



Migration Toolkit for Applications 4.2

Maven プラグインガイド

Red Hat Application Migration Toolkit の Maven ビルドプロセスへの統合

Migration Toolkit for Applications 4.2 Maven プラグインガイド

Red Hat Application Migration Toolkit の Maven ビルドプロセスへの統合

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律上の通知

Copyright © 2023 | You need to change the HOLDER entity in the en-US/Maven_Plugin_Guide.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

本ガイドでは、Migration Toolkit for Applications Maven プラグインを使用して Java アプリケーションの移行を単純化する方法を説明します。

目次

第1章 はじめに	3
1.1. MAVEN プラグインガイドについて	3
1.2. MIGRATION TOOLKIT FOR APPLICATIONS について	3
Migration Toolkit for Applications とは	3
Migration Toolkit for Applications で移行を単純化する方法	3
詳細情報	3
1.3. MAVEN プラグインについて	3
第2章 スタートガイド	4
2.1. 前提条件	4
2.2. MAVEN プラグインの実行	4
2.3. 複数のモジュールを使用した MAVEN プラグインの実行	5
2.4. レポートへのアクセス	6
第3章 レポートを CSV 形式でエクスポートする	7
3.1. レポートをエクスポートする	7
3.2. CSV ファイルのスプレッドシートプログラムへのインポート	7
3.3. CSV データ構造の概要	7
付録A 参考資料	9
A.1. MAVEN プラグイン引数	9
A.1.1. 入力ディレクトリーの指定	11
A.1.2. 出力ディレクトリーの指定	12
A.1.3. ソーステクノロジーの設定	12
A.1.4. ターゲット引数の設定	12
A.1.5. パッケージの選択	13
A.2. MAVEN プラグインのデフォルトの LOGGING.PROPERTIES	13
A.3. ルールのストーリーポイント	14
A.3.1. ストーリーポイントとは	14
A.3.2. ルールにおけるストーリーポイントの見積方法	14
A.3.3. タスクカテゴリー	15
A.4. 関連情報	15
A.4.1. 参加する	15
A.4.2. 重要なリンク	16
A.4.3. MTA の既知の問題	16
A.4.4. MTA の問題を報告する	16
A.4.4.1. JIRA の問題を作成する	17

第1章 はじめに

1.1. MAVEN プラグインガイドについて

Java アプリケーションやその他のコンポーネントを移行するために、Migration Toolkit for Applications (MTA) を使用するエンジニア、コンサルタント、およびその他のユーザーを対象としています。Maven プラグインのインストールおよび実行方法、生成されたレポートを確認し、追加機能を活用する方法を説明します。

1.2. MIGRATION TOOLKIT FOR APPLICATIONS について

Migration Toolkit for Applications とは

Migration Toolkit for Applications (MTA) は、拡張およびカスタマイズ可能なルールベースのツールセットで、Java アプリケーションの移行を容易にします。

MTA は、プロジェクトソースディレクトリーやアプリケーションアーカイブを含むアプリケーションアーティファクトを検査し、変更を必要とするエリアを強調表示する HTML レポートを作成します。MTA を使用して、以前のバージョンの Red Hat JBoss Enterprise Application Platform から、または Oracle® WebLogic Server や IBM® WebSphere® Application Server などの他のコンテナから Java アプリケーションを移行できます。

Migration Toolkit for Applications で移行を単純化する方法

Migration Toolkit for Applications は一般的なリソースを探し、アプリケーションを移行する際のテクノロジーと既知の問題点を明らかにします。この目的は、アプリケーションが使用するテクノロジーの概要を提供し、組織がエンタープライズアプリケーションを Java EE および Red Hat JBoss Enterprise Application Platform に推定、文書化、移行するために使用できる詳細なレポートを提供することです。

詳細情報

Migration Toolkit for Applications の機能、サポートされている設定、システム要件、および使用可能なツールの詳細については、[スタートガイド](#) を参照してください。

1.3. MAVEN プラグインについて

Migration Toolkit for Applications の Maven プラグインは Maven ビルドプロセスに統合されるため、開発者はソースコードの反復ごとに移行およびモダナイゼーション作業を継続的に評価できます。分析結果を強調表示するレポートが多数提供され、ビルドごとに更新が必要な開発者向けに設計されています。

第2章 スタートガイド

2.1. 前提条件

Maven プラグインを使用する前に、次の前提条件を満たしていることを確認してください。

- Java プラットフォーム、JRE バージョン 8+
- 最低 4 GB の RAM (8GB を推奨)
- Maven バージョン 3.2.5 以降



注記

macOS を実行している場合は、ユーザープロセスの最大数 **maxproc** を少なくとも **2048** に設定し、開いているファイルの最大数 **maxfiles** を **100000** に設定することを推奨します。

2.2. MAVEN プラグインの実行

Maven プラグインは、アプリケーションの **pom.xml** ファイルにプラグインへの参照を追加して実行されます。アプリケーションがビルドされると、Maven プラグインが実行され、分析用のレポートが生成されます。

Maven プラグインを実行するには、次の手順を実行します。

1. アプリケーションの **pom.xml** 内に次のプラグインを含めます。

```
[...]
<plugin>
  <groupId>org.jboss.windup.plugin</groupId>
  <artifactId>windup-maven-plugin</artifactId>
  <version>4.2.1.Final</version>
  <executions>
    <execution>
      <id>run-windup</id>
      <phase>package</phase>
      <goals>
        <goal>windup</goal>
      </goals>
    </execution>
  </executions>
  <configuration>
    <offlineMode>true</offlineMode>
  </configuration>
</plugin>
[...]
```

- **offlineMode**: オフラインモードで実行し、ネットワーク機能を無効にしてパフォーマンスを向上させることを示します。
上記の例は、最低限必要な引数を示しています。使用可能なすべての引数の詳細な説明については、[MTA Maven 引数](#) を参照してください。

2. Java 11 を使用している場合は、**--add-modules=java.se** を **MAVEN_OPTS** 環境変数に追加する必要があります。古いバージョンの Java を使用する場合は、これは必須ではないため、次のステップに進むことができます。

```
export MAVEN_OPTS="--add-modules=java.se"
```

重要

Java 11 での Maven プラグインの使用はテクノロジープレビューとしてのみ提供されます。テクノロジープレビュー機能は、Red Hat の実稼働環境でのサービスレベルアグリーメント (SLA) ではサポートされていないため、Red Hat では実稼働環境での使用を推奨していません。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。

テクノロジープレビュー機能のサポート範囲は、Red Hat カスタマーポータルの [テクノロジープレビュー機能のサポート範囲](#) を参照してください。

3. プロジェクトをビルドします。

```
$ mvn clean install
```

4. [生成されたレポートにアクセス](#) します。

2.3. 複数のモジュールを使用した MAVEN プラグインの実行

複数のモジュールを持つプロジェクトで Maven プラグインを使用するには、親の **pom.xml** 内に設定を配置します。実行中、Maven プラグインは親モジュールおよび子モジュールの分析が含まれる単一のレポートを生成します。

注記

マルチモジュールプロジェクトでは、**inherited** を **false** に設定することが強く推奨されます。それ以外の場合は、子の各モジュールがコンパイルされるときに Maven プラグインが実行し、子モジュールに対して Maven プラグインが複数回実行されます。**inherited** を **false** に設定すると、各プロジェクトで1回分析され、実行時間が大幅に短縮されます。

複数のモジュールを持つプロジェクトで Maven プラグインを実行するには、以下の手順を実行します。

1. 親プロジェクトの **pom.xml** に以下のプラグインを追加します。以下は、親モジュールの **pom.xml** の例になります。

```
<plugin>
  <groupId>org.jboss.windup.plugin</groupId>
  <artifactId>windup-maven-plugin</artifactId>
  <version>4.2.1.Final</version>
  <inherited>>false</inherited>
  <executions>
    <execution>
      <id>run-windup</id>
      <phase>package</phase>
```

```

        <goals>
          <goal>windup</goal>
        </goals>
      </execution>
    </executions>
    <configuration>
      <input>${project.basedir}</input>
      <offlineMode>true</offlineMode>
      <windupHome>PATH/TO/CLI</windupHome>
    </configuration>
  </plugin>

```

この **pom.xml** は、[Maven プラグインの実行](#) のものと次の属性で異なります。

- **inherited**: プラグインレベルで定義され、この設定を子モジュールで使用するかどうかを指定します。パフォーマンスを向上させるために **false** に設定します。
 - **input**: 分析されるプロジェクトが含まれるディレクトリーへのパスを指定します。この属性はデフォルトで **{project.basedir}/src/main** に設定されています。親プロジェクトに解析するソースコードがない場合は定義する必要があります。
 - **samupHome**: MTA CLI の抽出されたコピーへのパス。この属性は任意ですが、パフォーマンスを向上させることが推奨されます。
上記の例は、推奨される引数のセットを示しています。使用可能なすべての引数の詳細な説明については、[MTA Maven 引数](#) を参照してください。
2. 親プロジェクトをビルドします。ビルドプロセス中に、Maven プラグインは追加設定なしでプロジェクトのすべての子に対して実行されます。

```
$ mvn clean install
```

3. 完了したら、通常どおり、[生成されたレポートにアクセス](#) します。このレポートには、親およびすべての子に関する分析が含まれます。

2.4. レポートへのアクセス

Migration Toolkit for Applications を実行すると、**pom.xml** に **outputDirectory** 引数を使用して指定する **OUTPUT_REPORT_DIRECTORY** にレポートが生成されます。ビルドが完了すると、ビルドログに以下のメッセージが表示されます。

```
Windup report created: OUTPUT_REPORT_DIRECTORY/index.html
```

output ディレクトリーには、以下のファイルおよびサブディレクトリーが含まれます。

```

OUTPUT_REPORT_DIRECTORY/
├── index.html           // Landing page for the report
├── EXPORT_FILE.csv      // Optional export of data in CSV format
├── graph/              // Generated graphs used for indexing
├── reports/            // Generated HTML reports
└── stats/              // Performance statistics

```

MTA レポートの詳細と、それらを使用して移行またはモダナイゼーションの取り組みを評価する方法は、MTA の [CLI ガイド](#) の [レポートの確認](#) セクションを参照してください。

第3章 レポートを CSV 形式でエクスポートする

MTA には、分類やヒントなどのレポートデータをローカルファイルシステムのフラットファイルにエクスポートする機能があります。export 関数は現在 CSV ファイル形式をサポートし、レポートデータはコンマ (,) で区切られたフィールドとして示されます。

CSV ファイルは、Microsoft Excel、OpenOffice Calc、LibreOffice Calc などのスプレッドシートソフトウェアでインポートおよび操作できます。スプレッドシートソフトウェアは、MTA レポートから結果データを並べ替え、分析、評価、および管理する機能を提供します。

3.1. レポートをエクスポートする

レポートを CSV ファイルにエクスポートするには、**exportCSV** 引数が **true** に設定された MTA を実行します。CSV ファイルは、解析される各アプリケーションの、**--output** 引数で指定されたディレクトリに作成されます。分析されたすべてのアプリケーションにわたるすべての発見された問題は、**AllIssues.csv** ファイルに含まれます。

```
<exportCSV>true</exportCSV>
```

CSV ファイルは、**outputDirectory** 引数で指定されたディレクトリに作成されます。

3.2. CSV ファイルのスプレッドシートプログラムへのインポート

1. スプレッドシートソフトウェア (例: Microsoft Excel) を起動します。
2. **File** → **Open** を選択します。
3. CSV でエクスポートされるファイルを参照し、これを選択します。
4. これで、スプレッドシートソフトウェアでデータを分析できるようになりました。

詳細について、または問題を解決するには、スプレッドシートソフトウェアのヘルプを確認してください。

3.3. CSV データ構造の概要

CSV 形式の出力ファイルには、以下のデータフィールドが含まれます。

ルール ID

指定の項目を生成したルールの ID。

問題のタイプ

ヒント または 分類

件名

classification または **hint** の件名。このフィールドは、特定の項目の問題を要約します。

説明

指定項目の問題の詳細な説明。

リンク

問題に関する追加情報を提供する URL。リンクは、リンクとリンクの説明という 2 つの属性で設定されます。

アプリケーション

この項目が生成されたアプリケーションの名前。

ファイル名

指定項目のファイルの名前。

ファイルパス

指定項目のファイルパス。

行

指定項目のファイルの行番号。

ストーリーポイント

特定の項目に割り当てられた、努力のレベルを表すストーリーポイントの数。

付録A 参考資料

A.1. MAVEN プラグイン引数

以下は、利用可能な MTA Maven プラグイン引数の詳細な説明です。

表A.1 MTA Maven プラグイン引数

引数	説明
customLoggingPropertiesFile	java.util.logging.LogManager ログ設定が含まれる logging.properties ファイルへの絶対パス。指定されたパスが無効であるか、オプションが指定されていない場合、ログは Maven プラグインに含まれる logging.properties ファイルの使用に戻ります。
disableTattletale	Tattletale レポートの生成を無効にするフラグ。 enableTattletale と disableTattletale の両方が true に設定されていると、 disableTattletale は無視されます。また、Tattletale レポートは引き続き生成されます。
enableCompatibleFilesReport	Compatible Files レポートの生成を有効にするフラグ。問題が検出されない状態ですべてのファイル进行处理するため、このレポートには大規模なアプリケーションの処理に時間がかかる場合があります。
enableTattletale	各アプリケーションの Tattletale レポートの生成を有効にするフラグ。このオプションは、 eap が含まれるターゲットにある場合にデフォルトで有効になります。 enableTattletale と disableTattletale の両方が true に設定されていると、 disableTattletale は無視されます。また、Tattletale レポートは引き続き生成されます。
excludePackages	評価から除外するパッケージの一覧たとえば、com.mycompany.commonutilities と入力すると、パッケージ名が com.mycompany.commonutilities で始まるクラスをすべて除外します。
excludeTags	除外するタグの一覧。指定されている場合は、これらのタグを持つルールは処理されません。
explodedApps	指定された入力ディクショナリーに1つのアプリケーションのソースファイルが含まれていることを示すフラグ。詳細については、 入力ファイルの引数テーブル を参照してください。
exportCSV	レポートデータをローカルファイルシステムの CSV ファイルにエクスポートするフラグ。MTA は、 outputDirectory 引数で指定されたディレクトリーにファイルを作成します。CSV ファイルは、データの操作および分析のためにスプレッドシートプログラムにインポートできます。詳細については、 レポートを CSV 形式でエクスポートする を参照してください。

引数	説明
includeTags	使用するタグの一覧。指定されると、これらのタグを持つルールのみが処理されます。
inputDirectory	分析するアプリケーションが含まれるディレクトリーへのパスを指定します。この引数はデフォルトで {project.basedir}/src/main/ です。詳細については、 入力ディレクトリーの指定 を参照してください。
keepWorkDirs	グラフデータベースや展開されたアーカイブなどの一時作業ファイルを削除しないように MTA に指示するフラグ。これはデバッグに役立ちます。
packages	MTA が評価するパッケージの一覧。この引数は必須です。詳細については、 パッケージの選択 を参照してください。
offlineMode	フラグはオフラインモードで動作し、スキームの検証などのネットワークアクセス機能を無効にします。パフォーマンスの向上に使用します。
outputDirectory	MTA が生成したレポート情報を出力するディレクトリーへのパスを指定します。この引数はデフォルトで {project.build.directory}/windup-report に設定されます。詳細については、 出力ディレクトリーの指定 を参照してください。
overwrite	<p>outputDirectory で指定された既存の出力ディレクトリーを強制的に削除するためのフラグ。デフォルトは true です。</p> <div>  <p>警告</p> <p>重要な情報が含まれるレポート出力ディレクトリーを指定しないでください。</p> </div>
sourceTechnologies	移行元となる1つ以上のソーステクノロジー、サーバー、プラットフォーム、またはフレームワークの一覧。この引数は、 targetTechnologies 引数とともに、使用されるルールセットを判断するのに役立ちます。詳細については、 ソーステクノロジーの設定 を参照してください。
sourceMode	評価するアプリケーションに、コンパイルされたバイナリーではなくソースファイルが含まれていることを示すフラグ。デフォルトは true です。詳細については、 入力ファイルの引数テーブル を参照してください。

引数	説明
targetTechnologies	移行先の1つ以上のターゲットテクノロジー、サーバー、プラットフォーム、またはフレームワークの一覧。この引数は、 sourceTechnologies 引数とともに、使用されるルールセットを判断するのに役立ちます。詳細については、 ターゲットテクノロジーの設定 を参照してください。
userIgnorePath	MTA が無視されるファイルを識別する場所を指定します。
userRulesDirectory	カスタム MTA ルールを検索する場所を指定します。値は、単数または複数のルールセットファイルを含むディレクトリーです。ルールセットファイルには .windup.xml または .rhamt.xml 接尾辞を使用する必要があります。
windupHome	抽出した MTA CLI のルートを参照する任意の引数。CLI のローカルインストールを参照すると、Maven プラグインはすべてのインデックスに直接アクセスできるため、パフォーマンスが向上します。
windupVersion	実行する MTA のバージョンを指定します。デフォルトでは、これは Maven プラグインのビルドバージョンです。

A.1.1. 入力ディレクトリーの指定

分析する1つ以上のアプリケーションを含むファイルまたはディレクトリーへのパス。デフォルトは **{project.basedir}/src/main/** です。

使用方法

```
<inputDirectory>INPUT_ARCHIVE_OR_DIRECTORY</inputDirectory>
```

inputDirectory 引数に指定された入力ファイルタイプがファイルであるかディクショナリーであるかに応じて、指定された追加の引数に応じて次のように評価されます。

ディレクトリー

--explodedApp	--sourceMode	引数なし
ディレクトリーは1つのアプリケーションとして評価されます。	ディレクトリーは1つのアプリケーションとして評価されます。	各サブディレクトリーはアプリケーションとして評価されます。

ファイル

<code>--explodedApp</code>	<code>--sourceMode</code>	引数なし
引数は無視されます。ファイルは1つのアプリケーションとして評価されます。	ファイルは圧縮プロジェクトとして評価されます。	ファイルは1つアプリケーションとして評価されます。

A.1.2. 出力ディレクトリーの指定

MTA が生成したレポート情報を出力するディレクトリーへのパスを指定します。

使用方法

```
<outputDirectory>OUTPUT_REPORT_DIRECTORY</outputDirectory>
```

- 省略すると、レポートは `{project.build.directory}/windup-report` ディレクトリーに生成されます。
- 出力ディレクトリーが存在する場合は、**overwrite** 引数の値に基づいて上書きされます。この引数はデフォルトで **true** で、MTA はディレクトリーを削除し、再作成します。

A.1.3. ソーステクノロジーの設定

移行元となる1つ以上のソーステクノロジー、サーバー、プラットフォーム、またはフレームワークの一覧。この引数は、**targetTechnologies** 引数とともに、使用されるルールセットを判断するのに役立ちます。

使用方法

```
<sourceTechnologies>
  <source>eap:6</source>
</sourceTechnologies>
```

sourceTechnologies 引数は、[Maven バージョン範囲の構文](#) に続くバージョンサポートを提供するようになりました。これにより、指定されたバージョンに一致するルールセットのみを実行するように MTA が指示されます。たとえば、`<source>eap:5</source>` です。

A.1.4. ターゲット引数の設定

移行先の1つ以上のターゲットテクノロジー、サーバー、プラットフォーム、またはフレームワークの一覧。この引数は、**sourceTechnologies** 引数とともに、使用されるルールセットを判断するのに役立ちます。この引数は必須です。

使用方法

```
<targetTechnologies>
  <target>eap:7</target>
</targetTechnologies>
```


targetTechnologies 引数は、[Maven バージョン範囲の構文](#) に続くバージョンサポートを提供するようになりました。これにより、指定されたバージョンに一致するルールセットのみを実行するように MTA が指示されます。たとえば、`<target>eap:7</target>` です。



警告

JBoss EAP に移行する場合は、必ずバージョンをターゲットに指定してください (例: **eap:6**)。eap のみを指定すると、移行パスに関連しないものを含め、すべてのバージョンの JBoss EAP にルールセットが実行されます。

ソースプラットフォームに適した JBoss EAP バージョンについては、MTA スタートガイドの [サポート対象の移行パス](#) を参照してください。

A.1.5. パッケージの選択

MTA が評価するパッケージの一覧。この引数を使用することは強く推奨されます。

使用方法

```
<packages>
  <package>PACKAGE_1</package>
  <package>PACKAGE_2</package>
</packages>
```

- 多くの場合、関心があるのは、カスタムアプリケーションクラスパッケージの評価で、標準の Java EE パッケージやサードパーティーのパッケージではありません。**PACKAGE_N** 引数はパッケージ接頭辞で、すべてのサブパッケージがスキャンされます。たとえば、**com.mycustomapp** および **com.myotherapp** パッケージをスキャンするには、**pom.xml** で以下のスニペットを使用します。

```
<packages>
  <package>com.mycustomapp</package>
  <package>com.myotherapp</package>
</packages>
```

- org.apache** などの標準の Java EE サードパーティーソフトウェアにパッケージ名を指定することはできますが、通常は移行作業に影響しないため、追加しないことが推奨されます。

A.2. MAVEN プラグインのデフォルトの LOGGING.PROPERTIES

以下では、Maven プラグインに含まれるデフォルトの **logging.properties** ファイルが提供されます。この設定は、Maven プラグインの進捗を表示できる一方で、多くの余分なメッセージを省略します。

```
#
# Copyright 2013-2014 Red Hat, Inc. and/or its affiliates.
#
# Licensed under the Eclipse Public License version 1.0, available at
# http://www.eclipse.org/legal/epl-v10.html
#
```

```
# Additional loggers to configure (the root logger is always configured)
#loggers=
handlers=java.util.logging.ConsoleHandler
.level=INFO
#java.util.logging.ConsoleHandler.level=INFO

#loggers=org.jboss.forge,org.jboss.weld,org.xnio,org.jboss.forge,org.ocpsoft.rewrite,org.jboss.windup.graph.GraphModelScanner,org.jboss.windup.reporting.xml.ClassificationHandler,org.jboss.windup.graph.GraphType$
org.jboss.forge.level=SEVERE
org.janusgraph.level=SEVERE
org.janusgraph.diskstorage.berkeleyje.BerkeleyJEKeyValueStore.level=SEVERE
org.janusgraph.diskstorage.berkeleyje.level=SEVERE
org.jboss.weld.level=SEVERE
org.xnio.level=SEVERE
org.jboss.forge.level=SEVERE
org.ocpsoft.rewrite.level=SEVERE
org.jboss.windup.graph.GraphModelScanner.level=SEVERE
org.jboss.windup.reporting.xml.ClassificationHandler.level=SEVERE
org.jboss.windup.graph.GraphTypeManager.level=SEVERE
org.jboss.windup.graph.GraphContextImpl.level=SEVERE
org.jboss.windup.rules.files.FileMapping.level=SEVERE
org.jboss.windup.exec.level=SEVERE
org.jboss.windup.config.level=SEVERE
com.thinkaurelius.level=SEVERE
org.jboss.windup=INFO
```

A.3. ルールのストーリーポイント

A.3.1. ストーリーポイントとは

ストーリーポイントは、アジャイルソフトウェア開発で一般的に使用される抽象メトリクスで、機能や変更を実装するのに必要な作業量を予測します。

Migration Toolkit for Applications はストーリーポイントを使用して、特定のアプリケーションコンストラクトとアプリケーション全体を移行するために必要な作業のレベルを表現します。必ずしも工数に変換される訳ではありませんが、この値はタスク全体で一貫性を持たせる必要があります。

A.3.2. ルールにおけるストーリーポイントの見積方法

ルールのストーリーポイントの作業レベルを見積もることは複雑です。以下は、ルールに必要な作業レベルを見積もる際に MTA が使用する一般的なガイドラインです。

作業レベル	Story Points	説明
Information	0	移行の優先度が非常に低いか、優先度のない情報警告。
Trivial	1	移行は、些細な変更または単純なライブラリスワップであり、API の変更はないか、最小限となります。

作業レベル	Story Points	説明
Complex	3	移行タスクに必要な変更は複雑ですが、解決策が文書化されています。
Redesign	5	移行タスクでは、API が大幅に変更され、再設計または完全なライブラリーの変更が必要になります。
Rearchitecture	7	移行には、コンポーネントまたはサブシステムの完全な再アーカイブが必要です。
Unknown	13	移行ソリューションは不明なため、完全な再書き込みが必要になる場合があります。

A.3.3. タスクカテゴリー

作業量レベルに加えて、移行タスクを分類してタスクの重大度を示すことができます。移行作業の優先順位付けに役立つ問題をグループ化するために、以下のカテゴリーが使用されます。

必須

移行を成功させるには、タスクを完了する必要があります。変更が行われないと、生成されるアプリケーションはビルドまたは実行に成功しません。たとえば、ターゲットプラットフォームでサポートされないプロプライエタリー API の置き換え例が含まれます。

任意

移行タスクが完了しない場合、アプリケーションは動作しますが、結果が最適になるとは限りません。移行時に変更が行われなかった場合は、移行の完了後すぐにスケジュールに配置することが推奨されます。これには、EJB 2.x コードの EJB 3 へのアップグレードが挙げられます。

Potential

移行プロセス中にタスクを調べる必要があります。しかし、移行を成功させるためにタスクが必須かどうかを判断するのに十分な詳細情報がありません。この例は、直接互換性のあるタイプがないサードパーティーのプロプライエタリタイプの移行です。

Information

タスクは、特定のファイルの存在を通知するために含まれています。これらは、モダナイゼーション作業の一部として検証または変更する必要がある場合がありますが、通常は変更が必要ありません。これには、ロギング依存関係または Maven `pom.xml` があります。

タスクの分類に関する詳細は、[ルール開発ガイドのカスタムルールカテゴリーの使用](#)を参照してください。

A.4. 関連情報

A.4.1. 参加する

Migration Toolkit for Applications が、自分のものを含むほとんどのアプリケーション設定とサーバー設定に対応できるように、以下の項目のいずれかを支援できます。

- jboss-migration-feedback@redhat.com にメールを送信し、MTA 移行ルールが対象とすべき内容をご連絡ください。
- 移行ルールをテストするためのアプリケーションの例を指定します。

- 移行が困難なアプリケーションコンポーネントおよび問題の領域を特定してください。
 - これらの問題がある移行領域について簡単な説明を記入する。
 - 問題の移行領域を解決する方法を説明する簡単な概要を記述します。
- アプリケーションで Migration Toolkit for Applications を試行します。発生している **問題を必ず報告** してください。
- Migration Toolkit for Applications ルールリポジトリへの貢献。
 - Migration Toolkit for Applications ルールを記述して、移行プロセスを識別または自動化します。
 - 新規ルールのテストを作成します。
 - 詳細は、**ルール開発ガイド** を参照してください。
- プロジェクトのソースコードへの貢献。
 - コアルールを作成します。
 - MTA のパフォーマンスまたは効率が向上します。
 - 環境の設定およびプロジェクトの設定方法に関する詳細は、**Core Development Guide** を参照してください。

あらゆるレベルの貢献が大きく評価されます。

A.4.2. 重要なリンク

- MTA フォーク: <https://developer.jboss.org/en/windup>
- MTA JIRA 問題トラッカー
 - コア MTA: <https://issues.jboss.org/browse/WINDUP>
 - MTA ルール: <https://issues.jboss.org/browse/WINDUPRULE>
- MTA メーリングリスト: jboss-migration-feedback@redhat.com
- Twitter 上の MTA: [@JBossWindup](https://twitter.com/JBossWindup)
- MTA IRC チャンネル: Server FreeNode (irc.freenode.net) (チャンネル **#windup** ([transcripts](#)))

A.4.3. MTA の既知の問題

MTA の既知の問題は、**Open MTA の問題** で確認できます。

A.4.4. MTA の問題を報告する

Migration Toolkit for Applications は、問題追跡システムとして JIRA を使用します。MTA の実行で問題が発生した場合は、JIRA Issue を作成してください。



注記

JIRA で問題を作成するために、すでに JIRA アカウントにサインアップする必要があります。

A.4.4.1. JIRA の問題を作成する

1. ブラウザーを開き、JIRA の [Create Issue](#) ページに移動します。
ログインしていない場合は、ページの右上にある **Log In** リンクをクリックし、認証情報を入力します。
2. 以下のオプションを選択し、**Next** ボタンをクリックします。
 - **プロジェクト**
コア MTA の問題については、**Red Hat Application Migration Toolkit (WINDUP)**を選択してください。

MTA ルールに関する問題については、**Red Hat Application Migration Toolkit rules (WINDUPRULE)** を選択してください。
 - **問題のタイプ: バグ**
3. 次の画面では、以下のフィールドを入力します。
 - **Summary:** 問題またはその問題の簡単な説明を入力します。
 - **Environment:** オペレーティングシステム、Java のバージョン、およびその他の関連情報の詳細を指定します。
 - **Description:** 問題の詳細情報を指定します。ログと例外トレースを必ず含めてください。
 - **Attachment:** 問題を生じるアプリケーションまたはアーカイブに機密情報が含まれておらず、MTA 開発チームと共有する場合は、**browse** ボタンを使用して問題を添付します。
4. **Create** ボタンをクリックして JIRA の問題を作成します。

改訂日時: 2023-01-23 07:41:20 +1000