

 Personal Open source Business Explore Pricing Blog Support

This repository Search Sign in Sign up

deepmind / lab

Watch 210 Star 2,250 Fork 362

<> Code 1 Issues 2 2 Pull requests 0 0 Projects 0 Pulse Graphs

A customisable 3D platform for agent-based AI research

29 commits 1 branch 1 release 2 contributors

Branch: master New pull request Find file Clone or download

DeepMind Lab Team committed with tkoepp [README.md] Update paper link ...	Latest commit 432e063 2 days ago
assets	Assets created by DeepMind 12 days ago
assets_oa	Assets taken from OpenArena 12 days ago
deepmind	DeepMind hook system backend 12 days ago
docs	[python_api.md] Fix misspelled parameter name in the documentation of... 3 days ago
engine	Remove signal handlers. Standard "signal" must not be used in multith... 7 days ago
examples	[examples] A a human-playable, freestanding example executable 12 days ago
public	Public C API for DeepMind Lab 12 days ago
python	Python bindings for DeepMind Lab 10 days ago
q3map2	Main Bazel build rules 14 days ago
third_party	[third_party] Message-digest algorithm reference implementation 14 days ago
.gitignore	.gitignore rules for Bazel output 14 days ago
BUILD	Add missing dependency on zlib 6 days ago
CONTRIBUTING	Rules for contributing 16 days ago
LICENSE	Initial commit 16 days ago
README.md	[README.md] Update paper link 2 days ago
WORKSPACE	Bazel workspace and external build files 14 days ago
dmlab.lds	Linker script for isolating the main code into a DSO 14 days ago
eigen.BUILD	Bazel workspace and external build files 14 days ago
glib.BUILD	Bazel workspace and external build files 14 days ago
googletest.BUILD	Bazel workspace and external build files 14 days ago
jpeg.BUILD	Bazel workspace and external build files 14 days ago
libxml.BUILD	Bazel workspace and external build files 14 days ago
lua.BUILD	Bazel workspace and external build files 14 days ago
png.BUILD	Bazel workspace and external build files 14 days ago
python.BUILD	Bazel workspace and external build files 14 days ago
sdl.BUILD	Bazel workspace and external build files 14 days ago
zlib.BUILD	Bazel workspace and external build files 14 days ago

README.md



# DeepMind Lab

*DeepMind Lab* is a 3D learning environment based on id Software's [Quake III Arena](#) via [ioquake3](#) and [other open source software](#).



*DeepMind Lab* provides a suite of challenging 3D navigation and puzzle-solving tasks for learning agents. Its primary purpose is to act as a testbed for research in artificial intelligence, especially deep reinforcement learning.

## About

Disclaimer: This is not an official Google product.

If you use *DeepMind Lab* in your research and would like to cite the *DeepMind Lab* environment, we suggest you cite the [DeepMind Lab paper](#).

You can reach us at [lab@deepmind.com](mailto:lab@deepmind.com).

## Getting started on Linux

- Get [Bazel](#) from [bazel.io](#).
- Clone DeepMind Lab, e.g. by running

```
$ git clone https://github.com/deepmind/lab
$ cd lab
```

- For a live example of a random agent, run

```
lab$ bazel run :random_agent --define headless=false -- \
    --length=10000 --width=640 --height=480
```

Here is some [more detailed build documentation](#), including how to install dependencies if you don't have them.

## Play as a human

To test the game using human input controls, run

```
lab$ bazel run :game -- --level_script tests/demo_map
```

## Train an agent

*DeepMind Lab* ships with an example random agent in [python/random\\_agent.py](#) which can be used as a starting point for implementing a learning agent. To let this agent interact with DeepMind Lab for training, run

```
lab$ bazel run :random_agent
```

The Python API for the agent-environment interaction is described in [docs/python\\_api.md](#).

*DeepMind Lab* ships with different levels implementing different tasks. These tasks can be configured using Lua scripts, as described in [docs/lua\\_api.md](#).

---

## Upstream sources

*DeepMind Lab* is built from the *ioquake3* game engine, and it uses the tools *q3map2* and *bsp* for map creation. Bug fixes and cleanups that originate with those projects are best fixed upstream and then merged into *DeepMind Lab*.

- *bsp* is taken from [github.com/TTimo/bsp](#), revision e6f90a2dc02916aa2298da6ace70a8333b3f2405. There are virtually no local modifications, although we integrate this code with the main *ioq3* code and do not use their copy in the `deps` directory. We expect this code to be stable.
- *q3map2* is taken from [github.com/TTimo/GtkRadiant](#), revision 8557f1820f8e0c7cef9d52a78b2847fa401a4a95. A few minor local modifications add synchronization and use C99 constructs to replace formerly non-portable or undefined behaviour. We also expect this code to be stable.
- *ioquake3* is taken from [github.com/ioquake/ioq3](#), revision 1c1e1f61f180596c925a4ac0eddba4806d1369cd. The code contains extensive modifications and additions. We aim to merge upstream changes occasionally.

We are very grateful to the maintainers of these repositories for all their hard work on maintaining high-quality code bases.

---

## External dependencies, prerequisites and porting notes

*DeepMind Lab* currently ships as source code only. It depends on a few external software libraries, which we ship in several different ways:

- The `zlib`, `glib`, `libxml2`, `jpeg` and `png` libraries are referenced as external Bazel sources, and Bazel BUILD files are provided. The dependent code itself should be fairly portable, but the BUILD rules we ship are specific to Linux on x86. To build on a different platform you will most likely have to edit those BUILD files.
- Message digest algorithms are included in this package (in `//third_party/md`), taken from the reference implementations of their respective RFCs. A "generic reinforcement learning API" is included in `//third_party/rl_api`, which has also been created by the *DeepMind Lab* authors. This code is portable.
- Several additional libraries are required but are not shipped in any form; they must be present on your system:
  - SDL 2
  - Lua 5.1 (later versions might work, too)
  - gettext (required by `glib`)
  - OpenGL: a hardware driver and library are needed for hardware-accelerated human play, and OSMesa is required for the software-rendering, headless library that machine learning agents will want to use.
  - Python 2.7 (other versions might work, too)

The build rules are using a few compiler settings that are specific to GCC. If some flags are not recognized by your compiler (typically those would be specific warning suppressions), you may have to edit those flags. The warnings should be noisy but harmless.

