





# graphviz-dot— mode: Graphviz Dot Mode

Description	Keystroke	Function	Note
<b>Graphviz Dot</b>	With an external package installed (see below) Emacs can support rendering Graphviz-Dot markup files and can also render the resulting graph file when Emacs is running in graphics mode.   This requires the <a href="#">graphviz-dot-mode package</a>  PEL downloads, installs and activates it when the <a href="#">pel-use-graphviz-dot</a> user option is set to t.		
<b>Open this PDF file.</b> See also: <a href="#">Help/Info</a>	<ul style="list-style-type: none"><li>• <b>&lt;f11&gt; SPC g &lt;f1&gt;</b></li><li>• <b>&lt;f12&gt; &lt;f1&gt;</b></li></ul>	( <a href="#">pel-help-pdf</a> &optional OPEN-WEB-PAGE)	Open the local copy of the <a href="#">Graphviz Dot</a> PDF file unless a command prefix (like <b>C-u</b> ) was used. In that case it opens the Github-hosted file instead.
<b>Customize PEL Graphviz Dot Support</b> See also: <a href="#">Customize</a>	<ul style="list-style-type: none"><li>• <b>&lt;f11&gt; SPC g &lt;f2&gt;</b></li><li>• <b>&lt;f12&gt; &lt;f2&gt;</b></li></ul>	( <a href="#">pel-customize-pel</a> &optional OTHER-WINDOW)	Customize PEL Graphviz-Dot support. <ul style="list-style-type: none"><li>• If OTHER-WINDOW is non-nil (use <b>C-u</b>), display in another window.</li></ul>
<b>Customize Emacs Graphviz Dot Support</b> See also: <a href="#">Customize</a>	<ul style="list-style-type: none"><li>• <b>&lt;f11&gt; SPC g &lt;f3&gt;</b></li><li>• <b>&lt;f12&gt; &lt;f3&gt;</b></li></ul>	( <a href="#">pel-customize-library</a> &optional OTHER-WINDOW)	Customize Emacs Graphviz-Dot support. <ul style="list-style-type: none"><li>• If OTHER-WINDOW is non-nil (use <b>C-u</b>), display in another window.</li></ul>
<b>Activate Graphviz Dot mode</b>	<b>&lt;f11&gt; M-g</b>	( <a href="#">graphviz-dot-mode</a> )	Toggle use of the graviz-dot-mode to edit Graphviz Dot files.
<b>Editing Dot syntax</b>	The following commands help writing Graphviz DOT code. See the following reference documents: <ul style="list-style-type: none"><li>• <a href="#">Documentation</a>, which contains the following and much more:<ul style="list-style-type: none"><li>• <a href="#">The DOT Language</a></li><li>• <a href="#">Node, Edge and Graph Attributes</a></li><li>• <a href="#">Node Shapes</a></li><li>• <a href="#">Arrow Shapes</a></li><li>• <a href="#">Colors</a></li></ul></li></ul>		
<b>Indent current line</b> See also: <a href="#">Indentation</a>	<b>&lt;tab&gt;</b>	( <a href="#">indent-for-tab-command</a> &optional ARG)  ( <a href="#">graphviz-dot-indent-line</a> )	Indent current line of dot code (regardless of point position within the line)
<b>Indent the graph statement at point</b>	<ul style="list-style-type: none"><li>• <b>C-M-q</b></li><li>• <b>&lt;f12&gt; &lt;tab&gt;</b></li></ul>	( <a href="#">graphviz-dot-indent-graph</a> )	
<b>Break line and indent</b>	<b>M-j</b>	( <a href="#">indent-new-comment-line</a> &optional SOFT)	Break line at point and indent, continuing comment if within one. <ul style="list-style-type: none"><li>• This indents the body of the continued comment under the previous comment line.</li><li>• This command is intended for styles where you write a comment per line, starting a new comment (and terminating it if necessary) on each line.</li><li>• If a fill column is specified, it overrides the use of the comment column or comment indentation.</li><li>• The inserted newline is marked hard if variable ‘use-hard-newlines’ is true, unless optional argument SOFT is non-nil.</li></ul>
<b>Compiling</b>			
<b>Compile the graph</b>	<ul style="list-style-type: none"><li>• <b>C-c C-c</b></li><li>• <b>&lt;f12&gt; c</b></li></ul>	(compile COMMAND &optional COMINT)	Compile the graph using the program identified by graphviz-dot-dot-program to the output format identified by graphviz-dot-preview-extension. The command is shown in the mini buffer, allowing modification before execution.
<b><a href="#">Jump to next match</a></b>	<ul style="list-style-type: none"><li>• <b>C-x `</b></li><li>• <b>M-g n</b></li><li>• <b>M-g M-n</b></li></ul>	( <a href="#">next-error</a> &optional ARG RESET)	A prefix ARG specifies how many error messages to move; negative means move back to previous error messages. Just C-u as a prefix means reparse the error message buffer and start at the first error.
<b><a href="#">Jump to previous match</a></b>	<ul style="list-style-type: none"><li>• <b>M-g p</b></li><li>• <b>M-g M-p</b></li></ul>	( <a href="#">previous-error</a> &optional N)	Prefix arg N says how many error messages to move backwards (or forwards, if negative).
<b>Viewing</b>			
<b>Preview image in Emacs buffer</b>	<ul style="list-style-type: none"><li>• <b>C-C C-p</b></li><li>• <b>&lt;f12&gt; p</b></li></ul>	( <a href="#">graphviz-dot-preview</a> )	Compile the graph and preview it in an other buffer.   Only works in graphics mode.
<b>View image using external program</b>	<b>C-c C-v</b>	( <a href="#">graphviz-dot-view</a> )	Run an external viewer. <ul style="list-style-type: none"><li>• This creates an external process every time it is executed. If ‘graphviz-dot-save-before-view’ is set, the current buffer is saved before the command is executed.</li></ul>
<b>Graphviz Dot Variables</b>			
<a href="#">graphviz-dot-dot-program</a>	Program used to compile the graphs. Default := ‘dot’.		
<a href="#">graphviz-dot-preview-extension</a>	File type to use for output. Default := ‘png’.		
<a href="#">graphviz-dot-view-command</a>	Command to run when ‘graphviz-dot-view’ is executed. Default:= ‘dotty %s’		
<a href="#">graphviz-dot-view-edit-command</a>	If the user should be asked to edit the view command. Default:= nil		
<a href="#">graphviz-dot-save-before-view</a>	Automatically save current buffer berore ‘graphviz-dot-view’.		
<a href="#">graphviz-dot-mode-hook</a>	Hook variable for initialization of the mode.		

## Graphviz Dot Mode — References

Description & URL	Notes
<b>Graphviz Dot</b>	
<b>GraphViz @ Wikipedia</b>	Overview description with various links.
<b>GraphViz.org - Graphviz Graph Visualization Software</b>	Graphviz home page. Includes: <ul style="list-style-type: none"><li>• <a href="#">Gallery</a></li><li>• <a href="#">Documentation</a>, which contains the following and much more:<ul style="list-style-type: none"><li>• <a href="#">The DOT Language</a></li><li>• <a href="#">Command-line Usage</a></li><li>• <a href="#">Output formats</a></li><li>• <a href="#">Node, Edge and Graph Attributes</a></li><li>• <a href="#">Node Shapes</a></li><li>• <a href="#">Arrow Shapes</a></li><li>• <a href="#">Colors</a></li></ul></li><li>• <a href="#">Resources</a></li></ul>
<b>Graphviz Dot Mode for Emacs</b>	
<b><a href="#">graphviz-dot-mode @ GitHub</a></b>	