Unica JourneyV12.1.1インストールガイド

Contents

Chapter 1. インストールの概要..... 1
インストーラーの仕組み.....1
インストールのモード.....1
Chapter 2. Unica Journeyのインストールを計画する..... 3
前提条件.....3
展開図..... 7
Unica 製品のインストール順序..... 7
Unica Journey インストール・ワークシート.....9
のインストール順序Unica Journey..... 13
Journey Oracle 12Cデータベースサポート.....13
Chapter 3. Unica Journeyのデータソースを作成する..... 15
Web アプリケーション・サーバーでの JDBC 接続の作成..... 16
JDBC ドライバーを使用できるように Web アプリケーション・サーバーを構成する.....16
JDBC 接続を作成するための情報..... 18
Chapter 4. インストールUnica Journey.....22
Unica Journeyコンポーネント.....23
GUI モードを使用した Unica Journey のインストール.....23
コンソールモードによるUnica Journey のインストール..... 29
インストールUnica Journey.....
Chapter 5. 構成Unica Journey..... 31
Unica Journeyプロパティを構成する.....31
開始と検証のUnica Journeyインストール.....37
Unica 製品との統合のためのプロパティーの設定.....38
Journey Proxy 統合.....39
データベースの変更..... 40
Chapter 6. Unica Journey アプリケーションのデプロイメント..... 42
Apache Tomcat アプリケーションサーバーにUnica Journey を配置する。..... 42
WebSphere上にUnica Journeyを展開するためのガイドライン.....44
JBossにUnica Journeyをデプロイするためのガイドライン..... 47
Chapter 7. のアンインストールUnica Journey..... 49

Chapter 1. インストールの概要

のインストールHCL Unicaをインストール、構成、および展開すると、製品HCL Unicaは完成します。インストールガイドには、製品のインストール、構成、および展開に関する詳細情報が記載されています。

インストーラーの仕組み

Unica製品をインストールまたはアップグレードする際には、スイートインストーラーと製品インストーラーを使用する必要があります。例えば、Unica Journeyをインストールする場合、ユニカスイートインストーラーとUnica Journey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必ずご確認ください。

- Unicaのインストーラーと製品のインストーラーは、製品をインストールするコンピュータの同じディレクトリにある必要があります。Unicaインストーラーがあるディレクトリに複数のバージョンの製品インストーラーが存在する場合、Unicaインストーラーはインストールウィザードの Unica 製品画面に常に最新バージョンの製品を表示します。
- Unica製品をインストールした直後にパッチをインストールする場合、パッチインストーラがスイートや製品のインストーラと同じディレクトリにあることを確認してください。
- Unicaのインストールにおけるデフォルトのトップレベル・ディレクトリは、/HCL/Unica for UNIX™またはC:\HCL\Unica for 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。
- Unica Journeyのアップグレードを開始する前に、Unica Marketing Platformを12.1.1まで完了していることをご確認ください。

インストールのモード

ユニカスイートのインストーラは、以下のいずれかのモードで実行することができます。GUIモード、X Window Systemモード、コンソールモード、サイレントモード(無人モードとも呼ばれる)のいずれかで実行できます。Unica Journeyをインストールする際に、要件に合ったモードを選択します。

GUI X Window Systemモード

WindowsではGUIモード、UNIXではX Window Systemモードを使用し、グラフィカルユーザーインターフェースを用いてUnica Journeyをインストールします。

UNIX X Window Systemモード

UNIXのX Window Systemモードを使用し、グラフィカルユーザーインターフェースを使用してUnica Journeyをインストールします。

コンソール・モード

コンソールモードを使用して、コマンドラインウィンドウを使用してUnica Journeyをインストールします。

Note: 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。

Chapter 2. Unica Journeyのインストールを計画する

Unica Journeyを計画する際には、システムを正しくセットアップし、障害に対処できるように環境を構成しておく必要があります。

前提条件

Unica Journey 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

システム要件については、「推奨ソフトウェア環境と最低システム要件」ガイドをご覧ください。

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。

Note: Unica JourneyおよびUnica Link のインストールは、アプリケーションの URL にドメイン名を指定して行う必要があります。

JVM の要件

Unicaスイート内のアプリケーションは、専用のJava™仮想マシン（JVM）にデプロイ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によって使用される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、operating systems (OS)、データベース、Kafka、Webアプリケーションサーバーに関する知識も含まれます。

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。

  Note: 管理者は、テーブルとビューの両方について、`CREATE`、`SELECT`、`INSERT`、`UPDATE`、`DELETE`、`DROP`の権限を持っている必要があります。

- WebアプリケーションサーバーとUnicaコンポーネントを実行するため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とサブディレクトリへの読み取りと書き込みのアクセス権。
- 編集が必要なすべてのファイルに対して書き込み権限を与える。
- アップグレードする場合は、インストールディレクトリやバックアップディレクトリなど、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、製品のインストーラーファイルはすべて、`rwxr-xr-x`のようなフルパーミッションである必要があります。

UNIXの™場合は、さらに以下のパーミッションが必要です。

- Unica Journey および Unica Platform をインストールするユーザー・アカウントは、Unica Journey 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 製品のインストーラーファイルはすべて、`rwxr-xr-x` などのフルパーミッションである必要があります。

導入前の注意点Unica Journey

Unica Journey インストールには、以下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でJAVA_HOME環境変数が定義されている場合、その変数がサポートされているバージョンの JRE を指していることを確認します。システム要件については、『推奨ソフトウェア環境と最低システム要件』ガイドをご覧ください。

JAVA_HOME環境変数が不正なJREを指している場合、Unica インストーラーを実行する前に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で、環境変数JAVA_HOMEをクリアすることができます。

- Windowsの™場合: コマンドウィンドウで、set JAVA_HOME= (空のまま)と入力し、**Enter**キーを押します。
- UNIX™： ターミナルで、export JAVA_HOME= (空のまま)と入力し、**Enter**キーを押します。

ターミナルで以下のコマンドを実行すると、環境変数JAVA_HOMEをクリアすることができます。

export JAVA_HOME= (空のまま)

Unica インストーラーは、Unica インストール環境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のインストー

ラーによってインストールされるJREの場所を指します。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。

対応するJREの詳細については、「推奨ソフトウェア環境と最小システム要件」ガイドを参照してください。

Unica Platform要件

ユーザーは、ジャーニーのインストールとアップグレードの前に、サポートベースのUnica Platformバージョンをインストール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。Unica Platform お使い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よう促す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を設定する前に、デプロイして実行する必要があります。

PlatformとJourneyは異なるサーバーにインストールすることができます。そのような場合、Platformを別のサーバーにインストールし、JourneyアプリケーションがPlatformのURLにアクセスできるようにする必要があります。Journeyホストは、Unicaアプリケーションポートを介してプラットフォームホストと通信できるようにする必要があります。

Table 1. Journeyサポートインストールパス

ベースJourneyバージョン	アップグレード・パス	実行するタスク
Unica Journey 12.1.0 または 12.1.0.x (Oracle, MS SQL Server, OneDB, MariaDB 上のシステムテーブルを使用)	Unica Journey 12.1.1へのインプレースアップグレード。	1. Unica Marketing Platformを12.1.1にアップグレードする。 2. Unica Journeyを12.1.1にアップグレードするインストーラーを実行します。 3. Journeyアプリケーションの設定 4. Journeyアプリケーションのデプロイ 5. Journeyアプリケーション実行します
OneDB、MariaDB、SQL Serverなどのシステムテーブルがある場合、既存のUnica環境にJourneyをクリーンインストールします。	Unica Journey 12.1.1へのインプレースアップグレード。	1. ジャーニーを除く、Unica Marketing Platformおよび必要なUnica製品を12.1.1にアップグレードします。 2. Unica Journey 12.1.1のクリーンインストーラを実行します。 3. Journeyアプリケーションの設定

Table 1. Journeyサポートインストールパス (continued)

		4. Journeyアプリケーションのデプロイ 5. Journeyアプリケーション実行します

Note:

1. Oracleデータベースを持つ既存のUnica環境にJourneyがインストールされていない場合、Journeyバージョン12.1.0をインストールした後、Unica Journeyを12.1.1にアップグレードする必要があります。
2. MS SQL Server、OneDB、MariaDBのいずれかのデータベースを持つ既存のUnica環境にJourneyがインストールされていない場合、Journeyバージョン12.1.1をクリーンインストールとして直接インストールすることが可能です。

性能タブが効率的に動作するために、ユーザーはJourneyシステムDB権限をレポートDBユーザーに提供する必要があり、逆も同様です。

MariaDBの 場合は、以下のコマンドを使用します。

Journey_SystemDB}.* 上のすべての権限を '{Journey_Reports_User}'@'%' （'{Journey_Reports_User_Password}' で識別）に付与してください。

GRANT ALL ON {Journey_SystemDB}.* TO '{Journey_Reports_User}'@'%';

Journey_SystemDB_User_Password}' で識別される '{Journey_SystemDB_User}'@'%' に {Journey_ReportsDB}.* 上のすべての特権を付与してください。

GRANT ALL ON {Journey_ReportsDB}.* TO '{Journey_SystemDB_User}'@'%';

Oracleの場合

Oracleデータベースの場合、システムユーザアカウントとReportスキーマを作成するためのReportユーザを作成します。システム・ユーザー・アカウントには、以下の権限がなければなりません。

- CREATE TABLES
- CREATE VIEWS (レポート用)
- CREATE SEQUENCE (Oracle のみ)
- CREATE INDICES
- ALTER TABLE
- INSERT
- UPDATE
- DELETE

Note: レポートユーザーも上記の権利を有します。また、レポートユーザーは、システムユーザーにレポートスキーマテーブルへのアクセス権限を付与する必要があります。以下のコマンドを実行します。

GRANT ALL PRIVILEGES TO (SYSTEM_SCHEMA_USER_NAME)

詳しくは、Unica Journeyのデプロイメント（30ページ on page 42）をご覧ください。

Journeyのための分散環境。

Journeyエンジンのファイルの場所は、JourneyエンジンとWebマシンで共有する必要があります。Journeyエンジンが複数のマシンにインストールされている場合、このファイルディレクトリはすべてのマシンで同じパスに共有/マウントされる必要があります。

展開図

Unica Journeyを含むUnicaアプリケーションの展開図を以下に示します。Unica Journeyは、Unicaスイートの他の製品に使用されているUnica Platformの上にインストールする必要があります。

Unica Journeyには、以下の部品があります。

1. Unica Journey Web
2. Unica Journey Engine
3. 基礎となる通信に使用されるKafkaインスタンス。Kafkaインスタンスにはkafkaサーバーとzookeeperがあります。

現在、Journey Webはスタンドアロンデプロイメントとしてのみサポートされています。Journey Engineは、パフォーマンス要件に応じて複数のマシンにインストールすることが可能です。

Unica Journey WebとEngineのコンポーネントは、同じマシンまたは異なるマシンにデプロイすることができます。Unica Journey Web、Engine、Kafkaのコンポーネントは別々のマシンにインストールすることをお勧めします。

Unica 製品のインストール順序

複数インストールまたはアップグレードする場合Unica製品を特定の順序でインストールする必要があります。

次の表には、複数の Unica 製品をインストールまたはアップグレードするときに従う必要のある順序についての情報が示されています。

Table 2. Unica 製品のインストールまたはアップグレードの順序

製品または組み合わせ:	インストールまたはアップグレードの順序:
Unica Campaign（有無Unica Deliver) および Unica Optimize	1. Unica Platform 2. Unica Campaignおよびユニカ最適化 **Note:** バージョン 12.1 のクラスター モードの Campaign では、Deliver はサポートされていません。 **Note:** Unica Deliverインストール時に自動的にインストールされますUnica Campaign.ただし、Unica Deliver が Unica Campaign インストール・プロセス中に構成されたり有効にされたりすることはありません。
Unica Interact	1. Unica Platform 2. Unica Campaign 3. Unica Interactデザインタイム環境 4. Unica Interact実行時環境 5. Unica Interactエクストリーム スケール サーバー Interact 設計時環境だけをインストールまたはアップグレードする場合、 Interact 設計時環境を以下の順序でインストールまたはアップグレードします。 1. Unica Platform 2. Unica Campaign 3. Unica Interactデザインタイム環境 Interact ランタイム環境だけをインストールまたはアップグレードする場合、 Interact ランタイム環境を以下の順序でインストールまたはアップグレードします。 1. Unica Platform 2. Unica Interact実行時環境 Unica Interact Extreme Scale サーバーだけをインストールする場合、 Unica Interact Extreme Scale サーバーを以下の順序でインストールします。

Table 2. Unica 製品のインストールまたはアップグレードの順序

(continued)

製品または組み合わせ:	インストールまたはアップグレードの順序:
	1. Unica Platform 2. Unica Interact実行時環境 3. Unica Interactエクストリーム スケール サーバー
Unica Plan	1. Unica Platform 2. Unica Plan **Note:** Unica Plan を Unica Campaign に統合する場合、Unica Campaign もインストールする必要があります。それら 2 つの製品は任意の順序でインストールできます。
Unica Interact Advanced Patterns	1. Unica Platform 2. Unica Campaign 3. Unica Interact 4. Unica Interact Advanced Patterns
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	1. IBM SPSS Modeler Advantage Enterprise Marketing Management Edition
Unica Journey	1. Unica Platform 2. Unica Journey

Unica Journey インストール・ワークシート

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

次のテーブルを使用して、Unica Journey システムテーブル用に作成された空のデータベースに関する情報を収集します。Unica Journey に設定する空のデータベースは、任意の名前を付けることができます。

Table 3. 対応なデータベース

フィールド	メモ®
データベース・タイプ	

Table 3. 対応なデータベース

(continued)

フィールド	メモ®
データベース名	
データベース・アカウント・ユーザー名	
データベース・アカウント・パスワード	
JNDI 名	JourneyDS, JourneyReportDS
ODBC 名	

Table 4. Kafkaインスタンスに関する情報

フィールド	メモ
Kafkaサーバーのホスト	
Kafkaサーバーのポート	
Kafkaサーバー証明書（KafkaがSSLを有効にしている場合）	
Kafkaサーバー - ユーザーID（Kafka接続がSASLプレーンテキストの場合）。	
Kafkaサーバー - ユーザーパスワード（Kafka接続がSASLプレーンテキストの場合）。	

Oracle

- データベースドライバ:`oracle.jdbc.OracleDriver`
- デフォルトのポート: 1521
- ドライバークラス: `oracle.jdbc.OracleDriver`
- ドライバURL: `"jdbc:oracle:thin:@<Host>:<Port>:<SID_NAME>"`

```
<?xml version="1.0"?> <Context docBase="<Journeys_Install_Path>/Web/journey.war">
 <Environment name="journey.web.home" value="<Journeys_Install_Path>/Web/"
 type="java.lang.String"/> <Resource name="JourneyDS" type="javax.sql.DataSource"
 factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
 maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
```

```
driverClassName="oracle.jdbc.OracleDriver" url="jdbc:oracle:thin:@<Host>:<Port>:<SID_NAME>"/>
<Resource name="JourneyReportDS" type="javax.sql.DataSource"
factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
driverClassName="oracle.jdbc.OracleDriver" url="jdbc:oracle:thin:@<Host>:<Port>:<SID_NAME>"/>
</Context>
```

SQL サーバー

- データベースドライバ:`com.microsoft.sqlserver.jdbc.SQLServerDriver`
- デフォルトのポート: 1433
- ドライバークラス: `com.microsoft.sqlserver.jdbc.SQLServerDriver`
- ドライバURL: `jdbc:sqlserver://<your_db_host>\\<named_instance>:<your_db_port>;databaseName=<your_db_name>`
- プロパティを追加します。`user=<ユーザー名>`の追加

```
<?xml version="1.0"?> <Context docBase="<Journeys_Install_Path>/Web/journey.war">
<Environment name="journey.web.home" value="<Journeys_Install_Path>/Web/"
type="java.lang.String"/> <Resource name="JourneyDS" type="javax.sql.DataSource"
factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
url="jdbc:sqlserver://<your_db_host>\\<named_instance>:<your_db_port>;databaseName=<your_db_name>"/>
<Resource name="JourneyReportDS" type="javax.sql.DataSource"
factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
url="jdbc:sqlserver://<your_db_host>\\<named_instance>:<your_db_port>;databaseName=<your_db_name>"/>
</Context>
```

OneDBデータベース

- データベースドライバ:`com.informix.jdbc.IfxDriver`
- デフォルトのポート: 9088 <ユーザー定義のデータベースポート>。
- ドライバクラス: `javax.sql.DataSource`
- ドライバのURLです。`jdbc:Informix-sqli://host:port/database_name:informixserver=servername;`
- プロパティを追加します。`user=<ユーザー名>`の追加

```
<?xml version="1.0"?> <Context docBase="<Journeys_Install_Path>/Web/journey.war">
<Environment name="journey.web.home" value="<Journeys_Install_Path>/Web/"
type="java.lang.String"/> <Resource name="JourneyDS" type="javax.sql.DataSource"
factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory"
maxActive="30" maxIdle="10" maxWait="10000" username="<your_db_user_name>"
password="<your_db_user_password>" driverClassName="javax.sql.DataSource"
url="jdbc:Informix-sqli://host:port/<database_name>:informixserver=<servername>"/>
<Resource name="JourneyReportDS" type="javax.sql.DataSource"
factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory"
maxActive="30" maxIdle="10" maxWait="10000" username="<your_db_user_name>"
password="<your_db_user_password>" driverClassName="javax.sql.DataSource"
url="jdbc:Informix-sqli://host:port/<database_name>:informixserver=<servername>"/> </Context>
```

MariaDBデータベース

- データベースドライバ:`org.mariadb.jdbc.Driver`
- デフォルトのポート: 3306
- ドライバクラス： org.mariadb.jdbc.Driver
- ドライバURL: `="jdbc:mariadb://host:port/<DB_USER_NAME>"`
- プロパティを追加します。`user=<ユーザー名>`の追加

```
<?xml version="1.0"?> <Context docBase="<Journeys_Install_Path>/Web/journey.war">
 <Environment name="journey.web.home" value="<Journeys_Install_Path>/Web/"
 type="java.lang.String"/> <Resource name="JourneyDS" type="javax.sql.DataSource"
 factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
 maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
 driverClassName="org.mariadb.jdbc.Driver" url="jdbc:mariadb://host:port/<DB_USER_NAME>"/>
 <Resource name="JourneyReportDS" type="javax.sql.DataSource"
 factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
 maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
 driverClassName="org.mariadb.jdbc.Driver" url="jdbc:mariadb://host:port/<DB_USER_NAME>"/> </Context>
```

Unica Platform データベースのチェックリスト

各 Unica 製品のインストール・ウィザードは、製品を登録するために、Unica Platform システム・テーブル・データベースと通信可能でなければなりません。インストーラーを実行するたびに、 Unica Platform システム・テーブル・データベースの以下のデータベース接続情報を入力する必要があります。

- JDBC 接続 URL
- データベース・ホスト名
- データベースポート
- データベースの名前またはスキーマ ID
- データベース・アカウントのユーザー名とパスワード

Web アプリケーションサーバーへのUnica Platform の展開に関するチェックリスト

Unica Platform を配置する前に、以下の情報を入手してください。

- **プロトコル**: HTTP、またはWebアプリケーションサーバーにSSLが実装されている場合はHTTPS。
- **ホスト**: Unica Platform がデプロイされるマシンの名前。
- **ポート**： Webアプリケーションサーバーがリッスンするポート。
- **ドメイン名**： HCL製品がインストールされている各マシンの会社ドメイン。例えば、`example.com`。すべてのHCL製品は、同じ会社のドメインにインストールする必要があり、ドメイン名はすべて小文字で入力する必要があります。

  ドメイン名の入力に不一致があると、Unica Platform の機能を使用しようとした場合や、製品間を移動しようとした場合に、問題が発生することがあります。製品を配備した後にドメイン名を変更するには、ログインして、**[設定] > [構成]**ページの製品ナビゲーションカテゴリで関連する構成プロパティの値を変更することができます。

Unica Journey インストールのためのチェックリスト

Unica Journey の各コンポーネントをインストールするには、以下の情報を入手してください。

- hostname -Journey Web アプリケーションがインストールされるシステムの名前です。
- アプリケーション・サーバーが listen するポート。SSL の実装を計画している場合は、SSL ポートを入手します。
- 配置システムのネットワーク・ドメイン。例えば、`mycompany.com`。

のインストール順序Unica Journey

複数のUnica 製品をインストールする場合、特定の順序でインストールする必要があります。

次の表は、Unica Journey をインストールする際に従わなければならない順序についての情報です。

Table 5. のインストール順序Unica Journey

商品	この順番でインストールします。
Unica Journey	1. Unica Platform 2. Unica Journey

Note: Unica Journeyは、3つのコンポーネントをインストールします。

- Unica JourneyWebアプリケーション --Unica Journey Webアプリケーションは、Oracle、SQL サーバー、OneDB、MariaDBなど、サポートされているアプリケーションサーバーにデプロイすることができます。
- Unica Journeyエンジン：アプリケーションサーバーへのデプロイは不要で、Journey エンジンはスタンドアロンアプリケーションとしてコマンドライン/ターミナルから起動することができます。
- Apache Kafka。Kafkaサーバーとzookeeperは一緒にインストールされ、コマンドラインまたはターミナルで起動できます。Unica Journey 3つのコンポーネントは、同じマシンまたは異なるマシンにインストールすることができます。

Journey Oracle 12Cデータベースサポート

Oracle 12C (12.1.0.2) での Journey Upgrade 12.1 > 12.1.0.3

JOURNEY 12.1.0.3 を Oracle 12C でアップグレードする場合、まず Journey Web と Journey report のデータベーススキーマをクリーンにして、12.1.0.3 スクリプト（Hotfix として提供）を手動で実行します。

Oracle 12C (12.1.0.2) 上の Journey Upgrade 12.1 > 12.1.0.4

Oracle 12CでJourney 12.1.0.4にアップグレードしたい場合、まずバージョン12.1.0.3へのアップグレードが必要です。Journey WebとJourneyレポートのデータベーススキーマをクリーンアップし、12.1.0.3スクリプト（Hotfixとして提供）を手動で実行してから、12.1.0 FP4インストーラを実行してください。

Note: Hotfixについては、サポートにお問い合わせください。

Oracle 12C (12.1.0.2) での Journey Upgrade 12.1.0.4 > 12.1.1

Oracle 12CでJourney 12.1.1.0にアップグレードしたい場合、まず12.1.0から12.1.0.3にバージョンアップする必要があります。Journey WebとJourneyレポートのデータベーススキーマをクリーンアップし、12.1.0.3を実行します。スクリプト（Hotfixとして提供）を手動で実行した後、12.1.0 FP4インストーラーを実行します。

制限

Oracle 12Cでは、オブジェクト名は30文字を超えてはいけないという制限があります。

Chapter 3. Unica Journeyのデータソースを作成する

Unica Journeyをインストールする前に、Unica Journeyデータソースを作成する必要があります。以下のステップを完了し、Unica Journey のデータソースを準備します。

1. Unica JourneyおよびJourneyレポートシステムテーブルのデータベースまたはデータベーススキーマを作成します。
   次の表は、Journeyシステムテーブルのデータベースまたはデータベーススキーマを作成するためのベンダー固有のガイドラインに関する情報を提供します。

   Table 6. データ・ソース作成のためのガイドライン

データベース・ベンダー	ガイドライン
Oracle	環境を開くために自動コミット機能を有効にしてください。Oracle 資料の説明を参照してください。
MariaDB	Lower_case_table_names を 1 にすると、テーブル名の大文字と小文字が区別されない。wait_timeout=<`接続がアクティブになるまでサーバーが待機する時間（秒）を設定し、接続を閉じます。セッションの値は、スレッドの起動時に、非インタラクティブ接続の場合はグローバル値から、インタラクティブ接続の場合は`interactive_timeout`値から初期化`されます。 > 例: 30日間非アクティブに設定できる場合は 25,92,000（秒）に設定 max_connections=<`同時クライアント接続数の最大値`>。
SQL サーバー	プラットフォームにはSQL サーバー認証が必要なため、SQL サーバー認証のみ、またはSQL サーバー認証とWindows™認証の両方を使用します。必要であれば、データベース認証にSQL サーバーが含まれるように、データベース構成を変更します。また、SQL サーバー で TCP/IP を必ず有効にしてください。

   Note: マルチバイト文字 (中国語、韓国語、日本語など) を使用するロケールを使用可能にする予定の場合、それらをサポートするようデータベースが作成されていることを確認してください。

2. システム・ユーザー・アカウントを作成します。
   システム・ユーザー・アカウントには、以下の権限がなければなりません。
   - CREATE TABLES
   - CREATE VIEWS (レポート用)
   - CREATE SEQUENCE (Oracle のみ)
   - CREATE INDICES
   - ALTER TABLE
   - INSERT

    - UPDATE
    - DELETE
3. ODBC 接続またはネイティブ接続を作成します。
4. ご使用の JDBC ドライバー用に Web アプリケーション・サーバーを構成します。
5. Web アプリケーション・サーバーで JDBC 接続を作成します。

Web アプリケーション・サーバーでの JDBC 接続の作成

About this task

Unica Journeyウェブアプリケーションは、JDBC接続を使用して、そのシステムテーブルデータベースと通信できる必要があります。

このJDBC接続は、Unica Journeyをデプロイする予定のWebアプリケーションサーバーで作成する必要があります。

データ・ソースを手動で作成する場合は、以下のガイドラインに従ってください。

- WebSphere®では、このプロセスでデータベース・ドライバーのクラスパスを設定します。
- Unica Journeyシステムテーブルがデータベースログインユーザーのデフォルトスキーマとは異なるスキーマで作成されている場合、システムテーブルにアクセスするために使用するJDBC接続でそのデフォルトではないスキーマ名を指定する必要があります。
- Tomcat では、このプロセスの際に、ご使用のデータベース・ドライバーのクラスパスを設定してください。
- JBOSS では、JDBC ドライバーのモジュールを追加し、SQL JDBC ドライバーを登録することで、ご使用のデータベース・ドライバーのクラスパスを設定します。
- JNDI名には、`JourneyDS`と `JourneyReportDS`を使用する必要があります。この名前は必須であり、Unica Journey インストール・ワークシート on page 9に記載されています。

JDBC ドライバーを使用できるように Web アプリケーション・サーバーを構成する

Unica Journey をデプロイする予定の Web アプリケーション サーバーには、JDBC 接続をサポートするための正しい JAR ファイルが含まれている必要があります。この JAR ファイルによって、Web アプリケーションはシステム・テーブルに接続できます。Web アプリケーション・サーバーのクラスパスに、JAR ファイルの場所を含める必要があります。

WebSphere

About this task

インストーラーによる自動データソース作成は、Journey アプリケーションではサポートされていません。Journey アプリケーションのデータソースを作成するために、Manual ステップを行う必要があります。

以下の手順で、データソースを作成します。

1. WebSphere Admin Consoleにアクセスする
2. WebSphereでデータソースを設定する
3. ウィザードを続行します: JDBC プロバイダーのセットアップ

4. セキュリティ・エイリアスを指定する
5. データソースのテスト

詳しくは、WebSphereのドキュメントを参照してください。

JBoss

About this task

JBoss を使用している場合は、以下の手順をすべて実行する必要があります。

1. 「推奨されるソフトウェア環境および最小システム要件」のガイドの説明に従って、 でサポートされるシステム・テーブル・データベース用にUnica、ベンダー提供の最新のタイプ 4 JDBC ドライバーを入手します。JDBC ドライバーの入手後、以下のガイドラインを使用します。
   - Unica Journeyを展開する予定のサーバーにドライバーが存在しない場合は、ドライバーを入手し、サーバー上で解凍してください。スペースを含まないパスにドライバーを解凍してください。
   - データソースクライアントがインストールされているサーバーからドライバを取得する場合、そのバージョンがUnica Journeyでサポートされている最新版であることを確認します。
2. Unica Journeyを展開する予定のWebアプリケーションサーバーのクラスパスに、ファイル名を含むドライバーのフルパスを追加してください。
   以下のガイドラインを使用してください。
   - サポートされるすべてのバージョンの JBoss で、JDBC ドライバーをモジュールとして追加します。次の手順を使用してJDBCドライバーをモジュールとして追加します.

     例えば、SQL サーバーの場合:

     module add --name=com.microsoft.sqlserver.jdbc --resources=<JDBC_Driver_Location>jmssql-jdbc-7.0.0.jre8.jar --dependencies=javax.api,javax.transaction.api
   - 次のガイドラインを使用して、この SQL JDBC ドライバーを登録します。次に例を示します。
     - `/subsystem=datasources/jdbc-driver=sql:add(driver-module-name=com.microsoft.sqlserver.jdbc,driver-name=sql,driver-xa-datasource-class-name=com.microsoft.sqlserver.jdbc.SQLServerXADataSource)`
     - `/subsystem=datasources/jdbc-driver=sql:read-resource`
     - `/subsystem=ee/service=default-bindings:write-attribute(name=datasource,value=undefined)`
3. インストーラーを実行するときにパスを入力する必要があるため、Unica Journey インストール ワークシートのデータベース ドライバー クラス パスを書き留めます。
4. 変更内容を有効にするため、Web アプリケーション・サーバーを再始動します。

   始動時にコンソール・ログをモニターして、データベース・ドライバーのパスがクラスパスに含まれていることを確認してください。

Apache Tomcat

About this task

Apache Tomcat を使用している場合は、以下の手順をすべて実行する必要があります。

1. 「推奨されるソフトウェア環境および最小システム要件」のガイドの説明に従って、 でサポートされるシステム・テーブル・データベース用にUnica、ベンダー提供の最新のタイプ 4 JDBC ドライバーを入手します。JDBC ドライバーの入手後、以下のガイドラインを使用します。
   - Unica Journeyを展開する予定のサーバーにドライバーが存在しない場合は、ドライバーを入手し、サーバー上で解凍してください。スペースを含まないパスにドライバーを解凍してください。
   - データ・ソース・クライアントのインストール場所であるサーバーからドライバーを入手する場合、Unica でサポートされる最新バージョンであることを確認してください。
2. Unica Journeyを展開する予定のWebアプリケーションサーバー（<Tomcat_Installed Location>/lib）のクラスパスに、ファイル名を含むドライバーのフルパスを追加してください。
3. データベースドライバのクラスパスは、Unica Journey インストール・ワークシート on page 9 、インストーラを実行するときに入力する必要があるので、メモしておいてください。
4. 変更内容を有効にするため、Web アプリケーション・サーバーを再始動します。

   始動時にコンソール・ログをモニターして、データベース・ドライバーのパスがクラスパスに含まれていることを確認してください。

JDBC 接続を作成するための情報

特定の値が示されない場合は、JDBC 接続の作成時にデフォルト値を使用します。詳しくは、アプリケーション・サーバの資料を参照してください。

Note: データベースのデフォルト・ポート設定を使用しない場合は、正しい値に必ず変更してください。

WebSphere

アプリケーションサーバがWebSphereの場合、以下の値を使用します。

SQLサーバ

- ドライバー: 該当/なし
- デフォルトのポート: 1433
- ドライバークラス: `com.microsoft.sqlserver.jdbc.SQLServerConnectionPoolDataSource`
- ドライバーURL: `jdbc:sqlserver:/`<DBhostName>:1433;databaseName=<DBName>。

データベース タイプフィールドで、 ［User-defined］を選択します。

JDBCプロバイダとデータソースを作成した後、データソースのカスタムプロパティに移動し、以下のようにプロパティを追加、変更します。

- `サーバ名=<your_SQL_server_name>`
- `ポート番号 =<SQL_Server_Port_Number>`
- `データベース名=<your_database_name>`

- 名前: `webSphereDefaultIsolationLevel`
- 値: 1
- データタイプ: インテジャ

Oracle

- ドライバー: オラクルJDBCドライバー
- デフォルトのポート: 1521
- ドライバークラス: `oracle.jdbc.OracleDriver`
- ドライバーURL: `jdbc:oracle:thin:@<your_db_host>:<your_db_port>:<your_db_service_name>`

  示した形式を使用してドライバー URL を入力します。Unica アプリケーションでは、JDBC 接続に対する Oracle の RAC (Real Application Cluster) 形式の使用は許可されていません。

以下のカスタムプロパティを追加します:

- 名前: `webSphereDefaultIsolationLevel`
- 値: 2
- データタイプ: インテジャ

MariaDB

- データベースドライバー: mariadb-java-client-2.5.1.jar
- デフォルトのポート: 3306
- ドライバークラス: `org.mariadb.jdbc.Driver`
- ドライバーURL:`jdbc:mariadb://<your_db_host>>:<<PORT>/<Your_DB_user_name>>`
- プロパティ: ユーザー名を追加します= *<your_db_user_name>*
- プロパティ: ユーザーパスワードを追加します= *<your_db_password>*
- ドライバーモジュール xa-datasource-class= org.mariadb.jdbc.MySQLDataSource

OneDB

- データベースドライバー: onedb-jdbc-complete-8.0.0-SNAPSHOT.jre8.jar
- データベースポート :20195
- ドライバー: Informix JDBC ドライバー
- ドライバークラス:`com.informix.jdbc.IfxDriver`
- ドライバーURL: `jdbc:informix-sqli://<your_db_host>/<your_db_name>:INFORMIXSERVER=<your_db_servername>;`

JBoss

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: `db2jcc4.jar/ojdbc8.jar/mssql-jdbc-7.0.0.jre8.jar.jar`。

アプリケーション・サーバが JBoss の場合、次の値を使用します。

SQLサーバ

- データベースドライバー: Microsoft MS SQL サーバー ドライバー (タイプ 4). バージョン: 2012、2012 SP1およびSP3、2014、2014 SP1、2016 SP1
- デフォルトのポート: 1433
- ドライバークラス: `com.microsoft.sqlserver.jdbc.SQLServerDriver`
- ドライバーURL: `jdbc:sqlserver://<your_db_host>[ \<named_instance>]:<your_db_port>;databaseName=<your_db_name>,valid-connection-checker-class-name =org.jboss.jca.adapters.jdbc.extensions.mssql.MSSQLValidConnectionChecker`

例 :/subsystem=datasources/data-source=UnicaPlatformDS:add(jndi-name="java:/UnicaPlatformDS",connection-url="jdbc:sqlserver://localhost:1433;databaseName=plat11",driver-name=sql,user-name=sa,password=test1234,valid-connection-checker-class-name="org.jboss.jca.adapters.jdbc.extensions.mssql.MSSQLValidConnectionChecker")

Oracle

- ドライバー: オラクルJDBCドライバー
- デフォルトのポート: 1521
- ドライバークラス: `oracle.jdbc.OracleDriver`
- ドライバーURL: `jdbc:oracle:thin:@<your_db_host>:<your_db_port>:<your_db_service_name>`

次に例を示します。

MariaDB

- データベースドライバー: mariadb-java-client-2.5.1.jar
- デフォルトのポート: 3306
- ドライバークラス: `org.mariadb.jdbc.Driver`
- ドライバーURL:`jdbc:mariadb://<your_db_host>>:<<PORT>/<Your_DB_user_name>>`
- プロパティ: ユーザー名を追加します= *<your_db_user_name>*
- プロパティ: ユーザーパスワードを追加します= *<your_db_password>*
- ドライバーモジュール xa-datasource-class= org.mariadb.jdbc.MySQLDataSource

OneDB

- データベースドライバー: onedb-jdbc-complete-8.0.0-SNAPSHOT.jre8.jar
- データベースポート :20195
- ドライバー: Informix JDBC ドライバー
- ドライバークラス:`com.informix.jdbc.IfxDriver`
- ドライバーURL: `jdbc:informix-sqli://<your_db_host>/<your_db_name>:INFORMIXSERVER=<your_db_servername>;`

Tomcat

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: mariadb-java-client-2.5.2.jar/onedb-jdbc-8.0.0.1-complete.jar/ojdbc7.jar/mssql-jdbc-7.0.0.jre8.jar.

アプリケーション・サーバーが Tomcat である場合は、以下の値を使用します。

SQLサーバ

- データベースドライバー: Microsoft MS SQL サーバードライバー (タイプ 4). バージョン: SQL サーバ (e) 2014、2016 SP1、2017、2019
- デフォルトのポート`1433`
- ドライバークラス: `com.microsoft.sqlserver.jdbc.SQLServerDriver`
- ドライバータイプ: `javax.sql.DataSource`
- ドライバーURL:`jdbc:sqlserver://<your_db_host> [\\<named_instance>]:<your_db_port>;データベース名=<your_db_name>`

Oracle

- ドライバー: オラクルJDBCドライバー
- デフォルトのポート: 1521
- ドライバークラス: `oracle.jdbc.OracleDriver`
- ドライバーURL: `jdbc:oracle:thin:@<your_db_host>:<your_db_port>:<your_db_service_name>`

MariaDB

- ドライバー: MariaDB JDBC ドライバー
- デフォルトのポート: 3306
- ドライバークラス: `org.mariadb.jdbc.Driver`
- ドライバー URL: `jdbc:mariadb://<your_db_host>:<PORT>/<Your_DB_user_name>`
- プロパティ: ユーザー名を追加します= *<your_db_user_name>*

OneDB

- データベースドライバー: onedb-jdbc-complete-8.0.0-SNAPSHOT.jre8.jar
- データベースポート :20195
- ドライバー: Informix JDBC ドライバー
- ドライバークラス:`com.informix.jdbc.IfxDriver`
- ドライバーURL: `jdbc:informix-sqli://<your_db_host>/<your_db_name>:INFORMIXSERVER=<your_db_servername>;`

Chapter 4. インストールUnica Journey

Unica のインストールを開始するには、Unica Journey インストーラーを実行する必要があります。Unica インストーラーは、インストール作業中に製品インストーラーを起動します。Unica インストーラーと製品インストーラーが同じ場所に保存されていることを確認してください。

Unica suite のインストーラーを実行するたびに、まずUnica Platform システム・テーブルのデータベース接続情報を入力する必要があります。Unica Journey のインストーラーが起動したら、必要な情報を入力しUnica Journey

Note: Unica Journeytomcat の EAR ファイルを使用した Web アプリケーションのデプロイはサポートされていま せん。

インストール・ファイル

インストールファイルの名前は、製品のバージョンと、UNIXを™除くインストール先のOSに応じた名前になっています。UNIXの™場合、X Window システムモードとコンソールモードとで異なるインストールファイルが存在します。

Exemple

次の表に、製品のバージョンとオペレーティング・システムに従って命名されたインストール・ファイルの例を示します。

Table 7. インストール・ファイル

オペレーティング・システム	インストール・ファイル
Windowsの™場合。GUIおよびコンソールモード	*Product_.N.N.N.N*_win64.exe は、Productは製品名、*N.N.N.N*は製品のバージョン番号、Windows™64bit はこのファイルをインストールする必要があるオペレーティング システムを表します。 *Product_N.N.N.N*_win.exe, ここで、*Product*はお使いの製品の名前、*N.N.N.N*はお使いの製品のバージョン番号、Windows™32 はこのファイルをインストールする必要があるオペレーティングシステムです。
LINUX：X Window システムモード	*Product_N.N.N.N_linux*_linux.bin は、*Product*が製品名、*N.N.N.N*が製品のバージョン番号です。 *Product_N.N.N.N*_linuxrhel64.binは、*Product*はお使いの製品の名前、*N.N.N.N*はお使いの製品のバージョン番号です。
LINUX：コンソールモード	*Product_N.N.N.N*.bin、*Product*は製品名、*N.N.N.N*は製品のバージョン番号です。このファイルは、すべてのUNIX™オペレーティングシステムでのインストールに使用することができます。

Unica Journeyコンポーネント

追加のコンピューター上で Unica Journey ユーティリティーを使用するには、ユーティリティーと Web アプリケーションを対象の追加コンピューターにインストールする必要があります。これは、ユーティリティがWebアプリケーションのjarファイルを使用するために必要です。ただし、ユーティリティーを使用するために Unica Journey をインストールする場合、Unica Journey を再び配置する必要はなく、追加の Unica Journey システム・テーブルを作成する必要もありません。

以下の表に、Unica Journey のインストール時に選択できるコンポーネントを示します。

Table 8. Journeyコンポーネント

コンポーネント	説明
JourneyWebアプリケーション	Journeyウェブコンポーネントは、入力ソース、データ定義、ジャーニーフローを設計するために使用する能力をユーザーに提供します。
Journeyエンジン	Journeyエンジンは、オーディエンスのデータを処理し、タッチポイントにコミュニケーションを送り、レスポンス情報を聞き取り、捕捉します。
Kafkaスタンドアロン	Journeyは、現在のインストール作業と並行して、KafkaとZookeeperのコンポーネントをインストールする予定です。

GUI モードを使用した Unica Journey のインストール

Windows™ の場合、GUI モードを使用してUnica Journey をインストールします。LINUXの場合、X Window システムモードを使用して、Unica Journey をインストールします。

Before you begin

Important: GUI モードを使用してUnica Journey をインストールする前に、Unica Journey をインストールするコンピュータで利用可能な一時的なスペースがUnica Journey のインストーラのサイズの 3 倍以上であることを確認してください。

Unica インストーラーと Unica Journey インストーラー が Unica Journey のインストール先のコンピューターの同じディレクトリーにあることを確認してください。

以下のアクションを実行し、GUI モードで Unica Journey をインストールします。

1. Unica インストーラーを保存したフォルダーに移動して、 インストーラーをダブルクリックして開始します。
2. 最初の画面で**[OK]**をクリックすると、**Introduction**ウィンドウが表示されます。
3. インストーラーの指示に従って、**[次]**をクリックします。
   以下の表にある情報を使用して、Unica インストーラーの各ウィンドウで該当するアクションを実行します。

Table 9. Unicaインストーラー

ウィンドウ	説明
概要	Unica スイートのインストーラーの 最初の画面です。このウィンドウから、Unica Journey のインストール・ガイドとアップグレード・ガイドを開くことができます。また、インストーラーがインストール・ディレクトリーに保存されている製品のインストール・ガイドとアップグレード・ガイドのリンクも表示できます。 **[次]**をクリックします。
レスポンスファイル 送信先	製品のレスポンスファイルを生成する場合は、「**レスポンスファイルを生成する**」チェックボックスをクリックします。応答ファイルには、製品のインストールに必要な情報が保管されています。応答ファイルは、製品の無人インストールのため、または GUI モードでインストーラーを再実行する時に応答欄に設定値をあらかじめ入力しておくために使用できます。 **選択**］をクリックして、応答ファイルを保存する場所を参照します。 **[次]**をクリックします。
Unica製品	「**インストールセット**」リストで、「**カスタム**」を選択して、インストールする製品を選択します。 **インストールセット**]エリアには、お使いのコンピュータの同じディレクトリにインストーラが存在するすべての製品が表示されます。 **説明]**フィールドには、[**インストールセット**]領域で選択した製品の説明が表示されます。 **[次]**をクリックします。
インストール・ディレクトリー	「**インストール先ディレクトリ**の指定」フィールドで、「**選択**」をクリックして、製品をインストールするディレクトリを参照します。 インストーラーが保存されているフォルダーに製品をインストールする場合は、［**既定のフォルダーに戻す**］をクリックします。 **[次]**をクリックします。

ウィンドウ	説明
アプリケーションサーバーを選択します	アプリケーションサーバーの種類を選択します。Journey と共に他の製品をインストールする場合、Platform が配置されるアプリケーションサーバーの種類を選択することができます。 **[次]**をクリックします。
Platform データベースのタイプ	Oracle または OneDB を選択Unica Platformデータベースの種類。 **[次]**をクリックします。
Platform データベース接続	データベースに関する次の情報を入力します。 • データベース・ホスト名 • データベースポート • データベース名またはシステム ID (SID) • データベース・ユーザー名 • データベース・パスワード [**次へ**] をクリックします。
Platform データベース接続 (続き)	JDBC 接続を検討して確認します。 **[次]**をクリックします。必要な場合には、URL を追加パラメーターを使用してカスタマイズできます。
プリインストールのサマリー	インストール・プロセスで追加した値を検討して確認します。 **「インストール」**をクリックして、インストール・プロセスを開始します。 Unica Journey インストーラーが開きます。

4. の指示に従ってくださいUnica Platformインストーラーを使用して Unica Platform をインストールまたはアップグレードします。詳しくは、「 *Unica Platform*インストール・ガイド」を参照してください。
5. **「インストールの完了」**ウィンドウで、**「終了」**をクリックします。

   Result

   Unica Platform のインストールが完了し、Unica Journey のインストーラーが開きます。
6. 以下の表にある情報を使用して、Unica Journey インストーラーをナビゲートします。**Platform データベース接続**ウィンドウで、必要な情報をすべて入力し、**Nextを**クリックして、Unica Journey インストーラーを開始します。

Table 10. Unica Journeyインストーラーー GUI

ウィンドウ	説明
概要	これは Unica Journey のインストーラーの最初のウィンドウです。このウィンドウから、Unica Journey のインストール・ガイドとアップグレード・ガイドを開くことができます。 **[次]**をクリックします。
ソフトウェアのご使用条件	使用条件を注意深くお読みください。契約書を印刷する場合は、**Print**を使用します。同意の上、「**次**」をクリックします。
インストール・ディレクトリー	「**選択**」をクリックして、Unica Journey をインストールするディレクトリを参照します。 **[次]**をクリックします。
コンポーネント	インストールするコンポーネントを選択します。 コンポーネントを選択すると、そのコンポーネントに関する情報がインストーラーに表示されます。 **[次]**をクリックします。 **Note:** Unica Journeyこれら3つのコンポーネントは、同じマシンまたは異なるマシンにインストールすることができます。 • Unica JourneyWebアプリケーション • Unica Journeyエンジン • Apache Kafka
Unica Journeyデータベースのセットアップ	Unica Journeyデータベースのセットアップは自動的に行われます。デフォルトでは、Unicodeをサポートした SQLを実行します。 自動データベースセットアップを選択した場合、システムテーブルがUnicode用に設定されていれば、「Unicode SQLスクリプトを実行する」を選択します。 **[次]**をクリックします。
Unica Journeyデータベースタイプ	データベースの種類をOracle、SQLから選択します。MariaDBまたはOneDB。

ウィンドウ	説明
	[次]をクリックします。
Unica Journeyデータベース接続	Journey データベースについて、以下の詳細を入力します。 • データベース・ホスト名 • データベースポート • データベース・システム ID (SID) • データベース・ユーザー名 • パスワード **[次]**をクリックします。
JDBC 接続	JDBC 接続を検討して確認します。 **[次]**をクリックします。
Unica Journey接続設定	以下の接続設定を入力します。 • ネットワーク・ドメイン・ネーム **Note:** ネットワークドメイン名を追加すると、次のようなメッセージが表示されることがあります。 `警告サーバー名には、ドメイン名が含まれ、最終 URL にはドメイン名のいくつかのオカレンスが含まれます` ドメイン名を変更する場合は［**修正**］を、メッセージを取り消す場合は［**キャンセル**］を選択してください。 • ホスト名 • ポート番号 必要に応じて、「**安全な接続を使用**する」チェックボックスを選択します。 **[次]**をクリックします。
Unica Platform接続設定	以下の接続設定を入力します。 • ネットワーク・ドメイン・ネーム **Note:**

ウィンドウ	説明
	ネットワークドメイン名を追加すると、次のようなメッセージが表示されることがあります。 `警告サーバー名には、ドメイン名が含まれ、最終 URL にはドメイン名のいくつかのオカレンスが含まれます` ドメイン名を変更する場合は［**修正**］を、メッセージを取り消す場合は［**キャンセル**］を選択してください。 • ホスト名 • ポート番号 必要に応じて、「**安全な接続を使用**する」チェックボックスを選択します。 **[次]**をクリックします。
Kafkaスタンドアロンサーバーの詳細	このインスタンスと一緒にKafkaスタンドアロンインスタンスをインストールする場合、Kafkaの設定に以下の詳細が更新されます。 • ホスト名：Kafkaがインストールされているkafka Standaloneサーバーのホスト名を含めます。 • ポート番号：Kafka Zookeeperのポート番号を記載します。
プリインストールのサマリー	インストール・プロセスで追加した値を検討して確認します。 **「インストール」**をクリックして、インストール・プロセスを開始します。 Unica Journey インストーラーが開きます。
インストール完了	**完了**をクリックしてUnica Journey のインストーラーを終了し、Unica のインストーラーに戻ります。

7. **「インストールの完了」**ウィンドウで**「終了」**をクリックし、 Marketing Operations インストーラーを終了してUnica Journeyインストーラーーに戻ります。
8. Unica インストーラーの指示に従い、Unica Journey のインストールを完了させます。
   以下の表にある情報を使用して、Unica インストーラーの各ウィンドウで該当するアクションを実行します。

Table 11. HCL Unicaインストーラーー GUI

ウィンドウ	説明
デプロイメントEARファイル	Unica 製品を配置するために、エンタープライズ・アーカイブ (EAR) ファイルを作成するかどうかを指定します。 **[次]**をクリックします。
インストール完了	このウィンドウには、インストールで作成したログ・ファイルの場所が示されます。 インストールの詳細を変更する場合は、 [**前へ**] をクリックします。 「**完了**」をクリックして、Unica のインストーラーを閉じます。

Note: JourneyはEARのデプロイメントをサポートしていません。

コンソールモードによるUnica Journey のインストール

コンソールモードでは、コマンドラインウィンドウを使用して、Unica Journey をインストールすることができます。コマンドラインウィンドウで様々なオプションを選択することで、インストールする製品の選択やインストール先のホームディレクトリの選択などの作業を行うことができます。

Before you begin

Unica Journey をインストールする前に、必ず以下を構成しておいてください。

- アプリケーション・サーバー・プロファイル
- データベース・スキーマ

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、これらの文字エンコードを使用した一部の情報が読み取れなくなります。

1. コマンドラインプロンプトのウィンドウを開き、Unica インストーラーとUnica Journey インストーラーを保存したディレクトリに移動します。
2. 以下のアクションのいずれか 1 つを実行して、Unica インストーラーを実行します。
   Choose from:
   - Windows™場合、以下のコマンドを入力します。

     HCL_Unica_installer_12.1.1.0_win.exe -i コンソール

     たとえば、 HCL_Unica_Installer_12.1.1.0_win.exe -i コンソール

- UNIXの™場合は、*HCL_Unica_installer_12.1.1.0*.shファイルを起動します。

  例えばHCL_Unica_installer_12.1.1.0.sh

3. コマンド・ライン・プロンプトに表示される指示に従ってください。コマンドラインプロンプトでオプションを選択する必要がある場合は、以下のガイドラインを使用してください。
   - デフォルト・オプションはシンボル [X] で定義されます。
   - オプションを選択またはクリアするには、そのオプションに定義されている番号を入力し、`Enter`キーを押します。

   Example

   例えば、インストール可能なコンポーネントが以下のリストに表示されていると想定します。

   1 [X]Unica Platform

   2 [X]Unica Journey

   Note: すでにインストールされている場合を除き、Unica Platform のオプションはクリアしないでください。

4. Unica インストーラは、インストール作業中にUnica Journey インストーラを起動します。Unica Journey のインストーラのコマンドラインプロンプトウィンドウの指示に従ってください。
5. Unica Journey インストーラーのコマンドラインプロンプトウィンドウで`quit`を入力すると、ウィンドウは閉じます。Unica のインストーラーのコマンドラインプロンプトウィンドウの指示に従って、Unica Journey のインストールを完了します。

   Note: インストールの間にエラーが発生した場合、ログ・ファイルが生成されます。このログ・ファイルを表示するには、インストーラーを終了する必要があります。

Chapter 5. 構成Unica Journey

Unica Journey を配備する前に、Unica Journey システムユーザーを設定し、Unica Journey 構成プロパティを設定し、Unica Journey インストールを検証する必要があります。

インストール後、データベースでprocess_journey_goals_listプロシージャが正常にコンパイルされたかどうかを確認してください。プロシージャ - process_journey_goals_list が無効と表示された場合は、再コンパイルしてください。

Unica Journeyプロパティを構成する

Unica Journey には、構成ページで指定しなければならない重要な機能を実行する追加のプロパティがあります。その内容や設定方法については、「*Unica Journey*ユーザーガイド」をご覧ください。

Link とDeliver の構成プロパティを設定する

ユーザーは、Unica Platform の`Affinium|Journey|Link_Configurations` のパスにあるLink Configuration で、Unica Journey がUnica Link に接続するように設定することができます。

Journey >Link_Configurationsカテゴリにある以下のUnica Link 構成プロパティを手動で設定します。

- Link_URL を指定します。Unica Link デザインサーバーの URL を指定します。末尾に / が付いていないことを確認します。例: http://<FQDN>: <PORT>
- Link_Data_Source_ユーザ:Unica Link デザインサーバーの資格情報を保存するPlatform ユーザーを指定します。
- Link_Data_Source_名:認証情報を持つデータソース名を指定します。

また、パス`Affinium|Journey|Deliver_Configurations` の下にあるDeliver Configuration で、Unica Journey がUnica Deliver に接続するように設定することができます。

Journey >Deliver_Configurationsカテゴリにある以下のUnica Linkの構成プロパティを手動で設定します。

- Deliver_URL を指定します。Unica Deliver TMS サーバーの URL を指定します。例 `<UNICA_DELIVER_HOST>/delivertms/services/TMS` の場合。http://
- Deliver_Partition:Deliver TMS が設定されているCampaign のパーティション名を指定します。

Note: Deliver_Partitionに正しいパーティション名を入力したことを確認してください。

次の設定を使用して、Unica Link とUnica Deliver の統合を有効にすることができます。

Platform 構成設定 の下に移動することができます:

「Journey」 (Affinium|Journey) の設定

Link_Configured - この構成では、Unica Link がUnica Journey と統合されているかどうかを定義します (メール/SMS/CRM チャンネルの場合)。

可能な値 - はい/ いいえ

はい -Unica Link との統合を可能にします。Journey

Deliver_Configured - この構成では、Unica Deliver をUnica Journey と統合して電子メールを送信するかどうかを定義します。

可能な値 - はい/ いいえ

はい -Unica Deliver との統合を可能にします。Journey

Journey Webおよびエンジンコンポーネントを再起動する必要があります。

Platform_Configured- この構成では、Unica Platformが以下のものと統合されているかどうかを定義します。Unica Journey

Journeyを統合するためには、以下のPlatform APIを無効にする必要があります。

パス = コンフィギュレーション/Unica Platform/Security/API Management/Unica Platform

- 認証 = すべて無効にする
- データソース = すべてを無効にする
- ユーザー詳細 = すべて無効する
- 構成プロパティの取得 = すべて無効する
- コンフィグレーション = すべて無効する
- ログイン = すべて無効にする
- ユーザーの役割の許可 = すべてを無効にする
- ユーザー詳細 = すべて無効する
- ライセンス = すべて無効にする
- インストールされているアプリケーションを取得する = すべて無効にする

Note: Platformのトークン有効期限はデフォルトで15秒なので、手動で1800秒（30分）に延長する必要があります。Platformの ためにトークンの有効期限を延長するためのパスが表示されます: **Platform > 設定方法 > 設定方法 > 一般設定 > その他 > トークン有効期限**

JourneyをHTTPSでアクセスするためのAPI設定。

- 「認証」の設定 (Affinium|suite|security|apiSecurity|manager|managerAuthentication)
  - API URI - /authentication/login
  - APIアクセスをブロックする - 無効
  - HTTPSによるセキュアなAPIアクセス - 有効
  - APIアクセスに認証を要求する - 無効
- 「ユーザ」の設定 (Affinium|suite|security|apiSecurity|manager|managerUser)
  - API URI - /user/partitions/*
  - APIアクセスをブロックする - 無効
  - HTTPSによるセキュアなAPIアクセス - 無効
  - APIアクセスに認証を要求する - 有効
- 「ポリシー」の設定 (Affinium|suite|security|apiSecurity|manager|managerPolicy)
  - API URI - /policy/partitions/*
  - APIアクセスをブロックする - 無効

    - HTTPSによるセキュアなAPIアクセス - 無効
    - APIアクセスに認証を要求する - 有効
- 「構成」の設定 (Affinium|suite|security|apiSecurity|manager|Configuration)
    - API URI - /datasource/config
    - APIアクセスをブロックする - 無効
    - HTTPSによるセキュアなAPIアクセス - 有効
    - APIアクセスに認証を要求する - 有効
- 「データソース」の設定 (Affinium|suite|security|apiSecurity|manager|Datasource)
    - API URI - /datasource
    - APIアクセスをブロックする - 無効
    - HTTPSによるセキュアなAPIアクセス - 有効
    - APIアクセスに認証を要求する - 無効
- 「ログイン」の設定 (Affinium|suite|security|apiSecurity|manager|Login)
    - API URI - /authentication/v1/login
    - APIアクセスをブロックする - 無効
    - HTTPSによるセキュアなAPIアクセス - 有効
    - APIアクセスに認証を要求する - 無効
- 「User roles permissions」の設定 (Affinium|suite|security|apiSecurity|manager|managerGetRolesPermission)
    - API URI - /policy/roles-permissions
    - APIアクセスをブロックする - 無効
    - HTTPSによるセキュアなAPIアクセス - 無効
    - APIアクセスに認証を要求する - 有効
- 「ユーザ詳細」の設定 (Affinium|suite|security|apiSecurity|manager|managerGetUserDetails)
    - API URI - /user/user-details
    - APIアクセスをブロックする - 無効
    - HTTPSによるセキュアなAPIアクセス - 有効
    - APIアクセスに認証を要求する - 有効
- 「構成プロパティの取得」の設定 (Affinium|suite|security|apiSecurity|manager|managerGetConfigProperty)
    - API URI - /configuration/get
    - APIアクセスをブロックする - 無効
    - HTTPSによるセキュアなAPIアクセス - 有効
    - APIアクセスに認証を要求する - 無効
- 「ライセンス」の設定 (Affinium|suite|security|apiSecurity|manager|managerLicense)
    - API URI - /license/*
    - APIアクセスをブロックする - 無効
    - HTTPSによるセキュアなAPIアクセス - 無効
    - APIアクセスに認証を要求する - 無効

Note: これらの設定変更を適用した後、Platformアプリケーションを再起動します。

暗号化されたパスワードの生成方法

各プレーンテストパスワードに対して、暗号化されたパスワードを生成することが要求されます。暗号化ツールを複数回実行し、暗号化されたパスワードを生成します。

1. `<JOURNEY_WEB_HOME>/tools/` に移動してください。
2. JourneyEncryptionUtilityに`JAVA_HOME`を設定する .

   JAVA_HOME=<UNICA_HOME>/jre export JAVA_HOME
3. Linux OSをお使いの場合は、以下のコマンドでJourneyEncryptionUtility をUnixモードに変換してください。

   dos2unix JourneyEncryptionUtility
4. JourneyEncryptionUtilityを以下のコマンドで実行します。

   ジャーニー・EncryptionUtility<PASSWORD TEXT>
5. JourneyEncryptionUtilityは、コンソール出力に暗号化されたパスワードでプロンプトを表示します。

   何らかの理由でユーザーがJourneyシステムテーブルまたはJourney Reportsデータベースのユーザーパスワードを変更した場合、パスワード暗号化ユーティリティを使用して、それぞれのプロパティファイルでこれらのパスワードを更新することができます。

ClientIDとClientSecretを生成する手順

Unica PlatformのclientDetailsUtilityを実行し、以下のようにJourneyのクライアント詳細を生成します。

Linux システムでは、.bat の代わりに .sh ファイルを使用します。

1. PLATFORM_HOMEのtoolsbinディレクトリに移動します。Platformがインストールされているマシンが異なる場合は、Platformがインストールされているマシンでこのコマンドを実行します。
2. clientDetails -a Journeyとしてコマンドを実行します。これにより、ClientIDとClientSecretが生成される。以下はその例です。

   C:\Unica\Platform\tools\bin>clientDetails.bat -a Journey

   C:\Unica\Platform\tools\bin>echo off

   WARN com.unica.manager.configuration.ConfigurationManager - ローカルキャッシュがオフになっています。デフォルトの動作は、Hibernateのキャッシュに基づくことを意味します。

   パラメータ値

   ClientID: 885345

   ClientSecret：IfnKG2eqniVnaT8

   アプリ名ジャーニー

   ClientSecretとClientIdの生成に成功!
3. 生成されたClientIDとClientSecretをJourney Web アプリケーション.propertiesで使用します。

platform.clientId=上記の手順で生成されたClientID。

platform.clientSecret=上記の手順で暗号化されたClientSecret

Journey Web およびJourney Engine のアプリケーションプロパティを更新します。

Journey WebとJourney Engineのアプリケーションプロパティを更新します。以下の手順で、アップデートを実行します。

Journey Webアプリケーション.propertiesの更新を行うには、ユーザーが以下の手順を実行する必要があります。

1. 以下のプロパティは、PlatformとJourneyを並行して起動するために使用されます。アプリケーションサーバーによっては、Platformの起動に通常より多くの時間を要する場合があります。これらのプロパティはJourneyの起動時に使用され、指定された再試行回数と時間間隔でPlatformへの接続を試行します。
   - `platform.connect.retry.number`: Platformへの接続を何回再試行するか。
   - `platform.connect.retry.interval`: Platformへの接続の再試行間隔時間(ミリ秒)

     これらのプロパティの値は、<Journey_Home>/Web/ Properties/application.properties でユーザーが変更することができます。これらのプロパティの値は、どのアプリケーションサーバーを使用するかによって異なる。Tomcatの場合はデフォルト値、Websphere Application サーバー（WAS）の場合はリトライの時間間隔を長くする必要があります。
2. JOURNEYS_HOME/Web/properties/application.properties の "spring.entity.files.upload.defaultPath" パラメータのパスが、シングルフォワードスラッシュ（\\）からダブルフォワードスラッシュ（\）に変更になりました。
3. JOURNEYS_HOME/Web/properties/application.properties のパラメータ「spring.ignit.storage.path」のパスが、シングルフォワードスラッシュ(\\)からダブルフォワードスラッシュ(\)へ変更になりました。

Note: デフォルトでは、プロパティ`spring.entity.files.upload.defaultPath`と`spring.entity.files.upload.defaultFileReadBuffer`は1行で表示されます。以下のように、2つのプロパティに分ける必要があります。

`spring.entity.files.upload.defaultPath`を指定します。

`spring.entity.files.upload.defaultFileReadBuffer`を使用する。

Journey Engineのアップデート -`application.properties`ファイル

Engine application.propertiesファイル（Journeys_Install_location/Engine/）に暗号化されたパスワードを設定する必要がありますが、これは手作業です。

以下の手順で、アップデートを実施します。

1. 以下のプロパティの暗号化パスワードを生成し、Engine application.properties ファイルに記述する。`/JourneyEncryptionUtility.sh <JOURNEYS_HOME/tools>` を使用して、以下のプロパティを生成し、Engineの`application.properties`ファイルに記述します。
   - `journey.datasource.password`
   - `journey.report.datasource.password`

JourneyEncryptionUtility.sh (<JOURNEYS_HOME/tools>)<Journey System schema password> または <Journey Report schema password>としてコマンドを実行します。暗号化されたパスワードが生成されます。

以下はその例です。

[unica@cobra009 tools]$ ./JourneyEncryptionUtility.sh JourneySysctemschema

暗号化シェルスクリプトを開始しました...

入力された文字列は:JourneySysctemschema

暗号化された文字列は：3CKsX5SWYtGl+psHqlYUGkjXF9EVv6+XYP6GTIMa7WQ=です。

2. JOURNEYS_HOME/Engine/application.properties の "spring.entity.files.upload.defaultPath" パラメータのパスに、シングルフォワードスラッシュ (\) からダブルフォワードスラッシュ (\) に変更する必要があります。
3. JOURNEYS_HOME/Engine/application.properties の "spring.ignite.storage.path" パラメータのパスが、シングルフォワードスラッシュ(\\)からダブルフォワードスラッシュ(\)に変更になりました。
4. Journey Engineのapplication.propertiesに生成されたClientIDとClientSecretを使用します。

   platform.clientId=上記の手順で生成されたJourney Web アプリケーションのプロパティファイルのClientID

   platform.clientSecret=上記の手順で設定した暗号化されたクライアントシークレットJourney Webアプリケーションプロパティファイル

Note: デフォルトでは、プロパティ`spring.entity.files.upload.defaultPath`と`spring.entity.files.upload.defaultFileReadBuffer`は1行で表示されます。以下のように、2つのプロパティに分ける必要があります。

`spring.entity.files.upload.defaultPath`を指定します。

`spring.entity.files.upload.defaultFileReadBuffer`を使用する。

Note: アップグレード前に作成されたジャーニーのデータ処理に矛盾が生じないよう、アップグレード後の上記WebおよびEngineのイグナイトおよびtempフォルダのパスは、アップグレード前のものと同じにする必要があります。

KafkaStandalone server.properties および zookeeper.properties ファイルを更新します。

Windowsを使用している場合は、以下のサブステップを実行します。

1. `JOURNEY_HOME/KafkaStandalone/config/zookeeper.properties`の dataDir パラメータのパスが、1重のフォワードスラッシュ（\）から2重のフォワードスラッシュ（\）に変更になりました。
2. `JOURNEY_HOME/KafkaStandalone/config/server.properties`の log.dirs パラメータのパスに、シングルフォワードスラッシュ (\) からダブルフォワードスラッシュ (\) を含むように変更しました。

Tomcatのデプロイメントで使用するjourney.xmlを更新する。

Journey 配置 XML ファイルには、Journey システムテーブルのための暗号化されたパスワードが必要です。JourneyEncryptionUtilityを使用してパスワードを暗号化し、フィールド「password」にjourney.xmlファイルを指定することができます。

開始と検証のUnica Journeyインストール

Unica Journey をインストールおよび構成するためのすべてのステップを実行し終えたら、Unica Journey Web アプリケーションを配置して、それが終わった後に Unica Journey を構成します。これで、インストールを検査する準備が整います。

Before you begin

Journey アプリケーションを起動するための前提条件

Journey Web または Engine アプリケーションを起動するための前提条件:

- Unica Platformを起動する必要があります。
- Zookeeperのサーバーが稼働しています。
- Kafkaのサーバーが稼働しています。

起動とベリファイUnica Journey

Unica JourneyTomcatアプリケーションサーバに展開されたWebアプリケーションは、Tomcatインスタンスを起動することで起動する必要があります。

Unica Journey エンジン/サーバーの起動

- Unica JourneyEngineアプリケーションはスタンドアローンアプリケーションで、以下の手順で起動します。
  - `JOURNEY_HOME/Engine`ディレクトリに移動します。
  - java`-jar journeyEngine.jar`を実行して、Engineアプリケーションを実行します。オプションで、これをサービスとして起動するスクリプトを書くことができます。

KafkaサーバーとZookeeperの起動

KafkaサーバーとZookeperサーバーは、以下のコマンドで起動できます。

- `JOURNEY_HOME/KafkaStandalone/bin`(Linuxの場合)に移動します。
- `JOURNEY_HOME/KafkaStandalone/bin/windows`(Windowsの場合)に移動します。

以下のコマンドを実行して、まずZookeeperを起動します（ZookeeperはKafkaサーバーを停止している間に起動しておく必要があります）。

`zookeeper-server-start <PATH TO ZOOKEEPER CONF FILE>.`

例： `zookeeper-server-start JOURNEY_HOME/KafkaStandalone/config/zookeeper.` properties

`kafka-server-start <PATH TO SERVER CONF FILE>`を指定します。

例：`kafka-server-start JOURNEY_HOME/KafkaStandalone/config/server.properties`

インストールのJourney確認

まだの場合は、Unica Platform Administrators ロールに存在するユーザー（asm_admin など）でUnica にログインしてください。**設定 > ユーザーの役割と権限**を選択し、Unica Journey ユーザーのユーザー役割と権限を定義する必要があります。[ユーザーの役割とアクセス許可]で、次の役割とアクセス許可を割り当てる必要がありますUnica Journey応用。新しいユーザー ロールを割り当てたり、システムが提供するユーザー ロール（JourneyAdmin およびJourneyUser）を使用して作業することはできません。この2つのユーザーロールが実行できる役割を確認し、編集することができます。JourneyAdmin とJourneyUser にユーザーの役割と権限を定義したら、これらの役割をPlatform ユーザーに割り当てて、Journey アプリケーションでさまざまなアプリケーション機能にアクセスできるようにすることができます。

Unica 製品との統合のためのプロパティーの設定

Unica Journeyは、さまざまなアプリケーションと統合されています。

Unica Journey と他のUnica suite 製品との統合についての詳細は、以下の表のドキュメントマップを参照してください。

インストールと設定Unica Link

タスク	資料
のインストールと設定Unica Link	*Unica LinkV12.1*インストールガイド
のUnica Link コネクタアプリをインストールします。Journey	*Unica LinkV12.1*インストールガイド
Unica Link コネクタのインストール - MailChimp	*Unica LinkMailchimp Connector* ユーザーガイド
Unica Link コネクタの取り付け - Mandrill	*Unica LinkMandrill Connector* ユーザーガイド
Unica Link コネクタのインストール - Twilio	*Unica LinkTwilio*コネクタユーザーガイド
Unica Link コネクタのインストール - Salesforce	*Unica LinkSalesforce Connector* ユーザーガイド

Unica Campaign と他のHCL製品との統合

タスク	資料
Unica Campaign の統合とUnica Journey	*Unica Campaign* アドミニストレーションガイドおよび*Unica Campaign* ユーザーガイド
Unica Campaign の統合とUnica Interact	*Unica Interact*管理ガイド
Unica Deliver との統合Unica Journey	*Unica Journey*ユーザー・ガイド

Journey Proxy 統合

Proxyサーバーは、旅ウェブとエンジンのプロジェクトに統合され、これにより、ユーザーはセキュリティを追加し、アプリケーションサーバーをProxyサーバーの背後に保つことができるようになりました。Proxyサーバーは、Deliver、Link、Platformの各サーバーとやり取りを行います。

Journey Web - Deliver、Link、Platformサーバーと通信し、構成の詳細を取得したり、JourneyにメールSMS/AdTech Pointを統合したりします。

Journey Engine - Proxyを使用してDeliver/Link Serverとの通信を行い、Eメール/SMS/Adtechの詳細をエンドサーバーに送信します。

Journey Webでサポートされているproxy

1. SOCKS
2. HTTP
3. HTTPS

Journeyエンジンでサポートされているproxy

1. HTTP

Note: EngineがDeliverと通信するために使用するSOAP（Apache Axis2）では、SOCKSおよびHTTPS Proxyはサポートされていません。

エンジンの application.properties ファイルでエンジン用に設定するプロパティ

- journey.proxy.type=NONE
- spring.proxy.host=[IP]
- spring.proxy.port=[PORT]
- spring.proxy.username=[username]
- spring.proxy.password=[password]

Web application.properties ファイルで Web 用に設定されるプロパティ

- journey.proxy.type=NONE
- spring.proxy.host=[IP]
- spring.proxy.port=[PORT]
- spring.proxy.username=[username]
- spring.proxy.password=[password]
- server.use-forward-headers=true

Note: journey.proxy.typeプロパティのデフォルト値はNONEで、NONEに設定するとProxyは無効となる。

データベースの変更

メール配信停止イベントで実行されるMS SQLスクリプト。

Journeyシステムテーブルで以下のSQLスクリプトを実行してください。これは、Eメールの購読解除の詳細を入力するために必要です。

```
DROP TABLE IF EXISTS EmailUnsubscribedList;

CREATE TABLE EmailUnsubscribedList(

id BIGINT NOT NULL IDENTITY,

emailId NVARCHAR(200) NOT NULL,

status NVARCHAR(200) DEFAULT 0 NOT NULL,

channelAgent NVARCHAR(50),

eventID BIGINT NOT NULL,

audienceResponseId BIGINT,

audienceResponseExtendedId BIGINT,

createdBy NVARCHAR(200) DEFAULT 'SYSTEM' NOT NULL,

version BIGINT,

createdDate DATETIME2,

createdDateEpoch BIGINT NOT NULL,

modifiedDateTimeEpoch BIGINT,

FOREIGN KEY (eventID) REFERENCES AudienceResponseEventMaster(id),

FOREIGN KEY (audienceResponseId) REFERENCES AudienceResponse(id),

CONSTRAINT unique_emailId UNIQUE (emailId),

PRIMARY KEY (id)

);

DROP TABLE IF EXISTS AudienceResponseExtended;

CREATE TABLE AudienceResponseExtended(

id BIGINT NOT NULL IDENTITY,

AudienceResponseId BIGINT NOT NULL,

associatedAttributes NVARCHAR(MAX),

isProcessed BIT DEFAULT 0 NOT NULL,

createdDate DATETIME2,
```

```
createdBy NVARCHAR(200),
version BIGINT,
responseTimeEpoch BIGINT NOT NULL,
createdDateEpoch BIGINT,
FOREIGN KEY (audienceResponseId) REFERENCES AudienceResponse(id),
CONSTRAINT ensure_attribute_json CHECK (ISJSON(associatedAttributes) > 0),
PRIMARY KEY (id)
);
```

Chapter 6. Unica Journey アプリケーションのデプロイメント

Unica Journey Web アプリケーションを WAR ファイルを使用して配備することも、個々の WAR ファイルを配備することもできます。

Unica Journey を配置するには、このセクションのガイドラインに従ってから、Unica Journey サーバーを始動してください。

Unica JourneyWebアプリケーションは、別のtomcatインスタンスにデプロイされる必要があります。Unica Platform (unica.war) 配置のアプリケーションサーバプロファイル（tomcat インスタンス）には含めないでく ださい。

Tomcat、Webpshere、JBOSSアプリケーションサーバーでjourney.warを展開する際の推奨事項

unica.warがデプロイされていないアプリケーションサーバーにjourney.warをデプロイすることをお勧めします。ユーザーは、別のアプリケーションサーバーに journey.war をデプロイすることができます。

Journeyアプリケーションを起動するためには、Platformアプリケーションが起動している必要があります。ジャーニーアプリケーションとプラットフォームアプリケーションを同じJVMにデプロイした場合、アプリケーションサーバーの起動時に問題に直面します。

Apache Tomcat アプリケーションサーバーにUnica Journey を配置する。

About this task

以下のJourney のコンポーネントを配置または実行することができます。

- JourneyWeb - Tomcat にデプロイする必要があります。
- Journeyエンジン - スタンドアローンのアプリケーションとして実行されます。
- Kafkaサーバー - スタンドアローンアプリケーション（KafkaサーバーとZookeeper）として実行されます。

Tomcat に Unica Journey 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。

- Unica製品は、Tomcatが使用するJVMをカスタマイズします。Unica Journey Web アプリケーション展開専用の新しい tomcat インスタンスを作成する必要があります。
- 本番環境に導入する場合は、以下のset `CATALINA_OPTS=%CATALINA_OPTS% -Xms1024m -Xmx1024m -XX:MaxPermSize=512m`行を追加して、JVMのメモリーヒープサイズパラメーターを最低でも1024に設定してください。

  これらは推奨される最小値です。実際のサイズ要件を分析して、必要に合った適切な値を決定してください。システムの負荷に応じて、**-Xmxの**値を調整する必要があります。2048 より大きい値にする場合は、通常 64 ビット・アプリケーション・サーバーおよび JVM が必要です。
- `setenv.bat/sh`の `JAVA_OPTIONS`パラメータを変更して、以下の値を追加します。

  `JAVA_OPTS=%JAVA_OPTS% -DUNICA_PLATFORM_CACHE_ENABLED=true -Dclient.encoding.override=UTF-8`を設定してください.

`-Djourney.web.home=<Journeys_Install_location>/Web/` です。

- Unica Journey Tomcat インスタンスにjourney.war のパスと一緒に`journey.xml`という名前のUnica Journey 配置 XML ファイルを追加する必要があります。以下に例を示します。

```
<?xml version="1.0"?> <Context docBase="<Journeys_Install_Path>/Web/journey.war">
 <Environment name="journey.web.home" value="<Journeys_Install_Path>/Web/"
 type="java.lang.String"/> <Resource name="JourneyDS" type="javax.sql.DataSource"
 factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
 maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
 driverClassName="oracle.jdbc.OracleDriver" url="jdbc:oracle:thin:@<Host>:<Port>:<SID_NAME>"/>
 <Resource name="JourneyReportDS" type="javax.sql.DataSource"
 factory="com.hcl.journey.tomcat.util.JourneyTomcatDSFactory" maxActive="30" maxIdle="10"
 maxWait="10000" username="<your_db_user_name>" password="<your_db_user_password>"
 driverClassName="oracle.jdbc.OracleDriver" url="jdbc:oracle:thin:@<Host>:<Port>:<SID_NAME>"/>
 </Context>
```

Note:

- docBase = パスは、Journey Web War を指す必要があります
- {{You can encrypt DB password using }}、以下のJourneyEncryptionUtility.shは、`<Journey_Install_Path>/tools`に方法で行います。

データソース接続を作成するための重要なヒント

Oracle:

DRIVER_URL : jdbc:oracle:thin:@<DB_HOST_NAME:<DB_PORT>:<SID_NAME>.

DRIVER_CLASS_NAME : oracle.jdbc.OracleDriver

MariaDB:

DRIVER_URL : jdbc:mariadb://<DB_HOST_NAME>:<DB_PORT>/<DB_USER_NAME>

DRIVER_CLASS_NAME : org.mariadb.jdbc.Driver

SQL サーバー

DRIVER_URL : jdbc:sqlserver://<DB_HOST_NAME>:<DB_PORT>;databaseName=<DB_USER_NAME>

DRIVER_CLASS_NAME : com.microsoft.sqlserver.jdbc.SQLServerDriver

OneDB

DRIVER_URL : jdbc:informix-sqli://<DB_HOST_NAME>:<DB_PORT>/<DB_SCHEMA_NAME>:informixserver=<INFORMIX_SERVER_NAME>

DRIVER_CLASS_NAME : com.informix.jdbc.IfxDriver

- Tomcat アプリケーション・サーバーを再始動します。

AWS ELB (Elastic Load Balancing) が HTTPS 上にあり、Journey 製品が HTTP 上にある場合、Journey Swagger API ページがロードされていないか、Swagger API が実行されていない場合、ユーザーは以下の**パラメータ コネクタ タグをサーバーに追加して、SSL オフロード用に Tomcat を設定する必要があります。 xml** :

例: /opt/Tomcat/Journey_instance/conf/server.xml

<コネクタ ポート="7010" プロトコル="HTTP/1.1"

connectionTimeout="20000"

scheme="https" secure="true"

redirectPort="9010" />

Apache Tomcatアプリケーションサーバーのキャッシュのクリーニング

1. Unica Journeyに使用されるインスタンスロケーションにアクセスします。たとえば、 `/opt/Tomcat/instance1`です。
2. webappsとworkのフォルダの内容を削除してください。

WebSphere上にUnica Journeyを展開するためのガイドライン

WebSphere上にUnica Journeyを展開する際には、一連のガイドラインに従う必要があります。

WebSphere®のバージョンが、「推奨ソフトウェア環境および最小システム要件」文書に記載されている要件を満たしていることを確認します（フィックスパック（必要な場合）を含む）。以下は、WebSphere®上にUnica Journeyを展開するためのガイドラインです。

- journey.war Fileをエンタープライズアプリケーションとしてデプロイします。journey.warファイルをデプロイする際、JSPコンパイラのJDKソースレベルがJava 18 for SDK 1.8に設定されており、JSPページが以下の情報に従ってプリコンパイルされていることを確認してください。
  - WARファイルを参照・選択するフォームで、「**すべてのインストールオプションとパラメータを表示**する」を選択し、「**インストールオプションの選択**」ウィザードを実行するようにします。
  - **「インストール・オプションの選択」**ウィザードのステップ 1 で、**「JavaServer Pages ファイルのプリコンパイル」を**選択します。
  - **インストールオプションの選択**ウィザードのステップ 3 で、**JDK Source Level**が SDK 1.8 の 18 に設定されていることを確認します。
  - **インストールオプションの選択**ウィザードのステップ8で、**JourneyDSを**一致するターゲット・リソースとして選択します。
  - **インストールオプションの選択**ウィザードのステップ10で、コンテキストルートを`/journey`に設定する必要があります（すべて小文字）。
    - Finishをクリックし、アプリケーションがインストールされるのを待ちます。
    - WebSphere Enterprise Applications で、**Your Application を選択します（例：**journey.war）。
- **Webコンテナの「設定」→「Webコンテナ」→「セッション管理**」で、Cookieを有効にします。配置するアプリケーションごとに、異なるセッション Cookie 名を指定します。以下のいずれかの手順を使用して、Cookie 名を指定します。
  - **セッション 管理]の[セッション管理を上書き**する]チェックボックスを選択します。Unica製品用に別々のWARファイルをデプロイした場合は、WebSphereコンソールの［**アプリケーション**］ > ［**エンタープライズアプリケーション**］ > ［［**デプロイ済み_アプリケーション**］ > ［**セッション管**

理] > [**クッキーを有効にする**] > [**クッキー名**] セクションで、固有のセッションクッキーの名前を指定します。

- ポルトガル語など、非ASCII文字をサポートする必要がある場合や、マルチバイト文字を必要とするロケールでは、サーバー・レベルの**Generic JVM Argumentsに**以下の引数を追加してください。

  -Dfile.encoding=UTF-8

  -Dclient.encoding.override=UTF-8

  以下のように、**Journey Web Homeの**パス変数を、properties & configフォルダを置いた場所に追加します。

  -Djourney.web.home=<Journeys_Home>/Web/ です。

  実稼働環境では、このJavaオプションは削除するか、`false`に設定する必要があります。

  ナビゲーションのヒント：選択 **サーバー > アプリケーションサーバー > Javaとプロセス管理 > プロセス定義 > Java仮想マシン > 汎用JVM引数**.

その他の詳細

- の中にある **アプリケーション > エンタープライズアプリケーション**のセクションで、デプロイしたWARファイルを選択し、[**Class loading and update detection**]を選択して、以下のプロパティを指定します。

- の中の **アプリケーション > エンタープライズアプリケーション**のセクションで、デプロイしたEARファイルまたはWARファイルを選択し、[**Class loading and update detection**]を選択して、以下のプロパティを指定します。
  - WAR ファイルを配置する場合:
    - **クラス・ローダーの順序**では、 [**ローカル・クラス・ローダーで読み込まれたクラスを最初に（親を最後に）**] を選択します。
    - **WAR クラスローダー ポリシーで**、[**Single class loader for application**] を選択します。
- WebSphere Enterprise Applications で ***Your Application*> Manage Modules >*Your Application*> Class Loader Order > Class loaded with local class loader first (parent last)を選択します。**.
- アプリケーションの基本的な機能を実現するための推奨最小ヒープサイズは512で、推奨最大ヒープサイズは1024です。

  ヒープサイズを指定するには、次のタスクを実行します。

  1. WebSphere Enterprise Applicationsで®、 **Servers > WebSphere application servers > server1 > Server Infrastructure > Java and Process Management > Process definition > Java Virtual Machineを選択します。**.
  2. ヒープサイズの初期値を512に設定する。
  3. 最大ヒープ・サイズを 1024 に設定します。

  サイズ指定に関する詳細については、WebSphere の®ドキュメントを参照してください。

特定のウェブコンテナのカスタムプロパティを追加します。

1. サーバー > [サーバーの種類] > [アプリケーションサーバー]をクリックし、最初に作成したサーバーを選択します。
2. Webコンテナの設定 > [Webコンテナ]をクリックします。

3. ［カスタムプロパティ］をクリックします。
4. 「新規」をクリックする。
5. プロパティ値を入力します。

プロパティ	値
名前	com.ibm.ws.webcontainer.invokeFlushAfterService
値	False
説明	http://www.ibm.com/support/docview.wss?uid=swg1PM50111 をご覧ください。

6. 「OK」をクリックします。
7. 「保存」をクリックします。

Note: WebSphereに展開する場合は、HTTPS証明書をインポートする必要があります。JourneyはLink and Deliverと統合されているため、これらのアプリケーションがHTTPSでデプロイされている場合、WebSphereアプリケーションサーバーにHTTPS証明書をインポートする必要があり、そうしないとJourneyはLink and Deliverにアクセスできなくなります。

SSL証明書のインポート方法が必要な場合は、以下のURLを参照してください。https://www.ibm.com/support/knowledgecenter/en/SSEKCU_1.1.2.1/com.ibm.psc.doc/rs_original/installer/rs_t_import_client_cert_was.html

Note: WebSphere で OneDB データベースを使用していて、OneDB の DB_LOCALE が en_us.57372 に設定されている場合、WebSphere コンソールでデータソースのカスタムプロパティのロケールを ifxDB_LOCALE ="en_us.57372" と ifxCLIENT_LOCALE="en_us.57372" にも設定してください。

Websphere アプリケーションサーバーのキャッシュのクリーニング

1. Journeyのインストールに使用したWASプロファイルの場所に移動します。例: `/data/webservers/IBM/WebSphere85_ND/profiles/UMP9111`
2. そこに、tmpとwstempの2つのフォルダがあります。
3. この2つのフォルダの内容を削除してください。

- WebSphereを再起動する

Note: プラットフォームが正常に起動したら、WebsphereサーバーでJourneyアプリケーションを手動で起動する必要があります。

- journey.war (Journey Application)のデプロイを開始します。

JBossにUnica Journeyをデプロイするためのガイドライン

JBoss に 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。

JBoss のバージョンが、HCL 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 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 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. unica.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デプロイします。

   例： deploy <Journey_Install>\unica.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ディレクトリの下にある`standalone.conf`スクリプトを編集し、`JAVA_VENDOR`に

      `-Dfile.encoding=UTF-8`

      `-Dclient.encoding.override=UTF-8`

      `-Djboss.as.management.blocking.timeout=3600`

      を追加します。

      非本番環境でのデプロイを行う場合は、以下のコマンドを追加します。

      `-DENABLE_NON_PROD_MODE=true`

      実稼働環境では、このJavaオプションは削除するか、`false`に設定する必要があります。
   b. JBOSSサーバーを再起動します。
3. Schedulerが正しく動作するように、以下のサブステップを完了してください。
   - <JBOSS_HOME>/standalone/configuration/standalone.xml ファイルのバックアップを取ります。
   - <JBOSS_HOME>/standalone/configuration/standalone.xml で、モジュール名から driver を検索してください。

     ```
     <driver name="oracledriver" module="oracle.jdbc">
      <xa-datasource-class>oracle.jdbc.OracleDriver</xa-datasource-class> </driver>
     ```
   - インストーラーはデータソースを更新しないので、手動でデータソースを設定する必要があります。
   - `<subsystem xmlns="urn:jboss:domain:ee:4.0">` の下に以下の記述を追加して、モジュール名をグローバルにします。

     ```
     <global-modules> <module name="oracle.jdbc"/> </global-modules>
     ```
   - JBOSSサーバーを再起動します。

JBOSS アプリケーションサーバーのキャッシュのクリーニング

1. Journeyのインストールに使用したJBOSSのインストール先ロケーションに移動します。例えば、`/jboss-eap-7.1/standalone`の場合。
2. そこに、tmpとdeploymentsの2つのフォルダがあります。
3. この2つのフォルダの内容を削除する

Note: Journey Web コンポーネントは、application.properties ファイルから設定を読み取ります。Journey Webアプリケーションのプロパティファイルと Journey Engineアプリケーションのプロパティファイルは、journey.warをデプロイする前に更新しておく必要があります。ご参照ください。構成Unica Journey on page 31

Chapter 7. のアンインストールUnica Journey

Unica Journey アンインストーラーを実行し、Unica Journey をアンインストールしてください。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。アンインストール前にUnica Journey に関連するプロセスを停止してください。

About this task

製品をインストールするとUnica、`Uninstall_`*`Product`*ディレクトリ (*Product*は製品名) にアンインストーラが含まれます。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。

Note: UNIX™の場合、Unica Journey をインストールしたのと同じユーザーアカウントでアンインストーラーを実行する必要があります。