保存した後、AS クラスプロパティをプライマリキーとして指定します。

- Web ルートを基準として、リモート CFC クラスの相対パスをドット表記法で入力します。たとえば、CFC の絶対パスが "C:\ColdFusion9\wwwroot\CFProj\xyz.cfc" である場合、ドット表記法によるパスは "CFProj.xyz" となります。

**注意:** このエントリは、[AS Metadata Tags] チェックボックスを選択しなかった場合には無視されます。

- パッケージの名前を入力します。ここで使用するネーミング規則は、ActionScript のネーミング規則と同じです。

- 7 <データベーステーブル名>.as というネーミング規則で ActionScript クラスファイルが作成されます。

### 生成された ActionScript クラスファイルの確認

次のコード例は Employees.as という名前の ActionScript クラスファイルを示しています。このクラスファイルは cfdocexamples データソース内の Employees テーブルから生成されます。

コード	説明
package package1	"package1" という名前でパッケージが作成されます。
[RemoteClass(alias="cfproj.xyz")]	ActionScript クラスがサーバー上の "cfproj/xyz.cfc" にマッピングされます。
[Entity] public class EMPLOYEES { [Id] public var EMP_ID:int; public var FIRSTNAME:String; public var LASTNAME:String; public var EMAIL:String; public var PHONE:String; public var DEPARTMENT:String; public var LOCATION:String; public var IM_ID:String; public function EMPLOYEES() {} }	クラス名はデータベーステーブル名と同じになります。このクラスの各プロパティは、Employees データベーステーブル内の列に対応しています。各プロパティのデータ型は、列のデータ型のイントロスペクションが完了した後に自動的にマッピングされます。

## CFC からの AS クラスの生成

### ORM CFC から生成する場合

- 1 ORM CFC を作成します。

**注意:** ActionScript クラスの生成対象となる CFC は、ColdFusion Web ルート内に存在している必要があります。

- 2 プロジェクトナビゲーターで ORM CFC を右クリックし、[Generate AS Class from CFC]-[Generate AS Class] を選択します。
- 3 [Generate AS Class] ダイアログボックスで、次の詳細を入力します。
  - ActionScript クラスファイルの保存先を指定または選択します。
  - パッケージの名前を入力します。ここで使用するネーミング規則は、ActionScript のネーミング規則と同じです。
  - [RemoteClass] タグを含めるには、[Include AS Metadata tags] チェックボックスを選択します。
  - 次のタグを含めるには、[Include CF-AIR Offline Metadata] チェックボックスを選択します。
    - [Entity]
    - [Id]
  - 各プロパティのプライマリーキーやデータ型など、必要な情報が ORM CFC に既に含まれている場合は、ColdFusion データソースを入力する必要はありません。

#### 従来の CFC から生成する場合

- 1 [ナビゲーター]ビューで CFC を右クリックし、[Generate AS Class from CFC]-[Generate AS Class] を選択します。

**注意:** ActionScript クラスの生成対象となる CFC は、ColdFusion Web ルート内に存在している必要があります。
- 2 [Generate AS Class] ダイアログボックスで、次の詳細を入力します。
  - ActionScript クラスファイルの保存先を指定または選択します。
  - パッケージの名前を入力します。ここで使用するネーミング規則は、ActionScript のネーミング規則と同じです。
  - [RemoteClass] タグを含めるには、[Include AS Metadata tags] チェックボックスを選択します。
  - 次のタグを含めるには、[Include CF-AIR Offline Metadata] チェックボックスを選択します。
    - [Entity]
    - [Id]
  - 各プロパティのプライマリーキーとデータ型が CFC で指定されていない場合は、ColdFusion データソース名を入力します。

#### 注意事項

- CF-AIR オフライン統合を使用する場合、列固有のメタデータタグ [Column] の追加は現在のところサポートされていません。ただし、ActionScript クラスファイルを生成した後に、手動で列固有のメタデータタグを追加することは可能です。
- RDS は、データソース内のテーブルのプライマリーキーに関する情報を提供しません。プライマリーキーを指定するには cfdbinfo タグを使用します。cfdbinfo タグでは、Microsoft Access などの ODBC ドライバがサポートされていません。そのため、サポートされていないデータソースを使用する場合は、生成された ActionScript クラスファイル内で Id 列を手動で定義する必要があります。
- ActionScript クラスファイルの生成対象となる CFC は、ColdFusion Web ルート内に存在している必要があります。
- ORM CFC の場合、datatype/type は cfproperty タグのオプション属性です。この属性が CFC に存在していない場合、ActionScript クラスファイルは生成されません。ActionScript クラスファイルが生成されるようにするには、ColdFusion データソースを指定します。これを指定すると、データ型情報がデータベーステーブルから取得されて、ActionScript クラスファイルが作成されます。

## Model-Glue Assistant

Model-Glue Assistant は、Model-Glue を初めて利用する開発者向けの ColdFusion Builder アプリケーションです。

Model-Glue は、ColdFusion Web アプリケーションを開発するための Web アプリケーションフレームワークです。

Model-Glue の詳細については、<http://docs.model-glue.com> を参照してください。

### Model-Glue Assistant 拡張機能を使用するための前提条件

Model-Glue Assistant 拡張機能を使用する前に、次のサポートフレームワークをインストールする必要があります。

- Model-Glue Framework ([www.model-glue.com](http://www.model-glue.com) からダウンロードできます)
- ColdSpring Framework ([www.coldspringframework.org](http://www.coldspringframework.org) からダウンロードできます)
- Reactor Framework ([www.reactorframework.org](http://www.reactorframework.org) からダウンロードできます)

これらのサポートフレームワークが既にコンピュータにインストールされている場合、再インストールの必要はありません。

Model-Glue Assistant は、Model-Glue バージョン v2.0.304 で動作するように設計されています。コンピュータ上の Model-Glue がこれ以外のバージョンである場合は、Model-Glue Assistant を更新してください。

### Model-Glue Assistant 拡張機能のインストール

サーバーの作成時に [ 拡張機能のインストール ] オプションを選択した場合は、ColdFusion Builder とともに Model-Glue Assistant 拡張機能がインストールされます。

サーバーの作成時に Model-Glue Assistant 拡張機能をインストールしなかった場合は、88 ページの「[拡張機能のインストールとアンインストール](#)」の説明に従って拡張機能をインストールできます。Model-Glue Assistant 拡張機能のアーカイブファイル (ModelGlue Assistant.zip) は、ColdFusion Builder に出荷時から付属しています。このアーカイブファイルは、ColdFusion Builder のインストール先の "Extensions" ディレクトリにあります。

### ColdFusion プロジェクトの作成

- 1 Web ルートの下に ColdFusion プロジェクトを作成します。詳細については、56 ページの「[ColdFusion プロジェクトの作成](#)」を参照してください。
- 2 [Add Existing Sources and Referenced Project] ダイアログボックスで、アプリケーションのリストにある [ModelGlue Assistant Application] チェックボックスを選択します。
- 3 ColdFusion プロジェクトを作成した後、プロジェクトナビゲーターでそのプロジェクトを右クリックして [更新] を選択し、作成されたファイルとフォルダを確認します。"Config" フォルダにある "ModelGlue.xml" ファイルおよび "ColdSpring.xml" ファイルは、作成したプロジェクトで使用できるように自動的に設定されます。
- 4 Model-Glue Assistant が正しく設定されているかどうかを確認するには、Web ブラウザに次の URL を入力します。  
`http://<ローカルホスト>/<プロジェクト名>/index.cfm`

ModelGlue 2.0 の Web ページに切り替わり、ModelGlue 2.0 が実行中であることを示すメッセージが表示されます。コメントアウトできるデバッグメッセージも多数表示されます。詳細については、97 ページの「[Debug Trace Toggle](#)」を参照してください。

### Model-Glue Assistant の使用

プロジェクトナビゲーターで、設定したプロジェクトを右クリックし、[ModelGlue Assistant] を選択します。次のオプションが表示されます。関連するコードが、次の各アクションに対して生成されます。

## [Add Event]

Model-Glue では、イベントを追加するときに、CFM ページをそのイベントに関連付けます。イベントを追加したときに CFM ファイルに追加されるコードは、一般的に次のような内容になります。

```
<event-handler name="Event1">
  <broadcasts>
    <message name="Hello" />
  </broadcasts>
  <views>
    <include name="body" template="dspPhrase.cfm">
      <value name="xe.translationForm" value="translationForm" />
    </include>
  </views>
</event-handler>
```

このコードには、ハンドラ名、イベントの実行時にブロードキャストされるメッセージ、CFM ファイルに関連付けられているビューなどが含まれています。詳細については、<http://docs.model-glue.com/> を参照してください。

Model-Glue Assistant では、このコードを次の方法で簡単に追加できます。

- 1 [ナビゲーター] でプロジェクトを右クリックし、[ModelGlue Assistant]-[Add Event] を選択します。  
[Add Event] ダイアログボックスで、次の詳細を入力します。
- 2 イベントの名前を入力します。
- 3 必要に応じて、ブロードキャストメッセージを入力します。入力したメッセージはイベントの実行時にブロードキャストされます。
- 4 ビューの名前を入力します。
- 5 ビューに関連付けられている CFM ファイルの保存先を入力します。プロジェクトの "Views" フォルダ内のフォルダまたはサブフォルダを指定する必要があります。
- 6 必要に応じて、値の名前および値を入力します。
- 7 イベントが追加されたかどうかを確認するには、Web ブラウザに次の URL を入力します。  
`http://<ローカルホスト>/<プロジェクト名>/index.cfm?event=<イベント名>`

Model-Glue Assistant の Web ページに切り替わり、デフォルトのメッセージが表示されます。このメッセージを変更するには、ビューに関連付けた CFM ファイルを編集します。

### 生成されるコード

"ModelGlue.xml" ファイルにイベントハンドラエントリが追加されます。

## [Debug Trace Toggle]

[Debug Trace Toggle] 機能を使用すると、Model-Glue の Web ページでデバッグメッセージを表示するか非表示にするかを制御できます。

- 1 [ナビゲーター] でプロジェクトを右クリックし、[ModelGlue Assistant]-[Debug Trace Toggle] を選択します。
- 2 Web ブラウザの URL フィールドに `http://<ローカルホスト>/<プロジェクト名>/index.cfm` と入力します。  
デバッグメッセージが非表示になります。
- 3 デバッグメッセージを表示するには、[ナビゲーター] でプロジェクトを右クリックし、再度 [ModelGlue Assistant]-[Debug Trace Toggle] を選択します。

### 生成されるコード

Model-Glue Assistant によって "ColdSpring.xml" ファイル内の debug ノードの値が切り替えられます。

## [Enable Reactor]

データベースに関連するイベントの場合、Model-Glue は Reactor フレームワークと連動します。Model-Glue を Reactor と連動させるには、Reactor を有効にします。Reactor を有効化するには、"ColdSpring.xml" ファイル内の Reactor エントリのコメント化を解除します (このファイルはプロジェクトの "Config" フォルダにあります)。

- [ナビゲーター] でプロジェクトを右クリックし、[ModelGlue Assistant]-[Enable Reactor] を選択します。

### 生成されるコード

Model-Glue Assistant により、"ColdSpring.xml" ファイル内の Reactor エントリのコメント化が解除されます。

## Scaffold の作成

Model-Glue Assistant では、データベーステーブルに関連する基本的なユーザーインターフェイスをすばやく作成するための Scaffold を作成できます。Scaffold の詳細については、<http://docs.model-glue.com> および <http://livedocs.reactorframework.com> を参照してください。

- 1 ColdFusion Builder パースペクティブで、[RDS データビュー] を選択します。

- 2 RDS ユーザー名とパスワードの設定が既に完了している場合は、その RDS ユーザー名とパスワードを指定します。

**注意:** RDS パスワードの設定は ColdFusion Administrator で行います。同じく ColdFusion Administrator で管理される ColdFusion Administrator パスワードと RDS パスワードを混同しないでください。

- 3 設定したサーバーで利用可能なデータソースのリストが表示されます。

- 4 データソースを選択します。

**注意:** Scaffold では一部のタイプのデータソースがサポートされません。サポートされるデータソースについては、<http://livedocs.reactorframework.com/> を参照してください。

- 5 データソース内のテーブルを右クリックして、[ModelGlue Assistant]-[Create Scaffold Entry] を選択します。

**注意:** このオプションは、テーブル名を右クリックした場合にのみ表示されます。データソース自体またはデータソース内のレコードをクリックしても、[Create Scaffold Entry] オプションは表示されません。

[Create Scaffold Entry] ダイアログボックスで、次の詳細を入力します。

- 6 Model-Glue Assistant 用に設定されたプロジェクトを選択します。

- 7 データソースのユーザー名とパスワードを入力します。

**注意:** データソースのユーザー名とパスワードの設定は、ColdFusion Administrator で行います。データソースのパスワードは、RDS パスワードおよび ColdFusion Administrator パスワードとは異なります。

- 8 [DB Type] ドロップダウンリストからデータソースのタイプを選択します。

- 9 Scaffold エントリが追加されたかどうかを確認するには、Web ブラウザに次の URL を入力します。

`http://<ローカルホスト>/<プロジェクト名>/index.cfm?event=<テーブル名>.list`

- 10 Scaffold エントリが作成されると、プロジェクトの "Scaffolds" フォルダの下に次のファイルが作成されます。

- dsp<テーブル名>.cfm
- dsp<テーブル名>List.cfm
- frm<テーブル名>.cfm

### 生成されるコード

- 選択したテーブルの "Reactor.xml" に Scaffold エントリが追加されます。テーブルの相互関係情報は追加されません。
- "ColdSpring.xml" の Reactor 設定が正しいデータソースの場所に更新されます。

- "ModelGlue.xml" に Scaffold オブジェクトエントリが追加されます。

## Model-Glue Assistant の更新

Model-Glue Assistant 拡張機能は、コンピュータにインストールされている Model-Glue およびサポートフレームワークのバージョンに合わせて更新できます。

- 1 Model-Glue Assistant がインストールされている Web ルート内の場所へ移動します。
- 2 Model-Glue Assistant のフォルダには "Handlers" フォルダと "ide\_config.xml" ファイルがあります。
- 3 コンピュータ上の Model-Glue のバージョンに合わせて Model Glue Application テンプレートファイルを更新するには、"Handlers" フォルダ内の "modelapp.ini" ファイルを編集します。"modelapp.ini" ファイルに含まれるコードは次のような内容です。

```
<model>  
  <sampleapp name="modelglueapplicationtemplate" location="" />  
</model>
```

location 属性で "Model-Glue Application Template" フォルダの絶対パスを指定します。

たとえば、location="C:\ColdFusion9\wwwroot\ModelGlueApplicationTemplate" のように指定します。

Model-Glue Assistant により文字列処理操作が実行されて、コードが生成されます。生成されたコードをカスタマイズするには、必要に応じて "Handler" フォルダ内の CFM ファイルを更新します。