

Environment Setup Mars

- 1 [Setup for development of Ergo 4.0](#)
- 2 [Checking out Ergo projects with Git](#)
- 3 [Creating a Fork of Ergo](#)
- 4 [Code Format](#)

Setup for development of Ergo 4.0

1. **Download Eclipse.** You should download the latest version of Eclipse, which as of this writing is Eclipse Mars that you can download [here](#). Select the **Eclipse for RCP and RAP developers** download for your system specification.
2. **Install Eclipse.** Unpack the Eclipse zip file, find the folder where you unzipped it, and launch the IDE.
3. Select a workspace. The default is fine; however, you may want to consider naming it according to your development project (e.g. **ergo-workspace**) to distinguish it from other workspaces.
4. There are two Eclipse features that you will need to install before proceeding to the next section.
 - a. Go to **Help -> Install New Software...**
 - b. Where it says **Work with:** select Mars - <http://download.eclipse.org/releases/mars>
 - c. Expand **Modeling** and select **Graphical Editing Framework GEF SDK** and **Graphical Editing Framework Zest Visualization Toolkit SDK**.
 - d. Click Next, Click Next again, accept the license and Click Finish. After installing the plugins, restart Eclipse when it asks you to do so.

Checking out Ergo projects with Git

The plugin projects for Ergo can be checked out from the following Git repository:

- ergo.git

You can find this repository here: <https://opensource.ncsa.illinois.edu/stash/projects/ERGO> The other projects listed on that page are deprecated and can be ignored.

If you are not part of the **ergo-dev** group on opensource.ncsa.illinois.edu and are interested in contributing to Ergo, you have two options. You can either request to be added to **ergo-dev** and follow the instructions below or you can skip this section and go to the section on Forking the Ergo git repository. If you need an account on Opensource, see the guideline for [How to contribute changes to Ergo](#). If you have developer access and this is your first time cloning the repository, make sure you go to opensource.ncsa.illinois.edu and update your SSH key or you will not be able to clone the repository. If you haven't generated or aren't sure if you have setup an SSH Key on your computer, follow the instructions here: <http://git-scm.com/book/en/Git-on-the-Server-Generating-Your-SSH-Public-Key>

After generating your SSH Key, do the following:

1. Go to <https://opensource.ncsa.illinois.edu/stash>
2. In the upper right corner under your account profile, there should be a dropdown menu, from it select **Manage Account**
3. Select **SSH Keys** and click **Add Key**
4. Follow the instructions and be sure to add your **PUBLIC** key.

After setting up your SSH keys you will need to wait about 1 hour for the database on opensource to update with your new keys before you can clone the new repository.

Import all repositories by doing the following:

1. **File > Import > Git > Projects from Git** and click Next.
2. Select **Clone URI** and click Next.
3. If you haven't yet cloned the repository to your machine, do the following:
 - a. Go to <https://opensource.ncsa.illinois.edu/stash/projects/ERGO>, login to your account on opensource and select the repository you want to clone (e.g. ergo).
 - b. In the menu on the left side of the page, under **Actions**, select **CLONE** and copy the link. It should look something like `ssh://git@opensource.ncsa.illinois.edu:7999/ergo/ergo.git` for the ergo repository. If you only want read-only access, you can follow these steps, but don't login to opensource.
 - c. Go back Eclipse and paste the link Under Location: URI: for read-only access - <https://opensource.ncsa.illinois.edu/bitbucket/scm/ergo/ergo.git> for developer access- `ssh://git@opensource.ncsa.illinois.edu:7999/ergo/ergo.git`
 - d. Under Authentication: User: If you will be contributing source code, you should create a user account at <http://opensource.ncsa.illinois.edu> and specify your username for opensource. If you used the read-only link, you can ignore this part.
 - e. Under Authentication: Password: leave it blank if you are using the read only link, or your password if you are a developer and used the ssh link.
 - f. Click Next.
4. Choose the branches you want to clone. For the initial setup, check the box next to **master** and **develop-v1** and click Next.
5. Where it says **Destination**, change the **Directory** from `"/home/user/git/(repository-name)"` to `"/home/user/ergo-workspace/git/(repository-name)"`. This assumes you used **ergo-workspace** as your new workspace, if not, replace **ergo-workspace** with the name of the workspace you created. The **Directory** inside the dialog box should look similar to `/home/user/ergo-workspace/git/ergo` if you are cloning the **ergo** repository. This will keep your git clones inside the workspace (recommended, but not required). After modifying the **Directory** field, Click Next. Note: If you use the **Browse** button to select the location, you will need to browse for your workspace and create a folder inside your workspace folder called **git** and select the **git** folder. Either way, your **Directory** field should show `/home/user/ergo-workspace/git/ergo`

6. Select **Import existing projects** and click Next
7. Uncheck the box **Search for nested projects** and Click Finish.

Note: While trying to clone the repositories, if you see the error *1: incorrect URL 2: no network connection 3: SSH is not configured correctly*, you need to modify your **SSH2** preferences in Eclipse to point at your .ssh folder and your private key(s). For help with this, see [ssh2-preferences](#).

Creating a Fork of Ergo

If you are a developer interested in Ergo, but might not want your changes included in the master repository right away, then you may want to fork the Ergo repository. The only changes to the above would be you would go to <https://opensource.ncsa.illinois.edu/stash/projects/ERGO> and create a Fork of each repository before cloning. When creating your fork, it is recommended that you check the box to "Enable fork syncing". This will keep branches and tags in sync with the branch you forked (presumably master). This path is not recommended if you intend to make continuous contributions to Ergo because forks are usually intended to take development a very different direction than the main development branch. Thus, developers are encouraged to sign the license agreement found [here](#) so you can contribute to the master branch by creating feature branches from it and help guide the development process through interactions on the ergo-dev mailing list.

Code Format

Setup Ergo code formatter and cleanup profiles as specified in the [Developer Guidelines](#)