HITACHI
Inspire the Next

WEBアプリケーションサーバ JBossに触ってみる ～ JBoss移行支援ツール実演 ～

株式会社 日立ソリューションズ
OSSソリューションビジネス推進センタ

HITACHI
Inspire the Next

Contents

HITACHI
Inspire the Next

1. JBoss概要

HITACHI
Inspire the Next

JBossとは

JBossはJava Web Application Server である
JBoss Application Server を中心とした
オープンソースコミュニティプロジェクトです。

JBoss Application Server 以外にも
複数のソフトウェアで構成されています。

単にJBossというと、JBoss Application Server を
指していることが多いです。

http://www.jboss.org

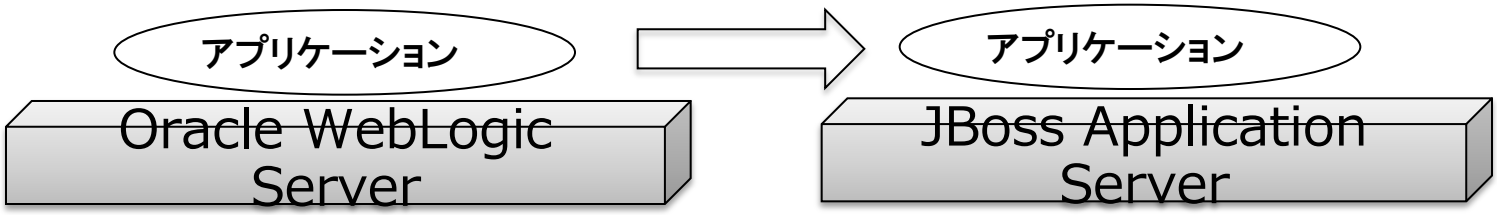
HITACHI
Inspire the Next

2. JBossへの移行

HITACHI
Inspire the Next

既存のアプリケーションをJBoss Application Serverにデプロイすること。

例えば、、



動かしてみると？

- コンパイルエラーや実行エラー発生
- 正常に動作する(ように見える)こともある

なぜ？

- Javaのバージョン差異
- アプリケーションサーバ固有のクラスに依存
- 古いバージョンのライブラリを使用

2-2 JBossに移行するメリット

HITACHI
Inspire the Next

- アプリケーションサーバのライセンスコスト、サポートコストを抑えたい
  - ✓ ライセンス費が不要となり、コスト削減が可能

- オープンスタンダードな技術を利用したい
  - ✓ 移植性の高いアプリケーションを開発可能
  - ✓ パブリッククラウドにて、多くの実績を持つオープンスタンダードな技術を採用
  - ✓ グローバルで利用可能

- 長期間システムを利用したい
  - ✓ JBoss Enterprise Application Platformであれば最大10年間の長期サポートを適用

HITACHI
Inspire the Next

本講演での対象

現状分析	設計・開発	環境構築・移行	テスト
● 移行アセスメント	● 基盤構築設計	● インフラ構築	● システムテスト
● 影響度調査	● システム移行設計	● データ移行	● 運用テスト
● 移行方式検討	● アプリケーション移行設計	● システム移行	
	● 運用設計		

現状分析フェーズでは、アプリケーションをJBossで
動作させるために、どの程度、既存アプリケーションに
改修が必要になるかを調査し、移行方式を確定します。

移行アセスメントには、静的検証と動的検証があります。

HITACHI
Inspire the Next

移行難易度や、発生し得る問題を事前に評価すること。

（1） 静的検証で調査
- 実行環境のJVMバージョン
- 準拠しているJavaEEの仕様
- 使っているライブラリ
- 依存しているアプリケーションサーバ固有のクラス
- 定義ファイル

（2） 動的検証で調査
- デプロイできるか
- 期待する動作をするか

HITACHI
Inspire the Next

3. Windupのご紹介

HITACHI
Inspire the Next

Windupとは

- JBoss 移行アセスメントツール
  JBossへの移行においてアプリケーション改修が必要になる箇所をレポート形式でHTML出力

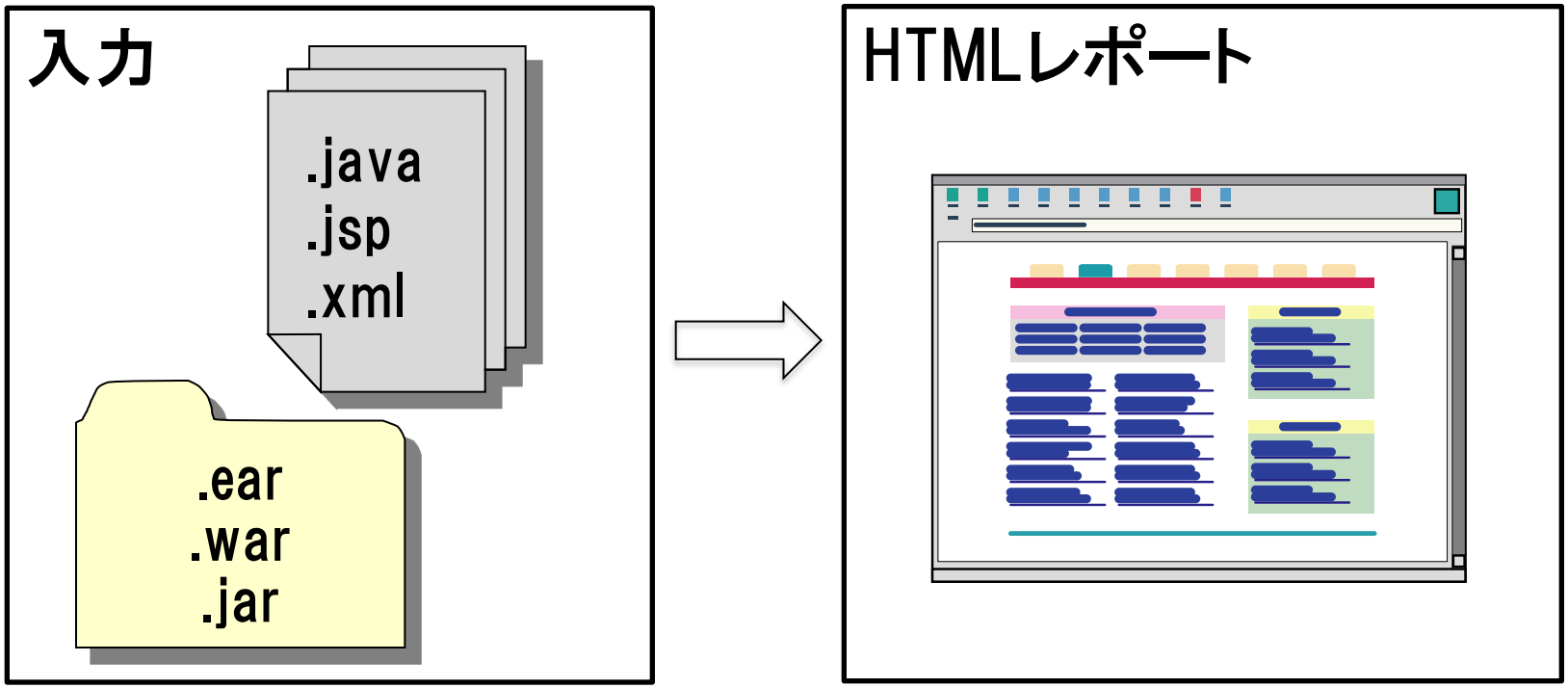
- オープンソース
  http://windup.jboss.org/index.html

- ライセンスはEPL(Eclipse Public License)
  http://www.eclipse.org/legal/epl-v10.html

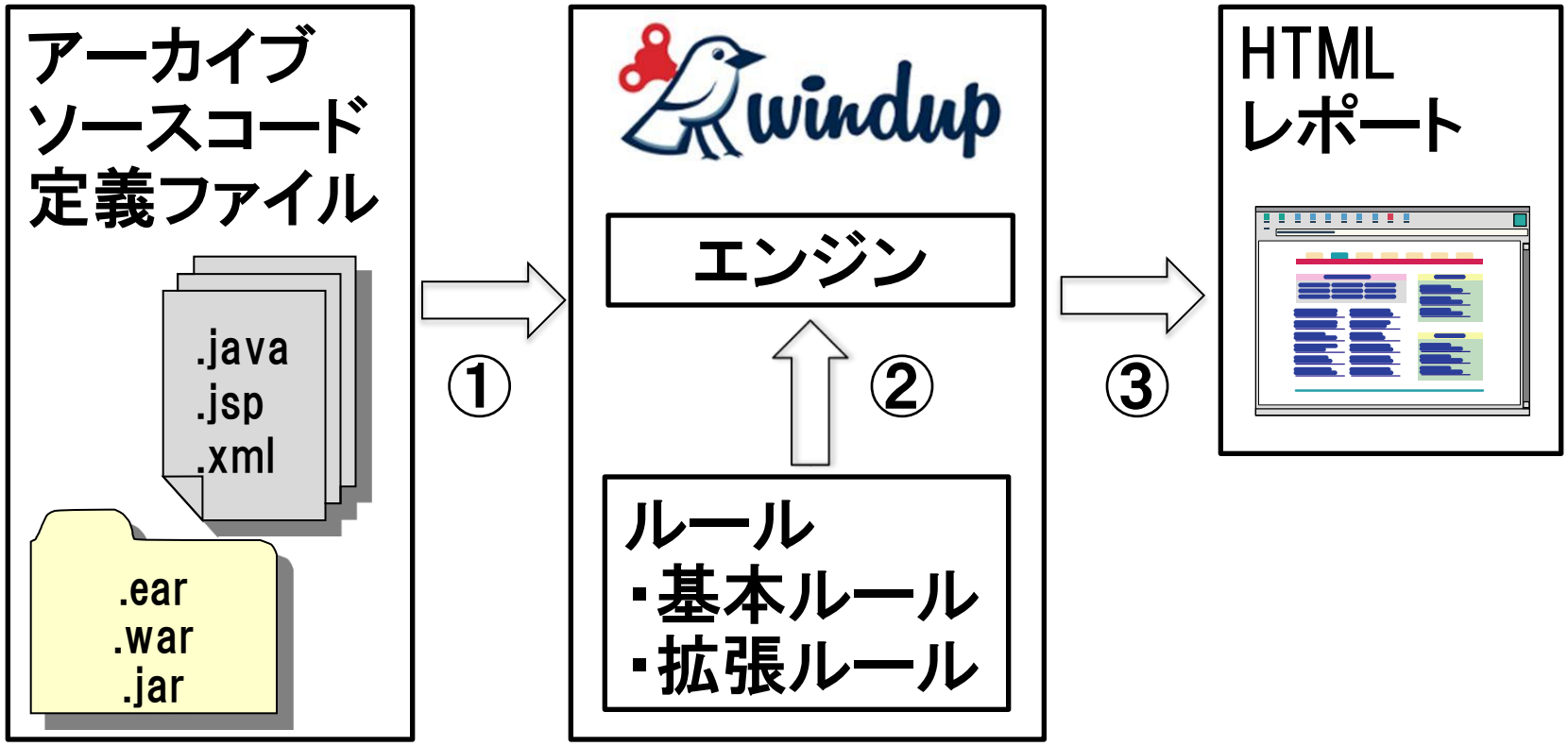
3-2 Windupでできること

HITACHI
Inspire the Next

- Javaアプリケーションを解析(.java .jsp .xmlを解析)
- 修正が必要な箇所をレポート形式でHTML出力
- 移行の規模感を把握



3-3 Windupの仕組み

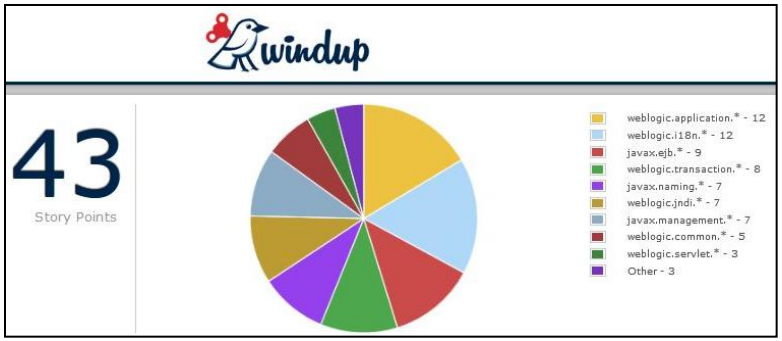


① warやear等の単位から単一ファイルでも読込可能
② ルールは自由にカスタマイズ可能
③ 評価結果をHTMLに出力

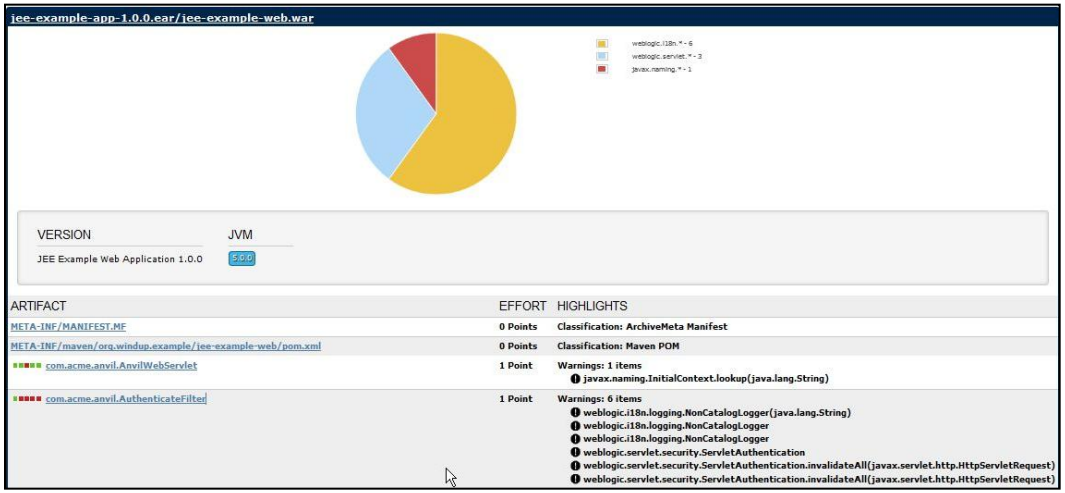
3-4 Windupのレポート

HITACHI
Inspire the Next

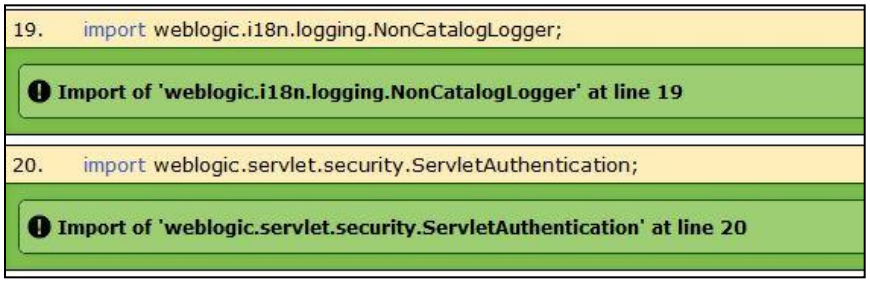
ルールに基づいて解析した結果をHTML出力します。



サマリーレポート



アーカイブレポート



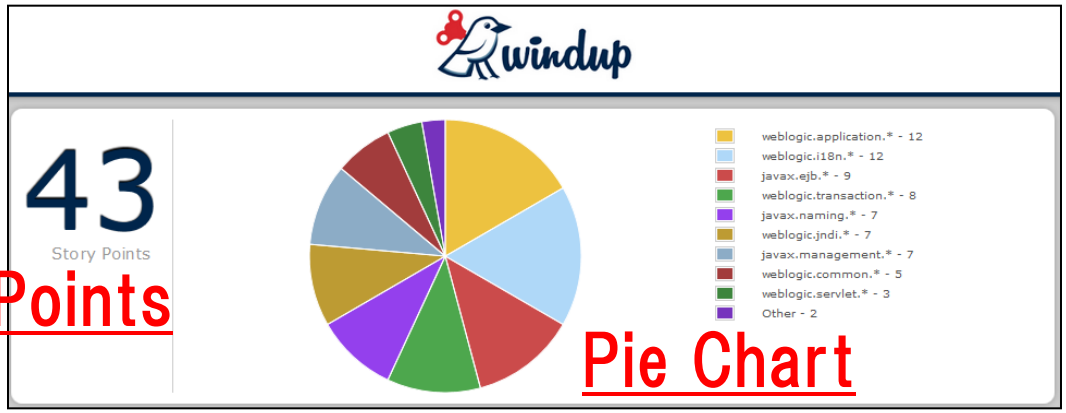
詳細レポート

3-5 サマリーレポート

HITACHI
Inspire the Next

- Story Points
  - 改修の難易度の指標を算出
  - 検証ルールに難易度の重み付けを定義
  - Story Pointsに移行作業者のスキルレベルを掛けておおよその移行工数を算出

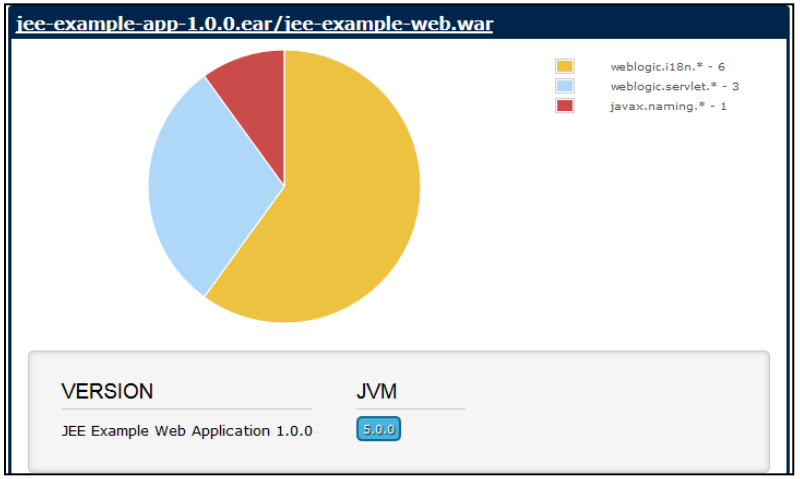
- Pie Chart
  - 変更が必要な箇所の割合を表示(全体)



3-6 アーカイブレポート

HITACHI
Inspire the Next

- Pie Chart
  - サマリーレポートを詳細化したもの(.warや.jar単位)で割合を出力



- 詳細
  - ファイル単位で改修難易度の指標を出力

ARTIFACT	EFFORT	HIGHLIGHTS
META-INF/MANIFEST.MF	0 Points	Classification: ArchiveMeta Manifest
META-INF/maven/org.windup.example/jee-example-web/pom.xml	0 Points	Classification: Maven POM
com.acme.anvil.AnvilWebServlet	1 Point	Warnings: 1 items javax.naming.InitialContext.lookup(java.lang.String)
com.acme.anvil.AuthenticateFilter	1 Point	Warnings: 6 items weblogic.i18n.logging.NonCatalogLogger(java.lang.String) weblogic.i18n.logging.NonCatalogLogger weblogic.i18n.logging.NonCatalogLogger weblogic.servlet.security.ServletAuthentication weblogic.servlet.security.ServletAuthentication.invalidateAll(javax.servlet.http.HttpServletRequest) weblogic.servlet.security.ServletAuthentication.invalidateAll(javax.servlet.http.HttpServletRequest)

3-7 詳細レポート

HITACHI
Inspire the Next

ソースコード上、修正すべき箇所を強調表示します。

HITACHI
Inspire the Next

4. デモ

4-1 デモ概要

HITACHI
Inspire the Next

デモ環境は、仮想環境(RHEL 6.2)を使用します。
環境構築からWindupの実行の流れは以下の通りです。

① Javaのインストール
② デコンパイラのインストール
③ Windupのインストール
④ Windupの設定
⑤ ツールの実行(HTML出力)

①～④についてはセットアップ済みの環境を使用します。

評価に使用するアプリケーションは、Windupのサイトで
公開されているサンプルアプリケーションを使用します。

HITACHI
Inspire the Next

6. まとめ

6 まとめ

HITACHI
Inspire the Next

- Windupを活用することで、JBossへ移行する際の修正箇所や作業量を把握することができます。
- Windupのルールは独自にカスタマイズできるため、独自フレームワークを使用したアプリケーションでも解析を行うことができます。
- WindupはOSSのため、無償で利用できます。

JBossの導入は日立ソリューションズにお任せください！！

URL：http://www.hitachi-solutions.co.jp/redhat/

HITACHI
Inspire the Next

END

WEBアプリケーションサーバ JBossに触ってみる ～ JBoss移行支援ツール実演 ～

株式会社 日立ソリューションズ
OSSソリューションビジネス推進センタ


HITACHI
Inspire the Next

HITACHI
Inspire the Next

Windupのインストールから実行の流れ

① Javaのインストール
② デコンパイラのインストール
③ Windupのインストール
④ Windupの設定
⑤ ルールのカスタマイズ（必要に応じて）
⑥ ツールの実行（HTML出力）

参考資料2　Javaのインストール

HITACHI
Inspire the Next

Windupを利用するためにはJavaが必要です。
Red Hat Enterprise Linux 6.4 に付属のOpenJDKを
利用します。

1. OpenJDKのインストール
  # yum install java-1.7.0-openjdk

参考資料3　デコンパイラのインストール

HITACHI
Inspire the Next

.classから構成されたアプリケーション(srcを含まない)
を評価するためにはデコンパイラが必要です。
Windupにはデコンパイラが同梱されていないため
コミュニティサイトで紹介されている
Jad(http://www.varaneckas.com/jad/)を使用します。

1. zipファイルのダウンロードおよび配置
　(jad158e.linux.static.zipを/opt/windupに配置します)

2. zipファイルの展開
```
# cd /opt/windup
# unzip jad158e.linux.static.zip
```

3. jadバイナリを配置
```
# cp jad /usr/bin
```

参考資料4　Windupのインストール

HITACHI
Inspire the Next

Windupのインストール方法はいくつかありますが、
最も汎用的なzipファイルでのインストール方法を
記載します。

1. Windupの実行ユーザ(OSユーザ)の作成
```
# useradd windup
```

2. zipファイルのダウンロードおよび配置
（今回windup-cli-0.6.8.zipを/opt/windupに配置済みです）

3. zipファイルの展開
```
# cd /opt/windup
# unzip windup-cli-0.6.8.zip
# chown -R windup:windup windup-cli-0.6.8
```

HITACHI
Inspire the Next

Windupのディレクトリ構成

```
windup-cli-0.6.8
|-- jboss-windup.sh                    //Windup起動ファイル　入出力ファイルはここで指定
|-- lib                                //Windupライブラリ郡
|-- rules                              //解析ルールを定義するディレクトリ
|   |-- base
|   |   |-- jboss-windup-context.xml   //解析対象のブラックリストを登録する
|   |   `-- windup                     //Windupの解析ルールのベース
|   `-- extensions
|       `-- extension-example.windup.xml    //解析ルールの拡張XMLファイル
`-- windup-cli.jar                     //Windupのmain関数を含むライブラリ
```

入力ファイル、レポート出力先はjboss-windup.shで指定します。
jboss-windup.shを実行することでレポートを出力します。

独自ルールを追加する場合、
jboss-windup-context.xml、extension-example.windup.xmlを編集します。

参考資料6　Windupの設定

HITACHI
Inspire the Next

Windupの設定はjboss-windup.shで行います。
ファイル内でjavaコマンドに入力ファイル、出力先等を指定します。

1. 起動ファイルを編集
   $ cd /opt/windup-cli-0.6.8
   $ vi jboss-windup.sh

```
java -jar windup-cli.jar -fetchRemote false -javaPkgs com.acme -
input /opt/windup/src/jee-example-app-1.0.0.ear
```

指定可能なコマンドライン引数は次ページでご紹介します。

参考資料7　Windupのコマンドライン引数

HITACHI
Inspire the Next

#	引数	説明	備考
1	-input <file> or <dir>	解析を行うファイル、ディレクトリを指定する。 -source trueを指定した場合、ディレクトリを指定する必要がある。	必須
2	-javaPkgs <string>	解析対象のプロジェクトのパッケージ名を指定する。	必須
3	-output <dir>	解析結果出力先を指定する。 指定しない場合の動作は以下となる。 (1)-inputでディレクトリ指定 -inputで指定したディレクトリと同階層に-docを付与して出力される。 (2)-inputでファイル指定 -inputで指定したファイルと同階層に-docを付与して出力される。	
4	-source <boolean>	解析対象がソースファイルかクラスファイルかを指定する。 デフォルト値はfalseでクラスファイル指定となる。ソースファイルを解析する場合は、trueを指定する。	
5	-logLevel <enum>	ログレベルを指定する。 INFO, DEBUG, WARN, ERROR, FATALで指定する。	
6	-fetchRemote <boolean>	JARファイルのPOMの取得を行う。ネットワーク接続を行わない環境では、falseを指定する。	
7	-excludePkgs	解析対象から外すプロジェクトのパッケージ名を指定する。 -source trueでも使用できる。	

参考資料8　Windupの実行

HITACHI
Inspire the Next

Windupの実行にはjboss-windup.shを使用します。

1. jboss-windup.shを実行

```
$ cd /opt/windup/windup-cli-0.6.8
$ ./jboss-windup.sh
```

```
INFO  [ClassPathXmlApplicationContext] Refreshing
org.springframework.context.support.ClassPathXmlApplicationContext@1e64a3b: startup date [Mon Oct 07
19:49:11 JST 2013]; root of context hierarchy
INFO  [CustomerPackageResolver] Found Package: com.acme
INFO  [ClassPathXmlApplicationContext] Refreshing
org.springframework.context.support.ClassPathXmlApplicationContext@11029a3: startup date [Mon Oct 07
19:49:14 JST 2013]; root of context hierarchy
INFO  [CustomerPackageResolver] Found Package: com.acme
INFO  [WindupReportEngine] Creating output path: /opt/windup/src/jee-example-app-1.0.0-ear-doc

....

INFO  [ReportEngine] Reporting complete.
```

Reporting complete.と出力されたらレポート出力完了です。