

# Receiving a digest of changes and comments from Jira issues

**Problem:** How to create a notification configuration that would send out digest of changes and comments in issues happened since last run of this notification?

Here's how you approach it in Raley:

1) Create a new Notification configuration of type "Jira issues digested messaging".

2) Set the value to **Fired on** input. This should be a CRON expression that represents when the notification configuration should be run. Use [www.cronmaker.com](https://www.cronmaker.com) to generate it for you.

For example, the value "0 0 9 1/1 \* ? \*" means that the notification configuration will be run at 9AM every day. When you enter your cron, Raley will show you when it will be run the next time as follows:

Fired on  for timezone UTC    Next run: 21-12-2023 09:00:00

Pay attention that all the times are evaluated against UTC time zone.

3) Now you specify a JQL rule that will be used as a filter for issues. Only those which match against it will be included into the digest. Use Jira issue navigator to make the JQL rule easier.

4) Specify subject of the email.

5) Specify recipients using TO, CC and BCC fields. You can use a mix of values picked from combo-box'es and free-text expressions.

Recipients will be calculated as following:

- a) For given list of issues retrieved according to JQL, resolve all recipients emails based on recipients codes specified in TO, CC, BCC combos.
- b) In addition to recipients found in a), add recipients defined in free-form text/expression for TO, CC and BCC textboxes to each issue
- c) Create sending groups: each containing TO, CC, and BCC email addresses AND list of issues to be sent with that group
- d) If Send As = "Single Email", then sent out those groups, each group at a time
- e) If Send As = "Multiple Email", then split each sending group into small groups where each subgroup has exactly 1 email address (TO, CC or BCC) and the same set of issues as parent group. Send those subgroups, 1 group at a time.

6) Finally, specify a message template. You can use [our Template Wizard](#) to simplify generation of the template, just make sure you use Table layout so, issues will be shown as HTML table.

Here's a little example on how to render both, changes made and comments added since the last run of this notification configuration.

```
#foreach ($issue in $issues)
  #foreach ($history in $issue.changelog.histories)
    #if ($jirassimo.isAfter($history.created, $context.last_run))
      #foreach ($item in $history.items)
        $item.field was changed from $item.fromString to $item.toString
      #end
    #end
  #end

  #foreach ($comment in $issue.fields.comment.comments)
    #if ($jirassimo.isAfter($comment.created, $context.last_run))
      Comment by $comment.author.name: $comment.body
    #end
  #end
#end
```

You probably noticed the usage of `$context.last_run` variable. This variable contains the datetime of last invocation of Scheduled notification configuration, no matter whether there were any emails sent with that invocation or not.

The value in `$context.last_run` is stored in format `yyyy-MM-dd HH:mm` and you use `$jirassimo.isAfter()` helper function to understand if a date in argument 1 is after the date in argument 2.

There are more variables available in `$context`. To learn about them take a look at [Context variables](#) wiki page. To learn more about functions available in `$jrassimo` check out [this page](#)