

## Redmine - Feature #4828

### Dynamic / modular workflows.

2010-02-13 05:01 - oliver stieber

<b>Status:</b>	New	<b>Start date:</b>	2010-02-13
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>	Issues workflow	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Resolution:</b>			
<b>Description</b>			
<p>Hi,</p> <p>I tried playing around with work-flows a bit, but they didn't seem to be as flexible or reusable / modular as I would have liked them to be to be really usable.</p> <p>I would like the ability to write a work flow in a modular fashion and maybe group sets of work-flows together so that a category of work flow could be linked to the higher work flow and one of a group of modules could be picked by the user to fulfill that particular work element and not just a specific module (maybe even have a work flow to pick which work-flow in the group to use).</p> <p>It would also be nice to have input and output from the particular modules and elements of the work-flow that could then be used as part of a script for the work flow, which could in turn make the work flow more powerfull, more automated and dynamic.</p> <p>I'd also like to be able to pick a particular work flow for a particular issue not just a particular type of issues (though this could be accomplished through the modularity feature able, but extending that so that it becomes a work flow for raising a new issue in general and then having a data input into the part of the work flow in which you decide which module to pick on the group of modules being the tracker that the issue is assigned to)</p> <p>It may also be nice to customize the work-flow for a particular issue etc... in a lot of instances a nice fast way of doing this would be as part of the work-flow for raising the issue to collect details that then define how the work flow works later on. I suppose this type of work flow activity could also be completed at other steps in the work-flow or for some types of things that need to be collected at any time and at any point in the work-flow (kind of work ahead flow). On a number of occasions it would be highly lightly that only some of the information required to fully complete work-ahead types flows or work-flow control isn't going to be available at any particular point in the work-flow, so being able to collect it throughout the life of the work-flow up until it's actually needed to perform the next step in the work-flow would be nice, and then when the step where it's required to determine what to do next that work-flow activity could be to complete the task.</p> <p>I can already think of lots of variations on this theme that are too great in number for me to write down here and should really be worked out in the design / development and implementation and not by me raising the feature request.</p> <p>output data from work-flow elements or modules could also be used to populate / update or add to other fields (especially for things like my super-tags idea)</p> <p>To help clarify and show just what kind of benefits what I'm suggesting would produce here's a real world example of something that really needed a dynamic / modular work-flow, infact a dynamic / modular work-flow would be the only way to do it</p> <p>"A company that I used to work for had a number of integrations that we used to integrate with our customers, the work-flows and data that need to be collected and used for each particular integration and sub-type/configuration of integration was pretty much unique, that is there was some information that all integrations needed, some other information that determined which integration and subtype of integration that should be used, first information required to determine the type and then information required to determine the subtype. There were also many different protocols and variations on common protocols some of which were specific to a type of integration or subtype of integration and some of which could be used by all integrations and subtypes and some of which could only be used by some integrations / subtypes.</p> <p>Sometimes when we setup a client they already had a backend system, sometimes this backend system only supported one integration type/ subtype / protocol, sometimes is supported different protocols and different types of integration / subtype dependent on what protocol was being use etc....</p> <p>Sometimes the client didn't have a backend or was looking to procure a backend system or was looking for us to write something client side for them or they were going to write something client side for them etc...</p> <p>A work flow was really needed for this, customer experience of the whole process was horrific, development was very untimley (the wrong information was collected, information wasn't collected at when required (or upfront which would have been best), wrong</p>			

decisions where made, frequently we didn't even keep track of half of the information so when dealing with customer problems or changing things is was an absolute nightmare etc....

You can also see that the information required to define the work-flow required and collect the additional data could be available at any time and that it would be really nice to collect as much information as soon as possible.

It would also be nice if new types of module grouping / categorization and information collecting / auto-completion etc... could be created when we came across a new system with new capabilities."

Having thought about it, I think something like this could be a real killer feature for redmine.

p.s. I'm not working at the moment so have plenty of time on my hands so am available for testing and even help with the coding of any features I suggest. I'm not familiar with Ruby or with the redmine code base so would need a little assistance to begin with. But I've been writing/ designing and teaching people how to do software for the bast part of 25 years in a wide range of languages etc... so it shouldn't take me very long to get up to speed.

---

**Related issues:**

Related to Redmine - Feature #973: Assign different status sets and workflows...	<b>New</b>	<b>2008-04-02</b>
--	------------	-------------------

---

**History****#1 - 2010-04-27 18:50 - Daniel Miller**

This feature-request appears to be in actuality multiple feature-requests (that perhaps should be tracked separately). For example,

1. One feature could be a static (for all issues) modular work-flow.
2. Another feature could be to choose one of several predefined static work-flows on a per-issue basis.
3. Yet another separate feature could be to extend these 2 concepts on a per-issue basis, where a base work-flow can be adjusted (additions or subtractions) per issue by a privileged role.

That third one could get rather complicated due to, in its most extreme form, an editor of an arbitrarily-complex graph or state-machine or petri net on a per-issue basis.

I do agree with the overall sentiment that the mental model currently in place in Redmine's work-flows is too simplistic to compete with DevTrack's rich work-flow feature-set.

**#2 - 2010-04-27 19:34 - Daniel Miller**

The multiple static work-flows in this feature-request might be a duplicate of feature-request [#973](#).

**#3 - 2011-03-23 11:23 - Toshi MARUYAMA**

- *Category set to Issues workflow*