


 Questions Developer Jobs Tags Users

Is this a pattern?

 0  

A naive and hard to test implementation of a method could look like this

```
public void getFromDatabaseComputeAndStore(SomeType param) {  
    Collection<Something> s = dao.retrieveSomethingFromDatabase(param);  
    Collection<Other> o = dao.retrieveOtherFromDatabase(param);  
    Result r = null;  
    <do massive computation, call other methods, figure out what r should be>  
    <based on s and o>  
    dao.store(r);  
}
```

Normally I'd refactor this into something like

```
public void getFromDatabaseComputeAndStore(SomeType param) {  
    Computer computer = new Computer();  
    Collection<Something> s = dao.retrieveFromDatabase(param);  
    Collection<Other> o = dao.retrieveOtherFromDatabase(param);  
    computer.setSomething(s);  
    computer.setOther(o);  
    computer.execute();  
    Result r = computer.getResult();  
    dao.store(r);  
}
```

where the `Computer` class is the key. This class does not interact with databases or other external systems and has no side effects, eg it's purely functional. Given the same `somethings` and `others`, the `result` will always be the same.

So my question are:


- Is this a known pattern with a name
- What would be the common name of a class with the functionality of `Computer`

I've looked at the `Strategy`, `Mediator`, and `Command` patterns, but I don't feel they fit perfectly.

[design-patterns](#)

[share](#) [edit](#) [close](#) [delete](#) [flag](#)

asked May 9 '12 at 9:44

 [slipset](#)

2,120 ● 2 ● 14 ● 16



are you writing java in Clojure?

erik assum
erik.assum@ardoq.com
@slipset
ClojureX 2018