

NX^Jアプリケーションのローカライズ

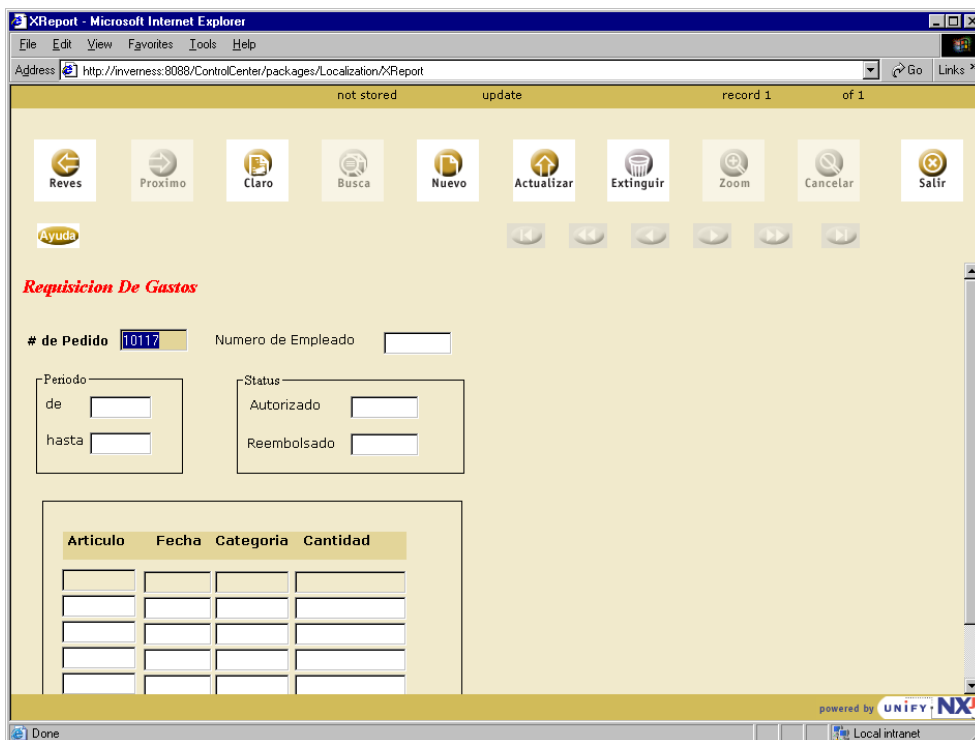
このドキュメントは、NX^Jアプリケーションをローカライズする方法を説明します。このドキュメントには、以下の章があります。

- ローカライゼーションとは？
- NX^Jアプリケーションのどの部分がローカライズできるのか？
- NX^Jアプリケーションをローカライズする手順
- 実行時にどのようなになるのか？
- 既存のNX^Jアプリケーションをローカライズ
- コントロールセンタのローカライズ

このドキュメントで説明されている機能は、リリース 9.1B またはそれ以降で利用できます。

ローカライゼーションとは？

ローカライゼーションは、特定の言語や領域用にカスタマイズしたアプリケーションのバージョンを作成することを意味します。日付と通貨のような文化的に依存するデータは、エンドユーザの地域と言語に従うフォーマットで現れます。例えば、NX^J チュートリアル の Expense Request フォームの日本語バージョンは以下の様に現れます。

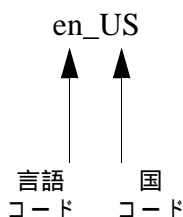


ローカライズされたデータの追加で、NX^J アプリケーションは、世界的に実行することができます。テキストエレメント、例えばフォームのステータスメッセージやラベルは、アプリケーション内でハードコードされていません。その代わりに、これらはフォーム定義 I8 階で格納され、実行時に動的に読み出されます。グラフィクスと HTML ファイルも同じく格納され、このように使用されます。新しい

言語のサポートは、アプリケーションの再コンパイルを必要としません。新しいローカライズデータと一緒に再パッケージを必要とするだけです。

NX^J アプリケーションは、指定されたロケールに基づいてローカライズされます。ロケールは、言語と地域の特定の組み合わせで設定されます。例えば、“en_US”は、アメリカ英語のロケール設定であり、そして“en_GB”は、イギリス英語のロケール設定となります。

ロケール名は、ISO 規格によって定義されています。ロケール名は、2文字の小文字の言語コード、アンダースコアの次に続くオプションで2文字の大文字の国コードが含まれます。例：



国と言語コードは、ISO 規格によって定義され、これらのコードのみ使用することができます。有効な言語コードは <http://ftp.ics.uci.edu/pub/ietf/http/related/iso639.txt> を参照します。

有効な国コードは <http://www.iso.org/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/list-en1.html> を参照します。

国コードは、他のアンダースコアの次に続く選択している可変要素です。

NX^J アプリケーションのどの部分がローカライズできるのか？

基本的には、顧客が見れるどんなものでもローカライズの対象となります。以下の図は、一般的な NX^J フォームのローカライズ項目を示しています。

The screenshot shows a web browser window titled 'XReport - Microsoft Internet Explorer'. The page displays an 'Expense Request Form' with the following elements:

- Navigation Bar:** Buttons for Back, Next, Clear, Search, New, Update, Delete, Zoom, Cancel, and Exit App.
- Form Fields:**
 - Request # (value: 10110)
 - Employee ID (value: 1105)
 - Request Period: from 10/1/2002 to 10/10/2002
 - Status: Authorized (with a text box 'Enter Authorization date') and Reimbursed.
- Table:**

ITEM	DATE	CATEGORY	XAMOUNT
1	10/5/2002	7	\$25.00
- Footer:** powered by UNIFY NXJ logo.

フォーム
タイトル

スタイル
テキスト
ボックス

ラベル

デフォルト
NX^J ツールバーラベル

グループ
ボックスラベル

表示
フォーマット

ツールチップ

他のアプリケーションエレメント、例えばアプリケーション情報フィールドと JRE からのメッセージはローカライズできないので、このドキュメントでは説明されていません。また、ターゲットフィールドに表示されるデータは、ローカライズ処理には関係ありません。そのデータは、データベースから獲得されます。

NX^J アプリケーションをローカライズする手順

NX^J アプリケーションをローカライズする手順は、各ターゲットロケール用に Static_Content フォルダ以下にフォルダを準備することが中心となります。各ロケールフォルダは、ロケールスペックデータ、例えば値にタグをマップする “application.properties” ファイルを持っています。

アプリケーションをローカライズするとき、カレントロケールでどのロケールフォルダのデータが使用されるかが決められ、デザインパネルやプレビューウィンドウに値が表示されます。タグは、カレントロケールフォルダ、それからその親フォルダ、そして最後に Static_Content フォルダを検索することで解決されます。

この章でのタスクは、新しいアプリケーションをローカライズしている状況で進められます。それが作成されると、同様にアプリケーションまたはフォームをローカライズすることができるようになります。[22 ページの「既存の NX^J アプリケーションをローカライズ」](#) 参照して下さい。この章は、以下のタスクを説明します。

- タスク 1: プロジェクトのターゲットキャラクタセットの指定
- タスク 2: フォームのローカライズの設定
- タスク 3: ターゲットロケールの指定
- タスク 4: 通貨記号の指定
- タスク 5: アプリケーションの開発
- タスク 6: ロケール固有プロパティファイルに翻訳
- タスク 7: 翻訳されたプロパティファイルをプロジェクトに追加
- タスク 8: Unify NX^J ツールバーを翻訳
- タスク 9: アプリケーションのローカライズされたフォームのテスト

タスク 1 : プロジェクトのターゲットキャラクタセットの指定

キャラクタセットは、NX^J アプリケーションの生成された JSP ページに適用するキャラクタエンコーディングの型です。適切なキャラクタセットを選択することは、アプリケーションの固有のキャラクタが Web ブラウザで表示されることを意味します。

デフォルトキャラクタセットは、西ヨーロッパとアメリカを意味する ISO-8859-1 です。アプリケーションのターゲットロケールがキャラクタセットを必要とする場合、別のキャラクタセットを指定することができます。共通のキャラクタセットは、世界のすべての言語を対象とする UTF-8 (8 ビット Unicode Transformation Format) です。

キャラクタセットを指定するステップは、以下のとおりです。

1. **プロジェクト > プロパティ > 全般** を選択します。
プロジェクトプロパティダイアログの全般ページが表示されます。
2. 作成する JSP のリストボックスのキャラクタセットでキャラクタセットを選択します。
リストされているキャラクタセットは、インターネットエクスプローラとネットスケープナビゲータでサポートされています。Unify NX^J の動作する Web ブラウザのバージョンの情報に関しては、Unify NX^J ReadMe を参照してください。
3. **適用** ボタンをクリックします。
次の作業で必要となるので、プロパティダイアログを開いたままにしておきましょう。

タスク 2: フォームのローカライズの設定

ローカライズされたフォーム、またはローカライズされなかったフォームのどちらかが、NX^J デザインセンタにあります。ローカライズされたフォームは、タグで識別される書き換えが可能なテキストのそれぞれの要素で構築されています。一旦、フォームがローカライズされると、フォームのテキストを定義する方法がローカライズされていないフォームとわずかに異なります。

プロジェクトは、ローカライズされたフォームとローカライズされていないフォームの両方を含むことができます。一旦、フォームがローカライズされると、アンローカライズすることはできません。

ローカライズされるように、プロジェクトの各新規フォームを指定するステップは、以下のとおりです。

1. プロジェクトプロパティダイアログで、全般タブをクリックします。
プロパティダイアログの全般パネルが、表示されます。
2. デフォルトのローカライズチェックボックスを設定します。
これ以降に作成する各新規フォームは、ローカライズ可能になります。
3. **適用** ボタンをクリックします。
次の作業で必要となるので、プロパティダイアログを開いたままにしておきましょう。

上記の代わりに、デフォルトフォームの作成 コマンドを使用する場合、デフォルトのデータベースコネクションパネルのローカライズチェックボックスを設定することができます。これで、デフォルトフォームの作成 コマンドによって作成されたすべてのフォームが、ローカライズ可能となります。

注 – ローカライゼーションがフォームに作用するとき、翻訳プロパティファイルがカレントロケールに適切な場合、フォームのテキストは翻訳されたフォームで表示されます。[17 ページの「タスク 9: アプリケーションのローカライズされたフォームのテスト」](#)を参照してください。

タスク 3: ターゲットロケールの指定

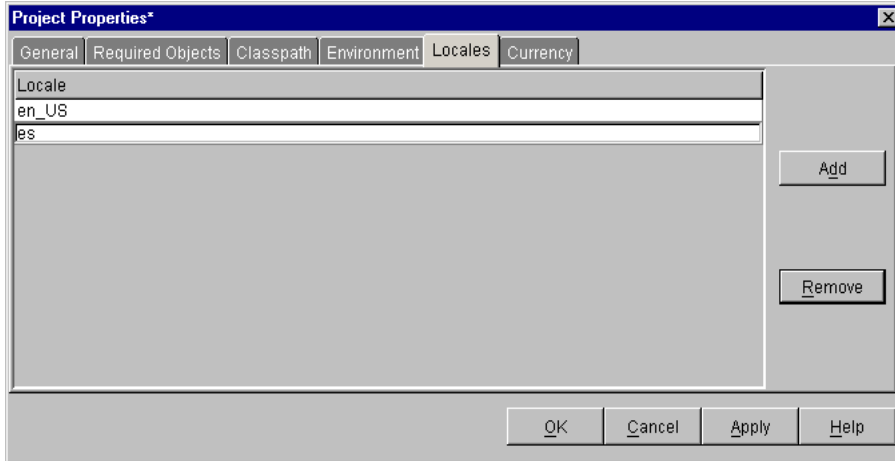
アプリケーションを実行するロケールは、言語コードと任意の国コードや相違するもので、識別される必要があります。アプリケーションが配備される実際のロケールは、プロジェクトに指定されたターゲットロケールと一致する必要があり、その他の場合は、アプリケーションの翻訳されたバージョンは表示されません。(NX^J インタラクションサーバは、実行環境のロケールがアプリケーションパッケージで定義されたものと正確に一致しない場合、アプリケーション用の適切なローカライズされたデータを見つけるために検索します。詳細については [18 ページの「実行時環境でどのようになるか？」](#)を参照してください。)

アプリケーションのターゲットロケールを指定するステップは、以下のとおりです。

1. プロジェクトプロパティダイアログのロケールタブをクリックします。
ダイアログのロケールパネルが表示されます。

-
- 最初に**追加** ボタンをクリックして、空のロケール名入力行にロケール名を入力します。

例えば、以下のロケールページに “en_US” と “es” のロケールを指定します。



- 適用** ボタンをクリックします。

次の作業で必要となるので、プロパティダイアログを開いたままにしておきましょう。

注 – このタスクで指定するロケールは、[15 ページの「タスク 7: 翻訳されたプロパティファイルプロジェクトに追加」](#)で記述されるように、Static_Content フォルダにどのように名前を付けてネストするかを決定します。

タスク 4: 通貨記号を指定

各ロケールは、地域の通貨で値の表示に適切な通貨記号を持っています。例えば、“en” のロケールでは \$123.45 で、“es” のロケールでは €123.45 です。アプリケーションは、実質的に 1 つの通貨記号だけを持つことができるので、アプリケーションでサポートされる各ロケールは、同じ通貨記号を使用しなければなりません。

通貨記号を指定するには、2つの方法があります。

- Project プロパティの場合：**プロジェクト > プロパティ > 通貨** を選択して、Currency Symbol フィールドに通貨記号を入力します。最初に、**シンボルを指定するラジオ** ボタンをクリックする必要があります。
- アプリケーションサーバに効力のあるロケールに基づいて、実行時に決定します。**プロジェクト > プロパティ > 通貨** を選択して、**シンボルを指定するラジオ** ボタンをクリックします。

通貨記号を指定するための詳細は、『Unify NX』開発者ガイド』の第3章「プロジェクトの設定」を参照して下さい。

タスク 5: アプリケーションの開発

ローカライズされたデータが“application.properties”ファイルで獲得することができるように、前のタスクはプロジェクトを設定しました。

ここでは、以下のように異なった NX』アプリケーション開発を続けることができます。

- フォームにテキストエレメントを設定するとき、テキストエレメントに関連したタグを入力します。プロジェクトの各テキストは、ユニークなタグを持ちます。
- jBiz スクリプトで提供される任意のメッセージ（例えば、**displayToMessageBox** メソッドを使用）は、同じようにそれらにユニークなタグを割り当てる必要があります。
- query-by-form 操作が表示フォーマットの文字と衝突する場合、ユーザが利用可能な操作を変更することで衝突を防ぐかどうかを決める必要があります。

これらのトピックスは、以降の章で説明されます。

テキストエレメントにタグを設定する

アプリケーションのフォーム上のすべてのテキストエレメントは、タグにマップされます。テキストエレメントは、以下のとおりです。

- フォームのタイトル (Title プロパティ)
- タブのタイトル (Title プロパティ)

- ラジオグループとグループボックスのタイトル (Title プロパティ)
- ラベル (Title プロパティ)
- ラジオグループに含まれるボタンラベル (Title プロパティ)
- スタイルテキストボックス (ボックスのテキスト)
- リストボックスとドロップダウンリストのラベル
- ツールチップ (Tooltip プロパティ)
- FYI Message (FYI Message プロパティ)
- フォームメニューラベル (エントリポイント定義)

テキストのすべての翻訳可能なタグを作成したことを確認するためのチェックリストとして、このリストを使用することができます。

原文の要素やプロパティをダブルクリックするか、ポップアップメニューからテキストの編集フォームを選択すると、タグの編集ダイアログが表示されます。



既存しているタグを見るため、または、選択するためにここをクリックします。

タグの編集ダイアログ上でタグフィールドは、テキスト部分のタグ名を示しています。新しいタグ名を作成するか、既存のタグ名を選択することができます。新しいタグ名を作成するか、既存のタグ名を選択することができます。タグ名は、文字で始まらなければなりません。タグ名は、文字、数字、下線を含むことができます。さらに、**新規** ボタンをクリックすることで、ユニークなタグ名を NX^J に作成することができます。

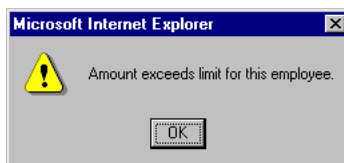
タグの値への変更は、カレントロケールの “application.properties” ファイルに設定されます。新しいタグは、デフォルトの application.properties ファイルに追加されます。

既存のタグ名を使用するには、ドロップダウンリストから値を選択します。タグに対応している値は、値フィールドで表示されます。可能な限り、翻訳コストをセーブするために、同じ値を持つすべてのテキストエレメントには単一のタグを使用するほうが良いでしょう。例えば、ラベル “Employee” は、いくつかのフォームに表示されますが、1つのタグだけを必要とします。

必要がある場合、値フィールドでカレントタグのテキストを入力します。これは、フォーム上（この場合、ラベル）に表示されるテキストです。このタスクには、デフォルトの言語でテキストを書き込みます。値の翻訳されたタグは、ここには表示しません。

メッセージ用のタグ設定

NX^J アプリケーションは、ユーザに情報を提供するためにメッセージを使用することができます。例えば、入力された総計が特定のフィールドの制限を超える場合、NX^J チュートリアルでは、右に示すメッセージボックスが使用されます。さらに、メッセージは、フィールドコントロールや、FYI Message アプリケーション情報フィールドを作成するために使用することができます。



各メッセージをローカライズ可能にするためには、メッセージにユニークなタグを指定することが必要です。application.properties ファイルに、これを設定します。例えば、先に説明されるメッセージは、application.properties に以下のエントリを含むことでできます。

“OverLimitMessage=Amount Exceeds Limit for this Employee.”

フォームスクリプトは、ファイルからメッセージを取り出す必要があり、それからその目的とするタグに、獲得したメッセージを書き込みます。

例：

```
String overLimit;
if ( (! ( limit.isNull() ) ) && ( XAMOUNT > limit ) )
{
    overLimit=session.getMessage
("OverLimitMessage");
    session.displayToMessageBox (overLimit);
}
```

jBiz スクリプトにメッセージを埋め込む代わりにメッセージファイルを使用することは、コードメンテナンスを著しくシンプルにするだけでなく、実行時のメモリ使用も減らします。

プロパティファイルについての詳細は、[15 ページの「タスク 7: 翻訳されたプロパティファイルプロジェクトに追加」](#)を参照してください。

検索演算子の衝突を防ぐ

ユーザがフォームに検索条件を入力するとき、演算子として特別な文字を使用することができます。例えば、“s*” は、“s” から始まるフィールドのすべての文字列を検索するというを意味し、“1-6” は、1 と 6 の間のすべての数値を検索するというを意味します。演算子として、デフォルトで使用される文字は、データベースに依存します。詳細については、『Unify NX^J ユーザーズガイド』の第 2 章「レコードの検索」を参照してください。

いくつかのケースで演算子は、データ値のフォーマットで使用される同じ特殊文字です。そして、特殊文字は NX^J インタラクションサーバによって誤解されます。

例えば、値の範囲を指定するデフォルトの検索演算子は、ハイフン (-) です。いくつかのロケールで、それは日付表示フォーマット (MM-dd-yyyy) と衝突することがあります。ユーザが、12-17 のような値を入力した場合、それが 12 月 17 日であるのか値の範囲 (12, 13, 14...17) であるのかは、NX^J インタラクションサーバには判別できません。

コンマ (,) は、グループセパレータ (アメリカとイギリスロケール) または基数セパレータ (フランスロケール) のときも、衝突が発生します。これらのロケールでは、デフォルトのメタキャラクタの値のセパレータは、同じくコンマです。

例えば、101.2 のフロートポイント値は、フランスロケールで 101,2 のように入力されます。これは、2 つの値 (10 と 2) か、1 つのフロートポイント値 (101.2) のようなどちらかに解釈されます。

アプリケーションで使用する検索演算子を変更することで、衝突を防ぐ必要があります。アプリケーションに影響する演算子を変更するには、以下のどちらかで行うことができます。

- 必要とする演算子を含むために、ロケールに固有のプロパティファイルを変更します。例えば：
 - `_NXJSearchOrOperator=;`
 - `_NXJSearchRangeOperator=*`
- アプリケーションの jBiz スクリプトの 1 つに、NXJSession ファウンデーションクラスの `searchRangeOperator` か `searchOrOperator` プロパティを設定します。実行時に決定されるロケール値に基づく、これらの値を変更することができます。[18 ページの「実行時環境でどのようになるか？」](#)を参照してください。

タスク 6：ロケール固有プロパティファイルに翻訳

前のタスクでテキストのタグを作成したことによって、“application.properties” ファイルは、タグと対応するテキスト含むために更新されています。また、そのファイルにメッセージのタグを入力することもできます。オリジナルテキストが書かれている言語は、*デフォルトの言語*であることはご存知ですね。

例えば、デフォルトの言語が英語の場合、チュートリアルアプリケーションの application.properties ファイルは、以下の通りです。

```
#Fri Dec 20 14:18:38 PST 2002
XAMOUNT=XAMOUNT
Request=Request \#
DATE=DATE
CATEGORY=CATEGORY
Reimbursed=Reimbursed
Status=Status
Authorized=Authorized
to=to
from=from
ITEM=ITEM
RequestPeriod=Request Period
EmployeeID=Employee
OverLimitMessage=Amount Exceeds Limit for this Employee.
```

テキストエレメントを
識別するが
表示しないタグ

表示するタグ

このファイルは、翻訳される際のベースファイルとして使用されます。
“application.properties” ファイルは、プロジェクトの Static_Content フォルダにあります。このファイル名と場所は、以下のとおりです。

<NXJWork>/projects/<yourproject>/sources/Static_Content/application.properties

このファイルのコピーを作成して、各ターゲットロケール用に翻訳します。翻訳されるテキストは、各エントリのイコール記号の右にテキストを置き換えなければなりません。例えば、翻訳されたスペイン語プロパティファイルは、次のとおりです。

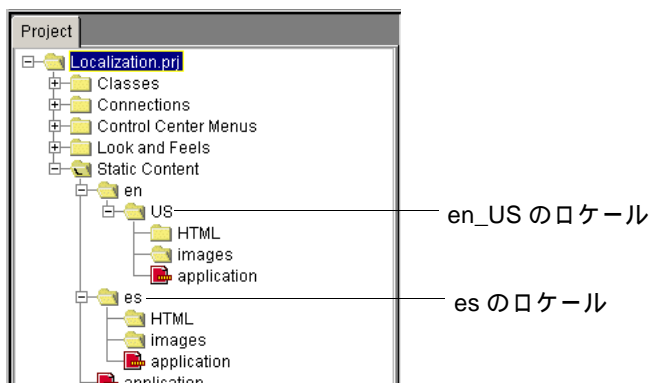
```
#C:/Unify/NxjWork/projects/tutorials/completed/
JBiz/sources/Static_Content/application.properties
#Fri Dec 20 14:18:38 PST 2002
XAMOUNT=XAMOUNT
Request=El Pedido\#
DATE=La Fecha
CATEGORY=La Categoria
Reimbursed=Reembolsado
```

Status=La Posicion
Authorized=Autorizado
to=a
from=de
ITEM=El Articulo
RequestPeriod=El Periodo
EmployeeID=El Empleado Identificacion
OverLimitMessage=La cantidad excede el límite para este empleado.

翻訳のために別の会社にプロパティファイルを送ることができ、NX^J デザインセンタで直接ファイルを編集することもできます。NX^J デザインセンタでプロパティファイルを編集する場合、スクリプトエディタと同様のエディタが使用されますが、カラーコーディングはありません。

タスク 7: 翻訳されたプロパティファイルをプロジェクトに追加

各翻訳されたプロパティファイルは、Static Content フォルダの適切なロケールフォルダでプロジェクトに追加されなければなりません。フォルダは、ディレクトリレベルでアンダーラインを含むロケール名を意味する名前であればなりません。例えば：



各翻訳されたプロパティファイルの場合、以下のステップに従います。

1. ブラウザパネルのプロジェクトビューで、Static Content フォルダを右クリックします。
コンテキストメニューが表示されます。
2. **新規 > フォルダ**を選択します。
3. 右クリックして、名前の変更を選択します。
ロケール名の最初のエレメントを入力します。例えば、“en”。
4. **OK** ボタンをクリックします。
新しいフォルダが追加されます。
5. ロケール名の各追加の部分のために、ステップ 2 から 4 を繰り返します。
6. 必要であれば、イメージと HTML フォルダを作成します。
各ロケールフォルダ内に、イメージフォルダを作成することができます。イメージがカレントロケールのイメージフォルダで存在する場合、Static Content/images フォルダのイメージの代わりに使用されることとなります。
各ロケールフォルダ内に、HTML フォルダを作成することができます。HTML ページがカレントロケールの HTML フォルダで存在する場合、Static Content/HTML フォルダの HTML の代わりに使用されることとなります。
7. ロケールフォルダに翻訳されたプロパティファイルを置きます。
8. ロケールフォルダのそれぞれのフォルダに、ロケールに固有の HTML とイメージファイルを追加するために、**プロジェクト > ファイルを追加** を選択します。

タスク 8: Unify NX^J ツールバーを翻訳

アプリケーションが Unify NX^J が提供するツールバーの 1 つを含んでいる場合、ボタンのラベルを翻訳することができます。ツールバーの翻訳をするには、ユーザが HTML の経験と JavaScript プログラマであることを必要とします。ツールバーを翻訳する代わりに、フォームのボタンとしてユーザが必要とするコマンドを提供しています。ボタンテキストは、プロパティファイルを使用することで、ローカライズ可能です。

ローカライズされた標準ツールバーの日本語版は、<NXJHome>\web-app\ja で Unify NX^J に含まれます。

ツールバーのローカライズされたバージョンを作成するステップは、以下のとおりです。

1. タスク7で作成したターゲットロケールディレクトリにツールバーのHTMLファイルと関連するイメージフォルダをコピーします。

Unify NX^J ツールバーは、NX^J <NXJHome>\webapp にあります。いくつかのツールバーの違ったバージョンを提供しています。デフォルトのツールバーは、“horizontalmenu.html” と名付けられています。

イメージフォルダは、ツールバーの様々なボタンのために .gif ファイルを含みます。各ボタンが “on” と “off” バージョンを持つことに注意します。必ずすべてのイメージフォルダをターゲットディレクトリにコピーして下さい。

ツールバーの作成についての詳細は、

http://www.unify-jp.com/products/nxj/dev_resources.htm で「カスタム NX^J ツールバーを作成」を参照してください。

2. 例えば、Adobe Photoshop のような、グラフィックデザインツールで、ラベルテキストがターゲットロケールにローカライズされるように、ラベルを持つ各 .gif ファイルを編集します。

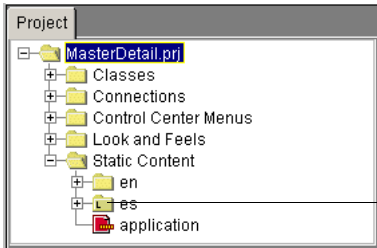
タスク9: アプリケーションのローカライズされたフォームのテスト

ロケール固有の “application.properties” ファイル、イメージ、HTML ファイルを有効にするために、ロケールにカレントロケールを設定する必要があります。カレントロケールが設定されていない場合、テキストはデフォルトの言語で表示されます。

フォームで翻訳されたテキストを見ることができるよう、カレントローケールの設定を変更するステップは、以下のとおりです。

1. 動作させたいロケールに対応しているフォルダを右クリップし、現在のロケールに設定 を選択します。

カレントロケールに対応しているフォルダは、Static Content フォルダのアイコン上に大文字の “L” でフラグを立てます。



カレントロケール

デザインパネルのフォームは、カレントロケールに基づいて更新されます。それは、翻訳された用語がタグのためのプロパティファイルに存在する場合、それは表示されます。用語が、まだ翻訳されていなければ、フォームの外観は変わりません。

2. デザインパネルの各フォームを検査します。

例えば、翻訳されたバージョンのラベルがデフォルト言語より長ければ、フォーム上の位置の調整を必要とします。

実行時環境でどのようになるか？

NX^J アプリケーションが最初に実行されたとき、ユーザの Web ブラウザは NX^J インタラクショナルサーバにロケールを知らせます。“application.properties” ファイルがそのロケールで見つかった場合、それが使用されます。それが見つからない場合、次の上位レベルの “application.properties” ファイルが、マッチするか、上位レベルファイルに達するまで検索されます。このセクションは、以下のトピックスを説明します。

- タグの階層を検索
- 実行環境のロケールの識別と設定

タグの階層を検索

カレントロケールの “application.properties” ファイルが、アプリケーションのすべてのタグを含む必要はありません。NX^J インタラクショナルサーバは、タグを検索するためにアプリケーションパッケージの全プロパティファイルを検索します。

最低レベルのディレクトリから探し始め、“application.properties” ファイルと指定したタグを探します。そこで見つからない場合、NX^J インタラクティブサーバは、次のより高いディレクトリを見ます。一番上のレベルで見つけれない場合、表示するための値としてタグ名を使用します。

実行環境のロケールの識別と設定

前に説明されるように、実行環境のロケールは、表示する正しくローカライズされたアプリケーションのためのアプリケーションパッケージで定義されるロケールと一致する必要があります。デフォルトで、NX^J アプリケーションが開始するとき、ユーザの Web ブラウザに効力のあるロケールが使用されます。

以下のセクションで説明されるようにロケールは、アプリケーションユーザ、またはアプリケーションの jBiz 文によって変更することができます。

SIInternet Explorer ユーザのロケール設定

Internet Explorer Web ブラウザのロケールを設定するステップは、以下のとおりです。

1. **ツール > インターネットオプション > 全般 > 言語** を選択します。
2. **追加** ボタンをクリックします。
3. ターゲットロケールに対応する言語をクリックします。
4. **OK** ボタンをクリックします。
言語エントリが、言語の優先順位ダイアログに追加されます。
5. 項目を選択して、リストの一番上に新しい言語を移動するために、**Move Up** ボタンを使用します。
6. **OK** ボタンをクリックします。

Netscape Navigator ユーザのロケール設定

Netscape Navigator Web ブラウザのロケール設定のステップは、以下のとおりです。

1. **編集 > 設定** を選択します。

-
2. **Navigator** > **言語** を選択します。
 3. **追加** を選択して、リストから使いたい言語を選択します。
 4. **OK** ボタンをクリックします。
 5. 項目を選択して、新しい言語をリストの一番上に移動するために、**上に移動** ボタンを使用します。
 6. **OK** ボタンをクリックします。

NX^J コマンドによるロケールの設定

NXJSession ファウンデーションクラスの `clientLocale` プロパティを使用することで、Web ブラウザのロケールの設定をオーバーライドすることができます。これを行うには、アプリケーションのエントリーフォームにコマンドを定義します。例えば：

```
BEFORE FORM
...
COMMAND switchLocaleToSpanish
{
session.clientLocale = new java.util.Locale("es", "MX");
}
```

コマンドはイベントに基づいて実行されるか、あるいはコマンドをロケール国フラグを表示するイメージボタンにマップすることができます。それから、ユーザはフラグ ボタンをクリックすることによって、フォームに必要な言語を選択することができます。



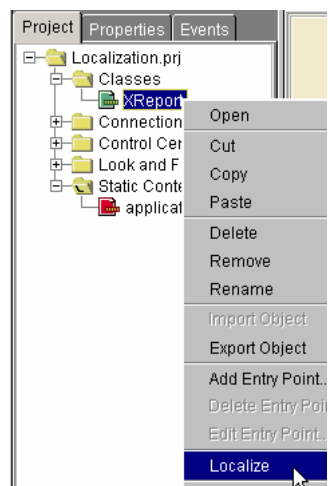
一旦、ロケールが変更されると、プロパティファイルからの新しいロケールのためにローカライズされたデータが、フォームの表示に使用されます。

注 - この方法でロケールを変更する場合、フィールドに表示されている Date, Time, Float と Amont は、フィールドコントロールの値が更新されるまで、ローカライズされたフォーマットで再表示されません。

既存の NX^J アプリケーションをローカライズ

ローカライズなしで作成されたフォームをローカライズするには、ブラウザパネルのプロジェクトビューでフォームのエントリを選択し、右クリックをしてローカライズを選択します。

各フォームの各テキストエレメントのためにタグと値のエントリの変更と確認をします。詳細については、[9 ページの「タスク 5: アプリケーションの開発」](#)を参照してください。



コントロールセンタのローカライズ

以下の図は、ローカライズできるコントロールセンタビューの項目を示しています。



メニュー項目 コントロールセンタ メニューラベル
 ツールバーラベル

NXJ コントロールセンタのメニューで使用されるラベルのロケール対応版を作成し、管理することができます。エントリポイントがメニューに表示される場合、そのラベルが使用されます。そのラベルを表示するとき、その最も近くで一致するロケール対応が使用されます。ロケールが見つからない場合、デフォルトのラベルが使用されます。

ローカライズされたラベルを作成する前に、NX^J コントロールセンタのロケールを設定する必要があります。これを行うには、**Control Center Properties > Locales** を選択します。

詳細については、『Unify NX^J 管理者ガイド』を参照してください。