When would you use the mediator design pattern

Asked 16 years ago Modified 8 months ago Viewed 10k times

17

As the title states when would you recommend the use of the mediator design pattern and where do you see it used incorrectly?

language-agnostic design-patterns mediator

Share

Improve this question

Follow

edited Dec 17, 2008 at 3:53

Blair Conrad
241k ● 25 ● 136 ● 112

asked Dec 17, 2008 at 3:46

hhafez
39.7k ● 39 ● 116 ● 146

4 Answers

Sorted by: Highest score (default)

25

Use a mediator when the **complexity of object communication** begins to **hinder object reusability**. This type of complexity often appears in view instances, though it could really be anywhere.

Misuse of a mediator can result in crippling the interfaces of the mediator's colleague classes.

It seems a little funny to talk about misusing a pattern. If your implementation follows the pattern, then you've used the pattern. Otherwise, you haven't. In other words, if your mediator is doing something else, then it probably isn't a mediator. Patterns are defined by what they do, what they in fact are. The names of things are simply labels.

The real question to ask yourself is whether your implementation of a pattern fulfills the pattern's promises for your design. The mediator pattern aims to encapsulate complex inter-object communication when it is becoming unmanageable. If it hasn't accomplished this, or hasn't done it very well, you could say that a mediator is being misused. At some point, it becomes a value judgement.

Share Improve this answer Follow

edited Dec 17, 2008 at 22:20

answered Dec 17, 2008 at 4:04

keparo

34k ● 13 ● 63 ● 66

A misuse I have encountered is to create a huge Mediator class with a lot of logic in it. – Stefan Iancu Jul 15, 2021 at 10:16

3

I have used it to deal with swing apps.

When I'm building a GUI I don't like each control knowing each other because that would require subclassing.

Instead I have a Main object which contains the listener and the widgets and let it mediate between the different controls, buttons, textfields etc.

Share Improve this answer Follow

edited Jan 14, 2019 at 18:42
StefanJM
1,614 ● 15 ● 19

answered Dec 17, 2008 at 3:59
OscarRyz
199k ● 119 ● 396 ● 573

I believe by the fact that you used the word "contains" here that this is actually not the Mediator pattern. It sounds more like the Facade pattern (or what I would call a Manager class). Mediator seems to be moreso having a shared class between 2 classes, perhaps that you pass from one to the other, or that you give a reference to to both classes, which then acts as an indirect connection between them. – Andrew May 27, 2020 at 6:34

And actually the shared class that you pass from one class to another sounds more like the Flyweight pattern lol. Idk I think maybe these definitions are too ambiguous and/or too tightly coupled to specific languages. – Andrew May 27, 2020 at 6:42

2

The mediator is also basically what an event-pump is. A very common pattern in GUI's and Games.

Also I've used mediator before to communicate among very dispar systems, and legacy frameworks.

Share Improve this answer Follow

answered Dec 17, 2008 at 4:37

Robert Gould

69.7k ● 61 ● 191 ● 275

Event-Pump == Event-Hub, right? Just a bussing system, basically? Or rather Event-Driven Mediation (routes requests)? – Cody Jan 19, 2017 at 22:39

0

When you are dealing with multiple objects and they need to communicate with each other but you want them not to know about each other directly

Share Improve this answer Follow

answered Apr 15 at 6:51

Kathiresh P

39 ● 5