

## **Oracle WebCenter Sites**

開発者ツール

11g リリース 1 (11.1.1)

部品番号 : B69682-01

2012 年 4 月

Oracle WebCenter Sites: 開発者ツール, 11g リリース 1 (11.1.1)

部品番号 : B69682-01

原本名 : Oracle® WebCenter Sites: Developer Tools, 11g Release 1 (11.1.1)

原本主著者 : Melinda Rubenau

原本協力著者 : Tatiana Kolubayev

原本協力者 : Valentin Vakar, Cory Lum

Copyright © 2012 Oracle and/or its affiliates. All rights reserved.

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバースエンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントが、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供される場合は、次の Notice が適用されます。

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このソフトウェアまたはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアは、危険が伴うアプリケーション(人的傷害を発生させる可能性があるアプリケーションを含む)への用途を目的として開発されていません。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用する場合、それを安全に使用するために、適切な安全装置、バックアップ、冗長性(redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

Oracle および Java は Oracle Corporation およびその関連企業の登録商標です。その他の名称は、他社の商標の可能性がります。

Intel および Intel Xeon は Intel Corporation の商標または登録商標です。すべての SPARC の商標はライセンスをもとに使用し、SPARC International, Inc. の商標または登録商標です。AMD、Opteron、AMD ロゴ、AMD Opteron ロゴは、Advanced Micro Devices の商標または登録商標です。UNIX は、X/Open Company, Ltd のライセンスによる登録商標です。

このソフトウェアまたはハードウェアおよびドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても、一切の責任を負いかねます。

# 目次

<b>このガイドについて</b> .....	<b>7</b>
対象読者 .....	7
関連ドキュメント .....	7
表記規則 .....	8
サード・パーティのライブラリ .....	8
<b>1 Oracle WebCenter Sites: 開発者ツールについて</b> .....	<b>9</b>
はじめに .....	10
開発者ツールのアーキテクチャ .....	10
IDE の統合 .....	12
開発者ツール・ワークスペース .....	12
同期 .....	13
JSP の管理 .....	13
コマンドライン・ツール .....	13
バージョン・コントロール・システムの使用方法 .....	14
次の手順 .....	15
<b>2 クイック・スタート</b> .....	<b>17</b>
前提条件 .....	18
開発者ツールの設定 .....	18
Eclipse での WebCenter Sites リソースの管理 .....	20
<b>3 Eclipse における WebCenter Sites の機能</b> .....	<b>23</b>
Oracle WebCenter Sites パースペクティブ .....	24
構成画面 .....	25
Eclipse のプロジェクトとワークスペース .....	26
開発者ツール・ビュー .....	27
Sites ワークビュー .....	27
Sites ログビュー .....	28
Sites プレビュー・ブラウザビュー .....	28
Sites UI ビュー .....	29
Sites ロギング構成ビュー .....	30

Sites 開発者リファレンスビュー .....	30
ウィザード .....	31
データ同期 ( エクスポート / インポート ) ツール .....	32
ワークスペースとの同期 ( WebCenter Sites からのエクスポート ) .....	32
WebCenter Sites との同期 ( WebCenter Sites へのインポート ) .....	33
次の手順 .....	34
<b>4 JSP の開発 .....</b>	<b>35</b>
開発者ツールを使用した JSP の開発 .....	36
タグおよび Java API 補完 .....	37
デバッグ .....	38
<b>5 同期とデータ交換 .....</b>	<b>41</b>
開発者ツールを使用した同期 .....	42
同期シナリオ .....	42
依存性解決 .....	43
データ交換とマッピング .....	44
ID マッピング .....	44
リソースの fw_uid のオーバーライド .....	48
既存のリソースと開発者ツールの併用 .....	49
サイト・マッピング .....	49
ナチュラル・サイト・マッピング .....	50
コマンドライン・ツールを使用したナチュラル・サイト・マッピングのオーバーライド .....	50
<b>6 ワークスペース .....</b>	<b>51</b>
概要 .....	52
ワークスペースの構造 .....	52
アセット・ストレージ構造 .....	53
コード・ベースのリソース・ストレージ構造 .....	53
属性エディタ・ストレージ構造 .....	54
アセット・タイプ・ストレージ構造 .....	54
<b>7 コマンドライン・ツール .....</b>	<b>57</b>
概要 .....	58
コマンドライン・ツールの実行と使用 .....	58
コマンド例 .....	60
モジュールの作成 .....	60
<b>8 バージョン・コントロール・システムとの統合に関する注意 .....</b>	<b>63</b>
バージョン・コントロールと開発者ツール .....	64
開発者ツールと VCS の統合 .....	64
開発者ツールが統合された VCS での作業 .....	65

## 付録

<b>A. 開発チームの統合の使用例</b> .....	<b>69</b>
今日: サイトおよび関連リソースの開発 .....	70
午前7時14分: 新しいプロジェクトの割当て .....	70
午前7時34分: 開発者ツールの設定 .....	70
午前7時45分: サイト定義の作成 .....	71
午前7時46分: サイトのリソースの作成 .....	72
午前8時12分: VCS に関するディスカッション .....	73
午前9時42分: ワークスペースと VCS の同期 .....	74
午前10時12分: 他のチーム・メンバーのワークスペースと SVN リポジトリの同期 .....	79
午前10時18分: ワークスペースと WebCenter Sites インスタンスの 同期 .....	80
午前10時21分: サイト権限の割当て .....	83
午前10時22分: スタート・メニューの問題 .....	84
午前10時24分: スタート・メニューの問題の解決 .....	84
午前11時17分: マーケティング・リクエストによる変更 .....	86
午前11時22分: 作成者定義への新しい属性の追加 .....	87
午前11時25分: サイトの変更の確認 .....	87
午前11時44分: 作成者定義の属性の変更 .....	88
午前11時53分: チームのワークスペースと WebCenter Sites インスタンスの更新 .....	89
午後12時27分: チームによるサイトの Template アセットの作成 .....	90
3日後 ... デプロイメント .....	93
午前9時32分: デプロイメントの準備 .....	93
午前10時04分: サイトとそのリソースのデプロイ .....	96
午前10時55分: デプロイメントの成功 .....	99
<b>B. コマンドライン・ツールを使用した再利用可能モジュールの作成</b> .....	<b>101</b>
再利用可能モジュールの作成 .....	102
手順 I. WebCenter Sites インスタンスでのリソースのリスト表示 .....	102
手順 II. スタート・メニュー・アイテムのリスト表示 .....	103
手順 III. 目的のワークスペースへのすべてのリソースのエクスポート ...	104
手順 IV. モジュールのコンテンツの調査 .....	105
手順 V. モジュールのアーカイブ .....	105
手順 VI. WebCenter Sites インスタンスへのモジュールのインポート ....	106



## このガイドについて

このガイドでは、Oracle WebCenter Sites: 開発者ツール (Oracle WebCenter Sites と Eclipse 統合開発環境 (IDE) の統合に使用されるツールキット) について説明します。開発者は開発者ツール・キットを使用すると、Eclipse IDE やバージョン・コントロール・システム (VCS) を使用して分散環境で作業できます。

このガイドで説明しているアプリケーションは、旧 FatWire の製品です。命名規則は次のとおりです。

- Oracle WebCenter Sites は、以前は *FatWire Content Server* と呼ばれていたアプリケーションの現在の名前です。このガイドでは、Oracle WebCenter Sites を *WebCenter Sites* と呼ぶこともあります。
- Oracle WebCenter Sites: 開発者ツールは、以前は *FatWire Content Server 開発者ツール (CSDT)* と呼ばれていたアプリケーションの現在の名前です。このガイドでは、Oracle WebCenter Sites: 開発者ツールを *開発者ツール* と呼ぶこともあります。

### 対象読者

このガイドは、WebCenter Sites の全体管理者や開発者を対象としています。ユーザーは、その企業のサイト設計に関する幅広い知識や JSP 開発の高度な知識を持っていることが前提になります。Eclipse IDE の使用経験が必要になります。また、コマンドライン・ツールやバージョン・コントロール・システム (VCS)、および Oracle WebCenter Sites Javadoc やタグ・リファレンスの使用経験も必要です。

### 関連ドキュメント

詳細は、次のドキュメントを参照してください。

- 『Oracle WebCenter Sites 管理者ガイド』
- 『Oracle WebCenter Sites 開発者ガイド』
- Oracle WebCenter Sites Javadoc
- Oracle WebCenter Sites タグ・リファレンス

## 表記規則

このガイドでは、次の表記規則を使用します。

- **太字**は、ユーザーが選択するグラフィカル・ユーザー・インタフェース要素を示します。
- *斜体*は、ドキュメントのタイトル、強調、またはユーザーが特定の値を指定する変数を示します。
- 等幅フォントは、ファイル名、URL、サンプル・コード、または画面に表示されるテキストを示します。
- 等幅太字フォントは、コマンドを示します。

## サード・パーティのライブラリ

Oracle WebCenter Sites およびそのアプリケーションには、サード・パーティのライブラリが含まれています。詳細は、*Oracle WebCenter Sites 11gR1: サード・パーティのライセンス*を参照してください。

## 第 1 章

# Oracle WebCenter Sites: 開発者ツールについて

この章では、Oracle WebCenter Sites: 開発者ツールの概要について説明します。  
この章は、次の項で構成されています。

- [はじめに](#)
- [開発者ツールのアーキテクチャ](#)
- [次の手順](#)

## はじめに

このガイドでは、Oracle WebCenter Sites: 開発者ツールを使用した WebCenter Sites CM サイトの開発に関する情報を提供し、その手順を説明します。開発者は開発者ツール・キットを使用すると、Eclipse 統合開発環境 (IDE) やバージョン・コントロール・システム (VCS) 統合などのツールを使用して分散環境で作業できます。開発者ツールは他の開発モデルに干渉したり、これらと統合することはありません。開発者ツールを使用すると、開発チームは WebCenter Sites リソースを管理して、他のチーム・メンバーとこれらのリソースを交換できます。

## 開発者ツールのアーキテクチャ

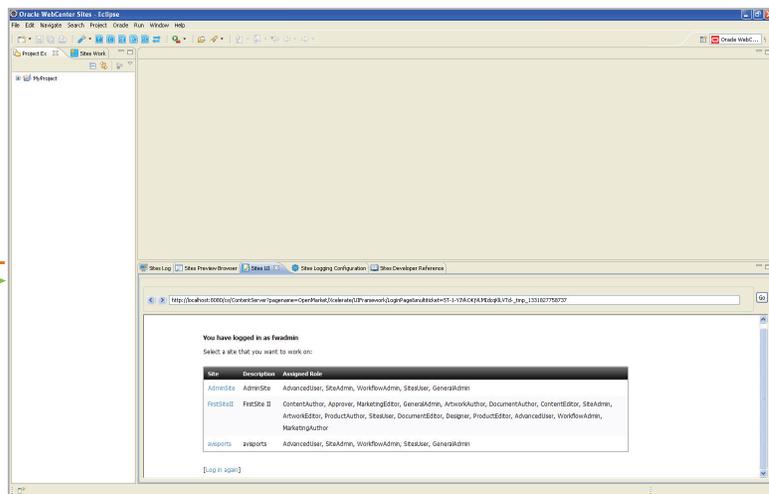
11 ページの図 1 で示されているように、Oracle WebCenter Sites を実行する任意のコンピュータ上で開発者ツールを使用して、WebCenter Sites を Eclipse IDE と統合し、パーソナルで柔軟な開発者環境を作成できます。

- 開発者は開発者ツール (および WebCenter Sites) と、主に Eclipse を通してやり取りします。Eclipse を統合すると、アセットおよび他のタイプの WebCenter Sites リソースを管理するための一連の充実した WebCenter Sites 固有のツールが提供されます。
- 開発者ツール・キットにより、Eclipse でのリソース開発を WebCenter Sites でのリソース開発に同期させたり、WebCenter Sites でのリソース開発を Eclipse でのリソース開発に同期できます。

開発者ツール・キットは概念と設計はシンプルであっても、多数の重要な意味を持っています。たとえば、IDE によって管理されるリソースは、ファイル・システムにファイルとして格納され、開発者は自分たちのバージョン・コントロール・システムと統合できます。同時に、ファイルは WebCenter Sites のネイティブ・アセット表現に自動的に変換され、WebCenter Sites データベースにインポートされます。反対方向の同期も実行できるので、開発者は Eclipse で作業することも、WebCenter Sites で直接作業することもできます。開発者ツールの機能の詳細は、この章で説明していきます。

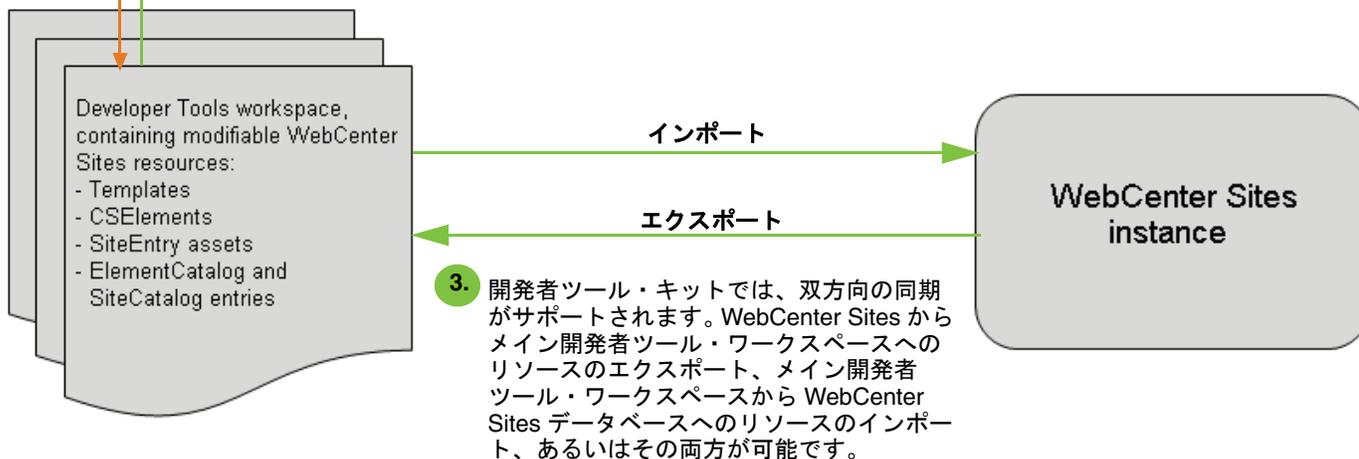
図 1: 開発者ツールのプロセス・フロー

## Eclipse プラグイン



1. 開発者は開発者ツールが統合された Eclipse で作業して、コードベースの WebCenter Sites リソースを作成および管理します。

2. Eclipse で保存されるリソースのメタデータは複数のファイルに変換され、メイン開発者ツール・ワークスペースというファイル・システム構造に格納されます (組込みの WebCenter Sites 管理インタフェースで作成および保存されたリソースは除きます)。これらのリソースに関連付けられた各 JSP および XML エレメントは、それぞれのファイルに配置されます。



## IDE の統合

開発者は Eclipse の統合を使用して、次を行うことができます。

- CSElement アセット、Template アセット、SiteEntry アセット、SiteCatalog エントリおよび ElementCatalog エントリの作成、編集および削除
- タグの補完、構文の強調表示、デバッグなどの標準 Eclipse 機能を使用した JSP エレメントの開発
- アセット、アセット・タイプ、フレックス・ファミリー、サイト、ロール、ツリー・タブおよびスタート・メニュー・アイテムのエクスポートとインポート
- 組込みのプレビュー・ブラウザを使用した、Eclipse IDE 内での WebCenter Sites ページのプレビュー
- 動的にリフレッシュされるパネルでの WebCenter Sites ログ・ファイルの表示
- WebCenter Sites リソースのファイル・システム表現を使用した、既存の Eclipse 機能の利用によるバージョン・コントロール・システムとの統合

### 注意

開発者ツール・キットと統合された Eclipse により、組込みの WebCenter Sites 管理インタフェースが提供され、これを使用してあらゆるタイプの WebCenter Sites リソースを管理できます。管理インタフェースで作成される WebCenter Sites リソースは、ファイル・システム構造に格納されるのではなく、WebCenter Sites データベースに直接格納されます。**組込みの管理インタフェースについては、このガイドでは説明されていません。**このインタフェースで実行できる開発タスクの詳細は、『Oracle WebCenter Sites 開発者ガイド』および『管理者ガイド』を参照してください。

## 開発者ツール・ワークスペース

Eclipse で管理される WebCenter Sites リソースは、メイン開発者ツール・ワークスペースというファイル・システム構造にファイルとして格納されます。これによってリソースを簡単に管理できることに加え、オプションとしてリソースを他の WebCenter Sites インスタンスと交換できます。メイン開発者ツール・ワークスペースは、Eclipse からアクセスできる唯一のワークスペースです。

### 注意

上級開発者は、コマンドライン・ツールを使用して、WebCenter Sites インスタンス用に任意の数のカスタム開発者ツール・ワークスペースを作成できます。Eclipse からはカスタム・ワークスペースにアクセスできません。カスタム・ワークスペースの作成はオプションで、ほとんどの分散環境で必要なワークスペースはメイン開発者ツール・ワークスペースのみです。詳細は、第 6 章「ワークスペース」を参照してください。

このガイドでは、開発者ツール・ワークスペースという用語は、メイン開発者ツール・ワークスペースを意味します。カスタム開発者ツール・ワークスペースを意味する場合は、明示的に記述します。

## 同期

開発者ツール・キットを使用すると、Eclipse IDE のリソース開発を WebCenter Sites のリソース開発と同期させて、開発者ツール・ワークスペースと WebCenter Sites データベースに確実に同じコンテンツが移入されるようにできます。手動同期は双方向で、リソースを WebCenter Sites にインポートすることも、開発者ツール・ワークスペースにエクスポートすることもできます。

Eclipse IDE で開発されたすべてのリソースは、1 つのファイルまたは関連する複数のファイルとして開発者ツール・ワークスペースに格納されます。WebCenter Sites にインポートされたリソースは、開発者ツール・キットによってネイティブ WebCenter Sites 形式 (データベース表現) に変換されて、WebCenter Sites データベースに格納されます。WebCenter Sites で直接開発されたリソースを Eclipse IDE にエクスポートすると、これらのリソースは開発者ツール・キットによってファイルに変換されて、開発者ツール・ワークスペースに格納されます。

### 注意

Eclipse で WebCenter Sites リソースを編集、作成または削除すると、自動同期が実行されます。すべての変更は Eclipse と統合された WebCenter Sites インスタンスと自動的に同期されて、ネイティブ・データベース構造に格納されます。WebCenter Sites への同期は、Eclipse と統合された WebCenter Sites インスタンスが実行されている場合のみ自動的に行われます。

## JSP の管理

開発者ツール・キットでは、開発者ツール・ワークスペースの既知の場所に JSP が公開されます。これによって開発者は、ファイルで直接作業を行って JSP の記述やデバッグが可能です。これにより、Eclipse での WebCenter Sites JSP の管理 (作成、編集、デバッグ) が、他の JSP ファイルでの作業と同じになります。開発者ツール・キットによって、開発者ツール・ワークスペースに格納されているファイルは、WebCenter Sites と自動的に同期されます。この同期には、WebCenter Sites でのページと結果セットのキャッシュの透過的なフラッシュも含まれます。

さらに、開発者ツール・キットでは、WebCenter Sites への変更を自動的に同期させることで、他のすべてのファイルに対する変更を管理します。ただし、Eclipse を使用しないで WebCenter Sites リソースを変更したり、WebCenter Sites で直接リソースを変更した場合は、手動同期が必要です。

## コマンドライン・ツール

開発者ツール・キットでは、自動化、デプロイメントおよび特定の開発作業に使用できるコマンドライン・ユーティリティが提供されます。コマンドライン・ツールは、リソースの大規模な移動のためのエクスポートおよびインポート機能です。ただし、メイン開発者ツール・ワークスペースにエクスポートされる WebCenter Sites リソースでのみ作業可能な Eclipse 統合とは異なり、コマンドライン・ツールでは任意のワークスペースに格納された WebCenter Sites リソースで作業できます。

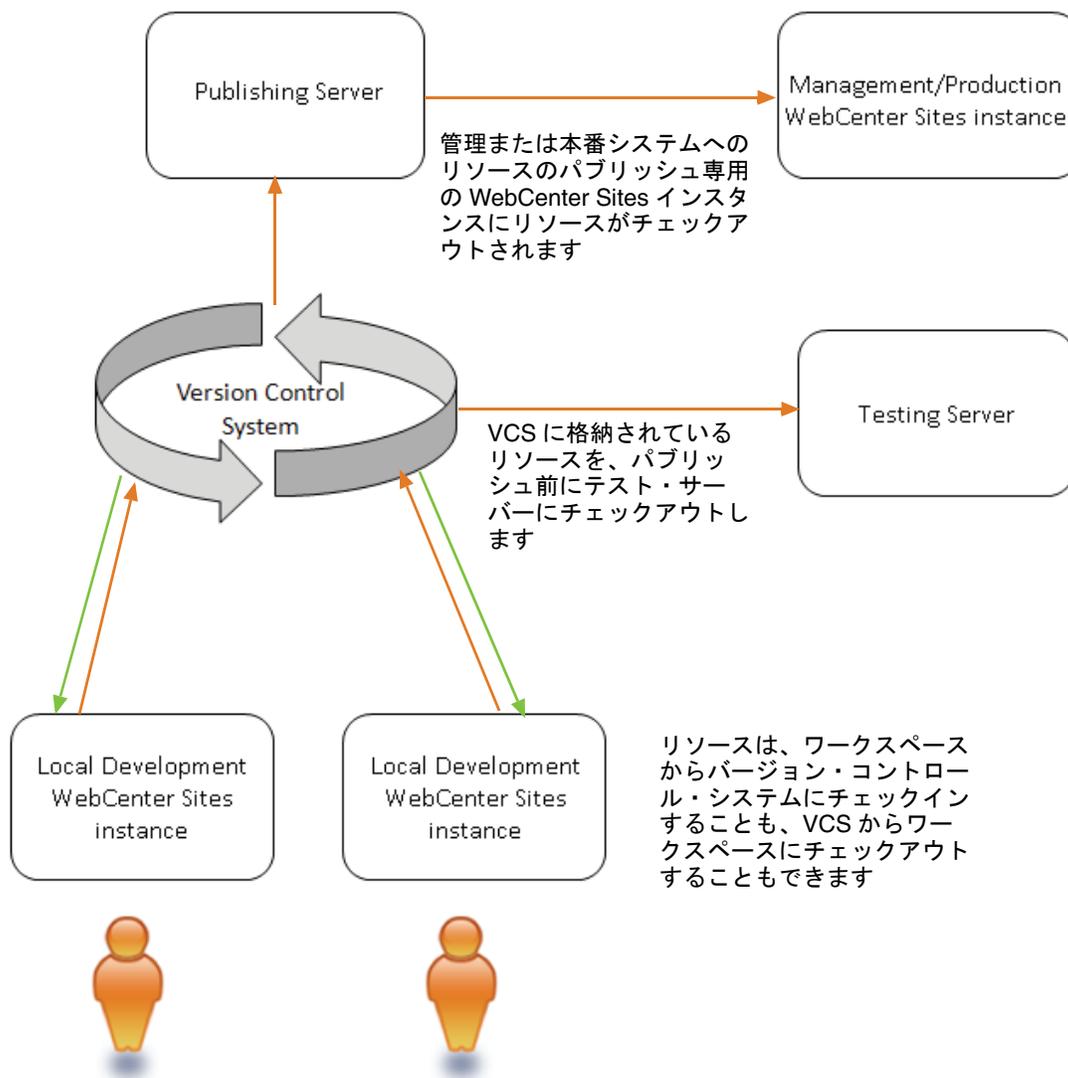
## バージョン・コントロール・システムの使用方法

開発者ツール・キットでは、WebCenter Sites インスタンス間のリソースの交換がサポートされます。これは、バージョン・コントロール・システムの実装によって実現できます。開発者ツール・キットではバージョン・コントロール・システムと統合するためのツールは提供されませんが、開発者ツール・キットによってワークスペースにリソースが格納されるファイル・システム構造では、バージョン・コントロール統合の実装がサポートされます。ワークスペースのファイル・システム構造では、バージョン・コントロール・システムによるリソースの追跡が可能です。

ワークスペースからバージョン・コントロール・システムへリソースをチェックインすることで、他の開発者とこれらのリソースを交換できます。また、他の開発者がバージョン・コントロール・システムにチェックインしたリソースを使用して、ワークスペースを更新できます。バージョン・コントロール・システムを使用すると、テスト・サーバー、管理または本番用の WebCenter Sites システム、別の開発者の WebCenter Sites インスタンスなど、任意のターゲット・システムにリソースをチェックアウトできます。

図 2 には、バージョン・コントロール・システムとともに開発者ツールを使用する例が示されています。この例では、専用 WebCenter Sites インスタンスを使用して、管理または本番用の WebCenter Sites インスタンスにリソースをパブリッシュしています。したがって、WebCenter Sites によって提供される承認 / パブリッシュ機能を使用して、バージョン・コントロール・システムからチェックアウトされたリソースをパブリッシュできます。この例は、開発者ツールとともに VCS を使用する際に推奨される方法ですが、必ずしも使用する必要はありません。

図 2: バージョン・コントロール・システムとともに開発者ツールを使用する方法



## 次の手順

このガイドでは以降、分散開発環境で WebCenter Sites リソースを管理するための開発者ツールの使用方法について説明していきます。次の章では、開発者ツールをインストールして WebCenter Sites インスタンスを Eclipse IDE と統合する手順を示します。また、次の章では、Eclipse と統合された WebCenter Sites インスタンスでリソースの作成と管理を開始するための情報も提供します。詳細と手順は、[第 2 章「クイック・スタート」](#)を参照してください。



## 第 2 章

# クイック・スタート

この章では、開発者ツールを設定して WebCenter Sites インスタンスを Eclipse IDE と統合する手順を示します。この章ではまた、Eclipse での WebCenter Sites リソースの管理の概要も説明します。

この章は、次の項で構成されています。

- [前提条件](#)
- [開発者ツールの設定](#)
- [Eclipse での WebCenter Sites リソースの管理](#)

## 前提条件

開発者ツールを設定する前に、次の要件が満たされていることを確認してください。

- 開発者ツール・キットを使用するには、完全に機能するライセンス付与された WebCenter Sites が必要です。
- Eclipse を WebCenter Sites と統合したら、全体管理者の資格証明 (fwadmin/xceladmin など) を使用して WebCenter Sites にログインする必要があります。このユーザーは RestAdmin グループに属している必要があります。
- コマンドライン・ツール機能を使用するには、開発者ツールの高度な知識が必要です。コマンドライン・ツールの実行と使用に関する詳細と手順は、第7章「コマンドライン・ツール」を参照してください。

## 開発者ツールの設定

1. Eclipse 3.6 Helios J2EE Edition を、WebCenter Sites がインストールされているコンピュータにインストールします。Eclipse は次の URL からダウンロードできます。

<http://eclipsesource.com/en/downloads/eclipse-helios-download/>

2. WebCenter Sites の配布パッケージにある csdt.zip を解凍します。csdt-eclipse フォルダを開き、com.fatwire.csdt.eclipsecsdt\_1.0.0.jar ファイルを Eclipse インストールの plugins フォルダに保存します。
3. ローカル WebCenter Sites を起動します。
4. Eclipse (eclipse.exe) を起動し、希望に応じて設定を構成します。
5. 「Oracle WebCenter Sites」 パースペクティブを開きます。  
Eclipse メニュー・バーで、「Window」→「Open Perspective」→「Other ...」→「Oracle WebCenter Sites」を選択します。
6. WebCenter Sites を Eclipse IDE と統合します。
  - 開発者ツールを初めて設定する場合は、構成画面が自動的に表示されます。
  - 過去に開発者ツールを使用したことがあり、別の WebCenter Sites インスタンスを統合する場合は、Eclipse メニュー・バーで「Oracle」→「Configure」を選択します。

構成画面で、使用する WebCenter Sites インスタンスの情報を次のフィールドに入力します。

- a. 「Sites Installation Directory」フィールドで、「Browse」をクリックして、WebCenter Sites インスタンスの futuretense.ini ファイルを含むディレクトリを選択します。
- b. 「Username」フィールドに、全体管理者のユーザー名を入力します。このユーザーは、RestAdmin グループのメンバーである必要があります。
- c. 「Password」フィールドに、手順 b で入力したユーザー名のパスワードを入力します。

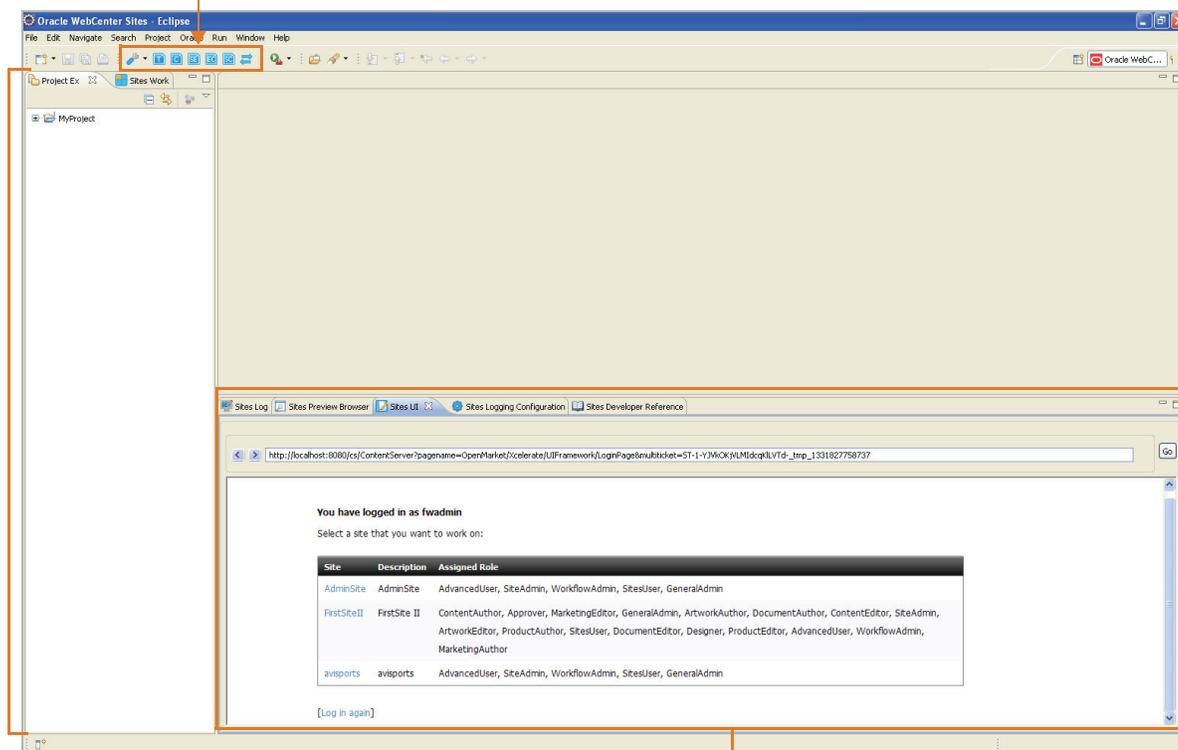
- d. 「Project name」フィールドに、作業するプロジェクトの名前を入力します。
- e. 「Sites Log File」フィールドに、ログ・ファイルの場所を入力します (例: logs/sites.log)。
- f. 「OK」をクリックします。

「Oracle WebCenter Sites」パースペクティブが開きます。WebCenter Sites が統合された Eclipse に初めてアクセスする場合、Oracle WebCenter Sites パースペクティブは次のようになります。

**WebCenter Sites ツールバー**は、次を提供します。

- Oracle WebCenter Sites 構成画面へのショートカット
- SiteEntry、CSElement および Template の各アセット作成のためのショートカット
- SiteCatalog および ElementCatalog の各エントリ作成のためのショートカット
- 同期ツール

「Project Explorer」  
ビューおよび Sites  
ワークスペース・エ  
レメントビュー



下部パネル・ビュー: Sites ログ、Sites プレビュー・ブラウザ、Sites UI、Sites ロギング構成および Sites 開発者リファレンス。

前述のように Oracle パースペクティブがレンダリングされたら、開発者ツール・プラグインのインストールと構成は成功です。

### 注意

Eclipse ビューと開発者ツール・ビューを含むパネルは、相互に交換できます。別のパネルにビューを移動するには、ビューのタブをクリックし、希望のパネルにそれをドラッグします。

7. (オプション) Oracle WebCenter Sites タグ・リファレンスと Javadoc を Eclipse と関連付ける手順は次のとおりです。
  - a. TagReference.zip と javadoc.zip をダウンロードします。
  - b. WebCenter Sites インストール・ディレクトリ内に developerdocs という名前のフォルダを作成します。
  - c. developerdocs フォルダ内に TagReference.zip と javadoc.zip を抽出します。
  - d. Eclipse で Sites 開発者リファレンスビューを開きます。タグ・リファレンスタブと Javadoc タブの両方で、「**Home**」をクリックします。Oracle WebCenter Sites タグ・リファレンスと Javadoc がそれぞれのタブに表示されます。
8. WebCenter Sites システムをアップグレードした場合に、開発者ツールを使用して、このリリースより前に作成されたリソース (既存のリソース) で作業を行うときは、49 ページの「既存のリソースと開発者ツールの併用」を参照してください。
9. Oracle WebCenter Sites パースペクティブでの WebCenter Sites リソースの管理をすぐに開始するには、次の項に進みます。手順 4 から開始してください。

## Eclipse での WebCenter Sites リソースの管理

この項では、次のコードベースの WebCenter Sites リソースの作成手順、編集手順およびそれ以外の管理手順をまとめることで、Eclipse Oracle WebCenter Sites パースペクティブをガイドします。

- SiteEntry アセット
- CSElement アセット
- Template アセット
- ElementCatalog エントリ
- SiteCatalog エントリ

### Eclipse で WebCenter Sites リソースを管理するには :

1. ローカル WebCenter Sites を起動します。
2. Eclipse を起動します。
3. 「Oracle WebCenter Sites」 パースペクティブを開きます。

Eclipse メニュー・バーで、「**Window**」→「**Open Perspective**」→「**Other ...**」→「**Oracle WebCenter Sites**」を選択します。

4. リソースを作成するには、次の手順に従います。
  - **SiteEntry** アセットを作成するには、**SE** アイコンをクリックして、フォームに入力します。
  - **CSElement** アセットを作成するには、**C** アイコンをクリックして、フォームに入力します。
  - **Template** アセットを作成するには、**T** アイコンをクリックして、フォームに入力します。
  - **ElementCatalog** エントリを作成するには、**EC** アイコンをクリックして、フォームに入力します。
  - **SiteCatalog** エントリを作成するには、**SC** アイコンをクリックして、フォームに入力します。

フィールドの定義については、『*Oracle WebCenter Sites 開発者ガイド*』の **Template**、**CSElement** および **SiteEntry** アセットの作成に関する項を参照してください。

5. 作成、編集、削除または他のサイトと共有するリソースを管理するには、**Sites** ワークスペース・エレメントビューを使用します。リソースを右クリックし、目的のオプションを選択します。使用可能なオプションの詳細は、[27 ページの「Sites ワークビュー」](#)を参照してください。
6. 開発者ツール・ビューをパネルに表示するには、次の手順に従います。
  - a. 「**Window**」→「**Show View**」→「**Other...**」を選択します。
  - b. 「**Show View**」ダイアログ・ボックスで、目的のビュー (**Oracle WebCenter Sites** フォルダの下に配置されています) を選択します。
    - **Sites UI** には、組込みの WebCenter Sites 管理インタフェースが表示されます。
    - **Sites ログ** には、WebCenter Sites のログ・ファイルが表示されます。このビューは、WebCenter Sites ログ・ファイルの場所を構成画面で指定した場合のみ表示されます (手順は [18 ページの手順 6](#) を参照)。
    - **Sites 開発者リファレンス** には、*Oracle WebCenter Sites* タグ・リファレンスと *Javadoc* が表示されます。これらは、タグ・リファレンスと *Javadoc* を現在の WebCenter Sites インスタンスと関連付けた場合のみ表示されます (手順は [20 ページの手順 7](#) を参照)。
    - **Sites ワークスペース・エレメント** では、コード関連のリソースにアクセスできます。このビューではツリーにリソースが表示され、そのサイト・アフィリエーションに従って各リソースがグループ化されます。
    - **Sites ロギング構成** には、log4j 構成の動的に更新されるビューが表示されます。このビューでは、各 WebCenter Sites ログ出力のログ・レベルを設定できます。
    - **Sites プレビュー・ブラウザ** には、組込みのプレビュー・ブラウザが表示されます。

開発者ツール・ビューの詳細は、[27 ページの「開発者ツール・ビュー」](#)を参照してください。

7.  アイコンを選択して、WebCenter Sites リソースを同期させます。同期ツールにより、WebCenter Sites から開発者ツール・ワークスペースにデータをエクスポートしたり、開発者ツール・ワークスペースから WebCenter Sites にデータをインポートできます。
  - 同期ツールの使用方法の概要は、32 ページの「データ同期 (エクスポート / インポート) ツール」を参照してください。
  - リソースの同期の詳細は、第5章「同期とデータ交換」を参照してください。

## 第 3 章

# Eclipse における WebCenter Sites の機能

この章では、開発者ツールと統合された Eclipse で提供される WebCenter Sites の機能について説明します。

- [Oracle WebCenter Sites パースペクティブ](#)
- [次の手順](#)

## Oracle WebCenter Sites パースペクティブ

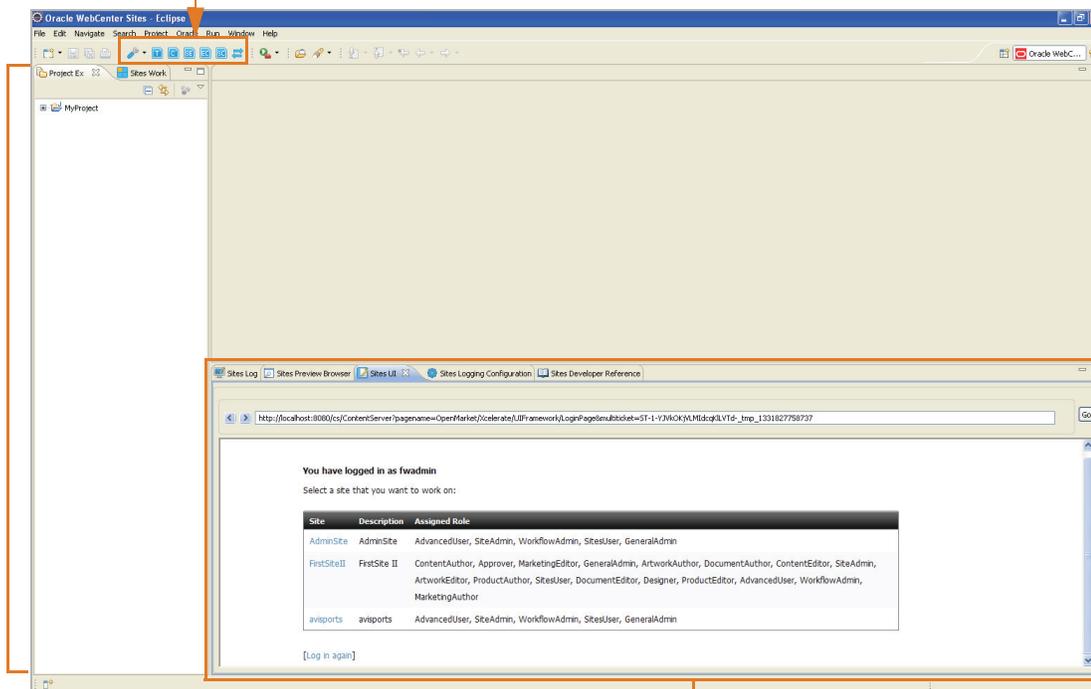
Eclipse の開発者ツール機能はすべて Oracle WebCenter Sites パースペクティブ (「Window」→「Open Perspective」→「Other...」→「Oracle WebCenter Sites」) の下でグループ化されます。

**図 3:** Oracle WebCenter Sites パースペクティブ : 「Eclipse」→「Window」→「Open Perspective」→「Other...」→「Oracle WebCenter Sites」

**WebCenter Sites ツールバー**は、次を提供します。

- WebCenter Sites 構成画面へのショートカット
- SiteEntry、CSElement および Template の各アセット作成のためのショートカット
- SiteCatalog および ElementCatalog の各エントリ作成のためのショートカット
- 同期ツール

左側のパネルには、「Project Explorer」ビューと Sites ワークスペース・エレメントビューが含まれます。



下部パネル・ビュー : Sites ログ、Sites プレビュー・ブラウザ、Sites UI、Sites ロギング構成および Sites 開発者リファレンス。

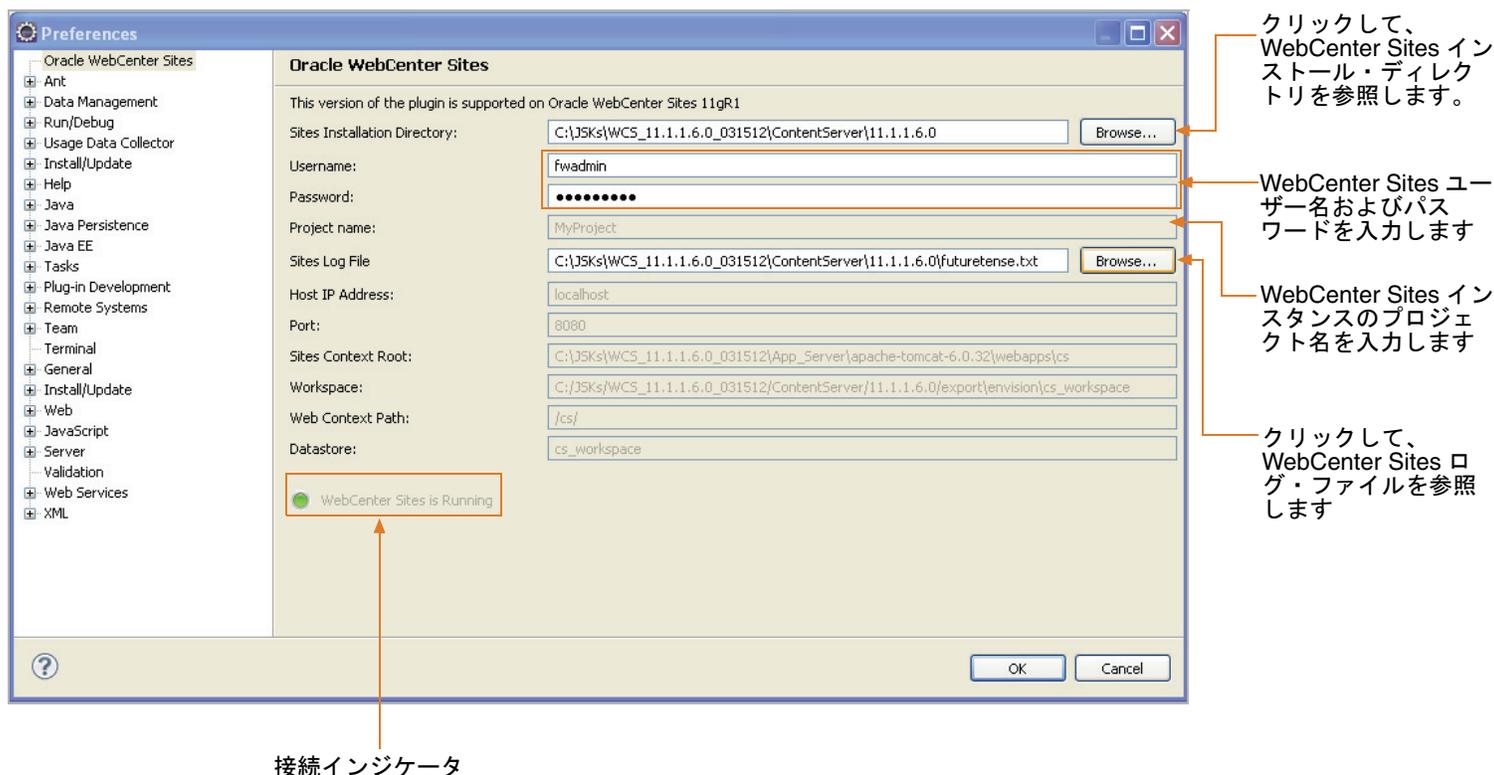
Oracle パースペクティブには次が含まれます。

- 構成画面
- Eclipse のプロジェクトとワークスペース

- 開発者ツール・ビュー
- データ同期 (エクスポート / インポート) ツール

## 構成画面

構成画面は、WebCenter Sites と統合された Eclipse への初回アクセス時に自動的に開きます。2 回目以降のアクセス時には、WebCenter Sites ツールバーの「構成」ボタンを選択すると、構成画面が開きます。この画面では、作業する WebCenter Sites インスタンスを指定できます。



構成画面では、WebCenter Sites インストール・ディレクトリのパス、RestAdmin グループに属する WebCenter Sites ユーザー、この WebCenter Sites インスタンスのプロジェクト名および WebCenter Sites ログ・ファイルのパスが要求されます。必要な情報をすべて入力すると、開発者ツールによって WebCenter Sites インスタンスの他の多数のパラメータが決定されて、読取り専用フィールドにそれらの情報が表示されます。さらに、接続インジケータによって、開発者ツールが指定の WebCenter Sites インスタンスに接続可能かどうかが表示されます。

## Eclipse のプロジェクトとワークスペース

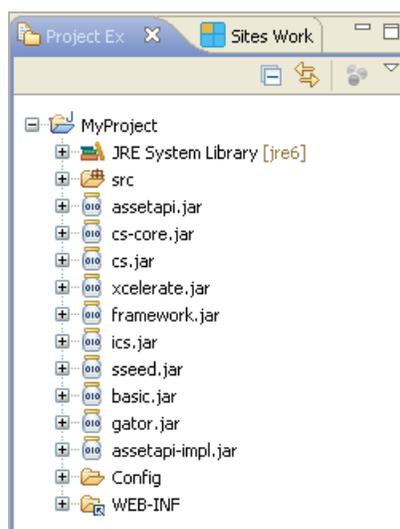
Eclipse を通じてアクセスする各 WebCenter Sites インスタンスには、Eclipse プロジェクトが割り当てられます。Eclipse プロジェクトのフォルダは、「Project Explorer」ビューに表示されます。プロジェクトは、開発者ツール・ワークスペース・アイテムの追跡のために必要です。デフォルトでは、各 WebCenter Sites インスタンスについて1つのプロジェクトのみが作成され、1つのプロジェクトによって対応できる WebCenter Sites インスタンスは1つのみです。

### 注意

プロジェクトの主な目的は、情報の追跡を促進し、Eclipse イベントを処理することです。プロジェクトは開発者ツール・プラグインによって管理されます。プロジェクトは開いたり、閉じたり、変更したりしないでください。

各開発者ツール Eclipse プロジェクトには、次のエレメントが含まれます。

- **src:** 現在の WebCenter Sites インスタンスの開発者ツール・ワークスペース・フォルダ。このフォルダには、開発者ツール・ワークスペースに格納されるリソースの全ファイルが含まれます。このフォルダ内のリソースは、バージョン・コントロール・システムにチェックインできます。
- **Config:** 現在の WebCenter Sites インスタンスに属する共通の構成ファイルへのリンク。
- **WEB-INF:** 現在の WebCenter Sites インスタンスの WEB-INF フォルダへのリンク。

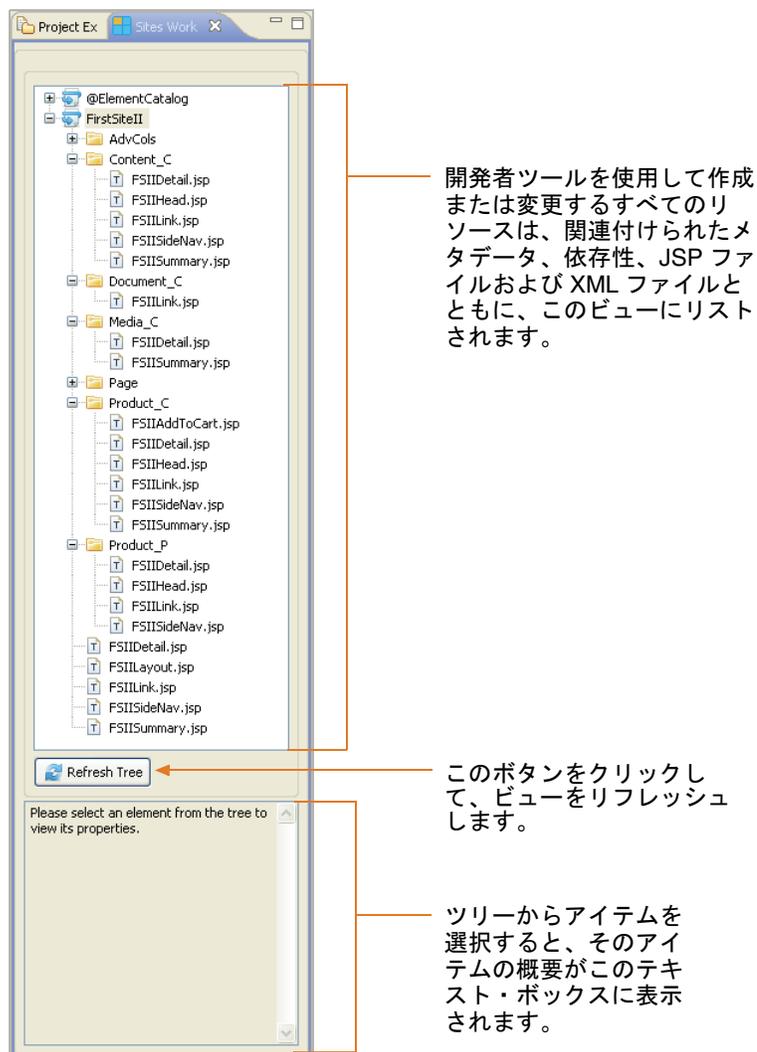


## 開発者ツール・ビュー

- Sites ワークビュー
- Sites ログビュー
- Sites プレビュー・ブラウザビュー
- Sites UI ビュー
- Sites ロギング構成ビュー
- Sites 開発者リファレンスビュー
- ウィザード

## Sites ワークビュー

このビューでは、コード関連のリソースにアクセスできます。リソースはサイト・アフィリエイトに従ってグループ化されます。リソースを選択すると、そのリソースの概要がビュー下部のテキスト・ボックスに表示されます。

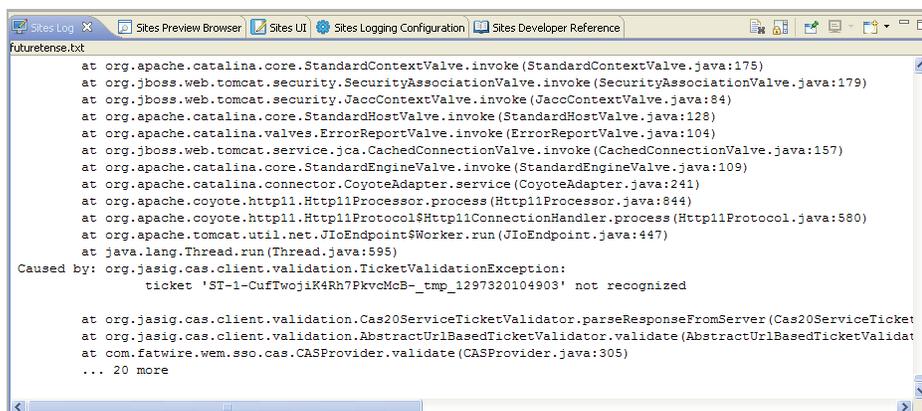


ツリーのリソースを右クリックして、使用可能な管理オプションを表示します。表示されるオプションは、選択したリソースによって異なります。

- **メタデータの表示**: 選択したアイテムのメタデータを含む `.main.xml` ファイルへのショートカット。
- **サイト・エントリ**: リソースのサイト・エントリの表示、サイト・エントリの他のサイトとの共有、新しいサイト・エントリの作成、サイト・エントリの削除を行います。
- **共有**: このリソースが関連付けられたサイトを管理します。
- **プロパティ**: キャッシュ基準やデフォルト引数など、このリソースのプロパティを管理します。
- **削除**: このリソースを削除します。

## Sites ログビュー

このビューには、WebCenter Sites ログ・ファイルの動的に更新されるレコードが表示されます。このビューを使用して、IDE と統合された WebCenter Sites インスタンスの動作を監視できます。



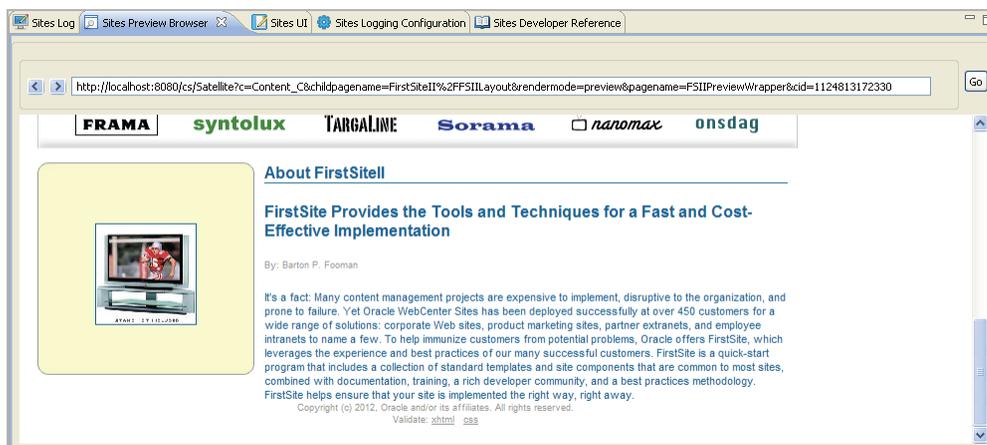
```
futuretense.txt
at org.apache.catalina.core.StandardContextValve.invoke (StandardContextValve.java:175)
at org.jboss.web.tomcat.security.SecurityAssociationValve.invoke (SecurityAssociationValve.java:179)
at org.jboss.web.tomcat.security.JaccContextValve.invoke (JaccContextValve.java:84)
at org.apache.catalina.core.StandardHostValve.invoke (StandardHostValve.java:128)
at org.apache.catalina.valves.ErrorReportValve.invoke (ErrorReportValve.java:104)
at org.jboss.web.tomcat.service.jca.CachedConnectionValve.invoke (CachedConnectionValve.java:157)
at org.apache.catalina.core.StandardEngineValve.invoke (StandardEngineValve.java:109)
at org.apache.catalina.connector.CoyoteAdapter.service (CoyoteAdapter.java:241)
at org.apache.coyote.http11.Http11Processor.process (Http11Processor.java:844)
at org.apache.coyote.http11.Http11Protocol$Http11ConnectionHandler.process (Http11Protocol.java:580)
at org.apache.tomcat.util.net.JIoEndpoint$Worker.run (JIoEndpoint.java:447)
at java.lang.Thread.run (Thread.java:595)
Caused by: org.jasig.cas.client.validation.TicketValidationException:
    ticket 'ST-1-CuffTwoj1K4Rh7FkvcMcB-_tmp_1297320104903' not recognized

at org.jasig.cas.client.validation.Cas20ServiceTicketValidator.parseResponseFromServer (Cas20ServiceTicket
at org.jasig.cas.client.validation.AbstractUrlBasedTicketValidator.validate (AbstractUrlBasedTicketValidat
at com.fatwire.wem.sso.cas.CASProvider.validate (CASProvider.java:305)
... 20 more
```

## Sites プレビュー・ブラウザビュー

このビューでは、ページをすばやくプレビューできます。このビューを使用して Web ページをプレビューするには、アドレス・バーに URL のページの名前を入力し、[Enter] を押すか「移動」をクリックします。現在のページをリフレッ

シユするには、[Ctrl] キーを押しながら [r] キーを押すか、「移動」をクリックします。



## Sites UI ビュー

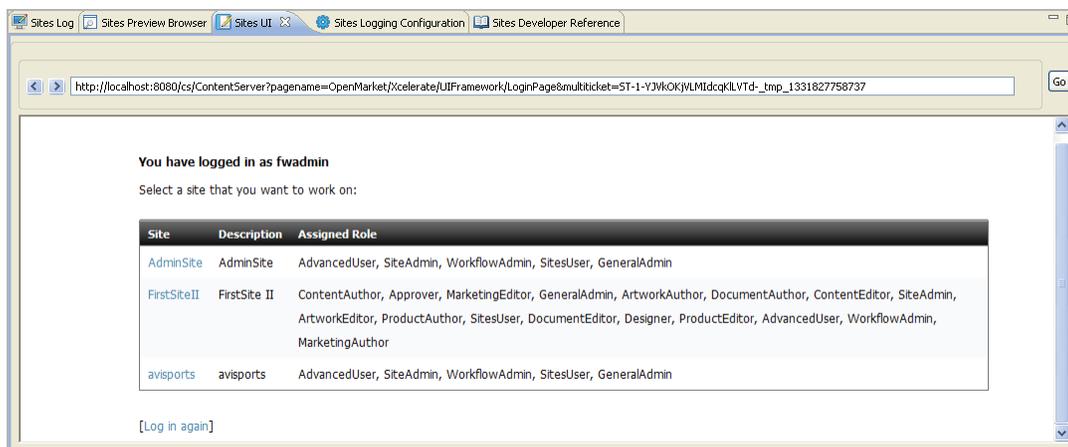
このビューでは、WebCenter Sites 管理インタフェースが組み込みブラウザに表示されます。これは、スタンドアロン・ブラウザで管理インタフェースを使用することに相当します。

### 注意

WebCenter Sites 管理インタフェースでは、Java アプレットを使用して左側のペインが表示されます。ブラウザ・ビューでのアプレットの実行に関する詳細は、次の URL にある Eclipse FAQ を参照してください。

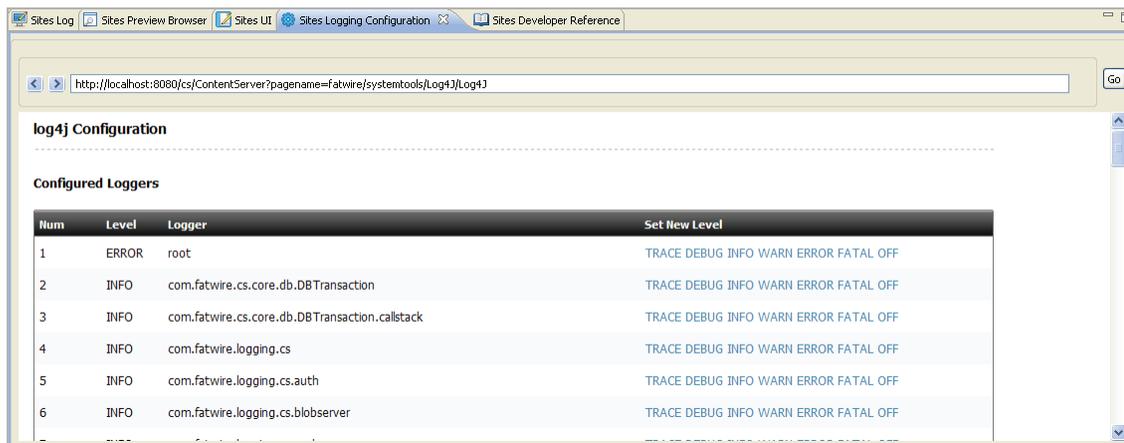
<http://www.eclipse.org/swt/faq.php#browserapplets>

アプレットを実行できない場合は、アプレットを使用しない管理インタフェース・モードを使用するか、スタンドアロン・ブラウザを使用して管理インタフェースで作業を行ってください。



## Sites ロギング構成ビュー

WebCenter Sites システムで Apache log4j が使用されている場合、このビューには動的に更新される log4j 構成画面が表示されます。log4j 構成画面では、現在のログ出力の表示、ログ出力レベルの変更、新規ログ出力の追加およびログの検索を実行できます。



log4j 構成画面の使用方法の詳細は、『Oracle WebCenter Sites 管理者ガイド』の「システム・ツール」の章を参照してください。

## Sites 開発者リファレンスビュー

このビューには2つのタブが含まれており、1つは Oracle WebCenter Sites タグ・リファレンスが、もう1つには Javadoc が表示されます。この情報は、タグ・リファレンスと Javadoc を Eclipse と関連付けている場合のみ表示されます。そうでない場合、このビューにはタグ・リファレンスと Javadoc を Eclipse と関連付け

るための手順が表示されます。詳細は、開発者ツールの設定の 20 ページの手順 7 も参照してください。

タグ・リファレンスと Javadoc が Eclipse と関連付けられていない場合、タブには次のように表示されます。

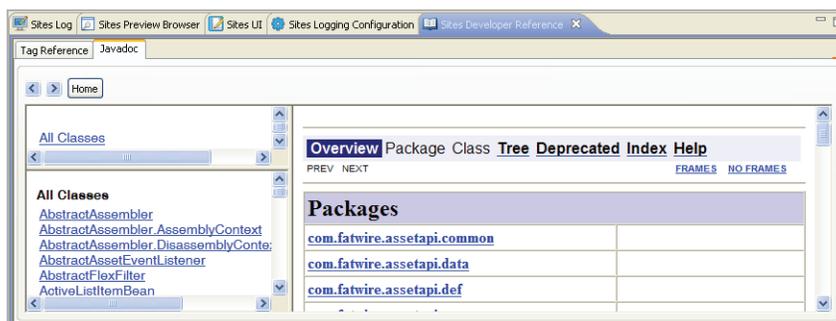


タグ・リファレンスと Javadoc のダウンロードおよびインストール手順

タグ・リファレンスと Javadoc が Eclipse と関連付けられている場合、タブには次のように表示されます。



タグ・リファレンスタブ

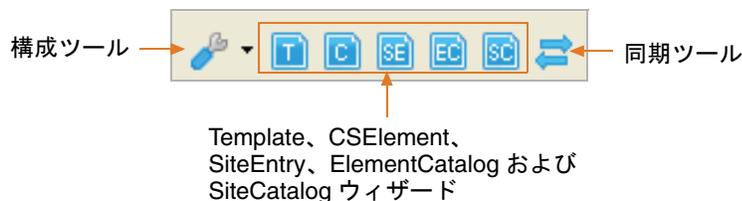


Javadoc タブ

## ウィザード

ウィザードは Oracle メニューまたは WebCenter Sites ツールバーから起動でき、これを使用してコードベースの WebCenter Sites リソースを作成できます。使用

可能なウィザードは、SiteEntry、CSElement、Template、ElementCatalog、SiteCatalog、構成ツールおよび同期ツールです。



## データ同期 (エクスポート/インポート) ツール

同期ツールには次の 2 つのタブがあります。

- ワークスペースとの同期 (WebCenter Sites からのエクスポート)
- WebCenter Sites との同期 (WebCenter Sites へのインポート)

### ワークスペースとの同期 (WebCenter Sites からのエクスポート)

ワークスペースとの同期を使用して、IDE と統合された WebCenter Sites から開発者ツール・ワークスペースヘータをエクスポートします。選択したリソースはこのプロセス時に開発者ツールによってシリアライズされ (データベース表現からファイルに変換され)、シリアライズされた表現は開発者ツール・ワークスペースにコピーされます。その後、Eclipse でリソースを変更できます。

図 4:  同期アイコン → ワークスペースとの同期タブ

Resource Type	Resource Id	Name	Description	Modified Date	Site
Template	1121304726674	FSILayout (Typeless, status=ED)	Generates an overall layout of a pag...	2005-08-29 14:50:30.0	FirstSiteII
Template	1121304726669	FSISummary (Typeless, status=PL)	Calls the Summary template directly, ...	2005-07-14 00:33:09.0	FirstSiteII
Template	1121304726698	FSIDetail (Typeless, status=ED)	Calls the Detail template directly, wit...	2005-08-30 11:29:19.0	FirstSiteII
Template	1121316511891	FSILink (Typeless, status=ED)	Calls the Link template directly, with ...	2005-07-14 00:50:17.0	FirstSiteII
Template	1121316511900	FSISideNav (Typeless, status=ED)	Generates the side nav bar given c, c...	2005-07-14 00:53:18.0	FirstSiteII
Template	1121316511962	FSIDetail (AdvCols, status=ED)	Generates a detailed view of the asset.	2005-06-28 22:03:45.0	FirstSiteII
Template	1121316511980	FSISummary (AdvCols, status=PL)	Generates a short summary for the a...	2005-07-14 01:31:09.0	FirstSiteII
Template	1121316511987	FSILink (AdvCols, status=PL)	Generates a link to the detail view for...	2005-07-14 01:32:25.0	FirstSiteII
Template	1121316511994	FSIHead (AdvCols, status=PL)	Generates the asset-specific header ...	2005-07-14 01:33:42.0	FirstSiteII
Template	1121316512001	FSISideNav (AdvCols, status=ED)	Generates the entry for this particula...	2005-07-14 01:35:21.0	FirstSiteII
Template	1121316512011	FSIDetail (Page, status=ED)	Generates a detail view of a Page. It...	2005-06-25 01:48:30.0	FirstSiteII
Template	1121394903860	FSISummary (Page, status=PL)	Generates a short summary for the a...	2005-07-14 22:37:29.0	FirstSiteII
Template	1121394903869	FSILink (Page, status=PL)	Generates a link to the detail view for...	2005-07-14 22:38:55.0	FirstSiteII
Template	1121394903876	FSIHead (Page, status=PL)	Generates the asset-specific header ...	2005-07-14 22:40:13.0	FirstSiteII
Template	1121394903883	FSISideNav (Page, status=ED)	Generates the entry for this particula...	2005-08-24 23:47:09.0	FirstSiteII
Template	1121394903892	FSIStandardDetailView (Page, statu...	Generates the detail for a standard.p...	2005-08-30 12:14:53.0	FirstSiteII
Template	1121398075705	FSITopNav (Page, status=ED)	Generates the top header, given the ...	2005-07-15 00:40:48.0	FirstSiteII
Template	1121398075721	FSIBottomNav (Page, status=PL)	Generates the footer, given the hom...	2005-07-15 00:36:04.0	FirstSiteII
Template	1121398075735	FSIStandardSideNavView (Page, sta...	Generates a simple side nav bar that ...	2005-06-30 12:34:47.0	FirstSiteII
Template	1123788998850	FSIProfileDetailView (Page, status=ED)	Generates the detail for a user profil...	2005-06-30 12:35:31.0	FirstSiteII
Template	1123788998868	FSIProfileSideNavView (Page, status...	Generates a side nav bar for user pr...	2005-06-30 12:36:07.0	FirstSiteII
Template	1124746375703	FSIDetail (Content_C, status=ED)	Renders the full version of the Conte...	2005-08-30 12:48:07.0	FirstSiteII
Template	1124746375728	FSISummary (Content_C, status=ED)	Creates a summary for the Content_...	2005-08-28 22:37:13.0	FirstSiteII
Template	1124813172560	FSILink (Content_C, status=PL)	This renders a link to content. It req...	2005-08-23 13:29:19.0	FirstSiteII
Template	1124813172578	FSIHead (Content_C, status=ED)	Displays the headline of the article fo...	2005-08-30 10:59:24.0	FirstSiteII
Template	1124813172646	FSISideNav (Content_C, status=ED)	The side nav bar for an article is the ...	2005-06-29 04:14:29.0	FirstSiteII
Template	1124902550532	FSIHead (Product_C, status=ED)	This template displays the name of th...	2005-06-30 10:59:54.0	FirstSiteII
Template	1124902550646	FSIDetail (Product_C, status=ED)	This template renders the product an...	2005-09-01 15:18:55.0	FirstSiteII
Template	1124902550794	FSIProductSideNavView (Page, stat...	This displays the product side nav ba...	2005-08-30 12:30:30.0	FirstSiteII
Template	1124902550803	FSIProductDetailView (Page, status...	This displays the detail view of the pr...	2005-08-30 12:15:19.0	FirstSiteII
Template	1124902551058	FSISideNav (Product_C, status=PL)	This displays the side nav for a Produ...	2005-08-24 23:33:02.0	FirstSiteII

### WebCenter Sites からワークスペースにアイテムをエクスポートするには:

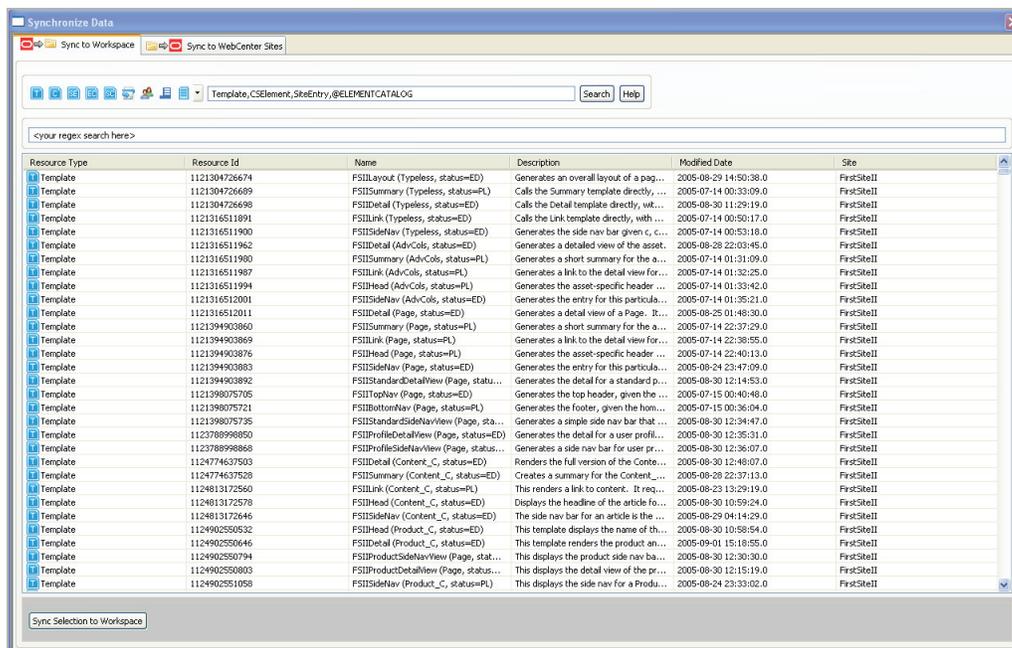
1. エクスポートするアイテムを選択します。(アイテムのリストを絞り込むには、**regex** 検索バーに移動し、検索するアセット・タイプの名前を入力します。複数のアセット・タイプを検索するには、カンマ区切りのリストを入力します。)
2. 選択項目をワークスペースと同期をクリックします。

開発者ツール・ワークスペースにエクスポートしたアセットが、Sites ワークツリー・タブにリストされます。

### WebCenter Sites との同期 (WebCenter Sites へのインポート)

WebCenter Sites との同期を使用して、開発者ツール・ワークスペースから IDE と統合された WebCenter Sites にリソースをインポートします。選択したリソースはこのプロセス時に開発者ツールによってそのネイティブ WebCenter Sites 表現に変換され、WebCenter Sites データベースにコピーされます。

図 5:  同期アイコン → WebCenter Sites との同期タブ



### ワークスペースから WebCenter Sites にアイテムをインポートするには:

1. インポートするアイテムを選択します。(アイテムのリストを絞り込むには、**regex** 検索バーに移動し、検索するアセット・タイプの名前を入力します。複数のアセット・タイプを検索するには、カンマ区切りのリストを入力します。)
2. 選択項目を WebCenter Sites と同期をクリックします。

## 次の手順

このガイドでは以降、Eclipse IDE で開発者ツールによって提供される WebCenter Sites の機能の使用方法について説明していきます。Eclipse での JSP の開発の詳細は、次の章 ( [第 4 章「JSP の開発」](#) ) に進んでください。

## 第 4 章

# JSP の開発

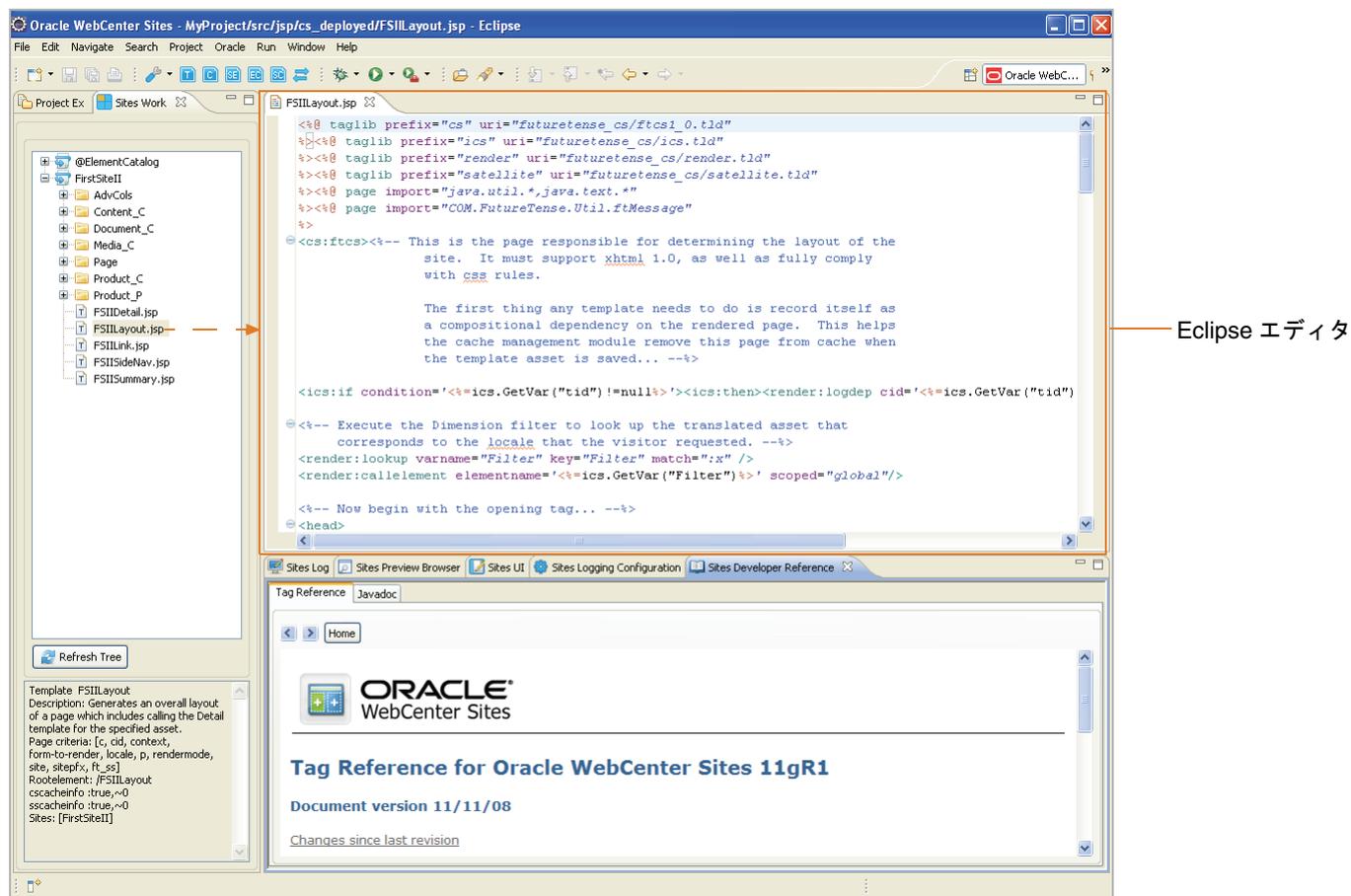
この章には、開発者ツールを使用して WebCenter Sites JSP を開発する方法に関する情報が含まれています。この章は、次の項で構成されています。

- [開発者ツールを使用した JSP の開発](#)
- [タグおよび Java API 補完](#)
- [デバッグ](#)

## 開発者ツールを使用した JSP の開発

開発者ツール・キットでは、ネイティブ Eclipse JSP エディタを使用した WebCenter Sites JSP の開発がサポートされます。Eclipse JSP エディタには、WebCenter Sites タグ、Java API 補完、構文の強調表示およびデバッグのサポートが含まれています。図 6 では、Eclipse エディタでの WebCenter Sites JSP の例を示しています。

図 6: Eclipse JSP エディタ



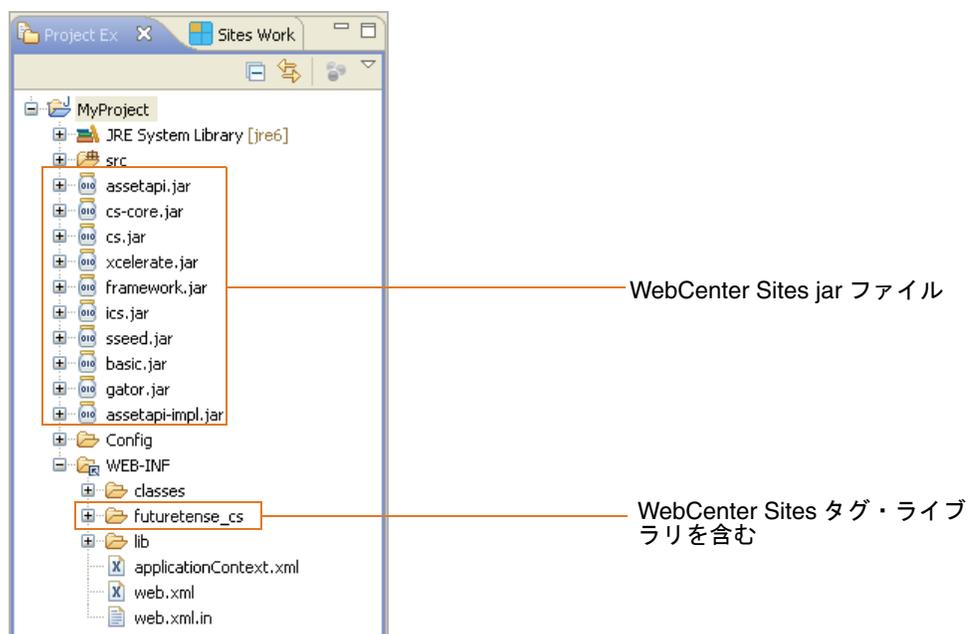
WebCenter Sites JSP には、ページのキャッシング、結果セットのキャッシング、および Template アセット、CSElement アセット、ElementCatalog エントリなどの関連メタデータを含めることができます。JSP のメタデータを使用して、WebCenter Sites は JSP を追跡および管理できます。開発者ツールは、JSP およびその対応するメタデータの追跡を含む、JSP の基礎となる WebCenter Sites プロセスを透過的に処理します。Eclipse で JSP を保存する場合、開発者ツール・キットは自動的にこれらの変更を現在の WebCenter Sites インスタンスと同期します。JSP に関連付けられているメタデータも、WebCenter Sites と自動的に同期されます。これにより、Eclipse で JSP を保存するとすぐに、WebCenter Sites で変更を表示できます。

たとえば、JSP が Template アセットに関連付けられている場合、開発者ツール・キットは更新された JSP とともに Template アセットを保存します。

## タグおよび Java API 補完

Eclipse では、タグおよび Java API 補完機能を提供します。Eclipse では、現在の WebCenter Sites インスタンスに属するタグ・ライブラリおよび jar ファイルを使用して、WebCenter Sites 関連タグおよび Java API 用の適切なコード補完を提供します。WebCenter Sites タグ・ライブラリおよび jar ファイルは、Eclipse プロジェクトに自動的にリンクされ、Eclipse プロジェクト・フォルダ (「Project Explorer」ビューにある) 内に含まれています。

- タグ・ライブラリは、WEB-INF フォルダ下の futuretense\_cs フォルダに含まれています。
- jar ファイルは、主要な Eclipse プロジェクト・フォルダ下に含まれています。



### 注意

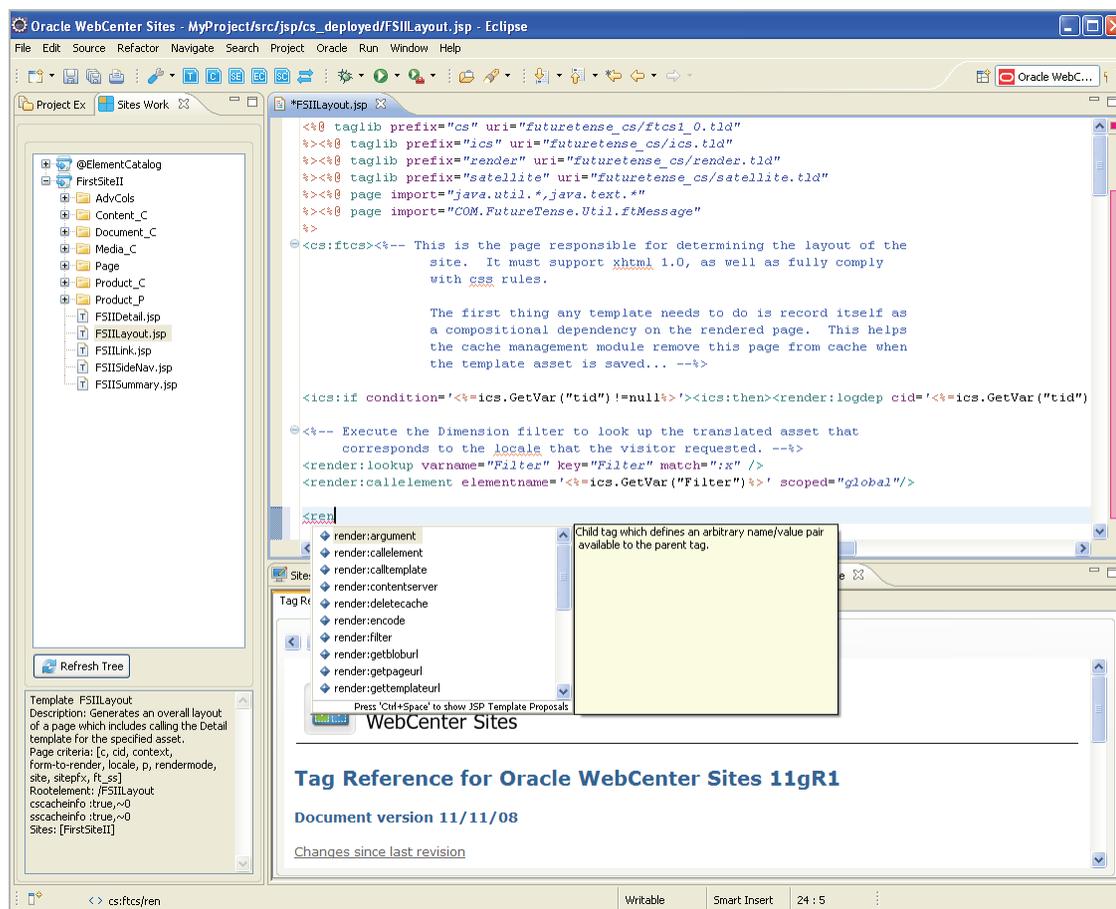
タグおよび Java API 補完機能を使用する場合は、次の点を念頭に置いてください。

- 厳格な JSP コーディング標準に確実に従ってください。これにより、コードをアプリケーション・サーバー上にデプロイできます。
- Eclipse コード補完により、WebCenter Sites jar 内に含まれているすべてのパブリック Java メソッドが表示されます。WebCenter Sites ドキュメント内にある API のみを使用してください。ドキュメントに記載されていない機能の使用は、リスクが高く、サポートされていません。

Oracle WebCenter Sites Javadoc とタグ・リファレンスを Eclipse に関連付けると、タグおよび Java API 補完機能が WebCenter Sites JSP を管理する際に使用する各

タグと Java コードの断片に関する情報を表示できます。たとえば、WebCenter Sites JSP で作業していて、タグ名の入力を開始すると、コード補完の提案をリストするウィンドウが表示されます。タグ・リファレンスと Javadoc を Eclipse に関連付けている場合は、各提案に関する情報を含む 2 番目のポップアップ・ウィンドウが表示されます(図 7 を参照)。

図 7: タグおよび Java API 補完機能



タグおよび Java コード補完機能に機能を追加するほか、Javadoc およびタグ・リファレンスは両方とも、「Sites Developer Reference」ビューにアクセスできます。詳細は、30 ページの「Sites 開発者リファレンスビュー」を参照してください。

## デバッグ

開発者ツールで Java および JSP コードをデバッグするには、WebCenter Sites を実行する JVM プロセスにデバッガを最初にアタッチする必要があります。リモート・デバッグでこれを実行することをお勧めします。WebCenter Sites JVM にアタッチするには、次の URL の Eclipse によって提供される手順に従ってください。

<http://www.ibm.com/developerworks/library/os-ecbug/>

JVM がデバッガにアタッチされると、JSP および Java コード、ビュー変数などでブレークポイントを設定できます。



## 第 5 章

# 同期とデータ交換

この章では、開発者ツールによってサポートされているエクスポートおよびインポート機能について説明します。また、この章では、**WebCenter Sites** インスタンスと開発者ツール・マッピング・プロセス間のリソース交換についても説明します。

この章は、次の項で構成されています。

- [開発者ツールを使用した同期](#)
- [データ交換とマッピング](#)

## 開発者ツールを使用した同期

同期は WebCenter Sites インスタンスとその関連ワークスペース間のリソースの双方向フローです。開発者ツールを使用して、次の同期操作を実行できます。

- 組込み依存性解決と ID マッピングを持つアセットをエクスポートおよびインポートします。
- フレックス・ファミリーや AssetMaker アセット・タイプなどの、アセット・タイプをエクスポートおよびインポートします。
- サイト定義、ロール、スタート・メニュー・アイテムおよびツリー・タブをエクスポートおよびインポートします。
- SiteCatalog および ElementCatalog エントリをエクスポートおよびインポートします。
- (コマンドライン・ツール操作) サイトの再マッピングを実行します。(たとえば、WebCenter Sites CM サイトにインポート可能な再使用可能なモジュールを作成するなど。)

指定されたサイトのすべてのリソースをエクスポートまたはインポートすると、バージョン・コントロール・システムでサイト全体を追跡できます。上級開発者はコマンドライン・ツールを使用して、再使用可能なモジュール(カスタム・ワークスペース)を作成することにより、あるサイトのリソースを別のサイトに再マッピングできます。

## 同期シナリオ

シナリオによって、リソースは自動または手動のいずれかで同期されます。

WebCenter Sites と Eclipse 間のリソースは、次のアクションが Eclipse で実行される場合に自動的に同期されます。

- コード・ベースのリソース (Templates、CSElements、SiteEntries、ElementCatalog エントリおよび SiteCatalog エントリ) が、Eclipse で開発者ツール・ウィザードを使用して作成されます。
- 開発者ツール・ワークスペースに格納されたコード・ベースのリソース (Templates、CSElements および ElementCatalog エントリ) が、Eclipse で編集されます。これには、JSP ファイル、XML ファイル、メタデータおよびリソースに関連付けられている他のファイルへの編集が含まれます。

たとえば、Eclipse エディタでリソースの関連付けられている JSP ファイルを編集する場合、開発者ツール・キットは WebCenter Sites インスタンス内に変更を自動的に同期します。Eclipse エディタを使用して、上級管理者はフレックス定義のメタデータ・ファイル (.main.xml) を編集することもできます。また、開発者ツール・キットは WebCenter Sites 内に変更を自動的に同期します。ただし、フレックス定義を編集する場合は WebCenter Sites 管理者インタフェースを使用することをお勧めします。

場合によっては、Eclipse IDE の同期ツールまたは (上級管理者の場合) コマンドライン・ツールのいずれかを使用して、リソースを手動で同期する必要があります。手動同期は次の場合に必要となります。

- Eclipse エディタが、開発者ツール・ワークスペースに格納されているリソースの編集に使用されない場合。たとえば、リソースが共有ネットワー

ク・ファイル・システムまたはバージョン・コントロール・システムから開発者ツール・ワークスペースにコピーされる場合などです。

- WebCenter Sites リソースが、WebCenter Sites インタフェースで変更される場合。

### 注意

Eclipse IDE は埋め込まれた WebCenter Sites 管理者インタフェースを提供します。ただし、Eclipse はこのインタフェースを使用して行われる変更を検出しません。したがって、埋め込まれた管理者インタフェースでの作業は、管理者インタフェースを実行しているスタンドアロン・ブラウザでの作業と同じです。

- WebCenter Sites は、Eclipse IDE でリソースを作成または編集している間は実行されません。WebCenter Sites が再起動したら、作成または編集したリソースを手動で同期する必要があります。

リソースの同期にコマンドライン・ツールを使用するのは、サーバーをテストするためにデプロイされるナイトリー・ビルドなど、主にデプロイメントのためです。たとえば、上級開発者は、自動化されたデプロイメント手順用のスクリプトに同期コマンドを埋め込むことができます。コマンドライン・ツールの実行および使用方法の詳細は、第 7 章「コマンドライン・ツール」を参照してください。

## 依存性解決

WebCenter Sites リソースは、多くの場合、他のリソースに依存します。たとえば、フレックス・アセットは、関連付けられたフレックス定義が作成される前に存在している必要があります。また、フレックス定義は、属性セットおよびおそらく他のリソースに依存します。したがって、すべてのフレックス・コンストラクトは、フレックス・ファミリーがシステム上に存在している必要があります。フレックス・アセットを空の WebCenter Sites システムにインポートするには、最初にフレックス・アセットが関連付けられるフレックス・ファミリーを作成する必要があります。その後、次の手順を実行します。

1. フレックス属性を作成します。(名前、住所、年齢など。)
2. 必要なフレックス親定義を作成します。
3. フレックス定義を作成します。
4. 必要なフレックス親を作成します。
5. フレックス・アセットを作成します。

フレックス・アセットをエクスポートすると、開発者ツール・キットはそのアセットのすべての依存性解決を実行し、その依存性のすべてを自動的にエクスポートします。したがって、必要なリソース(必要なフレックス・アセットなど)

を選択しさえすれば、開発者ツール・キットはアセットの依存性のすべてを計算します。

### 注意

開発者ツール・キットはサイト定義上でリソースの依存性を解決しません。これにより、サイト全体またはサイトのサブセットをエクスポートまたはインポートしたいか、あるいはサイト定義を完全に無視するかどうかを選択できます(たとえば、コマンドライン・ツールを使用して、任意のサイトにインポートできる再使用可能なモジュールを作成している場合)。再使用可能なモジュールの作成の詳細例は、[付録 B「コマンドライン・ツールを使用した再利用可能モジュールの作成」](#)を参照してください。

## データ交換とマッピング

開発者ツール・キットは ID とサイト・マッピング・プロセスを使用して、開発者が WebCenter Sites インスタンス間のリソースを交換できるようにします。この項は、次のトピックで構成されています。

- [ID マッピング](#)
- [サイト・マッピング](#)

### ID マッピング

WebCenter Sites で作成された各リソースは、一意のローカル ID を割り当てられます。リソースのローカル ID は、作成された WebCenter Sites インスタンスにのみ一意です。複数の WebCenter Sites インスタンスがリソースの作成に使用されるため、個別の WebCenter Sites インスタンス上の 2 つの異なるリソースが同じローカル ID を持つことが可能です。

一意にリソースを識別するために、開発者ツール・キットは、各リソースに対して、すべての WebCenter Sites インスタンス間で一意な、グローバル一意識別子 (fw\_uid) を割り当てます。また、WebCenter Sites インスタンスにリソースをインポートする場合、開発者ツール・キットはそのインスタンス上のそのリソースに新しいローカル ID を割り当てます。リソースが他のアセット (アソシエーション、アセット・ポインタ、フレックス定義など) を参照する場合、新しいローカル ID がそれらのアセットのそれぞれに対して生成されます。その WebCenter Sites インスタンスへの後続のインポートで、リソースは同じローカル ID が割り当てられます。開発者ツール・キットは、すべての WebCenter Sites インスタンス間でリソースの fw\_uid 値を維持します。リソースおよび参照され

るアセットが元の WebCenter Sites インスタンスにインポートされる場合、開発者ツール・キットはローカル ID を元の値に再マップします。

### 注意

Template アセット、フレックス属性およびツリー・タブなどの特定の WebCenter Sites リソースには一意の名前の制約があります。名前の競合を避けるために、各リソースがすべての WebCenter Sites インスタンス間で一意に名前が付けられていることを確認してください。

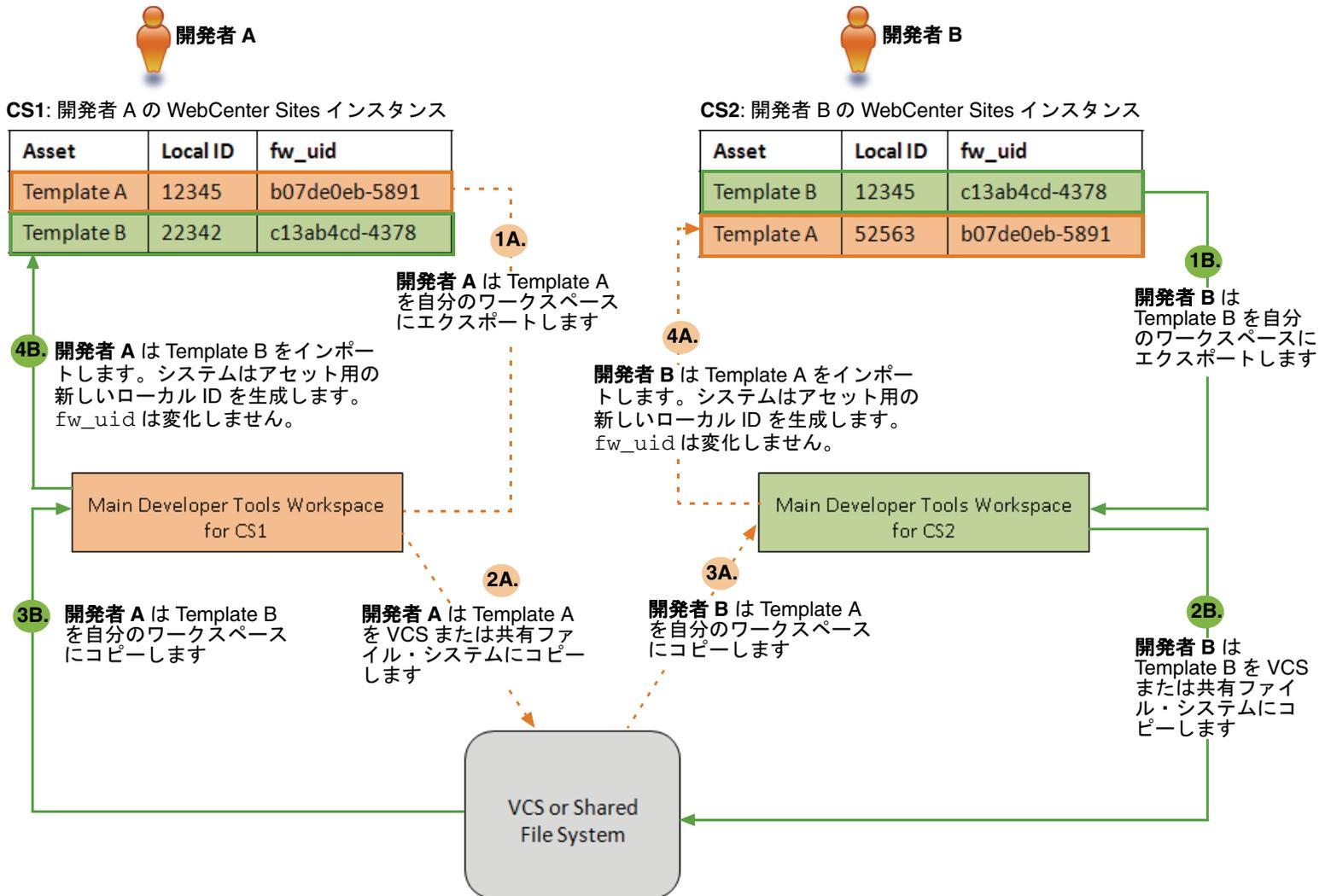
たとえば、(46 ページの図 8 で示すように) 開発者 A が CS1 という名前の WebCenter Sites インスタンスを使用していて、開発者 B が CS2 という名前の WebCenter Sites インスタンスを使用しているとします。両方の開発者は完全に異なる Template アセットを作成しました。開発者 A は Template A を作成し、開発者 B は Template B を作成しました。この 2 つの Template アセットは異なる fw\_uid 値と異なる名前を持っています。ただし、ローカル ID がランダムに割り当てられているため、偶然に両方の Template アセットが同じローカル ID (12345) を割り当てられている場合があります。開発者 A および B はその他の WebCenter Sites インスタンス間で Template アセットを交換したいと思っています。開発者 A は Template B を CS1 インスタンスにインポートしたいと思っており、開発者 B は Template A を CS2 インスタンスにインポートしたいと思っています。

図 8 は、WebCenter Sites インスタンス間で Template アセットを交換するための両方の開発者が取る手順について説明しています。両方の Template アセットのローカル ID は、その他の開発者の WebCenter Sites インスタンスにインポートされるときに再マップされます。Template A が CS2 インスタンスにインポートされる際、システムはそれにローカル ID 52563 を割り当てます。Template B が CS1 インスタンスにインポートされる際、システムはそれにローカル ID 22342 を割り当てます。それぞれの場合において、両方の Template アセットの fw\_uid 値は同じままです。

### 注意

WebCenter Sites インスタンス間でリソースを交換するため、この例の開発者は VCS または共有ファイル・システムを使用します。VCS の使用方法の詳細は、第 8 章「バージョン・コントロール・システムとの統合に関する注意」を参照してください。

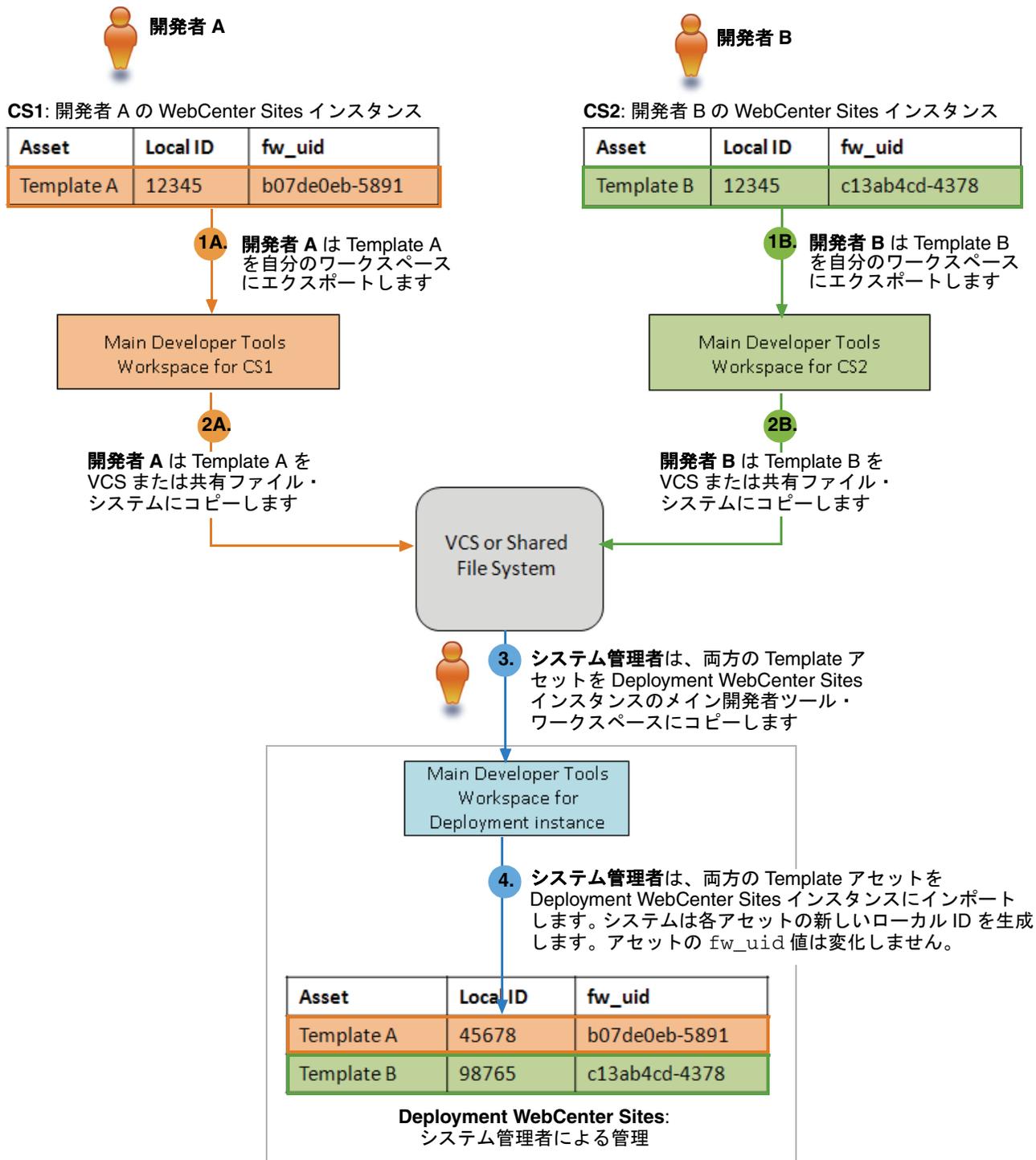
図 8: 2つの WebCenter Sites インスタンス間で同じローカル ID を持つ 2つの異なるアセットの交換



47 ページの図 9 では、開発者 A は Template A を Deployment WebCenter Sites インスタンス (システム管理者によって管理される) にデプロイしたいと思ひ、開発者 B は Template B を同じインスタンスにデプロイしたいと思ひています。両方の Template アセットは同じローカル ID (12345) を持ちます。

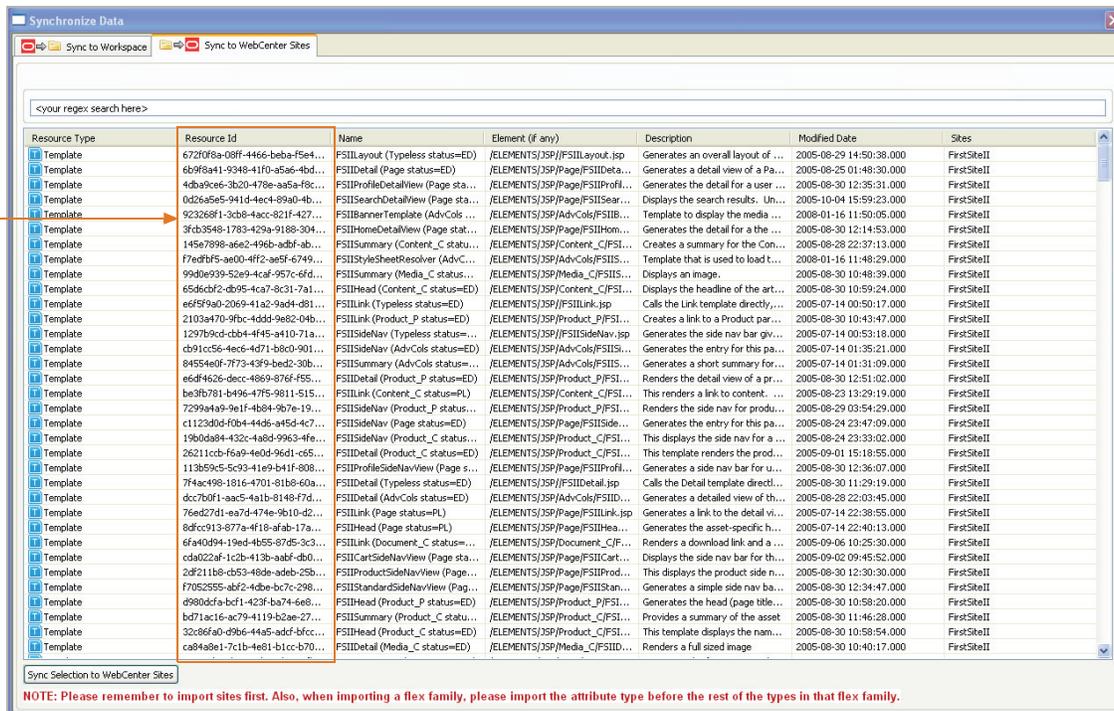
開発者 A と B は、自分たちの WebCenter Sites インスタンスのメイン開発者ツール・ワークスペースに Template をそれぞれエクスポートします。その後、Template を VCS または共有ファイル・システムにコピーします。ここから、システム管理者は両方の Template アセットを Deployment WebCenter Sites のメイン開発者ツール・ワークスペースにコピーします。その後、システム管理者は、2つの Template アセットをワークスペースから Deployment WebCenter Sites にインポートします。インポート時に、システムは両方の Templates に新しいローカル ID を割り当てます。Template A は、45678 のローカル ID を割り当てられ、Template B は 98765 のローカル ID を割り当てられます。アセットの fw\_uid 値は変化しません。

**図 9:** 3 番目の WebCenter Sites インスタンスへの同じローカル ID を持つ 2 つの異なるアセットのデプロイ



リソースは、ワークスペースにエクスポートされると、fw\_uid によってのみ識別されます。ElementCatalog および SiteCatalog エントリは、これらのエントリがエレメント名によって一意に識別されるため、fw\_uid を割り当てられません。

Resource ID 列は、fw\_uid 別に各 WebCenter Sites リソースをリスト表示します (ElementCatalog および SiteCatalog エントリを除く)。



## リソースの fw\_uid のオーバーライド

リソースが作成されると、UUID 値はグローバルな一意識別子として自動的に生成され、fw\_uid という名前のアセット属性に格納されます。上級開発者はアセット API を使用し、fw\_uid 属性を変更することによって、デフォルトの fw\_uid スキームを独自のスキームでオーバーライドします。アセット API の使用方法の詳細は、『Oracle WebCenter Sites 管理者ガイド』および Oracle WebCenter Sites Javadoc を参照してください。

### 注意

デフォルトの WebCenter Sites fw\_uid スキームの使用をお勧めします。リソースのデフォルト fw\_uid 値をオーバーライドする場合は、値がすべての WebCenter Sites インスタンス間で一意であることを確認する必要があります。リソースの fw\_uid 属性を設定したら、値を変更しないでください。

## 既存のリソースと開発者ツールの併用

Oracle WebCenter Sites システムが FatWire Content Server からのアップグレードである場合、その既存のリソースの一部の `fw_uid` 値が `CSSystem:[type]:id` に設定されている場合があります。ただし、Content Server 7.6 以降では、リソースの `fw_uid` は UUID 値として生成されます。開発者ツールは、リソースの `fw_uid` 値がグローバルに一意であるかぎり、`fw_uid` 値のいずれかのタイプでリソースをマップできます。したがって、既存のリソースの現在の `fw_uid` 値を (`CSSystem:[type]:id` の形式で) 引き続き使用できます。

開発者ツールを既存のリソースと併用する場合は、以下のいずれか(または両方)を実行してください。

- 既存のリソースの `CSSystem:[type]:id` の `fw_uid` 値を引き続き使用することをお勧めします。ただし、他の WebCenter Sites インスタンスが異なるリソースに同じ `fw_uid` 値を生成していないことを確認する必要があります。たとえば、WebCenter Sites 開発インスタンスがあり、管理インスタンスにリソースをパブリッシュしている場合、パブリッシュされたリソースの `fw_uid` 値は両方のインスタンス上で同じままです。したがって、開発者ツールを使用したこれらの 2 つのインスタンス間のリソースの同期によって、ID 競合が発生しません。
- 個別の FatWire Content Server インスタンスで作成されているが、同一の `fw_uid` 値を持つ既存のリソースがある場合、これらのリソースのそれぞれが、新しい、一意の `fw_uid` 値を割り当てられる必要があります。ID 競合を避けるには、現在の `fw_uid` 値を削除して、開発者ツールが WebCenter Sites からリソースをエクスポートする際に新しい UUID 値を生成できるようにするか、または独自の一意 ID をリソースに割り当てることができます。手順については、48 ページの「リソースの `fw_uid` のオーバーライド」を参照してください。

### 注意

新しい `fw_uid` をリソースに割り当てる場合は、新しい `fw_uid` 値をそのリソースのすべてのインスタンスに必ず割り当ててください。たとえば、`fw_uid` 値を変更する前に、リソースを別の WebCenter Sites インスタンスにパブリッシュする場合は、そのリソースの両方のコピーに同じ `fw_uid` を必ず割り当ててください。

## サイト・マッピング

アセットなどのほとんどの WebCenter Sites リソースは、少なくとも 1 つのサイトに関連付けられています。リソースが WebCenter Sites インスタンスからワークスペースにエクスポートされる場合は、その `.main.xml` ファイルに、関連付けられているサイトの完全な (canonical) リストが格納されます。リソースの canonical リストは、新しいサイト・アフィリエイトを追加するか、現在のサイト・アフィリエイトを削除するかまたは (上級管理者の場合) コマンドライン・ツールを使用してリソースのナチュラル・サイト・マッピングをオーバーライドする場合を除いて、すべての WebCenter Sites インスタンス上で同じままです。

## ナチュラル・サイト・マッピング

デフォルトでは、開発者ツールは、リソースの `.main.xml` ファイルに格納された `canonical` リストを参照することによって、関連サイトにリソースをマップします。このリストで参照されるサイトがリソースがインポートされる `WebCenter Sites` インスタンス上に存在する場合、開発者ツールはリソースをそれらのサイトにマップします。リソースの `canonical` リストで参照されるサイトのどれも `WebCenter Sites` インスタンス上に存在しない場合、インポートは失敗します。

たとえば、開発者 A が 2 つのサイト (`News` と `Sports`) をインストールするとします。別の `WebCenter Sites` インスタンス上で、開発者 B も 2 つのサイト (`News` と `Weather`) をインストールするとします。両方の開発者は同じ `Template` アセットを `WebCenter Sites` インスタンスにインポートします。この `Template` アセットは、`Sports` と `Weather` の両方のサイトに関連付けられます (両方のサイトがアセットの `canonical` リストで参照されます)。インポート時に、開発者ツールは `Template` アセットの `canonical` リストを参照し、そのアセットを開発者 A の環境の `Sports` サイトと開発者 B の環境の `Weather` サイトにマップします。

開発者 A と B が `Template` アセットに行った変更を互いに共有する場合、開発者ツールはそのアセットを両方の `WebCenter Sites` インスタンス上の適切なサイトにマップします。`canonical` リストにより、開発者ツールは、`Template` アセットが関連付けられているサイトを認識できます。これは、これらのサイトの一部がインストールされていないインスタンスにアセットがエクスポートされる場合であっても同様です。

## コマンドライン・ツールを使用したナチュラル・サイト・マッピングのオーバーライド

上級開発者はコマンドライン・ツールを使用して、`canonical` リストで参照されないサイトにリソースをインポートできます。コマンドライン・ツールにより、任意のサイトにインポート可能なリソースを含むワークスペースである、再使用可能なモジュールを作成できます。

たとえば、開発者が `FirstSiteII` サンプル・サイト内にブログ・ソリューションを作成するとします。このソリューションには、`Flex`・ファミリー、アセット、`Templates` などのリソースが含まれています。開発者は、まだ存在していないサイトを含む、様々なサイトにリソースをインポートしたいと思っています。上級管理者であるため、コマンドライン・ツールを使用して、空のワークスペースに必要なリソースをエクスポートし、このワークスペースのコンテンツをアーカイブします (`.zip` または `.tar` フォーマットを使用)。コマンドライン・ツールを使用して、他の開発者がこのモジュールに含まれるリソースのサイト・マッピングをカスタマイズし、モジュールがインポートされるサイトを手動で指定できます。

コマンドライン・ツールの使用方法の詳細は、第 7 章「[コマンドライン・ツール](#)」を参照してください。また、再使用可能なモジュールの作成の詳細シナリオは、[付録 B「コマンドライン・ツールを使用した再利用可能モジュールの作成」](#)を参照してください。

## 第 6 章

# ワークスペース

この章には、開発者ツールが、統合された **WebCenter Sites** インスタンスからエクスポートされたリソースを格納する方法に関する情報が含まれています。

この章は次の内容で構成されています。

- [概要](#)
- [ワークスペースの構造](#)

## 概要

ワークスペースとは、ワークスペースの WebCenter Sites インスタンスまたは別のインスタンスのワークスペースからのリソースを示すシリアル化された WebCenter Sites データのデータベースのリポジトリです。ワークスペースには、アセット、フレックス・ファミリー、サイトなどを含むあらゆるタイプの WebCenter Sites リソースを格納できます。各ワークスペースは、1つの WebCenter Sites インスタンスに関連付けられています。

デフォルトで、Eclipse では各 WebCenter Sites インスタンスに、Eclipse IDE での作業時に連続的な開発に使用される主要な開発者ツール・ワークスペース (Eclipse プロジェクト・フォルダにある) を提供します。カスタム・ワークスペースは、開発者ツール・コマンドライン・ツールを使用して上級開発者が作成できます。カスタム・ワークスペースは、モジュールの作成など、特殊なプロジェクトに使用できます。(カスタム・ワークスペースの作成方法の詳細は、第7章「コマンドライン・ツール」を参照してください。)

バージョン・コントロール・システム (Subversion など) や共有ファイル・システムを使用すると、1つのワークスペースに格納されたリソースは他のワークスペースと交換できます。WebCenter Sites インスタンスから関連したワークスペースにエクスポートされたすべてのリソースは、別の WebCenter Sites インスタンスのワークスペースにコピーできます。これにより、リソースが2番目のワークスペースの関連付けられた WebCenter Sites インスタンスへのインポートに使用できるようになります。異なるワークスペース間のリソースの共有方法の詳細は、第8章「バージョン・コントロール・システムとの統合に関する注意」を参照してください。

## ワークスペースの構造

ワークスペースは、WebCenter Sites インストール・ディレクトリ内の export/envision フォルダ下に作成されます。主要な開発者ツール・ワークスペースは、export/envision/cs\_workspace フォルダ下にあります。主要な開発者ツール・ワークスペースは、Eclipse プロジェクト・フォルダの唯一の表示可能なワークスペースです。

すべてのワークスペースは、同じ構造を持ちます。ワークスペースに含まれている各リソースは、単一ファイルまたはいくつかの関連するファイルとして格納されます。各リソースの主要なファイルは、.main.xml で終了し、リソース固有のメタデータが含まれています。この主要ファイルには、リソース (添付ドキュメント、JSP ファイルまたは BLOB など) に関連付けられている他のファイルへのリンクも含まれています。これにより、リソースの関連ファイルがワークスペースに格納される限り、各リソースは完全に自己完結型とすることができます。そうでない場合、リソースは不完全です。

リソースが複数のファイルを持つ場合、これらのファイルは storable0、storable1 などのように .main.xml ファイルの下部の項にリスト表示されます。指定されたリソースの関連ファイルには同様の名前があります。このようにして、元のルート・パスを保持するために別々に格納される ElementCatalog エントリを除き、リソースの関連ファイルのすべてが一緒に表示されます。

ワークスペース内のリソースのファイルの場所は、リソースのタイプによって異なります。ワークスペースは次のセクションに分けられます。

- `src/_metadata`: アセット、アセット・タイプ、サイト、ロールなどを含む指定のリソースのメタデータ・セクション。また、従来の XML コードは、`ELEMENTS/` サブフォルダ下に格納されます。
- `src/jsp/cs_deployed`: このセクションは、適切なパス下にリソースの JSP ファイルを格納します。

ワークスペースは非常に一貫した構造を持つため、1つのワークスペースから別のワークスペースへリソースをコピーできます。すべてのファイル・システムのコピー操作と同様に、同じ名前を持つファイルを上書きしないようにしてください。

## アセット・ストレージ構造

アセットは、`src/_metadata/ASSET/asset type` という名前のフォルダ名の下に格納されます。この構造下には、アセット・データを含む、2つのレベルのハッシュ・ベース階層があります。アセット・ファイルの名前は、アセット名とその `fw_uid` 値に基づきます。アセットに添付ドキュメントまたは BLOB が含まれる場合、ファイル名はアセット名、属性名、`fw_uid` 値、ドキュメントまたは BLOB(ある場合)の名前に基づきます。

たとえば、*FSII IES\_Manual.pdf* という名前の `Document_C` アセットには、*IES\_MDPlayer\_Manual.pdf* という添付ドキュメントが含まれています。したがって、このアセットは2つの別々のファイルとして格納されます。

- 最初は、`.main.xml` ファイルで、アセットのメタデータおよびアセットに関連付けられているファイルへのリンクが含まれます。

```
.src/_metadata/ASSET/Document_C/8/0/FSII IES_MDPlayer_Manual.pdf(aa0b47b5-f558-49d4-a6ac2ee012d1b75).main.xml
```

- 2番目は実際のドキュメントで、この例では PDF ファイルです。

```
.src/_metadata/ASSET/Document_C/8/0/FSII IES_MDPlayer_Manual.pdf.FSII DocumentFile(aa0b47b5-f558-49d4-8a6a-c2ee012d1b75).IES_MDPlayer_Manual.pdf
```

### 注意

アセットのすべてのファイル名はアセットの名前に基づいているため、アセットの名前を変更すると、ファイルの名前も変更されます。VCS でアセットを追跡する場合は、古い名前を持つファイルを削除してください。

## コード・ベースのリソース・ストレージ構造

`Templates`、`CSElements` および `ElementCatalog` エントリは、コード・エレメントによって要求されるストレージ・パス下に格納されます。コード・ベースのリソースに関連付けられている JSP ファイルは、`src/jsp/cs_deployed` の下のワークスペースに格納され、XML エレメントは `src/_metadata/ELEMENTS` の下に格納されます。コード・ベースのリソースのメタデータ・ファイルは、`.main.xml` 拡張子が付加されたリソースの JSP と同じ名前の下に格納されます。したがって、コード・ベースのリソースのメタデータ、JSP および XML ファイルはワークスペースでグループ化されます。

## 属性エディタ・ストレージ構造

属性エディタはアセットとして追跡されますが、一連の `ElementCatalog` エントリへの暗黙の参照も持ちます。属性エディタの `ElementCatalog` エントリは個別に追跡されます。

たとえば、`TextArea` エディタが、依存関係として登録される `OpenMarket/Gator/AttributeTypes/TEXTAREA` `ElementCatalog` エントリを使用します。開発者ツールは `TextArea` エディタ用の次のファイルを維持します。

- `.main.xml` ファイル:  
`src/_metadata/ASSET/AttrTypes/9/10/TextArea(e64f983d-9c7c-489baedb-476d56f8121e).main.xml`
- `urlxml` メタデータ・ファイル:  
`src/_metadata/ASSET/AttrTypes/9/10/TextArea.urlxml(e64f983d-9c7c-489b-aedb-476d56f8121e).1095346398911.txt`
- 個別のリソースとして追跡される、`ElementCatalog` エントリ:
  - `ElementCatalog` エントリの `.main.xml` ファイル:  
`src/_metadata/ELEMENTS/OpenMarket/Gator/AttributeTypes/TEXTAREA.xml.main.xml`
  - 属性エディタの要素・コード:  
`src/_metadata/ELEMENTS/OpenMarket/Gator/AttributeTypes/TEXTAREA.xml`

## アセット・タイプ・ストレージ構造

アセット・タイプは主要なメタデータ部分と一連の要素を持ちます。たとえば、次はページ・アセット・タイプの構造です。

- ページの主要なメタデータは、`.main.xml` ファイルに格納されます。  
`src/_metadata/Asset_Type/Page(b8d8ae9-14cc-4554-b80e-0c22e39a3ec8).main.xml`
- 関連要素は個別に追跡されます (各要素は独自の `.main.xml` ファイルを持ちます):  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/SearchForm.xml`  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/CheckDelete.xml`  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/ContentForm.xml.main.xml`  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/ContentDetails.xml.main.xml`  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/LoadSiteTree.xml`  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/IndexReplace.xml.main.xml`  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/LoadTree.xml`  
`src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/IndexAdd.xml.main.xml`

```
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  SearchForm.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  IndexReplace.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  PreviewPage.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  LoadTree.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  PreUpdate.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  Tile.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  SimpleSearch.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  SimpleSearch.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  ContentForm.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  AppendSelectDetailsSE.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  LoadSiteTree.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  AppendSelectDetails.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  ManageSchVars.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  PreviewPage.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  CheckDelete.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  ManageSchVars.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  PreUpdate.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  AppendSelectDetails.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  IndexCreateVerity.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  ContentDetails.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  PostUpdate.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  IndexAdd.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  IndexCreateVerity.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  Tile.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  AppendSelectDetailsSE.xml.main.xml  
src/_metadata/ELEMENTS/OpenMarket/Xcelerate/AssetType/Page/  
  PostUpdate.xml.main.xml
```



## 第 7 章

# コマンドライン・ツール

この章では、上級開発者向けにコマンドライン・ツールの実行と使用について説明します。

この章は、次の項で構成されています。

- [概要](#)
- [コマンドライン・ツールの実行と使用](#)
- [モジュールの作成](#)

## 概要

開発者ツール・コマンドライン・ツールは、デプロイメントやその他のリソースの移動作業に使用できます。開発者ツール・ワークスペースでのみ作業可能な Eclipse 統合とは異なり、コマンドライン・ツールでは、どのワークスペースでも作業できます。コマンドライン・ツールは、Eclipse IDE で作業する場合に利用できないインポート機能やエクスポート機能も備えています。たとえば、開発者は再利用可能なモジュールを作成できます。これらのモジュールは、任意のサイトにインポート可能なリソースを含むワークスペースになります。

## コマンドライン・ツールの実行と使用

コマンドライン・ツールを実行する方法は次のとおりです。

1. WebCenter Sites の配布パッケージにある `csdt.zip` を解凍します。 `csdt-client` フォルダを開き、 `csdt-client.jar` ファイルをクラスパスに置きます。 *Oracle WebCenter Sites 証明書* マトリックスに示されるすべての要件を満たしていることを確認します。
2. コマンドライン・ツール (`cmd`) を実行して次のコマンドを入力します。

```
java com.fatwire.csdt.client.main.CSDT [ContentServer url]
      username= username password= password
      cmd=export|import|listcs|listds [options]
```

プレースホルダのパラメータを、使用する開発環境の情報や、実行する目的のコマンドに置き換えます。

- `ContentServer url`: ContentServer サーブレットを含むローカル WebCenter Sites インスタンスの URL (`http://localhost:8080/cs/ContentServer` など)
- `username` および `password`: WebCenter Sites 全体管理者のユーザー名とパスワード。このユーザーは RestAdmin グループのメンバー (`fwadmin/xceladmin` など) であることが必要です。
- `cmd`: 実行するコマンド。次のコマンドを利用できます。
  - `export`: WebCenter Sites からワークスペースにデータをエクスポートします。
  - `import`: ワークスペースから WebCenter Sites にデータをインポートします。
  - `listcs`: WebCenter Sites のコンテンツをリスト表示します。
  - `listds`: ワークスペースのコンテンツをリスト表示します。
- `options`: 次のいずれかを `import` または `export` に指定します。
  - `resources`: リソース・タイプとリソース ID をセミコロンで区切り、インポートまたはエクスポートするリソースを指定します。複数のリソースを指定するには、カンマ区切りのリストを使用します。特定タイプのリソースをすべて指定するには、\* 記号を使用します。リソースを (ワークスペースに) エクスポートする場合、リソースのローカル ID を指定します。たとえば、特定の `Content_C` アセットとすべての `Product_C` アセットをエクスポートするには、`resources=Content_C:12345;Product_C:*` を使用します。

リソースを (WebCenter Sites インスタンスに) インポートする場合、リソースの `fw_uid` を指定します。リソースの `fw_uid` を取得するには、`listds` オプションを使用します。

次に、すべてのリソース・セレクタを示します。

- @SITE: 目的のサイトを指定します。
- @ROLE: 目的のロールを指定します。
- @ASSET\_TYPE: 目的のアセット・タイプを指定します。
- @TREETAB: 目的のツリー・タブを指定します。
- @STARTMENU: 目的のスタート・メニュー・アイテムを指定します。
- @ELEMENTCATALOG: 目的の ElementCatalog エントリを指定します。
- @SITECATALOG: 目的のサイト・カタログ・エントリを指定します。
- @ALL\_NONASSETS: この略語を使用して、すべての非アセット・リソースを選択します。
- @ALL\_ASSETS: この略語を使用して、利用可能なすべてのアセットを選択します。
- `asset type`: 特定タイプのアセットを指定します。

#### 注意

インポートまたはエクスポートの前にセレクタが正しいリソースを選択しているかどうかを確認するには、エクスポート作業の場合は `listcs`、インポート作業の場合は `listds` を使用します。これらのコマンドは、移動するリソースのリストを指定することにより、実行前にセレクタを微調整します。

リソースに依存性がある場合、これらは自動的にエクスポートおよびインポートされます。ただし、依存性は、`listcs` および `listds` コマンドを使用してリスト表示されません。

- `fromSites`: 指定のサイトからのみリソースを選択します。
- `toSites`: (インポート専用) インポート時にナチュラル・サイト・アプリケーションをサイトのカンマ区切りリストでオーバーライドします。指定するサイトはターゲット・システムに存在する必要があります。
- `modifiedSince`: (アセット専用) 指定した日付以降に変更されたリソースのみを選択します。日付の形式は `yyyy-mm-dd hh:MM:ss` です。
- `datastore`: (オプション) WebCenter Sites リソースのエクスポート先ワークスペース、または WebCenter Sites リソースのインポート元ワークスペースを指定します。このパラメータの値を指定しない場合、メイン開発者ツール・ワークスペースがデフォルトで指定されます。リソースをエクスポートする際に存在しないワークスペースを指定すると、コマンドライン・ツールは、そのワークスペースを自動的に作成し、そこに目的のリソースをエクスポートします。

## コマンド例

次に、コマンドライン・ツールを使用して実行可能なコマンドの例を示します。

- このコマンドは、FirstSiteII に属して指定日以降に変更された特定の Content\_C アセットとすべての Product\_C アセットをエクスポートします。ワークスペースが指定されていないため、デフォルトで開発者ツール・ワークスペースが使用されます。

```
java com.fatwire.csdt.client.main.CSDT http://localhost:8080/
cs/ContentServer username=bob password=password
resources=Content_C:123432123423,11234234212,111234341234;Pr
oduct_C:* fromSite=FirstSiteII modifiedSince=2010-08-08
19:14:00 cmd=export
```

- このコマンドは、ワークスペースにある特定の Content\_C アセットとすべての Product\_C アセットをインポートします。ワークスペースが指定されていないため、デフォルトで開発者ツール・ワークスペースが使用されます。

```
java com.fatwire.csdt.client.main.CSDT http://localhost:8080/
cs/ContentServer username=bob password=password
resources=Content_C:aad618e9-f04e-4ee4-b902-
076224bb6f7b;Product_C:* fromSite=FirstSiteII cmd=import
```

- このコマンドは、サイト SecondSiteII のすべてのリソースを、TheOutput という名前のワークスペースにエクスポートします。

```
java com.fatwire.csdt.client.main.CSDT http://localhost:8080/
cs/ContentServer username=bob password=password
resources=@ALL_ASSETS:*;@ALL_NONASSETS:*
fromSite=SecondSiteII datastore=TheOutput cmd=export
```

- このコマンドは、TheInput という名前のワークスペースすべてのアセットとツリー・タブをサイト MySite にインポートします。

```
java com.fatwire.csdt.client.main.CSDT http://localhost:8080/
cs/ContentServer username=bob password=password
resources=@ALL_ASSETS:*;@TREETAB:* toSites=MySite
datastore=TheInput cmd=import
```

## モジュールの作成

モジュールは、WebCenter Sites インスタンスから指定のワークスペースにエクスポートされる関連リソースのセットです。datastore パラメータでは、WebCenter Sites リソースのエクスポート先ワークスペース、または WebCenter Sites リソースのインポート元ワークスペースを指定できます。存在しないワークスペースに WebCenter Sites リソースをエクスポートする場合、コマンドライン・ツールは、そのワークスペースを自動的に作成し、そこに目的のリソースをエクスポートします。

モジュールは再利用可能であり、(リソースのサイトに関する canonical リストに該当サイトがリスト表示されない場合でも) そのコンテンツはどの CM サイトにもインポートできます。モジュールを CM サイトにインポートするには、import コマンドを実行する必要があります。datastore パラメータで、目的のリソースを含むワークスペースを指定し、toSites パラメータで、それらのリ

ソースのインポート先のサイトを指定します。これにより、ワークスペースのコンテンツは指定の CM サイトにインポートされます。



## 第 8 章

# バージョン・コントロール・システムとの統合 に関する注意

この章では、開発者ツール・ワークスペース・フォルダに含まれるリソースをバージョン・コントロール・システム (VCS) に保存する方法について説明します。これにより、開発者ツール・ワークスペースのリソースを他の開発者と共有できます。

この章は次の内容で構成されています。

- [バージョン・コントロールと開発者ツール](#)

## バージョン・コントロールと開発者ツール

バージョン・コントロール・システム (VCS) には、ソース・コード・リポジトリを作成する機能があります。VCS は、ソース・ファイルのバージョンニング、ブランチ化および管理のための高度なツールを備えています。開発者ツール・ワークスペースが WebCenter Sites リソースを格納するファイル・システム構造を使用すると、そのリソースを任意の VCS に格納でき、CM サイト全体を VCS で追跡できます。

### 開発者ツールと VCS の統合

開発者ツール・ワークスペースは、Eclipse プロジェクトの src フォルダにあります。このフォルダには、WebCenter Sites インストール・ディレクトリ (export/envision/cs\_workspace/src 配下) から直接アクセスできます。開発者ツール・ワークスペース・フォルダのコンテンツを VCS にコピーするには、使用する VCS を最初に決める必要があります。次に、開発者ツール・ワークスペースに格納されているリソースを VCS にチェックインします。選択する VCS により、Eclipse IDE からのリソースのチェックインに必要な手順が決まります。

リソースを Eclipse から VCS に直接チェックインできるプラグインを提供することで、Eclipse は、選択する VCS を場合によってはサポートします。たとえば、Subversion リポジトリを使用して開発者ツール・ワークスペースのコンテンツを格納する場合、Eclipse IDE は Subclipse プラグインをサポートします。そのため、リソースを Eclipse IDE から Subversion ディレクトリに直接チェックインできます。

開発者ツール・ワークスペースは、リソースのタイプに応じて、すべてのリソースを 1 つ以上のファイルで格納します。リソースを VCS にチェックインする場合、そのリソースの関連するファイルもすべてチェックインする必要があります。たとえば、添付ドキュメント (PDF など) を含むアセットは、メタデータ・ファイル (.main.xml) や関連ドキュメント・ファイルで表されます。アセットの関連ファイルはすべて VCS にチェックインする必要があります。これを行わない場合、チェックインは失敗します。開発者ツール・ワークスペースのレイアウトの詳細やリソースのワークスペース・ファイルへのマップ方法については、第 6 章「ワークスペース」を参照してください。

#### 注意

開発者ツール・ワークスペースから VCS にデータをチェックインする場合、開発者ツール・ワークスペース・ファイル構造について詳しく理解する必要はありません。かわりに、ほとんどの VCS クライアントでは、開発者ツール・ワークスペース・フォルダの徐々に増加する変化を検出し、VCS コミット操作時にその変化を示します。

## 開発者ツールが統合された VCS での作業

WebCenter Sites リソースを開発者ツール・ワークスペースから VCS にチェックインする場合、これらのリソースを他の開発者とやり取りして、時間の経過に伴うリソースの変化を追跡できます。次に、VCS を使用して WebCenter Sites リソースを共有する開発チームの例を示します。

開発者 A が WebCenter Sites にリソースを作成し、それを開発者ツール・ワークスペースにエクスポートします。次に、開発者 A はそのリソースを VCS にチェックインします。開発者 B は、そのリソースを VCS から自身の開発者ツール・ワークスペースにチェックアウトできます。この開発者は、リソースを変更してその変更を VCS にチェックインできます。開発者 A と開発チームのそれ以外の開発者は、リソースに加えられた変更を VCS から確認できます。これにより、開発チームのメンバーは、自身の開発者ツール・ワークスペースと、リソースに加えられた最新の変更を同期させることができます。また、他の開発者は、リソースを VCS から各自の開発者ツール・ワークスペースにチェックアウトすることでグループに参加できます。プロジェクトが進みながら、リソースの追加および変更のサイクルは続きます。

### 注意

WebCenter Sites には、指定の WebCenter Sites インスタンス内で保持されるリソースのリビジョン追跡システムがあります。WebCenter Sites リビジョン追跡システムは VCS に統合できません。



## 付録

この部には次の付録があります。

- [付録 A 「開発チームの統合の使用例」](#)
- [付録 B 「コマンドライン・ツールを使用した再利用可能モジュールの作成」](#)

### 注意

ここで示す付録には、WebCenter Sites 管理インタフェースと WebCenter Sites: 開発者ツールのスクリーン・キャプチャがあります。これらのスクリーン・キャプチャは再ブランド化されていませんが、コンテンツは引き続き有効です。



## 付録 A

# 開発チームの統合の使用例

この付録には、開発者チームが開発者ツールを使用して CM サイトやリソースを作成する場合の開発シナリオがあります。開発チームは、開発者ツールで提供される同期ツールを使用して、複数の **WebCenter Sites** インスタンス同士でリソースの管理およびやり取りを行います。コマンドライン・ツールを使用する場合、CM サイトとそのリソースはナイトリー・ビルドとしてデプロイされます。

シナリオのイベントのシーケンスは次のとおりです。

- 今日 : サイトおよび関連リソースの開発
- 3日後 ... デプロイメント

## 今日：サイトおよび関連リソースの開発

### 午前 7 時 14 分：新しいプロジェクトの割当て

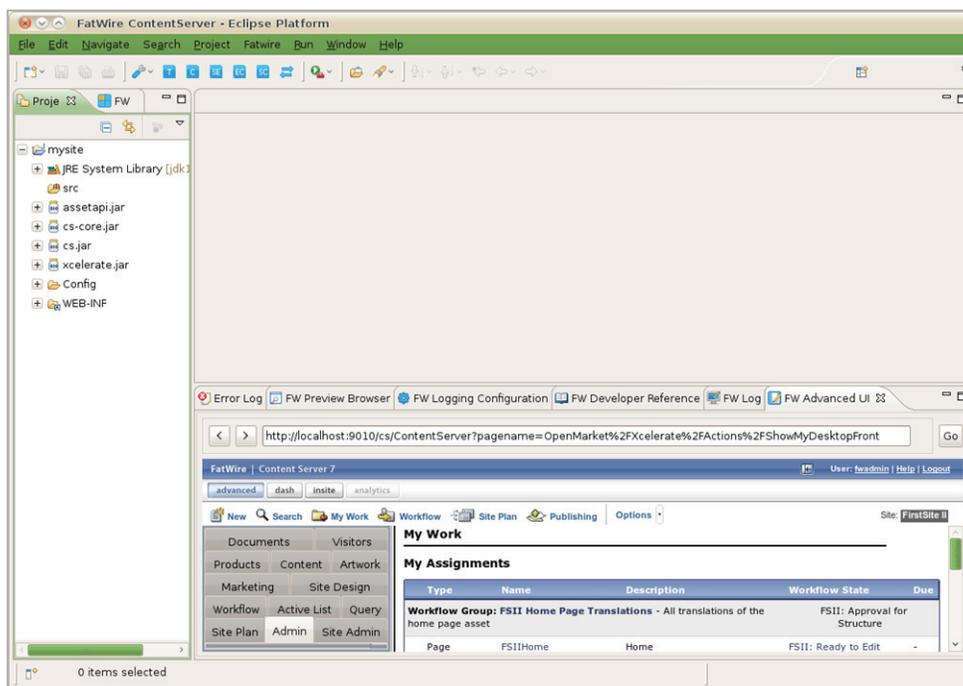
アーキテクトの Artie は目が覚めると、自分が新しい Web ベース・プロジェクトのリーダーに任命されていることに気がきます。

### 午前 7 時 34 分：開発者ツールの設定

Artie は、コーヒーを飲み、WebCenter Sites インスタンスをラップトップにインストールします。次に、Eclipse IDE を起動して開発者ツール・キットを構成します。

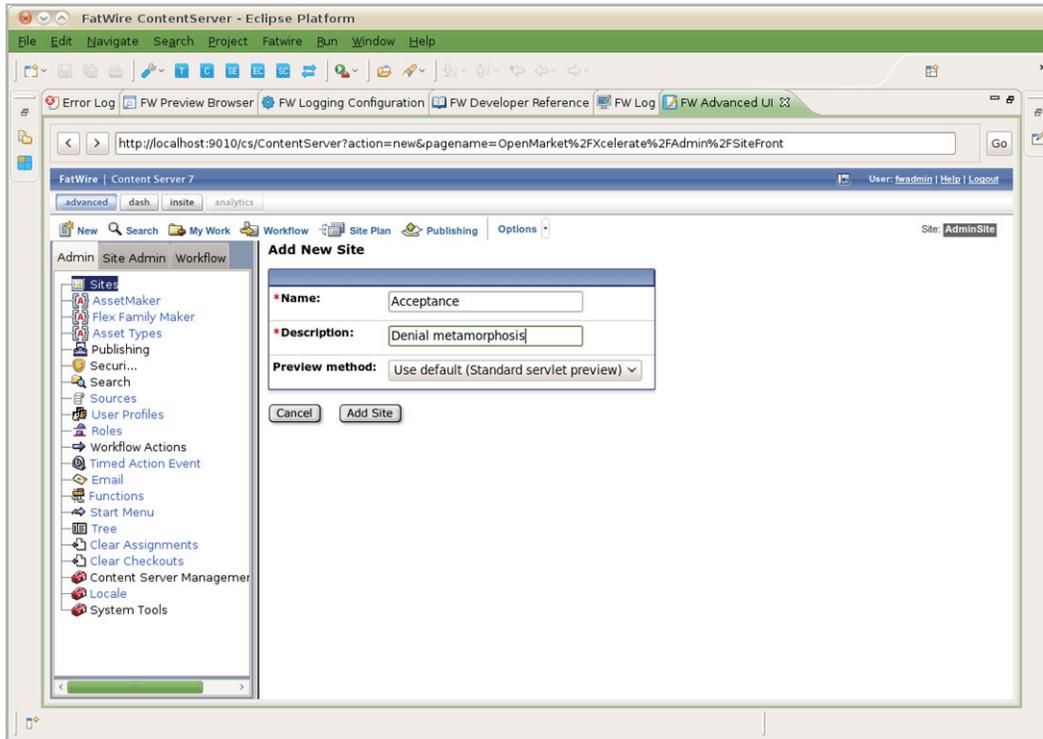
#### 注意

Eclipse と WebCenter Sites インスタンスを正常に統合するには、構成画面で全体管理者のユーザー名とパスワードを入力する必要があります。このユーザーは、RestAdmin グループのメンバーである必要があります。



## 午前 7 時 45 分：サイト定義の作成

Artie は、Eclipse の組み込みの WebCenter Sites 管理インタフェース・ビューを使用してサイト定義 (このシナリオではサイトの名前を Acceptance とする) を作成します。

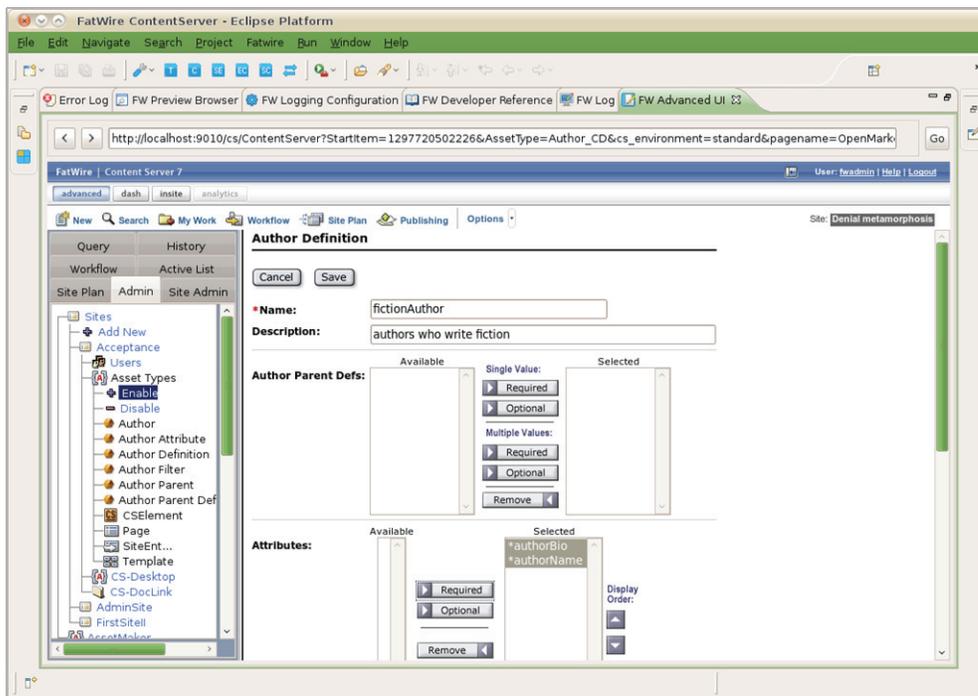


Artie は、WebCenter Sites 管理インタフェースを実行する別のブラウザ・ウィンドウを使用してサイト定義を作成しました。ただし、彼は Eclipse のかなり熱心な愛好者であるため、通常は Eclipse から離れずに完全な WebCenter Sites CM サイトを作成できるということに満足しています。

## 午前 7 時 46 分：サイトのリソースの作成

Artie は次のようにサイトを準備します。

- アセット・タイプを有効にします。
- 権限を割り当てます。
- サイトの情報アセット (このシナリオでは作成者情報アセット) を格納するフレックス・ファミリを作成して有効にします。
  - フレックス属性: Author\_A
  - フレックス親定義: Author\_PD
  - フレックス定義: Author\_CD
  - フレックス親: Author\_P
  - フレックス・アセット: Author\_C
  - フレックス・フィルタ: Author\_F
- フレックス属性 (authorName および authorBio) とフレックス定義 (fictionAuthor) を作成します。次に、属性をフレックス定義に追加します。



## 午前 8 時 12 分：VCS に関するディスカッション

Artie はオフィスに到着し、開発チームの他のメンバーである Sonoko (プログラマ)、Matthäus (プログラマ) および Yogesh (システム・エンジニア) とミーティングを行います。プロジェクトにバージョン・コントロール・システムを使用するかどうかについて話し合います。

**Yogesh:** 社内にバージョン・コントロール・システムを設定できますが、余計な作業は避けたいと思います。バージョン・コントロール・システムは本当に必要ですか？

**Artie:** このプロジェクトは数か月続く予定です。ネットワークに共有フォルダを作成して、すべての作業をそこに同期させます。ただし、お互いの作業を上書きしないように注意する必要があります。たとえば、2人が同じ **Template** アセットで作業している場合、互いに待機する必要があります。

**Sonoko:** Artie、前回のプロジェクトでは終わりが近づくにつれてかなり厳しい状況になりましたよね？このような管理上のプレッシャーを感じている場合、他の人の作業が終わるのを待っているとかなり気力が失われます。私はバージョン・コントロール・システムを使用する方がずっと良いと思います。また、今回は時間を持って余すときに **Stellarbucks** から作業できるように、リポジトリを Web 上に置くことはできますか？

**Matthäus:** Sonoko に賛成です。SVN ホスティングはただ同然で入手できます。安心のために SSL でも SVN を入手できます。

**Yogesh:** 社内 SVN を設定する時間がない場合、SVN ホスティングのサブスクリプションくらいは入手できます。

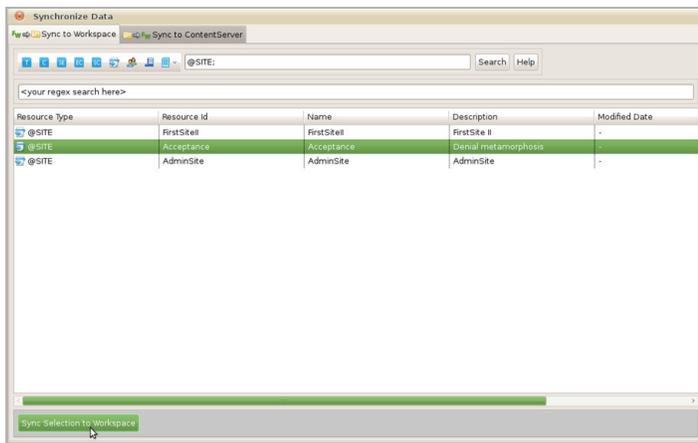
**Artie:** 了解、SVN で進めるとしよう。他に何かありますか？

Artie と開発チームのメンバーは、SVN を使用してサイトのリソースを追跡することになりました。

## 午前 9 時 42 分：ワークスペースと VCS の同期

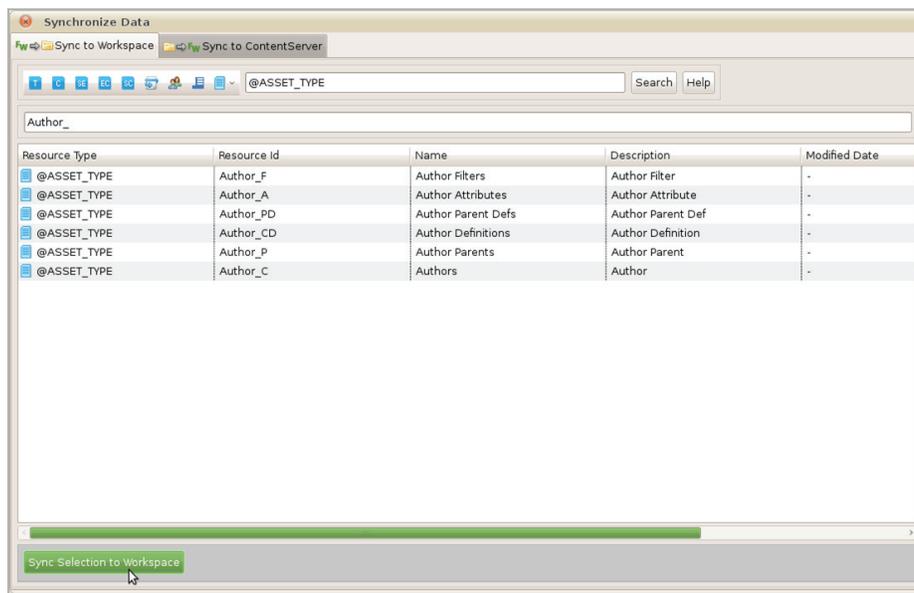
Artie と彼のチームは、<http://subclipse.tigris.org/> から Subclipse プラグインをインストールします。ここで、Artie は、以前に作成したサイトやリソースをチェックインする必要があります。

1. Artie は、Eclipse の「Developer Tools Synchronization」画面を使用して「Sync to Workspace」タブにアクセスし、検索フィールドに @site セレクタを入力して自身の WebCenter Sites インスタンス上のサイト一覧を取得します。



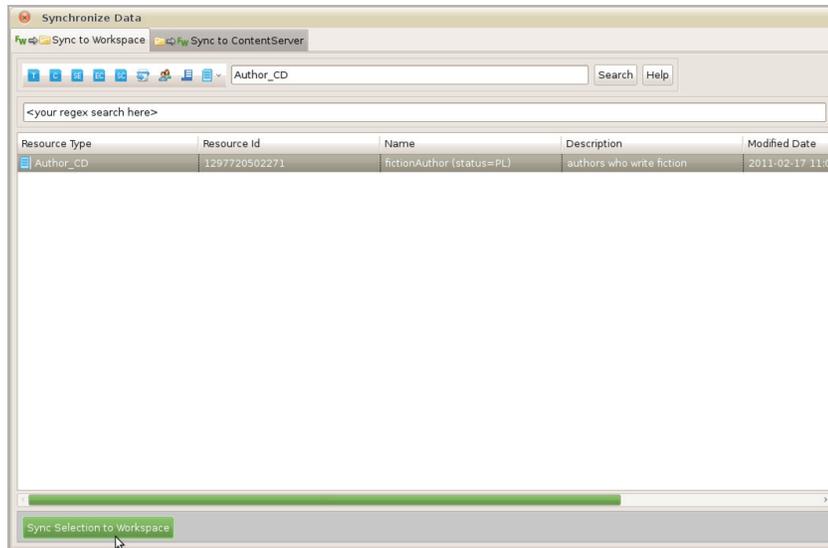
Artie は、以前に作成したサイト (Acceptance サイト) を選択し、「**Sync Selection to Workspace**」オプションをクリックしてサイト定義を自身の WebCenter Sites インスタンスからワークスペースにエクスポートします。

- 次に、サイトに関連付けられたフレックス・ファミリーをワークスペースにエクスポートします。Artie は、@ASSET\_TYPE セレクタを使用して、自身の WebCenter Sites インスタンス上のすべてのアセットをリスト表示します。結果を絞り込むには、検索文字列 Author\_ を使用します。その後、リストされているすべてのアイテムを選択して、「Sync Selection to Workspace」をクリックします。



フレックス・ファミリーのタイプは、タイプに固有の ElementCatalog エントリを含めてワークスペースにシリアルライズされます。

- ここで、Artie は、フレックス定義を自身のワークスペースにエクスポートします。該当タイプの利用可能な定義をすべてリスト表示するセクタ、Author\_CD を使用します。この場合、定義は 1 つのみです (fictionAuthor)。



### 注意

Artie は、サイト定義が依存するフレックス属性 (Author\_A インスタンス) を選択しませんでした。これは、開発者ツール・キットにより、フレックス属性が自動的に定義と同期することがわかっていたからです。

4. Artie は、Eclipse の「Project Explorer」ビューで自身のワークスペースにすばやく目を通して、自分のすべての作業内容がそこにあることを確認します。上から下まで見て、プロジェクトの src フォルダに次の内容があることを確認します。
- 同期したアセット・タイプごとの `_metadata.ASSET_TYPE` エントリ
  - 両方の `Author_A` 属性の `_metadata.ASSET.Author_A` ファイル
  - シリアライズした定義の `_metadata.ASSET.Author_CD` ファイル
  - シリアライズした各アセット・タイプに関連する `ElementCatalog` エントリの `_metadata.ELEMENTS` エントリ
  - サイト定義の `_metadata.SITE` エントリ

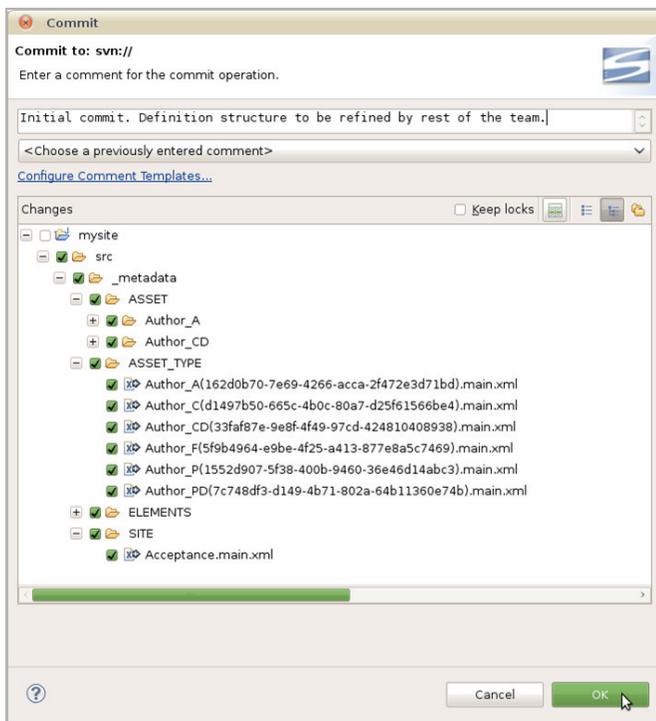


### 注意

Artie は、自身の WebCenter Sites インストール・ディレクトリの `export/envision/cs_workspace` フォルダに同じデータがあるかどうかを確認します。

すべてのリソースが Artie のワークスペース内にあるようです。ただし、これはすべて Artie のラップトップ上にあるため、チームはアクセスできません。チェックインの時間です。

5. Artie は、Subclipse を使用して、開発チームの SVN リポジトリに接続し、自身のメイン開発者ツール・ワークスペース・フォルダ (src フォルダ) を SVN リポジトリにコミットすることで自身の開発者ツール・プロジェクトを共有します。



### 注意

メイン開発者ツール・ワークスペースは、Eclipse の「Project Explorer」ビューの src フォルダにあります。src フォルダ内にあるファイルのみをコミットします。その他のファイルはすべて補助ローカル・リソースであるため、コミットしないでください。

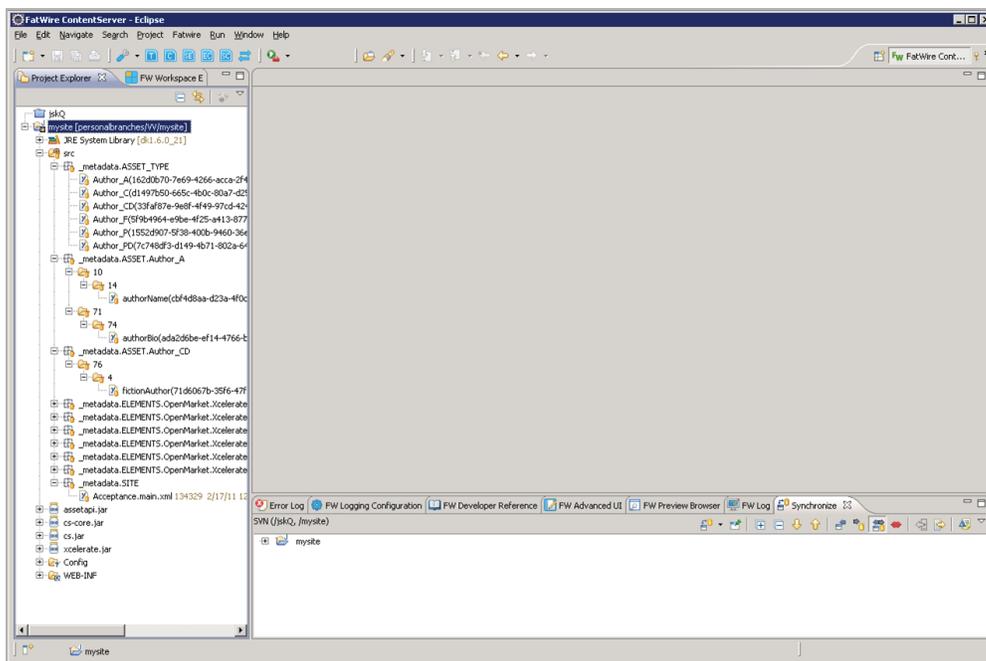
## 午前 10 時 12 分：他のチーム・メンバーのワークスペースと SVN リポジトリの同期

Sonoko と Matthäus は、各自の Eclipse 統合 WebCenter Sites インスタンスの設定を終えたところです。2 人ともそれぞれの Eclipse プロジェクトを SVN リポジトリに接続します。

Artie が以前にサイトとそのリソースを SVN リポジトリにチェックインしているため、Subclipse はターゲットの場所がすでにあることを検出します。



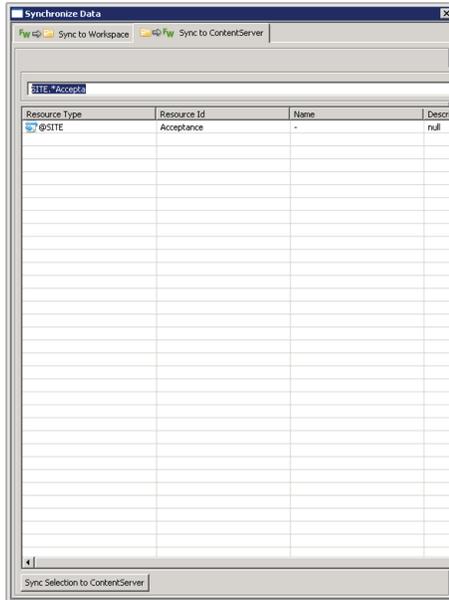
Sonoko と Matthäus は、自身のメイン開発者ツール・ワークスペースと、SVN リポジトリで入手可能な Artie 作成のリソースを同期させます。それらのリソースには、Sonoko と Matthäus のどちらのメイン開発者ツール・ワークスペースからもアクセスできるようになります。



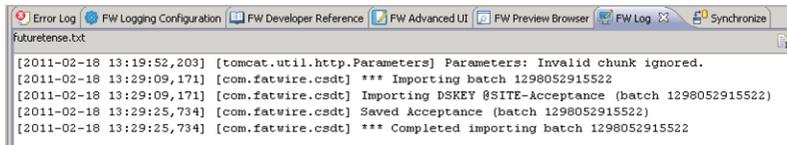
ただし、リソースは、Sonoko または Matthäus の WebCenter Sites インスタンスとまだ同期していません。



(Acceptance) を選択し、「**Sync to WebCenter Sites**」をクリックして自身の WebCenter Sites インスタンスとそのサイトを同期させます。



「FW Log」ビューを使用して、サイトが正常にインポートされていることを確認します。



2. Sonoko は、同期画面を再度開き、サイトのフレックス・ファミリーを、最初にフレックス属性 (このシナリオでは Author\_A) からインポートします。

Sonoko は自身の WebCenter Sites インスタンスで Acceptance サイトのフレックス・ファミリーを設定しなかったため、最初にフレックス属性 (Author\_A) を自身の WebCenter Sites インスタンスにインポートする必要があります。フレックス属性をインポートすると、サイトのフレックス・ファミリーを構成する残りのアセット・タイプを自身の WebCenter Sites インスタンスに同期できます。

Resource Type	Resource Id	Name	Element (if any)	Description	Modified On
@ASSET_TYPE	16303020-4e69-4326-gata-2f1...	Author_A ()		Author Attribute	2011-02-18
@ASSET_TYPE	138af67a-7e67-4f49-9703-4248...	Author_CD ()		Author Definition	2011-02-18
@ASSET_TYPE	5f5e4964-e5be-4f25-a113-877...	Author_F ()		Author Filter	2011-02-18
@ASSET_TYPE	15524907-5f38-400b-9460-36...	Author_P ()		Author Parent	2011-02-18
@ASSET_TYPE	7c748df3-d149-4b71-802a-64b...	Author_PD ()		Author Parent Def	2011-02-18
@ASSET_TYPE	d1497b50-665c-4b0c-80a7-d2...	Author_C ()		Author	2011-02-18

3. 最後に、フレックス定義を同期させると、必要な属性が自動的にインポートされます。

「FW Log」ビューでは、サイトのすべてのリソースのローカル・アセット ID が新しい WebCenter Sites インスタンスにインポートされると再マップされます。

```

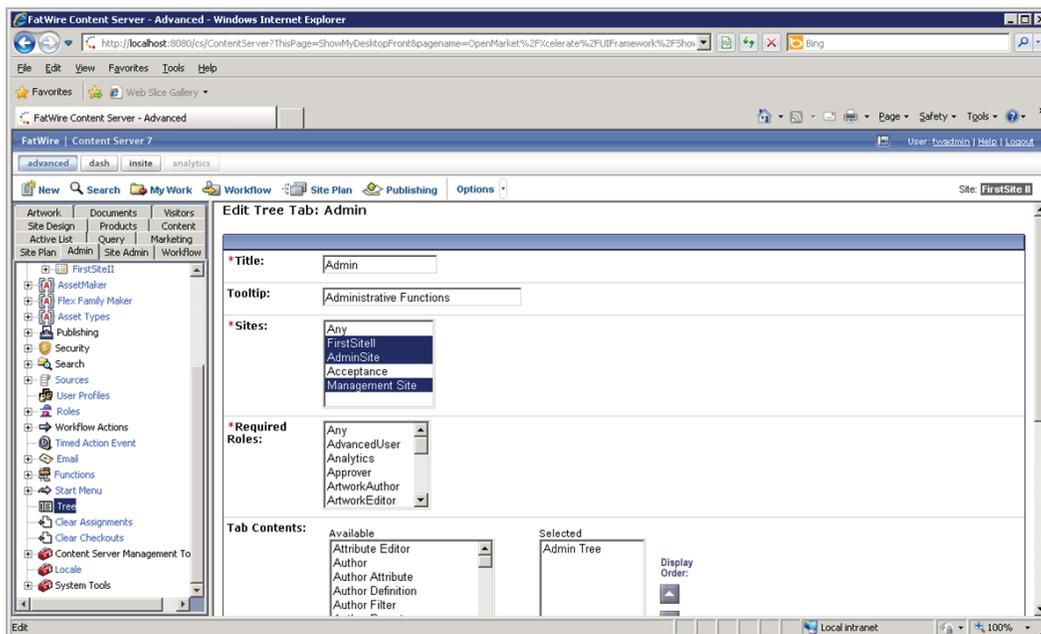
[2011-02-18 13:50:55,109] [com.fatwire.csdt] Mapped Author_CD:1297720502271 to Author_CD:1297837449943
[2011-02-18 13:50:55,109] [com.fatwire.csdt] Mapped Author_CD:1297720502271 to Author_CD:1297837449943
[2011-02-18 13:50:55,125] [com.fatwire.csdt] Mapped Author_A:1297720502260 to Author_A:1297837449935
[2011-02-18 13:50:55,125] [com.fatwire.csdt] Mapped Author_A:1297720502265 to Author_A:1297837449939
[2011-02-18 13:50:55,421] [com.fatwire.csdt] Saved Author_CD:1297837449943 (batch 1298052917205)
[2011-02-18 13:50:55,421] [com.fatwire.csdt] *** Completed importing batch 1298052917205
  
```

## 午前 10 時 21 分：サイト権限の割当て

Sonoko はリソースと自身の WebCenter Sites インスタンスの同期後、サイト権限を自身に割り当てます。これらの権限により、WebCenter Sites 管理インタフェースからサイトやそのリソースにアクセスできます。

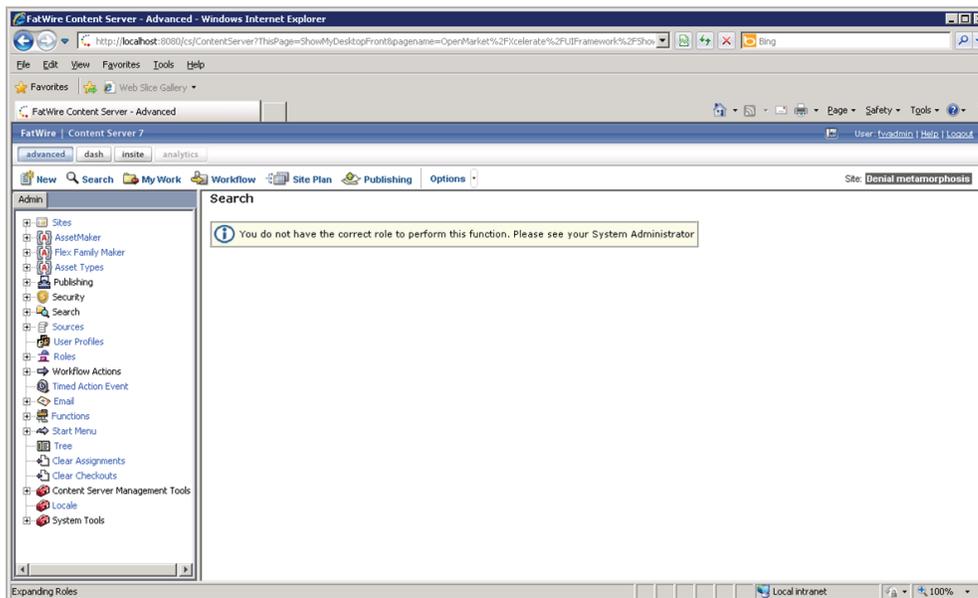
### 注意

新しいサイトのツリー・アプレットにアクセスするには、1 つ以上のツリー・タブをサイトに割り当てる必要があります。



## 午前 10 時 22 分：スタート・メニューの問題

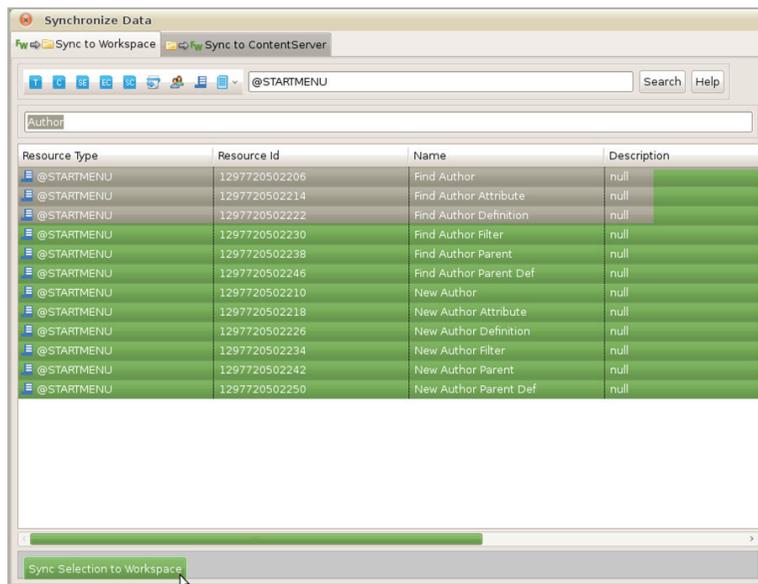
Sonoko はサイトにログインして「新規」オプションをクリックします。しかし、利用できるスタート・メニュー・アイテムがないことに気付きます。Artie がサイトのスタート・メニュー・アイテムを SVN リポジトリにチェックインしていませんでした。



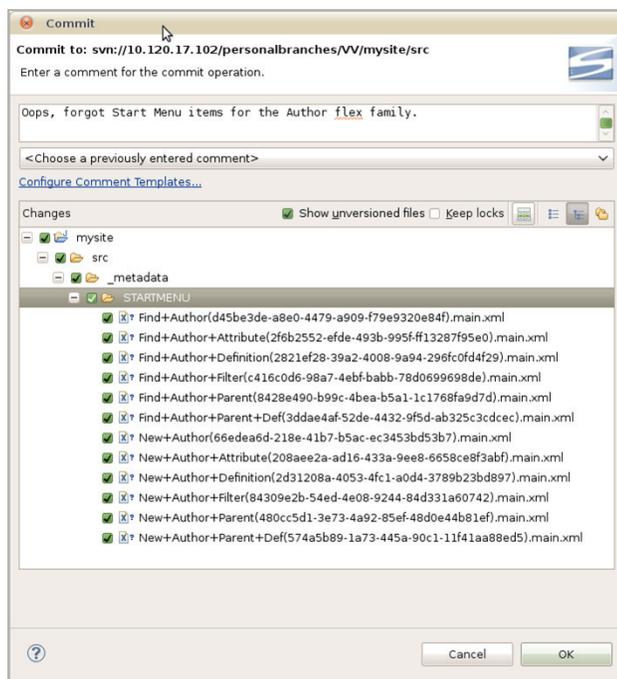
## 午前 10 時 24 分：スタート・メニューの問題の解決

Sonoko は、新しいサイトのスタート・メニュー・アイテムのチェックインを忘れていたことを知らせる IM を Artie に送ります。

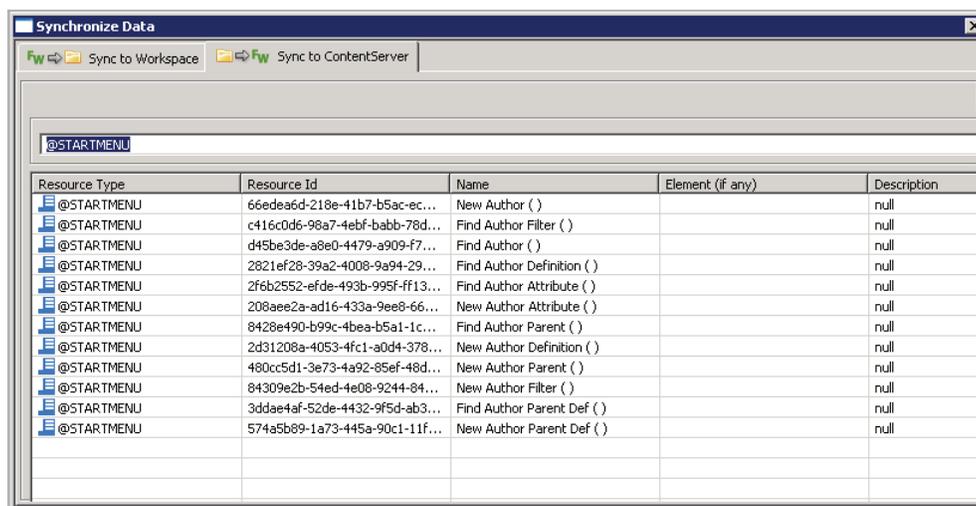
1. Artie は、サイトのスタート・メニュー・アイテムを自身のメイン開発者ツール・ワークスペースと同期させます。



2. Artie は、次にサイトのスタート・メニュー・アイテムを SVN リポジトリにチェックインします。



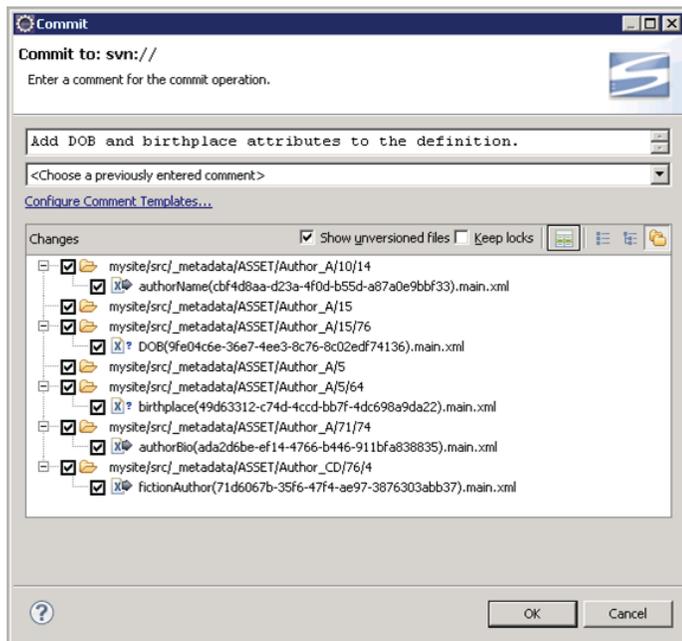
3. Sonoko は砂糖を 2 個入れてアールグレイ・ティーを飲んでいました。席に戻ると、Artie がスタート・メニュー・アイテムを SVN にコミットしたことに気がきます。Sonoko は自身の Eclipse プロジェクトを更新します。SVN リポジトリにアクセスして、スタート・メニュー・アイテムと自身のメイン開発者ツール・ワークスペースを同期させます。次に、これらのスタート・メニュー・アイテムを自身の WebCenter Sites インスタンスにインポートします。





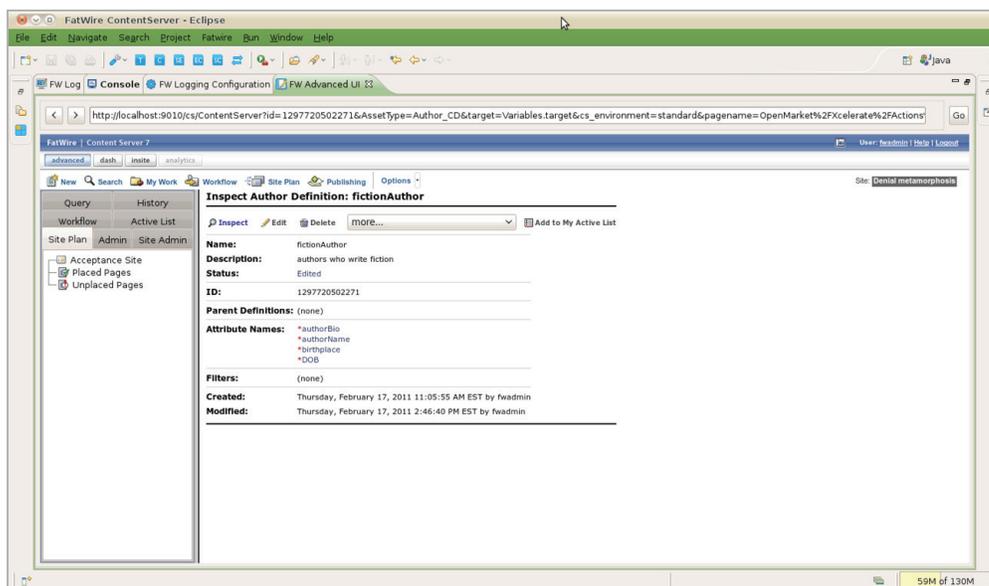
## 午前 11 時 22 分：作成者定義への新しい属性の追加

Matthäus は、マーケティングがリクエストした属性を作成し、それらをフレックス定義（このシナリオでは作成者定義）に追加します。次に、新しい属性とフレックス定義を自身のメイン開発者ツール・ワークスペースにエクスポートして、それらを SVN リポジトリにコミットします。



## 午前 11 時 25 分：サイトの変更の確認

Artie は、変更された作成者定義を SVN から取得して、それを自身の WebCenter Sites インスタンスにインポートします。



件名： RE: 作成者定義の変更の提案について

日時： 2011年2月16日 11:37:31

送信者： artie.companynone.com

宛先： Matthäus.companynone.com

Matthäus、

本件への対応、お疲れ様でした。企業基準により、後に続く各単語の先頭の文字を大文字にする必要があります。birthplace 属性を削除して、かわりに birthPlace を追加します。

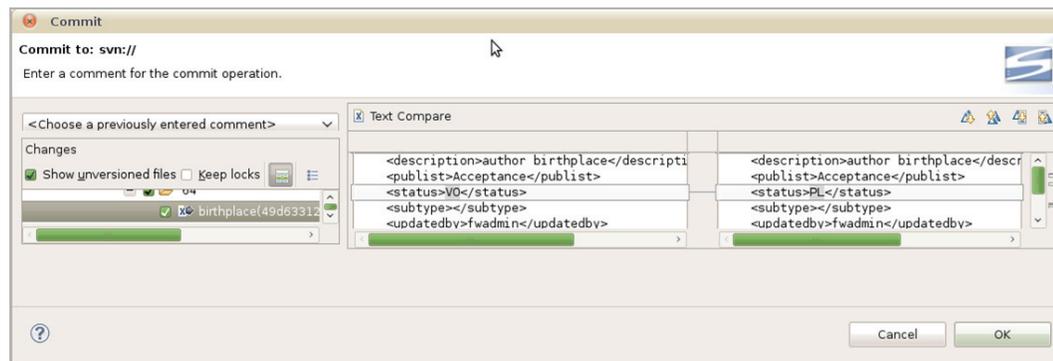
よろしくお願ひします。

Artie

## 午前 11 時 44 分：作成者定義の属性の変更

1. Artie は、birthPlace 属性を作成し、それをフレックス定義に追加します。次に元の birthplace 属性をサイト定義から削除します。
2. Artie は、新しい属性と、作成者定義の変更を SVN リポジトリにコミットします。次に、birthplace 属性のステータスが VO (属性が無効であることを示す) になっていることを確認します。

Sonoko と Matthäus が自身の WebCenter Sites インスタンスを更新すると、それに応じて birthplace 属性は彼らのワークスペースで無効になります。



## 午前 11 時 53 分：チームのワークスペースと WebCenter Sites インスタンスの更新

1. Sonoko と Matthäus は、自身のメイン開発者ツール・ワークスペースと、Artie が SVN リポジトリにチェックインしたリソースを同期させます。
2. 次に、「Synchronize to WebCenter Sites」タブを開いて、自身のワークスペースのリソースを各自の WebCenter Sites インスタンスにインポートします。便宜上、最新の変更が一番上に表示されるように「修正日」列でソートしました。

無効の属性 (Artie が無効にした birthplace 属性など) の「名前」列にはステータス・ヒント (status=VO) が表示されます。

Resource Type	Resource Id	Name	Element (if any)	Description	Modified Date
Author_CD	71d6067b-35f6-47f4-ae97-387...	FictionAuthor (status=ED)		authors who write fiction	2011-02-18 15:17:56.609
Author_A	42af4f58-e90c-4e18-a4b6-47d...	birthPlace (status=PL)		place of birth	2011-02-18 15:17:56.421
Author_A	ada2d6be-ef14-4766-b446-911...	authorBio (status=ED)		author biography	2011-02-18 15:17:56.046
Author_A	9fe04c6e-36e7-4ee3-8c76-8c0...	DOB (status=PL)		date of birth	2011-02-18 15:17:55.937
Author_A	49d63312-c74d-4c0d-bb7f-4dc...	birthplace (status=VO)		author birthplace	2011-02-18 15:17:55.765
Author_A	cbf4d9aa-d23a-4f0d-b55d-a87...	authorName (status=ED)		author name	2011-02-18 15:17:55.484
@STARTMENU	c416c0d6-96a7-4ebf-babb-78d...	Find Author Filter ()		null	2011-02-18 14:22:10.718
@STARTMENU	d45be3de-a8e0-4479-a909-f79...	Find Author ()		null	2011-02-18 14:22:10.718
@STARTMENU	2d31208a-4053-4fc1-a0d4-378...	New Author Definition ()		null	2011-02-18 14:22:10.703
@STARTMENU	480cc5d1-3e73-4a92-85ef-48d...	New Author Parent ()		null	2011-02-18 14:22:10.703
@STARTMENU	574a5b89-1a73-445a-90c1-11f...	New Author Parent Def ()		null	2011-02-18 14:22:10.703
@STARTMENU	66ede6d-218e-41b7-b5ac-ec3...	New Author ()		null	2011-02-18 14:22:10.687
@STARTMENU	3ddae4af-52de-4432-9f5d-ab3...	Find Author Parent Def ()		null	2011-02-18 14:22:10.687
@STARTMENU	2821ef28-39a2-400b-9a94-296...	Find Author Definition ()		null	2011-02-18 14:22:10.671
@STARTMENU	8428e490-b99c-4bea-b5a1-1c1...	Find Author Parent ()		null	2011-02-18 14:22:10.671
@STARTMENU	208aee2a-ad16-433a-9ee8-66...	New Author Attribute ()		null	2011-02-18 14:22:10.656
@STARTMENU	84309e2b-54ed-4e08-9244-84...	New Author Filter ()		null	2011-02-18 14:22:10.656
@STARTMENU	2f6b2552-efde-493b-995f-ff13...	Find Author Attribute ()		null	2011-02-18 14:22:10.609
@ELEMENT CATALOG	OpenMarket/Xcelerate/AssetTy...	-	/ELEMENTS/OpenMarket/Xceler...	null	2011-02-18 13:16:19.843
@ELEMENT CATALOG	OpenMarket/Xcelerate/AssetTy...	-	/ELEMENTS/OpenMarket/Xceler...	null	2011-02-18 13:16:19.843
@ELEMENT CATALOG	OpenMarket/Xcelerate/AssetTy...	-	/ELEMENTS/OpenMarket/Xceler...	null	2011-02-18 13:16:19.843
@ELEMENT CATALOG	OpenMarket/Xcelerate/AssetTy...	-	/ELEMENTS/OpenMarket/Xceler...	null	2011-02-18 13:16:19.828
@ELEMENT CATALOG	OpenMarket/Xcelerate/AssetTy...	-	/ELEMENTS/OpenMarket/Xceler...	null	2011-02-18 13:16:19.828

3. Sonoko と Matthäus は、これらの変更を自身のワークスペースからそれぞれの WebCenter Sites インスタンスにインポートします。これで、彼らのワークスペースと WebCenter Sites インスタンスは最新の状態になります。

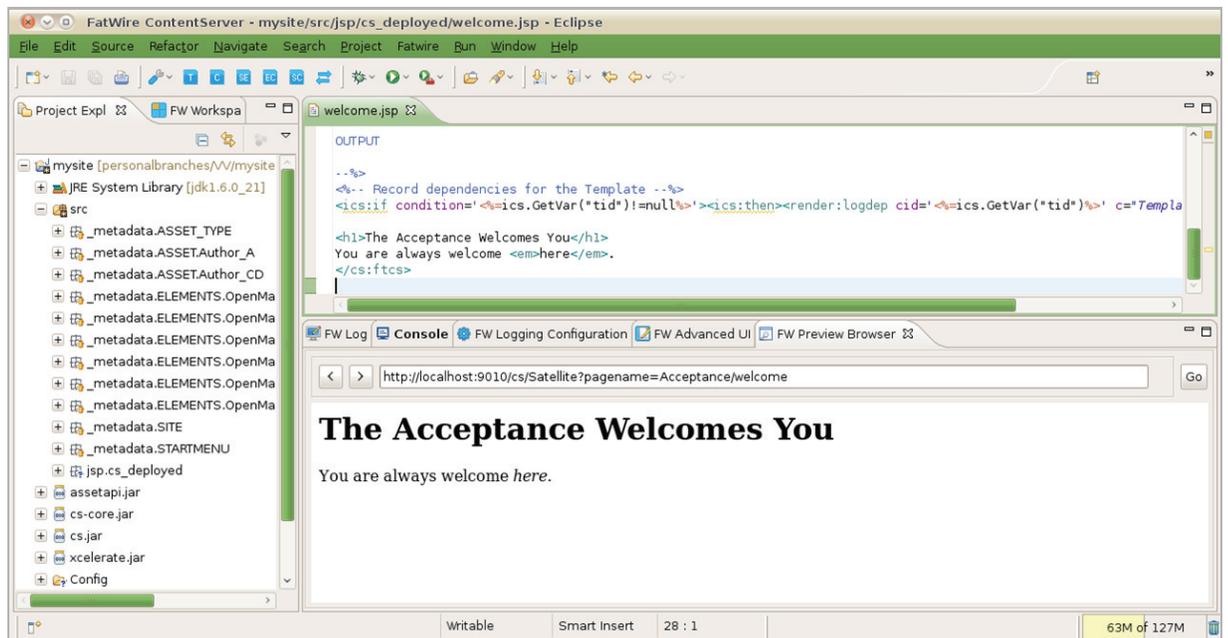
## 午後 12 時 27 分：チームによるサイトの Template アセットの作成

1. (午後 12 時 27 分) Matthäus はサイトの Welcome ページの Template アセットを作成します。

The 'New Template' dialog box is shown with the following configuration:

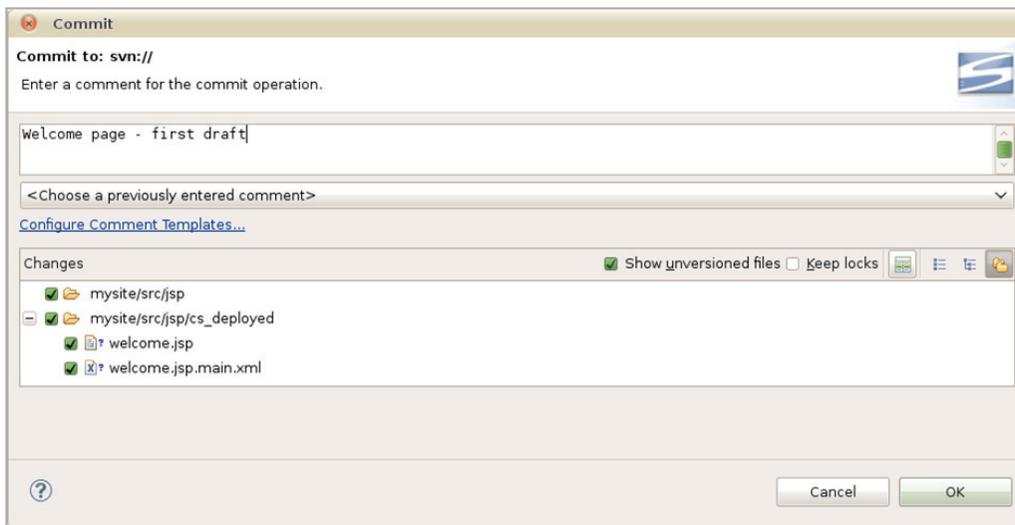
- Site: Acceptance
- Name: welcome
- Description: welcome page
- AssetType: can apply to any asset type
- Subtype: Any
- Usage: Usage unspecified
- Element Type: JSP
- Root Element: /welcome
- Storage path: /welcome.jsp

2. (午後 12 時 34 分) 次に、Template アセットを編集し、「FW Preview Browser」ビューで変更をプレビューします。変更内容を Template アセットの JSP に保存するとすぐに、[Ctrl] を押しながら [r] を押してプレビュー・ブラウザを更新します。

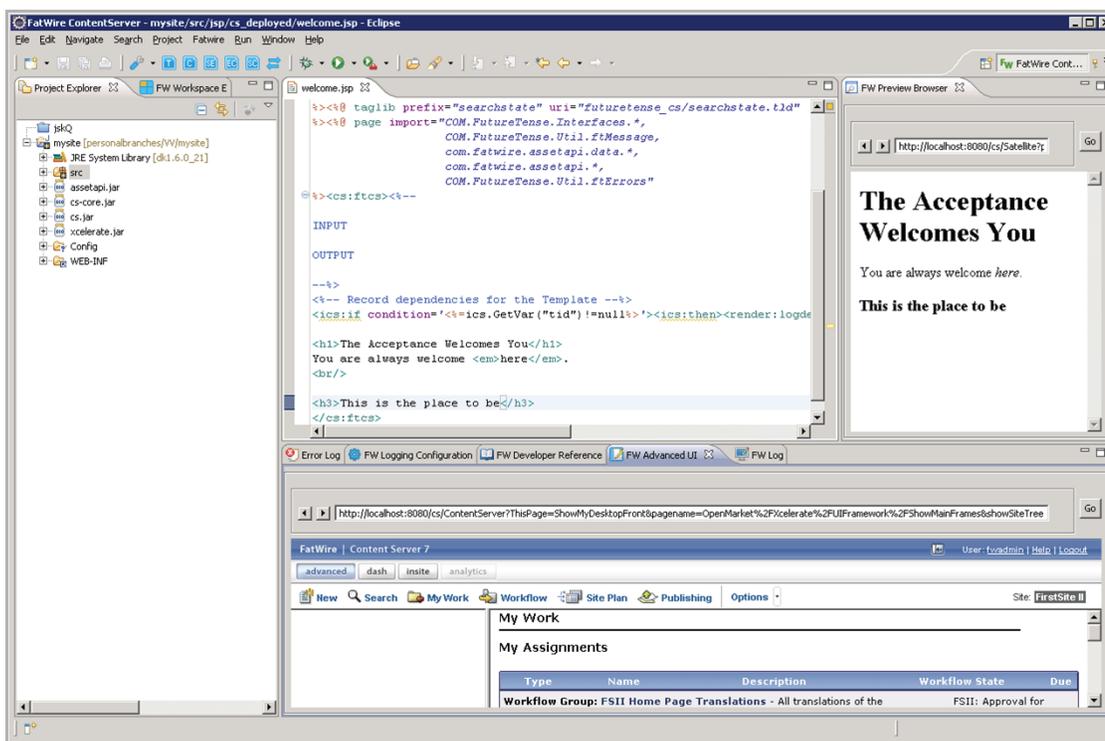


3. (午後 12 時 39 分)Matthäus は、Template の .jsp ファイルと .main.xml ファイルを SVN リポジトリにコミットします。

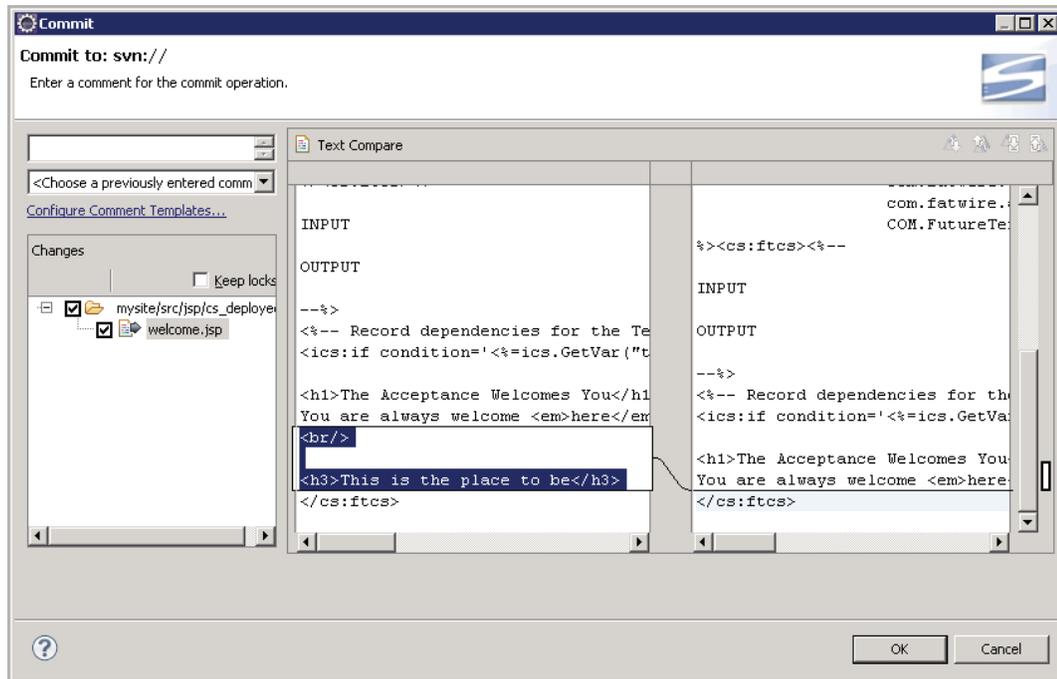
Subclipse は、プロジェクトのすべての変更を検出し、その変更を開発者に知らせます。Template アセットのみが新しいアセットだったため、Matthäus は .main.xml ファイルが Template のメタデータであり、JSP ファイルが Template のコードであることを推測できます。



4. (午後 12 時 44 分) Sonoko は、自身のワークスペースで Template の JSP ファイルを一部修正します。



5. Sonoko は、JSP ファイルの変更を確認してから、その変更を SVN にコミットします。



### 注意

別のチーム・メンバーが Sonoko と同時にこのファイルを修正してチェックインする場合、SVN は、別バージョンのファイルがすでにチェックインされていることを Sonoko に知らせます。Sonoko は、予想外に上書きされないようにその変更と自身の変更内容を統合できます。

## 3日後 ... デプロイメント

Yogesh はコマンドライン・ツールを使用してサイトをデプロイします。この項で使用されるコマンドの詳細は、第7章「コマンドライン・ツール」を参照してください。

### 午前 9 時 32 分 : デプロイメントの準備

Yogesh は、最終的にテスト環境の設定に取り掛かり、コマンドライン・ツールを使用して現在のビルドのデプロイを準備しています。彼は、本番で使用される環境に合う WebCenter Sites システムをハードウェアにインストールしました。

**開発者ツール・インポートを完全自動化ナイトリー・スクリプトに追加する前にテストするには：**

1. Yogesh は、コマンドライン・ツールを使用して、Acceptance サイトとそのリソースを SVN からチェックアウトして、ターゲット WebCenter Sites インスタンスのワークスペースにチェックインします。

コマンド：

```
## go to the workspace location under export/envision/  
cs_workspace in the CS install directory  
  
## create if not there  
  
/home$ mkdir /opt/cs/export/envision/cs_workspace  
/home$ cd /opt/cs/export/envision/cs_workspace  
  
## checkout site from svn
```

出力：

```
A mysite/src  
A mysite/src/_metadata  
A mysite/src/_metadata/ASSET  
A mysite/src/_metadata/ASSET/Author_A  
A mysite/src/_metadata/ASSET/Author_A/10  
A mysite/src/_metadata/ASSET/Author_A/10/14  
A mysite/src/_metadata/ASSET/Author_A/10/14/authorName(cbf4d8aa-  
d23a-4f0d-b55d-a87a0e9bbf33).main.xml  
A mysite/src/_metadata/ASSET/Author_A/11  
A mysite/src/_metadata/ASSET/Author_A/11/79  
A mysite/src/_metadata/ASSET/Author_A/11/79/birthPlace(42afd458-  
e90c-4e18-a4b6-47d322b46414).main.xml  
A mysite/src/_metadata/ASSET/Author_A/5  
A mysite/src/_metadata/ASSET/Author_A/5/64  
A mysite/src/_metadata/ASSET/Author_A/5/64/birthplace(49d63312-  
c74d-4ccd-bb7f-4dc698a9da22).main.xml  
A mysite/src/_metadata/ASSET/Author_A/15  
A mysite/src/_metadata/ASSET/Author_A/15/76  
A mysite/src/_metadata/ASSET/Author_A/15/76/DOB(9fe04c6e-36e7-4ee3-  
8c76-8c02edf74136).main.xml  
A mysite/src/_metadata/ASSET/Author_A/71
```

```

A   mysite/src/_metadata/ASSET/Author_A/71/74
A   mysite/src/_metadata/ASSET/Author_A/71/74/authorBio(ada2d6be-
ef14-4766-b446-911bfa838835).main.xml
A   mysite/src/_metadata/ASSET/Author_CD
A   mysite/src/_metadata/ASSET/Author_CD/76
A   mysite/src/_metadata/ASSET/Author_CD/76/4
A   mysite/src/_metadata/ASSET/Author_CD/76/4/fictionAuthor(71d6067b-
35f6-47f4-ae97-3876303abb37).main.xml
A   mysite/src/_metadata/ASSET_TYPE
A   mysite/src/_metadata/ASSET_TYPE/Author_F(5f9b4964-e9be-4f25-a413-
877e8a5c7469).main.xml
A   mysite/src/_metadata/ASSET_TYPE/Author_P(1552d907-5f38-400b-9460-
36e46d14abc3).main.xml
A   mysite/src/_metadata/ASSET_TYPE/Author_A(162d0b70-7e69-4266-acca-
2f472e3d71bd).main.xml
A   mysite/src/_metadata/ASSET_TYPE/Author_CD(33faf87e-9e8f-4f49-
97cd-424810408938).main.xml
A   mysite/src/_metadata/ASSET_TYPE/Author_PD(7c748df3-d149-4b71-
802a-64b11360e74b).main.xml
A   mysite/src/_metadata/ASSET_TYPE/Author_C(d1497b50-665c-4b0c-80a7-
d25f61566be4).main.xml
A   mysite/src/_metadata/STARTMENU
A   mysite/src/_metadata/STARTMENU/Find+Author+Attribute(2f6b2552-
efde-493b-995f-ff13287f95e0).main.xml
...

```

2. Yogesh はワークスペース・リスト (cmd=listds) を実行して、サイトとそのすべてのリソースが WebCenter Sites インスタンスにインポートされることを確認します。@ALL\_ASSETS および @ALL\_NONASSETS セレクタを使用して、ワークスペース内のアセット・リソースと非アセット・リソースの一覧を生成します。

- @ALL\_ASSETS セレクタを使用するコマンド:

```

/opt/cs/export/envision/cs_workspace$ export
CLASSPATH=csdt-client-1.0.2.jar
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdt.client.main.CSDT
http://localhost:9010/cs/ContentServer username=fwadmin
password=xceladmin resources=@ALL_ASSETS cmd=listds

```

#### 出力:

```

Resource Type ||| Resource Id ||| Name |||
Description ||| Modified On
-----
Author_A ||| cbf4d8aa-d23a-4f0d-b55d-a87a0e9bbf33 ||| authorName (
status=ED ) ||| author name ||| 2011-02-17 15:26:34.000
Author_A ||| 42afd458-e90c-4e18-a4b6-47d322b46414 ||| birthPlace (
status=PL ) ||| place of birth ||| 2011-02-17 15:26:34.000
Author_A ||| 9fe04c6e-36e7-4ee3-8c76-8c02edf74136 ||| DOB (
status=PL ) ||| date of birth ||| 2011-02-17 15:26:34.000
Author_CD ||| 71d6067b-35f6-47f4-ae97-3876303abb37 |||
fictionAuthor ( status=ED ) ||| authors who write fiction |||
2011-02-17 15:26:34.000

```

```
Author_A ||| ada2d6be-ef14-4766-b446-911bfa838835 ||| authorBio (
status=ED ) ||| author biography ||| 2011-02-17 15:26:34.000
Author_A ||| 49d63312-c74d-4ccd-bb7f-4dc698a9da22 ||| birthplace (
status=VO ) ||| author birthplace ||| 2011-02-17 15:12:43.000
Template ||| 89b05c0f-227b-4dcb-961e-2ab6e6af2dae ||| welcome
(Typeless status=PL) ||| welcome page ||| 2011-02-17 23:18:18.000
```

- @ALL\_NONASSETS セレクタを使用するコマンド:

```
/opt/cs/export/envision/cs_workspace$ export
CLASSPATH=csdt-client-1.0.2.jar
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdt.client.main.CSDT
http://localhost:9010/cs/ContentServer username=fwadmin
password=xceladmin resources=@ALL_NONASSETS cmd=listds
```

#### 出力:

```
Resource Type ||| Resource Id ||| Name |||
Description ||| Modified On
-----
-----
@STARTMENU ||| 66edea6d-218e-41b7-b5ac-ec3453bd53b7 ||| New Author
( ) ||| null ||| 2011-02-18 11:02:23.000
@STARTMENU ||| c416c0d6-98a7-4ebf-babb-78d0699698de ||| Find
Author Filter ( ) ||| null ||| 2011-02-18 11:02:23.000
@ASSET_TYPE ||| 162d0b70-7e69-4266-acca-2f472e3d71bd ||| Author_A
( ) ||| Author Attribute ||| 2011-02-18 11:02:23.000
@STARTMENU ||| 2821ef28-39a2-4008-9a94-296fc0fd4f29 ||| Find
Author Definition ( ) ||| null ||| 2011-02-18 11:02:23.000
@STARTMENU ||| d45be3de-a8e0-4479-a909-f79e9320e84f ||| Find
Author ( ) ||| null ||| 2011-02-18 11:02:23.000
@STARTMENU ||| 2f6b2552-efde-493b-995f-ff13287f95e0 ||| Find
Author Attribute ( ) ||| null ||| 2011-02-18 11:02:23.000
@ASSET_TYPE ||| 7c748df3-d149-4b71-802a-64b11360e74b ||| Author_PD
( ) ||| Author Parent Def ||| 2011-02-18 11:02:23.000
@STARTMENU ||| 208aee2a-ad16-433a-9ee8-6658ce8f3abf ||| New Author
Attribute ( ) ||| null ||| 2011-02-18 11:02:23.000
@STARTMENU ||| 8428e490-b99c-4bea-b5a1-1c1768fa9d7d ||| Find
Author Parent ( ) ||| null ||| 2011-02-18 11:02:23.000
@ASSET_TYPE ||| d1497b50-665c-4b0c-80a7-d25f61566be4 ||| Author_C
( ) ||| Author ||| 2011-02-18 11:02:23.000
```

3. Yogesh は、@ASSET\_TYPE:\* セレクタを使用して、必要なすべてのアセット・タイプがインポートされることを確認します。

コマンド:

```
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdt.client.main.CSDT
http://localhost:9010/cs/ContentServer username=fwadmin
password=xceladmin resources=@ASSET_TYPE:* cmd=listds
```

## 出力:

```
Resource Type ||| Resource Id ||| Name ||| Description
||| Modified On
-----
Author_A ||| cbf4d8aa-d23a-4f0d-b55d-a87a0e9bbf33 ||| authorName (
status=ED ) ||| author name ||| 2011-02-17 15:26:34.000
Author_A ||| 42afd458-e90c-4e18-a4b6-47d322b46414 ||| birthPlace (
status=PL ) ||| place of birth ||| 2011-02-17 15:26:34.000
Author_A ||| 9fe04c6e-36e7-4ee3-8c76-8c02edf74136 ||| DOB ( status=PL
) ||| date of birth ||| 2011-02-17 15:26:34.000
Author_CD ||| 71d6067b-35f6-47f4-ae97-3876303abb37 ||| fictionAuthor (
status=ED ) ||| authors who write fiction ||| 2011-02-17 15:26:34.000
Author_A ||| ada2d6be-ef14-4766-b446-911bfa838835 ||| authorBio (
status=ED ) ||| author biography ||| 2011-02-17 15:26:34.000
Author_A ||| 49d63312-c74d-4ccd-bb7f-4dc698a9da22 ||| birthplace (
status=VO ) ||| author birthplace ||| 2011-02-17 15:12:43.000
Template ||| 89b05c0f-227b-4dcb-961e-2ab6e6af2dae ||| welcome
(Typeless status=PL) ||| welcome page ||| 2011-02-17 23:18:18.000
```

4. Yogesh は、サイトの必要なすべてのリソースがビルドにインポートされることを確認します。

## 午前 10 時 04 分: サイトとそのリソースのデプロイ

Yogesh は、コマンドライン・ツールを使用して、インポート・シーケンスを実行します。

1. 最初にサイトをインポートします。

コマンド:

```
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdtd.client.main.CSDT
http://localhost:9999/cs/ContentServer username=fwadmin
password=xceladmin 'resources=@SITE:Acceptance' cmd=import
```

## 出力:

```
*** Importing batch 1297868431526
Importing DSKEY @SITE-Acceptance (batch 1297868431526)
Saved Acceptance (batch 1297868431526)
*** Completed importing batch 1297868431526
```

2. 次にフレックス・ファミリーをインポートします。

コマンド:

```
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdtd.client.main.CSDT
http://localhost:9999/cs/ContentServer username=fwadmin
password=xceladmin 'resources=@ASSET_TYPE:*' cmd=import
```

## 出力:

```
*** Importing batch 1298064678765
Importing DSKEY @ASSET_TYPE-162d0b70-7e69-4266-acca-2f472e3d71bd
(batch 1298064678765)
Importing DSKEY @ELEMENTCATALOG-OpenMarket/Xcelerate/AssetType/
Author_A/LoadTree (batch 1298064678765)
Saved OpenMarket/Xcelerate/AssetType/Author_A/LoadTree (batch
1298064678765)
...
```

## 3. 次にアセットをインポートします。

## コマンド:

```
/opt/cs/export/envision/cs_workspace java
com.fatwire.csdt.client.main.CSDT
http://localhost:9999/cs/ContentServer username=fwadmin
password=xceladmin 'resources=@ALL_ASSETS' cmd=import
```

## 出力:

```
*** Importing batch 1298064679760
Importing DSKEY Author_A-cbf4d8aa-d23a-4f0d-b55d-a87a0e9bbf33 (batch
1298064679760)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1295889071437 (batch 1298064679760)
Importing DSKEY Author_A-42afd458-e90c-4e18-a4b6-47d322b46414 (batch
1298064679760)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1295889071441 (batch 1298064679760)
Importing DSKEY Author_A-9fe04c6e-36e7-4ee3-8c76-8c02edf74136 (batch
1298064679760)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1295889071445 (batch 1298064679760)
Importing DSKEY Author_CD-71d6067b-35f6-47f4-ae97-3876303abb37 (batch
1298064679760)
Importing DSKEY Author_A-ada2d6be-ef14-4766-b446-911bfa838835 (batch
1298064679760)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1295889071449 (batch 1298064679760)
Dependency @ASSET_TYPE-Author_C already exists, skipping.
Dependency @ASSET_TYPE-Author_P already exists, skipping.
Dependency @ASSET_TYPE-Author_CD already exists, skipping.
Dependency @ASSET_TYPE-Author_PD already exists, skipping.
Dependency @ASSET_TYPE-Author_F already exists, skipping.
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_CD:1295889071453 (batch 1298064679760)
Importing DSKEY Author_A-49d63312-c74d-4ccd-bb7f-4dc698a9da22 (batch
1298064679760)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1295889071460 (batch 1298064679760)
Importing DSKEY Template-89b05c0f-227b-4dcb-961e-2ab6e6af2dae (batch
1298064679760)
Saved Template:1295889071461 (batch 1298064679760)
*** Completed importing batch 1298064679760
```

4. これは配布インストールのため、スタート・メニュー・アイテムはオプションです。ただし、Yogesh は WebCenter Sites 管理インタフェースを使用してすべてのリソースが正常にインポートされることを確認するため、スタート・メニュー・アイテムをインポートします。

コマンド:

```
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdt.client.main.CSDT
http://localhost:9999/cs/ContentServer username=fwadmin
password=xceladmin 'resources=@STARTMENU:*' cmd=import
```

出力:

```
*** Importing batch 1298064681075
Importing DSKEY @STARTMENU-66ede6d-218e-41b7-b5ac-ec3453bd53b7 (batch
1298064681075)
Saved 1297720502210 (batch 1298064681075)
Importing DSKEY @STARTMENU-c416c0d6-98a7-4ebf-babb-78d0699698de (batch
1298064681075)
Saved 1297720502230 (batch 1298064681075)
Importing DSKEY @STARTMENU-2821ef28-39a2-4008-9a94-296fc0fd4f29 (batch
1298064681075)
Saved 1297720502222 (batch 1298064681075)
Importing DSKEY @STARTMENU-d45be3de-a8e0-4479-a909-f79e9320e84f (batch
1298064681075)
Saved 1297720502206 (batch 1298064681075)
Importing DSKEY @STARTMENU-2f6b2552-efde-493b-995f-ff13287f95e0 (batch
1298064681075)
Saved 1297720502214 (batch 1298064681075)
Importing DSKEY @STARTMENU-208aee2a-ad16-433a-9ee8-6658ce8f3abf (batch
1298064681075)
Saved 1297720502218 (batch 1298064681075)
Importing DSKEY @STARTMENU-8428e490-b99c-4bea-b5a1-1c1768fa9d7d (batch
1298064681075)
Saved 1297720502238 (batch 1298064681075)
Importing DSKEY @STARTMENU-2d31208a-4053-4fc1-a0d4-3789b23bd897 (batch
1298064681075)
Saved 1297720502226 (batch 1298064681075)
Importing DSKEY @STARTMENU-480cc5d1-3e73-4a92-85ef-48d0e44b81ef (batch
1298064681075)
Saved 1297720502242 (batch 1298064681075)
Importing DSKEY @STARTMENU-84309e2b-54ed-4e08-9244-84d331a60742 (batch
1298064681075)
*** Completed importing batch 1298064681075
```

## 午前 10 時 55 分 : デプロイメントの成功

Yogesh は、インポート・シーケンスが成功したと判断します。次のスクリプトを作成して、このシステムに毎日自動的にインストールするように計画します。

```
## Reinstall ContentServer to start with a clean slate.
## Optionally skip this and just do an update
Reinstall_CS()

## Bring in the latest source from SVN
SVN_Update()

## Prepare for import: compile any Java code such as url
  assemblers and flex filters, etc
## Prepare the database with any custom settings, etc.
preImport()

## Run the CSDT import sequence
CSDT_Import()

## Run the test suite  sanity, performance, acceptance tests
runTestSuite()

## Report results to the team by email so they know about any
  failures first thing in the morning
runReports()
```

スクリプトは、夜中の 0 時 5 分に cron ジョブとして実行されます。



## 付録 B

# コマンドライン・ツールを使用した再利用可能モジュールの作成

開発者ツール・キットでは、モジュール形式でのリソースの再利用と共有が可能です。モジュールは、サイトに固有ではないワークスペースで、この中には **Templates**、フレックス・ファミリーおよび **ElementCatalog** エントリなどのリソースが含まれます。ナチュラル・マッピングを使用してアセットがサイトに追加される標準のエクスポートおよびインポート機能とは異なり、モジュールは、通常、指定したどのサイトにもインポートできるようにサイトのオーバーライドを使用します。

この付録には、次の項が含まれます。

- [再利用可能モジュールの作成](#)

## 再利用可能モジュールの作成

Artie は、他のサイトで再利用するフレックス定義のフレックス・ファミリーを所有しています。また、そのフレックス定義に関連付けられた **Template** アセットもあります。次のシナリオでは、Artie はこれらのリソースを含むモジュールを作成します。このシナリオでは、Artie とそのチームが、付録 A 「開発チームの統合の使用例」で開発したリソースを含むモジュールをコマンドライン・ツールを使用して作成します。

### 注意

Artie がコマンドライン・ツールを使用するには、実行する各コマンドに全体管理者のユーザー名とパスワードを指定する必要があります。このユーザーは、RestAdmin グループのメンバーである必要があります。このシナリオでは、Artie は fwadmin/xceladmin を使用します。

## 手順 I. WebCenter Sites インスタンスでのリソースのリスト表示

Artie は、コマンドライン・ツールを使用して自身の WebCenter Sites インスタンスを参照します。セクタ `resources=@ALL_ASSETS` および `fromSites=Acceptance` を使用して、Acceptance サイトのすべてのアセットをリスト表示します。Artie が使用するコマンドは `listcs` です。これは、彼の WebCenter Sites インスタンス上のすべてのリソースをリスト表示します。

コマンド:

```
/opt/cs/export/envision/cs_workspace$ export CLASSPATH=csdt-
client-1.0.2.jar
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdtd.client.main.CSDT
http://localhost:9010/cs/ContentServer username=fwadmin
password=xceladmin resources=@ALL_ASSETS
fromSites=Acceptance cmd=listcs
```

出力:

```
Resource Type ||| Resource Id ||| Name ||| Description
||| Modified On
-----
Author_CD ||| 1297720502271 ||| fictionAuthor (status=ED) ||| authors who
write fiction ||| 2011-02-17 15:10:41
Author_A ||| 1297720502260 ||| authorName (status=ED) ||| author name |||
2011-02-17 14:46:40
Author_A ||| 1297720502265 ||| authorBio (status=ED) ||| author biography
||| 2011-02-17 14:46:40
Author_A ||| 1297720502289 ||| 1297720502289 (status=VO) ||| author
birthplace ||| 2011-02-17 15:12:35
Author_A ||| 1297720502293 ||| DOB (status=PL) ||| date of birth |||
2011-02-17 14:46:40
```

```
Author_A ||| 1297720502305 ||| birthPlace (status=PL) ||| place of birth
||| 2011-02-17 15:10:22
Template ||| 1297720502331 ||| welcome (Typeless, status=ED) ||| welcome
page ||| 2011-02-17 23:18:18
```

Artie は、5つの Author\_A フレックス属性インスタンス (そのうち1つは無効) と、Author\_CD フレックス定義および Template アセットが1つあることを確認します。

## 手順 II. スタート・メニュー・アイテムのリスト表示

Artie は、コマンドライン・ツールを使用して、Acceptance サイトに割り当てられるスタート・メニュー・アイテムを参照します。

コマンド:

```
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdtd.client.main.CSDT
http://localhost:9010/cs/ContentServer username=fwadmin
password=xceladmin 'resources=@STARTMENU:*'
fromSites=Acceptance cmd=listcs
```

出力:

```
Resource Type ||| Resource Id ||| Name ||| Description
||| Modified On
-----
@STARTMENU ||| 1297720502206 ||| Find Author ||| null ||| -
@STARTMENU ||| 1297720502214 ||| Find Author Attribute ||| null ||| -
@STARTMENU ||| 1297720502222 ||| Find Author Definition ||| null ||| -
@STARTMENU ||| 1297720502230 ||| Find Author Filter ||| null ||| -
@STARTMENU ||| 1297720502238 ||| Find Author Parent ||| null ||| -
@STARTMENU ||| 1297720502246 ||| Find Author Parent Def ||| null ||| -
@STARTMENU ||| 1297720494070 ||| Find CSElement, FirstSiteII ||| Find
CSElement ||| -
@STARTMENU ||| 1297720494086 ||| Find Page, FirstSiteII ||| Find Page |||
-
@STARTMENU ||| 1297720494078 ||| Find SiteEntry, FirstSiteII ||| Find
SiteEntry ||| -
@STARTMENU ||| 1297720494066 ||| Find Template, FirstSiteII ||| Find
Template ||| -
@STARTMENU ||| 1297720502210 ||| New Author ||| null ||| -
@STARTMENU ||| 1297720502218 ||| New Author Attribute ||| null ||| -
@STARTMENU ||| 1297720502226 ||| New Author Definition ||| null ||| -
@STARTMENU ||| 1297720502234 ||| New Author Filter ||| null ||| -
@STARTMENU ||| 1297720502242 ||| New Author Parent ||| null ||| -
@STARTMENU ||| 1297720502250 ||| New Author Parent Def ||| null ||| -
@STARTMENU ||| 1297720501427 ||| New CSElement ||| null ||| -
@STARTMENU ||| 1297720494052 ||| New Page, FirstSiteII ||| New Page ||| -
@STARTMENU ||| 1297720501431 ||| New SiteEntry ||| null ||| -
@STARTMENU ||| 1297720501435 ||| New Template ||| null ||| -
```

### 手順 III. 目的のワークスペースへのすべてのリソースのエクスポート

Artie は、手順 I. と II. に示されるすべてのリソースを使用してモジュールを作成します。次のコマンドを実行して、指定のワークスペースにすべてのリソースを一度にエクスポートします。

```
/opt/cs/export/envision/cs_workspace$ java
com.fatwire.csdtd.client.main.CSDT
http://localhost:9010/cs/ContentServer username=fwadmin
password=xceladmin 'resources=@STARTMENU:*;@ALL_ASSETS'
fromSites=Acceptance cmd=export datastore=authorModule
```

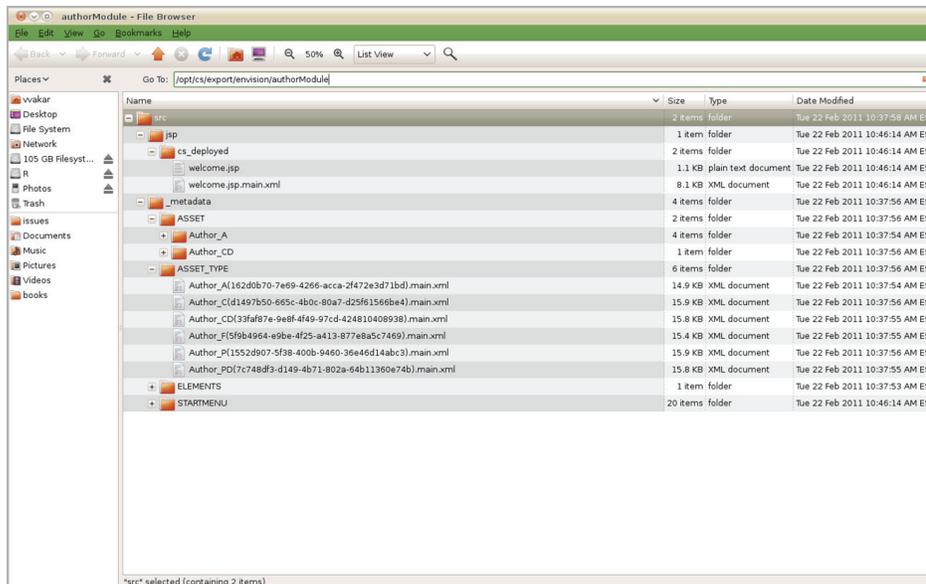
出力:

```
*** Exporting batch 1298385511005
Exporting ASSETDATA Author_CD:1297720502271 (batch 1298385511005)
Exporting ASSETDATA Author_A:1297720502260 (batch 1298385511005)
Exporting ASSET_TYPE Author_A (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
LoadSiteTree (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
AppendSelectDetails (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
AppendSelectDetailsSE (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/IndexAdd
(batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
IndexReplace (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
IndexCreateVerity (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
ContentDetails (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
ContentForm (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
PostUpdate (batch 1298385511005)
Exporting ELEMENTCATALOG OpenMarket/Xcelerate/AssetType/Author_A/
PreUpdate (batch 1298385511005)
...
```

フレックス・ファミリのすべてのアセット・タイプがエクスポートに含まれます。また、これらのタイプに属するすべてのエレメントも含まれます。Artie が作成しているモジュールをその他の WebCenter Sites インスタンスで再利用できるようにするには、この情報 (通常は変更されない) が必要です。

## 手順 IV. モジュールのコンテンツの調査

Artie は、自身のファイル・システムの authorModule ワークスペースを調べます。



Template アセット、フレックス・ファミリー・メンバー、アセット・タイプおよびスタート・メニュー・アイテムがすべて自身のファイル・システムのワークスペースにエクスポートされたことを確認します。

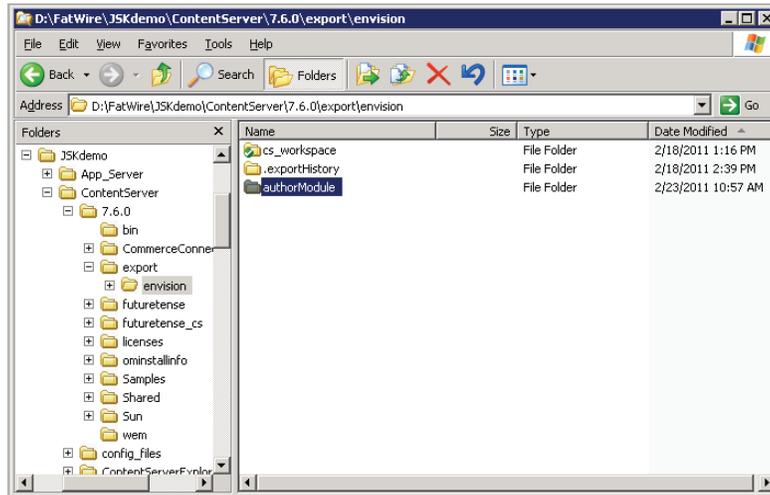
## 手順 V. モジュールのアーカイブ

Artie は authorModule ワークスペースの .zip ファイル・アーカイブを作成して保存します。

## 手順 VI. WebCenter Sites インスタンスへのモジュールのインポート

Artie は、モジュールを FirstSiteII サンプル・サイトにインポートすることになります。

1. ターゲット WebCenter Sites インスタンスのワークスペースの場所にモジュールを解凍します。



2. Artie は、コマンドライン・ツールを使用して、アセット・タイプとスタート・メニュー・アイテムをターゲット WebCenter Sites インスタンスにインポートします。

コマンド:

```
D:\FatWire\JSDKdemo\ContentServer>java com.fatwire.csdt
.client.main.CSDT
http://localhost:8080/cs/ContentServer username=fwadmin
password=xceladmin resources=@ALL_NONASSETS cmd=import
datastore=authorModule toSites=FirstSiteII
```

出力:

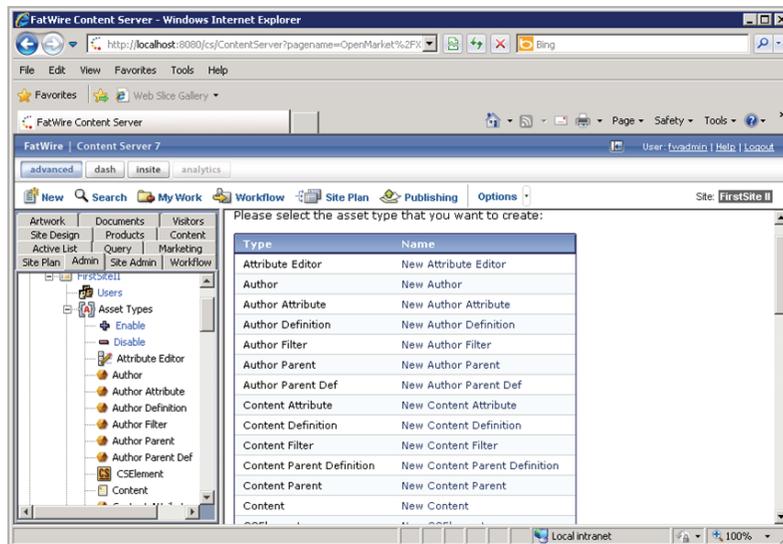
```
*** Importing batch 1298052933085
Importing DSKEY @STARTMENU-4340b65d-a9e4-4131-ac7f-51185a79b18d (batch
1298052933085)
Saved 1297720494070 (batch 1298052933085)
Importing DSKEY @STARTMENU-0a2dec4d-b6be-418c-9992-a4332480bb20 (batch
1298052933085)
Saved 1297720501435 (batch 1298052933085)
Importing DSKEY @STARTMENU-66e4ea6d-218e-41b7-b5ac-ec3453bd53b7 (batch
1298052933085)
Saved 1297720502210 (batch 1298052933085)
Importing DSKEY @STARTMENU-c416c0d6-98a7-4ebf-babb-78d0699698de (batch
1298052933085)
Saved 1297720502230 (batch 1298052933085)
Importing DSKEY @ASSET_TYPE-162d0b70-7e69-4266-acca-2f472e3d71bd
(batch 1298052933085)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
```

```

Importing DSKEY @ELEMENTCATALOG-OpenMarket/Xcelerate/AssetType/
Author_A/LoadSiteTree (batch 1298052933085)
Saved OpenMarket/Xcelerate/AssetType/Author_A/LoadSiteTree (batch
1298052933085)
Importing DSKEY @ELEMENTCATALOG-OpenMarket/Xcelerate/AssetType/
Author_A/AppendSelectDetails (batch 1298052933085)
Saved OpenMarket/Xcelerate/AssetType/Author_A/AppendSelectDetails
(batch 1298052933085)
Importing DSKEY @ELEMENTCATALOG-OpenMarket/Xcelerate/AssetType/
Author_A/AppendSelectDetailsSE (batch 1298052933085)
Saved OpenMarket/Xcelerate/AssetType/Author_A/AppendSelectDetailsSE
(batch 1298052933085)
Importing DSKEY @ELEMENTCATALOG-OpenMarket/Xcelerate/AssetType/
Author_A/IndexAdd (batch 1298052933085)
Saved OpenMarket/Xcelerate/AssetType/Author_A/IndexAdd (batch
1298052933085)
...

```

- Artie は、FirstSiteII サンプル・サイトの WebCenter Sites 管理インターフェースにアクセスし、アセット・タイプとスタート・メニュー・アイテムが正常にインポートされたことを確認します。



- ここで、アセットをインポートします。

コマンド:

```

D:\FatWire\JDKdemo\ContentServer>java com.fatwire.csd
t.client.main.CSDT http://localhost:8080/cs/ContentServer
username=fwadmin password=xceladmin resources=@ALL_ASSETS
cmd=import datastore=authorModule toSites=FirstSiteII

```

出力:

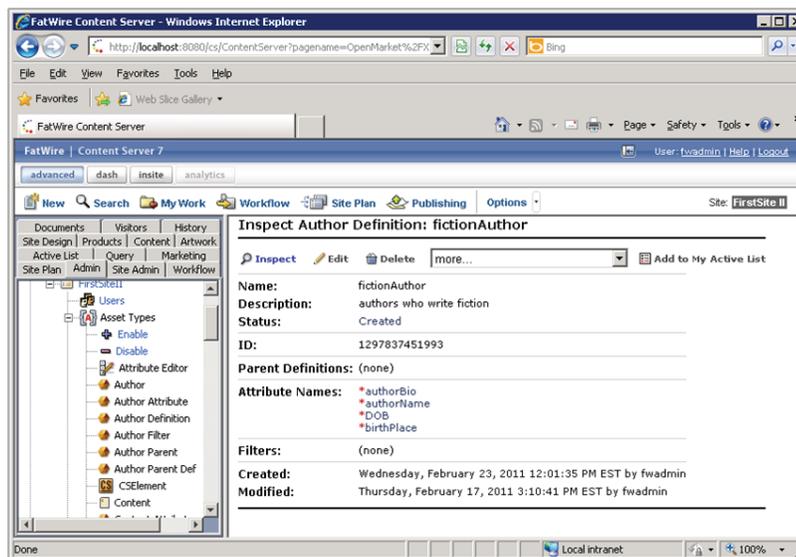
```

*** Importing batch 1298480206533
Importing DSKEY Author_A-cbf4d8aa-d23a-4f0d-b55d-a87a0e9bbf33 (batch
1298480206533)
Dependency @ASSET_TYPE-Author_A already exists, skipping.

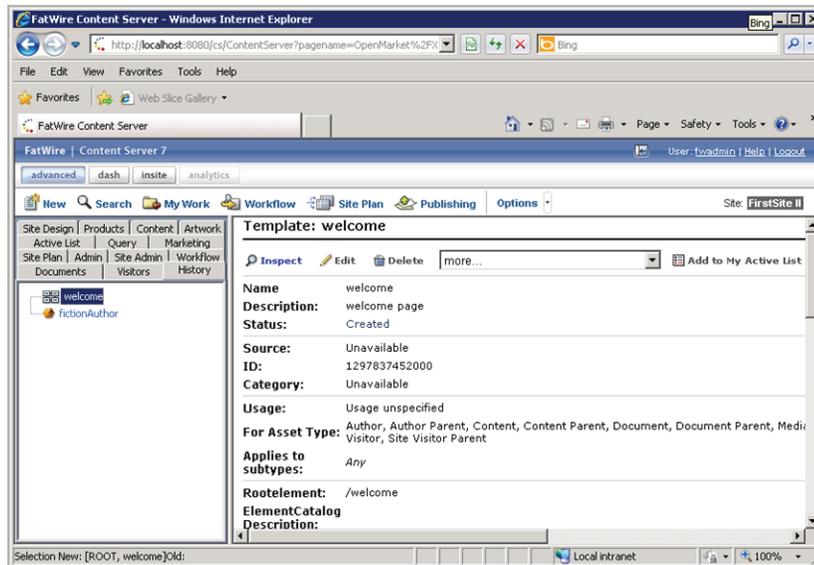
```

```
Saved Author_A:1297837451977 (batch 1298480206533)
Importing DSKEY Author_A-42afd458-e90c-4e18-a4b6-47d322b46414 (batch
1298480206533)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1297837451981 (batch 1298480206533)
Importing DSKEY Author_A-9fe04c6e-36e7-4ee3-8c76-8c02edf74136 (batch
1298480206533)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1297837451985 (batch 1298480206533)
Importing DSKEY Author_CD-71d6067b-35f6-47f4-ae97-3876303abb37 (batch
1298480206533)
Importing DSKEY Author_A-ada2d6be-ef14-4766-b446-911bfa838835 (batch
1298480206533)
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_A:1297837451989 (batch 1298480206533)
Dependency @ASSET_TYPE-Author_C already exists, skipping.
Dependency @ASSET_TYPE-Author_P already exists, skipping.
Dependency @ASSET_TYPE-Author_CD already exists, skipping.
Dependency @ASSET_TYPE-Author_PD already exists, skipping.
Dependency @ASSET_TYPE-Author_F already exists, skipping.
Dependency @ASSET_TYPE-Author_A already exists, skipping.
Saved Author_CD:1297837451993 (batch 1298480206533)
Importing DSKEY Template-89b05c0f-227b-4dcb-961e-2ab6e6af2dae (batch
1298480206533)
Saved Template:1297837452000 (batch 1298480206533)
*** Completed importing batch 1298480206533
```

5. Artie は、フレックス定義が FirstSiteII サンプル・サイトに正常にインポートされていることを確認します。



6. コマンドライン・ツールを使用して、Template アセットもインポートします。次に、WebCenter Sites 管理インターフェースに再度アクセスして、Template アセットが正しくインポートされていることを確認します。



モジュール全体が FirstSiteII サンプル・サイトに正常にインポートされています。このモジュールは再利用でき、目的の WebCenter Sites インスタンスにインポートできます。

