

# Nailgun

This page describes how to speed up execution of DSpace command-line scripts (class launcher) using [Nailgun](#) by Martian Software, Inc.

The reason why even the simplest DSpace command-line scripts usually take several seconds to run is because the Java class launcher has to create a new JVM instance every time it's run. Nailgun has a client/server architecture and implements a very lightweight JVM in the server part, which keeps running as a service and therefore doesn't have to be started up every time. You then use the command line nailgun client (ng) to connect to it to launch a class, which is considerably faster than starting up the JVM.

1. Download and build the nailgun client and server (this assumes that dspace is installed in /dspace, that you have gcc installed and that you can run binaries from ~/bin):

```
git clone git://github.com/martylamb/nailgun.git
cd nailgun
mvn clean install
cp nailgun-server/target/nailgun-server-0.9.2-SNAPSHOT.jar /dspace/lib/
cp nailgun-server/target/nailgun-server-0.9.2-SNAPSHOT.jar /dspace/webapps/xmlui/WEB-INF/lib/
cd nailgun-client
gcc ng.c -o ng
cp ng ~/bin
# exit your shell and launch it again
```

2. Start the nailgun server and add DSpace to its classpath:

```
java -classpath /dspace/lib/nailgun-server-0.9.2-SNAPSHOT.jar com.martiansoftware.nailgun.NGServer &
ng ng-cp /dspace/lib/*.jar /dspace/config
ng ng-alias dspace org.dspace.app.launcher.ScriptLauncher
```

3. Run the client:

```
# instead of this:
/dspace/bin/dspace
# use this:
ng dspace

# Example. Instead of
/dspace/bin/dspace update-discovery-index
# use this
ng dspace update-discovery-index
```

4. After you're finished running multiple invocations of the DSpace command-line scripts, you can stop the Nailgun server:

```
ng ng-stop
```