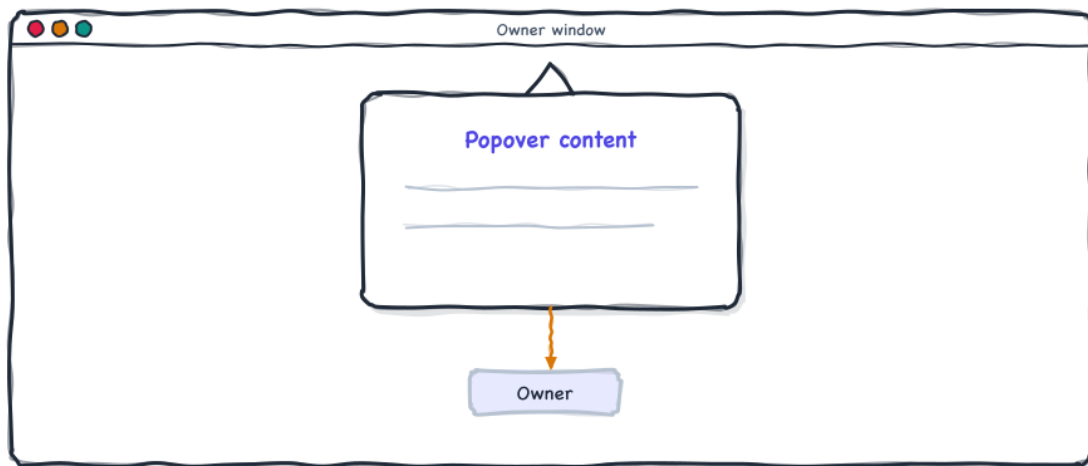


PopOver

Developer Manual

GemsFX • com.dlsc.gemsfx • JavaFX Custom Controls



The popover points at its owner with a styled arrow and can be detached into a movable popup.

PopOver is GemsFX's arrow-decorated PopupControl. It positions content around an owner node, draws an arrow path, flips to remain on screen, can detach by dragging and supports fade durations and arrow geometry styling.

Table of Contents

1. Introduction	2
1.1 Key features	2
1.2 Maven dependency	2
2. Getting started	2
3. Anatomy	3
4. Control API	4
5. Arrow geometry	4
6. Showing and screen fitting	5
7. Detachability and animation	5
8. Styling	6
9. Localization	7
10. Recipes	7
10.1 Calendar popover	7
10.2 Disable animations	7
10.3 Allow tear-off	7
10.4 Checklist	7
11. See also	7

1. Introduction

PopOver is a lightweight popup with a custom path background and an arrow pointing to an owner node. It is useful for contextual details, pickers and inspectors.

1.1 Key features

- Preferred and computed arrow locations.
- Screen fitting and side flipping when the preferred side is unavailable.
- Arrow size, indent and corner-radius properties exposed to CSS.
- Optional detachable mode by dragging more than 20 pixels.
- Fade-in and fade-out animations.
- CalendarPopOver subclass for single-date or date-range selection.

1.2 Maven dependency

```
<dependency>
  <groupId>com.dlsc.gemsfx</groupId>
  <artifactId>gemsfx</artifactId>
  <version>4.4.1</version>
</dependency>
```

GemsFX controls live in module com.dlsc.gemsfx.

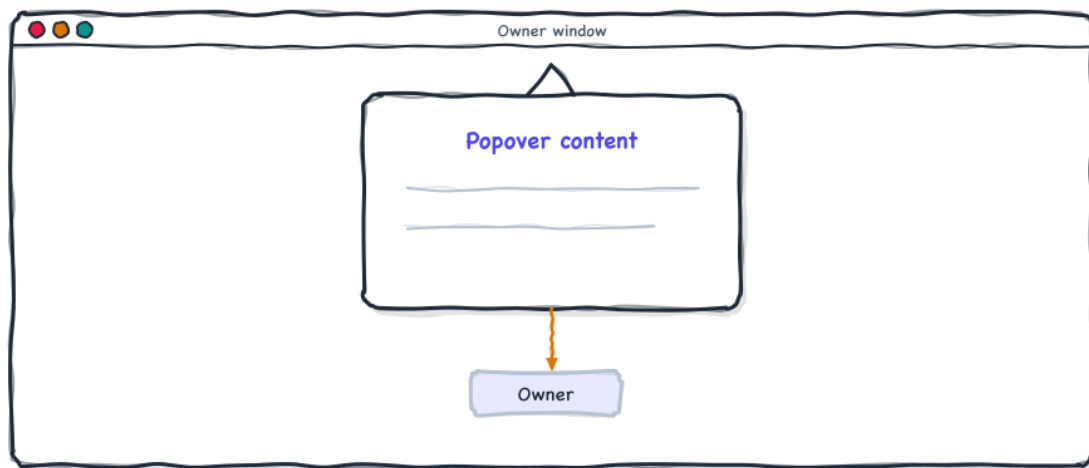
2. Getting started

```
Label content = new Label("More details");
content.setPrefSize(240, 120);

PopOver popOver = new PopOver(content);
popOver.setArrowLocation(PopOver.ArrowLocation.TOP_CENTER);

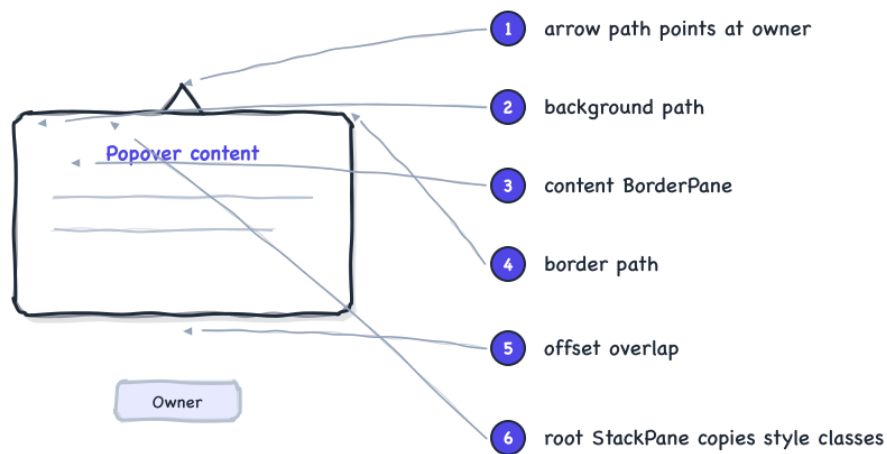
Button owner = new Button("Show");
owner.setOnAction(evt -> popOver.show(owner));
```

A minimal popover attached to a button.



The popover is positioned relative to an owner node.

3. Anatomy



The parts of the PopOver skin.

Part	Style class	Description
.popover	Root style class copied to the root StackPane.	
:detached	Pseudo class active while detached.	
.background	Path filled with control-inner background and drop shadow.	
.border	Path used for the border stroke.	
.content	BorderPane that hosts contentNode.	

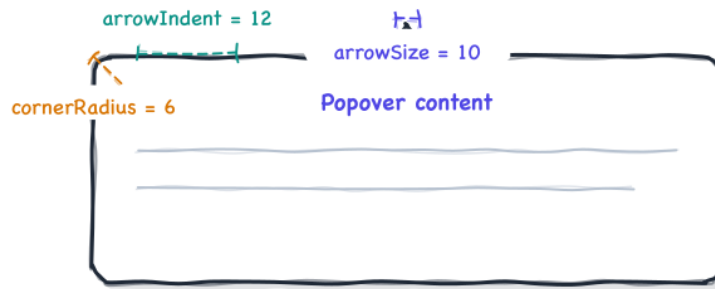
Part	Style class	Description
.calendar-popover	Extra style class used by CalendarPopOver.	

4. Control API

Property	Type	Default	Description
contentNode	ObjectProperty<Node>	Label("Content")	Node displayed in the popover; null is rejected.
detachable	BooleanProperty	false	Allows dragging the popup away from its owner.
detached	BooleanProperty	false	True after detach; hides the arrow and disables auto-hide.
arrowSize	DoubleProperty	10	Arrow size. Styleable as -fx-arrow-size.
arrowIndent	DoubleProperty	12	Distance from arrow to rounded corner. Styleable as -fx-arrow-indent.
cornerRadius	DoubleProperty	6	Rounded-corner radius. Styleable as -fx-corner-radius.
arrowLocation	ObjectProperty<ArrowLocation>	TOP_CENTER	Preferred arrow location.
computedArrowLocation	ReadOnlyObjectProperty<ArrowLocation>	TOP_LEFT initially	Effective location after fitting/flipping on screen.
fadeInDuration / fadeOutDuration	ObjectProperty<Duration>	100 ms / 100 ms	Durations used by show/hide when animated.
animated	BooleanProperty	true	Enables fade-in and fade-out animations.

5. Arrow geometry

The skin builds one rounded path and inserts arrow path elements on the computed edge. The minimum size is bound to $2 * \text{cornerRadius} + 2 * \text{arrowIndent} + 2 * \text{arrowSize}$.

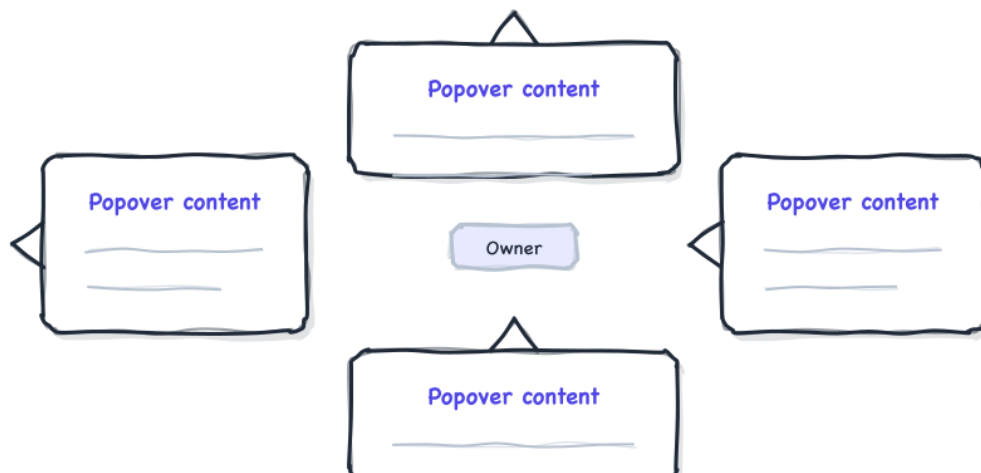


The skin builds one path and inserts arrow segments for the computed location

Arrow size, indent and corner radius.

6. Showing and screen fitting

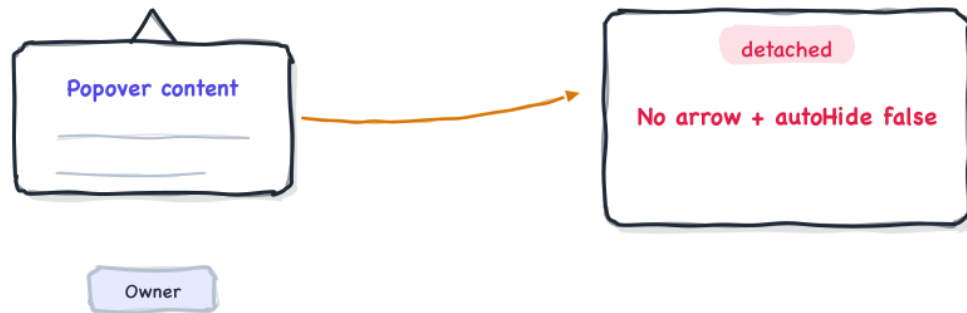
`show(owner)` uses the owner bounds in screen coordinates and the default positive offset 4, which slightly overlaps the owner. `show(owner, bounds, offset)` accepts explicit bounds and offset. The preferred arrow location is never modified; `computedArrowLocation` reports the effective fitted location.



Preferred arrowLocation may flip to stay on screen; `computedArrowLocation` reports the effective side

Preferred locations can flip or adjust along the edge to stay on screen.

7. Detachability and animation



Dragging a detachable popover away hides the arrow and disables auto-hide.

- `setDetachable(true)` enables tear-off dragging.
- Once detached, the `:detached` pseudo class is active and `autoHide` is set to `false`.
- `hide()` delegates to `hide(fadeOutDuration)`.
- `setAnimated(false)` skips fade transitions.

8. Styling

Style class / pseudo class	Description
<code>.popover</code>	Root style class copied to the root StackPane.
<code>:detached</code>	Pseudo class active while detached.
<code>.background</code>	Path filled with control-inner background and drop shadow.
<code>.border</code>	Path used for the border stroke.
<code>.content</code>	BorderPane that hosts contentNode.
<code>.calendar-popover</code>	Extra style class used by CalendarPopOver.

CSS property	Type	Default
<code>-fx-arrow-size</code>	size	10
<code>-fx-arrow-indent</code>	size	12
<code>-fx-corner-radius</code>	size	6

```

.popover {
    -fx-arrow-size: 14px;
    -fx-arrow-indent: 18px;
    -fx-corner-radius: 10px;
}
.popover > .background {
    -fx-fill: white;
}
  
```

Example PopOver CSS.

9. Localization

Key	English default
content.default-label	Content

10. Recipes

10.1 Calendar popover

```
Popover.CalendarPopover picker = new Popover.CalendarPopover(true);  
picker.show(dateButton);
```

10.2 Disable animations

```
popover.setAnimated(false);
```

10.3 Allow tear-off

```
popover.setDetachable(true);
```

10.4 Checklist

- 1 Choose arrowLocation based on where the popup should appear.
- 2 Use computedArrowLocation for skin logic, not arrowLocation.
- 3 Set contentNode before showing.
- 4 Remember detached popovers no longer auto-hide.

11. See also

- Demo app: PopOverApp. Run with `mvn javafx:run -f gemsfx-demo/pom.xml -Dmain.class=com.dlsc.gemsfx.demo.PopOverApp`.
- Related controls: CalendarPicker and DateRangePicker use calendar popovers.
- API docs: <https://dlsc-software-consulting-gmbh.github.io/GemsFX/api/>