

- [Home](#)
- [Design Patterns](#)
- [Blog](#)
- [Products](#)
- [Sitemap](#)
- [Contact](#)

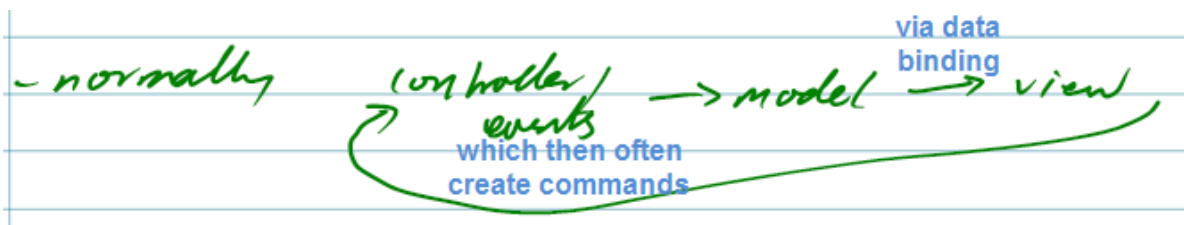
AndyPatterns



Communicating events from Model to the view - Cairngorm vs. PureMVC

The [universal mind extensions](#) to Cairngorm introduce a callback facility. The main use for this is to allow view controls to be notified of events outside of the normal model->view "data binding" mechanism. View controls can subscribe to all sorts of events via the callback mechanism thus allowing e.g. buttons to disable/enable etc.

Of course you can achieve the same effect by having an event trigger a command which changes some part of the model. A view control that is bound to that property in the model will thus get notified via this Flex data binding mechanism. The universal mind callback extensions to Cairngorm provide a more direct way of doing the same thing.



The above illustrates the normal control flow in Cairngorm.

If you are using Cairngorm and if you want to bypass the model and get to the view directly (without having to set a state on the model in order to communicate via data binding to the view) then you need to use the universal mind cairngorm callback extensions.

With PureMVC this is not necessary.

The PureMVC approach

Such callback extensions are not necessary in PureMVC because the view (via the mediator) already can generate and listen for **any event** in the pureMVC system.

In fact mediators rely on the fact that they listen to events from the model in order to update the view. This is especially true because in PureMVC there is no reliance on "data binding" to get information from the model to the view - the view (via the mediators) must manually listen for events from the model. If view mediators can listen to events from the model, then they can [Update](#) listen to events from other parts of the system.

Thus the more generic PureMVC architecture doesn't need fancy universal mind type callback extensions - it sort of comes for free.

[concrete5 - open source CMS](#) © 2020 [AndyPatterns](#). All rights reserved. [Sign In to Edit this Site](#)

