

# Trac Ticket Queries

## Error: Macro TracGuideToc(None) failed

'NoneType' object has no attribute 'find'

In addition to [reports](#), Trac provides support for *custom ticket queries*, used to display lists of tickets meeting a specified set of criteria.

To configure and execute a custom query, switch to the *View Tickets* module from the navigation bar, and select the *Custom Query* link.

## Filters

When you first go to the query page the default filters will display all open tickets, or if you're logged in it will display open tickets assigned to you. Current filters can be removed by clicking the button to the right with the minus sign on the label. New filters are added from the pulldown list in the bottom-right corner of the filters box. Filters with either a text box or a pulldown menu of options can be added multiple times to perform an *or* of the criteria.

You can use the fields just below the filters box to group the results based on a field, or display the full description for each ticket.

Once you've edited your filters click the *Update* button to refresh your results.

## Navigating Tickets

Clicking on one of the query results will take you to that ticket. You can navigate through the results by clicking the *Next Ticket* or *Previous Ticket* links just below the main menu bar, or click the *Back to Query* link to return to the query page.

You can safely edit any of the tickets and continue to navigate through the results using the *Next/Previous/Back to Query* links after saving your results. When you return to the query *any tickets which were edited* will be displayed with italicized text. If one of the tickets was edited such that it no longer matches the query criteria the text will also be greyed. Lastly, if **a new ticket matching the query criteria has been created**, it will be shown in bold.

The query results can be refreshed and cleared of these status indicators by clicking the *Update* button again.

## Saving Queries

While Trac does not yet allow saving a named query and somehow making it available in a navigable list, you can save references to queries in Wiki content, as described below.

## Using TracLinks

You may want to save some queries so that you can come back to them later. You can do this by making a link to the query from any Wiki page.

```
[query:status=new|assigned|reopened&version=1.0 Active tickets against 1.0]
```

Which is displayed as:

## Active tickets against 1.0

This uses a very simple query language to specify the criteria (see [Query Language](#)).

Alternatively, you can copy the query string of a query and paste that into the Wiki link, including the leading ? character:

```
[query:?status=new&status=assigned&status=reopened&group=owner Assigned tickets by owner]
```

Which is displayed as:

### Assigned tickets by owner

## Using the `[[TicketQuery]]` Macro

The `?TicketQuery` macro lets you display lists of tickets matching certain criteria anywhere you can use [WikiFormatting](#).

Example:

```
[[TicketQuery(version=0.6|0.7&resolution=duplicate)]]
```

This is displayed as:

No results

Just like the [query: wiki links](#), the parameter of this macro expects a query string formatted according to the rules of the simple [ticket query language](#).

A more compact representation without the ticket summaries is also available:

```
[[TicketQuery(version=0.6|0.7&resolution=duplicate, compact)]]
```

This is displayed as:

No results

Finally if you wish to receive only the number of defects that match the query using the count parameter.

```
[[TicketQuery(version=0.6|0.7&resolution=duplicate, count)]]
```

This is displayed as:

0

## Customizing the *table* format

You can also customize the columns displayed in the table format (*format=table*) by using *col=<field>* - you can specify multiple fields and what order they are displayed by placing pipes (|) between the columns like below:

```
[[TicketQuery(max=3, status=closed, order=id, desc=1, format=table, col=resolution|summary|owner|reporter)]]
```

This is displayed as:

## Results (1 - 3 of 530)

1 [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) ?

<u>Ticket</u>	<u>Resolution</u>	<u>Summary</u>	<u>Owner</u>	<u>Reporter</u>
<a href="#">#592</a>	invalid	<a href="#">Wrong characters in text</a>		rmsyam
<a href="#">#591</a>	fixed	<a href="#">using the docky to close yam does not work</a>	tboeckel	MichaelMerkel
<a href="#">#590</a>	wontfix	<a href="#">Crash when clicking on Start while the email list is being retrieved</a>		Elwood

1 [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) ?

### Full rows

In *table* format you can also have full rows by using *rows=<field>* like below:

```
[[TicketQuery(max=3,status=closed,order=id,desc=1,format=table,col=resolution|summary|owner|reporter,r
```

This is displayed as:

## Results (1 - 3 of 530)

1 [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) ?

<u>Ticket</u>	<u>Resolution</u>	<u>Summary</u>	<u>Owner</u>	<u>Reporter</u>
<a href="#">#592</a>	invalid	<a href="#">Wrong characters in text</a>		rmsyam

### Summary

E-mail has wrong characters in text

### Steps to reproduce

1. read the mail 2.

### Expected results

#### Description

Mail should have accented characters

### Actual results

Accented characters are replaced by "?"

### Regression

### Notes

<a href="#">#591</a>	fixed	<a href="#">using the docky to close yam does not work</a>	tboeckel	MichaelMerkel
----------------------	-------	--	----------	---------------

### Description Summary

when using the context menu on the YAM docky in amidock ans selecting "beenden" it does not work. this is with yam minimized to the docky.

## Steps to reproduce

1. open yam on its own screen.
2. close/minimize yam
3. -> only the docky is visible now. no yam screen or other yam icon.
4. select "beenden" (quit/close?) from the context menu of the yam docky.

## Expected results

yam should be quit. the docky be removed.

## Actual results

yam still runs. docky does not vanish. and yam can not be opened again using the docky! but you can make it show up by starting yam again. then the screen and window opens. and the "do you want to quit yam" requester is visible then (in the background unfortunately - actually a second problem?)

i can quit it then from there.

## Regression

always behaves like that.

## Notes

current nightly (2.10-dev 15.12.2014 r8294) os4.1 FE beta system.

#590      wontfix      Crash when clicking on Start while the email list is being retrieved      Elwood

## Description Summary

## Steps to reproduce

1. I click get, the "retrieving window" opens
2. while the list is being populated in the window, I immediately click on Start or Abort
3. crash occurs

## Expected results

## Actual results

Crashlog attached done with debug version

## Regression

# Notes

Yam debug 16/10/2014

[1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) [?](#)

## Query Language

query: [TracLinks](#) and the `[[TicketQuery]]` macro both use a mini 'query language' for specifying query filters. Basically, the filters are separated by ampersands (&). Each filter then consists of the ticket field name, an operator, and one or more values. More than one value are separated by a pipe (|), meaning that the filter matches any of the values.

The available operators are:

- = the field content exactly matches the one of the values
- ~= the field content contains one or more of the values
- ^= the field content starts with one of the values
- \$= the field content ends with one of the values

All of these operators can also be negated:

- != the field content matches none of the values
- !~= the field content does not contain any of the values
- !^= the field content does not start with any of the values
- !\$= the field content does not end with any of the values

---

See also: [TracTickets](#), [TracReports](#), [TracGuide](#)