So far we have learned about the basics of the Silverlight technology platform and core capabilities of Silverlight for media and data integration and networking. This chapter is solely focused on new and extended features and capabilities provided in Silverlight 4. New features and capabilities of Silverlight 4 extend the user interface and functionalities by providing an opportunity to developers to develop true line-of-business (LoB) applications and thus improve the user experience significantly.

Enhancements in Silverlight 4

Silverlight 4 introduces the following key enhancements, enabling the development of LoB applications and providing a better and more familiar user experience:

- One of the widely implemented features for any LoB application is drag-and-drop functionality. Silverlight 4 allows UIElements as a drop target and thus enables the implementation of drag-and-drop functionality by introducing the UIElement.AllowDrop property and related drag-and-drop events.

- Silverlight 4 extends the UIElement.MouseWheel event implementation through the managed code, which allows mouse-wheel support in full-screen mode as well as in an application running in out-of-browser mode.

- Like drag-and-drop functionality, right-click functionality is also a popular feature for any LoB application. With Silverlight 4, two new MouseRightButtonDown and MouseRightButtonUp events for UIElement are introduced to handle the right-click functionality and implement required custom features.

- Another critical feature of any LoB application is printing capability. Silverlight 4 introduces this capability and enables you to print either the existing visual tree or a custom virtual visual tree, using PrintDocument class. The printing functionality can be implemented in both in-browser and out-of-browser modes.

- Silverlight 4 also introduces the System.Windows.Clipboard class to provide access to the system clipboard to copy and paste data.

- Silverlight 4 extends the capabilities by supporting bi-directional text and adding support to many more languages, including right-to-left languages. This is critical for developing global aware applications.
Silverlight 4 brings the Silverlight version of the XAML parser one step closer to the regular WPF parser. We will take a look at key enhancements before ending this chapter.

Drag-and-Drop Functionality

Drag-and-drop functionality is always a favorite feature for developers to implement and it also brings a great end-user experience. Silverlight 4 allows Silverlight UIElements as a drop target and thus enables the implementation of the drag-and-drop functionality for partially trusted in-browser and out-of-browser (OOB) applications, as well as elevated-trusted OOB applications.

We already looked at a brief example of drag-and-drop functionality in Chapter 1, and we learned about key property and events of UIElement, enabling drag-and-drop functionality, in Chapter 3. Let’s revisit the properties and events of UIElement that are required to enable the drag-and-drop functionality.

Properties of UIElement to Enable Drag-and-Drop Functionality

The System.Windows.UIElement class contains two properties—AllowDrop and Visibility—that need to be enabled for the user control to make it a drop target.

The AllowDrop dependency property is the key property to enable drag-and-drop functionality, which is introduced in Silverlight 4. Set it to true if you want to allow that specific UIElement to be a drop target. The default value of AllowDrop property is false.

Along with the AllowDrop property setting of true, the Visibility property of that specific UIElement must be set to Visibility.Visible, to make the control visible and allow it to be a drop target. Note that Visibility.Visible is the default value of the UIElement, so you do not need to explicitly mention that.

Events of UIElement to Enable Drag-and-Drop Functionality

The System.Windows.UIElement class contains four new events—DragEnter, DragLeave, DragOver, and Drop—in Silverlight 4 that can be implemented to enable drag-and-drop functionality. The AllowDrop property must be set to true in order to raise these events. If you set the AllowDrop property to false, even if these events are implemented, they will not be raised. Table 7-1 details these properties of the System.Windows.UIElement class.

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DragEnter</td>
<td>Fires when the input system reports an underlying drag event with the specific event as the target. The event occurs only if the AllowDrop property of that element is set to true. Event args class: DragEventHandler. New event in Silverlight 4.</td>
</tr>
<tr>
<td>DragLeave</td>
<td>Fires when the input system reports an underlying drag event with the specific event as the origin. The event occurs only if the AllowDrop property of that element is set to true. Event args class: DragEventHandler. New event in Silverlight 4.</td>
</tr>
</tbody>
</table>