

## Redmine - Patch #20384

### Proposal: Workflow enhancement

2015-07-21 15:20 - Frederico Camara

<b>Status:</b>	New	<b>Start date:</b>	
<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>Description</b>			
<p>I work at a large ICT public company in Brazil and we are using Redmine here.</p> <p>At first it didn't become apparent that the workflow tools in Redmine would be limiting in any way. But as we start to build more complex workflows and reutilize roles and tracker names for different projects and teams, we noticed that a workflow being inherent to a specific role and tracker, was making us come with similar names and synonyms to roles and trackers so we could develop new workflows for a different areas in our company. Examples of this are: using lower and uppercase in names, adding numbers, using words in both genders (substantives in portuguese have gender inflexion). Ultimately it became an annoyance, as similar entries would populate various menus across Redmine.</p> <p>Also, the closed state being inherent to a issue status, also made we duplicate entries, baccuse in a workflow, some state would close the issue whereas in another workflow, the same state would not.</p> <p>I still think workflows should still be inherent to roles and trackers, but projects could have diferent workflow sets to choose from, so that you could still have different projects using the same workflow and at the same time could have different projects using different workflows.</p> <p>Also, the closed state should be inherent to the workflow being used.</p> <p>For a first proposal, I would have another entry in the admin menu for a workspace, workflow set, or workflow space, or whatever the best term is in english. From there, workflows would be part of a workspace and in each project you would select the workspace to use. For migration purposes, all existing and new workflows and projects would be assigned to the default workspace. New subprojects should have selected the same workspace as the parent at creation. There is no inheritance of workspace between projects.</p> <p>The main advantages are:</p> <ul style="list-style-type: none"><li>• fewer roles created;</li><li>• fewer issue statuses created;</li><li>• simpler migration between workflows;</li><li>• more control over complex infrastructure;</li><li>• very low impact on simpler infrastructure.</li></ul>			

### History

#### #1 - 2016-03-15 23:48 - Frederico Camara

Implemented it as 'Workspaces', (except for the closed state) on my git repository with a few improvements, and it is awesome!

Now I can have different workflows assigned to the same tracker and role names, in different projects.

- In Administration, you can add workspaces in 'Workspaces'.
- In Administration > Workflow, there is a new filter 'Workspace'.
- Changing the workspace switches between different sets of workflows (also working if the Field Permission tabs).
- Summary has a dropdown menu to switch between different workspaces.
- On Project > Settings, you can change the workspace the project is using.
- You can't delete the default workspace (also, when migrating, this is where all previous workflows go to).
- Patched against Redmine 3.2 stable.

How to get it:

```
git clone -b "3.2-patched" https://github.com/fredsdc/redmine <target directory name>
```

#### #2 - 2017-03-13 19:27 - Frederico Camara

- File *workspaces.patch* added

I cherry-picked the commits and made the necessary adjustments against redmine 3.3 stable. Here's the patch.

It adds "Workspaces" in the Administration menu and migrates all existing workflows to the "default" Workspace.

For each project, you can select what workspace it will use. Workspaces separate workflows (transitions and permission) from each other so that different projects can have different workflows. It also tries to help administration.

This patch was made on Linux, for Windows you have to fix newline chars.

## Changes:

- In Administration, you can add workspaces in 'Workspaces'.
- In Administration > Workflow, there is a new filter 'Workspace'.
  - Changing the workspace switches between different sets of workflows.
  - This is also working if the Field Permission tabs.
- Administration > Workflow > Summary has a dropdown menu to switch between views from different workspaces.
- Administration > Projects shows which workspace each project is using.
- On each project > Settings, you can change the workspace the project is using.
  - On Information, Trackers are filtered by which trackers have workflow on the workspace
    - You can show/hide filtered trackers clicking on "all"
  - On Members, Roles are filtered by which roles have workflow on the workspace and those that can't have issues assigned to them (on Administration > Roles)
    - You can show/hide filtered roles clicking on "all"
- You can't delete the default workspace (also, when migrating, this is where all previous workflows go to).

## To patch (redmine 3.3-stable):

```
patch -p1 < workspaces.patch
bundle exec rake db:migrate
touch tmp/restart.txt
```

### #3 - 2017-03-13 19:52 - Federico Camara

Could someone change this Feature to Patch?

### #4 - 2017-03-14 02:05 - Jan from Planio [www.plan.io](http://www.plan.io)

- Tracker changed from Feature to Patch

It's an interesting (albeit far-reaching and complex) patch, thank you for proposing it. In order for a patch of this magnitude to be considered though, it would need full test coverage and it shouldn't break any existing tests. If possible, I'd also recommend to break it down into a series of patches in which each patch results in a working version of Redmine with all tests passing, but separating your work into separate chunks that can stand by themselves.

From our work with Redmine in very large organisations at [Planio](http://Planio) the main requirement was for project teams to create their own trackers, statuses and workflows without the need of an admin and without interfering with other projects. Your patch seems to solve the latter challenge, but not the former one. With your solution, project teams requiring different workflows would still have to ask the Redmine Admin to create them for them, am I correct?

Especially in large organisations, this creates a real challenge because admins wear many hats and administer many different systems. They may not be 100% familiar with the configuration interfaces in the Admin area of Redmine and how they correspond to the business needs of a project manager requesting a certain workflow change. This creates potential for misunderstandings when project managers are requesting a different set of workflows via email or on the phone. Sitting down with the admin to create them is often hard to realize and can take weeks for a project to get their required workflows implemented.

I think that a solution where certain roles can manage trackers, statuses, and workflows for their projects individually would be a better suited solution to the challenge we've seen with many large organisations.

### #5 - 2017-03-14 15:18 - Federico Camara

Jan from Planio [www.plan.io](http://www.plan.io) wrote:

It's an interesting (albeit far-reaching and complex) patch, thank you for proposing it. In order for a patch of this magnitude to be considered though, it would need full test coverage and it shouldn't break any existing tests. If possible, I'd also recommend to break it down into a series of patches in which each patch results in a working version of Redmine with all tests passing, but separating your work into separate chunks that can stand by themselves.

I should prepare a testing environment :-). The patch is the result of four commits on my github redmine tree, the main one changes about 30 files and is hard to break down, as soon as I add workflows to different workspaces, the test should break somewhere

From our work with Redmine in very large organisations at [Planio](http://Planio) the main requirement was for project teams to create their own trackers, statuses and workflows without the need of an admin and without interfering with other projects. Your patch seems to solve the latter challenge, but not the former one. With your solution, project teams requiring different workflows would still have to ask the Redmine Admin to create them

for them, am I correct?

Yes, at least in this present state. We were already numbering roles and trackers to make different workflows, already not knowing which workflows were used in which projects, and querying the database. I started by trying to solve that problem.

Especially in large organisations, this creates a real challenge because admins wear many hats and administer many different systems. They may not be 100% familiar with the configuration interfaces in the Admin area of Redmine and how they correspond to the business needs of a project manager requesting a certain workflow change. This creates potential for misunderstandings when project managers are requesting a different set of workflows via email or on the phone. Sitting down with the admin to create them is often hard to realize and can take weeks for a project to get their required workflows implemented.

I think that a solution where certain roles can manage trackers, statuses, and workflows for their projects individually would be a better suited solution to the challenge we've seen with many large organisations.

You have to isolate these resources before empowering roles, or it could get out of control fast, for the Admins.

Workspaces could be extended to isolate Trackers, Issue statuses and Custom fields. You can extend roles so they could manage everything inside a Workspace. On top of that, the whole thing should be manageable by the admins, so the whole information has to be broken in manageable bits.

#### #6 - 2019-11-21 13:21 - Federico Camara

- File 0004-Add-workspace-functionality-to-reuse-roles-and-track.patch added

- File 0005-Hide-trackers-and-roles-by-workspace-workflow-color-.patch added

Patch for Redmine 4.0

#### #7 - 2022-11-07 17:48 - Federico Camara

- File redmine\_5.0\_patches.zip added

For Redmine 5.0, I zipped my many patches adding this functionality. In the zip, there is a big .patch file with everything together at once, and separated patches 0001 through 0011 for each incremental functionality/fix.

#### #8 - 2022-11-09 11:59 - thuruk thuruk

Hi Federico !

~~May I suggest making a branch of 5.0 stable on your public git ?~~

~~I'd like to use this via Docker, and having a public git is much cleaner for that. I'd do it myself but I wouldn't want to steal this work :P-~~

Thanks !

Edit : Of course, if the format of a plugin is possible for such deep changes, that would be even greater, as I don't think this is gonna get patched into master anytime soon...

Edit again : nvm, found your branch... Thanks !

#### #9 - 2024-01-25 03:24 - Jet Yan

Hi Federico Camara,

The latest redmine revision is now 5.1.1.

It is cheerful if you can get a chance to update your workspace patches to support the latest 5.1.1 revision. Thank you !

I just installed your patches but found not work correctly.

found 2 failures during the patching:

logs: =====

C:\Sites\redmine-5.1.1>"C:\Program Files (x86)\GnuWin32\bin\patch.exe" p1<-workspaces\_redmine\_5.0\_all.patch 2>conflict.txt

patching file app/controllers/admin\_controller.rb

Hunk #1 succeeded at 44 with fuzz 2 (offset 4 lines).

patching file app/controllers/workflows\_controller.rb

Hunk #7 FAILED at 171.

1 out of 7 hunks FAILED - saving rejects to file app/controllers/workflows\_controller.rb.rej

patching file app/controllers/workspaces\_controller.rb

patching file app/helpers/members\_helper.rb

Hunk #1 succeeded at 48 with fuzz 1.

patching file app/helpers/projects\_helper.rb

patching file app/helpers/workflows\_helper.rb

patching file app/models/issue.rb

Hunk #1 succeeded at 693 (offset 7 lines).

Hunk #2 succeeded at 1076 with fuzz 2 (offset 7 lines).

patching file app/models/issue\_status.rb

Hunk #1 succeeded at 56 (offset 2 lines).

patching file app/models/project.rb

Hunk #1 succeeded at 41 with fuzz 2 (offset 4 lines).  
Hunk #2 succeeded at 368 (offset 3 lines).  
Hunk #3 succeeded at 491 (offset 7 lines).  
Hunk #4 succeeded at 857 (offset 8 lines).  
patching file app/models/role.rb  
Hunk #1 succeeded at 276 (offset 7 lines).  
patching file app/models/tracker.rb  
Hunk #1 succeeded at 143 (offset 3 lines).  
patching file app/models/workflow\_permission.rb  
patching file app/models/workflow\_rule.rb  
patching file app/models/workflow\_transition.rb  
patching file app/models/workspace.rb  
patching file app/views/admin/projects.html.erb  
Hunk #1 FAILED at 23.  
Hunk #2 FAILED at 32.  
2 out of 2 hunks FAILED-- saving rejects to file app/views/admin/projects.html.erb.rej  
patching file app/views/members/\_edit.html.erb  
patching file app/views/members/\_new\_form.html.erb  
patching file app/views/principal\_memberships/\_edit.html.erb  
patching file app/views/principal\_memberships/\_new\_form.html.erb  
patching file app/views/projects/\_form.html.erb  
patching file app/views/projects/index.api.rsb  
Hunk #1 succeeded at 8 with fuzz 1 (offset 1 line).  
patching file app/views/projects/settings/\_issues.html.erb  
patching file app/views/projects/show.api.rsb  
patching file app/views/projects/show.html.erb  
patching file app/views/workflows/\_index\_table.html.erb  
patching file app/views/workflows/copy.html.erb  
patching file app/views/workflows/edit.html.erb  
patching file app/views/workflows/index.html.erb  
patching file app/views/workflows/permissions.html.erb  
patching file app/views/workspaces/\_form.html.erb  
patching file app/views/workspaces/edit.html.erb  
patching file app/views/workspaces/index.api.rsb  
patching file app/views/workspaces/index.html.erb  
patching file app/views/workspaces/new.html.erb  
patching file config/locales/en.yml  
Hunk #1 succeeded at 646 (offset 9 lines).  
Hunk #2 succeeded at 1006 (offset 10 lines).  
patching file config/locales/pt-BR.yml  
Hunk #2 succeeded at 876 (offset -5 lines).  
patching file config/routes.rb  
Hunk #1 succeeded at 381 (offset 7 lines).  
patching file db/migrate/20160314174310\_add\_workspace\_to\_projects.rb  
patching file db/migrate/20160314174311\_add\_workspace\_to\_workflows.rb  
patching file db/migrate/20160314174312\_create\_workspaces.rb  
patching file lib/redmine/preparation.rb  
patching file public/stylesheets/application.css  
Hunk #1 succeeded at 673 (offset 3 lines).  
Hunk #2 succeeded at 999 (offset -3 lines). =====

#### #10 - 2024-01-25 19:18 - Frederico Camara

- File 0001-Implements-Workspaces-in-redmine-5.1.patch added

Jet Yan wrote in [#note-9](#):

The latest redmine revision is now 5.1.1.  
found 2 failures during the patching:

Thank you for reporting.

I updated the patch for Redmine 5.1 Stable from git (probably it's the same as 5.1.1).

Admin > Projects changed a little, some things went to a ProjectQuery model that didn't exist.

You can use git am now:

```
git am 0001-Implements-Workspaces-in-redmine-5.1.patch
```

Best regards

#### #11 - 2024-01-26 03:46 - Jet Yan

Thank you Frederico Camara.  
It did the job!

Files

workspaces.patch	51.9 KB	2017-03-13	Frederico Camara
0005-Hide-trackers-and-roles-by-workspace-workflow-color-.patch	8.83 KB	2019-11-21	Frederico Camara
0004-Add-workspace-functionality-to-reuse-roles-and-track.patch	52.9 KB	2019-11-21	Frederico Camara
redmine_5.0_patches.zip	39.3 KB	2022-11-07	Frederico Camara
0001-Implements-Workspaces-in-redmine-5.1.patch	65.6 KB	2024-01-25	Frederico Camara