

## Re: Better architecture for cross-platform project

**Source:**

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.compactframework/2004-09/1780.htm>

---

**From:** Stefan Hong (*stefan\_at\_mosp.net*)

**Date:** 09/28/04

Date: Tue, 28 Sep 2004 13:02:14 +0800

Actually I've already put all non-UI functions in a shared CF project, and fortunately they don't use webservices so that works great. But for UI part, there is still a great deal of code that could be shared, such as the logic swapping content view in a form based on current context.

I recently came up with another idea. First making those code work in an abstract way – manipulate on pure interfaces that represent visual components in my app, then implementing those interfaces in concrete UI classes in separate projects for each platform. But turns out this is a bad idea 'cause I have to export too many things onto the interfaces that making the entire project over-designed.

The current Configuration Manager has a 'Platform' drop-down setting but does not allow us changing anything. Will it be supported in vs.net 2005?

Stefan

Robert Levy [MS] wrote:

- > *My suggestion would be to put the actual functionality of your app into a*
- > *DLL and then creating 3 seperate projects for each platform – each of which*
- > *would simply be for a UI that leverages the shared DLL.*
- >
- > *When creating the DLL, be sure to do it as a .NET Compact Framework project*
- > *so that the IDE still prevents you from using supported on the desktop but*
- > *not the device.*
- >
- > *The only complication I've ever had with this approach is with having the*
- > *shared DLL call a webservice – the code generated for device projects to do*
- > *this doesn't work on the desktop.*
- >