

Adobe® LiveCycle® ES4 の インストールの準備 (シングルサーバー)

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 このガイドで使用する表記	1
1.2 追加情報	3

第 2 章：必要システム構成

2.1 サーバー環境の準備	4
2.2 必要システム構成	4
2.3 LDAP の設定	20
2.4 グローバルドキュメントストレージディレクトリ	20
2.5 インストールに関する考慮事項	21
2.6 LiveCycle IPv6 サポート	22
2.7 Central Migration Bridge サービス	25

第 3 章：LiveCycle の秘密鍵証明書と公開鍵証明書

3.1 Reader Extensions 使用権限秘密鍵証明書の取得	27
3.2 Digital Signatures で使用する電子証明書の取得	27

第 4 章：LiveCycle データベースの作成

4.1 データベース設定の要件	29
4.2 データベースを使用するための最小限のユーザー権限	30
4.3 Oracle データベースの作成	30
4.4 SQL Server データベースの作成	31
4.5 DB2 データベースの作成	35
4.6 MySQL データベースの作成	38

第 5 章：JBoss Application Server の設定

5.1 設定オプション	41
5.2 JBoss 用 JDK のインストール	42
5.3 JBoss の起動と停止	43
5.4 アドビの事前設定 JBoss のデータベース接続の設定	44
5.5 次の手順	51

第 6 章：WebSphere Application Server の設定

6.1 WebSphere のインストール	52
6.2 WebSphere Application Server の準備	55
6.3 LiveCycle データベース接続の設定	57
6.4 グローバルセキュリティを有効にする場合の WebSphere Application Server の設定	57

第 7 章：WebLogic Server の設定

7.1 WebLogic 用の LiveCycle の要件	59
7.2 WebLogic Server のインストール	59
7.3 JDK のインストール	59
7.4 WebLogic Server ドメインの作成	61
7.5 WebLogic 管理対象サーバー	62
7.6 WebLogic の停止と再起動	64
7.7 データベースの初期化に関する JMX ポリシーの作成	65
7.8 LiveCycle データベース接続の設定	66

第 8 章：事前設定環境のチェックリスト**第 9 章：付録 - JBoss の手動設定**

9.1 JBoss 用 JDK のインストール	69
9.2 JBoss の手動インストール	70
9.3 JBoss の起動と停止	70
9.4 JBoss 設定の変更	71
9.5 jar ファイルのコピー	74
9.6 手動でインストールした JBoss 用の LiveCycle データベース接続	75
9.7 次の手順	82

第 1 章：このドキュメントの内容

Adobe® LiveCycle® Enterprise Suite 4 (ES4) 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 Adobe® LiveCycle® Foundation 11。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 このガイドで使用する表記

一般的なファイルパスに対して、次の命名規則を使用しています。

名前	説明	デフォルト値
[LiveCycle root]	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、Adobe® LiveCycle® Configuration Manager、LiveCycle SDK およびインストールされた各 LiveCycle モジュールが (製品ドキュメントと共に) サブディレクトリに分けられて格納されています。このディレクトリには、サードパーティのテクノロジーに関連したディレクトリも含まれます。	Windows : C:\Adobe\Adobe LiveCycle ES4\ AIX, Linux, and Solaris: /opt/adobe/adobe_livecycle_es4/
[appserver root]	LiveCycle に含まれてい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。	JBoss (Windows) : C:\jboss JBoss (Linux および Solaris) : /opt/jboss JBoss Enterprise Application Platform (Windows) : C:\jboss-eap-<バージョン>\jboss-as JBoss Enterprise Application Platform (Linux および Solaris) : /opt/jboss-eap-<バージョン>/jboss-as WebSphere (Windows) : C:\Program Files\IBM\WebSphere\AppServer WebSphere (Linux および Solaris) : /opt/IBM/WebSphere/AppServer WebSphere (AIX) : /usr/IBM/WebSphere/AppServer、または /opt/IBM/WebSphere/AppServer WebLogic (Windows) : C:\Oracle\Middleware\wlserver_<バージョン> WebLogic (Linux および Solaris) : /opt/Oracle/Middleware/wlserver_<バージョン>
[WL_Home]	WL_HOME 環境変数に指定されている、WebLogic のインストールディレクトリ。	WebLogic (Windows) : C:\Oracle\Middleware\ WebLogic (Linux および Solaris) : /opt/Oracle/Middleware
[appserverdomain]	WebLogic で設定したドメイン。デフォルトのドメインは base_domain です。	WebLogic (Windows) : C:\Oracle\Middleware\user_projects\domains\base_domain WebLogic (Linux および Solaris) : /opt/Oracle/Middleware/user_projects/domains/base_domain
[WebSphere ND root]	WebSphere Application Server Network Deployment のインストールディレクトリ。	WebSphere (Windows) : C:\Program Files\IBM\WebSphere\AppServer WebSphere (Linux および Solaris) : /opt/IBM/WebSphere/AppServer WebSphere (AIX) : /usr/IBM/WebSphere/AppServer

名前	説明	デフォルト値
[server_name]	WebLogic サーバーまたは WebSphere サーバー上に設定されたサーバーの名前。	WebLogic : server1 WebSphere : server1
[profile_name]	JBoss Application Server のプロファイル名。	アドビにより事前設定された JBoss : lc_<db-name> 手動設定またはダウンロードした JBoss : standard
[dbserver root]	データベースサーバーがインストールされている場所。	データベースの種類とインストール時の設定によって異なります。

このドキュメント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 (AIX、Linux および Solaris 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。

1.2 追加情報

次の表に、LiveCycle の詳細について参照できるリソースを示します。

情報	参照先
一般的な LiveCycle およびモジュール	LiveCycle の概要
LiveCycle に統合できる他のサービスや製品	LiveCycle デベロッパーセンター
LiveCycle のすべてのドキュメント	LiveCycle ドキュメント

第 2 章：必要システム構成

2.1 サーバー環境の準備

サーバー環境を準備するために次のタスクを実行します。

- 1 インストールメディアが損傷していないことを確認します。インストールメディアから直接インストールするのではなく、LiveCycle をインストールする予定のコンピューターのハードディスクにインストーラーファイルをコピーすることをお勧めします。
また、インストールメディアの MD5 チェックサムが、[アドビの Web サイト](#)に公開されているチェックサムと一致していることを確認します。
- 2 4 ページの「[2.2 必要システム構成](#)」を参照して、ソフトウェアインフラストラクチャが要件に合致していることを確認します。
- 3 「必要システム構成」を参照して、ハードウェアインフラストラクチャが要件に従って配置されていることを確認します。
- 4 オペレーティングシステムをインストールし、必要なすべてのパッチとサービスパックを適用して更新します。
- 5 データベースサーバーをインストールおよび設定します。
- 6 アプリケーションサーバーをインストールおよび設定します。
- 7 対象のコンピューターに、4 ページの「[2.2.1 最小ハードウェア要件](#)」に示されている最小必要量以上の RAM が搭載されていることを確認します。

2.2 必要システム構成

[ここをクリックして](#)、クラスターデプロイメントの必要システム構成を表示します。

2.2.1 最小ハードウェア要件

次の表に、LiveCycle でサポートされる最小ハードウェア要件を示します。

オペレーティングシステム	最小ハードウェア要件
Microsoft Windows Server® 2008 R1/R2 (64 ビットアーキテクチャ)	Intel® Pentium® 4、2.8 GHz または同等のプロセッサ VMWare ESX 3.0 以降 RAM : 4 GB (64 ビット OS および 64 ビット JVM) ディスクの空き容量: 5.4 GB の一時ディスクスペースと LiveCycle 用の 3.4 GB のディスクスペース
Sun Solaris 10 (64 ビットアーキテクチャ)	UltraSPARC® IIIi、1.5 GHz プロセッサ Solaris コンテナ (ゾーン) パーティション RAM : 4 GB (64 ビット OS および 64 ビット JVM) ディスクの空き容量: 5.4 GB の一時ディスクスペースと LiveCycle 用の 3.4 GB のディスクスペース
IBM AIX 6.1, 7.1	P5 pSeries 520 (Model 52A) 9131-52A、1.8 GHz プロセッサ LPAR 分割 RAM : 4 GB (64 ビット OS および 64 ビット JVM) ディスクの空き容量: 5.4 GB の一時ディスクスペースと LiveCycle 用の 3.4 GB のディスクスペース
SUSE Linux Enterprise Server 11.0 (64 ビット版のみ)	デュアルコア 1 GHz プロセッサ VMWare ESX 3.0 以降 RAM : 4 GB (64 ビット OS および 64 ビット JVM) ディスクの空き容量: 5.4 GB の一時ディスクスペースと LiveCycle 用の 3.4 GB のディスクスペース
Red Hat Enterprise Linux AP または ES 5.5 または 6 (64 ビット版のみ)	デュアルコア 1 GHz プロセッサ VMWare ESX 3.0 以降 RAM : 4 GB (64 ビット OS および 64 ビット JVM) ディスクの空き容量: 5.4 GB の一時ディスクスペースと LiveCycle 用の 3.4 GB のディスクスペース

2.2.1.1 Intel x86 の互換性

サポート対象の Windows および UNIX 系環境において、LiveCycle では、サポート対象の 32 ビットまたは 64 ビットのいずれかのオペレーティングシステムを実行している Intel および AMD64 互換チップセットがサポートされています。

注意: 32 ビットのプラットフォームは、開発および評価の目的でのみサポートされています。

2.2.1.2 推奨ハードウェア要件

小規模な実稼働環境の場合:

Intel 環境: Pentium 4、2.8 GHz 以上。デュアルコアプロセッサを使用するとパフォーマンスがさらに向上します。

Sun SPARC 環境: UltraSPARC V 以降。

IBM AIX 環境: Power4 以降。

メモリ要件: 4 GB の RAM。

注意: 同じマシン上で Correspondence Management Solution の発行インスタンスを実行する場合は、2 GB の RAM を追加することをお勧めします。

2.2.1.3 クライアント側コンピューターの推奨ハードウェア要件

Adobe® LiveCycle® Workbench 11

- インストール用のディスク容量：

1.5 GB (Workbench のみの場合) 1.7 GB (Workbench、Adobe® LiveCycle® Designer 11、およびサンプルアセンブリのフルインストールを単一ドライブで行う場合)、400 MB (一時的インストールディレクトリ)、200 MB (ユーザー一時ディレクトリ)、および 200 MB (Windows 一時ディレクトリ)

注意：これらの場所がすべて 1 つのドライブ上にある場合は、インストール時に 1.5 GB の空き容量が必要です。一時ディレクトリにコピーされるファイルは、インストールが完了すると削除されます。

- Workbench を実行するためのメモリ：2 GB の RAM
- ハードウェア要件：Intel® Pentium® 4 または AMD の同等の 1 GHz プロセッサ
- 1024 X 768 ピクセル以上のモニター解像度、16 ビットカラー以上
- LiveCycle サーバーに対する TCP/IPv4 または TCP/IPv6 ネットワーク接続

注意：Workbench を Windows にインストールするには、管理者権限が必要です。管理者以外のアカウントを使用してインストールする場合は、適切なアカウントの資格情報が求められます。

Adobe® LiveCycle® Designer 11

- 最低 Adobe® Acrobat® 9 Professional、Acrobat 9 Standard、または Adobe Reader® 9。Designer 11 の新機能を十分に活用するには、Acrobat XI Professional、Acrobat XI Standard、または Reader XI が必要です。
- Adobe® Flash® Player 11.1 以降。
- (オプション) Adobe® Flash® Builder® 4.5 以降。
- (オプション) Adobe® Flex® SDK 4.1 (Designer に付属のフォームガイドコンポーネントをカスタマイズするために必要)。

注意：Designer インストーラーに含まれている Flex SDK 4.1 のみを使用してください。アドビの Web サイトから入手したバージョンの Flex SDK 以外は使用しないでください。

エンドユーザーのハードウェア要件

- Adobe® LiveCycle® Workspace 11：1 GB の RAM (Adobe Flash と Adobe Reader の要件を含む)
- Adobe Flash Player 11.1 以降：512 MB の RAM (1 GB 推奨)
- Adobe Reader 9 以降：128 MB の RAM (256 MB 推奨)

注意：詳細は、[Adobe LiveCycle ES4 サポートプラットフォーム](#)を参照してください。

2.2.2 サポートされているプラットフォーム

オペレーティングシステム、アプリケーションサーバー、データベース、JDK などのサポートされている組み合わせについて詳しくは、「[Adobe LiveCycle ES4 でサポートされているプラットフォーム](#)」を参照してください。

2.2.3 Linux および Solaris に関するその他の要件

注意：Linux および Solaris マシンでは、LiveCycle インストーラーはマシンにインストールされている JDK を使用します。そのため、サポートされている JDK バージョンをインストールしてください。他のオペレーティングシステムでは、インストーラーはインストーラーにバンドルされている JVM を使用します。

2.2.3.1 UTF-8 のインストールおよび設定

AIX、Linux または Solaris オペレーティングシステムに LiveCycle をインストールする場合、US English バージョンの UTF-8 ロケールをインストールおよび設定する必要があります (まだインストールされていない場合)。このタスクを実行するには、オペレーティングシステムのインストールメディア (CD または DVD) が必要です。

注意: Linux プラットフォームの場合、このロケールはデフォルトで en_US.utf8 という名前でインストールされます。ロケールは locale -a コマンドを使用して確認できます。

AIX への UTF-8 のインストール

- 1 US English UTF-8 ロケールがインストールされていないことを確認するには、コマンドプロンプトに "locale -a" と入力します。コマンドの出力結果に EN_US.UTF-8 というエントリが含まれないことを確認します。
- 2 コマンドプロンプトのルートで "smitty mle_add_lang" と入力し、AIX SMIT ユーティリティに (テキストモードで) アクセスします。
- 3 表示される画面の **CULTURAL CONVENTION** ドロップダウンリストと **LANGUAGE TRANSLATION** ドロップダウンリストの両方で、「**UTF-8 US English[EN_US]**」を選択します。

注意: 「INPUT DEVICE/DIRECTORY」はデフォルトの「/dev/cd0」設定のままにします。

- 4 **Enter** キーを押して先に進みます。次のようなメッセージが表示されます。

```
installp: Device /dev/cd0 not ready for operation.  
Please insert media and press Enter to continue.
```

- 5 ディスクドライブに適切な AIX インストールディスクを挿入します。
- 6 コマンドが完了したら、SMIT ユーティリティを終了し、"locale -a" コマンドを入力してロケールに EN_US.UTF-8 が設定されたことを確認します。

Solaris への UTF-8 のインストール

- 1 US English UTF-8 ロケールがインストールされていないことを確認するには、コマンドプロンプトに "locale -a" と入力します。コマンドの出力結果に EN_US.UTF-8 というエントリが含まれないことを確認します。
- 2 ディスクドライブに Solaris のインストール CD #1 を挿入し、次のような適切な場所にマウントします。

```
/cdrom/sol_10_807_sparc/s0
```

- 3 root として "localeadm -a nam -d /cdrom/sol_10_807_sparc/s0" コマンドを入力します。

注意: このコマンドを実行すると、en_US.UTF-8 ロケールのみを指定した場合でも、North America (nam) 地域のすべてのロケールがインストールされます。

- 4 コマンドが完了したら、"locale -a" コマンドを入力してロケールに EN_US.UTF-8 が設定されたことを確認します。

注意: 「[FAQs on Solaris locales](#)」を参照してください。

2.2.3.2 Solaris

注意: オペレーティングシステムに X Window ライブラリがインストールされていることを確認してください。これは Adobe® LiveCycle® PDF Generator 11 および Adobe® LiveCycle® Forms Standard 11 が必要です。詳しくは、オペレーティングシステムのドキュメントを参照してください。

重要: ファイルを抽出するのに Solaris tar コマンドを使用しないでください。このコマンドを使用すると、ファイルが失われるなどのエラーが発生します。GNU tar ツールをダウンロードし、このツールを使用して、Solaris 環境ですべてのファイルを抽出します。

2.2.3.3 Linux

Linux オペレーティングシステムでは、次のことを確認してください。

- **すべての Linux ディストリビューション**：オペレーティングシステムに X Window ライブラリがインストールされていることを確認してください。これは、PDF Generator および Forms で必要です。詳しくは、オペレーティングシステムのドキュメントを参照してください。
- **SUSE Linux**：SUSE Linux Enterprise Server 付属の glibc-locale-32bit ライブラリをインストールしないと、LiveCycle 3 で PDF ファイルが生成されません。デフォルトでこのライブラリファイルはインストールされないため、インストールするには YaST を使用する必要があります（詳しくは、[SUSE Linux Enterprise Server のドキュメント](#)を参照）。

LiveCycle を SUSE Linux 11 にインストールする予定になっている場合は、libstdc++-libc6.2-2.so.3 ライブラリもインストールする必要があります。SUSE Linux 11 には、これらのライブラリがデフォルトでは含まれていません。詳しくは、[Novell Web](#) ページを参照してください。これらのライブラリは、Adobe Central Pro Output Server を実行するために必要です。

2.2.3.4 Solaris および Linux でのファイル制限値の設定

Solaris または Linux の環境での StuckThread 問題を回避するには、/etc/system ファイルで rlim 値を追加するか、大きい値に変更します。

- 1 **(Linux)** /etc/security/limits.conf ファイルを探して開きます。

(Solaris) /etc/system ファイルを探して開きます。

- 2 **(Linux)** /etc/security/limits.conf ファイルに次の行を追加します。

```
<app_group> soft nofile 8192  
<app_group> hard nofile 8192
```

<app_group> を、アプリケーションサーバーを実行するユーザーグループに置き換えます。すべてのユーザーおよびユーザーグループと一致するように、<app_group> をアスタリスク (*) に置き換えることもできます。

(Solaris) /etc/system ファイル内の rlim 値を探して、次のように変更します。

set rlim_fd_cur：プロセスごとのファイル記述子の初期の（不確定な）最大数。この値を 8192 以上に設定します。

set rlim_fd_max：プロセスごとのファイル記述子の確定した最大数。この値を 8192 以上に設定します（この変更は、デフォルト値が 8192 未満の場合にのみ必要です）。この値を変更するには、スーパーユーザーの権限が必要です。

注意：rlim_fd_max 値は、rlim_fd_cur 値以上にする必要があります。

- 3 ファイルを保存して閉じます。
- 4 コンピューターを再起動します。

更新された設定の確認

- 1 新しいシェルを起動します。
- 2 ulimit -n と入力して **Enter** キーを押します。
- 3 返される値が、設定した rlim の値に一致していることを確認します。

2.2.4 Windows でのインストールに必要な権限

Windows にインストールする場合は、管理者権限を持つアカウントを使用する必要があります。管理者以外のアカウントでインストーラーを実行する場合は、管理者権限を持つアカウントの資格情報を入力します。UAC をオフにして、インストールと設定プロセスを実行します。

2.2.5 Windows インストールの設定

64 ビット版の Windows Server 2008 R1、Windows Server 2008 R2、または Vista オペレーティングシステムでは、次のように管理者承認モードのセキュリティオプションを変更します。

- 1 (Windows 2008 R1 および Vista) スタート/コントロールパネル/管理ツール/ローカルセキュリティポリシー/ローカルポリシー/セキュリティオプションを選択します。
(Windows 2008 R2 および Windows 7) スタート/管理ツール/ローカルセキュリティポリシー/ローカルポリシー/セキュリティオプションに移動します。
- 2 「ユーザーアカウント制御：管理者承認モードでの管理者に対する昇格時のプロンプトの動作」を「確認を要求しないで昇格する」に設定します。
- 3 コンピューターを再起動します。

2.2.5.1 Vista および Windows Server 2008 R1 上での Windows UAC の無効化

- 1 システム構成ユーティリティにアクセスするには、スタート/ファイル名を指定して実行を選択し、MSCONFIG と入力します。
- 2 「ツール」タブをクリックし、スクロールして「UAC の無効化」を選択します。
- 3 「起動」をクリックして新しいウィンドウでコマンドを実行します。
- 4 完了したら、コマンドウィンドウを閉じ、システム構成ウィンドウを閉じます。
- 5 コンピューターを再起動します。

UAC を再度有効にするには、上記の手順を繰り返し、「UAC の有効化」を選択してから「起動」をクリックします。

2.2.5.2 Windows 2008 R2 および Windows 7 上での Windows UAC の無効化

- 1 システム構成ユーティリティにアクセスするには、スタート/ファイル名を指定して実行を選択し、MSCONFIG と入力します。
- 2 「ツール」タブをクリックし、スクロールして「UAC 設定の変更」を選択します。
- 3 「起動」をクリックして新しいウィンドウでコマンドを実行します。
- 4 スライダーを「通知しない」のレベルに設定します。
- 5 完了したら、コマンドウィンドウを閉じ、システム構成ウィンドウを閉じます。
- 6 コンピューターを再起動します。

UAC を再度有効にするには、上記の手順を繰り返し、スライダーを目的のレベルに設定してからコンピューターを再起動します。

2.2.6 PDF Generator に関するその他の要件

注意：PDF Generator がデプロイされた Windows 2008 マシン上の SendToPrinter API で、共有プリンターのプロトコルを使用することはできません。CIFS または DirectIP などの代替プロトコルを使用してください。

2.2.6.1 Windows のユーザーアカウント

次のタスクには管理者権限があるユーザーアカウントを使用する必要があります。

- Microsoft Office のインストール
- PDF Generator のインストール
- PDF Generator 用の Acrobat のインストール

- アプリケーションサーバープロセスの実行

注意: PDF Generator 用のユーザーを追加する場合は、アプリケーションサーバーを実行するユーザーに「プロセスレベルトークンの置き換え」権限を付与する必要があります。

2.2.6.2 Linux および Solaris のユーザーアカウント

次のタスクには管理者権限があるユーザーアカウントを使用する必要があります。

- PDF Generator のインストール
- アプリケーションサーバープロセスの実行
- sudo コマンドの実行

注意: PDF Generator 用のユーザーを追加する場合は、アプリケーションサーバーを実行するユーザーに「プロセスレベルトークンの置き換え」権限を付与する必要があります。

2.2.6.3 PDF Generator での 64 ビットアプリケーションサーバーの使用

PDF Generator をシステム上の 64 ビットアプリケーションサーバーで使用する場合は、アプリケーションサーバーが使用する 64 ビット Java 6 JDK の他に、32 ビット Java 6 JDK がインストールされている必要があります。環境変数 JAVA_HOME_32 を設定します。この変数は、64 ビットアプリケーションサーバーが使用しているシステム上の 32 ビット JDK を示す必要があります。指定するパスは、指定したインストールディレクトリと、インストール先のオペレーティングシステムによって変わります。

注意: 32 ビット Sun JDK をインストールし、そのインストールディレクトリを指定するように JAVA_HOME_32 を設定します。Sun Java 6 Release Notes / Supported System Configurations を参照し、使用しているオペレーティングシステム用の 32 ビットバージョンをダウンロードしてください。ただし、AIX を使用している場合は、JAVA_HOME_32 を設定しないでください。

重要: JAVA_HOME_32 は環境変数としてのみ設定し、PATH には含めないでください。JAVA_HOME_32 を PATH に含めると、EAR のデプロイ時、またはサーバーの再起動時に **Java core** ダンプが表示される場合があります。

2.2.6.3.1 Windows での JAVA_HOME_32 変数の設定

- 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 「詳細なシステム設定」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 環境変数 JAVA_HOME_32 を入力します。この値は、JDK を含むディレクトリです。例えば、次のように入力します。

```
C:\Program Files (x86)\Java\jdk1.6.0_31
```

2.2.6.3.2 Linux または Solaris での JAVA_HOME_32 変数の設定

次の例に示すように、Bourne シェルおよび Bash シェルについて、サポート対象の JDK の JAVA_HOME_32 変数を設定します。

```
JAVA_HOME_32=/opt/jdk1.6.0_31  
export JAVA_HOME_32
```

2.2.6.4 ネイティブファイルを変換するためのソフトウェアのインストール

PDF Generator をインストールする前に、PDF 変換サポートを必要とするファイル形式に対応したソフトウェアをインストールし、アプリケーションサーバープロセスの実行に使用されているユーザーアカウントを使用して、ソフトウェアのライセンスを手動でアクティベートします。

LiveCycle デプロイメントで使用するネイティブアプリケーションごとにライセンス契約を参照し、LiveCycle デプロイメントで指定されたライセンス要件を満たしていることを確認してください。通常、ネイティブアプリケーションサポートを使用する各 LiveCycle ユーザーは、そのネイティブアプリケーションを使用するコンピューターでライセンスをアクティベートしている必要があります。

PDF Generator の機能を拡張すると、次のアプリケーションを使用して PDF に変換できるファイルの種類を追加できます。

注意：PDF Generator では、サポート対象アプリケーションの英語版、フランス語版、ドイツ語版および日本語版のみがサポートされています。

- Microsoft Office 2007、2010 (DOC、DOCX、XLS、XLSX、PPT、PPTX、RTF、TXT)
- Microsoft Office Visio 2007、2010 (VSD)
- Microsoft Publisher 2007、2010 (PUB)
- Microsoft Project 2007、2010 (MPP)
- Acrobat で変換された AutoCAD ファイル (DWG、DWF)
- Corel WordPerfect X4 (WPD)(英語のみ)
- Adobe FrameMaker® 8.0 (FM)
- Adobe PageMaker® 7.0 (PMD、PM6、P65、PM)
- OpenOffice 3.3、(ODT、ODP、ODS、ODG、ODF、SXW、SXI、SXC、SXD、SXM)
- Acrobat で変換された Paper Specification 文書 (XPS)

注意：LiveCycle では、上記のすべてのソフトウェアについて、32 ビット版のみがサポートされています。

注意：OpenOffice 3.3 で作成したドキュメントを変換するには、OpenOffice 3.3 以降をサーバーにインストールする必要があります。

注意：Linux プラットフォームでは、OpenOffice は /root ユーザーの下でインストールされていることが必要です。OpenOffice が特定のユーザー用にインストールされている場合は、PDFG が OpenOffice ドキュメントを変換できない可能性があります。

注意：LiveCycle PDF Generator は AutoCAD 2013 で作成された DWF と DWG ファイルをサポートしません。

注意：これらのソフトウェアアプリケーションをサーバー上で使用しないでください。使用した場合、PDF Generator の変換が失敗する可能性があります。

次のネイティブファイルの形式を変換するために、ネイティブソフトウェアアプリケーションをインストールする必要はありません。

- Print ファイル (PS、PRN、EPS)
- Web ファイル (HTML)
- 画像ファイル (JPEG、GIF、BMP、TIFF、PNG)
- Flash ビデオ (SWF、FLV)XML

2.2.6.5 PDF Generator 用の Acrobat のインストール

LiveCycle インストーラーを実行する前に Acrobat XI Pro をインストールします。PDF Generator の設定の問題を回避するために、Acrobat をインストールした後必ず 1 回は Acrobat を起動してください。Acrobat の起動時に表示されるすべてのモーダルダイアログボックスを閉じます。

注意：LiveCycle のインストールで使用するユーザーアカウントで、Acrobat をインストールしてください。

ただし、LiveCycle がインストールされていて Acrobat XI Pro がインストールされていない場合は、Acrobat XI Pro をインストールした後、**[LiveCycle root]\pdfg_config** フォルダーにある Acrobat_for_PDFG_Configuration.bat スクリプトを実行します。これを実行しないと、PDF 変換が失敗します。

Acrobat_PATH (大文字と小文字が区別されます) 環境変数が、Configuration Manager によって自動的に設定されます。この環境変数を手動で設定する方法については、12 ページの「[2.2.6.8 環境変数の設定](#)」を参照してください。環境変数を設定したら、アプリケーションサーバーを再起動します。

注意：使用する JDK に応じて、Sun または IBM の Web サイトから必要な JAR ファイルをダウンロードします。

2.2.6.6 SHX フォントを使用するための Acrobat の設定 (Windows のみ)

注意：PDF Generator で SHX フォントを使用して、AutoCAD をインストールせずに AutoCAD DWG ファイルを変換する場合、次の手順を実行して Acrobat を設定してください。また、次の手順は、管理コンソールで設定されたすべてのユーザーアカウントに対して実行する必要があります。

- 1 Acrobat を開きます。
- 2 **編集／環境設定**を選択します。
- 3 **PDF への変換／Autodesk AutoCAD**を選択します。
- 4 「**設定を編集**」をクリックします。
- 5 「**設定の環境設定**」をクリックします。
- 6 SHX フォントファイルの検索パスの横にある「**参照**」をクリックして、SHX フォントファイルへのパスを指定します。
- 7 開いているダイアログのそれぞれで「**OK**」をクリックします。

2.2.6.7 QuickTime 7

PDF Generator では、PowerPoint プレゼンテーションや PDF マルチメディアファイルなどのファイルに埋め込まれているビデオを変換する場合は、QuickTime 7 (Player または Pro) がインストールされている必要があります。このアプリケーションは、Apple Downloads サイトから入手できます。

2.2.6.8 環境変数の設定

FrameMaker、Photoshop、PageMaker、WordPerfect などのアプリケーションから PDF ドキュメントを作成する場合は、Windows の環境変数を設定する必要があります。

これらの環境変数の名前を以下に示します。

- FrameMaker_PATH
- Notepad_PATH
- OpenOffice_PATH
- PageMaker_PATH
- WordPerfect_PATH
- Acrobat_PATH

これらの環境変数はオプションであり、PDF Generator で対応するアプリケーションを使用して PDF ファイルを変換する場合にのみ設定する必要があります。環境変数の値には、対応するアプリケーションを起動する際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。

例えば、FrameMaker_PATH 変数には C:\ProgramFiles\Adobe\FrameMaker7.2\FrameMaker.exe を含めることができます。ただし、OpenOffice_PATH は他の環境変数とは設定方法が異なります。この変数は、OpenOffice のインストールフォルダーに設定する必要があります (実行ファイルへのパスに設定するものではありません)。Windows での OpenOffice_PATH の一般的な値は、C:\Program Files (x86)\OpenOffice.org 3 です。

Word、PowerPoint、Excel、Visio、Publisher、および Project などの Microsoft Office アプリケーションのパスや AutoCAD のパスは不要です。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。

Windows 環境変数を設定するには

- 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 「詳細なシステム設定」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 設定が必要な環境変数の名前 (FrameMaker_PATH など) を入力します。この値は、実行ファイルを含むディレクトリです。例えば、次のように入力します。

```
C:\Program Files\Adobe\FrameMaker7.2\FrameMaker.exe
```

2.2.6.9 リモートマシン上での PDF Generator の設定

クラスターの場合、LiveCycle は 1 台のマシンにのみインストールされます。次の手順を実行して、クラスター内の他のマシン上の PDF Generator を設定します。

- 1 リモートマシンで、以前のバージョンの Acrobat がインストールされている場合は、Windows のコントロールパネルにある「プログラムの追加と削除」を使ってそれ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 がインストールされているマシンから、pdfg_config と plugins フォルダを、リモートマシンの任意のディレクトリの下にコピーします。
- 4 リモートマシン上で、/pdfg_config/ Acrobat_for_PDFG_Configuration.bat ファイルを開いて編集します。
- 5 goto locationerror というコメント行を探します。

前

```
goto locationerror
```

後

```
rem goto locationerror
```

- 6 Acrobat_for_PDFG_Configuration.bat ファイルを保存し閉じます。
- 7 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。

```
Acrobat_for_PDFG_Configuration.bat <Path of the pdfg_Configuration folder>
```

2.2.6.10 Service Control Manager コマンドラインツール

Windows で PDF Generator の自動インストールを行う場合は、インストール前に sc.exe (Service Control Manager コマンドラインツール) が Windows 環境にインストールされていることを確認します。一部の Windows サーバーでは、sc.exe がプレインストールされていません。デフォルトでは、sc.exe ファイルは C:\Windows\system32 ディレクトリにインストールされます。ほとんどの OS のインストールでは、このツールがインストールされます。このツールがインストールされていない場合は、使用しているバージョンの Windows 用の Windows リソースキットでこのツールを入手できます。サーバーにツールがインストールされていることを確認するには、コマンドプロンプトに sc.exe と入力します。ツールの使用状況が返されます。

注意：PDF Generator が正しく機能するためには、LiveCycle が Windows サービスとして実行されている必要があります。

2.2.6.11 ヘッドレスモードの設定

ヘッドレスモード環境 (モニター、キーボードまたはマウスのないサーバー) で PDF Generator を実行する場合、x11 ライブラリをインストールする必要があります。一部の Linux では、これらのライブラリがデフォルトでインストールされません。そのため、ライブラリを取得して手動でインストールする必要があります。

注意：シェルセッションで x11 転送をアクティブにすると、SOAP 要求中に SOAP UI によって UI 要素が作成され、要求は失敗します。要求のエラーを回避するには、-Djava.awt.headless=true JVM 引数を追加する必要があります。

2.2.6.12 PDF Generator のマルチスレッドファイル変換およびマルチユーザーサポートの有効化

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：(Microsoft Office による) マルチスレッドファイル変換は Microsoft Word 2007 または 2010、および PowerPoint 2007 または 2010 のみでサポートされています。

注意：Microsoft Excel、Publisher、Project、Visio ファイルは同時には変換されません。変換中、EXCEL.exe、PUBLISHER.exe、PROJECT.exe、および VISIO.exe はタスクマネージャーで監視されます。Visio ファイルは、Excel ファイルの処理が完了した後に処理されます。

OpenOffice または PDFMaker の各インスタンスは、それぞれ別のユーザーアカウントを使用して起動されます。追加する各ユーザーアカウントは、LiveCycle サーバーコンピューター上での管理者権限を持つ有効なユーザーである必要があります。詳しくは、9 ページの「[2.2.5 Windows インストールの設定](#)」を参照してください。

LiveCycle サーバーの設定が完了したら、管理コンソールで LiveCycle ユーザーアカウントを追加します。使用しているアプリケーションサーバー版の『LiveCycle インストールガイド』のマルチスレッドファイル変換のユーザーアカウントの項を参照してください。Windows 環境でネイティブファイルおよび OpenOffice ファイルのマルチユーザーサポートを有効にするには、次の権限を持つユーザーを 3 人以上追加します。

PDF Generator ネイティブ変換用にユーザーを追加する場合は、アプリケーションサーバーを実行するユーザーに「プロセスレベルトークンの置き換え」権限を付与します。詳しくは、16 ページの「[2.2.6.14 「プロセスレベルトークンの置き換え」権限の付与 \(Windows のみ\)](#)」を参照してください。

ネイティブアプリケーションの最初のダイアログの解除と自動アップデートの無効化

ネイティブファイルを PDF Generator から変換するには、最初の登録、アクティベート、向上プログラムのダイアログを、これらを表示しないようにするオプションを使用して解除する必要があります。このようなアップデートダイアログは実行中のサーバーに障害を起こす場合があるため、これらのアプリケーションの自動アップデートも無効にする必要があります。

サーバーを実行しているユーザー、およびマルチユーザーサポート用の PDFG アカウントで設定されたすべてのユーザーアカウントでは、ダイアログと自動アップデートを無効にする必要があります。次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、ダイアログを解除する必要があります。

- Microsoft Word (Windows)
- Microsoft Excel (Windows)
- Microsoft PowerPoint (Windows)
- Microsoft Project (Windows)
- Microsoft Publisher (Windows)

- Microsoft Visio (Windows)
- OpenOffice (Windows/ Solaris / Linux)
- Adobe Acrobat (Windows)
- Adobe FrameMaker (Windows)
- Adobe PageMaker (Windows)
- Autodesk AutoCAD (Windows)
- Corel WordPerfect (Windows)

Windows Server 2008 で報告されたエラーの無効化 (オプションですが推奨)

PDF Generator on Windows Server 2008 を使用してドキュメントを PDF に変換中、Windows が実行ファイルに問題が見つかり、ファイルを閉じる必要があると報告する場合があります。ただし、PDF 変換はバックグラウンドで続行されるため、影響を与えません。

エラーを受信しないようにするために、Windows エラー報告を無効にすることができます。エラーレポート機能を無効にする手順については、<http://technet.microsoft.com/ja-jp/library/cc754364.aspx> を参照してください。

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

- 1 /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 lcadm、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。

```
lcadm myhost=(user1) NOPASSWD: ALL
```

```
lcadm myhost=(user2) NOPASSWD: ALL
```

この設定により、lcadm 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。

- 2 すべての LiveCycle ユーザーが LiveCycle Server に接続できるようにします。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@
```

アプリケーションサーバーを起動したセッションが終了されないようにしてください。

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。

- 3 サーバーを再起動します。

2.2.6.13 PDF Generator のマルチユーザーサポート

Windows 環境でネイティブファイルおよび OpenOffice ファイルのマルチユーザーサポートを有効にするには、次の権限を持つユーザーを 3 人以上追加する必要があります。Linux または Solaris プラットフォームでは、ユーザーを 1 人以上作成する必要があります。

プラットフォーム	ユーザー権限
Windows Server 2008	管理者権限を持つユーザー、 LiveCycle の一時ディレクトリ、PDF Generator の一時ディレクトリ、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対する読み取り/書き込み権限。
Linux および Solaris	sudo 権限を持つユーザー LiveCycle 一時ディレクトリ、PDF Generator 一時ディレクトリ、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対する読み取り/書き込み権限。

PDF Generator ネイティブ変換用のユーザーを追加する場合は、アプリケーションサーバーを実行するユーザーに「プロセスレベルトークンの置き換え」権限を付与する必要があります。16 ページの「[2.2.6.14 「プロセスレベルトークンの置き換え」権限の付与 \(Windows のみ\)](#)」を参照してください。

2.2.6.14 「プロセスレベルトークンの置き換え」権限の付与 (Windows のみ)

アプリケーションサーバーを起動したユーザーアカウントは、ローカル管理者グループの一部になっている必要があります。プロセスレベルトークンを置換する権限を持っている必要があります。プロセスレベルトークンを置換する権限を提供するには、次の操作を実行します。

- 1 スタート/ファイル名を指定して実行をクリックして、gpedit.msc と入力します。
- 2 グループポリシーダイアログボックスで、**コンピューターの構成 / Windows の設定 / セキュリティの設定 / ローカルポリシー / ユーザー権限の割り当て**を選択して、「プロセスレベルトークンの置き換え」をダブルクリックします。
- 3 「**ユーザーまたはグループの追加**」をクリックし、アプリケーションサーバーを起動するコマンドプロンプトを開くための Windows ユーザーアカウントを追加します。
- 4 Windows を再起動して、アプリケーションサーバーを起動します。

2.2.6.15 Linux プラットフォームのシンボリックリンク

Linux プラットフォームで HTML から PDF への変換に必要なフォントを置き換えるため、PDF Generator は /usr/share/X11/ ディレクトリへのシンボリックリンクを作成します。

アプリケーションサーバーを実行するユーザーが、シンボリックリンクの作成に必要な権限を所有していない場合があります。そのようなシステムでは、/usr/lib/X11/fonts から /usr/share/X11/fonts ディレクトリへのシンボリックリンクを作成します。

2.2.6.16 Solaris 11 プラットフォームのシンボリックリンク

Solaris 11 では、HTML から PDF への変換に必要ないくつかのフォントは /usr/openwin/lib/X11/fonts の場所から /usr/share/fonts の場所へと移動されています。PDF Generator でこれらのフォントにアクセスできるようにするには、/usr/share/fonts の場所を参照し、/usr/openwin/lib/X11/fonts でシンボリックリンクを作成します。Solaris 11 プラットフォームで HTML から PDF への変換を実行するには、次の手順を実行します。

- 1 ターミナルウィンドウを開きます。
- 2 次のコマンドを実行します。

```
ln -s /usr/share/fonts /usr/openwin/lib/X11/fonts/usr_share_fonts
```

2.2.6.17 Windows Server 2008 および Windows Server 2008 R2 に関するその他の要件

次の記事に従って、Windows Server 2008 バージョンの修正プログラムをインストールします。

<http://support.microsoft.com/kb/972616> (Windows Server 2008 の場合)

<http://support.microsoft.com/kb/2815716> (Windows Server 2008 R2 の場合)

2.2.6.18 Red Hat Enterprise Linux 6 (RHEL6) に関するその他の要件

RHEL6 上で PDF Generator の変換を実行するためには、RPM パッケージとフォントを追加する必要があります。次の手順を実行して、RHEL6 上の PDF Generator を設定します。

1 RHEL6 インストールメディアから次の RPM パッケージをインストールします。

- glibc-2.12-1.25.el6.i686.rpm
- nss-softokn-freebl-3.12.9-3.el6.i686.rpm
- libX11-1.3-2.el6.i686.rpm
- libxcb-1.5-1.el6.i686.rpm
- libXau-1.0.5-1.el6.i686.rpm
- zlib-1.2.3-25.el6.i686.rpm
- libXext-1.1-3.el6.i686.rpm
- fontconfig-2.8.0-3.el6.i686.rpm
- expat-2.0.1-9.1.el6.i686.rpm
- freetype-2.3.11-6.el6_0.2.i686.rpm
- libSM-1.1.0-7.1.el6.i686.rpm
- libICE-1.0.6-1.el6.i686.rpm
- libuuid-2.17.2-12.el6.i686.rpm
- libXrandr-1.3.0-4.el6.i686.rpm
- libXrender-0.9.5-1.el6.i686.rpm
- libXinerama-1.1-1.el6.i686.rpm

2 ブラウザーで、Web サイト <http://cgit.freedesktop.org/xorg/font/ibm-type1/> を開きます

3 圧縮ファイル [font-ibm-type1-1.0.3.tar.gz](#) または [font-ibm-type1-1.0.3.zip](#) をダウンロードします。圧縮ファイルには、必要なフォントが含まれます。

4 ダウンロードした zip ファイルを `/usr/share/fonts` ディレクトリに解凍します。

2.2.7 Connector for Documentum に関するその他の要件

LiveCycle が Documentum に接続する場合、LiveCycle をホストするマシンに Document Foundation Classes をインストールする必要があります。

2.2.8 Connector for IBM Content Manager に関するその他の要件

Adobe® LiveCycle® 11 Connector for IBM® Content Manager では、次のソフトウェアがインストールされている必要があります (両方とも IBM の Web サイトから入手可能)。

- DB2 Universal Database Client
- IBM Information Integrator for Content (II4C)

使用しているアプリケーションサーバー版の『LiveCycle のインストールおよびデプロイ』ドキュメントの「デプロイメント完了後の作業」の章を参照してください。

2.2.8.1 単一の IBM Content Manager データストアに対する接続の設定

- 1 DB2 Configuration Assistant を起動します。
- 2 **Selected / Add Database Using Wizard** をクリックします。
- 3 「**Manually Configure a Connection to a Database**」を選択し、「**Next**」をクリックします。
- 4 「**TCP/IP**」を選択して、「**Next**」をクリックします。
- 5 以下の TCP/IP 通信オプションを指定して、「**Next**」をクリックします。
 - 「**Host Name**」ボックスに、DB2 Content Manager をホストするサーバーのホスト名を入力します。
 - 「**Service Name**」ボックスは空にしておきます。
 - 「**Port Number**」ボックスに、ポート番号を入力します。DB2 Content Manager のデフォルトのポート番号は 50000 です。
- 6 「**Database Name**」ボックスに IBM Content Manager データストア名を入力し、「**Database Alias**」ボックスにデータストアのエイリアス名を入力して、「**Next**」をクリックします。
- 7 「**Next**」をクリックして、デフォルトのデータソース設定を受け入れます。
- 8 「**Operating System**」リストで、使用しているオペレーティングシステムを選択し、「**Next**」をクリックします。
- 9 以下のシステムオプションを指定して、「**Next**」をクリックします。
 - 「**System Name**」ボックスに、DB2 をホストするサーバー名を入力します。「**Discover**」をクリックすると、DB2 Content Manager では指定したシステム名を検索し、システムが見つからない場合、すべての DB2 インスタンスを示します。
 - 「**Host Name**」ボックスにホスト名を入力するか、または「**View Details**」をクリックして、前の手順で指定したシステムのドメインと IP アドレスを表示します。
 - **Operating System** リストで、DB2 Content Manager をデプロイしたオペレーティングシステム (Windows、Linux または AIX) を選択します。
- 10 (オプション) 「**Security**」オプションを指定するには、「**Use Authentication Value in Server's DBM Configuration**」を選択して、「**Finish**」をクリックします。
- 11 **Test Connection** ダイアログボックスで、必要に応じて接続をテストします。

2.2.8.2 複数の IBM Content Manager データストアに対する接続の設定

- 1 18 ページの「[2.2.8.1 単一の IBM Content Manager データストアに対する接続の設定](#)」の手順に従って初期接続を設定します。
- 2 `cmbicmsrvs.ini` ファイル (データストア情報を格納するファイル) を以下のように変更して、データベース接続を追加します。
 - コマンドプロンプトウィンドウで、ディレクトリを `[II4C home]/bin` (例えば Windows では `C:\Program Files\ibm\db2cmv8`、AIX、Linux または Solaris では `/opt/IBM/db2cmv8`) に変更します。
 - `cmbenv81.bat` (Windows) または `cmbenv81.sh` (AIX、Linux または Solaris) ファイルを実行して、II4C の Java ユーティリティ用の環境およびクラスパスを設定します。
 - ディレクトリを `[II4C working directory]/cmgmt/connectors` に変更します。ここで、`[II4C working directory]` は以下のいずれかです。
(Windows) `C:\Program Files\ibm\db2cmv8`

(Linux、AIX) /home/ibmcmadm

(Solaris) /export/home/ibmcmadm

- コマンドを実行します。

```
java com.ibm.mm.sdk.util.cmbsrvsicm -a add -s <library server database name> -sm <database schema name>
```

ここで、<library server database name> は、上記の手順 6 で設定した Database Alias と同じです。

注意: 次の手順では、DB2 の権限を持たないユーザーが cmbicmenv.ini ファイルを使用して接続証明書を共有することができます。

2.2.8.3 IBM Content Manager データストアへのマルチユーザー接続の設定

- 1 コマンドプロンプトウィンドウで、ディレクトリを [II4C home]/bin (例えば Windows では C:\Program Files\ibm\db2cmv8、AIX、Linux または Solaris では /opt/IBM/db2cmv8) に変更します。
- 2 cmbenv81.bat (Windows) または cmbenv81.sh (AIX、Linux または Solaris) ファイルを実行して、II4C の Java ユーティリティ用の環境およびクラスパスを設定します。
- 3 ディレクトリを [II4C working directory]/cmgmt/connectors に変更します。ここで、[II4C working directory] は以下のいずれかです。

(Windows) C:\Program Files\ibm\db2cmv8

(Linux、AIX) /home/ibmcmadm

(Solaris) /export/home/ibmcmadm

- 4 コマンドを実行します。

```
java com.ibm.mm.sdk.util.cmbenvicm -a add -s <library server database name> -u <database user ID> -p <database password>
```

ここで、<library server database name> は、上記の手順 6 で設定した Database alias と同じです。

2.2.9 Connector for IBM FileNet に関するその他の要件

これらの要件はオプションであり、Adobe® LiveCycle® 11 Connector for IBM® FileNet をインストールする場合のみ必要です。

IBM FileNet 4.0

LiveCycle を IBM FileNet 4.0 Content Engine に接続する場合は、Content Engine Java Client をインストールする必要があります。デフォルトで C:\Program Files\ibm\filenet\CEClient に配置される IBM FileNet 4.0 Content Engine クライアントインストーラーを使用します。インストール時に、コンポーネント画面で、Java クライアントコンポーネントのみを選択します。

IBM FileNet 4.0 Process Engine を設定するには、pe.jar ファイルを、Process Engine ディレクトリから LiveCycle をホストするコンピューターにコピーします。C:\FileNetPE\files ディレクトリを作成して、ここに pe.jar ファイルをコピーします。Process Engine クライアントインストールディレクトリは、C:\FileNetPE に変わりました。

IBM FileNet 4.5

LiveCycle を IBM FileNet 4.5 Content Engine に接続する場合は、Content Engine Java Client をインストールする必要があります。デフォルトで C:\Program Files\ibm\filenet\CEClient に配置される IBM FileNet 4.5 Content Engine クライアントインストーラーを使用します。インストール時に、コンポーネント選択画面で、Application Engine または Process Engine から 1 つ以上のコンポーネントを選択します。

IBM FileNet 4.5 Process Engine の場合は、デフォルトで C:\ProgramFiles\FileNet\BPMClient に配置される IBM FileNet 4.5 Process Engine Client をインストールする必要があります。インストール時に、コンポーネント選択画面で「Other」オプションを選択します。

IBM FileNet 5.0

LiveCycle を IBM FileNet 5.0 Content Engine に接続する場合は、Content Engine Java Client をインストールする必要があります。デフォルトで C:\Program Files\FileNet\CEClient に配置される IBM FileNet 5.0 Content Engine クライアントインストーラーを使用します。インストール時に、コンポーネント選択画面で、Application Engine または Process Engine から 1 つ以上のコンポーネントを選択します。

IBM FileNet 5.0 Process Engine の場合は、デフォルトで C:\ProgramFiles\FileNet\BPMClient に配置される IBM FileNet 5.0 Process Engine Client をインストールする必要があります。インストール時に、コンポーネント選択画面で「Other」オプションを選択します。

2.3 LDAP の設定

この設定はオプションであり、ユーザーを認証するために LDAP ディレクトリを使用する場合にのみ必要です。

既存の LDAP サーバーおよびデータベースがない場合、ベンダーのドキュメントに従って LDAP サーバーおよびデータベースをインストールし、設定してください。LiveCycle 設定プロセス中に使用する LDAP の管理者名とパスワードを控えておいてください。LiveCycle に含まれるサービスをインストールおよびデプロイした後、LDAP データベースに接続するように LiveCycle を設定します。この設定には User Manager サービスを使用します。

使用しているアプリケーションサーバーの『[LiveCycle のインストールおよびデプロイ](#)』を参照してください。

2.4 グローバルドキュメントストレージディレクトリ

グローバルドキュメントストレージディレクトリ (GDS) は、プロセス内で使用される長期間有効なファイルや重要な LiveCycle 製品コンポーネントを格納するために使用します。長期間有効なファイルの有効期間内には、LiveCycle システムが複数回再起動され、期間は数日間、数年間にわたる場合もあります。該当するファイルには、PDF ファイル、ポリシー、フォームテンプレートなどがあります。

長期間有効なファイルは、多くの LiveCycle デプロイメントの全体的な状態の中で重要な部分です。非同期ジョブの呼び出しの入力ドキュメントも GDS ディレクトリに保存されます。これらのドキュメントは、要求を処理するために使用可能である必要があります。

GDS ディレクトリは、LiveCycle データベースを初期化する前に作成する必要があります。詳しくは、20 ページの「[2.4.1 グローバルドキュメントストレージディレクトリの場所](#)」を参照してください。

2.4.1 グローバルドキュメントストレージディレクトリの場所

LiveCycle のインストール後に Configuration Manager を使用して GDS ディレクトリの場所を設定します。指定する GDS ディレクトリは可用性が高く、パフォーマンスを向上させるために短時間でアクセスできなければなりません。GDS ディレクトリが共有ネットワークドライブ上にある場合は、場所を UNC 形式で \\computer_name\GDS として指定することをお勧めします。

インストールの完了後に GDS ディレクトリの場所を変更する場合 ([LiveCycle 管理ヘルプ](#)を参照)、GDS ディレクトリ用の適切な場所を計画する必要があります。

重要：GDS ディレクトリがドライブのルートにある場合 (D:¥ など) は、Windows ではモジュールのデプロイメントが失敗します。GDS の場合、ディレクトリがドライブのルートではなく、サブディレクトリに配置されていることを確認する必要があります。例えば、ディレクトリは単に D:¥ ではなく D:¥GDS にする必要があります。

2.4.2 グローバルドキュメントストレージディレクトリのサイズ決定要因

グローバルドキュメントストレージディレクトリのサイズは、デプロイメントに予測される LiveCycle の使用状況によって決まります。GDS ディレクトリには、少なくとも 10 GB のディスク容量を割り当てる必要があります。

以下の要因もサイズの決定に影響します。

- LiveCycle で処理されるドキュメントの標準的な量。大量のドキュメントを処理するには、大容量の GDS ディレクトリが必要です。
- LiveCycle で処理されるドキュメントの標準的なサイズ。サイズの大きいドキュメントを処理するには、大容量の共有 GDS ディレクトリが必要です。
- LiveCycle で処理されるドキュメントの複雑性。LiveCycle に含まれる複数のサービスによって処理されるような複雑なドキュメントの処理には、大容量の GDS ディレクトリが必要です。

2.4.3 グローバルドキュメントストレージディレクトリの保護

GDS ディレクトリへのアクセスは保護する必要があります。このディレクトリの長期間有効なドキュメントには、LiveCycle SDK やユーザーインターフェイスを使用してアクセスするときに特殊な秘密鍵証明書が必要とする情報など、機密性の高いユーザー情報が含まれる場合があります。

使用しているオペレーティングシステムに適したセキュリティ方式を使用してください。アプリケーションサーバーを実行する際に使用するオペレーティングシステムアカウントだけが、このディレクトリに対する読み取りと書き込みの権限を持つようにすることをお勧めします。

注意：GDS ディレクトリのファイルまたはディレクトリを削除すると、LiveCycle サーバーが動作不能になる場合があります。

2.4.4 グローバルドキュメントストレージディレクトリのバックアップ

グローバルドキュメントストレージディレクトリをバックアップして、障害が発生した場合に管理者が LiveCycle を復元できるようにする必要があります。

グローバルドキュメントストレージディレクトリが使用できなくなるかまたはエラーにより失われた場合、GDS ディレクトリおよびデータベースが一貫したバックアップによって復元されるか、LiveCycle が新規インストールにより再び初期化されるまで、LiveCycle は実行されません。

2.5 インストールに関する考慮事項

2.5.1 ウイルススキャンの無効化 (Windows のみ)

インストールの速度を上げるには、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ができます。

2.5.2 ネットワークドライブからのインストール

LiveCycle は、インストールメディアまたはローカルのディスクドライブからのみインストールすることをお勧めします。ネットワーク経由でソフトウェアをインストールしようとする、起動およびインストールにおいて大幅な遅延が発生します。また、ネットワークドライブからインストールすると、ディレクトリパスが長くなり、LiveCycle のインストーラーがインストールを続行できなくなる可能性があります。

2.5.3 手動による Acrobat の使用制限

ネイティブドキュメントの変換に使用する PDF Generator をインストールした場合は、収録されている Acrobat インストールの使用が Generate PDF サービスに制限され、他の使用にはライセンスが供与されません。

2.5.4 一時ディレクトリ

LiveCycle では、ドキュメントオブジェクト用に設定された最大インラインサイズより容量の大きいドキュメントを格納するために一時ディレクトリが必要です。この設定に適切な値を指定することによって LiveCycle サーバーのパフォーマンスを向上させる方法について詳しくは、この[ドキュメント最大インラインサイズの設定に関するブログ](#)を参照してください。

(AIX、Linux および Solaris のみ) root 以外のユーザーがアプリケーションサーバーを実行している場合、ユーザーは指定された一時ディレクトリに対して、読み取り、書き込みおよび実行の権限を持つ必要があります。

重要：(クラスターデプロイメントの場合のみ) 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た場合、以降のアップグレードが失敗することがあります。

2.6 LiveCycle IPv6 サポート

LiveCycle には IPv6 のサポートが含まれます。LiveCycle のインストールドキュメントに定義されているデフォルトの設定では、IPv4 をデフォルトの IP プロトコルとして設定しています。このプロトコルは、IPv4 がサードパーティのインフラストラクチャと最も互換性があるからです。

デプロイメントに必要な場合を除き、IPv6 は有効にしないでください。LiveCycle で IPv6 サポートを有効にすると、サポート対象のプラットフォーム設定が少なくなります。IPv6 を有効にする場合は、その前に、使用するすべてのサードパーティソフトウェア、ハードウェアおよびネットワークが IPv6 をサポートしていることを確認する必要があります。

注意：IPv6 環境で CIFS を有効にする場合は、Configuration Manager を使用して LiveCycle インストールを設定した後に、IPv6 設定を明示的に有効にする必要があります。使用しているアプリケーションサーバー版ガイドの「IPv6 モードでの CIFS の有効化」を参照してください。

2.6.1 サポートされている IPv6 の設定

IPv6 はすべてのインフラストラクチャコンポーネントでサポートされているわけではありません。例えば、Oracle データベースは IPv6 をサポートしていません。アプリケーションサーバーとデータベースの間の接続を IPv4 で、残りの通信を IPv6 経由で設定することにより、これらのデータベースを使用できます。

IPv6 がサポートされていることをコンポーネントのベンダーに確認してください。

2.6.2 IPv6 実装のガイドライン

IPv6 実装を部分的または全体的に使用する場合は、次の点に注意してください。

- LiveCycle のインストール後は、LiveCycle インストーラーから直接 Configuration Manager を起動するオプションを使用しないでください。代わりに、`[LiveCycle root]¥configurationManager¥bin¥IPv6` ディレクトリに移動して、

IPv6 固有のスクリプト (ConfigurationManager_IPv6.bat または ConfigurationManager_IPv6.sh) を実行して Configuration Manager を起動します。

- Configuration Manager を使用してアプリケーションサーバーの設定を検証することを選択している場合は、アプリケーションサーバーに対して IPv6 を有効にした後に検証が失敗します。プロセス中はこのエラーメッセージは無視して構いません。IPv6 モードでアプリケーションサーバーを再起動した後で、アプリケーションサーバーをデータベースに接続できます。
- (WebLogic のみ) 管理対象サーバーに関してのみ IPv6 を有効にする必要があります。Admin Server は引き続き IPv4 上で実行でき、IPv4 アドレスを使用してアクセスできます。ただし、IPv6 環境で起動した管理対象サーバーには、IPv6 アドレスで、または DNS によって解決されたホスト名でのみアクセスできます。
- (WebLogic のみ) アプリケーションサーバーのホストコンピューター上で Configuration Manager を実行している場合であっても、ブートストラップや LiveCycle モジュールのデプロイの際には管理対象サーバーのリスンアドレスを指定する必要があります。このリスンアドレスは、コンピューターの IPv6 アドレスに解決される DNS 名である必要があります。
- データベースサーバーと Pure IPv6 通信を行うには、数値の IPv6 アドレスに解決されるデータベースのホスト名を使用するように、EDC_DS および IDP_DS の両方の接続設定を変更します。
- データベースドライバーなど、多くのソフトウェアコンポーネントでは、数値の IPv6 アドレスが完全にはサポートされていません。そのため、数値の IPv6 アドレスの代わりに DNS 解決されたホスト名を使用することをお勧めします。
- IPv6 のマッピングに使用される名前が CSRF (フィルターセクション) に追加されていることを確認します。名前が追加されていない場合は、[管理ヘルプ](#)の「CSRF 攻撃の防止」を参照してください。

注意: IPv6 のマッピングに使用される名前には、角括弧 ([]) を含めないでください。

- IPv6 環境では、Microsoft SQL Server を使用している場合は、データベースサーバーの IP アドレスを次の形式で指定する必要があります。この文字列で、;serverName はキーワードなので、実際のサーバー名には置き換えないでください。

```
jdbc:sqlserver://;serverName=<IPv6 address>;portNumber=<port>;databaseName=<db_name>
```

ここで、数値の IPv6 アドレスの代わりに、SQL Server データベースのホスト名を指定することもできます。

2.6.3 JBoss 用の IPv6 の設定

- 1 JBoss は、<http://www.jboss.org/jbossas/downloads/> からダウンロードしてインストールするか、インストールメディアのサードパーティディレクトリから jboss zip ファイルを取得して、バンドルされた JBoss を抽出できます。
- 2 adobe-ds.xml ファイルおよびデータベースに固有のデータソース設定ファイルを、LiveCycle データベースに接続するように変更します。
- 3 login-config.xml ファイルを、LiveCycle データベースに接続するように変更します。
- 4 次のファイルを変更して IPv6 を有効にします。
 - (Windows 上の JBoss) [appserver root]¥bin¥run.conf.bat
(他のプラットフォーム上の JBoss) [appserver root]/bin/run.conf
 - -Djava.net.preferIPv4Stack=true を -Djava.net.preferIPv6Stack=true に変更します。
 - -Djava.net.preferIPv6Addresses=true 引数を追加します。
 - (Windows 上の JBoss 5.2 のみ) [appserver root]\bin\run.bat
(その他のプラットフォーム上の JBoss 5.2 のみ) [appserver root]\bin\run.sh
 - -Djava.net.preferIPv4Stack=true を削除します。

- [appserver root]\bin\service.bat
 - -b 0.0.0.0 を -b <hostname mapped to IPv6 address> に置き換えます。
- 5 [LiveCycle root]¥configurationManager¥bin¥IPv6¥ ConfigurationManager_IPv6.bat または ConfigurationManager_IPv6.sh スクリプトを呼び出して、Configuration Manager を起動します。
- 6 Configuration Manager で、EAR ファイルを設定するための手順を選択し、LiveCycle モジュールをブートストラップおよびデプロイします。
- 7 Configuration Manager のプロセスが完了したら、これらの EAR ファイルを [appserver root]\server\<profile_name>\deploy ディレクトリにコピーします。
- 8 コマンドラインから JBoss を起動します。
- 9 IPv6 アドレスにマップされるコンピューターの Configuration Manager ホスト名を指定してから、アプリケーションサーバーをブートストラップして LiveCycle モジュールをデプロイします。

2.6.4 WebLogic 用の IPv6 の設定

- 1 インストーラーを使用して LiveCycle をインストールします。
- 2 インストーラーの終了時に Configuration Manager を起動しないでください。[LiveCycle root]¥configurationManager¥bin¥IPv6¥ ConfigurationManager_IPv6.bat または ConfigurationManager_IPv6.sh スクリプトを呼び出して、Configuration Manager を起動します。
- 3 LiveCycle EAR、WebLogic アプリケーションサーバーを設定するオプションを選択し、Configuration Manager を使用してアプリケーションサーバーの設定を検証します。

注意： Configuration Manager によるデータソースの検証が失敗したというエラーメッセージが表示されます。これは、アプリケーションサーバーがまだ IPv6 モードで起動されていないのに、データソースが IPv6 モードで設定されていることが原因です。この段階では、この警告は無視して構いません。

- 1 WebLogic Server Administration Console から、管理対象サーバーのアプリケーション「**Server Start**」引数を変更して IPv6 を有効にします。
 - -Djava.net.preferIPv4Stack=true を -Djava.net.preferIPv6Stack=true に変更します。
 - -Djava.net.preferIPv6Addresses=true 引数を追加します。
- 2 管理対象サーバーのリスンアドレスを変更し、IPv6 アドレスを使用して有効にします。
 - WebLogic Server Administration Console で、**Environment** / **Servers** / 「[managed server name]**Configuration**」タブを選択します。
 - 「Listen Address」フィールドに、コンピューターのホスト名を入力します。このホスト名が、このコンピューターの IPv6 アドレスに解決されることを確認してください。
- 3 変更を保存して、管理対象サーバーを再起動します。
- 4 [LiveCycle root]¥configurationManager¥bin¥IPv6¥ ConfigurationManager_IPv6.bat または ConfigurationManager_IPv6.sh スクリプトを呼び出して、Configuration Manager を起動します。
- 5 Configuration Manager で、EAR ファイルをデプロイするための手順を選択し、LiveCycle モジュールをブートストラップおよびデプロイします。
- 6 管理対象サーバーのリスンアドレスフィールドで指定したものと同一ホスト名を入力します。

注意： Configuration Manager を同じコンピューターで実行している場合であっても、ブートストラップや LiveCycle モジュールのデプロイの際には管理対象サーバーのリスンアドレスを指定する必要があります。

2.6.5 WebSphere 用の IPv6 の設定

- 1 インストーラースクリプトを使用して LiveCycle をインストールします。インストール完了後、指示があっても Configuration Manager を起動しないでください。
- 2 **[LiveCycle root]**¥configurationManager¥bin¥IPv6 ディレクトリに移動して、IPv6 固有のスクリプト (ConfigurationManager_IPv6.bat または ConfigurationManager_IPv6.sh) を実行して LCM を起動します。
- 3 Configuration Manager のオプションを使用して、EAR およびアプリケーションサーバーを設定します。
- 4 Configuration Manager の手順に従ってアプリケーションサーバーを設定します。データベースを設定中に、IPv6 アドレスにマップするデータベースのホスト名を指定します。
- 5 Configuration Manager によって、アプリケーションサーバー設定を検証します。データソースの検証が失敗しても、警告を無視します。データソースの検証は、WebSphere Administrative Console から実行できます。
- 6 WebSphere Administrative Console で、**Servers / Server Types / WebSphere application servers / [server name] / Java and Process Management / Process definition / Java Virtual Machine** を選択します。「**Generic JVM arguments**」で、引数 `-Djava.net.preferIPv6Addresses=true` を追加します。
- 7 **Servers / Server Types / WebSphere application servers / [server name] / Java and Process Management / Process definition / Java Virtual Machine** を選択し、「**Custom Properties**」をクリックします。`java.net.preferIPv4Stack=true` を `java.net.preferIPv6Stack=true` に変更します。
- 8 WebSphere Administrative Console を使用して、WebSphere Application Server に EAR ファイルを手動でデプロイします。設定済みの EAR ファイルは、**[LiveCycle root]/configurationManager/export** フォルダにあります。
- 9 WebSphere Application Server を再起動します。
- 10 **[LiveCycle root]**¥configurationManager¥bin¥IPv6 ディレクトリに移動し、ConfigurationManager_IPv6.bat または ConfigurationManager_IPv6.sh を実行して Configuration Manager を起動します。
- 11 Configuration Manager で、オプションを選択して LiveCycle モジュールをブートストラップおよびデプロイします。IPv6 アドレスにマップされているホスト名をアプリケーションサーバーに指定します。

注意： IPv6 環境でアプリケーションサーバーを起動後 (`-Djava.net.preferIPv6Stack=true` フラグを使用) は、IPv6 アドレスまたは IPv6 アドレスにマップされているホスト名でのみアクセスできます。

2.7 Central Migration Bridge サービス

Central Migration Bridge サービスを使用すると、Adobe Central Pro Output Server または Adobe Web Output Pak の製品から既存のアプリケーションを移行して、Output サービスで動作させることができます。Central Migration Bridge サービスを使用した移行を行うと、LiveCycle 環境で、現在の IFD/MDF テンプレート、データ変換スクリプトおよび DAT ファイルを使用できるようになります。

注意： Central Migration Bridge が有用なのは、移行対象の既存の Central Pro アプリケーションがある場合のみです。

Central Migration Bridge の使用許可

Central Migration Bridge サービスを使用するには、Central Pro Output Server 5.7 の有効なライセンスを所有しているか、または Central Pro Output Server 5.7 移行契約を締結している必要があります。Central Pro Output Server 5.7 をインストールするには、既存のメディアおよび既存の製品認証コード (PAC) を使用します。PAC は特定のオペレーティングシステムプラットフォーム用です。これが LiveCycle のインストール先のオペレーティングシステムプラットフォームと異なる場合は、そのオペレーティングシステムの PAC を取得する必要があります。移行または Central Pro Output Server 5.7 メディアや PAC の取得方法について詳しくは、アドビの営業担当者にお問い合わせください。

インストールに関する考慮事項

Central Migration Bridge サービスは、Central Pro (バージョン 5.7) 実行可能ファイルと直接やり取りします。Central Pro は LiveCycle と同じサーバーにインストールする必要がありますが、LiveCycle をインストールするための前提条件ではありません (つまり、LiveCycle より前でも後でもインストールできます)。インストール手順については、Central Pro のドキュメントセットを参照してください。

重要: Central Pro を起動したり、自動的に実行するように Central Pro のプロパティを変更したりしないでください。

Windows では、Central Pro サービス Adobe Central Output Server は、手動のサービスとしてインストールされます。このサービスを実行したり、自動的に実行するようにこのサービスのプロパティを変更したりしないでください。

AIX、Linux または Solaris では、Central Pro デーモン jfdaemon を起動しないでください。コンピューターの再起動時に jfdaemon を起動するようにコンピューターの起動スクリプトを編集している場合は、このデーモンが自動的に起動しないようにスクリプトを変更します (Central Pro のインストールドキュメントを参照)。Central は、コマンドラインから jfserver プロセスを起動することによって起動しないでください。

注意: Central Migration Bridge サービスを呼び出す LiveCycle ユーザーには、Central Pro インストールディレクトリに対するアクセス権と、Central Pro 実行ファイルの実行権限が必要です。

LiveCycle 自動インストール

高速モードを使用して、LiveCycle を自動環境でインストールおよび設定する場合は、Central Migration Bridge サービスはデフォルトでインストールおよび設定されます。何らかの入力を求められることはありません。

注意: Adobe Central Pro 製品がデフォルトのディレクトリにインストールされていることを確認してください。

LiveCycle カスタムインストール

カスタムモード (部分的な自動または手動) を使用して LiveCycle をインストールおよび設定する場合は、Configuration Manager で、Central Migration Bridge をデプロイに含めるよう求められます。

デフォルトでは、サービスは Central Pro のデフォルトのインストールパスを使用します。Central Pro が別の場所にインストールされている場合は、Administration Console に移動して、Central Migration Bridge Service 用の [Central Install Dir] の設定を更新してください。

LiveCycle のインストールが完了したら、Central Pro がデフォルトの場所にインストールされていない場合は、次の手順を実行して、LiveCycle が適切なディレクトリを参照するように指定します。

- 1 Administration Console にログインします。
- 2 サービス/アプリケーションおよびサービス/サービスの管理をクリックします。
- 3 「Central Migration Bridge:1.0」 サービスをクリックします。
- 4 Central Pro インストールディレクトリの正しいパスを入力します。
- 5 「保存」をクリックします。

注意: この設定は、Workbench でも可能ですプロセスの作成と管理を参照してください。

第 3 章：LiveCycle の秘密鍵証明書と公開鍵証明書

この章では、次のタスクを実行する方法について説明します。

- Adobe® LiveCycle® Reader® Extensions 11 の使用権限秘密鍵証明書の取得
- Adobe® LiveCycle® Digital Signatures 11 で使用する電子証明書の取得

3.1 Reader Extensions 使用権限秘密鍵証明書の取得

Reader Extensions 使用権限秘密鍵証明書は、Reader Extensions に固有な電子証明書で、これにより、PDF ドキュメントで Adobe Reader の使用権限を有効にすることができます。秘密鍵証明書がインストールされていない場合、Reader Extensions ユーザーは使用権限をドキュメントに適用できません。この機能に標準の電子証明書を使用することはできません。専用の使用権限秘密鍵証明書を使用する必要があります。

使用権限秘密鍵証明書は、Reader Extensions で処理する各 PDF ファイルの使用権限を拡張します。ソフトウェアライセンスの最も重要な部分であり、安全性の高い環境で注意して保管する必要があります。

次のタイプの使用権限秘密鍵証明書を取得できます。

Customer Evaluation (カスタマー評価用)：Reader Extensions を評価するお客様に提供される、有効期限が短い秘密鍵証明書。この秘密鍵証明書を使用してドキュメントに適用されている使用権限は、証明書の有効期限が切れると失効します。この種類の秘密鍵証明書は、2～3 か月間のみ有効です。

Production (実稼働環境用)：全製品を購入したお客様に提供される、有効期間が長い秘密鍵証明書。Production (実稼働環境用) 証明書は、ユーザーごとに固有ですが、複数のシステムにインストールできます。

使用権限秘密鍵証明書は、公開鍵と秘密鍵の両方を含んでいる電子証明書、および秘密鍵証明書にアクセスするためのパスワードとして提供されます。

Reader Extensions の評価版を注文した場合、評価用の使用権限秘密鍵証明書は、製品注文先の営業担当者または評価版をダウンロードした Web サイトから提供されます。

Reader Extensions 製品版を購入した場合、実稼働環境用の使用権限秘密鍵証明書は、Electronic Software Download (ESD) によって配信されます。実稼働環境用の使用権限秘密鍵証明書は、会社に固有のものであり、必要とする特定の使用権限を有効にすることができます。

Reader Extensions を各自のソフトウェアに統合したパートナーまたはソフトウェアプロバイダーを通じて Reader Extensions を入手した場合、使用権限秘密鍵証明書は、アドビからパートナーに提供され、その後パートナーからお客様に提供されます。

注意：使用権限秘密鍵証明書は、一般的なドキュメントの署名や識別情報のアサーションには使用できません。これらの場合は、自己署名証明書を使用するか、認証局 (CA) から識別情報証明書を取得してください。

3.2 Digital Signatures で使用する電子証明書の取得

電子証明書は、Digital Signatures で使用するために必要です。電子証明書は、LiveCycle をインストールおよび設定した後で設定および管理できますが、インストールの前に取得しておくこと、LiveCycle をデプロイした時点で使用できる状態になります。

公開鍵の電子証明書は認証局 (CA) から発行され、電子メールか Web を介して公開鍵証明書ファイルとして送信されます。この公開鍵証明書ファイルには、ドキュメントの暗号化と署名に使用する公開鍵 (「公開鍵証明書」とも呼ばれる) と秘密鍵 (「秘密鍵証明書」とも呼ばれる) への参照が含まれます。公開鍵証明書には実際の秘密鍵は含まれませんが、その代わりに、暗号化されたファイルまたは HSM に秘密鍵を安全に保管しているユーザーを識別するための参照が含まれます。

Internet Explorer (Windows) または OpenSSL (AIX、Linux および Solaris) を使用して、コンピューターで利用可能な任意の互換性のある公開鍵証明書ストアに保管されている公開鍵証明書の PFX ファイル、P12 ファイルおよび CER ファイルを書き出すことができます。PFX ファイルは、公開鍵証明書ストアまたは秘密鍵証明書自体によって許可された方式でのみ書き出すことができます。秘密鍵証明書に対応する公開鍵を保持する CER ファイルは、Internet Explorer または OpenSSL のいずれかを使用して PFX ファイルから書き出すこともできます。

注意: LiveCycle で使用する公開鍵証明書、秘密鍵証明書および証明書失効リスト (CRL) は、Web ベースの管理コンソールからアクセス可能な Trust Store の管理を使用して設定および管理できます ([管理ヘルプ](#)を参照)。

CRL 配布ポイントは、特定の CER ファイルまたは PFX ファイルに対応する CRL をダウンロードできる場所を示しています。

次の種類のファイルがサポートされます。

公開鍵証明書: DER でエンコードされた X509v3 および base64 でエンコードされた証明書 (.cer) ファイル。trust.xml ファイルを検証する公開鍵証明書には、DER でエンコードされた証明書または base64 でエンコードされた証明書のいずれかを使用します。

秘密鍵証明書: 標準の PKCS12 形式 (.pfx ファイルおよび .p12 ファイル) の、最大 4096 ビットの RSA 秘密鍵証明書および DSA 秘密鍵証明書をサポートします。

CRL: base64 および DER でエンコードされた CRL ファイル。

秘密鍵 (秘密鍵証明書) のセキュリティを保護することは、機密情報の安定性を確保する上で欠かせません。通常、物理的なストレージデバイス (一般的に「ハードウェアセキュリティモジュール」(HSM) と呼ばれる) を使用すると、秘密鍵のセキュリティを最大限に保護することができます。物理デバイスを使用しない場合は、機密性の高い秘密鍵や公開鍵証明書は安全な場所にある暗号化ファイルに格納することが重要です。

Digital Signatures では、業界標準である PKCS #11 インターフェイスを使用して HSM と通信することができます。HSM ストレージシステムのインストールと設定に必要なリソースおよびツールは、HSM ベンダーから取得できます。

第4章：LiveCycle データベースの作成

4.1 データベース設定の要件

ここでは、LiveCycle で使用するデータベースを設定する方法について説明します。また、LiveCycle データベースに必要な特殊なテーブル、インデックス、その他のプロパティのうち、Configuration Manager で設定されないものについて説明します。

データベースには以下のエレメントが含まれます。

- LiveCycle サービス
- LiveCycle 実行時の設定
- LiveCycle のプロセスデータ
- 顧客のプロセスの定義とテンプレート
- アプリケーションサーバー管理データ

データベースを作成する前に、プリインストール要件を読み、必要なソフトウェアがインストールされていることを確認する必要があります

LiveCycle を初めてインストールする場合は、空のデータベースを作成する必要があります。LiveCycle をサポートするのに必要な各テーブルは、Configuration Manager を使用して LiveCycle データベースを初期化するときを作成されます (使用しているアプリケーションサーバーの『LiveCycle のインストールおよびデプロイ』を参照)。

4.2 データベースを使用するための最小限のユーザー権限

データベース	初期化権限	ランタイム権限
Oracle	CREATE SESSION CREATE TABLE CREATE VIEW CREATE SEQUENCE UNLIMITED TABLE SPACE	CREATE SESSION UNLIMITED TABLE SPACE (ユーザーのクォータを設定しない場合にのみ必要) CREATE TABLE
MySQL	SELECT INSERT UPDATE DELETE CREATE DROP REFERENCES INDEX ALTER CREATE_TEMP_TABLE LOCK_TABLES	SELECT INSERT UPDATE DELETE
SQL Server - DB レベル	Create Table Create View Connect	Connect
SQL Server - スキーマレベル	Alter Insert References Select Update Delete	Insert Select Update Delete
DB2	一覧については、36 ページの「 4.5.1 DB2 ユーザーアカウント 」を参照してください。	一覧については、36 ページの「 4.5.1 DB2 ユーザーアカウント 」を参照してください。

4.3 Oracle データベースの作成

Oracle 11g のインストール時に作成されたデフォルトデータベースを使用しない場合は、Database Configuration Assistant ツールを使用して新しいデータベースを作成します。LiveCycle は、WebLogic、WebSphere および JBoss Application Server 上の Oracle 11g をサポートします。

注意: LiveCycle 用の Oracle データベースインスタンスの設定では、トランザクション処理用のテンプレートまたは一般処理用のテンプレートを使用することができます。データベースインスタンスの設定にカスタムデータベーステンプレートを使用する場合は、最低限のコンポーネントとして **Oracle JVM** および **Enterprise Manager Repository** を含める必要があります。

Oracle データベースを作成するときは次の手順を実行します。

- 初期データベースサイズは 500 MB 以上に設定します。
- LiveCycle アプリケーションからの永続データに応じてデータベースのサイズを増やせるようにユーザーのクォータを作成します。
- UTF-8 エンコードのサポートを有効にします。
- 「Database Character Set」を「Unicode (AL32UTF8)」に、「National Character Set」を「AL16UTF16 (Unicode UTF-16 universal character set)」に設定します。
- NLS_LENGTH_SEMANTICS を BYTE に設定します (必要な場合)。他の値を設定するとデータベースの初期化が失敗します。
- 「Transaction Processing」を使用して Oracle をインストールし、サーバーの接続モードを「Dedicated Processing」に設定する必要があります。

4.3.1 ユーザーアカウントと権限

データベース上に新規ユーザーアカウントを作成し、次のシステム権限を割り当てます。

- CREATE SEQUENCE
- CREATE VIEW
- UNLIMITED TABLESPACE
- CREATE TABLE
- CREATE CLUSTER
- CREATE SESSION

重要： (WebLogic のみ) WebLogic と Oracle データベースを使用する場合は、データベースユーザー名の先頭の文字として数字を使用したり、ユーザー名にハイフン (-) を使用したりできません。この制限を守らないと、データベースは正常にブートストラップされません。

注意：ユーザー名は、AIX、Linux または Solaris 上にデプロイする場合は 8 文字以下に、Windows の場合は 12 文字以下にする必要があります。

アプリケーションサーバーでデータソースを設定する場合は、次の情報が必要です。

- SID (サービス ID)
- Oracle ユーザーアカウントのユーザー名とパスワード
- データベースサーバーのホスト名または IP アドレス
- Oracle リスナーポート番号 (デフォルトのポートは 1521)

Oracle 11g の使用方法について詳しくは、該当するユーザーマニュアルを参照してください。

4.4 SQL Server データベースの作成

LiveCycle が実行時データと設定データを格納するために使用する SQL Server データベースを作成することができます。SQL Server データベースの作成について詳しくは、SQL Server のマニュアルを参照してください。LiveCycle は、JBoss、WebLogic および WebSphere 上の SQL Server 2005 および 2008 R2 (英語版および日本語版) をサポートします。

SQL Server データベースを作成し、アプリケーションサーバー上でデータソースを設定する際に使用するユーザーアカウントを作成し、DB_OWNER の権限を割り当てます。データベースおよびユーザーの作成について詳しくは、SQL Server のマニュアルを参照してください。

アプリケーションサーバーでデータソースを設定する場合は、次の情報が必要です。

- データベース名
- SQL Server ユーザーアカウントのユーザー名とパスワード
- データベースサーバーのホスト名または IP アドレス
- SQL Server ポート番号

4.4.1 LiveCycle 用の SQL Server のセットアップ

LiveCycle データベースを作成する前に、次の設定を変更して SQL Server を最適化します。

4.4.1.1 メモリの拡張

SQL Server のデフォルトの設定では、積極的なメモリ割り当ては行われていません。この場合、SQL Server データベースのほとんどのデプロイメントでパフォーマンスが大きく低下します。

注意：この節の処理は実行することをお勧めしますが、省略することもできます。

- 1 Microsoft SQL Server Management Studio を使用して、LiveCycle データベースをホストするデータベースサーバーに接続します。
- 2 データベースサーバー接続を右クリックして、「**Properties**」をクリックします。
- 3 **Memory** ページを選択し、「**Minimum Server Memory (in MB)**」ボックスにサーバー上の空きメモリのサイズと等しいサイズを入力します。
- 4 SQL Server データベースを再起動します。

4.4.1.2 プロセッサの優先度の設定

専用のデータベースサーバー (LiveCycle の実稼働環境へのインストールに推奨) 上では、SQL Server プロセスはシステムの CPU リソースを過度に消費しないように設定されます。

注意：この節の処理は実行することをお勧めしますが、省略することもできます。

- 1 Microsoft SQL Server Management Studio を使用して、LiveCycle データベースをホストするデータベースサーバーに接続します。
- 2 データベースサーバー接続を右クリックして、「**Properties**」をクリックします。
- 3 **Processors** ページを選択し、「**Boost SQL Server Priority**」を選択します。
- 4 SQL Server データベースを再起動します。

4.4.1.3 復旧間隔の拡張

この設定では、クラッシュの後、回復までデプロイメントを待機する時間を指定します。SQL Server のデフォルトの設定は 1 分です。この設定値を大きくすると、データベースログからデータベースファイルに変更を書き込む頻度が低くなるので、パフォーマンスが向上します。この設定では、トランザクションの動作が損なわれることはありませんが、起動時に再生されるログファイルのサイズは影響を受けます。

注意：この節の処理は実行することをお勧めしますが、省略することもできます。

- 1 Microsoft SQL Server Management Studio を使用して、LiveCycle データベースをホストするデータベースサーバーに接続します。
- 2 データベース接続を右クリックして、「**Properties**」をクリックします。
- 3 **Database Settings** ページを選択し、「**Recovery Interval (Minutes)**」ボックスに 5 を入力します。
- 4 SQL Server データベースを再起動します。

4.4.1.4 統合セキュリティ

注意：これはオプション設定です。

SQL Server 統合セキュリティを使用する場合は、SQL Server データベースを混合モードまたは Windows 認証モードに設定する必要があります。ただし、Windows 認証モードを使用する場合、SQL Server との信頼された接続を確立するために、Windows に統合セキュリティを設定する必要があります。

- JBoss の場合は、82 ページの「[9.6.3.6 Windows での統合セキュリティの設定](#)」を参照してください。
- WebLogic の場合は、『[LiveCycle のインストールおよびデプロイ \(WebLogic 版\)](#)』の「SQL Server データベース接続の設定」を参照してください。
- WebSphere の場合は、『[LiveCycle のインストールおよびデプロイ \(WebSphere 版\)](#)』の「SQL Server データベース接続の設定」を参照してください。

4.4.2 SQL Server データベースのサイズ変更

SQL Server で提供されるデフォルトのデータベースサイズは、LiveCycle には小さすぎます。データベースが自動拡張に設定されている場合でも、データベースが大きくなると、または大きくなってディスクがフラグメント化し始めると、パフォーマンスの低下などの予期しない結果が生じることがあります。デプロイメント要件に合わせて、作成時に事前にデータベースサイズを割り当てることをお勧めします。

中規模のデプロイメント：LDAP ディレクトリ内におよそ 100,000 のユーザーと 10,000 のグループが存在する環境。「Database Data Initial Size」を 1 GB に設定し、自動拡張を 250 MB に設定します。

大規模のデプロイメント：LDAP ディレクトリ内におよそ 350,000 のユーザーと 10,000 を超えるグループが存在する環境。「Database Data Initial Size」を 2 GB に設定し、自動拡張を 1 GB に設定します。

注意：データベースの拡張が、常に特定のサイズに制限されます。管理者は、LiveCycle データベースのリソース利用状況を監視して、限定された領域、すなわちデータベースが存在するディスク上で使用可能な領域が失われないことを確認します。

4.4.3 LiveCycle データベースユーザー、スキーマおよびログインの作成

これで、LiveCycle データベースユーザー、スキーマおよびログインを作成することができます。

重要：LiveCycle のデータベースインスタンスを作成するときには、必ず SQL_Latin1_General_CP1_CI_AS 照合（データベースを日本語環境で実行する場合は Japanese_CI_AS 照合）を使用してください。それ以外の照合を使用すると、データベースの初期化に失敗する場合があります。SQL Server データベースの作成時に使用した照合と異なる照合を、LiveCycle データベースインスタンスに使用できません。

- 1 Microsoft SQL Server Management Studio で「**Server**」をクリックし、「**Database**」を右クリックして「**New Database**」を選択します。
- 2 任意のデータベース名を入力します。

注意: データベース名は非常に重要です。指定した名前は、以降の手順で一貫して使用する必要があります。以降の手順では database_name として参照します。

- 3 (SQL Server 2005 のみ) 「Use Full-Text Indexing」が選択解除されていることを確認します。
- 4 「Database Data Initial Size MB」ボックスに適切な値を入力します。
 - 小規模な開発または小規模な実稼働システムでは、200 MB を指定します。
 - 大規模なシステムの場合は、33 ページの「4.4.2 SQL Server データベースのサイズ変更」を参照してください。
- 5 「Database DataAutogrowth」ボックスに、50 % と入力します。
- 6 「Database LogInitial Size」ボックスに適切な値を入力します。
 - 小規模な開発または小規模な実稼働システムでは、20 MB を指定します。
 - 大規模なシステムの場合は、33 ページの「4.4.2 SQL Server データベースのサイズ変更」を参照してください。
- 7 「Database LogAutogrowth」ボックスに、50 % と入力します。
- 8 「OK」をクリックすると、データベースが作成されます。

4.4.3.1 SQL Server での LiveCycle ユーザーの作成

以下の手順では、[database_name] はデータベースの作成時に指定した名前を表し、[database_username] は新規ユーザーに指定する名前を表します。

- 1 Microsoft SQL Server Management Studio を使用して、LiveCycle データベースを作成したデータベースサーバーに接続します。
- 2 **Server / Security** をクリックし、「Logins」を右クリックして、「New Login」を選択します。
- 3 ログイン名 [database_username] を入力し、次に、「SQL Server Authentication」を選択して、新しいパスワードを入力します。
- 4 「Enforce Password Expiration, User must change password on next login」が選択解除されていることを確認します。
- 5 デフォルトのデータベースは「Master」のままにし、「OK」をクリックします。
- 6 **Server / Databases / [database_name] / Security** をクリックし、「Schemas」を右クリックして、「New Schema」を選択します。
- 7 「Schema Name」ボックスに [database_username] を入力し、「OK」をクリックします。
- 8 **Server / Databases / [database_name] / Security** をクリックし、「Users」を右クリックして、「New User」を選択します。
- 9 New User ダイアログボックスで、ログイン名とユーザー名 [database_username] を入力します。
- 10 デフォルトのスキーマを [database_username] に設定して「OK」をクリックします。

注意: スキーマ名は [database_username] と同じである必要があります。
- 11 (SQL Server 2008 R2) 「Schemas Owned By This User」領域で、[database_username] スキーマを選択します。
- 12 **サーバー / データベース / [database_name] / セキュリティ** をクリックし、[database_username] スキーマを右クリックして、**プロパティ / 権限** を選択します。
 - a 「ユーザー」または「ロール」で「検索」をクリックして [database_username] を入力し、「OK」をクリックします。
 - b 「明示的」タブで、次の権限を許可します。
 - Alter

- Insert
- リファレンスドキュメント
- Select
- Update
- Delete

c 「OK」をクリックします。

13 サーバー/データベース/[database_name] を右クリックし、[database_username] スキーマを右クリックして、プロパティ/権限を選択します。

a 「明示的」タブで、「テーブルを作成」、「ビューを作成」、「接続」の権限を許可します。

b 「OK」をクリックします。

4.4.3.2 LiveCycle ユーザーとデータベースとの関連付け

LiveCycle ユーザーを作成した後に、このユーザーを LiveCycle データベースに関連付けます。

- 1 Security / Logins をクリックし、[database_username] を右クリックして、「Properties」をクリックします。
- 2 Login Properties の General ページで、ユーザーのデフォルトのデータベースを [database_name] に設定します。
- 3 User Mapping ページを選択し、「Users Mapped To This Login」セクションで、[database_name] が選択されていること、「User」が [database_username] に設定されていること、「Default Schema」が [database_username] に設定されていることを確認します。
- 4 「Users Mapped To This Login」テーブルで [database_name] が選択されていることを確認し、「Database Role Membership For[database_name]」テーブルで「public」が選択されていることを確認し、「OK」をクリックします。

4.4.3.3 LiveCycle データベースの分離レベルの設定

LiveCycle では、長時間実行されているトランザクションと同時期に短時間の読み取りが何度も発生する場合のデッドロックを管理するために、固有の分離レベルが必要です。

重要：MS SQL Server で分離レベルを設定し、デッドロックの問題を回避する必要があります。

- 1 「Databases」をクリックし、[database_name] を右クリックして、「New Query」を選択します。

注意：[database_name] は、データベースの作成時に指定した名前を表します。

- 2 Query パネルで、次のテキストを入力します。

```
ALTER DATABASE [database_name]
SET READ_COMMITTED_SNAPSHOT ON
GO
```

- 3 「Execute」をクリックします。応答がメッセージパネルに表示されます。

4.5 DB2 データベースの作成

この節で説明しているスクリプトのいずれかを実行して DB2 データベースを作成します。このスクリプトは、1 GB のメモリをデータベースに使用するシステムに合わせたものです。システムのデータベース用に割り当てられているメモリ容量がこれより少ない場合、システム設定の調整について詳しくは、DB2 のドキュメントを参照してください。LiveCycle は、DB2 と WebSphere および WebLogic の組み合わせをサポートしています。

アプリケーションサーバーでデータソースを設定する場合は、次の情報が必要です。

- データベース名
- DB2 ユーザーアカウントのユーザー名とパスワード
- データベースサーバーのホスト名または IP アドレス
- DB2 ポート番号

4.5.1 DB2 ユーザーアカウント

LiveCycle では、専用のシステムデータベースアカウントが必要です。アカウントがスキーマ所有者の場合は、それ以外の権限は必要ありません。スキーマ所有者は、スキーマ内の任意のオブジェクトを変更、作成、破棄できます。アカウントがスキーマ所有者ではない場合は、スキーマの権限 ALTERIN、CREATEIN、DROPIN が必要です。さらに、ユーザーは DB2USERS グループに属している必要があります。

DB2 ユーザーアカウントは、製品のインストール時とシステムの実行時の両方にテーブルを作成するために、スキーマ所有者の権限を持つ必要があります。アプリケーションサーバーによっては、データベースユーザーとスキーマに対して、データベースを内部で使用するための要件が追加される場合があります。

ユーザーアカウントの作成について詳しくは、DB2 のドキュメントを参照してください。

注意：ユーザー名は、AIX、Linux または Solaris 上にデプロイする場合は 8 文字以下に、Windows の場合は 12 文字以下にする必要があります。

4.5.2 DB2 データベースの作成

- ❖ DB2 をホストするコンピューターで、使用しているデータベースバージョン用の次の DB2 スクリプトを含む新しいテキストファイルを作成します。

注意：次のテキストには、改行が含まれています。このドキュメント以外の場所にこのテキストをコピーする場合は、改行を削除してください。


```
-- Create a database, using a custom USERSPACE1 definition
-- Configures the database for typical LC use. For large number of connections, add:
--   num_remote_apps
-- to the AUTOCONFIGURE section
-- To use: Globally replace (case sensitive) DB_NAME with the name of the database to be created.
-- Modify the FILE locations based on system configuration
-- Note: The size of the file has to be specified in pages (4K pages in this case) when used as part of the
create database command
create database DB_NAME using codeset utf-8 territory default
USER TABLESPACE MANAGED BY DATABASE USING (FILE'C:\Db2\DB_NAME\DATA_4K_1.db2' 8192) AUTORESIZE YES
AUTOCONFIGURE using workload_type mixed isolation CS admin_priority performance apply DB and DBM;
-- Create required buffer pools. Let the DB manage the sizes based on load.
-- Creates 8K pool for LC
-- Let DB2 manage the sizes based on usage patterns
connect to DB_NAME;
CREATE BUFFERPOOL DB_NAME_BP8K IMMEDIATE SIZE AUTOMATIC PAGESIZE 8192;
ALTER BUFFERPOOL IBMDEFAULTBP IMMEDIATE SIZE AUTOMATIC;
connect reset;
-- Create required tablespaces for tables requiring 8K and 32K pools.
connect to DB_NAME;
-- 8K tablespace used by a number of tables.
-- Note that the majority of data is stored in the 4K tablespace (created as part of the databse create)
CREATE TEMPORARY TABLESPACE DB_NAME_TEMP_8K IN DATABASE PARTITION GROUP IBMTEMPGROUP PAGESIZE 8192 MANAGED
BY SYSTEM
USING ('C:\Db2\DB_NAME\TEMP_8K') PREFETCHSIZE AUTOMATIC BUFFERPOOL DB_NAME_BP8K;
-- Start off large enough to allow a reasonable test to run without needing to extend the space
-- Should be modified to fit expected usage patterns
CREATE LARGE TABLESPACE DB_NAME_DATA_8K IN DATABASE PARTITION GROUP IBMDEFAULTGROUP PAGESIZE 8192 MANAGED BY
DATABASE USING (FILE'C:\Db2\DB_NAME\DATA_8K.db2'1G)
PREFETCHSIZE AUTOMATIC BUFFERPOOL DB_NAME_BP8K AUTORESIZE YES INCREASESIZE 1 G;
commit work;
connect reset;
deactivate database DB_NAME;
activate database DB_NAME;
```

1 スクリプトを次のように変更します。

- **dbname** および **DBNAME** のインスタンスを、使用する LiveCycle データベースの名前に置き換えます。
- 同じホストに複数のデータベースインスタンスを作成する場合、総物理メモリの 10% 以下のバッファプールサイズを使用して、各データベースインスタンスに対して一意の名前のバッファプールを作成します (BP8K_1 など)。例えば、物理メモリが 1 GB のマシンでは、合計のバッファプールサイズが 100 MB (「SIZE 100000」) 以下となるようにします。
- DB2_root を、DB2 がインストールされているルートディレクトリへのパスに置き換えます。
- コマンドに改行が含まれていないことと、各コマンドがセミコロン (;) で終了していることを確認します。
- 次の行の 9000 を、データベースのサイズに基づいて変更します。

```
(FILE'DB2_root\DBNAME_DATA'9000)
```

この値は、データベースを初期化するのに必要な最小ページ数を指定します。データベースを初期化した後に、DB2 管理ツールを使用してこの値を変更することもできます。

2 テキストファイルを DB2 コマンドラインプロセッサがアクセスできる場所に保存します。

3 DB2 コマンドプロンプトを開き、次のコマンドを入力してスクリプトを実行します。

```
db2 -tf <path_to_script_file>/<script_file_name>
```

4.5.3 DB2 の同時使用対応の設定

- 1 DB2 Control Center を開きます。
 - (Windows) スタート / すべてのプログラム / IBM DB2 / General Administration Tools / Control Center を選択します。
 - (AIX、Linux および Solaris) コマンドプロンプトから、db2jcc コマンドを入力します。
- 2 DB2 Control Center のオブジェクトツリーで、「All Databases」をクリックします。
- 3 LiveCycle 製品用に作成したデータベースを右クリックし、「Configuration Advisor」をクリックします。
- 4 Configuration Advisor ウィザードの手順に従い、次のプロパティを設定します。

DB2 のプロパティ	必要な値
Workload type	Mixed
Average number of SQL transactions per unit of work	10 未満
Database Administration Priority	Faster transaction performance
Populated Database	データベースインスタンスの現在の状態に従って設定します。Configuration Manager によって初期化されていない場合、インスタンスは設定されません。
Isolation Level	Read Committed

- 5 「Finish」をクリックします。

4.5.4 新規データベースへのユーザーの追加

- 1 IBM Control Center にログインします。
- 2 [database_name] / User and Group Objects / DB Users をクリックします。
- 3 「Add User」を選択してユーザーを選択します。
- 4 「Authorities」で、「Connect to database」、「Create tables」および「Database administration authority」を選択し、「Apply」を選択します。
- 5 「OK」をクリックします。

4.6 MySQL データベースの作成

注意：このドキュメントで説明する内容は、MySQL を手動でインストールするユーザーを対象にしています。自動インストールに関する説明はありません。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。

MySQL ツールを使用して、LiveCycle で使用する MySQL 5 データベースを作成し、データベースに接続する際にアプリケーションサーバーで使用する MySQL ユーザーアカウントを作成します。また、MySQL データベースサーバーの設定も変更する必要があります。データベースおよびユーザーアカウントの作成について詳しくは、MySQL のマニュアルを参照してください。LiveCycle は、MySQL 5 と JBoss の組み合わせをサポートしています。

注意：MySQL では、ユーザー名やパスワードで特殊文字やスペースの使用はサポートされません。ユーザー名とパスワードがこの制限に従っていることを確認してください。また、Scheduler のエラーを避けるために MySQL データベース名ではハイフン (-) を使用しないでください。

注意：AIX、Linux または Solaris では、`lower_case_table_names` システム変数に 1 を設定して、テーブル名で大文字と小文字が区別されるようにする必要があります。Windows システムでは、このパラメーターを指定しても影響は出ません。テーブル名での大文字と小文字の区別の設定について詳しくは、<http://dev.mysql.com/doc/refman/5.1/ja/identifier-case-sensitivity.html> を参照してください。

注意：グラフィカルユーザーインターフェイス (GUI) を使用して、MySQL でデータベーススキーマおよび新規ユーザーを設定する場合は、MySQL Administrator ツールをインストールする必要があります

MySQL データベースをインストールするときに、MYSQL UI の Variables セクションで UTF-8 文字エンコードを指定する必要があります。

JBoss でデータソースを設定する際は、次の情報が必要です。

- MySQL ユーザーアカウントのユーザー名とパスワード
- データベースサーバーのホスト名または IP アドレス
- MySQL ポート番号

4.6.1 MySQL データベースの場合のプロセスでのドキュメントの使用に関する制限

MySQL データベースでは、テーブルの行に格納できるデータの容量が制限されます。ドキュメントの複数インスタンスに関連するプロセスでは、MySQL の行サイズ制限を上回る記憶領域が必要となることがあります。LiveCycle では、1 つのプロセスインスタンスのデータがテーブルの 1 つの行に格納されるので、大容量データに関連するプロセスの場合、行サイズに関する MySQL の制限を超えて、エラーが発生することがあります。

ドキュメントの多数のインスタンスに関連するプロセスを実行して、データベースでエラーが発生した場合は、ドキュメントを処理する操作が複数のサブプロセスで実行されるようにプロセスを再設計することができます。例えば、プロセスを再設計して操作を 2 つのプロセスで実行することができます。最初のプロセスが元のプロセスの前半の操作に対応し、2 番目のプロセスが後半の操作に対応します。この場合、最初のプロセスに 2 番目のプロセスを呼び出す処理を追加して、2 番目のプロセスがサブプロセスとして実行されるようにする必要があります。

4.6.2 MySQL ユーザーアカウント

作成する MySQL ユーザーアカウントには、LiveCycle データベース内のテーブルにアクセスするために次の権限が必要です。

- SELECT
- INSERT
- UPDATE
- DELETE
- CREATE
- DROP
- REFERENCES
- INDEX
- ALTER
- CREATE_TMP_TABLE
- LOCK_TABLES

4.6.3 MySQL 初期サーバー設定についての要件

my.ini ファイル (Windows) または /etc/my.cnf ファイル (AIX、Linux または Solaris) の mysqld セクションでは次の設定が必要です。

- max_allowed_packet=25M

注意: この設定に失敗すると、「パケットが大きすぎる」というエラーが発生し、結果としてモジュールのデプロイが失敗します。

my.ini ファイル (Windows) または /etc/my.cnf ファイル (AIX、Linux または Solaris) の mysqld セクションでは次の設定をお勧めします。

- key_buffer_size=64M
- sort_buffer_size=1M
- default-storage-engine= INNODB
- innodb_buffer_pool_size=100M
- innodb_additional_mem_pool_size=5M
- innodb_log_file_size=170M
- innodb_log_buffer_size=8M
- innodb_flush_log_at_trx_commit=2
- query_cache_type=2
- default-character-set=utf8
- read_buffer_size=1M
- read_rnd_buffer_size=1M

第 5 章：JBoss Application Server の設定

5.1 設定オプション

JBoss Application Server の使用時、次のインストールオプションが用意されています。

- アドビ事前設定 JBoss
- 手動による JBoss の設定（69 ページの「付録 - JBoss の手動設定」を参照。）

ここでは、自動インストールオプションを使用しない場合に、アドビにより事前設定された JBoss Application Server を使用する方法について説明します。以下のいずれかのシナリオに当てはまる場合は、自動インストール以外の方法を使用する必要があります。

- Linux または Solaris オペレーティングシステムを使用している。
- Oracle または SQL Server を使用しているか、自動セットアップ（部分自動）によりデフォルトでインストールされる MySQL の使用を希望していない。
- JBoss 設定を詳細に制御する必要がある。

LiveCycle は、Windows Server 2008 R1 および R2（Enterprise Edition または Standard Edition）、Red Hat Enterprise Linux 5.5 または 6、Red Hat Enterprise Linux Advanced Platform 5.5 または 6、Solaris 10、SUSE Enterprise Linux 11 のプラットフォームの JBoss 上で動作します

注意：JBoss 版での LiveCycle の自動インストールを使用する場合は、このドキュメントを読む必要はありません。自動インストールの環境を準備するために必要なすべての手順は、『[自動インストールおよびデプロイ \(JBoss 版\)](#)』に記載されています。

5.1.1 JBoss の設定に関する注意事項

JBoss アプリケーションサーバーは様々な 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。Linux または Solaris でも、JBoss は .property 設定ファイルをいくつか使用しますが、これらは UNIX テキストファイルでなければなりません。どのような場合でも Windows 環境でこれらのファイルを編集するときには注意してください。

5.1.2 アドビにより事前設定された JBoss

アドビにより事前設定された JBoss ソフトウェアは、LiveCycle メディアの一部として提供されます。これは、JBoss をインストールする最も単純な方法です。JBoss の詳しい知識は必要ありません。

JBoss のサポート対象バージョンをインターネットからダウンロードする場合は、69 ページの「[付録 - JBoss の手動設定](#)」の設定手順を実行する必要があります。

アドビが JBoss のアーカイブを Red Hat のウェブサイトからダウンロードし、スタンドアロン設定で LiveCycle を実行するために必要なオプションでそれを設定します。これらの設定については、次の節で確認できます。

注意：JBoss のデプロイメントによって、[appserver root]/server/ フォルダー以下に、各データベースタイプのデータソースのディレクトリが個別に作成されます。各データベースの先頭には「lc_」が付きます。これにより、MySQL、Oracle および MS SQL データベースのデータソースファイルは、それぞれ lc_mysql、lc_oracle および lc_sqlserver の各ディレクトリに保存されます。以降の節では、これらの個別ディレクトリを示す際に lc_<db-name> を使用します。

Electronic Software Distribution (ESD) をアドビの Web サイトから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) ファイルを、ファイルシステムに展開します。メインのアーカイブファイルを展開したら、ディレクトリ merge-modules、server および third_party を抽出します。

注意：元の ESD ファイルのディレクトリ階層は変更しないでください。

事前設定された JBoss ソフトウェアは、インストールメディアまたは Electronic Software Distribution (ESD) の /third_party ディレクトリにあります。jboss.zip ファイルにはアドビ事前設定 JBoss が含まれています。

インストール要件に応じて、JBoss のディレクトリおよびその内容を、JBoss のインストール先にコピーします。JBoss をインストールするとすべての設定手順が完了しますが、44 ページの「5.4 アドビの事前設定 JBoss のデータベース接続の設定」で説明している手順は除きます。MySQL を使用する場合は、この節の手順を完了する必要があります。MySQL データベースを使用しない場合は、データベースの適切な設定手順を完了する必要があります。設定手順の詳細は、この章で後述します。

注意：事前設定された JBoss には多数の JSafe JAR ファイルが含まれています。これらのファイルによって、暗号化および公開鍵基盤 (PKI) に関連した FIPS 認定の実装が LiveCycle 内に提供されます。製品ファイルをインストールすると、JSafe JAR ファイルは、[appserver root]/server/lc_<db-name>/lib ディレクトリにデプロイされます。お使いのアプリケーションに独自のバージョンの JSafe JAR ファイルが存在する場合でも、読み込むクラスに応じて、デプロイされた JSafe JAR ファイルが使用されます。

5.2 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。

Java 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。

5.2.1 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」タブをクリックして、「環境変数」をクリックします。
- 3 「新しいシステム変数」領域で、「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、値として Java JDK をインストールしたディレクトリを入力します。このディレクトリには /bin サブディレクトリが含まれています。例えば、次のパスを入力します。

```
C:\Program Files\Java\jdk1.6.0_31
```

5.2.2 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」を選択して、「編集」をクリックします。
- 4 値の先頭に次のテキストを追加します。

```
%JAVA_HOME%\bin;
```

5.2.3 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 シェルで 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
```

5.2.4 PATH 環境変数の設定 (Linux または Solaris)

- 次の例に示すように、Bourne および Bash シェルで PATH 変数を設定することをお勧めします。

```
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

5.2.5 (Windows、Linux または Solaris) JAVA_HOME 環境変数の設定の検証

(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
```

インストールした Java バージョン (1.6.0_31 など) で始まる応答を受信します。

5.2.6 (Solaris) run.conf ファイルの変更

JVM を 64 ビットモードで実行するには、[appserver root]/bin/run.conf ファイル内の次の行に、JVM 引数 **-D64** を追加します。

```
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"
```

5.3 JBoss の起動と停止

この章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。

注意：これらの手順は、アドビにより事前設定された JBoss および手でインストールした JBoss ソフトウェアの両方に適用されます。

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。アドビにより事前設定された JBoss の場合、どの起動設定を呼び出すかは、インストール済みのデータベース (lc_mysql、lc_oracle および lc_sqlserver) によって異なります。

5.3.1 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。

注意：アドビにより事前設定された JBoss にリモートからアクセスする場合は、特定の IP アドレス、または -b 0.0.0.0 を使用するすべてのインターフェイスにバインドする必要があります。ホスト名または IP アドレスを使用する場合、localhost を含む URL を使用しても JBoss に接続できません。

例えば、Windows で実行している SQL Server データベース用に、事前に設定された JBoss インスタンスを起動するには、次のように入力します。

```
run.bat -c lc_sqlserver -b [server_IP_Address]
```

5.3.2 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 name]:<jndi -port>
 - (Linux および Solaris) ./shutdown.sh -s [server name]:<jndi -port>

5.3.3 Linux および Solaris のログオンスクリプト

実稼働で使用するための環境を準備するために、ユーザーのログイン時に環境を自動的に設定することをお勧めします。それには、JBoss プロセスの実行に使用するユーザーのユーザーログインスクリプトで JAVA_HOME および PATH 環境変数を設定する必要があります。

5.4 アドビの事前設定 JBoss のデータベース接続の設定

LiveCycle データベースへの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。

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、SQL Server または Oracle 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。

5.4.1 アドビの事前設定 JBoss での MySQL の設定

アドビにより事前設定された JBoss が、LiveCycle データを格納している MySQL データベースに接続できるようにするには、データソースファイルを作成して、LiveCycle のデプロイ先となる JBoss のインスタンスにデプロイする必要があります。

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 (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_clearite.html の手順を使用することもできます。

5.4.1.1 adobe-ds.xml ファイルの編集

MySQL データソースを設定する前に、38 ページの「[4.6 MySQL データベースの作成](#)」の説明に従って、データベースを MySQL に作成しておく必要があります。

- 1 テキストエディターで [appserver root]/server/lc_mysql/deploy/adobe-ds.xml ファイルを開き、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>  
<driver-class>com.mysql.jdbc.Driver</driver-class>  
<security-domain>MySqlDbRealm</security-domain>
```


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。

5.4.1.2 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/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>DefaultDS</jndi-name>  
<connection-url>jdbc:mysql://localhost:3306/adobe/</connection-url>  
<driver-class>com.mysql.jdbc.Driver</driver-class>  
<security-domain>MySQLDbRealm</security-domain>
```

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。

5.4.1.3 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。

5.4.2 アドビの事前設定 JBoss 用の Oracle の設定

JBoss が、LiveCycle データを格納している Oracle データベースに接続できるようにするには、データソースファイルを作成して、LiveCycle のデプロイ先となる JBoss のインスタンスにデプロイする必要があります。

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 (adobe-ds.xml および oracle-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_clearite.html の手順を使用することもできます。

5.4.2.1 adobe-ds.xml ファイルの編集

- 1 [appserver root]/server/lc_oracle/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>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<security-domain>OracleDbRealm</security-domain>
```

- 2 上述の行の以下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の SID。デフォルト値 adobe をデータベースの SID 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 (Oracle RAC のみ) 手順 1 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。

5 (Oracle RAC のみ) 手順 4 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。

6 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。

7 ファイルを保存して閉じます。

5.4.2.2 oracle-ds.xml ファイルの編集

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/lc_oracle/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_oracle/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>DefaultDS</jndi-name>  
<connection-url>jdbc:oracle:thin@localhost:1521:adobe</connection-url>  
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>  
<security-domain>OracleDbRealm</security-domain>
```

2 上述の行の太字のテキストをデータベースに固有の次の値に置き換えます。

localhost : 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置き換えます。

adobe : この値をデータベースの SID 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 (Oracle RAC のみ) 最初の手順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。

- 5 (Oracle RAC のみ) 前の手順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。
- 6 ファイルを保存して閉じます。

5.4.2.3 login-config.xml ファイルの編集

- 1 **[appserver root]/server/lc_oracle/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="OracleDbRealm">
  <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=DefaultDS</module-option>
      </login-module>
    </authentication>
  </application-policy>
```

- 2 太字のテキスト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。

注意 : Oracle RAC の場合、**adobe** をデータベースに固有のサービス名に置き換えます。
- 3 ファイルを保存して閉じます。
- 4 JBoss を起動します。

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

JBoss が、LiveCycle データを格納している SQL Server データベースに接続できるようにするには、SQL Server のデータソースファイルを作成して、LiveCycle のデプロイ先となる JBoss のインスタンスにデプロイする必要があります (例えば、**[appserver root]/server/lc_sqlserver/deploy**)。

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 (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_clearart.html の手順を使用することもできます。

5.4.3.1 adobe-ds.xml ファイルの編集

- 1 **[appserver root]/server/lc_sqlserver/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>  
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>  
<security-domain>MSSQLDbRealm</security-domain>
```

2 上述の行の以下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、独自のデータベース名に更新する必要があります。

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。

5.4.3.2 mssql-ds.xml ファイルの編集

MS SQL Server データベースと共に LiveCycle を実行する場合は、SQL Server 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、SQL Server JDBC ドライバーが [appserver root]/server/lc_sqlserver/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_sqlserver/deploy/mssql-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 SQL Server の接続設定の内容に変更します。

```
<jndi-name>DefaultDS</jndi-name>  
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>  
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>  
<security-domain>MSSQLDbRealm</security-domain>
```

2 上述の行の太字のテキストをデータベースに固有の次の値に置き換えます。

localhost : この値を SQL Server のホスト名に置き換えます。

1433 : SQL Server 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置き換えます。

adobe : この値を SQL Server データベース名に置き換えます。

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。

注意 : SQL Server の JDBC ドライバーでは数値の IPv6 アドレスは解析されません。IPv6 環境に SQL Server をデプロイする場合、次の形式でサーバー名を指定する必要があります。

```
<connection-url>jdbc:sqlserver://;serverName=<IPv6 address>;  
portNumber=<port>;databaseName=<db_name></connection-url>
```

注意: この形式でサーバーの IPv6 アドレスを指定する場合、`serverName` という文字列はキーワードであることに注意し、このキーワードを実際のサーバー名に置き換えないようにしてください。

- 4 ファイルを保存して閉じます。

5.4.3.3 login-config.xml ファイルの編集

- 1 `[appserver root]/server/lc_sqlserver/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="MSSQLDbRealm">
  <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=DefaultDS</module-option>
      </login-module>
    </authentication>
  </application-policy>
```

- 2 太字のテキスト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。

5.4.3.4 Windows ログインの LiveCycle データベースユーザーへのマップ

- 1 Microsoft SQL Server Management Studio を使用して、LiveCycle データベースをホストするデータベースサーバーに接続します。
- 2 認証モードを「**Windows 認証**」に設定します。
- 3 **セキュリティ/ログイン**で、Windows ドメインユーザーの新しいアカウントを作成して、「**Windows 認証**」を選択します。
- 4 ログイン - 新規作成画面で「**ユーザーマッピング**」をクリックして、新しいユーザーのデータベースおよびデフォルトスキーマを設定します。
- 5 データベースロールとして「**db_owner**」を選択し、「**OK**」をクリックします。

ユーザーを作成したことを確認するには、ツリーで LiveCycle データベースを展開し、**セキュリティ/ユーザー**を開きます。新しいユーザーがユーザーのリストに表示されます。

5.4.3.5 SQL Server データベースドライバーのインストール

注意: まだ実行していない場合は、Microsoft の Web サイトの Microsoft ダウンロードセンターから SQL Server JDBC Driver 3.0 (全プラットフォーム用) をダウンロードするか、ドライバーを `[DVD root]/third_party/to [appserver root]/server/<profile_name>/lib` ディレクトリからコピーしてください。

Microsoft ダウンロードセンターからインストールしている場合は、ドライバーのダウンロードおよびインストールについての Web サイトの指示に従ってください。システム上でドライバーをインストールしたディレクトリの場所を控えておいてください。

注意: Microsoft SQL Server 2005 SP2 および Microsoft SQL Server 2008 のどちらに対しても SQL Server JDBC Driver 3.0 を使用します。

5.4.3.6 Windows 上での統合セキュリティの設定

注意: JBoss サービスは、50 ページの「[5.4.3.4 Windows ログインの LiveCycle データベースユーザーへのマップ](#)」で設定した Windows ユーザーとして実行する必要があります。JBoss がサービスとして実行されていない場合は、そのユーザーとして Windows にログインし、コマンドプロンプトから JBoss を起動する必要があります。

- 1 次の例に示すように、**[appserver root]\server\lc_sqlserver\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。

```
jdbc:sqlserver://<hostname>:<port>;databaseName=<db-name>;integratedSecurity=true.
```

- 2 上述の例の太字の値を、データベースサーバーに適した値に置き換えます。
- 3 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先 (32 ビットオペレーティングシステムの場合のデフォルトは [SQL_root]/sqljdbc_3.0/enu/auth/x86、64 ビットオペレーティングシステムの場合のデフォルトは [SQL_root]/sqljdbc_3.0/enu/auth/x64) にあります。

注意: Microsoft SQL Server 2008 に対しては SQL Server JDBC Driver 3.0 を使用します。

- 4 JBoss for Adobe LiveCycle ES4 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。
- 5 「**アカウント**」を選択し、50 ページの「[5.4.3.4 Windows ログインの LiveCycle データベースユーザーへのマップ](#)」でマップした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、このユーザーとしてログインする必要があります。

5.5 次の手順

『[LiveCycle のインストールおよびデプロイ \(JBoss 版\)](#)』の手順に従って、LiveCycle をインストールします。

第6章：WebSphere Application Server の設定

この章では、LiveCycle のインストールをホストする WebSphere Application Server のインストールと設定の方法について説明します。

6.1 WebSphere のインストール

LiveCycle 製品を実行するには、WebSphere Application Server をインストールする必要があります。LiveCycle で使用する WebSphere は、WebSphere Base として、またはベースプロファイルを 1 つ持つ WebSphere ND としてインストールできます。どちらの方法が適しているか判断するには、WebSphere のドキュメントを参照してください。

6.1.1 WebSphere への Fix Pack のインストール

WebSphere 7.0 のインストール後、LiveCycle をデプロイする前に WebSphere 7.0.0.15 および SR 9 に更新する必要があります。

Fix Pack のインストール手順および WebSphere の更新については、[WebSphere のサポート Web サイト](#)を参照してください。

6.1.2 WebSphere Feature Pack for JPA 2.0 のインストール

WebSphere 7.0 のインストール後、WebSphere Feature Pack for JPA 2.0 をインストールする必要があります。Feature Pack のダウンロードおよびインストールについては、[WebSphere のサポート Web サイト](#)を参照し、「Installing and uninstalling the feature pack」を検索してください。

6.1.3 JAVA_HOME および PATH 環境変数

WebSphere のインストール時に、Java SDK (JDK) がインストールされています。JAVA_HOME および PATH 環境変数には、LiveCycle がデプロイされる JDK を指定できます。

6.1.3.1 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、「新規」をクリックします。
- 4 JAVA_HOME を変数名として入力し、WebSphere にバンドルされた JDK のインストール先ディレクトリを値として指定します。例えば、次のパスを入力します。

```
C:\Program Files\IBM\WebSphere\AppServer\java
```

6.1.3.2 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」を選択して「編集」をクリックし、変数値の先頭に次のテキストを追加します。

```
%JAVA_HOME%\bin;
```

6.1.3.3 DISPLAY 設定の指定 (AIX)

- Windows マシンを使用して AIX コンソールにアクセスしている場合、コマンドプロンプトで次のテキストを入力します。

```
export DISPLAY= <IP Address of the Windows machine telnet, or cygwin to Linux or Solaris  
machine>:0.0
```

注意: DISPLAY 設定が正しくない場合、Configuration Manager の起動時に「No JRE is found...」というエラーが発生します。

6.1.3.4 JAVA_HOME 環境変数の設定 (AIX、Linux および Solaris)

- 次の例に示すように、Borne シェルおよび Bash シェルの JAVA_HOME 変数を設定します。

```
JAVA_HOME=/opt/IBM/WebSphere/AppServer/java  
export JAVA_HOME
```

注意: 指定するパスは、指定したインストールディレクトリと、インストール先のオペレーティングシステムによって変わります。

6.1.3.5 PATH 環境変数の設定 (AIX、Linux および Solaris)

- 次の例に示すように、Borne シェルおよび Bash シェルの PATH 変数を設定します。

```
PATH=$JAVA_HOME/bin:$PATH  
export PATH
```

6.1.3.6 (Windows、AIX、Linux または Solaris) JAVA_HOME 環境変数の検証

(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
```

インストールされている Java のバージョンを示す応答が返されます。

6.1.4 WebSphere Base の起動と WebSphere Administrative Console へのアクセス

6.1.4.1 WebSphere Base の起動

- ❖ サーバーがまだ実行されていない場合、WebSphere Administrative Console を起動し、[appserver root]/profiles/<profile_name>/bin で、次の該当するコマンドを入力します。**server1** は、使用しているサーバーインスタンスの名前に置き換えます。

- (Windows) startServer.bat **server1**
- (AIX、Linux または Solaris) ./startServer.sh **server1**

6.1.4.2 WebSphere Administrative Console へのアクセス

- WebSphere Application Server を起動します。
- Web ブラウザーのアドレスバーに適切な URL を入力します。

```
http://[host name]:[port]/ibm/console
```

現在使用しているサーバーにログインしている場合は、[hostname] を localhost に置き換えることができます。ポート番号はアプリケーションサーバーによって異なり、管理セキュリティが有効かどうかによっても異なります。WebSphere のデフォルトのポート番号は 9060 です。管理セキュリティが有効な場合、デフォルトの SSL ポート番号は 9043 です。

- 1 WebSphere 管理セキュリティが有効になっている場合は、表示されるボックスに WebSphere のユーザー ID とパスワードを入力します。
- 2 「Log In」をクリックします。

6.1.5 WebSphere ND の起動と WebSphere Administrative Console へのアクセス

- 1 コマンドプロンプトを開き、[WebSphere ND root]/profiles/Dmgr01/bin に移動します。
- 2 (Windows) startManager.bat を実行します。
(AIX, Linux および Solaris) startManager.sh を実行します。
- 3 [WebSphere ND root]/profiles/[profile name]/bin に移動し、startNode.bat を実行します。
- 4 サービスが起動したら、Web ブラウザーのアドレスバーに http://[host name]:[port]/ibm/console という URL を入力してサーバーに接続します。

現在使用しているサーバーにログインしている場合は、[hostname] を localhost に置き換えることができます。ポート番号はアプリケーションサーバーによって異なり、管理セキュリティが有効かどうかによっても異なります。WebSphere のデフォルトのポート番号は 9060 です。管理セキュリティが有効な場合、デフォルトの SSL ポート番号は 9043 です。

- 5 WebSphere 管理セキュリティが有効になっている場合は、表示されるボックスに WebSphere のユーザー ID とパスワードを入力して、「Log In」をクリックします。
- 6 WebSphere Administrative Console のナビゲーションツリーで、**Servers / Server Types / WebSphere application servers** をクリックします。
- 7 リストからサーバーを選択し、「Start」をクリックします。サーバーの起動時にサーバーの状態が変化することに注意してください。

6.1.6 ディレクトリ権限

LiveCycle アプリケーションは、[appserver root]/installedApps ディレクトリにファイルを抽出します。これにより、そのディレクトリには書き込み権限を付与することが重要です。次の節では、書き込み権限を付与できない場合にファイルの抽出先を変更する方法について説明します。

注意：ファイルの抽出先となる場所は、[appserver root]/profiles/<profile_name>/installedApps に変更することをお勧めします。

6.1.6.1 ファイルの抽出先の変更

- 1 WebSphere Administrative Console にログインします。
- 2 **Servers / Server Types / WebSphere Application servers** をクリックし、サーバー名 (「server1」など) をクリックします。
- 3 「Server Infrastructure」で、**Java and Process Management / Process Definition** をクリックします。
- 4 「Additional Properties」で「**Java Virtual Machine**」をクリックし、次に「**Custom Properties**」をクリックします。
- 5 「New」をクリックし、adobeidp.RootDirectory という名前のカスタムプロパティを作成します。

- 6 adobeidp.RootDirectory の値を、アドビのネイティブファイルの抽出先となるパス ([appserver root]/profiles/<profile_name>/installedApps など) に設定します。
- 7 「OK」または「Apply」をクリックします。
- 8 「Messages」ボックスで「Save directly to master configuration」をクリックし、アプリケーションサーバーを再起動します。

6.2 WebSphere Application Server の準備

ここでは、LiveCycle のデプロイメント用のアプリケーションサーバーインスタンスを準備および設定する方法について説明します。

6.2.1 SOAP 要求タイムアウト値の延長

- 1 [appserver root] ディレクトリに移動し、soap.client.props という名前のすべてのファイルを検索します。この名前のファイルは複数存在する場合があります。例えば、AIX、Linux または Solaris サーバー上には、次のファイルが存在します。
 - [appserver root]/profileTemplates/default/documents/properties/soap.client.props
 - [appserver root]/profileTemplates/management/documents/properties/soap.client.props
 - [appserver root]/profiles/<profile_name>/properties/soap.client.props
 - [appserver root]/profiles/<profile_name>/temp/soap.client.props
- 2 各 soap.client.props ファイルをテキストエディターで開き、com.ibm.SOAP.requestTimeout プロパティを探して、値を 180 から 1800 に変更します。
- 3 各 soap.client.props ファイルを保存して閉じます。
- 4 WebSphere Administrative Console のナビゲーションツリーで、**Servers** / **Server Types** / **WebSphere Application servers** をクリックし、右側のウィンドウでサーバー名をクリックします。
- 5 「Server Infrastructure」で、**Administration** / **Administration Services** をクリックします。
- 6 「Additional Properties」で「**JMX Connectors**」をクリックし、「**SOAPConnector**」をクリックします。
- 7 次の画面で「**Custom properties**」をクリックし、「**requestTimeout**」をクリックします。
- 8 必要に応じて、次の画面で「**Value**」ボックスの 600 を 1800 に変更します。「OK」または「Apply」をクリックします。
- 9 「Messages」ボックスで、「**Save directly to master configuration**」をクリックします。

6.2.2 デプロイヤーのヒープサイズの拡大

タイムアウトエラーを回避するために、ejbdeploy.bat/sh スクリプトでヒープサイズを拡大する必要があります。

注意： WebSphere 8 以降のバージョンでは、ejbdeploy.bat/sh スクリプトはデフォルトでインストールされません。そのため、インストール中のエラーを回避するために別途インストールする必要があります。スクリプトの追加については、「[EJBDeploy - Optional feature in WebSphere Application Server Version 8 \(WebSphere Application Server Version 8 のオプション機能\)](#)」を参照してください。

6.2.2.1 AIX、Linux または Solaris

- 1 [appserver root]/deploytool/itp/ ディレクトリに移動して、ejbdeploy.sh を編集用に開きます。

- 2 (Solaris のみ) SunOS セクションで、EJBDEPLOY_JVM_OPTIONS 属性を探して -XX:PermSize オプションの値を 256m に変更し、-Xverify オプションの値が none であることを確認します。
- 3 JAVA_CMD \ セクションの最後に次のようなヒープサイズパラメーターを追加します：
-Xms256m -Xmx512m
- 4 ファイルを保存して閉じます。

6.2.2.2 Windows

- 1 [appserver root]\¥deploytool¥itp に移動して、ejbdeploy.bat ファイルをテキストエディターで開きます。
- 2 %JAVA_HOME% で始まる行を探して、引数 -Xmx を探します。
- 3 この引数を -Xmx512M に変更します。
- ❖ ファイルを保存して閉じます。

6.2.3 MaxPermSize の拡大 (Solaris 上の WebSphere)

- 1 WebSphere Administrative Console にログインします。
- 2 WebSphere Administrative Console のナビゲーションツリーで、**Servers / Server Types / WebSphere Application servers** をクリックし、右側のウィンドウでサーバー名をクリックします。
- 3 「Server Infrastructure」で、**Java and Process Management / Process Definition** をクリックします。
- 4 「Additional Properties」で、「**Java Virtual Machine**」をクリックします。
- 5 「**Generic JVM Arguments**」に、MaxPermSize パラメーターとして -XX:MaxPermSize=512m を入力します。
- 6 「**OK**」または「**Apply**」をクリックします。
- 7 「Messages」ボックスで「**Save directly to master configuration**」をクリックし、アプリケーションサーバーを再起動します。

6.2.4 受信および送信用の通信設定

注意：これらの手順は、WebSphere アプリケーションサーバーでグローバルセキュリティを有効にしている場合にのみ実行してください。

- 1 WebSphere 管理コンソールで、「**Security > Global Security**」に移動します。
- 2 「Authentication」セクションで「**RMI/IIOP security**」をクリックします。
- 3 「**CSIV2 inbound communication**」をクリックし、「**Transport**」を **SSL-supported** に設定します。「**OK**」をクリックします。
- 4 「Messages」ボックスで、「**Save directly to master configuration**」をクリックします。
- 5 「**CSIV2 outbound communication**」をクリックし、「**Transport**」を「**SSL-supported**」に設定します。
- 6 「**OK**」をクリックします。
- 7 「Messages」ボックスで、「**Save directly to master configuration**」をクリックします。「**OK**」をクリックします。

6.3 LiveCycle データベース接続の設定

次に、『LiveCycle のインストールおよびデプロイ (WebSphere 版)』の手順に従って、データベース接続を設定し、LiveCycle をインストールします。

6.4 グローバルセキュリティを有効にする場合の WebSphere Application Server の設定

グローバルセキュリティを使用する場合、適切なロールを持つユーザーとして WebSphere Application Server を実行する必要があります。WebSphere グローバルセキュリティを有効にする場合、次のいずれかの方法で WebSphere Application Server を設定できます。

- 必要なロールを持つ新規ユーザーを作成し、そのユーザーとして WebSphere Application Server を実行します。WebSphere Application Server を実行するユーザーが既に存在する場合は、必要なロールをそのユーザーに割り当てます。

重要：このユーザーとして WebSphere Application Server を起動します。グローバルセキュリティが有効になっているときに、他のユーザーとして WebSphere Application Server を起動すると、一部の WebSphere プロセスが停止する可能性があります。

安全性の高い環境では、この方法を使用することをお勧めします。

- 必要なロールを EVERYONE グループに設定します。

新しい WebSphere Application Server ユーザーを作成するには：

- 1 WebSphere Administrative Console のナビゲーションツリーで、**Environment / Naming / CORBA Naming Service Users** をクリックした後、右側のウィンドウの「**Add**」をクリックします。
- 2 「**Roles**」で、すべてのロールを選択します。
- 3 「**Search and Select Users**」で、「**User Realm**」を選択します。
- 4 検索ボックスに検索文字列を入力し、「**Search**」をクリックします。
注意：すべてのユーザーを取得するには、アスタリスク (*) を入力します。
- 5 「**Available**」テキストボックスから、必要なユーザーを選択し、右矢印をクリックして「**Mapped to role**」ボックスに追加します。
- 6 「**Save directly to master configuration**」をクリックします。

既存の WebSphere Application Server ユーザーを設定するには：

- 1 WebSphere Administrative Console のナビゲーションツリーで、**Environment / Naming / CORBA Naming Service Users** をクリックした後、右側のウィンドウで該当ユーザーをクリックします。
- 2 「**Roles**」で、必要なロールを選択します。
- 3 「**OK**」または「**Apply**」をクリックします。
- 4 「**Save directly to master configuration**」をクリックします。

EVERYONE グループを設定するには

- 1 WebSphere Administrative Console のナビゲーションツリーで、**Environment / Naming / CORBA Naming Service Groups** をクリックします。

2 「Roles」で、必要なロールを選択します。

3 「Select from special subjects」を有効にし、Special subjects リストから「EVERYONE」グループを選択します。

注意：EVERYONE グループが設定済みの場合、グループは Special subjects リストに表示されません。このグループに必要なロールが割り当てられていない場合、割り当てる必要があります。

1 「OK」または「Apply」をクリックします。

2 「Save directly to master configuration」をクリックします。

6.4.1 CSiv2 Inbound Transport の設定

デフォルトの Global Security が有効な状態での IBM WebSphere をインストールすると、CSiv2 Inbound Transport オプションが SSL-required に設定されます。この設定は、Output および Forms コンポーネントの失敗を引き起こします。CSiv2 Inbound Transport オプションを SSL-Supported に変更したことを確認します。オプションを変更するには、次の操作を行います。

1 IBM WebSphere Administration Console にログインします。

2 「Security」を展開して、「Global security」をクリックします。

3 Authentication セクションで、「RMI/IIOP security」を展開して、「CSiv2 inbound communications」をクリックします。

4 CSiv2 Transport Layer セクションで、「Transport」の値を「SSL-Supported」に設定します。

5 「適用」をクリックします。

第7章：WebLogic Server の設定

7.1 WebLogic 用の LiveCycle の要件

WebLogic の起動方法には、主に次の2つがあります。

- ノードマネージャーを使用して、管理サーバーから設定する（推奨）
- StartManagedWebLogic スクリプトの設定で、直接管理対象サーバーを起動する

Configuration Manager の自動設定機能を使用するには（この方法を選択することをお勧めします）、ノードマネージャーを使用し、管理サーバーによって設定する必要があります。

管理対象サーバーを直接起動する場合も、Configuration Manager の自動設定機能は使用できます。ただし、Configuration Manager に関連する変更内容については StartManagedWebLogic スクリプトに手動で入力する必要があります。適切な手順については、『LiveCycle のインストールおよびデプロイ（WebLogic 版）』を参照してください。

7.2 WebLogic Server のインストール

使用しているオペレーティングシステム版の Oracle WebLogic Server を入手し、付属のマニュアルに従ってインストールします（http://download.oracle.com/docs/cd/E14571_01/wls.htm を参照してください）。

注意： WebLogic Server のインストールと実行には、コンピューターの管理者の権限を持つユーザーアカウントを使用する必要があります。

7.2.1 Solaris（64 ビット Java）での WebLogic の設定

Solaris で 64 ビット Java を使用するように WebLogic をインストールする際には、追加の手順を実行する必要があります。

- 1 -d64 フラグを使用して WebLogic インストーラーを実行します。
- 2 新たに起動したコマンドプロンプトで [appserver_root]/server/bin ディレクトリに移動します。
- 3 export JAVA_OPTIONS=-d64 と入力します。
- 4 スクリプトを使用してノードマネージャーを起動します。startNodeManager.sh と入力します。

注意： 64 ビット Java を使用して WebLogic Administration Server を起動する必要があります。startWeblogic.sh スクリプトを使用して WebLogic Administration Server を起動する前に、環境で JAVA_OPTIONS=-d64 が設定されていることを確認してください。

7.3 JDK のインストール

7.3.1 JDK のインストール

オペレーティングシステムに応じて、Oracle JRockit® または Oracle JAVA を入手しインストールします。サポートされているバージョンについては詳しくは、『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。

JAVA_HOME および PATH 環境変数は、LiveCycle のデプロイ先となるサーバーの JDK を示している必要があります。

7.3.2 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、「新規」をクリックします。
- 4 変数名として JAVA_HOME と入力し、Java SDK 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、WebLogic によって Java SDK がインストールされた場所であり、/bin サブディレクトリが含まれています。例えば、次のテキストを入力します。

```
C:\Program Files\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
```

注意：OpenOffice.org ファイルの変換に PDF Generator を使用している場合、Sun JDK がインストールされているディレクトリを示すように JAVA_HOME_32 環境変数を設定する必要があります。

<http://www.oracle.com/technetwork/java/index.html> から Oracle JDK をダウンロードします。

7.3.3 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;
```

注意：PATH 環境変数の最後のエントリの末尾にスラッシュ (¥) が付いていないことを確認します。スラッシュがあると、WebLogic サーバーインスタンスは起動時に失敗します。スラッシュが付いている場合は削除し、変更内容を保存します。

7.3.4 JAVA_HOME 環境変数の設定 (Linux および Solaris)

次の例に示すように、Bourne シェルおよび Bash シェルの JAVA_HOME 変数を設定します。

Linux :

```
JAVA_HOME=/opt/jrockit 1.6.0.28  
export JAVA_HOME
```

Solaris :

```
JAVA_HOME=/usr/java  
export JAVA_HOME
```

注意：指定するパスは、指定したインストールディレクトリと、インストール先のオペレーティングシステムによって変わります。

注意：JAVA_HOME_32 環境変数を設定する必要があるのは、PDF Generator を使用して WebLogic Server 環境で Open Office ファイルを変換する場合です。

7.3.5 PATH 環境変数の設定 (Linux および Solaris)

次の例に示すように、Bourne シェルおよび Bash シェルの PATH 変数を設定します。

```
PATH=$JAVA_HOME/bin:$PATH  
export PATH
```


7.3.6 JAVA_HOME 環境変数の検証

(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
```

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。

7.4 WebLogic Server ドメインの作成

WebLogic Server にアプリケーションをデプロイするには、WebLogic Server ドメインが存在する必要があります。ドメインは、WebLogic Server の基本的な管理単位です。WebLogic Server ドメインの基本的な 2 種類は次のとおりです。

管理対象サーバーを含むドメイン (推奨)：通常、実稼働環境は管理サーバーと 1 つ以上の管理対象サーバーで構成されません。管理サーバーでは管理操作が実行されます。アプリケーションとリソースは個々の管理対象サーバーにデプロイされません。

スタンドアロンサーバードメイン：この種類のドメインは、1 つのサーバーインスタンスが管理サーバーと管理対象サーバーの両方の機能を果たす開発環境またはテスト環境に使用できます。これによって管理が容易になり、開発環境で WebLogic のホットデプロイ機能を利用できるようになります (非推奨)。

注意：Oracle は、実稼働環境ではアプリケーションをドメイン内の管理対象サーバーのみにデプロイし、管理サーバーは管理タスク専用とすることを推奨しています。

7.4.1 管理対象サーバーが含まれる WebLogic ドメインの作成

- 1 コマンドプロンプトで、[appserver root]/common/bin ディレクトリに移動し、次の該当するコマンドを入力して WebLogic Configuration Wizard を起動します。
 - (Windows) config.cmd
 - (Linux および Solaris) ./config.sh
- 2 「Welcome」画面で、「**Create a new WebLogic domain**」を選択して「**Next**」をクリックします。
- 3 「Select Domain Source」画面で、「**Generate a domain configured automatically to support the following products**」を選択し、「**Next**」をクリックします。
- 4 「Create WebLogic Domain」画面で、デフォルトの値を受け入れるか、必要に応じてドメイン名とドメインロケーションを入力し、「**Next**」をクリックします。
- 5 「Configure Administrator Username and Password」画面で、WebLogic ユーザー名とパスワードを入力し、パスワードを再入力して確認し、「**Next**」をクリックします。Configuration Manager で入力を求められるので、このユーザー名とパスワードを記録します。
- 6 WebLogic Domain Startup Mode パネルで、「**Production Mode**」を選択します。

重要：「Development Mode」の使用はお勧めしません。
- 7 右側のウィンドウで、使用しているオペレーティングシステムに応じた JDK を選択し、「**Next**」をクリックします。
 - (Windows、Linux) **JRockit SDK 1.6** 以降の場所。例えば、C:\Program Files\JRockit Real Time\jrockit-jdk1.6.0_24-R28.1.3-4.0.1 (Windows) または /opt/JRockit Real Time/jrockit-jdk1.6.0_24-R28.1.3-4.0.1 (Linux)
 - (Solaris) **Oracle JDK 1.6** 以降のロケーション。
- 8 「Select Optional Configuration」画面で「**Next**」をクリックします。
- 9 「Configuration Summary」画面で「**Create**」をクリックします。

10 「Creating Domain」画面で、設定の作成が完了したら次のタスクを実行します。

- (Windows) 「Start Admin Server」を選択して「Done」をクリックします。
- (Linux および Solaris) 「Done」をクリックし、コマンドプロンプトで [appserverdomain] に移動して、
./startWebLogic.sh と入力して、サーバーを起動します。

11 (Windows、Linux または Solaris) プロンプトが表示されたら、手順 5 で入力した WebLogic のユーザー名とパスワードを入力します。

注意：コマンドプロンプトで管理サーバーが起動します。コマンドプロンプトにユーザー名とパスワードを入力します。

7.5 WebLogic 管理対象サーバー

LiveCycle アプリケーションは、ドメイン内の管理対象サーバーにデプロイする必要があります。管理サーバーは管理タスク専用にご覧ください。サーバードメイン、管理サーバーおよび管理対象サーバーの作成方法について詳しくは、WebLogic の製品マニュアルを参照してください。

注意：64 ビット Java を使用して WebLogic Administration Server を起動する必要があります。startWeblogic.sh スクリプトを使用して WebLogic Administration Server を起動する前に、環境に JAVA_OPTIONS=-d64 が設定されていることを確認してください。

注意：次の手順では、デフォルトプロパティに必要な変更について説明します。明記されていないプロパティについては、既存の設定を使用してください。各画面について詳しくは、WebLogic Server Administration Console の WebLogic のヘルプを参照してください。

7.5.1 ノードマネージャーの起動

この章の残りの節に進む前に、ノードマネージャーを起動する必要があります。

- 新しいコマンドプロンプトで、[appserver root]\server\bin ディレクトリに移動して、該当するコマンドを入力します。

(Windows) startNodeManager.cmd

(UNIX) ./startNodeManager.sh

注意：Windows で、ノードマネージャーを Windows サービスとしてインストールした場合は、ノードマネージャーサービスを開始する必要があります。それ以外の場合は、コマンドラインオプションを使用してノードマネージャーを起動します。

7.5.2 新しい WebLogic 管理対象サーバーの作成

1 WebLogic Administration Server がまだ実行されていない場合は、コマンドプロンプトで [appserverdomain] ディレクトリに移動して、該当するコマンドを入力します。

- (Windows) startWebLogic.cmd
- (Linux および Solaris) ./startWebLogic.sh

2 WebLogic Server Administration Console にアクセスするには、Web ブラウザーの URL 行に http://[hostname]:7001/console と入力します。

3 この WebLogic 設定の作成時に使用したユーザー名とパスワードを入力して、「Log In」をクリックします。

4 Change Center で、「Lock & Edit」をクリックします。

5 「Domain Structure」で、Environment / Servers をクリックし、右側のウィンドウで「New」をクリックします。

6 Create New Server ページの「Server Name」ボックスに、管理対象サーバーの名前 (server1 など) を入力します。

- 7 「**Server Listen Address**」ボックスに、コンピューター名または IP アドレスを入力します。
注意：アプリケーションサーバーを設定する場合と、**Configuration Manager** の実行時にデータベースを初期化する場合にも同じ値を使用する必要があるため、この値 (デフォルトは localhost) を控えておいてください。
- 8 「**Server Listen Port**」ボックスに、現在使用中でないポート番号 (8001 など) を入力します。ポート 7001 は既に管理サーバーに使用されています。
- 9 「**Finish**」をクリックし、「**Activate Changes**」をクリックします。
- 10 Change Center で、「**Lock & Edit**」をクリックします。
- 11 「Domain Structure」で、「**Environment** / **Machines**」をクリックし、右側のウィンドウで「**New**」をクリックします。
- 12 Create New Machine ページの「**Name**」ボックスにマシン名を入力し、使用しているオペレーティングシステムを「**Machine OS**」リストから選択し、「**OK**」をクリックします。
- 13 **Environment** / **Servers** をクリックし、作成した管理対象サーバーの名前をクリックします。
- 14 「**Machine**」リストで、作成したマシンを選択します。
- 15 リスポートが手順 8 で入力したポート番号と同じであることを確認します。
- 16 「**Save**」をクリックします。

7.5.3 メモリ設定

Solaris では、Configuration Manager で WebLogic 用の「Maximum Heap Size」が設定されないため、WebLogic Server Administration Console を使用して LiveCycle を手動で設定する必要があります。

注意：LiveCycle ES4 Service Pack 1 以降にアップグレードし、Oracle JDK 1.7 On Red Hat Enterprise Linux を使用する予定の場合は、この設定を行います。

- 1 「Configuration」タブで、「**Server Start**」タブをクリックします。
- 2 「**Arguments**」ボックスに、次に示す適切なテキストを入力して、管理対象サーバーのメモリサイズを設定します。
 - -XX:MaxPermSize=512m
- 3 「**Save**」をクリックし、「**Activate Changes**」をクリックします。

7.5.4 WebLogic 上の管理対象サーバーのメモリ設定

- 1 [WL_HOME]\user_projects\domains\[appserverdomain]\bin ディレクトリに移動します。
- 2 テキストエディターで次のファイルを編集します。
 - (Windows) setDomainEnv.cmd
 - (Linux および Solaris) setDomainEnv.sh
- 3 WLS_MEM_ARGS_64BIT=-Xms256m -Xmx512m という行を探して、WLS_MEM_ARGS_64BIT=-Xms256m -Xmx1024m に変更します。
- 4 -XX:MaxPermSize=256m という行を探して、XX:MaxPermSize=512m に変更します。
注意：64 ビット JVM for Solaris の場合のみ、MaxPermSize パラメーターを設定する必要があります。
- 5 変更内容を保存してファイルを閉じます。
注意：LiveCycle ES4 Service Pack 1 以降にアップグレード、Oracle JDK 1.7 with Red Hat Enterprise Linux を使用する予定の場合は、MaxPermSize を設定します。

7.6 WebLogic の停止と再起動

すべての設定変更を完了した後、その変更を有効にするには WebLogic を再起動します。WebLogic 管理対象サーバー、Node Manager、WebLogic Administration Server も再起動する必要があります。

64 ビット Java を使用して WebLogic Administration Server を起動する必要があります。startWeblogic.sh スクリプトを使用して WebLogic Administration Server を起動する前に、環境に JAVA_OPTIONS=-d64 が設定されていることを確認してください。

7.6.1 WebLogic 管理対象サーバーの停止

- 1 WebLogic Server Administration Console の「Domain Structure」で、ドメイン名をクリックします。
- 2 「Control」タブをクリックし、停止するサーバーの横にあるチェックボックスを選択します。
- 3 「Shutdown」をクリックし、次のいずれかのオプションを選択します。

When work completes : 選択されたサーバーが正常に停止され、管理対象サーバーは処理中のすべての要求を完了するようサブシステムに通知します。正常に停止する処理では、現在処理中の特定のアプリケーションを完了するための時間が WebLogic Server のサブシステムに与えられます。

Force Shutdown Now : 強制的な停止が開始され、管理対象サーバーは処理中の要求を直ちに破棄するようサブシステムに指示します。

- 4 WebLogic Server Administration Console のプロンプトで、「Yes」をクリックしてコマンドの実行を確認します。
- 管理対象サーバーが停止したことを確認するには、「Control」タブの一番下のテーブルを参照します。このテーブルには、すべてのサーバーのリストが表示され、それぞれの現在の状態が示されます。

7.6.2 WebLogic Administration Server の停止

- 1 コマンドプロンプトで、[appserverdomain]\bin に移動します。
- 2 次のコマンドを入力します。
 - (Windows) stopWebLogic.cmd
 - (Linux、UNIX) /stopWebLogic.sh
- 3 WebLogic のユーザー名とパスワードを入力します (WebLogic のインストール時にセキュリティを有効にした場合)。

7.6.3 WebLogic Administration Server の起動

- 1 コマンドプロンプトで [appserverdomain] に移動します。
- 2 次のコマンドを入力します。
 - (Windows) startWebLogic.cmd
 - (Linux、UNIX) /startWebLogic.sh
- 3 WebLogic のユーザー名とパスワードを入力します (WebLogic のインストール時にセキュリティを有効にした場合)。

7.6.4 WebLogic Node Manager の起動

- コマンドプロンプトで、[appserver root]/server/bin ディレクトリに移動して、該当するコマンドを入力します。
 - (Windows) startNodeManager.cmd
 - (Linux、UNIX) ./startNodeManager.sh

7.6.5 WebLogic 管理対象サーバーの起動

- 1 WebLogic Administration Server が起動されている場合は、WebLogic Server Administration Console にログインします。
- 2 「Domain Structure」で、**Environment** / **Servers** をクリックし、右側のウィンドウで管理対象サーバーをクリックします。
- 3 次の画面で、「**Control**」タブをクリックし、起動する管理対象サーバーの横にあるチェックボックスを選択します。
- 4 「**Start**」をクリックして、「**Yes**」をクリックします。

7.7 データベースの初期化に関する JMX ポリシーの作成

JMX ポリシーを作成して、LiveCycle のコアコンポーネント用のデータベースが正しく初期化されることを確認する必要があります。

次の手順を実行します。

7.7.1 MBean 認証の領域への委任

JMX ポリシーを作成する前に、セキュリティ領域が MBean へのアクセスを制御するように設定されていることを確認します。詳しくは、WebLogic Administration Console のドキュメントを参照してください。

- 1 WebLogic Administration Console で、**Domain Structure** / **Security Realms** をクリックします。
- 2 **Summary of Security Realms** ページの「Realms」リストで「**myrealm**」をクリックします。
- 3 **Configuration** / **General** ページで、「**Use Authorization Providers to Protect JMX Access**」が選択されていることを確認します。このオプションが選択されていない場合は、次の手順を実行します。
 - 「Change Center」で「**Lock & Edit**」をクリックします。
 - 「**Use Authorization Providers to Protect JMX Access**」を選択します。
 - 「**Save**」をクリックします。
 - 「Change Center」で、「**Activate Changes**」をクリックします。
 - 管理サーバーと管理対象サーバーを再起動します。

7.7.2 JMX ポリシーの作成

- 1 WebLogic Administration Console で、**Domain Structure** / **Security Realms** をクリックします。
- 2 **Summary of Security Realms** ページで、JMX ポリシーの変更対象となる領域の名前をクリックします。
- 3 **Settings** ページで、「**Roles and Policies**」タブをクリックして、「**Realm Policies**」サブタブをクリックします。
- 4 「**Policies**」テーブルの「Name」列で、「**JMX Policy Editor**」をクリックします。
- 5 JMX Policy Editor ページで、「**GLOBAL SCOPE**」オプションが選択されていることを確認します。「**Next**」をクリックします。
- 6 次のページで「**ALL MBEANS TYPES**」オプションが選択されていることを確認します。「**Next**」をクリックします。
- 7 「**Attributes: Permission to Write**」オプションを選択し、「**Create Policy**」をクリックします。
- 8 **Edit JMX Policies** ページで、「**Add Conditions**」をクリックします。
- 9 「**Predicate List**」ドロップダウンメニューの「**Role**」を選択し、「**Next**」をクリックします。

10 「Role Argument Name」ボックスに **Anonymous** と入力し、「Add」をクリックします。

注意： Anonymous ロールはデフォルトの WebLogic ロールで、すべてのランタイムプロセスユーザー（アプリケーションをブートストラップするために必要なユーザーなど）向けのロールです。

11 「Finish」をクリックします。

12 Edit JMX Policies ページで、「Save」をクリックします。

13 手順 1 ～ 6 を繰り返します。

14 JMX Policy Editor - Attributes and Operations ページで、「Unregister instances of this MBean using MBean server」オプションを選択し、「Create Policy」をクリックします。

15 手順 8 ～ 12 を繰り返します。

7.8 LiveCycle データベース接続の設定

次に、『[LiveCycle のインストールおよびデプロイ \(WebLogic 版\)](#)』の手順に従って、データベース接続を設定し、LiveCycle をインストールします。

第 8 章：事前設定環境のチェックリスト

アプリケーションサーバーインストールガイドに進む前に、システム設定に関する次の情報を記録済みであることを確認してください。

サーバー設定

事前設定	データを記録	必須？
必要システム構成が満たされているかどうか	<ここにテキストを入力>	はい
一時ディレクトリ		
JDK のインストール場所	<ここにテキストを入力>	はい
グローバルストレージディレクトリの作成	<ここにテキストを入力>	デフォルトの場所を使用しない場合のみ。

アプリケーションサーバーの設定

事前設定	データを記録	必須？
アプリケーションサーバーのホスト名または IP アドレス	<ここにテキストを入力>	はい
アプリケーションサーバーの資格情報	<ここにテキストを入力>	はい
アプリケーションサーバーのポート番号	<ここにテキストを入力>	はい
41 ページの「 JBoss Application Server の設定 」、59 ページの「 WebLogic Server の設定 」または 52 ページの「 WebSphere Application Server の設定 」の説明どおりに事前設定が完了しているか	<ここにテキストを入力>	はい

データベースの設定

事前設定	データを記録	必須？
データベースのホスト名または IP アドレス	<ここにテキストを入力>	はい
データベースアカウントの資格情報	<ここにテキストを入力>	はい
データベースのポート番号	<ここにテキストを入力>	はい
29 ページの「 LiveCycle データベースの作成 」の説明どおりに事前設定が完了しているか	<ここにテキストを入力>	はい

LDAP サーバー

事前設定	データを記録	必須？
LDAP サーバーのホスト名または IP アドレス	<ここにテキストを入力>	LDAP を使用してユーザーを認証する場合にのみ必要。
LDAP アカウントの資格情報	<ここにテキストを入力>	LDAP を使用してユーザーを認証する場合にのみ必要。

PDF Generator の設定 (Windows)

事前設定	データを記録	必須 ?
Microsoft Office のインストール	<ここにテキストを入力>	はい。Acrobat および Microsoft Office をインストールするには、管理者権限を持つユーザーアカウントを使用する必要があります。
Acrobat のインストール	<ここにテキストを入力>	はい。Acrobat および Microsoft Office をインストールするには、管理者権限を持つユーザーアカウントを使用する必要があります。
その他のネイティブアプリケーションソフトウェアのインストール	<ここにテキストを入力>	はい。
Windows 環境変数の設定	<ここにテキストを入力>	はい。
32 ビット JDK のインストール		

第9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。

注意：JBoss 版での LiveCycle ES3 の自動インストールを使用する場合は、このドキュメントを読む必要はありません。自動インストールの環境を準備するために必要なすべての手順は、『[LiveCycle の自動インストールおよびデプロイ \(JBoss 版\)](#)』に記載されています。

9.1 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。

9.1.1 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
```

9.1.2 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;
```

9.1.3 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
```

9.1.4 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
```

9.1.5 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。

9.2 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。

9.3 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。

9.3.1 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。

9.3.2 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>`

9.4 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。

9.4.1 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

9.4.2 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。

- jacob.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。

9.4.2.1 jacob.properties ファイルの更新

- 1 [**appserver root**]/server/<profile_name>/conf/jacob.properties ファイルをテキストエディターで開きます。
- 2 jacob.poa.thread_pool_max という設定を見つけて、値を 16 に変更します。
- 3 ファイルを保存して閉じます。

9.4.2.2 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。

9.4.2.3 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。

9.4.2.4 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。

9.4.2.5 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。

9.4.2.6 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="jacorb.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。

9.4.2.7 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。

9.4.2.8 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。

9.5 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-name>/lib から、ダウンロードした JBoss の [アプリサーバールート]/server/<profile_name>/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
- jgroups.jar

9.6 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。

9.6.1 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_clearite.html の手順を使用することもできます。

9.6.1.1 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)。

9.6.1.2 MySQL データベースドライバーのインストール

- 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。

9.6.1.3 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。

9.6.1.4 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]/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 [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lement 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>
```

- 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。

- 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。

- 5 <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>
```

- 6 ファイルを保存して閉じます。

9.6.1.5 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。

9.6.2 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_clearart.html の手順を使用することもできます。

9.6.2.1 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。

9.6.2.2 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。

9.6.2.3 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。

9.6.2.4 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">
        jdbc: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。

- JBoss を起動します。

9.6.3 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_clearart.html の手順を使用することもできます。

9.6.3.1 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)。

9.6.3.2 JBoss 用 SQL データベースドライバのインストール

- Microsoft の Web サイトから SQL Server JDBC 3.0 データベースドライバを取得します。

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。

- (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)。
- (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)。
- [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。
- [**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。

9.6.3.3 adobe-ds.xml ファイルの編集

- [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。
- 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>
```

- 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> エlement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。

9.6.3.4 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 **mssql-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>** エlement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>** Elementを DefaultDS に変更
- **<datasource-mapping>** Element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** Elementを独自のユーザー名に変更
- **<password>y** Element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>** Element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。

9.6.3.5 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>** Element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。

9.6.3.6 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。

9.7 次の手順

『[LiveCycle サーバーのインストールおよびデプロイ](#)』に記載されている指示に従って LiveCycle をインストールします。