

## Text labels provided with elements

For elements that provide a visual label, the visual label will automatically be provided as the accessible name of the element. Examples of such elements include Button, CheckBox, RadioButton, LinkButton, and Menu components. In the following Button declaration, for example, the `accessibilityName` will be "Search" because that is the button's label:

```
<s:Button x="296" y="28" label="Search"/>
```

If the label does not sufficiently describe the purpose of the button, the developer should explicitly set the `accessibilityName` property.

## Text labels positioned near an element

If a text label is positioned near an element, developers can define the text label as part of a parent `FormItem` container in Flex. The `FormItem` will export the label of the Flex control it is grouped with. In the following example, the `TextInput` element is grouped with a "Search" text label.

```
<mx:FormItem label="Search">
<s:TextInput width="252" height="21" accessibilityDescription=
    "Enter a search term into this field."/>
</mx:FormItem>
```

## No text labels present

For items that do not provide a text label, such as images, custom controls, and complex controls, the name of the element can be exported either via the `toolTip` attribute or directly through the `accessibilityName` property of the element. When the `toolTip` for an element is set, the `toolTip` text will be appended to the accessible name of the element.

```
//Option one - use the accessibilityName attribute
<mx:Image width="60" height="56" source="assets/icecreampint.jpg"
    accessibilityName="Ice Cream Pint"/>
//Option two - use the toolTip attribute
<mx:Image width="60" height="56" source="assets/icecreampint.jpg"
    toolTip="Ice Cream Pint"/>
```

**Note:** When no text label is present for a standard component (or when a label is provided as part of the element itself) do not include "image", "button", or any other description of the role of the component in the accessible name. This information will be announced automatically by screen reader software.

## Setting the accessible name at runtime

If the accessible name changes at runtime or it cannot be set at compile time because the name of an element is unknown, the accessible name can be set using ActionScript and the convenience accessor `accessibilityName`. When the accessor is used, the developer neither needs to verify that an `AccessibilityProperties` object has been created nor call `updateProperties()`, as these are handled automatically; for example:

```
btnSearch.accessibilityName = "Search";
```

If the accessor is not used, those extra steps are required:

```
if (!btnSearch.accessibilityProperties)
    btnSearch.accessibilityProperties = new AccessibilityProperties();
btnSearch.accessibilityProperties.name = "Search";
Accessibility.updateProperties();
```

## Accessible Spark components and containers

The following Spark components and containers (<http://goo.gl/JeZcz>) are accessible:

- ButtonBar component
- Button component