# The uMAC Utilities

## June 2018

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

## 1 Overview

In this section the utilities for viewing appcasts are discussed. The include:

- The uMACView utility: A GUI tool containing navigation tools for viewing appcasts across multiple vehicles or nodes. A snapshot is shown in Figure 1.
- The uMAC utility: A utility implement in the terminal window. Simpler in the interface, but notable in that it allows a user to remotely launch the tool on deployed vehicle.

## 2 The uMACView Utility

The uMACView utility is an appcast viewer with a graphical user interface using the FLTK library. It contains three panes as shown in Figure 1. The bottom pane renders incoming appcasts. The top right pane lists possible applications for selecting appcast viewing for the present node. The upper left pane shows all presently known nodes from which appcasts have been received.
The uMACView utility receives appcasts from perhaps several different nodes and applications on each node. The interface allows the user to select a node and application for viewing the selected application’s latest appcast. The viewer will raise alerts from non-selected nodes and applications when or if a run warning occurs.

The content of the *appcast pane* is solely determined by the appcast content itself. From the viewer’s perspective it is simply pushing a list of strings to a browser pane. Even the formatting of the header lines showing the application name, node name and application iteration, are formatted by a method defined over the AppCast class.

The content of the *application pane* lists the applications known thus far to the viewer from incoming appcasts for a particular node. The order is shown by the order in which they were received. The three columns, "AC", "CW", and "RW", show the number of appcasts received from the application, and the number of run warnings and configuration warnings in the latest appcast.

The content of the *node pane* in the upper left lists the nodes known thus far to the viewer from incoming appcasts. The order is shown by the order in which they were received. The three columns, "AC", "CW", and "RW", show the number of appcasts received from a given node for all applications from that node, as well as the sum of all run warnings and configuration warnings for all applications received from the given node.
2.1 Publications and Subscriptions

The sole subscription for uMACView is the appcast message in the MOOS variable APPCAST. The sole publication is the appcast request, in the variable APPCAST_REQ.

2.2 Configuration File Parameters

A few configuration parameters are exposed to facilitate visual preference settings that would otherwise need to be done each time the application is launched via pull-down menus. These parameters are listed below, but may be recalled anytime by typing on the command line: "uMACView -e".

Listing 2.1: Configuration Parameters for uMACView.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Possible Settings</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>nodes_font_size</td>
<td>xsmall, small, medium, large</td>
<td>medium</td>
</tr>
<tr>
<td>procs_font_size</td>
<td>xsmall, small, medium, large</td>
<td>medium</td>
</tr>
<tr>
<td>appcast_font_size</td>
<td>xsmall, small, medium, large</td>
<td>small</td>
</tr>
<tr>
<td>appcast_color_scheme</td>
<td>default, beige, indigo</td>
<td>default</td>
</tr>
<tr>
<td>appcast_height</td>
<td>[30,35,40,...,85,90]</td>
<td>70</td>
</tr>
<tr>
<td>refresh_mode</td>
<td>paused, events, streaming</td>
<td>events</td>
</tr>
</tbody>
</table>

The appcast_height refers to the relative height of the appcast pane to the window. The default of 70 means the pane will be 70% of the overall window height.

The refresh_mode refers to the policy of refreshing appcasts via appcast requests sent to the nodes and their applications. This is discussed below in Section 2.4.

2.3 Command Line Arguments and Options

A few command line arguments are available. They are similar to most other MOOS-IvP applications. These arguments are listed below, but may be recalled anytime by typing on the command line:

```
$ MACView --help or -h
```

- **--alias=**<ProcessName>: Launch with the given process name rather than uMACView.
- **--example, -e**: Display example MOOS configuration block.
- **--help, -h**: Display command line usage.
- **--interface, -i**: Display MOOS publications and subscriptions.
- **--version, -v**: Display release version information.

2.4 Refresh Modes

The uMACView utility, operates in one of three refresh modes. This mode is always shown on the upper right in the title bar in reverse color. The three modes are the streaming, events, and paused modes.
The Paused Refresh Mode  In the paused refresh mode, no appcasts are solicited by the uMAC utility. The information being rendered should remain constant, virtually paused. Since appcast requests have a duration associated with them, prior appcast requests received by an application may need some time before expiring. Therefore the uMAC user may still see a trickle of updates after entering the paused mode. Furthermore, if there is another uMAC utility open, generating appcast requests, updates might still be seen after pausing.

The Events Refresh Mode  In the events refresh mode, appcast requests of two types are sent to all known nodes and applications. The first type of request is sent only to the selected node and application. This type requests a continual update of appcasts, unconditionally. This application is, after all, the selected application for viewing. The second type of request is sent to all other nodes and applications requesting an appcast only if a new run warning is generated. The events refresh mode is the default mode upon launch.

The Streaming Refresh Mode  In the streaming refresh mode, appcast requests are sent to all nodes and all applications, all the time. Furthermore, the request type is unconditional, meaning the application is requested to post a new appcast regardless of whether anything has changed in the application status. This mode is normally used for brief debugging, perhaps to check whether a node or application fails to respond. It creates a lot of appcast messaging traffic. It is not recommended to operate in this mode normally.

3 The uMAC Utility

The uMAC utility is an appcast viewer implemented in the terminal console. It acts the same as uMACView in terms of soliciting appcasts by publishing APPCAST_REQ messages and receiving APPCAST mail. The advantage over the GUI tool is the ability to remotely log into a vehicle and launch uMAC in a terminal window. This is especially useful if the vehicle is otherwise not behaving in its communication with a shoreside MOOS community. Like uMACView, multiple versions of the utility may be running at the same time without interference with one another.
3.1 Content Modes

The viewable information in the uMAC tool is rendered in one of four content modes. Besides a help mode, the other three modes correlate to one of the three panes of the uMACView window in Figure 1. The primary content mode is the appcast content mode, where the output is an appcast of a particular application as shown in Figure 3.

Figure 3: The uMAC utility monitors appcasts from a terminal window. The user may switch between appcasting sources by navigating with a keyboard menu. The primary advantage of uMAC is the ability to run it on a remotely deployed vehicle with a network connection.

The content of this window should look very similar to the bottom pane of the uMACView window in Figure 1. Two other content modes are supported, the nodes content mode, and the procs content mode. The former allows the user to select between different nodes, i.e., vehicles. The latter allows selection between different processes (MOOS applications) for the selected node. These modes correlate to the top two panes in the uMACView tool shown in Figure 1. A fourth, help, content mode is also supported. Transitioning between modes usually is done by selecting one of the choices presented, or popping back up a mode to allow a higher level choice. This done with three reserved keyboard keys, p, n, h, implementing the transitions shown in Figure 4.
An example rendering of uMAC in the *nodes content mode* is shown on the left in Figure 5. Each discovered node is assigned a character ID in the left-most column for selecting the node. The ID’s are assigned as the nodes are discovered from incoming APPCAST messages.

The title line at the top of the report shows, on the left, the name of the uMAC application as it is known to the MOOSDB, and the number of nodes discovered. When uMAC is launched it gives itself a random suffix, such as uMAC_5991 in this example, to allow multiple uMAC sessions connect with the same MOOSDB. Recall the MOOSDB requires unique names across applications. The righthand side of the title line shows the number of iterations of the uMAC in parentheses, and the *refresh mode* shown in reverse color.

An example from the *procs content mode* is shown on the right in Figure 5. Each discovered process (MOOS application) is assigned an ID in the left-most column for selecting the process. The ID’s are assigned as the apps are discovered from incoming APPCAST messages. The title line at the top is nearly the same format as in the *nodes* content mode, except that the selected node is shown rather than the total nodes. The appcast counter just below the title line indicates the total number of appcasts received over all nodes, not just the selected node.

### 3.2 Refresh Modes

The uMAC utility, like the uMacView utility, operates in one of three *refresh modes*. This mode is always shown on the upper right in the title bar in reverse color. The three modes are the *streaming*,
events, and paused modes. The description of these modes, given in Section 2.4, is also applicable to the operation for the uMAC utility.

3.3 A Tip Regarding Process Monitoring and uMAC Sessions

The uProcessWatch application is a utility for monitoring the presence of applications connected to the MOOSDB. It is appcast enabled, and will post a run warning when it detects the disappearance of a prior noted process. If a uMAC is launched and then exited, the uProcessWatch utility may interpret this as a problem and post a run warning. The effect may be that real run warnings are then later ignored. Tip: Add the following configuration line to uProcessWatch to ignore the exit of a uMAC session: nowatch = uMAC*.

3.4 Publications and Subscriptions

The sole subscription for uMACView is the appcast message in the MOOS variable APPCAST. The sole publication is the appcast request, in the variable APPCAST_REQ.

3.5 Configuration File Parameters

There are no configuration file parameters specific to uMAC.

3.5.1 Command Line Arguments and Options

A few command line arguments are available. They are similar to most other MOOS-IvP applications. These arguments are listed below, but may be recalled anytime by typing on the command line:

```
$ uMAC --help or -h
```

- `--alias=<ProcessName>`: Launch with the given process name rather than uMACView.
- `--example`, `-e`: Display example MOOS configuration block.
- `--help`, `-h`: Display command line usage.
- `--interface`, `-i`: Display MOOS publications and subscriptions.
- `--version`, `-v`: Display release version information.
The final uMAC tool is essentially uMACView embedded in the pMarineViewer application as shown in Figure 6. This is mostly just a convenience for users already using pMarineViewer. The appcasting mode may be toggled with the ‘a’ key to return to the traditional viewing layout. The AppCasting pull-down menu offers the same set of selections as uMACView with the exception of the hot keys used to switch between appcasting refresh modes since those keys were already used for other things in pMarineViewer.

In addition to toggling on/off the appcasting portion of the window, the width of the set of appcasting panes may be made wider or thinner using the \texttt{CTRL-ALT-ARROW} keys. By default the appcasting panes consume 30\% of the width. The default height of the appcasting (bottom) portion of the appcasting panes consume 75\% of the height of the set of appcasting panes. These startup extents may be changed with configuration the parameters:

\begin{verbatim}
appcasting_width = 25 // legal values [20, 25,...,65, 70]
appcasting_height = 80 // legal values [30, 35,..., 85, 90]
\end{verbatim}

The prevailing value of these parameters can always be discovered by checking which radio button in the pull-down menu is presently selected.