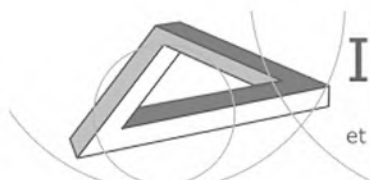




ECOLE  
POLYTECHNIQUE  
DE BRUXELLES



IRIDIA  
Institut de Recherches  
et de Développement Interdisciplinaires  
en Intelligence Artificielle

---

# IRIDIA Documentation

---

10 mars 2025

# Argos3 installation

1. Install all prerequisites (specified at the github page). You can do so by running the following command in your terminal :

```
$ sudo apt-get install libfreeimage-dev libfreeimageplus-dev qt5-default  
└─ freeglut3-dev libxi-dev libxmu-dev liblua5.2-dev lua5.2 doxygen graphviz  
└─ graphviz-dev asciidoc
```

2. Download and install argos :

- Be aware that we are currently not using the most recent version but instead the release "3.0.0-beta48" :

(a) `git clone https://github.com/ilpincy/argos3.git`

(b) `cd argos3 && git checkout 3.0.0-beta48`

checks out the version that we are using at the moment

- If you are on the cluster : Change the CMake-version from 2.8.12 to 2.8.8 in the src/CMakeLists.txt file. There are two occurrences that need to be replaced.

- Prepare the build of argos3 :

(a) `export ARGOS_INSTALL_PATH=$HOME`

if you want to install argos3 into \$HOME/argos3-dist (recommended on the cluster), otherwise choose the location to your liking

(b) `mkdir build && cd build`

(c) `cmake -DCMAKE_INSTALL_PREFIX=$ARGOS_INSTALL_PATH/argos3-dist  
└─ -DCMAKE_BUILD_TYPE=Release -DARGOS_INSTALL_LDSOCONF=OFF  
└─ -DARGOS_DOCUMENTATION=OFF ../src`

(d) `make`

(e) `make doc`

if you have forgotten the -DARGOS\_DOCUMENTATION=OFF

(f) `make install`

- Before you can install our epuck-plugin, you have to remove the default plugin first. In argos3-dist run the following commands :

- (a) 

```
rm -rf  
  ↳ $ARGOS_INSTALL_PATH/argos3-dist/include/argos3/plugins/robots/e-puck
```
- (b) 

```
rm -rf $ARGOS_INSTALL_PATH/argos3-dist/lib/argos3/lib*epuck*.so
```

— Add the following lines to your `.bashrc` (or create a separate file and source it from your `.bashrc`) :

```
export ARGOS_INSTALL_PATH=$HOME (or whatever you chose in the earlier  
  ↳ step)  
export PKG_CONFIG_PATH=_ARGOS_INSTALL_PATH/argos3-dist/lib/pkgconfig  
export ARGOS_PLUGIN_PATH=_ARGOS_INSTALL_PATH/argos3-dist/lib/argos3  
export LD_LIBRARY_PATH=$ARGOS_PLUGIN_PATH:$LD_LIBRARY_PATH  
export PATH=$ARGOS_INSTALL_PATH/argos3-dist/bin/:$PATH
```

### 3. Download and install the e-puck plugin :

— Get the repository :

- (a) 

```
git clone https://github.com/demiurge-project/argos3-epuck.git
```
- (b) 

```
cd argos3-epuck && git checkout v48
```

— If you are on the cluster :

- (a) Open `src/plugins/robots/e-puck/CMakeLists.txt` and comment the line with `include(VisionTools.cmake)`
- (b) Open `src/cmake/ARGoSBuildChecks.cmake` and remove the checks for Lua and Qt/OpenGL

— Prepare the build of the epuck-plugin :

- (a) 

```
mkdir build && cd build
```
- (b) 

```
cmake -DCMAKE_INSTALL_PREFIX=$ARGOS_INSTALL_PATH/argos3-dist  
  ↳ -DCMAKE_BUILD_TYPE=Release ../src
```
- (c) 

```
make
```
- (d) 

```
make install
```

# AutoMoDe installation

## 1. Download and install the loopfunctions

- (a) 

```
git clone  
└─ https://github.com/demiurge-project/experiments-loop-functions.git
```
- (b) 

```
cd AutoMoDe-loopfunctions
```
- (c) 

```
git checkout dev
```
- (d) 

```
mkdir build && cd build
```
- (e) 

```
cmake -DCMAKE_INSTALL_PREFIX=$ARGOS_INSTALL_PATH/argos3-dist  
└─ -DCMAKE_BUILD_TYPE=Release ..
```
- (f) 

```
make
```
- (g) 

```
make install
```

## 2. Download and install the e-puck DAO

- (a) 

```
git clone https://github.com/demiurge-project/demiurge-epuck-dao.git
```
- (b) 

```
cd AutoMoDe-DAO
```
- (c) 

```
mkdir build && cd build
```
- (d) 

```
cmake -DCMAKE_INSTALL_PREFIX=$ARGOS_INSTALL_PATH/argos3-dist  
└─ -DCMAKE_BUILD_TYPE=Release ..
```
- (e) 

```
make
```
- (f) 

```
make install
```

## 3. Download and install AutoMoDe

- (a) 

```
git clone https://github.com/demiurge-project/ARGoS3-AutoMoDe.git AutoMoDe
```
- (b) 

```
cd AutoMoDe
```
- (c) 

```
mkdir build && cd build
```
- (d) 

```
cmake ..
```
- (e) 

```
make
```

## Irace installation

— In order to install irace on the cluster, follow these steps :

1. Create folder "~/R/library/"
2. In R console, run :

```
install.packages("Rmpi", "~/R/library",  
  ↪ configure.args="--with-mpi=/opt/ohpc/pub/mpi/openmpi4-gnu9/4.0.5")
```

3. Download the irace 2.2 package (<https://nextcloud.ananas.space/s/WhQ2yqhmqoaBRdC>)
4. Upload the .tar.gz file to your home directory (for example in ~/irace/)
5. Find the file main.R and find line 178 (after :

```
parameters <- readParameters (file = scenario$parameterFile,  
                               dependencyFile = scenario$dependencyFile,  
                               digits = scenario$digits,  
                               debugLevel = debug.level)
```

Add the following two lines here :

```
parameters$names = unlist(parameters$names)  
parameters$isFixed = unlist(parameters$isFixed)
```

6. In your home directory, run :

```
R CMD INSTALL --library=~ /R/library/ irace/
```

to install the contents of irace/ into ~/R/library.

7. Add the R library path, IRACE\_HOME, etc. environment variables in ~/.bashrc :

```
export R_LIBS_USER=~ /R/library  
export IRACE_HOME=${R_LIBS_USER}/irace  
export PATH=${IRACE_HOME}/bin/:${PATH}  
export R_LIBS=${R_LIBS_USER}:${R_LIBS}
```