

Are there any good guidelines, references, design patterns or just good advice for building HTML/JavaScript Air Apps

Asked 15 years, 11 months ago Modified 15 years, 11 months ago

Viewed 136 times



1

I'm currently building a prototype AIR application for work. It allows our clients to download and work with their data and then synchronise with the server at some later time. Pretty standard stuff.



I'm an experienced web developer and so I have been fairly successful at getting this app into a reasonable state for demonstration, but in the near future I'll have to get it production ready. In preparation for this I'll need to do a bit of research on best practices for this kind of thing.

Any advice you can give would be most appreciated. I would like to hear about

Architecture

I've organised my app into a roughly MVC pattern with a rudimentary signal/slot system for inter object communication. this works quite well but I think it may begin to creak if the project were very much larger.

I have an object that communicates with the database server, another that handles the local, SQLite data. An object that handles the various views, both static and dynamic html. A controller that marshals the other objects and handles the flow and a small config object that loads, stores and handles configuration data.

Does this sound reasonable? What have other people done? Are there any good demos/tutorials or good references?

Security

I haven't really spend much time on security because we're at prototype stage but I'm all ears! I'm using CSV to move the data around at the moment but in the end it'll be AMF over HTTPS.

Distribution and updates

I'm developing in Linux. Is this going to be a problem when the app is packaged up? Is AIR on Linux as capable as on Windows or Mac? Will I be able to make a proper installer/badge? Will I be able to get the update framework working?

In general though, because I'm not this far down the line, is making the .air file as simple as it seems? Works like a charm on my machine.

I think that's enough for now unless anyone spots something I've left out.

Thanks :)

[javascript](#)[user-interface](#)[air](#)[design-patterns](#)[ria](#)[Share](#)[Improve this question](#)[Follow](#)

asked Jan 19, 2009 at 20:33

[meouw](#)

42.1k ● 11 ● 54 ● 69

1 Answer

Sorted by:

Highest score (default)



1



I have used AIR quite successfully for this sort of thing, and intend to converge with you at the MVC abstraction level. From one of my SO questions, my attention was directed to [this](#). Maybe you could look it over and let me know what you think. At a high level it seems to coincide with your description of your design.



I'm using AMF - tried starting with JSON but AMF is more direct and for my purposes functionally equivalent.

My app is used by a couple dozen people across random platforms, Windows and Mac and Linux. Installation is painless. I post one installation package, and the Adobe installer figures out what runtime stuff is needed for the platform being served, and makes sure things are set up correctly with no attention on my part. It completely abstracts out platform dependencies, and there was no learning curve.

answered Jan 19, 2009 at 20:56

Share Improve this answer

Follow



dkretz

37.6k ● 13 ● 83 ● 140
