

Software 1

Recitation No. 11: SWT GUI Package

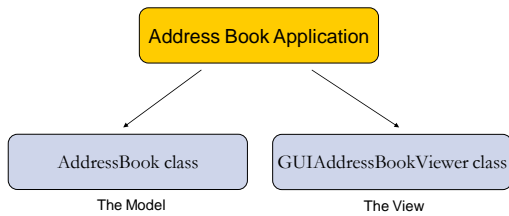
1

Model-View Separation

- Separate between the application logic (**model**) part and the GUI (**view**) part.
- Ensures that view changes have no effect on the basic model
- Enables us to maintain one model for several different views

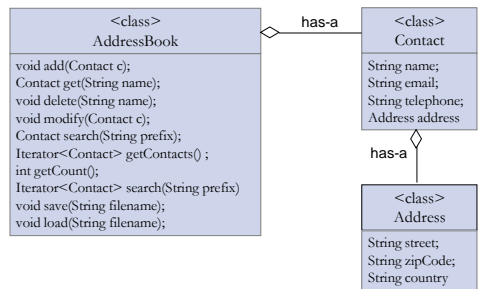
4

Example: Address Book



5

The Model



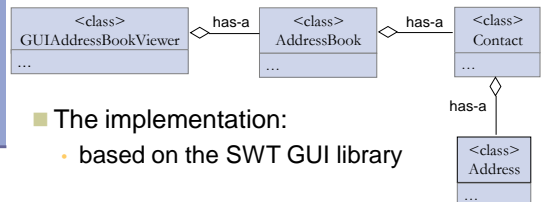
6

The View

7

The View

- The class diagram:



- The implementation:

- based on the SWT GUI library

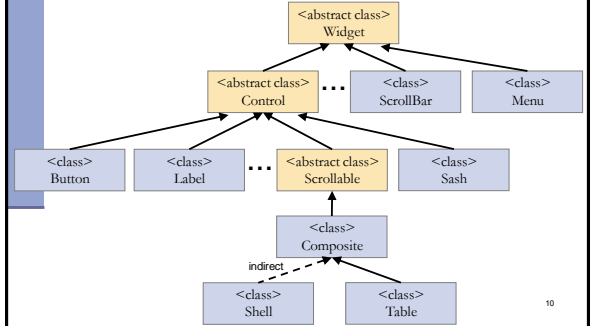
8

SWT

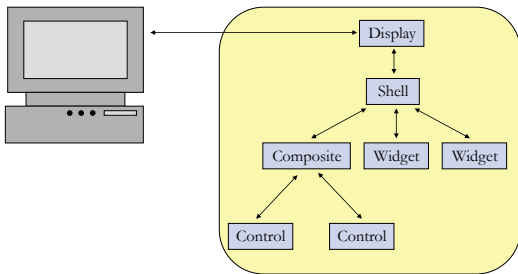
- Online Documentation:
 - SWT HomePage: <http://www.eclipse.org/swt/>
 - JavaDoc
 - Snippets
 - Getting Started with Eclipse and the SWT: <http://www.cs.umanitoba.ca/~eclipse/>

9

Widgets



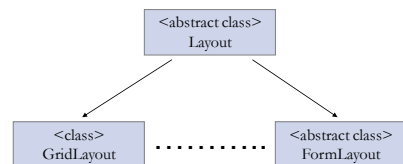
10



11

Layouts

- A Layout controls the position and size of Control widgets in a Composite.



12

GridLayout

- Lays out the Control widgets in a grid.



Each column is as wide as Wide Button 2

14

GridLayout

- Lays out the Control widgets in a grid.
- GridLayout Configuration fields:

Field	Default	Description
horizontalSpacing		Horizontal/vertical space between the grid cells
verticalSpacing	5	Horizontal/vertical space between the grid cells
marginHeight		The size of the horizontal/vertical margins of the layout
marginWidth	5	The size of the horizontal/vertical margins of the layout
numColumns	1	Number of columns
makeColumnsEqualWidth	false	If true, all columns will have the same size

15

GridLayout (cont.)

GridData:

- Use GridData objects to configure the Control widgets in a GridLayout.
- Use the `setLayoutData()` to set a GridData object into a Control, e.g.
`label.setLayoutData(new GridData(...));`
- Do not reuse GridData objects

16

GridLayout (cont.)

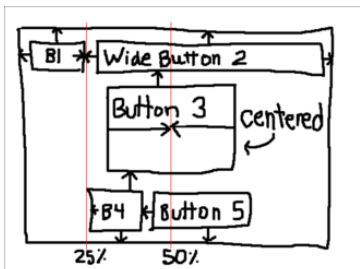
■ GridData Configuration Fields:

Field	Default	Description
<code>grabExcessHorizontalSpace</code> <code>grabExcessVerticalSpace</code>	false	If true, the width/length of the widget will be as large as possible to fit the remaining space.
<code>heightHint</code> <code>widthHint</code>	SWT.DEFAULT minimum) (no	A minimum width/height for the widget.
<code>horizontalSpan</code> <code>verticalSpan</code>	1	the number of column/row cells that the widget will take up.
<code>horizontalIndent</code>	0	the number of indentation pixels along the left side of the cell.
<code>horizontalAlignment</code> <code>verticalAlignment</code>	GridData.BEGINNING GridData.CENTER	how controls will be positioned horizontally/vertically within a cell.

17

FormLayout

- A very flexible layout



18

FormLayouts

- A very flexible layout
- FormLayout Configuration Properties:

Field	Default	Description
<code>marginHeight</code> <code>marginWidth</code>	0	the margin width/height
<code>spacing</code>	0	the number of pixels between the edge of one control and the edge of its neighbouring control.

19

FormLayouts (cont.)

- Use FormData objects to configure the Control widgets in a FormLayout.
- Use the `setLayoutData()` to set a FormData object into a Control widget.
- A FormData object has a FormAttachment object for each edge of the Control.

Field	Description
<code>width/height</code>	the desired width/height in pixels.
<code>top/bottom/left/right</code>	Specifies the position of the control attachment.

20

FormLayouts (cont.)

- A FormAttachment defines where to attach the side of a Control by using the equation: $y = ax + b$.

A fraction defined by:
-numerator
-denominator

an offset, in pixels

the width/height of a Control to which the control side is attached (control).

21

FormLayouts (cont.)

■ Main FormAttachment Constructors:

- `public FormAttachment(Control control)`
- `public FormAttachment(Control control, int offset)`
- `public FormAttachment(int numerator)`
- `public FormAttachment(int numerator, int offset)`

Field	Default
control	Parent Composite
numerator	100
denominator	100
offset	0

$$y = \frac{\text{numerator}}{\text{denominator}} \cdot x + \text{offset}$$

x – control's width/height