

# HP ALM ベストプラクティス・シリーズ

ALM 実践者向け

アップグレードのベスト・プラクティス



ドキュメントリリース日: 2016 年 5 月 (英語版)

## ご注意

### 保証

HP製品、またはサービスの保証は、当該製品、およびサービスに付随する明示的な保証文によってのみ規定されるものとして、ここでの記載で追加保証を意図するものは一切ありません。ここに含まれる技術的、編集上の誤り、または欠如について、HPIはいかなる責任も負いません。

ここに記載する情報は、予告なしに変更されることがあります。

### 権利の制限

機密性のあるコンピューターソフトウェアです。これらを所有、使用、または複製するには、HPからの有効な使用許諾が必要です。商用コンピューターソフトウェア、コンピューターソフトウェアに関する文書類、および商用アイテムの技術データは、FAR 12.211および12.212の規定に従い、ベンダーの標準商用ライセンスに基づいて米国政府に使用許諾が付与されます。

### 著作権について

© Copyright 2002 - 2016 Hewlett Packard Enterprise Development LP

### 商標について

Oracle®は、Oracle Corporationおよびその関連会社の登録商標です。

Microsoft®、Windows®およびWindows® XPは、Microsoft Corporationの米国における登録商標です。

## ドキュメントの更新情報

このマニュアルの表紙には、次の識別情報が記載されています。

- ソフトウェアバージョンの番号は、ソフトウェアのバージョンを示します。
- ドキュメントリリース日は、ドキュメントが更新されるたびに変更されます。
- ソフトウェアリリース日は、このバージョンのソフトウェアのリリース期日を表します。

更新状況、およびご使用のドキュメントが最新版かどうかは、次のサイトで確認できます。

<https://softwaresupport.hpe.com/group/softwaresupport/search-result?doctype=manuals?keyword=>

このサイトを利用するには、HP Passportへの登録とサインインが必要です。HP Passport IDの登録は、次のWebサイトから行なうことができます。

<https://softwaresupport.hpe.com/>

または、HP Passport のログインページの **[New users - please register]** リンクをクリックします。

適切な製品サポートサービスをお申し込みいただいたお客様は、更新版または最新版をご入手いただけます。詳細は、HPの営業担当にお問い合わせください。

## サポート

HPソフトウェアサポートWebサイトを参照してください。

<https://softwaresupport.hpe.com/>

このサイトでは、HPのお客様窓口のほか、HPソフトウェアが提供する製品、サービス、およびサポートに関する詳細情報をご覧いただけます。

HPソフトウェアオンラインではセルフソルブ機能を提供しています。お客様のビジネスを管理するのに必要な対話型の技術サポートツールに、素早く効率的にアクセスできます。HPソフトウェアサポートのWebサイトでは、次のようなことができます。

- 関心のあるナレッジドキュメントの検索
- サポートケースの登録とエンハンスメント要求のトラッキング
- ソフトウェアパッチのダウンロード
- サポート契約の管理
- HPサポート窓口の検索
- 利用可能なサービスに関する情報の閲覧
- 他のソフトウェアカスタマーとの意見交換
- ソフトウェアトレーニングの検索と登録

一部のサポートを除き、サポートのご利用には、HP Passportユーザーとしてご登録の上、サインインしていただく必要があります。また、多くのサポートのご利用には、サポート契約が必要です。アクセスレベルの詳細については、次のWebサイトをご覧ください。

<https://softwaresupport.hpe.com/web/softwaresupport/access-levels>

HP Passport IDを登録するには、次のWebサイトにアクセスしてください。

<https://softwaresupport.hpe.com/>

# 目次

<b>第 1 章</b>	<b>HP ALM のアップグレードについて</b>	<b>6</b>
	はじめに	6
	対象となる読者	6
<b>第 2 章</b>	<b>定義</b>	<b>7</b>
	ALM アップグレードの必要性	7
	アップグレードの範囲と戦略の定義	7
	インフラストラクチャのアップグレード	7
	ALM のアップグレード	9
	サイトのアップグレード方法	10
	サイト管理の構造	11
	プロジェクトのアップグレード方法	12
	プロジェクトの設定とカスタマイズ	12
<b>第 3 章</b>	<b>テスト</b>	<b>13</b>
	テスト環境とは	13
	テスト環境を設定する理由	13
	テストの対象と方法	14
	テスト・プロジェクトの選定	14
	テスト戦略	14
	テスト環境のセットアップ	16
	インストール・モード	16
	インストール後のチェックリスト	16
	テスト結果の分析	17
<b>第 4 章</b>	<b>運用環境のアップグレード計画</b>	<b>18</b>
	インフラストラクチャ・チェックリストの作成	18
	制限リストの作成	19
	プロジェクト・アップグレードのスケジュール作成	19
	プロジェクトの担当者	19
	ロールバック手順の作成	20
	アップグレード後の動作確認テストの作成	20
	アップグレード後のチェックリストのレビュー	21
	新バージョンのトレーニング資料とリリース・ノートの作成	21
	アップグレード後のサポート	21

# 第 1 章 HP ALM のアップグレードについて

## はじめに

HP Application Lifecycle Management (ALM) は、要件定義から展開まで、コア・アプリケーションのライフサイクル全体を管理する強力なツールです。アプリケーション・チームは ALM を使用することによって、最新のアプリケーションを予測可能、繰り返し可能、柔軟な適応が可能な方法で提供するために不可欠な可視性とコラボレーション環境を実現できます。ALM の各バージョンでは、アプリケーション管理プロセスを最も効率的な方法で行うための新機能が提供されています。

このような新機能を既存の ALM 資産で最大限に活用するためには、製品で提示されているアップグレード手順に従う必要があります。ALM のアップグレード手順の詳細については、『HP Application Lifecycle Management インストール・ガイド』を参照してください。

サイトのアップグレードは、次のフェーズに従って計画することをお勧めします。



1. **準備と計画** — 定義、テスト、計画という 3 つのサブステップで構成されます。この 3 つの段階のベスト・プラクティスは、次のとおりです。
  - アップグレードが失敗する原因になる潜在的な問題点を早い段階で見つけます。
  - 運用環境のアップグレードに必要なリソースや時間を適切な方法で計画および評価することによって、サーバのダウンタイムを最小限まで短縮します。

このフェーズの結果として、ALM アップグレード計画が作成されます。

2. **アップグレードの実施** — ALM アップグレード計画の作成と承認が完了したら、アップグレードを開始します。対象の ALM プロジェクトを新バージョンにアップグレードした後、プロジェクトを検証し、エンド・ユーザがアクセス可能な状態かどうかを確認します。検証が完了したら、エンド・ユーザは作業を継続できます。その間、プロジェクトのカスタマイズと設定を行います。
3. **アップグレード後のプロジェクトの設定とカスタマイズ** — このフェーズでは、新バージョンで提供されている新機能のサポートやワークフローの変更に対応できるように、ALM プロジェクトをカスタマイズします。HP ソフトウェアおよびソリューションのプロフェッショナル・サービスが、ALM に関する豊富な知識や経験でお客様をサポートします。

本書では、準備および計画フェーズに含まれる 3 つのサブステップに関する手法と、HP が推奨するガイドラインを紹介します。

## 対象となる読者

本書は、現在の ALM 環境から新バージョンへのアップグレードを担当する管理者を対象としています。

## 第 2 章 定義

### ALM アップグレードの必要性

ALM の各バージョンでは、アプリケーション管理プロセスを最も効率的かつ経済的な方法で行うための新機能が提供されています。

ALM の新バージョンがリリースされた場合、まず最初に、アップグレードの必要性の有無を検討することが重要です。この評価プロセスでは、次の項目を検討してください。

- 運用環境で現在使用している ALM バージョンは引き続きサポートされるか。サポートされる場合、サポート期間はどの程度か。
- ALM の新バージョンでは、ユーザ環境のプロセス改善につながる機能や修正が提供されているか。
- 都合のよいタイミング(メジャー・リリース、リソースの追加または削除などの前)で ALM をアップグレードできるか。

ALM アップグレードの必要性の評価については、HP プロフェッショナル・サービスまたはカスタマー・サポートをご利用ください。

ALM の新バージョンが必要であると判断した場合、アップグレードの実施対象と戦略を定義します。

### アップグレードの範囲と戦略の定義

アップグレードの範囲と戦略の定義では、次の項目について検討します。

1. インフラストラクチャを変更する必要があるか。
2. どのような戦略に基づいて、インフラストラクチャを変更するか。
3. どのような戦略に基づいて、ALM サイトおよびプロジェクトをアップグレードするか。

上記の項目の検討結果に基づいて、次のようなアップグレード戦略を採用することができます。

1. 運用環境にある既存のサーバのアップグレードと、新規サーバまたは別のサーバを使用したアップグレード
  2. 段階的アップグレードと、一括アップグレード
  3. クリーン・インストールによってサイト管理の構造を変更する方法と、既存のサイト管理の構造を使用する方法
- 以下では、上記の項目の説明と、その検討結果に基づいて採用可能なアップグレード戦略を紹介します。

### インフラストラクチャのアップグレード

ALM のアップグレードは、ユーザ環境を変更およびアップグレードするチャンスです。このような変更は、次に示す要因によって発生します。

- ALM サポート対象マトリックスの変更  
最新の ALM システム構成 ([http://www.hp.com/go/TDQC\\_SysReq](http://www.hp.com/go/TDQC_SysReq)) をレビューし、ALM の新バージョンで提供される Readme ファイルを確認してください。
- ソフトウェアの新規またはアップグレード・バージョンのリリース。
- 別のソフトウェアまたはハードウェアへの切り替え。

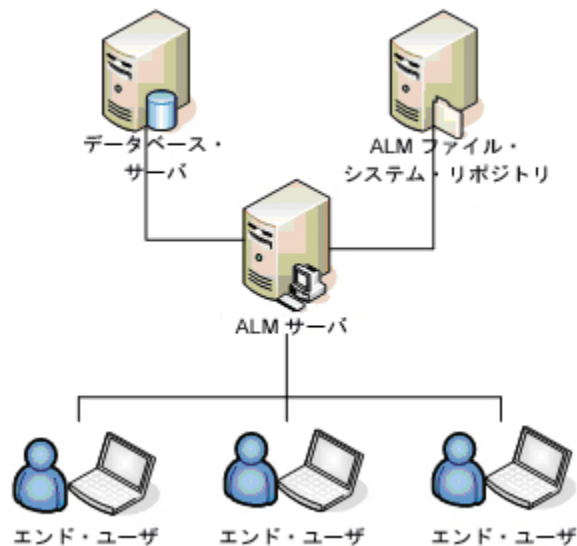
ALM 環境は、ハードウェアとソフトウェアの 2 つの部分に分かれています。ハードウェアには、物理サーバが含まれます。これに対してソフトウェアには、オペレーティング・システム、Web サーバ、アプリケーション・サーバ、データベースが含まれます。

アップグレードの計画では、ソフトウェアで発生する可能性のある変更点をすべて検討する必要があります。この評価作業では、次の点を考慮してください。

- ALM の成長速度。ユーザ数、データベースおよびファイル・システム・リポジトリのサイズなど。
- 使用方法の変化。たとえば、可用性、セキュリティ、パフォーマンス、統合ツールの使用の変化など。
- 他の HP 製品 (QuickTest Professional など) や ALM と統合されている HP 以外の製品の使用。このような製品がインフラストラクチャ要件に与える影響。
- ポリシーの変更。ALM での認証方法の変更など。

HP カスタマー・サポート技術情報には、ALM の実装に必要な環境を特定する上で役に立つ多くの記事が用意されています。

ソフトウェアの変更だけでなく、アーキテクチャの変更も考慮する必要があります。ALM アーキテクチャでは、ALM インストール・サーバ、データベース、ファイル・システム・リポジトリが必要です。このようなコンポーネントは、それぞれ別のサーバ上にセットアップすることができます。サイトに多数のプロジェクトが存在する場合や、プロジェクトに大量のデータが含まれる場合は、各コンポーネントを別々のサーバにセットアップすることをお勧めします。次に例を示します。



このような方法で構築したアーキテクチャには、次のようなメリットがあります。

- 各プロジェクトのスキーマやリポジトリに問題が発生するリスクを軽減できる
- 各コンポーネントで使用できるストレージを増大できる
- プロジェクトやサイトを柔軟に拡張できる

ALM で使用する物理環境は、コストとパフォーマンスを考慮して設計を行う必要があります。ただし、ほとんどの場合、ALM の実装には、現在のユーザ環境と同じ構成が使用されます。ユーザ・ニーズに合った環境設定を特定するための参考として、最新の ALM 構成 ([http://www.hp.com/jp/TDQC\\_SysReq](http://www.hp.com/jp/TDQC_SysReq)) を参照してください。

ソフトウェア環境やハードウェア環境を変更する場合は、次の KPI (主要業績評価指標) を検討します。

- 1 か月あたりに実施する一般的なメンテナンス・サービスの回数
- 1 か月あたりに実施する一般的なメンテナンス・サービスの中で、運用環境に影響しないサービスの回数
- 1 か月あたりに実施する ALM インフラストラクチャ・サービスの回数



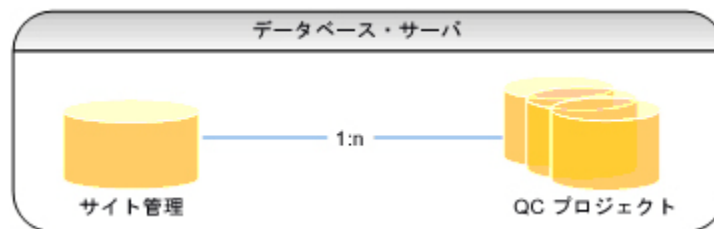
- 生産性に影響を与える ALM インフラストラクチャ・サービスの回数
- 1 年間に適用する ALM パッチの数
- 生産性に影響を与える ALM パッチの数
- 生産性に影響を与える警告の数
- 予想される ALM 使用率の増大により、データベースやリポジトリの拡張が必要になるもの

## ALM のアップグレード

ALM では、次のデータベース・ユーザ・スキーマを使用できます。

- サイト管理データベース・ユーザ・スキーマ  
すべてのサイト管理テーブル(SITE\_USERS, PROJECTS など)を管理します。このスキーマは、各 ALM サイトに 1 つのみ存在し、ALM のインストール時に作成されます。
- プロジェクト・データベースのユーザ・スキーマ  
特定のプロジェクトに関連するデータをすべて管理します。プロジェクトには、それぞれ専用のスキーマがあります。

各 ALM インスタンスは、1 つのサイト管理スキーマと、サイトで定義されている任意の数のプロジェクト・スキーマを使用できます。クラスタ環境では、すべてのインスタンスが同じサイト管理スキーマを共有します。



アップグレードでは、上記のスキーマをすべてアップグレードする必要があります。すべてのスキーマをアップグレードするには、サイトをアップグレードしてから(既存のサイト管理スキーマで新しいバージョンをインストールする際に自動的にアップグレードされます)、プロジェクト・スキーマを個別にアップグレードします。

## サイトのアップグレード方法

物理的なアップグレードには、次の方法があります。

1. 既存サーバをアップグレードする方法
2. 新規サーバでアップグレードする方法

次のような場合は、通常、既存サーバでアップグレードを行います。

1. サーバとして使用できる新規ハードウェアがない場合
2. プロジェクトの数が少ない場合
3. プロジェクトの規模が小さく、リスクが低い場合
4. テスト環境のアップグレードを行う場合

別のサーバを使ってアップグレードを行う場合は、ALM のインストールで使用する新規サーバが必要になります。ただし、ファイル・システム・リポジトリやデータベース用の新規サーバは不要です。データベースとファイル・システム・リポジトリが異なるサーバ上にある場合、データベースとファイル・システム・リポジトリのいずれにも、アップグレードに伴う変更は必要ありません。

物理アップグレードの方法を選択する際には、次の点を検討してください。

アップグレードの方法	運用サーバのアップグレード	新規サーバでのアップグレード
ハードウェア	追加ハードウェアは不要	新規サーバが必要
アップグレードの手法 (12 ページを参照)	一括アップグレード	一括アップグレードまたは段階的アップグレード
ダウンタイム	既存の ALM バージョンのアンインストールから、新規バージョンのインストールとプロジェクトのアップグレードが完了するまでの間、プロジェクトは使用不能になります。	既存バージョンでプロジェクトがアクティブである場合、ダウンタイムが発生するのは、プロジェクトのアップグレード中のみです。
サイト管理	新規のサイト管理スキーマを使ってインストールすると、サイトの構造と設定はすべて使用できなくなります。したがって、既存のサイト管理スキーマを使用することをお勧めします。	インストールでは、新規と既存のどちらのサイト管理を使用することも可能です。  注: 新規のサイト管理スキーマを使用すると、ユーザ情報(パスワードやユーザの詳細情報)はすべて削除されます。
バックアップ	新規インストールを行う前には、すべてのプロジェクト(サイト管理スキーマを含む)をバックアップすることをお勧めします。	インストールで新規サイト管理を使用する場合には、プロジェクトが 1 つずつアップグレードされるので、その後でバックアップを作成してください。既存のサイト管理スキーマを使用する場合は、インストールの前にバックアップを作成してください。

アップグレードの方法	運用サーバのアップグレード	新規サーバでのアップグレード
ユーザ・アクセス	最初に、サイト全体をシャットダウンする必要があります。プロジェクトのアップグレードが完了したら、プロジェクトを使用可能な状態に戻すことができます。	段階的アップグレードでは、既存バージョンと最新バージョンを混在できるので、高い柔軟性を実現できます。
バージョン管理プロジェクト	既存の ALM バージョンをアンインストールする前に、バージョン管理が有効になっているすべてのプロジェクトで、エンティティをすべてチェックインしておく必要があります。	新しいサーバにプロジェクトを移行する前に、既存の ALM バージョンでエンティティをチェックインする必要があります。

## サイト管理の構造

ALM のインストールには、クリーン・インストールを行う方法と、運用環境のサイト管理スキーマのコピーを使用する方法があります。運用環境のサイト管理スキーマを使用する方法と新規サイトを使用する方法のどちらを採用するかは、次の点を検討して決定してください。

- サイト構造の再編成 — サイト構造(ドメイン内のプロジェクト)の再編成が必要な場合、新規サイト管理を使って運用環境をインストールします。プロジェクトの数が多い場合、次の理由からサイトの再編成はお勧めしません。
  - 再編成が、非常に面倒な作業になることがあります。
  - 再編成に時間がかかることがあります(完全なアップグレードを計画することにより、再編成にかかる時間を短縮できます。アップグレード計画の作成方法については、19 ページを参照してください。)
  - サイト構造の再編成には、サイト・ユーザの再編成が必要になることがあります。
  - すべてのプロジェクトのすべてのユーザに、変更内容の周知を徹底する必要があります。
- ユーザ管理の再編成 — ALM では、サイト管理のユーザ・リストに多数のユーザ・リストが存在する可能性があります。ユーザが存在しないクリーンなサイト管理スキーマを使って ALM をインストールすることによって、すべてのユーザが必要かどうかを確認できます。運用環境のサイト管理スキーマを使用すると、すべての運用ユーザ・リストがテスト環境に追加されます。注: プロジェクトをサイト管理に復元すると、リストのユーザがすべて追加されます。
- サイト構成のメンテナンス — 新規サイト管理でインストールを行うと、[サイト設定]タブで行った設定内容をすべてコピーする必要があります。
- 管理にかかる時間 — 既存のサイト管理スキーマを使用するには、サイト内にある設定、プロジェクト、ユーザのメンテナンスが必要になります。この作業には、時間と手間がかかることがあります。これに対して、新規サイト管理スキーマを使ってインストールを行う場合は、すべてのデータをリセットする必要があります。サイト管理スキーマを選択する際には、手間の大きさと利便性を考慮してください。

既存のサイト管理スキーマを使用するには、まず最初に、運用環境のサイト管理スキーマをバックアップします。ALM スキーマのバックアップに関する詳細については、技術情報の記事 KM205839 および KM169526 を参照してください。バックアップしたサイト管理スキーマを復元する場合は、インストール時に使用したスキーマ名を使用してください。既存のサイト管理スキーマを使ったインストール方法については、『HP Application Lifecycle Management インストールおよびアップグレード・ガイド』を参照してください。

注: 新規のサイト管理スキーマを使用する場合、この新規スキーマにユーザ・データは格納されていません。したがって、ソースのサイト管理スキーマからターゲットのサイト管理スキーマに Users テーブルを移動し、SEQUENCES テーブルの USER\_SEQ エントリを更新する必要があります。インデックス番号の値は、Users テーブルの USER\_ID の最大値に設定してください。

## プロジェクトのアップグレード方法

プロジェクトのアップグレード方法には、一括アップグレードと段階的アップグレードがあります。

アップグレードの方法	検討事項
一括アップグレード	<ul style="list-style-type: none"><li>➤ 一括アップグレードの実行中、エンド・ユーザは ALM Platform サーバを含む運用環境を使用できなくなります。</li><li>➤ テスト結果(13 ページのテストの項を参照)から推定されるシステムのダウンタイムをもとに、この方法がユーザの運用環境に適しているかどうかを判断してください。</li><li>➤ 一括アップグレードが適しているのは、プロジェクトの数が多くなく、システム・ダウンタイムが発生しても許容される環境です。</li></ul>
段階的アップグレード	<ul style="list-style-type: none"><li>➤ 段階的アップグレードの場合、ユーザはアップグレード中、アップグレード対象のプロジェクト・グループにアクセスできなくなります。</li><li>➤ このプロジェクト・グループのアップグレードが完了すると、ALM 運用環境内のすべてのエンド・ユーザがアクセスできるようになります。</li><li>➤ この方法は、アップグレード作業を小さな単位に分割することにより、エンド・ユーザがプロジェクトにアクセスできなくなる時間を最小限に短縮できるので、アップグレード・プロセスの柔軟な管理が可能になります。</li><li>➤ 既存の運用サーバでアップグレードを行う場合には、段階的アップグレードは使用できません。</li><li>➤ 新規サーバと既存サーバが両方存在する間は、両方のサーバのメンテナンスが必要になります。たとえば、新規ユーザは両方に追加する必要があります(パスワードの変更、サイト管理パラメータなども同様です)。</li></ul>

## プロジェクトの設定とカスタマイズ

アップグレードが終わった時点では、プロジェクトは既存バージョンに基づいた設定のままであり、新規バージョンで提供されている新機能のサポートに必要な設定やカスタマイズは行われていない状態です。このフェーズでは、各プロジェクトをカスタマイズすることによって、新機能のサポートや、ワークフローの変更への対応を行います。

このフェーズには、現行の QA プロセスをよく理解し、ALM プロジェクトのカスタマイズ全体を担当する ALM プロジェクトの管理者が参加します。

## 第 3 章 テスト

アップグレード・プロセスは、機能、パフォーマンス、他の HP 製品 (HP LoadRunner, HP Performance Center, HP Business Process Testing など) やサードパーティ製品の互換性に影響を及ぼします。運用環境に ALM をインストールする前に、特定の設定を反映したテスト環境を作成し、そこで ALM をテストします。

### テスト環境とは

テスト環境とは、運用環境とは分離された環境であり、運用環境とまったく同じ構成が行われています。つまり、運用環境の設定やインストールされているアプリケーション (データベース・サーバ、ソフトウェア、運用プロジェクトなど) を疑似的に再現した環境です。テスト環境でアップグレードをテストすることにより、アップグレードの結果を明確に把握できるだけでなく、運用環境で発生する可能性のある問題を特定して回避することが可能になります。このテスト環境は、新バージョンの評価にも使用できます。

### テスト環境を設定する理由

- 運用環境に移行する前に、アップグレード手順をテストできる
- アップグレードに失敗しても、テスト環境は運用環境と分離されているので実害がない
- 運用環境で発生する可能性のある問題を早い段階で特定し、検出できる
- このプラットフォームに基づいて、アップグレードの計画や、運用環境のアップグレードに必要なリソースと時間に関する最終的な判断ができる
- 新しい機能を評価するプラットフォームとして活用できる
- 新しいバージョンのトレーニング資料の作成に活用できる (新バージョンのトレーニング作成に関する詳細は、22 ページの新バージョンのトレーニングに関する項を参照してください)。

# テストの対象と方法

## テスト・プロジェクトの選定

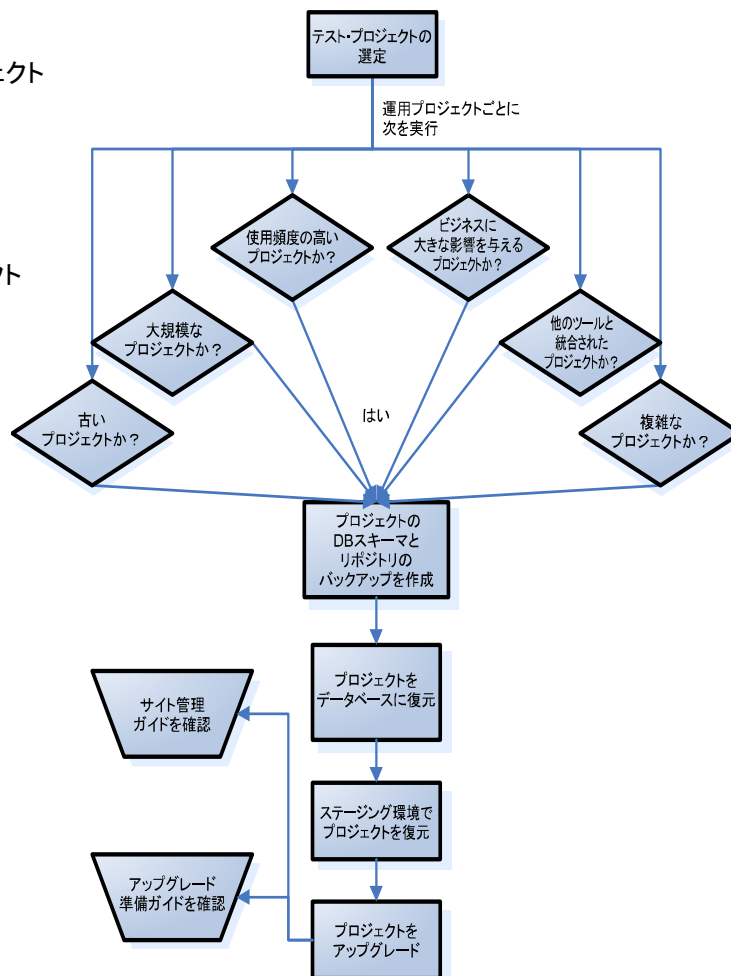
テスト環境で使用するプロジェクトは、次の条件を1つ以上満たしたプロジェクトが適しています。

- 使用頻度の高いプロジェクト
- 多数のユーザが使用する戦略的プロジェクト
- 大規模なプロジェクト(資産数が多い)
- 古いプロジェクト
- 複雑なプロジェクト
- ビジネスに大きな影響を与えるプロジェクト

テスト環境でテスト用に使用するプロジェクトを選択したら、プロジェクトのデータベース・ユーザ・スキーマをバックアップし、リポジトリをコピーします。ALM プロジェクトのバックアップ方法の詳細については、技術情報の記事 KM205839 および KM169526 を参照してください。

テスト環境の準備にかかる時間を短縮するために、データベース・スキーマやリポジトリの格納場所が記載された dbid.xml ファイルを事前に用意しておいてください。

注: テスト環境でアップグレード作業を行うことによって、運用環境でのアップグレード中に発生する現象を正確にシミュレーションすることができます。また、テスト環境は独立しているので、運用プロジェクトを実際にアップグレードする際の最適な方法を検討することもできます。



## テスト戦略

テスト・プロジェクトのテスト方法を決定するには、テストにかかる手間と時間を考慮してください。時間やリソースに厳しい制約がない場合は、すべてのプロジェクトを対象にすべてのタイプのテストを実施します。時間やリソースに制約がある場合は、プロジェクトごとに異なるテストを実施することができます。ただし、テスト範囲が広がるほど、アップグレード・プロセスや ALM の新バージョンに対する信頼性も高まります。

テスト環境では、次のようなタイプのテストを検討してください。

- パフォーマンス・テスト: 新バージョンでのパフォーマンスと比較することにより、運用環境でのパフォーマンスをテストします。注: 正確なテスト結果を得るためには、運用サーバと同じ設定(作業負荷など)を使用してテスト環境のシミュレーションを行う必要があります。
- データ整合性テスト: 既存のプロジェクト情報がすべて存在し、破損していないことを確認します。

- ワークフローの下位互換性/回帰テスト: 組織で定義されている手順とアクションをテストします。包括的な回帰テストでは、アクションを個別に評価し、期待されるデータが正しく取得されることを確認します。
- エンド・ツー・エンド (E2E) テスト: プロセス全体の状態とテスト結果をチェックします。回帰テストではアクションを個別にチェックしますが、E2E テストでは、1 つまたは複数のアクションを含むプロセス全体をチェックします。
- ユーザビリティ・テスト: 人為的な製品が想定通りに機能する能力を測定します。
- 新機能の検証テスト: 運用環境のデータとプロセスを使用することによって、新バージョンで提供される新機能がユーザ・ニーズを満たし、既存のプロセスに統合可能かどうかを判断します。

テストのタイプを選定したら、各プロジェクトで実施するテストを選択します。このマトリックスを作成する際には、次の点を考慮してください。

要素	検討内容
各プロジェクトで実施するテスト・タイプにかかるテスト時間	テストの実施回数に影響を与える可能性があります。所定の時間枠でテスト可能な回数を選択してください。
テスト・プロジェクトで定義されているプロセスを熟知したテスト担当者の参加	プロジェクト・プロセスを熟知したテスト担当者が参加できない場合、ワークフロー・テストまたはデータ整合性テストは実施できません。
プロジェクトに含まれる資産の数	プロジェクトに多数の資産が含まれている場合は、パフォーマンス・テストを検討してください。
旧バージョンのプロジェクトにパフォーマンスの問題が発生していたかどうか	パフォーマンス・テストの実施を検討してください。
プロジェクトはビジネスに大きな影響を与えるかどうか	大きな影響を与える場合は、次のテストの実施を検討します。 - テスト整合性テスト - ワークフロー・テスト - E2E テスト
運用サーバでプロジェクトがアクティブな状態かどうか	アクティブな状態である場合、次のテストの実施を検討します。 - E2E テスト - ユーザビリティ・テスト
プロジェクトには、新バージョンの機能によるメリットがあるか	プロジェクトで新機能を実装した場合の効果を評価します。

テスト環境でのテスト対象となるプロジェクトごとに、このようなテスト・マトリックスを作成するのがベスト・プラクティスです。テスト・マトリックスでは、各テスト・タイプのテスト・レベルをプロジェクトごとに定義します。例:

	プロジェクト A	プロジェクト B	プロジェクト C	プロジェクト D	プロジェクト E
パフォーマンス	完全	なし	なし	完全	基本
データ整合性	なし	完全	完全	完全	完全
ワークフロー	なし	基本	基本	完全	なし
E2E	部分	基本	完全	完全	部分
ユーザビリティ	なし	基本	基本	完全	部分
新機能	なし	なし	完全	完全	部分



# テスト環境のセットアップ

## インストール・モード

テスト環境をセットアップする前に、新バージョンのテストに使用するリソースが準備されていることを確認してください。テスト用のリソースやサーバを準備できない場合、アップグレード・プロセスを再作成するか、次の手順に進みアップグレード計画を作成します。

サイト管理スキーマのインストールでは、運用環境のサイト管理スキーマと新規スキーマのどちらを使用するかを選択します。次の事項を考慮してください。

- シミュレーションの整合性 — 整合性を最大限に高めるためには、既存のサイト管理スキーマまたは新規スキーマのどちらを使用するかに基づいて、テスト環境をインストールする必要があります。
- 完全なインストール・テスト — 運用環境のアップグレード時に選択したインストール方法をそのまま使用します。運用環境にあるサイト管理スキーマを使用して運用環境をアップグレードする場合、インストールに伴うリスクを軽減するために、テスト環境で事前にテストを行うことをお勧めします。
- テスト管理の簡易化 — 運用環境のサイト構造を使用する場合、テスト・プロジェクトを運用環境と同じツリー構造内の場所に復元する方法を選択する場合があります（ただし、必須ではありません）。この場合、ツリー内でテスト・プロジェクトを探して管理する作業が複雑になることがあります。
- ユーザ管理 — 運用環境のサイト管理スキーマをテスト環境で使用するには、運用環境のすべてのユーザ・リストをテスト環境でも含める必要があります。サイトのユーザをすべてリストに含めても、シミュレーションの完全性には影響しないことを確認してください。

**注:** プロジェクトをサイトに復元すると、プロジェクトに存在するがリストにないサイト・ユーザはすべてサイト・ユーザ・リストに自動的に追加されます。

- 管理にかかる時間 — 管理が面倒であるほど、管理にかかる時間は長くなります。運用サイト構造での作業が面倒である場合には、クリーン・インストールを検討してください。ただし、運用サイト構造を使用する場合でも、テスト専用ドメインを定義することができます。これにより、管理にかかる時間が短縮するだけでなく、シミュレーションの完全性も低下します。

**注:** 運用サイト管理スキーマのコピーを使用する場合、運用環境のドメインとプロジェクトがすべてリストに含まれますが、アクティブまたはアクセス可能な状態ではありません。さらに、管理とテスト上のニーズに応じて、単一のローカライズ・ドメインまたは運用プロジェクト・ツリーと同じ場所にプロジェクトを復元します。

## インストール後のチェックリスト

テスト環境のインストールが完了したら、チェックリストを作成し、ALM で必要な設定がすべて行われていることを確認します。このチェックリストの重要な項目を次に示します。

- テストのニーズに合ったライセンス
- ALM サイト管理でのデータベース・サーバの定義と設定
- インストール前にテスト対象プロジェクトを選択している場合は、選択したすべてのプロジェクトに関する dbid.xml ファイルを作成します。ALM プロジェクトを復元する方法の詳細については、『HP Application Lifecycle Management インストールおよびアップグレード・ガイド』でプロジェクトの復元に関する内容を参照してください。
- 統合ツールの設定とインストール
- サイト管理へのログイン接続の確認
- テスト関係者にサイト管理へのアクセス権限が割り当てられていることの確認



- ログの記録を行うレベルの設定(これにより、問題が発生した場合の追跡作業が簡単になります)
- 運用環境でメール・サーバが設定されている場合、テスト環境でも設定を行ってください。ただし、テスト中に運用環境のユーザにメールが自動送信されないようにしてください。これには、プロジェクト内のすべてのメール・アドレスをリセットする方法と、運用環境のメールにアクセスできないメール・サーバを別途設定する方法があります。メール・アドレスをリセットする場合は、DBA に問い合わせの上、プロジェクト・スキーマで次のクエリを実行します。

```
UPDATE USERS  
SET US_MAIL_ADDRESS = ''
```

## テスト結果の分析

アップグレード手順の定義とテスト結果がユーザ・ニーズを満たしているかどうかを判断するには、運用環境の ALM バージョンと新バージョンを機能とパフォーマンスの点で比較することにより、テスト環境の結果を検証します。HP やサードパーティのツールを ALM と統合している場合は、ツールの下位互換性も確認します。たとえば、スクリプトで OTA API 機能を使用する場合、アップグレード後にスクリプトを実行し、アップグレードでインストールされた新しい OTA API コンポーネントの互換性を確認します。

問題が発生した場合は、そのリスクの大きさに基づいて判断します。たとえば、小規模な機能変更が発生した場合、必要となるのがワークフローの調整時間であれば、リスクは低いと判断されます。リスクの大きさに基づき、アップグレード手順を続行できるかどうかを決定してください。

## 第 4 章 運用環境のアップグレード計画

作業を始める前に、綿密な計画を作成することが最も重要です。そのために、マスター・タスク・リストまたは作業計画書を作成します。これを元にタスクを追跡し、取りこぼしがないことを確認します。

作業計画書には、実施すべきすべてのアクション、スケジュール、関係者を記載します。たとえば、プロジェクト・スキーマのバックアップについては、DBA は重要な関係者として、実施にかかる時間の見積りを提示します。問題が発生しないように、問題点の洗い出しを事前に行い、回避策を検討します。たとえば、DBA が作業に参加できない場合は、別途日程を調整するか、DBA が作業計画書を実行する上での補佐役を指名します。

作業計画書では、決定したアップグレード戦略（一括アップグレードまたは段階的アップグレード）に必要な段階をすべて網羅します。さらに、それぞれの段階の詳細を定義し、各段階の実施範囲を定義します。

### インフラストラクチャ・チェックリストの作成

定義した通りの内容で作業を進めるために、次に示すチェックリストを作成します。最新の ALM 構成 ([http://www.hp.com/jp/TDQC\\_SysReq](http://www.hp.com/jp/TDQC_SysReq)) を参考に、推奨されるサポート対象マトリックスに基づいて定義をチェックします。チェックリストには、サーバの名前、設定、ストレージ、メモリ、ファイル・システムの重要なパス、ソフトウェアなどすべての要件を記載します。たとえば、次のような表をプロジェクトごとに作成します。

サーバ A- IP XXX.X.XX.XX	
用途	このサーバは、新しい ALM バージョン 11.50 アプリケーション・サーバ専用
ハードウェア	メモリ: CPU: CPU クロック: ディスクの空き容量:
オペレーティング・システム	Windows 2003 Server SP 2, 64 ビット
ソフトウェア	IBM WebLogic 9 Apache
備考	* サーバ B (DB サーバ) への直接接続が必要です。 * サーバ C (リポジトリ・ストレージ・サーバ) への直接接続が必要です。 * \\x\y\z ディレクトリは、サーバ C のリポジトリ・ストレージにマッピングする必要があります。 * インストール・パスは c:\Program Files\HP\ALM です。

サーバのインストールが完了したら、このチェックリストに基づいてインストールと設定を検証します。

## 制限リストの作成

アップグレード中に対応が必要な制限事項の一覧を作成します。たとえば、データベースの認証パスワードの制限、セキュリティ上の制限、クライアントに関する制限などがあります。リスト内のすべての項目に対して、解決方法を詳細に記載します。

制限リストの作成では、現在、クライアント・マシンで ALM を展開している方法 (ALM のダウンロード機能やユーザ独自のパッケージ機能) に対応できるようにし、必要な準備や変更点があれば併せて記載します。

## プロジェクト・アップグレードのスケジュール作成

段階的アップグレードまたは一括アップグレードのいずれの場合も、プロジェクトのアップグレード・スケジュールを作成する必要があります。このスケジュールでは、各時間枠の中でプロジェクトの優先順位を指定します。プロジェクトのアップグレードに関する優先順位を決定した後、次に示す技術的な項目とプロジェクトの項目をチェックします。

- **技術的な項目:**
  - プロジェクトのアップグレードに伴うリスクの大きさ — たとえば、アーカイブ・プロジェクトのアップグレードは、日常業務で頻繁に使用するライブ・プロジェクトのアップグレードと比較すると、リスクは非常に低いと考えることができます。リスクの低いプロジェクトから作業を開始することによって、アップグレード・プロセスの経験と自信を積み重ねる方法もあります。反対に、リスクの低いプロジェクトは緊急性が低いため、後に残しておく方法もあります。
  - バックアップ、復元、アップグレードにかかる時間 — どのような側面から考えても、バックアップ、復元、アップグレードには時間がかかります。たとえば、1 日に多数のプロジェクトを詰め込み、バックアップと復元だけでも 1 日では終わらないような状況にならないように注意してください。DB 内のあらゆるサイズのプロジェクトについて、バックアップと復元にかかる時間を DBA に問い合わせた上で、テスト環境で実際にかかった時間を測定します。
- **プロジェクト:**
  - 所要時間 — 選択したアップグレード時間内に予定している作業について、プロジェクト管理者に問い合わせます。主要なマイルストーン (メジャー・リリースやパッチのテストなど) が計画されているかどうかを確認します。
  - 機能の緊急性 — 新リリースに含まれる新機能について、プロジェクトに参加しているマネージャとプロジェクト管理者に問い合わせます。新機能をプロジェクトに実装するか、どのような方法で実装するのかを把握します。新機能によって大きく影響されるプロジェクトがあれば、アップグレードで優先度の高いプロジェクトと見なします。
  - ユーザの依存度 — 段階的アップグレードでは、両方の ALM バージョンを同時に稼働することができます。複数のプロジェクトで作業しているユーザは、異なるバージョンの複数のプロジェクトへのログインが必要になる可能性があります。このような場合、ALM クライアントの各バージョンをダウンロードする必要があります。クライアントの権限の点でこのような処理ができない場合は、複数のプロジェクトを使用するユーザ・グループごとのアップグレードを検討してください。
  - バージョン管理 — バージョン管理が有効なプロジェクトでは、プロジェクトのアップグレード前に、すべてのプロジェクト・エンティティをチェックインする必要があります。チェックインを行わないと、アップグレードが失敗します。

## プロジェクトの担当者

アップグレード中のスケジュール調整や連絡のために、プロジェクトごとに 1~2 名の担当者を指名します。この担当者は、プロジェクトのアップグレードのスケジュール作成の関係者であることを確認してください。段階的アップグレー

ドの場合、アップグレードの実行前、実行中、実行後のプロジェクト・メンテナンスをユーザに通知するのは担当者の責任であるため、移行スケジュールについて担当者の承認を得てください。プロジェクトのアップグレード・スケジュールには、各プロジェクトについて次の内容を記載することをお勧めします。

- ドメイン
- プロジェクト名
- データベース・スキーマ
- リポジトリの場所
- アクティブかどうか
- 使用中かどうか
- 予定されているメンテナンス時間
- アップグレードのスケジュール
- 担当者
- 特別な考慮事項(移行作業に外部ツールを使用する, など)
- カスタマイズによる変更

この表を作成することにより、実際のアップグレード作業での進捗を追跡できます。

## ロールバック手順の作成

アップグレード手順で発生する可能性のある問題をまとめます。このような問題への対応として、リスクを軽減する移行プランを作成し、さらにリスクごとに結果プランを作成します。また必要に応じて、既存の環境での作業を容易にするロールバック手順を作成します。これは、一括アップグレードを行う場合に非常に重要な役割を果たします。予防措置として、アップグレードを開始する前に、プロジェクトとサイト管理スキーマをすべてバックアップしておきます。バックアップは、ダウンタイムの開始にできるだけ近く、プロジェクトを使用していない時間帯に作成するようにします。これにより、プロジェクトで作業できる時間を最大限に確保し、データ損失の危険性を最小限に抑えることができます。ロールバック手順の作成には、必ず担当者を割り当ててください。

## アップグレード後の動作確認テストの作成

テスト環境の検証が正式に完了したら、テスト結果に基づいて、アップグレード中、主にどのような領域で影響が発生するのかを特定します。影響を受ける領域は、ハイリスク領域として見なされ、各プロジェクトのアップグレード完了後に行う動作確認テストの対象になります。さらに、使用頻度の高い基本機能があれば、これも各プロジェクトの動作確認テストの対象になります。

## アップグレード後のチェックリストのレビュー

アップグレードが完了したら、次の内容をチェックします。

- プロジェクトのアップグレードが完了したことを、プロジェクトの担当者に通知します。
- サイト管理のプロジェクトに接続できることを確認します。
- ユーザに依頼して、ユーザが各自のアカウントを使用してすべての ALM プロジェクトにログインできることを確認します。
- 事前に作成しておいた動作確認テストの実施をプロジェクトの担当者に通知します。
- ユーザに依頼して日常業務を実行し、応答時間やエラーの発生状況に関するフィードバックを報告してもらいます。
- ユーザに依頼し、ALM の新機能をチェックし、フィードバックを報告してもらいます。
- ユーザ・グループに標準で設定される新機能の権限をチェックし、必要に応じて変更します。
- テスト環境で負荷テストを実行し、所定のユーザ数に対応できるかどうかを確認します。
- HP の統合ツールやサードパーティ・ツールを ALM で使用する場合は、下位互換性を検証し、フィードバックを報告します。

## 新バージョンのトレーニング資料とリリース・ノートの作成

ALM での日常業務に関する問い合わせを減らし、導入作業をスムーズに進める目的で、大きな改善点、制限事項、新機能の概要、ヒントなどを記載したリリース・ノートを作成します。このドキュメントは、テスト段階で報告されたフィードバックと、ALM のリリース時に発行される『HP Application Lifecycle Management ユーザーズ・ガイド』や『新機能』を元に作成します。

リリース・ノートやトレーニング資料を実際のアップグレード前に提供することにより、サポート・ケースの件数を減らすことができます。

## アップグレード後のサポート

アップグレードを計画する際には、実際のアップグレード作業が完了した後で問題が発生した場合や不明な点の説明のために、クライアントに必要なサポート期間を確保するようにしてください。サポート担当者を指名し、担当者の名前をプロジェクトのユーザに通知します。この担当者は、問題の解決と機能に関する質問への対応を行います。したがって、あらゆる新機能や変更内容に精通している人を指名する必要があります。また、トレーニングの実施も検討してください。

サポート期間中に使用する不具合の追跡システムの作成も検討するとよいでしょう。