Welcome to the XVM 10.2.2™ AMI.

This AMI includes XVM, Logtalk, and demo Jupyter notebooks.

Script for AMI Creation

Insert the following user data script before EC2 Launch:

https://xvm-public.s3.us-east-1.amazonaws.com/XVM_10.2.2_user_data.txt

SSH access to the AMI from VSCode

First, install the "Remote - SSH" extension in VSCode running on your host system. Then, follow the extension instructions to connect to the AMI SSH server. After connecting, create a new terminal using the VSCode Terminal menu.

Running XVM

In a terminal (see above), type:

```
$ xvmpl
```

Running Logtalk with XVM

In a terminal (see above), type:

```
$ xvmlgt
```

XVM documentation

The XVM documentation is available from the `/opt/permion/xvm/doc` in HTML, PDF, ePub, and Texinfo formats. There are two sets of documentation: the XVM Handbook and the APIs documentation. The best user experience is using the HTML version. The entry point is `/opt/permion/xvm/doc/handbook/index.html`. In a terminal (see above), type:

```
$ cd /opt/permion/xvm/doc
$ python3 -m http.server 8080
```

The Texinfo versions can be browsed from the XVM top-level interpreter by calling the `handbook/0` and `apis/0` predicates.

Logtalk documentation

The Logtalk documentation is available from the `~/logtalk/manuals` (Handbook) and `~/logtalk/docs` (APIs documentation) directories in HTML, PDF, PDF, ePub, and Texinfo formats. The Texinfo versions can be browsed from the

`xvmlgt` top-level interpreter by loading the `help` tool (using the `{help(loader)}` goal) and calling the `help::handbook/0` and `help::apis/0` predicates. But the best user experience is using the HTML version. In a terminal (see above), type:

```
$ cd ~/logtalk
$ python3 -m http.server 8081
```

After, click in the `manuals` link to open the Logtalk Handbook.

Additional documentation and learning resources can be found at the Logtalk website:

https://logtalk.org

Demo Jupyter notebooks

The `~/demos` directory contains demo Jupyter notebooks of XVM and Logtalk. See the `~/demos/README.md` file for details on the bundled notebooks.

Running the JupyterLab server

Open a new integrated terminal in VSCode and run the following commands:

```
$ cd ~/demos
$ jupyter-lab --allow-root --NotebookApp.token='' --NotebookApp.password=''
```

VSCode will display a dialog with a button to open the JupyterLab start page on your host web browser. If you need a different port than the default 8888, e.g. 8891, use instead:

```
$ jupyter-lab --allow-root --NotebookApp.token='' --NotebookApp.password='' --port=8891
```

See the JupyterLab documentation for more details on available options.

Creating Jupyter notebooks

Create a new notebook using the JupyterLab web interface. If necessary, select the Logtalk kernel by clicking in the "Select Kernel" button in the top right corner of the notebook file. If you create a new notebook outside the `~/demos` directory, copy the `~/demos/logtalk_kernel_config.py` file to the notebook directory to ensure that XVM is used as the backend used to run the notebook.

To check that everything is running, create a code cell in your new notebook and type the following line magic:

```
%versions
```

Executing the cell (by default, Shift-Enter) should print the Logtalk version, the backend version, and the Logtalk Jupyter kernel version.

Running Logtalk examples

Start by installing the "Logtalk for VSCode" extension in the remote and configure it by going into "Settings", typing "Logtalk" in the search box, and selecting "Remote" settings. Open a terminal in the remote to find the values of the Logtalk environment variables to be used for the configuration of the Logtalk extension:

```
$ echo $LOGTALKHOME
$ echo $LOGTALKUSER
```

For the Logtalk backend use `xvm`. The extension documentation (that you can access at any time be selecting the Logtalk extension in the VSCode "Extensions" pane) describes the extension functionality.

Next, select the "File" menu "Open Folder..." command and open e.g. the `~/logtalk/examples/planets`. See the example `NOTES.md` file for details. In the Logtalk website (https://logtalk.org) you can find a "Learning by examples" guide.

Creating a XVM/Logtalk project

1. Be sure to install the "Logtalk for VSCode" extension in the remote (see the previous section for instructions).

2. Using a terminal create a new empty directory for your project.

3. Open the VSCode Command Palette and type "Logtalk" to select the "Logtalk: Create Project" command. Navigate and select the project directory, which will be populated with a set of sample files as a starting point. See the comments in the files themselves for guidance.

4. Before closing the SSH connection, backup you project by copying its files to your host computer.