

Help Topic: The Emacs MOOS Mode

Spring 2020

Michael Benjamin, mikerb@mit.edu
Department of Mechanical Engineering
MIT, Cambridge MA 02139

Emacs Modes for MOOS-IvP Mission and Behavior Files

The Emacs `moos-mode` was written by MIT student Jacob Gerlach in 2014, in Release 14.7, to provide syntax highlighting in Emacs for mission (`.moos`) files and behavior (`.bhv`) files. The aim is to make simple syntax errors easier to detect, before a mission is launched. Once a mission is launched there are other provisions for catching syntax errors, especially in appcast-enabled applications. The Emacs mode makes this a lot easier, especially for newcomers. The idea is shown in the figure below. Note how much easier it is, with highlighting on the right, to detect the syntax error in the first line of the `pNodeReporter` configuration.

Using the Emacs MOOS Mode

The Emacs MOOS mode is implemented in three files distributed with MOOS-IvP source code:

```
$ cd moos-ivp/editor-modes
$ ls
moos-apps.el moos-bhvs.el moos-mode.el
```

To use the mode in your Emacs session, just add a couple lines in your `.emacs` file like this:

```
(add-to-list 'load-path "~/moos-ivp/editor-modes/")
(require 'moos-mode)
```

The exact path on the first line above may depend on where you have `moos-ivp` checked out on your system.

Adding Your Own Apps and Behaviors to the Modes

At some point you may want to add your own MOOS applications to this mode. This can be done by adding further lines to your `.emacs` file, below the two lines mentioned above. For example, if you have a `pFooBar` app that has two parameters `time` and `date`, the following in your `/.emacs` file should do the trick:

```
(add-to-list 'load-path "~/moos-ivp/editor-modes/")
(require 'moos-mode)
...
(moos-add-user-app '("pFooBar" "time" "date"))          <<-- Add this line
```

The operation is similar for IvP behaviors. If you have a behavior `BHV_Smarty` with parameters `insight` and `intuition`, the following in your `/.emacs` file should do the trick:

```
(add-to-list 'load-path "~/moos-ivp/editor-modes/")
(require 'moos-mode)
...
(moos-add-user-bhv '("BHV_Smarty" "insight" "intuition"))    <<-- Add this line
```

Using the Emacs MOOS Mode in Older MOOS-IvP Releases

If you are working with an older release of MOOS-IvP, using the Emacs MOOS mode is still possible. Just download the three lisp files separately:

```
svn co https://oceanai.mit.edu/svn/moos-ivp-aro/releases/moos-ivp-14.7/editor-modes
```

And adjust the line in your `.emacs` file accordingly. For the most part, the supported MOOS application names and parameters defined in these lisp files should be applicable with older releases of MOOS-IvP.

Further Emacs Links:

- Getting Emacs in Linux: http://oceanai.mit.edu/ivpman/help/emacs_get_for_gnulinux
- Getting Emacs in Mac OS X: http://oceanai.mit.edu/ivpman/help/emacs_get_for_osx
- Getting a more Graphical Emacs for Mac OS X: http://oceanai.mit.edu/ivpman/help/emacs_graphical
- Survival Emacs: http://oceanai.mit.edu/ivpman/help/emacs_survival
- Configuring Emacs: http://oceanai.mit.edu/ivpman/help/emacs_config
- Why use Emacs: <http://xkcd.com/378/>

Contact Information

For bug reports, suggestions or feature requests, email Jake at jacobgerlach@gmail.com.

```

// FILE: alpha.moos
ServerHost = localhost
ServerPort = 9000
Community = alpha
LatOrigin = 43.825300
LongOrigin = -70.330400

//-----
// Antler configuration block
ProcessConfig = ANTLER
{
  MSBetweenLaunches = 200

  Run = MOOSDB @ NewConsole = false
  Run = pLogger @ NewConsole = false
  Run = uSimMarine @ NewConsole = false
  Run = pMarinePID @ NewConsole = false
  Run = pHelmIvP @ NewConsole = false
  Run = pMarineViewer @ NewConsole = false
  Run = uProcessWatch @ NewConsole = false
  Run = pNodeReporter @ NewConsole = false
}

//-----
// uProcessWatch config block
ProcessConfig = uProcessWatch
{
  AppTick = 4
  CommsTick = 4

  summary_wait = 5
  nowatch = uMAC*
  nowatch = uPokeDB*
  watch_all = true
}

//-----
// pNodeReporter config block
ProcessConfig = pNodeReporter
{
  platform_tpye = KAYAK
  platform_length = 4
}

```

```

// FILE: alpha.moos
ServerHost = localhost
ServerPort = 9000
Community = alpha
LatOrigin = 43.825300
LongOrigin = -70.330400

//-----
// Antler configuration block
ProcessConfig = ANTLER
{
  MSBetweenLaunches = 200

  Run = MOOSDB @ NewConsole = false
  Run = pLogger @ NewConsole = false
  Run = uSimMarine @ NewConsole = false
  Run = pMarinePID @ NewConsole = false
  Run = pHelmIvP @ NewConsole = false
  Run = pMarineViewer @ NewConsole = false
  Run = uProcessWatch @ NewConsole = false
  Run = pNodeReporter @ NewConsole = false
}

//-----
// uProcessWatch config block
ProcessConfig = uProcessWatch
{
  AppTick = 4
  CommsTick = 4

  summary_wait = 5
  nowatch = uMAC*
  nowatch = uPokeDB*
  watch_all = true
}

//-----
// pNodeReporter config block
ProcessConfig = pNodeReporter
{
  platform_tpye = KAYAK
  platform_length = 4
}

```

Figure 1: **The MOOS Emacs Mode:** On the left is a sample mission file without syntax highlighting. On the right, highlighting is enabled. Note the syntax error in the first line of `pNodeReporter` is more apparent.