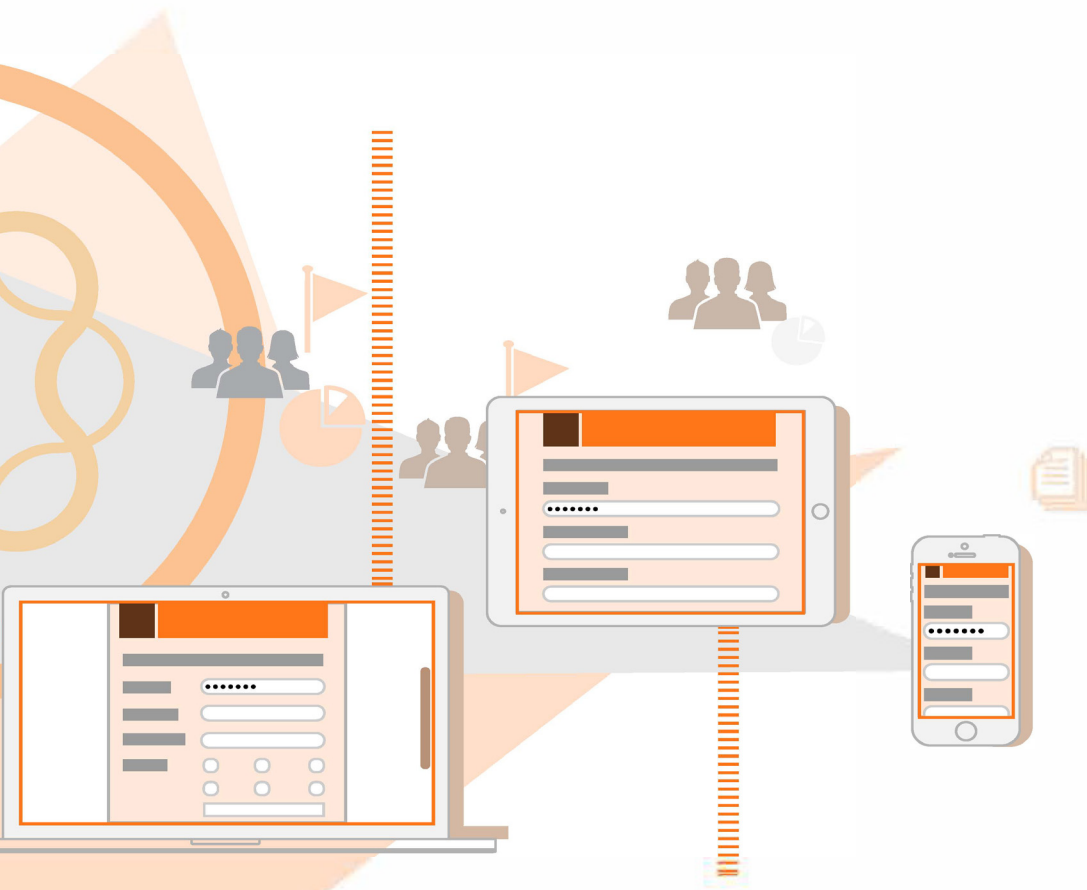


# AEM Formsのインストールの準備 (シングルサーバー)



**AEM 6.3 Forms**

## 法的通知

法律上の注意については、<https://helpx.adobe.com/jp/legal/legal-notices.html>を参照してください。

---

# 目次

<b>章1</b>	<b>このドキュメントの内容</b>	<b>1</b>
	このガイドで使用する表記	1
	追加情報	2
<b>章2</b>	<b>必要システム構成</b>	<b>3</b>
	サーバー環境の準備	3
	Windowsでのインストールに必要な権限	3
	クロックタイムの同期	3
	(オプション) その他の必要システム構成	4
<b>章3</b>	<b>AEM Formsの秘密鍵証明書および証明書</b>	<b>5</b>
	Acrobat Reader DC Extensions使用権限秘密鍵証明書の取得	5
	Digital Signaturesで使用する電子証明書の取得	6
<b>章4</b>	<b>AEM Formsデータベースの作成</b>	<b>7</b>
	データベース設定の要件	7
	データベースを使用するための最小限のユーザー権限	7
	Oracleデータベースの作成	8
	ユーザーアカウントと権限	9
	SQL Serverデータベースの作成	9
	AEM Forms用のSQL Serverのセットアップ	10
	メモリの拡張	10
	プロセッサの優先度の設定	10
	復旧間隔の拡張	10
	統合セキュリティ	11
	SQL Server データベースのサイズ変更	11

AEM Forms データベースユーザー、スキーマおよびログインの作成	11
SQL Server での AEM Forms ユーザーの作成	12
AEM Forms ユーザーとデータベースとの関連付け	13
AEM Forms データベースの分離レベルの設定	13
MySQL データベースの作成	14
MySQL ユーザーアカウント	14
MySQL 初期サーバー設定についての要件	15
<b>章5 JBoss Application Server の設定</b>	<b>16</b>
設定オプション	16
アドビにより事前設定された JBoss	16
JBoss 用 JDK のインストール	17
Linux および Solaris のログオンスクリプト	17
アドビの事前設定 JBoss のデータベース接続の設定	18
データベースパスワードの暗号化	18
アドビの事前設定 JBoss での MySQL の設定	18
lc_<db-name>.xml ファイルの編集	19
lc_<db-name>.xml ファイルの編集	19
アドビの事前設定 JBoss 用の Oracle の設定	21
lc_<db-name>.xml ファイルの編集	21
lc_<db-name>.xml ファイルの編集	22
lc_<db-name>.xml ファイルの編集	22
アドビの事前設定 JBoss 用の SQL Server の設定	24
lc_<db-name>.xml ファイルの編集	24
lc_<db-name>.xml ファイルの編集	24
Windows ログインの JEE 上の AEM forms データベースユーザーへのマップ	26
SQL Server データベースドライバのインストール	26
Windows 上での統合セキュリティの設定	26
<b>章6 WebSphere Application Server の設定</b>	<b>27</b>
WebSphere のインストール	27
WebSphere への Fix Pack のインストール	27
JAVA_HOME および PATH 環境変数	27
JAVA_HOME 環境変数の設定 (Windows)	27
PATH 環境変数の設定 (Windows)	28
DISPLAY 設定の指定 (AIX)	28
JAVA_HOME 環境変数の設定 (AIX、Linux および Solaris)	28
PATH 環境変数の設定 (AIX、Linux および Solaris)	28
(Windows、AIX、Linux または Solaris) JAVA_HOME 環境変数の検証	28

WebSphere Base の起動と WebSphere Administrative Console へのアクセス . . . . .	29
WebSphere Base の起動 . . . . .	29
WebSphere Administrative Console へのアクセス . . . . .	29
WebSphere ND の起動と WebSphere Administrative Console へのアクセス . . . . .	29
ディレクトリ権限 . . . . .	30
ファイルの抽出先の変更 . . . . .	30
WebSphere Application Server の準備 . . . . .	31
SOAP 要求タイムアウト値の延長 . . . . .	31
デプロイヤーのヒープサイズの拡大 . . . . .	31
AIX、Linux または Solaris . . . . .	32
ウィンドウ . . . . .	32
受信および送信用の通信設定 . . . . .	32
JEE 上の AEM Forms データベース接続の設定 . . . . .	33
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 . . . . .	33
新しい WebSphere Application Server ユーザーを作成するには . . . . .	33
既存の WebSphere Application Server ユーザーを設定するには . . . . .	34
EVERYONE グループを設定するには . . . . .	34
CSIv2 Inbound Transport の設定 . . . . .	34
<b>章7 WebLogic Server の設定 . . . . .</b>	<b>35</b>
JEE 上の AEM Forms の WebLogic 必要システム構成 . . . . .	35
WebLogic Server のインストール . . . . .	35
Solaris (64 ビット Java) での WebLogic の設定 . . . . .	35
JDK のインストール . . . . .	36
JDK のインストール . . . . .	36
JAVA_HOME 環境変数の設定 (Windows) . . . . .	36
PATH 環境変数の設定 (Windows) . . . . .	36
JAVA_HOME 環境変数の設定 (Linux および Solaris) . . . . .	37
PATH 環境変数の設定 (Linux および Solaris) . . . . .	37
JAVA_HOME 環境変数の検証 . . . . .	37
WebLogic Server ドメインの作成 . . . . .	38
管理対象サーバーが含まれる WebLogic ドメインの作成 . . . . .	38
管理サーバーのスニアドレスの設定 . . . . .	39
WebLogic 管理対象サーバー . . . . .	39
ノードマネージャーの起動 . . . . .	40
新しい WebLogic 管理対象サーバーの作成 . . . . .	40
メモリ設定 . . . . .	41
WebLogic 上の管理対象サーバーのメモリ設定 . . . . .	41
Web サービス用の WebLogic の設定 . . . . .	42

WebLogicの停止と再起動 . . . . .	43
WebLogic 管理対象サーバーの停止 . . . . .	43
WebLogic 管理サーバーの停止 . . . . .	44
WebLogic 管理サーバーの起動 . . . . .	44
WebLogic Node Manager の起動 . . . . .	44
WebLogic 管理対象サーバーの起動 . . . . .	44
データベースの初期化に関するJMXポリシーの作成 . . . . .	45
MBean 認証の領域への委任 . . . . .	45
JMX ポリシーの作成 . . . . .	45
スタックスレッド最大時間の延長 . . . . .	46
JEE上のAEM Formsデータベース接続の設定 . . . . .	46
<b>章8 事前設定環境のチェックリスト . . . . .</b>	<b>47</b>
サーバー設定 . . . . .	47
アプリケーションサーバーの設定 . . . . .	47
データベースの設定 . . . . .	48
LDAPサーバー . . . . .	48
PDF Generatorの設定 (Windows) . . . . .	48
<b>章9 付録- JBossの手動設定 . . . . .</b>	<b>49</b>
JBoss用JDKのインストール . . . . .	49
JAVA_HOME 環境変数の設定 (Windows) . . . . .	49
PATH 環境変数の設定 (Windows) . . . . .	49
JAVA_HOME 環境変数の設定 (Linux および Solaris) . . . . .	50
PATH 環境変数の設定 (Linux および Solaris) . . . . .	50
JAVA_HOME 環境変数の設定の検証 (Windows、Linux または Solaris) . . . . .	50
JBossの手動インストール . . . . .	50
JBossの起動と停止 . . . . .	50
JBoss の起動 . . . . .	50
JBoss の停止 . . . . .	51
JBoss設定の変更 . . . . .	51
JBoss 設定の変更 . . . . .	51
standalone.conf.bat ファイルの変更 (Windows のみ) . . . . .	51
standalone.bat ファイルの変更 (Windows のみ) . . . . .	52
jboss.cli.bat の変更 (Windows) . . . . .	53
standalone.conf ファイルの変更 (Solaris 11、Red Hat 6.5 または 7 の JBoss、 64 ビットのみ) . . . . .	53
domain.xml ファイルの変更 (Windows および Unix) . . . . .	54
host.xml ファイルの変更 . . . . .	56

modules.xml ファイルの変更	.57
JVM引数について	.57
管理対象ドメイン	.57
ドメイン設定ファイル内の JVM 設定	.57
jar ファイルのコピー	.58
手動でインストールした JBoss 用の AEM Forms データベース接続	.60
手動でインストールした JBoss での MySQL の設定	.60
MySQL データソースの設定	.60
データソースとしての MySQL の設定	.60
standardjbosscmp-jdbc.xml ファイルの編集	.61
lc_turnkey.xml ファイルを編集します	.62
手動でインストールした JBoss 用の Oracle の設定	.63
Oracle 12c データベースドライバのインストール	.63
lc_turnkey.xml ファイルを編集します	.63
データソースとしての Oracle の設定	.64
lc_turnkey.xml ファイルを編集します	.67
手動でインストールした JBoss 用の SQL Server の設定	.68
SQL Server データベース接続の設定	.68
lc_turnkey.xml ファイルを編集します	.68
データソースとしての MSSQL の設定	.69
lc_turnkey.xml ファイルを編集します	.72
Windows での統合セキュリティの設定	.73
次の手順	.73
<b>章10 付録- その他の必要システム構成</b>	<b>.74</b>
Linux および UNIX ベースのプラットフォームに関するその他の要件	.74
UTF-8 のインストールおよび設定	.74
Solaris への UTF-8 のインストール	.74
Solaris	.75
Linux	.75
Windows 以外のオペレーティングシステムでのファイル制限値の設定	.76
更新された設定の確認	.76
LDAP の設定	.77
PDF Generator に関するその他の要件	.77
Windows のユーザーアカウント	.77
Windows 以外のオペレーティングシステムのユーザーアカウント	.77
PDF Generator での 64 ビットアプリケーションサーバーの使用	.78
Windows での JAVA_HOME_32 変数の設定	.78
Windows 以外のオペレーティングシステムでの JAVA_HOME_32 変数の設定	.78

---

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	79
PDF Generator用のAcrobatのインストール	80
SHXフォントを使用するためのAcrobatの設定（Windowsのみ）	80
QuickTime 7	80
環境変数の設定	81
リモートマシン上でのPDF Generatorの設定	81
Service Control Manager コマンドラインツール	82
ヘッドレスモードの設定	82
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	82
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	83
Windows Server 2012で報告されたエラーの無効化（オプションですが推奨します）	83
Windows以外のオペレーティングシステムでOpenOfficeに必要な追加設定	83
PDF Generatorのマルチユーザーサポート	84
「プロセスレベルトークンの置き換え」権限の付与（Windowsのみ）	84
Linuxプラットフォームのシンボリックリンク	84
Solaris 11プラットフォームのシンボリックリンク	85
Red Hat Enterprise Linux 6（RHEL6）に関するその他の要件	85
マルチスレッドファイル変換のユーザーアカウントの設定	86
ユーザーアカウントの追加	86
手動によるAcrobatの使用制限	86
Connector for Documentumに関するその他の要件	86
Connector for IBM Content Managerに関するその他の要件	87
単一のIBM Content Manager データストアに対する接続の設定	87
複数のIBM Content Manager データストアに対する接続の設定	88
IBM Content Manager データストアへのマルチユーザー接続の設定	88
Connector for IBM FileNetに関するその他の要件	89
IBM FileNet 5.0	89
IBM FileNet 5.2	89
Central Migration Bridgeサービス	89
Central Migration Bridgeの使用許可	89
インストールに関する考慮事項	90
JEE上のAEM Formsの自動インストール	90
JEE上のFormsのカスタムインストール	90
JEE上のAEM FormsのIPv6サポート	91
サポートされているIPv6の設定	91
IPv6実装のガイドライン	91
JBoss用のIPv6の設定	92



# 1. このドキュメントの内容

JEE 上の AEM Forms 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。AEM Forms は次のコンポーネントで構成されます。

- サーバー機能とランタイム環境を提供する J2EE ベースの Adobe Experience Manager Forms
- JEE 上の AEM Forms アプリケーションを設計、開発、テストするためのツール
- JEE 上の AEM Forms サーバーにデプロイされる、機能サービスを提供するモジュールとサービス

このドキュメントは、[JEE 上の AEM Forms ドキュメント](#) 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、既存の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、「[JEE 上の AEM Forms のインストールおよびデプロイ（JBoss 自動インストールを使用）](#)」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
[aem-forms root]	JEE 上の AEM Forms モジュールすべてに使用されてい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、Configuration Manager、SDK およびインストールされた各 JEE 上の AEM Forms モジュールが（製品マニュアルと共に）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_Experience_Manager_Forms AIX、Linux および Solaris : /opt/adobe/Adobe_Experience_Manager_Forms
[appserver root]	AEM Forms に含まれるサービスを実行するアプリケーションサーバーのホームディレクトリ。	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>

名前	説明	デフォルト値
[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
[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 : domain_<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

次の表では、JEE 上の AEM Forms について詳しく知るために役立つリソースを紹介します。

情報	参照先
AEM Forms の概要	<a href="#">概要</a>
JEE 上の AEM Forms 用のすべてのドキュメント	<a href="#">ドキュメント</a>

## 2. 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) 「[JEE 上の AEM Forms でサポートされているプラットフォーム](#)」を参照し、お使いのソフトウェア、ハードウェア、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、その他の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。
- 2) オペレーティングシステムをインストールおよび設定し、必要なパッチとサービスパックを適用して更新します。
- 3) 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。
- 4) 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。

### 2.2. 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。

### 2.3. クロックタイムの同期

水平クラスターのすべてのコンピューターがクロックタイムを定期的に同期することを確認する必要があります。ノードの時刻が数秒以上ずれていると、JEE 上の AEM Forms のインストールに問題が発生する場合があります。

使用しているネットワークで採用されている標準的な同期方法を、JEE 上の AEM Forms クラスターのすべてのコンピューターに適用してください。

## 2.4. (オプション) その他の必要システム構成

特定の機能またはプラットフォームには、いくつかの追加要件があります。AEM Forms でこれらの機能またはプラットフォームを使用する場合は、「付録 - その他の必要システム構成」を参照し、追加要件の詳細を確認してください。

- Linux および UNIX ベースのプラットフォーム
- PDF Generator
- Central Migration Bridge サービス
- AEM Forms IPv6 サポート
- Connector for IBM File Net、Connector for Documentum、Connector for IBM Content Manager
- Forms サービス、Output サービス、ConvertPDF サービス
- JEE 上の AEM Forms と Luna HSM クラスターの併用
- LDAP の設定
- ドキュメントフォーム変数および電子署名を使用するプロセス
- AEM Forms の秘密鍵証明書および証明書

## 3. AEM Forms の秘密鍵証明書および証明書

この節では、次のタスクを実行する方法について説明します。

- Document Security 使用権限秘密鍵証明書を取得します。
- Digital Signatures で使用する電子証明書を取得します。

### 3.1. Acrobat Reader DC Extensions 使用権限秘密鍵証明書の取得

Acrobat Reader DC Extensions 使用権限秘密鍵証明書は、Acrobat Reader DC Extensions に固有な電子証明書で、これにより、PDF ドキュメントで Acrobat 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、Acrobat Reader DC 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、Acrobat Reader DC 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。

Acrobat Reader DC 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。

Acrobat Reader DC 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 Acrobat Reader DC 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：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、AEM Forms 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、AEM Forms 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（Windows 以外のオペレーティングシステム）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：AEM Forms 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](#)を参照してください。）

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。

## 4. AEM Forms 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、AEM Forms 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、AEM Forms 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。ここにはサポートされるすべてのデータベースの使用手順が記載されています。これらの手順はお使いのデータベースに対してのみ実行してください。

データベースには以下のエレメントが含まれます。

- AEM Forms サービス
- AEM Forms ランタイム設定
- AEM Forms 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。

### 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



データベース	初期化権限	ランタイム権限
SQL Server - DB レベル	Create Table Create View Connect	Connect
SQL Server - スキーマレベル	Alter 挿入 参照 選択 更新 削除	挿入 選択 更新 削除
DB2	詳細な説明については、「DB2 ユーザーアカウント」を参照してください。	詳細な説明については、「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 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。

注：AEM Forms 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。
- 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

注：Windows以外のオペレーティングシステム上にデプロイする場合、ユーザー名は8文字以下にする必要があります。Windowsにデプロイする場合は、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の使用方法について詳しくは、該当するOracleの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

AEM Forms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。

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. AEM Forms用のSQL Serverのセットアップ

AEM Forms 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。

#### 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：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、AEM Forms 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。

#### 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（AEM Forms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：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、AEM Forms 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。

#### 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：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、AEM Forms 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。

## 統合セキュリティ

注：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、『[AEM Forms のインストールの準備 \(シングルサーバー\)](#)』の「Windows での統合セキュリティの設定」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、AEM Forms 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：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、AEM Forms 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. AEM Forms 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、AEM Forms 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：AEM Forms 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、AEM Forms データベースインスタンスに使用できます。

- 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：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) 「**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、SQL Server データベースのサイズ変更を参照してください。
- 4) 「**Database Data Autogrowth**」ボックスに、50% と入力します。

- 5) 「**Database Log Initial 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、SQL Server データベースのサイズ変更を参照してください。
- 6) 「**Database Log Autogrowth**」ボックスに、50%と入力します。
- 7) 「**OK**」をクリックすると、データベースが作成されます。

### SQL Server での AEM Forms 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、AEM Forms 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：スキーマ名は [database\_username] と同じである必要があります。

- 11) **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
    - 挿入
    - 参照

- 選択
  - 更新
  - 削除
- c) 「OK」をクリックします。
- 12) 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。

### AEM Forms ユーザーとデータベースとの関連付け

AEM Forms ユーザーを作成した後に、このユーザーを AEM Forms 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て「OK」をクリックします。

### AEM Forms データベースの分離レベルの設定

AEM Forms では、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。デッドロックが発生するのは、トランザクションが長時間実行されているときに、短時間の読み取りが何度も発生する場合です。

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：[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. MySQL データベースの作成

注：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。「[自動オプションを使用した AEM Forms の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、AEM Forms で使用する MySQL 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。

注：MySQL では、ユーザー名やパスワードに特殊文字やスペースを使用することはできません。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：Windows 以外のオペレーティングシステムでは、`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/en/identifier-case-sensitivity.html> を参照してください。

注：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

### 4.5.1. 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、AEM Forms 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

#### 4.5.2. 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 (Windows 以外のオペレーティングシステム) の `mysqld` セクションでは、次の設定が必要です。

- `max_allowed_packet=64M`

注：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 (Windows 以外のオペレーティングシステム) の `mysqld` セクションでは、次のように設定することをお勧めします。

- `key_buffer_size=16M`
- `port=3306`
- `socket = /tmp/mysql.sock`
- `skip-external-locking`
- `table_open_cache = 64`
- `innodb_log_file_size=170M`
- `sort_buffer_size = 512K`
- `net_buffer_length = 8K`
- `read_buffer_size=1M`
- `read_rnd_buffer_size=1M`
- `myisam_sort_buffer_size = 35M`



## 5. 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
- JBoss の手動設定（「付録 - JBoss の手動設定」を参照してください。）

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。以下のケースに該当する場合は、[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。

- オペレーティングシステムとして Linux または Solaris を使用している場合。
- Oracle または SQL Server を使用している場合。または、自動セットアップ（部分的な自動インストール）によってデフォルトでインストールされる MySQL を使用したくない場合。
- JBoss の設定を詳細に制御する必要がある場合。

#### 5.1.1. 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、JEE 上の AEM Forms 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、「付録 - JBoss の手動設定」に記載されている設定手順を実行する必要があります。

アドビは 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 JEE 上の AEM forms 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：JBoss をデプロイすると、[appserver root]/standalone/configuration 内に、各データベースタイプのデータソースの設定ファイル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。したがって、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.xml、lc\_oracle.xml、lc\_sqlserver.xml になります。以降の節では、これらの個別のファイルを示す際に lc\_<db-name> を使用します。

Electronic Software Distribution (ESD) を Adobe の Web サイトからダウンロードした場合は、lces\_server\_6\_3\_0\_jboss\_all\_win.zip ファイル（Windows の場合）または lces\_server\_6\_3\_0\_jboss\_all\_unix.tar.gz ファイル（Linux または Solaris の場合）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：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。

事前に設定された JBoss ソフトウェアは、インストールメディアまたは /third\_party ディレクトリ内の ESD 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、「[アドビの事前設定 JBoss のデータベース接続の設定](#)」節に記載されている手順を除くすべての設定手順が完了します。MySQL を使用する場合は、この節に記載されている手順を実行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：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 JEE 上の AEM Forms 内に提供されます。製品ファイルをインストールすると、それらのファイルは [appserver root]/standalone/configuration 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。

注：standalone.sh ファイルには、Linux オペレーティングシステム上で稼働するアプリケーションサーバーの起動パラメーターが記述されています。ファイルに必要なすべてのパラメーターが含まれていることと、すべてのパラメーターが開始引用符と終了引用符（"）で囲まれていることを確認してください。

## 5.2. JBoss 用 JDK のインストール

Oracle JDK 8.0（またはバージョン 8.0 に対する新しい更新プログラム）をダウンロードしてインストールする必要があります。JDK のインストールが完了したら、以下の手順を実行します。

- 1) 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。
- 2) JAVA\_HOME/bin を指すように PATH 環境変数を設定します。
- 3) （Solaris のみ）standalone.conf ファイルを変更します。

JVM を 64 ビットモードで実行するには、[appserver root]/bin/standalone.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 -Xmx4096m -XX:MaxPermSize=768m  
-Dorg.jboss.resolver.warning=true -Dsun.rmi.dgc.client.gcInterval=3600000  
-Dsun.rmi.dgc.server.gcInterval=3600000  
-Dsun.lang.ClassLoader.allowArraySyntax=true"
```

### 5.2.1. 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。

## 5.3. アドビの事前設定 JBoss のデータベース接続の設定

JEE 上の AEM Forms 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。

- JEE 上の AEM Forms 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：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。

### 5.3.1. データベースパスワードの暗号化

lc\_<db-name> ファイルのパスワードを暗号化します。パスワードを暗号化するには、次のコマンドを使用します。

```
java -cp %JBOSS_HOME%\modules\system\layers\base\.overlays\layer-base
-jboss-eap-6.4.5.CP\org\picketbox\main picketbox-4.1.2.Final-redhat-1.jar
org.picketbox.datasource.security.SecureIdentityLoginModule <password>
```

注：JBoss のデータソースパスワードを暗号化するために使用する picketbox-4.1.2.Final-redhat-1.jar ファイルがない場合は、[/techpreview/all/org/picketbox/picketbox/4.1.2.Final-redhat-1](http://techpreview.all.org/picketbox/picketbox/4.1.2.Final-redhat-1) のインデックスからダウンロードしてください。<http://wiki.jboss.org> にある JAR ファイルは使用しないでください。

### 5.3.2. アドビの事前設定 JBoss での MySQL の設定

アドビによって事前に設定された JBoss から、JEE 上の AEM Forms データが格納されている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、JEE 上の AEM Forms のデプロイ先となる JBoss インスタンスにそのデータソースファイルをデプロイする必要があります。

lc\_<db-name>.xml ファイル内のパスワードを暗号化します。パスワードを暗号化するには、次のコマンドを使用します。

```
java -cp %JBOSS_HOME%\modules\system\layers\base\.overlays\layer-base
-jboss-eap-6.4.5.CP\org\picketbox\main picketbox-4.1.2.Final-redhat-1.jar
org.picketbox.datasource.security.SecureIdentityLoginModule <password>
```

注：JBoss のデータソースパスワードを暗号化するために使用する picketbox-4.1.2.Final-redhat-1.jar ファイルがない場合は、[/techpreview/all/org/picketbox/picketbox/4.1.2.Final-redhat-1](http://techpreview.all.org/picketbox/picketbox/4.1.2.Final-redhat-1) のインデックスからダウンロードしてください。<http://wiki.jboss.org> にある JAR ファイルは使用しないでください。

## lc\_<db-name>.xml ファイルの編集

MySQL データソースを設定する前に、「MySQL データベースの作成」の説明に従い、MySQL 上にデータベースを作成しておく必要があります。

- 1) [appserver root]\standalone\configuration\lc\_<db-name>.xml ファイルを任意のテキストエディターで開き、IDP\_DS、AEM\_DS、EDC\_DS で次の行を検索します。

```
<connection-url>
|jdbc:mysql://localhost:3306/adobe
</connection-url>
<driver>mysql</driver>
```

- 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) 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>30</max-pool-size>
```

- AEM\_DS の場合 :

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

注 : JEE 上の AEM Forms 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。例えば、IDP\_DS、AEM\_DS、EDC\_DS で、<max-pool-size> に 50 以上の値を設定します。

- 4) ファイルを保存して閉じます。

## lc\_<db-name>.xml ファイルの編集

- 1) [appserver root]\standalone\configuration\lc\_<db-name>.xml ファイルを任意のテキストエディターで開き、<authentication> エlement 内に記述されている次のコードを変更します。

```
<security-domain name="EncryptDBPassword">
    <authentication>
        <login-module
```

```

code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
    <module-option name="userName" value="adobe"/>
    <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
    <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
    </login-module>
</authentication>
</security-domain>
<security-domain name="EncryptDBPassword_IDP_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
                <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
            </login-module>
        </authentication>
    </security-domain>

    <security-domain name="EncryptDBPassword_AEM_DS">
        <authentication>
            <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
                <module-option name="userName" value="adobe"/>
                <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
                    <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                </login-module>
            </authentication>
        </security-domain>
<security-domain name="EncryptDBPassword_EDC_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
                <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
            </login-module>
        </authentication>
    </security-domain>

```

- 2) データベースに固有の値を指定して、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。

### 5.3.3. アドビの事前設定 JBoss 用の Oracle の設定

JBoss が、JEE 上の AEM Forms 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、JEE 上の AEM Forms 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。

#### lc\_<db-name>.xml ファイルの編集

- 1) [appserver root]\standalone\configuration\lc\_<db-name>.xml ファイルを任意の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>
```

- 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**：JEE 上の AEM Forms 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。

注：プラグ可能なデータベースを使用している場合は、<port> の後ろに指定されているコロン (:) をスラッシュ (/) に置き換えてください。

- 3) <connection-url> 設定に続く行で user-name 設定と password 設定を探し、データベースへの接続時にアプリケーションサーバーで使用されるユーザー名とパスワードでデフォルト値を置き換えます。
- 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>30</max-pool-size>
```

- AEM\_DS の場合：

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

注：JEE 上の AEM Forms 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。例えば、IDP\_DS、AEM\_DS、EDC\_DS で、<max-pool-size> に 50 以上の値を設定します。

- 5) ファイルを保存して閉じます。

## lc\_<db-name>.xml ファイルの編集

Oracle データベースと共に JEE 上の AEM Forms 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]/standalone/configuration 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]/standalone/configuration/lc\_<db-name>.xml ファイルを任意のテキストエディターで開き、<datasource> エlement を Oracle の接続設定に変更します。

```
<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 設定と password 設定を探し、データベースへの接続時にアプリケーションサーバーで使用されるユーザー名とパスワードでデフォルト値を置き換えます。

- 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：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：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。

## lc\_<db-name>.xml ファイルの編集

- 1) [appserver root]/standalone/configuration/lc\_<db-name>.xml ファイルを任意のテキストエディターで開き、<authentication> エlement 内に記述されている次のコードを変更します。

Replace :

```
<security-domain name="EncryptDBPassword">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_IDP_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_AEM_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_EDC_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
```

- 2) データベースに固有の値を指定して、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：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。



### 5.3.4. アドビの事前設定 JBoss 用の SQL Server の設定

JBoss から、JEE 上の AEM Forms データが格納されている 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、JEE 上の AEM Forms のデプロイ先となる JBoss のインスタンスにそのデータソースファイルをデプロイする必要があります (例: [appserver root]\standalone\configuration\).

#### lc\_<db-name>.xml ファイルの編集

- 1) [appserver root]/standalone/configuration/lc\_<db-name>.xml ファイルを任意の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>
```

- 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**: JEE 上の AEM Forms 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 設定と password 設定を探し、データベースへの接続時にアプリケーションサーバーで使用されるユーザー名とパスワードでデフォルト値を置き換えます。
- 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>30</max-pool-size>
```

- AEM\_DS の場合:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

注: JEE 上の AEM Forms 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。例えば、IDP\_DS、AEM\_DS、EDC\_DS で、<max-pool-size> に 50 以上の値を設定します。

- 5) ファイルを保存して閉じます。

#### lc\_<db-name>.xml ファイルの編集

- 1) [appserver root]/standalone/configuration/lc\_<db-name>.xml ファイルを任意のテキストエディターで開き、<authentication> エlement 内に記述されている次のコードを変更します。



Replace:

```
<security-domain name="EncryptDBPassword">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_IDP_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>

<security-domain name="EncryptDBPassword_AEM_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>

<security-domain name="EncryptDBPassword_EDC_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="5c2f412d6fa61722"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
```

- 2) データベースに固有の値を指定して、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。

## Windows ログインの JEE 上の AEM forms 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、JEE 上の AEM Forms 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 JEE 上の AEM Forms 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。

## SQL Server データベースドライバーのインストール

注：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 (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/ から [appserver root]\modules\system\layers\base\com\microsoft\main ディレクトリにコピーしてください。

注：サポートされているドライバーのバージョンについては、「[JEE 上の AEM Forms 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。

## Windows 上での統合セキュリティの設定

注：JBoss サービスは、[Windows ログインの JEE 上の AEM forms 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) [appserver root]\standalone\configuration\ に保管されている lc\_<db-name>.xml ファイルを編集用として開き、integratedSecurity=true というコードを接続 URL に追加します。以下に例を示します。  
`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 4.0 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\_4.0/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\_4.0/enu/auth/x64 (64 ビットオペレーティングシステムの場合) です。

注：サポートされているドライバーのバージョンについては、「[JEE 上の AEM Forms 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。

- 4) JBoss for Adobe Experience Manager Forms 6.3 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、[Windows ログインの JEE 上の AEM forms 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。

## 6. WebSphere Application Server の設定

この章では、JEE 上の AEM Forms 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

JEE 上の AEM Forms 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。JEE 上の AEM Forms 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 8.5 のインストール後、JEE 上の AEM Forms をデプロイする前に WebSphere 8.5.5.8 に更新する必要があります。

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

#### 6.1.2. 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、JEE 上の AEM Forms がデプロイされる JDK を指定できます。

##### 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\8.0
```

### 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;
```

### 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：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。

### 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/8.0  
export JAVA_HOME
```

注：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。

### 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
```

### (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。

### 6.1.3. WebSphere Base の起動と WebSphere Administrative Console へのアクセス

#### WebSphere Base の起動

- 1) 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**

#### WebSphere Administrative Console へのアクセス

- 1) WebSphere Application Server を起動します。
- 2) 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

使用中のサーバーにログインしている場合は、[hostname] を実際の IP アドレスまたはホスト 名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。

### 6.1.4. 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。
- 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。

### 6.1.5. ディレクトリ権限

JEE 上の AEM Forms 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：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。

#### 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 forms workflow / 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、JEE 上の AEM Forms 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：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詳しくは、「[EJBDeploy - Optional feature in WebSphere Application Server Version 8](#)」を参照してください。

## 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 -Xmx4096M
- 4) ファイルを保存して閉じます。

## ウィンドウ

- 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。
- 1) ファイルを保存して閉じます。

### 6.2.3. 受信および送信用の通信設定

注：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. JEE 上の AEM Forms データベース接続の設定

次に、「[JEE 上の AEM Forms 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、JEE 上の AEM Forms 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。

### 6.4.1. 新しい WebSphere Application Server ユーザーを作成するには

- 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。
- 「**Roles**」で、すべてのロールを選択します。
- 「**Search and Select Users**」で、「**User Realm**」を選択します。
- 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：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。

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

### 6.4.2. 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。

### 6.4.3. 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：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、割り当てする必要があります。

- 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。

### 6.4.4. 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 管理コンソールにログインします。
- 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。

注：Global Security を有効にし、アプリケーションサーバーに対して適切なロールを設定したら、追加の設定を行って、JEE スタック内の OSGi 機能を有効にする必要があります。これらの設定処理は、JEE 上の AEM Forms を WebSphere アプリケーションサーバーにインストールして設定が完了してから実行します。詳しい手順については、「[WebSphere Global Administrative Security の有効化](#)」を参照してください。

## 7. WebLogic Server の設定

### 7.1. JEE 上の AEM Forms の WebLogic 必要システム構成

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、「JEE 上の AEM Forms 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](http://download.oracle.com/docs/cd/E14571_01/wls.htm) を参照してください）。

注：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：64 ビット Java を使用して WebLogic 管理サーバー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 管理サーバーを起動する前に、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 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、JEE 上の AEM Forms 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¥jdk1.8.0_74
```

注：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：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/Java/jdk1.8.0_74
export JAVA_HOME
```

**Solaris :**

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

注：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：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す。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す。

**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：**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) **Oracle JDK 1.8**以降のロケーション。  
例：C:\Program Files\Java\jdk1.8.0\_XX (Windows) または /opt/Java/jdk1.8.0\_XX (Linux)
  - (Solaris) **Oracle JDK 1.8**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：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。

### 管理サーバーのリスンアドレスの設定

- 1) WebLogic Server 管理コンソールの「Domain Structure」で、**Environment** / **Servers** をクリックします。
- 2) 「**Change Center**」で「**Lock & Edit**」をクリックします。
- 3) 「**Servers table**」で「**AdminServer**」をクリックします。
- 4) 「リスンアドレス」ボックスにホスト名または IP アドレスを入力します。
- 5) 「**Save**」と「**Activate Changes**」をクリックします。
- 6) 管理サーバーを再起動します。

注：管理サーバーの特定の IP アドレスまたはホスト名を入力します。「Listen Address」ボックスには必ず入力してください。ボックスを空のままにすると、管理対象サーバーの起動時の問題や設定上の問題が生じる可能性があります。

## 7.5. WebLogic 管理対象サーバー

JEE 上の AEM Forms 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してください。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：64 ビット Java を使用して WebLogic 管理サーバー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 管理サーバーを起動する前に、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：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 管理コンソールの 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。

ヒント：WebLogic 12.1.3 および 12.2.1 では、[appserver root]\bin ディレクトリに移動します。

(Windows) startNodeManager.cmd

(UNIX) ./startNodeManager.sh

注：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 管理サーバー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 管理コンソール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：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ので、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) 「**保存**」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 管理コンソールを使用して JEE 上の AEM Forms を手動で設定する必要があります。

注：JEE 上の AEM Forms 以降にアップグレードし、Oracle JDK 1.8 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=1024m`
- 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 -Xmx4096m` に変更します。
- 4) `-XX:MaxPermSize=256m` という行を探して、`XX:MaxPermSize=1024m` に変更します。

注：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。

注：JEE 上の AEM Forms 以降にアップグレード、Oracle JDK 1.8 with Red Hat Enterprise Linux を使用する予定の場合は、MaxPermSize を設定します。

- 5) 変更内容を保存してファイルを閉じます。

## 7.6. Web サービス用の WebLogic の設定

JEE 上の AEM Forms が Web サービスを使用して要求を受け入れるようにするには、サーブレットコンテナの認証を変更する次の手順を実行する必要があります。

- 1) WebLogic 管理サーバーを起動します。
- 2) コマンドプロンプトを開き、次の例のように setWLSEnv を使用して環境を設定します。
  - (Windows) [WL\_HOME]\wlserver\_¥server¥bin¥ フォルダから、setWLSEnv.cmd と入力します
  - (Linux および Solaris) [WL\_HOME/wlserver\_/server/bin/ ディレクトリから、setWLSEnv.sh と入力します
- 3) 次のコマンドを入力して WebLogic のスクリプトツールを起動します。

```
java weblogic.WLST
```

「Exception in thread "main" java.lang.NoClassDefFoundError: weblogic/WLST」と表示された場合は、次のタスクを実行します。

- コマンドプロンプトで、<WL\_Home>/wlserver\_/server/bin/setWLSEnv.cmd を使用して環境を設定します
- コマンドプロンプトで、setWLSEnv.cmd または ./setWLSEnv.sh コマンドから返されたクラスパスをコピーして、export CLASSPATH を追加します。次に例を示します。

```
CLASSPATH="C:\PROGRA~1\Java\JDK18~1.0_7\lib\tools.jar;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\lib\weblogic_sp.jar;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\lib\weblogic.jar;C:\Oracle\MIDDLE~1\ORACLE~1\oracle_common\modules\net.sf.antcontrib_1.1.0.0_1-0b3\lib\ant-contrib.jar;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\modules\features\oracle.wls.common.nodemanager_2.0.0.0.jar;
```

- コマンドプロンプトで、setWLSEnv.cmd または ./setWLSEnv.sh から返されたパスをコピーして、export PATH を追加します。次に例を示します。

```
PATH=";C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\native\win\x64;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\bin;C:\Oracle\MIDDLE~1\ORACLE~1\oracle_common\modules\org.apache.ant_1.9.2\bin;C:\PROGRA~1\Java\JDK18~1.0_7\jre\bin;C:\PROGRA~1\Java\JDK18~1.0_7\bin;C:\PROGRA~3\Oracle\Java\javapath;C:\PROGRA~1\Java\JDK18~1.0_7\bin;C:\Windows\System32;C:\Windows;C:\Windows\System32\wbem;C:\Windows\System32\WINDOW~1\v1.0\;C:\PROGRA~1\Perforce;C:\PROGRA~2\IBM\db2cmv8\inso;C:\PROGRA~2\IBM\db2cmv8;C:\PROGRA~2\IBM\db2cmv8\bin;C:\PROGRA~2\IBM\db2cmv8\dll;C:\PROGRA~1\Perforce\Server;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\native\win\x64\oci920_8;C:\Oracle\MIDDLE~1\ORACLE~1\oracle_common\modules\org.apache.maven_3.0.5\bin"
```

- WLST で次のコマンドを入力し、サブレットコンテナの認証を更新します。

```
connect('<WebLogic username>','<WebLogic password>','<WebLogic URL>')
edit()
startEdit()
cd('SecurityConfiguration')
cd('<domain name>')
set('EnforceValidBasicAuthCredentials','false')
activate()
exit()
```

注：WebLogic URL は、`t3://hostname:[port]` の形式で指定します。`[port]` は管理サーバーポート（通常 7001）です。

## 7.7. 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 管理サーバーも再起動する必要があります。

64 ビット Java を使用して WebLogic 管理サーバー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 管理サーバーを起動する前に、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。

### 7.7.1. WebLogic 管理対象サーバーの停止

- 1) WebLogic Server 管理コンソールの「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 管理コンソールのプロンプトで、「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。

### 7.7.2. WebLogic 管理サーバーの停止

- 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)。

### 7.7.3. WebLogic 管理サーバーの起動

- 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)。

### 7.7.4. 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。

ヒント：WebLogic 12.1.3 では、[appserver root]\bin ディレクトリに移動します。

(Windows) startNodeManager.cmd

(Linux、UNIX) ./startNodeManager.sh

### 7.7.5. WebLogic 管理対象サーバーの起動

- 1) WebLogic 管理サーバーが起動されている場合は、WebLogic Server 管理コンソールにログインします。
- 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。

## 7.8. 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、JEE 上の AEM Forms 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ようにする必要があります。

次の手順を実行します。

### 7.8.1. 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 管理コンソールのドキュメントを参照してください。

- 1) WebLogic 管理コンソールで、**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。
  - 「保存」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。

### 7.8.2. JMX ポリシーの作成

- 1) WebLogic 管理コンソールで、**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。「次へ」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。「次へ」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：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。

### 7.8.3. スタックスレッド最大時間の延長

- 1) WebLogic 管理コンソールで、**Environment** / **Servers** / **Admin Server** を選択します。
- 2) 右側のウィンドウで「Admin Server」をクリックします。「Configuration」タブで、「Tuning」タブを開きます。
- 3) 左側のウィンドウで「Lock and Edit」をクリックして、スタックスレッドの最大値を 1200 に設定します。
- 4) 「Save」をクリックし、「Activate Changes」をクリックします。
- 5) WebLogic 管理サーバーを再起動します。

## 7.9. JEE 上の AEM Forms データベース接続の設定

次に、「[JEE 上の AEM Forms 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、JEE 上の AEM Forms をインストールします。

## 8. 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。

### 8.1. 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>	Yes
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>	Yes
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。

### 8.2. 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>	Yes
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>	Yes
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>	Yes
「JBoss Application Server の設定」または「WebLogic Server の設定」または「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>	Yes

### 8.3. 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>	Yes
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>	Yes
データベースのポート番号	<ここにテキストを入力>	Yes
「 <a href="#">AEM Forms データベースの作成</a> 」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>	Yes

### 8.4. 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。

### 8.5. 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		



## 9. 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。

AEM Forms は、Windows Server 2012 (Enterprise または Standard Edition)、Red Hat Linux ES / AS 6.5 または 7、SUSE Linux ES 12 のプラットフォームおよび Solaris 11 の JBoss 上で動作します。

注：JBoss 版での AEM Forms 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、『[AEM Forms 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 8.0 以降の更新バージョンは、<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\jdk1.8.0_74
```

#### 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 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
```

### 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
```

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。

### 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) domain.bat -c [config\_file\_name] -b [server\_IP\_Address]
  - (Linux および Solaris) ./standalone.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) jboss-cli.bat --connect command=:shutdown
  - (Linux および Solaris) ./jboss-cli.sh --connect command=:shutdown

## 9.4. JBoss 設定の変更

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。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]\standalone\configuration 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]\domain\ 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. JBoss 設定の変更

JBoss 設定を変更して Adobe Experience Manager Forms 用に JBoss をカスタマイズするには、次の手順を実行します。

- EAR ファイルのクラスローディングの分離の変更
- standalone.conf.bat ファイルの変更 (Windows)
- standalone.conf の変更 (Linux および Solaris)

#### standalone.conf.bat ファイルの変更 (Windows のみ)

- 1) [appserver root]/bin/standalone.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、64 ビット JVM のメモリ引数を追加します。

```
set "JAVA_OPTS=-Xms64M -Xmx512M -XX:MaxPermSize=256M"
```

- 64 ビット JVM のメモリ引数：

```
set ""JAVA_OPTS=%JAVA_OPTS% -Xms1024M -Xmx2048M -XX:MaxPermSize=768M"
```

3) 64ビットJVMの次の引数を追加します。

```
set "JAVA_OPTS=%JAVA_OPTS% -Dadobeidp.serverName=server1
-Dadobe.cache.multicast-port=33891 -Dfile.encoding=utf8
-Djava.net.preferIPv4Stack=true"
set "JAVA_OPTS=%JAVA_OPTS% -DentityExpansionLimit=10000"
set "JAVA_OPTS=%JAVA_OPTS% -XX:+HeapDumpOnOutOfMemoryError"
set "JAVA_OPTS=%JAVA_OPTS% -Dorg.jboss.net.protocol.file.useURI=false
-Dorg.jboss.as.logging.per-deployment=false"
set "JAVA_OPTS=%JAVA_OPTS% -XX:+UseCompressedOop"
```

4) (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：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.PlainDatagram-
SocketImpl.socketSetOption(Native Method)at java.net.PlainDatagramSocketImpl.setOption
(PlainDatagramSocketImpl.java:260)"
```

5) ファイルを保存して閉じます。

### standalone.bat ファイルの変更 (Windowsのみ)

1) [appserver root]/bin/standalone.bat ファイルをエディターで開きます。

2) rem Setup JBoss specific properties の直後の行を次のように変更し、UTF-8へのエンコードを設定します。

```
set JAVA_OPTS=%JAVA_OPTS% -Dprogram.name=%PROGNAME%
```

注：この新しい行に改行が含まれないようにしてください。

注：メモリ引数はドメインコントローラーとプロセスコントローラーの両方を対象とし、クラスターの各ノードに設定する必要があります。

1) (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：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.PlainDatagram-
SocketImpl.socketSetOption(Native Method)at java.net.PlainDatagramSocketImpl.setOption
(PlainDatagramSocketImpl.java:260)"
```

2) ファイルを保存して閉じます。

**jboss.cli.bat の変更 (Windows)**

- 1) [JBoss\_root]\bin\jboss-cli.bat ファイルを開いて編集します。
- 2) 次の行の太字のテキストに二重引用符 (“”) を追加します。

```
set "JAVA_OPTS=%JAVA_OPTS% -Djboss.modules.system.pkgs=com.sun.java.swing
-Dlogging.configuration=file: %JBoss_HOME%\bin\jboss-cli-logging.properties"
```

- 3) ファイルを保存して閉じます。

**standalone.conf ファイルの変更 (Solaris 11、Red Hat 6.5 または 7 の JBoss、64 ビットのみ)**

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：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/standalone.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：このエントリが standalone.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。

**domain.xml ファイルの変更 (Windows および Unix)**

- 1) [JBoss\_root]/domain/configuration/domain\_<DBType>.xml ファイルを開いて編集します。
- 2) `<property name="java.net.preferIPv4Stack" value="true"/>` ノードの子として、次のシステムプロパティを追加します。

```
<property name="com.arjuna.ats.arjuna.allowMultipleLastResources" value="true"/>
<property name="org.apache.catalina.connector.URI_ENCODING" value="UTF-8"/>
```

- 3) domain.xml ファイルのプロパティには、default、ha、full、full-ha の4つのプロファイルが含まれます。JBoss アプリケーションサーバーのクラスターに構成された JEE 上の AEM Forms には、full プロファイルのみが必要です。次のノードを検索して削除します。

- `<profile name="default">`
- `<profile name="ha">`
- `<profile name="full-ha">`

- 4) ロギングプロパティを追加またはアップデートするには：
  - a) `<subsystem xmlns="urn:jboss:domain:logging:1.3">` ノードで、子ノード `<append value="true"/>` の値を false に設定します。
  - b) `</periodic-rotating-file-handler>` ノードの後に、次のコードを追加します。

```
<logger category="com.adobe">
    <level name="INFO"/>
</logger>
<logger category="org.springframework">
    <level name="WARN"/>
</logger>
<logger category="org.mc4j.ems">
    <level name="WARN"/>
</logger>
```

- c) `<subsystem xmlns="urn:jboss:domain:ee:1.1">` ノードの子として、次のコードを追加します。

```
<global-modules>
    <module name="org.jgroups" slot="main"/>
    <module name="org.jacorb" slot="main"/>
</global-modules>
```

- 5) 次のテキストを探し、値を追加するか、太字で表示されている値を 100 に変更します。

```
<strict-max-pool name="slsb-strict-max-pool" max-pool-size="20"
instance-acquisition-timeout="5" instance-acquisition-timeout-unit="MINUTES"/>
```

- 6) `<subsystem xmlns="urn:jboss:domain:jacorb:1.3">` ノードを探し、子ノード `<initializers transactions="spec" security="identity"/>` を `<initializers security="identity" transactions="spec"/>` に置き換えます。

- 7) ノードを探し、

```
<orb socket-binding="jacorb" ssl-socket-binding="jacorb-ssl">
  <initializers security="identity" transactions="spec"/>
</orb>
```

その後に次の行を追加します。

```
<properties>
<property name="jacorb.connection.client.pending_reply_timeout" value="360000"/>
</properties>
```

- 8) 次のテキストを探し、太字で表示されているテキストを削除します。

```
<!--<remoting-connector use-management-endpoint="false"/>-->
```

- 9) `<subsystem xmlns="urn:jboss:domain:messaging:1.4">` ノードで、子ノード `<address-settings>` を探します。その子ノードで、`<address-full-policy>PAGE</address-full-policy>` 行の後に次のテキストを追加します。

```
<page-size-bytes>2097152</page-size-bytes>
```

- 10) `<subsystem xmlns="urn:jboss:domain:remoting:1.1">` ノードで、太字で表示されている値を削除します。

```
<connector name="remoting-connector" socket-binding="remoting"
security-realm="ApplicationRealm"/>
```

- 11) `<interfaces>` ノードの子として、次の行を追加します。

```
<interface name="any">
  <any-ipv4-address/> <!-- 0.0.0.0 -->
</interface>
```

- 12) すべてのプロパティで、すべての `<socket-binding-group>` ノードのすべての子ノードを探して削除します。

- 13) すべてのプロパティで、`<socket-binding-groups>` ノードの子ノードを探して置換します。

```
<socket-binding-group name="full-sockets" default-interface="public">
  <!-- Needed for server groups using the 'default' profile -->
  <socket-binding name="http" interface="any" port="8080"/>
  <socket-binding name="https" interface="any" port="8443"/>
<socket-binding name="jacorb" interface="public" port="3528"/>
  <socket-binding name="remoting" interface="any" port="4447"/>
</socket-binding-group>
```

- 14) すべてのプロパティで、`<socket-binding-group name="ha-sockets" default-interface="public">` ノードのすべての子ノードを探して削除します。

- 15) すべてのプロパティで、`<socket-binding-group name="full-ha-sockets" default-interface="public">` ノードのすべての子ノードを探して削除します。

- 16) <server-groups> ノードの次の子ノードを削除します。

```
<server-group name="other-server-group" profile="full-ha">
  <jvm name="default">
    <heap size="1303m" max-size="1303m"/>
    <permgen max-size="256m"/>
  </jvm>
  <socket-binding-group ref="full-ha-sockets"/>
</server-group>
```

- 17) ヒープサイズを<heap size="1303m" max-size="1303m"/>から<heap size="1024m" max-size="4096m"/>に変更します。

- 18) perm サイズを<permgen max-size="256m"/>から<permgen max-size="1024m"/>に変更します。

- 19) ファイルを保存して閉じます。

注: domain\_db.xml ファイルで指定されたメモリ引数は、一部の設定パラメーターが host.xml に設定されていない限り、other-server-group の各ノードで server process fork に使用されます。

## host.xml ファイルの変更

- 1) [JBoss\_root]/domain/configuration/hosts.xml ファイルを開いて編集します。

- 2) <jvms> ノードを探し、太字の値を変更します。

```
<jvms>
  <jvm name="default">
    <heap size="1024m" max-size="2048m"/>
    <permgen size="256m" max-size="768m"/>
    <jvm-options>
      <option value="-server"/>
    </jvm-options>
  </jvm> </jvms>
```

- 3) <server> ノードを探し、次の子ノードを削除します。

```
<server name="server-two" group="main-server-group" auto-start="true">
  <!-- server-two avoids port conflicts by incrementing the ports in
    the default socket-group declared in the server-group -->
  <socket-bindings port-offset="150"/>
</server>
<server name="server-three" group="other-server-group" auto-start="false">
  <!-- server-three avoids port conflicts by incrementing the ports in
    the default socket-group declared in the server-group -->
  <socket-bindings port-offset="250"/>
</server>
```

- 4) ファイルを保存して閉じます。

注: host.xml で指定されたメモリ引数は、domain\_db.xml ファイルで指定されたメモリ引数より優先されます。



## modules.xml ファイルの変更

- 1) module.xml ファイルを \Program Files\EAP-6.4.0\jboss-eap-6.4\modules\system\layers\base\sun\jdk\main ディレクトリから編集用を開きます。
- 2) 次のタグを <paths> タグ内のファイルに追加します。

```
<path name="com/sun/org/apache/xerces/internal/jaxp"/>
<path name="sun/net/util"/>
```

- 3) 保存して閉じます。

## 9.5. JVM 引数について

domain.conf.bat ファイルで定義されているメモリー引数は、ドメインコントローラーおよびプロセスコントローラーののプロセスに適用することができます。クラスターの各ノードに、これらのメモリー引数を必ず設定するようにしてください。

管理ドメインにおいて、JVM 設定は host.xml および domain.xml 設定ファイル内で宣言されます。サーバーの起動および停止プロセスを行うドメインコントローラーコンポーネントは、これらの設定を使用します。スタンドアロンサーバーインスタンスにおいて、サーバー起動プロセスで起動時にコマンドライン設定を渡すことができます。これらの設定は、コマンドラインから、または Management Console の「システムプロパティ」画面から宣言することができます。

### 9.5.1. 管理対象ドメイン

管理対象ドメインの重要な機能には、複数レベルでの JVM 設定の定義機能があります。ホストレベルや、サーバーグループ単位、サーバーインスタンス単位でカスタムの JVM 設定を構成することができます。特殊な子要素が親構成をオーバーライドするほど、グループやホストレベルにおいて例外を必要とせずに特定のサーバー構成の宣言が可能になります。これにより、設定が構成ファイルで宣言されるか、ランタイムで渡されるかのいずれかの時点まで、親構成を他のレベルに継承できるようになります。

### 9.5.2. ドメイン設定ファイル内の JVM 設定

次の例では、domain\_db.xml 設定ファイルのサーバーグループの JVM 宣言を示しています。

```
<server-groups>
<server-group name="main-server-group" profile="default">
<jvm name="default">
<heap size="64m " max-size="512m "/>
</jvm >
<socket-binding-group ref="standard-sockets"/>
</server-group>
</server-groups>
```

このインスタンス内では、`main-server-group` という名前のサーバーグループは、ヒープサイズ 64 メガバイト、最大ヒープサイズ 512 メガバイトを宣言しています。このグループに属するサーバーは、これらの設定を引き継ぎます。これらの設定は、グループ全体、ホスト単位、または個別のサーバーごとに変更することができます。

次の例では、`host.xml` 設定ファイルのサーバーグループの JVM 宣言を示しています。

```
<servers>
<server name="server-one" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="64m " max-size="256m "/>
</jvm >
</server>
</servers>
```

このインスタンス内では、`server-one` という名前のサーバーは `main-servergroup` という名前のサーバーグループに属し、デフォルトの JVM グループから JVM 設定を継承しています。以前の例では、`main-server-group` のメインヒープサイズは 512 メガバイトに設定されています。最大ヒープサイズの下限である 256 メガバイトを宣言することによって、`server-one` は `domain.xml` の設定を必要なレベルに微調整することができます。

## 9.6. jar ファイルのコピー

- 1) 次の JAR ファイルを `[DVD root]\third_party\jboss.zip\[JBoss_root]\modules\system\layers\base\com\adobe\lifecycle\main` から、ダウンロードした JBoss の `[JBoss_root]\modules\system\layers\base\com\adobe\lifecycle\main` ディレクトリにコピーします。

- `certjFIPS.jar`
- `cglib.jar`
- `jsafeFIPS.jar`
- `jsafeJCEFIPS.jar`
- `module.xml`

注：`module.xml` には、すべての jar ファイルのパスエントリが含まれます。

- 2) 使用するサーバー上に、フォルダー構造 `[JBoss_root]\modules\system\layers\base\com\microsoft\main` を作成します。

次の JAR ファイルを `[DVD root]\third_party\jboss.zip\[JBoss_root]\modules\system\layers\base\com\microsoft\main` から、ダウンロードした JBoss の `[JBoss_root]\modules\system\layers\base\com\microsoft\main` ディレクトリにコピーします。

- `sqljdbc4.jar`
- `module.xml`

注：`module.xml` には、すべての jar ファイルのパスエントリが含まれます。

- 3) 使用するサーバー上に、フォルダー構造 [JBoss\_root]\modules\system\layers\base\com\mysql\main を作成します。

次の JAR ファイルを [DVD root]\third\_party\jboss.zip\[JBoss\_root]\modules\system\layers\base\com\mysql\main から、ダウンロードした JBoss の [JBoss\_root]\modules\system\layers\base\com\mysql\main ディレクトリにコピーします。

- mysql-connector-java-bin.jar
- module.xml

注：module.xml には、すべての jar ファイルのパスエントリが含まれます。

- 4) 使用するサーバー上に、フォルダー構造 [JBoss\_root]\modules\system\layers\base\com\oracle\main を作成します。

次の JAR ファイルを [DVD root]\third\_party\jboss.zip\[JBoss\_root]\modules\system\layers\base\com\oracle\main から、ダウンロードした JBoss の [JBoss\_root]\modules\system\layers\base\com\oracle\main ディレクトリにコピーします。

- ojdbc6.jar
- module.xml

注：module.xml には、すべての jar ファイルのパスエントリが含まれます。

- 5) 使用するサーバー上に、フォルダー構造 [JBoss\_root]\modules\system\layers\base\org\apache\commons を作成します。

次の JAR ファイルを [DVD root]\third\_party\jboss.zip\[JBoss\_root]\modules\system\layers\base\org\apache\commons\validator\main から、ダウンロードした JBoss の [JBoss\_root]\modules\system\layers\base\org\apache\commons ディレクトリにコピーします。

- commons-validator.jar
- commons-validator.jar.index
- module.xml

注：module.xml には、すべての jar ファイルのパスエントリが含まれます。

- 6) 使用するサーバー上に、フォルダー構造 [JBoss\_root]\modules\system\layers\base\org\hibernate\3 を作成します。

次の JAR ファイルを [DVD root]\third\_party\jboss.zip\[JBoss\_root]\modules\system\layers\base\org\hibernate\3 から、ダウンロードした JBoss の [JBoss\_root]\modules\system\layers\base\org\hibernate\3 ディレクトリにコピーします。

- hibernate-annotations.jar
- hibernate-entitymanager.jar
- module.xml

注：module.xml には、すべての jar ファイルのパスエントリが含まれます。

## 9.7. 手動でインストールした JBoss 用の AEM Forms データベース接続

AEM Forms 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。

- AEM Forms 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をインストールする必要があります。ドライバは、Configuration Manager およびアプリケーションサーバーが AEM Forms データベースに接続できるようにするために必要です。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：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。

### 9.7.1. 手動でインストールした JBoss での MySQL の設定

JBoss から AEM Forms 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 ドライバを取得して、AEM Forms 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、AEM Forms をデプロイする JBoss のインスタンスにデプロイします。
- lc\_<db-name>.xml ファイル内のパスワードを暗号化します。パスワードを暗号化するには、次のコマンドを使用します。

```
java -cp %JBOSS_HOME%\modules\system\layers\base\overlays\layer-base-jboss-eap-6.4.5.CP\org\picketbox\main picketbox-4.1.2.Final-redhat-1.jar org.picketbox.datasource.security.SecureIdentityLoginModule <password>
```

注：JBoss のデータソースパスワードを暗号化するために使用する picketbox-4.1.2.Final-redhat-1.jar ファイルがない場合は、[/techpreview/all/org/picketbox/picketbox/4.1.2.Final-redhat-1](http://techpreview.all.org/picketbox/picketbox/4.1.2.Final-redhat-1) のインデックスからダウンロードしてください。<http://wiki.jboss.org> にある JAR ファイルは使用しないでください。

### 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）。

### データソースとしての MySQL の設定

- 1) データベースプロファイルを [DVD root]\third\_party\jboss.zip\ [JBoss\_root]\standalone\configuration から [AppServer\_root]\standalone\configuration ディレクトリにコピーします。
- 2) [JBoss\_root]/domain/configuration/domain\_oracle.xml ファイルを編集用に開きます。

3) `<datasources>` タグを探し、すべての子ノードを削除します。

```
<datasources>
    <datasource jta="true" jndi-name="java:/IDP_DS"
pool-name="IDP_DS" enabled="true" use-java-context="true">
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>
<driver>mysql</driver>
<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
<pool>
    <min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
</pool>
<security>
    <user-name>adobe</user-name>
<password>password</password>
</security>
<validation>
    <valid-connection-checker
class-name="org.jboss.jca.adapters.jdbc.extensions.novendor.JDBC4ValidConnection
Checker"/>
    <check-valid-connection-sql>SELECT count(*) from
DUAL</check-valid-connection-sql>
    <!--exception-sorter
class-name="com.mysql.jdbc.integration.jboss.ExtendedMysqlExceptionSorter"/-->
<exception-sorter
class-name="org.jboss.jca.adapters.jdbc.extensions.mysql.MySQLExceptionSorter"/>
</validation>
    <timeout>
<blocking-timeout-millis>20000</blocking-timeout-millis>
<idle-timeout-minutes>2</idle-timeout-minutes>
</timeout>
<statement>
<prepared-statement-cache-size>20</prepared-statement-cache-size>
</statement>
</datasource>
```

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**、**jbosssdb**、**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) ファイルを保存して閉じます。

### standardjbosscmp-jdbc.xml ファイルの編集

1) [appserver root]\server\<profile\_name>\conf\<profile\_name>jbosscmp-jdbc.xml ファイルを開き、次のエレメントを変更します（これらのエレメントは連続しているとは限りません）。

```
<!-- optional since 4.0 <datasource-mapping>Hypersonic SQL </datasource-mapping> -->
<fk-constraint>false</fk-constraint>
```

- `<datasource-mapping>` タグから `<!--optional since 4.0` テキストおよび末尾の `-->` コメントタグを削除して、コメントを解除します。
- `<datasource-mapping>` エレメントで Hypersonic SQL を MySQL に置き換えます。

最終的に、`<datasource-mapping>` タグは次の行のようになります。

```
<datasource-mapping>MySQL</datasource-mapping>
```

- `<fk-constraint>` エレメントで `false` を `true` に置き換えます。

2) ファイルを保存して閉じます。

## lc\_turnkey.xml ファイルを編集します

- 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、<authentication> エlement 内に次のテキストを追加します。

```
<security-domain name="EncryptDBPassword">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_IDP_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_EDC_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
```

- 2) ファイル内で、データベースに固有の値を指定します。

注：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。

## 9.7.2. 手動でインストールした JBoss 用の Oracle の設定

### Oracle 12c データベースドライバのインストール

[aem-forms root]/lib/db/oracle ディレクトリの JDK 1.8 ドライバーファイル ojdbc6.jar を、[appserver root]/modules/system/layers/base/com/oracle/main ディレクトリにコピーします。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 12c 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。

### lc\_turnkey.xml ファイルを編集します

- 1) データベースプロファイルを [DVD root]\third\_party\jboss.zip\ [JBoss\_root]\standalone\configuration から [AppServer\_root]\standalone\configuration ディレクトリにコピーします。
- 2) [appserver root]/standalone/configuration/lc\_turnkey.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>
```

- 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。

注：プラグ可能なデータベースを使用している場合は、<port> の後にあるコロン (:) をスラッシュ (/) で置き換えてください。

- 4) (Oracle RAC 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：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：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。

注：プラグ可能なデータベースを使用している場合は、<port> の後にあるコロン (:) をスラッシュ (/) で置き換えてください。

6) 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>30</max-pool-size>
```

注：Forms 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。

## データソースとしての Oracle の設定

プラグ可能な Oracle データベースを使用する場合は、「プラグ可能な Oracle データベースの作成と設定」を参照してください。

Oracle データベースと共に LiveCycle を実行する場合は、3 つのデータソース (IDP\_DS、EDC\_DS、DefaultDS) を作成する必要があります。

- 1) データベースプロファイルを [DVD root]\third\_party\jboss.zip\ [JBoss\_root]\standalone\configuration から [AppServer\_root]\standalone\configuration ディレクトリにコピーします。
- 2) [JBoss\_root]/domain/configuration/domain\_oracle.xml ファイルを編集用を開きます。
- 3) <datasources> タグを探し、すべての子ノードを削除します。
- 4) IDP\_DS データソースを作成するには、<datasources> ノードに次の行を追加します。

```
<datasource jta="true" jndi-name="java:/IDP_DS" pool-name="IDP_DS" enabled="true"
use-java-context="true">
<connection-url>jdbc:oracle:thin:@localhost:1521:ORACLE_SID</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<driver>oracle</driver>
```

```
<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
    <pool>
        <min-pool-size>1</min-pool-size>
        <max-pool-size>30</max-pool-size>
    </pool>
    <security>
```



```

        <user-name>DB_NAME</user-name>
        <password>DB_PASSWORD</password>
    </security>

<validation>
<exception-sorter class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"/>
</validation>

        <timeout>
            <blocking-timeout-millis>20000</blocking-timeout-millis>
            <idle-timeout-minutes>2</idle-timeout-minutes>
        </timeout>
        <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
    </statement>
</datasource>

```

5) EDC\_DS データソースを作成するには、<datasources> ノードの IDP\_DS の後に次の行を追加します。

```

    <datasource jndi-name="java:/EDC_DS" pool-name="EDC_DS" enabled="true" use-java-
context="true">
<connection-url>jdbc:oracle:thin:@localhost:1521:ORACLE_SID</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<driver>oracle</driver>

<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
        <pool>
            <min-pool-size>1</min-pool-size>
            <max-pool-size>30</max-pool-size>
        </pool>
        <security>
            <user-name>DB_NAME</user-name>
            <password>DB_PASSWORD</password>
        </security>

<validation>
<exception-sorter class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"/>
</validation>

        <timeout>
            <blocking-timeout-millis>20000</blocking-timeout-millis>
            <idle-timeout-minutes>2</idle-timeout-minutes>
        </timeout>
        <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
    </statement>
</datasource>

```

6) DefaultDS データソースを作成するには、<datasources> ノードの EDC\_DS の後に次の行を追加します。

```
<datasource jndi-name="java:/DefaultDS" pool-name="DefaultDS" enabled="true"
use-java-context="true">
<connection-url>jdbc:oracle:thin:@localhost:1521:ORACLE_SID</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<driver>oracle</driver>

<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
    <pool>
        <min-pool-size>1</min-pool-size>
        <max-pool-size>30</max-pool-size>
    </pool>
    <security>
        <user-name>DB_NAME</user-name>
        <password>DB_PASSWORD</password>
    </security>
<validation>
<exception-sorter class-
name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter"/>
</validation>

    <timeout>
        <blocking-timeout-millis>20000</blocking-timeout-millis>
        <idle-timeout-minutes>2</idle-timeout-minutes>
    </timeout>
    <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
    </statement>
</datasource>
```

7) データソース IDP\_DS、EDC\_DS、および DefaultDS のデータベース固有の値を設定します。

- a) **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。
- b) **1521** : Oracle でデフォルトのポートを使用していない場合は、適切なポート番号を指定します。
- c) **ORACLE\_SID** : ORACLE\_SID を Oracle System Identifier で置き換えます。
- d) **DB\_USER**、**DB\_PASSWORD** : 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秘密鍵証明書を入力します。
- e) 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。

注 : プラグ可能なデータベースを使用している場合は、<port> の後にあるコロン (:) をスラッシュ (/) で置き換えてください。

- 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>30</max-pool-size>
```

注：Forms 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。

- f) <datasource> タグの子として、次の行を追加します。

```
<drivers>
<driver name="oracle" module="com.oracle">
<xa-datasource-class>oracle.jdbc.xa.client.OracleXADataSource</xa-datasource-class>
</driver>
</drivers>
```

- 8) ファイルを保存して閉じます。

## lc\_turnkey.xml ファイルを編集します

- 1) [appserver root]/standalone/configuration/lc\_turnkey.xml ファイルをテキストエディターで開き、<authentication> エレメント内に次のテキストを追加します。

```
<security-domain name="EncryptDBPassword">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_IDP_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
```

```

        </login-module>
    </authentication>
</security-domain>
<security-domain name="EncryptDBPassword_EDC_DS">
    <authentication>
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">
            <module-option name="userName" value="adobe"/>
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
        </login-module>
    </authentication>
</security-domain>

```

- 2) ファイル内で、データベースに固有の値を指定します。
- 3) ファイルを保存して閉じます。
- 4) JBoss を起動します。

### 9.7.3. 手動でインストールした JBoss 用の SQL Server の設定

#### 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、AEM Forms 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してください。)

#### lc\_turnkey.xml ファイルを編集します

- 1) データベースプロファイルを [DVD root]\third\_party\jboss.zip\ [JBoss\_root]\standalone\configuration から [AppServer\_root]\standalone\configuration 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、<datasource>エレメントを 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>

```

- 3) 次の値を、データベースに固有の値に置き換えます。
  - **localhost** : データベースをホストするコンピューターの名前、IP アドレスまたは完全修飾パス。デフォルトは localhost です。
  - **1433** : データベースへのアクセスに使用するポート。
  - **adobe** : AEM Forms 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>エレメント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>30</max-pool-size>
```

注：Forms 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。

## データソースとしての MSSQL の設定

MSSQL データベースと共に AEM Forms を実行する場合は、3 つのデータソース (IDP\_DS、EDC\_DS、DefaultDS) を作成する必要があります。

- 1) データベースプロファイルを [DVD root]\third\_party\jboss.zip\ [JBoss\_root]\standalone\configuration から [AppServer\_root]\standalone\configuration ディレクトリにコピーします。
- 2) [JBoss\_root]/domain/configuration/domain\_mssql.xml ファイルを開いて編集します。
- 3) <datasources> タグを探し、すべての子ノードを削除します。
- 4) IDP\_DS データソースを作成するには、<datasources> ノードに次の行を追加します。

```
<datasources>
    <datasource jta="true" jndi-name="java:/IDP_DS" pool-name="IDP_DS"
    enabled="true" use-java-context="true">

<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>

<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
    <driver>sqlserver</driver>

<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
    <pool>
        <min-pool-size>1</min-pool-size>
        <max-pool-size>30</max-pool-size>
    </pool>
```

```

        <security>
            <user-name>DB_USER</user-name>
            <password>DB_PASSWORD</password>
        </security>
        <timeout>
            <blocking-timeout-millis>20000</blocking-timeout-millis>
            <idle-timeout-minutes>2</idle-timeout-minutes>
        </timeout>
        <statement>

</prepared-statement-cache-size>20</prepared-statement-cache-size>
    </statement>
</datasource>

```

5) EDC\_DS データソースを作成するには、<datasources> ノードの IDP\_DS の後に次の行を追加します。

```

<datasource jndi-name="java:/EDC_DS" pool-name="EDC_DS" enabled="true" use-java-
context="true">

<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>

<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
    <driver>sqlserver</driver>

<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
    <pool>
        <min-pool-size>1</min-pool-size>
        <max-pool-size>30</max-pool-size>
    </pool>
    <security>
        <user-name>DB_USER</user-name>
        <password>DB_PASSWORD</password>
    </security>
    <timeout>
        <blocking-timeout-millis>20000</blocking-timeout-millis>
        <idle-timeout-minutes>2</idle-timeout-minutes>
    </timeout>
    <statement>

</prepared-statement-cache-size>20</prepared-statement-cache-size>
    </statement>
</datasource>

```

6) DefaultDS データソースを作成するには、<datasources> ノードの EDC\_DS の後に次の行を追加します。

```

<datasource jndi-name="java:/DefaultDS" pool-name="DefaultDS" enabled="true"
use-java-context="true">

```

```

<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>

<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
    <driver>sqlserver</driver>

<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
    <pool>
        <min-pool-size>1</min-pool-size>
        <max-pool-size>30</max-pool-size>
    </pool>
    <security>
        <user-name>DB_USER</user-name>
        <password>DB_PASSWORD</password>
    </security>
    <timeout>
        <blocking-timeout-millis>20000</blocking-timeout-millis>
        <idle-timeout-minutes>2</idle-timeout-minutes>
    </timeout>
    <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
    </statement>
</datasource>

```

7) データソース IDP\_DS、EDC\_DS、および DefaultDS のデータベース固有の値を設定します。

- a) **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。
- b) **1433** : MSSQL でデフォルトのポートを使用していない場合は、適切なポート番号を指定します。
- c) **DB\_USER、DB\_PASSWORD** : 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秘密鍵証明書を入力します。
- d) 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>30</max-pool-size>

```

注 : Forms 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。

**lc\_turnkey.xml ファイルを編集します**

- 1) [appserver root]/standalone/configuration/lc\_turnkey.xml ファイルをテキストエディターで開き、<authentication> エlement 内に次の行を追加します。

Replace :□

□

```
<security-domain name="EncryptDBPassword">□
    <authentication>□
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">□
            <module-option name="userName" value="adobe"/>□
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>□
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>□
        </login-module>□
    </authentication>□
</security-domain>□
<security-domain name="EncryptDBPassword_IDP_DS">□
    <authentication>□
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">□
            <module-option name="userName" value="adobe"/>□
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>□
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>□
        </login-module>□
    </authentication>□
</security-domain>□
<security-domain name="EncryptDBPassword_EDC_DS">□
    <authentication>□
        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule" flag="required">□
            <module-option name="userName" value="adobe"/>□
            <module-option name="password" value="-
3bfaa32dfe43f65b207a6df87216de44"/>□
            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>□
        </login-module>□
    </authentication>□
</security-domain>□
```

- 2) ファイル内で、データベースに固有の値を指定します。
- 3) ファイルを保存して閉じます。
- 4) JBoss を起動します。



## Windows での統合セキュリティの設定

- 1) 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 4.0 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。
- 2) JBoss for Adobe Experience Manager Forms 6.3 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。
- 3) 「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。
- 4) 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。

## 9.8. 次の手順

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

## 10. 付録 - その他の必要システム構成

PDF Generator、Central Migration Bridge サービス、AEM Forms IPv6 サポート、Connector for IBM File Net、Connector for Documentum、Connector for IBM Content Manager、Forms サービス、Output サービス、ConvertPDF サービス、その他のコンポーネントには、いくつかの追加設定が必要になります。これらの機能を構成する場合にのみ追加設定を行ってください。

### 10.1. Linux および UNIX ベースのプラットフォームに関するその他の要件

注：Linux および UNIX プラットフォームでは、JEE 上の AEM Forms 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。

#### 10.1.1. UTF-8 のインストールおよび設定

Linux および UNIX ベースのオペレーティングシステムに JEE 上の AEM Forms 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：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。

#### 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。

TF-8 oTF-8 o

```
/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：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：詳しくは、「[Installing additional locales for Solaris](#)」を参照してください。

### 10.1.2. Solaris

注：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 Forms Standard 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。

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。

### 10.1.3. 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。
- 最新バージョンの 32 ビットの `libcurl`、`libcrypto` および `libssl` ライブラリをインストールします。
- `/usr/lib/X11/fonts` and `/usr/share/fonts` ディレクトリが存在することを確認してください。このディレクトリが存在しない場合は、`ln` コマンドを使用して `/usr/share/X11/fonts` から `/usr/lib/X11/fonts` へのシンボリックリンク、さらに `/usr/share/fonts` から `/usr/share/X11/fonts` への別のシンボリックリンクを作成します。また、`courier` フォントが使用可能であることを `/usr/lib/X11/fonts` で確認してください。
- すべてのフォント（Unicode および非 Unicode）が `/usr/share/fonts` or `/usr/share/X11/fonts` ディレクトリで使用できることを確認してください。
- OnRedhat Enterprise Linux 6.x で `courier` フォントは使用できません。`font-ibmtype1-1.0.3.zip` アーカイブをダウンロードしてください。`/usr/share/fonts` でアーカイブを抽出します。`/usr/share/X11/fonts` から `/usr/share/fonts` へのシンボリックリンクを作成します。`Html2PdfSvc/bin` と `/usr/share/fonts` のディレクトリから、`.lst` フォントのキャッシュをすべて削除します。

- **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、JEE 上の AEM Forms 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](#)を参照してください。）

JEE 上の AEM Forms 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。

### 10.1.4. Windows 以外のオペレーティングシステムでのファイル制限値の設定

Windows 以外のオペレーティングシステム環境で 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 nfile 65553
<app_group> hard nfile 65553
```

<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 65553 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 65553 以上に設定します (この変更は、デフォルト値が 65553 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: 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。

## 10.2. 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。JEE 上の AEM Forms 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。JEE 上の AEM Forms を、JEE 上の AEM Forms の一部であるサービスをインストールおよびデプロイした後に、LDAP データベースに接続するように設定します。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「[JEE 上の AEM Forms のインストールおよびデプロイ](#)」を参照してください。

## 10.3. PDF Generator に関するその他の要件

注：PDF Generator がデプロイされた Windows 2012 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 Direct IP などの代替プロトコルを使用してください。

### 10.3.1. 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：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。

### 10.3.2. Windows 以外のオペレーティングシステム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：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。

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

アプリケーションサーバーが使用する 64 ビット Java 8 JDK の他に、32 ビット Java 8 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：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 8s 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。

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。

#### 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\jdk1.8.0_74
```

#### Windows 以外のオペレーティングシステムでの `JAVA_HOME_32` 変数の設定

Linux の場合は、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/jdk1.8.0_74
export JAVA_HOME_32
```

Solaris の場合は、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/jdk1.8
export JAVA_HOME_32
```

### 10.3.4. 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。

JEE 上の AEM Forms 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、JEE 上の AEM Forms 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 JEE 上の AEM Forms 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 は、サードパーティのネイティブファイル変換アプリケーションを使用して、追加のファイルタイプを PDF ファイルに変換するように拡張することができます。サポート対象のアプリケーションとファイルフォーマットについての完全なリストは、「[サポートされているプラットフォームの組み合わせ](#)」ドキュメントを参照してください。

注：PDF Generator は、サポート対象のファイルフォーマットを PDF に変換するネイティブアプリケーションを使用します。特に説明がない限り、これらのアプリケーションとプラットフォーム（オペレーティングシステム）は、ドイツ語版、フランス語版、英語版および日本語版のみサポートされています。また、サポート対象の言語が基本プラットフォーム（オペレーティングシステム）にインストールされていることを確認してください。

注：JEE 上の AEM Forms では、上記すべてのソフトウェアについて、32 ビット版のみサポートします。

注：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：ネイティブファイルを変換するためのソフトウェアには、初期登録／アクティベーションダイアログがあります。サーバー上で設定されているすべての PDFG ユーザーアカウントに対するすべての初期登録／アクティベーションダイアログは却下します。

注：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：エンドユーザーは、サーバー上の PDF Generator によって使用されているソフトウェアアプリケーションを使用しないようにしてください。使用すると、PDF Generator の変換が干渉され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）



### 10.3.5. PDF Generator用のAcrobatのインストール

JEE上のAEM Formsインストーラーを実行する前に、Acrobat DC 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：JEE上のAEM Forms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、JEE上のAEM Forms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、[aem-forms 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、「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。

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

注：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。

### 10.3.7. 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 7.7.9以降（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。



### 10.3.8. 環境変数の設定

Photoshop、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。

- Notepad\_PATH
- OpenOffice\_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。

### 10.3.9. リモートマシン上でのPDF Generatorの設定

クラスターの場合、JEE上のAEM Forms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 DC Proをインストールします。
- 3) JEE上のAEM Formsがインストールされているマシンから、pdfg\_configとpluginsフォルダーを、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>
```

### 10.3.10. 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：PDF Generator が正しく機能するためには、JEE 上の AEM Forms が Windows のサービスとして実行され、そのサービスがローカルシステムアカウントの下で実行されている必要があります。

### 10.3.11. 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：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、アプリケーションサーバーのスタートアップパラメーターに追加する必要があります。具体的な手順については、アプリケーションサーバーのマニュアルを参照してください。

### 10.3.12. 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

PDF Generator では、一度に 1 つの OpenOffice、Microsoft Word または PowerPoint ドキュメントのみをデフォルトで変換できます。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：(Microsoft Office による) マルチスレッドファイル変換は Microsoft Word 2007、2010、2013、2016、および PowerPoint 2007、2010、2013、2016 のみでサポートされています。

注：Microsoft Excel、Publisher、Project ファイルは同時には変換されません。変換中、EXCEL.exe、PUBLISHER.exe、および PROJECT.exe はタスクマネージャーで監視されます。

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、JEE 上の AEM Forms 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、「Windows インストールの設定」を参照してください。

JEE 上の AEM Forms サーバーの設定後、JEE 上の AEM Forms ユーザーアカウントを管理コンソールに追加します。使用しているアプリケーションサーバー版の『JEE 上の AEM Forms 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、「[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。サードパーティー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。

注：サーバー上で設定されているすべてのPDFGユーザーアカウントに対して、Adobe Acrobat Distillerを少なくとも1度必ず起動してください。

## Windows Server 2012で報告されたエラーの無効化（オプションですが推奨します）

PDF Generator on Windows Server 2012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、<https://technet.microsoft.com/en-us/library/gg232692%28v=ws.10%29.aspx>を参照してください。

## Windows以外のオペレーティングシステムでOpenOfficeに必要な追加設定

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

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

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

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

- 2) すべてのJEE上のAEM Formsユーザーに、JEE上のAEM Formsサーバーへの接続を許可します。例えば、user1 というローカルユーザーにJEE上のAEM Forms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。

### 10.3.13. 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。Windows 以外のオペレーティングシステム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 2012	管理者権限を持つユーザー、JEE 上の AEM Forms 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。
Windows 以外のオペレーティングシステム	sudo 権限を持つユーザー JEE 上の AEM Forms 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。

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。[「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。

### 10.3.14. 「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、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。

### 10.3.15. 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。

### 10.3.16. 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
```

### 10.3.17. 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。

### 10.3.18. マルチスレッドファイル変換のユーザーアカウントの設定

PDF Generator では、一度に 1 つの OpenOffice、Microsoft Word または PowerPoint ドキュメントのみをデフォルトで変換できます。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)。

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

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

#### ユーザーアカウントの追加

- 1) 管理コンソールで、サービス / **PDF Generator** / ユーザーアカウントをクリックします。
- 2) 「追加」をクリックし、JEE 上の AEM Forms サーバー上での管理者権限を持つユーザーのユーザー名とパスワードを入力します。OpenOffice のユーザーを設定する場合は、最初に表示される OpenOffice のアクティベート用のダイアログを閉じます。

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

- 3) JEE 上の AEM Forms サーバーを再起動します。

### 10.3.19. 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。

## 10.4. Connector for Documentum に関するその他の要件

JEE 上の AEM Forms が Documentum に接続している場合、JEE 上の AEM Forms をホストしているマシンに Document Foundation Classes をインストールする必要があります。



## 10.5. Connector for IBM Content Managerに関するその他の要件

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『JEE 上の AEM Forms 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。

### 10.5.1. 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。

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

- 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\db2cmv8\`、Windows 以外のオペレーティングシステムでは `/opt/IBM/db2cmv8`）に変更します。
  - `cmbenv81.bat`（Windows）または `cmbenv81.sh`（Windows 以外のオペレーティングシステム）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\db2cmv8`
    - （Linux） `/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：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。

### 10.5.3. 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\db2cmv8\`、Windows 以外のオペレーティングシステムでは `/opt/IBM/db2cmv8`）に変更します。
- 2) `cmbenv81.bat`（Windows）または `cmbenv81.sh`（Windows 以外のオペレーティングシステム）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\db2cmv8`
  - （Linux） `/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。



## 10.6. 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、Connector for IBM®FileNetをインストールする場合のみ必要です。

### 10.6.1. IBM FileNet 5.0

JEE 上の AEM Forms 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:\Program Files\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。

### 10.6.2. IBM FileNet 5.2

JEE 上の AEM Forms を IBM FileNet 5.2 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 5.2 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 5.2 Process Engine の場合は、デフォルトで C:\Program Files\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。

## 10.7. 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、JEE 上の AEM Forms 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：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。

### 10.7.1. 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 JEE 上の AEM Forms 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。

### 10.7.2. 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 を JEE 上の AEM Forms と同じサーバーにインストールしておく必要がありますが、JEE 上の AEM Forms のインストールは前提条件ではありません（すなわち、JEE 上の AEM Forms の前または後にインストールすることができます）。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。

Windows 以外のオペレーティングシステムでは、Central Pro デモン `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：** Central Migration Bridge サービスを呼び出す JEE 上の AEM Forms 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。

### 10.7.3. JEE 上の AEM Forms の自動インストール

高速モードを使用して、JEE 上の AEM Forms 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：** 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。

### 10.7.4. JEE 上の Forms の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 JEE 上の AEM Forms 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、管理コンソールに移動して、Central Migration Bridge Service 用の [Central Install Dir] の設定を更新してください。

JEE 上の AEM Forms 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、JEE 上の AEM Forms が適切なディレクトリを参照するように指定します。

- 1) 管理コンソールにログインします。
- 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：** この設定は、Workbench でも可能ですプロセスの作成について詳しくは、Workbench ドキュメントの「プロセスの作成と管理」を参照してください。

## 10.8. JEE 上の AEM Forms の IPv6 サポート

JEE 上の AEM Forms には、IPv6 サポートが含まれています。JEE 上の AEM Forms 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。JEE 上の AEM Forms 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：IPv6 環境で CIFS を有効にする場合は、Configuration Manager を使用して JEE 上の AEM Forms 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。

### 10.8.1. 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。

### 10.8.2. 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。

- JEE 上の AEM Forms をインストールした後に、JEE 上の AEM Forms 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、[aem-forms 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。
- 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、AEM\_DS、IDP\_DS 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していません。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：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。

### 10.8.3. JBoss 用の IPv6 の設定

- 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。
- lc\_turnkey.xml およびデータベース固有のデータソース設定ファイルを、JEE 上の AEM Forms データベースに接続するように変更します。
- lc\_turnkey.xml ファイルを、JEE 上の AEM Forms データベースに接続するように変更します。
- 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\standalone.conf.bat
  - (他のプラットフォーム上の JBoss) [appserver root]\bin\standalone.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。
- [aem-forms root]\configurationManager\bin\IPv6\ ConfigurationManager\_IPv6.bat または Configuration Manager\_IPv6.sh スクリプトを呼び出して、Configuration Manager を起動します。
- Configuration Manager で、EAR ファイルを設定するための手順を選択し、JEE 上の AEM Forms モジュールをブートストラップおよびデプロイします。
- 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]\standalone\deployments ディレクトリにコピーします。
- コマンドラインから JBoss を起動します。
- 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 JEE 上の AEM Forms モジュールをデプロイします。