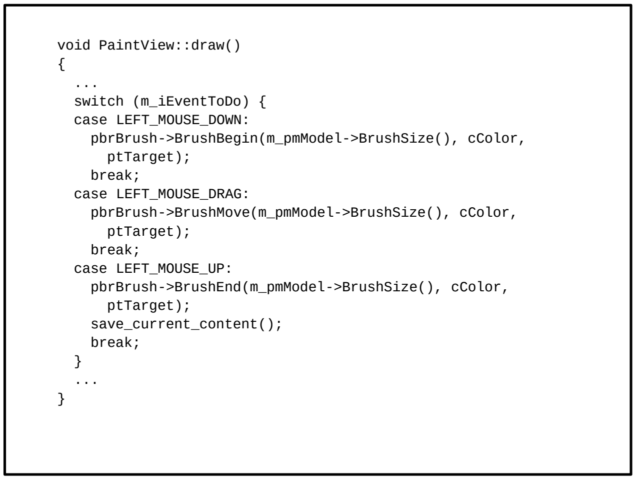
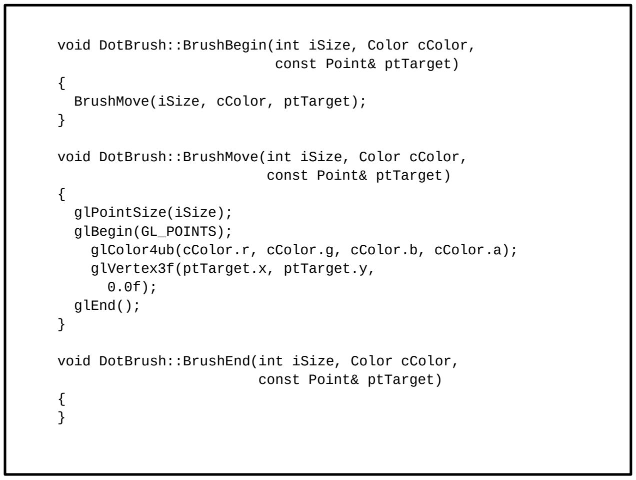


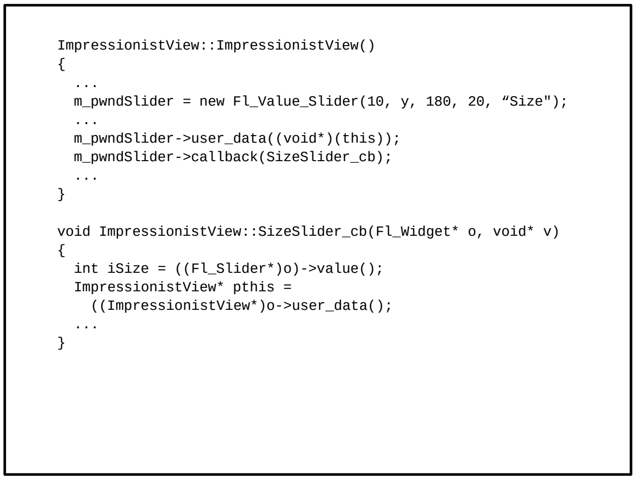
```
void PaintView::draw()
{
  ...
  switch (m_iEventToDo) {
  case LEFT_MOUSE_DOWN:
    pbrBrush->BrushBegin(m_pmModel->BrushSize(), cColor,
      ptTarget);
    break;
  case LEFT_MOUSE_DRAG:
    pbrBrush->BrushMove(m_pmModel->BrushSize(), cColor,
      ptTarget);
    break;
  case LEFT_MOUSE_UP:
    pbrBrush->BrushEnd(m_pmModel->BrushSize(), cColor,
      ptTarget);
    save_current_content();
    break;
  }
  ...
}
```



```
void DotBrush::BrushBegin(int iSize, Color cColor,
                          const Point& ptTarget)
{
  BrushMove(iSize, cColor, ptTarget);
}

void DotBrush::BrushMove(int iSize, Color cColor,
                         const Point& ptTarget)
{
  glPointSize(iSize);
  glBegin(GL_POINTS);
    glColor4ub(cColor.r, cColor.g, cColor.b, cColor.a);
    glVertex3f(ptTarget.x, ptTarget.y,
      0.0f);
  glEnd();
}

void DotBrush::BrushEnd(int iSize, Color cColor,
                        const Point& ptTarget)
{
}
```



```
ImpressionistView::ImpressionistView()
{
  ...
  m_pwndSlider = new Fl_Value_Slider(10, y, 180, 20, “Size");
  ...
  m_pwndSlider->user_data((void*)(this));
  m_pwndSlider->callback(SizeSlider_cb);
  ...
}

void ImpressionistView::SizeSlider_cb(Fl_Widget* o, void* v)
{
  int iSize = ((Fl_Slider*)o)->value();
  ImpressionistView* pthis =
    ((ImpressionistView*)o->user_data();
  ...
}
```