The Trac Ticket System

Error: Macro TracGuideToc(None) failed

```
'NoneType' object has no attribute 'find'
```

The Trac ticket system provides a simple but effective way to track issues and software bugs within a project.

As the central project management element of Trac, tickets can be used for **project tasks**, **feature requests**, **bug reports**, **software support issues** among others.

As with the TracWiki, this subsystem has been designed to make user contribution and participation as simple as possible.

An issue is assigned to a person who must resolve it or reassign the ticket to someone else. All tickets can be edited, annotated, assigned, prioritized and discussed at any time.

However, a Trac installation may place restrictions on who can change what. For example, the default installation doesn't permit to non-authenticated users ("anonymous" users) to change anything, even to comment on an issue, for obvious spam prevention reasons. Check the local contributing policy, which you can usually find on the front page of WikiStart, or contact your local Trac administrator.

Ticket Fields

A ticket contains the following information:

- **Reporter** ? The author of the ticket.
- **Type** ? The category of the ticket. The default types are `defect`, `enhancement` and `task`.
- **Component** ? The project module or subsystem that this ticket concerns.
- **Version** ? Version of the project that this ticket pertains to.
- **Keywords** ? Keywords that a ticket is tagged with. Useful for searching and report generation.
- **Priority** ? The importance of this issue, ranging from *trivial* to *blocker*. A dropdown list when multiple priorities are defined.
- **Milestone** ? Due date of when this issue should be resolved. A dropdown list containing the milestones.
- **Assigned to/Owner** ? Principal person responsible for handling the issue.
- **Cc** ? A comma-separated list of other users or email addresses to notify. Note that this does not imply responsibility or any other policy.
- **Resolution** ? Reason for why a ticket was closed. One of `fixed`, `invalid`, `wontfix`, `duplicate`, `worksforme`.
- **Status** ? What is the current status? The statuses are defined in the ticket workflow. For the default workflow the statuses are `new`, `assigned`, `accepted`, `closed` and `reopened`.
- **Summary** ? A description summarizing the issue. Simple text without WikiFormatting.
- **Description** ? The body of the ticket. A good description should be specific, descriptive and to the point. Accepts WikiFormatting.

Notes:

- Versions of Trac prior to 0.9 did not have the *type* field, but instead provided a *severity* field and different default values for the *priority* field. This change was done to simplify the ticket model by removing the somewhat blurry distinction between *priority* and *severity*. However, the old model is still available if you prefer it: just add/modify the default values of the *priority* and *severity*, and optionally hide the *type* field by removing all the possible values through trac-admin.

- The ?type, ?component, version, priority and severity fields can be managed with trac-admin or through the WebAdmin.

- Description of the builtin *priority* values is available at ?TicketTypes.

Changing and Commenting Tickets

With appropriate permissions, as already mentioned above, a ticket entered into Trac can at any time be modified by **annotating**.

Then, annotations like changes and comments to the ticket are logged as a part of the ticket itself. When viewing a ticket, the history of changes will appear below the main ticket area.

Comment editing (available since Trac version 0.12) is meant to be used to make small corrections to comments, like fixing formatting, forgotten WikiFormatting or spelling errors, not major edits. For longer edits, you should be adding a new comment instead. Editing a comment will not produce a new entry on timeline, while entering a new comment or other changes will do.

All edits (field changes, new comments, comment edits) update the "last changed" time of the ticket.

Notes:

- An important feature is being able to use TracLinks and WikiFormatting in ticket descriptions and comments. Use TracLinks to refer to other issues, changesets or files to make your ticket more specific and easier to understand.

- See TracNotification for how to configure email notifications of ticket changes.

- See TracWorkflow for information about the state transitions (ticket lifecycle), and how this workflow can be customized.

Default Values for Drop-Down Fields

The option selected by default for the various drop-down fields can be set in trac.ini, in the `[ticket]` section:

- `default_component`: Name of the component selected by default.
- `default_milestone`: Name of the default milestone.
- `default_priority`: Default priority value.
- `default_severity`: Default severity value.
- `default_type`: Default ticket type.
- `default_version`: Name of the default version.
- `default_owner`: Name of the default owner. If set to the text `< default >` (the default value), the component owner is used.

If any of these options are omitted, the default value will either be the first in the list, or an empty value, depending on whether the field in question is required to be set. Some of these can be chosen through the WebAdmin in the "Ticket System" section, others can be set in the [ticket] section in `trac.ini`.

Hiding Fields and Adding Custom Fields

Many of the default ticket fields can be hidden from the ticket web interface simply by removing all the possible values through trac-admin. This of course only applies to drop-down fields, such as *type*, *priority*, *severity*,

component, *version* and *milestone*.

Trac also lets you add your own custom ticket fields. See TracTicketsCustomFields for more information.

Assign-to as Drop-Down List

If the list of possible ticket owners is finite, you can change the *assign-to* ticket field from a text input to a drop-down list. This is done by setting the `restrict_owner` option of the `[ticket]` section in trac.ini to `true`. In that case, Trac will populate the list with all users who **have an authenticated session** and possess the `TICKET_MODIFY` permissions.

An authenticated session will be created the first time a user authenticates with the project. You can manually add an authenticated session using the trac-admin `session add` command. The `:1` suffix on the session id (i.e. username) is the key to creating an authenticated session:

```
trac-admin /path/to/projenv session add <sid>:1 [name] [email]
```

You may find the dropdown list is *overpopulated* with users that are no longer active in the project. Revoking authentication privileges will not remove the session data that is used to populate the dropdown list. The trac-admin command can be used to list and remove sessions:

- List all sessions:
  ```
  trac-admin /path/to/projenv session list
  ```
- Remove a session:
  ```
  trac-admin /path/to/projenv session delete SID
  ```

Alternatively, you can just revoke `TICKET_MODIFY` from users that you don't want to be included in the list. However, that will not be possible if you've granted `TICKET_MODIFY` to all *anonymous* or *authenticated* users.

Notes:

- If you need more flexibility and aren't afraid of a little plugin coding of your own, see the ?FlexibleAssignTo plugin.

- Activating this option may cause some performance degradation. Read more about this in the ?Trac performance page.

Preset Values for New Tickets

To create a link to the new-ticket form filled with preset values, you need to call the `/newticket?` URL with `variable=value` separated by `&`. Possible variables are:

- **type** ? The type droplist.
- **reporter** ? Name or email of the reporter.
- **summary** ? Summary line for the ticket.
- **description** ? Long description of the ticket.
- **component** ? The component dropdown list.
- **version** ? The version dropdown list.
- **severity** ? The severity dropdown list.
- **keywords** ? The keywords or tags.
- **priority** ? The priority dropdown list.
- **milestone** ? The milestone dropdown list.
- **owner** ? The person responsible for the ticket.

- **cc** ? The list of emails for notifying about the ticket change.

Example: `[/newticket?summary=Compile%20Error&version=1.0&component=gui]`

Deleting Tickets

Ticket delete and ticket change delete functions are enabled through an optional component. To enable the functionality edit the [components] section of TracIni:

```
[components]
tracopt.ticket.deleter.* = enabled
```

The *Delete* buttons appears next to the *Reply* buttons in the ticket description and ticket change areas. `TICKET_ADMIN` permission is required for deleting tickets and ticket changes.

See also: TracTicketsCustomFields, TracNotification, TracReports, TracQuery