HCLSoftware

Unica Plan V12.1.8 アップグレードガイド

Contents

Chapter 1. アップグレードの概要 ... 3
- アップグレードロードマップ ... 3
- インストーラーの機能 ... 4
- インストールのモード ... 5
  - サンプル応答ファイル ... 5
- アップグレード・インストールが失敗した場合のレジストリー・ファイルの修正 ... 6
- Unica Plan の資料とヘルプ ... 7

Chapter 2. Unica Plan のアップグレードの計画 ... 9
- 前提条件: ... 9
- 全 Unica 製品に関するアップグレード前提条件 ... 12
- クリーンアップのためのデータベース照会の実行 ... 13
- エラー・ログおよび警告メッセージ ... 13
- 既存のキャンペーン・プロジェクトまたは要求でのアップグレード ... 14
- Unica Plan データ・ソース情報ワークシート ... 14

Chapter 3. Unica Plan をアップグレードするには ... 15
- アップグレードの前にシステムをバックアップする ... 15
- インストーラーの実行および構成プロパティーの更新 ... 15
- 12.0.0 以前のバージョンからアップグレードする場合に実行する構成 ... 16
- データベースの手動アップグレード ... 17
- アップグレードされた Web アプリケーションの配置とアップグレード・プロセスの実行 ... 19
- Unica Planのアップグレードの確認 ... 20
- トリガー手順の復元 ... 21
- クラスター環境での Unica Plan のアップグレード ... 21

Chapter 4. 概要 ... 22
- Unica 製品の JVM パラメーターの構成 ... 22
- Websphere での Unica Plan の配置 ... 28
- WebLogic 上での Unica Plan のデプロイ ... 33
- JBoss への Unica Plan の配置 ... 36
- Unica Plan の Apache Tomcat へのデプロイ® ... 38
- セキュリティを強化するための追加構成 ... 40
  - X-Powered-By フラグを無効にする ... 40
  - 制限された Cookie パスの構成 ... 41
- チェックリストメニューのアクティブ化 ... 41
- Plan タスクでのラベルの同期 ... 42

Chapter 5. Unica Plan のアンインストール ... 43

Chapter 6. configTool ... 44

Chapter 7. Unica Plan 構成プロパティー ... 49
- Unica Plan ... 49
- Unica Plan | navigation ... 49
- Unica Plan | 概要 ... 52
- Unica Plan | umoConfiguration ... 53
- Unica Plan | umoConfiguration | 承認 ... 61
- Unica Plan | umoConfiguration | テンプレート ... 62
- Unica Plan | umoConfiguration | attachmentFolders ... 64
- Unica Plan | umoConfiguration | E メール ... 67
- Unica Plan | umoConfiguration | マークアップ ... 67
- Unica Plan | umoConfiguration | グリッド ... 69
- Unica Plan | umoConfiguration | ワークフロー ... 71
- Unica Plan | umoConfiguration | integrationServices ... 73
- Unica Plan | umoConfiguration | campaignIntegration ... 74
- Unica Plan | umoConfiguration | レポート ... 74
- Unica Plan | umoConfiguration | invoiceRollup ... 75
- Unica Plan | umoConfiguration | データベース ... 76
- Unica Plan | umoConfiguration | listingPages ... 80
- Unica Plan | umoConfiguration | objectCodeLocking ... 81
- Unica Plan | umoConfiguration | thumbnailGeneration ... 83
- Unica Plan | umoConfiguration | スケジューラー | intraDay ... 85
- Unica Plan | umoConfiguration | スケジューラー | 日次 ... 85
- Unica Plan | umoConfiguration | 通知 ... 86
- Unica Plan | umoConfiguration | 通知 | E メール ... 88
- Unica Plan | umoConfiguration | 通知 | プロジェクト ... 90
- Unica Plan | umoConfiguration | 通知 | projectRequest ... 93
- Unica Plan | umoConfiguration | 通知 | プログラム ... 94
- Unica Plan | umoConfiguration | 通知 | marketingObject ... 94
- Unica Plan | umoConfiguration | 通知 | 承認 ... 95
- Unica Plan | umoConfiguration | 通知 | 資産 ... 96
- Unica Plan | umoConfiguration | 通知 | 請求書 ... 97
- Unica Plan | umoConfiguration | 通知 | 不在中 ... 97

第 1 章. アップグレードの概要

Unica Plan のアップグレードは、Unica Plan をアップグレード、構成、および配置すると、完了です。Unica Plan アップグレード・ガイドには、Unica Plan のアップグレード、構成、および配置に関する詳細情報が記載されています。

アップグレード・ロードマップセクションを使用して、Unica Plan の使用に関する幅広い理解を得るアップグレード・ガイド。

アップグレードロードマップ

アップグレードロードマップを使用して、Unica のアップグレードに必要な情報を素早く見つけるPlan.

次を使用できますTable 1: Unica Plan アップグレード ロードマップ on page 3 Unica のアップグレードのために完了する必要があるタスクをスキャンするためのテーブルPlan:

Table 1. Unica Plan アップグレード ロードマップ

トピック	情報
アップグレードの概要 on page 3	このトピックには、以下の情報が含まれています。 • インストーラーの機能 on page 4 • インストールのモード on page 5 • Unica Plan の資料とヘルプ on page 7
Unica Plan のアップグレードの計画 on page 9	このトピックには、以下の情報が含まれています。 • 前提条件: on page 9 • 全 Unica 製品に関するアップグレード前提条件 on page 12 • エラー・ログおよび警告メッセージ on page 13 • Unica Plan データ・ソース情報ワークシート on page 14
Unica Plan をアップグレードするには on page 15	このトピックには、以下の情報が含まれています。 • アップグレードの前にシステムをバックアップする on page 15 • インストーラーの実行および構成プロパティーの更新 on page 15 • データベースの手動アップグレード on page 17 • アップグレードされた Web アプリケーションの配置とアップグレード・プロセスの実行 on page 19 • Unica Planのアップグレードの確認 on page 20 • トリガー手順の復元 on page 21

Table 1. Unica Plan アップグレード ロードマップ

(continued)

トピック	情報
	• 既存のキャンペーン・プロジェクトまたは要求でのアップグレード on page 14 • クラスター環境での Unica Plan のアップグレード on page 21
概要 on page 22	このトピックには、以下の情報が含まれています。 • Websphere での Unica Plan の配置 on page 28 • WebLogic 上での Unica Plan のデプロイ on page 33 • JBoss への Unica Plan の配置 on page 36 • Unica Plan の Apache Tomcat へのデプロイ on page 38
Unica Plan のアンインストール on page 43	このトピックでは、Unica をアンインストールする方法についての情報を提供しますPlan.
設定ツールユーティリティ on page 44	UnicaPlan の設定ツールユーティリティの詳細を読む.

インストーラーの機能

どの Unica 製品をインストールまたはアップグレードする場合も、スイート・インストーラーおよび製品インストーラーを使用する必要があります。たとえば、Unica Plan をインストールするには、Unica スイート・インストーラーと Unica Plan インストーラーを使用する必要があります。

Unica スイート・インストーラーおよび製品インストーラーを使用する前に、以下のガイドラインを確認してください。

- Unicaインストーラーと製品インストーラーは、製品をインストールするコンピューターの同じディレクトリーにある必要があります。Unica インストーラーのあるディレクトリーに製品インストーラーの複数のバージョンが存在する場合、Unica インストーラーはインストール・ウィザードの「Unica 製品」画面で常に製品の最新バージョンを表示します。
- Unica 製品のインストール直後にパッチをインストールする場合は、パッチのインストーラーがスイートおよび製品のインストーラーと同じディレクトリーにあるようにしてください。
- Unica インストールのデフォルトの最上位ディレクトリーは `/HCL/Unica` (UNIX™) または `C:\HCL\Unica` (Windows™) です。ただし、このディレクトリーはインストール時に 変更できます。

インストールのモード

Unica スイート・インストーラーは、GUI モード、X Window System モード、コンソール・モード、またはサイレント・モード (無人モードとも呼ぶ) のいずれかのモードで実行できます。Unica Plan をインストールする際は要件に見合ったモードを選択してください。

アップグレードの場合は、初期インストール時に実行するタスクと同じ多くのタスクをインストーラーを使用して実行します。

GUIX Window System モード モード

グラフィカル・ユーザー・インターフェースを使用して Unica Plan をインストールするには、Windows™ の GUI モード、または UNIX™ の X Window System モードを使用します。

UNIX™ X Window System モード

グラフィカル・ユーザー・インターフェースを使用して Unica Plan をインストールするには、UNIX™ の X Window System モードを使用します。

コンソール・モード

コマンド・ライン・ウィンドウを使用して Unica Plan をインストールするには、コンソール・モードを使用します。

注: コンソールモードでインストーラー画面を正しく表示するために、端末ソフトをUTF-8の文字コードに対応するように設定してください。ANSI などその他の文字エンコードでは、テキストが正しくレンダリングされず、一部の情報が読み取れなくなります。

サイレント・モード

Unica Plan を複数回インストールするには、サイレント・モード (無人モード) を使用します。サイレント・モードは、インストールに応答ファイルを使用し、インストール・プロセスの間にユーザー入力を必要としません。

注: クラスター化された Web アプリケーションやクラスター化されたリスナー環境では、サイレント・モードはアップグレード・インストールでサポートされていません。

サンプル応答ファイル

Unica Plan のサイレント・インストールをセットアップするため、応答ファイルを作成する必要があります。応答ファイルを作成する際には、サンプル応答ファイルを利用できます。サンプル応答ファイルは、インストーラーの `ResponseFiles` 圧縮アーカイブに含まれています。

次の表には、サンプル応答ファイルに関する情報が示されています。

表 2. サンプル応答ファイルの説明

サンプル応答ファイル	説明
`installer.properties`	Unica マスター・インストーラーのサンプル応答ファイル。
`installer_`*`product initials and product version number`*`.properties`	Unica Plan インストーラーのサンプル応答ファイル。 例えば、`installer_uc`*`n.n.n.n`*`.properties` (ここで、*n.n.n.n* はバージョン番号) は、Unica Campaign インストーラーの応答ファイルです。 例えば、`installer_ump`*`n.n.n.n`*`.properties` (ここで、*n.n.n.n* はバージョン番号) は、Unica Plan インストーラーの応答ファイルです。 例えば、`installer_ul`*`n.n.n.n`*`.properties` (ここで、*n.n.n.n* はバージョン番号) は、Leads インストーラーの応答ファイルです。
`installer_`*`report pack initials, product initials, and version number`*`.properties`	レポート・パック・インストーラーのサンプル応答ファイル。 例えば、`installer_urpc`*`n.n.n.n`*`.properties` (*n.n.n.n* はバージョン番号) は、Unica Campaign レポート・パック・インストーラーの応答ファイルです。 例えば、`installer_urpl.properties` は、Leads レポート・パック・インストーラーの応答ファイルです。

アップグレード・インストールが失敗した場合のレジストリー・ファイルの修正

インストール済み製品の基本バージョンをインストーラーが検出できなかったためにインストールが失敗した場合、ここに説明されている方法でレジストリー・ファイルを修正できます。

このタスクについて

`.com.zerog.registry.xml` という名前の InstallAnywhere Global レジストリー・ファイルは、Unica 製品のインストール時に作成されます。このレジストリー・ファイルは、そのサーバー上にインストールされているすべての Unica 製品 (その各機能とコンポーネントを含む) をトラッキングします。

1. `.com.zerog.registry.xml` ファイルを見つけます。

   製品をインストールするサーバーに応じて、`.com.zerog.registry.xml` ファイルは次のいずれかの場所にあります。

- Windows サーバーの場合、ファイルは `Program Files/Zero G Registry` フォルダーにあります。

  `Zero G Registry` 非表示ディレクトリーです。非表示のファイルとフォルダーを表示する設定を有効にする必要があります。

- UNIX システムの場合、ファイルは以下のいずれかのディレクトリーにあります。
  - root ユーザー - `/var/`
  - 非ルートユーザー - `$HOME/`

2. ファイルのバックアップ・コピーを作成します。
3. ファイルを編集して、インストール済み製品のバージョンを参照するすべての項目を変更します。

   たとえば、これはUnicaPlanバージョン8.6.0.3に対応するファイルのセクションです。

```
<product name="Plan" id="dd6f88e0-1ef1-11b2-accf-c518be47c366"
version=" 8.6.0.3 " copyright="2013" info_url="" support_url=""
location="<HCL_Unica_Home>\Plan" last_modified="2013-07-25 15:34:01">
```

   この例では、`version=" 8.6.0.3 "` を参照するすべての項目を基本バージョン (このケースでは 8.6.0.0) に変更します。

Unica Plan の資料とヘルプ

以下の表では、Unica Plan のインストールに関する様々なタスクについて説明しています。

「文書」 列には、タスクに関して詳しい情報が記載されている資料の名前が含まれています。

表 3. 起動して稼働状態にする

タスク	ドキュメンテーション
新機能、既知の問題、および回避策についてのリストを表示	Unica Plan リリース・ノート
Unica Plan をインストールまたはアップグレードし、Unica Plan Web アプリケーションを配置する	以下のいずれかのガイド: • Unica *Plan* インストール・ガイド • Unica *Plan* アップグレード・ガイド

以下の表には、Plan における管理タスクが記述されています。**「文書」** 列には、タスクに関して詳しい情報が記載されている資料の名前が含まれています。

表 4. Unica を構成して使用する Plan

タスク	ドキュメンテーション
• ユーザー用にシステムをセットアップおよび構成する • セキュリティー設定の調整 • テーブルのマッピング、オファー・テンプレートの定義、およびカスタム属性 • ユーティリティーの実行およびメンテナンスの実行	*Unica Plan* 管理者ガイド
• マーケティング計画、プログラム、プロジェクトの作成および管理 • 進捗状況と結果の分析	*Unica Plan* ユーザーズ・ガイド

以下の表には、Unica Plan のオンライン・ヘルプおよび PDF の取得に関する情報が含まれています。**「指示」**列には、オンライン・ヘルプの開き方および Unica Plan の文書へのアクセス方法が説明されています。

表 5. ヘルプの使用

タスク	手順
オンライン・ヘルプを開く	1. **「ヘルプ」** > **「このページのヘルプ」**を選択して、コンテキスト・ヘルプのトピックを開きます。 2. ヘルプ・ウィンドウ内の**「ナビゲーションの表示」**アイコンをクリックすると、ヘルプ全体が表示されます。 オンラインのコンテキスト・ヘルプを表示するには、Web アクセスが必要です。オフライン資料として Knowledge Center をローカルで利用する方法、およびインストールする方法について詳しくは、HCL サポートにお問い合わせください。
PDF の取得	以下のいずれかの方法に従います: • **「ヘルプ」** > **「製品資料」**を選択して、Plan の PDF にアクセスします。 • **「ヘルプ」** > **「Unica Suite のすべての資料」**を選択して、使用可能なすべての資料にアクセスします。

Chapter 2. Unica Plan のアップグレードの計画

アップグレード対象Unica Plan 12.1.8 を更新するには、どのバージョンからアップグレードしようとしているかを確認する必要があります。アップグレードのシナリオは、Unica Plan の現行バージョンに基づいています。

Unica Plan は以下のアップグレードパスをサポートしています:

- 12.1.x → 12.1.8
- 12.1.0.x → 12.1.8

8.6.x より前のバージョンを使用している場合:

- 既存のバージョンからバージョン 8.6.0 への高速アップグレードを実行します (詳しくは、 *HCL Unica 8.6.0* 高速アップグレードガイドを参照してください)。
- バージョン 8.6.0 からバージョン 12.1.0 への高速アップグレードを実行します (詳細については、 *HCL Unica 12.1.0* 高速アップグレードガイドを参照してください)。
- バージョン 12.1.0 からバージョン 12.1.8 へのインプレースアップグレードを実行します。

11.1.xx より前のバージョンのユーザーは:

- 既存のバージョンからバージョン 12.1.0 への高速アップグレードを実行します (詳しくは、 *HCL Unica 12.1.0* 高速アップグレードガイドを参照してください)。
- バージョン 12.1.0 からバージョン 12.1.8 へのインプレースアップグレードを実行します。

バージョン 11.1.xx/12.0.xx のお客様は、以下のアップグレードオプションのいずれかを使用できます:

- **オプション 1**
  - 既存のバージョンからバージョン 12.1.0 へのインプレースアップグレードを実行します。
  - バージョン 12.1.0 からバージョン 12.1.8 へのインプレースアップグレードを実行します。
- **オプション 2**
  - 既存のバージョンからバージョン 12.1.0 への高速アップグレードを実行します (詳しくは、 *HCL Unica 12.1.0* 高速アップグレードガイドを参照してください)。
  - バージョン 12.1.0 からバージョン 12.1.8 へのインプレースアップグレードを実行します。

前提条件:

Unica 製品をインストールまたはアップグレードするには、その前に、ご使用のコンピューターがすべてのソフトウェアおよびハードウェアの前提条件を満たしていることを確認する必要があります。

システム要件

サポートされるバージョンについて詳しくは、「推奨されるソフトウェア環境および最小システム要件」を参照してください。

ネットワーク・ドメイン要件

スイートとしてインストールされる Unica 製品は同じネットワーク・ドメインにインストールする必要があります。これは、クロスサイト・スクリプティングで生じ得るセキュリティー・リスクを制限することを目的としたブラウザー制限に準拠するためです。

JVM 要件

スイートに含まれる Unica アプリケーションは、専用 Java™ 仮想マシン (JVM) に配置する必要があります。Unica 製品は、Web アプリケーション・サーバーが使用する JVM をカスタマイズします。

知識要件

Unica 製品をインストールするには、製品をインストールする環境全般に関する知識が必要です。この知識には、オペレーティング・システム、データベース、および Web アプリケーション・サーバーに関する知識が含まれます。

インターネット・ブラウザー設定

ご使用のインターネット・ブラウザーが、以下の設定に準拠していることを確認してください。

- ブラウザーで Web ページをキャッシュしない。
- ブラウザーはポップアップ・ウィンドウをブロックしてはなりません。

アクセス権限

インストール作業を完了するため、以下のネットワーク権限を保持していることを確認してください。

- 必要なすべてのデータベースに対する管理権限。

  注: 管理者は、テーブルとビューの両方に関する `CREATE`、`SELECT`、`INSERT`、`UPDATE`、`DELETE`、および `DROP` 権限がなければなりません。

- Webアプリケーション・サーバーの実行に使用するオペレーティング・システム・アカウントの関連ディレクトリーおよびサブディレクトリーへの読み取りおよび書き込みアクセスと Unica コンポーネント。
- 編集する必要があるすべてのファイルに対する書き込み権限。
- アップグレードする場合は、インストール・ディレクトリーやバックアップ・ディレクトリーなど、ファイルを保存する必要があるすべてのディレクトリーに対する書き込み権限。
- インストーラーを実行するための適切な読み取り、書き込み、および実行のアクセス許可。

Web アプリケーション・サーバーの管理パスワードを保持していることを確認してください。

UNIX™ の場合、製品のすべてのインストーラー・ファイルはフル権限 (例えば、rwxr-xr-x) が必要です。

UNIX™ の場合、以下の追加の権限が必要です。

- Unica Plan と Unica Platform をインストールするユーザー・アカウントは、Unica Campaign ユーザーと同じグループのメンバーである必要があります。このユーザーアカウントには、有効なホーム・ディレクトリーがあり、そのディレクトリへの書き込み権限が必要です。
- HCL Unica製品のすべてのインストーラー・ファイルには、rwxr-xr-xなどの完全な権限が必要です。

Unica Planをインストールする前に考慮すべきポイント

Unica Planのインストールでは、次の点を考慮する必要があります。

JAVA_HOME 環境変数

Unica 製品をインストールするコンピューターに JAVA_HOME 環境変数が定義されている場合、サポートされる JRE のバージョンがこの変数で指定されていることを確認してください。システム要件については、以下を参照してください。*Unica 推奨されるソフトウェア環境と最小システム要件ガイド*。

JAVA_HOME 環境変数が正しくない JRE を指している場合、Unica インストーラーを実行する前に、その JAVA_HOME 変数をクリアする必要があります。

以下のいずれかの方法により、JAVA_HOME 環境変数をクリアできます。

- Windows™: コマンド・ウィンドウで、set JAVA_HOME= (空のままにする) と入力して、**Enter** キーを押します。
- UNIX™: 端末で、export JAVA_HOME= (空のままにする) と入力して、**Enter** キーを押します。

端末で次のコマンドを実行して、JAVA_HOME 環境変数をクリアできます。

export JAVA_HOME= (空のままにする)

Unica インストーラーは、Unica インストールの最上位ディレクトリーに JRE をインストールします。個々の Unica アプリケーションのインストーラーは、JRE をインストールしません。その代わりに、Unica インストーラーによってインストールされた JRE の場所を指定します。すべてのインストールが完了した後に環境変数を再設定することができます。

サポートされているJREの詳細については、Unica推奨ソフトウェア環境および最小システム要件ガイドを参照してください。

Unica Platform 要件

Unica 製品をインストールまたはアップグレードする前に、Unica Platform をインストールまたはアップグレードする必要があります。一緒に機能する製品のグループごとに、Unica Platform を 1 回だけインストールまたはアップグレードする必要があります。各製品インストーラーは、必要な製品がインストールされているかどうかを検査します。ご使用の製品またはバージョンが Unica Platform に登録されていない場合、インストールを続行する前に、Unica Platform をインストールまたはアップグレードすることを求めるメッセージが表示されます。**「設定」>「構成」**ページでプロパティーを設定するには、その前に、Unica Platform が配置済みであり、稼働している必要があります。

注: UNIX にインストールする場合、Web アプリケーション・サーバーで、`Djava.awt.headless` プロパティーを true に設定しなければならないことがあります。この設定は、Unica Optimizeレポートを表示できない場合にのみ必要です。詳細については、Unica Campaign インストール・ガイドを参照してください。Unica Optimize は Unica

Campaign システム・テーブルのデータ・ソースを使用するため、Unica Optimize 用に追加のデータ・ソースを準備する必要はありません。

注: バージョン 12.0.0 以降では、データベース・タイプ Informix が機能していないため、選択しないようにしてください。バージョン 12.1.0.3 以上では、ユーザーは OneDB データベースをシステム・テーブルおよびユーザー・テーブルとして使用できます。詳しくは、「*Unica V12.1.0.3* インストール・ガイド *(OneDB* 用)」を参照してください。

注: バージョン 12.1.8 の場合、PostgreSQL は Interact レポートと PatternStateETL ddl をサポートしません。

全 Unica 製品に関するアップグレード前提条件

Plan をアップグレードする前に、権限、オペレーティング・システム、および知識に関するすべての要件を正しく満たしてくださいシームレスなアップグレード・エクスペリエンスを保証します。

以前のインストールで生成された応答ファイルの削除

8.6.0 より前のバージョンからアップグレードする場合は、以前の Unica Plan のインストールによって生成された応答ファイルを削除する必要があります。古い応答ファイルには、8.6.0 以降のインストーラーとの互換性がありません。

古い応答ファイルを削除しないと、インストーラーの実行時にインストーラーフィールドに誤ったデータが事前に入力されたり、インストーラーが一部のファイルのインストールに失敗したり、構成手順をスキップしたりする可能性があります。

応答ファイルの名前は `installer.properties` です。

各製品の応答ファイルの名前は `installer_productversion.properties` です。

インストーラーは、インストール時に指定したディレクトリーに応答ファイルを作成します。デフォルトの場所は、ユーザーのホーム・ディレクトリーです。

UNIX™ のユーザー・アカウント要件

UNIX™ では、製品をインストールしたユーザー・アカウントでアップグレードを実行する必要があります。そうしないと、インストーラーは前のインストールの検出に失敗します。

32 ビットから 64 ビットへのバージョンアップ

Unica Plan の 32 ビット・バージョンから 64 ビット・バージョンに移行する場合、次のタスクを完了していることを確認してください。

- 製品データ・ソースのデータベース・クライアント・ライブラリが 64 ビットであることを確認してください。
- スタートアップ・スクリプトや環境スクリプトなど、関連するすべてのライブラリパスが、64 ビット・バージョンのデータベース・ドライバーを正しく参照していることを確認してください。

以下のメモリーからの未使用ファイルのアンロード: AIX®

AIX® でのインストールの場合、インストーラーをアップグレード・モードで実行する前に、AIX® インストールに付属の slibclean コマンドを実行して、未使用のライブラリーをメモリーからアンロードします。

注: slibclean コマンドは root ユーザーとして実行する必要があります。

カスタムファイルのバックアップ

Unica 12.1.8 へのアップグレードを開始する前に、`<UNICA_HOME>/jre/` の場所に存在しているすべてのカスタム・ファイルをバックアップしてください。`<UNICA_HOME>/jre/` の場所に存在するカスタム・ファイルをバックアップしない場合、Unica 12.1.8 のアップグレードによって既存の `jre` フォルダーが削除され、Oracle JRE ファイルを含む新しい `jre` フォルダーがインストールされるため、ファイルが失われます。

注: オペレーティング・システムが IBM AIX の場合、Unica 12.18 は IBM JRE をインストールします。

クリーンアップのためのデータベース照会の実行

Unica Plan をアップグレードする前にデータベース照会を実行して、重複するプロジェクト要求 ID があれば削除します。

このタスクについて

Unica Plan のアップグレードの成功を確実にするため、データベース内で照会を実行し、この照会によって戻された重複している結果をすべて見つけて削除します。

以下のステップを実行して、データベース照会を実行します。

1. Unica Plan システム・テーブルを保持するデータベース・コンソールを開きます。
2. 以下の照会を入力します。

```
 SELECT proj_request_id, count(proj_request_id) num
 FROM uap_projects
 WHERE proj_request_id in (SELECT project_id FROM uap_projects WHERE
state_code = 'ACCEPTED')
 group by proj_request_id
 having count(proj_request_id) > 1
```

3. この照会は重複するプロジェクト要求 ID を戻します。結果を分析して、重複している行のうちどちらを使用していてどちらを削除できるのかを判別します。削除するレコードを決定する参考にするため、`uap_projects_last_mod_date` 表を見たり参照テーブルのデータを表示したりできます。行を削除するには、データベース上で 削除照会を実行します。重複している行が削除されないと、アップグレードが失敗する場合があります。

エラー・ログおよび警告メッセージ

アップグレードの際、システムはプロセス中に生成されたメッセージを記録します。アップグレード中に発生した情報またはエラー・メッセージを見るには、ログ・ファイルを参照してください。

参照情報として、それらのメッセージが含まれるログ・ファイルは以下のファイルおよびデータベース表にあります。

- *<HCL_Unica_Home>*/HCL_Unica_Installer_Install<date_time>.log
- *<Plan_Home>*/Plan_Install_<date_time>.log
- *<Platform_Home>*/Platform_Install<date_time>.log
- *<USER_HOME>*/HCL_Unica_Installer_stdout.log
- *<USER_HOME>*/HCL_Unica_Installer_stderr.log
- *<USER_HOME>*/Platform_stdout.log
- *<USER_HOME>*/Platform_stderr.log
- *<USER_HOME>*/Plan_stdout.log
- *<USER_HOME>*/Plan_stderr.log

既存のキャンペーン・プロジェクトまたは要求でのアップグレード

Unica Plan と統合された Unica Campaign システムをアップグレードする場合で、既存のキャンペーン・プロジェクトに対応するリンクされたキャンペーンがない場合、Unica Plan へアップグレードする前にリンクされたキャンペーンを作成します。同様に、キャンペーン・プロジェクト用の既存のプロジェクト要求がある場合は、Unica Plan にアップグレードする前に、要求を受け入れるか、または拒否してください。

アップグレード前にリンクしない場合、システムのアップグレード後にそれらのプロジェクト用にキャンペーンの作成を試みたり、または要求を受け入れたりする場合に、キャンペーンが正しく Unica Plan プロジェクトへリンクされません。

Unica Plan データ・ソース情報ワークシート

Unica Plan のインストール・ワークシートを使用して、Unica Plan データベースに関する情報と、Unica Plan のインストールに必要なその他の Unica 製品に関する情報を収集してください。

表 6. データ・ソース情報ワークシート

項目	値
データ・ソース・タイプ	SQL Server / IBM DB2 / Oracle / MariaDB / OneDB / PostgreSQL。 **注:** PostgreSQL サポートは、12.1.7 からアップグレードしている場合のみ使用可能です。
データ・ソース名	Plan データ・ソースの名前。
データ・ソースのアカウント・ユーザー名	データベース・ユーザー名。
データ・ソースのアカウント・パスワード	データベース・パスワード。
JNDI 名	`plands`
JDBC ドライバーへのパス	JDBC ドライバー・ファイルのパス。

第 3 章. Unica Plan をアップグレードするには

Unica Plan をアップグレードするには、既存のインストール済み環境をバックアップし、Unica Platform がアップグレードされて稼働していることを確認し、インストーラーを実行し、トリガー手順があればすべて復元し、アップグレードされたアプリケーションを配置し、次にいくつかの配置後の処理を実行します。

このタスクについて

Unica Plan の以前のバージョンは Affinium Plan または Marketing Operations という名前でした。本書では、すべてのバージョンを Unica Plan と呼んでいます。

アップグレードの前にシステムをバックアップする

アップグレード・プロセスを始める前にシステムのバックアップを取ります。アップグレードが失敗した場合に、直近バージョンの Unica Plan を復元できます。

このタスクについて

システムをバックアップするには、以下の手順を完了します。

1. 既存のバージョンの Unica Plan を配置解除します。
2. 既存のインストール・フォルダー内のすべてのファイルおよびディレクトリーをバックアップします。

   注: サンプル・トリガー手順または `procedure_plugins.xml` ファイルを変更していた場合、トリガー手順が失われないために、アップグレード後にバックアップからファイルを復元する必要があります。復元する必要のあるファイルは、`/devkits/integration/examples/src/procedure` フォルダー内にあります。

3. Unica Plan システム・テーブルを保持するデータベースをバックアップします。

インストーラーの実行および構成プロパティーの更新

インストーラーを実行する前に、Unica Platform データベースおよび Unica Plan データベースについて、適切なデータベース接続情報を保有していることを確認してください。

このタスクについて

重要: Microsoft Windows 2019 サーバーおよび Microsoft Windows 2022 サーバーで、インストーラーを起動するには、コマンド・プロンプトにパラメーター `SET JAVA_TOOL_OPTIONS="-Dos.name=Windows 7"` を設定して、同じコマンド・プロンプトからインストーラーを起動します。

インストーラーを実行して構成プロパティーを更新するには、以下の手順を完了します。

1. インストーラーを実行し、使用するインストール・ディレクトリーとして、既存のインストール・ディレクトリーを指定します。詳しくは、「インストーラーの機能 ページ 4」を参照してください。

   インストーラーは、以前のバージョンがインストールされていることを検出し、アップグレード・モードで実行されます。

2. インストール・ウィザードの指示に従います。

   注: インストーラーが自動的にデータベースをアップグレードできることに注意してください。会社の方針が、この機能の使用をユーザーに許可していない場合は、ソフトウェアのインストール後、Web アプリケーションを配置する前に、**「手動データベース・セットアップ」**オプションを選択してから手動でスクリプトを実行します。

3. インストーラーが終わったら、conf/plan_ehcache.xml ファイルに属性名 ="`PlanEhCacheManager`" が含まれていることを確認します。含まれていない場合は、ファイルを編集して追加します。例: `<ehcache updateCheck="false" name="PlanEhCacheManager">`
4. アップグレードされた Unica Platform アプリケーションにログインし、**「設定」**>**「構成」**を選択します。Unica Plan カテゴリーでプロパティーを確認し、任意の項目を設定または変更します。

12.0.0 以前のバージョンからアップグレードする場合に実行する構成

12.0.0 以前のバージョンの Unica Plan からアップグレードしている場合は、アップグレードの後で Unica Platform に2 つの Plan メニュー (1 つは Plan のすべての機能があるメニュー、もう 1 つはチェックリスト機能だけがあるメニュー) が存在する可能性があります。

このタスクについて

2 つの Plan メニューの問題を回避するには、以下の手順を実行します。

1. アップグレードを完了していない場合は、以下の手順を実行します。

   a. アップグレードインストーラーを実行します。

   b. アップグレードの後で Webアプリケーション・サーバーの開始前に、ロケールごとに以下の手順を実行します。

      i. `conf/locale` の場所でファイル `sysmenu.xml` を確認します。

      ii. `<menugroup id="projectmanagement">` が見つかった場合は、`<menugroup id="Operations">` と置き換えます。

2. アップグレードを完了していて、2 つの Plan メニューがすでに存在する場合は、以下の手順を実行して、問題を修正します。

   a. Plan Web アプリケーションを停止します。

   b. `conf/locale` の場所からファイル `sysmenu.xml` にアクセスします。

c. このファイル内で、`conf/locale/<menugroup id="projectmanagement">` が見つかった場合は、その中の `<menugroup id="Operations">` セクションを削除し、`<menugroup id="projectmanagement">` の名前を `<menugroup id="Operations">` に変更します。

d. すべてのロケールに対してステップ *c* を実行します。

e. Plan Web アプリケーションを再始動します。

f. Unica Platform ホームページから、**「設定」 > 「Plan 設定」**に移動します。
結果
「管理設定」ページが表示されます。

g. **「システム管理」**設定内の**「制限オプション」**内で、**「メニューの同期」**を選択します。

h. Unica Platform と Unica Plan の両方を再始動します。

データベースの手動アップグレード

インストーラーは、アップグレードプロセス中に、 Unica Plan データベースをアップグレードできます。会社の方針でデータベースのアップグレードが許可されない場合は、データベース・セットアップ・ユーティリティーである `umodbsetup` を使用してテーブルを手動でアップグレードできます。

`umodbsetup` ユーティリティーにより、以下のアクションのいずれかを実行します。

- オプション 1: Unica Plan データベースでシステム・テーブルをアップグレードし、必要なデフォルト・データをシステム・テーブルに追加します。
- オプション 2: データベースをアップグレードしてデータを追加するためのスクリプトをファイルに出力します (このファイルは、後で、ユーザーまたはデータベース管理者がユーザーのデータベース・クライアントで実行できます)。

環境変数の構成

`umodbsetup` を実行する前に、以下の手順を実行して、環境変数を適切に構成します。

1. テキスト・エディターで、`<Plan_Home>\tools\bin` ディレクトリーにある `setenv` ファイルを見つけ、開きます。
2. JAVA_HOME 変数が正しい Java™ インストールのディレクトリーを示しており、DBDRIVER_CLASSPATH 変数の最初の項目が JDBC ドライバーであることを確認します。環境変数の設定について詳しくは、「Plan インストール・ガイド」を参照してください。
3. ファイルを保存して閉じます。
4. `<Plan_Home>\tools\bin` ディレクトリーにある `umo_jdbc.properties` ファイルを見つけ、開きます。
5. 以下のパラメーターの値を設定します。
    - `umo_driver.classname`
    - `umo_data_source.url`

- umo_data_source.login
- umo_data_source.password

6. ファイルを保存して閉じます。

データベース・セットアップ・ユーティリティー

コマンド・プロンプトまたは UNIX™ シェルから、*<Plan_Home>*\tools\bin ディレクトリーにナビゲートします。umodbsetup ユーティリティーを実行し、自身の状況に必要なパラメーターに適切な入力データを指定してください。

例えば、次のコマンドは、アップグレードを実行し、ロケールを en_US に設定して、ロギング・レベルを medium に設定します。

```
./umodbsetup.sh -t upgrade -L en_US -l medium
```

ユーティリティーについて指定できるすべての変数の説明は以下のとおりです。

表 7. umodbsetup.sh ユーティリティーの変数

Variable	説明
-b	アップグレードの場合のみ。アップグレードしようとしているデータベースの基本バージョンを識別します。 デフォルトで、ユーティリティーは、アップグレードしようとしているデータベースのバージョンを検出します。ただし、以前にデータベースをアップグレードしようとしたときに何らかの形で失敗していた場合、アップグレードが失敗してもバージョンが更新されていることがあります。問題を修正して再びユーティリティーを実行するときには、この変数を -f 変数と共に使用して、正しい基本バージョンを指定してください。 例えば、次のようになります。 -f -b 12.1.0.0
-f	アップグレードの場合のみ。データベースで検出される基本バージョンをオーバーライドして、-b 変数で指定された基本バージョンがユーティリティーで使用されるようにします。-b 変数の説明を参照してください。
-h	ユーティリティーのヘルプを表示します。
-l	umodbsetup ユーティリティーによって実行されるアクションからの出力を umo-tools.log ファイルに記録します。このファイルは、*<HCL_Unica_Home>**<Plan_Home>*\tools\logs ディレクトリーにあります。この変数はロギング・レベルを指定します。 ロギング・レベルは high、medium、または low に設定できます。

表 7. umodbsetup.sh ユーティリティーの変数

(続く)

Variable	説明
-L	インストールのデフォルト・ロケールを設定します。例えば、ドイツ語版のインストールでは `-L de_DE` を使用してください。 ロケールについて有効な入力値としては、de_DE、en_GB、en_US、es_ES、fr_FR、it_IT、ja_JP、ko_KR、pt_BR、ru_RU、zh_TW、zh_CN があります。
-m	スクリプトを *<HCL_Unica_Home>\<Plan_Home>\tools* ディレクトリー内のファイルに出力します。このファイルは後で手動で実行することができます。このオプションは、データベース・クライアント・アプリケーションからスクリプトを実行する必要がある場合に使用してください。この変数を使用すると、スクリプトが `umodbsetup` ツールによって実行されなくなります。
-t	データベース・インストールのタイプ。有効な値は `full` と `upgrade` です。例えば、次のようになります。 `-t full`
-v	冗長。

データベース・スクリプトの手動での実行 (必要な場合)

`-m` 変数を使用してスクリプトを出力し、データベース・クライアント・アプリケーションから実行できるようにしてある場合は、ここで、そのスクリプトを実行してください。

システム・テーブルをアップグレードしてデータを追加する前に `plan.war` ファイルを配置しないでください。

アップグレードされた Web アプリケーションの配置とアップグレード・プロセスの実行

アップグレードされた Web アプリケーションを Web アプリケーション・サーバーへ配置する必要があります。Web アプリケーションを配置した後で、アップグレード・プロセスを開始できます。

開始する前に

注: Unica Plan が Campaign と統合されている場合、続行する前に、Campaign がアップグレードされており実行中であることを確認してください。

1. 概要 ページ 22 で説明するように、Unica Plan をご使用の Web アプリケーション・サーバーに配置します。
2. アプリケーション・サーバーを再起動します。

3. アプリケーションが稼働しているときに、ログインして、アップグレードが正しく行われたことを確認します。**「設定」>「構成」**を選択し、左側のリストに「**Unica Plan**」があることを確認します。その後、「**Unica Plan**」セクションを展開して、**「umoConfiguration」**カテゴリーがリストに含まれていることを確認します。
4. **「設定」>「Plan 設定」**を選択します。
5. 下にスクロールして**「Unica Plan のアップグレード」**をクリックします。アップグレード・プロセスのリストが表示されます。
   これらのプロセスは、データベース表と、サイトに特定のカスタマイズを保管するファイルとをアップグレードすることにより、アプリケーションの構成を変更します。

   アップグレード・プロセスについて詳しくは、そのプロセスの横にある**「ヘルプ」**をクリックしてください。
6. 選択したプロセスを実行するには、**「アップグレード」**をクリックします。

Unica Planのアップグレードの確認

Unica Platform をアップグレードする前に、Unica Plan をアップグレードおよび配置する必要があります。

About this task

アップグレードを検証するには、以下のステップを完了します。

1. アプリケーション サーバーのログ ディレクトリ内のログ ファイルでエラー メッセージを確認しますメッセージ「UAPContext Init failed」は、アップグレードが正常に完了しなかったことを示しています。
2. Internet Explorer またはサポートされる他のブラウザーを使用して、Unica URL にアクセスします。
3. 資産ファイルなどの、さまざまな Unica Plan フィーチャーに移動します。[操作] または [プラン設定] メニューが 2 つ表示されている場合は、**設定 > プラン設定**に移動し、 [**メニューの同期**] をクリックし、アプリケーションからログアウトして、もう一度ログインします。
4. 計画、プログラム、プロジェクト、独自のカスタム・マーケティング・オブジェクト・タイプなど、 さまざまな Unica Plan オブジェクトのインスタンスを作成します。

5. **Note:** このステップは、オファー管理の一元管理**への移行**ユーティリティを使用して、オファー管理を Unca Planから Unica 中央集中管理に移行する予定の場合にのみ適用できます。

   リバースプロキシまたは Web アクセス管理ソフトウェア (ISAM、Siteminder など) を使用している場合、`serverURLInternal`に Unica 中央集権管理の直接内部 URL を入力します。
   Example
   `serverURLInternal=http://<hostname>:<port>/Offer`
6. **[設定] > [プランの設定] > [テンプレートの設定]**を選択し、**[テンプレートの検証]** をクリックします。
7. **[メニューの同期]**をクリックします。アプリケーションからログアウトし、再度ログインして、**「プランの設定」メニュー**項目を確認してください。
8. インストール済み環境で Unica Plan がアプリケーション・プログラミング・インターフェースによってカスタマイズされている場合、 そのカスタマイズが互換性問題の影響を受けないことを確認してください。
9. トリガー手順を使用する場合は、それらを復元します。

トリガー手順の復元

Unica Plan アプリケーションをアップグレードした後で、トリガー手順を復元できます。

このタスクについて

トリガー手順を復元するには、以下の手順を完了します。

1. 以前に作成したバックアップから、手順と `procedure_plugins.xml` ファイルを復元します。それらをファイル用の以下のデフォルト・ロケーションに入れます。

   `<Plan_Home>\devkits\integration\examples\src\procedure`

2. 必要な場合は、Unica Plan インストール済み環境下の `<Plan_Home>\devkits\integration\examples\build` ディレクトリーにある `build` ファイルを使用して、統合サービス手順を再ビルドします。
3. **「設定」>「構成」>「Plan」>「umoConfiguration」>「attachmentFolders」**ページで、以下のパラメーターを更新します。前のステップで作成したディレクトリーを指すように、値を設定します。
   - **graphicalRefUploadDir** を次に設定 `<Plan_Home>\graphicalrefimages`
   - **templateImageDir** を次に設定 `<Plan_Home>\images`
   - **recentDataDir** を次に設定 `<Plan_Home>\recentdata`
   - **workingAreaDir** を次に設定 `<Plan_Home>\umotemp`

クラスター環境での Unica Plan のアップグレード

クラスター環境で Unica Plan の複数のインスタンスをアップグレードする場合には、以下のガイドラインを使用してください。

- Unica Plan のすべてのインスタンスを配置解除します。
- この章の指示に従ってアップグレードします。
- ご使用の Web アプリケーション・サーバーの自動配置機能を使用して、クラスター内の EAR ファイルを配置します。

クラスター環境に インストールUnica Planする場合の考慮事項については、*Unica Plan*「インストールガイド」を参照してください。

Note: すべての plan_ehcache.xml ファイルを確認または更新する必要があります。

一意の名前が存在しない場合は、これを指定する必要があります (例: "`<ehcache updateCheck="false" name="PlanEhCacheManager">`

第 4 章. 概要

Unica Plan を WebSphere® および WebLogic に配置する際の一般ガイドラインがあります。

インストーラーを実行した後に EAR ファイルを作成して他の製品をその EAR ファイルに含めた場合は、この章に記載されているガイドラインに従うほか、EAR ファイルに含めた製品の個々のインストール・ガイドに記載されているすべての配置ガイドラインに従う必要があります。

ここでは、読者が Web アプリケーション・サーバーの使用方法を理解しているものと想定しています。「管理」コンソールの使用方法などに関する詳細は、Web アプリケーション・サーバーの資料を参照してください。

Unica 製品の JVM パラメーターの構成

Unica 製品の以下の JVM パラメーターを構成する必要があります。

表 8. 構成する Unica 製品と JVM パラメーター

Unica 製品名	JVM パラメーター
Unica Campaign	• `-DUNICA_PLATFORM_CACHE_ENABLED=true` • `-Dcampaign.log4j.async=true` • `-Dcom.sun.management.jmxremote` `-Dcom.sun.management.jmxremote` • `-Dcom.sun.management.jmxremote.port=1007` • `-Dcom.sun.management.jmxremote.ssl=false` • `-Dcom.sun.management.jmxremote.authenticate=false` • `-Dcom.sun.management.jmxremote.password.file=../conf/jmxremote.password` • `-Dcom.sun.management.jmxremote.access.file=../conf/jmxremote.access` • `-DUNICA_GOTO_CREATEEARFILE=TRUE` • `-Dfile.encoding=UTF-8` • `-Dclient.encoding.override=UTF-8` • `-DUNICA_GOTO_CREATEEARFILE=TRUE` • `-Dclient.encoding.override=UTF-8` • `-Djboss.as.management.blocking.timeout=3600` SET SSL_OPTIONS=-Djavax.net.ssl.keyStoreType="JKS"

表 8. 構成する Unica 製品と JVM パラメーター (続く)

Unica 製品名	JVM パラメーター

	• `-Djavax.net.ssl.trustStore="C:\security\myTrustStore.jks"`
	• `-Djavax.net.ssl.trustStorePassword=myPassword`
	
	`ACOOptAdmin` ツール (`ACOOptAdmin.sh` (UNIX) または `ACOOptAdmin.bat` (Windows) ファイル) の構成
	
	• `-Djavax.net.ssl.keyStoreType=keyStoreTypeValue`
	• `-Djavax.net.ssl.keyStore=<keyStoreValue>`
	• `-Djavax.net.ssl.keyStorePassword=<keyStorePasswordValue>`
	• `-Djavax.net.ssl.trustStore=<trustStoreValue>`
	• `-Djavax.net.ssl.trustStorePassword=<trustStorePasswordValue>`
	• `-Djava.awt.headless=true`
	• `-Dcampaign.deliveretl.disabled=true`
	• `-Dcampaign.interactetl.disabled=true`
	• `-Dcampaign.journeyetl.disabled=false`
	• `-Dcampaign.journeyetl.disabled=true`
	
	`EMAIL_TLS_ENABLED`
	
	`EMAIL_SSL_TRUST_ENABLED`
	
	`SCH_ASYNCH_EXECUTION_ENABLED`

Unica Audience Central	• `-DAUDIENCE_CENTRAL_HOME=<Install_Base_DIR>/AudCentral`
Unica Contact Central	• `CONTACTCENTRAL_HOME` • `UNICA_PLATFORM_HOME`
Unicaコンテンツの統合	• `-DASSET_PICKER_HOME` • `-DUNICA_PLATFORM_HOME`

表 8. 構成する Unica 製品と JVM パラメーター (続く)

Unica 製品名	JVM パラメーター
Unica Deliver	• `-Dcampaign.deliveretl.disabled=true`
Unica Interact	**Interact 設計時間** 以下に Interact 設計時間の JVM パラメーターを示します。 • `INTERACTDT_HOME` • `Interact.UsernameToAlwaysDeployFor` • `com.unicacorp.interact.deliver.templateTimeout` • `DeploymentServletParameterEncoding` • `com.unicacorp.interact.flexoffers.defaultDateTimeFormat` • `com.unicacorp.interact.flexoffers.defaultDateFormat` • `com.unicacorp.Campaign.interact.offermapping.batchsize` • `com.unicacorp.Campaign.interact.offermapping.service.synctimeout` • `com.unicacorp.interact.cacheTTL` • `com.unicacorp.interact.cacheRefreshIntervalInMin` • `com.unicacorp.interact.enableDTPerfLogging` • `com.unicacorp.interact.compressAPIResponse` • `ignoreSpecialCharacterValidator` • `Interact.CustomStringDelimiter` • `com.unicacorp.interact.playback.maxFilteredIdsCount` • `com.unicacorp.interact.playback.APITimeoutInSecs` **Interact 実行時間** 以下に Interact 実行時間の JVM パラメーターを示します。 • `INTERACT_HOME` • `com.unicacorp.interact.deliver.templateTimeout` • `INBOUND_GATEWAYS_USING_MAPPING_FROM_PROPERTIES` • `com.hcl.interact.http.proxyProtocol` • `com.hcl.interact.http.proxyHost` • `com.hcl.interact.http.proxyPort` • `com.hcl.interact.http.proxyUsername` • `com.hcl.interact.http.proxyPassword` • `interact.jmx.monitoring.port` • `interact.offerserving.maxOfferAllocationInMemoryPerInstance`

表 8. 構成する Unica 製品と JVM パラメーター (続く)

Unica 製品名	JVM パラメーター
	• `interact.offerserving.maxDistributionPerIntervalPerInstanceFactor` • `interact.ignitePort` • `com.unicacorp.interact.chDupeCheckLimit` • `com.unicacorp.interact.rhDupeCheckLimit` • `com.unicacorp.interact.chSuppressDupe` • `com.unicacorp.interact.rhSuppressDupe` • `com.unicacorp.interact.testclient.nullValue` • `interact.ehcache.config` • `interact.api.dateFormat` • `com.hcl.interact.testrun.rowlimit` • `Interact.DisableExceptionStackTracesInMacros` • `com.unicacorp.interact.enableDetailStats` • `com.unica.interact.deployment.timeoutInSecs` • `com.ibm.interact.instance.name` • `com.unicacorp.interact.invalidPaths` • `interact.XSessResponseConsumerManager.generateOnlyOneResponse` • `tryToPreserveInexactFloatValues` • `com.unicacorp.interact.propertyRefreshInterval` • `com.unicacorp.interact.scheduledTasksProcessInterval` • `com.unicacorp.interact.eventpatterns.parallelism` • `com.unicacorp.interact.eventpatterns.restartRetries` • `com.unicacorp.interact.eventpatterns.evaluateTimeoutMilli` • `com.unicacorp.interact.eventpatterns.restartRetryDelayInSec` • `Interact.advisoryMessageEncodingOverrides` • `com.unica.interact.api.insertSessionIDAsCookie` • `com.unica.interact.api.SessionIDCookieName` • `InteractMsgCode` • `com.ibm.interact.triggeredmessage.enableJMSConsumer` • `com.unicacorp.interact.maxStringLengthInFormatMacro` • `ContinueEvaluatingBranchAndAdvOptTreatmentLogicDespiteExMessageList` • `DisableDecisionProcessBoxAndAdvOptTreatmentLogging` 有効な値は `True` または `False` です。デフォルト値は `False` です。値を `False` に設定した場合、各ブランチの式を評価

表 8. 構成する Unica 製品と JVM パラメーター (続く)

Unica 製品名	JVM パラメーター
	するとき、システムでエラーが発生すると、エラーをログに記録します。 • `Interact.DPB.MuffleErrorsForMutuallyExclusiveBranchesWithNullValues` 有効な値は `True` と `False` です。デフォルト値は `False` です。値を `False` に設定し、`-DDisableDecisionProcessBoxAndAdvOptTreatmentLogging` の値も `False` に設定されている場合、決定プロセス・ボックスの各ブランチの式の評価中、このブランチは相互に排他的であるため、システムでエラーが発生すると、エラーをログに記録します。それ以外の場合、システムはエラーをデバッグとしてログに記録します。 • `TwoDigitYearStartDate` • `Interact.enableTwoDigitYearFix` • `com.ibm.interact.evpatetl.conf` • `com.unicacorp.interact.minTreatmentsPerThread` • `com.unicacorp.interact.maxTreatmentPoolSize` • `CircuitBreaker.processTimeoutMillis` • `com.unicacorp.interact.event.asyncTimeoutMSec` • `com.unicacorp.interact.eventActionTimeout` • `Interact.HTMI.Enabled` • `Interact.HTMI.MaxRequestDurationInMs` • `Interact.HTMI.RecordIndividualAPIs` • `Interact.HTMI.MaxStartSessionDurationInMs` • `Interact.HTMI.MaxGetOffersDurationInMs` • `Interact.HTMI.MaxPostEventDurationInMs` • `Interact.HTMI.MaxGetProfileDurationInMs` • `Interact.HTMI.LogErrorsEveryNthTime` • `Interact.HTMI.UseMillisecondTimers` • `Interact.HTMI.Debug` • `com.unicacorp.interact.suppressWarningOnAnonymousUser` • `com.hcl.interact.eventpatterns.printPatternAction` • `com.hcl.interact.eventpatterns.eagerPersist` • `com.ibm.interact.triggeredmessage.addPerfData` • `com.unicacorp.interact.learning.disableAggregator` • `com.unicacorp.interact.learning.disableDeletion` • `com.unicacorp.interact.learning.ignoreInterval`

表 8. 構成する Unica 製品と JVM パラメーター (続く)

Unica 製品名	JVM パラメーター
	• `interact.services.loader.saveLoaderFiles` • `ConvertEveryNULLAttributeValueToAJEPNullConstant` • `includeJoinInfo` • `com.unicacorp.interact.deployment.reloadTimeout` • `com.ibm.interact.lockTimeWarningThreshold` • `com.unicacorp.interact.cache.maxWaitTime` • `DEFAULT_PERSISTENCE_PROVIDER` • `com.unicacorp.interact.playback.purgeBatchSize` **注:** JVM パラメーター `interact.runtime.instance.name` を削除しました。JVM パラメーター `com.ibm.interact.instance.name` を使用して Interact 実行時間インスタンスの名前を設定します。
Unica Journey	
Unica Plan	• `-Dplan.log4j.config=<plan_home>/conf/plan_log4j.xml` • `-Dplan.log4j.async=true` • `-Dplan.home=<plan_home>` • `-Dplan.slow.query.threshold=<value in milliseconds>` (オプション) • `-Dplan.slow.query.show.stack.trace=true` (オプション) • `-Dcampaign.https.port=<campaign's https port>` Plan が `https` を介して Campaign と通信する場合に、Campaign が 8443 以外の `https` ポートで実行するように構成されている場合は、このシステムプロパティーを Plan アプリケーションの JVM の JVM 引数に追加します。
Unica Platform	• `-DENABLE_PLATFORM_LOG4J_XML_LOGGING=TRUE` • `-DENABLE_PERSISTENT_NAMEID_FORMAT=TRUE` • `-DUNICA_PLATFORM_HOME=<platform_home_directory_path>`
Unica 一元化されたオファー管理	• `-DOFFER_HOME=<Install_Base_DIR>/CentralizedOffer` • `-DUNICA_PLATFORM_CACHE_ENABLED=false`

表 8. 構成する Unica 製品と JVM パラメーター (続く)

Unica 製品名	JVM パラメーター
	• `-DUNICA_PLATFORM_LOCAL_CACHE_ENABLED=false` • `-DOffer.isDebug=false` • `-Dlog4j2.formatMsgNoLookups=true`
Unica Segment Central	• `-DSEGMENT_CENTRAL_HOME=<Install_Base_DIR>/SegCentral`

Websphere での Unica Plan の配置

WebSphere® Application Server (WAS) に、WAR ファイルまたは EAR ファイルから Unica Plan アプリケーションを配置できます。

開始する前に

Websphere に Unica Plan を配置する前に以下の点を考慮してください。

- ご使用の WebSphere® のバージョンが、必要なフィックス・パックまたはアップグレードも含めて、「推奨されるソフトウェア環境および最小システム要件」の資料に記載された要件を満たしていることを確認します。
- WebSphere® Integrated Solutions コンソールを使用して、WebSphere® Application Server を構成します。以下のステップでは、個々の制御を設定するためのガイドラインを示します。
- WAS サーバーの lib ディレクトリーに `javax.el-3.0.1-b11.jar` を追加していることを確認してください。`javax.el-3.0.1-b11.jar` を https://mvnrepository.com/artifact/org.glassfish/javax.el/3.0.1-b11 からダウンロードできます。

注: WebSphere® Application Server のバージョンによって、ユーザー・インターフェース制御が表示される順序が異なり、別のラベルが使用されていることもあります。

このタスクについて

以下の手順を実行して Unica Plan の配置のための環境をセットアップします。

1. カスタム・プロパティーを定義します。**「アプリケーション・サーバー」** > **「<servers>」** > **「Web コンテナー」** > **「カスタム・プロパティー」**フォームで、**「新規」**をクリックして以下の値を入力します。
   - **Name**:`com.ibm.ws.webcontainer.invokefilterscompatibility`
   - **Value**: `true`
2. JDBC プロバイダーを作成します。**「リソース」** > **「JDBC」** > **「JDBC プロバイダー」**フォームで、**「新規」**をクリックします。以下のフィールドも含めて、**「新規 JDBC プロバイダーの作成」**ウィザードを完了します。

   インストーラーを使用して構成する場合、Web アプリケーション・サーバーでのデータ・ソースの作成を省略できます。

a. **「実装タイプ」**で**「接続プール」**データ・ソースを選択します。

b. サーバー上のデータベース・ドライバー JAR ファイルのネイティブ・ライブラリー・パスを指定します。
例
例: *`db2jcc4.jar / ojdbc6.jar / sqljdbc4.jar / mariadb-java-client-2.4.1 / onedb-jdbc-8.0.0.1-complete.jar` / postgresql-42.5.4.jar*。

注: PostgreSQL サポートは、12.1.7 からアップグレードしている場合のみ使用可能です。

c. MariaDB の場合は、実装クラス名を `org.mariadb.jdbc.MariaDbDataSource` として入力します

3. データ・ソースを作成します。**「リソース」** > **「JDBC」** > **「データ・ソース」**フォームで、**「新規」**をクリックします。以下の操作を実行して、データ・ソースの作成ウィザードを完了します。

インストーラーを使用して構成する場合、Web アプリケーション・サーバーでのデータ・ソースの作成を省略できます。

a. データ・ソース名を指定します。
b. 「JNDI 名」に `plands` と入力します。
c. ステップ 2 で作成した JDBC プロバイダーを選択します。
d. データベース名およびサーバー名を指定します。
e. **「マッピング構成」**別名で **WSLogin** を選択します。

4. データ・ソースのカスタム・プロパティーを定義します。**「JDBC プロバイダー」** > **「<database provider>」** > **「データ・ソース」** > **「カスタム・プロパティー」**フォームで、**「新規」**をクリックして、以下の 2 つのプロパティーを追加します。

- **Name**: `user`
- **Value**: <user_name>
- **Name**: `password`
- **Value**: <password>

Unica Plan システム・テーブルが DB2® 内にある場合は、`resultSetHoldability` プロパティーを見つけ、その値を `1` に設定します。このプロパティーが存在しない場合は、追加してください。

5. MariaDB データ・ソースのカスタム・プロパティーを作成します。
選択
- ポート: `<portnumber>`
- databaseName: `<database name>`
- serverName: `<server name>`
- driverType : 6

6. JVM を構成します。**「アプリケーション・サーバー」** > **「<server>」** > **「プロセス定義」** > **「Java 仮想マシン」**フォームで、「クラスパス」を見つけ、以下の項目をスペースで区切って「汎用 JVM 引数」として追加します。

- `-Dplan.home=<installation_directory>\<Plan_Home>`

  ここで、*<Plan_Home>* は Unica Plan をインストールしたディレクトリーへのパスです。通常、このパスは `HCL_Unica/Plan` です。

- `-Dclient.encoding.override = UTF-8`
- 次の JVM パラメーターが存在することを確認してください。存在しない場合、追加します。
  - バージョン 12.1.0、12.1.0.1、12.1.0.2、および 12.1.0.3 では、以下のパラメーターを追加します。
    - `-Dlog4j2.contextSelector=org.apache.logging.log4j.core.async.AsyncLoggerContextSelector`
    - `-Dlog4j2.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
    - `-Dlog4j.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
  - バージョン 12.1.0.4 以降では、インストールする場合は、以下のパラメーターを追加します。バージョン 12.1.0 以降からアップグレードする場合は、前述 ページ 30のパラメーターを削除し、以下のパラメーターを追加します。
    - `-Dlog4j.configuration=file:///<Plan_home>\conf\plan_log4j_1x.xml`
    - `-Dplan.log4j.config=<Plan_home>\conf\plan_log4j.xml`
    - `-Dplan.log4j.async=true (If not set, deafults to true)`

注:

- バージョン 12.1.0、12.1.0.1、12.1.0.2、12.1.0.3 では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を `-Dlog4j2.configurationFile` に従ってここで指定されているように行い、`-Dlog4j.configurationFile` が配置されていることを確認してください。
- バージョン 12.1.0.4 以降では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を、配置されている `-Dplan.log4j.config` ファイルに従って、ここで指定されているように行ってください。
- バージョン 12.1.0、12.1.0.1、12.1.0.2、12.1.0.3 では、`log4j2.configurationFile` と `log4j.configurationFile` にポイントされている `xml` ファイルが同じで、`log4j2` 構文に従っていることを確認してください。ファイルの先頭には、以下の 2 行が含まれている必要があります。

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration packages="com.unica.afc.logger" monitorInterval="60">
```

- バージョン 12.1.0.4 以降では、`-Dplan.log4j.config` ファイルにポイントされる `xml` ファイルが `log4j2` 構文に従っていることを確認してください。ファイルの先頭には、以下の 2 行が含まれている必要があります。

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration packages="com.unica.afc.logger" monitorInterval="60">
```

- 詳細については、次のリンクを参照しながら log4j2 の資料を参照してください。 https://logging.apache.org/log4j/2.0/manual/configuration.html

- Windows では、パスにバックスラッシュが含まれます。それに対し、UNIX ではスラッシュを含む必要があります。
- log4j 構成ファイル名のパス区切り記号は、基盤となるオペレーティング・システムに基づき、Windows スタイル () または UNIX (/) スタイルで設定される必要があります。

7. `<Plan_home>/conf/` の場所で、12.1.8 インストーラーは既存のファイルのバックアップを作成します。

バックアップ前のファイル名	バックアップ後のファイル名
`plan_log4j.xml`	`plan_log4j.xml_pre_12.1.8`
`plan_log4j_client.xml`	`plan_log4j_client.xml_pre_12.1.8`
`plan_log4j_socket_server.xml`	`plan_log4j_socket_server.xml_pre_12.1.8`

バックアップ・ファイルにカスタム設定が含まれる場合、そのカスタム設定が 12.1.8 インストーラーにより新しく追加された構成ファイルにコピーされていることを確認する必要があります。

8. WebSphere Application Server の JSP コンパイル・レベルを 18 に設定します。

WAR または EAR ファイルの配置

新規エンタープライズ・アプリケーションを配置する場合、WebSphere® Integrated Solutions Console に一連のフォームが表示されます。以下のステップでは、それらのフォームで個々の制御を設定するためのガイドラインを示します。WebSphere® のバージョンによって、制御が表示される順序が異なる可能性があります。また、別のラベルが使用されている場合もあります。

このタスクについて

以下の手順を実行して、WAR または EAR ファイルを配置します。

1. **「アプリケーション」** > **「新規アプリケーション」** > **「新規エンタープライズ・アプリケーション」**を選択します。
2. 初期フォームで、**「リモート」**ファイルシステムを選択してから、**「参照」**で `plan.war` ファイルまたは EAR ファイルを指定します。
3. 次の**「アプリケーション・インストールの準備」**ウィンドウで、以下のようにします。
   - **「詳細」**を選択します。
   - **「デフォルト・バインディングの生成」**を選択します。
   - **「既存バインディングをオーバーライドする」**を選択します。
4. **「インストール・オプションの選択」**ウィンドウで以下の操作を完了します。
   - **「JavaServer Pages ファイルのプリコンパイル」**を選択します。
   - **「アプリケーション名」**に `plan` と入力します。
   - **「Web および EJB モジュールのクラス再ロード設定をオーバーライドする」**を選択します。
   - **「再ロード間隔 (秒)」**では、`4` などの整数を入力します。
5. **「サーバーにモジュールをマップ」**ウィンドウで、**「モジュール」**を選択します。EAR を配置した場合は、すべての WAR ファイルを選択してください。

6. **「JSP をコンパイルするためのオプションを指定」**ウィンドウで、**「Web モジュール」**を選択します。EAR を配置した場合は、すべての WAR ファイルを選択してください。
7. **「JDK ソース・レベル」**を `18` に設定します。
8. **「Web モジュールの JSP 再ロード・オプション」**フォームで、**「JSP: クラスの再ロードを有効にする」**を選択し、**「JSP: 再ロード間隔 (秒)」**に `5` と入力します。
9. **「共有ライブラリーをマップ」**ウィンドウで、**「アプリケーション」**および**「モジュール」**を選択します。
10. **「共有ライブラリーの関係をマップ」**ウィンドウで、**「アプリケーション」**および**「モジュール」**を選択します。
11. **「リソース参照をリソースにマップ」**ウィンドウでモジュールを選択し、**「ターゲット・リソース JNDI 名」**に `plands` と入力します。
12. **「Web モジュールのコンテキスト・ルートをマップ」**ウィドウで、**「コンテキスト・ルート」**に `/plan` と入力します。
13. 設定を確認して保存します。

クラス・ローダー・ポリシーの定義

クラス・ローダー・ポリシーは、WAS でアプリケーションを構成する方法を定義します。Unica Plan を配置する前に WAS のデフォルトの設定をいくつか変更する必要があります。

このタスクについて

以下の手順を完了して、クラス・ローダー・ポリシーを定義します。

1. **「エンタープライズ・アプリケーション」** > **「plan」** > **「クラス・ローダー」**で、**「Web および EJB モジュールのクラス再ロード設定をオーバーライドする」**を選択します。
2. **「クラス・ローダー順序」**では、**「最初にローカル・クラス・ローダーをロードしたクラス (親は最後)」**を選択します。
3. **「WAR クラス・ローダー・ポリシー」**では、**「アプリケーションの単一クラス・ローダー」**を選択します。
4. **「適用」**および**「設定の保存」**をクリックします。

Cookie の設定の定義

「Websphere エンタープライズ・アプリケーション」の**「セッション管理」**オプションを使用し、Cookie の設定を定義してセットする必要があります。

このタスクについて

以下の手順を完了して、Cookie の設定を定義します。

1. **「エンタープライズ・アプリケーション」** > **「*plan*」** > **「セッション管理」**へ移動します。
2. **「セッション管理のオーバーライド」**を選択します。
3. **「Cookie を使用可能にする」**を選択します。
4. **「適用」**をクリックして、**「エンタープライズ・アプリケーション」** > ***plan*** > **「セッション管理」** > **「Cookie」**に移動します。
5. Unica Plan **「Cookie 名」**を `JSESSIONID` から `UMOSESSIONID` に変更します。
6. **「適用」**および**「設定の保存」**をクリックします。

EAR モジュール設定の定義

EAR ファイルを配置した場合は、EAR ファイルに含まれている個々の WAR ファイルの設定を定義する必要があります。

このタスクについて

以下の手順を完了して、EAR ファイル・モジュールの設定を定義します。

1. **「エンタープライズ・アプリケーション」**に移動して、EAR ファイルを選択します。
2. **「モジュールの管理」**フォームで、WAR ファイルの 1 つ (例えば、`MktOps.war`) を選択します。
3. **「エンタープライズ・アプリケーション」 > 「*EAR*」 > 「モジュールの管理」 > 「WAR」**フォームで以下の手順を実行します。
   a. **「開始ウェイト」**を `10000` に設定します。
   b. **「クラス・ローダー順序」**では、**「最初にアプリケーション・クラス・ローダーをロードしたクラス」**を選択します。
4. **「エンタープライズ・アプリケーション」 > 「EAR」 > 「モジュールの管理」 > 「WAR」 > 「セッション管理」**で、**「Cookie を使用可能にする」**を選択します。
5. **「エンタープライズ・アプリケーション」 > 「EAR」 > 「モジュールの管理」 > 「WAR」 > 「セッション管理」 > 「Cookie」**で以下の手順を実行します。
   a. **「Cookie 名」**を `CMPJSESSIONID` に設定します。
   b. **「Cookie 最大存続期間」**では、**「現行のブラウザー・セッション」**を選択します。
6. **「エンタープライズ・アプリケーション」 > 「EAR」 > 「モジュールの管理」 > 「WAR」 > 「セッション管理」**で以下の情報を入力します。
   a. **「オーバーフローの許可」**を選択します。
   b. **「メモリー内の最大セッション・カウント」**に `1000` と入力します。
   c. **「セッション・タイムアウト」**で**「タイムアウトの設定」**を選択し、`30` と入力します。
7. 他の WAR ファイル (`unica.war` や `plan.war` など) のそれぞれについても同じ設定を定義します。

注: `Campaign.war` ファイルが EAR ファイル内にも存在し、Unica Plan と Unica Campaign とを統合する計画の場合、`Campaign.war` ファイルに対して同じ設定を定義してください。

WebLogic 上での Unica Plan のデプロイ

WebLogic での Unica Plan のデプロイについては、以下のガイドラインを使用してください。

このタスクについて

- Unica 製品により、WebLogic で使用される JVM がカスタマイズされます。JVM 関連のエラーが発生した場合に、Unica 製品専用の WebLogic インスタンスを作成することができます。
- 同一の WebLogic ドメインに複数の Unica Plan アプリケーションをインストールしないでください。
- 始動スクリプト (`startWebLogic.cmd`) で JAVA_VENDOR 変数を調べて、使用する WebLogic ドメイン用に選択された Software Developement Kit (SDK) が Sun SDK であることを確認します。その変数は、`JAVA_VENDOR=Sun` に設定されている必要があります。それが `JAVA_VENDOR=BEA` に設定されている場合、JRockit が選択されています。JRockit はサポートされていません。選択された SDK を変更するには、WebLogic の資料を参照してください。

WebLogic へ Unica Plan を配置するには以下の手順を実行します。

1. ご使用のオペレーティング・システムが AIX® である場合は、Unica Plan `WAR` ファイルを解凍し、`xercesImpl.jar` ファイルを `WEB_INF/lib` ディレクトリーから削除して、`WAR` ファイルを再作成します。
   インストーラーによって製品が `EAR` ファイルにまとめられている場合は、まず、そのファイルを解凍して `WAR` ファイルを取得してから、`EAR` ファイルを再作成する必要があります。
2. IIS プラグインを使用するように WebLogic を構成する場合は、WebLogic の資料を見直して、他にも要件があるかどうかを確認します。
3. WebLogic ドメイン・ディレクトリーの下の `bin` ディレクトリーで、`setDomainEnv` スクリプトを見つけ、テキスト・エディターで開きます。スクロールして `JAVA_OPTIONS` プロパティーを表示し、次の項目を追加します。項目を区切るにはスペースを使用します。

   - `-Dplan.home=<HCL_Unica_Home>\<Plan_Home>`

     ここで、 *`<HCL_Unica_Home>`* は最上位のディレクトリーへのパスであり、*`<Plan_Home>`* は Unica Plan がインストールされているディレクトリーへのパスです。通常、このディレクトリーは `HCL_Unica/Plan` です。

   - `-Dfile.encoding=UTF-8`
   - 次の JVM パラメーターが存在することを確認してください。存在しない場合、追加します。
     - バージョン 12.1.0、12.1.0.1、12.1.0.2、および 12.1.0.3 では、以下のパラメーターを追加します。
       - `-Dlog4j2.contextSelector=org.apache.logging.log4j.core.async.AsyncLoggerContextSelector`
       - `-Dlog4j2.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
       - `-Dlog4j.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
     - バージョン 12.1.0.4 以降では、インストールする場合は、以下のパラメーターを追加します。バージョン 12.1.0 以降からアップグレードする場合は、前述 ページ 34のパラメーターを削除し、以下のパラメーターを追加します。
       - `-Dlog4j.configuration=file:///<Plan_home>\conf\plan_log4j_1x.xml`
       - `-Dplan.log4j.config=<Plan_home>\conf\plan_log4j.xml`
       - `-Dplan.log4j.async=true (If not set, deafults to true)`

注:

- バージョン 12.1.0、12.1.0.1、12.1.0.2、12.1.0.3 では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を `-Dlog4j2.configurationFile` に従ってここで指定されているように行い、`-Dlog4j.configurationFile` が配置されていることを確認してください。
- バージョン 12.1.0.4 以降では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を、配置されている `-Dplan.log4j.config` ファイルに従って、ここで指定されているように行ってください。
- バージョン 12.1.0、12.1.0.1、12.1.0.2、12.1.0.3 では、`log4j2.configurationFile` と `log4j.configurationFile` にポイントされている `xml` ファイルが同じで、`log4j2` 構文

に従っていることを確認してください。ファイルの先頭には、以下の 2 行が含まれている必要があります。

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration packages="com.unica.afc.logger" monitorInterval="60">
```

- バージョン 12.1.0.4 以降では、`-Dplan.log4j.config` ファイルにポイントされる `xml` ファイルが `log4j2` 構文に従っていることを確認してください。ファイルの先頭には、以下の 2 行が含まれている必要があります。

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration packages="com.unica.afc.logger" monitorInterval="60">
```

- 詳細については、次のリンクを参照しながら log4j2 の資料を参照してください。 https://logging.apache.org/log4j/2.0/manual/configuration.html
- Windows では、パスにバックスラッシュが含まれます。それに対し、UNIX ではスラッシュを含む必要があります。
- log4j 構成ファイル名のパス区切り記号は、基盤となるオペレーティング・システムに基づき、Windows スタイル () または UNIX (/) スタイルで設定される必要があります。

注:

- Windows では、パスにバックスラッシュが含まれます。それに対し、UNIX ではスラッシュを含む必要があります。
- log4j 構成ファイル名のパス区切り記号は、基盤となるオペレーティング・システムに基づき、Windows スタイル () または UNIX (/) スタイルで設定される必要があります。

4. `<Plan_home>/conf/` の場所で、12.1.8 インストーラーは既存のファイルのバックアップを作成します。

バックアップ前のファイル名	バックアップ後のファイル名
`plan_log4j.xml`	`plan_log4j.xml_pre_12.1.8`
`plan_log4j_client.xml`	`plan_log4j_client.xml_pre_12.1.8`
`plan_log4j_socket_server.xml`	`plan_log4j_socket_server.xml_pre_12.1.8`

バックアップ・ファイルにカスタム設定が含まれる場合、そのカスタム設定が 12.1.8 インストーラーにより新しく追加された構成ファイルにコピーされていることを確認する必要があります。

5. ファイルを保存して閉じます。
6. WebLogic を再始動します。
7. Unica Plan を Web アプリケーション・モジュールとして配置します。`plan.war` を選択します。
8. 配置した Web アプリケーションを開始します。

次にやるべきこと

WebLogic 12.2.1.3 は、少数の HTTP の方法、例えば、DELETE や PATCH をブロックします。これらの方法は、REST API により新しく追加された拡張ワークフローに使用されます。この問題を解決するには、WebLogic パッチを適用します (適用されていない場合)。

パッチの詳細については、https://support.oracle.com/knowledge/Middleware/2331453_1.html にアクセスしてください。リンクにアクセスするには、登録ユーザーである必要があります。認証に成功した後、パッチ番号 `26923558` を検索します。

注: パッチに関して提供される情報はサードパーティーの Web サイトに属します。情報が不正確な場合でも HCL Unica は責任を負いません。

JBoss への Unica Plan の配置

JBoss に Unica Plan を配置する際には、一連のガイドラインに従う必要があります。

1. 実行する手順
    - すべてのユーザーの最新項目リストを消去します。ViewState に情報を格納するには: 管理者: JBOSS に移行する前に、それぞれの最新項目を消去するすべてのユーザーに通知してください。
    - ディレクトリー `<Plan_home>/recentdata` をクリアします。
    - JBOSS + 12.1.8 に移行します。

JBoss インストール・ディレクトリーの下の `bin` ディレクトリーから、ご使用のオペレーティング・システムに応じた `standalone.conf` または `standalone.conf.bat` スクリプトを見つけ、テキスト・エディターで開きます。スクロールして `JAVA_OPTIONS` プロパティーを表示し、次の項目を追加します。項目を区切るにはスペースを使用します。

- `-Dplan.home=<Plan_Home>`

  ここで、*<HCL_Unica_Home>* は最上位の ディレクトリーへのパスであり、*<Plan_Home>* は Unica Plan がインストールされているディレクトリーへのパスです。通常、このディレクトリーは `HCL_Unica/Plan` です。

- `-Dfile.encoding=UTF-8`
- 次の JVM パラメーターが存在することを確認してください。存在しない場合、追加します。
    - バージョン 12.1.0、12.1.0.1、12.1.0.2、および 12.1.0.3 では、以下のパラメーターを追加します。
        - `-Dlog4j2.contextSelector=org.apache.logging.log4j.core.async.AsyncLoggerContextSelector`
        - `-Dlog4j2.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
        - `-Dlog4j.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
    - バージョン 12.1.0.4 以降では、インストールする場合は、以下のパラメーターを追加します。バージョン 12.1.0 以降からアップグレードする場合は、前述 ページ 36のパラメーターを削除し、以下のパラメーターを追加します。
        - `-Dlog4j.configuration=file:///<Plan_home>\conf\plan_log4j_1x.xml`
        - `-Dplan.log4j.config=<Plan_home>\conf\plan_log4j.xml`
        - `-Dplan.log4j.async=true (If not set, deafults to true)`

注:

- バージョン 12.1.0、12.1.0.1、12.1.0.2、12.1.0.3 では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を `-Dlog4j2.configurationFile` に従ってここで指定されているように行い、`-Dlog4j.configurationFile` が配置されていることを確認してください。
- バージョン 12.1.0.4 以降では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を、配置されている `-Dplan.log4j.config` ファイルに従って、ここで指定されているように行ってください。
- バージョン 12.1.0、12.1.0.1、12.1.0.2、12.1.0.3 では、`log4j2.configurationFile` と `log4j.configurationFile` にポイントされている `xml` ファイルが同じで、`log4j2` 構文に従っていることを確認してください。ファイルの先頭には、以下の 2 行が含まれている必要があります。

```
 <?xml version="1.0" encoding="UTF-8"?>
<Configuration packages="com.unica.afc.logger" monitorInterval="60">
```

- バージョン 12.1.0.4 以降では、`-Dplan.log4j.config` ファイルにポイントされる `xml` ファイルが `log4j2` 構文に従っていることを確認してください。ファイルの先頭には、以下の 2 行が含まれている必要があります。

```
 <?xml version="1.0" encoding="UTF-8"?>
<Configuration packages="com.unica.afc.logger" monitorInterval="60">
```

- 詳細については、次のリンクを参照しながら log4j2 の資料を参照してください。 https://logging.apache.org/log4j/2.0/manual/configuration.html
- Windows では、パスにバックスラッシュが含まれます。それに対し、UNIX ではスラッシュを含む必要があります。
- log4j 構成ファイル名のパス区切り記号は、基盤となるオペレーティング・システムに基づき、Windows スタイル () または UNIX (/) スタイルで設定される必要があります。

- `<Plan_home>/conf/` の場所で、12.1.8 インストーラーは既存のファイルのバックアップを作成します。

バックアップ前のファイル名	バックアップ後のファイル名
`plan_log4j.xml`	`plan_log4j.xml_pre_12.1.8`
`plan_log4j_client.xml`	`plan_log4j_client.xml_pre_12.1.8`
`plan_log4j_socket_server.xml`	`plan_log4j_socket_server.xml_pre_12.1.8`

バックアップ・ファイルにカスタム設定が含まれる場合、そのカスタム設定が 12.1.8 インストーラーにより新しく追加された構成ファイルにコピーされていることを確認する必要があります。

JBoss のバージョンが、Unica Enterprise Products Recommended Software Environments and Minimum System Requirements ドキュメントに記載されている要件を満たしていることを確認してください.Jboss に Unica Plan を配置する場合には、以下のガイドラインに従ってください。

サポートされるバージョンの JBoss に Unica Plan 製品を配置する場合には、以下のガイドラインに従ってください。

1. HCL plan.war ファイルを、エンタープライズ・アプリケーションとして配置します。

   例: 配置する `<Plan_home>\plan.war`

   Web Server Application の JBoss への配置の手順については、https://docs.jboss.org/jbossweb/3.0.x/deployer-howto.html を参照してください。

2. インストール済み環境で非 ASCII 文字をサポートする必要がある場合 (例えば、ポルトガル語や、マルチバイト文字を必要とするロケール) は、以下のタスクを実行してください。
   a. JBoss /bin ディレクトリーの下の bin ディレクトリーにある `standalone.conf` スクリプトを編集して、`-Dfile.encoding=UTF-8 -Dclient.encoding.override=UTF-8 to JAVA_VENDOR` を追加します。
   b. JBOSSサーバーを再起動します。

デフォルトでは、JBOSS は最大サイズ 10 MB の要求を処理するように構成されています。ユーザーが大きなファイルをダウンロードすると予想される場合は、必要に応じて **max-post-size** を構成します。

このパラメーターの値を増やすには、次のように独立した構成またはドメイン名構成を直接編集します。

```
<buffer-cache name="default"/>
           <server name="default-server">
               <http-listener name="default" socket-binding="http" redirect-socket="https"
max-post-size="20971520" enable-http2="true"/>
```

詳細については、*JBOSS* 資料を参照してください。

Unica Plan の Apache Tomcat へのデプロイ®

Apache Tomcat に Unica Plan を配置する際には、一連のガイドラインに従う必要があります。

removeAbandoned=true および **removeAbandonedTimeout=300** (5 分間) を `plands` データソース構成にパラメーター **Resource name="plands"** で始まるセクション内の `<catalina home>/conf/Catalina/localhost/plan.xml` ファイルで設定します。

以下に例を示します。

```
<Resource name="plands" auth="Application" type="javax.sql.DataSource"
 factory="org.apache.tomcat.jdbc.pool.DataSourceFactory" maxTotal="100" maxIdle="30" maxWaitMillis
 ="10000" username="abc" password="abc" driverClassName="org.mariadb.jdbc.Driver" removeAbandoned="true"
 removeAbandonedTimeout="300" url="jdbc:mysql://lp2-ap-51728467.prod.hclpnp.com:3306/pln121" />
```

注: アプリケーションが `300` 秒を超える実行時間が必要な SQL 照会を実行する場合、**removeAbandonedTimeout** の値を変更します。

Apache Tomcatのバージョンが、Unica 推奨ソフトウェア環境と最小システム要件のドキュメントに記載されている要件を満たしていることを確認してください。Apache Tomcat に Unica Plan を配置する場合には、以下のガイドラインに従ってください。

Apache Tomcat インストール・ディレクトリーの下の `bin` ディレクトリーから、ご使用のオペレーティング・システムに応じた `setenv.sh` または `setenv.bat` スクリプトを見つけるか、またはテキスト・エディターで作成し、`JAVA_OPTIONS` プロパティーを追加または変更し、次のエントリーを追加します。項目を区切るにはスペースを使用します。

- `-Dplan.home=<Plan_Home>`

  ここで、*<Plan_Home>* は Unica Plan をインストールしたディレクトリーへのパスです。

- `-Dfile.encoding=UTF-8`
- 次の JVM パラメーターが存在することを確認してください。存在しない場合、追加します。
  - バージョン 12.1.0、12.1.0.1、12.1.0.2、および 12.1.0.3 では、以下のパラメーターを追加します。
    - `-Dlog4j2.contextSelector=org.apache.logging.log4j.core.async.AsyncLoggerContextSelector`
    - `-Dlog4j2.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
    - `-Dlog4j.configurationFile=file:///<Plan_home>\conf\plan_log4j.xml (in case of fresh install of 12.1)`
  - バージョン 12.1.0.4 以降では、インストールする場合は、以下のパラメーターを追加します。バージョン 12.1.0 以降からアップグレードする場合は、前述 ページ 39のパラメーターを削除し、以下のパラメーターを追加します。
    - `-Dlog4j.configuration=file:///<Plan_home>\conf\plan_log4j_1x.xml`
    - `-Dplan.log4j.config=<Plan_home>\conf\plan_log4j.xml`
    - `-Dplan.log4j.async=true (If not set, deafults to true)`

注:

- バージョン 12.1.0、12.1.0.1、12.1.0.2、12.1.0.3 では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を `-Dlog4j2.configurationFile` に従ってここで指定されているように行い、`-Dlog4j.configurationFile` が配置されていることを確認してください。
- バージョン 12.1.0.4 以降では、`plan_log4j.xml` をカスタマイズして組織似合うように変更を加えている場合は、`plan_log4j.xml` でのそれらの設定を、配置されている `-Dplan.log4j.config` ファイルに従って、ここで指定されているように行ってください。
- Windows では、パスにバックスラッシュが含まれます。それに対し、UNIX ではスラッシュを含む必要があります。
- log4j 構成ファイル名のパス区切り記号は、基盤となるオペレーティング・システムに基づき、Windows スタイル () または UNIX (/) スタイルで設定される必要があります。
- パラメーター `relaxQueryChars` が Tomcat の `conf/server.xml` のコネクター・タグに存在することを確認します。

```
<Connector port="7002" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443"
relaxedQueryChars="|,[,],\,`,{,},^" />
```

- `<Plan_home>/conf/` の場所で、12.1.8 インストーラーは既存のファイルのバックアップを作成します。

バックアップ前のファイル名	バックアップ後のファイル名
`plan_log4j.xml`	`plan_log4j.xml_pre_12.1.8`
`plan_log4j_client.xml`	`plan_log4j_client.xml_pre_12.1.8`
`plan_log4j_socket_server.xml`	`plan_log4j_socket_server.xml_pre_12.1.8`

バックアップ・ファイルにカスタム設定が含まれる場合、そのカスタム設定が 12.1.8 インストーラーにより新しく追加された構成ファイルにコピーされていることを確認する必要があります。

1. HCL plan.war ファイルを、エンタープライズ・アプリケーションとして配置します。

   注: HCL EARファイルのデプロイメントは、Tomcatではサポートされていません。

2. インストール済み環境で非 ASCII 文字をサポートする必要がある場合 (例えば、ポルトガル語や、マルチバイト文字を必要とするロケール) は、以下のタスクを実行してください。
   a. tomcat インスタンス・ディレクトリーの下の bin ディレクトリー内のそれぞれの製品インスタンス・スクリプトに応じた setenv.sh ファイルを編集して、次の追加を行います: `Dfile.encoding=UTF-8 -Dclient.encoding.override=UTF-8 to JAVA_VENDOR`。
   b. Tomcat を再始動します。
3. 実稼働環境に配置している場合は、その tomcat インスタンスの JVM ヒープ設定を、すべてのインスタンスに対してそれぞれの app-one/bin/setenv.sh ファイルに追加できます

セキュリティを強化するための追加構成

このセクションの手順では、Web アプリケーション・サーバーの追加構成について説明します。これらはオプションの構成ですが、実行するとセキュリティーを強化できます。

X-Powered-By フラグを無効にする

組織で、ヘッダー変数内の X-Powered-By フラグがセキュリティー・リスクになることが懸念される場合、次の手順を使用してこのフラグを無効にすることができます。

1. WebLogic を使用している場合は、管理コンソールの**「*domainName*」 > 「構成」 > 「Web アプリケーション」**で、**「X-Powered-By ヘッダー」**を**「X-Powered-By ヘッダーは送信されません」**に設定します。
2. WebSphere® を使用している場合は、以下の手順を実行します。
   a. WebSphere® 管理コンソールで、**「サーバー」 > 「サーバー・タイプ」 > 「WebSphere Application Servers」 > 「*server_name*」 > 「Web コンテナー設定」 > 「Web コンテナー」**に移動します。
   b. **「追加プロパティー」**で、**「カスタム・プロパティー」**を選択します。
   c. 「カスタム・プロパティー」ページで、**「新規」**をクリックします。
   d. 「設定」ページで、`com.ibm.ws.webcontainer.disablexPoweredBy` という名前のカスタム・プロパティーを作成し、値を `false` に設定します。
   e. **「適用」**または**「OK」**をクリックします。

f. コンソール・タスクバーの**「保存」**をクリックして、構成の変更を保存します。
g. サーバーを再起動します。

制限された Cookie パスの構成

Web アプリケーション・サーバーでは、セキュリティーを強化するために Cookie アクセスを特定のアプリケーションに制限できます。制限しない場合、Cookie は、配置されたすべてのアプリケーションで有効になります。

1. WebLogic を使用している場合は、以下の手順を実行します。

   a. 制限された Cookie パスを追加する WAR パッケージまたは EAR パッケージから `weblogic.xml` ファイルを抽出します。

   b. 以下のコードを `weblogic.xml` ファイルに追加します。*context-path* は配置されているアプリケーションのコンテキスト・パスです。Unica アプリケーションでは、コンテキスト・パスは通常 /unica です。

   ```
   <session-descriptor>
       <session-param>
         <param-name>CookiePath</param-name>
         <param-value>/context-path> </param-value>
       </session-param>
   </session-descriptor>
   ```

   c. WAR または EAR ファイルを再ビルドします。

2. WebSphere® を使用している場合は、以下の手順を実行します。

   a. WebSphere® 管理コンソールで、**「セッション・マネージャー」** > **「Cookie」**タブにナビゲートします。

   b. **「Cookie パス」**にアプリケーションのコンテキスト・パスを設定します。

   Unica アプリケーションでは、コンテキスト・パスは通常 /unica です。

チェックリストメニューのアクティブ化

Planのチェックリストメニューをアクティブにするには、いくつかの構成を実行する必要があります。

このタスクについて

チェックリストメニューをアクティブにするには、次の手順を実行します。

1. **「設定」** > **「Plan 設定」**を選択します。
2. **「Unica Plan のアップグレード」**をクリックします。
3. **「アップグレード構成」**がチェックされていることを確認します。そうでない場合は、それを選択します。
4. **「アップグレード」**をクリックします。
5. アップグレード後、**「設定」** > **「Plan 設定」**を選択します。
6. サーバーを再起動します。
7. **「メニューの同期」**を選択します。

8. ログアウトしてから再度ログインします。
9. 確認するには、**「Plan」**メニューを選択します。リストに**「チェックリスト」**メニュー項目が表示されます。

Plan タスクでのラベルの同期

12.1.8 にアップグレードした後、**「Plan」 > 「タスク」 > 「すべてのタスク」**にアクセスすると、ラベルのないチェックボックスが表示されます。以下の手順を実行してこの問題を修正します。

1. **「設定」 > 「Plan 設定」**を選択します。
2. **管理設定**ページで、**その他のオプション**セクションまで下にスクロールして**「Unica Plan のアップグレード」**を選択します。
3. 列タイトル**モジュール**の前にあるチェックマークを選択して、すべての選択を選択解除します。
4. **「アップグレード構成」**を選択します。
5. **「アップグレード」**をクリックします。
6. ログアウトしてアプリケーション・サーバーを再起動します。
7. 再度 Unica Plan にログインして変更を確認します。

第 5 章. Unica Plan のアンインストール

Unica Plan のアンインストーラーを実行して、Unica Plan をアンインストールします。アンインストーラーを実行すると、インストール時に作成されたファイルが削除されます。例えば、構成ファイル、インストーラーの登録情報、およびユーザー・データなどのファイルがコンピューターから削除されます。

このタスクについて

Unica 製品をインストールする際、アンインストーラーが `Uninstall_Product` ディレクトリーに組み込まれます。*Product* は、製品の名前です。Windows™ の場合、コントロール・パネルの**「プログラムの追加と削除」**リストにもエントリーが追加されます。

アンインストーラーを実行せず、インストール・ディレクトリーのファイルを手動で削除した場合、後で同じ場所に製品を再インストールしたときに、インストールが不完全になる可能性があります。製品をアンインストールしても、そのデータベースは削除されません。アンインストーラーは、インストール中に作成されたデフォルト・ファイルのみを削除します。インストール後に作成または生成されたファイルはいずれも削除されません。

注: UNIX™ の場合、Unica Plan をインストールしたものと同じユーザー・アカウントがアンインストーラーを実行する必要があります。

1. Unica Plan Web アプリケーションを配置した場合、Web アプリケーション・サーバーから Web アプリケーションを配置解除します。
2. Web アプリケーション・サーバーをシャットダウンします。
3. Unica Plan に関連するプロセスを停止します。
4. 以下のいずれかの手順を実行して Unica Plan をアンインストールします。

   選択

   - `Uninstall_Product` ディレクトリー内に存在する Unica Plan アンインストーラーをクリックします。アンインストーラーは、Unica Plan をインストールする際に使用したモードで実行します。
   - コンソール・モードを使用して Unica Plan をアンインストールする場合は、コマンドライン・ウィンドウで、アンインストーラーが存在するディレクトリーにナビゲートして、次のコマンドを実行します。

     Uninstall_***Product*** -i console

   - サイレント・モードを使用して Unica Plan をアンインストールする場合は、コマンドライン・ウィンドウで、アンインストーラーが存在するディレクトリーにナビゲートして、次のコマンドを実行します。

     Uninstall_***Product*** -i silent

     サイレント・モードを使用して Unica Plan をアンインストールする場合、アンインストール・プロセスでは、ユーザーとの対話用のダイアログが表示されません。

   注: Unica Plan のアンインストールに関するオプションを指定しなかった場合、アンインストーラーは、Unica Plan のインストール時に使用されたモードで実行されます。

第 6 章. configTool

「構成」ページのプロパティーと値は、Unica Platform システム・テーブルに保存されます。`configTool` ユーティリティーを 使用して、構成設定をシステム・テーブルにインポートしたり、システム・テーブルからエクスポートしたりできます。

configTool をいつ使用するか

`configTool` は、次のような目的で使用できます。

- Unica Campaign に付属のパーティションおよびデータ・ソース・テンプレートをインポートするには、**「構成」**ページを使用して変更および複製できます。
- 製品インストーラーがプロパティーをデータベースに自動的に追加できない場合に Unica 製品を登録する (その構成プロパティーをインポートする)。
- バックアップ用、または Unica の他のインストール済み環境へのインポート用に、XML バージョンの構成設定をエクスポートする。
- **「カテゴリーの削除」**リンクを持たないカテゴリーを削除する。これを行うには、`configTool` を使用して構成をエクスポートし、カテゴリーを作成する XML を手動で削除し、`configTool` を使用して、編集された XML をインポートします。

重要: このユーティリティーは、構成プロパティーとその値を含む Unica Platform システム・テーブルデータベースの `usm_configuration` テーブルと `usm_configuration_values` テーブルを変更します。最良の結果を得るために、それらのテーブルのバックアップ・コピーを作成するか、`configTool` を使って既存の構成をエクスポートし、生成されるファイルをバックアップしてください。 そうすることで、`configTool` を使ったインポートに失敗した場合に構成を復元することができます。

Syntax

```
configTool -d -p "elementPath" [-o]

configTool -i -p "parent ElementPath" -f importFile [-o]

configTool -x -p "elementPath" -f exportFile

configTool -vp -p "elementPath" -f importFile [-d]

configTool -r productName -f registrationFile [-o] configTool -u productName
```

コマンド

`-d -p "elementPath" [o]`

構成プロパティー階層内のパスを指定して、構成プロパティーとそれらの設定を削除します。

エレメント・パスには、カテゴリーおよびプロパティーの内部名が 使用されている必要があります。それらを 得るには、**「構成」**ページの 目的のカテゴリーまたはプロパティーを選択して、右のペインに ある括弧内に示されているパスを確認します。| 文字を使って構成プロパティー階層のパスを区切り、パスを二重引用符で囲みます。

次のことに注意してください。

- このコマンドで削除できるのは、アプリケーション内のカテゴリーおよびプロパティーのみで、アプリケーション全体は削除できません。アプリケーション全体を登録解除するには、`-u` コマンドを使用します。
- **「構成」**ページに**「カテゴリーの削除」**リンクがないカテゴリーを削除するには、`-o` オプションを使用します。

`-vp` コマンドで `-d` を使用すると、configTool は指定されたパスに含まれるどの子ノードも削除します (指定された XML ファイルにそれらのノードが含まれない場合)。

`-i -p "parentElementPath" -f importFile [o]`

指定された XML ファイルから、構成プロパティーとそれらの設定をインポートします。

インポートするには、親要素へのパスを指定します。 この親要素の下に、カテゴリーがインポートされます。`configTool` ユーティリティーは、パス内で指定するカテゴリーの下にプロパティーをインポートします。

最上位より下のいずれのレベルでもカテゴリーを追加できますが、最上位カテゴリーと同じレベルではカテゴリーを追加できません。

親エレメント・パスには、カテゴリーおよびプロパティーの内部名が 使用されている必要があります。これらの内部名は、**「構成」**ページに移動して、必要なカテゴリーまたはプロパティーを選択し、右側のペインの括弧内に表示されるパスを調べることによって得ることができます。| 文字を使って構成プロパティー階層のパスを区切り、パスを二重引用符で囲みます。

`tools/bin` ディレクトリーからの相対的なインポート・ファイル場所を指定するか、ディレクトリーの絶対パスを指定することができます。相対パスを指定した場合、またはパスを指定しない場合、`configTool` は `tools/bin` ディレクトリーから相対的な場所にあるファイルを最初に探します。

デフォルトでこのコマンドは既存のカテゴリーを上書きしませんが、`-o` オプションを使用して上書きを強制することができます。

`-x -p "elementPath" -f exportFile`

指定された名前の XML ファイルに、構成プロパティーとそれらの設定をエクスポートします。

すべての構成プロパティーをエクスポートできます。 あるいは、構成プロパティー階層内のパスを指定することで、特定のカテゴリーに限定してエクスポートすることもできます。

要素パスにはカテゴリーおよびプロパティーの内部名を使用する必要があります。 これは、**「構成」**ページに移動し、必要なカテゴリーまたはプロパティーを選択して、右側のペインで括弧付きで表示されるパスを見ると分かります。| 文字を使って構成プロパティー階層のパスを区切り、パスを二重引用符で囲みます。

現行ディレクトリーからの相対的なエクスポート・ファイル場所を指定するか、ディレクトリーの絶対パスを指定することができます。ファイル指定に区切り記号 (UNIX™ の場合は /、Windows™ の場合は / または \ ) が含まれていない場合、`configTool` はファイルを Unica Platform インストール済み環境の `tools/bin` ディレクトリーの下に作成します。`xml` 拡張子を付けない場合、`configTool` によってそれが追加されます。

`-vp -p "elementPath" -f importFile [-d]`

このコマンドは 、主に手動アップグレードにおける構成プロパティーのインポートに使用されます。新しい構成プロパティーが含まれるフィックスパックを適用した後にアップグレードする場合、手動アップグレード・プロセスの一部として構成ファイルをインポートすることにより、フィックスパックが適用されたときに設定された値をオーバーライドできます。`-vp` コマンドは、既に設定されている構成値がインポートによってオーバーライドされないようにします。

重要: `configTool` ユーティリティーを -vp オプションを指定して使用したら、変更が適用されるように、Unica Platform が配置されている Web アプリケーション・サーバーを再始動する必要があります。

`-vp` コマンドで `-d` を使用すると、configTool は指定されたパスに含まれるどの子ノードも削除します (指定された XML ファイルにそれらのノードが含まれない場合)。

`-r productName -f registrationFile`

アプリケーションを登録します。`tools/bin` ディレクトリーに相対する登録ファイルの場所を指定することも、絶対パスを指定することもできます。デフォルトでこのコマンドは既存の構成を上書きしませんが、`-o` オプションを使用して上書きを強制することができます。*`productName`* パラメーターは、上記にリストされている名前のいずれかでなければなりません。

次のことに注意してください。

- `-r` コマンドを使用する際、登録ファイルには XML 内の最初のタグとして `<application>` を指定する必要があります。

  Unica Platform データベースに構成プロパティーを 挿入するために使用できる他のファイルが、製品と一緒に提供されることがあります。それらのファイルについては、`-i` コマンドを使用します。最初のタグとして `<application>` タグがあるファイルだけを `-r` コマンドとともに使用できます。
- Unica Platform の登録ファイルの名前は `Manager_config.xml` で、最初のタグは `<Suite>` です。新規インストールでこのファイルを登録するには、`populateDb` ユーティリティーを使用するか、「*Unica Platform* インストール・ガイド」にある説明に従って Unica Platform インストーラーを再実行します。
- 最初のインストールの後、Unica Platform 以外の製品を再登録する場合、`configTool` を `-r` コマンドおよび `-o` を指定して実行して、既存のプロパティーを上書きします。

`configTool` ユーティリティーは、製品の登録または登録解除を行うコマンドのパラメーターとして製品名を使用します。Unica の 8.5.0 リリースでは、多くの製品名が変更されました。ただし、`configTool` によって認識される名前は変更されていません。`configTool` で使用できる有効な製品名を、現在の製品名とともに以下にリストします。

表 9. configTool 登録および登録解除で使用する製品名

製品名	configTool で使用する名前
Unica Platform	マネージャー
Unica Campaign	`Campaign`
Unica Collaborate	`Collaborate`
Unica Deliver	配信
Unica Journey	`Journey`
Unica Insights	`UnicaInsights`
Unicaコンテンツの統合	`assetPicker`
Unica Offer	`Offer`
Unica Interact	`interact`
Unica Optimize	最適化
Unica Plan	`Plan`
Opportunity Detect	`Detect`
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	`SPSS`
Digital Analytics	`Coremetrics`

`-u` *`productName`*

`productName` によって指定されたアプリケーションを登録解除します。製品カテゴリーにパスを含める必要はありません。製品名は必須で、それのみで十分です。このプロセスで、製品のすべてのプロパティーと構成設定が削除されます。

オプション

`-o`

`-i` または `-r` と 共に使用すると、既存のカテゴリーまたは製品登録 (ノード) を上書きします。

`-d` とともに使用すると、**「構成」**ページ に**「カテゴリーの削除」**リンクがないカテゴリー (ノード) を削除することができます。

例(X)

- Unica Platform インストール済み環境の下の `conf` ディレクトリーの `Product_config.xml` という名前のファイルから構成設定をインポートします。

  ```
  configTool -i -p "Affinium" -f Product_config.xml
  ```

- 提供されている Unica Campaign データ・ソース・テンプレートの 1 つを、デフォルトの Unica Campaign パーティションである partition1 にインポートします。この例では、Unica Platform インストールの下にある Oracle データ・ソース・テンプレート`OracleTemplate.xml` を `tools/bin` ディレクトリーの下に配置したことを前提としています。

```
configTool -i -p "Affinium|Campaign|partitions|partition1|dataSources" -f OracleTemplate.xml
```

- すべての構成設定を `D:\backups` ディレクトリーの `myConfig.xml` という名前のファイルにエクスポートします。

```
configTool -x -f D:\backups\myConfig.xml
```

- 既存の Unica Campaign パーティション (データ・ソース・エントリーを含む) をエクスポートし、`partitionTemplate.xml` という名前のファイルに保存して、Unica Platform インストールの下にあるデフォルトの `tools/bin` ディレクトリーの下に保存します。

```
configTool -x -p "Affinium|Campaign|partitions|partition1" -f partitionTemplate.xml
```

- Unica Platform インストールの下にあるデフォルトの `tools/bin` ディレクトリーにある `app_config.xml` という名前のファイルを使用して、productNameという名前のアプリケーションを手動で登録し、このアプリケーションの既存の登録を上書きするように強制します。

```
configTool-r製品名-fapp_config.xml-o
```

- productName という名前のアプリケーションを登録解除します。

```
configTool -u productName
```

- 次のコマンドを実行して、encodeCSV 機能を有効にします。

```
configTool -vp -p "Affinium|Plan|umoConfiguration" -f Plan_Home\conf\Plan_encodeProperty_11.1.xml
```

- 以下を使用して `AffiniumWebApps\Campaign\interact\conf\interact_setup_navigation.xml` に Unica Interact 設定を構成メニューとして登録します:

```
configTool.bat -v -i -p "Affinium|suite|uiNavigation|settingsMenu" -f "interact_setup_navigation.xml"
```

第 7 章. Unica Plan 構成プロパティー

Unica Plan 構成プロパティーは、**「設定」**>**「構成」**ページで利用できます。構成プロパティーの詳細については、『Plan 管理者ガイド』を参照してください。

Unica Plan

このカテゴリーのプロパティーは、Unica Plan インストール済み環境のデフォルトとサポート対象のロケールを指定します。

supportedLocales

説明

Unica Plan のインストール済み環境で使用できるロケールを指定します。使用しているロケールだけをリストしてください。リストするロケールごとにサーバー上のメモリーが使用されます。使用されるメモリーの量は、テンプレートのサイズと数によって異なります。

初期インストールまたはアップグレード後にロケールを追加する場合は、アップグレード・サーブレットを再実行する必要があります。詳しくは、アップグレードの資料を参照してください。

この値を変更した場合、その変更を有効にするには、Unica Plan 配置を停止し、再始動する必要があります。

デフォルト値

en_US

defaultLocale

説明

Unica Plan において、Unica Plan 管理者が特定のユーザーについて明示的にオーバーライドしない限り、すべてのユーザーに対して表示されるサポート対象のロケールを指定します。

この値を変更した場合、その変更を有効にするには、Unica Plan 配置を停止し、再始動する必要があります。

デフォルト値

en_US

Unica Plan | navigation

このカテゴリーのプロパティーは、Uniform Resource Identifier、URL、ポートなどのナビゲーション用のオプションを指定します。

welcomePageURI

説明

Unica Plan 索引ページの Uniform Resource Identifier。この値は、Unica アプリケーションによって内部的に使用されます。この値を変更することは勧められていません。

デフォルト値

```
affiniumPlan.jsp?cat=projectlist
```

projectDetailpageURI

説明

Unica Plan 詳細設定ページの Uniform Resource Identifier。この値は、Unica アプリケーションによって内部的に使用されます。この値を変更することは勧められていません。

デフォルト値

blank

seedName

説明

Unica アプリケーションによって内部的に使用されます。この値を変更することは勧められていません。

デフォルト値

```
Plan
```

type

説明

Unica アプリケーションによって内部的に使用されます。この値を変更することは勧められていません。

デフォルト値

```
Plan
```

httpPort

説明

アプリケーション・サーバーで Unica Plan アプリケーションとの接続に使用されるポート番号。

デフォルト値

```
7001
```

httpsPort

説明

アプリケーション・サーバーで Unica Plan アプリケーションとのセキュア接続に使用されるポート番号。

デフォルト値

```
7001
```

serverURL

説明

Unica Plan インストールの URL。HTTP または HTTPS プロトコルのロケーターを受け入れます。

ユーザーが Chrome ブラウザーを使用して Unica Plan にアクセスする場合は、URL に完全修飾ドメイン・ネーム (FQDN) を使用します。FQDN を使用しない場合は、Chrome ブラウザーで製品 URL にアクセスできません。

デフォルト値

```
http://<server>:<port>/plan
```

注: <server> は小文字にする必要があります。

logoutURL

説明

内部的に使用されます。この値を変更することは勧められていません。

Unica Platform は、ユーザーがスイートでログアウト・リンクをクリックしたときに、この値を使用して、それぞれの登録済みアプリケーションのログアウト・ハンドラーを呼び出します。

デフォルト値

```
/uapsysservlet?cat=sysmodules&func=logout
```

表示名

説明

内部的に使用されます。

デフォルト値

```
Unica Plan
```

serverURLInternal

説明

これは、新しいパラメーターで、Unica Plan サーバーの内部 URL を指定します。リバース・プロキシーまたは Webor アクセス管理ソフトウェア (ISAM、Siteminder、など) を使用している場合、このパラメーターを Unica Plan の内部 URL に設定するか、その値を serverURL (Affinium|Plan|navigation|serverURL) の値に設定します。

この値は、Plan の新しい 'REST APIs v2、例えば、一元化されたオファー管理のオファー承認を使用するとき、Unica アプリケーションにより使用されます。

デフォルト値

`http://<server>:<port>/plan`

Unica Plan | 概要

このセクションの構成プロパティーは、Unica Plan インストール済み環境に関する情報をリストします。これらのプロパティーは編集できません。

表示名

説明

製品の表示名。

値

Unica Plan

releaseNumber

説明

現在インストールされているリリース。

値

<version>.<release>.<modification>

copyright

説明

著作権の年。

値

<year>

os

説明

Unica Plan がインストールされているオペレーティング・システム。

値

<operating system and version>

java

説明

Java™ の現在のバージョン。

値

<version>

サポート

説明

文書を読み取り、サービス要求を出します。

値

`https://hclpnpsupport.hcltech.com/csm`

appServer

説明

Unica Plan がインストールされているアプリケーション・サーバーのアドレス。

値

<IP address>

otherString

説明

値

blank

Unica Plan | umoConfiguration

これらのプロパティーは、Unica Plan の基本構成についての情報を指定します。

serverType

説明

アプリケーション・サーバー・タイプ。カレンダーのエクスポートに使用されます。

有効な値

`WEBLOGIC`、`WEBSPHERE`、`JBOSS`、`TOMCAT`

デフォルト値

Websphere

DBType

説明

データベース・タイプ。システム・テーブルがどのタイプのデータベースに保存されるかを示します。

有効な値

DB2、ORACLE、SQLSERVER、MARIADB、POSTGRESQL

注: PostgreSQL サポートは、12.1.7 からアップグレードしている場合のみ使用可能です。

デフォルト値

:NONE.

usermanagerSyncTime

説明

スケジュール設定された Unica Platform との同期化の時間間隔 (ミリ秒)。

デフォルト値

`10800000` (ミリ秒: 3 時間)

encodeCSV

説明

この設定の値が true の場合、Campaign 統合プロジェクトの TCS グリッドがエクスポートされると、セル内の文字列の値が引用符で囲まれます。このようなデータがインポートされ、文字列に式が含まれている場合、ユーザーが TCS グリッド・ページにアクセスしても実行されません。

有効な値

`True | False`

デフォルト値

偽

firstMonthInFiscalYear

説明

会計年度が開始する月を設定します。アカウントの「サマリー」タブには、そのアカウントの各会計年度の月別予算情報をリストした表示専用テーブルがあります。このテーブルの最初の月は、このパラメーターによって決まります。

1 月は 0 で表されます。会計年度を 4 月から開始するには、**firstMonthInFiscalYear** を 3 に設定します。

有効な値

`0` から `11` の整数

デフォルト値

`0`

maximumItemsToBeRetainedInRecentVisits

説明

「最新」メニューに表示する、最近表示したページへのリンクの最大数。

デフォルト値

`10` (リンク)

maxLimitForTitleString

説明

ページ・タイトルに表示できる最大文字数。指定された文字数よりもタイトルが長い場合、Unica Plan はタイトルを切り取って短くします。

デフォルト値

`40` (文字)

maximumLimitForBulkUploadItems

説明

同時にアップロードできる添付ファイルの最大数。

デフォルト値

`5` (添付ファイル)

workingDaysCalculation

説明

Unica Plan が期間を計算する方法を制御します。

有効な値

- `営業日`: 営業日のみ、営業日のみを含みます。休日も週末も含まれません。
- `週末`: 営業日 + 週末、営業日と週末を含みます。休日は含まれません。
- `off`: 営業日 + 休日。すべての営業日と休日を含みます。週末は含まれません。
- `すべて`: カレンダーのすべての日が含まれます。

デフォルト値

`all`

validateAllWizardSteps

説明

ユーザーがウィザードを使用してプログラム、プロジェクト、または要求を作成するときに、Unica Plan によって、現行ページの必須フィールドに値が設定されているかどうかが自動的に検証されます。このパラメーターは、ユーザーが**「終了」**をクリックしたときに、Unica Plan がすべてのページ (タブ) の必須フィールドを検証するかどうかを制御します。

有効な値

- `True`: Unica Plan は、ユーザーが表示しなかったページの必須フィールドを検査します (ワークフロー、トラッキング、添付ファイルを除く)。必須フィールドがブランクの場合、ウィザードはそのページを開き、エラー・メッセージを表示します。
- `False`: Unica Plan は、ユーザーが表示しなかったページの必須フィールドを検証しません。

デフォルト値

`真`

enableRevisionHistoryPrompt

説明

ユーザーがプロジェクト、要求、または承認を保存するときに変更コメントを追加するよう求めるプロンプトが出るようにします。

有効な値

`True | False`

デフォルト値

`偽`

useForecastDatesInTaskCalendar

説明

タスクがカレンダー・ビューに表示されるときに使用される日付のタイプを指定します。

有効な値

- `True`: 予測/実際の日付を使用してタスクを表示します。
- `False`: ターゲット日を使用してタスクを表示します。

デフォルト値

偽

copyRequestProjectCode

説明

プロジェクト・コード (PID) を要求からプロジェクトに引き継ぐかどうかを制御します。このパラメーターを `False` に設定した場合、プロジェクトと要求は、異なるコードを使用します。

有効な値

`True | False`

デフォルト値

真

projectTemplateMonthlyView

説明

プロジェクト・テンプレートのワークフローで月次ビューが許可されるかどうかを制御します。

有効な値

`True | False`

デフォルト値

偽

disableAssignmentForUnassignedReviewers

説明

承認のために作業を役割別に割り当てる方法を指定します。**disableAssignmentForUnassignedReviewers** パラメーターは、「スタッフ」タブにある**「役割別に作業を割り当て」**の、ワークフロー承認における承認者の割り当てに関する動作を制御します。

有効な値

- `True`: 「スタッフ」タブにおいて未割り当てのレビュー担当者は、新しいステップとして承認に追加されません。
  - 追加オプション: 所有者によって割り当てられた既存の承認者で、割り当てられた役割を持たないものは、変更されません。「スタッフ」タブに役割が「未割り当て」のレビュー担当者が存在しても、新しい承認者ステップは追加されません。
  - 置換オプション: 所有者によって割り当てられた既存の承認者で、役割を持たないものは、ブランクに置き換えられます。「スタッフ」タブに役割が「未割り当て」のレビュー担当者が存在しても、新しい承認者ステップは追加されません。
- `False`: 未割り当てのレビュー担当者は、承認に追加されます。
  - 追加オプション: 定義された役割がない所有者割り当てステップが承認に存在する場合は、役割を持たないすべてのレビュー担当者が、レビュー担当者として承認に追加されます。
  - 置換オプション: 承認における既存の承認者は、「スタッフ」タブの未割り当て承認者に置き換えられます。

デフォルト値

偽

enableApplicationLevelCaching

説明

アプリケーション・レベルのキャッシングを有効にするかどうかを示します。キャッシング・メッセージのマルチキャストが有効になっていないクラスター環境で最良の結果を得るには、Unica Plan のアプリケーション・レベルのキャッシングをオフにすることを検討してください。

有効な値

`True | False`

デフォルト値

真

customAccessLevelEnabled

説明

カスタム・アクセス・レベル (プロジェクトの役割) を Unica Plan で使用するかどうかを決定します。

有効な値

- `True`: プロジェクトおよび要求に対するユーザーアクセスは、オブジェクト・アクセス・レベルおよびカスタム・アクセス・レベル (プロジェクトの役割) に従って評価されます。カスタム・タブのタブ・セキュリティーが有効になります
- `False`: プロジェクトおよび要求へのユーザーアクセスは、オブジェクト・アクセス・レベル (オブジェクトの暗黙の役割) のみに従って評価され、カスタム・タブのタブ・セキュリティーは無効になります。

デフォルト値

真

enableUniqueIdsAcrossTemplatizableObjects

説明

プログラム、プロジェクト、計画、請求書を含むテンプレートから作成されたすべてのオブジェクトにおいて、固有の内部 ID を使用するかどうかを決定します。

有効な値

- `True` に設定すると、テンプレートから作成されたすべてのオブジェクトにおいて固有の内部 ID を使用できます。この構成を使用すると、オブジェクト・タイプが異なる場合でも、同じテーブルをシステムが使用できるようになるため、オブジェクトをまたがるレポート作成が簡単になります。
- `False` に設定すると、テンプレートから作成されたすべてのオブジェクトにおいて固有の内部 ID を使用できなくなります。

デフォルト値

真

FMEnabled

説明

財務管理モジュールを有効または無効にします。これにより、製品に「アカウント」、「請求書」、および「予算」のタブが表示されるかどうかが決まります。

有効な値

`True | False`

デフォルト値

偽

FMProjVendorEnabled

説明

プロジェクト明細項目のベンダー列の表示/非表示を指定するためのパラメーター。

有効な値

`True | False`

デフォルト値

偽

FMPrgmVendorEnabled

説明

プログラム明細項目のベンダー列の表示/非表示を指定するためのパラメーター。

有効な値

`True | False`

デフォルト値

偽

マーケティング・オブジェクトのデフォルトのレポート・タイプ

説明

オファーを含むカスタム・マーケティング・オブジェクトのデフォルトのレポート・タイプを定義します。このパラメーターの値を変更した場合は、アプリケーションを再起動する必要があります。

有効な値

改訂履歴 | 相互参照

デフォルト値

改訂履歴

enablePlanAssetPickerIntegration

説明

`True` を設定して Unica コンテンツの統合を有効にして Unica Plan と統合します

有効な値

`True | False`

デフォルト値

偽

Unica Plan | umoConfiguration | 承認

これらのプロパティーは、承認に関するオプションを指定します。

specifyDenyReasons

説明

承認を拒否する理由のカスタマイズ可能なリストを有効にします。有効にされると、管理者は「承認拒否理由」リストにオプションを設定してから、 拒否理由を各ワークフロー・テンプレートおよびワークフローを定義する各プロジェクト・テンプレートに関連付けます。承認または承認に含まれる項目を拒否するユーザーは、事前定義されたこれらの理由のいずれかを選択する必要があります。

有効な値

`True | False`

デフォルト値

偽

approveWithChanges

説明

承認に関する**「変更付きで承認」**オプションを有効にします。有効にすると、ユーザーがプロジェクト・テンプレート、プロジェクト、または独立した承認で承認をセットアップする際に、デフォルトで**「承認者が変更付きで承認することを許可する」**オプションが選択されます。**overrideApproveWithChanges** プロパティーを `True` に設定すると、**「承認者が変更付きで承認することを許可する」**オプションを編集できます。

承認のセットアップ時に**「承認者が変更付きで承認することを許可する」**オプションを選択すると、承認者は**「変更付きで承認」**オプションを選択してタスクを承認できます。

有効な値

`True | False`

デフォルト値

真

overrideApproveWithChanges

説明

ユーザーがプロジェクト・テンプレート、プロジェクト、または独立した承認で承認をセットアップする際に**「承認者が変更付きで承認することを許可する」**オプションのデフォルト設定を編集できるようにするには、`True` に設定します。デフォルト設定は、**approveWithChanges** プロパティーによって決まります。

有効な値

```
True | False
```

デフォルト値

```
真
```

Unica Plan | umoConfiguration | テンプレート

これらのプロパティーは、Unica Plan におけるテンプレートについての情報を指定します。最良の結果を得るには、これらのパラメーターのデフォルト値を変更しないでください。

templatesDir

説明

すべてのプロジェクト・テンプレート定義を格納する XML ファイルを入れるためのディレクトリーを指定します。

絶対パスを使用してください。

デフォルト値

```
<HCL_Unica_Home>/<Plan_Home>/templates
```

assetTemplatesFile

説明

資産のテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
asset_templates.xml
```

planTemplatesFile

説明

計画のテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
plan_templates.xml
```

programTemplatesFile

説明

プログラムのテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
program_templates.xml
```

projectTemplatesFile

説明

プロジェクトのテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
project_templates.xml
```

invoiceTemplatesFile

説明

請求書のテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
invoice_templates.xml
```

componentTemplatesFile

説明

カスタム・マーケティング・オブジェクト・タイプのテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
component_templates.xml
```

metricsTemplateFile

説明

メトリックのテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
metric_definition.xml
```

teamTemplatesFile

説明

チームのテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
team_templates.xml
```

offerTemplatesFile

説明

オファーのテンプレートを定義する XML ファイル。このファイルは、**templatesDir** で指定されたディレクトリー内にあることが必要です。

デフォルト値

```
uap_sys_default_offer_comp_type_templates.xml
```

Unica Plan | umoConfiguration | attachmentFolders

これらのプロパティーは、添付ファイルのアップロードと保管に使用するディレクトリーを指定します。

uploadDir

説明

プロジェクトの添付ファイルが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/projectattachments
```

planUploadDir

説明

計画の添付ファイルが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/planattachments
```

programUploadDir

説明

プログラムの添付ファイルが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/programattachments
```

componentUploadDir

説明

マーケティング・オブジェクトの添付ファイルが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/componentattachments
```

taskUploadDir

説明

タスクの添付ファイルが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/taskattachments
```

approvalUploadDir

説明

承認項目が保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/approvalitems
```

assetUploadDir

説明

資産が保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/assets
```

accountUploadDir

説明

アカウントの添付ファイルが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/accountattachments
```

invoiceUploadDir

説明

請求書の添付ファイルが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/invoiceattachments
```

graphicalRefUploadDir

説明

属性イメージが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/graphicalrefimages
```

templateImageDir

説明

テンプレート・イメージが保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/images
```

recentDataDir

説明

各ユーザーの最近のデータ (直列化済み) を保管する一時ディレクトリー。

デフォルト値

```
<Plan_Home>/recentdata
```

workingAreaDir

説明

グリッドのインポート時にアップロードされた CSV ファイルを保管する一時ディレクトリー。

デフォルト値

```
<Plan_Home>/umotemp
```

managedListDir

説明

管理対象のリスト定義が保管されるアップロード・ディレクトリー。

デフォルト値

```
<Plan_Home>/managedList
```

Unica Plan | umoConfiguration | E メール

これらのプロパティーは、Unica Plan における E メール通知の送信に関する情報を指定します。

notifyEMailMonitorJavaMailHost

説明

E メール通知メール・サーバーの DNS ホスト名またはそのドット形式の IP アドレスのいずれかを指定するストリング (オプション)。SMTP サーバーのマシン名または IP アドレスに設定されます。

セッションパラメーターを使用する既存の JavaMail™ セッションを Unica Plan に提供しておらず、委任が「完了」とマークされている場合は、このパラメーターが必要です。

デフォルト値

```
[CHANGE-ME]
```

notifyDefaultSenderEmailAddress

説明

有効な E メール・アドレスを設定します。システムは、通知 E メール・メッセージを送信するための有効な E メール・アドレスがない場合には、このアドレスに E メール・メッセージを送信します。

デフォルト値

```
[CHANGE-ME]
```

notifySenderAddressOverride

説明

このパラメーターを使用して、通知における「返信」および「差出人」の E メール・アドレスの標準値を指定します。デフォルトでは、これらのアドレスには、イベント所有者の E メール・アドレスが設定されます。

デフォルト値

blank

Unica Plan | umoConfiguration | マークアップ

これらのプロパティーは、マークアップ・オプションを指定します。Unica Plan には、添付ファイルのコメントを作成するためのマークアップツールが用意されています。Adobe™ Acrobat マークアップまたはネイティブ Unica Plan マークアップのいずれかを使用できます。使用するオプションを構成するには、このカテゴリーのプロパティーを使用します。

markupServerType

説明

使用するマークアップ・オプションを決定します。

有効な値

- `SOAP` を指定すると、ユーザーは PDF 文書のマークアップを編集および表示できます。マークアップには Adobe™ Acrobat Professional が必要です。これを指定した場合、ユーザーはネイティブ Unica Plan メソッドを使用して Web ブラウザーで以前に作成されたマークアップを表示できません。

  `SOAP` を指定する場合は、**markupServerURL** パラメーターも構成する必要があります。

  `SOAP` を指定する場合、Adobe Acrobat がインストールされているディレクトリーの JavaScripts サブディレクトリーにコピーされたカスタマイズ済み `UMO_Markup_Collaboration.js` を削除する必要があります。以下に例を示します。 `C:\Program files (x86)\Adobe\Acrobat 10.0\Acrobat\Javascripts\UMO_Markup_Collaboration.js`. このファイルは不要になりました。

- `MCM` を指定すると、ユーザーが Web ブラウザーでマークアップを編集および表示できるネイティブ Unica Plan マークアップメソッドを使用できます。これを指定した場合、ユーザーは、以前に Adobe™ Acrobat を使用して PDF で作成されたマークアップを編集することも表示することもできません。
- `PDF Embed API` は、PDF 埋め込み API を有効にします。この機能を使用してフィードバックを提供し、注釈を操作できます。
- ブランクの場合、マークアップ機能は無効になり、**「マークアップの表示/追加」**リンクは表示されません。

デフォルト値

`MCM`

markupServerURL

説明

markupServerType = `SOAP` に依存しています。

マークアップ・サーバーをホストするコンピューターの URL を設定します (Web アプリケーション・サーバーが listen に使用するポートの番号を含みます)。この URL には、完全修飾ホスト名が含まれていなければなりません。

HTTP または HTTPS プロトコルのロケーターを受け入れます。

デフォルト値

`http://<server>:<port>/plan/services/collabService?wsdl`

instantMarkupFileConversion

説明

`True` の場合、Unica Plan は、ユーザーがマークアップの項目を初めて開くときに PDF 添付資料からイメージへの変換を実行するのではなく、PDF 添付資料がアップロードされるとすぐにこの変換を実行します。

有効な値

`True | False`

デフォルト値

偽

adobeConsoleKey

説明

`http://developer.adobe.com` Web サイトから生成されたコンソールキー。詳しくは、「Unica *Plan* 管理者ガイド」を参照してください。このフィールドは、`PDF 埋め込み API` を markupServerType ページ 68 パラメーターに選択した場合のみ適用されます。

デフォルト値

-

Unica Plan | umoConfiguration | グリッド

これらのプロパティーは、グリッドに関するオプションを指定します。

gridmaxrow

説明

グリッドで取得される最大行数を定義する整数 (オプション)。デフォルトの `-1` の場合は、すべての行が取得されます。

デフォルト値

`-1`

reloadRuleFile

説明

グリッド検証プラグインを再ロードする必要があるかどうかを示すブール・パラメーター (オプション)。

有効な値

`True | False`

デフォルト値

真

gridDataValidationClass

説明

カスタム・グリッド・データ検証クラスを指定するパラメーター (オプション)。指定しない場合は、デフォルトの組み込みプラグインがグリッド・データ検証に使用されます。

デフォルト値

blank

tvcDataImportFieldDelimiterCSV

説明

グリッドにインポートされたデータの解析に使用する区切り文字。デフォルトはコンマ (,) です。

デフォルト値

, (カンマ)

maximumFileSizeToImportCSVFile

説明

TVC のコンマ区切りデータをインポートするときにアップロードできる最大ファイル・サイズ (MB) を表します。

デフォルト値

`0` (無制限)

maximumRowsToBeDisplayedPerPageInGridView

説明

グリッド・ビューの 1 ページ当たりの表示行数を指定します。

有効な値

正整数

デフォルト値

`100`

griddataxsd

説明

グリッド・データ XSD ファイルの名前。

デフォルト値

`griddataschema.xsd`

gridpluginxsd

説明

グリッド・プラグイン XSD ファイルの名前。

デフォルト値

```
gridplugin.xsd
```

gridrulesxsd

説明

グリッド・ルール XSD ファイルの名前。

デフォルト値

```
gridrules.xsd
```

Unica Plan | umoConfiguration | ワークフロー

これらのプロパティーは、Unica Plan におけるワークフローについてのオプションを指定します。

hideDetailedDateTime

説明

タスク・ページにおける詳細な日時のパラメーターの表示/非表示パラメーター (オプション)。

有効な値

```
True | False
```

デフォルト値

偽

daysInPastRecentTask

説明

このパラメーターは、タスクが「最新」と見なされる期間を決めます。タスクが「アクティブ」であり、開始されてからの期間がこの日数未満であるか、またはタスクの「ターゲット終了日」が現在日付とこの日数前の日付との間にある場合、そのタスクは最新のタスクとして表示されます。

有効な値

正整数

デフォルト値

`14` (日)

daysInFutureUpcomingTasks

説明

このパラメーターは、将来の何日間について次回のタスクを検索するかを決定します。タスクが次の **daysInFutureUpcomingTasks** の期間に開始する場合、または現在日付の前に終了しない場合、そのタスクは次回のタスクとなります。

有効な値

正整数

デフォルト値

`14` (日)

beginningOfDay

説明

営業日の始業時間。このパラメーターは、小数形式の期間を使用したワークフローの日時の計算に使用されます。

有効な値

`0` から `12` の整数

デフォルト値

`9` (9 AM)

numberOfHoursPerDay

説明

1 日当たりの時間数。このパラメーターは、小数形式の期間を使用したワークフローの日時の計算に使用されます。

有効な値

`1` から `24` の整数

デフォルト値

`8` (時間)

mileStoneRowBGColor

説明

ワークフロー・タスクの背景色を定義します。この値を指定するには、色を表す 6 文字の 16 進コードの前に # 文字を挿入します。例えば、#0099CC と指定します。

デフォルト値

```
#DDDDDD
```

UI

説明

ワークフロー UI のバージョンを定義します。可能な値は、最新の UI を表示する拡張、または 12.1 より前のバージョンで UI を表示する従来型です。これは、システムレベルの設定であり、ユーザーレベルの設定またはプロジェクトレベルの設定ではありません。この設定を変更する場合は、Plan アプリケーションを再始動する必要があります。

デフォルト値

拡張

Unica Plan | umoConfiguration | integrationServices

これらのプロパティーは、Unica Plan 統合サービスモジュールについての情報を指定します。統合サービスモジュールは、Unica Plan の機能を Web サービスとトリガーを使用して拡張します。

enableIntegrationServices

説明

統合サービス・モジュールを有効および無効にします。

有効な値

```
True | False
```

デフォルト値

偽

integrationProcedureDefinitionPath

説明

カスタム・プロシージャー定義 XML ファイルへの絶対ファイル・パス (オプション)。

デフォルト値

[plan-home]/devkits/integration/examples/src/procedure/procedure-plugins.xml

integrationProcedureClasspathURL

説明

カスタム・プロシージャーのクラスパスへの URL。

デフォルト値

`file:///[plan-home]/devkits/integration/examples/classes/`

Unica Plan | umoConfiguration | campaignIntegration

このカテゴリーのプロパティーは、Unica Campaign 統合用のオプションを指定します。

defaultCampaignPartition

説明

Unica Plan が Unica Campaign と統合されていると、このパラメーターは、プロジェクト・テンプレートに campaign-partition-id が定義されていない場合にデフォルトの Unica Campaign パーティションを指定します。

デフォルト値

`partition1`

webServiceTimeoutInMilliseconds

説明

Web サービス統合 API 呼び出しに追加されます。このパラメーターは、Web サービス API 呼び出しのタイムアウトとして使用されます。

デフォルト値

`1800000` ミリ秒 (30 分)

Unica Plan | umoConfiguration | レポート

これらのプロパティーは、Unica Plan が使用するレポートについての情報を指定します。

reportsAnalysisSectionHome

説明

分析セクション・レポートのホーム・ディレクトリーを示します。Cognos レポートの値は `/content/folder[@name='Affinium Plan']` である必要があります。Unica Insights レポートの値は Plan/Affinium Plan である必要があります。

デフォルト値

`/content/folder[@name='Affinium Plan']`

reportsAnalysisTabHome

説明

分析タブ・レポートのホーム・ディレクトリーを示します。Cognos レポートの値は `/content/folder[@name='Affinium Plan - Object Specific Reports']` である必要があります。Unica Insights レポートの値は Plan/Affinium Plan - Object Specific Reports である必要があります。

デフォルト値

```
/content/folder[@name='Affinium Plan - Object Specific Reports']
```

cacheListOfReports

説明

このパラメーターは、オブジェクト・インスタンスの分析ページにおけるレポート・リストのキャッシングを有効にします。

有効な値

```
True | False
```

デフォルト値

偽

Unica Plan | umoConfiguration | invoiceRollup

このカテゴリーのプロパティーは、請求書ロールアップ用のオプションを指定します。

invoiceRollupMode

説明

ロールアップがどのように発生するかを指定します。許容値は以下のとおりです。

有効な値

- `immediate`: 請求書が支払い済みとマークされるたびに、ロールアップが発生します。
- `schedule`: スケジュールに基づいてロールアップが発生します。

  このパラメーターが `schedule` に設定されると、システムは以下のパラメーターを使用して、ロールアップ発生のタイミングを決定します。

  - `invoiceRollupScheduledStartTime`
  - `invoiceRollupScheduledPollPeriod`

デフォルト値

即時

invoiceRollupScheduledStartTime

説明

invoiceRollupMode が `schedule` である場合、このパラメーターは以下のように使用されます。

- このパラメーターに値 (例えば、11:00 pm) が含まれている場合、その値は、スケジュールが開始するための開始時刻となります。
- このパラメーターが未定義の場合は、サーバーの始動時にロールアップ・スケジュールが開始します。

invoiceRollupMode が `immediate` である場合、このパラメーターは使用されません。

デフォルト値

`11:00 pm`

invoiceRollupScheduledPollPeriod

説明

invoiceRollupMode が `schedule` である場合、このパラメーターは、ロールアップが発生するためのポーリング期間 (秒) を指定します。

invoiceRollupMode が `immediate` である場合、このパラメーターは使用されません。

デフォルト値

`3600` (1 時間)

Unica Plan | umoConfiguration | データベース

これらのプロパティーは、Unica Plan に使用するデータベースについての情報を指定します。

fileName

説明

JNDI 検索を使用してデータ・ソースをロードするためのファイルへのパス。

デフォルト値

`plan_datasources.xml`

sqlServerSchemaName

説明

使用するデータベース・スキーマを指定します。このパラメーターは、Unica Plan データベースに SQL Server を使用している場合にのみ適用されます。

デフォルト値

`dbo`

db2ServerSchemaName

重要: このパラメーター用に提供されたデフォルト値を変更することは勧められていません。

説明

Unica アプリケーションによって内部的に使用されます。

デフォルト値

blank

thresholdForUseOfSubSelects

説明

ここで指定したレコード数を超えると、(リスト・ページの) SQL の IN 節で、IN 節内の実際のエンティティー ID の代わりに副照会を使用する必要があります。このパラメーターを設定すると、大規模なアプリケーションデータセットが含まれる Unica Plan インストール済み環境のパフォーマンスが向上します。ベスト・プラクティスとして、パフォーマンスの問題が発生しない限りこの値を変更しないでください。このパラメーターがないか、あるいはコメント化されている場合、データベースは、しきい値が大きな値に設定されるかのように動作します。

デフォルト値

`3000` (レコード)

commonDataAccessLayerFetchSize

説明

このパラメーターは、パフォーマンスに影響されやすい特定の重要な照会について、結果セットの取り出しサイズを指定します。

デフォルト値

`0`

commonDataAccessLayerMaxResultSetSize

説明

このパラメーターは、パフォーマンスに影響されやすい特定の重要な照会について、結果セットの最大サイズを指定します。

デフォルト値

`-1`

useDBSortForAllList

説明

このパラメーターは、すべての Unica Plan リストハンドラーを構成するために使用されます。特定のリストのページング動作をオーバーライドするには、別の **useDBSortFor<module>List** パラメーターを使用します。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

真

useDBSortForPlanList

説明

このパラメーターは、計画リスト・ハンドラーを構成するために使用されます。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

真

useDBSortForProjectList

説明

このパラメーターは、プロジェクト・リスト・ハンドラーを構成するために使用されます。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

真

useDBSortForTaskList

説明

このパラメーターは、タスク・リスト・ハンドラーを構成するために使用されます。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

真

useDBSortForProgramList

説明

このパラメーターは、プログラム・リスト・ハンドラーを構成するために使用されます。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

真

useDBSortForApprovalList

説明

このパラメーターは、承認リスト・ハンドラーを構成するために使用されます。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

真

useDBSortForInvoiceList

説明

このパラメーターは、請求書リスト・ハンドラーを構成するために使用されます。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

`真`

useDBSortForAlerts

説明

このパラメーターは、アラート・リスト・ハンドラーを構成するために使用されます。

有効な値

- `True`: データベースから一度に 1 ページのリスト・データを取得します。
- `False`: すべてのリスト・データをキャッシュに入れます。

デフォルト値

`真`

Unica Plan | umoConfiguration | listingPages

これらのプロパティーは、Unica Plan のページ上におけるマーケティングオブジェクトやマーケティングプロジェクトなどのリスト項目についての情報を指定します。

listItemsPerPage

説明

1 つのリスト・ページに表示される項目 (行) の数を指定します。この値は、0 より大きくする必要があります。

デフォルト値

`10`

listPageGroupSize

説明

リスト・ページのリスト・ナビゲーターに表示されるページ番号のサイズを指定します。例えば、ページ 1 - 5 は、ページ・グループです。この値は、0 より大きくする必要があります。

デフォルト値

`7`

maximumItemsToBeDisplayedInCalendar

説明

カレンダーに表示されるオブジェクト (計画、プログラム、プロジェクト、またはタスク) の最大数。このパラメーターは、ユーザーがカレンダー・ビューを選択した場合に表示するオブジェクトの数を制限します。数値 0 は、制限がないことを示します。

デフォルト値

```
0
```

listDisplayShowAll

説明

リスト・ページに「すべて表示」リンクを表示します。

デフォルト値

```
偽
```

有効な値

```
True | False
```

自動折り返し

説明

末尾に省略記号を表示する代わりに、長い名前を切り捨てずに表示できます。プロジェクト名、タスク名、承認名、オファー名、およびオファーリスト名に適用されます。長い名前は折り返され、同じ行の次のラインに表示されます。

デフォルトでは、**自動折り返し**は `False` に設定されています。**自動折り返し** の値を `True` に設定した後、Platform と Plan アプリケーションを再始動して、この構成をアクティブ化する必要があります。

デフォルト値

```
偽
```

有効な値

```
True | False
```

Unica Plan | umoConfiguration | objectCodeLocking

これらのプロパティーは、Unica Plan における計画、プログラム、プロジェクト、資産、およびマーケティングオブジェクトのオブジェクトロックについての情報を指定します。

enablePersistentObjectLock

説明

Unica Plan がクラスター環境に配置されている場合は、 このパラメーターを `True` に設定する必要があります。データベースにおいてオブジェクト・ロック情報は永続的です。

有効な値

`True | False`

デフォルト値

偽

lockProjectCode

説明

ユーザーがプロジェクトの「サマリー」タブでプロジェクト・コードまたは PID を編集できるかどうかを決定します。

有効な値

- `True`: ロックを有効にします。
- `False`: ロックを無効にします。

デフォルト値

真

lockProgramCode

説明

ユーザーがプログラムの「サマリー」タブでプログラム・コードまたは PID を編集できるかどうかを決定します。

有効な値

- `True`: ロックを有効にします。
- `False`: ロックを無効にします。

デフォルト値

真

lockPlanCode

説明

ユーザーが計画の「計画サマリー」タブで計画コードまたは PID を編集できるかどうかを決定します。

有効な値

- `True`: ロックを有効にします。
- `False`: ロックを無効にします。

デフォルト値

真

lockMarketingObjectCode

説明

ユーザーがマーケティング・オブジェクトの「サマリー」タブでマーケティング・オブジェクト・コードまたは PID を編集できるかどうかを決定します。

有効な値

- `True`: ロックを有効にします。
- `False`: ロックを無効にします。

デフォルト値

真

lockAssetCode

説明

ユーザーが資産の「サマリー」タブで資産コードまたは PID を編集できるかどうかを決定します。

有効な値

- `True`: ロックを有効にします。
- `False`: ロックを無効にします。

デフォルト値

真

Unica Plan | umoConfiguration | thumbnailGeneration

これらのプロパティーは、Unica Plan がサムネールを生成する方法とタイミングについての情報を指定します。

trueTypeFontDir

説明

True Type フォントが存在するディレクトリーを指定します。このパラメーターは、Aspose を使用する非 Windows™ オペレーティング・システムでサムネールを生成する場合には必須です。Windows™ インストール済み環境の場合、このパラメーターはオプションです。

デフォルト値

blank

coreThreadPoolSize

説明

サムネール・ジェネレーター・スレッドのスレッド・プールに保持される永続スレッド数を指定します。

デフォルト値

```
7
```

maxThreadPoolSize

説明

サムネール・ジェネレーター・スレッドのスレッド・プールで許可される最大スレッド数を指定します。

デフォルト値

```
10
```

threadKeepAliveTime

説明

サムネール・ジェネレーター・スレッドのキープアライブ時間を構成するためのパラメーター。

デフォルト値

```
60
```

threadQueueSize

説明

サムネール・ジェネレーター・スレッドのスレッド・キュー・サイズを構成するためのパラメーター。

デフォルト値

```
20
```

disableThumbnailGeneration

説明

アップロードされた文書のためにサムネール・イメージを生成するかどうかを決めます。値 `True` は、サムネールの生成を有効にします。

デフォルト値

偽

有効な値

`True | False`

markupImgQuality

説明

レンダリングされるページに適用される拡大率またはズーム係数。

デフォルト値

`1`

Unica Plan | umoConfiguration | スケジューラー | intraDay

このプロパティーは、対象日におけるスケジューラーの実行頻度を指定します。

schedulerPollPeriod

説明

バッチ・ジョブが、プロジェクトの正常性ステータスの実行を毎日計算する際の頻度を秒数で定義します。

注: 日次のバッチ・ジョブだけが、レポートで使用されるプロジェクトの正常性ステータスの履歴を更新します。

デフォルト値

`60` (秒)

Unica Plan | umoConfiguration | スケジューラー | 日次

このプロパティーは、スケジューラーの毎日の開始時刻を指定します。

schedulerStartTime

説明

プロジェクトの正常性ステータスを計算するバッチ・ジョブの開始時刻を定義します。このジョブは、以下のことも行います。

- レポートで使用されるプロジェクトの正常性ステータスの履歴を更新します。
- E メール通知を配信登録しているユーザーへの配布を開始します。

注: システムがこのバッチ・ジョブを開始するのは、計算がまだ実行されていない場合だけです。ジョブが **intraDay** パラメーターとは異なる時刻に、そしてユーザーがこの計算を手動で要求する可能性の低い時刻に開始するように、このパラメーターを定義してください。

デフォルト値

```
11:00 pm
```

Unica Plan | umoConfiguration | 通知

これらのプロパティーは、イベントモニターについての情報を含む、Unica Plan における通知に関する情報を指定します。

notifyPlanBaseURL

説明

Unica Plan 配置の URL (ホスト名とポート番号を含む)。Unica Plan では、Unica Plan 内の他の情報へのリンクを含む通知に、この URL が組み込まれます。

注: メールクライアントと Unica Plan サーバーを同じサーバー上で実行している場合以外は、「localhost」をサーバー名として使用しないでください。

デフォルト値

```
http://<server>:<port>/plan/affiniumplan.jsp
```

notifyDelegateClassName

説明

サービスによってインスタンス化される委任実装の完全修飾 Java™ クラス名。このクラスには、`com.unicacorp.afc.service.IServiceImpl` インターフェースをインプリメントする必要があります。指定しない場合は、デフォルトでローカル実装になります。

デフォルト値

blank

notifyIsDelegateComplete

説明

委任実装が完了したかどうかを示すブール・ストリング (オプション)。指定しない場合は、デフォルトで `True` に設定されます。

デフォルト値

`真`

有効な値

`True | False`

notifyEventMonitorStartTime

説明

Unica Plan の始動後、初めてイベント通知モニターの処理が開始される時刻を指定します。値の形式は、現行ロケールの `java.text.DateFormat` クラスのショート・バージョンに従ってください。例えば、米国英語ロケールの場合、有効なストリングとしては `11:45 pm` などが考えられます。

デフォルト値

ブランク (Unica Plan の始動直後)。

notifyEventMonitorPollPeriod

説明

イベント・モニターがポーリングとポーリングの間にスリープするおおよその時間 (秒) を定義します。ポーリング期間とポーリング期間の間、イベントはイベントキューに累積されます。 ポーリング期間が短いほど通知の処理が早く行われますが、システムのオーバーヘッドが大きくなる場合があります。既定値を削除して値をブランクのままにすると、ポーリング期間はデフォルトで短時間 (通常は 1 分未満) に設定されます。

デフォルト値

`5` (秒)

notifyEventMonitorRemoveSize

説明

1 回でキューから削除するイベント数を指定します。イベント・モニターは、イベント・キューからイベントを、この値で指定された数ずつキューが空になるまで削除します。

注: イベント処理のパフォーマンスを向上させるために、この値を 1 以外の数に設定することもできます。ただし、削除されたイベントが処理される前にサービス・ホストがダウンした場合にイベントが失われる恐れがあります。

デフォルト値

`10`

alertCountRefreshPeriodInSeconds

説明

アラート数に関するシステム全体のアラート数リフレッシュ期間 (秒) を指定します。この数は、ユーザーのログイン後にナビゲーション・バーの上部付近に表示されます。

注: マルチユーザー環境では、 リフレッシュ期間を変更してポーリングを高速にすると、パフォーマンスに影響が出る場合があります。

デフォルト値

`180` (3 分)

Unica Plan | umoConfiguration | 通知 | E メール

これらのプロパティーは、Unica Plan における E メール通知についての情報を指定します。

notifyEMailMonitorStartTime

説明

Unica Plan の始動後、初めて E メールモニターが処理を開始する時刻を指定します。値の形式は、現行ロケールの `java.text.DateFormat` クラスのショート・バージョンに従ってください。例えば、米国英語ロケールの場合、有効なストリングとしては `11:59 pm` などが考えられます。

デフォルト値

ブランク (Unica Plan の始動直後)。

notifyEMailMonitorPollPeriod

説明

E メール・モニターがポーリングとポーリングの間にスリープするおおよその時間 (秒) を定義します。

注: イベントと同様に、ポーリング期間とポーリング期間の間、E メール・メッセージはキューに累積されます。 ポーリング時間が短いほど E メール・メッセージが早く送信されますが、システムのオーバーヘッドが大きくなる場合があります。

デフォルト値

`60` (秒)

notifyEMailMonitorJavaMailSession

説明

E メール通知に使用する、既存の初期化済み JavaMail™ セッションの JNDI 名。これが未指定であり、委任が「完了」とマークされている場合は、Unica Plan がセッションを作成できるように JavaMail™ ホストパラメーターを指定する必要があります。

デフォルト値

blank

notifyEMailMonitorJavaMailProtocol

説明

E メール通知に使用するメール・サーバー・トランスポート・プロトコルを指定します。

デフォルト値

`smtp`

notifyEMailMonitorRemoveSize

説明

1 回にキューから削除する E メール・メッセージ数を指定します。E メール・モニターは、E メール・キューからメッセージを、この値で指定された数ずつ削除し、これをキューが空になるまで続けます。

注: E メール処理のパフォーマンスを向上させるために、この値を 1 以外の数に設定することもできます。ただし、削除された E メール・メッセージが処理される前にサービス・ホストがダウンした場合、メッセージが失われる恐れがあります。

デフォルト値

`10` (メッセージ)

notifyEMailMonitorMaximumResends

説明

最初の送信試行が失敗した E メール・メッセージの送信を試行する最大回数を指定します。送信が失敗した場合、E メールは、このパラメーターで許可される最大試行回数に到達するまでキューに戻されます。

例えば、**notifyEMailMonitorPollPeriod** が 60 秒ごとにポーリングするよう設定されているとします。**notifyEMailMonitorMaximumResends** プロパティーを試行回数 60 に設定すると、E メールモニターは失敗したメッセージをポーリングごと (つまり毎分) に 1 回、最大 1 時間、再試行を試みます。値 1440 (24x60) を設定した場合、E メール・モニターは、1 分間隔で最大 24 時間試行します。

デフォルト値

`1` (試行)

showUserNameInEmailNotificationTitle

説明

Unica Plan 通知およびアラートシステムで、E メール通知の**「差出人」**フィールドにユーザー名を入れるかどうかを指定します。

注: この設定は、Unica Plan の通知およびアラートシステムによって送信される E メールメッセージにのみ適用されます。

有効な値

- `True`: Unica Plan はメッセージタイトルの後ろにユーザー名を追加し、その両方を E メールの**「差出人」**フィールドに表示します。
- `False`: Unica Plan はメッセージタイトルのみを**「差出人」**フィールドに表示します。

デフォルト値

偽

notifyEMailMonitorJavaMailDebug

説明

JavaMail™ デバッグモードを設定するかどうかを指定します。

有効な値

- `True`: JavaMail™ デバッグを有効にします。
- `False`: デバッグトレースを無効にします。

デフォルト値

偽

Unica Plan | umoConfiguration | 通知 | プロジェクト

これらのプロパティーは、Unica Plan におけるプロジェクトアラームについての情報を指定します。

notifyProjectAlarmMonitorStartTime

説明

Unica Plan の始動後、初めてプロジェクトアラームモニターが実行される時刻を指定します。値の形式は、現行ロケールの `java.text.DateFormat` クラスのショート・バージョンに従ってください。例えば、米国英語ロケールの場合、有効なストリングとしては `11:59 pm` などが考えられます。デフォルトを削除し、値をブランクのままにすると、このモニターは、作成された直後に開始します。

デフォルト値

```
22:00 PM
```

notifyProjectAlarmMonitorPollPeriod

説明

プロジェクト・アラーム・モニターおよびプログラム・アラーム・モニターがポーリングとポーリングの間にスリープするおおよその時間 (秒) を定義します。

デフォルト値

ブランク (60 秒)

notifyProjectAlarmMonitorScheduledStartCondition

説明

プロジェクトの開始日の何日前に、Unica Plan がユーザーに通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`1` (日)

notifyProjectAlarmMonitorScheduledEndCondition

説明

プロジェクトの終了日の何日前に、Unica Plan がユーザーに終了通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

notifyProjectAlarmMonitorTaskScheduledStartCondition

説明

タスクの開始日の何日前に、Unica Plan がユーザー開始通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`1` (日)

notifyProjectAlarmMonitorTaskScheduledEndCondition

説明

タスクの終了日の何日前に、Unica Plan がユーザーに終了通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

notifyProjectAlarmMonitorTaskLateCondition

説明

タスクの開始日の何日後に、Unica Plan がユーザーに、タスクが開始しなかったことを示す通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

notifyProjectAlarmMonitorTaskOverdueCondition

説明

タスクの終了日の何日後に、Unica Plan がユーザーに、タスクが終了しなかったことを示す通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

notifyProjectAlarmMonitorTaskScheduledMilestoneCondition

説明

マイルストーンタスクの開始日の何日前に、Unica Plan が通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`1` (日)

Unica Plan | umoConfiguration | 通知 | projectRequest

これらのプロパティーは、Unica Plan におけるプロジェクト要求アラームについての情報を指定します。

notifyRequestAlarmMonitorLateCondition

説明

要求が遅れているという通知を Unica Plan が送信する日数を定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

notifyRequestAlarmMonitorScheduledEndCondition

説明

要求の終了日の何日前に、Unica Plan がユーザーに終了通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`1` (日)

Unica Plan | umoConfiguration | 通知 | プログラム

このカテゴリーのプロパティーは、プログラム通知スケジュール用のオプションを指定します。

notifyProgramAlarmMonitorScheduledStartCondition

説明

プログラムの開始日の何日前に、Unica Plan がユーザーに開始通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`1` (日)

notifyProgramAlarmMonitorScheduledEndCondition

説明

プログラムの終了日の何日前に、Unica Plan がユーザーに終了通知を送信するかを定義します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

Unica Plan | umoConfiguration | 通知 | marketingObject

これらのプロパティーは、Unica Plan におけるマーケティングオブジェクトアラームについての情報を指定します。

notifyComponentAlarmMonitorScheduledStartCondition

説明

マーケティングオブジェクトの開始日の何日前に、Unica Plan がユーザーに開始通知を送信するかを指定します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`1` (日)

notifyComponentAlarmMonitorScheduledEndCondition

説明

マーケティングオブジェクトの終了日の何日前に、Unica Plan がユーザーに終了通知を送信するかを指定します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

Unica Plan | umoConfiguration | 通知 | 承認

これらのプロパティーは、Unica Plan における承認アラームについての情報を指定します。

notifyApprovalAlarmMonitorStartTime

説明

Unica Plan の始動後、初めて承認アラームモニターが処理を開始する時刻を指定します。値の形式は、現行ロケールの `java.text.DateFormat` クラスのショート・バージョンに従ってください。例えば、米国英語ロケールの場合、有効なストリングとしては `11:59 pm` などが考えられます。デフォルトを削除し、この値をブランクのままにすると、モニターは、作成された直後に開始します。

注: 最良の結果を得るためには、アラーム・モニターの開始をオフピーク時間帯にし、それぞれのモニターの開始時刻をずらすように構成して、データ処理のロードを分散します。

デフォルト値

`9:00 pm`

notifyApprovalAlarmMonitorPollPeriod

説明

承認アラーム・モニターがポーリングとポーリングの間にスリープするおおよその時間 (秒) を指定します。

デフォルト値

ブランク (60 秒)

notifyApprovalAlarmMonitorLateCondition

説明

承認の開始日の何日後に、システムがユーザーに承認が遅れていることを通知し始めるかを指定します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`3` (日)

notifyApprovalAlarmMonitorScheduledEndCondition

説明

承認の終了日の何日前に、システムが終了通知をユーザーに送信し始めるかを指定します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`1` (日)

Unica Plan | umoConfiguration | 通知 | 資産

これらのプロパティーは、Unica Plan における資産アラームについての情報を指定します。

notifyAssetAlarmMonitorStartTime

説明

Unica Plan の始動後、初めて資産アラームモニターが処理を開始する時刻を指定します。値の形式は、現行ロケールの `java.text.DateFormat` クラスのショート・バージョンに従ってください。例えば、米国英語ロケールの場合、有効なストリングとしては `11:59 pm` などが考えられます。デフォルトを削除し、この値をブランクのままにすると、モニターは、作成された直後に開始します。

注: 最良の結果を得るためには、アラーム・モニターの開始をオフピーク時間帯にし、それぞれのモニターの開始時刻をずらすように構成して、データ処理のロードを分散します。

デフォルト値

`11:00 pm`

notifyAssetAlarmMonitorPollPeriod

説明

資産アラーム・モニターがポーリングとポーリングの間にスリープする時間 (秒) を指定します。

デフォルト値

ブランク (60 秒)

notifyAssetAlarmMonitorExpirationCondition

説明

資産が期限切れになる何日前に、Unica Plan がユーザーに対して資産がもうすぐ期限切れになることを通知するかを指定します。

注: この値が `-1` の場合、Unica Plan は有効期限をチェックしません。

デフォルト値

`5` (日)

Unica Plan | umoConfiguration | 通知 | 請求書

これらのプロパティーは、Unica Plan における請求書アラームについての情報を指定します。

notifyInvoiceAlarmMonitorStartTime

説明

Unica Plan の始動後、初めて請求書アラームモニターが処理を開始する時刻を指定します。値の形式は、現行ロケールの `java.text.DateFormat` クラスのショート・バージョンに従ってください。例えば、米国英語ロケールの場合、有効なストリングとしては `11:59 pm` などが考えられます。デフォルトを削除し、値をブランクのままにすると、モニターは、作成された直後に開始します。

注: 最良の結果を得るためには、アラーム・モニターの開始をオフピーク時間帯にし、それぞれのモニターの開始時刻をずらすように構成して、データ処理のロードを分散します。

デフォルト値

`21:00 PM`

notifyInvoiceAlarmMonitorDueCondition

説明

期日の何日前に、Unica Plan がユーザーに対して請求書の期日が近づいていることを通知するかを指定します。

注: この値が `-1` の場合、Unica Plan はこれらの通知を送信しません。

デフォルト値

`5` (日)

Unica Plan | umoConfiguration | 通知 | 不在中

notifyOutOfOfficeMonitorStartTime

説明

Plan ユーザーの不在中ステータスのモニターを開始する時間。

デフォルト値

-

notifyOutOfOfficeMonitorPollPeriod

説明

Plan ユーザーの不在中ステータスを再確認する間隔 (秒)。例えば、値を `600` に設定すると、システムは `10` 分ごとに Plan ユーザーの不在中ステータスを確認します。

デフォルト値

`1800`