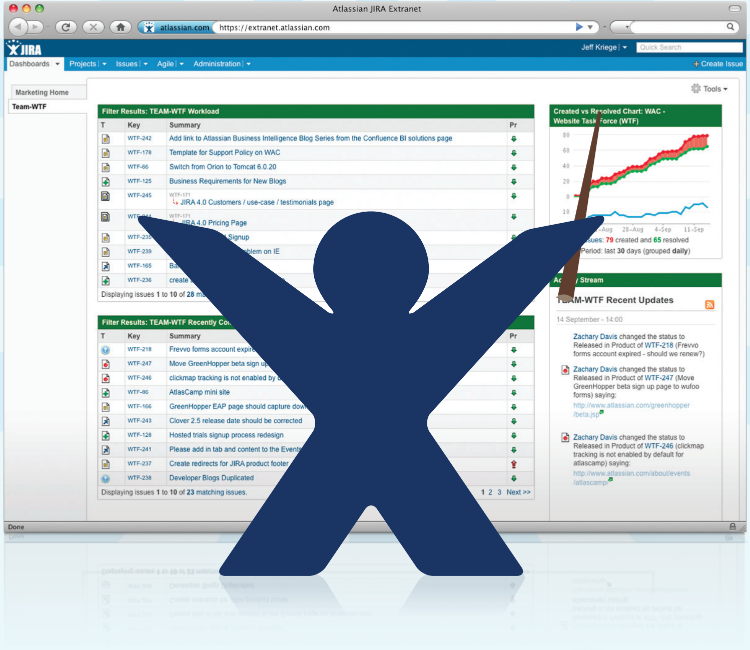
JIRA Workflows
GUIDEBOOK



Atlassian JIRA Extranet

atlassian.com https://extranet.atlassian.com

Filter Results: TEAM-WTF Workload

T	Key	Summary	Pr
	WTF-242	Add link to Atlassian Business Intelligence Blog Series from the Confluence BI solutions page	↓
	WTF-178	Template for Support Policy on WAC	↓
	WTF-66	Switch from Orion to Tomcat 6.0.20	↓
	WTF-125	Business Requirements for New Blogs	↓
	WTF-245	WTF-171 JIRA 4.0 Customers / use-case / testimonials page	↓
	WTF-244	WTF-171 JIRA 4.0 Pricing Page	↓
	WTF-235	Signup	↓
	WTF-239	blem on IE	
	WTF-165	Ba	
	WTF-236	create	

Displaying issues 1 to 10 of 28 ma

Filter Results: TEAM-WTF Recently Co

T	Key	Summary	Pr
	WTF-218	Frevvo forms account expire	↓
	WTF-247	Move GreenHopper beta sign up	↓
	WTF-246	clickmap tracking is not enabled by	↓
	WTF-86	AtlasCamp mini site	↓
	WTF-166	GreenHopper EAP page should capture dow	↓
	WTF-243	Clover 2.5 release date should be corrected	↓
	WTF-128	Hosted trials signup process redesign	↓
	WTF-241	Please add in tab and content to the Event	↓
	WTF-237	Create redirects for JIRA product footer	↑
	WTF-238	Developer Blogs Duplicated	↓

Displaying issues 1 to 10 of 23 matching issues. 1 2 3 Next >>

Created vs Resolved Chart: WAC - Website Task Force (WTF)

Issues: 79 created and 65 resolved
Period: last 30 days (grouped daily)

Activity Stream

TEAM-WTF Recent Updates

14 September - 14:00

Zachary Davis changed the status to Released in Product of WTF-218 (Frevvo forms account expired - should we renew?)

Zachary Davis changed the status to Released in Product of WTF-247 (Move GreenHopper beta sign up page to wufoo forms) saying: http://www.atlassian.com/greenhopper/beta.jsp

Zachary Davis changed the status to Released in Product of WTF-246 (clickmap tracking is not enabled by default for atlascamp) saying: http://www.atlassian.com/about/events/atlascamp/

Done

Table Of Contents

Workflow Guide

This is a guide to help you with JIRA workflows. Workflows interact with many other parts of administration, including fields, screens, statuses, and resolutions. Should you want to modify any of these pieces, you will want to do so in a specific order to prevent yourself from having to back-track later. Anything that you do not wish to modify, where you would rather just use the defaults, you may just glance at them to double-check that they are appropriate to your project, without having to make modifications. For best results, start at Step 1, even if the customization you're looking for is from a middle step.

The first page of this guide contains a Master List of tasks. This Master List contains the navigation details to get to any piece. Refer back to the Master List from any of the Examples. Everything is done from the JIRA ADMINISTRATION tab, and you will need to be a JIRA Administrator to make any changes. If you cannot find the link you are looking for, try ctrl+f (PC) or cmd+f (Mac) to do a page search. Type in the term you are looking for and your browser will highlight it for you.

A Note About Plugins

There are a few references to plugins in this guide—there are hundreds of JIRA plugins, including many that can extend the built-in functionality of workflows. Some were created by Atlassian, others by third parties. Most are free but unsupported—this means that if the plugin is causing problems in your JIRA instance and you come to Atlassian Technical Support, we may not be able to fix the problem without removing the plugin.

Master List

1. Add or modify Issue Settings > Issue Types

2. Add or modify Issue Settings > Issue Types > Issue Types Scheme (found at the top of the page when you view Issue Types in Step 1, this will be to the right of 'Global Issue Types')

3. Add any Issue Fields > Custom Fields

4. Add or modify any views (when your users create, view, edit, or transition through the workflow from one step to another) from Issue Fields > Screens

5. Add or modify Issue Fields > Screen Schemes (this is where you'll apply the new screens you just created to issue operations like Create, Edit, View—you will apply your transition views, such as Close Issue, later)

6. Add or modify Issue Fields > Field Configuration to determine which fields are optional vs required

7. Add or modify Issue Fields > Field Configuration Schemes to determine which issue types will use which field configurations of the ones you have just created

8. Add or modify Issue Fields > Issue Types Screen Scheme to determine which issue types will use which screen schemes

9. Projects > pick your project > Next to each Scheme: Issue Type Scheme, Field Configuration Scheme, Issue Type Screen Scheme: hit 'Select' and choose your new one

10. Add any new Issue Settings > Statuses

11. Add any new Issue Settings > Resolutions

12. Global Settings > Workflows > 'Add new workflow' (at the bottom)

13. Click 'Steps' for your new workflow (to the very right of the screen)

14. Add New Steps (at the bottom)

15. Next to each new step, click Add Transition (to the right). Note you will always them as outgoing transitions, then pick the new step they will go into. The Transition View (screens) you pick is from the views you created for Close Issue, Start Progress, Send for Approval, etc.

16. Optional: edit workflow in XML to use any common transitions. Note that this is not trivial, but you can use the default workflow as an example.

17. Go to Workflows > click Steps on the right > Click on each transition to add Conditions, Validators and Post-Functions.

18. Go to Schemes > Workflow Schemes and Add new Workflow Scheme

19. Go to Projects > pick your project > next to Workflow Scheme click 'select' and pick the one you just created to associate Workflow Scheme with Project

20. Test!

EXAMPLE:

From Scratch Workflow—Blog Entry

1. Add Project = Website

2. Add Component = Blogs

3. Add Issue Type = Blog Entry Add Issue Type Scheme

4. Add Custom Field type = select list name = 'Blog Category'. Hit Configure to the right. Hit 'Edit Options' to the right. Add options of News, Developer, etc.

5. Add Custom Field type = Date Picker name = 'Target Publish Date'

6. Add New Screen 'Blog Entry' with fields: Blog Category, Target Publish Date, Summary, and any others (attachment, assignee, etc)

7. Add New Screen Scheme 'Website Screen Scheme' and for issue type 'Blog Entry' use the 'Blog Entry' screen

8. Add New Field Configuration = Blog Field Configuration where 'Blog Category' field is required, 'Target Publish Date' field is optional

9. Add New Field Configuration Scheme = Website Field Configuration Scheme where 'Blog Entry' issue type uses 'Blog Field Configuration' and 'all unmapped issue types' use Default Field Configuration

10. Edit default Issue Type Screen Scheme so Issue Type Blog Entry uses Website Screen Scheme

11. Associate new schemes with Website Project

12. Go to Statuses and add new Statuses: Draft Waiting for Review, Scheduled, Published

13. Go to Resolutions and add new Resolutions: Published and Won't Publish

14. Go to Workflows and add new workflow 'Blog Entry Workflow'

15. Add New Steps In Progress, Draft Waiting for Review, Publish Date Scheduled, Published, Close Idea—Not Published

16. Go to Open step, hit 'Add Transition' called 'Start Progress' going to In Progress (view—Blog Entry Screen), then add another Transition called 'Close Idea' and select 'Resolve Issue Screen'

17. Go to In Progress step, hit 'Add Transition' called 'Draft Complete' going to Draft Waiting for Review step (view—Blog Entry Screen), then add another Transition called 'Close Idea' and select 'Resolve Issue Screen'

18. Go to Draft Waiting for Review step, hit ‘Add Transition’ called ‘Schedule Blog’ going to Publish Date Scheduled (view—Blog Entry Screen), then add another Transition called ‘Close Idea’ and select ‘Resolve Issue Screen’

19. Go to Publish Date Scheduled step, hit ‘Add Transition’ called ‘Publish’ going to Publish (view—Blog Entry Screen), then add another Transition called ‘Send Back Draft’ and select ‘Blog Entry Screen’

20. Go to Workflow Schemes and add a new Workflow Scheme called Website Workflow Scheme. For Issue Type Blog Entry select ‘Blog Entry Workflow’ and for all unassigned issue types select default workflow (jira).

21. Go to Projects > Website Project and select your new Workflow Scheme

22. Create a test issue, and click through the workflow to check to see that your steps and views are mapped appropriately.

23. If anything doesn’t work properly, you’ll need to create a draft of your workflow and/or make any schemes inactive to edit them.

EXAMPLE:

Add a Step (With a Condition)—Needs to be an Inactive Workflow

1. Create your workflow by copying the default. We're only adding one step, so this is easier than creating a new workflow from scratch.
2. You'll want to add a 'waiting for approval' step, and add transitions going into and out of that step. You'll also add a 'deny' transition going out of the 'waiting for approval' step that will send it back to In Progress.
3. We'll add a Condition to the 'approve' and 'deny' transitions that User is in Group and select the group Managers. This means that first we'll need to create that group and put users in it.
4. Add new group from Administration > Users, Groups & Roles > Group Browser called Managers, add users to this group
5. Add new custom field: Reason for Denial
6. Add new screen: Deny Screen
7. Add new statuses: Waiting for Approval, Approved
8. Go to Administration > Workflows > Copy default workflow
9. Click Steps next to new workflow
10. Add new Step called 'Waiting for Approval' and give it linked status of 'waiting for approval'
11. Edit the Resolved Step and name it Approved and give it linked status of Approved
12. Edit the existing Resolve Issue transitions going out of In Progress and call them Send for Approval going into the Waiting for Approval step
13. Add new transition going out of Waiting for Approval called Approve into Approved
14. Add new transition going out of Waiting for Approval called Deny into In Progress; view = Deny Screen
15. Add a Condition to the Approve and Deny transitions: User is in Group = Managers
16. Take a look at your Transitions. They have small numbers in parentheses next to them. That is the 'transition ID' in the database. Some of them will be the same, but some will be unique because you have modified the workflow. Note that any transition with a unique transition ID can have different Conditions, Validators, Post-Functions, and View from other transitions.

EXAMPLE:

Add a Custom Field to Transition Screen—Can Do on an Active Workflow

1. Go to Administration > Workflows and click on the appropriate workflow

2. Click Steps

3. Click the transition you want to modify the view for

4. Top left box shows Transition View: Screen (click on that link)

5. This takes you to the screen you'll want to modify. Add the field(s), move them around, etc

6. If you want to make the field required vs optional:

7. Go to Administration > Projects > Click appropriate project

8. Next to Field Configuration Scheme click Edit

9. For appropriate Issue Type, click link to Field Configuration (in the middle)

10. For appropriate field, click Required

11. Repeat for other Issue Types if necessary, or add a new Field Configuration for other issue types

12. Note that the Field Configuration for the Project means that a field will either ALWAYS be required or ALWAYS be option, no matter where the issue is in the workflow, or what screen your users are viewing. If you make a field required but don't put it on the screen, your users will see an error

13. If you want to make a field required sometimes but optional other times:

14. This is not possible in JIRA as it ships. However, there is a workaround:

15. You would need to use a Validator. Essentially, what you'll do is make a Validator for certain transitions that a field must be filled out—your users won't see that it is Required (the little * next to other Required fields) but if they do not give the field a value screen will show them they have to fill in the field.

16. There is a third party plugin that has this Validator. It is free, and has TONS of other useful workflow customizations (Conditions, Validators, and Post-Functions) but is unsupported by Atlassian—that means if it's causing problems in your JIRA and you come to technical support, we'll tell you to remove it before we can help.

17. JIRA Suite Utilities! Check it out here: https://plugins.atlassian.com/plugin/details/5048

18. If you want to add a field to the 'edit issue' screen:

19. Create a new screen with the desired fields if you will not be using an existing screen—note that you can Copy an existing screen, then modify it by just adding the new field desired

20. Go to Administration > Projects > Click appropriate project

21. Next to Issue Type Screen Scheme click Edit

22. For appropriate Issue Type click link to Screen Scheme (in the middle)

23. Add new Issue Operation from the select list: Edit Issue and choose the new screen

EXAMPLE:

Change Order and Number of Transitions Displayed—JIRA 1+

1. Note that this is a good way to 'suggest' certain transitions to your users with actually restricting transitions

2. Change number shown:

3. Shutdown JIRA: Go to atlassian-jira-standalone/bin/shutdown.sh

4. Go to atlassian-jira-standalone/WEB-INF/classes and edit 'jira-application.properties' file in a text editor

5. Uncomment the property 'ops.bar.group.size.opsbar-transitions' and change its value to the required number of transition buttons required before the 'Workflow' button.

6. Save the file.

7. Restart JIRA: Go to atlassian-jira-standalone/bin/startup.sh

8. Change order of transitions:

9. Copy the workflow or make a draft

10. Click Steps then click on the Transition you want to change the order of: for example, go to In Progress Step then Resolve Issue Transition

11. Click 'view properties of this transition' in box on top left

12. Delete current 'opsbar-sequence' property and give it a new one: 'opsbar-sequence' and value = 0 (or 1, 2, 3, etc)

13. Repeat for any other transitions

14. Associate your new workflow with a workflow scheme and associate that with your project, or publish the draft workflow

EXAMPLE:

Subtasks With A Diff Workflow

1. Enable Subtasks by going to Administration > Global Settings > Sub-Tasks
2. Create your new workflow for subtasks (follow Master List or another guide)
3. You will need to create a Workflow Scheme if you do not already have one (there isn't one in the system by default). Go to Schemes > Workflow Schemes to create a new one.
4. If you already have a Workflow Scheme for this project, copy it so that you can make changes.
5. Next to your new, inactive Workflow Scheme click 'Workflows' (on the left)
6. Click 'assign a new workflow to an issue type for this scheme' in box on top right
7. Pick Issue Type = Subtask, Workflow = new subtask workflow you created, click Add
8. Associate this modified Workflow Scheme to appropriate project from Administration > Projects > your project.

EXAMPLE:

Notify a Particular User on a Specific Transition, But Not on Just ‘Update Issue’

1. Example: Notify Bob if a Subtask is Reopened
2. We are going to create a Custom Event for this.
3. Go to Administration > Events and at the bottom add a new Custom Event
4. Call this Subtask Reopened and give it a template (this will determine what email notifications are sent by default. Pick Generic).
5. Go to Administration > Projects > Click appropriate project
6. Next to Notification Scheme click Edit
7. Your new event is at the bottom, Subtask Reopened. Next to it hit Add.
8. Pick ‘Subtask Reopened’ event on the right and Pick notify Single User = Bob on the left
9. Go to the workflow for the Subtasks in this project—you will need to make a draft of it to make changes
10. Click Steps and click the Reopen Issue Transition
11. Click Post-Functions
12. The last Post-Function will be ‘Fire Issue Resolved Event’ by default. Click Edit.
13. Select New Event ‘Subtask Reopened’
14. Publish Workflow Draft or
15. Associate new workflow with appropriate by creating a Workflow Scheme from Administration > Schemes > Workflow Schemes, then apply it to your Project from Administration > Projects > pick your project > Workflow Schemes > select.

EXAMPLE:

Email Every User Who is Reporter, Assignee or Commented on an Issue

1. This will require a plugin: JIRA Toolkit, which was created by Atlassian but is officially Unsupported. If you would like to set up a custom event to happen on a specific transition, see previous Example first (Notify on a specific transition, but not just on 'update issue'). Add your Custom Event to any transitions first, before following these steps.

2. The JIRA Toolkit plugin contains a custom field called Participants Field. You may have seen this field on jira.atlassian.com.

3. Download and install the plugin from https://plugins.atlassian.com/plugin/details/5142

4. First you will need to add this field to your issues. Go to Administration > Custom Fields > Add New Custom Field. Pick the custom field type of 'Participants' and name your field (you can call it Participants).

5. Apply that field to whichever Projects you wish by going to Screens and adding that field.

6. Now we'll set up the email notification part. Go to Administration > Projects > pick your project > Notification Scheme > Edit.

7. For any event, click 'Add' to the right.

8. Pick to notify User is in Custom Field = Participants.