

6/27/2014

Qt SizePolicy and sizeHint

Michael Brehler

dkfz.


GERMAN
CANCER RESEARCH CENTER
IN THE HELMHOLTZ ASSOCIATION

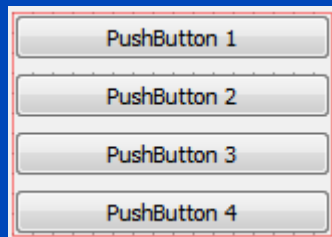



50 Years – Research for
A Life Without Cancer

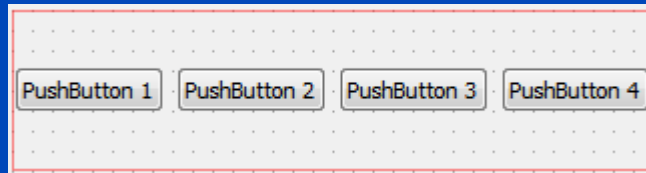
QLayouts


- size policies only have influence on widgets that are inside a layout

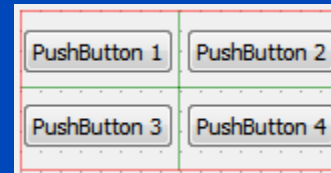
 Vertical Layout




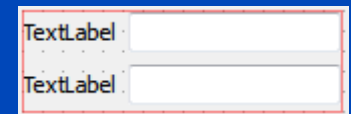
 Horizontal Layout



 Grid Layout



 Form Layout



- The **QSizePolicy** class is a layout attribute describing horizontal and vertical resizing policy.

QSizePolicy

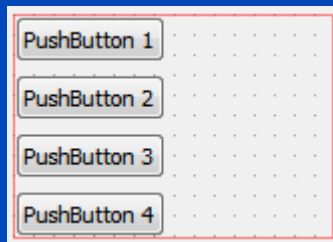
- The size policy of a widget is an expression of its willingness to be resized in various ways, and affects how the widget is treated by the layout engine.
- You can change this for a specific widget by changing its `QWidget::sizePolicy` property.

Constant	Value	Description
<code>QSizePolicy::Fixed</code>	0	The <code>QWidget::sizeHint()</code> is the only acceptable alternative, so the widget can never grow or shrink (e.g. the vertical direction of a push button).
<code>QSizePolicy::Minimum</code>	<code>GrowFlag</code>	The <code>sizeHint()</code> is minimal, and sufficient. The widget can be expanded, but there is no advantage to it being larger (e.g. the horizontal direction of a push button). It cannot be smaller than the size provided by <code>sizeHint()</code> .
<code>QSizePolicy::Maximum</code>	<code>ShrinkFlag</code>	The <code>sizeHint()</code> is a maximum. The widget can be shrunk any amount without detriment if other widgets need the space (e.g. a separator line). It cannot be larger than the size provided by <code>sizeHint()</code> .
<code>QSizePolicy::Preferred</code>	<code>GrowFlag</code> <code>ShrinkFlag</code>	The <code>sizeHint()</code> is best, but the widget can be shrunk and still be useful. The widget can be expanded, but there is no advantage to it being larger than <code>sizeHint()</code> (the default <code>QWidget</code> policy).
<code>QSizePolicy::Expanding</code>	<code>GrowFlag</code> <code>ShrinkFlag</code> <code>ExpandFlag</code>	The <code>sizeHint()</code> is a sensible size, but the widget can be shrunk and still be useful. The widget can make use of extra space, so it should get as much space as possible (e.g. the horizontal direction of a horizontal slider).
<code>QSizePolicy::MinimumExpanding</code>	<code>GrowFlag</code> <code>ExpandFlag</code>	The <code>sizeHint()</code> is minimal, and sufficient. The widget can make use of extra space, so it should get as much space as possible (e.g. the horizontal direction of a horizontal slider).
<code>QSizePolicy::Ignored</code>	<code>ShrinkFlag</code> <code>GrowFlag</code> <code>IgnoreFlag</code>	The <code>sizeHint()</code> is ignored. The widget will get as much space as possible.

- This property holds the recommended size for the widget.
- If the value of this property is an invalid size, no size is recommended.
- The default implementation of `sizeHint()` returns an invalid size if there is no layout for this widget, and returns the layout's preferred size otherwise.
- Standard Qt widgets reimplement the `sizeHint` e.g. `QPushButton`

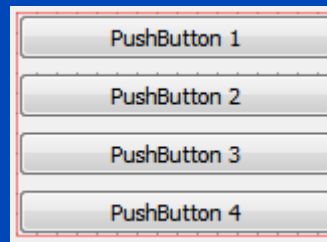
- Example:
 - Vertical Layout
 - sizeHint is set to text size
 - Changing HorizontalPolicy

Fixed



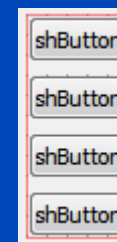
- Never grow or shrink
- Size of sizeHint

Minimum



- Not smaller than sizeHint
- Can grow

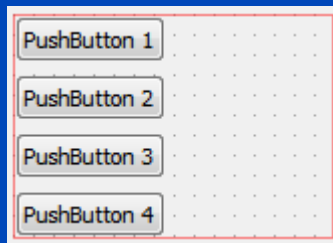
Maximum



- Not larger than sizeHint
- Can shrink

- Example:
 - Vertical Layout
 - sizeHint is set to text size
 - Changing HorizontalPolicy

Preferred



Expanding



MinimumExpanding



- sizeHint is best
- Can grow and shrink
- Default Qwidget policy
- Can be shrunk and still be useful
- Can make use of extra space => it should get as much space as possible
- Not smaller than sizeHint
- Can make use of extra space => it should get as much space as possible

Remeber

- Size policies only have influence on widgets that are inside a layout
- The default implementation of `sizeHint()` returns the layout's preferred size

Summary:

- **sizeHint** is the preferred size of the widget, layouts will try to keep it as close to this as possible
- **sizePolicy** describes how the size may change when the preferred size cannot be used (can it stretch or shrink)
- **sizeConstraint** are the `maximumSize` and `minimumSize` the widget can be

Thank you for
your attention!



dkfz.

GERMAN
CANCER RESEARCH CENTER
IN THE HELMHOLTZ ASSOCIATION



50 Years – Research for
A Life Without Cancer