

Remove Middleman

The *Remove Middleman* refactoring allows you to replace all calls to delegating methods in a class with the equivalent calls directly to the field delegated to. Additionally, you can automatically remove the classes delegating methods, which will now be unused.

This refactoring is useful if you have a class that simply forwards many of its method calls to objects of other classes, and you wish to simplify your design.

This refactoring is also available from [UML Class diagram](#).

Example

Before	After
<pre>public class Foo { Bar bar; public Foo getImpValue() { return bar.getImpValue(); } } public class Bar { private Foo impValue1; public Bar(Foo impValue) { impValue1 = impValue; } public Foo getImpValue() { return impValue1; } } public class Client { Foo a; Foo impValue = a.getImpValue(); }</pre>	<pre>public class Foo { Bar bar; public Bar getbar() { return bar; } } public class Bar { private Foo impValue1; public Bar(Foo impValue) { impValue1 = impValue; } public Foo getImpValue(){ return impValue1; } } public class Client { Foo a; Foo impValue = a.getbar().getImpValue(); }</pre>

To remove a middleman

1. Open the class in question in the editor, and position the caret at the name of the delegating field. Alternatively, select the delegating field in the Structure view of the desired class.
2. Choose **Refactor | Remove Middleman** on the main menu or on the context menu of the selection.
3. In the **Remove Middleman** dialog box:
 - Select whether you wish any methods which simply forward to calls to this field to be delegated. These methods will be unused by code in the current project, but may still be needed by code outside the project. You may wish to retain these delegating methods for backward compatibility.
 - Click **Preview** to make IntelliJ IDEA search for usages of the selected field, and display the refactoring preview results in the [Find tool window](#). In the preview, you can include usages into refactoring or skip them. Click **Do Refactor** to apply refactoring to the selected usages.
If you do not want to view usages, click **Refactor**. In this case, the usages will be changed immediately.

The Refactoring preview window may appear anyway, if the files to be affected are read-only.

See Also

Concepts:

- [Refactoring Source Code](#)

Web Resources:

- [Developer Community](#) 