画面のカスタマイズ

ロゴの変更

個別のページに移動

ロゴをデフォルトのOur Identity Providerから機関のロゴに変更する手順は下記の通りです。

1. ロゴファイル organization-logo.pngをedit-webapp/images/以下に配置します。

```
$ ls edit-webapp/images/
dummylogo-mobile.png  dummylogo.png  organization-logo.png
```

2. bin/build.shを実行して、war/idp.warを作り直します。

```
$ sudo bin/build.sh
Installation Directory: [/opt/shibboleth-idp]

Rebuilding /opt/shibboleth-idp/war/idp.war ...
...done

BUILD SUCCESSFUL
Total time: 16 seconds
```

3. messages/messages.propertiesのidp.logoに上記1.で配置したファイル名を設定します。 なお、ファイル名は/images/から始めます。また、 idp.logo.alt-textを設定します。

messages/messages.properties

```
# You can define message properties here to override messages defined in
# system/messages/ or to add your own messages.
idp.logo = /images/organization-logo.png
idp.logo.alt-text = Organization logo
```

差分

```
 # You can define message properties here to override messages defined in
 # system/messages/ or to add your own messages.
+idp.logo = /images/organization-logo.png
+idp.logo.alt-text = Organization logo
```

4. messages/以下にmessages_ja.properties等のファイルがあれば、それらのファイルにも3.同様の設定を行います。

messages/以下のファイルは更新したら、Tomcatの再起動等を行わなくても、しばらくすれば反映されます。

メッセージの多言語化

個別のページに移動

日本語を含む英語以外のメッセージファイルは下記ページからダウンロードできます。

- [Shibboleth wiki] IdP3 / DeployerResources / Productionalization / Internationalization / MessagesTranslation

ダウンロードしたメッセージファイルをmessages/ディレクトリ配下にコピーすることで、ブラウザに表示されるメッセージを英語以外に変更できます。
以下に記載の通り既存のmessages.propertiesよりも配置したmessages_ja.properties等が優先することになりますので、ロゴの指定等messages.propertiesに記述があればそれをmessages_ja.properties等にも記述することを忘れないでください。

messages/（およびsystem/messages/）以下のファイルは更新したら、Tomcatの再起動等行わなくても、しばらくすれば反映されます。
これはconf/services.propertiesの idp.message.cacheSeconds で制御されており、デフォルトは300で、300秒（5分）以内に反映されます。

3.2.1およびそれ以前と3.3.0およびそれ以降で配置ファイルおよび配置方法が変更になっています。3.3.0およびそれ以降にアップデートした場合は、新たに上記ページから対応するファイルをダウンロードしsystem/messages/ディレクトリに配置してください。（system/以下ですが、一度配置すれば、以降のアップデートで削除されることはありません。）としておりましたが、3.3.1へのアップデートで削除されるようですので、ひとまずmessages/ディレクトリに配置するのがよいです。

conf/services.xmlを3.3.0以降の版に更新したら、旧来のmessages/以下の3ファイルは削除可能になります。（逆に言うと、services.xmlを更新しない限り旧来のメッセージファイルが参照されていますので削除すると日本語にならなくなります。）

上記URLで提供している日本語メッセージファイルについての注意点

1. ボタンに表示するメッセージ(idp.login.login)は、Password認証フローのExtendedフローを使用した際に他の認証と合わせてアルファベットが並ぶように「Login」と英語のままです。
2. ユーザが存在しない場合のエラーメッセージ(bad-username.message)とパスワードが間違った場合のエラーメッセージ(bad-password.message)は、どちらも「ユーザ名かパスワードが違います。」と同じメッセージです。(英語では、「The username you entered cannot be identified.」と「The password you entered was incorrect.」と異なります)

表示されるメッセージの優先順位は下記の通りです。messages/以下およびsystem/messages/以下のファイルが候補になります。

1. Accept-Languageヘッダーの一番目に一致するmessages_言語_国.properties
2. Accept-Languageヘッダーの一番目に一致するmessages_言語.properties
3. Javaのシステムプロパティに一致するmessages_言語_国.properties
4. Javaのシステムプロパティに一致するmessages_言語.properties
5. デフォルトのmessages.properties

Javaのシステムプロパティの優先順位は下記の通りです。

1. コマンドラインオプション user.languageとuser.country
2. 環境変数 LC_MESSAGES
3. 環境変数 LANG

ブラウザの使用言語によらずある言語を強制する方法は以下のページに説明があります。
https://wiki.shibboleth.net/confluence/display/IDP30/Switching+locale+on+the+login+page

多言語化の方法

3.3以降で、表示するメッセージを英語から日本語などに変更する場合は、下記のメッセージファイルを用意します。文字コードはUTF-8である必要があります。

- system/messages/messages_言語[_国].properties

日本語の場合の例を下記に示します。

- system/messages/messages_ja.properties

イギリス英語の例を下記に示します。

- system/messages/messages_en_GB.properties

どうしてもlogin.jspを使い回したいのですが

個別のページに移動

以下の手順でlogin.vmの代わりにlogin.jspを使って、正常時の挙動は確認できましたが、パスワード不一致時の挙動等さらに修正すべきところがあると思われますので、お勧めできません。

1. login.jspについて、"actionUrl"となっている部分を"flowExecutionUrl"に修正し、さらにsubmitボタンに name="_eventId_proceed" を付与する。

修正例

```
@@ -17,8 +17,8 @@

         <div class="content">
           <div class="column one">
-            <% if(request.getAttribute("actionUrl") != null){ %>
-              <form id="login" action="<%=request.getAttribute("actionUrl")%>" method="post">
+            <% if(request.getAttribute("flowExecutionUrl") != null){ %>
+              <form id="login" action="<%=request.getAttribute("flowExecutionUrl")%>" method="post">
             <% }else{ %>
               <form id="login" action="j_security_check" method="post">
             <% } %>
@@ -44,7 +44,7 @@
               </section>

               <section>
-                <button class="form-element form-button" type="submit" onClick="this.childNodes[0].nodeValue='Logging in,
please wait...'">Login</button>
+                <button class="form-element form-button" type="submit" name="_eventId_proceed" onClick="this.childNodes[0].
nodeValue='Logging in, please wait...'">Login</button>
               </section>
             </form>
```

2. /opt/shibboleth-idp/views/login.vm を login.vm.dist 等にリネームした上で、上記login.jspを /opt/shibboleth-idp/edit-webapp/WEB-INF/jsp/ に配置し、build.shを実行する。
3. （必要であれば /opt/shibboleth-idp/war/idp.war を /usr/java/tomcat/webapps/ に配置する。）

login.vm等のデバッグ

login.vm等にはVelocity templateに従った書式を含めることができます。その部分をデバッグするときは、logback.xmlに以下のような行を挿入してください。

logback.xml

```
    <logger name="org.apache.velocity" level="DEBUG"/>
```