appmon

appmon 2.1.14.1
September 11 2012

September 11 2012

1 Appmon User's Guide

The Application Monitor, *Appmon*, is a graphical utility used to supervise applications executing either locally or on remote nodes. The process tree of an application can furthermore be monitored.

1.1 Appmon

1.1.1 Introduction

> **Warning:**
>
> The Appmon application has been superseded by the Observer application. Appmon will be removed in R16.

The application monitor Appmon is a graphical node and application viewer. The tool shows an overview of all applications on all known nodes, and it is possible to view the process tree for an application running on any of the nodes.

> **Note:**
>
> If the Appmon code is not available at a node, for example an embedded node, this node is ignored by Appmon and is not shown in the Appmon window.

1.1.2 Getting Started with Appmon

Start Appmon by calling `appmon:start()`. It will start the *main window* showing a load meter and the applications running at the current node. By clicking on one of the applications a window showing the process tree of the application will be opened, the *application window*.

The main window is equipped with a menubar from which it is possible to:

- exit Appmon
- perform some operations on the node
- set how information should be displayed
- select which node to show
- open help (this document).

The application window shows the process tree for an application with each process displayed as a box. It is possible to view information about the processes, to send messages to them, and to trace and kill them.

1.1.3 The Main Window

The main window shows a load meter and all applications running at the displayed node. Select which node to display in the window by choosing the node name from the Nodes menu. It is also possible to run Appmon in a many-window mode where a new instance of the main window is opened for each node to be displayed.

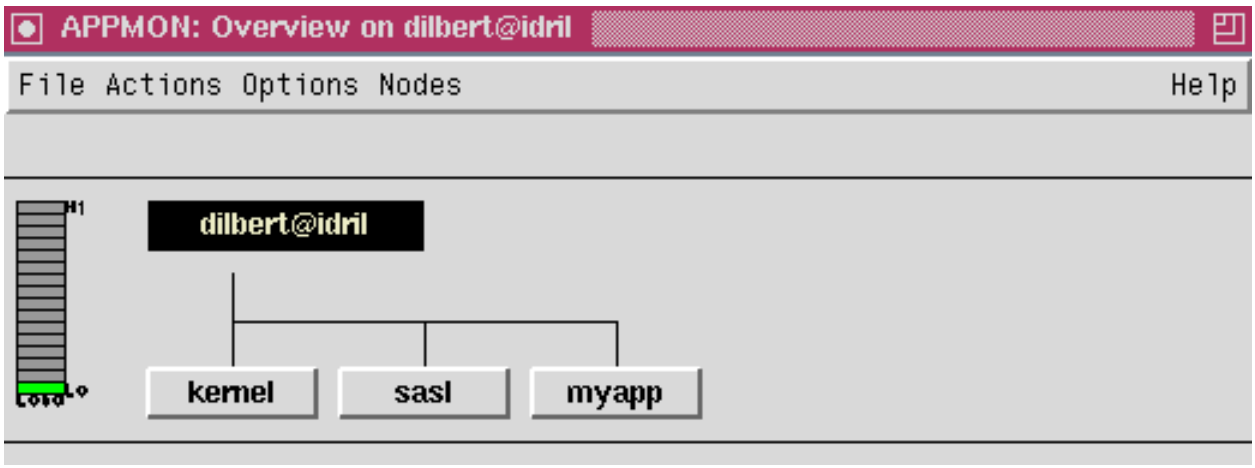

Figure 1.1: The Main Window.

The load meter shows load measured as processor time, or as the length of the ready queue.

Every application running at the node is shown as a button. Clicking the button will open the application window showing the process tree for the application.

The File Menu

Show List Box...
: This will open the *listbox window* which lists all nodes and applications. This window can be more easy to use than the normal, graphical user interface when the system consists of a large number of nodes and/or applications.

Close
: Close the window. If no other instance of the main window exists, Appmon will be stopped.

Exit
: Stop Appmon.

The Actions Menu

Reboot
: Call `init:reboot()` at the currently displayed node. This will stop the node. If the `-heart` system flag was given, the heart program will try to reboot the system.

Restart
: Call `init:restart()` at the currently displayed node. This will restart the node.

Stop
: Call `init:stop()` at the currently displayed node. This will stop the node.

Ping
: Call `net:ping(Node)` where `Node` is the currently displayed node. This can be useful when the connection to the node has been lost.

The Options Menu

One window/Many windows
: Select one of these radio button to run Appmon in one-window or many-window mode. In many-windows mode, a new instance of the main window is opened for each node to be displayed. The default value is one-window mode.

Load: time/queue
: Select one of these radio buttons to either calculate load as processor time or as the length of the ready queue, which is the number of processes ready to execute. By default, the load is calculated as processor time.

Load: progressive/linear

Select one of these radio buttons to show load either according to a progressive or a linear scale. By default, the load is shown according to a progressive scale.

The Nodes Menu

The Nodes menu contains all currently and previously known nodes, where the Appmon code is available. Nodes where the Appmon code is not available, for example embedded nodes, are ignored by Appmon and are not shown.

Selecting a node from the Nodes menu will cause that node to be displayed in either the same window, or in a new window, depending on if Appmon is run in one-window or many-windows mode. If the connection to the node has been lost, this will be shown in the window.

The Help Menu

Help

Selecting Help from the Help menu will cause the HTML version of the Appmon User's Guide (this document) to be displayed. Currently this function requires Netscape to be up and running.

1.1.4 The Application Window

The application window shows the process tree for an application. The window title contains the application name and the node name. The window also contains a menubar and a toolbar.

The application window can be opened from the main window by clicking on the button denoting the application, or from the listbox window by selecting the application and clicking on the *Load* button.

Figure 1.2: The Application Window.

The application can be shown either as a strict supervision tree, or as a process view with all linked processes. In supervision mode, the tree-gathering and -building algorithm assumes conformance to the OTP design principles.

The File Menu

Close
Close the application window.

The Options Menu

Refresh
Refresh the application window.

Sup. view/ Proc. view
Select one of these radio buttons to show the application as a strict supervision tree, or as a process view with all linked processes. By default, the process view is used.

The Toolbar

The toolbar consists of four buttons: *Info*, *Send*, *Trace* and *Kill*. First select one of these buttons and then select to which process the action should apply by clicking on a process in the process tree. By default *Info* is selected which means that clicking on a process, without selecting *Send*, *Trace* or *Kill* first, will open the *process information window*.

Info
: Open the *process information window*, which displays the information about the process given by `process_info(Pid)`.

Send
: Send a message to a process. A window is opened where the message can be specified. Click *Ok* to send the message or *Cancel* to cancel.

Trace
: Switch on OTP tracing of a process. `sys:trace(Pid,true)` is called. Selecting *Trace* a second time for the same process will switch the tracing off instead.

Kill
: Kill a process. `exit(Pid,kill)` is called.

1.1.5 The Listbox Window

The listbox window lists all nodes and applications. This window can be more easy to use than the normal, graphical user interface when the system consists of a large number of nodes and/or applications.

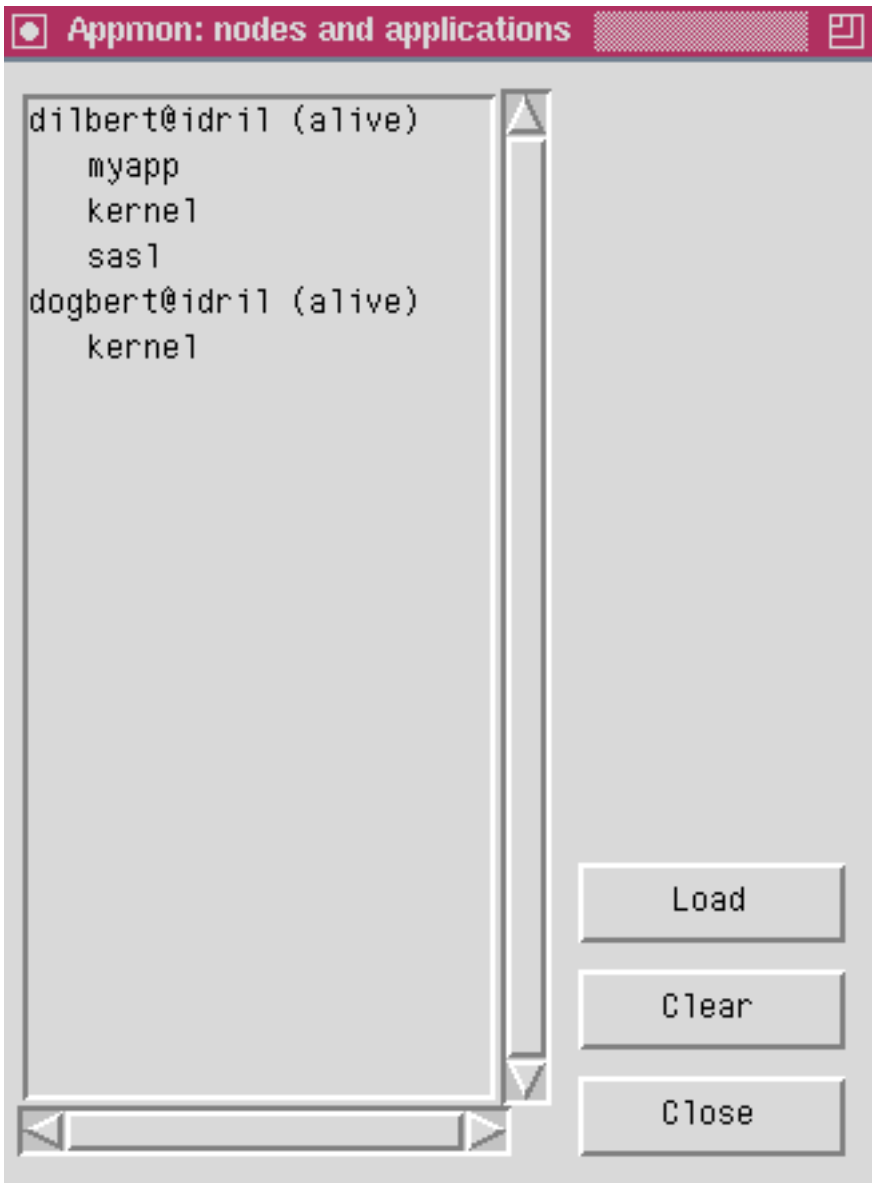

Figure 1.3: The Listbox Window.

The window contains three buttons:

Load
: First selecting an application and then clicking the *Load* button will open the *application window* for that application.

Clear
: De-select any selected node or application name.

Close
: Close the listbox window.

1.1.6 The Process Information Window

The process information window shows information about different processes as given by `process_info(Pid)`.

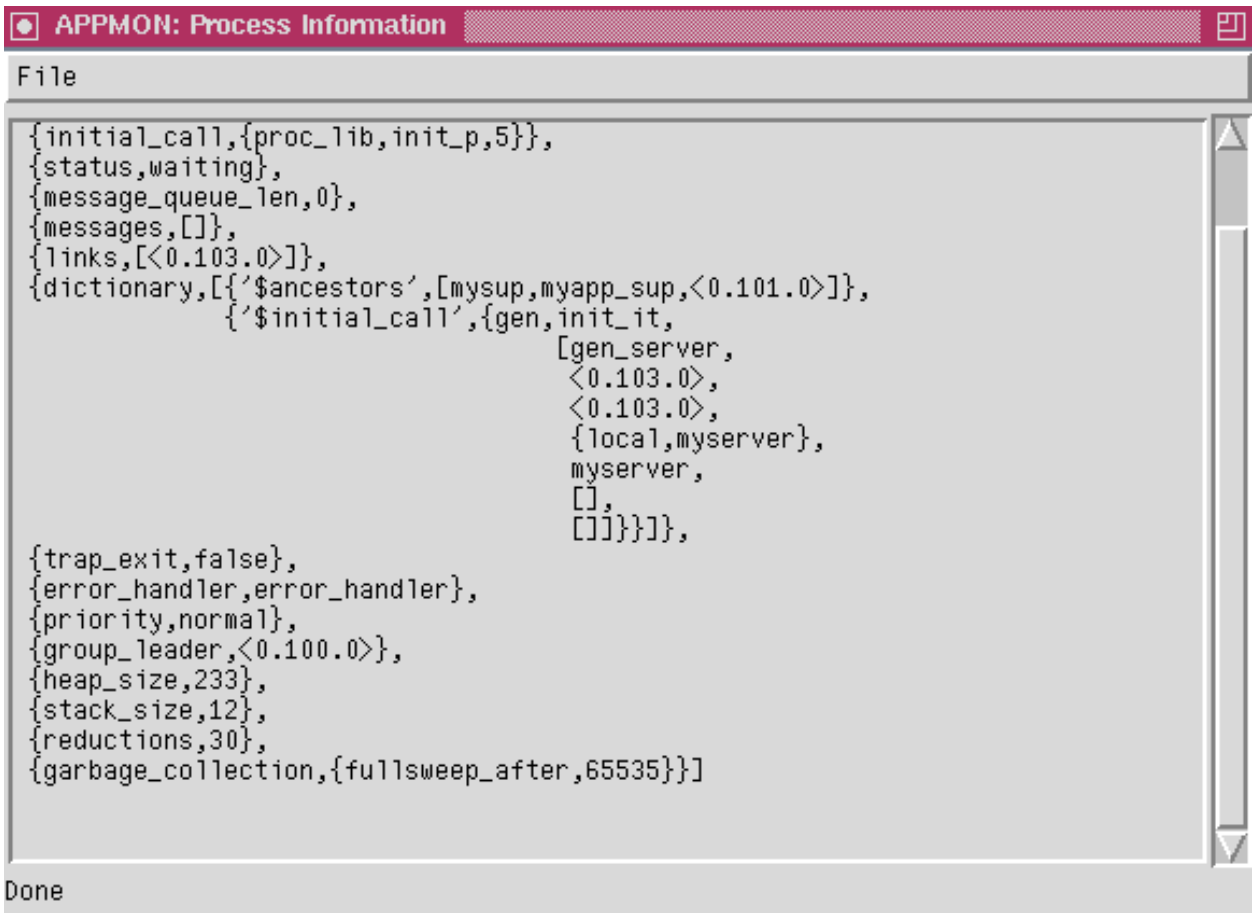

Figure 1.4: The Process Information Window.

The File Menu

Close

Close the process information window.

1.1.7 Using the Web Based version of Appmon

Introduction

The web based version of Appmon is an alternative version of Appmon. The main difference between the web based version of Appmon and the original version of Appmon is that the web based version of Appmon can monitor nodes and applications on nodes where Appmon not is installed.

Start the Web Based version of Appmon

To start the web based user interface configure and start WebTool, see *WebTool User's Guide*. The web based version of Appmon is compatible with the browsers Netscape Navigator and Internet Explorer 4.0 and higher.

When WebTool is started, start the Web based version of Appmon via WebTool. If WebTool succeeded to start the web based version of Appmon a link named WebAppmon will appear in the topmost frame. Click on the link and the main frame of the browser will show two frames.The left frame will show a combo box for node selection and a list of all applications on the currently selected node.

In the left frame it is possible to:

- Select which node to supervise.
- Select an application to view its process tree.
- Select an application to view its specification.

The right frame shows the selected information, either the application specification or the process tree and process information.

Selecting a Different Node

In the top of the left frame there is a combo box with all known nodes. The name of the node that is monitored is the node whose name is visible in the combo box. At startup the node on which the web based version of Appmon is running on will be monitored.

To change node, select another node in the combo box, and the list of running applications will change to the applications that runs on the selected node.

Viewing the Process Tree of an Application

To view the process tree of an application click on the application name in the list of applications. The process tree of the selected application will then appear in the right frame.

The processes in the process tree can have three different colors:

Blue
: The relation to the process above in the process hierarchy is a primary relation. That means that the process above is the only process that link to it or the process above is a supervisor.

Red
: The relation to the process above in the process hierarchy is a secondary relation. This means that more than one process has a relation to it and the process above in the hierarchy is not a supervisor.

Black
: The process isn't a regular process instead it's a Port. The name of the process will also begin with *Port:*

If the process name begins with *Runs on another node:* The monitored application runs on more than one node and this process is the first process on a branch that runs on another node.

Viewing the Application Specification

After the application name in the list of applications there are a link named *Spec*, Click on this link and the application specification will appear in the right frame. The application specification is the data in the *.app* file of the selected application.

Select Process View

In the top of the page that shows the process tree, there are three radio buttons for selecting which processes that shall be included in the process tree. The default mode is that all processes that one process in the application tree has a relation to is included. It's possible to filter which of the processes in the process tree that will be included. To reduce the number of processes in the process tree select one of the radio buttons on the top of the page

All processes
: All processes that at least one process in the application tree has a link to.

Supervised processes
: All processes that are supervised by one of the supervisors in the application

Supervisors only
: Only the processes that actually are supervisors in the applications supervision tree is included

Process Information

To see more information about a specific process click on the process name and more information will be loaded under the application tree.

The Process information page shows information about a process as given by the `process_info(Pid)`, but formatted in a more human readable form.

Trace a process

If the link after the process name in the process tree is *start trace* the process is not traced. If the link is *stop trace* then the process id traced.

To toggle the trace flag for a process click on the link *start trace* or *stop trace* after the process name in the process tree.

2 Reference Manual

The Application Monitor, *Appmon*, is a graphical utility used to supervise applications executing either locally or on remote nodes. The process tree of an application can furthermore be monitored.

appmon

Erlang module

> ## Warning:
>
> The Appmon application has been superseded by the Observer application. Appmon will be removed in R16.

The application monitor Appmon is a graphical utility used to supervise applications executing either locally or on remote nodes. The process tree of an application can furthermore be monitored.

Exports

`start()`

Starts Appmon.

`stop()`

Stops Appmon.

See Also

Appmon User's Guide