Subject: Re: Development Diary (markcarranza)
Posted by markcarranza on Sun, 30 Dec 2012 01:28:36 GMT
View Forum Message <> Reply to Message

Again, thank you for replies.

1) If you want your new application to have the same look and feel...
Not the same, but ... a rich web UI. I'm used to (framework guy) spending 80% of development time on user interface.

There are many things I like about Radicore architecture, especially business layer + workflow, and data access layer + audit. I build highly-customized Rich Internet Applications (RIAs) with JavaScript.

I think I can replace the presentation layer, using a similar small-component philosophy, and still reap some benefits of the Radicore back-end, but it will be a risk. If you think this is an unwise risk, please let me know.

2)d. >resize, reposition
easy to fix location with css. It's my experience that webapp users prefer 'fixed locations and scrolling' rather than 'scrunching.' The interface of this forum's post form is a good example: some things reposition intelligently with resize, but most fixed.

4) With Radicore I might, for example, need to change user_IDs from VARCHAR to INT.
>no sensible reason
For the nonsensible reason that (major client) employees want to change their login name to reflect name changes. So I separate the string used for logins (login_name) from the primary key (user_id, employee_id).

For the sensible reason that users are (typically) employees, and while not all employees are users, using employee_id in 'created_user' and 'revised_user', as well as other application 'user' fields, keeps joins sane in applications with many employee/application integrations. For enterprise clients I use employee_id (technical key) as the primary key of an application user table, the login name is a separate, unique VARCHAR.

7)>The Radicore Security Model
Thank you for the link

9) I'd love to see a system allowing user-created queries and reports.
>customisable report generator which you can obtain from a third party
I haven't found a good one for HTML reporting. They typically have data dictionaries...
Suggestions welcome.

In the past I've made a number of custom print-related plug-ins, but all browser specific, IE or Mozilla. Print requirements are still TBD.