Representing enumeration values

To use the ImageSwitcher and ActiveImage to represent an enumeration value

1. Create an ImageSwitcher element, and specify the enumeration, whose values will be represented by an ActiveImage of the ImageSwitcher, as the tagged value of the **represent** tag definition of the ImageSwitcher element.
2. Create an Active image for each enumeration literal owned by the enumeration.
3. Specify an image that will be the **image** attribute of each created ActiveImage, and set the enumeration literal that will be represented by the ActiveImage as the **activeElement** tag definition.

You can use *test_imageswitcher_for_enum.mdzip* as your example. This file is a sample project of Magic Model Analyst. It is located in the samples /simulation/tests folder of the MagicDraw installed directory. In this example, an ImageSwitcher was created to represent a VerdictKind enumeration. An ActiveImage was created for each enumeration literal of the VerdictKind as follows
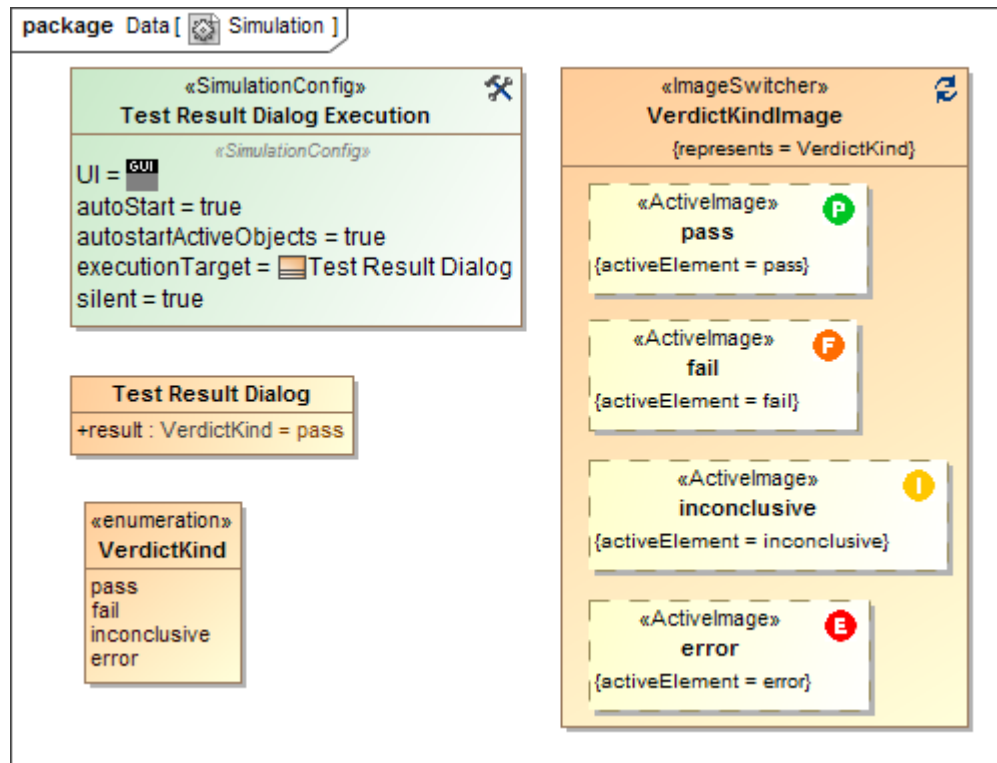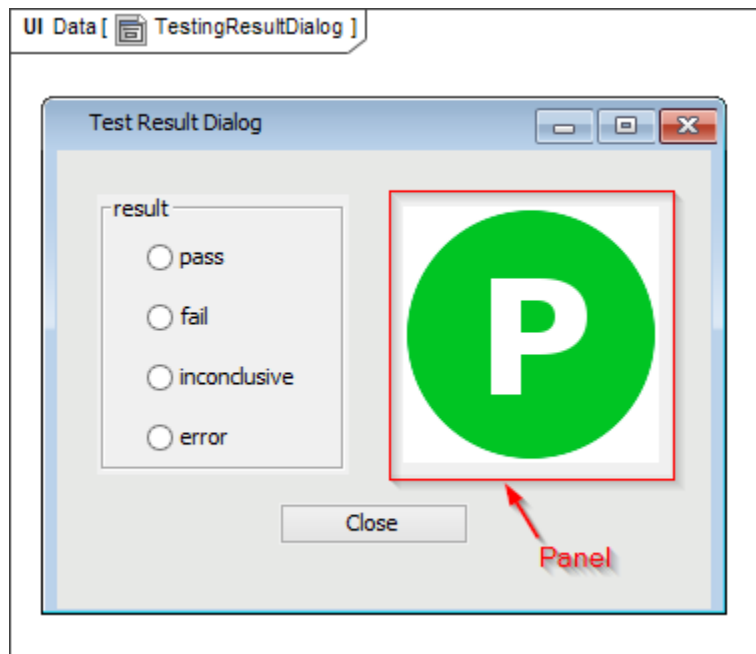

Image switcher and Active Image representing Enumeration and Enumeration Literals.

If a property is typed by an enumeration, you can use the ImageSwitcher and the ActiveImage to represent it with a UI frame mockup of the Classifier that owns the property. Once you drag the property to the panel owned by the UI frame mockup, Magic Model Analyst will prompt you to select either an ImageSwitcher or a group of radio buttons to represent the value. If you select the ImageSwitcher, the ActiveImage that represents the enumeration literal, which is the default value of the property, will open in the panel or in the group box to which the property has been dragged.

The figure below shows the User Interface Modeling diagram of the *test_imageswitcher_for_enum.mdzip* sample project. It contains a UI frame that represents the **Test Result** dialog, which is the Classifier owning the property typed by the VerdictKind enumeration. The property is **result:VerdictKind,** whose default value is **pass**. Now, the ActiveImage that represents the value **pass** will be shown in the panel.

UI Modeling diagram containing a UI frame with Image Switcher.

When the UI mockup opens in the MagicDraw window during model simulation, the image shown in the panel containing the ImageSwitcher will be the ActiveImage that represents the enumeration literal, which is the runtime value of the property.

You can simulate the Test Result dialog in the sample project *test_imageswitcher_for_enum.mdzip*. You can also change a result value by selecting one of the radio buttons that represents the value: pass, fail, inconclusive, or error. Now the image that corresponds to the value will appear.
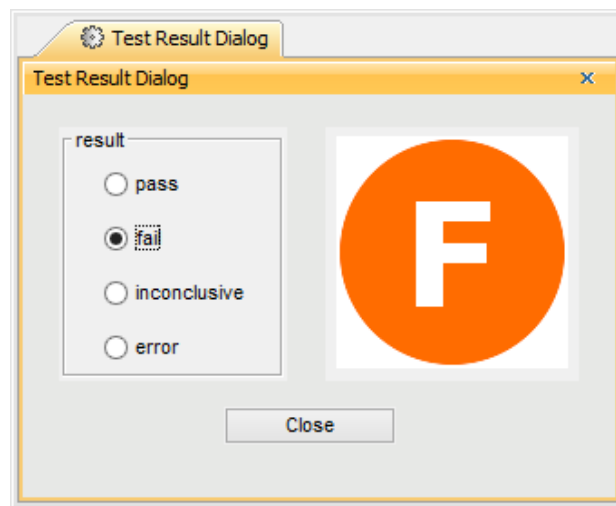

UI Mockup whowing the property's value typed by enumeration.

You can view more information on reusing ImageSwitcher in the UI mockup in Nested UI Configuration stereotype.

The image that appears on the web UI through ImageSwitcher is the same image that represents the current State of simulation UI. If the image on the simulation UI is changed, the image on the Web UI will change accordingly and vice versa. Setting the **show empty image for unspecified value** option (true/ false) on the «ImageSwitcher» will change the image on the web UI through ImageSwitcher to be the same as the image that represents the simulation UI.
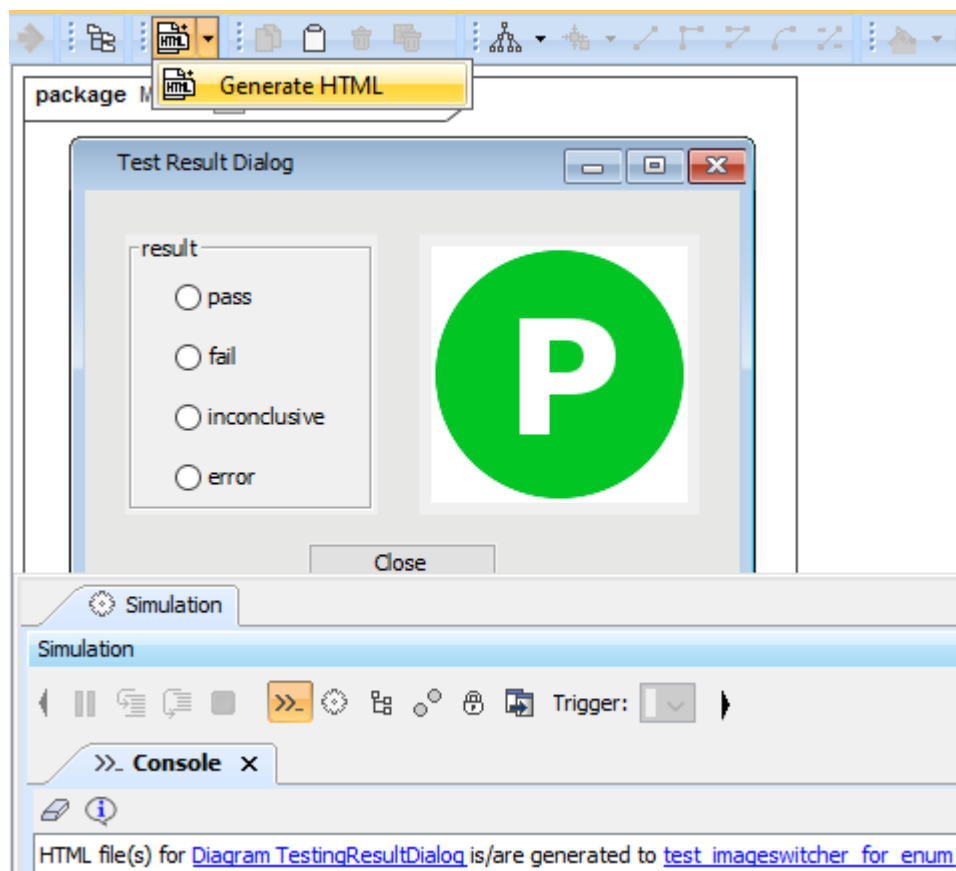
The generated HTML button on the toolbar of the UI diagram.
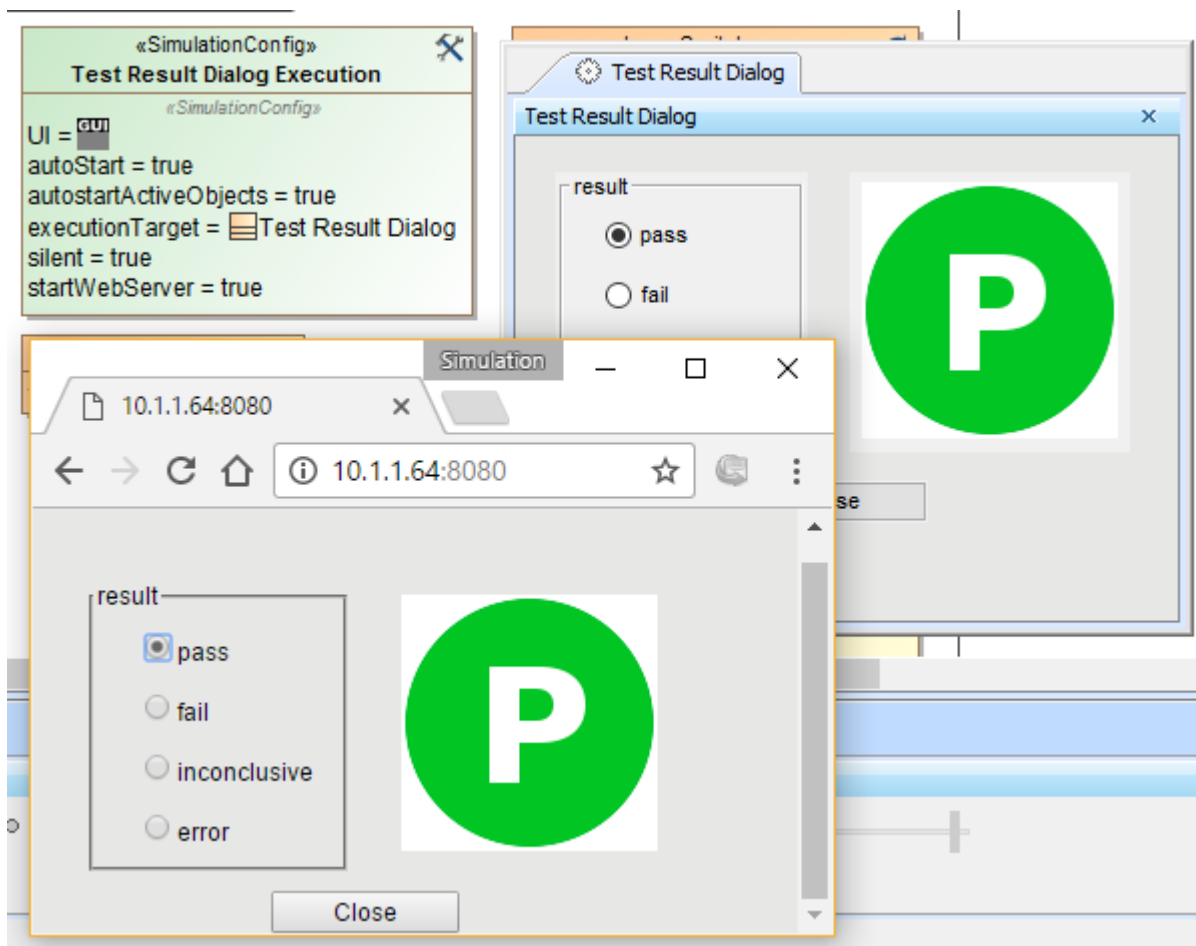
ImageSwitcher on the web UI representing the enumeration value that corresponds to the simulation UI.

Another more comprehensive and realistic sample is available in Magic Model Analyst is *Blackjack.mdzip*. This sample contains a Blackjack game simulation in which Players can only stand or hit, and the Dealer can stand on 17 or higher. Once you run the Run Blackjack Simulation Config, the Blackjack game will begin by showing a Blackjack game table UI (see the following blackjack figure).

Note that this sample also demonstrates the **Nested UI Configuration** feature (see Nested UI Configuration stereotype for more information). As the Dealer/Player has 7 possible card slots, according to the **Nested UI Configuration** feature, the Dealer/Player thus has 7 corresponding Value Properties, e.g., 'Card1', 'Card2', 'Card3', and so on.
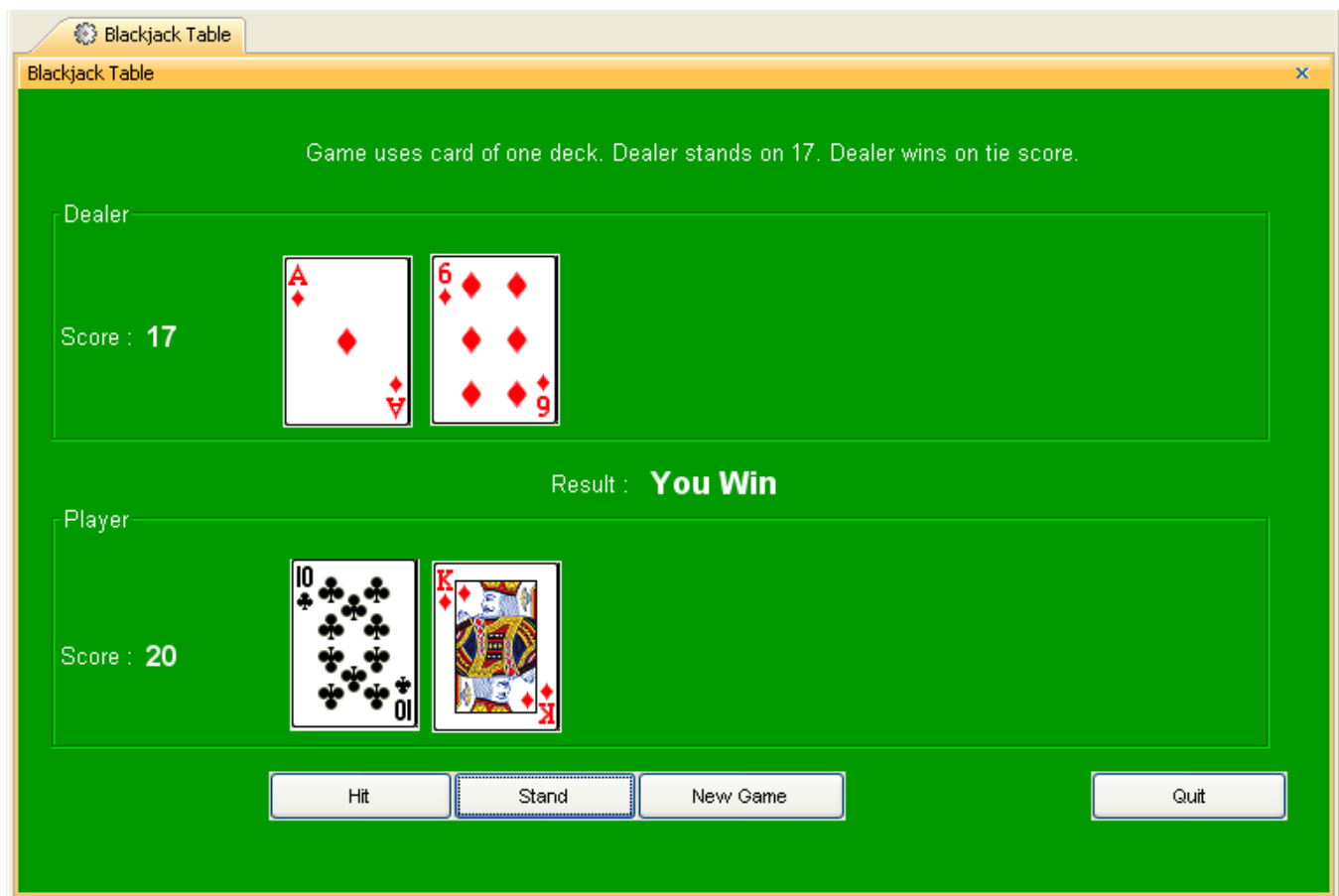
UI showing Blackjack Table in the Blackjack.mdzip sample.

The image showing the card slots demonstrates the application of ImageSwitcher and enumeration. Each card slot is actually a UI Panel having the **Nested UIConfig** stereotype applied in which the **NestedUIConfig's config** tag is set to the **Card Image Switcher** ImageSwitcher, and the **NestedUIConfig's feature** tag is set to the corresponding Value Property, e.g., 'Card1'. The **Card Image Switcher** ImageSwitcher is set to represent the **Card Value** enumeration. The **Card Image Switcher** ImageSwitcher contains 53 ActiveImages, each of which is the corresponding Enumeration Literal of the **Card Image** enumeration. Note the **Card Value** enumeration also contains 53 Enumeration Literals.

- '0' Enumeration Literal represents a blank image; a card slot without any card.
- '1' - '52' Enumeration Literals represent the images of 52 distinct cards.

«ImageSwitcher»
Card Image Switcher

{represents = Card Value}

| «ActiveImage» 0 {activeElement = 0}			
«ActiveImage» 1 {activeElement = 1}	«ActiveImage» 14 {activeElement = 14}	«ActiveImage» 27 {activeElement = 27}	«ActiveImage» 40 {activeElement = 40}
«ActiveImage» 2 {activeElement = 2}	«ActiveImage» 15 {activeElement = 15}	«ActiveImage» 28 {activeElement = 28}	«ActiveImage» 41 {activeElement = 41}
«ActiveImage» 3 {activeElement = 3}	«ActiveImage» 16 {activeElement = 16}	«ActiveImage» 29 {activeElement = 29}	«ActiveImage» 42 {activeElement = 42}
«ActiveImage» 4 {activeElement = 4}	«ActiveImage» 17 {activeElement = 17}	«ActiveImage» 30 {activeElement = 30}	«ActiveImage» 43 {activeElement = 43}
«ActiveImage» 5 {activeElement = 5}	«ActiveImage» 18 {activeElement = 18}	«ActiveImage» 31 {activeElement = 31}	«ActiveImage» 44 {activeElement = 44}
«ActiveImage» 6 {activeElement = 6}	«ActiveImage» 19 {activeElement = 19}	«ActiveImage» 32 {activeElement = 32}	«ActiveImage» 45 {activeElement = 45}
«ActiveImage» 7 {activeElement = 7}	«ActiveImage» 20 {activeElement = 20}	«ActiveImage» 33 {activeElement = 33}	«ActiveImage» 46 {activeElement = 46}
«ActiveImage» 8 {activeElement = 8}	«ActiveImage» 21 {activeElement = 21}	«ActiveImage» 34 {activeElement = 34}	«ActiveImage» 47 {activeElement = 47}
«ActiveImage» 9 {activeElement = 9}	«ActiveImage» 22 {activeElement = 22}	«ActiveImage» 35 {activeElement = 35}	«ActiveImage» 48 {activeElement = 48}
«ActiveImage» 10 {activeElement = 10}	«ActiveImage» 23 {activeElement = 23}	«ActiveImage» 36 {activeElement = 36}	«ActiveImage» 49 {activeElement = 49}
«ActiveImage» 11 {activeElement = 11}	«ActiveImage» 24 {activeElement = 24}	«ActiveImage» 37 {activeElement = 37}	«ActiveImage» 50 {activeElement = 50}

| «ActiveImage»
12
\|{activeElement = 12} | «ActiveImage»
25
\|{activeElement = 25} | «ActiveImage»
38
\|{activeElement = 38} | «ActiveImage»
51
\|{activeElement = 51} |
| «ActiveImage»
13
\|{activeElement = 13} | «ActiveImage»
26
\|{activeElement = 26} | «ActiveImage»
39
\|{activeElement = 39} | «ActiveImage»
52
\|{activeElement = 52} |
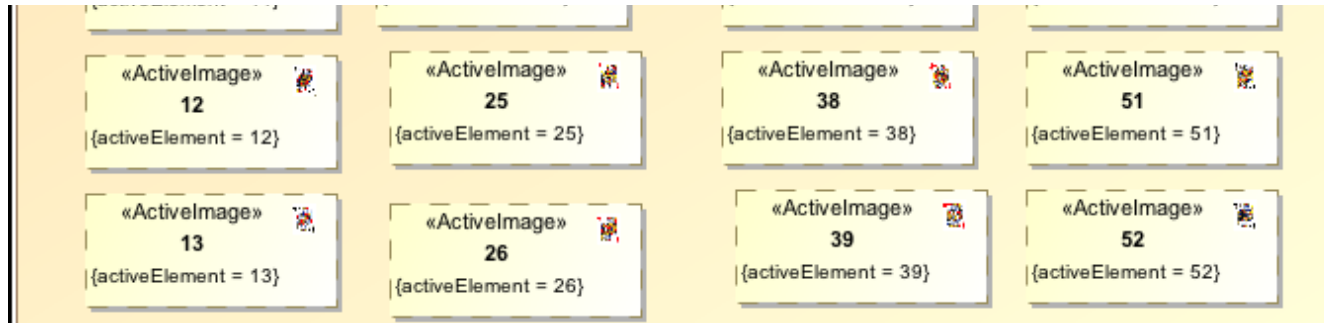
Image Switcher containing active image of 52 cards and blank image.

When the value of one of the 7 Value Properties changes, the card slot image will change accordingly. See the Player's cards in the figure to better understand how it works as follows

- The value of the Player's 'Card1' Value Property is '49' Enumeration Literal, which is the activeElement of the '49' ActiveImage. The '49' ActiveImage represents the ten of spades.
- The value of the Player's 'Card2' Value Property is '28' Enumeration Literal, which is the activeElement of the '28' ActiveImage. The '28' ActiveImage represents the two of hearts.
- The value of the Player's 'Card3' Value Property is '0' Enumeration Literal, which represents a blank card slot. Therefore, no card is open next to the two of hearts card.

UI and Variables panel showing the value of 7 value properties and the displayed image.

Related pages

- ImageSwitcher and ActiveImage
- Representing object states