

Adobe® LiveCycle® ES4

アプリケーションサーバークラスターの 設定 (JBoss® 版)

法律上の注意

法律上の注意については、http://help.adobe.com/ja_JP/legalnotices/index.htmlを参照してください。

目次

第1章：このドキュメントの内容

1.1 このドキュメントの対象読者	1
1.2 このガイドで使用する表記	1
1.3 追加情報	2

第2章：インストール、設定およびデプロイメントプロセスの概要

2.1 インストール、設定およびデプロイメントの概要	3
2.2 設定およびデプロイのためのタスクの選択	3
2.3 LiveCycle インストールおよびデプロイメントリスト	4

第3章：JBoss をクラスターで使用する場合の設定

3.1 インストールの準備	5
3.2 JBoss Application Server ソフトウェアのインストール	6
3.3 クラスターでの JBoss の実行	7
3.4 JBoss run.conf ファイルの変更	8
3.5 LiveCycle データベース接続の設定	10
3.6 JBoss Application Server クラスターのテスト	16

第4章：LiveCycle モジュールのインストール

4.1 事前準備	17
4.2 インストールに関する考慮事項	18
4.3 LiveCycle のインストール	19
4.4 クラスター内のキャッシュロケーターの設定 (TCP を使用するキャッシュのみ)	21
4.5 グローバルドキュメントストレージディレクトリ (GDS)	23
4.6 クラスターノードでのフォントディレクトリのインストール	24
4.7 次の手順	24

第5章：LiveCycle をデプロイするための設定

5.1 LiveCycle の設定およびデプロイにおける考慮事項	25
5.2 LiveCycle の事前設定タスク	26
5.3 LiveCycle の構成およびデプロイ	27

第6章：デプロイメント後のタスク

6.1 一般的なタスク	33
6.2 LiveCycle クラスターの検証	37
6.3 CRX クラスターの検証	37
6.4 モジュールの Web アプリケーションへのアクセス	38
6.5 Correspondence Management Solution の設定	40
6.6 PDF Generator の設定	46
6.7 Rights Management の最終設定	55

目次

6.8 LDAP アクセスの設定	55
6.9 FIPS モードの有効化	56
6.10 HTML 電子署名の設定	56
6.11 Connector for EMC Documentum の設定	57
6.12 Connector for IBM Content Manager の設定	61
6.13 Connector for IBM FileNet の設定	65
6.14 JBoss クラスターの隔離	69
6.15 (オプション) JMX コンソールセキュリティを有効にする	69
第7章：ロードバランシングの設定	
7.1 HTTP コネクターベースのロードバランシング	71
7.2 AJP コネクターベースのロードバランシング	73
第8章：高度な実稼働環境の設定	
8.1 Output および Forms のプールサイズの設定	76
8.2 PDF Generator	76
8.3 Windows での CIFS の有効化	77
第9章：付録 - コマンドラインインターフェイスのインストール	
9.1 概要	79
9.2 LiveCycle のインストール	79
9.3 エラーログ	80
9.4 コンソールモードでの LiveCycle のアンインストール	81
第10章：付録 - Configuration Manager コマンドラインインターフェイス	
10.1 操作の順序	82
10.2 コマンドラインインターフェイスのプロパティファイル	82
10.3 一般的な設定プロパティ	83
10.4 使用例	93
10.5 Configuration Manager CLI のログ	93
10.6 次の手順	93
第11章：付録 - Windows サービスとしての JBoss の設定	
11.1 Web Native Connector のダウンロード	94
11.2 Windows サービスのインストール	95
11.3 Windows サービスとしての JBoss Application Server の開始および停止	95
11.4 インストールの確認	95
11.5 追加の設定	96
第12章：付録 - JBoss の手動設定	
12.1 JBoss 用 JDK のインストール	97
12.2 JBoss の手動インストール	98
12.3 JBoss の起動と停止	98
12.4 JBoss 設定の変更	99

12.5 jar ファイルのコピー	103
12.6 手動でインストールした JBoss 用の LiveCycle データベース接続	103
12.7 次の手順	111

第 1 章：このドキュメントの内容

LiveCycle は、ビジネスプロセスの自動化と効率化を支援するエンタープライズサーバープラットフォームです。LiveCycle は次のコンポーネントから構成されます。

- J2EE ベースの Foundation。サーバー機能とランタイム環境を提供します。
- LiveCycle アプリケーションの設計、開発およびテストを行うためのツール。
- LiveCycle サーバーにデプロイされるモジュールとサービス。機能的なサービスを提供します。

LiveCycle のアーキテクチャと機能について詳しくは、「[LiveCycle の概要](#)」を参照してください。

このドキュメントは、[LiveCycle ドキュメントのページ](#)から入手できる大きなドキュメントセットの一部です。新規インストール（シングルサーバーまたはクラスターセットアップ）か、または既存の LiveCycle デプロイメントのアップグレードかに応じて、準備ガイドから始めて、インストールおよび設定ガイドに進むことをお勧めします。自動デプロイメント（評価目的のみ）の場合は、「[LiveCycle の自動インストールおよびデプロイ \(JBoss 版\)](#)」を参照してください。

1.1 このドキュメントの対象読者

このドキュメントは、LiveCycle コンポーネントのインストール、アップグレード、設定、管理またはデプロイを担当する管理者や開発者を対象にしています。このドキュメントで扱う内容は、J2EE アプリケーションサーバー、オペレーティングシステム、データベースサーバーおよび Web 環境に関する十分な知識がある読者を想定しています。

1.2 このガイドで使用する表記

LiveCycle のインストールおよび設定に関するドキュメントでは、共通のファイルパスについて次の命名規則を使用します。

名前	デフォルト値	説明
[LiveCycle root]	Windows : C:\Adobe\Adobe LiveCycle ES4 Linux および Solaris : /opt/adobe/adobe_livecycle_es4	すべての LiveCycle モジュールで使用するインストールディレクトリ。インストールディレクトリには、LiveCycle Configuration Manager 用のサブディレクトリが含まれます。このディレクトリには、サードパーティ製品に関連したディレクトリも含まれます。
[appserver root]	これらはインストール場所の例です。ご使用のマシンでのインストール場所は異なる場合があります。 Windows 上の JBoss Application Server : C:\jboss¥ Linux 上の JBoss Application Server : /opt/jboss/	LiveCycle に含まれるサービスを実行するアプリケーションサーバーのホームディレクトリ。

名前	デフォルト値	説明
[dbserver root]	データベースタイプとインストール時の設定によって異なります。	LiveCycle データベースサーバーがインストールされている場所。
[lc_temp_dir]	Windows の場合 : C:\Adobe\Adobe LiveCycle ES4\tmp Linux、UNIX、AIX の場合 : /opt/adobe/adobe_lifecycle_es4/tmp	LiveCycle サーバーの一時ディレクトリ。
[CRX_home]	Windows の場合 : C:\Adobe\Adobe LiveCycle ES4\crx-repository Linux、UNIX、AIX の場合 : /opt/adobe/adobe_lifecycle_es4/crx-repository	CRX レポジトリをインストールするために使用するディレクトリ。

このガイドに記述されているディレクトリの場所に関するほとんどの情報は、すべてのプラットフォームに当てはまります (Windows 以外のオペレーティングシステムでは、すべてのファイル名とパスにおいて大文字と小文字が区別されます)。プラットフォーム固有の情報は、必要に応じて特記します。

1.3 追加情報

次の表に、LiveCycle の詳細について参照できる情報を示します。

情報	参照先
LiveCycle およびモジュールに関する一般的な情報	LiveCycle の概要
LiveCycle モジュール	LiveCycle モジュール
LiveCycle に統合できる他のサービスや製品	Adobe Developer Connection
Adobe® LiveCycle® Workbench 11 のインストール	Adobe LiveCycle - Workbench 11 のインストール
LiveCycle のインストールの準備	LiveCycle のインストールの準備 (サーバークラスター)
LiveCycle ES4 のアップグレードのチェックリストと計画	LiveCycle ES4 のアップグレードのチェックリストと計画
LiveCycle のトラブルシューティング	LiveCycle のトラブルシューティング
LiveCycle の管理タスクの実行	LiveCycle 管理ヘルプ
LiveCycle に使用できるすべてのマニュアル	LiveCycle ドキュメント
現在のバージョンに関するパッチアップデート、テクニカルノート、および追加情報	アドビエンタープライズサポート

第 2 章：インストール、設定およびデプロイメントプロセスの概要

2.1 インストール、設定およびデプロイメントの概要

LiveCycle のインストール、設定およびデプロイには、次のプロセスが含まれます。

- **インストール**：インストールプログラムを実行して LiveCycle をインストールします。LiveCycle をインストールすると、必要なすべてのファイルが、使用するコンピューター上の 1 つのインストールディレクトリ構造内に配置されます。デフォルトのインストールディレクトリは C:\¥Adobe¥Adobe LiveCycle ES4 (Windows) または /opt/adobe/adobe_lifecycle_es4 (Windows 以外) ですが、これ以外のディレクトリにファイルをインストールすることもできます。
- **設定**：LiveCycle の設定では、LiveCycle の動作方法を決定する様々な設定を変更します。製品のアセンブリでは、設定の指示に従って、すべてのインストール済みコンポーネントがデプロイ可能な EAR および JAR ファイルに配置されます。コンポーネントに対してデプロイメントのための設定とアセンブリを行うには、Configuration Manager を実行します。複数の LiveCycle モジュールに対して同時に設定およびアセンブリを行うこともできます。
- **デプロイ**：製品のデプロイでは、アセンブリされた複数の EAR ファイルといくつかの補助ファイルを、LiveCycle を実行する予定のアプリケーションサーバーにデプロイします。複数のモジュールを設定した場合は、デプロイ可能なコンポーネントがデプロイ可能な EAR ファイル内にパッケージ化されます。また、コンポーネントおよび LiveCycle アーカイブファイルは、JAR ファイルとしてパッケージ化されています。

注意：LiveCycle アーカイブファイルは、ファイル拡張子 .lca を使用します。

- **LiveCycle データベースの初期化**：LiveCycle で使用されるデータベースを初期化すると、User Management および他のコンポーネントで使用するためのテーブルが作成されます。LiveCycle データベースに接続するモジュールをデプロイする場合は、デプロイメントプロセスの完了後に LiveCycle データベースを初期化する必要があります。

LiveCycle のインストールと設定を開始する前に、該当する準備ガイドで説明されているように環境の準備が整っていることを確認します。

2.2 設定およびデプロイのためのタスクの選択

LiveCycle のインストール後は、Configuration Manager を実行して次のタスクを実行できます。

- アプリケーションサーバーまたはアプリケーションサーバーのクラスターにデプロイするために EAR ファイルの LiveCycle モジュールを設定
- LiveCycle データベースの初期化
- LiveCycle コンポーネントのデプロイ
- LiveCycle コンポーネントのデプロイメントの検証
- LiveCycle コンポーネントの設定

2.3 LiveCycle インストールおよびデプロイメントリスト

次のリストに、手動オプションを使用して LiveCycle をインストールする場合に必要な手順を示します。インストールを実行する前に、アプリケーションサーバーまたはクラスターをインストールおよび設定しておく必要があります。

- 必要なソフトウェアが各マシンにインストールされ、ターゲット環境で設定されていることを確認します。
- インストール先の環境でアプリケーションサーバークラスターが作成および設定されていることを確認します。JBoss を手動で設定するか、アドビにより事前設定された JBoss を使用するかを選択します。
- 1 台のマシンでのみインストールプログラムを実行します。
- Configuration Manager を実行し、「LiveCycle EAR を設定」タスクを選択します。このタスクでは、LiveCycle の設定とアセンブリを行います。

LiveCycle がインストールされていないマシンで、手動設定手順を実行します。例えば、コンテンツレポジトリをコピーします。

- EAR ファイルをアプリケーションサーバーまたはクラスターにデプロイします。これは手動で実行する必要があります。
- Configuration Manager を実行して、LiveCycle データベースを初期設定し、LiveCycle コンポーネントファイルをデプロイします。
- Administration Console および User Management にアクセスします。
- (オプション) LDAP アクセスを設定します。

第3章：JBoss をクラスターで使用する場合の設定

JBoss Application Server の設定は、複数のディレクトリにある多数の設定ファイルで定義されます。クラスターで使用するように JBoss を設定するには、多数の設定ファイルを変更する必要があります。変更には任意のテキストエディターを使用できます。

JBoss クラスター環境を設定するには、次のタスクを実行する必要があります。

- クラスター内のすべてのコンピューターが正しく準備されていることを確認します (5 ページの「[3.1 インストールの準備](#)」を参照)。
- JBoss Application Server ソフトウェアをインストールします (6 ページの「[3.2 JBoss Application Server ソフトウェアのインストール](#)」を参照)。
- JBoss 実行ファイルを変更します (8 ページの「[3.4 JBoss run.conf ファイルの変更](#)」を参照)。
- LiveCycle データベース接続を設定します (10 ページの「[3.5 LiveCycle データベース接続の設定](#)」を参照)。
- JBoss クラスター設定をテストします (16 ページの「[3.6 JBoss Application Server クラスターのテスト](#)」を参照)。

3.1 インストールの準備

JBoss Application Server をクラスター内のコンピューターにインストールする前に、システムが以下の設定要件を満たしていることを確認してください。

ディスク容量：アプリケーションサーバーをインストールするパーティションに 10 GB 以上の空きディスク容量があることを確認します。製品のインストールに必要な容量に加えて、環境変数 TEMP または TMP が、最低 500 MB の空き容量がある有効な一時ディレクトリを指している必要があります。ダウンロード可能な実行ファイルに約 500 MB、イメージの展開用にさらに 1.0 GB 必要です。

IP アドレス設定：すべてのコンピューターに、単一の DNS で管理される固定 IP アドレスが必要です。

IP マルチキャスト：すべてのコンピューターが、IP マルチキャストパケットの転送を完全にサポートしている必要があります。つまり、すべてのルーターやその他のトンネル用テクノロジーが、クラスターサーバーインスタンスにマルチキャストメッセージを転送するように設定されている必要があります。ネットワーク遅延は、最低でも、ほとんどのマルチキャストメッセージの最終到達先に 200 ~ 300 ミリ秒以内に到達するレベルにあることが必要です。また、クラスターのマルチキャスト time-to-live (TTL) 値は、マルチキャストパケットが最終到達先に届く前にルーターによって破棄されることのないよう、十分に大きい値である必要があります。

バージョン：クラスター内のすべてのコンピューターは、JBoss Application Server ソフトウェアの同一バージョンおよび同一サービスパックを使用している必要があります。

水平クラスター：水平クラスター構成では (つまり JBoss Application Server のインスタンスが別々のコンピューターにインストールされている場合は)、すべてのコンピューターが同一のネットワークサブネット上にあり、コンピューターの時間が同期されている必要があります (『[LiveCycle のインストールの準備 \(サーバークラスター\)](#)』を参照)。

アカウントの権限：(Windows 上の PDF Generator のみ) 管理者権限を持つユーザーアカウントで JBoss Application Server をインストールおよび実行する必要があります。

共有ネットワークドライブ：クラスター内のすべてのコンピューターに読み取りと書き込み権限がある、安全な共有ネットワークドライブを作成しておく必要があります (『[LiveCycle のインストールの準備 \(サーバークラスター\)](#)』を参照)。

J2SE SDK バージョン: J2SE SDK バージョンの詳細については、『[サポートされているプラットフォームの組み合わせ](#)』ドキュメントを参照してください。

クラスターのすべてのシステムのクロックが共通のタイムサーバーと同期されている可能性があります。Windows ドメインでは、クロックの同期は自動的に行われます。Windows 以外のシステムでは、Network Time Protocol を設定する必要があります。

3.2 JBoss Application Server ソフトウェアのインストール

JBoss Application Server は、クラスター内のコンピューターごとにインストールおよび設定します。垂直クラスターを実装するコンピューターには、複数のインスタンスをインストールできます。『LiveCycle のインストールの準備 (サーバークラスター)』ドキュメントに、LiveCycle ES4 用にサポートされている JBoss Application Server のバージョンが記載されています。

LiveCycle インストールメディアの `third_party` ディレクトリ内にある、アドビの事前設定 JBoss Application Server をインストールします。`third_party\jboss.zip` ファイルを展開すると、`[appserver root]/server/` ディレクトリに次のサブディレクトリが作成されます。

- (シングルサーバー) `lc_oracle`
- (シングルサーバー) `lc_sqlserver`
- (シングルサーバー) `lc_mysql`
- (クラスター) `lc_sqlserver_cl`
- (クラスター) `lc_oracle_cl`
- (クラスター) `lc_mysql_cl`

実際の設定に関係のないディレクトリは削除しても問題ありません。例えば、クラスター設定でアドビの事前設定 JBoss 版 Oracle を使用する場合、`lc_oracle_cl` を残し、他のディレクトリを削除します。

注意: `[appserver root]/server/all` ディレクトリは、JBoss を手動で設定した場合にのみ関係します。アドビの事前設定 JBoss では、`/all` ディレクトリではなく、該当するデータベースに特有の上記のディレクトリを使用できます。

重要: 上記のアドビの事前設定 JBoss Application Server のみをインストールしてから、このドキュメントの以降のセクションを参照して、ノードをクラスター用に設定します。『[LiveCycle のインストールの準備 \(シングルサーバー\)](#)』で説明されている JBoss 設定手順は使用しないでください。この手順はスタンドアロン設定用であり、クラスター設定には適していません。

3.2.1 水平クラスター用の JBoss Application Server のインストール

アドビの事前設定 JBoss Application Server をインストールするには、クラスターの各コンピューター上で JBoss Application Server をインストールする予定の場所に、`JBoss.zip` ディレクトリの内容を展開します。このインストールは、水平クラスター用に完全に設定されています。

3.2.2 JBoss Application Server 用 Windows サービスの設定

クラスターの JBoss Application Server が Windows オペレーティングシステムで実行される場合は、必要に応じて Windows サービスをインストールして管理できます。Windows サービスには、クラスター構成のアプリケーションサーバーを容易に起動および停止するための GUI が用意されています。

アプリケーションサーバーを管理するための Windows サービスを作成するには、JBoss Application Server がインストールされている必要があります。Windows サービスは、管理するクラスターの JBoss Application Server ごとに個別に作成します。JBoss Web Native Connector を使用して JBoss を Windows サービスとして設定する方法については、「94 ページの「[付録 - Windows サービスとしての JBoss の設定](#)」を参照してください。

JBoss Application Server を Windows サービスとして起動するには：

- ❖ クラスターの JBoss Application Server で、**スタート/コントロールパネル/管理ツール/サービス**を選択し、JBoss Application Server の Windows サービスを選択して、「**開始**」をクリックします。

注意：JBoss Application Server を Windows サービスとして開始すると、コンソールの出力結果は `run.log` ファイルにリダイレクトされます。このファイルを調べると、サービスの開始時に発生したエラーを検出できます。

JBoss Application Server を Windows サービスとして停止するには：

- ❖ クラスターの JBoss Application Server で、**スタート/コントロールパネル/管理ツール/サービス**を選択し、JBoss Application Server の Windows サービスを選択して、「**停止**」をクリックします。

注意：JBoss Application Server を Windows サービスとして停止すると、コンソールの出力結果は `run.log` ファイルにリダイレクトされます。このファイルを調べると、サービスの停止時に発生したエラーを検出できます。

3.3 クラスターでの JBoss の実行

JBoss の起動

次のコマンドを入力して、JBoss Application サーバーを起動します。

```
./run.sh -b <ipaddress> -c <server_profile>
```

注意：これにより、JBoss インスタンスがデフォルトのパーティション名、マルチキャストアドレス、およびマルチキャストポートを使用して起動します。

クラスターパーティション名の変更

```
./run.sh -g <partition_name> -b <ipaddress> -c <server_profile>
```

注意：このコマンドは、デフォルトのマルチキャストポートおよびアドレスを使用します。クラスターのすべてのノードは `<partition name>` である必要があるため、この値が JBoss クラスターで一意であることを確認してください。

また、「8 ページの「[3.4 JBoss run.conf ファイルの変更](#)」」セクションで説明しているように JVM 引数としてパーティション名に言及することもできます。

マルチキャストアドレスの変更

一部のユーザーは特定のセットのマルチキャストアドレスを持っています。デフォルトのマルチキャストアドレスをお好きなアドレスに変更するには、次のコマンドを入力します。

```
/run.sh -u <UDP group Ip address> -g <partition_name> -b <ipaddress> -c <server_profile>
```

注意：上記のコマンドで、`<UDP group Ip address>` は UDP マルチキャストアドレスを示しています。

マルチキャストポートの変更

マルチキャストポートを変更するには、「8 ページの「[3.4 JBoss run.conf ファイルの変更](#)」」で説明している `run.conf` ファイルで次の `jvm` 引数を入力します。

```
-Djboss.jgroups.udp.mcast_port=<port_number>
```

注意: デフォルトでは、JBoss クラスターは UDP ベースのクラスターを使用するように設定されます。TCP ベースのクラスターを使用するようにクラスターを設定できます。TCP ベースのクラスターを使用するためのクラスターの設定方法について詳しくは、「JBoss クラスター」を参照してください。

3.4 JBoss run.conf ファイルの変更


LiveCycle オプションを追加するには、LiveCycle クラスター内の各 JBoss Application Server インスタンスの JBoss 実行ファイルを変更します。

この手順を開始する前に、クラスターキャッシュの引数を正しく設定できるように、LiveCycle クラスターでのクラスターキャッシュの実装方法を判断してください。クラスターキャッシュは、TCP または UDP のいずれかを使用して実装できますが、両方を使用することはできません。選択に影響する要因は次のとおりです。

- (推奨) クラスターが IPv4 ベースまたは IPv6 ベースの場合は、TCP を使用します。IPv6 ベースのクラスターには、IPv6 に準拠するように TCP を使用する必要があります。

TCP を使用してクラスターキャッシュを実装する場合は、TCP ロケーターも正しく設定するようにしてください (「キャッシュロケーターの設定 (TCP を使用するキャッシュのみ)」を参照)。

- UDP を使用するオプションは、IPv4 ベースのクラスターでのみ利用できます。

 TCP プロトコルには継承の信頼性があるので、実稼働システムには UDP マルチキャストではなく TCP を使用することをお勧めします。

JBoss run.conf ファイルを変更するには:

- 1 テキストエディターで次のファイルを開きます。

- (Windows) [appserver root]¥bin¥run.conf.bat
- (UNIX) [appserver root]/bin/run.conf

- 2 JAVA_OPTS 行で、次の引数を追加または変更します。

```
-Djboss.partition.name=<partition_name>
```

注意: <partition_name> には、LiveCycle クラスターを一意に識別する値を指定します。次の例に示すように、LiveCycle クラスターの各ノードで、同じ <partition_name> の値を設定します。

```
-Djboss.partition.name=livecycle_cluster
```

注意: JBoss サーバーの起動中にこの値をコマンドとして渡すこともできます。JBoss クラスター内のすべてのノードで同じ値である必要があります。値は run.conf ファイル内の JVM 引数、またはコマンドライン引数 '-g <partition_name>' として渡す必要があります。

- 3 JAVA_OPTS 行で、次の引数を追加または変更します。

```
-Dadobeidp.serverName=<server name>
```

注意: <server name> には任意の値を指定できますが、次の例に示すように、<server name> には **LiveCycle** クラスターの各ノードで一意の値を指定する必要があります。

- クラスターの 1 つのノードで、引数 -Dadobeidp.serverName=server1 を設定します。
- クラスターの別のノードで、引数 -Dadobeidp.serverName=server2 を設定します。

LiveCycle クラスターのその他のノードも同様に設定できますが、<server name> には一意の値を設定します。

- 4 IPv4 の場合、JAVA_OPTS 行で次の引数が既に設定されている場合があります。設定されていない場合は、この引数を設定します。

```
-Djava.net.preferIPv4Stack=true
```

IPv6 の場合、`-Djava.net.preferIPv4Stack=true` を削除し、次の引数を追加します。

```
-Djava.net.preferIPv6Addresses=true  
-Djava.net.preferIPv6Stack=true
```

- 5 クラスターキャッシュ用に JVM 引数を構成します。JAVA_OPTS 行で、次のいずれか 1 つの引数を追加または変更します。

UDP 検索を使用したキャッシュ

- マルチキャストポート引数を次の形式で設定します。

```
-Dadobe.cache.multicast-port=<port number>
```

注意：<port number> の値は、1025 ~ 65535 の使用可能なポートにすることができます。マルチキャストポートは LiveCycle クラスターに対して一意である必要があります (同じネットワーク上の別のクラスターが、そのポートを使用しているはいけません。同じネットワーク上の別のクラスターで同じポートを使用すると、ブートストラップに失敗します)。次の例のように、LiveCycle クラスターのすべてのノードに同じ <port number> を設定することをお勧めします。

```
-Dadobe.cache.multicast-port=33456
```

- マルチキャストアドレス引数の設定はオプションです。IPv4 および IPv6 用のデフォルトのマルチキャストアドレスは、次のとおりです。

```
IPv6 - FF38::1234  
IPv4 - 239.192.81.1
```

ネットワークのマルチキャストアドレスに制限を設けている場合は、次の引数を使用してマルチキャストアドレスを設定します。

```
-Dadobe.cache.multicast-address=<ip address>
```

<ip address> の値は、マルチキャストネットワークに使用する IP アドレスです。adobe.cache.multicast-port の値が 0 の場合、この IP アドレスは無視されます。

マルチキャストアドレスは LiveCycle クラスターに対して一意である必要があります、同じネットワーク上の別のクラスターがそのアドレスを使用しているはいけません。LiveCycle クラスターのすべてのノードに同じ <ip address> を設定することをお勧めします。次に例を示します。

```
-Dadobe.cache.multicast-address=239.192.81.1
```

- 複数のネットワークインターフェイスを持つマシンの場合

複数のネットワークインターフェイスカード (NIC) を介して複数のネットワークに接続するマシンもあります。そのようなマシンでは、JVM プロパティ `-Dadobe.cache.bind-address` を、LiveCycle Server に使用するネットワークインターフェイスカードの IP アドレスに設定します。

```
-Dadobe.cache.bind-address=<IP Address>
```

注意：1 つのネットワークインターフェイスカードを持つマシンに対しても、JVM プロパティ `-Dadobe.cache.bind-address` を設定することをお勧めします。

TCP のみを使用したキャッシュ

- IPv4 の場合、クラスターロケーター引数を次の形式で設定します。

```
-Dadobe.cache.cluster-locators=<IPaddress>[<port number>],<IPaddress> [ <port number>]
```

IPv6 の場合、クラスターロケーター引数を次の形式で設定します。

```
-Dadobe.cache.cluster-locators=<hostname>@<IPv6 address>[<port number>],<hostname>@<IPv6 address>[<port number>]
```

注意：クラスターのすべてのノードのロケーターを、カンマ区切りのリストで設定します。<IPaddress> の値には、ロケーター稼働しているコンピューターの IP アドレスを指定します。<port number> の値には、1025 ~ 65535 の未使用のポートを指定します。次の例のように、LiveCycle クラスターのすべてのノードに同じ <port number> を設定することをお勧めします。

```
-Dadobe.cache.cluster-locators=10.20.30.5[22345],10.20.30.6[22345]
```

注意：LiveCycle Cluster のすべてのノードで TCP ロケーターを実行しないでください。ロケーターは、最低 2 つ実行することをお勧めします。1 つの TCP ロケーターがプライマリロケーターとなり、他の TCP ロケーターがフェイルオーバーの問題を処理するセカンダリロケーターになります。3 つ以上の TCP ロケーターをバックアップロケーターとして追加できますが、必須ではありません。TCP ロケーターの設定の詳細については、「21 ページの「[4.4 クラスター内のキャッシュロケーターの設定 \(TCP を使用するキャッシュのみ\)](#)」」を参照してください。

- 6 アプリケーションサーバーへのサービス拒否攻撃を防ぐには、次の JVM 引数を設定します。

```
-DentityExpansionLimit=10000
```

- 7 編集したファイルを保存します。
- 8 クラスター内のノードごとに、手順 1 ~ 6 を繰り返します。

3.5 LiveCycle データベース接続の設定

次のタスクを実行し、クラスター内の各 JBoss Application Server から LiveCycle データベースへのデータベース接続を有効にする必要があります。

- クラスター内の JBoss Application Server のインスタンスごとに、正しい JDBC ドライバーが存在することを確認します。
- データソースファイルを作成し、クラスター内の JBoss Application Server の各インスタンスにデプロイします。adobe-ds.xml ファイルで、LiveCycle で使用するデータソースを設定します。データベースが存在するコンピューターのホスト名、データベース名、ポート番号、データベースのユーザー名とパスワードなどのパラメーターを設定します。

このタスクは、次の手順で簡単に行うことができます。

- 1 LiveCycle インストールメディアから任意のコンピューターに、必要なファイルをコピーします。
- 2 次のサブセクションで説明するようにファイルを編集します。
- 3 編集したファイルをクラスターの各ノードに保存します。

手順については、次のうちご使用のデータベースに関連するセクションを参照してください。

- 10 ページの「[3.5.1 アドビの事前設定 JBoss 用の Oracle の設定](#)」
- 12 ページの「[3.5.2 アドビの事前設定 JBoss 用の SQL Server の設定](#)」
- 14 ページの「[3.5.3 アドビの事前設定 JBoss での MySQL の設定](#)」

3.5.1 アドビの事前設定 JBoss 用の Oracle の設定

LiveCycle データを格納する Oracle データベースに JBoss から接続できるようにするには、アドビの事前設定 JBoss に付属している次のファイルが必要です。

- [appserver_root]/server/lc_oracle_cl/lib の Oracle JDBC ドライバーファイル
- [appserver_root]/server/lc_oracle_cl/deploy のアドビデータソースファイル
- [appserver_root]/server/lc_oracle_cl/deploy の Oracle データソースファイル

注意: <http://community.jboss.org/wiki/EncryptingDataSourcePasswords> で説明されているいずれかの方法を使用して、データソースファイル内のパスワードを暗号化します。 http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clear.html の手順を使用することもできます。

3.5.1.1 データソースファイルの設定

Oracle データソースを設定する前に、Oracle にデータベースを作成しておく必要があります (『LiveCycle のインストールの準備 (サーバークラスター)』を参照)。

アドビデータソースファイルを変更するには:

- 1 [appserver root]/server/lc_oracle_cl/deploy/adobe-ds.xml ファイルをテキストエディターで開き、次の行を検索します。

```
<connection-url>jdbc:oracle:thin:@localhost:1521:adobe</connection-url>  
<user-name>adobe</user-name>  
<password>adobe</password>
```

- 2 次のテキストをデータベースに固有の値に置き換えます。

- **localhost**: データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは localhost です。
- **1521**: データベースへのアクセスに使用するポート。デフォルトのポートは 1521 です。
- **adobe**: LiveCycle データを格納しているデータベースインスタンスのシステム ID (SID)。デフォルト値の adobe を、お使いのデータベースのシステム ID に変更する必要があります。

- 3 <user-name> タグと <password> タグに、アプリケーションサーバーがデータベースにアクセスするときに使用するユーザー名とパスワードを指定します。これらのデフォルト値 (どちらも adobe) を、お使いのデータベースの資格情報に変更する必要があります。

- 4 IDP_DS および EDC_DS の残りの要素について、手順 1 ~ 3 を繰り返します。

- 5 ファイルを保存します。

Oracle データソースファイルを変更するには:

- 1 [appserver root]/server/lc_oracle_cl/deploy/oracle-ds.xml ファイルをテキストエディターで開き、次の行を検索します。

```
<connection-url>jdbc:oracle:thin:@localhost:1521:adobe</connection-url>  
<user-name>adobe</user-name>  
<password>adobe</password>
```

- 2 次のテキストをデータベースに固有の値に置き換えます。

- **localhost**: データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは localhost です。
- **1521**: データベースへのアクセスに使用するポート。デフォルトのポートは 1521 です。
- **adobe**: LiveCycle データを格納しているデータベースインスタンスのシステム ID (SID)。デフォルト値の adobe を、お使いのデータベースのシステム ID に変更する必要があります。

- 3 <user-name> タグと <password> タグに、アプリケーションサーバーがデータベースにアクセスするときに使用するユーザー名とパスワードを指定します。これらのデフォルト値 (どちらも adobe) を、お使いのデータベースの資格情報に変更する必要があります。

- 4 ファイルを保存します。

3.5.1.2 login-config.xml ファイルの編集

- 1 [appserver root]/server/lc_oracle_cl/conf/login-config.xml ファイルをテキストエディターで開き、<policy> 要素内の次のコードを変更します。

```
<application-policy name="OracleDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.ConfiguredIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=Default DS </module-option>
      </login-module>
    </authentication>
  </application-policy>
```

- 2 太字のテキスト (principal、userName および password パラメーターの値) をデータベースに固有の値に置き換えて、アプリケーションサーバーがデータベースにアクセスできるようにします。
- 3 ファイルを保存して閉じます。
- 4 JBoss を再起動します。

3.5.2 アドビの事前設定 JBoss 用の SQL Server の設定

LiveCycle データを格納する SQL Server データベースに JBoss から接続できるようにするには、アドビの事前設定 JBoss に付属している次のファイルが必要です。

- [appserver_root]/server/lc_sqlserver_cl/lib の SQL Server JDBC ドライバーファイル
- [appserver_root]/server/lc_sqlserver_cl/deploy のアドビデータソースファイル
- [appserver_root]/server/lc_sqlserver_cl/deploy の SQL Server データソースファイル

注意: <http://community.jboss.org/wiki/EncryptingDataSourcePasswords> で説明されているいずれかの方法を使用して、データソースファイル内のパスワードを暗号化します。http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearte.html の手順を使用することもできます。

3.5.2.1 データソースファイルの設定

SQL Server データソースを設定する前に、SQL Server に LiveCycle データベースを作成しておく必要があります (『LiveCycle のインストールの準備 (サーバークラスター)』を参照)。

アドビデータソースファイルを変更するには：

- 1 [appserver root]/server/lc_sqlserver_cl/deploy/adobe-ds.xml ファイルをテキストエディターで開き、次の行を検索します。

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 2 次のテキストをデータベースに固有の値に置き換えます。
 - localhost：データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは localhost です。
 - 1433：データベースへのアクセスに使用するポート。デフォルトのポートは 1433 です。

- **adobe** : LiveCycle データを格納しているデータベースの名前。デフォルト値の **adobe** を、お使いのデータベース名に変更する必要があります。
- 3 <user-name> タグと <password> タグに、アプリケーションサーバーがデータベースにアクセスするときに使用するユーザー名とパスワードを指定します。これらのデフォルト値 (どちらも **adobe**) を、お使いのデータベースの資格情報に変更する必要があります。
 - 4 IDP_DS および EDC_DS の残りの要素について、手順 1 ~ 3 を繰り返します。
 - 5 ファイルを保存します。

SQL Server データソースファイルを変更するには :

- 1 [appserver root]/server/lc_sqlserver_cl/deploy/mssql-ds.xml ファイルをテキストエディターで開き、次の行を検索します。

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>  
<user-name>adobe</user-name>  
<password>adobe</password>
```

- 2 次のテキストをデータベースに固有の値に置き換えます。
 - **localhost** : データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは **localhost** です。
 - **1433** : データベースへのアクセスに使用するポート。デフォルトのポートは **1433** です。
 - **adobe** : LiveCycle データを格納しているデータベースの名前。デフォルト値の **adobe** を、お使いのデータベース名に変更する必要があります。
- 3 <user-name> タグと <password> タグに、アプリケーションサーバーがデータベースにアクセスするときに使用するユーザー名とパスワードを指定します。これらのデフォルト値 (どちらも **adobe**) を、お使いのデータベースの資格情報に変更する必要があります。
- 4 ファイルを保存します。

Windows 上で統合セキュリティを設定するには :

- 1 次の例に示すように、[appserver root]¥server¥all¥deploy にある **adobe-ds.xml** ファイルを変更し、**integratedSecurity=true** を接続 URL に追加します。

```
jdbc:sqlserver://<serverhost>:<port>;databaseName=<dbname>;integratedSecurity=true.
```
- 2 JBoss を実行するコンピューターの Windows システムパス (C:¥Windows) に **sqljdbc_auth.dll** ファイルを追加します。sqljdbc_auth.dll ファイルは、Microsoft SQL JDBC 3.0 ドライバーのインストール先 (デフォルトは <InstallDir>/sqljdbc_3.0/enu/auth/x86) と同じ場所にあります。
- 3 JBoss for Adobe LiveCycle ES4 サービスのプロパティを開き、「**ログオン**」タブをクリックします。
- 4 「**アカウント**」を選択し、有効なユーザーアカウントの値を入力します。コマンドラインから JBoss を実行する場合は、この変更は必要ありません。
- 5 SQL Server のセキュリティを混合モードから Windows 認証のみに変更します。

3.5.2.2 login-config.xml ファイルの編集

- 1 [appserver root]/server/lc_sqlserver_cl/conf/login-config.xml ファイルをテキストエディターで開き、<policy> 要素内の次のコードを変更します。

```
<application-policy name="MSSQLDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.ConfiguredIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=Default DS </module-option>
      </login-module>
    </authentication>
  </application-policy>
```

- 2 太字のテキスト (principal、userName および password パラメーターの値) をデータベースに固有の値に置き換えて、アプリケーションサーバーがデータベースにアクセスできるようにします。
- 3 ファイルを保存して閉じます。
- 4 JBoss を再起動します。

3.5.3 アドビの事前設定 JBoss での MySQL の設定

注意: <http://community.jboss.org/wiki/EncryptingDataSourcePasswords> で説明されているいずれかの方法を使用して、データソースファイル内のパスワードを暗号化します。 http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clear.html の手順を使用することもできます。

3.5.3.1 adobe-ds.xml ファイルの編集

MySQL データソースを設定する前に、データベースを MySQL に作成しておく必要があります。

- 1 [appserver root]/server/lc_mysql_cl/deploy/adobe-ds.xml ファイルをテキストエディターで開き、IDP_DS と EDC_DS の両方について次の行を検索します。

```
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 2 次のテキストをデータベースに固有の値に置き換えます。
 - **localhost**: データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは localhost です。
 - **3306**: データベースへのアクセスに使用するポート。デフォルトのポートは 3306 です。
 - **adobe**: データを格納しているデータベースの名前。デフォルト値 adobe をデータベースの名前に置き換えます。
- 3 <connection-url> 設定に続く行で、user-name および password 設定を探し、デフォルト値を、アプリケーションサーバーがデータベースにアクセスするために使用するユーザー名とパスワードに置き換えます。
- 4 データソース接続の最小値と最大値が次のように設定されていることを確認します。
 - IDP_DS の場合:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```
 - EDC_DS の場合:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>20</max-pool-size>
```

注意： LiveCycle サーバーで処理する負荷が大きい場合は、JDBC 接続の最大数を増やして、すべてのジョブが確実に処理されるようにします。そのような場合は、IDP_DS と EDC_DS の両方で <max-pool-size> を 50 以上に増やします。

5 ファイルを保存して閉じます。

3.5.3.2 mysql-ds.xml ファイルの編集

MySQL データベースと共に LiveCycle を実行する場合は、MySQL を JBoss のデフォルトのデータソースに設定する必要があります。この手順では、MySQL JDBC ドライバーが [appserver root]/server/lc_mysql/lib ディレクトリにインストールされていることを前提としています。

1 [appserver root]/server/lc_mysql_cl/deploy/mysql-ds.xml ファイルをテキストエディターで開き、<local-tx-datasource> 要素を MySQL の接続設定の内容に変更します。

```
<jndi-name>DefaultDS</jndi-name>
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

2 次のテキストをデータベースに固有の値に置き換えます。

- **localhost** : この値を、データベースをホストするサーバーの名前に置き換えます。
- **3306** : この値を、データベースサーバーのポート番号に置き換えます。
- **adobe** : この値を、LiveCycle に接続するデータベースに置き換えます。

3 <connection-url> 設定に続く行で、user-name および password 設定を探し、デフォルト値を、アプリケーションサーバーがデータベースにアクセスするために使用するユーザー名とパスワードに置き換えます。

4 ファイルを保存して閉じます。

3.5.3.3 login-config.xml ファイルの編集

1 [appserver root]/server/lc_mysql/conf/login-config.xml ファイルをテキストエディターで開き、<policy> エレメント内の次のコードを変更します。

```
<application-policy name="MySqlDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.SecureIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=Default DS </module-option>
      </login-module>
    </authentication>
  </application-policy>
```

2 **太字のテキスト** (principal、userName および password パラメーターの値) をデータベースに固有の値に置き換えて、アプリケーションサーバーがデータベースにアクセスできるようにします。

3 ファイルを保存して閉じます。

4 JBoss を起動します。

3.6 JBoss Application Server クラスターのテスト

JBoss Application Server クラスターをテストして、すべてのメンバーがアクティブであること、およびクラスターが設計に従って動作することを確認できます。LiveCycle のインストールおよび設定の前に、JBoss Application Server クラスターが正しく動作することを確認する必要があります。

JBoss Application Server クラスターをテストするには：

- 1 適切なコマンドを入力して、クラスターのすべての JBoss Application Server インスタンスを起動します。

(Windows) `run.bat -g <partition_name> -b <ipaddress or hostname> -c <server_profile>`

(UNIX ベースの環境) `run.sh -g <partition_name> -b <ipaddress or hostname> -c <server_profile>`

注意：IPv6 の場合、上記のコマンドで、システムのホストファイルに IPv6 アドレスにマッピングされる IPv6 アドレスまたは IPv6 ホスト名を使用します。

注意：JBoss Application Server インスタンスを起動する際に、コンピューター（ローカルホストを含む）上のすべてのアドレスをバインドするために、IP アドレスまたはホスト名の代わりに `-b 0.0.0.0` を指定できます。

クラスターの場合、すべての IP アドレスではなく、特定の IP アドレスにバインドすることをお勧めします。

- 2 `[appserver root]/server/<server_profile>/log` にある `server.log` ファイルを表示します。次のようなメッセージで、クラスターのアクティブなメンバーを確認できます。

```
INFO [org.jboss.ha.framework.interfaces.HAPartition.DefaultPartition]
Number of cluster members: 2
INFO [org.jboss.ha.framework.server.DistributedReplicantManagerImpl.
DefaultPartition] All Members : 2
([<IPAddress1>:<Port1>], [<IPAddress2>:<Port2>])
```

第 4 章：LiveCycle モジュールのインストール

4.1 事前準備

4.1.1 インストールの概要

モジュールをインストールする前に、LiveCycle の実行に必要なソフトウェアとハードウェアが使用環境に含まれていることを確認してください。また、各インストールオプションについて理解し、必要に応じて環境を整えておく必要があります。詳しくは、インストールの準備（シングルサーバーまたはサーバークラスター）、アップグレードの準備に関する各ガイドを参照してください。LiveCycle のドキュメント一式は http://www.adobe.com/go/learn_lc_documentation_11_jp から入手できます。

LiveCycle には、インストールプログラム用にコマンドラインインターフェイス（CLI）も用意されています。CLI の使用に関する説明については、79 ページの「[付録 - コマンドラインインターフェイスのインストール](#)」を参照してください。Configuration Manager 用の CLI もあります。82 ページの「[付録 - Configuration Manager コマンドラインインターフェイス](#)」を参照してください。これらの CLI は、LiveCycle の上級ユーザーが使用したり、インストールプログラムや Configuration Manager でグラフィカルユーザーインターフェイスがサポートされていないサーバー環境で使用したり、ユーザーがバッチ（非インタラクティブ）インストール機能を実装したりする場合を想定しています。

4.1.2 インストーラーの確認

インストールプロセスを開始する前に、インストーラーファイルについて、次のベストプラクティスを確認してください。

DVD インストールメディアの確認

入手したインストールメディアが破損していないことを確認します。LiveCycle をインストールするコンピューターのハードディスクにインストーラーのメディアコンテンツをコピーする場合は、必ず、すべての DVD コンテンツをハードディスクにコピーしてください。インストールエラーを避けるには、Windows のパスの最大長を超えるディレクトリパスに DVD インストールイメージをコピーしないでください。

インストールファイルのローカルコピーを使用するか DVD から直接 LiveCycle をインストールします。LiveCycle をネットワークを介してインストールすると、インストールが失敗する場合があります。また、ローカルパスに特殊文字（例えば、文字「#」）は使用しないでください。

ダウンロードしたファイルの確認

アドビの Web サイトからインストーラーをダウンロードした場合は、MD5 チェックサムを使用してインストーラーファイルの整合性を検証してください。次のいずれかを実行し、ダウンロードファイルの MD5 チェックサムを計算して、アドビのダウンロード用 Web ページで公開されているチェックサムと比較します。

- **Linux** : md5sum コマンドを実行します。
- **Solaris** : digest コマンドを実行します。
- **Windows** : WinMD5 などのツールを実行します。

ダウンロードしたアーカイブファイルの展開

アドビの Web サイトから ESD をダウンロードした場合は、lces_server_11_0_0_jboss_all_win.zip (Windows) または lces_server_11_0_0_jboss_all_unix.tar.gz (Linux または Solaris) アーカイブファイル全体をコンピューターに展開します。Solaris の場合は、gunzip コマンドを使用して .gz ファイルを展開します。

注意：元の ESD ファイルのディレクトリ階層は変更しないようにしてください。

4.2 インストールに関する考慮事項

4.2.1 インストールパス

正常にインストールするには、インストールディレクトリに対する読み取り、書き込みおよび実行権限が必要です。デフォルトのインストールディレクトリは以下のとおりですが、必要に応じて、別のディレクトリを指定することもできます。

- (Windows) C:\Adobe\Adobe LiveCycle ES4
- (Linux または Solaris) /opt/adobe/adobe_lifecycle_es4

重要: LiveCycle をインストールするときに、インストールパスに 2 バイト文字または拡張ラテン文字 (ââçèèëïïòùüÄÖÛ など) を使用しないでください。

重要: (Windows のみ) LiveCycle インストールディレクトリのパスには、非 ASCII 文字 (例えば、é や ñ などのインターナショナル文字) を使用しないでください。使用した場合には、JBoss for Adobe LiveCycle ES4 サービスを起動できません。

モジュールを UNIX 系のシステムにインストールする際に、デフォルトのインストール先である /opt/adobe/adobe_lifecycle_es4 に正常にインストールするには、ルートユーザーでログインする必要があります。ルートユーザー以外でログインした場合は、権限 (読み取り、書き込み、実行権限) を持っている別のディレクトリにインストール先を変更してください。例えば、ディレクトリを /home/[username]/adobe/adobe_lifecycle_es4 に変更します。

注意: UNIX 系のシステムでは、ソース (インストールメディア) からファイルをコピーまたはダウンロードすると、install.bin で実行権限が失われる場合があります。ファイルをコピーまたはダウンロードした後で、書き込み、実行権限を復元してください。

Windows では、LiveCycle のインストールには管理者権限が必要です。

4.2.2 一時ディレクトリ

一時ファイルは、一時ディレクトリに生成されます。生成された一時ファイルが、インストーラーの終了後も残る場合があります。これらのファイルは手動で削除することができます。

Linux でのインストールでは、インストールプログラムにより、ログインしているユーザーのホームディレクトリがファイルを格納するための一時ディレクトリとして使用されます。そのため、次のようなメッセージがコンソールに表示される場合があります。

```
WARNING: could not delete temporary file /home/<username>/ismp001/1556006
```

インストールが完了したら、次のディレクトリから一時ファイルを手動で削除する必要があります。

- (Windows) 環境変数で設定されている TMP または TEMP パス
- (Linux または Solaris) ログインユーザーのホームディレクトリ

UNIX 系のシステムでは、root 以外のユーザーは次のディレクトリを一時ディレクトリとして使用できます。

- (Linux) /var/tmp or /usr/tmp
- (Solaris) /var/tmp または /usr/tmp

4.2.3 Linux または UNIX にインストールするための Windows ステージングプラットフォームへのインストール

Linux または UNIX プラットフォームにデプロイするために、LiveCycle を Windows にインストールして設定することができます。この機能を使用して、ロックダウンされた Linux または UNIX 環境にインストールできます。ロックダウンされた環境にはグラフィカルユーザーインターフェイスはインストールされていません。Linux または UNIX プラットフォームの場合、インストールプログラムにより、Configuration Manager で製品を設定するために使用されるバイナリがインストールされます。

その後、Windows を実行するコンピューターを、デプロイ可能なオブジェクトのステージング場所として使用できます。これらのオブジェクトは、アプリケーションサーバーへのデプロイメント用に Linux または UNIX コンピューターにコピーできます。Windows ベースのコンピューター上のアプリケーションサーバーと、LiveCycle をインストールする Linux または UNIX ターゲットコンピューターは、同じである必要があります。

4.2.4 JAVA_HOME 環境変数の設定

JAVA_HOME 環境変数は、準備ガイドに説明されているように、アプリケーションサーバーの Java SDK を指している必要があります。詳しくは、『[LiveCycle のインストールの準備 \(シングルサーバー\)](#)』または『[LiveCycle のインストールの準備 \(サーバークラスター\)](#)』を参照してください。

4.2.5 インストールに関する一般的な注意

- Windows の場合は、インストール中にオンアクセスウイルススキャンソフトウェアを無効にすることにより、インストールに要する時間が短縮されます。
- UNIX 系のシステムにインストールするが、リリース DVD からは直接インストールしない場合は、インストールファイルに実行権限を設定します。
- デプロイメントの際に権限の問題を回避するため、アプリケーションサーバーを実行する場合と同じユーザーで、LiveCycle インストーラーおよび Configuration Manager を実行してください。
- UNIX 系コンピューターにインストールする場合は、指定するインストールディレクトリ名にスペースを含めないでください。
- インストール中にエラーが発生した場合は、install.log ファイルが作成され、エラーメッセージが記録されます。このログファイルは、[LiveCycle root]/log ディレクトリに作成されます。
- JAVA_HOME 環境変数が互換性のある JDK を含むディレクトリを指していることを確認します。詳しくは、[サポートされているプラットフォームの組み合わせ](#)を参照してください。

4.3 LiveCycle のインストール

1 インストールプログラムを起動します。

- (Windows) インストールメディア上、またはインストーラーをコピーしたハードディスク上のフォルダーの %server%\Disk1\InstData\Windows_64\VM ディレクトリに移動します。install.exe ファイルを右クリックし、「管理者として実行」を選択します。

注意: 32 ビットバージョンの LiveCycle も使用できます。対応するディレクトリに移動し、インストールファイルを選択して、インストーラーを起動します。ただし、32 ビットバージョンは、開発または評価目的のみサポートされており、実稼働環境ではサポートされていません。

- (Windows 以外) 適切なディレクトリに移動して、コマンドプロンプトで .install.bin と入力します。
 - (Linux) /server/Disk1/InstData/Linux/NoVM

- (Solaris) /server/Disk1/InstData/Solaris/NoVM
- 2 プロンプトが表示されたら、インストールプログラムで使用する言語を選択して、「OK」をクリックします。
 - 3 ようこそ画面で「次へ」をクリックします。
 - 4 インストーラーを実行するコンピューターに、LiveCycle ES 2、ADEP、または LiveCycle ES3 の以前のバージョンがインストールされている場合は、アップグレードの準備画面が表示されます。
注意：新しいコンピューターでアウトオブプレースアップグレードを実行する場合は、この画面は表示されません。
- **既存のインストールを Adobe LiveCycle ES4 にアップグレードする準備**
新規インストールを行う場合は、このオプションを選択しないでください。
 - **Adobe LiveCycle ES4 をインストール：**LiveCycle を新規にインストールします。
「次へ」を選択して、続行します。
- 5 インストールフォルダーを選択画面で、デフォルトのディレクトリをそのまま使用するか、「**選択**」をクリックして LiveCycle のインストール先ディレクトリを選択してから、「次へ」をクリックします。存在しないディレクトリの名前を入力すると、そのディレクトリが作成されます。
「デフォルトのフォルダーに戻す」をクリックすると、デフォルトのディレクトリパスに戻すことができます。
 - 6 インストールタイプを選択画面で、**カスタム／手動**を選択して、「次へ」をクリックします。
 - 7 (**Windows のみ**) 手動インストールオプション画面で、目的のデプロイメントオプションを選択し、「次へ」をクリックします。
 - **Windows (ローカル)：**ローカルサーバーに LiveCycle をインストールおよびデプロイする場合は、このオプションを選択してください。
 - **リモート (下記のリモートオペレーティングシステムを対象とする)：**デプロイメント用のステージングプラットフォームとして Windows を使用する場合は、このオプションを選択します。その後で、リモートサーバー上のターゲットオペレーティングシステムを選択します。Windows 上でインストールを行っている場合でも、デプロイメント対象として UNIX オペレーティングシステムを選択できます (19 ページの「[4.2.3 Linux または UNIX にインストールするための Windows ステージングプラットフォームへのインストール](#)」を参照)。
 - 8 Adobe LiveCycle ES4 の使用許諾契約書を読み、「**同意します**」を選択して使用許諾契約書の条件に同意し、「次へ」をクリックします。使用許諾契約書に同意しない場合は、操作を継続することはできません。
 - 9 プリインストールの概要画面で、詳細を確認して「**インストール**」をクリックします。インストールプログラムによりインストールの進行状況が表示されます。
 - 10 リリースノートの情報を確認して「次へ」をクリックします。
 - 11 インストール完了画面の詳細情報を確認します。
 - 12 「**LiveCycle Configuration Manager を起動**」チェックボックスはデフォルトで選択されています。「完了」をクリックして Configuration Manager を実行します。
注意：(Windows 版 Adobe® LiveCycle® PDF Generator 11 のみ) クラスター内のすべてのノードに Acrobat がインストールされていない場合は、今すぐインストールしてください。次に、46 ページの「[6.6 PDF Generator の設定](#)」の手順を実行します。
- 注意：** Configuration Manager を後で実行するには、「完了」をクリックする前に、「**LiveCycle Configuration Manager を起動**」オプションの選択を解除します。[LiveCycle root]/configurationManager/bin ディレクトリにある該当するスクリプトを使用して、Configuration Manager を後で起動することができます。このガイドの「LiveCycle をデプロイするための設定」の章を参照してください。

4.4 クラスター内のキャッシュロケータの設定 (TCP を使用するキャッシュのみ)

TCP を使用して LiveCycle クラスターのキャッシュを実装する場合、LiveCycle クラスターの他のメンバーを検索するために TCP ロケータを設定します。

注意: この節は、UDP を使用して LiveCycle クラスター用にキャッシュを実装する場合には適用されません (UDP を使用して LiveCycle クラスター用にキャッシュを設定するには、「8 ページの「[3.4 JBoss run.conf ファイルの変更](#)」を参照)。

TCP を使用して LiveCycle クラスターのキャッシュを有効にするには、次の作業を行う必要があります。

- TCP ロケータがインストールおよび設定されていることを確認します。TCP ロケータは、LiveCycle のインストール時に、デフォルトの設定で、ディレクトリ **[LiveCycle root]/lib/caching** にインストールされます。デフォルトの設定は変更することができます (「TCP ロケータの変更」を参照)。
- ロケータを使用するように LiveCycle クラスター内の各ノードを設定します (8 ページの「[3.4 JBoss run.conf ファイルの変更](#)」を参照)。
- TCP ロケータが実行されていることを確認します。

4.4.1 TCP ロケータの変更

LiveCycle インストーラーによって、変更せずに使用できる TCP ロケータのデフォルト設定が作成されます。ロケータをネットワーク上の任意のコンピューターに移動し、そのコンピューターで実行できます。ロケータが存在するコンピューターは、LiveCycle クラスターのメンバーである必要はありません。クラスターで高可用性をサポートするために、フェイルオーバーロケータを追加作成することもできます (「TCP ロケータをインストールするには:」を参照)。

TCP ロケータを変更して、デフォルトのポート (22345) 以外のポートを使用することもできます (「デフォルトのロケータポートの変更 (Windows)」または「デフォルトのロケータポートの変更 (UNIX)」を参照)。

4.4.2 TCP ロケータのインストール

1 LiveCycle がインストールされているコンピューターにログオンし、**[LiveCycle root]/lib/caching** ディレクトリに移動します。

2 **caching** ディレクトリとその内容を、ロケータを実行するコンピューターにコピーします。

デフォルトの場所から TCP ロケータを開始できます。次の場合のみ、**caching** ディレクトリを別の場所にコピーする必要があります。

- LiveCycle がインストールされていないマシンで TCP ロケータを実行します。
- デフォルトの場所から TCP ロケータを開始しないでください。

注意: LiveCycle Cluster のすべてのノードで TCP ロケータを実行しないでください。最低 2 つ実行することをお勧めします。1 つの TCP locator がプライマリロケータとなり、他の TCP ロケータがフェイルオーバーの問題を処理するセカンダリロケータになります。3 つ以上の TCP ロケータをバックアップロケータとして追加できますが、必須ではありません。

4.4.3 デフォルトのロケータポートの変更 (Windows)

1 **startlocator.bat** ファイルをテキストエディターで開きます。デフォルトインストールの **startlocator** ファイルは、LiveCycle をインストールしたコンピューターの **[LiveCycle root]/lib/caching** ディレクトリにあります。

2 次のプロパティで、デフォルトのポート番号 (22345) を任意のポート番号に変更します。

```
set port=22345
```

ポート番号は、1025 ~ 65535 の使用可能なポートにすることができます。

注意: ここで設定するポート番号は、LiveCycle クラスターの各ノードの JVM 引数で設定したものと一致させる必要があります。このポート番号は、1025 から 65535 の範囲の任意の使用可能なポートが使用できます。設定を完了する手順については、8 ページの「[3.4 JBoss run.conf ファイルの変更](#)」を参照してください。

- 3 フェイルオーバーとして複数のロケーターを使用する場合は、これらのロケーターのすべてを、startlocator.bat ファイルのサイドの JVM 引数 -Dlocators に割り当ててください。

```
-Dlocators=localhost[22345]
```

- 4 (複数のネットワークカードがあるコンピューターのみ) ロケーターをホストするコンピューターに複数のネットワークカードがある場合は、スクリプト内の次のプロパティを設定します。

```
set bindaddr=<bind IP address>
```

<bind IP address> は、ロケーターがリスンする IP アドレスです。LiveCycle クラスター内の各ノードで JVM 引数 adobe.cache.cluster-locators の <bind IP address> を指定する必要があります。

注意: startlocator スクリプトにバインドアドレスとバインドポートを指定しない場合、スクリプトの実行時にこれらの値を入力するよう求められます。ただし、IPv6 の場合は、startlocator スクリプト自体にバインドアドレスとバインドポートを指定する必要があります。

- 5 編集したファイルを保存します。
- 6 LiveCycle クラスターのその他のロケーターについて、手順 1 ~ 4 を繰り返します。

4.4.4 デフォルトのロケーターポートの変更 (UNIX)

- 1 startlocator.sh ファイルをテキストエディターで開きます。デフォルトインストールの startlocator ファイルは、LiveCycle をインストールしたコンピューターの [LiveCycle root]/lib/caching ディレクトリに配置されます。
- 2 次のプロパティで、デフォルトのポート番号 (22345) を任意のポート番号に変更します。

```
GF_PORT=22345
```

ポート番号は、1025 ~ 65535 の使用可能なポートにすることができます。

重要: ここで設定するポート番号が、LiveCycle クラスターの各ノードの JVM 引数で設定されるポート番号と一致することを確認します。

- 3 フェイルオーバーとして複数のロケーターを使用する場合は、これらのロケーターのすべてを、startlocator.sh ファイルのサイドの JVM 引数 -Dlocators に割り当ててください。

```
-Dlocators=localhost[22345]
```

- 4 (複数のネットワークカードがあるコンピューターのみ) ロケーターをホストするコンピューターに複数のネットワークカードがある場合は、次の引数を変更します。

```
GF_BIND_ADDRESS="<bind IP address>"
```

<bind IP address> は、ロケーターがリスンする IP アドレスです。LiveCycle クラスター内の各ノードで JVM 引数 adobe.cache.cluster-locators の <bind IP address> を指定する必要があります。

注意: IPv6 の場合は、startlocator スクリプト自体にバインドアドレスとバインドポートを指定することをお勧めします。

- 5 編集したファイルを保存します。
- 6 LiveCycle クラスターのその他のロケーターについて、手順を繰り返します。

4.4.5 TCP ロケーターの起動

TCP ロケーターを使用して LiveCycle クラスターに対して TCP ベースのキャッシングを使用するには、クラスターを開始する前に、TCP ロケーターを開始する必要があります。LiveCycle クラスターのメンバーを起動するときに TCP ロケーターが実行されていない場合、LiveCycle クラスターは機能しません。

1 TCP ロケーターがインストールされているコンピューターで、 `caching` ディレクトリに移動します。デフォルトインストールでは、TCP ロケーターは、LiveCycle をインストールしたコンピューターの `[LiveCycle root]/lib/caching` ディレクトリにインストールされています。

2 (IPv6 のみ) `startlocator.bat` (Windows) または `startlocator.sh` (UNIX) を変更して、次の JVM 引数を追加します。

```
-Djava.net.preferIPv6Stack=true  
-Djava.net.preferIPv6Addresses=true
```

3 適切なファイルを実行します。

- (Windows) `startlocator.bat`
- (UNIX) `startlocator.sh`

4 LiveCycle クラスターのその他のロケーターについて、以上の手順を繰り返します。

注意: (Windows のみ) `startlocator` スクリプトの実行時に、デフォルト値を変更するように求められます。スクリプトに指定されたデフォルト値をそのまま使用することも、新しい値を指定することもできます。

4.4.6 TCP ロケーターの停止

1 TCP ロケーターがインストールされているコンピューターで、 `caching` ディレクトリに移動します。デフォルトインストールでは、TCP ロケーターは、LiveCycle をインストールしたコンピューターの `[LiveCycle root]/lib/caching` ディレクトリにインストールされています。

2 適切なファイルを実行します。

- (Windows) `stoplocator.bat`
- (UNIX) `stoplocator.sh`

3 LiveCycle クラスターのその他のロケーターについて、手順 1 ~ 2 を繰り返します。

注意: `startlocator` スクリプトおよび説明した特定の IP アドレスやポートの値にデフォルト値を使用していない場合は、`stoplocator` スクリプトの値と同じ値を指定します。このように指定しないと、`stoplocator` スクリプトはロケーターの停止に失敗します。

4.5 グローバルドキュメントストレージディレクトリ (GDS)

クラスターの設定で、LiveCycle クラスターの各ノードにアクセス可能な DGS ディレクトリのために共有ファイルシステムを作成します。この共有ファイルシステムは、コンピューター上のローカルストレージまたは専用のネットワークストレージシステム上の共有のいずれでもかまいません。クラスターのすべてのノードが、共有ストレージに対して読み書き権限を持つようにしてください。GDS ディレクトリは、低アクセス時間、高稼働可能時間を持てるようにし、UNC 形式のパスとしてアクセスできるようにしてください。例えば、`\\storagename\shared\GDS`。

4.6 クラスターノードでのフォントディレクトリのインストール

クラスター内の各ノードに、**[LiveCycle ルート]**¥fonts ディレクトリにインストールされている LiveCycle のフォントを含めて、フォントディレクトリをインストールする必要があります。

フォントはクラスター内の各ノードで同じパスに存在する必要があり、フォントディレクトリのコンテンツもクラスター内のすべてのノードで同一になっている必要があります。このように設定するには、次のいずれかの操作を行います。

- クラスター内のすべてのノードがアクセスできる共有ディレクトリを使用します。**[LiveCycle ルート]** ¥fonts にあるフォントを共有ディレクトリにコピーします。共有フォントディレクトリを使用すると、フォントへのアクセスが遅くなったり、パフォーマンスの問題が発生したりする可能性があります。
- 同じパスを使用して、クラスター内の各ノードに **[LiveCycle root]**¥fonts ディレクトリをコピーします。

これらの共有ディレクトリを作成した場所を記録しておき、後で Configuration Manager を使用して LiveCycle を設定するときに使用できるようにします。

注意：フォントディレクトリは、GDS ディレクトリと別にする必要があります。ただし、単一の共有親ディレクトリの個別の兄弟サブディレクトリとして存在させることができます。前述の必要システム条件と手順は、カスタムフォントディレクトリに対して適用できます。

4.7 次の手順

デプロイする LiveCycle を設定する必要があります。**[LiveCycle root]**¥configurationManager¥bin にある ConfigurationManager.bat ファイルまたは ConfigurationManager.sh ファイルを使用して、Configuration Manager を後で実行することもできます。

第 5 章：LiveCycle をデプロイするための設定

5.1 LiveCycle の設定およびデプロイにおける考慮事項

5.1.1 一般的な考慮事項

- IPv6 の場合は、IPv6 LiveCycle Configuration Manager を実行します。詳しくは、[インストール準備ガイド](#) の LiveCycle IPv6 サポートの項を参照してください。
- Configuration Manager のデフォルトのフォントを上書きできます。これを行うには、[LiveCycle root]¥ConfigurationManager¥Bin¥ConfigurationManager.bat (Windows) または [LiveCycle root]/ConfigurationManager/Bin/ConfigurationManager.sh (Linux、UNIX) に、次の JVM 引数を追加します。

```
-Dlcm.font.override=<FONT_FAMILY_NAME>
```

次に例を示します。

```
-Dlcm.font.override=SansSerif
```

JVM 引数を追加したら、Configuration Manager を再起動します。

- 設定中に、「Reset to Default」オプションを使用して Configuration Manager 内のデータをリセットする必要がある場合は、Configuration Manager を必ず再起動してください。再起動しない場合、一部の設定画面が表示されない場合があります。
- 設定では、データベースの JDBC ドライバーの場所を指定する必要があります。Oracle、SQL Server および DB2 のドライバーは、[LiveCycle root]/lib/db/[database] ディレクトリにあります。MySQL では、必要な JDBC ドライバーをダウンロードしてインストールします。

JBoss を手動で設定した場合は、データベースドライバーをダウンロードして、[appserver root]/server/<profile_name>/lib にコピーする必要があります。

- 一時ディレクトリ：クラスター設定時に、共有ネットワークディレクトリを一時ディレクトリとして指定しないでください。ローカルディレクトリを一時ディレクトリとして使用することをお勧めします。一時ディレクトリはクラスターのすべてのノード上に存在していなければならない、かつ一時ディレクトリのパスはクラスターのすべてのノードで同じでなければなりません。
- グローバルドキュメントストレージ (GDS) ディレクトリ：インストールの準備 (シングルサーバーまたはサーバークラスター) ガイドで説明されている要件を満たす GDS ディレクトリを指定してください。最新のドキュメントについては、http://www.adobe.com/go/learn_lc_documentation_11_jp を参照してください。
- クラスター環境では、Configuration Manager が行う自動設定に加えて、いくつかの手順を手動で実行する必要があります。

5.1.2 Configuration Manager の CLI バージョンと GUI バージョンの比較

この項では、Configuration Manager の GUI バージョンについて説明します。Configuration Manager のコマンドラインインターフェイス (CLI) バージョンの使用については、82 ページの「[付録 - Configuration Manager コマンドラインインターフェイス](#)」を参照してください。

LiveCycle の設定のタスク	Configuration Manager GUI	Configuration Manager CLI	手動
LiveCycle を設定	○	○	×
LiveCycle データベースの初期化	○	○	×
LiveCycle サーバー接続を検証	○	○	×
LiveCycle コンポーネントのデプロイ	○	○	×
LiveCycle コンポーネントのデプロイメントの検証	○	○	○
LiveCycle コンポーネントの設定	○	○	○

5.1.3 JBoss Application Server の考慮事項

JBoss の場合、「アプリケーションサーバーを設定」、「アプリケーションサーバーの設定を検証」、「LiveCycle EAR をデプロイ」の各タスクはありません。

LiveCycle の JBoss へのデプロイの説明に従って、JBoss を設定し、LiveCycle EAR を手動でデプロイする必要があります。

5.1.4 LiveCycle サーバークラスターの設定時の考慮事項

- Configuration Manager を使用して IPv6 ベースのクラスターの設定を指定することはできません。
- クラスター内の各ノードで、同じパスにローカルサーバーフォントとカスタムフォントのディレクトリを配置することをお勧めします。ローカルフォントディレクトリの代わりに共有フォントディレクトリを使用すると、パフォーマンスの問題が発生する可能性があります。

5.1.5 日付、時刻およびタイムゾーンの設定

LiveCycle 環境に接続するすべてのサーバーで正しい日付、時刻およびタイムゾーンを設定することで、時間に依存するモジュール (Adobe® LiveCycle® Digital Signatures 11 や Reader Extensions 11 など) が正常に機能するようになります。例えば、未来の時間に作成された署名は、有効になりません。

時間同期を必要とするサーバーは、データベースサーバー、LDAP サーバー、HTTP サーバーおよび J2EE サーバーです (アプリケーションサーバー)。

注意: LiveCycle クラスター内で使用するすべてのマシンは時間を同期させてください。

5.2 LiveCycle の事前設定タスク

注意: Configuration Manager の実行中に **F1** キーを押すと、現在表示されている画面に関するヘルプ情報が表示されます。「進行状況ログを表示」をクリックすると、いつでも設定の進行状況を確認できます。

- 1 インストールプログラムで Configuration Manager が自動的に起動しなかった場合は、**[LiveCycle root]/configurationManager/bin** ディレクトリに移動し、**ConfigurationManager.bat/sh** スクリプトを実行します。
- 2 プロンプトが表示されたら、Configuration Manager で使用する言語を選択して、「**OK**」をクリックします。
- 3 ようこそ画面で「**次へ**」をクリックします。
- 4 アップグレードタスクの選択画面では、どのオプションも選択しないで、「**次へ**」をクリックします。
- 5 モジュール画面で、設定する Adobe LiveCycle ES4 モジュールを選択し、「**次へ**」をクリックします。

LiveCycle をデプロイするための設定

注意：適切な設定と機能のために、一部のモジュールは他のモジュールとのテクニカルな依存関係をもちます。相互依存するモジュールが選択されていない場合、Configuration Manager はダイアログを表示し、それより先の操作はできなくなります。例えば、Correspondence Management Solution を設定する際には、Adobe LiveCycle Forms、Adobe LiveCycle Output、コンテンツリポジトリモジュールを選択する必要があります。

- 6 タスク選択画面で、実行するすべてのタスクを選択し、「次へ」をクリックします。

注意：JBoss アプリケーションサーバーの場合、「アプリケーションサーバーを設定」、「アプリケーションサーバーの設定を検証」、「LiveCycle ES4 EAR ファイルをデプロイ」の各タスクはありません。

5.3 LiveCycle の構成およびデプロイ

注意：Configuration Manager の実行中に **F1** キーを押すと、現在表示されている画面に関するヘルプ情報が表示されません。

LiveCycle の設定

- 1 LiveCycle ES4 を設定 (1/5) 画面で、「設定」をクリックし、完了後に「次へ」をクリックします。
- 2 LiveCycle ES4 を設定 (2/5) 画面で、LiveCycle でフロントへのアクセスに使用するディレクトリを設定し、「次へ」をクリックします。



この画面上の値を変更するには、「設定を編集」をクリックします。このボタンは、Configuration Manager を最初に実行したときには使用できませんが、2 回目およびそれ以降の実行では使用できるようになります。

- (オプション)「**Adobe サーバーフォントディレクトリ**」のデフォルトの場所を変更するには、パスを入力するか、ディレクトリを参照します。
- 「**カスタマーフォントディレクトリ**」のデフォルトの場所を変更するには、「参照」をクリックするか、カスタマーフォントの新しい場所を指定します。

注意：アドビ システムズ社以外が提供しているフォントを使用するユーザーの権利は、それらのフォントを所有する会社が提供する使用許諾契約書に拘束されるもので、アドビソフトウェアを使用するための使用許諾契約書は適用されません。アドビ システムズ社以外が提供しているフォントをアドビソフトウェアで使用する前に、適用されるアドビ システムズ社以外の使用許諾契約書すべてに準拠していることを確認してください。特に、サーバー環境でフォントを使用する際は注意が必要です。

- (オプション)「**システムフォントディレクトリ**」のデフォルトの場所を変更するには、パスを入力するか、ディレクトリを参照します。リストにさらにディレクトリを追加するには、「追加」をクリックします。
- (オプション) FIPS を有効にするには、「**FIPS を有効にする**」を選択します。このオプションは、連邦情報処理規格 (FIPS) を適用する場合にのみ選択してください。

- 3 LiveCycle ES4 を設定 (3/5) 画面で、「参照」をクリックし、「一時ディレクトリの場所」を指定します。

注意：一時ディレクトリを指定しない場合は、システム設定のデフォルトの一時ディレクトリが使用されます。

- 4 LiveCycle ES4 を設定 (4/5) 画面で、「参照」をクリックし、グローバルドキュメントストレージ (GDS) ディレクトリのパスを指定します。

注意：GDS ディレクトリのフィールドを空白のままにすると、LiveCycle によって、アプリケーションサーバーのディレクトリツリーにあるデフォルトの場所にディレクトリが作成されます。設定手順の完了後、Administration Console / 設定 / コアシステム設定 / 設定からその場所にアクセスできます。

注意：クラスターのすべてのノードから GDS ディレクトリにアクセスできることを確認してください。クラスターについては、このディレクトリフィールドを空白のままにしないでください。

5 永続的なドキュメントストレージを設定 (5/5) 画面で、GDS ディレクトリのほかに、永続的なドキュメントストレージのオプションを選択します。次のいずれかを選択します。

- **GDS を使用**：すべての永続的なドキュメントストレージにファイルシステムベースの GDS を使用します。このオプションでは、最高のパフォーマンスを実現し、ストレージの場所として GDS だけを使用します。
- **データベースを使用**：永続的なドキュメントや長期間有効な成果物の保存に、LiveCycle データベースを使用します。ただし、ファイルシステムベースの GDS も必要です。データベースを使用することにより、バックアップと復元の手順が簡単になります。

「設定」をクリックし、LiveCycle EAR にこのディレクトリ情報を設定します。設定が完了したら、「次へ」をクリックします。

PDF Generator 用の Acrobat の設定

❖ (Windows のみ) Acrobat を LiveCycle PDF Generator に合わせて設定画面で、「設定」をクリックして、Adobe Acrobat および必要な環境設定を設定するスクリプトを実行します。完了したら「次へ」をクリックします。

注意：この画面では、Configuration Manager がローカルで実行されている場合にのみ、必要な設定が実行されます。Adobe Acrobat XI Pro が既にインストールされている必要があります。インストールされていないと、この手順は失敗します。

注意：クラスターの他のすべてのノードに、PDF Generator 用の Acrobat を手動で設定する必要があります。デプロイメント後のタスクに関する章の 46 ページの「6.6 PDF Generator の設定」を参照してください。

LiveCycle の設定の概要

❖ LiveCycle ES4 の概要を設定画面で、「次へ」をクリックします。設定したアーカイブは [LiveCycle root]/configurationManager/export ディレクトリに配置されます。

注意：クラスター内の各 JBoss Application Server のインスタンスを停止します。

CRX の設定

❖ CRX 設定画面では、CRX レポジトリを設定し、それを LiveCycle Core EAR ファイルにインストールすることができます。この画面で、レポジトリへのパスを指定し、「設定」をクリックして、指定した場所に必要なレポジトリを作成します。レポジトリに対してカスタムパスを使用する場合は、ファイルシステムにすでにカスタムディレクトリが含まれていることを確認してください。

注意：(自動以外のみ) LiveCycle サーバーをリモートで実行している場合、「Server is running on remote host」を選択し、リモートホスト上のレポジトリへのパスを指定します。

「Next」をクリックして、続行します。

注意：CRX レポジトリパスに空白が含まれていないことと、コンテンツレポジトリがクラスターのすべてのノードで使用できることを確認してください。設定が完了したら、コンテンツレポジトリをローカルノードから (CRX 設定画面で指定した) 同じ場所にあるすべてのノードにコピーします。

注意：パッケージが構成済みになると、Configuration Manager を再実行して削除することはできません。デプロイ済みパッケージをアンインストールするには、Package Manager を使用してアンインストールおよび削除する必要があります。

(リモートホストのみ) CRX 設定サマリー

❖ リモートでデプロイする場合は、[LiveCycle root]/configurationManager/export/crx-quickstart/ ディレクトリの内容を、CRX 設定画面で指定したリモートホストの場所へコピーします。

CRX コンテンツのコピー

[LiveCycle root]/configurationManager/export/crx-quickstart/ ディレクトリのすべてのコンテンツを、すべてのクラスターノード上の、CRX 設定画面で指定した場所にコピーします。

LiveCycle EAR のデプロイ

❖ Configuration Manager を実行したまま、LiveCycle EAR ファイルを JBoss に手動でデプロイします。これを行うには、次のファイルを、[LiveCycle root]/configurationManager/export ディレクトリから、指定されたディレクトリにコピーします。

- adobe-livecycle-native-jboss-[OS].ear
- adobe-livecycle-jboss.ear
- adobe-workspace-client.ear (Adobe® LiveCycle® Process Management 11 のみ)

手動で設定した JBoss (クラスター上) [appserver root]/server/all/deploy

アドビにより事前設定されたクラスター上の JBoss [appserver root]/server/lc_<db-name>_cl/deploy。

これらのファイルは各クラスターノードにコピーする必要があります。

必要に応じて、Adobe® LiveCycle® Forms Standard 11、Adobe® LiveCycle® Output 11、Adobe® LiveCycle® Mobile Forms、および Assembler IVS EAR もデプロイできます。

Correspondence Management の発行インスタンスを作成するには、dobe-livecycle-cq-publish.ear をデプロイします。adobe-livecycle-cq-publish.ear が別のサーバーにデプロイされていることを確認します。adobe-livecycle-cq-publish.ear を LiveCycle サーバーにデプロイしないでください。発行インスタンスの設定について詳しくは、40 ページの「[6.5.2 発行インスタンスの設定](#)」を参照してください。

注意：データソース定義ファイルを、データベースサーバーとデータベースを指すように変更する必要があります。詳しくは、「付録 - データソースの手動設定」を参照してください。

重要：IVS EAR ファイルを実稼働環境にデプロイすることは、お勧めしません。

JBoss を起動して LiveCycle アプリケーションが正常に起動されたことを確認し、Configuration Manager に戻ります。

LiveCycle データベースの初期化

1 LiveCycle ES4 データベースの初期化画面で、アプリケーションサーバーに指定したホスト名とポート番号が正しいことを確認してから、「**初期化**」をクリックします。データベースの初期化タスクによって、データベースにテーブルが作成され、デフォルトのデータがテーブルに追加されて、データベースに基本的なロールが作成されます。初期化が正常に完了したら、「**次へ**」をクリックします。指示があったら、アプリケーションサーバーを手動で再起動します。

注意：データベースの初期化は、クラスター内の 1 つのサーバーに対してのみ実行します。それ以降の手順は、初期化したサーバーに対してのみ実行します。

2 LiveCycle ES4 情報画面で、「**LiveCycle ES4 ユーザー ID**」と「**パスワード**」(デフォルト値はそれぞれ、administrator と password) を入力します。

「**サーバー接続を検証**」をクリックし、完了したら、「**次へ**」をクリックします。

注意：この画面に表示されるサーバー情報はデプロイメントの既定値です。

サーバー接続の検証は、デプロイメントや検証でエラーが発生した場合に、トラブルシューティングの対象を絞り込むのに役立ちます。接続テストが正常に終了しても以降の段階でデプロイメントや検証のエラーが発生する場合は、接続の問題をトラブルシューティングのプロセスから除外できます。

Central Migration Bridge Service のデプロイ

❖ Central Migration Bridge Service デプロイメント設定画面が表示される場合は、この画面で「**Central Migration Bridge Service をデプロイメントに含める**」オプションを選択し、「**次へ**」をクリックします。

LiveCycle コンポーネントのデプロイ

- 1 LiveCycle ES4 コンポーネントのデプロイメント画面で、「**デプロイ**」をクリックします。ここでデプロイされるコンポーネントは、サービスのデプロイ、統合および実行を目的として LiveCycle サービスコンテナにプラグインされている Java アーカイブファイルです。デプロイメントが正常に完了したら、「**次へ**」をクリックします。
- 2 LiveCycle ES4 コンポーネントのデプロイメントの検証画面で、「**検証**」をクリックします。検証が正常に完了したら、「**次へ**」をクリックします。

LiveCycle コンポーネントの設定

- ❖ LiveCycle コンポーネントを設定画面で、Configuration Manager で実行するタスクを選択し、「**次へ**」をクリックします。

LiveCycle サーバー JNDI 情報

- ❖ LiveCycle ES4 サーバー JNDI 情報画面で、JNDI サーバーのホスト名、ポート番号および JBoss クライアント JAR の場所を入力します。詳しくは、F1 キーを押してください。「**サーバー接続を検証**」をクリックし、Configuration Manager が JNDI サーバーに接続できることを確認します。「**次へ**」をクリックして、続行します。

Connector for EMC Documentum

注意: リモート LiveCycle デプロイメントの場合は、Configuration Manager を使って EMC ドキュメンタムのコネクターを設定することはできません。

- 1 EMC Documentum のクライアントを指定画面で、「**Connector for EMC Documentum コンテンツサーバーを設定します**」を選択して、次の情報を指定します。詳細情報を入力して、「**確認**」をクリックし、完了したら、「**次へ**」をクリックして次に進みます。
 - **EMC Documentum クライアントバージョンを選択:** EMC Documentum コンテンツサーバーで使用するクライアントバージョンを選択します。
 - **EMC Documentum クライアントのインストールディレクトリのパス:**「**参照**」をクリックしてディレクトリパスを選択します。

注意: Documentum 6.7 については手動で設定してください。LCM では Documentum 6.7 はサポートされません。
- 2 EMC Documentum Content Server 設定を指定画面で、EMC Documentum Server の詳細情報を入力し、「**次へ**」をクリックします。入力する必要がある情報について詳しくは、F1 キーを押してください。
- 3 Connector for EMC Documentum を設定画面で、「**Documentum Connector を設定**」をクリックします。完了したら、「**次へ**」をクリックします。
- 4 Connector for EMC Documentum に必要な手動設定画面で、一覧の手動による手順を確認および実行し、「**次へ**」をクリックします。

Connector for IBM Content Manager

注意: リモート LiveCycle デプロイメントの場合は、Configuration Manager を使って IBM Content Manager のコネクターを設定することはできません。

- 1 IBM Content Manager のクライアントを指定画面で、「**Connector for IBM Content Manager を設定**」を選択し、「**IBM Content Manager クライアントのインストールディレクトリのパス**」を入力します。「**確認**」をクリックし、完了したら、「**次へ**」をクリックして次に進みます。
- 2 IBM Content Manager サーバーの設定を指定画面で、IBM Content Manager Server の詳細情報を入力し、「**次へ**」をクリックします。
- 3 Connector for IBM Content Manager を設定画面で「**IBM Content Manager Connector を設定**」をクリックします。完了したら、「**次へ**」をクリックします。

- 4 Connector for IBM Content Manager に必要な手動設定画面で、一覧の手動による手順を確認および実行し、「次へ」をクリックします。

Connector for IBM FileNet

注意：リモート LiveCycle デプロイメントの場合は、Configuration Manager を使って IBM FileNet のコネクタを設定することはできません。

- 1 IBM FileNet のクライアントを指定画面で、「**Connector for IBM FileNet Content Manager を設定**」を選択し、次の設定を指定します。
 - **IBM FileNet クライアントのバージョンを選択：**IBM FileNet Content Server で使用するクライアントバージョンを選択します。
 - **IBM FileNet クライアントのインストールディレクトリのパス：**「参照」をクリックしてディレクトリパスを選択します。

注意：IBM FileNet クライアントを含むディレクトリ名に、ハイフン (-)、下線 (_)、カンマ (,)、ドット (.) などの特殊文字がある場合は、IBM FileNet の検証に失敗する場合があります。「確認」をクリックし、完了したら、「次へ」をクリックして次に進みます。
- 2 IBM FileNet Content Server の設定を指定画面で、必要な詳細情報を入力し、「次へ」をクリックします。
- 3 IBM FileNet Process Engine のクライアントを指定画面で、必要な詳細情報を入力し、「確認」をクリックします。完了したら、「次へ」をクリックします。
- 4 IBM FileNet Process Engine サーバーの設定を指定画面で、必要な詳細情報を入力し、「次へ」をクリックします。
- 5 Connector for IBM FileNet を設定画面で、「**FileNet Connector を設定**」をクリックします。完了したら、「次へ」をクリックします。
- 6 Connector for IBM FileNet に必要な手動設定画面で、一覧の手動による手順を確認および実行し、「次へ」をクリックします。

Connector for Microsoft SharePoint

注意：リモート LiveCycle デプロイメントの場合は、Configuration Manager を使って Microsoft SharePoint のコネクタを設定することはできません。

Adobe LiveCycle ES4 Connector for Microsoft SharePoint を設定画面で、次のいずれかのタスクを実行します。

- 後で Microsoft Sharepoint を手動設定するには、「**Adobe LiveCycle ES4 Connector for Microsoft SharePoint を設定**」オプションの選択を解除し、「次へ」をクリックします。
- 「**Adobe LiveCycle ES4 Connector for Microsoft SharePoint を設定**」オプションを選択したままにします。必要な値を入力し、「SharePoint Connector を設定」をクリックします。完了したら、「次へ」をクリックします。

注意：Administration Console を使用して後で Connector for Microsoft SharePoint を設定する場合は、この手順をスキップできます。

ネイティブファイル変換のための LiveCycle サーバーの設定

- ❖ (PDF Generator のみ) PDF のネイティブ変換に必要な管理者のユーザー資格情報画面で、サーバーコンピューターの管理者権限を持つユーザーのユーザー名とパスワードを入力して、「**ユーザーを追加**」をクリックします。

注意：Windows Server 2008 の場合は、管理ユーザーを 1 人以上追加する必要があります。Windows Server 2008 では、追加するユーザーのユーザーアカウント制御 (UAC) を無効にする必要があります。UAC を無効にするには、**コントロールパネル/ユーザーアカウント/ユーザーアカウント制御の有効化または無効化**を順にクリックし、「ユーザーアカウント制御 (UAC) を使ってコンピューターの保護に役立たせる」の選択を解除し、「OK」をクリックします。変更を適用するには、コンピューターを再起動します。

PDF Generator の System Readiness Test

- ❖ **Document Services PDF Generator System Readiness Test** 画面で、「開始」をクリックして、システムが適切に PDF Generator を設定しているかを検証します。System Readiness Tool レポートを確認し、「次へ」をクリックします。LiveCycle がリモートマシンにデプロイされている場合は、System Readiness Test が失敗します。

Reader Extensions の設定

- ❖ Reader Extensions の秘密鍵証明書の設定画面で、モジュールサービスをアクティブにする Reader Extensions 秘密鍵証明書に関連付けられている以下の詳細を指定します。

注意：「Administration Console を使用して後から設定」を選択して、さしあたり、この手順をスキップすることもできます。デプロイメントを完了した後で、Administration Console を使用して Reader Extensions 秘密鍵証明書を設定できます (Administration Console にログインしたら、ホーム / 設定 / Trust Store の管理 / ローカル秘密鍵証明書ををクリックします)。

「設定」をクリックし、「次へ」をクリックします。

サマリー、および次の手順

- ❖ Configuration Manager のタスクの概要リストを確認し、適切なオプションを選択します。
 - 「次の手順を開始」を選択して、LiveCycle ユーザーと管理インターフェイスに関する情報を表示し、LiveCycle の起動と使用に関する手順を説明した html ページを開きます。
- 「完了」をクリックして Configuration Manager を終了します。

第 6 章：デプロイメント後のタスク

6.1 一般的なタスク

6.1.1 許可されているリファラーの設定

Configuration Manager を実行すると、デフォルトホスト、IPv4 アドレス、IPv6 アドレス、ループバックアドレス、およびローカルホストアドレスが許可されているリファラーリストに追加されます。これらのアドレスは、LCM が実行されるマシンに対してのみ追加されます。LiveCycle クラスターの場合は、その他のすべてのクラスターノードをリストに手動で追加します。

- 1 Administration Console で、**設定 / User Management / 設定 / 許可されているリファラー URL** を設定をクリックします。許可されているリファラーリストがページ下部に表示されます。
- 2 許可されているリファラーを追加するには
 - a ホストの名前または IP アドレスを「許可されているリファラー」ボックスに入力します。一度に複数の許可されているリファラーを追加するには、各ホストの名前または IP アドレスを 1 行ごとに入力します。すべてのクラスターノード / ロードバランスの `hostName` と IP アドレスを指定します。
 - b 「HTTP ポート」と「HTTPS ポート」ボックスについて、HTTP または HTTPS のいずれか、または両方のポートを指定します。これらのポートを空のままにした場合、デフォルトのポート（HTTP ではポート 80、HTTPS ではポート 443）が使用されます。ボックスに 0（ゼロ）を入力した場合、そのサーバー上のすべてのポートが有効化されます。特定のポート番号を入力すると、そのポートのみが有効化されます。
 - c 「追加」をクリックします。
- 3 「保存」をクリックします。

許可されているリファラーリストが空の場合、CSRF 機能は動作を停止し、システムのセキュリティが低下します。
- 4 許可されているリファラーリストを変更したら、LiveCycle クラスターを再起動してください。

6.1.2 CRX レポジトリクラスタリングの設定

注意： LiveCycle ES4 Service Pack 1 をインストールしていない場合は、次の手順をスキップして、[LiveCycle ES4 Service Pack 1 の LiveCycle ES4 への適用によって生じる主な相違点](#)の記事に記載されている CRX レポジトリクラスタリングの設定手順を実行します。

次の手順を実行して CRX レポジトリクラスタリングを設定します。

- 1 `http://[Host]:[Port]/lc/libs/granite/cluster/content/admin.html` に移動します。管理者としてログインします。
- 2 マスター URL を `http://[Master_host]:[port]/lc` に変更します。マスターとして機能するノードのホスト名とポートを指定します。
- 3 ユーザー名 / パスワードを `admin/admin` に変更して、「参加」をクリックします。設定を完了するには少し時間がかかる場合があります。更新や戻るを押さないでください。設定が完了すると、成功したというメッセージが表示されます。
- 4 さらに多くのスレーブノードを接続するには、各スレーブノードに対して手順 1 から 4 を繰り返します。各スレーブに対して同じマスター URL（手順 4 を参照）を指定してください。

注意：これらの手順をマスターノードに対しては実行しないでください。

クラスターを開始するときは、すべてのスレーブノードの前にマスターノードが開始されていることを確認してください。クラスターを停止するときは、マスターノードの前にすべてのスレーブを停止してください。ある特定のシナリオでは、Master ノードと Slave ノードは役割を交代できます。この場合、クラスターを停止する前にマスターを停止してください。

CRX クラスターリングでは特定の開始 / 停止クラスター順序が実施されますが、これは LiveCycle 内に埋め込まれているので、LiveCycle クラスターを開始および停止するときは、この手順に従ってください。

スレーブノードは、マスターノードが立ち上がって稼働するまで指定秒数の時間を待機します。この指定秒数以内にマスターノードが立ち上がらなかった場合は、スレーブノードはそのレポジトリを停止します。クラスター内にスレーブノードを参加させるには、そのスレーブノードを再起動します。デフォルトのノード待機時間は 60 秒です。スレーブノードに対する秒数を設定するには、次の JVM 引数を使用します。

```
-Dcom.day.crx.core.cluster.WaitForMasterRetries=<value>
```

クラスターのすべてのノードを一度に開始した場合、開始順序による依存関係が成立しなくなり、クラスターのスレーブノードが開始できなくなる場合があります。このような問題を回避するために、ノードの待機時間を 300 秒以上にしてください。

注意：古くなったセッションを回避するには、スレーブインスタンスを再起動します。

重要：クラスター内のすべての作成者インスタンスは、時間を同期させる必要があります。NTP (Network Time Protocol) サーバーを使用して時間を同期させることができます。

6.1.3 システムイメージバックアップの実行

実稼働環境に LiveCycle をインストールおよびデプロイした後、このシステムを稼働する前に、LiveCycle を実装したサーバーのシステムイメージバックアップを実行することをお勧めします。CRX レポジトリのバックアップもとってください。

このバックアップには、LiveCycle のデータベース、GDS ディレクトリおよびアプリケーションサーバーを含める必要があります。これは、ハードドライブまたはコンピューター全体が動作しなくなった場合に、コンピューターの内容の復元に使用できる完全なシステムバックアップです。[管理ヘルプ](#)の「LiveCycle のバックアップと回復」トピックを参照してください。

6.1.4 アプリケーションサーバーの再起動

LiveCycle を初めてデプロイする際、サーバーはデプロイメントモードになっています。このモードでは、ほとんどのモジュールがメモリ内に置かれます。このため、メモリの消費量が大きく、サーバーは実稼働に適した状態ではありません。アプリケーションサーバーを再起動して、サーバーをクリーンな状態に戻す必要があります。

注意：CRX クラスターリングの設定後または許可されるリファラーのリストを更新後にサーバーを再起動した場合は、LiveCycle サーバーの再起動を省略してもかまいません。

注意：LiveCycle Server をアップグレードするときまたはサービスパックをデプロイしたときは、アプリケーションサーバーを再起動する前に、[Jboss ルート]\server\<サーバー名>\work および [Jboss ルート]\server\<サーバー名>\tmp フォルダを削除してください。

6.1.5 デプロイメントの確認

Administration Console にログインして、デプロイメントを確認できます。正常にログインできる場合は、LiveCycle がアプリケーションサーバーで実行されており、データベースにデフォルトのユーザーが作成されています。CRX レポジトリデプロイメントを検証するには、CRX ようこそページにアクセスします。

アプリケーションサーバーのログファイルを確認して、コンポーネントが正しくデプロイされたことを確認したり、発生する可能性のあるデプロイメントの問題の原因を特定したりすることができます。

6.1.5.1 LiveCycle 管理コンソールへのアクセス

Administration Console は、LiveCycle の各種の設定ページにアクセスするための Web ベースのポータルです。これらの設定ページでは、LiveCycle ES3 の動作を制御する実行時プロパティを設定できます。Administration Console にログインすると、User Management、監視フォルダー、電子メールクライアント設定および他のサービスの管理設定オプションにアクセスできます。また、Administration Console では「アプリケーションおよびサービス」にアクセスすることもできます。これは、管理者がアーカイブの管理や、実稼働環境へのサービスのデプロイに使用します。

ログインする場合のデフォルトのユーザー名とパスワードは、それぞれ administrator と password です。初回のログイン後は、User Management にアクセスしてパスワードを変更してください。

Administration Console にアクセスするには、デプロイ済みの LiveCycle がアプリケーションサーバー上で実行されている必要があります。

Administration Console の使用方法については、[管理ヘルプ](#)を参照してください。

- 1 Web ブラウザーに次の URL を入力します。

```
http://[hostname]:[port]/adminui
```

例：http://localhost:8080/adminui

- 2 LiveCycle にアップグレードした場合、以前の LiveCycle インストールと同じ管理者ユーザー名およびパスワードを入力します。新規インストールの場合は、デフォルトのユーザー名とパスワードを入力します。
- 3 ログイン後、「サービス」をクリックして、サービスの管理ページにアクセスするか、「設定」をクリックして、様々なモジュールの設定を管理できるページにアクセスします。

6.1.5.2 LiveCycle 管理者のデフォルトパスワードの変更

LiveCycle では、インストール時に 1 つ以上のデフォルトのユーザーが作成されます。これらのユーザーのパスワードは製品資料に記載され、公開されています。セキュリティ要件に応じて、このデフォルトのパスワードを変更する必要があります。

LiveCycle 管理者のユーザーパスワードは、デフォルトで「password」に設定されています。Administration Console / 設定 / User Management でパスワードを変更してください。

CRX 管理者のデフォルトパスワードの変更もお勧めします。

詳細については、デフォルト管理者パスワードの変更を参照してください。

6.1.5.3 CQ ようこそページへのアクセス

CQ ようこそページは、さまざまな CQ コンポーネント、管理、デプロイメント、開発ツールにアクセスするための Web ベースのポータルです。ログイン用のデフォルトのユーザー名とパスワードは、administrator と password です (LiveCycle 管理者と同じです)。

次の手順を使用してようこそページにアクセスします。

- 1 Web ブラウザーに次の URL を入力します。

```
http://[ホスト名]:[ポート]/lc/welcome
```

- 2 上記の管理者ユーザー名とパスワードを入力します。
- 3 ログインすると、さまざまなコンポーネント、管理、デプロイメント、開発ユーザーインターフェイスにアクセスできます。

6.1.5.4 OSGi Management Console へのアクセス

CQ では、コンポーネントは OSGi バンドルの形式で、Apache Felix OSGi コンテナにデプロイされています。OSGi コンソールは、OSGi バンドルとサービス設定を管理するための手段を提供します。ログイン用のデフォルトのユーザー名とパスワードは、admin と ad,om です (CRX 管理者と同じです)。

次の手順お Windows 使用して OSGi 管理コンソールにアクセスします。

- 1 Web ブラウザーに次の URL を入力します。

`http://[ホスト名]:[ポート]/lc/system/console`

- 2 上記と同じ管理者ユーザー名とパスワードを入力します。
- 3 ログインすると、さまざまなコンポーネント、サービス、バンドル、その他の設定にアクセスできます。

6.1.5.5 CQ 管理者のデフォルトパスワードの変更

LiveCycle 内に埋め込まれている CQ は、前述のような 2 人の管理者ユーザーを持っています。

- **スーパー管理者 (administrator):** スーパー管理者ユーザーはさまざまな CQ/CRX ユーザーインターフェイスにアクセスでき、admin 操作を実行できます。デフォルトのユーザー名とパスワードは、LiveCycle 管理者と同じ **administrator/password** です。このユーザーは OSGi Management Console へのアクセスを持っていません。このユーザーのデフォルトパスワードは、「**デフォルト LiveCycle パスワードの変更**」の項で述べたように、LiveCycle 管理者コンソールを使用してのみ変更できます。変更されたパスワードは、LiveCycle と CQ の両方に適用されます。
- **管理者 (admin):** このユーザーは、CQ/CRX ユーザーインターフェイスのほかに OSGi コンソールにもアクセスでき、管理者特権を持っています。ユーザーのデフォルトのユーザー名とパスワードは、**admin/admin** です。デフォルトのパスワードを変更するには、以下の手順に従います。

- 1 Web ブラウザーに次の URL を入力します。

`http://[ホスト名]:[ポート]/lc/libs/granite/security/content/admin.html`

- 2 次の資格情報を使ってログインします。

ユーザー名: admin

パスワード: admin

- 3 ユーザー **Administrator** を検索します。
- 4 左メインでこのユーザーをクリックすると、ユーザーの詳細が右ペインに表示されます。
- 5 右ペインで**編集**アイコンをクリックします。
- 6 右ペインの編集ペインで、「**新しいパスワード**」フィールドに新しいパスワードを、「**パスワード**」フィールドに現在のパスワードを入力します。
- 7 右ペインで**保存**アイコンをクリックします。
- 8 変更したパスワードを使って再びログインし、検証します。

6.1.5.6 ログファイルの表示

実行時や起動時のエラーなどのイベントは、アプリケーションサーバーのログファイルに記録されます。アプリケーションサーバーへのデプロイ中に何らかの問題が発生した場合には、ログファイルを参照して問題を見つけることができます。ログファイルは、テキストエディターを使用して開くことができます。

手動で設定された JBoss の場合、ログファイルは次の場所にあります。

- (スタンドアロン JBoss) [appserver root]/server/standard/logs ディレクトリ
- (クラスター) [appserver root]/server/all/logs ディレクトリ

アドビの事前設定 JBoss の場合、ログファイルは次の場所にあります。

- (スタンドアロン) [appserver root]/server/lc_<dbname>/logs ディレクトリ
- (クラスター) [appserver root]/server/lc_<dbname>_cl/logs ディレクトリ

次のログファイルがあります。

- server.log
- boot.log

次の CRX ログファイルは [CRX_home]/ にあります。

- error.log
- audit.log
- access.log
- request.log
- update.log

6.2 LiveCycle クラスターの検証

- 1 クラスターのすべてのアプリケーションサーバーインスタンスが起動していることを確認します。
- 2 Gemfire.log ファイルを参照します。このファイルは、アプリケーションサーバーに応じて、次のディレクトリに配置されています。
 - Jboss: [lc_temp_dir]/adobejb_[idp_server_name]/caching

注意：idp_server_name は、JBoss インスタンスに渡される JVM 引数 -Dadobeidp.serverName の値です。

- 3 次のようなメッセージで、キャッシュがクラスターのすべてのサーバーに接続されていることを確認できます。

```
[info 2008/01/22 14:24:31.109 EST GemfireCacheAdapter <UDP mcast receiver> nid=0x5b611c24] Membership: received new view [server-0:2916|1] [server-0:2916/2913, server-1:3168/3165] [info 2008/01/22 14:24:31.125 EST GemfireCacheAdapter <View Message Processor> nid=0x7574d1dc] DMMembership: admitting member <server-1:3168/3165>; now there are 2 non-admin member(s)
```

注意：非管理メンバーの数 (上記のログエントリの例では 2) がクラスター内のメンバーの数と一致することを確認してください。一致しない場合は、クラスター内の一部のメンバーがキャッシュに接続されていないことを意味します。

6.3 CRX クラスターの検証

- 1 `http://<オーサーホスト>:<オーサーポート>/lc/system/console` に移動します。OSGi Management Console のユーザー資格情報を使ってログインします。デフォルトの資格情報は `admin/admin` です。
- 2 Main / JMX を選択し、ドメイン: `com.adobe.granite` の行を探し、「Repository」と入力します。
- 3 Repository をクリックし、Attribute Name: `ClusterNodes` を探します。

ClusterNodes 属性はテーブルを含みます。このテーブルの各行は、クラスター内のノードを表します。各行は、オペレーティングシステム、ホスト名、ID、およびクラスターノードの `repositoryHome` が含まれています。

すべてのスレーブノード上では `crx.cluster.master` の値は `false` であり、マスターノード上では `crx.cluster.master` の値は `true` です。

6.4 モジュールの Web アプリケーションへのアクセス

LiveCycle のデプロイ後は、次のモジュールに関連付けられた Web アプリケーションにアクセスできます。

- Reader Extensions
- Adobe® LiveCycle® Workspace 11
- HTML ワークスペース
- ユーザー管理
- コレスポネデンス管理
- PDF Generator Web アプリケーション
- Adobe® LiveCycle® PDF Generator 11
- Adobe® LiveCycle® Rights Management 11

デフォルトの管理者権限を使用して Web アプリケーションにアクセスし、そのアプリケーションにアクセス可能であることを確認したら、他のユーザーがログインしてアプリケーションを使用できるように追加のユーザーとロールを作成できます (管理ヘルプを参照)。

6.4.1 Reader Extensions Web アプリケーションへのアクセス

注意： Reader Extensions 秘密鍵証明書を適用して、新しいユーザーのユーザーロールを適用する必要があります (LiveCycle 管理ヘルプの「秘密鍵証明書を Reader Extensions で使用するための設定」を参照)。

- 1 Web ブラウザーを開き、次の URL を入力します。

`http://[hostname]:[port]/ReaderExtensions`

- 2 LiveCycle のユーザー名とパスワードを使用してログインします。

注意： ログインするには、管理者またはスーパーユーザーの権限が必要です。他のユーザーが Reader Extensions Web アプリケーションにアクセスできるようにするには、User Management でユーザーを作成し、そのユーザーに Reader Extensions Web アプリケーションロールを付与する必要があります。

6.4.2 Workspace へのアクセス

- 1 Web ブラウザーを開き、次の URL を入力します。

`http://[hostname]:[port]/workspace`

- 2 LiveCycle のユーザー名とパスワードを使用してログインします。

6.4.3 HTML ワークスペースへのアクセス

- 1 Web ブラウザーを開き、次の URL を入力します。

`http://[ホスト名]:[ポート]/lc/ws`

- 2 LiveCycle のユーザー名とパスワードを使用してログインします。

6.4.4 フォームマネージャーへのアクセス

- 1 Web ブラウザーを開き、次の URL を入力します。

`http://[ホスト名]:[ポート]/lc/fm`

- 2 LiveCycle のユーザー名とパスワードを使用してログインします。

6.4.5 PDF Generator Web アプリケーションへのアクセス

- 1 Web ブラウザーを開き、次の URL を入力します。

`http://[ホスト名]:[ポート]/pdfgui`

- 2 LiveCycle のユーザー名とパスワードを使用してログインします。

6.4.6 Rights Management へのアクセス

User Management で Rights Management End User ロールのユーザーを作成し、そのユーザーに関連付けられたログイン情報を使用して Rights Management の管理者またはエンドユーザーアプリケーションにログインする必要があります。

注意: デフォルトの管理者ユーザーは、Rights Management エンドユーザー Web アプリケーションにはアクセスできません。ただし、このユーザーのプロファイルに必要なロールを追加できます。新しいユーザーを作成したり、既存のユーザーを修正したりするには、Administration Console を使用します。

Rights Management エンドユーザー Web アプリケーションへのアクセス

- ❖ Web ブラウザーを開き、次の URL を入力します。

`http://[hostname]:[port]/edc`

Rights Management 管理 Web アプリケーションへのアクセス

- 1 Web ブラウザーを開き、次の URL を入力します。

`http://[hostname]:[port]/adminui`

- 2 サービス / LiveCycle Rights Management 11 をクリックします。

ユーザーおよびロールの設定について詳しくは、管理ヘルプを参照してください。

Rights Management End User ロールのアサイン

- 1 Administration Console にログインします (「35 ページの「6.1.5.1 LiveCycle 管理コンソールへのアクセス」」を参照)。
- 2 **設定 / User Management / ユーザーとグループ** をクリックします。
- 3 「キーワード」ボックスに all と入力し、**条件 2** リストで「**グループ**」を選択します。
- 4 「**検索**」をクリックし、該当するドメインについて、表示されるリストの「**すべてのプリンシパル**」をクリックします。
- 5 「**ロールアサイン**」タブをクリックし、「**ロールを検索**」をクリックします。
- 6 ロールのリストで、「**Rights Management End User**」の横にあるチェックボックスを選択します。
- 7 「**OK**」をクリックし、「**保存**」をクリックします。

6.4.7 User Management へのアクセス

User Management を使用すると、管理者は 1 つまたは複数のサードパーティユーザーディレクトリに同期するすべてのユーザーおよびグループのデータベースを管理できます。User Management には、Reader Extensions、Workspace、Rights Management、Adobe® LiveCycle® Process Management 11、Adobe® LiveCycle® Forms Standard 11、PDF Generator などの LiveCycle モジュールの認証、権限付与およびユーザー管理の機能があります。

- 1 Administration Console にログインします。
- 2 ホームページで、**設定 / User Management** をクリックします。

注意：User Management でのユーザー設定について詳しくは、User Management ページの右上隅にある「**User Management ヘルプ**」をクリックしてください。

6.4.8 Correspondence Management Solution テンプレートへのアクセス

http://[ホスト名]:[ポート]/lc/cm に行って LiveCycle 管理者資格情報を使ってログインすることで、Correspondence Management Solution デプロイメントを検証できます。ソリューションテンプレートは、Correspondence Management Solution の参照実装です。

注意：非自動のデプロイメントでは、ソリューションテンプレートにアクセスしたときにエラーが発生した場合は、LiveCycle と Correspondence Management Solution を統合する必要があります。詳細については、このドキュメントの「**パブリッシュノードを設定して LiveCycle と統合する**」の項を参照してください。

6.5 Correspondence Management Solution の設定

次のタスクを実行して Correspondence Management Solution を設定します。

Correspondence Management Solution の推奨設定については、Correspondence Management Solution トポロジーを参照してください。

6.5.1 作成者インスタンスの設定

作成者インスタンスは LiveCycle サーバー内に埋め込まれています。このことは、作成者インスタンスに対して設定アップデートをまったく行う必要がないことを意味しています。このインスタンスは、LiveCycle サーバーからすべての構成設定を継承します。

6.5.2 発行インスタンスの設定

Correspondence Management Solution では、個別の作成者インスタンスと発行インスタンスを実行する必要があります。ただし、この 2 つのインスタンスは、同じマシンに設定することも、それぞれ別のマシンに設定することもできます。作成者インスタンスは、LiveCycle コアアプリケーション内に埋め込まれており、LiveCycle サーバー上で実行します。発行インスタンスの場合は、LCM が発行者 EAR (adobe-livecycle-cq-publish.ear) を設定します。別のサーバーインスタンスに発行 EAR をデプロイします。

注意：発行インスタンスを設定する前に、作成者インスタンスが設定およびデプロイ済みであることを確認します。

Correspondence Management Solution 用のソリューションテンプレートにログインできれば、これを確認できたこととなります。詳細については、このドキュメントの「**Correspondence Management Solution テンプレートへのアクセス**」の項を参照してください。

- 1 発行インスタンスのために新しいサーバーを作成します。JBoss インストールを実行し、標準プロファイルを使用します。
- 2 作成者インスタンスで、**[LiveCycle root]/configurationManager/export/** ディレクトリに移動します。
- 3 **[LiveCycle ルート]/configurationManager/export/crx-repository** ディレクトリを発行インスタンスマシンにコピーします。
- 4 発行インスタンスマシン上の **crx-repository** ディレクトリの場所に行きます。crx-repository/install フォルダーを開きます。次のパッケージを残して、その他すべてのパッケージをインストールフォルダーから削除します。
 - dataservices-pkg.zip
 - platform-common-pkg.zip

- platform-content-pkg.zip
- platform-security-pkg.zip
- solution-correspondencemanagement-pkg.zip

5 -Dcom.adobe.livecycle.crx.home=<location for crx-repository> パラメーターを使用して、発行サーバーを起動します。ここで、<location for crx-repository> は発行インスタンス用の crx-repository ディレクトリのコピー元の場所です。

WebSphere および WebLogic のための汎用 JVM 引数を設定する方法については、[WebSphere](#) および [WebLogic](#) の、[JVM 引数の設定](#)の項を参照してください。

6 adobe-livecycle-cq-publish.ear ファイルをコピーし、手順 1 で作成したアプリケーションサーバーのプロファイルにデプロイします。

注意：同じコンピューター上に作成者インスタンスと発行インスタンスが両方ある場合には、発行インスタンスを起動する際に必ず別のポートを使用するようにしてください。

発行インスタンスを起動して実行したら、2つのインスタンスが互いに通信できるように設定する必要があります。

6.5.3 発行ノードの「LiveCycle に統合」への設定

すべての発行インスタンスに対して次の手順を実行します。発行インスタンスと LiveCycle サーバーとの間の通信を有効にするには、次の操作を実行します。

1 `http://[発行ホスト]:[発行ポート]/lc/system/console/configMgr` に行き、OSGi Management Console ユーザー資格情報を使ってログインします。デフォルトの資格情報は `admin/admin` です。

2 「**Adobe LiveCycle Client SDK Configuration**」設定の横にある「**編集**」アイコンをクリックします。

3 サーバー URL フィールドで、`http://[lc ホスト]:[lc ポート]` が指定されていることを確認します。

重要：LiveCycle サーバーが、指定されたホストとポートの組み合わせでリスンしていることを確認してください。LiveCycle サーバークラスターの場合、次の3つのシナリオが可能です。

- すべての LiveCycle サーバーインスタンスが **localhost** と同じポート上で実行している。この場合は、**localhost:[ポート]** を使用します。
- すべての LiveCycle サーバーインスタンスが **localhost** の異なるポート上で実行している。この場合は、ロードバランサーホスト名とポートの組み合わせ、すなわち `[ロードバランサーホスト]:[ロードバランサーポート]` を使用します。
- すべての LiveCycle サーバーインスタンスが特定のホスト名 (**localhost** ではない) と異なる / 同じポート上で実行している。この場合は、ロードバランサーホスト名とポート、すなわち `[ロードバランサーホスト]:[ロードバランサーポート]` を使用します。

LiveCycle サーバークラスターにアクセスするためにロードバランサー URL を使用する必要がある場合 (上記参照)、作成者インスタンスとロードバランサー間の必要な通信ポートが開いていることを確認してください。

4 LiveCycle の管理者資格情報を、「Username」フィールドと「Password」フィールドにそれぞれ入力します。

5 「**Save**」をクリックします。

6.5.4 作成者インスタンスと発行インスタンス間の通信

作成者インスタンスと発行インスタンス間で双方向通信を有効にするには、いくつかの設定変更を行う必要があります。

6.5.4.1 複製エージェントの設定 (発行インスタンス URL の定義)

作成者インスタンスで、各発行インスタンスごとに複製エージェントを設定する必要があります。これらのエージェントは作成者インスタンスのコンテンツをすべての発行インスタンスに複製します。

- 1 `http://<authorHost>:<authorPort>/lc/miscadmin` で Tools UI にログインします。
- 2 「複製」を選択してから、左パネルで「作成者のエージェント」を選択します。
右パネルには、作成者インスタンスのために設定されたさまざまなエージェントがあります
- 3 右パネルで、「新規...」を選択し、「新規ページ」をクリックします。
ページの作成ダイアログが表示されます。
- 4 タイトルと名前を設定し、複製エージェントを選択します。
- 5 「作成」をクリックして、新しいエージェントを作成します。
- 6 新しいエージェントをダブルクリックして設定パネルを開きます。
- 7 「編集」をクリックすると、エージェント設定ダイアログが表示されます。
 - a 設定タブで次の操作をします。
 - 説明を入力します。
 - 「有効」にチェックを付けます。
 - 「シリアライゼーションタイプをデフォルトにする」を選択します。
 - 「試行遅延」を「60000」に設定します。
 - 「ログレベル」を「Info」として設定します。
 - b トランスポートタブで次の操作をします。
 - 発行インスタンスの必要 URI `http://<発行ホスト>:<発行ポート>/lc/bin/receive?sling:authRequestLogin=1` を入力します。
 - ユーザーとパスワードを設定します。デフォルトの資格情報は `admin/admin` です。
- 8 「OK」をクリックして設定を保存します。
- 9 エージェント設定パネルで、「接続のテスト」をクリックします。
接続に成功すると、設定が正しく行われたことがわかります。

注意：場合によっては、発行インスタンスを1つだけ持っている場合は、デフォルトの複製エージェントを `publish` という名前を付けて使用できます。手順 b(i) で説明したように、トランスポートタブでそれを編集して、発行 URI を指定してください。この場合は、新しい複製エージェントを作成する必要はありません。

注意：場合によっては、発行ファーム (複数の非クラスター発行インスタンス) を持っている場合は、手順 1 から 9 で説明されているように、各発行インスタンスごとに複製エージェントを作成する必要があります。これらの各複製エージェントに対して、タイトルと名前は重要で一意でなければならず、対応する発行インスタンスの識別を簡単にできるようにする必要があります。これらの各複製エージェントは、特定の発行インスタンスを示す異なる URI をトランスポートタブに持っています。複数の発行インスタンスの場合は、デフォルトのエージェント `publish` をコピーし、作成したエージェントのトランスポートタブで名前と URI を編集することで、複製エージェントを作成することもできます。デフォルトの複製エージェントを使用しない場合は、それを無効にして、不必要な複製が行われないようにできます。

注意：別のクラスターに対しては、1つの作成者インスタンス (できればマスターインスタンス) でこれらの手順を実行する必要があります。

6.5.4.2 ActivationManagerImpl の発行インスタンス URL の定義

- 1 `http://<authorHost>:<authorPort>/lc/system/console/configMgr` に移動します。OSGi Management Console のユーザー資格情報を使ってログインします。デフォルトの資格情報は `admin/admin` です。
- 2 「`com.adobe.livecycle.content.activate.impl.ActivationManagerImpl.name`」設定の横にある「編集」アイコンをクリックします。
- 3 「ActivationManager Publish URL」フィールドで、発行インスタンス ActivationManager にアクセスするための URL を指定します。次の URL を指定できます。
 - a **ロードバランサー URL (推奨)**: 発行ファーム (複数の非クラスター発行インスタンス) の前にロードバランサーとして機能する Web サーバーを持っている場合は、そのロードバランサーの URL を指定します。
 - b **発行インスタンス URL**: 単一の発行インスタンスのみを持っている場合、あるいは発行ファーム前段の Web サーバーが何らかの理由で作成者完了からアクセスできない場合、任意の発行インスタンス URL を指定します。指定した発行インスタンスがダウンした場合は、フォールバックメカニズムが機能して作成者側で処理します。

URL 設定: `http://<ホスト名>:<ポート>/lc/bin/remoting/lc.content.remote.activate.activationManager`
- 4 「保存」をクリックします。

6.5.4.3 逆複製キューの設定

作成者インスタンスで、各発行インスタンスごとに逆複製エージェントを設定する必要があります。これらのエージェントは発行インスタンスのコンテンツを作成者インスタンスに複製します。

- 1 `http://<authorHost>:<authorPort>/lc/miscadmin` で Tools UI にログインします。
- 2 「複製」を選択してから、左パネルで「作成者のエージェント」を選択します。

右パネルには、作成者インスタンスのために設定されたさまざまなエージェントがあります
- 3 右パネルで、「新規」を選択してから、「新規ページ」をクリックします。

ページの作成ダイアログが表示されます。
- 4 **タイトル** と **名前** を設定し、**逆複製エージェント** を選択します。
- 5 「作成」をクリックして、新しいエージェントを作成します。
- 6 新しいエージェントをダブルクリックして設定パネルを開きます。
- 7 「編集」をクリックすると、**エージェント設定**ダイアログが表示されます。
 - a **設定**タブで次の操作をします。
 - 説明を入力します。
 - 「有効」にチェックを付けます。
 - 「試行遅延」を「60000」に設定します。
 - 「ログレベル」を「Info」として設定します。
 - b **トランスポート**タブで次の操作をします。
 - 発行インスタンスの必要 URI を入力します - `http://<発行ホスト>:<発行ポート>/lc/bin/receive?slingsling:authRequestLogin=1`
 - ユーザーとパスワードを設定します - `admin/admin`
 - c **拡張**タブで: HTTP メソッドを GET として設定します
- 8 「OK」をクリックして設定を保存します。
- 9 エージェント設定パネルで、「接続のテスト」をクリックします。

接続に成功すると、設定が正しく行われたことがわかります。

注意: 場合によっては、発行インスタンスを 1 つだけ持っている場合は、デフォルトの逆複製エージェントを **publish_reverse** という名前を付けて使用できます。手順 b(i) で説明したように、**トランスポート** タブでそれを編集して、発行 URI を指定してください。この場合は、新しい逆複製エージェントを作成する必要はありません。

注意: 場合によっては、発行ファーム (複数の非クラスター発行インスタンス) を持っている場合は、手順 1 から 9 で説明されているように、各発行インスタンスごとに逆複製エージェントを作成する必要があります。これらの各複製エージェントに対して、**タイトル** と名前は重要で一意でなければならず、対応する発行インスタンスの識別を簡単にできるようにする必要があります。これらの各複製エージェントは、特定の発行インスタンスを示す異なる URI を **トランスポート** タブに持っています。複数の発行エージェントの場合は、デフォルトのエージェント **publish_reverse** をコピーし、作成したエージェントの **トランスポート** タブで **名前** と **URI** を編集することで、逆複製エージェントを作成することもできます。デフォルトの逆複製エージェントを使用しない場合は、それを無効にして、不必要な複製が行われないようにできます。

注意: 別のクラスターに対しては、1 つの作成者インスタンス (できればマスターインスタンス) でこれらの手順を実行する必要があります。

6.5.4.4 VersionRestoreManagerImpl の作成者インスタンス URL の定義

1 `http://<publishHost>:<publishPort>/lc/system/console/configMgr` に移動します。OSGi Management Console のユーザー資格情報を使ってログインします。デフォルトの資格情報は `admin/admin` です。

2 「com.adobe.livecycle.content.activate.impl.VersionRestoreManagerImpl.name」設定の横にある「編集」アイコンをクリックします。

3 「VersionRestoreManager Author URL」フィールドで、作成者インスタンス VersionRestoreManager の URL を指定します。

URL string: `http://<ホスト名>:<ポート>/lc/bin/remoting/lc.content.remote.activate.versionRestoreManager`

注意: ロードバランサーの前に複数の作成者インスタンス (クラスター化) がある場合は、「VersionRestoreManager Author URL」フィールドにその URL を指定します。

4 「Save」をクリックします。

6.5.5 サンプルのユーザーとアセットのインストール

ユーザー権限が事前に定義されたサンプルユーザーをインストールして、ソリューションテンプレートを検索し、独自のソリューションを構築するようカスタマイズすることができます。

1 `http://<作成者ホスト>:<作成者ポート>/lc/crx/explorer/index.jsp` に行きます。

2 LiveCycle 管理者資格情報を使ってログインし、**Package Manager** をクリックします。

3 **Package Manager** で、`samples-correspondencemanagement-pkg-<バージョン>.zip` パッケージを `<LC ホーム>/deploy/crx` からアップロードします。

4 パッケージのアップロードに成功したら、「インストール」をクリックします。

5 確認ダイアログで「インストール」をクリックし、サンプルユーザーとアセットをインストールします。

Correspondence Management サンプルユーザー

Correspondence Management Solution Accelerator には、次のサンプルユーザーが含まれています。これらのユーザーは、対話型カスタマー通信の生成に導かれるアクティビティに参加することを期待されています。

パッケージのインストール時に、次の役割がユーザーに自動的に割り当てられます。

ユーザー名	割り当てられた役割	役割
Todd Goldman	Correspondence Management 管理者	このユーザーは、システム全般の管理者です。このロールを持つユーザーは、すべてのアセットを変更できます。また、カテゴリを定義することもできます。
Heather Douglas	Correspondence Management SME	この人物は、テキストと画像を CRUD できる役割を持っています。
Caleb Lopez	Correspondence Management アプリケーションスペシャリスト	このユーザーは、テキスト、写真、条件、リストの各オブジェクトを慎重に選択してレターテンプレートを定義します。この役割では、ユーザーはレターテンプレート、レイアウト、リスト、条件、テキスト、および画像を CRUD できます。
Gloria Rios	Correspondence Management 要求処理担当者	エージェントユーザーは、ビジネスユーザーによって定義されたレターテンプレートを使用して、カスタマーに配信するレター通信を生成します。
Jocelyn Robinson	Correspondence Management フォーム開発者	このユーザーには、LiveCycle Designer を使用したフォームレイアウトのデザインスキルがあります。このユーザーは、レスポンス管理で使用するためのフォームレイアウトをデザインする必要なノウハウを持っており、LiveCycle Designer を使用して、XDP テンプレートをデザインします。これはレターの雛形として機能します。
Frank Kricfalusi	Correspondence Management 開発者	このユーザーには、XSD スキーマおよびデータモデリングの概念についての知識があり、データディクショナリの作成および管理を行う役割があります。

ソリューションテンプレートを使用してソリューションを実装する場合のサンプルユーザーとガイドラインについて詳しくは、『[Correspondence Management Solution ガイド](#)』を参照してください。

注意: 別のクラスターに対しては、1 つの作成者インスタンス (できればマスターインスタンス) でこれらの手順を実行する必要があります。

6.5.6 IPv6 実装の設定

注意: この手順は、IPv6 アドレスを使用するコンピューター上で Correspondence Management Solution が実行されている場合のみ実行します。

IPv6 アドレスをサーバーおよびクライアントコンピューターにマップするには:

- 1 C:\Windows\System32\drivers\etc ディレクトリを開きます。
- 2 hosts ファイルをテキストエディターで開きます。
- 3 IPv6 アドレスのマッピングをホスト名に追加します。次に例を示します。

```
2001:1890:110b:712b:d1d:9c99:37ef:7281 <ipv6_hostname>
```

- 4 ファイルを保存して閉じます。

Correspondence Management Solution へのアクセスに IPv6 アドレスではなくマップされたホスト名が使用されていることを確認します。

6.5.7 Adobe Reader 用日本語フォントのインストール

Correspondence Management のアセットで日本語フォントを使用する場合は、Adobe Reader 用日本語サポートパッケージをインストールする必要があります。インストールしないと、文字やフォームのレンダリングおよび機能が正常に実行されません。言語パックをインストールするには、Adobe Reader のダウンロードページにアクセスします。

6.6 PDF Generator の設定

PDF Generator を LiveCycle の一部としてインストールしている場合は、次のタスクを実行します。

6.6.1 環境変数

PDF Generator モジュールをインストールして、ファイルを PDF に変換するように設定した場合、一部のファイル形式については、環境変数を手動で設定して、対応するアプリケーションの起動に使用する実行ファイルの絶対パスを含める必要があります。次の表に、インストールされたネイティブアプリケーション用の環境変数の一覧を示します。

注意：クラスター内のすべてのノードに、必要なアプリケーションがインストールされていることを確認してください。

注意：すべての環境変数とそれぞれのパスでは、大文字と小文字が区別されます。

アプリケーション	環境変数	例
Adobe Acrobat	Acrobat_PATH	C:\Program Files (x86)\Adobe\Acrobat 11.0\Acrobat\Acrobat.exe
Adobe FrameMaker®	FrameMaker_PATH	C:\Program Files (x86)\Adobe\FrameMaker8.0\FrameMaker.exe
メモ帳	Notepad_PATH	C:\WINDOWS\notepad.exe Notepad_PATH 変数は空欄でかまいません。
OpenOffice	OpenOffice_PATH	C:\Program Files (x86)\OpenOffice.org 3.3
Adobe PageMaker®	PageMaker_PATH	C:\Program Files (x86)\Adobe\PageMaker 7.0.2\PageMaker.exe
WordPerfect	WordPerfect_PATH	C:\Program Files (x86)\WordPerfect Office 12\Programs\wpwin12.exe
Adobe Photoshop®	Photoshop_PATH	C:\Program Files (x86)\Adobe\Adobe Photoshop CS4\Photoshop.exe

注意：これらの環境変数は、クラスター内のすべてのノードに対して設定する必要があります。

注意：環境変数 OpenOffice_PATH は、実行ファイルではなく、インストールフォルダーのパスに設定します。

Word、PowerPoint、Excel、Visio、Project などの Microsoft Office アプリケーションまたは AutoCAD のパスを設定する必要はありません。これらのアプリケーションがサーバーにインストールされている場合は、Generate PDF サービスが自動的にこれらのアプリケーションを起動します。

新しい Windows 環境変数の作成

- 1 スタート/コントロールパネル/システムを選択します。
- 2 「詳細設定」タブをクリックして、「環境変数」をクリックします。
- 3 「システム環境変数」セクションで、「新規」をクリックします。
- 4 設定が必要な環境変数の名前（例えば、Photoshop_PATH など）を入力します。このフォルダーは、実行ファイルを含むフォルダーです。例えば、次のパスを入力します。

D:\Program Files\Adobe\Adobe Photoshop CS4\Photoshop.exe

Linux または UNIX での PATH 変数の設定 (OpenOffice のみ)

次のコマンドを実行します。

```
export OpenOffice_PATH=/opt/openoffice.org3.3
```

6.6.2 HTTP プロキシサーバーを使用するようにアプリケーションサーバーを設定

LiveCycle が実行されているコンピューターが、プロキシ設定を使用して外部 Web サイトにアクセスしている場合、アプリケーションサーバーは、次の値を Java 仮想マシン (JVM) 引数として設定して起動する必要があります。

```
-Dhttp.proxyHost=[server host]  
-Dhttp.proxyPort=[server port]
```

アプリケーションサーバーを HTTP プロキシホスト設定で起動するには、次の手順を完了します。

1 コマンドラインから、[appserver root]/bin/ ディレクトリ内の run スクリプトを編集します。

- (Windows)
 - run.conf.bat
- (Linux, UNIX)
 - run.conf

2 次のテキストをスクリプトファイルに追加します。

```
Set JAVA_OPTS=%JAVA_OPTS%  
-Dhttp.proxyHost=[server host]  
-Dhttp.proxyPort=[server port]
```

3 ファイルを保存して閉じます。

6.6.3 Adobe PDF プリンターをデフォルトのプリンターとして設定

Adobe PDF プリンターを、サーバーのデフォルトプリンターに設定する必要があります。Adobe PDF プリンターがデフォルトとして設定されていない場合、PDF Generator ではファイルを変換できません。

クラスターの場合、Adobe PDF プリンターを、すべてのノードのデフォルトプリンターに設定する必要があります。

デフォルトプリンターの設定

1 スタート/プリンターと FAX を選択します。

2 プリンターと FAX ウィンドウで、「Adobe PDF」を右クリックし、「通常使うプリンターに設定」を選択します。

6.6.4 Acrobat Professional の設定 (Windows ベースのコンピューターのみ)

注意：この手順は、LiveCycle のインストールを完了後に Acrobat へのアップグレードまたは Acrobat のインストールを行った場合にのみ必要です。Acrobat のアップグレードは、Configuration Manager を実行してアプリケーションサーバーに LiveCycle をデプロイした後に実行できます。Acrobat Professional のルートディレクトリは、[Acrobatroot] と表記します。通常、ルートディレクトリは C:\Program Files\Adobe\Acrobat 11.0\Acrobat です。

PDF Generator で使用するための Acrobat の設定

- 1 Acrobat の以前のバージョンがインストールされている場合、Windows コントロールパネルの「プログラムの追加と削除」を使用して Acrobat をアンインストールします。
- 2 インストーラーを実行して Acrobat XI Pro をインストールします。
- 3 LiveCycle インストールメディアの additional\scripts フォルダーに移動します。

4 次のバッチファイルを実行します。

```
Acrobat_for_PDFG_Configuration.bat [LiveCycle root]/pdfg_config
```

注意：クラスターでは、LiveCycle がインストールされているクラスターノード上でコマンドを実行する必要があります。

5 LiveCycle Configuration Manager を実行しない他のクラスターノード上で、次の手順を実行します。

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print に、SplWOW64TimeOut という名前の新しいレジストリ DWORD エントリを追加します。値を 60000 に設定します。
- LiveCycle がインストールされているノード上の [LiveCycle ルート]/plugins/x86_win32 ディレクトリにある PDFGen.api を、現在設定しているノード上の [Acrobat ルート]/plug_ins ディレクトリにコピーします。

6 Acrobat を開き、ヘルプ／アップデートの有無をチェック／環境設定を選択します。

7 「自動的に新しいアップデートを確認する」を選択解除します。

Acrobat のインストールの検証

1 システム上の PDF ファイルに移動し、そのファイルをダブルクリックして Acrobat で開きます。PDF ファイルが正常に開いた場合は、Acrobat が正しくインストールされています。

2 PDF ファイルを正しく開くことができない場合は、Acrobat をアンインストールしてから再インストールします。

注意：Acrobat のインストール完了後に表示される Acrobat のすべてのダイアログボックスを閉じてから、Acrobat の自動アップデートを無効化してください。環境変数 Acrobat_PATH を、Acrobat.exe を指すように設定してください（例えば、C:\Program Files\Adobe\Acrobat 11.0\Acrobat\Acrobat.exe）。

ネイティブアプリケーションサポートの設定

1 前の手順で説明したように、Acrobat をインストールして検証します。

2 Adobe PDF プリンターをデフォルトのプリンターとして設定します。

Acrobat の信頼できるディレクトリリストへの一時ディレクトリの追加

OptimizePDF サービスでは、Adobe Acrobat を使用し、LiveCycle の一時ディレクトリおよび PDF Generator t の一時ディレクトリを Acrobat の信頼できるディレクトリリストに作成します。

LiveCycle の一時ディレクトリおよび PDF Generator の一時ディレクトリが信頼できるディレクトリリストに追加されない場合、OptimizePDF サービスの実行は失敗します。一時ディレクトリリストにディレクトリを追加するには、次の手順を実行します。

1 Acrobat を開き、編集／環境設定を選択します。

2 左側のカテゴリから、「セキュリティ（強化）」を選択し、「拡張セキュリティを有効にする」オプションを選択します。

3 LiveCycle の一時ディレクトリおよび PDF Generator の一時ディレクトリを信頼できるディレクトリリストに追加するには、「フォルダーパスの追加」をクリックし、ディレクトリを選択して「OK」をクリックします。

6.6.5 マルチスレッドファイル変換のユーザーアカウントの設定

デフォルトでは、PDF Generator は、一度に 1 つの OpenOffice、Microsoft Word または PowerPoint ドキュメントのみを変換できます。マルチスレッド変換を有効にすると、OpenOffice または PDFMaker の複数のインスタンスを起動して PDF Generator で同時に複数のドキュメントを変換できます（PDFMaker は、Word 文書と PowerPoint ドキュメントの変換に使用されます）。

注意：マルチスレッドファイル変換は、Microsoft Word 2007 および Microsoft PowerPoint 2007 のみでサポートされています。Microsoft Excel 2003 および Microsoft Excel 2007 ではサポートされていません。

マルチスレッドファイル変換を有効にする必要がある場合は、[LiveCycle のドキュメント](#)から入手可能な『インストールの準備』または『アップグレードの準備』の「マルチスレッドファイル変換の有効化」の節で説明されているタスクを実行する必要があります。

Linux および Solaris ユーザーの場合、ユーザーを作成して、パスワードプロンプトが表示されないようにシステムを設定する必要があります。次の項では、ユーザーを作成し、追加の設定を行う方法の概要について説明します。

6.6.5.1 ユーザーアカウントの追加

1 管理コンソールで、サービス / **LiveCycle PDF Generator 11** / **ユーザーアカウント** をクリックします。

2 「追加」をクリックし、LiveCycle サーバー上での管理者権限を持つユーザーのユーザー名とパスワードを入力します。OpenOffice のユーザーを設定する場合は、最初に表示される OpenOffice のアクティベート用のダイアログを閉じます。

注意： OpenOffice のユーザーを設定する場合、OpenOffice のインスタンス数を、この手順で指定したユーザーアカウント数よりも大きくすることはできません。

3 LiveCycle サーバーを再起動します。

注意： 追加したユーザーアカウントがクラスターのすべてのノードで定義されていることを確認してください。

6.6.5.2 Linux または Solaris での OpenOffice に必要な追加設定

1 上記の説明に従って、ユーザーアカウントを追加します。

2 `/etc/sudoers` ファイルで、追加のユーザー (LiveCycle サーバーを実行する管理者以外) のエントリを追加します。例えば、ユーザーを `lcamd`、サーバーを `myhost` として LiveCycle を実行している場合、`user1` および `user2` として動作させるには、`/etc/sudoers` に次のエントリを追加します。

```
lcamd myhost=(user1) NOPASSWD: ALL
lcamd myhost=(user2) NOPASSWD: ALL
```

この設定により、`lcamd` は、ホスト `myhost` において `user1` または `user2` として、パスワードの入力を求められることなくすべてのコマンドを実行できるようになります。

3 ユーザーアカウントの追加で追加したすべてのユーザーが LiveCycle サーバーに接続できるようになります。例えば、`user1` というローカルユーザーに LiveCycle サーバーに接続する権限を許可するには、次のコマンドを使用します。

```
xhost +local:user1@
```

詳しくは、`xhost` コマンドのドキュメントを参照してください。

4 `/etc/sudoers` ファイルで `requiretty` を有効にします。

5 サーバーを再起動します。

注意： アプリケーションサーバーを起動するセッションがオープンになっていることを確認してください。セッションを終了すると、一部の変換は間歇的に失敗する場合があります。

6.6.6 PDF Generator へのフォントの追加

LiveCycle にはフォントの中央リポジトリがあり、すべての LiveCycle モジュールがアクセスできます。サーバー上の LiveCycle 以外のアプリケーションで、追加フォントを使用できるように設定します。これにより、PDF Generator では、そのアプリケーションを使用して作成された PDF ドキュメントで追加フォントを使用できるようになります。

注意： 指定したフォントフォルダーに新しいフォントを追加したら、アプリケーションサーバーを再起動します。

6.6.6.1 LiveCycle 以外のアプリケーション

次のリストには、PDF Generator でサーバー側の PDF 生成に使用できる LiveCycle 以外のアプリケーションが含まれています。

Windows 専用アプリケーション

- Microsoft Office Word
- Microsoft Office Excel
- Microsoft Office PowerPoint
- Microsoft Office Project
- Microsoft Office Visio
- Microsoft Office Publisher
- AutoDesk AutoCAD
- Corel WordPerfect
- Adobe Photoshop CS
- Adobe FrameMaker
- Adobe PageMaker
- Adobe Acrobat Professional

マルチプラットフォームアプリケーション

- OpenOffice Writer
- OpenOffice Calc
- OpenOffice Draw
- OpenOffice Impress

注意：これらのアプリケーションの他にも、各ユーザーが追加したアプリケーションが含まれている場合があります。

上記のアプリケーションのうち OpenOffice スイート (Writer、Calc、Draw および Impress) は、他のアプリケーションが Windows にのみ対応しているのに対して、Windows、Solaris および Linux プラットフォームに対応しています。

6.6.6.2 Windows 専用アプリケーションへの新しいフォントの追加

上記のすべての Windows 専用アプリケーションでは、C:\Windows\Fonts (または同等の) フォルダにあるすべてのフォントにアクセスできます。これらのアプリケーションには、C:\Windows\Fonts に加えて、それぞれ固有のフォントフォルダが存在する場合があります。

このため、LiveCycle フォントディレクトリにカスタムフォントを追加する場合、C:\Windows\Fonts (または同等の) フォルダにそのフォントをコピーして、Windows 専用のアプリケーションでもこれらのフォントを使用できるようにする必要があります。

カスタムフォントの使用に際しては、使用許諾契約に基づくライセンスを取得して、そのフォントにアクセスするアプリケーションでの使用が許可されている必要があります。

6.6.6.3 その他のアプリケーションへの新しいフォントの追加

他のアプリケーションに PDF 作成のサポートを追加した場合、これらのアプリケーションのヘルプを参照して新しいフォントを追加します。Windows では、通常はカスタムフォントを C:\Windows\Fonts (または同等の) フォルダに追加すれば十分です。

6.6.7 HTML から PDF への変換の設定

HTML から PDF への変換プロセスは、Acrobat XI Pro の設定を使用するように設計されています。この設定は、PDF Generator の設定よりも優先されます。

注意：この設定は、HTML から PDF への変換プロセスを有効にするために必要です。設定が行われていない場合、この変換タイプは失敗します。

6.6.7.1 HTML から PDF への変換の設定

- 1 Acrobat のインストールおよび検証は、47 ページの「[6.6.4 Acrobat Professional の設定 \(Windows ベースのコンピューターのみ\)](#)」で説明されています。
- 2 `[LiveCycle root]¥plugins¥x86_win32` ディレクトリにある `pdfgen.api` ファイルを探し、`[Acrobat root]¥Acrobat¥plug_ins` ディレクトリにコピーします。

6.6.7.2 HTML から PDF への変換における Unicode フォントのサポート

重要：入力用 zip ファイルにファイル名が 2 バイト文字の HTML ファイルが含まれている場合、HTML から PDF への変換は失敗します。この問題を回避するには、HTML ファイルに名前を付けるときに 2 バイト文字を使用しないようにします。

- 1 Unicode フォントを、使用しているシステムに応じて、次のいずれかのディレクトリにコピーします。

- Windows

`[Windows root]¥Windows¥fonts`

`[Windows root]¥WINNT¥fonts`

- UNIX

`/usr/lib/X11/fonts/TrueType`

`/usr/openwin/lib/X11/fonts/TrueType`

`/usr/share/fonts/default/TrueType`

`/usr/X11R6/lib/X11/fonts/ttf`

`/usr/X11R6/lib/X11/fonts/truetype`

`/usr/X11R6/lib/X11/fonts/TrueType`

`/usr/X11R6/lib/X11/fonts/TTF`

`/Users/cfqauser/Library/Fonts`

`/System/Library/Fonts`

`/Library/Fonts`

`/Users/ + System.getProperty(<user name>, root) + /Library/Fonts`

`System.getProperty(JAVA_HOME) + /lib/fonts`

`/usr/share/fonts (Solaris)`

注意：`/usr/lib/X11/fonts` ディレクトリが存在することを確認します。ディレクトリがない場合は、`ln` コマンドを使用して `/usr/share/X11/fonts` から `/usr/lib/X11/fonts` へのシンボリックリンクを作成します。

- 2 `[LiveCycle root]/deploy/adobe-generatepdf-dsc.jar` ファイルにある `cffont.properties` ファイルで、フォント名マッピングを変更します。
 - このアーカイブを展開し、`cffont.properties` ファイルを探して、エディターで開きます。

デプロイメント後のタスク


- Java フォント名のコンマ区切りリストで、フォントタイプごとに、Unicode システムフォントにマップを追加します。以下の例では、kochi mincho が Unicode システムフォントの名前です。

```
dialog=Arial, Helvetica, kochi mincho
```

```
dialog.bold=Arial Bold, Helvetica-Bold, kochi mincho ...
```

- プロパティファイルを保存して閉じ、adobe-generatepdf-dsc.jar ファイルを再パッケージ化して再デプロイします。

注意：日本語のオペレーティングシステムでは、`cffont.properties` ファイルでもフォントマッピングを指定します。これは、標準の `cffont.properties` ファイルよりも優先されます。

 リスト内のフォントは、左から右に検索され、最初に見つかったフォントが使用されます。HTML から PDF の変換ログでは、システム内で見つかったすべてのフォント名のリストが返されます。マップが必要なフォント名を特定するには、前述したいずれかのディレクトリにフォントを追加し、サーバーを再起動して変換を実行します。マッピングに使用するフォント名は、ログファイルから特定できます。

生成された PDF ファイルにフォントを埋め込むには、`cffont.properties` ファイル内の `embedFonts` プロパティを `true` に設定します (デフォルトは `false`)。

6.6.8 Microsoft Visio のデフォルトのマクロ設定の変更

マクロを含む Microsoft Visio のファイルを変換しようとする、Microsoft Office Visio のセキュリティに関する通知ダイアログが表示され、変換がタイムアウトします。マクロが含まれているファイルを正常に変換するには、Visio のデフォルトのマクロ設定を変更する必要があります。

❖ Visio で、**ツール/セキュリティセンター/マクロの設定**をクリックし、次のいずれかのオプションを選択して、「OK」をクリックします。

- 警告を表示せずにすべてのマクロを無効にする
- すべてのマクロを有効にする

6.6.9 Network Printer Client のインストール

PDF Generator には、クライアントコンピューターに PDF Generator ネットワークプリンターをインストールするための実行ファイルが含まれています。インストールが完了すると、PDF Generator プリンターがクライアントコンピューターの既存のプリンターのリストに追加されます。その後、このプリンターを使用してドキュメントを送信し、PDF に変換することができます。

注意：Administration Console のネットワークプリンタークライアントのインストールウィザードでは、Windows オペレーティングシステムのみがサポートされています。ネットワークプリンタークライアントのインストールウィザードの起動には、32 ビット JVM を使用してください。64 ビット JVM を使用した場合は、エラーが発生します。

Windows で PDFG ネットワークプリンターのインストールが失敗する場合や、プリンターを UNIX または Linux のプラットフォームにインストールする場合は、各オペレーティングシステムのネイティブのプリンター追加ユーティリティを使用して、53 ページの「[6.6.9.2 Windows でネイティブのプリンターの追加ウィザードを使用した PDFG ネットワークプリンターの設定](#)」の説明に従って設定してください。

6.6.9.1 PDF Generator ネットワークプリンタークライアントのインストール

注意：Windows Server 2008 で PDF Generator ネットワークプリンタークライアントをインストールする前に、Windows Server 2008 にインターネット印刷クライアント機能がインストールされていることを確認してください。機能のインストールについては、Windows Server 2008 のヘルプを参照してください。

- 1 PDF Generator をサーバーに正常にインストールしたことを確認します。

2 次のいずれかを実行します。

- Windows クライアントコンピューターから、Web ブラウザーに次の URL を入力します。[host] は PDF Generator をインストールしたサーバーの名前、[port] は使用しているアプリケーションサーバーポートです。

`http://[host]:[port]/pdfg-ipp/install`

- Administration Console で、ホーム/サービス/ PDF Generator / PDFG ネットワークプリンターをクリックします。「PDFG ネットワークプリンターのインストール」セクションで、「ここをクリックしてください」をクリックして、PDFG ネットワークプリンターのインストールを起動します。

3 インターネットポートの構成画面で、「指定されたユーザーアカウントを使う」オプションを選択して、PDFG 管理者またはユーザーのロールを持つ LiveCycle ユーザーの資格情報を指定します。このユーザーには電子メールアドレスも必要です。このアドレスは、変換済みのファイルを受信する際に使用できます。このセキュリティ設定をクライアントコンピューター上のすべてのユーザーに適用するには、「すべてのユーザーに同じセキュリティ設定を使う」を選択して、「OK」をクリックします。

注意：ユーザーのパスワードが変更された場合、ユーザーは使用しているコンピューターに PDFG ネットワークプリンターを再インストールする必要があります。パスワードを Administration Console から更新することはできません。

インストールが終了すると、Adobe LiveCycle PDF Generator 11 が正常にインストールされたことを示すダイアログボックスが表示されます。

4 「OK」をクリックします。使用可能なプリンターのリストに「Adobe LiveCycle PDF Generator 11」という名前のプリンターが追加されます。

6.6.9.2 Windows でネイティブのプリンターの追加ウィザードを使用した PDFG ネットワークプリンターの設定

1 スタート/プリンターと FAX をクリックし、「プリンターの追加」をダブルクリックします。

2 「次へ」をクリックし、「ネットワークプリンター、または他のコンピューターに接続されているプリンター」を選択して、「次へ」をクリックします。

3 「インターネット上または自宅/会社のネットワーク上のプリンターに接続する」を選択し、次の PDFG プリンターの URL を入力します。[host] はサーバー名、[port] はサーバーを実行しているポート番号です。

`http://[host]:[port]/pdfg-ipp/printer`

4 インターネットポートの構成画面で、「指定されたユーザーアカウントを使う」を選択し、ユーザーの有効な資格情報を指定します。

5 「プリンタードライバーの選択」ボックスで、任意の標準的な PostScript ベースのプリンタードライバー (HP Color LaserJet PS など) を選択します。

6 適切なオプション (このプリンターをデフォルトに設定するなど) を選択してインストールを完了します。

注意：プリンターの追加の際に使用するユーザーの資格情報では、応答を受信するために、有効な電子メール ID を User Management で設定する必要があります。

7 電子メールサービスの sendmail サービスを設定します。サービスの設定オプションで有効な SMTP サーバーと認証情報を指定します。

6.6.9.3 プロキシサーバーのポート転送を使用するように PDF Generator ネットワークプリンタークライアントをインストールして設定する

1 CC プロキシサーバーで特定のポートについて LiveCycle サーバーへのポート転送を設定し、プロキシサーバーレベルで認証を無効にします (LiveCycle で独自の認証を使用するので)。転送を設定したポートでクライアントがこのプロキシサーバーに接続すると、すべての要求が LiveCycle サーバーに転送されます。

2 次の URL を使用して、PDFG ネットワークプリンターをインストールします。

```
http://[proxy server]:[forwarded port]/pdfg-ipp/install.
```

- PDFG ネットワークプリンターの認証に必要な資格情報を指定します。
- PDFG ネットワークプリンターがクライアントマシンにインストールされます。これにより、ファイアウォールで保護されている LiveCycle サーバーを使用した PDF 変換が可能になります。

6.6.10 ファイル制限機能の設定の変更

Microsoft Office のセキュリティセンター設定を変更して、PDFG が古いバージョンの Microsoft Office ドキュメントを変更できるようにします。

- 任意の Office 2010 アプリケーションで、「ファイル」タブをクリックします。「ヘルプ」の下の「オプション」をクリックします。オプションダイアログボックスが表示されます。
- 「セキュリティセンター」をクリックし、「セキュリティセンターの設定」をクリックします。
- セキュリティセンターダイアログで、「ファイル制限機能の設定」をクリックします。
- 「ファイルの種類」リストで、PDFG に変換させるファイルの種類に対して、「開く」チェックボックスをオフにします。

6.6.11 監視フォルダーのパフォーマンスパラメーター

監視フォルダーを使用した PDF の変換を実行するための十分なディスク容量がないことを示す java.io.IOException エラーメッセージが発生しないように、Administration Console で PDF Generator の設定を変更できます。

PDF Generator のパフォーマンスパラメーターの設定

- Administration Console にログインして、サービス/アプリケーションおよびサービス/サービスの管理を選択します。
- サービスのリストで PDFGConfigService を探してクリックし、以下の値を設定します。
 - PDFG Cleanup Scan Seconds : 1800
 - Job Expiration Seconds : 6000
 - Server Conversion Timeout : デフォルト値の 270 を、450 などの大きい値に変更します。
- 「保存」をクリックして、サーバーを再起動します。

6.6.12 保護フィールドを含む Microsoft Word 文書に対して PDF 変換を有効にする

PDF Generator は保護フィールドを含む Microsoft Word 文書をサポートします。保護フィールドを含む Microsoft Word 文書に対して PDF 変換を有効にするには、次のようにファイルタイプ設定を変更します。

- Administration Console で、Services / PDF Generator / File Type Settings に行き、ファイルタイプ設定プロファイルを開きます。
- Microsoft Word オプションを展開し、「Adobe PDF でドキュメントマークアップを保持 (Microsoft Office 2003 以降)」オプションを選択します。
- 「名前を付けて保存」をクリックし、ファイルタイプ設定の名前を指定し、「OK」をクリックします。

6.7 Rights Management の最終設定

Rights Management では、SSL を使用するようにアプリケーションサーバーを設定する必要があります ([管理ヘルプ](#)を参照)。

6.8 LDAP アクセスの設定

6.8.1 User Management の設定 (ローカルドメイン)

- 1 Web ブラウザーを開き、[http://\[host\]:\[port\]/adminui](http://[host]:[port]/adminui) にアクセスしてログインします (35 ページの「[6.1.5.1 LiveCycle 管理コンソールへのアクセス](#)」を参照)。
- 2 **設定 / User Management / ドメインの管理** をクリックし、「**新規ローカルドメイン**」をクリックします。
- 3 該当するボックスにドメイン ID とドメイン名を入力します ([管理ヘルプ](#)の「ローカルドメインの追加」を参照)。
- 4 (オプション)「**アカウントロックを有効にする**」オプションの選択を解除して、アカウントロックを無効にします。
- 5 「**OK**」をクリックします。

6.8.2 User Management の LDAP 設定 (エンタープライズドメイン)

- 1 Web ブラウザーを開き、[http://\[host\]:\[port\]/adminui](http://[host]:[port]/adminui) にアクセスしてログインします (35 ページの「[6.1.5.1 LiveCycle 管理コンソールへのアクセス](#)」を参照)。
- 2 **設定 / User Management / ドメインの管理** をクリックし、「**新規エンタープライズドメイン**」をクリックします。
- 3 「**ID**」ボックスにドメインの一意の ID を入力し、「**名前**」ボックスにドメインの識別名を入力します。
注意: LiveCycle データベースとして MySQL を使用している場合、ID には 1 バイト (ASCII) 文字のみを使用してください ([管理ヘルプ](#)の「エンタープライズドメインの追加」を参照)。
- 4 「**認証を追加**」をクリックし、**認証プロバイダー**リストで「**LDAP**」を選択します。
- 5 「**OK**」をクリックします。
- 6 「**ディレクトリを追加**」をクリックし、「**プロファイル名**」ボックスに、LDAP プロファイルの名前を入力します。
- 7 「**次へ**」をクリックします。
- 8 「**サーバー**」、「**ポート**」、「**SSL**」、「**バインド**」の各ボックスに値を指定し、「**ページに次の情報を入力**」ボックスで、ディレクトリ設定オプション (「**Sun ONE のデフォルト値**」など) を選択します。また、「**名前**」ボックスと「**パスワード**」ボックスで、匿名アクセスが無効な場合に LDAP データベースへの接続に使用する値を指定します ([管理ヘルプ](#)の「ディレクトリ設定」を参照)。
- 9 (オプション) 設定をテストします。
 - 「**テスト**」をクリックします。画面に、サーバーのテストが成功したか、または設定エラーが存在することを示すメッセージが表示されます。
- 10 「**次へ**」をクリックして、必要に応じて、「**ユーザー設定**」を設定します ([管理ヘルプ](#)の「ディレクトリ設定」を参照)。
- 11 (オプション) 設定をテストします。
 - 「**テスト**」をクリックします。
 - 「**検索フィルター**」ボックスで、検索フィルターを確認するか新しい検索フィルターを指定してから、「**送信**」をクリックします。画面に検索条件に一致するエントリのリストが表示されます。
 - 「**閉じる**」をクリックしてユーザー設定画面に戻ります。

12 「次へ」をクリックして、必要に応じて、「グループ設定」を設定します（管理ヘルプの「ディレクトリ設定」を参照）。

13 (オプション) 設定をテストします。

- 「テスト」をクリックします。
- 「検索フィルター」ボックスで、検索フィルターを確認するか新しい検索フィルターを指定してから、「送信」をクリックします。画面に検索条件に一致するエントリのリストが表示されます。
- 「閉じる」をクリックしてグループの設定画面に戻ります。

14 「完了」をクリックして新規ディレクトリページを閉じ、「OK」をクリックして終了します。

6.9 FIPS モードの有効化

LiveCycle には FIPS モードがあり、RSA BSAFE Crypto-C 2.1 暗号化モジュールを使用して、データ保護を連邦情報処理規格 (FIPS) 140-2 承認アルゴリズムに限定しています。

LiveCycle の設定中に Configuration Manager を使用してこのオプションを有効化しなかった場合、または有効化した設定を無効化する場合は、Administration Console からこの設定を変更できます。

FIPS モードを変更した場合は、サーバーを再起動する必要があります。

FIPS モードは Acrobat 7.0 より前のバージョンをサポートしていません。FIPS モードが有効で、パスワードによる暗号化およびパスワード削除のプロセスに Acrobat 5 の設定が含まれる場合、このプロセスは失敗します。

通常、FIPS が有効化されていると、Assembler サービスでは、どのドキュメントにもパスワードの暗号化が適用されません。この処理が試行されると、FIPSMoException が発生し、FIPS モードではパスワードを暗号化できないことが示されます。また、ベースドキュメントがパスワードで暗号化されている場合、PDFsFromBookmarks エレメントは FIPS モードではサポートされません。

FIPS モードのオンまたはオフ

- 1 Administration Console にログインします。
- 2 設定 / コアシステム設定 / 設定をクリックします。
- 3 「FIPS を有効にする」を選択して FIPS モードを有効化するか、選択を解除して FIPS モードを無効化します。
- 4 「OK」をクリックして、アプリケーションサーバーを再起動します。

注意：LiveCycle ソフトウェアでは、コードを検証して FIPS の互換性を確認しません。FIPS 操作モードは、FIPS で承認されたライブラリ (RSA) の暗号化サービスで、FIPS で承認されたアルゴリズムが使用されるようにするために提供されています。

6.10 HTML 電子署名の設定

Forms の HTML 電子署名機能を使用するには、次の手順を実行します。

- 1 [LiveCycle root]/deploy/adobe-forms-ds.ear ファイルをアプリケーションサーバーに手動でデプロイします。
- 2 管理コンソールにログインし、サービス / LiveCycle Forms ES4 をクリックします。
- 3 「HTML 電子署名が有効です」を選択し、「保存」をクリックします。

6.11 Connector for EMC Documentum の設定

注意: LiveCycle が EMC Documentum をサポートしているのは、バージョン 6.0、6.5 および 6.7 SP1 のみです。ECM が適切にアップグレードされていることを確認してください。

注意: コネクタ用のクライアントのインストール、JAR のファイルおよび設定変更のコピーの作業が、クラスターのすべてのノードで実行されていることを確認してください。

Connector for EMC Documentum を LiveCycle の一部としてインストールした場合は、次の手順を実行して、Documentum リポジトリに接続するように、このサービスを設定します。

Connector for EMC Documentum の設定

1 [appserver root]/bin フォルダにある adobe-component-ext.properties ファイルを開きます (ファイルが存在しない場合は、ファイルを作成します)。

2 次の Documentum Foundation Classes JAR ファイルを指定する新しいシステムプロパティを追加します。

- dfc.jar
- aspectjrt.jar
- log4j.jar
- jaxb-api.jar
- (Connector for EMC Documentum 6.5 のみ)
 - configservice-impl.jar
 - configservice-api.jar

新しいシステムプロパティは、次の形式にする必要があります。

[component id].ext=[JAR files and/or folders]

例えば、デフォルトの Content Server と Documentum Foundation Classes のインストールを使用して、次のいずれかのシステムプロパティをファイルに追加します。その際、システムプロパティは新しい行に記述し、行中に改行を入れず、末尾で改行してください。

- Connector for EMC Documentum 6.0 のみ :

```
com.adobe.livecycle.ConnectorforEMCDocumentum.ext=  
C:/Program Files/Documentum/Shared/dfc.jar,  
C:/Program Files/Documentum/Shared/aspectjrt.jar,
```

- Connector for EMC Documentum 6.5 のみ :

```
com.adobe.livecycle.ConnectorforEMCDocumentum.ext=  
C:/Program Files/Documentum/Shared/dfc.jar,  
C:/ProgramFiles/Documentum/Shared/aspectjrt.jar,  
C:/Program Files/Documentum/Shared/log4j.jar,  
C:/Program Files/Documentum/Shared/jaxb-api.jar,  
C:/Program Files/Documentum/Shared/configservice-impl.jar,  
C:/Program Files/Documentum/Shared/configservice-api.jar
```

注意: 上記のテキストには、改行が含まれています。このテキストをコピー&ペーストする場合、改行を削除してください。

- Connector for EMC Documentum 6.7 SP1 のみ :

```
com.adobe.livecycle.ConnectorforEMCDocumentum.ext=  
C:/Program Files/Documentum/Shared/dfc.jar,  
C:/ProgramFiles/Documentum/Shared/aspectjrt.jar,  
C:/Program Files/Documentum/Shared/log4j.jar,  
C:/Program Files/Documentum/Shared/jaxb-api.jar,  
C:/Program Files/Documentum/Shared/configservice-impl.jar,  
C:/Program Files/Documentum/Shared/configservice-api.jar  
C:/Program Files/Documentum/Shared/commons-codec-1.3.jar  
C:/Program Files/Documentum/Shared/commons-lang-2.4.jar
```

注意：上記のテキストには、改行が含まれています。このテキストをコピー&ペーストする場合、改行を削除してください。

3 クラスターの各アプリケーションサーバーインスタンスで、前の手順を繰り返します。

4 Web ブラウザーを開き、次の URL を入力します。

```
http://[host]:[port]/adminui
```

5 次のデフォルトのユーザー名とパスワードを使用してログインします。

ユーザー名：administrator

パスワード：password

6 サービス / **LiveCycle 11 Connector for EMC Documentum** / **環境設定** に移動して、以下のタスクを実行します。

- 必要な Documentum リポジトリ情報のすべてを入力します。
- Documentum をリポジトリプロバイダーとして使用するには、「リポジトリサービスプロバイダー」で「**EMC Documentum リポジトリプロバイダー**」を選択し、「**保存**」をクリックします。[管理ヘルプ](#)のページの右上にあるヘルプリンクをクリックしてください。

7 (オプション) サービス / **LiveCycle 11 Connector for EMC Documentum** / **リポジトリ証明書の設定** に移動して、「**追加**」をクリックし、Docbase 情報を指定して、「**保存**」をクリックします (詳しくは、右上隅の「ヘルプ」をクリックしてください)。

8 アプリケーションサーバーが現在実行されていない場合は、サーバーを起動します。実行されている場合は、サーバーを停止し、再起動します。

9 Web ブラウザーを開き、次の URL を入力します。

```
http://[host]:[port]/adminui
```

10 次のデフォルトのユーザー名とパスワードを使用してログインします。

ユーザー名：administrator

パスワード：password

11 サービス / **アプリケーションおよびサービス / サービスの管理** に移動して、以下のサービスを選択します。

- EMCDocumentumAuthProviderService
- EMCDocumentumContentRepositoryConnector
- EMCDocumentumRepositoryProvider

12 「**開始**」をクリックします。サービスのいずれかが正常に起動されない場合は、前の手順で実行した設定を確認します。

13 次のいずれかの操作を行います。

- Documentum Authorization サービス (EMCDocumentumAuthProviderService) を使用して、Workbench の Resources ビューで Documentum リポジトリのコンテンツを表示するには、この手順を続行します。Documentum Authorization サービスを使用すると、デフォルトの LiveCycle 認証が上書きされるので、Documentum の資格情報を使用して Workbench にログインするように設定する必要があります。

- LiveCycle リポジトリを使用するには、LiveCycle の上級管理者の資格情報（デフォルトは **administrator** と **password**）を使用して Workbench にログインします。

これで、この手順に必要なステップを完了しました。この場合、手順 19 で指定した資格情報を使用してデフォルトリポジトリにアクセスし、デフォルトの LiveCycle 認証サービスを使用します。

14 アプリケーションサーバーを再起動します。

15 Administration Console にログインし、**設定 / User Management / ドメインの管理** をクリックします。

16 「新規エンタープライズドメイン」をクリックして、ドメイン ID と名前を入力します。ドメイン ID は、ドメインの一意の識別子です。名前は、ドメインの識別名です。

注意： LiveCycle データベースとして MySQL を使用している場合、ID には 1 バイト (ASCII) 文字のみを使用してください (LiveCycle の管理ヘルプの「エンタープライズドメインの追加」を参照)。

17 カスタム認証プロバイダーを追加します。

- 「認証を追加」をクリックします。
- 認証プロバイダーリストで「**カスタム**」を選択します。
- 「EMCDocumentumAuthProvider」を選択し、「OK」をクリックします。

18 LDAP 認証プロバイダーを追加します。

- 「認証を追加」をクリックします。
- 認証プロバイダーリストで「**LDAP**」を選択し、「OK」をクリックします。

19 LDAP ディレクトリを追加します。

- 「ディレクトリを追加」をクリックします。
- 「プロファイル名」ボックスに一意の名前を入力し、「次へ」をクリックします。
- 「サーバー」、「ポート」、「SSL」、「バインド」および「ページに次の情報を入力」オプションの値を指定します。「バインド」オプションで「ユーザー」を選択する場合は、「名前」と「パスワード」フィールドにも値を指定する必要があります。
- (オプション) 必要に応じてベースドメイン名を取得するには、「BaseDN を取得」を選択します。
- 「次へ」をクリックし、ユーザー設定を指定して「次へ」をクリックし、必要に応じてグループ設定を指定して「次へ」をクリックします。

設定について詳しくは、ページの右上隅にある「**User Management ヘルプ**」をクリックしてください。

20 「OK」をクリックして「ディレクトリを追加」ページを閉じ、もう一度「OK」をクリックします。

21 新しいエンタープライズドメインを選択し、「今すぐ同期」をクリックします。LDAP ネットワークのユーザーとグループ数および接続の速度によって、同期処理には数分かかる場合があります。

(オプション) 同期のステータスを確認するには、「更新」をクリックし、「現在の同期の状態」列にステータスを表示します。

22 **設定 / User Management / ユーザーとグループ** をクリックします。

23 LDAP から同期されたユーザーを検索し、以下のタスクを実行します。

- 1 つ以上のユーザーを選択し、「**ロールをアサイン**」をクリックします。
- 1 つ以上の LiveCycle のロールを選択し、「OK」をクリックします。
- 「OK」をもう一度クリックして、ロールアサインを確認します。

ロールをアサインするすべてのユーザーについて、この手順を繰り返します。詳しくは、ページの右上隅にある「**User Management ヘルプ**」をクリックしてください。

24 Workbench を起動し、Documentum リポジトリ用の次の資格情報を使用してログインします。

Username : [username]@[repository_name]

Password : [password]

ログイン後は、Documentum リポジトリは、Workbench 内の Resources ビューに表示されます。

username@repository_name を使用してログインしない場合、Workbench では、デフォルトリポジトリへのログインが試行されます。

25 (オプション) Connector for EMC Documentum の LiveCycle サンプルをインストールするには、Samples という名前の Documentum リポジトリを作成して、その中にインストールします。

Connector for EMC Documentum サービスの設定後の、Documentum リポジトリでの Workbench の設定については、LiveCycle 管理ヘルプを参照してください。

6.11.1 Documentum リポジトリでの XDP MIME 形式の作成

ユーザーが Documentum リポジトリから XDP ファイルを取得し、保存できるようにするには、次のタスクのいずれかを実行する必要があります。

- ユーザーがアクセスする XDP ファイルが置かれている各リポジトリに、対応する XDP 形式を作成します。
- Documentum リポジトリにアクセスするときに Documentum 管理者アカウントを使用するように、Connector for EMC Documentum サービスを設定します。この場合、Connector for EMC Documentum サービスでは必要に応じて XDP 形式が使用されます。

Documentum 管理者アカウントを使用した Documentum Content Server での XDP 形式の作成

1 Documentum 管理者アカウントにログインします。

2 「形式」をクリックし、**ファイル／新規作成／形式**を選択します。

3 次の情報を対応するフィールドに入力します。

名前 : xdp

デフォルトのファイル拡張子 : xdp

Mime タイプ : application/xdp

4 ユーザーが XDP ファイルを保存する他のすべての Documentum リポジトリについて、手順 1 ~ 3 を繰り返します。

Documentum 管理者アカウントを使用するための Connector for EMC Documentum サービスの設定

1 Web ブラウザーを開き、次の URL を入力します。

http://[host]/:[port]/adminui

2 次のデフォルトのユーザー名とパスワードを使用してログインします。

ユーザー名 : administrator

パスワード : password

3 **サービス／LiveCycle 11 Connector for EMC Documentum** / **環境設定** をクリックします。

4 「Documentum プリンシパル秘密鍵証明書に関する情報」領域で、次の情報を更新し、「保存」をクリックします。

ユーザー名 : [Documentum Administrator user name]

パスワード : [Documentum Administrator password]

5 「リポジトリ証明書の設定」をクリックして、リストからリポジトリを選択します。リストにない場合は、「追加」をクリックします。

- 6 対応するフィールドで適切な情報を指定して、「保存」をクリックします。

リポジトリ名：[Repository Name]

リポジトリ証明書のユーザー名：[Documentum Administrator user name]

リポジトリ証明書のパスワード：[Documentum Administrator password]

- 7 ユーザーが XDP ファイルを保存するすべてのリポジトリについて、手順 5～6 を繰り返します。

6.11.2 複数の接続ブローカーのサポートの追加

LiveCycle Configuration Manager がサポートする接続ブローカーは 1 つのみです。LiveCycle 管理コンソールを使用して、複数の接続ブローカーのサポートを追加します。

- 1 「LiveCycle 管理コンソール」を開きます。
- 2 ホーム/サービス/LiveCycle 11 Connector for EMC Documentum /環境設定に移動します。
- 3 「接続ブローカーのホスト名または IP アドレス」で、別の接続ブローカーのホスト名のカンマで区切りられたリストを入力します。例えば、host1、host2、host3 と入力します。
- 4 「接続ブローカーのポート番号」で、対応する接続ブローカーのポートのカンマで区切りられたリストを入力します。例えば、1489、1491、1489 を入力します。
- 5 「Save」をクリックします。

6.12 Connector for IBM Content Manager の設定

注意： LiveCycle が IBM Content Manager をサポートしているのは、バージョン 8.4 のみです。ECM が適切にアップグレードされていることを確認してください。

注意： コネクタ用のクライアントのインストール、JAR のファイルおよび設定変更のコピーの作業が、クラスターのすべてのノードで実行されていることを確認してください。

Connector for IBM Content Manager サービスを LiveCycle の一部としてインストールした場合は、次の手順を実行して、IBM Content Manager データストアに接続するようサービスを設定します。

Connector for IBM Content Manager の設定

- 1 [appserver root] フォルダーにある adobe-component-ext.properties ファイルを開きます。ファイルが存在しない場合は、ファイルを作成します。
- 2 次の IBM I14C JAR ファイルの場所を指定する、新しいシステムプロパティを追加します。
 - cmb81.jar
 - cmbcm81.jar
 - cmbicm81.jar
 - cmblog4j81.jar
 - cmbsdk81.jar
 - cmbutil81.jar
 - cmbutilicm81.jar
 - cmbview81.jar
 - cmbwas81.jar

- cmbwcm81.jar
- cmgmt

注意: cmgmt は JAR ファイルではありません。Windows では、このフォルダーはデフォルトで C:\Program Files\IBM\db2cmv8\ にあります。

- common.jar
- db2jcc.jar
- db2jcc_license_cisuz.jar
- db2jcc_license_cu.jar
- ecore.jar
- ibmjgssprovider.jar
- ibmjsseprovider2.jar
- ibmpkcs.jar
- icmrm81.jar
- jcache.jar
- log4j-1.2.8.jar
- xerces.jar
- xml.jar
- xsd.jar

新しいシステムプロパティは次のようになります。

[component id].ext=[JAR files and/or folders]

例えば、デフォルトの DB2 Universal Database Client および II4C インストールを使用する場合、次のシステムプロパティをファイルに追加します。その際、システムプロパティは新しい行に記述し、行中に改行を入れず、末尾で改行してください。

```
C:/Program Files/IBM/db2cmv8/cmgmt,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/ibmjsseprovider2.jar,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/ibmjgssprovider.jar,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/ibmpkcs.jar,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/xml.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbview81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmb81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbcm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/xsd.jar,  
C:/Program Files/IBM/db2cmv8/lib/common.jar,  
C:/Program Files/IBM/db2cmv8/lib/ecore.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbicm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbwcm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/jcache.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbutil81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbutilicm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/icmrm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/db2jcc.jar,  
C:/Program Files/IBM/db2cmv8/lib/db2jcc_license_cu.jar,  
C:/Program Files/IBM/db2cmv8/lib/db2jcc_license_cisuz.jar,  
C:/Program Files/IBM/db2cmv8/lib/xerces.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmblog4j81.jar,  
C:/Program Files/IBM/db2cmv8/lib/log4j-1.2.8.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbSDK81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbwas81.jar
```

- 3 アプリケーションサーバーが現在実行されていない場合は、サーバーを起動します。実行されている場合は、サーバーを停止し、再起動します。

これで、IBMCMConnectorService プロパティシートから IBM Content Manager データストアに、「Use User credentials」をログインモードとして使用して接続できます。

これで、この手順に必要なステップを完了しました。

(オプション) IBMCMConnectorService プロパティシートから IBM Content Manager データストアに、「Use Credentials From Process Context」をログインモードとして使用して接続するには、次の手順を実行します。

「Use Credentials from process context」 ログインモードを使用した接続

- 1 Web ブラウザーを開き、次の URL を入力します。

`http://[host]/:[port]/adminui`

- 2 上級管理者の資格情報を使用してログインします。インストール中に設定されたデフォルト値は、次のとおりです。

ユーザー名: administrator

パスワード: password

- 3 サービス / **LiveCycle 11 Connector for IBM Content Manager** をクリックします。
- 4 必要なりポジトリ情報のすべてを入力して「保存」をクリックします。IBM Content Manager リポジトリ情報について詳しくは、ページの右上隅にある「ヘルプ」リンクをクリックします。
- 5 次のいずれかのタスクを実行します。

- IBM Content Manager Authorization サービス (IBMCMAuthProvider) を使用して IBM Content Manager データストアのコンテンツを Workbench の Processes ビューで使用するには、この手順を続行します。IBM Content Manager Authorization サービスを使用すると、デフォルトの LiveCycle 認証が上書きされるので、IBM Content Manager の資格情報を使用して Workbench にログインするように設定する必要があります。
- Workbench の Processes ビューで IBM Content Manager データストアのコンテンツを使用するために手順 4 で指定したシステム資格情報を使用するには、LiveCycle の上級管理者の資格情報 (デフォルトは **administrator** と **password**) を使用して、Workbench にログインします。これで、この手順に必要なステップを完了しました。こ

の場合、手順 4 で指定したシステム資格情報は、デフォルトリポジトリにアクセスするためのデフォルトの LiveCycle 認証サービスを使用します。

- 6 Administration Console にログインし、**設定 / User Management / ドメインの管理**をクリックします。
- 7 「新規エンタープライズドメイン」をクリックして、ドメイン ID と名前を入力します。ドメイン ID は、ドメインの一意的識別子です。名前は、ドメインの識別名です。

注意： LiveCycle データベースとして MySQL を使用している場合、ID には 1 バイト (ASCII) 文字のみを使用してください (管理ヘルプの「エンタープライズドメインの追加」を参照)。

- 8 カスタム認証プロバイダーを追加します。
 - 「認証を追加」をクリックします。
 - **認証プロバイダー**リストで「**カスタム**」を選択し、「**IBMCMAuthProviderService**」を選択して、「**OK**」をクリックします。
- 9 LDAP 認証プロバイダーを追加します。
 - 「認証を追加」をクリックします。
 - **認証プロバイダー**リストで「**LDAP**」を選択し、「**OK**」をクリックします。
- 10 LDAP ディレクトリを追加します。
 - 「ディレクトリを追加」をクリックします。
 - 「プロファイル名」ボックスに一意的名前を入力し、「次へ」をクリックします。
 - 「サーバー」、「ポート」、「SSL」、「バインド」および「ページに次の情報を入力」オプションの値を指定します。「バインド」オプションで「ユーザー」を選択する場合は、「名前」と「パスワード」フィールドにも値を指定する必要があります。(オプション) 必要に応じてベースドメイン名を取得するには、「BaseDN を取得」を選択します。完了したら、「次へ」をクリックします。
 - ユーザー設定を指定し、「次へ」をクリックし、必要に応じてグループ設定を指定して「次へ」をクリックします。

上記の設定について詳しくは、ページの右上隅にある「ヘルプ」リンクをクリックしてください。
- 11 「OK」をクリックして「ディレクトリを追加」ページを閉じ、もう一度「OK」をクリックします。
- 12 新しいエンタープライズドメインを選択し、「今すぐ同期」をクリックします。LDAP ネットワークのユーザーとグループ数および接続の速度によって、同期処理には数分かかる場合があります。
- 13 同期のステータスを確認するには、「更新」をクリックし、「現在の同期の状態」列にステータスを表示します。
- 14 **設定 / User Management / ユーザーとグループ**をクリックします。
- 15 LDAP から同期されたユーザーを検索し、以下のタスクを実行します。
 - 1 つ以上のユーザーを選択し、「**ロールをアサイン**」をクリックします。
 - 1 つ以上の LiveCycle のロールを選択し、「**OK**」をクリックします。
 - 「OK」をもう一度クリックして、ロールアサインを確認します。

ロールをアサインするすべてのユーザーについて、この手順を繰り返します。詳しくは、ページの右上隅にある「ヘルプ」リンクをクリックします。
- 16 Workbench を起動し、IBM Content Manager データストア用の次の資格情報を使用してログインします。

Username : [username]@[repository_name]

Password : [password]

これで、IBMCMConnectorService オーケストレーション可能コンポーネントのログインモードが「**Use Credentials from process context**」と選択されている場合に、Workbench の Processes ビューで IBM Content Manager データストアを使用できます。

6.13 Connector for IBM FileNet の設定

LiveCycle が IBM FileNet をサポートしているのは、バージョン 4.0、4.5 および 5.0 のみです。ECM が適切にアップグレードされていることを確認してください。

Connector for IBM FileNet を LiveCycle の一部としてインストールした場合は、FileNet オブジェクトストアに接続するように、このサービスを設定する必要があります。

注意：コネクタ用のクライアントのインストール、JAR のファイルおよび設定変更のコピーの作業が、クラスターのすべてのノードで実行されていることを確認してください。

次の手順を実行して、Connector for IBM FileNet を設定します。

FileNet 4.x または FileNet 5.0 および CEWS トランSPORTを使用して Connector for IBM FileNet を設定するには：

1 アプリケーションサーバーの実行ファイルをテキストエディターで開きます。実行ファイルは次の場所にあります。

- (Windows)[**appserver root**]/bin/run.conf
- (Windows 以外)[**appserver root**]/bin/run.conf

2 (FileNet 4.x の場合のみ) FileNet 設定ファイルの場所を、アプリケーションサーバーの start コマンドに Java オプションとして追加し、ファイルを保存します。

注意：JBoss がサービスとして実行されている場合、他の JVM 引数が定義されているレジストリに、Java オプションを追加します。

```
-Dwasp.location= <configuration files location>
```

例えば、デフォルトの FileNet Application Engine インストールを Windows オペレーティングシステムで使用する場合、次の Java オプションを追加します。

```
-Dwasp.location=C:/Progra~1/FileNet/AE/CE_API/wsi
```

3 デプロイメントで Process Engine Connector サービスを使用している場合は、[**appserver root**]\client\logkit.jar を次のディレクトリにコピーします。

- (手動で設定した JBoss、クラスター) [**appserver root**]/server/all/lib
- (手動で設定した JBoss、シングルサーバー) [**appserver root**]/server/standard/lib
- (アドビにより事前設定された JBoss、クラスター) [**appserver root**]/server/lc_<db-name>_cl/lib
- (アドビにより事前設定された JBoss、シングルサーバー) [**appserver root**]/server/lc_<db-name>/lib

4 [**appserver root**]/bin/bin フォルダにある adobe-component-ext.properties ファイルを開きます (ファイルが存在しない場合は、ファイルを作成します)。

5 次の FileNet Application Engine JAR ファイルの場所を指定する、新しいシステムプロパティを追加します。

FileNet 4.x の場合、次の JAR ファイルを追加します。

- javaapi.jar
- soap.jar
- wasp.jar
- builtin_serialization.jar (FileNet 4.0 のみ)

- wsdl_api.jar
- jaxm.jar
- jaxrpc.jar
- saaj.jar
- jetty.jar
- runner.jar
- p8cjares.jar
- Jace.jar
- (オプション) pe.jar

FileNet 5.0 の場合、次の JAR ファイルを追加します

- Jace.jar
- javaapi.jar
- log4j.jar
- pe.jar
- stax-api.jar
- xlpScanner.jar
- xlpScannerUtils.jar

注意：pe.jar ファイルは、デプロイメントで IBMFileNetProcessEngineConnector サービスを使用する場合にのみ追加します。新しいシステムプロパティには、次の構造を反映させる必要があります。

```
[component id].ext=[JAR files and/or folders]
```

例えば、デフォルトの FileNet Application Engine インストールを Windows オペレーティングシステムで使用する場合、次のシステムプロパティをファイルに追加します。その際、システムプロパティは新しい行に記述し、行中に改行を入れず、末尾で改行してください。

注意：次のテキストには、改行が含まれています。このテキストを、このドキュメント以外の場所にコピーする場合は、新しい場所に貼り付けるときに改行を削除してください。

```
com.adobe.livecycle.ConnectorforIBMFileNet.ext=  
C:/Program Files/FileNet/AE/CE_API/lib2/javaapi.jar,  
C:/Program Files/FileNet/AE/CE_API/lib2/log4j-1.2.13.jar
```

6 (FileNet Process Engine Connector のみ) 次の手順で、プロセスエンジンの接続プロパティを設定します。

- テキストエディターを使用してファイルを作成し、次のコンテンツを 1 行で入力します。末尾で改行してください。

```
RemoteServerUrl = cemp:http://[contentserver_IP]:[contentengine_port]/ wsi/FNCEWS40DIME/
```

- このファイルを WcmApiConfig.properties という名前で別のフォルダーに保存して、そのフォルダーの場所を adobe-component-ext.properties ファイルに追加します。

例えば、このファイルを c:\pe_config\WcmApiConfig.properties として保存して、パス c:\pe_config を adobe-component-ext.properties ファイルに追加します。

注意：ファイル名では大文字と小文字が区別されます。

7 次のフォルダーで login-config.xml ファイルを探し、次のアプリケーションポリシーを <policy> ノードの子として追加します。

- (手動で設定した JBoss、シングルサーバー) [appserver root]/server/standard/conf

- (手動で設定した JBoss、クラスター) [appserver root]/server/all/conf
- (アドビにより事前設定された JBoss、シングルサーバー) [appserver root]/server/lc_<dbname>/conf
- (アドビにより事前設定された JBoss、クラスター) [appserver root]/server/lc_<dbname>_cl/conf

```
<application-policy name = "FileNetP8WSI">
<authentication>
  <login-module code = "com.filenet.api.util.WSILoginModule" flag =
    "required" />
</authentication>
</application-policy>
```

- 8 (FileNet Process Engine Connector のみ) 実際のデプロイメントでプロセスエンジンを使用している場合は、次のノードを login-config ファイルに追加します。

```
<application-policy name = "FileNetP8">
<authentication>
  <login-module code = "com.filenet.api.util.WSILoginModule" flag =
    "required" />
</authentication>
</application-policy>
```

- 9 アプリケーションサーバーが現在実行されていない場合は、サーバーを起動します。実行されている場合は、サーバーを停止し、再起動します。

- 10 JBoss がサービスとして実行されている場合は、JBoss for Adobe LiveCycle ES4 サービスを開始 (または再開) します。

- 11 (クラスターのみ) クラスターの各インスタンスに対して、これまでのすべての手順を繰り返します。

- 12 Web ブラウザーを開き、次の URL を入力します。

http://[host]:[port]/adminui

- 13 次のデフォルトのユーザー名とパスワードを使用してログインします。

ユーザー名 : administrator

パスワード : password

- 14 サービス / LiveCycle 11 Connector for IBM FileNet をクリックします。

- 15 必要なすべての FileNet リポジトリ情報を入力し、「リポジトリサービスプロバイダー」の下で「IBM FileNet リポジトリプロバイダー」を選択します。

オプションのプロセスエンジンサービスをデプロイメントで使用する場合、「プロセスエンジン設定」領域で「プロセスエンジンコネクタサービスを使用」を選択し、プロセスエンジンの各設定を指定します。詳しくは、ページの右上隅にある「ヘルプ」リンクをクリックします。

注意：この手順で指定する資格情報は、IBM FileNet リポジトリサービスを後で起動するときに検証されます。資格情報が無効な場合はエラーが発生し、サービスは起動されません。

- 16 「保存」をクリックし、サービス / アプリケーションおよびサービス / サービスの管理に移動します。

- 17 次の各サービスの横にあるチェックボックスを選択して「開始」をクリックします。

- IBMFileNetAuthProviderService
- IBMFileNetContentRepositoryConnector
- IBMFileNetRepositoryProvider
- IBMFileNetProcessEngineConnector (設定されている場合)

サービスのいずれかが正常に開始されない場合は、Process Engine 設定を確認します。

18 次のいずれかの操作を行います。

- FileNet Authorization サービス (IBMFileNetAuthProviderService) を使用して Workbench の Resources ビューで FileNet オブジェクトストアからコンテンツを表示するには、この手順を続行します。FileNet Authorization サービスを使用すると、デフォルトの LiveCycle 認証が上書きされるので、FileNet の資格情報を使用して Workbench にログインするように設定する必要があります。
- LiveCycle リポジトリを使用するには、LiveCycle の上級管理者の資格情報 (デフォルトは **administrator** と **password**) を使用して Workbench にログインします。この場合、手順 16 で指定した資格情報は、デフォルトリポジトリにアクセスするためにデフォルトの LiveCycle 認証サービスを使用します。

19 アプリケーションサーバーを再起動します。

20 Administration Console にログインし、**設定 / User Management / ドメインの管理** をクリックします。

21 「**新規エンタープライズドメイン**」をクリックして、ドメイン ID と名前を入力します。ドメイン ID は、ドメインの一意の識別子です。名前は、ドメインの識別名です。

LiveCycle データベースとして MySQL を使用している場合、ID には 1 バイト (ASCII) 文字のみを使用してください ([LiveCycle 管理ヘルプ](#)の「エンタープライズドメインの追加」を参照)。

22 カスタム認証プロバイダーを追加します。

- 「**認証を追加**」をクリックします。
- 「**認証プロバイダー**」リストで「**カスタム**」を選択します。
- 「**IBMFileNetAuthProviderService**」を選択し、「**OK**」をクリックします。

23 LDAP 認証プロバイダーを追加します。

- 「**認証を追加**」をクリックします。
- 「**認証プロバイダー**」リストで「**LDAP**」を選択し、「**OK**」をクリックします。

24 LDAP ディレクトリを追加します。

- 「**ディレクトリを追加**」をクリックし、「**プロファイル名**」ボックスに一意の名前を入力して、「**次へ**」をクリックします。
- 「**サーバー**」、「**ポート**」、「**SSL**」、「**バインド**」および「**ページに次の情報を入力**」オプションの値を指定します。「**バインド**」オプションで「**ユーザー**」を選択する場合は、「**名前**」と「**パスワード**」フィールドにも値を指定する必要があります。
- (オプション) 必要に応じてベースドメイン名を取得するには、「**BaseDN を取得**」を選択します。完了したら、「**次へ**」をクリックします。
- ユーザー設定を指定し、「**次へ**」をクリックし、必要に応じてグループ設定を指定して「**次へ**」をクリックします。
設定について詳しくは、ページの右上隅にある「**ヘルプ**」リンクをクリックしてください。

25 「**OK**」をクリックして「**ディレクトリを追加**」ページを閉じ、もう一度「**OK**」をクリックします。

26 新しいエンタープライズドメインを選択し、「**今すぐ同期**」をクリックします。LDAP ネットワークのユーザーとグループ数および接続の速度によって、同期処理には数分かかる場合があります。

(オプション) 同期のステータスを確認するには、「**更新**」をクリックし、「**現在の同期の状態**」列にステータスを表示します。

27 **設定 / User Management / ユーザーとグループ** をクリックします。

28 LDAP から同期されたユーザーを検索し、以下のタスクを実行します。

- 1 つ以上のユーザーを選択し、「**ロールをアサイン**」をクリックします。
- 1 つ以上の LiveCycle のロールを選択し、「**OK**」をクリックします。

- 「OK」をもう一度クリックして、ロールアサインを確認します。

ロールをアサインするすべてのユーザーについて、この手順を繰り返します。詳しくは、ページの右上隅にある「ヘルプ」リンクをクリックします。

29 Workbench を起動して、IBM FileNet リポジトリ用の次の資格情報を使用してログインします。

ユーザー名：[username]@[repository_name]

パスワード：[password]

これで、FileNet オブジェクトストアが Workbench の Resources ビューに表示されます。**username@repository name** を使用してログインしない場合、Workbench では、手順 16 で指定したデフォルトリポジトリへのログインが行われます。

30 (オプション) Connector for IBM FileNet の LiveCycle サンプルをインストールする場合、**Samples** という名前の FileNet オブジェクトストアを作成してその中にインストールします。

Connector for IBM FileNet を設定したら、FileNet リポジトリを使用した Workbench の機能の設定について、LiveCycle 管理ヘルプを参照することをお勧めします。

6.14 JBoss クラスターの隔離

多くの JBoss サービスが、複数の JGroup チャンネルサービスを作成します。これらのチャンネルは、特定のチャンネルとのみ通信する必要があります。

ネットワーク上の他のクラスターから JGroups クラスターを隔離するには、次のことを確認します。

- 様々なクラスターのチャンネルが、異なるグループ名を使用していること。./run.sh -g QAPartition -b <ipaddress> -c all を使用して、一意のグループを作成します。
- 様々なクラスターのチャンネルが、異なるマルチキャストアドレスを使用していること。/run.sh -u <UDP group Ip address> -g QAPartition -b <ipaddress> -c all を使用して、マルチキャストアドレスを制御します。
- 各クラスターのチャンネルで、異なるマルチキャストポートを使用していること。/run.sh -u <UDP group Ip address> -g QAPartition -b <ipaddress> -c all \-Djboss.jgroups.udp.mcast_port=12345 -Djboss.messaging.datachanneludpport=23456 を使用して、マルチキャストソケットを制御します。

JBoss クラスターの隔離の詳細な手順については、jbossclustering ガイド (<http://docs.jboss.org/>) の「Isolating JGroups Channels」を参照してください。

6.15 (オプション) JMX コンソールセキュリティを有効にする

LiveCycle のデフォルト設定では、JBoss JMX コンソールセキュリティは無効になっています。セキュリティを有効にするには、以下に説明する手順に従います。

1 アプリケーションサーバーを停止します。

2 [appserver root]/server/<profile_name>/deploy ディレクトリに移動して、jmx-invoker-service.xml ファイルをテキストエディターで開きます。

3 invoke セクションで次の行がコメントアウトされていないことを確認します。

```
<interceptor code="org.jboss.jmx.connector.invoker.AuthenticationInterceptor"
securityDomain="java:/jaas/jmx-console"/>
```

- 4 ファイルを保存して閉じます。
- 5 新規ファイル `work-manager.properties` を `[appserver root]/server/` で作成します。
- 6 `work-manager.properties` ファイルをテキストエディターで開き、次のコードを追加します。

```
adobe.work-manager.jboss.jmx.lookup.java.naming.factory.initial=org.jboss.security.jndi.JndiLo
ginInitialContextFactory
adobe.work-manager.jboss.jmx.lookup.java.naming.provider.url=jnp://localhost:1099/
adobe.work-manager.jboss.jmx.lookup.java.naming.security.credentials=<password>
adobe.work-manager.jboss.jmx.lookup.java.naming.security.principal=<username>
adobe.work-manager.jboss.jmx.lookup.java.naming.security.protocol=jmx-console
```

注意：特定の秘密鍵証明書が `jmx-console-users.properties` ファイルおよび `work-manager.properties` ファイルに記載されていることを確認してください。デフォルトの管理者の秘密鍵証明書は `/admin` です。

- 7 ファイルを保存して閉じます。
- 8 `[appserver root]/server/<profile_name>/conf/props` ディレクトリに移動して、`jmx-console-users.properties` ファイルをテキストエディターで開きます。
- 9 `work-manager.properties` ファイルで使用されている秘密鍵証明書が含まれるエントリがコメントアウトされていないことを確認してください。
- 10 ファイルを保存して閉じます。
- 11 `[appserver root]/bin` に移動して、`run.conf.bat` ファイルをテキストエディターで開き、次のコードを追加します。

```
set "JAVA_OPTS=%JAVA_OPTS% -Dadobe.workmanager.properties = <path of the work-manager.properties file>
```
- 12 ファイルを保存して閉じます。
- 13 アプリケーションサーバーを起動します。

第7章：ロードバランシングの設定

ロードバランシング機能が動作するように JBoss クラスターを設定できます。ロードバランサーを使用すると、作業負荷をクラスターのすべてのノードに均等に分散することができます。Apache Web サーバーとさまざまなプラグインを使用してクラスターのロードバランシングを実装できます。LiveCycle クラスターはスティッキー機能を有効にしたロードバランサーのみをサポートします。

使用しているオペレーティングシステムに適した Apache Web サーバーソフトウェアを入手します。

- Windows のばあい、Apache Web サーバーを Apache HTTP Server Project サイトからダウンロードします。
- Solaris 64 ビットの場合、Apache Web サーバーを Sunfreeware for Solaris Web サイトからダウンロードします。
- Linux の場合、Apache Web サーバーは、Linux システムにプレインストールされています。

Apache は HTTP または AJP プロトコルを使用して JBoss と通信できます。両方のプロトコルを使用したロードバランシングの設定を次に示します。

7.1 HTTP コネクターベースのロードバランシング

7.1.1 mod_proxy の使用

この設定はプロキシモジュールと一緒に Apache プロキシバランサーモジュールを使用して、HTTP プロトコルでの JBoss クラスターのロードバランスを取ります。

APACHE_HOME/conf/httpd.conf ファイル内で次のモジュール設定に対してコメント解除します。

```
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_balancer_module modules/mod_proxy.so
LoadModule proxy_balancer_module modules/mod_proxy_http.so
```

注意：Linux の場合、デフォルトの APACHE_HOME は `/etc/httpd/` です

mod_proxy ベースのロードバランシングを実装するには次の 2 つの方法があります。

- mod_headers の使用
- JSESSIONID の使用

7.1.1.1 mod_headers の使用

この設定は、mod_headers を使用してスティッキー機能によるロードバランシングを行う方法を提供します。

次の設定を APACHE_HOME/conf/httpd.conf 設定ファイルに追加してください。

```
LoadModule headers_module modules/mod_headers.so
<VirtualHost *:80>
ServerName lb.xyz.com
ServerAlias lb.xyz.com
ProxyRequests Off
ProxyPreserveHost On
Header add Set-Cookie "ROUTEID=.%{BALANCER_WORKER_ROUTE}e; path=/" env=BALANCER_ROUTE_CHANGED
<Proxy balancer://cluster>
BalancerMember http://abc.com:8080 route=1
BalancerMember http://xyz.com:8080 route=2
ProxySet stickySession=ROUTEID
</Proxy>
ProxyPass / balancer://cluster/
ProxyPassReverse / balancer://cluster/

</VirtualHost>
```

7.1.1.2 JSESSIONID の使用

この設定は、JSESSIONID を使用してロードバランシングを行う方法を提供します。

JBoss アプリケーションサーバーでは、tomcat コンテナがインスタンスの名前をセッション ID クッキーの最後に、セッション ID からドット (.) で区切って追加します。したがって、Apache Web サーバーがセッション持続クッキーの値の中にドットを検出すると、ドットの後ろの部分のみを使用して、ルートを検索します。Tomcat Server インスタンス ro がそのインスタンス名を認識されるためには、JBoss 設定ファイルデプロイ /jbossweb.sar/server.xml 内部の属性 jvmRoute を、それぞれの JBoss に接続するワーカーのルートの値に設定します。ここで使用しているセッションクッキーの名前は JSESSIONID です。

次の設定を APACHE_HOME/conf/httpd.conf ファイルに追加してください。

```
<VirtualHost *:80>
ServerName lb.xyz.com
ServerAlias lb.xyz.com
ProxyRequests Off
ProxyPreserveHost On
<Proxy balancer://cluster>
BalancerMember http:// abc.com:8080 route=node1
BalancerMember http:// xyz.com:8080 route=node2
</Proxy>
ProxyPass / balancer://cluster/ stickySession=JSESSIONID|jsessionid
ProxyPassReverse / balancer://cluster/
</VirtualHost>
```

サーバー側の設定

- 1 クラスターの各ノードごとに、次の場所からテキストエディターで、server.xml ファイルを開きます：[appserver root]/server/<profile_name>/deploy/jbossweb.sar
- 2 server.xml ファイルで Engine name 要素を検索し、jvmRoute 属性を追加します。例えば、node1 というノードでは、エレメントを次のように編集します：<Engine name="jboss.web" defaultHost="localhost" jvmRoute="node1">。
- 3 編集した server.xml ファイルを保存します。

注意：各クラスターノードごとに、jvmRoute の値は一意でなければならず、それぞれの JBoss サーバーに接続するワーカーのルートの値と一致していなければなりません (httpd.conf での MalancerMember 設定を参照)。

7.2 AJP コネクタベースのロードバランシング

7.2.1 mod_jk の使用

この設定は、Apache mod_jk プラグインを使用して、AJP プロトコルでの JBoss クラスターのロードバランスを取ります。これは、Apache JServ プロトコル (AJP) に対するサポートを提供します。

次の手順を実行して設定を行います。

- 1 Apache Tomcat Connector サイトから、ご使用のオペレーティングシステム用の mod_jk プラグインをダウンロードします。

注意：ご使用の Apache サーバーが、ダウンロードした mod_jk プラグインファイルをサポートしていることを確認してください。

- 2 ダウンロードしたファイルの名前を mod_jk.so に変更し、APACHE_HOME/modules/ ディレクトリに保存します。
- 3 テキストエディターで、APACHE_HOME/conf にある httpd.conf ファイルを開いて、ファイルの末尾に次の行を追加します。Include conf/mod-jk.conf
- 4 次の内容の新しいファイル APACHE_HOME/conf/mod-jk.conf を作成します。

```
# Load mod_jk module
# Specify the filename of the mod_jk lib
LoadModule jk_module modules/mod_jk.so
# Where to find workers.properties
JkWorkersFile conf/workers.properties
# Where to put jk logs
JkLogFile logs/mod_jk.log
# Set the jk log level [debug/error/info]
JkLogLevel info
# Select the log format
JkLogStampFormat "[%a %b %d %H:%M:%S %Y]"
# JkOptions indicates to send SSK KEY SIZE
JkOptions +ForwardKeySize +ForwardURISCompat -ForwardDirectories
# JkRequestLogFormat
JkRequestLogFormat "%w %V %T"
# Mount your applications
JkMount /* loadbalancer
# You can use external file for mount points.
# It will be checked for updates each 60 seconds.
# The format of the file is: /url=worker
# /examples/*=loadbalancer
#JkMountFile conf/uriworkermap.properties
# Add shared memory.
# This directive is present with 1.2.10 and
# later versions of mod_jk, and is needed
# for load balancing to work properly
JkShmFile logs/jk.shm
# Add jkstatus for managing run-time data
<Location /jkstatus/>
JkMount status
Order deny,allow
Deny from all
Allow from 127.0.0.1
</Location>
```

- 5 conf/workers.properties に次の内容のファイルを作成します。

```
# for mapping requests
worker.list=loadbalancer,status
# Define Node1
# modify the host as your host IP or DNS name.
worker.node1.port=8009
worker.node1.host=node1.mydomain.com
worker.node1.type=ajp13
worker.node1.lbfactor=1
worker.node1.cachesize=10
# Define Node2
# modify the host as your host IP or DNS name.
worker.node2.port=8009
worker.node2.host= node2.mydomain.com
worker.node2.type=ajp13
worker.node2.lbfactor=1
worker.node2.cachesize=10
# Load-balancing behavior
worker.loadbalancer.type=lb
worker.loadbalancer.balance_workers=node1,node2
worker.loadbalancer.sticky_session=1
#worker.list=loadbalancer
# Status worker for managing load balancer
worker.status.type=status
```

6 このファイルで、次の項目を定義します。

- クラスターの各ノード（この例では、2つのノードに `node1`、`node2` という名前が付けられています）
- ファイル内で定義したすべてのノードを `worker.loadbalancer.balance_workers` エントリに追加します。

7 クラスターの各ノードごとに、`[appserver root]/server/<profile_name>/deploy/jbossweb.sar` にある `server.xml` ファイルを開き編集します。

8 `server.xml` ファイルで `Engine name` 要素を検索し、`jvmRoute` 属性を追加します。例えば、`node1` というノードでは次のようになります：`<Engine name="jboss.web" defaultHost="localhost" jvmRoute="node1">`。

9 編集した `server.xml` ファイルを保存します。

注意： `jvmRoute` の各クラスターノードの値は一意でなければならず、それぞれの JBoss に接続するワーカーの値と一致していなければなりません。

7.2.2 mod_proxy_ajp の使用

この設定は、Apache `mod_proxy_ajp` プラグインを使用して、AJP プロトコルでの JBoss クラスターのロードバランスを取ります。このモジュールは、`mod_proxy` のサービスを必要とします。したがって、AJP プロトコルを処理できるようにするために、`mod_proxy` と `mod_proxy_ajp` がサーバーに存在していなければなりません。

このモジュールは、AJP プロトコルを使用してバックエンドアプリケーションサーバーへの逆プロキシのために使われます。使い方は HTTP 逆プロキシと同じですが、これは `ajp://` プレフィクスを使用します。

`APACHE_HOME/conf/httpd.conf` ファイル内で次のモジュール設定をコメント解除します。

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_ajp_module modules/mod_proxy_ajp.so
```

次の設定を `APACHE_HOME/conf/httpd.conf` ファイルに追加します。

```
<VirtualHost *:80>
  ServerName lb.xyz.com
  ServerAlias lb.xyz.com
  ProxyRequests Off
  ProxyPreserveHost On
  <Proxy balancer://cluster>
    BalancerMember ajp://abc.com:8009 route=node1
    BalancerMember ajp://xyz.com:8009 route=node2
    ProxySet lbmethod=byrequests
  </Proxy>
  ProxyPass / balancer://cluster/ stickysession=JSESSIONID|jsessionid
</VirtualHost>
```

通常、ProxyPassReverse ディレクティブは不要です。AJP リクエストはプロキシに与えられたオリジナルのホストヘッダーを含み、アプリケーションサーバーはこのホストに相対的な自己参照ヘッダーを生成すると見なせるので、書き換えは不要です。

サーバー側の設定：

- 1 クラスターの各ノードごとに、[appserver root]/server/<profile_name>/deploy/jbossweb.sar にある server.xml ファイルを開き編集します。
- 2 server.xml ファイルで Engine name 要素を検索し、jvmRoute 属性を追加します。例えば、node1 というノードでは、エレメントを次のように編集します：<Engine name="jboss.web" defaultHost="localhost" jvmRoute="node1">。
- 3 編集した server.xml ファイルを保存します。

注意：各クラスターごとに、jvmRoute のノード値は一意でなければならず、それぞれの JBoss ーに接続するワーカーのルートの値と一致していなければなりません (httpd.conf での MalancerMember 設定を参照)。

第 8 章：高度な実稼働環境の設定

ここでは、Adobe® LiveCycle® Output 11、Adobe® LiveCycle® Forms Standard 11 および PDF Generator の高度なチューニングについて説明します。この節に記載されている作業は、上級アプリケーションサーバー管理者が実稼働システムに対してのみ行ってください。

8.1 Output および Forms のプールサイズの設定

PoolMax の現在のデフォルト値は 4 です。実際に設定する値は、使用環境のハードウェア構成と予想される使用量によって異なります。

最適な使用方法としては、PoolMax の下限を使用可能な CPU の数以上に設定し、上限はサーバーの負荷パターンによって決めることをお勧めします。一般的に、上限はサーバー上にある CPU コアの数に 2 倍に設定します。

既存の PoolMax 値の変更

1 テキストエディターを使用して、JBoss 起動スクリプトを編集します。

2 ConvertPdf の以下のプロパティを追加します。

- com.adobe.convertpdf.bmc.POOL_MAX=[new value]
- com.adobe.convertpdf.bmc.MAXIMUM_REUSE_COUNT=5000
- com.adobe.convertpdf.bmc.REPORT_TIMING_INFORMATION=true
- com.adobe.convertpdf.bmc.CT_ALLOW_SYSTEM_FONTS=true

3 XMLFM の以下のプロパティを追加します。

- com.adobe.xmlform.bmc.POOL_MAX=[new value]
- com.adobe.xmlform.bmc.MAXIMUM_REUSE_COUNT=5000
- com.adobe.xmlform.bmc.REPORT_TIMING_INFORMATION=true
- com.adobe.xmlform.bmc.CT_ALLOW_SYSTEM_FONTS=true

8.2 PDF Generator

PDF Generator では、一部の種類の入力ファイルについて、複数の PDF 変換を同時に行うことができます。これは、ステートレスセッションビーンを使用して実行されます。

8.2.1 EJB プールサイズの設定

以下の種類の入力ファイルについて個別のプールサイズを適用するために、4 種類のステートレスセッションビーンがあります。

- Adobe PostScript® および Encapsulated PostScript (EPS) ファイル
- 画像ファイル (BMP、TIFF、PNG、JPEG ファイルなど)
- OpenOffice ファイル

- Microsoft Office ファイル、Photoshop® ファイル、PageMaker® ファイル、FrameMaker® ファイルなど、その他すべての種類のファイル (HTML ファイルを除く)

HTML から PDF への変換時のプールサイズは、ステートレスセッションビーンでは管理されません。

PostScript および EPS ファイルと画像ファイルのデフォルトのプールサイズは 3 に設定され、OpenOffice とその他の種類のファイル (HTML を除く) のデフォルトのプールサイズは 1 に設定されます。

CPU の数や各 CPU 内のコアの数など、使用しているサーバーハードウェア構成に基づいて、PS/EPS と画像のプールサイズを別の値に設定できます。ただし、PDF Generator を正常に機能させるためには、OpenOffice とその他の種類のファイルのプールサイズを 1 のままにする必要があります。

この節では、サポートされるアプリケーションサーバーのそれぞれについて、PS2PDF (PS から PDF への変換) と Image2PDF (画像から PDF への変換) のプールサイズを設定する方法を説明します。

以下の説明は、次の 2 つの LiveCycle アプリケーション EAR ファイルがアプリケーションサーバーにデプロイされていることを前提としています。

- adobe-livecycle-jboss.ear
- adobe-livecycle-native-jboss-[platform].ear

この [platform] は、オペレーティングシステムに応じて、次のいずれかの文字列に置き換えられます。

- (Windows) x86_win32
- (Linux) x86_linux
- (SunOS™) sparc_sunos

PS2PDF および Image2PDF のプールサイズの設定

LiveCycle 管理ヘルプの「サービスの管理」の節にある、「Distiller サービスの設定」および「Generate PDF サービスの設定」を参照してください。

8.3 Windows での CIFS の有効化

LiveCycle をホストする Windows Server マシンを手動で設定する必要があります。

注意: サーバーには、静的 IP アドレスが必要です。

Windows マシンで、次の作業を行う必要があります。

8.3.1 NetBIOS over TCP/IP の有効化

LiveCycle サーバーに接続するクライアントの要求がサーバーホスト名で解決されるように、NetBIOS over TCP/IP を有効にする必要があります。

- 1 ローカルエリアの接続プロパティダイアログボックスの「全般」タブで、「インターネットプロトコル」を選択して、「プロパティ」をクリックします。
- 2 インターネットプロトコル (TCP/IP) のプロパティダイアログボックスの「全般」タブで、サーバーに静的 IP アドレスがあることを確認します。「詳細設定」をクリックします。
- 3 TCP/IP 詳細設定ダイアログボックスで、「WINS」タブを選択して「NetBIOS over TCP/IP を有効にする」を選択します。

8.3.2 他の IP アドレスの追加

- 1 ローカルエリアの接続プロパティダイアログボックスの「全般」タブで、「インターネットプロトコル」を選択して、「プロパティ」をクリックします。
- 2 インターネットプロトコル (TCP/IP) のプロパティダイアログボックスの「全般」タブで、サーバーに静的 IP アドレスがあることを確認します。「詳細設定」をクリックします。
- 3 TCP/IP 詳細設定ダイアログボックスで、「IP 設定」タブを選択して「追加」をクリックします。
- 4 静的 IP アドレスを指定して「追加」をクリックします。

8.3.3 ファイルとプリンターの共有の無効化 (Windows Server 2008 のみ)

- 「ネットワークの設定」に移動し、「Microsoft ネットワーク用ファイルとプリンター共有」の選択を解除して、「適用」をクリックします。

第9章：付録 - コマンドラインインターフェイスのインストール

9.1 概要

LiveCycle には、インストールプログラム用にコマンドラインインターフェイス (CLI) が用意されています。CLI は、LiveCycle の上級ユーザーが使用したり、インストールプログラムのグラフィカルユーザーインターフェイス (GUI) がサポートされていないサーバー環境で使用したりすることを前提としています。CLI はコンソールモードで実行します。1つのインタラクティブセッションで、すべてのインストール操作を行うことができます。

CLI インストールオプションを使用してモジュールをインストールする前に、該当する準備ガイド (新規のシングルサーバーインストール、クラスターセットアップまたはアップグレード) に従って、LiveCycle の実行に必要な環境の準備が整っていることを確認します。LiveCycle のドキュメント一式は http://www.adobe.com/go/learn_lc_documentation_11_jp から入手できます。

インストール手順の概要については、17 ページの「4.1 事前準備」を参照してください。

インストールプロセスを開始したら、画面の指示に従ってインストールオプションを選択します。各プロンプトに応答しながらインストールを進めてください。

注意：前の手順で選択した内容を変更する場合は、back と入力します。quit と入力すれば、いつでもインストールをキャンセルできます。

9.2 LiveCycle のインストール

1 コマンドプロンプトを開き、実行可能なインストーラーが含まれるインストールメディアまたはハードディスクのフォルダーに移動します。

- (Windows) server¥Disk1¥InstData¥Windows_64¥VM
- (Linux) server/Disk1/InstData/Linux/NoVM
- (Solaris) server/Disk1/InstData/Solaris/NoVM

2 コマンドプロンプトを開いて、次のコマンドを実行します。

- (Windows) install.exe -i console
- (Windows 以外) ./install.bin -i console

注意：-i console オプションを指定せずにコマンドを入力すると、GUI ベースのインストーラーが起動します。

3 次の表の説明に従って、プロンプトに応答します。

プロンプト	説明
Choose Locale	<p>インストールで使用するロケールを値 1 ~ 2 を入力して選択します。デフォルト値を選択するには、Enter キーを押します。</p> <p>English、または日本語を選択できます。デフォルトの言語は日本語です。</p>
Choose Install Folder	<p>Destination 画面で、Enter キーを押してデフォルトディレクトリを使用するか、新しいインストールディレクトリの場所を入力します。</p> <p>デフォルトのインストールフォルダーは次のとおりです。</p> <p>(Windows) : C:\¥Adobe¥Adobe LiveCycle ES4</p> <p>(Windows 以外) : /opt/adobe/adobe_lifecycle_es4</p> <p>ディレクトリ名にアクセント記号付きの文字を使用しないでください。アクセント記号付きの文字を使用すると、CLI によってアクセントが無視され、アクセント記号付きの文字が変更されてからディレクトリが作成されます。</p>
Choose Operating System	<p>(Windows のみ)</p> <p>LiveCycle のインストール先のオペレーティングシステムを選択します。</p> <p>Windows、Linux、Solaris のいずれかを選択できます。Windows (Local) がデフォルトです。</p> <p>異なるターゲットオペレーティングシステムを選択すると、LiveCycle を別のオペレーティングシステムにデプロイするために Windows 上のインストールをステージングプラットフォームとして使用できます。</p>
LiveCycle Server License Agreement	<p>Enter キーを押して、使用許諾契約のページに目を通します。</p> <p>契約に同意する場合は、Y を入力し、Enter キーを押します。</p>
Pre-Installation Summary	<p>選択したインストール内容を確認し、その内容でインストールを続行する場合は Enter キーを押します。</p> <p>前の手順に戻って設定を変更するには、back と入力します。</p>
Ready To Install	<p>インストーラーによってインストールディレクトリが表示されます。</p> <p>Enter キーを押すと、インストールプロセスが開始します。</p>
Installing	<p>インストール中、進行状況バーによりインストールの進行状況が示されます。</p>
Configuration Manager	<p>Enter キーを押すと、LiveCycle のインストールが完了します。</p> <p>Configuration Manager を GUI モードで実行するには、次のスクリプトを呼び出します。</p> <p>(Windows) : C:\¥Adobe¥Adobe LiveCycle ES4¥configurationManager¥bin¥ConfigurationManager.bat</p> <p>(Windows 以外) :</p> <p>/opt/adobe/adobe_lifecycle_es4/configurationManager/bin/ConfigurationManager.sh</p>
Installation Complete	<p>インストールの完了画面にインストールのステータスと場所が表示されます。</p> <p>Enter キーを押すと、インストーラーが終了します。</p>

9.3 エラーログ

エラーが発生した場合は、次のインストールのログディレクトリで `install.log` を確認できます。

- (Windows) [LiveCycle root]¥log
- (Linux、Solaris) [LiveCycle root]/log

インストール中に発生するおそれのあるエラーについて詳しくは、適切なトラブルシューティングガイドを参照してください。

9.4 コンソールモードでの LiveCycle のアンインストール

注意: コマンドラインオプションを使用して LiveCycle をインストールした場合は、コマンドラインからアンインストーラーを実行するだけで LiveCycle ES4 をアンインストールできます。サイレントアンインストールを実行する場合は、「-i console」フラグを省略します。

1 コマンドプロンプトを開き、アンインストールスクリプトが含まれるディレクトリに移動します。

注意: UNIX システムの場合は、ディレクトリ名にスペースが含まれているので、アンインストールスクリプトが含まれるディレクトリには手動で移動する必要があります。

- (Windows) `cd C:\Adobe\Adobe LiveCycle ES4\Uninstall_Adobe LiveCycle ES4`
- (UNIX 系のシステム) `cd /opt/adobe/adobe_livecycle_es4/Uninstall_Adobe Livecycle ES4`

2 プロンプトで次のコマンドを入力し、Enter キーを押します。

- (Windows) `Uninstall Adobe LiveCycle ES4-i console`
- (Linux、Solaris) `./Uninstall Adobe Livecycle ES4 -i console`

3 画面の指示に従って操作します。

プロンプト	説明
LiveCycle ES4 のアンインストール	Enter キーを押すと、アンインストールが続行します。 quit と入力すると、アンインストールプログラムが終了します。
Uninstalling... Uninstall Complete	アンインストールが開始したら、残りのアンインストールプロセスが完了し、カーソルがプロンプトに戻ります。 一部の項目については削除されない可能性があります。また、LiveCycle のインストール後に作成されたフォルダーは削除されません。これらのファイルやフォルダーは手動で削除する必要があります。

第 10 章：付録 - Configuration Manager コマンドラインインターフェイス

LiveCycle には、Configuration Manager のコマンドラインインターフェイス (CLI) が用意されています。CLI は、LiveCycle の上級ユーザーが使用したり、Configuration Manager のグラフィカルユーザーインターフェイス (GUI) がサポートされていないサーバー環境で使用したりすることを前提としています。

10.1 操作の順序

Configuration Manager CLI は、GUI バージョンの Configuration Manager の操作と同じ順序で実行する必要があります。CLI の操作は以下の順序で実行してください。

- 1 LiveCycle を設定します。
- 2 CRX の設定
- 3 設定済みの EAR ファイルを手動でデプロイします
- 4 LiveCycle を初期化します。
- 5 LiveCycle を検証します。
- 6 LiveCycle モジュールをデプロイします。
- 7 LiveCycle モジュールデプロイメントを検証します。
- 8 PDF Generator のシステム準備設定を確認します。
- 9 PDF Generator に管理者ユーザーを追加します。
- 10 Connector for IBM Content Manager を設定します。
- 11 Connector for IBM FileNet を設定します。
- 12 Connector for EMC Documentum を設定します。
- 13 Connector for SharePoint を設定します。

重要： Configuration Manager CLI の操作を完了したら、各クラスターノードを再起動する必要があります。

10.2 コマンドラインインターフェイスのプロパティファイル

プロパティファイルは、インストールの状態に応じて作成する必要があります。次のいずれかの方法を使用します。

- プロパティファイルを作成し、インストールシナリオおよび構成シナリオに応じて値を設定します。
- プロパティファイル `cli_propertyFile_template.txt` をテンプレートとして使用し、使用する Configuration Manager 操作に基づいて値を編集します。
- Configuration Manager の GUI を使用し、GUI バージョンによって作成されたプロパティファイルを CLI バージョンのプロパティファイルとして使用します。[LiveCycle root]/configurationManager/bin/ConfigurationManager.bat/sh ファイルを実行すると、`userValuesForCLI.properties` ファイルが [LiveCycle root]/configurationManager/config ディレクトリに作成されます。このファイルを Configuration Manager CLI の入力として使用できます。

注意：CLI プロパティファイルでは、Windows パスのディレクトリ区切り文字 (¥) にエスケープ文字 (¥) を使用する必要があります。例えば、指定する Fonts フォルダが C:¥Windows¥Fonts である場合、Configuration Manager CLI スクリプトでは C:¥¥Windows¥¥Fonts と入力する必要があります。

注意：次のモジュールは、ALC-LFS-ContentRepository に依存します。cli_propertyFile_template.txt をテンプレートとして使用する場合は、ALC-LFS-ContentRepository を excludedSolutionComponents リストから削除するか、あるいは次の LFS を excludedSolutionComponents リストに追加してください。

- ALC-LFS-ProcessManagement
- ALC-LFS-CorrespondenceManagement
- ALC-LFS-ContentRepository
- ALC-LFS-MobileForms
- ALC-LFS_FormsManager

10.3 一般的な設定プロパティ

10.3.1 共通のプロパティ

共通のプロパティは以下のとおりです。

LiveCycle Server 固有のプロパティ：LiveCycle の初期化および LiveCycle コンポーネントのデプロイ操作に必要です。

以下の操作に必要なプロパティは次の表のとおりです。

- LiveCycle を初期化します。
- LiveCycle コンポーネントをデプロイします。

プロパティ	値	説明
LiveCycle サーバー固有のプロパティ		
LCHost	文字列	LiveCycle をデプロイするサーバーのホスト名。 クラスターデプロイメントの場合、アプリケーションサーバーを実行しているいずれかのクラスターノードのホスト名。
LCPort	整数値	LiveCycle のデプロイ先の Web ポート番号。

プロパティ	値	説明
excludedSolutionComponents	文字列。次の値がサポートされています。 ALC-LFS-Forms、 ALC-LFS-ConnectorEMCDocumentum、 ALC-LFS-ConnectorIBMFileNet、 ALC-LFS-ConnectorIBMContentManager、 ALC-LFS-DigitalSignatures、 ALC-LFS-DataCapture、 ALC-LFS-Output、 ALC-LFS-PDFGenerator、 ALC-LFS-ProcessManagement、 ALC-LFS-ReaderExtensions、 ALC-LFS-RightsManagement ALC-LFS-CorrespondenceManagement、 ALC-LFS-ContentRepository、 ALC-LFS-MobileForms、 ALC-LFS_FormsManager	(オプション) 構成しない LiveCycle モジュールを指定します。構成対象から除外するモジュールが複数ある場合はコンマで区切ります。
includeCentralMigrationService	true: サービスを含める false: サービスを含めない	Central Migration Bridge Service を含めるまたは除外するためのプロパティ。
CRX Content レポジトリ 次のプロパティは、cli_propertyFile_crx_template.txt ファイルで指定されます。		
contentRepository.rootDir		CRX レポジトリのパス。

10.3.2 LiveCycle の構成プロパティ

以下のプロパティは LiveCycle の構成操作にのみ適用されます。

プロパティ	値	説明
AdobeFontsDir	文字列	Adobe サーバーフォントディレクトリの場所。 このパスは、デプロイ先のすべてのクラスターノードからアクセスできるようにする必要があります。
customerFontsDir	文字列	カスタマーフォントディレクトリの場所。 このパスは、デプロイ先のすべてのクラスターノードからアクセスできるようにする必要があります。
systemFontsDir	文字列	システムフォントディレクトリの場所。 このパスは、デプロイ先のすべてのクラスターノードからアクセスできるようにする必要があります。

プロパティ	値	説明
LCTempDir	文字列	一時ディレクトリの場所。 このパスは、デプロイ先のすべてのクラスターノードからアクセスできるようにする必要があります。
LCGlobalDocStorageDir	文字列	グローバルドキュメントストレージのルートディレクトリ。 長期間有効なドキュメントを保存したり、それらをすべてのクラスターノードで共有したりするために使用する、NFS 共有ディレクトリのパスを指定します。 このパスは、デプロイ先のすべてのクラスターノードからアクセスできるようにする必要があります。
EnableDocumentDBStorage	true または false デフォルト : false	永続ドキュメントについて、データベースへのドキュメントの保存を有効または無効にします。 データベースへのドキュメントの保存を有効にしても、GDS のファイルシステムディレクトリは必要です。

10.3.3 アプリケーションサーバーの設定および検証のプロパティ

10.3.3.1 JBoss の設定プロパティ

JBoss Application Server と共に LiveCycle をインストールする場合、JBoss を手動で設定する必要があります。
 『[LiveCycle のインストールの準備 \(シングルサーバー\)](#)』ガイドの JBoss の手動設定に関する節を参照してください。
 LiveCycle DVD で提供されるアドビにより事前設定された JBoss を使用して、インターネットからダウンロードするか JBoss 自動オプションを使用します。

10.3.4 LiveCycle の初期化プロパティ

LiveCycle の初期化プロパティは LiveCycle の初期化操作にのみ適用されます。

プロパティ	値	説明
詳しくは、83 ページの「 10.3.1 共通のプロパティ 」を参照してください		

10.3.5 LiveCycle コンポーネントのデプロイプロパティ

以下の操作に適用されるプロパティは次の表のとおりです。

- LiveCycle コンポーネントのデプロイ
- LiveCycle コンポーネントのデプロイメントの検証
- LiveCycle サーバーの検証

プロパティ	値	説明
LiveCycle サーバー情報セクションを構成する必要があります。詳しくは、「共通のプロパティ」を参照してください		
LCAdminUserID	文字列	LiveCycle 管理者ユーザーに割り当てるユーザー ID。このユーザー ID は、Administration Console へのログオンに使用されます。
LCAdminPassword	文字列	LiveCycle 管理者ユーザーに割り当てるパスワード。このパスワードは、Administration Console へのログオンに使用されます。

10.3.6 PDF Generator 用の管理者ユーザーの追加

以下のプロパティは、PDF Generator 用の管理者ユーザーを追加する場合にのみ適用されます。これらのプロパティは、cli_propertyFile_pdfg_template.txt にあります。

プロパティ	値	説明
LCHost	文字列	LiveCycle サーバーがインストールされるホスト名。
LCPort	整数値	LiveCycle アプリケーションサーバーが構成されているポート番号。
LCAdminUserID	文字列	LiveCycle 管理者ユーザーに割り当てるユーザー ID。このユーザー ID は、Administration Console へのログオンに使用されます。
LCAdminPassword	文字列	LiveCycle 管理者ユーザーに割り当てるパスワード。このパスワードは、Administration Console へのログオンに使用されます。
LCServerMachineAdminUser	文字列	LiveCycle をホストする運用システムの管理者ユーザーのユーザー ID。
LCServerMachineAdminUserPasswd	文字列	LiveCycle をホストする運用システムの管理者ユーザーのパスワード。

10.3.7 Connector for IBM Content Manager の設定

プロパティ	値	説明
LCHost	文字列	LiveCycle サーバーがインストールされるホスト名。
LCPort	整数値	LiveCycle アプリケーションサーバーが構成されているポート番号。
LCAdminUserID	文字列	LiveCycle 管理者ユーザーに割り当てるユーザー ID。このユーザー ID は、Administration Console へのログオンに使用されます。
LCAdminPassword	文字列	LiveCycle 管理者ユーザーに割り当てるパスワード。このパスワードは、Administration Console へのログオンに使用されます。
jndiPortNumber	文字列	LiveCycle アプリケーションサーバーに対応する JNDI ポート。
jboss.clientjar.location	文字列	jbossall-client.jar ファイルの場所 (JBoss のみ)。

プロパティ	値	説明
CDVTopology.appserverrootdir	文字列	リモートサーバー上に設定するアプリケーションサーバーインスタンスのルートディレクトリ (LiveCycle をデプロイするディレクトリ)。
ConfigureIBMCM	true または false	Connector for IBM Content Manager を設定するには、true を指定します。
IBMCMClientPathDirectory	文字列	IBM Content Manager クライアントのインストールディレクトリの場所。
DataStoreName	文字列	接続する IBM Content Manager サーバーのデータストアの名前。
IBMCMUsername	文字列	IBM Content Manager 管理者ユーザーに割り当てるユーザー名。このユーザー ID は、IBM Content Manager へのログインに使用されます。
IBMCMPassword	文字列	IBM Content Manager 管理者ユーザーに割り当てるパスワード。このパスワードは、IBM Content Manager へのログインに使用されます。
ConnectionString	文字列	IBM Content Manager に接続するための接続文字列内に使用される追加の引数 (オプション)。

10.3.8 Connector for IBM FileNet を設定

プロパティ	値	説明
LCHost	文字列	LiveCycle サーバーがインストールされるホスト名。
LCPort	整数値	LiveCycle アプリケーションサーバーが構成されているポート番号。
LCAdminUserID	文字列	LiveCycle 管理者ユーザーに割り当てるユーザー ID。このユーザー ID は、Administration Console へのログインに使用されます。
LCAdminPassword	文字列	LiveCycle 管理者ユーザーに割り当てるパスワード。このパスワードは、Administration Console へのログインに使用されます。
jndiPortNumber	文字列	LiveCycle アプリケーションサーバーに対応する JNDI ポート。
jboss.clientjar.location	文字列	jbossall-client.jar ファイルの場所 (JBoss のみ)。
CDVTopology.appserverrootdir	文字列	リモートサーバー上に設定するアプリケーションサーバーインスタンスのルートディレクトリ (LiveCycle をデプロイするディレクトリ)。
ConfigureFileNetCE	true または false	Connector for IBM FileNet を設定するには、true を指定します。
FileNetConfigureCEVersion	文字列	設定する FileNet クライアントのバージョン。FileNetClientVersion4.0 または FileNetClientVersion5.0 を指定します。
FileNetCEClientPathDirectory	文字列	IBM FileNet Content Manager クライアントのインストールディレクトリの場所。
ContentEngineName	文字列	IBM FileNet Content Engine がインストールされているマシンのホスト名または IP アドレス。

プロパティ	値	説明
ContentEnginePort	文字列	IBM FileNet Content Engine が使用するポート番号。
CredentialProtectionSchema	CLEAR または SYMMETRIC	保護のレベルを指定します。
EncryptionFileLocation	文字列	暗号化ファイルの場所。これは、CredentialProtectionSchema 属性に対して SYMMETRIC オプションを選択した場合にのみ必要です。 パス区切り文字には、スラッシュ (/) または二重の円記号 (\\) を使用します。
DefaultObjectStore	文字列	Connector for IBM FileNet Content Server のオブジェクトストアの名前。
FileNetContentEngineUsername	文字列	IBM FileNet Content Server に接続するためのユーザー ID。 読み取りアクセス権限を持つユーザー ID では、デフォルトのオブジェクトストアへの接続が許可されます。
FileNetContentEnginePassword	文字列	IBM FileNet ユーザーに割り当てるパスワード。このパスワードは、デフォルトのオブジェクトストアに接続する際に使用されます。
ConfigureFileNetPE	true または false	Connector for IBM FileNet を設定するには、true を指定します。
FileNetPEClientPathDirectory	文字列	IBM FileNet クライアントのインストールディレクトリの場所。
FileNetProcessEngineHostname	文字列	プロセスルーターのホスト名または IP アドレス。
FileNetProcessEnginePortNumber	整数値	IBM FileNet Content Server のポート番号。
FileNetPERouterURLConnectionPoint	文字列	プロセスルーターの名前。
FileNetProcessEngineUsername	文字列	IBM FileNet Content Server に接続するためのユーザー ID。
FileNetProcessEnginePassword	文字列	IBM FileNet Content Server に接続するためのパスワード。

10.3.9 Connector for EMC Documentum の設定

プロパティ	値	説明
LCHost	文字列	LiveCycle サーバーがインストールされるホスト名。
LCPort	整数値	LiveCycle アプリケーションサーバーが構成されているポート番号。
LCAdminUserID	文字列	LiveCycle 管理者ユーザーに割り当てるユーザー ID。このユーザー ID は、Administration Console へのログオンに使用されます。
LCAdminPassword	文字列	LiveCycle 管理者ユーザーに割り当てるパスワード。このパスワードは、Administration Console へのログオンに使用されます。
jndiPortNumber	文字列	LiveCycle アプリケーションサーバーに対応する JNDI ポート。

プロパティ	値	説明
jboss.clientjar.location	文字列	jbossall-client.jar ファイルの場所 (JBoss のみ)。
CDVTopology.appserverrootdir	文字列	リモートサーバー上に設定するアプリケーションサーバーインスタンスのルートディレクトリ (LiveCycle をデプロイするディレクトリ)。
ConfigureDocumentum	true または false	Connector for EMC Documentum を設定するには、true を指定します。
DocumentumClientVersion	文字列	設定する EMC Documentum クライアントのバージョン。DocumentumClientVersion6.5 または DocumentumClientVersion6.0 を指定します。
DocumentumClientPathDirectory	文字列	EMC Documentum クライアントのインストールディレクトリの場所。
ConnectionBrokerHostName	文字列	EMC Documentum Content Server のホスト名または IP アドレス。
ConnectionBrokerPortNumber	文字列	EMC Documentum Content Server のポート番号。
DocumentumUsername	文字列	EMC Documentum Content Server に接続するためのユーザー ID。
DocumentumPassword	文字列	EMC Documentum Content Server に接続するためのパスワード。
DocumentumDefaultRepositoryName	文字列	MC Documentum Content Server のデフォルトリポジトリの名前。

10.3.10 Connector for Microsoft SharePoint の設定

プロパティ	値	説明
LCHost	文字列	LiveCycle サーバーがインストールされるホスト名。
LCPort	整数値	LiveCycle アプリケーションサーバーが構成されているポート番号。
LCAdminUserID	文字列	LiveCycle 管理者ユーザーに割り当てるユーザー ID。このユーザー ID は、Administration Console へのログオンに使用されます。
LCAdminPassword	文字列	LiveCycle 管理者ユーザーに割り当てるパスワード。このパスワードは、Administration Console へのログオンに使用されます。
jndiPortNumber	文字列	LiveCycle アプリケーションサーバーに対応する JNDI ポート。
jboss.clientjar.location	文字列	jbossall-client.jar ファイルの場所 (JBoss のみ)。
CDVTopology.appserverrootdir	文字列	リモートサーバー上に設定するアプリケーションサーバーインスタンスのルートディレクトリ (LiveCycle をデプロイするディレクトリ)。
ConfigureSharePoint	true または false	Connector for Microsoft SharePoint を設定するには、true を指定します。
SharePointServerAddress	文字列	SharePoint Server のホスト名または IP アドレス。
SharePointUsername	文字列	SharePoint Server に接続するためのユーザー ID。

プロパティ	値	説明
SharePointPassword	文字列	SharePoint Server に接続するためのパスワード。
SharePointDomain	文字列	SharePoint Server のドメイン名。
ConnectionString	文字列	SharePoint Server に接続するための接続文字列内に使用される追加の引数 (オプション)。

10.3.11 コマンドラインインターフェイスの使用

プロパティファイルを設定したら、[LiveCycle root]/configurationManager/bin フォルダに移動する必要があります。

Configuration Manager CLI のコマンドの詳細な説明を表示するには、ConfigurationManagerCLI help<command name> と入力します。

10.3.11.1 CRX CLI の使用の設定

CRX リポジトリの設定では、次の構文を使用する必要があります。

```
configureCRXRepository -f <propertyFile>
```

説明：

- -f<propertyFile>：必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。

10.3.11.2 設定済みの EAR ファイルの手動デプロイ

設定済みの EAR ファイルを手動でデプロイするための詳細手順は、JBoss アプリケーションサーバーへのデプロイを参照してください。

10.3.11.3 LiveCycle の初期化 CLI の使用

LiveCycle の初期化操作では、次の構文を使用する必要があります。

```
initializeLiveCycle -f <propertyFile>
```

説明：

- -f<propertyFile>：必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。

10.3.11.4 LiveCycle サーバーの検証 CLI の使用

LiveCycle サーバーの検証操作 (オプション) では、次の構文を使用する必要があります。

```
validateLiveCycleServer -f <propertyFile> -LCAdminPassword <password>
```

説明：

- -f<propertyFile>：必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。
- -LCAdminPassword <password>：コマンドライン上で管理者パスワードを設定できます。この引数を指定すると、プロパティファイルの targetServer.adminPassword プロパティが上書きされます。

10.3.11.5 LiveCycle コンポーネントのデプロイ CLI の使用

LiveCycle コンポーネントのデプロイ操作では、次の構文を使用する必要があります。

deployLiveCycleComponents -f <propertyFile> -LCAdminPassword <password>

説明:

- -f <propertyFile>: 必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。
- -LCAdminPassword <password>: コマンドライン上で管理者パスワードを設定できます。この引数を指定すると、プロパティファイルの targetServer.adminPassword プロパティが上書きされます。

10.3.11.6 LiveCycle コンポーネントのデプロイメントの検証 CLI の使用

LiveCycle コンポーネントのデプロイメントの検証操作 (オプション) では、次の構文を使用する必要があります。

validateLiveCycleComponentDeployment -f <propertyFile> -LCAdminPassword <password>

説明:

- -f <propertyFile>: 必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。
- -LCAdminPassword <password>: コマンドライン上で管理者パスワードを設定できます。この引数を指定すると、プロパティファイルの targetServer.adminPassword プロパティが上書きされます。

10.3.11.7 PDF Generator のシステム準備設定を確認します。

PDF Generator のシステム準備設定の確認操作では、次の構文を使用する必要があります。

pdfg-checkSystemReadiness

10.3.11.8 PDF Generator の管理者ユーザーの追加

PDF Generator の管理者ユーザーの追加操作では、次の構文を使用する必要があります。

pdfg-addAdminUser -f <propertyFile>

説明:

- -f <propertyFile>: 必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。

10.3.11.9 Connector for IBM Content Manager の設定

Connector for IBM Content Manager の設定操作 (オプション) では、次の構文を使用する必要があります。

IBMCM-configurationCLI -f <propertyFile>

説明:

- -f <propertyFile>: 必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。

重要: [LiveCycle root]¥configurationManager¥bin¥ ディレクトリにある cli_propertyFile_ecm_ibmcm_template.txt という名前の <propertyFile> を修正します。

- 1 [LiveCycle root]/configurationManager/configure-ecm/jboss の **adobe-component-ext.properties** ファイルを次の [appserver root]/bin ディレクトリにコピーします。
- 2 アプリケーションサーバーの再起動
- 3 LiveCycle 管理コンソールから以下のサービスを開始します。
 - IBMCMAuthProviderService

- IBMCMConnectorService

10.3.11.10 Connector for IBM FileNet の設定

Connector for IBM FileNet の設定操作 (オプション) では、次の構文を使用する必要があります。

```
filenet-configurationCLI -f <propertyFile>
```

説明:

- -f <propertyFile>: 必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。

重要: [LiveCycle root]¥configurationManager¥bin¥ ディレクトリにある cli_propertyFile_ecm_filenet_template.txt という名前の <propertyFile> を修正します。

Connector for IBM Content Manager の設定を完了するには、次の手順を手動で実行してください。

- 1 [LiveCycle root]/configurationManager/configure-ecm/jboss の **adobe-component-ext.properties** ファイルを次の [appserver root]/bin ディレクトリにコピーします。
- 2 [appserver root]/server/[profile]/conf フォルダ内の login-config.xml ファイルを見つけ、そのファイルに [LiveCycle root]/configurationManager/configure-ecm/jboss ディレクトリにある login-config.xml ファイルの内容を追加します。
デフォルトの JBoss 設定では、[profile] 値は「all」です。ただし、アドビ用に設定されている JBoss では [lc_DatabaseName] (例 lc_mysql、lc_oracle など) を使用します。
- 3 logkit.jar ファイルを [appserver root]/client から次の [appserver root]/server/[profile]/lib ディレクトリにコピーします。デフォルトの JBoss 設定では、[profile] 値は「all」です。ただし、アドビ用に設定されている JBoss では [lc_DatabaseName] (例 lc_mysql、lc_oracle など) を使用します。
- 4 (FileNet 4.x の場合のみ) Java オプション -Dwaspl.location=[FileNetClient root]/wsi をアプリケーションサーバー起動オプションに追加します。
- 5 アプリケーションサーバーを再起動します。
- 6 LiveCycle 管理コンソールから以下のサービスを開始します。
 - IBMFileNetAuthProviderService
 - IBMFileNetContentRepositoryConnector
 - IBMFileNetRepositoryProvider
 - IBMFileNetProcessEngineConnector (設定されている場合)

10.3.11.11 Connector for EMC Documentum の設定

Connector for EMC Documentum の設定操作 (オプション) では、次の構文を使用する必要があります。

```
documentum-configurationCLI -f <propertyFile>
```

説明:

- -f <propertyFile>: 必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。

重要: [LiveCycle root]¥configurationManager¥bin¥ ディレクトリにある cli_propertyFile_ecm_documentum_template.txt という名前の <propertyFile> を修正します。

Connector for EMC Documentum の設定を完了するには、次の手順を手動で実行してください。

- 1 [LiveCycle root]/configurationManager/configure-ecm/jboss の **adobe-component-ext.properties** ファイルを次の [appserver root]/bin ディレクトリにコピーします。
- 2 アプリケーションサーバーの再起動
- 3 LiveCycle 管理コンソールから以下のサービスを開始します。
 - EMCDocumentumAuthProviderService
 - EMCDocumentumRepositoryProvider
 - EMCDocumentumContentRepositoryConnector

10.3.11.12 Connector for Microsoft SharePoint の設定

Connector for Microsoft SharePoint の設定操作 (オプション) では、次の構文を使用する必要があります。

```
sharepoint-configurationCLI -f <propertyFile>
```

説明：

- -f <propertyFile>：必要な引数が含まれるプロパティファイル。プロパティファイルの作成について詳しくは、「コマンドラインインターフェイスのプロパティファイル」を参照してください。

重要： [LiveCycle root]¥configurationManager¥bin¥ ディレクトリにある cli_propertyFile_ecm_sharepoint_template.txt という名前の <propertyFile> を修正します。

10.4 使用例

C:¥Adobe¥Adobe LiveCycle ES4¥configurationManager¥bin から次のコマンドを入力します。

```
ConfigurationManagerCLI configureLiveCycle -f cli_propertyFile.txt
```

cli_propertyFile.txt には、作成済みのプロパティファイルの名前を指定します。

10.5 Configuration Manager CLI のログ

エラーが発生した場合は、[LiveCycle root]¥configurationManager¥log フォルダーにある CLI ログで確認できます。生成されるログファイルには、命名規則に基づいて lcmCLI.0.log のような名前が付けられます。ファイル名の数字 (ここでは 0) は、ログファイルがロールオーバーされるたびに増加します。

10.6 次の手順

Configuration Manager CLI を使用して LiveCycle の設定とデプロイを行った場合は、この時点で次のタスクを実行できます。

- デプロイメントの確認 (34 ページの「[6.1.5 デプロイメントの確認](#)」を参照)
- Administration Console へのアクセス (35 ページの「[6.1.5.1 LiveCycle 管理コンソールへのアクセス](#)」を参照)。
- LDAP にアクセスするための LiveCycle モジュールの構成 (55 ページの「[6.8 LDAP アクセスの設定](#)」を参照)。

第 11 章：付録 - Windows サービスとしての JBoss の設定

この付録では JBoss Application Server を Windows サービスとして起動するための、JBoss Web Native Connectors を使用した設定方法を説明します。Windows Server 2008 の 32 ビットおよび 64 ビットバージョンで、この手順を使用してください。

11.1 Web Native Connector のダウンロード

- 1 「JBoss Web Native Connectors - Current packages」のダウンロードページで、Windows 用の JBoss Web Native Connector をダウンロードします。使用している Windows のバージョンに応じて、次のいずれかのファイルをダウンロードします。
 - (64 ビット) : <http://download.jboss.org/jbossweb/2.0.8.GA/jboss-native-2.0.8-windows-x64-ssl.zip>
 - (32 ビット) : <http://download.jboss.org/jbossweb/2.0.8.GA/jboss-native-2.0.8-windows-x86-ssl.zip>
 - 2 ZIP ファイルを解凍し、¥bin フォルダにある ¥native フォルダ以外の内容をすべて JBoss インストールフォルダの ¥bin フォルダにコピーします。
 - 3 service.bat ファイルをテキストエディターで開き、変数を更新します。
サービス名 (SVCNAME)、サービスディスプレイ名 (SVCDISP) およびサービスの説明 (SVCDESC) の変数を、JBoss 環境を反映した値に更新する必要があります。例えば JBoss のバージョンが 5.1 の場合は、次のように入力します。

```
set SVCNAME=JBAS51SVC  
  
set SVCDISP=JBossAS 5.1 for Adobe LiveCycle ES4  
  
set SVCDESC=JBoss Application Server 5.1 GA/ Platform: Windows x64
```
 - 4 :cmdStart セクションで call run.bat 行を探し、次のように設定名とバインド IP アドレス (サーバーのすべての IP アドレスにバインドする場合は 0.0.0.0) を追加します。

```
call run.bat -c <profilename> -b 0.0.0.0 <.r.lock >> run.log 2>&1
```
 - 5 :cmdRestart セクションで、手順 4 の編集を繰り返し行います。

```
call run.bat -c <profilename> -b 0.0.0.0 <.r.lock >> run.log 2>&1
```
 - 6 ファイルを保存して閉じます。
- 注意：**手順 4 および 5 で JBoss クラスター引数を指定し、JBoss インスタンスをクラスターに含めます。JBoss クラスター引数についての詳細は、7 ページの「[3.3 クラスターでの JBoss の実行](#)」を参照してください。

11.2 Windows サービスのインストール

- 1 JBoss の \bin フォルダーから次のコマンドを実行して Windows サービスを作成します。

```
service.bat install
```

コマンドが正常に実行されると、次のような応答が返されます。

```
Service JBossAS 5.1 for Adobe LiveCycle ES4 installed
```

- 2 Windows のコントロールパネルの「サービス」アプレットに、作成したサービスが **JBossAS 5.1 for Adobe LiveCycle ES4** と表示されているのを確認します。これは service.bat ファイルの SVCDISP 変数に設定した値です。
- 3 Windows のコントロールパネルの「サービス」アプレットで、「スタートアップの種類」を「自動」に設定します。
- 4 (オプション)「回復」タブで、「最初のエラー」に「サービスを再起動する」を、「次のエラー」に「コンピューターを再起動する」を設定します。

注意: 必要に応じて、「ログオン」の値をデフォルトの「ローカルシステムアカウント」から他のユーザーまたはサービスアカウントに変更できます。

11.3 Windows サービスとしての JBoss Application Server の開始および停止

Windows サービスとしての JBoss の開始

- ❖ Windows サーバーで、**スタート/コントロールパネル/管理ツール/サービス**を選択し、JBoss Application Server 用の Windows サービスを選択して、「**開始**」をクリックします。

注意: JBoss Application Server を Windows サービスとして開始すると、コンソールの出力結果は run.log ファイルにリダイレクトされます。このファイルを調べると、サービスの開始時に発生したエラーを確認できます。

Windows サービスとしての JBoss の停止

- ❖ Windows サーバーで、**スタート/コントロールパネル/管理ツール/サービス**を選択し、JBoss Application Server 用の Windows サービスを選択して、「**停止**」をクリックします。

注意: JBoss Application Server を Windows サービスとして停止すると、コンソールの出力結果は run.log ファイルにリダイレクトされます。このファイルを調べると、サービスのシャットダウン時に発生したエラーを検出できます。

11.4 インストールの確認

- 1 Windows のコントロールパネルの「サービス」アプレットでサービスを起動します。
- 2 [appserver root]\%<profile_name>%\logs\server.log ファイルの末尾を監視し、サービスが正常に起動することを確認します。
- 3 Windows のコントロールパネルの「サービス」アプレットでサービスをシャットダウンし、サービスが正常にシャットダウンすることを確認します。
- 4 Windows のコントロールパネルの「サービス」アプレットで、サービスが再起動できることを確認します。

11.5 追加の設定

上記の手順に加え、Windows のコントロールパネルの「サービス」アプレット、または Windows に組み込まれているサービス設定ユーティリティ (sc) のいずれかを使用して、追加の設定を実行できます。

例えば Microsoft SQL Server をデータベースとして使用し、データベースサービスが同じマシンのインスタンスで起動する場合、次のコマンドでデータベースサービスへの依存関係を作成できます。

```
sc config JBAS51SVC depend= MSSQL$MYSERVER
```

MSSQL\$MYSERVER 変数の値を、同じサーバーインスタンスで起動する Microsoft SQL Server 2005 サービスのサービス名に更新します。

注意： = の前にはスペースがなく、 = の後にスペースがあることを確認してください。

コマンドが正常に実行されると、次のような応答が返されます。

```
[SC] ChangeServiceConfig SUCCESS
```

第 12 章：付録 - JBoss の手動設定

この付録では、Red Hat からダウンロードできる JBoss EAP で必要な設定について説明します。このオプションは、詳細インストールの場合にのみ検討してください。通常、JBoss の高度な知識が必要です。

LiveCycle ES3 は、Windows Server 2008 R1 / R2 (Enterprise または Standard Edition)、Red Hat Linux ES / AS 5.5 または 6、SUSE Linux ES 11 のプラットフォームおよび Solaris 10 の JBoss 上で動作します。

12.1 JBoss 用 JDK のインストール

Oracle JDK 6.0 Update 31 (またはバージョン 6.0 の以降の更新) は、<http://www.oracle.com/technetwork/java/javase/downloads/index.html> からダウンロードおよびインストールする必要があります。

JDK がインストールされている場所を示す JAVA_HOME 環境変数を作成または設定します。

12.1.1 JAVA_HOME 環境変数の設定 (Windows)

- 1 スタート/コントロールパネル/システムを選択します。
- 2 「詳細設定」タブをクリックします。
- 3 「環境変数」をクリックし、「システム環境変数」で、「新規」をクリックします。
- 4 新しいシステム変数ダイアログボックスで、変数名として JAVA_HOME を入力し、JDK をインストールしたディレクトリを入力します。このディレクトリは、/bin サブディレクトリを含むディレクトリです。例えば、次のパスを入力します。

```
C:\Program Files\Java\jdk1.6.0_31
```

12.1.2 PATH 環境変数の設定 (Windows)

- 1 スタート/コントロールパネル/システムを選択します。
- 2 「詳細設定」タブをクリックして、「環境変数」をクリックします。
- 3 「システム環境変数」領域で、「PATH」変数を選択して、「編集」をクリックします。
- 4 変数値の先頭に次のテキストを追加します。

```
%JAVA_HOME%\bin;
```

12.1.3 JAVA_HOME 環境変数の設定 (Linux および Solaris)

- 次の例に示すように、Bourne および Bash で JAVA_HOME 変数を設定することをお勧めします。

```
JAVA_HOME=/usr/java
export JAVA_HOME
```

12.1.4 PATH 環境変数の設定 (Linux および Solaris)

- 次の例に示すように、Bourne および Bash の PATH 変数を設定します。

```
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

12.1.5 JAVA_HOME 環境変数の設定の検証 (Windows、Linux または Solaris)

(オプション) コマンドプロンプトを開いて、次のコマンドを実行します。

```
java -version
```

このコマンドによって、システムにインストールされている Java のバージョンが返されます。

12.2 JBoss の手動インストール

JBoss EAP は、<http://www.jboss.org/jbossas/downloads/> からダウンロードしてインストールできます。

12.3 JBoss の起動と停止

この付録で説明する一部の手順では、製品をデプロイする JBoss のインスタンスを停止して起動する必要があります。

JBoss のすべての起動設定は **[appserver root]/server** ディレクトリにあります。Red Hat から直接入手した JBoss の場合は、JBoss で提供されている起動設定のいずれか (all、default または minimal)、またはカスタム設定を使用します。

12.3.1 JBoss の起動

1 コマンドプロンプトで **[appserver root]/bin** に移動します。

2 次のコマンドを入力して、アプリケーションサーバーを起動します。

- (Windows) `run.bat -c [profile_name] -b [server_IP_Address]`
- (Linux および Solaris) `./run.sh -c [profile_name] -b [server_IP_Address]`

[profile_name] はデータベースに必要な設定、[server_IP_Address] はサーバーの IP アドレスに置き換えます。

12.3.2 JBoss の停止

1 コマンドプロンプトで **[appserver root]/bin** に移動します。

2 次のコマンドを入力して、アプリケーションサーバーを停止します。

- (Windows) `shutdown.bat -s [server_IP_Address]:<jndi -port>`
- (Linux および Solaris) `./shutdown.sh -s [server_IP_Address]:<jndi -port>`

12.4 JBoss 設定の変更

JBoss Application Server は様々な XML 設定ファイルを使用して設定されます。これらの設定ファイルのいずれかを編集するためには、事前に JBoss を停止する必要があります。JBoss の実行中にこれらのファイルを変更すると、JBoss で障害が発生する可能性があります。JBoss には、.property ファイルという形式の設定ファイルがいくつかあります。.property ファイルを Windows 環境で編集する場合は、.property ファイルが常に Linux または Solaris 上の UNIX テキストファイルとして保存されている必要があります。

シングルサーバーインストールの場合、[appserver root]\server\standard\ にある jboss プロファイルをテンプレートとして使用することもできます。クラスターインストールの場合、[appserver root]\server\all\ にある jboss プロファイルをテンプレートとして使用します。

プロファイル (all または standard) のコピーを作成して、コピーしたプロファイルに変更を加えることをお勧めします。

12.4.1 JMS および設定ファイルの削除 (オプション)

LiveCycle では JMS 設定を使用しないので、JBoss に含まれる以下のファイルとディレクトリを削除できます。以下に示すディレクトリとその内容の全体を削除してください。

注意： LiveCycle Foundation JMS サービスを有効にする必要がある場合や、JMS サービスに依存する他のサービスやアプリケーションをデプロイする必要がある場合は、この手順をスキップしてください。

- [appserver root]\server\\deploy\messaging
- [appserver root]\server\\deploy\jms-ra.rar

12.4.2 JBoss 設定の変更

JBoss 設定を変更して LiveCycle 用に JBoss をカスタマイズするには、次の手順を実行します。

- jacorb.properties ファイルを更新します。
- JBoss server.xml ファイル内の URI エンコードの更新
- EAR ファイルのクラスローディングの分離の変更
- run.conf.bat ファイルを変更します (Windows)。
- run.conf (Linux および Solaris) を変更します。
- log4j.xml を変更します。
- jbossts-properties.xml ファイルを変更します。
- セッション Cookie のパスの変更

12.4.2.1 jacorb.properties ファイルの更新

- 1 [appserver root]/server/<profile_name>/conf/jacorb.properties ファイルをテキストエディターで開きます。
- 2 [appserver root]/server/<profile_name>/conf/jacorb.properties ファイルをテキストエディターで開きます。
- 3 jacorb.poa.thread_pool_max という設定を見つけて、値を 16 に変更します。
- 4 ファイルを保存して閉じます。

12.4.2.2 JBoss server.xml ファイル内の URI エンコードの更新

- 1 [appserver root]/server/<profile_name>/deploy/jbossweb.sar/server.xml ファイルをテキストエディターで開きます。

- 2 次の行を探します。

```
<Connector protocol="HTTP/1.1" port="8080" address="${jboss.bind.address}" connectionTimeout="20000"
redirectPort="8443" />
```

- 3 URIEncoding="UTF-8" を末尾に追加します。

```
<Connector protocol="HTTP/1.1" port="8080" address="${jboss.bind.address}" connectionTimeout="20000"
redirectPort="8443" URIEncoding="UTF-8" />
```

- 4 ファイルを保存して閉じます。

12.4.2.3 EAR ファイルのクラスローディングの分離の変更

- 1 [appserver root]/server/<profile_name>/deployers/ejb-deployer-jboss-beans.xml ファイルをエディターで開きます。
- 2 <property name="CallByValue"> の値を true に変更します。
- 3 ファイルを保存して閉じます。
- 4 [appserver root]/server/<profile_name>/deployers/ear-deployer-jboss-beans.xml ファイルをエディターで開きます。
- 5 <property name="isolated"> の値を true に変更します。
- 6 <property name="CallByValue">false</property> の値を true に変更します。
- 7 ファイルを保存して閉じます。
- 8 [appserver root]/server/<profile_name>/conf/jboss-service.xml ファイルをエディターで開きます。
- 9 <attribute name="CallByValue">false</attribute> の値を true に変更します。
- 10 ファイルを保存して閉じます。

12.4.2.4 run.conf.bat ファイルの変更 (Windows のみ)

- 1 [appserver root]/bin/run.conf.bat ファイルをエディターで開きます。
- 2 次の行の太字のテキストを削除し、必要に応じて 32 ビットおよび 64 ビット JVM のメモリ引数を追加します。

```
set "JAVA_OPTS=-Xms1303m -Xmx1303m -XX:MaxPermSize=256m -Dorg.jboss.resolver.warning=true -
Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -
Dsun.lang.ClassLoader.allowArraySyntax=true"
```

- 32 ビット JVM のメモリ引数:

```
set "JAVA_HEAP_ARGS=-Xms1024m -Xmx1024m -XX:PermSize=128m -XX:MaxPermSize=192m"
```

- 64 ビット JVM のメモリ引数:

```
set "JAVA_HEAP_ARGS=-Xms1024m -Xmx2048m -XX:PermSize=256m -XX:MaxPermSize=512m -
XX:+UseCompressedOops"
```

- 3 次の行を追加します。

```
set "JAVA_OPTS=%JAVA_OPTS% %JAVA_HEAP_ARGS%"
```

- 4 32 ビットおよび 64 ビット JVM の次のメモリ引数を追加します。

```
set "JAVA_OPTS=%JAVA_OPTS% -Dadobeidp.serverName=server1 -Dfile.encoding=utf8 -
Djava.net.preferIPv4Stack=true"
set "JAVA_OPTS=%JAVA_OPTS% -DentityExpansionLimit=10000"
set "JAVA_OPTS=%JAVA_OPTS% -XX:+HeapDumpOnOutOfMemoryError"
```

- 5 (オプション) IPv6 モードで実行するように次のように JBoss Application Server を変更します。

- -Djava.net.preferIPv4Stack=false のように変更します。

- 文字列 `-Djava.net.preferIPv6Stack=true` を挿入します。

注意: アプリケーションサーバーログで起動時に次のエラーが記録される場合は、IPv6 スタックの値を削除して、IPv4 の値を true に戻します。

```
"13:37:44,488 WARN [HANamingService] Failed to start AutomaticDiscovery java.net.SocketException: bad
argument for IP_MULTICAST_IF: address not bound to any interface at
java.net.PlainDatagramSocketImpl.socketSetOption(Native Method)at
java.net.PlainDatagramSocketImpl.setOption(PlainDatagramSocketImpl.java:260)"
```

- 6 ファイルを保存して閉じます。

12.4.2.5 run.conf ファイルの変更 (JBoss、Solaris 10、Red Hat 5.5、64 ビット専用)

Sun の Solaris JDK では、64 ビット機能を使用するために追加の引数が必要です。この設定を変更しないと、Sun JDK では既定で 32 ビットしかサポートされません。

注意: ルートユーザー以外で JBoss を実行している場合、`-Djava.io.tmpdir="location"` を使用して、一時ディレクトリの場所をアクセス権があるディレクトリに設定します。

- 1 `[appserver root]/bin/run.conf` ファイルをエディターで開きます。

- 2 次の行で始まるセクションを探します。

```
if [ "x$JAVA_OPTS" = "x" ]; then
```

このセクションを次のように変更します。

```
if [ "x$JAVA_OPTS" = "x" ]; then
    #JAVA_OPTS="-Xms1303m -Xmx1303m -XX:MaxPermSize=256m -Dorg.jboss.resolver.warning=true -
Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -
Dsun.lang.ClassLoader.allowArraySyntax=true"
    JAVA_OPTS="$JAVA_OPTS -Xms1024m -Xmx2048m -XX:MaxPermSize=512m -Dorg.jboss.resolver.warning=true -
Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -
Dsun.lang.ClassLoader.allowArraySyntax=true"
    JAVA_OPTS="$JAVA_OPTS -Dadobeidp.serverName=server1 -Dfile.encoding=utf8 -
Djava.net.preferIPv4Stack=true"
    JAVA_OPTS="$JAVA_OPTS -DentityExpansionLimit=10000"
    JAVA_OPTS="$JAVA_OPTS -XX:+UseCompressedOops -XX:+HeapDumpOnOutOfMemoryError"
```

注意: このエントリが `run.conf` ファイル内に 1 行で表示されることを確認してください。

- 3 (オプション) IPv6 モードで実行するように次のように JBoss Application Server を変更します。

- `-Djava.net.preferIPv4Stack=false` のように変更します。
- `-Djava.net.preferIPv6Stack=true` を追加します。

- 4 ファイルを保存して閉じます。

12.4.2.6 log4j.xml ファイルの変更

- 1 `[appserver root]/server/<profile_name>/conf/jboss-log4j.xml` ファイルをエディターで開きます。

- 2 FILE appender セクションで次のテキストを見つけて、太字で表示されている行を追加し、太字で表示されている値を変更します。

```
<appender name="FILE" class="org.jboss.logging.appender.DailyRollingFileAppender">
  <errorHandler class="org.jboss.logging.util.OnlyOnceErrorHandler"/>
  <param name="File" value="{jboss.server.home.dir}/log/server.log"/>
  <param name="Append" value="false"/>
```

- 3 Limit categories セクションを見つけて、以下を追加します。

```
<category name="jacob.config">
  <priority value="ERROR"/>
</category>
<category name="com.adobe">
  <priority value="INFO"/>
</category>
<category name="org.springframework">
  <priority value="WARN"/>
</category>
<!-- Limit the verbose MC4J EMS (lib used by admin-console) categories -->
<category name="org.mc4j.ems">
  <priority value="WARN"/>
</category>
```

- 4 ファイルを保存して閉じます。

12.4.2.7 jbossts-properties.xml ファイルの変更：

- 1 [appserver root]/server/<profile_name>/conf ディレクトリにある jbossts-properties.xml ファイルを探して、エディターで開き、トランザクション管理が期待どおり機能することを確認します。
- 2 プロパティエレメントを探して次のように変更します (太字で示してあります)。太字のプロパティが jbossts-properties.xml ファイルにない場合は、このプロパティを追加します。

```
<properties depends="arjuna" name="jta">
  <property name="com.arjuna.ats.jta.allowMultipleLastResources" value="true"/>
  <!-- ... other properties ... -->
</properties>
```

- 3 メッセージングを使用しない場合は、次の行を見つけ、コメントアウトします。

```
<property
  name="com.arjuna.ats.jta.recovery.XAResourceRecovery.JBMESSAGING1" value="org.jboss.jms.server.recovery
  .MessagingXAResourceRecovery;java:/DefaultJMSProvider"/>
```

- 4 ファイルを保存して閉じます。

注意：詳しくは、JBoss の記事 11443 (<http://www.jboss.org/community/docs/DOC-11443>) を参照してください。

12.4.2.8 jmx-invoker-service.xml ファイルの変更

- 1 編集のために [アプリケーションサーバールート]/server/<プロファイル名>/deploy/jmx-invoker-service.xml ファイルを開き、次の行をコメントにします。

```
<interceptor code="org.jboss.jmx.connector.invoker.AuthenticationInterceptor"
  securityDomain="java:/jaas/jmx-console"/>
```

- 2 ファイルを保存して閉じます。

12.4.2.9 セッション Cookie のパスの変更

- 1 次の場所で context.xml ファイルを探し、エディターで開きます。

```
[appserver root]/server/all/deploy/jbossweb.sar
```

- 2 次の行を探します。

```
<InstanceListener>org.jboss.web.tomcat.security.RunAsListener</InstanceListener>
```

- 3 上記の行の後に、次のテキストを追加します。

```
<SessionCookie path="/" />
```

- 4 ファイルを保存して閉じます。

12.5 jar ファイルのコピー

次の JAR ファイルを [DVD ルート]/third_party/jboss.zip/server/lc_<db 名>/_cl/lib から、ダウンロードした JBoss の [アプリサーバールート]/server/< プロファイル名 >/lib ディレクトリにコピーします。

- certjFIPS.jar
- cglib.jar
- hibernate-annotations.jar
- hibernate-entitymanager.jar
- hibernate3.jar
- jsafeFIPS.jar
- jsafeJCEFIPS.jar

12.6 手動でインストールした JBoss 用の LiveCycle データベース接続

LiveCycle データベース接続を設定するには、次のタスクを実行する必要があります。

- LiveCycle データソースを設定します。
- データベースをデフォルトのデータソースとして使用するよう JBoss を設定します。

アプリケーションサーバーのインストールディレクトリにデータベースドライバーをインストールする必要があります。ドライバーは、Configuration Manager およびアプリケーションサーバーが LiveCycle データベースに接続できるようにするのに必要です。データベースに使用するデータベースの種類に対応するドライバーをインストールします。

データベースに接続するデータソースを設定する必要があります。JBoss の場合は、MySQL、Oracle または SQL Server データソースを設定できます。

注意：以下の操作に進む前に、JBoss が実行されていないことを確認してください。

12.6.1 手動でインストールした JBoss での MySQL の設定

JBoss から LiveCycle データを格納している MySQL データベースへの接続を有効にするには、次のタスクを実行する必要があります。

- MySQL JDBC ドライバーを取得して、LiveCycle をデプロイする JBoss のインスタンスにコピーします。
- データソースファイルを作成し、LiveCycle をデプロイする JBoss のインスタンスにデプロイします。
- <http://community.jboss.org/wiki/EncryptingDataSourcePasswords> で説明されているいずれかの方法を使用して、データソースファイル (adobe-ds.xml および mysql-ds.xml) および login-config.xml ファイルのパスワードを暗号化します。 http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_cleararte.html の手順を使用することもできます。

12.6.1.1 MySQL データソースの設定

MySQL データソースを設定する前に、データベースを MySQL に作成しておく必要があります (MySQL データベースの作成を参照)。

12.6.1.2 MySQL データベースドライバのインストール

- インストール DVD の [DVD_root]/third_party/db/mysql ディレクトリから [appse ver root]/server/<profile_name>/lib ディレクトリに、mysql-connector-java-5.1.6-bin.jar ドライバーファイルをコピーします。
- インストール DVD の [DVD_root]/third_party/db/mysql ディレクトリから [appse ver root]/server/<profile_name>/lib ディレクトリに、mysql-connector-java-5.1.6-bin.jar ドライバーファイルをコピーします。

12.6.1.3 adobe-ds.xml ファイルの編集

- 1 インストール DVD の [DVD_root]/third_party/additional/datasources/lc_mysql/deploy ディレクトリから [appse ver root]/server/<profile_name>/deploy ディレクトリに adobe-ds.xml ファイルをコピーします。

- 2 adobe-ds.xml ファイルをテキストエディターで開き、次の行を見つけます。

```
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>  
<driver-class>com.mysql.jdbc.Driver</driver-class>  
<user-name>adobe</user-name>  
<password>adobe</password>
```

- 3 次の値をデータベースに固有の値に置き換えます。

- localhost : データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは localhost です。
- 3306 : データベースへのアクセスに使用するポート。デフォルトのポートは 3306 です。
- adobe : LiveCycle データを格納しているデータベースの名前。デフォルト値の adobe を、独自のデータベース名に更新する必要があります。

- 4 <connection-url> 設定に続く行で、user-name および password 設定を探し、デフォルト値をアプリケーションサーバーがデータベースへのアクセスに使用するユーザー名とパスワードに置き換えます。

- 5 データソース接続の最小値と最大値を次のように変更します。

- IDP_DS :

```
<min-pool-size>1</min-pool-size>  
<max-pool-size>30</max-pool-size>
```

- EDC_DS :

```
<min-pool-size>1</min-pool-size>  
<max-pool-size>20</max-pool-size>
```

注意 : LiveCycle サーバーで処理する負荷が大きい場合は、JDBC 接続の最大数を増やして、すべてのジョブが確実に処理されるようにします。そのような場合は、IDP_DS と EDC_DS の両方で <max-pool-size> を 50 以上に増やします。

- 6 ファイルを保存して閉じます。

12.6.1.4 データソースとしての MySQL の設定

- 1 [appserver root]/server/<profile_name>/deploy ディレクトリに移動し、hsqldb-ds.xml ファイルを削除します。
- 2 [appserver root]/server/<profile_name>/deploy ディレクトリに移動し、hsqldb-ds.xml ファイルを削除します。
- 3 [appserver root]/docs/examples/jca ディレクトリに移動し、[appserver root]/server/<profile_name>/deploy ディレクトリに mysql-ds.xml ファイルをコピーします。
- 4 [appserver root]/docs/examples/jca ディレクトリに移動し、[appserver root]/server/<profile_name>/deploy ディレクトリに mysql-ds.xml ファイルをコピーします。

- 5 **[appserver root]/server/<profile_name>/deploy/mysql-ds.xml** ファイルをテキストエディターで開き、<local-tx-datasource> エレメントを MySQL の接続設定の内容に変更します。

```
<jndi-name>MySQLDS</jndi-name>
<connection-url>jdbc:mysql://mysql-hostname:3306/jbossdb</connection-url>
<driver-class>com.mysql.jdbc.Driver</driver-class>
<user-name>x</user-name>
<password>y</password>
```

- 6 太字の値をデータベースに固有の次の値に置き換えます。

- **MySQLDS** : DefaultDS に変更します。
- **mysql-hostname**, **3306**, **jbossdb**, **x** および **y** : アプリケーションサーバーがデータベースへのアクセスに使用するデータベースの値。

- 7 <local-tx-datasource> セクションに次の行を追加します (まだ存在しない場合)。

```
<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
```

- 8 ファイルを保存して閉じます。

12.6.1.5 login-config.xml ファイルの編集

- 1 **[appserver root]\server\<profile_name>\conf\login-config.xml** ファイルをテキストエディターで開き、<policy> エレメント内に次のテキストを追加します。

```
<application-policy name="MySQLDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.ConfiguredIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=DefaultDS </module-option>
    </login-module>
  </authentication>
</application-policy>
```

- 2 太字の値をデータベースに固有の次の値に置き換えます。

- 3 ファイルを保存して閉じます。

- 4 JBoss を起動します。

12.6.2 手動でインストールした JBoss 用の Oracle の設定

JBoss から LiveCycle データを格納している Oracle データベースへの接続を有効にするには、LiveCycle 製品を手動でデプロイする場合、次のタスクを実行する必要があります。

- Oracle JDBC ドライバーを取得して、LiveCycle をデプロイする JBoss のインスタンスにコピーします。
- データソースファイルを作成し、LiveCycle をデプロイする JBoss のインスタンスにデプロイします。
- <http://community.jboss.org/wiki/EncryptingDataSourcePasswords> で説明されているいずれかの方法を使用して、データソースファイル (adobe-ds.xml および oracle-ds.xml) と login-config.xml ファイルのパスワードを暗号化します。
http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearite.html の手順を使用することもできます。

12.6.2.1 Oracle 11g データベースドライバのインストール

[LiveCycle root]/lib/db/oracle ディレクトリから [appserver root]/server/<profile_name>/lib ディレクトリに、JDK 1.6 ドライバーファイル用の ojdbc6.jar をコピーします。JDBC Driver Downloads サイトから Oracle 11g ドライバーをダウンロードすることもできます。サポートされている Oracle 11g ドライバーのバージョンについては、[サポートされているプラットフォームの組み合わせ](#)を参照してください。

12.6.2.2 adobe-ds.xml ファイルの編集

1 [DVD_root]/third_party/additional/datasources/lc_oracle/deploy ディレクトリの adobe-ds.xml ファイルを、[appserver root]/server/<profile_name>/deploy ディレクトリにコピーします。

2 adobe-ds.xml ファイルをテキストエディターで開き、次の行を見つけます。

```
<connection-url>jdbc:oracle:thin:@localhost:1521:adobe</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<user-name>adobe</user-name>
<password>adobe</password>
```

3 次の値をデータベースに固有の値に置き換えます。

- **localhost** : データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは localhost です。
- **1521** : データベースへのアクセスに使用するポート。デフォルトのポートは 1521 です。
- **adobe** : デフォルト値 adobe をデータベース SID に変更します。

4 <connection-url> 設定に続く行で、**user-name** および **password** 設定を探し、デフォルト値をアプリケーションサーバーがデータベースへのアクセスに使用するユーザー名とパスワードに置き換えます。

5 (Oracle RAC でのみ) 手順 2 で示されている接続 URL を次の接続 URL に置き換えます。

```
jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=broken) (ADDRESS_LIST=(ADDRESS= (PROTOCOL=TCP) (HOST=yourhost1) (PORT=1521)) (ADDRESS= (PROTOCOL=TCP) (HOST=yourhost2) (PORT=1521)) (LOAD_BALANCE=on) (FAILOVER=on)) (CONNECT_DATA=(SERVER=dedicated) (SERVICE_NAME=service.yourcompany.com) (FAILOVER_MODE=(TYPE=session) (METHOD=basic) (RETRIES=10) (DELAY=3))))
```

注意 : このエントリが adobe-ds.xml ファイル内に 1 行で表示されることを確認してください。

6 (Oracle RAC でのみ) 手順 5 で示されている接続 URL に含まれる次のテキストをデータベースに固有の値に置き換えます。

- **yourhost1** : データベースをホストするクラスター内のプライマリノードの名前、IP アドレスまたは完全修飾ドメイン名。
- **yourhost2** : データベースをホストするクラスター内のセカンダリノードの名前、IP アドレスまたは完全修飾ドメイン名。

注意 : データベースをホストするクラスターは任意の数のノードで構成することができます。**yourhost1** および **yourhost2** は、クラスターが 2 つのノードで構成される場合の例です。

- **service.yourcompany.com** : Oracle RAC データベースのサービス名。

7 データソース接続の最小値と最大値を次のように変更します。

- **IDP_DS** :

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

- **EDC_DS** :

```
<min-pool-size>1</min-pool-size>
<max-pool-size>20</max-pool-size>
```

注意： LiveCycle サーバーで処理する負荷が大きい場合は、JDBC 接続の最大数を増やして、すべてのジョブが確実に処理されるようにします。そのような場合は、IDP_DS と EDC_DS の両方で <max-pool-size> を 50 以上に増やします。

8 ファイルを保存して閉じます。

12.6.2.3 データソースとしての Oracle の設定

Oracle データベースと共に LiveCycle を実行する場合は、Oracle を JBoss のデフォルトのデータソースに設定する必要があります。この手順では、Oracle JDBC ドライバーが **[appserver root]/server/<profile_name>/lib** ディレクトリにインストールされていることを前提としています。

- 1 **[appserver root]/server/<profile_name>/deploy** ディレクトリに移動し、**hsqldb-ds.xml** ファイルを削除します。
- 2 **[appserver root]/docs/examples/jca** ディレクトリの **oracle-ds.xml** ファイルを、**[appserver root]/server/<profile_name>/deploy** ディレクトリにコピーします。
- 3 **[appserver root]/server/<profile_name>/deploy/oracle-ds.xml** ファイルをテキストエディターで開き、<local-tx-datasource> エレメントを Oracle の接続設定の内容に変更します。

```
<jndi-name>OracleDS</jndi-name>
<connection-url>jdbc:oracle:thin:@youroraclehost:1521:yourSID</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<user-name>x</user-name>
<password>y</password>
```

- 4 太字の値をデータベースに固有の次の値に置き換えます。
 - **OracleDS** : この値を DefaultDS に変更します。
 - **youroraclehost** : この値を Oracle サーバーのホスト名に置き換えます。
 - **1521** : Oracle でデフォルトのポートを使用していない場合は、この値を適切なポート番号に変更します。
 - **yourSID** : この値を Oracle システム識別子に置き換えます。
- 5 <connection-url> 設定に続く行で、**user-name** および **password** 設定を探し、デフォルト値をアプリケーションサーバーがデータベースへのアクセスに使用するユーザー名とパスワードに置き換えます。

6 (Oracle RAC でのみ) 手順 3 で示されている接続設定を次の接続 URL に置き換えます。

```
jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=broken) (ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP) (HOST=yourhost1) (PORT=1521)) (ADDRESS=(PROTOCOL=TCP) (HOST=yourhost2) (PORT=1521)) (LOAD_BALANCE=on) (FAILOVER=on)) (CONNECT_DATA=(SERVER=dedicated) (SERVICE_NAME=service.yourcompany.com) (FAILOVER_MODE=(TYPE=session) (METHOD=basic) (RETRIES=10) (DELAY=3))))
```

注意： このエントリが **oracle-ds.xml** ファイル内に 1 行で表示されることを確認してください。

- 7 (Oracle RAC でのみ) 手順 6 で示されている接続 URL に含まれる次のテキストをデータベースに固有の値に置き換えます。
 - **yourhost1** : データベースをホストするクラスター内のプライマリノードの名前、IP アドレスまたは完全修飾ドメイン名。
 - **yourhost2** : データベースをホストするクラスター内のセカンダリノードの名前、IP アドレスまたは完全修飾ドメイン名。
注意： データベースをホストするクラスターは任意の数のノードで構成することができます。**yourhost1** および **yourhost2** は、クラスターが 2 つのノードで構成される場合の例です。
 - **service.yourcompany.com** : Oracle RAC データベースのサービス名。

8 ファイルを保存して閉じます。

12.6.2.4 login-config.xml ファイルの編集

- 1 **[appserver root]/server/<profile_name>/conf/login-config.xml** ファイルをテキストエディターで開き、<policy> エレメント内に次のテキストを追加します。

```
<application-policy name = "OracleDbRealm">
  <authentication>
    <login-module code =
      "org.jboss.resource.security.ConfiguredIdentityLoginModule" flag =
      "required">
      <module-option name = "principal">adobe</module-option>
      <module-option name = "userName">adobe</module-option>
      <module-option name = "password">adobe</module-option>
      <module-option name = "managedConnectionFactoryName">
        jboss.jca:service=LocalTxCM,name=DefaultDS</module-option>
    </login-module>
  </authentication>
</application-policy>
```

- 2 太字の値をデータベースに固有の次の値に置き換えます。
- 3 ファイルを保存して閉じます。
- 4 JBoss を起動します。

12.6.3 手動でインストールした JBoss 用の SQL Server の設定

JBoss から LiveCycle データを格納している SQL Server データベースへの接続を有効にするには、次のタスクを実行する必要があります。

- SQL Server JDBC ドライバーファイルを取得して、LiveCycle をデプロイする JBoss のインスタンスにコピーします。
- SQL Server データソースファイルを作成し、LiveCycle をデプロイする JBoss のインスタンスにデプロイします ([appserver_root]/server/<profile_name>/deploy など)。
- <http://community.jboss.org/wiki/EncryptingDataSourcePasswords> で説明されているいずれかの方法を使用して、データソースファイル (adobe-ds.xml および mssql-ds.xml) および login-config.xml ファイルのパスワードを暗号化します。 http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clear.html の手順を使用することもできます。

12.6.3.1 SQL Server データベース接続の設定

SQL Server データソースを設定する前に、LiveCycle データベースを SQL Server に作成しておく必要があります (SQL Server データベースの作成を参照)。

12.6.3.2 JBoss 用 SQL データベースドライバーのインストール

- 1 Microsoft の Web サイトから SQL Server JDBC 3.0 データベースドライバーを取得します。

注意: Microsoft SQL Server 2005 SP2 および Microsoft SQL Server 2008 のどちらのサーバーにも SQL Server JDBC ドライバー 3.0 を使用します。

- 2 (Windows) *.exe ファイルをダウンロードして実行し、これらのファイルを一時ディレクトリに抽出します (この節の以降の部分では、[SQL_root] ディレクトリと呼びます)。
- 3 (Linux) *.tar.gz ファイルを一時ディレクトリに抽出します (この節の以降の部分では、[SQL_root] ディレクトリと呼びます)。
- 4 [SQL_root]/sqljdbc_3.0/enu ディレクトリから [appserver root]/server/<profile_name>/lib ディレクトリに sqljdbc.jar ファイルをコピーします。

- 5 **[appserver root]/server/<profile_name>/lib** ディレクトリにある `mysql-connector-java-3.1.12-bin.jar` ファイルを削除します。

12.6.3.3 adobe-ds.xml ファイルの編集

- 1 **[DVD_root]/third_party/additional/datasources/lc_sqlserver/deploy** ディレクトリの `adobe-ds.xml` ファイルを、**[appserver root]/server/<profile_name>/deploy** ディレクトリにコピーします。
- 2 `adobe-ds.xml` ファイルをテキストエディターで開き、`<local-tx-datasource>` エlement を SQL Server の接続設定の内容に変更します。

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>  
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>  
<user-name>adobe</user-name>  
<password>adobe</password>
```

- 3 次の値をデータベースに固有の値に置き換えます。
 - `localhost` : データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは `localhost` です。
 - `1433` : データベースへのアクセスに使用するポート。
 - `adobe` : LiveCycle データを格納しているデータベースの名前。デフォルト値の `adobe` を、独自のデータベース名に更新する必要があります。

- 4 `<driver-class>` Element を次のように変更します。

```
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
```

- 5 `<driver-class>` 設定に続く行で、`user-name` および `password` 設定を探し、デフォルト値をアプリケーションサーバーがデータベースへのアクセスに使用するユーザー名とパスワードに置き換えます。データソース接続の最小値と最大値を次のように変更します。

- `IDP_DS` :

```
<min-pool-size>1</min-pool-size>  
<max-pool-size>30</max-pool-size>
```

- `EDC_DS` :

```
<min-pool-size>1</min-pool-size>  
<max-pool-size>20</max-pool-size>
```

注意 : LiveCycle サーバーで処理する負荷が大きい場合は、JDBC 接続の最大数を増やして、すべてのジョブが確実に処理されるようにします。そのような場合は、`IDP_DS` と `EDC_DS` の両方で `<max-pool-size>` を 50 以上に増やします。

- 6 ファイルを保存して閉じます。

12.6.3.4 mssql-ds.xml ファイルの編集

- 1 **[appserver root]/server/<profile_name>/deploy** ディレクトリに移動し、`hsqldb-ds.xml` ファイルを削除します。
- 2 **[appserver root]/docs/examples/jca** ディレクトリに移動し、**[appserver root]/server/<profile_name>/deploy** ディレクトリに `mysql-ds.xml` ファイルをコピーします。
- 3 `mssql-ds.xml` ファイルをテキストエディターで開き、`<local-tx-datasource>` Element を SQL Server の接続設定の内容に変更します (これらの Element は連続した行とは限りません)。

```
<jndi-name>MSSQLDS</jndi-name>
<datasource-mapping>MS SQLSERVER2000</datasource-mapping>
<connection-url>jdbc:sqlserver://localhost:1433; DatabaseName=MyDatabase</connection-url>
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver </driver-class>
<user-name>x</user-name>
<password>y</password>
```

- <jndi-name> エlementを DefaultDS に変更
- <datasource-mapping> エlementを MS SQLSERVER2000 に変更
- データベース名 MyDatabase を独自のデータベース名に変更
- <user-name>x エlementを独自のユーザー名に変更
- <password>y エlementを独自のパスワードに変更

注意: MS SQL Server 2005 を使用していても、<datasource-mapping> エlementでは常に MS SQLSERVER2000 を指定する必要があります。

- 4 ファイルを保存して閉じます。

12.6.3.5 login-config.xml ファイルの編集

- 1 [appserver root]/server/<profile_name>/conf/login-config.xml ファイルをテキストエディターで開き、<policy> エlement内に次の行を追加します。

```
<application-policy name = "MSSQLDbRealm">
  <authentication>
    <login-module code =
      "org.jboss.resource.security.ConfiguredIdentityLoginModule" flag = "required">
      <module-option name = "principal">adobe</module-option>
      <module-option name = "userName">adobe</module-option>
      <module-option name = "password">adobe</module-option>
      <module-option name =
        "managedConnectionFactoryName">jboss.jca:service=LocalTxCM,name= DefaultDS </module-option>
    </login-module>
  </authentication>
</application-policy>
```

- 2 太字の値をデータベースに固有の次の値に置き換えます。
- 3 ファイルを保存して閉じます。
- 4 JBoss を起動します。

12.6.3.6 Windows での統合セキュリティの設定

- 1 次の例に示すように、[appserver root]\server\<profile_name>\deploy にある adobe-ds.xml および mssql-ds.xml ファイルを変更し、integratedSecurity=true を接続 URL に追加します。

```
<connection-url>jdbc:sqlserver://<serverhost>:<port>; databaseName=<dbname>;integratedSecurity=<true></connection-url>
```

- 2 JBoss を実行しているコンピューターの Windows システムパス (C:\¥Windows) に sqljdbc_auth.dll ファイルを追加します。sqljdbc_auth.dll ファイルは、Microsoft SQL JDBC 3.0 ドライバーのインストールフォルダー内にあります。デフォルトの場所は、[SQL_root]/sqljdbc_3.0/enu/auth/x86 (32 ビットオペレーティングシステムの場合) および [SQL_root]/sqljdbc_3.0/enu/auth/x64 (64 ビットオペレーティングシステムの場合) です。
- 3 JBoss for Adobe LiveCycle ES3 サービスまたは設定した JBoss サービスのプロパティを開き、「ログオン」タブをクリックします。
- 4 「アカウント」を選択し、有効なユーザーアカウントの値を入力します。コマンドラインから JBoss を実行する場合は、この変更は必要ありません。

- 5 SQL Server のセキュリティを混合モードから Windows 認証のみに変更します。

12.7 次の手順

『[LiveCycle アプリケーションサーバークラスターの設定 \(JBoss 版\)](#)』に記載されている指示に従って JBoss クラスターの LiveCycle を設定します