

Работа с мышью и клавиатурой

**Отвагин Алексей Владимирович,
доцент каф. ЭВМ, к.т.н., а. 505-5**

Содержание

- Работа с мышью
- Работа с клавиатурой

Мышь

- Манипулятор с одной, двумя или тремя кнопками
- Перемещение отображается на дисплее в виде небольшого изображения – курсора
- Курсор имеет специальный пиксел, координаты которого принимаются за координаты мыши

Получение характеристик мыши

- Опрос свойств объекта класса `SystemInformation`

Свойство	Описание
<code>DoubleClickSize</code>	Размер области, в которой двойное нажатие кнопки мыши рассматривается как двойной щелчок.
<code>DoubleClickTime</code>	Период между двумя соседними щелчками, рассматриваемый как двойной щелчок.
<code>MouseButtons</code>	Количество кнопок мыши.
<code>MouseHoverTime</code>	Время зависания курсора над элементом управления для генерации соответствующего события.
<code>MousePresent</code>	Наличие мыши.
<code>MouseSpeed</code>	Скорость перемещения мыши.
<code>MouseWheelPresent</code>	Наличие колесика у мыши.

Сообщения мыши

- Оконная процедура получает сообщения при проходе мыши над окном или нажатии клавиш мыши внутри области окна
- Сообщения проходят ко всем окнам, даже если окно не имеет фокуса
- Фокус – это свойство активного в данный момент окна

Основные сообщения

Событие	Описание
Click	Событие перед отпусканием кнопки мыши, приходит перед событием MouseUp.
DoubleClick	Двойной щелчок кнопкой мыши по элементу управления.
MouseDown	Событие нажатия кнопки мыши, когда курсор находится над элементом управления.
MouseMove	Событие движения указателя мыши над элементом управления.
MouseUp	Событие отпускания кнопки мыши над элементом управления.
MouseWheel	Событие поворота колесика мыши для элемента управления, имеющего фокус.

Дополнительные сообщения

Событие	Описание
MouseClicked	Щелчок кнопкой мыши по элементу управления.
DoubleClick	Двойной щелчок кнопкой мыши по элементу управления.
MouseEnter	Событие вхождения курсора в клиентскую область элемента управления.
MouseHover	Событие остановки указателя мыши над элементом управления.
MouseLeave	Событие выхода курсора за пределы клиентской области элемента управления.

Параметры сообщений мыши

MouseEventArgs

Имя	Описание
<u>Button</u>	Определяет, какая кнопка была нажата.
<u>Clicks</u>	Количество нажатий кнопки.
<u>Delta</u>	Количество делений, на которое повернулось колесико мыши.
<u>X</u>	Координата мыши по X.
<u>Y</u>	Координата мыши по Y.

Последовательность сообщений от мыши

- Одинарный щелчок

MouseDown->Click->MouseClicked->MouseUp

- Двойной щелчок

MouseDown->Click->MouseClicked->

MouseUp-> MouseDown->

DoubleClick-> MouseDoubleClick->

MouseUp event.

Пример обработчика сообщений МЫШИ

```
private void Control1_MouseClick(Object sender, MouseEventArgs e) {  
    System.Text.StringBuilder messageBoxCS = new  
        System.Text.StringBuilder();  
    messageBoxCS.AppendFormat("{0} = {1}", "Button", e.Button );  
    messageBoxCS.AppendLine();  
    messageBoxCS.AppendFormat("{0} = {1}", "Clicks", e.Clicks );  
    messageBoxCS.AppendLine();  
    messageBoxCS.AppendFormat("{0} = {1}", "X", e.X );  
    messageBoxCS.AppendLine();  
    messageBoxCS.AppendFormat("{0} = {1}", "Y", e.Y );  
    messageBoxCS.AppendLine();  
    messageBoxCS.AppendFormat("{0} = {1}", "Delta", e.Delta );  
    messageBoxCS.AppendLine();  
    messageBoxCS.AppendFormat("{0} = {1}", "Location", e.Location );  
    messageBoxCS.AppendLine();  
    MessageBox.Show(messageBoxCS.ToString(), "MouseClicked Event" );  
}
```

Drag-n-Drop

Событие	Описание
DragEnter	Перетаскиваемый объект входит в границы элемента управления.
DragOver	Перетаскиваемый объект движется в границах элемента управления.
DragDrop	Завершение операции перетаскивания.
DragLeave	Перетаскиваемый объект выходит за границы элемента управления.

Параметры сообщений перетаскивания DragEventArgs

Имя	Описание
Data	Ассоциированные с событием данные.
Effect	Эффект операции перетаскивания
KeyState	Состояние клавиш SHIFT, CTRL, ALT и кнопок мыши.
<u>X</u>	Координата мыши по X.
<u>Y</u>	Координата мыши по Y.

Начало и окончание перетаскивания

```
private void button1_MouseDown(object sender,
    System.Windows.Forms.MouseEventArgs e) {
    button1.DoDragDrop(button1.Text, DragDropEffects.Copy |
        DragDropEffects.Move);
}
```

```
private void textBox1_DragEnter(object sender,
    System.Windows.Forms.DragEventArgs e)
{
    if (e.Data.GetDataPresent(DataFormats.Text))
        e.Effect = DragDropEffects.Copy;
    else e.Effect = DragDropEffects.None;
}
}
```

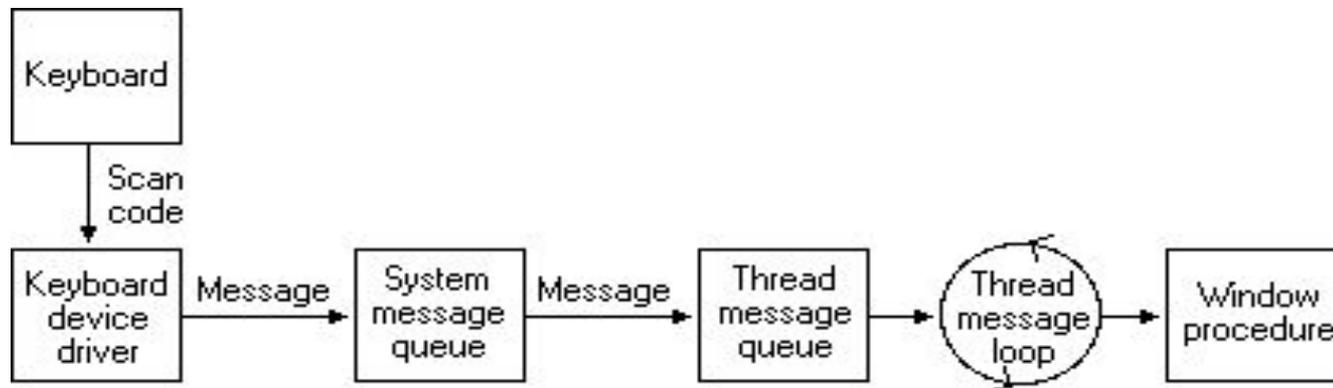
```
private void textBox1_DragDrop(object sender,
    System.Windows.Forms.DragEventArgs e) {
    textBox1.Text = e.Data.GetData(DataFormats.Text).ToString();
}
```

Клавиатура

- Система обслуживает клавиатуру через соответствующий драйвер устройства
- Драйвер возвращает скэн-коды нажатой клавиши
- Поддержка конкретного языка ввода обеспечивается трансляцией скэн-кодов в код виртуальной клавиши

Сообщения клавиатуры в очереди

- Сообщение со скэн-кодом и кодом виртуальной клавиши помещается в системную очередь сообщений
- Система выбирает сообщения и передает их в очередь активного потока, окно которого содержит фокус
- Сообщения обрабатываются процедурой окна



Фокус клавиатуры

- Может передаваться между окнами приложения (в случае многодокументного интерфейса)
- `GetFocus()` – сообщает потоку, какое из его окон владеет фокусом
- `SetFocus()` – устанавливает фокус в определенное окно
- Передача фокуса клавиатуры от одного окна другому:
 - `WM_KILLFOCUS` посылается окну, теряющему фокус
 - `WM_SETFOCUS` – окну, получившему его

Буферизация сообщений клавиатуры

- Сообщения о нажатии клавиш хранятся в системной очереди
- При нажатии некоторой клавиши несколько раз сообщения накапливаются и оформляются в одно сообщение с указанием количества повторений
- Некоторые комбинации клавиш вызывают переключение фокуса, при этом поток сообщений перенаправляется в другое окно

Сообщения о нажатии клавиш

- ❑ Сочетания клавиш могут производить «системные» сообщения, связанные с акселераторами
- ❑ Сообщения приходят парами, кроме случаев повторения символа

	<i>Нажатие</i>	<i>Отпускание</i>
Несистемная комбинация	WM_KEYDOWN	WM_KEYUP
Системная комбинация	WM_SYSKEYDOWN	WM_SYSKEYUP

Предобработка KeyDown

Действие	Метод	Пояснение
Проверка акселератора или быстрой клавиши.	ProcessCmdKey	Возвращает true, если нажатая комбинация является акселератором. В этом случае срабатывает Соответствующий обработчик, а генерация событий клавиатуры прекращается. Если нажатая комбинация не соответствует акселераторам, вызывается метод IsInputKey
Проверка, является ли клавиша специальной	IsInputKey	Если метод возвращает true(обычная клавиша), то генерируется событие KeyDown, иначе вызывается функция ProcessDialogKey.
Проверка, служит ли клавиша для Навигации	ProcessDialogKey	Определяет действия программы с навигационными клавишами. Он позволяет передавать сообщения по иерархии окон. Если метод возвращает true(обработка завершена), то событие KeyDown не генерируется, иначе создается KeyDown.

Предобработка KeyPress

Действие	Метод	Пояснение
Соответствует ли событие нажатию символьной клавиши	<code>IsInputChar</code>	True означает, что символ относится к простым символам, false – что символ требует дальнейшей обработки методом <code>ProcessDialogChar</code>
Обработка специальных символов	<code>ProcessDialogChar</code>	Используется для передачи управления по иерархии аналогично <code>ProcessDialogKey</code>

Обработчик KeyDown

```
// флаг определяет ввод нечислового символа.  
private bool nonNumberEntered = false;  
private void textBox1_KeyDown(object sender,  
    System.Windows.Forms.KeyEventArgs e) {  
    nonNumberEntered = false;  
    // Это цифра с клавиатуры?  
    if (e.KeyCode < Keys.D0 || e.KeyCode > Keys.D9) {  
        // Это цифра с боковой клавиатуры?  
        if (e.KeyCode < Keys.NumPad0 || e.KeyCode > Keys.NumPad9) {  
            //Это Backspace?  
            if(e.KeyCode != Keys.Back) {  
                // Это не цифра  
                nonNumberEntered = true;  
            }  
        }  
    }  
}
```

Обработчик KeyPress

```
private void textBox1_KeyPress(object sender,  
    System.Windows.Forms.KeyPressEventArgs e) {  
    // Проверка флага из обработчика KeyDown  
    if (nonNumberEntered == true) {  
        // отбросить символ.  
        e.Handled = true;  
    }  
}
```

Параметры сообщений клавиатуры KeyEventArgs

Имя	Описание
Alt	Показывает, нажат ли ALT
Control	Показывает, нажат ли CTRL
Handled	Признак того, что сообщение уже
KeyCode	Код нажатой клавиши
Modifiers	Комбинация ALT,CTRL,SHIFT
Shift	Показывает, нажат ли SHIFT