

32 混合方式UI设计

Qt设计师的功能虽然强大，但也不是万能的。纵使在其中可以完成有关用户界面设计的绝大部分任务，但仍有少量工作需要靠手工编写C++代码来完成。这种UI设计模式即所谓混合方式UI设计。

32.1 创建项目

通过QtCreator，在C:\Users\Minwei\Projects\Qt路径下，创建名为Editor的项目。

32.2 添加资源

C:\Users\Minwei\Projects\Qt\Editor\Editor.qrc:

```
1 <RCC>
2     <qresource prefix="/">
3         <file>images/bld.bmp</file>
4         <file>images/clear.bmp</file>
5         <file>images/copy.bmp</file>
6         <file>images/cut.bmp</file>
7         <file>images/exit.bmp</file>
8         <file>images/itl.bmp</file>
9         <file>images/new.bmp</file>
10        <file>images/open.bmp</file>
11        <file>images/paste.bmp</file>
12        <file>images/under.bmp</file>
13        <file>images/AppIcon.ico</file>
14    </qresource>
15 </RCC>
16
```

32.3 设计界面

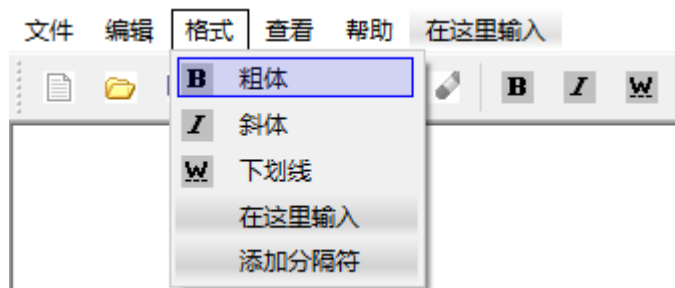
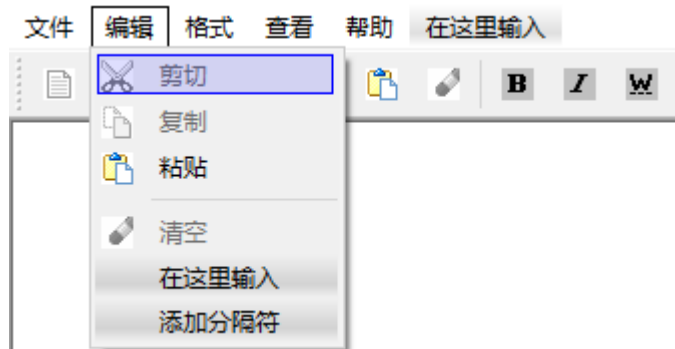
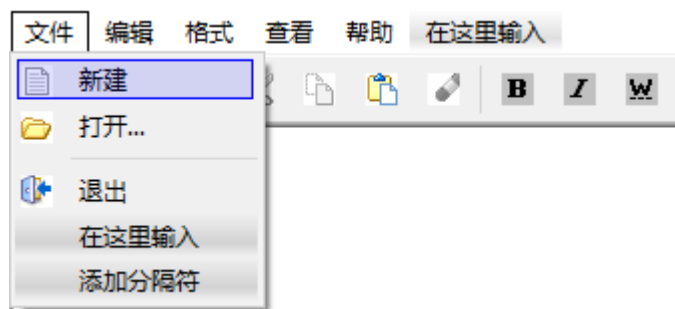
32.3.1 在Qt设计师中完成的界面设计

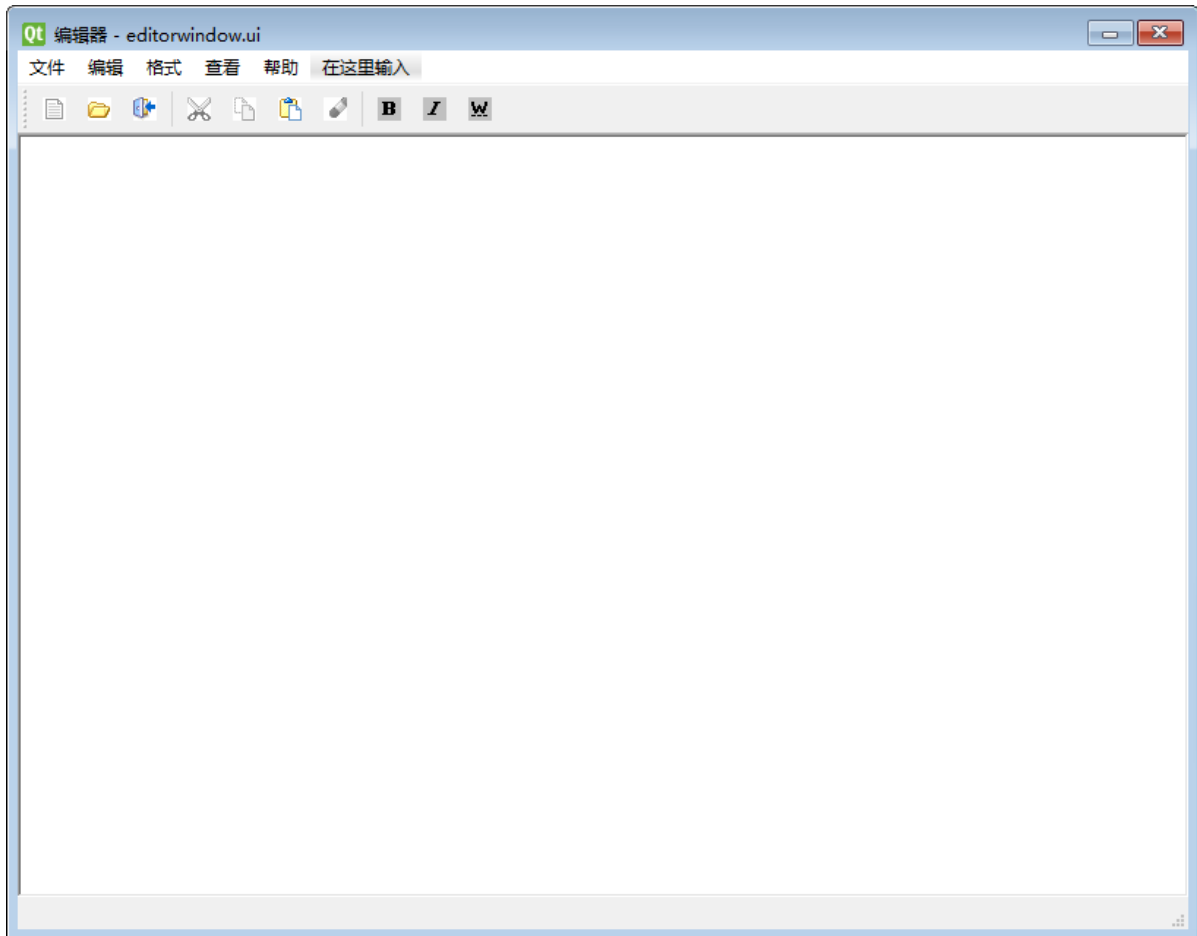
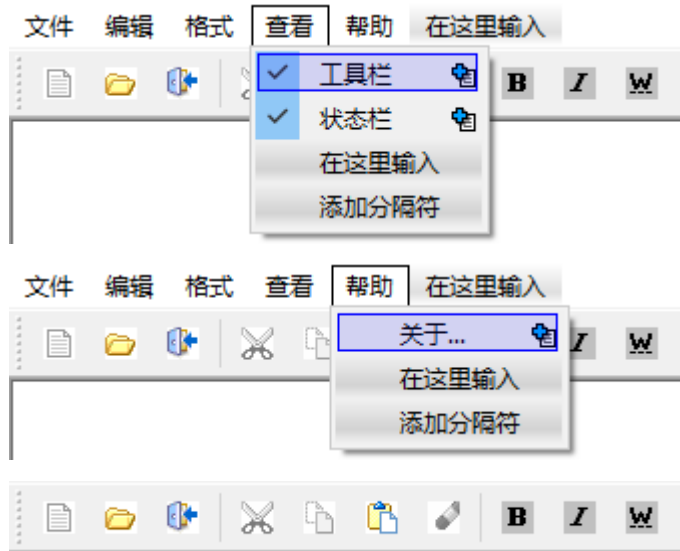
在Qt设计师中完成界面设计一般遵循如下步骤：

1. 设计动作
2. 设计菜单栏
3. 设计工具栏
4. 设计中心组件
5. 添加预定义信号——槽连接

名称	使用	文本	快捷键	可选的	工具提示
m_actNew	<input checked="" type="checkbox"/>	新建	Ctrl+N	<input type="checkbox"/>	新建
m_actOpen	<input checked="" type="checkbox"/>	打开...	Ctrl+O	<input type="checkbox"/>	打开
m_actExit	<input checked="" type="checkbox"/>	退出		<input type="checkbox"/>	退出
m_actCut	<input checked="" type="checkbox"/>	剪切	Ctrl+X	<input type="checkbox"/>	剪切
m_actCopy	<input checked="" type="checkbox"/>	复制	Ctrl+C	<input type="checkbox"/>	复制
m_actPaste	<input checked="" type="checkbox"/>	粘贴	Ctrl+V	<input type="checkbox"/>	粘贴
m_actClear	<input checked="" type="checkbox"/>	清空		<input type="checkbox"/>	清空
B m_actBold	<input checked="" type="checkbox"/>	粗体		<input checked="" type="checkbox"/>	粗体
<i>I</i> m_actItalic	<input checked="" type="checkbox"/>	斜体		<input checked="" type="checkbox"/>	斜体
<u>W</u> m_actUnderline	<input checked="" type="checkbox"/>	下划线		<input checked="" type="checkbox"/>	下划线
<input type="checkbox"/> m_actToolBar	<input checked="" type="checkbox"/>	工具栏		<input checked="" type="checkbox"/>	工具栏
<input type="checkbox"/> m_actStatusBar	<input checked="" type="checkbox"/>	状态栏		<input checked="" type="checkbox"/>	状态栏
<input type="checkbox"/> m_actAbout	<input checked="" type="checkbox"/>	关于...		<input type="checkbox"/>	关于

发送者	信号	接收者	槽
m_actPaste	triggered()	m_edit	paste()
m_actExit	triggered()	EditorWindow	close()
m_actCut	triggered()	m_edit	cut()
m_actCopy	triggered()	m_edit	copy()
m_actClear	triggered()	m_edit	clear()





对象	类
▼ EditorWindow	QMainWindow
▼ m_central	QWidget
m_edit	QTextEdit
▼ m_menuBar	QMenuBar
▼ m_menuFile	QMenu
m_actNew	QAction
m_actOpen	QAction
分隔符	QAction
m_actExit	QAction
▼ m_menuEdit	QMenu
m_actCut	QAction
m_actCopy	QAction
m_actPaste	QAction
分隔符	QAction
m_actClear	QAction
▼ m_menuFormat	QMenu
m_actBold	QAction
m_actItalic	QAction
m_actUnderline	QAction
▼ m_menuView	QMenu
m_actToolBar	QAction
m_actStatusBar	QAction
▼ m_menuHelp	QMenu
m_actAbout	QAction
m_statusBar	QStatusBar
▼ m_toolBar	QToolBar
m_actNew	QAction
m_actOpen	QAction
m_actExit	QAction
分隔符	QAction
m_actCut	QAction
m_actCopy	QAction
m_actPaste	QAction
m_actClear	QAction
分隔符	QAction
m_actBold	QAction
m_actItalic	QAction
m_actUnderline	QAction

C:\Users\Minwei\Projects\Qt\Editor\editorwindow.ui:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <ui version="4.0">
3 <class>Editorwindow</class>
4 <widget class="QMainWindow" name="EditorWindow">
5 <property name="geometry">
6 <rect>
7 <x>0</x>
8 <y>0</y>
9 <width>800</width>

```

```
10     <height>600</height>
11     </rect>
12 </property>
13 <property name="windowTitle">
14     <string>编辑器</string>
15 </property>
16 <property name="windowIcon">
17     <iconset resource="Editor.qrc">
18
19 <normaloff>:/images/AppIcon.ico</normaloff>:/images/AppIcon.ico</iconset>
20 </property>
21 <widget class="Qwidget" name="m_central">
22     <layout class="QHBoxLayout" name="m_layout">
23         <property name="leftMargin">
24             <number>0</number>
25         </property>
26         <property name="topMargin">
27             <number>0</number>
28         </property>
29         <property name="rightMargin">
30             <number>0</number>
31         </property>
32         <property name="bottomMargin">
33             <number>0</number>
34         </property>
35         <item>
36             <widget class="QTextEdit" name="m_edit">
37                 <property name="font">
38                     <font>
39                         <family>微软雅黑</family>
40                         <pointsize>18</pointsize>
41                     </font>
42                 </property>
43                 <property name="frameShape">
44                     <enum>QFrame::WinPanel</enum>
45                 </property>
46                 <property name="frameShadow">
47                     <enum>QFrame::Sunken</enum>
48                 </property>
49             </widget>
50         </item>
51     </layout>
52 </widget>
53 <widget class="QMenuBar" name="m_menuBar">
54     <property name="geometry">
55         <rect>
56             <x>0</x>
57             <y>0</y>
58             <width>800</width>
59             <height>23</height>
60         </rect>
61     </property>
62     <widget class="QMenu" name="m_menuFile">
63         <property name="title">
64             <string>文件</string>
65         </property>
```

```
65     <addaction name="m_actNew"/>
66     <addaction name="m_actOpen"/>
67     <addaction name="separator"/>
68     <addaction name="m_actExit"/>
69 </widget>
70 <widget class="QMenu" name="m_menuEdit">
71     <property name="title">
72         <string>编辑</string>
73     </property>
74     <addaction name="m_actCut"/>
75     <addaction name="m_actCopy"/>
76     <addaction name="m_actPaste"/>
77     <addaction name="separator"/>
78     <addaction name="m_actClear"/>
79 </widget>
80 <widget class="QMenu" name="m_menuView">
81     <property name="title">
82         <string>查看</string>
83     </property>
84     <addaction name="m_actToolBar"/>
85     <addaction name="m_actStatusBar"/>
86 </widget>
87 <widget class="QMenu" name="m_menuFormat">
88     <property name="title">
89         <string>格式</string>
90     </property>
91     <addaction name="m_actBold"/>
92     <addaction name="m_actItalic"/>
93     <addaction name="m_actUnderline"/>
94 </widget>
95 <widget class="QMenu" name="m_menuHelp">
96     <property name="title">
97         <string>帮助</string>
98     </property>
99     <addaction name="m_actAbout"/>
100 </widget>
101 <addaction name="m_menuFile"/>
102 <addaction name="m_menuEdit"/>
103 <addaction name="m_menuFormat"/>
104 <addaction name="m_menuView"/>
105 <addaction name="m_menuHelp"/>
106 </widget>
107 <widget class="QStatusBar" name="m_statusBar"/>
108 <widget class="QToolBar" name="m_toolBar">
109     <property name="windowTitle">
110         <string>工具栏</string>
111     </property>
112     <attribute name="toolBarArea">
113         <enum>TopToolBarArea</enum>
114     </attribute>
115     <attribute name="toolBarBreak">
116         <bool>>false</bool>
117     </attribute>
118     <addaction name="m_actNew"/>
119     <addaction name="m_actOpen"/>
120     <addaction name="m_actExit"/>
```

```
121 <addaction name="separator"/>
122 <addaction name="m_actCut"/>
123 <addaction name="m_actCopy"/>
124 <addaction name="m_actPaste"/>
125 <addaction name="m_actClear"/>
126 <addaction name="separator"/>
127 <addaction name="m_actBold"/>
128 <addaction name="m_actItalic"/>
129 <addaction name="m_actUnderline"/>
130 </widget>
131 <action name="m_actNew">
132 <property name="icon">
133 <iconset resource="Editor.qrc">
134 <normaloff>:/images/new.bmp</normaloff>:/images/new.bmp</iconset>
135 </property>
136 <property name="text">
137 <string>新建</string>
138 </property>
139 <property name="shortcut">
140 <string>Ctrl+N</string>
141 </property>
142 </action>
143 <action name="m_actOpen">
144 <property name="icon">
145 <iconset resource="Editor.qrc">
146 <normaloff>:/images/open.bmp</normaloff>:/images/open.bmp</iconset>
147 </property>
148 <property name="text">
149 <string>打开...</string>
150 </property>
151 <property name="iconText">
152 <string>打开</string>
153 </property>
154 <property name="shortcut">
155 <string>Ctrl+O</string>
156 </property>
157 </action>
158 <action name="m_actExit">
159 <property name="icon">
160 <iconset resource="Editor.qrc">
161 <normaloff>:/images/exit.bmp</normaloff>:/images/exit.bmp</iconset>
162 </property>
163 <property name="text">
164 <string>退出</string>
165 </property>
166 </action>
167 <action name="m_actCut">
168 <property name="enabled">
169 <bool>>false</bool>
170 </property>
171 <property name="icon">
172 <iconset resource="Editor.qrc">
173 <normaloff>:/images/cut.bmp</normaloff>:/images/cut.bmp</iconset>
174 </property>
175 <property name="text">
176 <string>剪切</string>
```

```
177     </property>
178     <property name="shortcut">
179         <string>Ctrl+X</string>
180     </property>
181 </action>
182 <action name="m_actCopy">
183     <property name="enabled">
184         <bool>>false</bool>
185     </property>
186     <property name="icon">
187         <iconset resource="Editor.qrc">
188             <normaloff>:/images/copy.bmp</normaloff>:/images/copy.bmp</iconset>
189     </property>
190     <property name="text">
191         <string>复制</string>
192     </property>
193     <property name="shortcut">
194         <string>Ctrl+C</string>
195     </property>
196 </action>
197 <action name="m_actPaste">
198     <property name="icon">
199         <iconset resource="Editor.qrc">
200             <normaloff>:/images/paste.bmp</normaloff>:/images/paste.bmp</iconset>
201     </property>
202     <property name="text">
203         <string>粘贴</string>
204     </property>
205     <property name="shortcut">
206         <string>Ctrl+V</string>
207     </property>
208 </action>
209 <action name="m_actClear">
210     <property name="enabled">
211         <bool>>false</bool>
212     </property>
213     <property name="icon">
214         <iconset resource="Editor.qrc">
215             <normaloff>:/images/clear.bmp</normaloff>:/images/clear.bmp</iconset>
216     </property>
217     <property name="text">
218         <string>清空</string>
219     </property>
220 </action>
221 <action name="m_actBold">
222     <property name="checkable">
223         <bool>>true</bool>
224     </property>
225     <property name="icon">
226         <iconset resource="Editor.qrc">
227             <normaloff>:/images/bld.bmp</normaloff>:/images/bld.bmp</iconset>
228     </property>
229     <property name="text">
230         <string>粗体</string>
231     </property>
232 </action>
```



```
233 <action name="m_actItalic">
234   <property name="checkable">
235     <bool>true</bool>
236   </property>
237   <property name="icon">
238     <iconset resource="Editor.qrc">
239       <normaloff>:/images/it1.bmp</normaloff>:/images/it1.bmp</iconset>
240     </property>
241     <property name="text">
242       <string>斜体</string>
243     </property>
244   </action>
245 <action name="m_actUnderline">
246   <property name="checkable">
247     <bool>true</bool>
248   </property>
249   <property name="icon">
250     <iconset resource="Editor.qrc">
251       <normaloff>:/images/under.bmp</normaloff>:/images/under.bmp</iconset>
252     </property>
253     <property name="text">
254       <string>下划线</string>
255     </property>
256   </action>
257 <action name="m_actToolBar">
258   <property name="checkable">
259     <bool>true</bool>
260   </property>
261   <property name="checked">
262     <bool>true</bool>
263   </property>
264   <property name="text">
265     <string>工具栏</string>
266   </property>
267 </action>
268 <action name="m_actStatusBar">
269   <property name="checkable">
270     <bool>true</bool>
271   </property>
272   <property name="checked">
273     <bool>true</bool>
274   </property>
275   <property name="text">
276     <string>状态栏</string>
277   </property>
278 </action>
279 <action name="m_actAbout">
280   <property name="text">
281     <string>关于...</string>
282   </property>
283 </action>
284 </widget>
285 <resources>
286   <include location="Editor.qrc"/>
287 </resources>
288 <connections>
```

```
289 <connection>
290   <sender>m_actExit</sender>
291   <signal>triggered</signal>
292   <receiver>Editorwindow</receiver>
293   <slot>close</slot>
294   <hints>
295     <hint type="sourcelabel">
296       <x>-1</x>
297       <y>-1</y>
298     </hint>
299     <hint type="destinationlabel">
300       <x>399</x>
301       <y>299</y>
302     </hint>
303   </hints>
304 </connection>
305 <connection>
306   <sender>m_actCut</sender>
307   <signal>triggered</signal>
308   <receiver>m_edit</receiver>
309   <slot>cut</slot>
310   <hints>
311     <hint type="sourcelabel">
312       <x>-1</x>
313       <y>-1</y>
314     </hint>
315     <hint type="destinationlabel">
316       <x>399</x>
317       <y>317</y>
318     </hint>
319   </hints>
320 </connection>
321 <connection>
322   <sender>m_actCopy</sender>
323   <signal>triggered</signal>
324   <receiver>m_edit</receiver>
325   <slot>copy</slot>
326   <hints>
327     <hint type="sourcelabel">
328       <x>-1</x>
329       <y>-1</y>
330     </hint>
331     <hint type="destinationlabel">
332       <x>399</x>
333       <y>317</y>
334     </hint>
335   </hints>
336 </connection>
337 <connection>
338   <sender>m_actPaste</sender>
339   <signal>triggered</signal>
340   <receiver>m_edit</receiver>
341   <slot>paste</slot>
342   <hints>
343     <hint type="sourcelabel">
344       <x>-1</x>
```

```

345     <y>-1</y>
346 </hint>
347 <hint type="destinationlabel">
348     <x>399</x>
349     <y>317</y>
350 </hint>
351 </hints>
352 </connection>
353 <connection>
354     <sender>m_actClear</sender>
355     <signal>triggered()</signal>
356     <receiver>m_edit</receiver>
357     <slot>clear()</slot>
358     <hints>
359     <hint type="sourceLabel">
360     <x>-1</x>
361     <y>-1</y>
362 </hint>
363 <hint type="destinationlabel">
364     <x>399</x>
365     <y>317</y>
366 </hint>
367 </hints>
368 </connection>
369 </connections>
370 </ui>

```

32.3.2 手工编写代码完成的界面设计

C:\Users\Minwei\Projects\Qt\Editor\editorwindow.h:

```

1  ...
2  #include <QFontComboBox>
3  #include <QSpinBox>
4  #include <QLabel>
5  ...
6  class EditorWindow : public QMainWindow
7  {
8      ...
9  private:
10     ...
11     QFontComboBox* m_combFontName;
12     QSpinBox* m_spinFontSize;
13     QLabel* m_labCurFile;
14     ...
15 };
16 ...

```

C:\Users\Minwei\Projects\Qt\Editor\editorwindow.cpp:

```

1  ...
2  EditorWindow::EditorWindow(QWidget *parent)
3      : QMainWindow(parent)
4      , ui(new Ui::EditorWindow)

```

```

5  {
6      ...
7      ui->m_toolBar->addSeparator();
8
9      ui->m_toolBar->addWidget(new QLabel(" 字体名称: "));
10     m_combFontName = new QFontComboBox;
11     m_combFontName->setMinimumWidth(80);
12     ui->m_toolBar->addWidget(m_combFontName);
13
14     ui->m_toolBar->addWidget(new QLabel(" 字体大小: "));
15     m_spinFontSize = new QSpinBox;
16     m_spinFontSize->setMinimumWidth(60);
17     m_spinFontSize->setRange(8, 72);
18     ui->m_toolBar->addWidget(m_spinFontSize);
19
20     m_labCurFile = new QLabel("当前文件");
21     m_labCurFile->setMinimumWidth(400);
22     ui->m_statusBar->addWidget(m_labCurFile);
23     ...
24 }

```

32.4 实现功能

通过自定义槽函数，响应用户触发的各种UI事件，实现业务逻辑。

C:\Users\Minwei\Projects\Qt\Editor\editorwindow.h:

```

1  ...
2  class EditorWindow : public QMainWindow
3  {
4      ...
5  private slots:
6      void on_m_actNew_triggered();
7      void on_m_actOpen_triggered();
8
9      void on_m_actBold_triggered(bool checked);
10     void on_m_actItalic_triggered(bool checked);
11     void on_m_actUnderline_triggered(bool checked);
12
13     void on_m_actToolBar_triggered(bool checked);
14     void on_m_actStatusBar_triggered(bool checked);
15
16     void on_m_actAbout_triggered();
17
18     void on_m_combFontName_currentIndexChanged(const QString& fontName);
19     void on_m_spinFontSize_valueChanged(int fontSize);
20
21     void on_m_edit_copyAvailable(bool b);
22     void on_m_edit_textChanged();
23     void on_m_edit_selectionChanged();
24     ...
25 };
26 ...

```

C:\Users\Minwei\Projects\Qt\Editor\editorwindow.cpp:

```

1  #include <QMessageBox>
2  ...
3  EditorWindow::EditorWindow(QWidget *parent)
4      : QMainWindow(parent)
5      , ui(new Ui::EditorWindow)
6  {
7      ...
8      on_m_edit_selectionChanged();
9
10     connect(m_combFontName, SIGNAL(currentIndexChanged(QString)),
11             this, SLOT(on_m_combFontName_currentIndexChanged(QString)));
12     connect(m_spinFontSize, SIGNAL(valueChanged(int)),
13             this, SLOT(on_m_spinFontSize_valueChanged(int)));
14     ...
15 }
16 ...
17 void EditorWindow::on_m_actNew_triggered()
18 {
19     QMessageBox::information(this, windowTitle(), "处理“新建”菜单项");
20 }
21
22 void EditorWindow::on_m_actOpen_triggered()
23 {
24     QMessageBox::information(this, windowTitle(), "处理“打开...”菜单项");
25 }
26
27 void EditorWindow::on_m_actBold_triggered(bool checked)
28 {
29     QTextCharFormat fmt;
30     fmt.setFontWeight(checked ? QFont::Bold : QFont::Normal);
31     ui->m_edit->mergeCurrentCharFormat(fmt);
32 }
33
34 void EditorWindow::on_m_actItalic_triggered(bool checked)
35 {
36     QTextCharFormat fmt;
37     fmt.setFontItalic(checked);
38     ui->m_edit->mergeCurrentCharFormat(fmt);
39 }
40
41 void EditorWindow::on_m_actUnderline_triggered(bool checked)
42 {
43     QTextCharFormat fmt;
44     fmt.setFontUnderline(checked);
45     ui->m_edit->mergeCurrentCharFormat(fmt);
46 }
47
48 void EditorWindow::on_m_actToolBar_triggered(bool checked)
49 {
50     ui->m_toolBar->setHidden(!checked);
51 }
52
53 void EditorWindow::on_m_actStatusBar_triggered(bool checked)
54 {
55     ui->m_statusBar->setHidden(!checked);

```

```

56 }
57
58 void EditorWindow::on_m_actAbout_triggered()
59 {
60     QMessageBox::about(this, windowTitle(), "混合方式UI设计\n版本: 1.0");
61 }
62
63 void EditorWindow::on_m_combFontName_currentIndexChanged(
64     const QString& fontName)
65 {
66     QTextCharFormat fmt;
67     fmt = ui->m_edit->currentCharFormat();
68     fmt.setFontFamily(fontName);
69     ui->m_edit->mergeCurrentCharFormat(fmt);
70 }
71
72 void EditorWindow::on_m_spinFontSize_valueChanged(int fontSize)
73 {
74     QTextCharFormat fmt;
75     fmt = ui->m_edit->currentCharFormat();
76     fmt.setFontPointSize(fontSize);
77     ui->m_edit->mergeCurrentCharFormat(fmt);
78 }
79
80 void EditorWindow::on_m_edit_copyAvailable(bool b)
81 {
82     ui->m_actCut->setEnabled(b);
83     ui->m_actCopy->setEnabled(b);
84 }
85
86 void EditorWindow::on_m_edit_textChanged()
87 {
88     ui->m_actClear->setEnabled(ui->m_edit->toPlainText().size());
89 }
90
91 void EditorWindow::on_m_edit_selectionChanged()
92 {
93     QFont font = ui->m_edit->currentFont();
94
95     ui->m_actBold->setChecked(font.bold());
96     ui->m_actItalic->setChecked(font.italic());
97     ui->m_actUnderline->setChecked(font.underline());
98
99     m_combFontName->setCurrentFont(font);
100     m_spinFontSize->setValue(font.pointSize());
101 }
102 ...

```

32.5 测试验证

运行效果如图所示:

