

# Object-Oriented Programming with Java

## Recitation No. 11: Custom Widgets

# Custom Widgets

- Non-standard SWT widgets
- The [org.eclipse.swt.custom](http://org.eclipse.swt.custom) package:
  - provides the standard look and feel of Eclipse
- Before writing a custom widget consider the following issues:
  - Can a standard SWT widget be used as a simpler version of the new widget?
  - Can the function be achieved by extending the behavior of a standard SWT widget?
  - Does the new widget have to be portable?

# Portability

---

## ■ Non-portable Implementations:

- Writing a new widget that wraps a native widget
- Extending a standard SWT widget by calling natives

## ■ Portable Implementations:

- Writing a new pure Java widget
- Extending a standard SWT widget in a pure Java manner

# Writing Custom Widgets

- Custom widgets are written by extending a standard widget:

Basic Widgets	Compound Widgets
Draw themselves Do not contain other widgets	Contain other widgets
Extends <code>Canvas</code>	Extends <code>Composite</code>

- `Canvas` is a subclass of `Composite`.

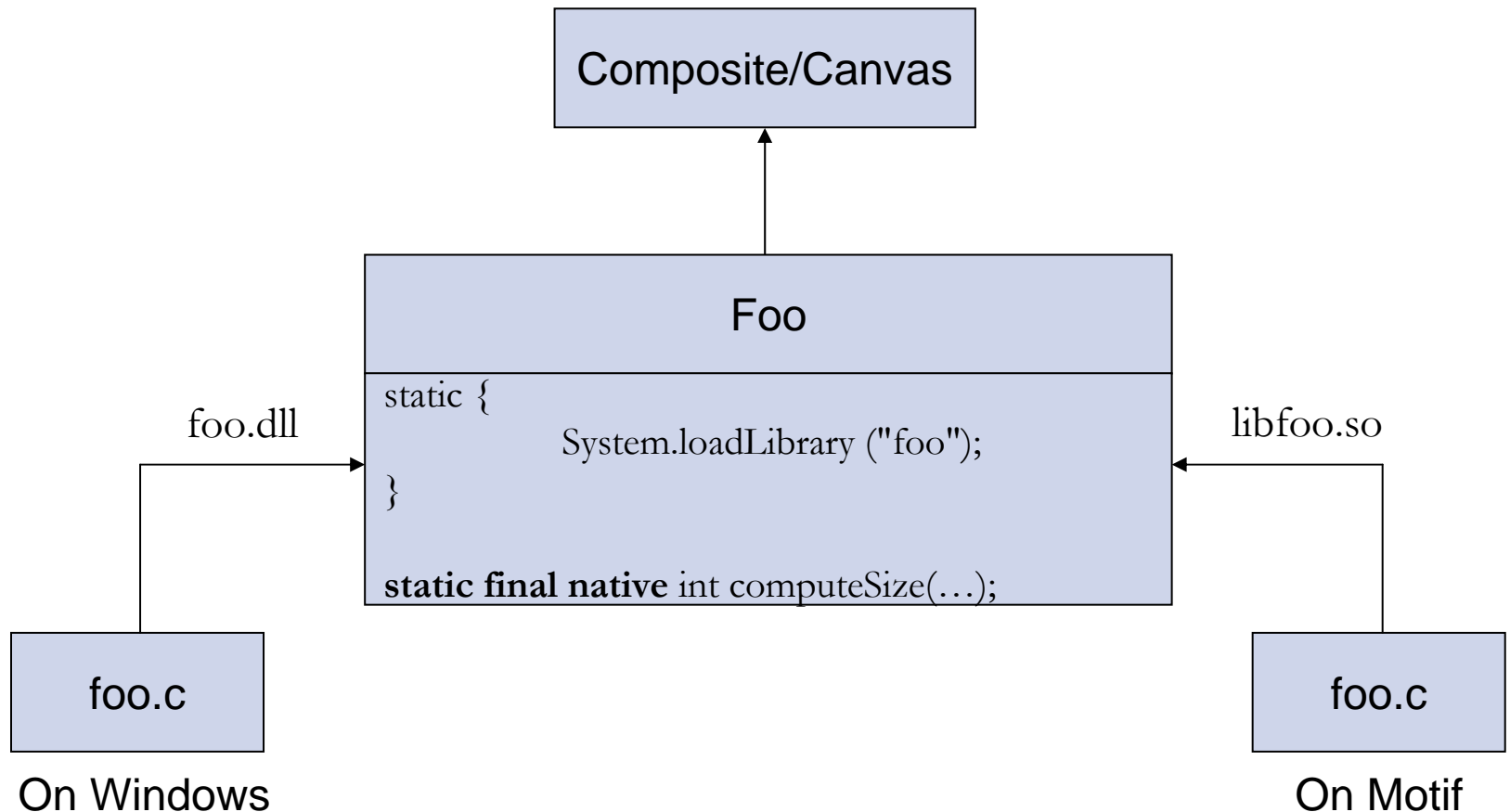
# Writing Custom Widgets

---

- Try to avoid extending a widget other than Canvas or Composite
  - Other widgets may be native
  - Other widgets may have name conflicts in future releases
- Wrapping the widget is preferable

# Wrapping Native Widgets

- Based on JNI (Java Native Interface)



# Mixing Native and Java Widgets

