

vHutコンパイル手順書

1. ライセンス

vHut は、Apache License Version 2.0 (January 2004)に基づいて使用、若しくは利用して下さい。

Copyright 2011 NTT Software Corporation

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.

See the License for the specific language governing permissions and
limitations under the License.

なお、vHut では GNU Lesser General Public License, Version 2 でライセンスされている Granite Data Services の Code generation tools (Gas3) を使用しており、改変・再配布しています。vHut を利用・再配布する場合、これらのファイルについては、GNU Lesser General Public License, Version 2 のライセンス規約に従って下さい。Granite Data Services のダウンロード先は、下記の通りです。

<http://www.graniteds.org>

2. 商標

記載されている会社および製品の名称は、各社の商標及び登録商標です。

3. 前提条件

vHut ソースコードをコンパイルする際の、前提条件を以下に示します。

3.1. コンパイル環境

コンパイル環境として要件を満たす環境が準備されていること。

コンパイル環境要件は【別紙_開発環境構成】を参照すること。

3.2. コンパイル資材

下記配置ディレクトリを作成していること。

下記資材を解凍したディレクトリ配下を、配置ディレクトリ配下に配置していること。

- 資材
vHut¥src¥vHut サーバ¥vhut_v.zip
- 配置ディレクトリ
C:¥workspace(以降、[配置ディレクトリ]と記載)

4. 手順概要

下記に本手順の概要を示します。

- ライブラリ取得
Maven と Eclipse の連携
ライブラリ取得
- コンパイル
Flex コンパイル
Java コンパイル

5. ライブラリ取得

5.1. Maven と Eclipse の連携

(1) Eclipse workspace の設定

Eclipse IDE for Java Developers(以後 Eclipse)を起動し、workspace に[配置ディレクトリ]を指定します。

[Package Explorer]内で右クリックし、[Import] > [General] > [Existing Projects into Workspace]から、3.2 で配置した資材を workspace にプロジェクトとしてインポートします。

(2) Maven と Eclipse の連携

Eclipse の[Window]メニューバーから[Preferences]を選択します。

[Preferences]から[Maven] > [User Settings]を選択し、以下の値を設定します。

- 項目
User Settings
- 値
C:\¥maven¥conf¥settings.xml

(3) Eclipse の設定

Eclipse にて、[Package Explorer]内の vhut_server_java を右クリックし、[Properties]を選択します。

[Validation]を選択し、[Configure Workspace Settings]のリンクをクリックします。

[Disable All]ボタンを押下し、[Suspend all validators]にチェックを入れ、[OK]ボタンを押下します。

※ ライブラリ未取得で PATH 未設定のため、コンパイルエラーが発生しますが、構わず作業を進めます。

5.2. ライブラリ取得

(1) 手動でのライブラリダウンロード

<http://opensource.adobe.com/wiki/display/blazeds/BlazeDS> から

"BlazeDS binary distribution(blazeds-bin-4.0.0.14931.zip)" をダウンロード＋解凍します。

解凍した zip ファイル内に格納されている"blazeds.war"を再度解凍します。

解凍した war ファイル内に格納されている下記ファイルを[配置ディレクトリ]¥vhut_parent 配下にディレクトリを作成後、copy+rename します。

No	original (全て war ファイルのディレクトリ配下)	copy+rename 後 (全て[配置ディレクトリ]配下)
1	blazeds¥WEB-INF¥lib¥flex-messaging-common.jar	vhut_parent¥repo¥com¥adobe¥blazeds¥blazeds-common¥4.0.0.14931¥blazeds-common-4.0.0.14931.jar
2	blazeds¥WEB-INF¥lib¥flex-messaging-core.jar	vhut_parent¥repo¥com¥adobe¥blazeds¥blazeds-core¥4.0.0.14931¥blazeds-core-4.0.0.14931.jar
3	blazeds¥WEB-INF¥lib¥flex-messaging-opt.jar	vhut_parent¥repo¥com¥adobe¥blazeds¥blazeds-opt¥4.0.0.14931¥blazeds-opt-4.0.0.14931.jar
4	blazeds¥WEB-INF¥lib¥flex-messaging-proxy.jar	vhut_parent¥repo¥com¥adobe¥blazeds¥blazeds-proxy¥4.0.0.14931¥blazeds-proxy-4.0.0.14931.jar
5	blazeds¥WEB-INF¥lib¥flex-messaging-remoting.jar	vhut_parent¥repo¥com¥adobe¥blazeds¥blazeds-remoting¥4.0.0.14931¥blazeds-remoting-4.0.0.14931.jar
6	blazeds¥WEB-INF¥lib¥flex-rds-server.jar	vhut_parent¥repo¥com¥adobe¥blazeds¥flex-rds-server¥272564¥flex-rds-server-272564.jar
7	blazeds¥WEB-INF¥lib¥flex-messaging-core.jar	vhut_parent¥repo¥flex¥flex-messaging-core¥4.0.0.14931¥flex-messaging-core-4.0.0.14931.jar

http://code.google.com/p/swfobject/から

"swfobject.swfobject_2_2.zip"をダウンロード＋解凍します。

解凍した zip ファイル内に格納されている下記ファイルを修正します。

➤ ファイル名

(zip ファイルのディレクトリ配下)src¥swfobject.js

修正前	fv = "MMredirectURL=" + win.location.toString().replace(/&/g,"%26") + "&MMplayerType=" + pt + "&MMdoctitle=" + doc.title;
修正後	fv = "MMredirectURL=" + encodeURIComponent(window.location).toString().replace(/&/g,"%26") + "&MMplayerType=" + pt + "&MMdoctitle=" + doc.title;

修正したファイルを下記ディレクトリに copy します。

No.	copy 元 (zip ファイルのディレクトリ配下)	copy 先 (全て[配置ディレクトリ]配下)
1	src¥swfobject.js	vhut_client_flex_aigmod¥html-template¥swfobject.js
2	src¥swfobject.js	vhut_client_flex_appmod¥html-template¥swfobject.js
3	src¥swfobject.js	vhut_client_flex_cnfmod¥html-template¥swfobject.js
4	src¥swfobject.js	vhut_client_flex_logmod¥html-template¥swfobject.js
5	src¥swfobject.js	vhut_client_flex_mngmod¥html-template¥swfobject.js
6	src¥swfobject.js	vhut_client_flex_rolmod¥html-template¥swfobject.js
7	src¥swfobject.js	vhut_client_flex_skmportal¥html-template¥swfobject.js
8	src¥swfobject.js	vhut_client_flex_tmpmod¥html-template¥swfobject.js
9	src¥swfobject.js	vhut_client_flex_topmod¥html-template¥swfobject.js
10	src¥swfobject.js	vhut_client_flex_usrmod¥html-template¥swfobject.js

(2) Maven コマンドによるライブラリダウンロード

コマンドプロンプトにて、Maven コマンドを実行し、ライブラリをダウンロードします。

実行ディレクトリおよび実行コマンドは、【別紙_Maven コマンド一覧】の[分類=ライブラリ取得]を対象と

すること。

(3) PATH 設定確認

Eclipse にて、全プロジェクトをオープンし、[Project]メニューバーから[Clean]を選択します。

[Clean all projects]にチェックを入れ、[OK]ボタンを押下します。

[OK]ボタン押下後、正常に PATH が設定され、5.1 で発生したコンパイルエラーが発生しなくなることを確認します。

6. コンパイル

6.1. Flex コンパイル

Flex のプロジェクト(vhut_client_flex_**)を全てオープンし、[Project]メニューバーから[Clean]を選択します。

[Clean all projects]にチェックを入れ、[OK]ボタンを押下します。

[OK]ボタン押下後、コンパイルエラーが発生せず、下記ディレクトリにコンパイルされたファイルが出力されることを確認します。

- 出力ディレクトリ

[配置ディレクトリ]¥vhut_server_java¥src¥main¥webapp¥vhut_client_flex-debug¥

6.2. Java コンパイル

6.1 でコンパイルされたファイル全てを、下記ディレクトリに copy します。

- copy 先

[配置ディレクトリ]¥vhut_server_java¥src¥main¥webapp¥

コマンドプロンプトにて、Maven コマンドを実行し、ライブラリをダウンロードします。

実行ディレクトリおよび実行コマンドは、【別紙_Maven コマンド一覧】の[分類=Java コンパイル]を対象とすること。