

0.1 Configuration

The `HardwareEmulator` and the `EmulatorManager` support a simple fail-safe way of configuring them and making queries to them. A message is made of a `String` where each line (separated by a newline character) is a name followed by optional parameters.

The syntax is whitespace insensitive and as follows:

```
name = param1, param2, param...
or
name
```

Whitespace is optional, and the only separating characters are the equal sign (=), the comma (,) and the newline character (\n).

0.1.1 Configuration

The current settings supported when allocating an emulator are shown in Table 1.

<code>autopilot=name</code> Name of the autopilot program file, without extension.
<code>os=name</code> Name of the operating system used. Currently <code>linux</code> or <code>windows</code> . This implies the extension of the autopilot file searched. If the <code>os</code> command is not given, the <code>Emulator</code> will search for any autopilot with the given name in the search folder.
<code>debug=flag1, flag2, ...</code> Enable the debugging outputs for the specified flags, where the flags are any of <code>mem</code> , <code>regs</code> , <code>reg.update</code> , <code>syscalls</code> , <code>unsupported.syscalls</code> , <code>code</code> or <code>call</code> .
<code>cpu_frequency=value</code> The CPU frequency in Hertz used to evaluate the computation time.
<code>memory_frequency=value</code> The RAM frequency in Hertz used to evaluate the computation time.
<code>cache_name=size, read_ticks, write_ticks</code> Where <code>name</code> is <code>IL1</code> , <code>DL1</code> , <code>L2</code> or <code>L3</code> . This setting enables and sets the parameters of the named cache layer. Set <code>size</code> to 0 to disable this cache level. The ticks refer to the CPU ticks.
<code>no_time</code> Disables the delaying of the autopilot in the simulation.
<code>test_real</code> If the <code>RMIModelServer</code> and the autopilot are of the same operating system type, this flag makes the emulator load the autopilot directly as a library (not emulated) and uses its outputs to validate those of the emulated instance.
<code>export</code> Enables the <code>HardwareEmulator</code> to export the deviation from the planned trajectory and the evaluated execution time of the autopilot. Data is currently exported for the first 60 seconds of the simulation. The output files are <code>autopilot_dist.txt</code> and <code>autopilot_time.txt</code> .

Table 1: `HardwareEmulator` Configuration

The `EmulatorManager` itself also supports a config string on initialization. Table 2 shows the supported config.

<pre>autopilots_folder=folder</pre> <p>Sets the folder in which to look for autopilot programs.</p>

Table 2: EmulatorManager Configuration

0.1.2 Queries

The query system uses the same presented syntax, but also sends a response for every query, using the same syntax.

Queries supported by the `EmulatorManager` are shown in Table 3.

<i>Query</i>	<code>get_error_msg</code>
<i>Response</i>	<code>error_msg=msg</code> Returns the last error message that occurred while allocating an autopilot emulator.
<i>Query</i>	<code>get_available_autopilots</code>
<i>Response</i>	<code>available_autopilots=autopilot1, autopilot2, ...</code> Returns the list of autopilot files available in the autopilot folder.
<i>Query</i>	<code>get_available_threads</code>
<i>Response</i>	<code>available_threads=thread_count</code> Returns the number of concurrent threads the machine running the RMI-ModelServer supports.
<i>Query</i>	<code>get_autopilots_folder</code>
<i>Response</i>	<code>autopilots_folder=folder</code> Returns the current autopilots folder.

Table 3: EmulatorManager queries.

Queries supported by querying a specific emulator are shown in Table 4.

<i>Query</i>	<code>get_avg_runtime</code>
<i>Response</i>	<code>avg_runtime=time</code> Returns the mean average time of autopilot execution.
<i>Query</i>	<code>is_computing</code>
<i>Response</i>	<code>computing=1 or 0</code> Returns 1 if the computer is considered to be computing in simulation time.
<i>Query</i>	<code>get_computer_time</code>
<i>Response</i>	<code>computer_time=time</code> Returns the time counter of the computer, in microseconds.

Table 4: HardwareEmulator queries.