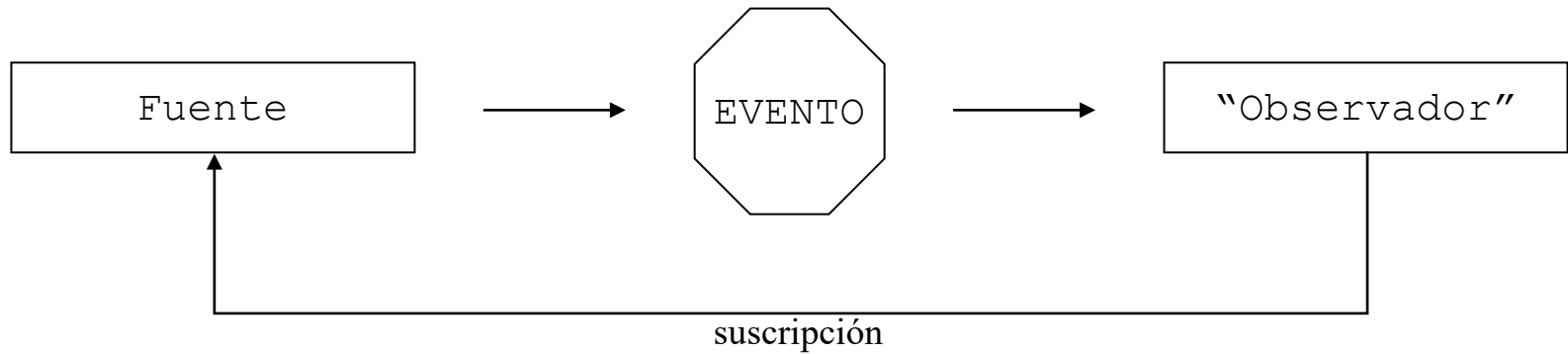
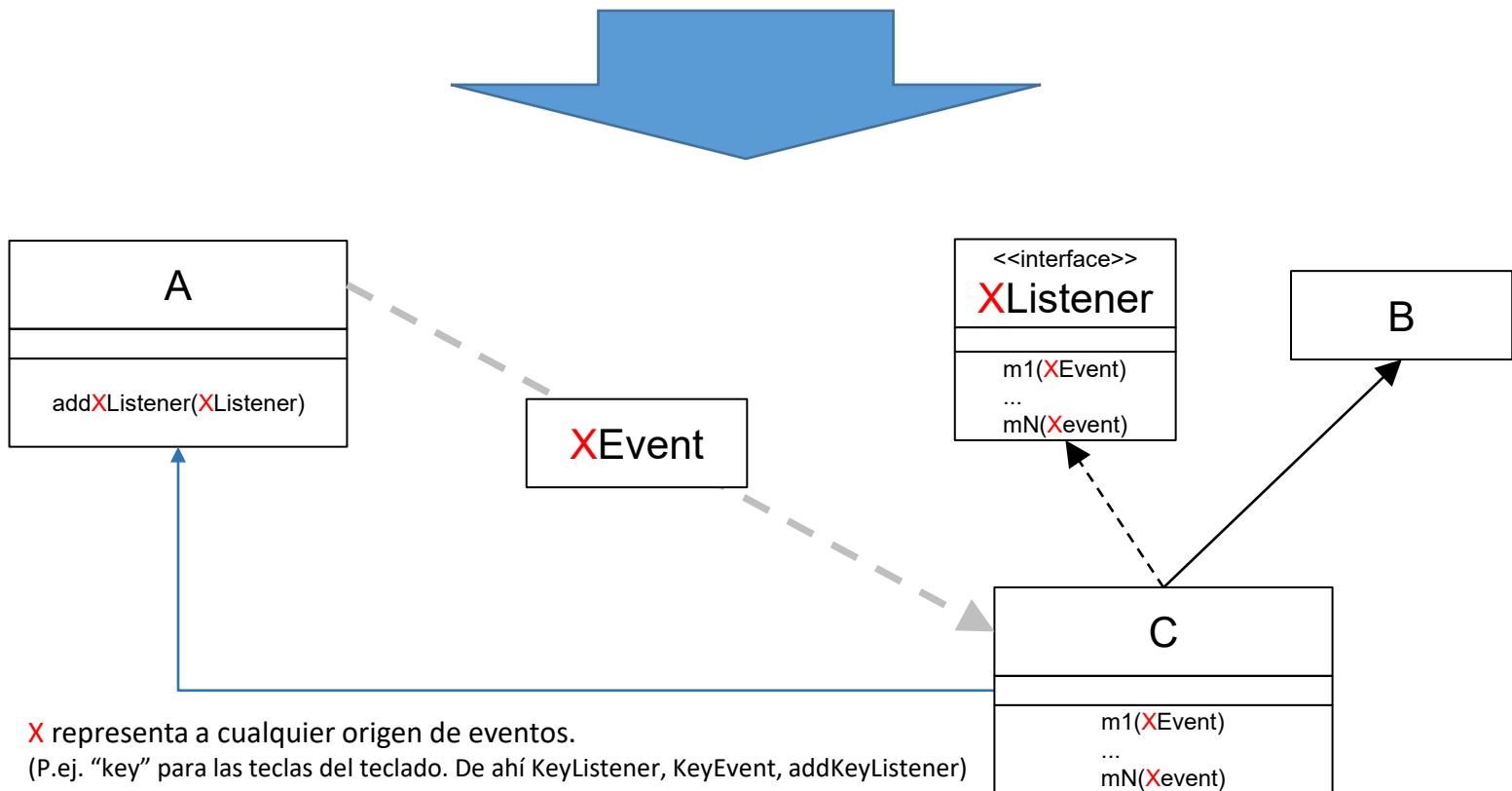
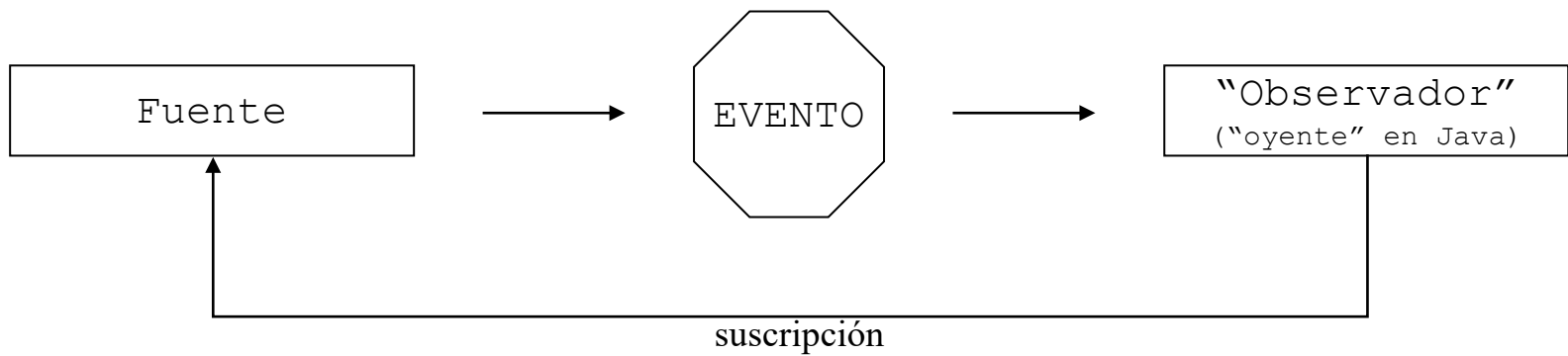


Interacción con GUIs

Un uso concreto del patrón “Delegation Event Model”
Más conocido como “[Observer](#)” o “Publish/subscribe”





Module java.desktop

Package java.awt.event

Aunque nos limitemos a Java7, podemos “familiarizarnos” con versiones posteriores a Java9 aquí se añade la información de que el package java.awt.event se sitúa en el módulo java.desktop

Interface **KeyListener**

All Superinterfaces:
EventListener

All Known Implementing Classes:
AWTEventMulticaster, BasicComboBoxUI.KeyHandler, BasicComboPopup.InvocationKeyHandler, BasicTableUI.KeyHandler, BasicTreeUI.KeyHandler, KeyAdapter

```
public interface KeyListener  
extends EventListener
```

The listener interface for receiving keyboard events (keystrokes). The class that is interested in processing a keyboard event either implements this interface (and all the methods it contains) or extends the abstract KeyAdapter class (overriding only the methods of interest).

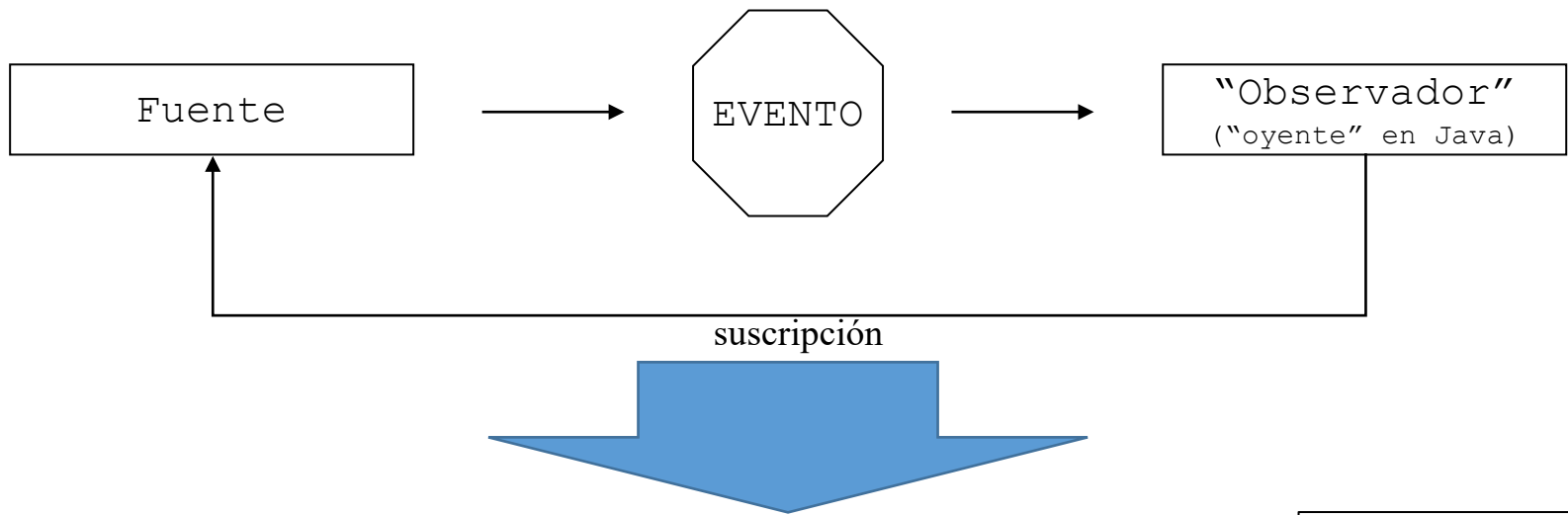
The listener object created from that class is then registered with a component using the component's addKeyListener method. A keyboard event is generated when a key is pressed, released, or typed. The relevant method in the listener object is then invoked, and the KeyEvent is passed to it.

Since:
1.1

See Also:
KeyAdapter, KeyEvent, Tutorial: Writing a Key Listener

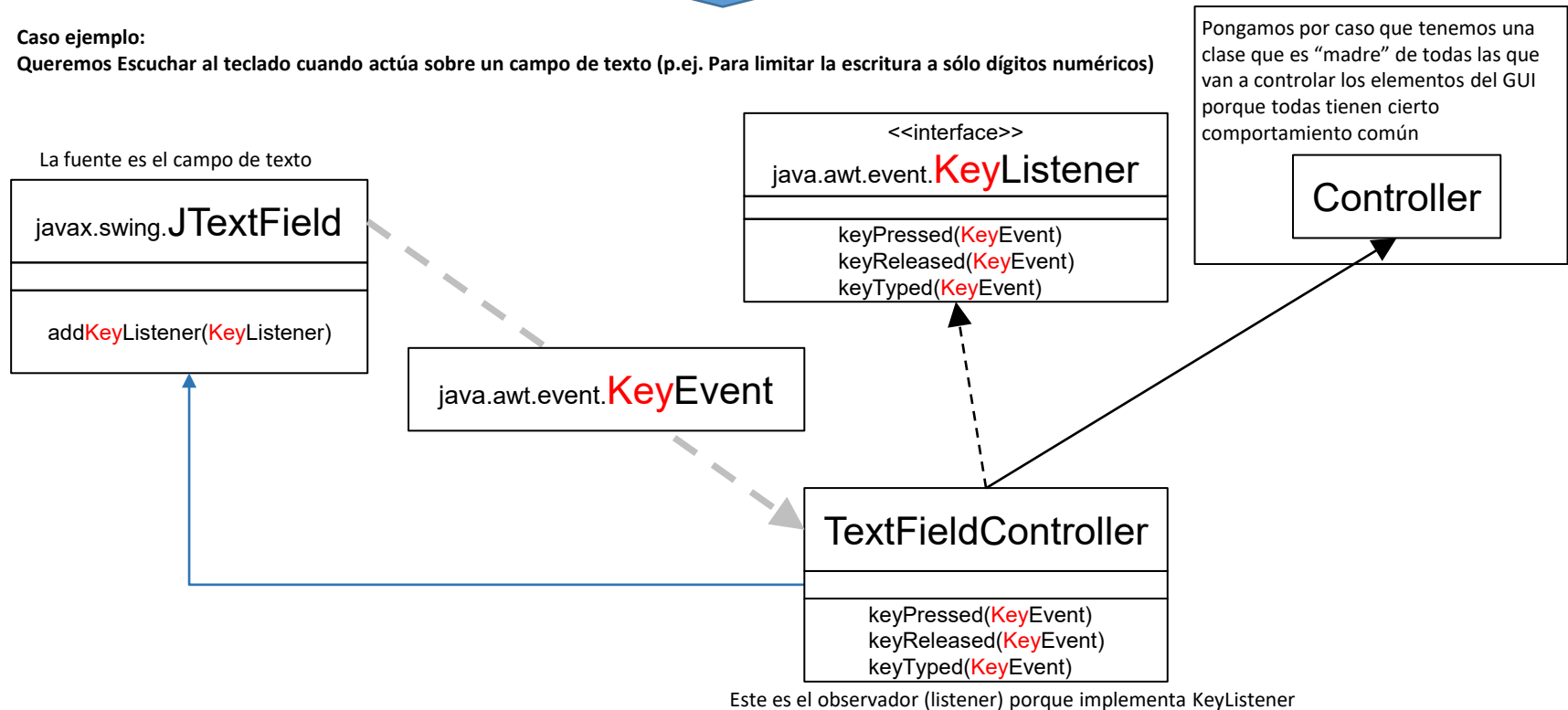
Method Summary

All Methods	Instance Methods	Abstract Methods
Modifier and Type	Method	Description
void	keyPressed(KeyEvent e)	Invoked when a key has been pressed.
void	keyReleased(KeyEvent e)	Invoked when a key has been released.
void	keyTyped(KeyEvent e)	Invoked when a key has been typed.



Caso ejemplo:

Queremos Escuchar al teclado cuando actúa sobre un campo de texto (p.ej. Para limitar la escritura a sólo dígitos numéricos)



Un ejemplo concreto sencillo

```
1 package edu.upveh.gbg.tap.ejemploobserver;
2
3 public class EjemploObserver {
4
5     public static void main(String[] args) {
6
7         //por simplificar arrancamos fuera del hilo del GUI
8         new EjemploObserverGUI();
9     }
10
11 }
```

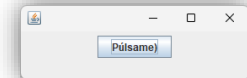
MVC

```
1 package edu.upveh.gbg.tap.ejemploobserver;
2
3 import java.awt.event.MouseEvent;
4 import java.awt.event.MouseListener;
5
6 public class EjemploMouseListener implements MouseListener {
7
8     @Override public void mouseClicked(MouseEvent e) {System.out.println(x: "Clicked");}
9     @Override public void mousePressed(MouseEvent e) {System.out.println(x: "Pressed");}
10    @Override public void mouseReleased(MouseEvent e) {System.out.println(x: "Released");}
11    @Override public void mouseEntered(MouseEvent e) {System.out.println(x: "Entered");}
12    @Override public void mouseExited(MouseEvent e) {System.out.println(x: "Exited");}
13 }
```

MVC

```
1 package edu.upveh.gbg.tap.ejemploobserver;
2
3 import java.awt.FlowLayout;
4 import javax.swing.JButton;
5 import javax.swing.JFrame;
6
7 public class EjemploObserverGUI extends JFrame{
8
9     EjemploObserverGUI() {
10         setLayout(new FlowLayout());
11         JButton boton = new JButton(string: "Púlsame");
12         add(comp: boton);
13         boton.addMouseListener(new EjemploMouseListener());
14         pack();
15         setVisible(b: true);
16     }
17
18 }
```

MVC



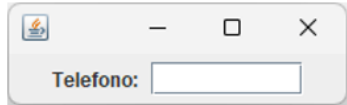
Un ejemplo concreto un poco más real

```
1 package edu.upveh.gbg.tap.ejemploobserver;
2
3 public class EjemploObserver {
4
5     public static void main(String[] args) {
6
7         //por simplificar arrancamos fuera del hilo del GUI
7         new EjemploObserverGUI();
8     }
9
10 }
11
```

MVC

```
1 package edu.upveh.gbg.tap.ejemploobserver;
2
3 import java.awt.FlowLayout;
4 import javax.swing.JFrame;
5 import javax.swing.JLabel;
6 import javax.swing.JTextField;
7
8 public class EjemploObserverGUI extends JFrame{
9
10     EjemploObserverGUI(){
11         setLayout(new FlowLayout());
12         add(new JLabel(string: "Telefono: "));
13         JTextField telefono = new JTextField();
14         telefono.setColumns(columns:8);
15         add(comp: telefono);
16         telefono.addKeyListener(new EjemploKeyListener());
17         pack();
18         setVisible(b: true);
19     }
20
21 }
```

MVC



```
1 package edu.upveh.gbg.tap.ejemploobserver;
2
3 import java.awt.event.KeyEvent;
4 import java.awt.event.KeyListener;
5 import javax.swing.JTextField;
6
7 public class EjemploKeyListener implements KeyListener {
8
9     @Override
10     public void keyTyped(KeyEvent e) {
11         if (((JTextField)e.getSource()).getText().length()>=9 || !Character.isDigit(ch: e.getKeyChar())) e.consume();
12     }
13
14     @Override public void keyPressed(KeyEvent e) {}
15     @Override public void keyReleased(KeyEvent e) {}
16 }
```

MVC

Alterémoslo para ver clases anónimas evitando definir EjemploKeyListener
(pero dejando claro que no es lo mejor, al mezclar view y controller)

Cómo lo hace Netbeans

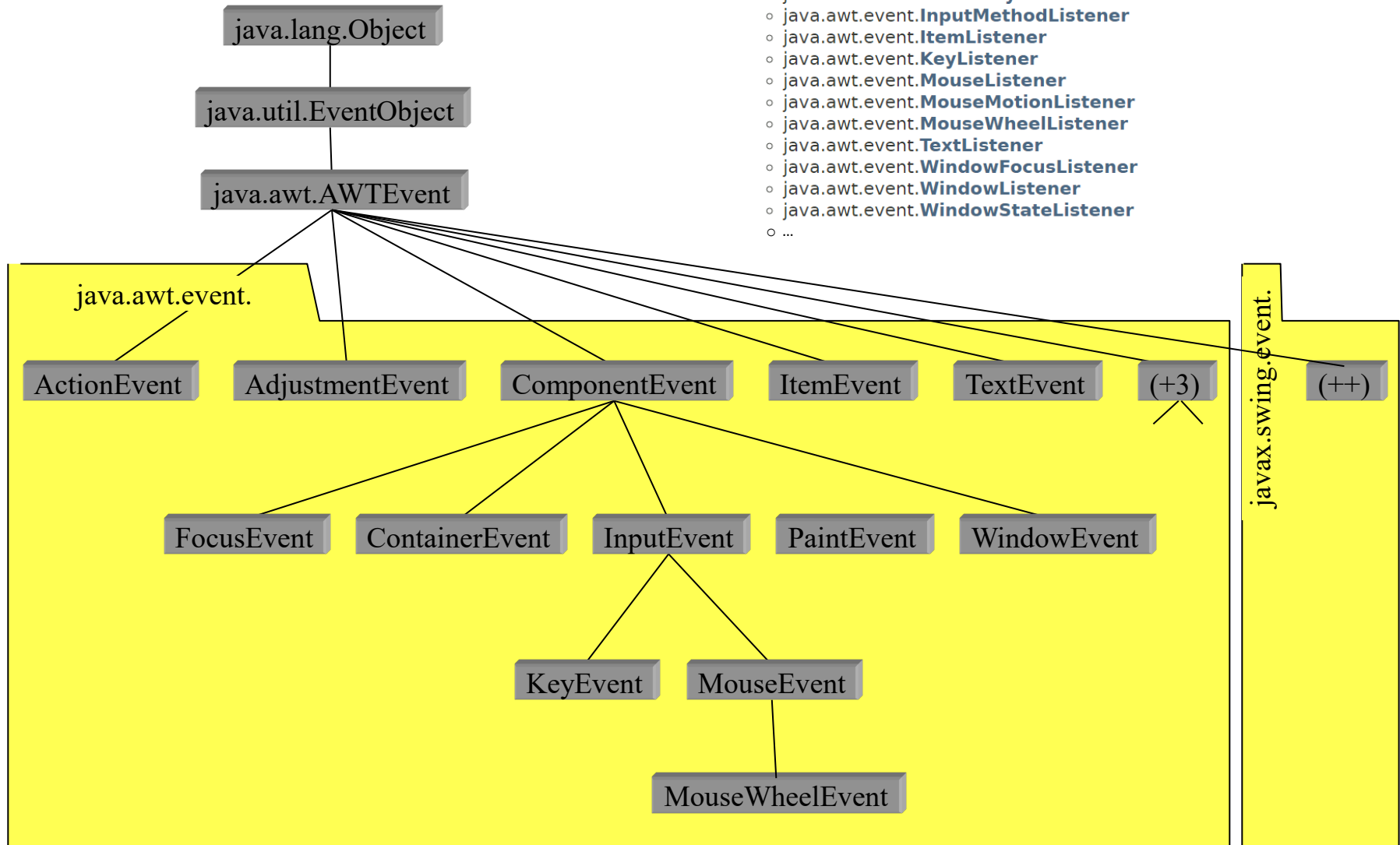
(código con eliminación de comentarios y reformateado ligeramente para ocupar menos líneas)

- * Genera Listener abstracto
- * Lo hace extendiendo un Adapter
- * Delega el método en otro para mantener como no editable la zona de inicialización del JFrame

```
1 package edu.upveh.gbg.tap.ejemploobserver;
2
3 import javax.swing.JTextField;
4
5 public class NewJFrame extends javax.swing.JFrame {
6
7     /**
8      * Creates new form NewJFrame
9      */
10    public NewJFrame() { initComponents(); }
11
12
13    @SuppressWarnings("unchecked")
14    // <editor-fold defaultstate="collapsed" desc="Generated Code">
15    private void initComponents() {
16
17        jLabel1 = new javax.swing.JLabel();
18        jTextField1 = new javax.swing.JTextField();
19
20        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
21        getContentPane().setLayout(new java.awt.FlowLayout());
22
23        jLabel1.setText(text: "Teléfono: ");
24        getContentPane().add(comp: jLabel1);
25
26        jTextField1.setColumns(columns: 8);
27        jTextField1.addKeyListener(new java.awt.event.KeyAdapter() {
28            public void keyTyped(java.awt.event.KeyEvent evt) {
29                jTextField1KeyTyped(evt);
30            }
31        });
32        getContentPane().add(comp: jTextField1);
33
34        pack();
35    } // </editor-fold>
36
37    private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {
38        if (((JTextField) evt.getSource()).getText().length() >= 9 || !Character.isDigit(ch: evt.getKeyChar())) evt.consume();
39    }
40
41    public static void main(String args[]) {
42        /* Set the Nimbus look and feel */
43        Look and feel setting code (optional)
44
45        /* Create and display the form */
46        java.awt.EventQueue.invokeLater(new Runnable() {
47            @Override public void run() { new NewJFrame().setVisible(true); }
48        });
49    }
50
51    // Variables declaration - do not modify
52    private javax.swing.JLabel jLabel1;
53    private javax.swing.JTextField jTextField1;
54    // End of variables declaration
55 }
```

“Listeners” y eventos disponibles en un GUI con awt/swing

- java.util.**EventListener**
 - java.awt.event.**ActionListener**
 - java.awt.event.**AdjustmentListener**
 - java.awt.event.**AWTEventListener**
 - java.awt.event.**ComponentListener**
 - java.awt.event.**ContainerListener**
 - java.awt.event.**FocusListener**
 - java.awt.event.**HierarchyBoundsListener**
 - java.awt.event.**HierarchyListener**
 - java.awt.event.**InputMethodListener**
 - java.awt.event.**ItemListener**
 - java.awt.event.**KeyListener**
 - java.awt.event.**MouseListener**
 - java.awt.event.**MouseMotionListener**
 - java.awt.event.**MouseWheelListener**
 - java.awt.event.**TextListener**
 - java.awt.event.**WindowFocusListener**
 - java.awt.event.**WindowListener**
 - java.awt.event.**WindowStateListener**
 - ...



Package java.util

Interface `EventListener`

All Known Subinterfaces:

Action, ActionListener, AdjustmentListener, AncestorListener, AppForegroundListener, AppHiddenListener, AppReopenedListener, AWTEventListener, BeanContextMembershipListener, BeanContextServiceRevokedListener, BeanContextServices, BeanContextServicesListener, CaretListener, CellEditorListener, ChangeListener, ComponentListener, ConnectionEventListener, ContainerListener, ControllerEventListener, DocumentListener, DragGestureListener, DragSourceListener, DragSourceMotionListener, DropTargetListener, FlavorListener, FocusListener, GUIInitializedListener, HandshakeCompletedListener, HierarchyBoundsListener, HierarchyListener, HyperlinkListener, IIOReadProgressListener, IIOReadUpdateListener, IIOReadWarningListener, IIOWriteProgressListener, IIOWriteWarningListener, InputMethodListener, InternalFrameListener, ItemListener, KeyListener, LineListener, ListDataListener, ListSelectionListener, MenuDragMouseListener, MenuItemListener, MouseEventListener, MouseInputListener, MouseListener, MouseMotionListener, MouseWheelListener, NamespaceChangeListener, NamingListener, NodeChangeListener, NotificationListener, ObjectChangeListener, PopupMenuListener, PreferenceChangeListener, PropertyChangeListener, RowSetListener, RowSorterListener, ScreenSleepListener, SSLSessionBindingListener, StatementEventListener, SystemEventListener, SystemSleepListener, TableColumnModelListener, TableModelListener, TextListener, TopLevelWindowListener, TreeExpansionListener, TreeModelListener, TreeSelectionListener, TreeWillExpandListener, UndoableEditListener, UnsolicitedNotificationListener, UserSessionListener, VetoableChangeListener, WindowFocusListener, WindowListener, WindowStateListener

All Known Implementing Classes:

AbstractAction, AbstractButton, ButtonChangeListener, AWTEventListenerProxy, AWTEventMulticaster, BasicActionButtonHandler, BasicActionBarUI, BasicActionBarUI.Resource, BasicActionBarUI.UIResource, BasicActionBarUI.FocusHandler, BasicActionBarUI.ItemHandler, BasicActionBarUI.KeyHandler, BasicActionBarUI.ListDataHandler, BasicActionBarUI.ListDataHandler, BasicActionBarUI.PropertyChangeHandler, BasicActionBarUI.InvocationKeyHandler, BasicActionBarUI.InvocationMouseHandler, BasicActionBarUI.InvocationMouseMoveHandler, BasicActionBarUI.ItemHandler, BasicActionBarUI.ListDataHandler, BasicActionBarUI.ListModelHandler, BasicActionBarUI.ListSelectionHandler, BasicActionBarUI.ListSelectionHandler, BasicActionBarUI.NavigateAction, BasicActionBarUI.OpenAction, BasicDirectoryModel, BasicFileChooserUI.ApproveSelectionAction, BasicFileChooserUI.CancelSelectionAction, BasicFileChooserUI.ChangeToParentDirectoryAction, BasicFileChooserUI.DoubleClickListener, BasicFileChooserUI.GoHomeAction, BasicFileChooserUI.NewFolderAction, BasicFileChooserUI.SelectionListener, BasicFileChooserUI.UpdateAction, BasicInternalFrameTitlePane.CloseAction, BasicInternalFrameTitlePane.IconifyAction, BasicInternalFrameTitlePane.MaximizeAction, BasicInternalFrameTitlePane.MoveAction, BasicInternalFrameTitlePane.PropertyChangeHandler, BasicInternalFrameTitlePane.RestoreAction, BasicInternalFrameTitlePane.SizeAction, BasicInternalFrameUI.BasicInternalFrameListener, BasicInternalFrameUI.BorderListener, BasicInternalFrameUI.ComponentHandler, BasicInternalFrameUI.GlassPaneDispatcher, BasicInternalFrameUI.InternalFramePropertyChangeListener, BasicLabelUI, BasicListUI.FocusHandler, BasicListUI.ListDataHandler, BasicListUI.ListSelectionHandler, BasicListUI.MouseInputHandler, BasicListUI.PropertyChangeHandler, BasicMenuItemUI.MouseInputHandler, BasicMenuItemUI.ChangeHandler, BasicMenuItemUI.MouseInputHandler, BasicOptionPaneUI.ButtonActionListener, BasicOptionPaneUI.PropertyChangeHandler, BasicProgressBarUI.ChangeHandler, BasicRootPaneUI, BasicScrollBarUI.ArrowButtonListener, **BasicScrollBarUI.ModelListener**, BasicScrollBarUI.PropertyChangeHandler, BasicScrollBarUI.ScrollListener, BasicScrollBarUI.TrackListener, BasicScrollPaneUI.HSBChangeListener, BasicScrollPaneUI.MouseWheelHandler, BasicScrollPaneUI.PropertyChangeHandler, BasicScrollPaneUI.ViewportChangeHandler, BasicScrollPaneUI.VSBChangeListener, BasicSliderUI.ActionScroller, BasicSliderUI.ChangeHandler, BasicSliderUI.ComponentHandler, BasicSliderUI.FocusHandler, BasicSliderUI.PropertyChangeHandler, BasicSliderUI.ScrollListener, BasicSliderUI.TrackListener, BasicSplitPaneDivider, BasicSplitPaneDivider.MouseHandler, BasicSplitPaneUI.FocusHandler, BasicSplitPaneUI.KeyboardDownRightHandler, BasicSplitPaneUI.KeyboardEndHandler, BasicSplitPaneUI.KeyboardHomeHandler, BasicSplitPaneUI.KeyboardResizeToggleHandler, BasicSplitPaneUI.KeyboardUpLeftHandler, BasicSplitPaneUI.PropertyHandler, BasicTabbedPaneUI.FocusHandler, BasicTabbedPaneUI.MouseHandler, BasicTabbedPaneUI.PropertyChangeHandler, BasicTabbedPaneUI.TabSelectionHandler, BasicTableHeaderUI.FrameInputHandler, BasicTableUI.FocusHandler, BasicTableUI.KeyHandler, BasicTableUI.MouseInputHandler, BasicTextUI.BasicCaret, BasicToolBarUI.DockingListener, BasicToolBarUI.FrameListener, BasicToolBarUI.PropertyListener, BasicToolBarUI.ToolBarContListener, BasicToolBarUI.ToolBarFocusListener, BasicTreeUI.CellEditorHandler, BasicTreeUI.ComponentHandler, BasicTreeUI.FocusHandler, BasicTreeUI.KeyHandler, BasicTreeUI.MouseHandler, BasicTreeUI.MouseInputHandler, BasicTreeUI.PropertyChangeHandler, BasicTreeUI.SelectionModelPropertyChangeHandler, BasicTreeUI.TreeCancelEditingAction, BasicTreeUI.TreeExpansionHandler, BasicTreeUI.TreeHomeAction, BasicTreeUI.TreeIncrementAction, BasicTreeUI.TreeModelHandler, BasicTreeUI.TreePageAction, BasicTreeUI.TreeSelectionHandler, BasicTreeUI.TreeToggleAction, BasicTreeUI.TreeTraversalAction, BeanContextChildSupport, BeanContextServicesSupport, BeanContextServicesSupport.BCSProxyServiceProvider, BeanContextSupport, Checkbox.AccessibleAWTCheckbox, Component.AccessibleAWTComponent, AccessibleAWTComponentHandler, Component.AccessibleAWTFocusHandler, ComponentAdapter, Container.AccessibleAWTContainer, AccessibleContainerHandler, ContainerAdapter, DefaultCaret, DefaultCellEditor, EditorDelegate, DefaultEditorKit.BeepAction, DefaultEditorKit.CopyAction, DefaultEditorKit.CutAction, DefaultEditorKit.DefaultKeyTypedAction, DefaultEditorKit.InsertBreakAction, DefaultEditorKit.InsertContentAction, DefaultEditorKit.InsertTabAction, DefaultEditorKit.PasteAction, DefaultTableModel, DefaultTableCellEditor, DragSourceAdapter, DragSourceContext, DropTarget, DropTarget.DropTargetAutoScroller, DropTargetAdapter, EventListenerProxy, EventQueueMonitor, FocusAdapter, FormView, FormView.MouseEventListener, HierarchyBoundsAdapter, HTMLEditorKit.HTMLTextAction, HTMLEditorKit.InsertHTMLTextAction, HTMLEditorKit.LinkController, InternalFrameAdapter, JCheckBox.AccessibleJCheckBox, JCheckBoxMenuItems.AccessibleJCheckBoxMenuItem, JComboBox, JComponent.AccessibleJComponent, AccessibleContainerHandler, JComponent.AccessibleJComponent, AccessibleFocusHandler, JEditorPane.AccessibleJEditorPane, JEditorPane.AccessibleJEditorPaneHTML, JEditorPane.JEditorPaneAccessibleHypertextSupport, JLayer, JList.AccessibleJList, JMenu.AccessibleJMenu, JMenu.WinListener, JMenuItem.AccessibleJMenuItem, JPasswordField.AccessibleJPasswordField, JPopupMenu.AccessibleJPopupMenu, JRadioButton.AccessibleJRadioButton, JRadioButtonMenuItem.AccessibleJRadioButtonMenuItem, JScrollPane.AccessibleJScrollPane, JSlider.AccessibleJSlider, JSpinner.AccessibleJSpinner, JSpinner.DateEditor, JSpinner.DefaultEditor, JSpinner.Listener, JSpinner.NumberEditor, JTabbedPane.AccessibleJTabbedPane, JTabbedPane.ModelListener, JTable.AccessibleJTable, JTableHeader, JTextArea.AccessibleJTextArea, JTextComponent.AccessibleJTextComponent, JTextField.AccessibleJTextField, JToggleButton.AccessibleJToggleButton, JTree.AccessibleJTree, JTree.TreeModelHandler, JTree.TreeSelectionRedirector, JViewport.ViewListener, KeyAdapter, List.AccessibleAWTList, MetalComboBoxEditor, MetalComboBoxEditor.UIResource, MetalPropertyChangeListener, MetaFileChooserUI.DirectoryComboBoxAction, MetaFileChooserUI.FilterComboBoxModel, MetaFileChooserUI.SingleClickListener, MetaLabelUI, MetalRootPaneUI, MetalSliderUI.MetalPropertyListener, MetalToolBarUI.MetalContainerListener, MetalToolBarUI.MetalDockingListener, MetalToolBarUI.MetalRollOverListener, MouseAdapter, MouseDragGestureRecognizer, MouseInputAdapter, MotionEventAdapter, ProgressMonitor.AccessibleProgressMonitor, PropertyChangeListenerProxy, RelationService, StyledEditorKit.AlignmentAction, StyledEditorKit.BoldAction, StyledEditorKit.FontFamilyAction, StyledEditorKit.FontSizeAction, StyledEditorKit.ForegroundAction, StyledEditorKit.ItalicAction, StyledEditorKit.StyledTextAction, StyledEditorKit.UnderlineAction, SynthButtonUI, SynthCheckBoxMenuItemUI, SynthCheckBoxBoxUI, SynthColorChooserUI, SynthComboBoxUI, SynthDesktopIconUI, SynthDesktopPaneUI, SynthInternalFrameUI, SynthLabelUI, SynthListUI, SynthMenuBarUI, SynthMenuItemUI, SynthMenuUI, SynthOptionPaneUI, SynthPanelUI, SynthPopupMenuUI, SynthProgressBarUI, SynthRadioButtonMenuItemUI, SynthRadioButtonUI, SynthScrollPaneUI, SynthScrollPaneUI, SynthSeparatorUI, SynthSliderUI, SynthSpinnerUI, SynthSplitPaneUI, SynthTabbedPaneUI, SynthTableHeaderUI, SynthTableUI, SynthToggleButtonUI, SynthToolBarUI, SynthToolTipUI, SynthTreeUI, SynthViewportUI, TextAction, TextArea.AccessibleAWTTextArea, TextComponent.AccessibleAWTTextComponent, TextField.AccessibleAWTTextField, ToolTipManager, ToolTipManager.insideTimerAction, ToolTipManager.outsideTimerAction, ToolTipManager.stillInsideTimerAction, UndoManager, VetoableChangeListenerProxy, WindowAdapter

```
public interface EventListener
```

A tagging interface that all event listener interfaces must extend.

El patrón Observer es muy utilizado

Este es un ejemplo de Interfaz “etiquetador”

Métodos de los principales Listeners

ActionListener	actionPerformed (ActionEvent)
ItemListener	itemStateChanged (ItemEvent)
MouseListener	mousePressed (MouseEvent) mouseReleased (MouseEvent) mouseEntered (MouseEvent) mouseExited (MouseEvent) mouseClicked (MouseEvent)
MouseMotionListener	mouseDragged (MouseEvent) mouseMoved (MouseEvent)
KeyListener	keyPressed (KeyEvent) keyReleased (KeyEvent) keyTyped (KeyEvent)
FocusListener	focusGained (FocusEvent) focusLost (FocusEvent)
AdjustmentListener	adjustmentValueChanged (AdjustmentEvent)
ComponentListener	componentMoved (ComponentEvent) componentHidden (ComponentEvent) componentResized (ComponentEvent) componentShown (ComponentEvent)
WindowListener	windowClosing (WindowEvent) windowOpened (WindowEvent) windowIconified (WindowEvent) windowDeiconified (WindowEvent) windowClosed (WindowEvent) windowActivated (WindowEvent) windowDeactivated (WindowEvent)
ContainerListener	componentAdded (ContainerEvent) componentRemoved (ContainerEvent)
TextListener	textValueChanged (TextEvent)