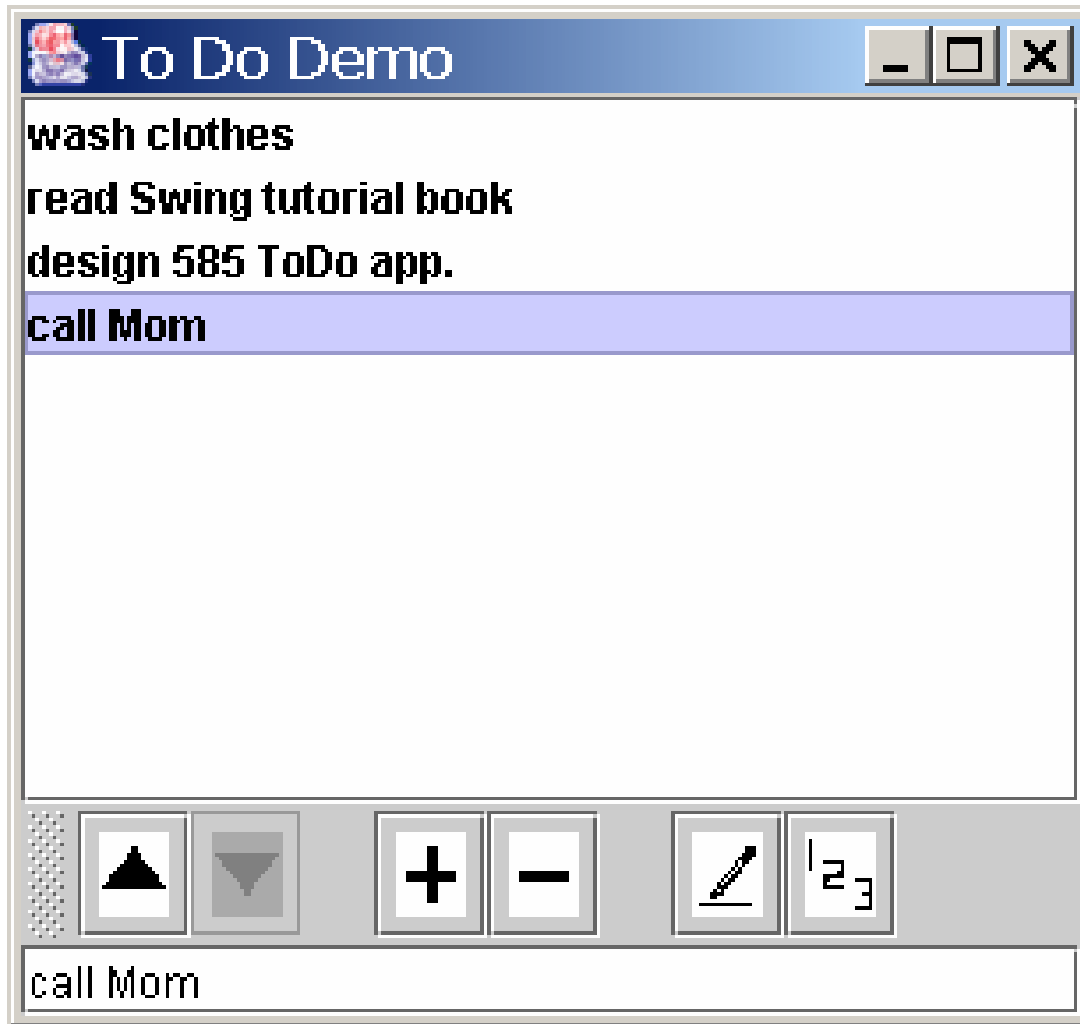


ObjectListDemo



Maintains a list of objects.

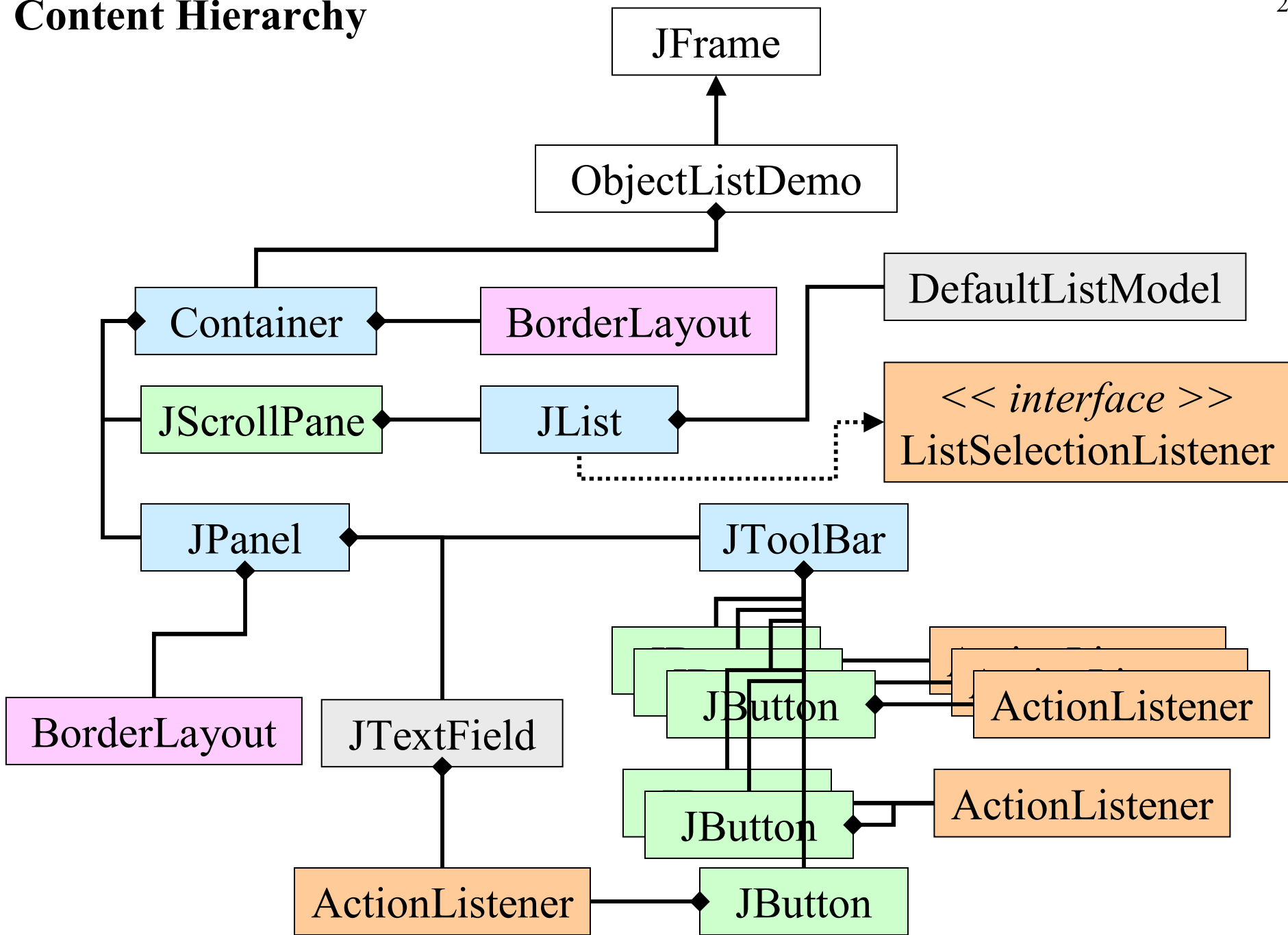
Each object has an item and a rank (randomly set)

Items sortable by rank.

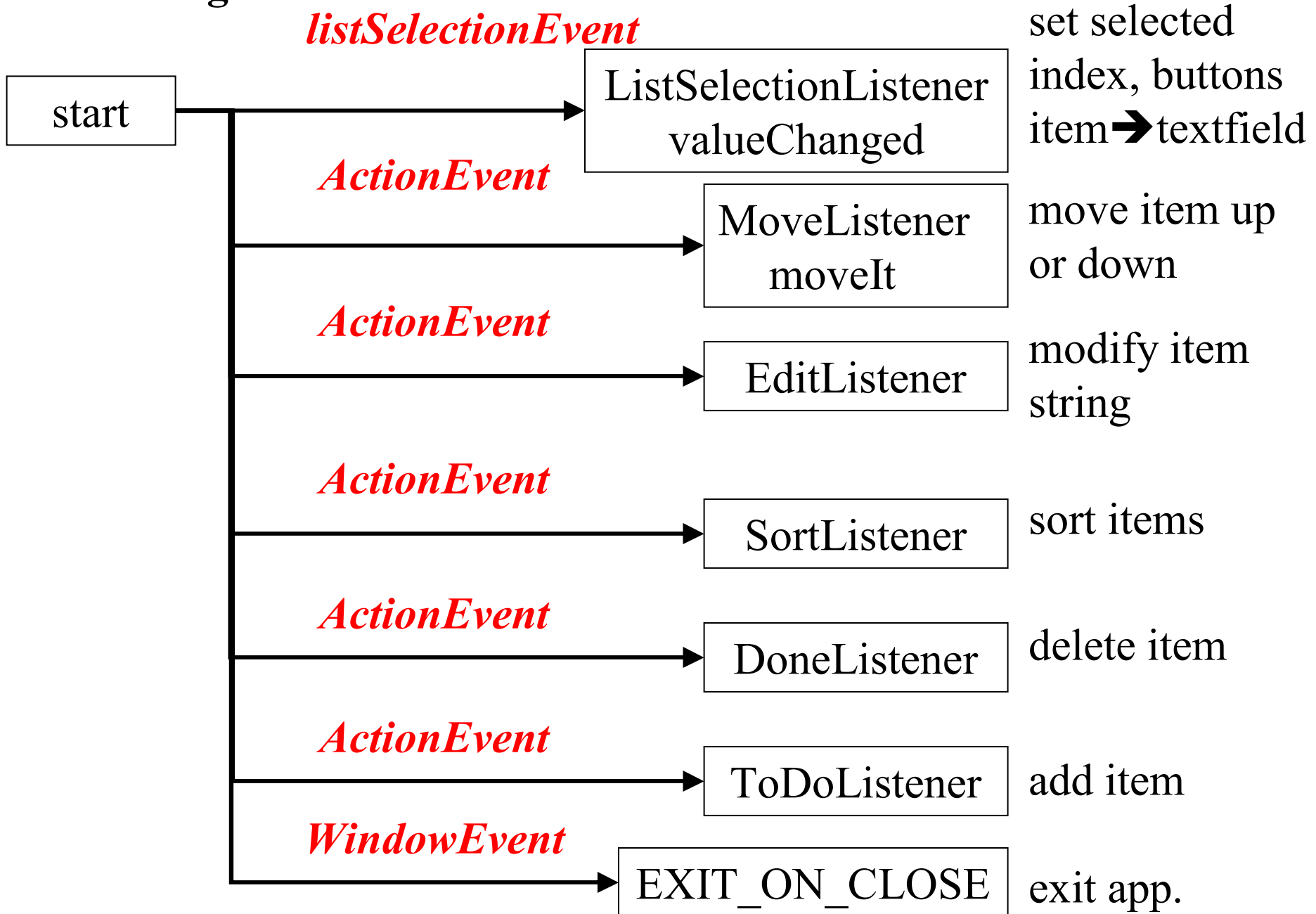
Icon button labels

Floating toolbar

Content Hierarchy



Event Diagram



Event	Listener	Intention
ListSelectionEvent	ListSelectionListener valueChanged	set selected index enable/disable buttons item → textfield
ActionEvent	MoveListener moveIt	move item up / down
ActionEvent	EditListener	edit item
ActionEvent	SortListener	sort items on rank
ActionEvent	DoneListener	delete item
ActionEvent	ToDoListener	add item
WindowEvent	EXIT_ON_CLOSE	kill app

TableDemo

Date	Calories	Fat	Exercise
Tue, Sep 9, 03	1600	500	<input type="checkbox"/>
Tue, Sep 9, 03	2150	650	<input checked="" type="checkbox"/>
Tue, Sep 9, 03	2600	850	<input type="checkbox"/>
Tue, Sep 9, 03	2150	650	<input checked="" type="checkbox"/>

Cell editors use UI controls (JComboBox, JCheckBox) to select values for cells.

Cell renderers use UI controls (JLabel, JCheckBox) to represent the values of cells.

Cell renderers can be for a column, Class type, or individual cells.

Date column used a customized DefaultTableCellRenderer

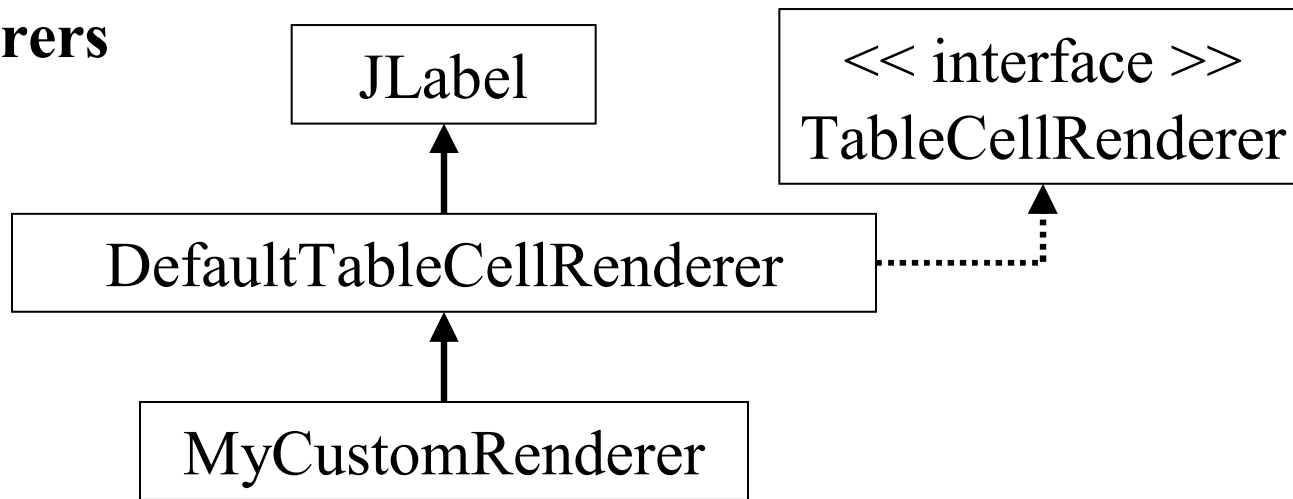
Calories and Fat columns uses a JComboBox cell editor.

Calories column uses a customized DefaultTableCellRenderer

Fat column uses the DefaultTableCellRenderer

Exercise column uses a JCheckbox cell editor and renderer.

Renderers



```

class MyCustomRenderer extends DefaultTableCellRenderer {
    public Component getTableCellRendererComponent(
        JTable t, Object value, boolean isSelected,
        boolean hasFocus, int row, int col) {
        ...
        setValue(...);
        ...
        return this;
    }
}
  
```

```

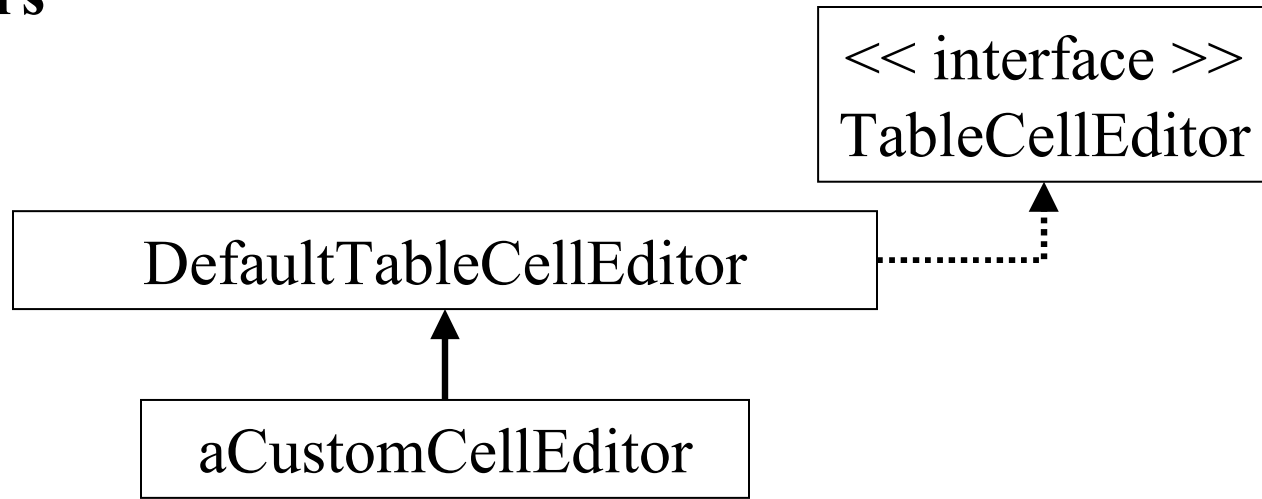
classDiagram
    class JCheckBox
    class TableCellRenderer {
        <<interface>>
    }
    class MyCheckboxRenderer
    JCheckBox --|> MyCheckboxRenderer
    MyCheckboxRenderer ..> TableCellRenderer
  
```

The diagram illustrates the following relationships:

- `MyCheckboxRenderer` inherits from `JCheckBox` (solid arrow with open arrowhead).
- `MyCheckboxRenderer` implements the `TableCellRenderer` interface (dotted arrow with open arrowhead).

```

class MyCheckboxRenderer extends JCheckBox
    implements TableCellRender { ... }
  
```



If not `JTextField`, `JCheckBox`, or `JComboBox` you need to:

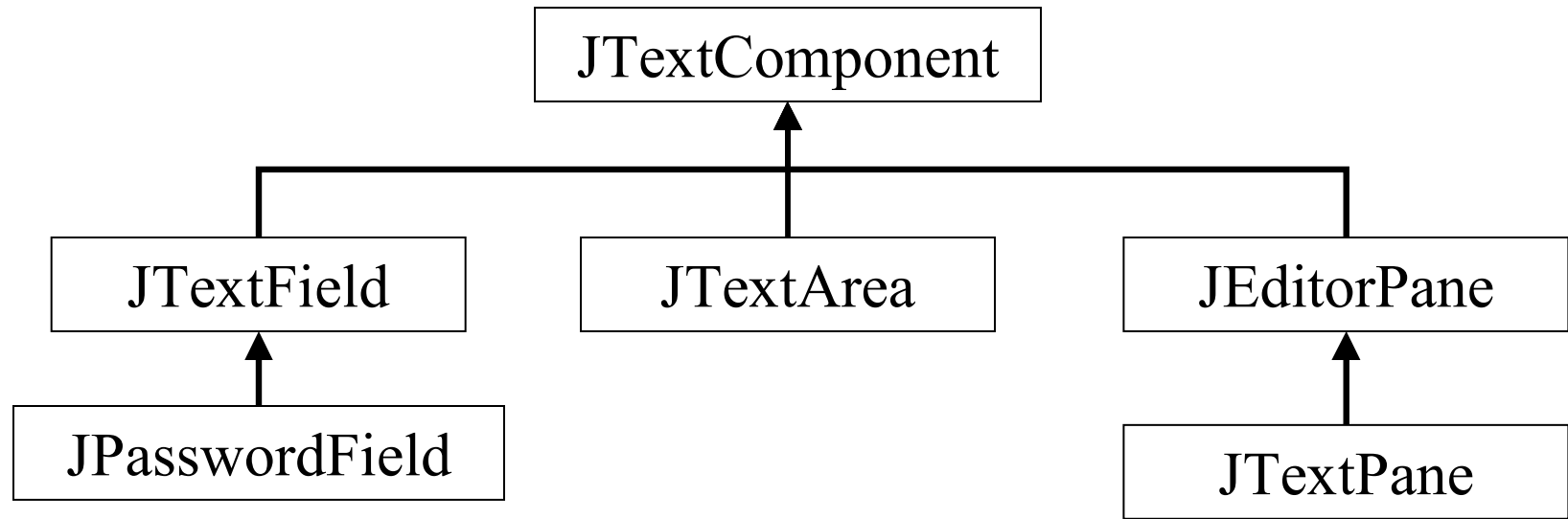
- substituted the component to be the editor.
- set the default editor – `setDefaultEditor` or `setCellEditor`

see Sun's `TableDialogEditDemo.java` example and description on the Sun Java GUI trail (under Tables).

<http://java.sun.com/docs/books/tutorial/uiswing/components/example-1dot4/TableDialogEditDemo.java>

Substitutes a button (for a `JCheckbox`) that is used to display a `JColorChooser` to get a color for the cell's background color.

TextComponents



JTextComponents

document, the model, manages components content
view displays content

editor kit, implements editing of content (*controller*)

supports infinite undo / redo

pluggable caret and caret change listeners

JEditorPane easy way to have *.rtf files created to load document for help dialogs – can also be hyper text (html editor kit).