Subject: Business Process Mangement and Rules
Posted by stephenboey on Mon, 09 Apr 2007 21:32:13 GMT
View Forum Message <> Reply to Message

Just some suggestions.

Based on a Role:
Based on a Business Rule:
1. Add features that allow a screen/view to be preconfigured by developer/administrator from a form. Preferably, these properties are stored in database.
example: field visibility, field enabled/disabled.

Currently this is only achievable through scripting in the classes. This can be very tedious if too many rules.

Would be good if business rules can be factored out and stored in database as well. Similar to BPM workflows.

Subject: Re: Business Process Mangement and Rules
Posted by AJM on Mon, 09 Apr 2007 21:57:08 GMT
View Forum Message <> Reply to Message

You need to take a look at http://www.tonymarston.net/php-mysql/menuguide/mnu_role_task field(multi)a.html and http://www.tonymarston.net/php-mysql/menuguide/mnu_role_task field(multi)b.html

These tasks allow you to change the accessibility of individual fields within a form based on the user's role. By default each field is granted FULL access (read and write), but this can be changed to NOEDIT (read only) or NODISPLAY (hidden) for individual roles and tasks.

Is this what you are after?

Subject: Re: Business Process Mangement and Rules
Posted by stephenboey on Tue, 10 Apr 2007 00:03:21 GMT
View Forum Message <> Reply to Message

Based on Role (yes)
Based on Rules:
1. Choose a preconfigure Task? Meaning we have a few versions of Task (file sets) for different Rules?

additional suggestions if there is solution to factoring out of business rules: (idea borrowed from drools)
2. Allow users to define business rules in sentences. (bmp attached)

3. Meaning that, may need to allow users access/edit Petri Nets?

## File Attachments

1) User defined rules.zip, downloaded 2096 times

---

Subject: Re: Business Process Management and Rules
Posted by AJM on Tue, 10 Apr 2007 08:35:06 GMT
View Forum Message <> Reply to Message

No, you do not need different versions of the same task. You have one task, which has a single screen with a defined set of fields. If you want to have different accessibility to these fields for different Roles the old fashioned way would be to create a different task with a different screen. This would also mean that you would have to change menu/navigation buttons in order to activate the relevant task with the relevant accessibility, and this would require a great deal of effort.

The Radicore approach is much easier. There is a single Task with a single screen that is accessible to one or more Roles. By default all the users of those Roles see exactly the same screen. By using either of those two functions I mentioned previously it is possible to change the accessibility of certain fields in that screen for specified Roles. There is still a single task with a single screen, but the contents of the MNU_ROLE_TASKFIELD table is checked at runtime to see if the default accessibility of any fields in that screen need to be changed.

As for a rules engine based on the screen sample you supplied, I have no plans to implement such a thing. The Radicore framework has been designed for developers, not untrained users, so business rules are implemented by putting the correct PHP code in the correct method of the correct class. Any alternative would simply impose a great overhead for no benefit. It would also make debugging a lot harder.

The theory that it is possible to write a general-purpose rules engine that can deal with any rules in any circumstances is just that - a theory. I have seen it attempted and I have seen the failures.

The only time I have personally built a successful rules engine was for a payroll system in the 1990s in which I made it possible for all the calculations to be defined in screens and maintained on the database. This was a closed system in that all the possible inputs were known, all the possible outputs were known, and all the possible actions (add, subtract, multiply, divide) were also known.

---

Subject: Re: Business Process Management and Rules
Posted by stephenboey on Tue, 10 Apr 2007 09:24:23 GMT
View Forum Message <> Reply to Message

If we don't create another screen for a different Rule means we have to resort to coding in the classes. I don't see other way. Of course ideal case is based on Roles with no complex Rules.

Question is of course, how do developer find those code if business rules begins to pile up? Would they have to over analyse the source code in order to change certain functionality if application is huge?

Take for example (Simple):
1. When Contact status = 'Prospect'. All fields editable.
2. When Contact status = 'Customer'. All fields readonly except for phone number, fax numbers.

A more complex one
1. When a Loan status = 'Prospecting'. All fields editable. Collateral fieldNot Displayed.
2. When a Loan status = 'In Progress'. All fields editable Collateral field Displayed.
3. When a Loan status = 'Approved'. All fields read only. Loan Status field readonly. Collateral field Displayed.

---

Subject: Re: Business Process Management and Rules
Posted by AJM on Tue, 10 Apr 2007 09:55:02 GMT
View Forum Message <> Reply to Message

Such rules cannot easily be handled by the framework for the simple reason that the framework knows only about Tasks and Roles, and nothing about the state of any records within your application database.

If you have such complex rules, what is the difference between having to define them using PHP code and defining them in a rules engine? They still have to be defined, and doing it in code avoids the overhead of going through an intermediate rules engine.

If you look inside file dict_column.class.inc you will see where I turn access to different fields either ON or OFF depending on the datatype of the current column. This is just a single task with a single screen, yet the screen can change at will to suit the circumstances. I am afraid that trying to write a rules engine to do the same thing would require an enormous amount of effort, but for what benefit?

Also, the task of maintaining a rules engine which people would expect to be able to handle ANY condition in ANY set of circumstances would be just too great.

If you think that such a rules engine is possible then why not try writing it yourself?

---

Subject: Re: Business Process Management and Rules
Posted by AJM on Tue, 10 Apr 2007 10:34:13 GMT
View Forum Message <> Reply to Message

If you think that writing the same block of code in several classes is not such a good idea, then

why not try the alternative method of putting that code into a centralised function which can be called from anywhere?

---

Subject: Re: Business Process Management and Rules
Posted by stephenboey on Tue, 10 Apr 2007 11:07:59 GMT
View Forum Message <> Reply to Message

Its only a suggestion to improve radicore and to tickle your brain at the same time as well

Agree with you that in reality, new ideas do require resource and flexible thinking.

If I encounter such a situation,  perhaps might resort to a parameter table or central function suggestion at the moment.

Some Java technologies are really nice ideas, like Rules and Business Flow Engines. However, I can foresee that these guys will experience a slower application which might frustrate users who do not have a clue about what is going on at the background.

However, similar applications in php will be faster due to php's track record thus far. After all, its written in C programming languange.

---