

# Upgrade Instructions

## Error: Macro TracGuideToc(None) failed

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

A Trac environment sometimes needs to be upgraded before it can be used with a new version of Trac. This document describes the steps necessary to upgrade an environment.

**Note:** *Environment upgrades are not necessary for minor version releases unless otherwise noted. For example, there's no need to upgrade a Trac environment created with (or upgraded) 0.8.0 when installing 0.8.4 (or any other 0.8.x release).*

## General Instructions

Typically, there are four steps involved in upgrading to a newer version of Trac:

### Update the Trac Code

Get the new version of Trac, either by downloading an official release package or by checking it out from the [Subversion repository](#).

If you have a source distribution, you need to run

```
python setup.py install
```

to install the new version. If you've downloaded the Windows installer, you execute it, and so on.

In any case, if you're doing a major version upgrade (such as from 0.8 to 0.9), it is *highly* recommended that you first remove the existing Trac code. To do this, you need to delete the `trac` directory from the Python `lib/site-packages` directory. You may also want to remove the `Trac cgi-bin`, `htdocs` and `templates` directories that are commonly found in a directory called `share/trac` (the exact location depends on your platform).

### Upgrade the Trac Environment

Unless noted otherwise, upgrading between major versions (such as 0.8 and 0.9) involves changes to the database schema, and possibly the layout of the [environment directory](#). Fortunately, Trac provides automated upgrade scripts to ease the pain. These scripts are run via [trac-admin](#):

```
trac-admin /path/to/projenv upgrade
```

This command will do nothing if the environment is already up-to-date.

Note that if you are using a PostgreSQL database, this command will fail with the message that the environment can only be backed up when you use an SQLite database. This means that you will have to backup the repository and the database manually. Then, to perform the actual upgrade, run:

```
trac-admin /path/to/projenv upgrade --no-backup
```

## Update the Trac Documentation

Every [Trac environment](#) includes a copy of the Trac documentation for the installed version. As you probably want to keep the included documentation in sync with the installed version of Trac, [trac-admin](#) provides a command to upgrade the documentation:

```
trac-admin /path/to/projenv wiki upgrade
```

Note that this procedure will of course leave your WikiStart page intact.

## Restart the Web Server

In order to reload the new Trac code you will need to restart your web server (note this is not necessary for [CGI](#)).

## Specific Versions

The following sections discuss any extra actions that may need to be taken to upgrade to specific versions of Trac.

### From 0.9-beta to 0.9

If inclusion of the static resources (style sheets, javascript, images) is not working, check the value of the `htdocs_location` in `trac.ini`. For [mod\\_python](#), [Tracd](#) and [FastCGI](#), you can simply remove the option altogether. For [CGI](#), you should fix it to point to the URL you mapped the Trac `htdocs` directory to (although you can also remove it and then [map the static resources](#)). If you're still having problems after removing the option, check the paths in the `trac/siteconfig.py` file and fix them if they're incorrect.

If you've been using plugins with a beta release of Trac 0.9, or have disabled some of the built-in components, you might have to update the rules for disabling/enabling components in [trac.ini](#). In particular, globally installed plugins now need to be enabled explicitly. See [TracPlugins](#) and [TracIni](#) for more information.

If you want to enable the display of all ticket changes in the timeline (the `?Ticket Details?` option), you now have to explicitly enable that in [trac.ini](#), too:

```
[timeline]
ticket_show_details = true
```

### From 0.8.x to 0.9

[mod\\_python](#) users will also need to change the name of the `mod_python` handler in the Apache HTTPD configuration:

```
from: PythonHandler trac.ModPythonHandler
to:   PythonHandler trac.web.modpython_frontend
```

If you have [?PySQLite 2.x](#) installed, Trac will now try to open your SQLite database using the SQLite 3.x file format. The database formats used by SQLite 2.8.x and SQLite 3.x are incompatible. If you get an error like *?file is encrypted or is not a database?* after upgrading, then you must convert your database file.

To do this, you need to have both SQLite 2.8.x and SQLite 3.x installed (they have different filenames so can coexist on the same system). Then use the following commands:

```
$ mv trac.db trac2.db
$ sqlite trac2.db .dump | sqlite3 trac.db
```

After testing that the conversion was successful, the `trac2.db` file can be deleted. For more information on the SQLite upgrade see [?http://www.sqlite.org/version3.html](http://www.sqlite.org/version3.html).

## From 0.7.x to 0.8

0.8 adds a new roadmap feature which requires additional permissions. While a fresh installation will by default grant `ROADMAP_VIEW` and `MILESTONE_VIEW` permissions to anonymous, these permissions have to be granted manually when upgrading:

```
$ trac-admin /path/to/projectenv permission add anonymous MILESTONE_VIEW
$ trac-admin /path/to/projectenv permission add anonymous ROADMAP_VIEW
```

---

See also: [TracGuide](#), [TracInstall](#)