

## Trac Backup

Since Trac uses a database backend, some extra care is required to safely create a backup of a [project environment](#). Luckily, [trac-admin](#) has a command to make backups easier: `hotcopy`.

*Note: Trac uses the `hotcopy` nomenclature to match that of [Subversion](#), to make it easier to remember when managing both Trac and Subversion servers.*

### Creating a Backup

To create a backup of a live [TracEnvironment](#), simply run:

```
$ trac-admin /path/to/projenv hotcopy /path/to/backupdir
```

[trac-admin](#) will lock the database while copying.

The resulting backup directory is safe to handle using standard file-based backup tools like `tar` or `dump/restore`.

Please, note, that `hotcopy` command does not overwrite target directory and when such exists, `hotcopy` ends with error: `Command failed: [Errno 17] File exists: This is discussed in ■#3198.`

### Restoring a Backup

Backups are simply a copied snapshot of the entire [project environment](#) directory, including the SQLite database.

To restore an environment from a backup, stop the process running Trac (i.e. the Web server or [tracd](#)), restore the contents of your backup (`path/to/backupdir`) to your [project environment](#) directory and restart the service.

*Note: Automatic backup of environments that don't use SQLite as database backend is not supported at this time. As a workaround, we recommend that you stop the server, copy the environment directory, and make a backup of the database using whatever mechanism is provided by the database system.*

See also: [TracAdmin](#), [TracEnvironment](#), [TracGuide](#), [■TracMigrate](#)