

# 顧客向けに重点を置いた Web 画面

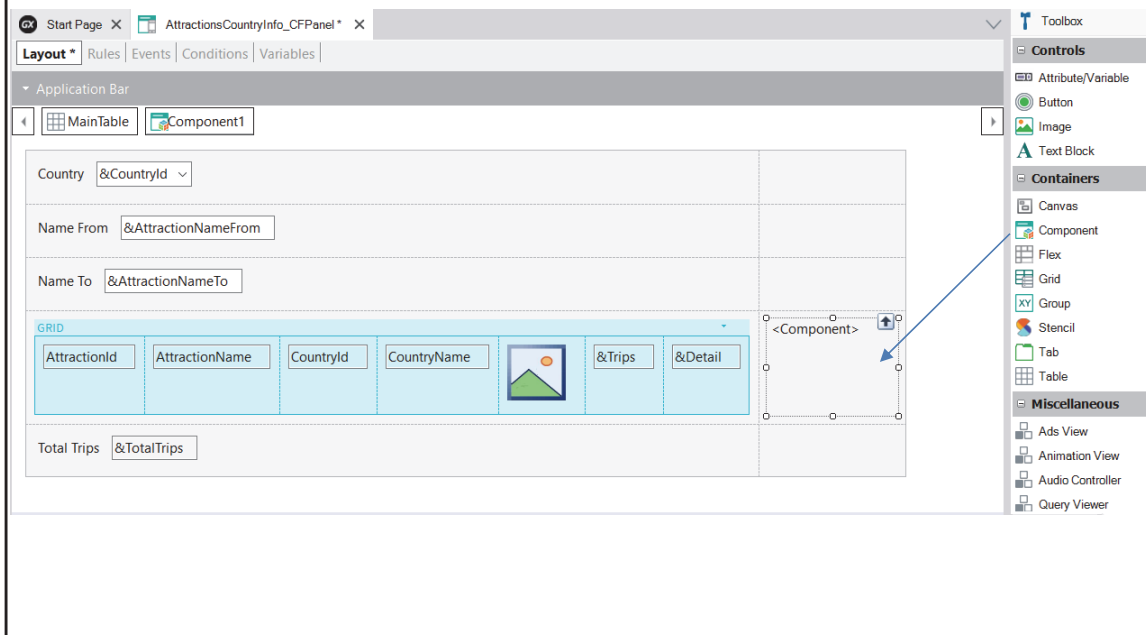
コンポーネントと Master Panel オブジェクトの使用

*GeneXus*<sup>™</sup>

# コンポーネント

Panel オブジェクトを使用すると、Web コンポーネントで構成された Web パネルのように、コンポーネントを使用して機能をカプセル化することができます。これを例を使って説明します。

## パネル内部での別のパネルの使用



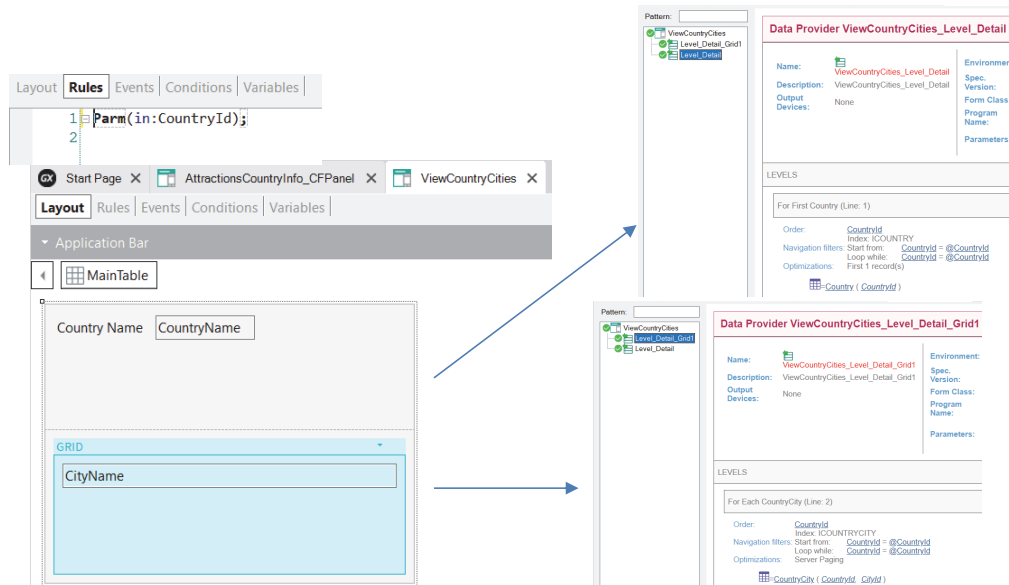
選択した国の観光名所とデータを表示する Panel オブジェクトを構築するために、今回は、作成したパネル内にコンポーネントを使用して情報を展開します。

コンポーネントを含むパネルを作成するには、Attractions\_CFPanel を別名で保存し、AttractionsCountryInfo\_CFPanel という名前を付けます。

ツールボックスからコンポーネントコントロールを選択し、観光名所グリッドの右側に挿入します。

前に行ったように、グリッドで国の名前をクリックすると、その国に関する情報が開きます。

## コンポーネントとして使用するパネルの作成



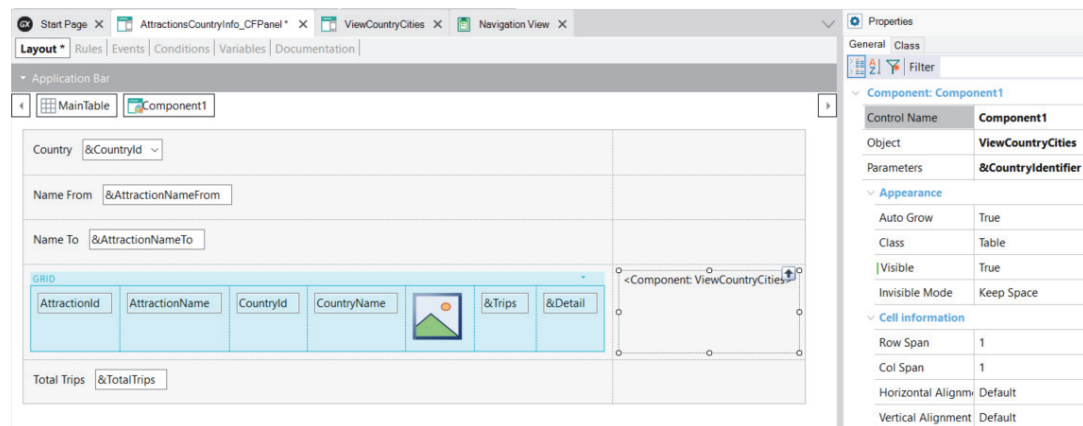
次に、コンポーネントに関連付けるパネルを作成します。名前は ViewCountryCities にします。ツールバーから CountryName 項目属性をドラッグし、CityName 項目属性を含むグリッドをドラッグします。Country.City でグリッドにベーストランザクションを割り当てます。

次に、CountryId 項目属性を含む Parm を追加します。

パネルのナビゲーション表示を確認すると、Level\_Detail ノードで、Country テーブルが Parm ルールの CountryId 項目属性でフィルタリングして照会され、パラメーターで受け取った国に対応する 1 つのレコードが取得されることが分かります。

Level\_Detail\_Grid1 ノードでは、同様にパラメーターで受け取った CountryId でフィルタリングして CountryCity テーブルが照会されることが分かります。

## コンポーネントのプロパティ



次に、Component1 の [Object] プロパティで ViewCountryCities パネルを選択し、[Parameters] に &CountryIdentifier 変数を追加します。この変数には、選択したグリッドの明細行の CountryId 項目属性の値が保存されます。

ここで、Web パネルとの関連で相違点が 1 つあります。それは、コンポーネントコントロールを挿入するときに、作成済みの任意のパネルを挿入できることから、ViewCountryCities\_CFPANEL をコンポーネントにするのに、タイプの変更が不要なことです。

## コンポーネントとして使用されるパネルの動的作成



観光名所グリッドで国名をクリックしたときにコンポーネントパネルを呼び出す必要があるため、その呼び出しで CountryName 項目属性の Tap イベントをプログラムします。

まず、選択された CountryName に対応する CountryId を定義済みの &CountryIdentifier 変数に保存してから、Create コマンドを使用して、パラメーターで選択された国を渡すことによって、Component1 に関連付けられた ViewCountryCities\_CFPANEL オブジェクトを動的に作成します。

構築しているパネルは既にメインになっているため、[実行] を使用して実行します。

## 実行時...

| Name From           | Name To       | Country |  |   |         |
|---------------------|---------------|---------|--|---|---------|
| Christ the Redemmer | Brazil        |         |  | 1 | Details |
| Cinque Terre        | Italy         |         |  | 0 | Details |
| Eiffel Tower        | France        |         |  | 1 | Details |
| Forbidden city      | China         |         |  | 0 | Details |
| Glenfinnan Viaduct  | Scotland      |         |  | 0 | Details |
| London Towers       | England       |         |  | 0 | Details |
| Long Bridges        | United States |         |  | 0 | Details |
| Louvre              | France        |         |  | 0 | Details |
| Matisse Museum      | France        |         |  | 1 | Details |
| Meet the            | China         |         |  | 0 | Details |

Total Trips: 3

| Country Name | City Name |
|--------------|-----------|
| China        | Beijing   |
|              | Shanghai  |
|              | Hong Kong |

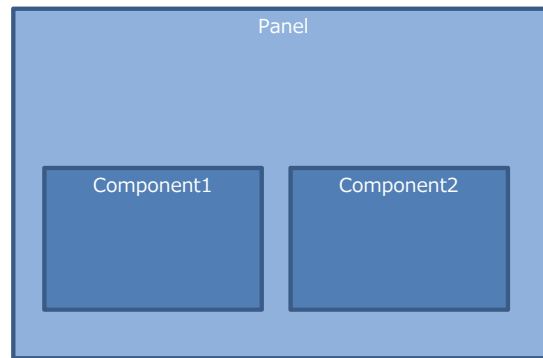
中国をクリックすると、中国の都市である北京、上海、および香港が表示されます。

## イベントのトリガー順序

ClientStart (Panel)  
Start (Panel)  
Refresh (Panel)  
Load (Panel)

ClientStart (Component1)  
Start (Component1)  
Refresh (Component1)  
Load (Component1)

ClientStart (Component2)  
Start (Component2)  
Refresh (Component2)  
Load (Component2)



別のパネル内にコンポーネントとして挿入するパネルオブジェクトには、パネルと同じイベント、つまり、クライアント側のシステムイベント (ClientStart および Back) とサーバー側のイベント (Start、Refresh、および Load) が含まれています。

これは、ホストパネルのイベントとの関連で、パネルにある 1 つまたは複数のコンポーネントからイベントがトリガーされる順序を考慮する必要があることを意味します。

最初にトリガーされるイベントは、コンポーネントを含むパネル内のイベントで、前に説明した通常の順序、つまり ClientStart、Start、Refresh、Load の順にトリガーされます。

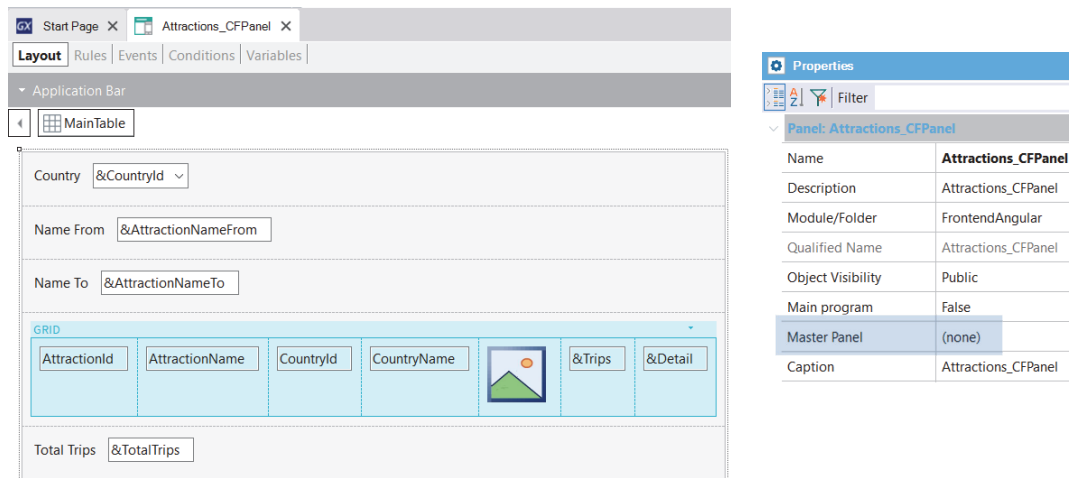
次に、上から下、左から右に向かって、コンポーネントとして含まれている各パネルのイベントが同様に通常の順序でトリガーされます。



# Master Panel オブジェクト

次に、Master Panel オブジェクトについて見てみましょう。

## Master Panel オブジェクト



The screenshot displays the GeneXus IDE interface for the **Attractions\_CFPanel** object. The **Layout** tab is active, showing a form with the following components:

- Country**: A dropdown menu with the value `&CountryId`.
- Name From**: A text field with the value `&AttractionNameFrom`.
- Name To**: A text field with the value `&AttractionNameTo`.
- GRID**: A table with columns: `AttractionId`, `AttractionName`, `CountryId`, `CountryName`, `&Trips`, and `&Detail`.
- Total Trips**: A text field with the value `&TotalTrips`.

The **Properties** window on the right shows the configuration for **Panel: Attractions\_CFPanel**:

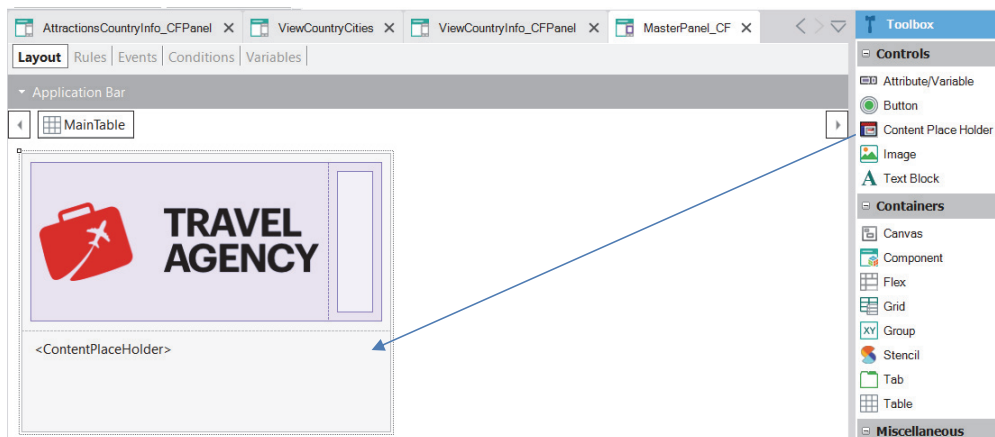
| Property          | Value               |
|-------------------|---------------------|
| Name              | Attractions_CFPanel |
| Description       | Attractions_CFPanel |
| Module/Folder     | FrontendAngular     |
| Qualified Name    | Attractions_CFPanel |
| Object Visibility | Public              |
| Main program      | False               |
| Master Panel      | (none)              |
| Caption           | Attractions_CFPanel |

Panel オブジェクト **Attractions\_CFPanel** のプロパティを見ると、そのうちの 1 つに **[Master Panel]** プロパティがあり、パネルに空で表示されることが分かります。これは、Panel オブジェクトが配置されてロードされるマスターページがあることを示しています。

この概念は、Web パネル用の Web マスターパネルの概念に似ています。

ただし、後者とは異なり、Panel オブジェクトにはタイプを変更するためのプロパティがありません。ここで、Master Panel オブジェクトは、Panel オブジェクトから独立したオブジェクトです。

## Master Panel オブジェクト (続き)



次に、MasterPanel\_CF という名前のマスター パネル タイプのオブジェクトを作成します。

ツールバーに移動すると、パネルオブジェクトがロードされるコンテナコントロールであるコンテンツ プレース ホルダーが利用可能であることが分かります。

次に、コンテンツ プレース ホルダーをパネルに挿入します。次に、その上にテーブルを挿入し、テーブル内に代理店のロゴが入った画像を挿入します。その右側にテーブルを配置して、ロゴが左揃えになるようにします。

## Master Panel オブジェクト (続き)

**Table: MainTable**

|               |           |
|---------------|-----------|
| Control Name  | MainTable |
| ▼ Appearance  |           |
| Columns Style | 100%      |
| Rows Style    | 100dip;pd |
| Width         | 100%      |
| Height        | 100%      |
| Auto Grow     | True      |

**Table: Table1**

|               |             |
|---------------|-------------|
| Control Name  | Table1      |
| ▼ Appearance  |             |
| Columns Style | 320dip;100% |
| Rows Style    | 90dip       |
| Width         | 100%        |
| Height        | 90dip       |

**Image Class Hierarchy**

- Image
  - ImageHeader\_CF
    - ImageHeaderItems
      - Scale Type: Fit

テーブルのプロパティと MainTable のプロパティを調整して、画像が適切に表示されるようにします。また、クラス ImageHeader\_CF を作成し、[Scale Type] プロパティに値 Fit を割り当て、そのクラスをロゴ画像に割り当てます。これを保存します。

## Master Panel オブジェクト (続き)

### Panel: Attractions\_CFPanel

|                   |                            |
|-------------------|----------------------------|
| Name              | <b>Attractions_CFPanel</b> |
| Description       | Attractions_CFPanel        |
| Module/Folder     | FrontendAngular            |
| Qualified Name    | Attractions_CFPanel        |
| Object Visibility | Public                     |
| Main program      | <b>True</b>                |
| Master Panel      | <b>MasterPanel_CF</b>      |
| Caption           | Attractions_CFPanel        |

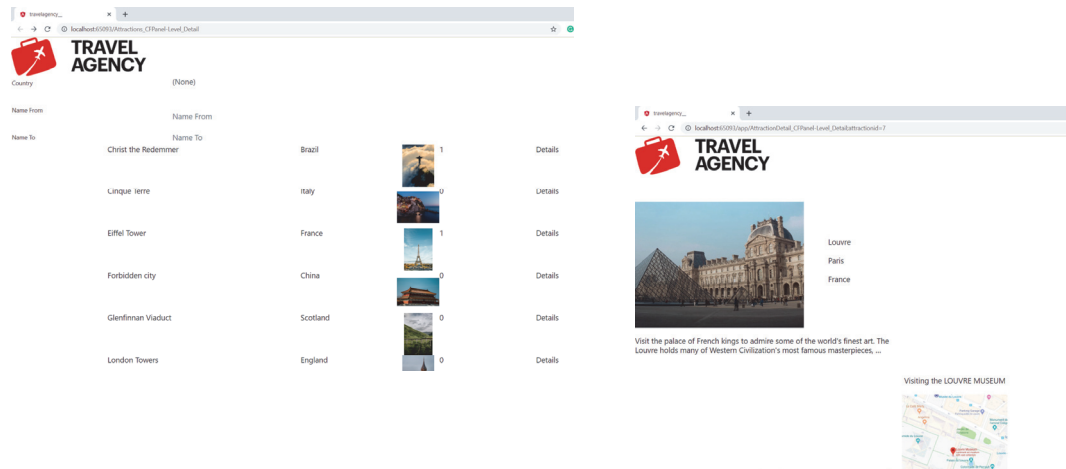
### Panel: AttractionDetail\_CFPanel

|                   |                                 |
|-------------------|---------------------------------|
| Name              | <b>AttractionDetail_CFPanel</b> |
| Description       | Attraction Detail_CFPanel       |
| Module/Folder     | FrontendAngular                 |
| Qualified Name    | AttractionDetail_CFPanel        |
| Object Visibility | Public                          |
| Main program      | False                           |
| Master Panel      | <b>MasterPanel_CF</b>           |
| Caption           | Attraction Detail_CFPanel       |

次に、作成した MasterPanel\_CF オブジェクトを含む Attractions\_CFPanel パネルと AttractionDetail\_CFPanel パネルに [Master Panel] プロパティを割り当てます。

そして実行します。

## 割り当てたマスターパネルを使用したアプリケーションの実行



代理店のロゴが表示され、その下に Attractions\_CFPANEL がロードされることが確認できます。フランスの [Details] をクリックすると、観光名所の詳細パネルにも同じロゴが表示されます。

今回は、Master Panel オブジェクトのデザインシステムの詳細は考慮しませんが、Theme オブジェクトとクラスの使用方法は Web パネルの場合と似ていると言えます。

ここで作成したデザインは、以降の章でも使用します。