

Continuous integration

Basically, check out latest version of source, build, and run automated tests.

Shockingly useful.

Using buildbot

Master/slave setup; one master,
multiple slaves (one for each
platform).

Each slave runs one or more buildbot
processes.

Using buildbot

```
def make_darcs_build_factory(url, env={ }):  
    f = factory.BuildFactory()  
    f.addStep(Darcs, repourl=url, mode="update")  
    f.addStep(Test, command=["nosetests", "-v"], env=env)  
    return f
```

Using buildbot

```
f1 = factory.BuildFactory()
```

```
f1.addStep(SVN, mode="update", workdir="build",  
           svnurl="http://svn.idyll.org/repos/cartwheel/trunk/")
```

```
f1.addStep(ShellCommand, command=["cmake", "."])
```

```
f1.addStep(Compile, command=["make"])
```

Why is continuous integration effective?

Developer-independent test reporting.

Multiple (cleanroom!) environments & platforms.

Can test against “legacy” software dependencies.

Runs long tests while you do other things.

Records test failures for posterity.