



UNIVERSIDAD  
POLITECNICA  
DE VALENCIA



Escuela Técnica Superior de Ingeniería del Diseño

# Qt svg

Pablo Laseca Jiménez

Francesc Navarro Mari

# Scalable Vector Graphics

**Scalable Vector Graphics (SVG)** es una especificación para describir [gráficos vectoriales](#) bidimensionales, tanto estáticos como animados en formato **XML**.

## Vista general

- La imagen vectorial puede ser escalada, tanto como se requiera, sin pérdida de calidad de imagen.



**BITMAP**  
.jpeg .gif .png



**OUTLINE**  
.svg

## El SVG permite tres tipos de objetos gráficos:

- Formas gráficas de vectores (consistentes en rectas, curvas, y áreas limitadas por ellos).
- Imágenes de [mapa de bits](#) / [digitales](#).
- Texto.

## Impacto en la Web

- La amplia adopción de clientes **SVG**, particularmente aquellos integrados nativamente en los navegadores, como [Firefox](#), [Opera](#) o [Safari](#), puede traer un significativo cambio de imagen en la Web.

## Complejidad

- A pesar de ser un lenguaje vectorial, **SVG** permite crear imágenes complejas.



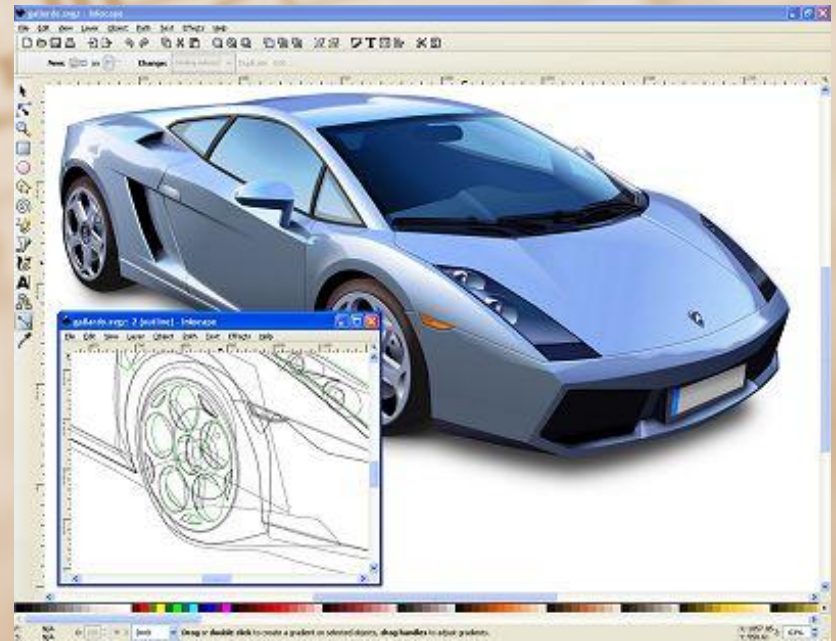
Tomate en SVG

# Inkscape



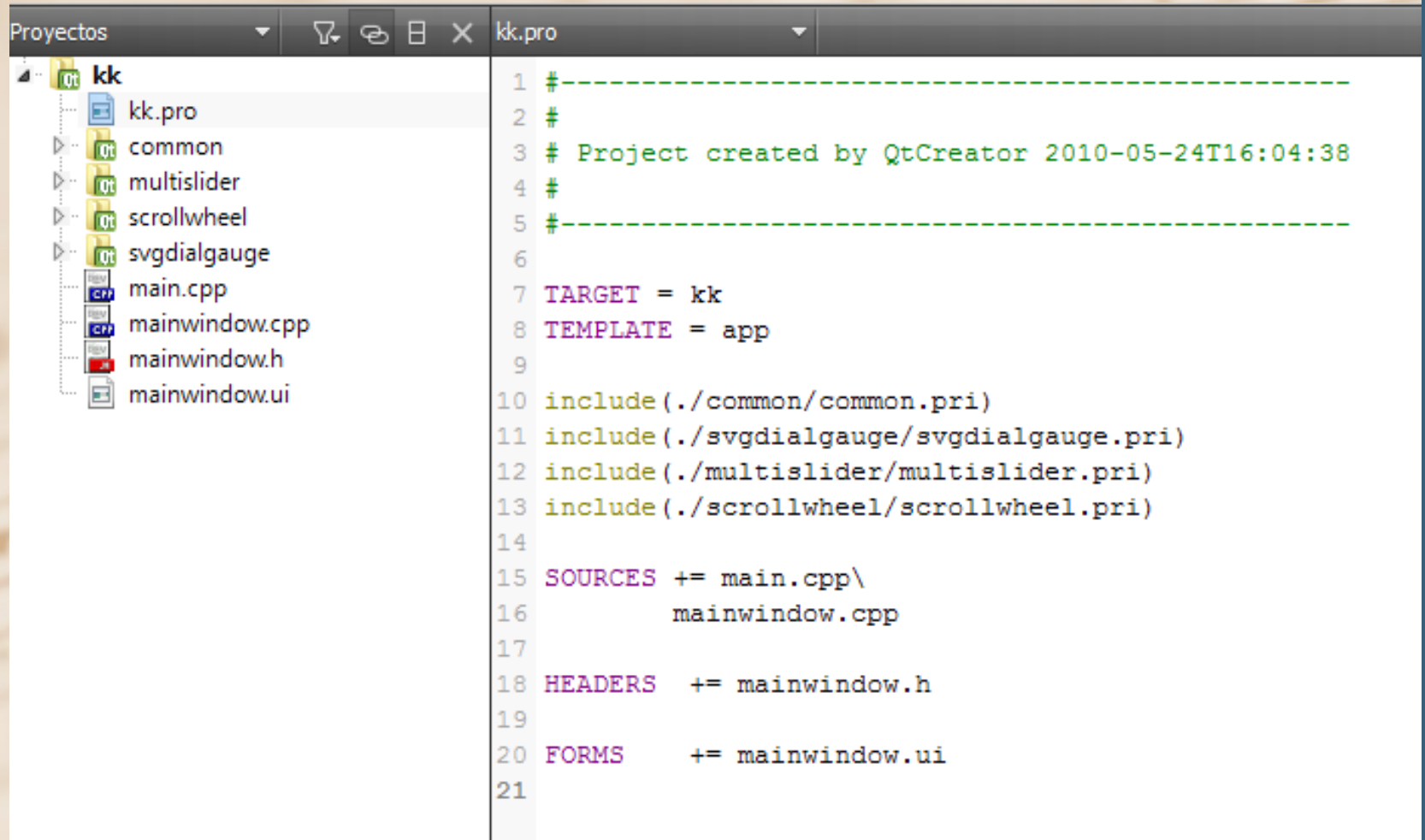
***Inkscape*** es una herramienta de *dibujo libre* y *multiplataforma* para gráficos vectoriales SVG.

- Las características de **SVG** soportadas incluyen formas básicas, trayectorias, texto, canal alfa, transformaciones, gradientes, edición de nodos, exportación de SVG a PNG, agrupación de elementos, etc.
- El objetivo principal de ***Inkscape*** es crear una herramienta de dibujo potente y cómoda, totalmente compatible con los estándares **XML**, **SVG** y **CSS**.



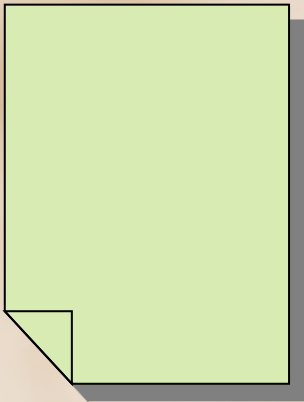


# Creación de Widget's industriales



The image shows a screenshot of the Qt Creator IDE. On the left, the 'Proyectos' (Projects) pane displays a tree view for a project named 'kk'. The project structure includes a 'kk.pro' file, a 'common' sub-project, and three widget sub-projects: 'multislider', 'scrollwheel', and 'svgdialgauge'. The main project files listed are 'main.cpp', 'mainwindow.cpp', 'mainwindow.h', and 'mainwindow.ui'. On the right, the 'kk.pro' file is open, showing the following content:

```
1 #-----  
2 #  
3 # Project created by QtCreator 2010-05-24T16:04:38  
4 #  
5 #-----  
6  
7 TARGET = kk  
8 TEMPLATE = app  
9  
10 include(./common/common.pri)  
11 include(./svgdialgauge/svgdialgauge.pri)  
12 include(./multislider/multislider.pri)  
13 include(./scrollwheel/scrollwheel.pri)  
14  
15 SOURCES += main.cpp\  
16           mainwindow.cpp  
17  
18 HEADERS += mainwindow.h  
19  
20 FORMS   += mainwindow.ui  
21
```



mainwindow.cpp

```
14 MainWindow::MainWindow(QWidget *parent) :
15     QMainWindow(parent),
16     ui(new Ui::MainWindow)
17 {
18     ui->setupUi(this);
19
20     // Tachometer
21     gauge = new QtSvgDialGauge(this);
22     gauge->setSkin("Tachometer"); //Indica la carpeta con las Imagenes SVG
23     gauge->setNeedleOrigin(0.486, 0.466); //Punto central de la aguja
24     gauge->setMinimum(0); //Valor minimo que toma la aguja
25     gauge->setMaximum(100); //Valor Maximo que toma la aguja
26     gauge->setStartAngle(-130);
27     gauge->setEndAngle(133);
28     gauge->setValue(nivel); //Valor que marca la aguja
29     gauge->setMaximumSize(200, 200); //Tamaño
30
31     ui->ly1->addWidget(gauge);
32
33
34     //Deposito
35     multiSlider = new QtMultiSlider(this);
36     multiSlider->setSkin("Botella"); //Indica la carpeta con las Imagenes SVG
37     multiSlider->setValue(nivel); //Valor que tomo el deposito
38
39     ui->ly3->addWidget(multiSlider);
40
41
42     //Motor
43     scrollWheel = new QtScrollWheel(this);
44     scrollWheel->setSkin("Motor"); //Indica la carpeta con las Imagenes SVG
45
46     ui->ly2->addWidget(scrollWheel);
47
48
49     //Valvula
50     scrollWheel2 = new QtScrollWheel(this);
51     scrollWheel2->setSkin("Valvula"); //Indica la carpeta con las Imagenes SVG
52
53     ui->ly4->addWidget(scrollWheel2);
54 }
55
56 MainWindow::~MainWindow()
57 {
58     delete ui;
59 }
```