Chime Embed Tags

Tag name: **`altscript`**

Description: Specifies RasMol commands to execute when a button goes from a "pushed" state to an "unpushed" state. Multiple commands can be separated with "|" or ";". This tag applies only to toggle or radio buttons, as specified in the **`button`** tag.

Syntax: **`altscript={RasMol script commands}`**

See also: `button, buttonstate, script`

Tag name: **`animfps`**

Description: Specifies the animation events (frames) per second

Note: The value specified for **`animfps`** may not be possible on the viewer's platform. Chime will get as close as the system allows.

Syntax: **`animfps={animation events per second}`**

See also: `AnimFrameCallback, animmode, staranim`

Tag name: **`AnimFrameCallback`**

Description: Specifies the name of a JavaScript function in the current document that is to be called before each events ("frame") of an animation (.xyz) file is displayed. The function should conform the prototype:

`function JSFunctionName` *`(pluginName, frameNumber)`*

Where *`pluginName`* is the value specified in the **`name`** parameter for the Chime structure window in which the animation is playing and the *`frameNumber`* is an integer indicating which number event in the file is about to be displayed.

Syntax: **`AnimFrameCallback={JSFunctionName}`**

See also: `animfps, animmode, startanim`

Tag name: **`animmode`**

Description: Specifies how the animation is to be presented

Syntax: **`animmod={loop|once|palindrome|ping|pong}`**

`loop` - Play the animation from the first frame to the last and then start again from the first

`once` - Play the animation from the first frame to the last and then stop.

`palindrome` - Play the animation from the first frame to the last and then backwards from the last to the first, continuously

`ping` - Synonym for **`palindrome`**

pong - Synonym for **palindrome**

See also: animfps, AnimFrameCallback, startanim

Tag name: **atommapnum2d**

Description: An atom-atom map on the reaction components specifies exactly which atoms in the reactants correspond to the atoms in the products. Applies to reactions in 2D

Syntax: **atommapnum2d={off|on}**

off - do not display atom-atom maps

on - display atom-atom maps

See also: invretmarks2d, reactingcenters2d

Tag name: **atomnum2d**

Description: Specifies whether to display atom numbers

Syntax: **atomnum2d={on|off}**

on - display atom numbers

off - do not display atom numbers.

See also: bondscale2d, bondlen2d, fontsize2d, fontname2d, hlabels2d

---

Tag name: **bgcolor2d**

Description: Sets the background color for 2d rendering

Syntax: **bgcolor={black|white|#rrggbb}**

black - make the background black

white - make the background white

#rrggbb - set the background to a specific color using an HTML-style color value. For example: #777777 gives a dark gray

See also: color3d, palette

Tag name: **bgcolor3d**

Description: Sets the background color for 3d rendering

Syntax: **bgcolor={black|white|#rrggbb}**

black - make the background black

white - make the background white

#rrggbb - set the background to a specific color using an HTML-style color value. For example: #777777 gives a dark gray

See also: color2d, palette

Tag name: **bondlen2d**

Description: Specifies the standard bond length, in decipoints, used to display a 2D image when using the tag **bondscaling=stdbond**. The default length is 180

Syntax: **bondlen2d={###}**

See also: bondscale2d

Tag name: **bondscale2d**

Description: Specifies the type of scaling to use when displaying a 2D image

Syntax: **bondscale2d={asdrawn|asdrawn_fitbox|fixbox|stdbond}**

asdrawn - display the image exactly as it was drawn, with the same bond lengths and coordinates.

asdrawn_fitbox - display the image with the same relative bond lengths as were drawn, but scale the whole image to best fit the Chime structure window.

fitbox - display the image to best fit the Chime structure window, and make all bonds equal in length.

stdbond - set all bond lengths to the standard bond length. The standard bond length is specified by the **bondlen** tag.

See also: atomnum2d, bondlen2d, fontsize2d, fontname2d, hlabels2d

---

Tag name: **button**

Description: Display the Chime plug-in as a button. Three types of buttons are available: push, radio and toggle.

When the user clicks a button, the script attached to the button through the **script** or **csml** tag executes. The script acts on the Chime plug-in indicated by the **target** tag. A Chime button can also trigger a JavaScript routine through the **ButtonCallback** tag.

In the case of toggle and radio buttons, a second script can be activated when the button goes from a "pushed" to an "unpushed" state. This script is specified by the **altscript** tag Because a Chime button does not display a file, you must specify the mime type manually using the type tag: type="application/x-spt", as shown in the example

Syntax: **`button={push|radio#|toggle}`**

`push` - a simple push-button.

`radio#` - a member of a radio group. Only one member of the group is pushed at any given time. The group is identified by the number attached to the tag, for example "radio1", "radio2".

`toggle` - a toggle button which alternates between a "pushed" state and an "unpushed" state, similar to check box.

See also: `alscript, ButtonCallback, buttonstate, csml, immediate, target`

Tag name: **`buttonstate`**

Description: Used to specify that a button's initial state is "pushed". If this tag is not used, the button's initial state will be "unpushed".

Syntax: **`buttonstate={pushed}`**

See also: `altscript, button`

Tag name: **`ButtonCallback`**

Description: This tag is only be used in a Chime button(**`button`**`=true`). It specifies a JavaScript function in the current document that is to be called whenever the button is pushed. The function is called twice, once before execution of the attached RasMol or CSML script, and a second time afterwards. If there is no attached script, the function is still called twice.

The JavaScript function must conform to the prototype:

`function JSFunctionName(`*`pluginName, executedYet?`*`)`

...where *`pluginName`* is the name of the Chime button which the user has pressed, and the boolean *`executeYet?`* is *`false`* when *`JSFunctionName`* is first called (before execution of the script) and the *`true`* the second time.

With the LiveConnect interface in Chime, you can start RasMol scripts from a JavaScript routine when running in Netcape Navigator, as shown in the example below. In other browsers, you can achieve the same result by writing a new button with the **`immediate`** tag. You can also see an example of the LiveConnect interface in "Using LiveConnect" under the Tutorials section.

Syntax: **`ButtonCallback={JSFunctionName}`**

See also: `button, csml, immediate, script`

Tag name: **`color2d`**

Description: Specifies the forground color for a 2D rendering.

Syntax: **`color2d={black|sketch|white|#rrggbb}`**

`black` - Sets the foreground color to black

sketch - Sets the foreground color to the colors specified in the sketch. Use this option if you want to display:

- A sketch (`.skc`) file, or
- The highlighted atoms, bonds, and Sgroup brackets in a molfile. The highlight color is blue.
  Note: Highlighting is supported only in 2D. If the structure is not in 2D display, set the `display2d` embed tag to `true`

white - Sets the foreground color to white

#rrggbb - Set the foreground color to a specific color using an HTML-style color value. For example:#777777 sets the foreground color to dark gray.

See also: `bgcolor, display2d`

Tag name: **color3d**

Description: Specifies the color scheme for 3D display

Syntax: **color3d={chain|cpk|group|monochrome|shapely|structure|temperature|user}**

The options correspond to those available in the Chime menu.

See also: `bgcolor, display3d, option3d, palette, scale3d`

Tag name: **csml**

Description: Specifies any valid CSML script commands to apply to the plug-in. Multiple commands can be separated with a "|" or ";"

When attached to a Chime structure window, the script executes immediately when the plug-in loads. When attached to a Chime button (**button**=`true`), the script executes in the Chime structure window indicated by the **target** tag when the button is pushed

Syntax: **csml={valid CSML script commands}**

See also: `button, immediate, script, target`

Tag name: **debugscript**

Description: When **debugscript**=`true`, each line of an executing Rasmol script is echoed to the browser's status line.

Syntax: **debugscript={false|no|true|yes}**

fasle or **no** - do not echo script commands

true or **yes** - echo script commands to status line

See also: `script`

Tag name: **display3d**

Description: Specifies the type of 3D display

Syntax: **`display3D= {backbone|ball&stick|cartoons|ribbons|spacefill|sticks|strands|wireframe}`**

The options correspond to those available in the Chime menu.

See also: `color3d, display2d, options3d, scale3d`

Tag name: **`expandResidue2d`**

Description: Specifies wether to display expanded or contracted residues in Chime Pro is the contracted (in Chime Pro 2.0 the default was the expanded form).

Syntax: **`expandResidue2d = {on|off}`**

`on` - Display expanded residues

`off` - Do not display expanded residues (default)

See also: (none)

Tag name: **`fontname2d`**

Description: Sets the font used to display atom symbols and other text in 2D display. The default in Window is Arial.

Syntax: **`fontname2d={font name}`**

See also: `atomnum2d, bondscale2d, bondlen2d, fontsize2d, hlabels2d`

Tag name: **`fontsize2d`**

Description: Sets the font size, in decipoints, used to display atom symbols and other text in 2D display. The default size is 120.

Syntax: **`fontsize2d={###}`**

See also: `atomnum2d, bondscale2d, fontname2d, hlabels2d`

Tag name: **`frank`**

Description: When **`frank`** = `true`, the "MDL" trademark is displayed in the lower right corner of the Chime structure window

Syntax: **`frank ={false|no|true|yes}`**

See also: (none)

Tag name: **`hbonds2d`**

Description: Specifies whether, and how, to display hydrogen bond.

Syntax: **hbonds2d={off|on|number}**

off - do not display hydrogen bonds

on - display hydrogen bonds as dashed lines

number -display hydrogen bonds as cylinders whose diameter is proportional to the number. See example

See also: ssbonds

Tag name: **hlabels2d**

Description: Specifies how to display implicit hydrogen in a 2D structure

Syntax: **hlabels2d={asdrawn|false|hetero|terminalhetero|true}**

asdrawn - display hydrogens as originally drawn

false - display no implicit hydrogens

hetero -display implicit hydrogens only on hereoatoms

terminalhetero - display implicit hydrogens only on terminal and hetero atoms

true - display implicit hydrogens on all atoms

See also: atomnum2d, bondscale2d, bondlen2d, fontsize2d, fontname2d

---

Tag name: **immediate**

Description: Specifies whether the script attached to a Chime button should be run automatically when the Chime is loaded. This feature allows JavaScript to communicate with Chime by writting a Chime button (to an unseen part of the screen) that will execute immediately to perform some operation on a Chime structure window

NOTE: Chime supports a LiveConnect interface in Netscape Navigator, providing an easier way to run RasMol scripts from JavaScript. See **ButtonCallback** and **button** for an example. You can also see an example of the LiveConnect interface in "Using LiveConnect" under the Tutorials section.

Syntax: **immediate={false|no|true|yes}**

false or **no** - do not run script immediately. Script will run only when button is pushed

true or **yes** - run script immediately upon loading Chime

See also: button, csml, preloadscript, script, target

Tag name: **invretmarks2d**

Description: Inversion marks specify that a stereo center must be inverted by the reaction. Retention marks specify that a stereo center must retain its configuration. Applies to reactions in 2D

Syntax: **invretmarks2d={off|on}**

off - do not display the stereochemical changes that are apparent in reactions

on - display the stereo chemical changes that are apparent in reactions

See also: atommapnum2d, reactingcenters2d

Tag name: **jcamp_grid**

Description: Specifies whether to display JCAMP grid

Syntax: **jcamp_grid={false|true}**

false - do not display JCAMP grid

true - display JCAMP grid

See also: jcamp_revplot, jcamp_help

Tag name: **jcamp_help**

Description: Specifies whether to turn on/off JCAMP interpret mode

Syntax: **jcamp_help={false|true}**

false - set interpret off

true - set interpret on

See also: jcamp_grid, jcamp_revplot

Tag name: **jcamp_revplot**

Description: Specifies whether to reverse JCAMP plot

Syntax: **jcamp_revplot={false|true}**

false - do not reverse JCAMP plot

true - reverse JCAMP plot

See also: jcamp_grid, jcamp_help

Tag name: **LoadStructCallback**

Description: Specifies the name of a JavaScript function in the current document that is to be called whenever the user pastes or transfers a structure or opens a file into a Chime Pro query form box. The function should conform to the prototype:

```
function JSFunctionName(pluginName)
```

where `pluginName` is the **`name`** specified for the Chime structure window.

Syntax: **`LoadStructCallbback={JSFunctionName}`**

Note that if you are using frames you should explicitly specify the document containing the function, for example:

```
LoadStructCallback=parent.frameName.MyLoadStructCallback
```

See also: (none)

Tag name: **`messages3d`**

Description: If `true`, messages generated by normal operation of Chime's 3D rendering engine are echoed to the browser's status line

Syntax: **`messages3d={false|no|true|yes}`**

`false` or **`no`** - do not echo messages

`true` or **`yes`** - echo messages to status line

See also: `MessageCallback`

Tag name: **`MessageCallback`**

Description: Specifies a JavaScript function in the document that is to be called whenever an informational message is generated by Chime's 3D rendering processes. The function should conform to the prototype:

```
function JSFunctionName(pluginName, messageText)
```

where `pluginName` is the **`name`** specified for the Chime structure window and `messageText` is the informational message from Chime. Note that if you are using frames you should explicitly specify the document containing the function, i.e., specify:

```
MessageCallback=parent.frameName.MyCallBack
```

Syntax: **`MessageCallback={JSFunctionName}`**

See also: `message3d`

Tag name: **`name`**

Description: Specifies the name of the plug-in. This is needed to make a Chime the **`target`** of a Chime button. It is also used in JavaScript when executing a script through Chime 2.0's LiveConnect interface. Both techniques are shown in the example below.

Syntax: **`name={name}`**

See also: `button, ButtonCallback, target`

Tag name: **nmrpdb**

Description: Sets the MIME type to the NMR pdb mime type so that "multi-frame" pdb files can be loaded

Syntax: **nmrpdb={false|true|auto}**

false - use default PDB file reading, even if there are MODEL records present

true - treat the PDB file as an NMR model and ignore all CONECT records

auto - treat the PDB file as an NMR model if a MODEL record is found

See also: (none)

Tag name: **nomenus**

Description: Specifies whether to display Chime menus. This feature allows you to turn off Chime's menus except the About menu.

Syntax: **nomenus={true|false}**

true - do not display Chime menus except for the About menu

false - display all Chime menus

See also: (none)

Tag name: **options3d**

Description: Specifies 3D display options. If more than one options3d is specified for the EMBED tag the options will be combined.

Syntax: **options3d={dots|hetero|hydrogen|labels|shadows|slab|specular|stereo}**

The options correspond to those available in the Chime menu

See also: `color3d, display3d, scale3d`

Tag name: **palette**

Description: "Foreground" allows Chime to use the colors it needs outside of the current color palette to smoothly display spacefilling structures. This corresponds to the "Force Palette" command on the Chime menu

Syntax: **palette={background|forground}**

See also: `bgcolor, color3d`

Tag name: **preloadscript**

Description: Specifies a RasMol script to be executed immediately when the plugin is loaded, before any file is loaded from the **src** tag. Multiple commands can be separated with a "|" or ";".

Syntax: **`preloadscript={RasMol script commands}`**

See also: `immediate, script`

Tag name: **`PauseCallback`**

Description: Specifies the name of a JavaScript function in the current document that is to be called whenever a `pause` command is encountered in a RasMol script. The function is called twice: once when the `pause` is first encountered and again when the script resumes. The function must conform to the prototype:

`function JSFunctionName(`*`pluginName, begPause?`* `)`

...where *`pluginName`* is the **name** of the Chime in which the script is running, and the boolean *`begPause?`* is `true` when the `pause` is first encountered and `faulse` when execution of the script resumes

Syntax: **`PauseCallback={JSFunctionName}`**

See also: (none)

Tag name: **`PickCallback`**

Description: Specifies the name of a JavaScript function in the current document that is to be called whenever the user clicks on an atom in a Chime structure window. The function must conform to the prototype:

`function JSFunctionName(`*`pluginName, atomExpression`* `)`

...where *`pluginName`* is the **name** of the Chime structure window and *`atomExpression`* is a string representing the atom (including residue information for a pdb file) on which the user has clicked.

Syntax: **`PickCallback={JSFunctionName}`**

See also: (none)

Tag name: **`reactingcenters2d`**

Description: Reacting centers indicate where specific transformations occur on a reaction. Applies to reactions in 2D

Syntax: **`reactingcenters2d={off|color|thicken|hash}`**

`off` - do not display reacting centers

`color` - display reacting centers in color

`thicken`- display reacting centers thickened

`hash` - display reacting centers with hashes

See also: `atommapnum2d, invretmarks2d`

Tag name: **`scale3d`**

Description: Specifies a scaling value (in Angstroms per inch) to be used when a 3D structure is displayed. When the same **scale3d** value is used for multiple Chime structure windows on a page, the viewer can get a feel for the relative size of the structures, as shown in the example

Syntax: **scale3d={real value}**

The value should be at least 10 angstroms/inch. The higher the scaling value, the smaller the molecule will appear.

See also: color3d, display3d, options3d, scale3d

Tag name: **script**

Description: Specifies any valid RasMol script commands to apply to the plug-in. Multiple commands can be separated with a "|" or ";"

When attached to a Chime structure window, the script executes immediately when the plug-in loads. When attached to a Chime button (**button**=true), the script executes in the Chime structure window indicated by the**target** tag when the button is pushed.

Syntax: **script={valid RasMol script commands}**

See also: altscript, csml, debugscript, immediate, preloadscript, target

---

Tag name: **sendmouse**

Description: Specifies whether to send Chime's mouse events to another target Chime plug-in

Syntax: **sendmouse={false|true}**

false - do not send mouse events to another target Chime plug-in

true - send mouse events to another target Chime plug-in

See also: (none)

Tag name: **spinfps**

Description: Specifies the frame rate per second for spinning. The default is 10

Note: The value specified for **spinfps** may not be possible on the viewer's platform. Chime will get as close as the system allows.

Syntax: **spinfps={spin frame rate / second}**

See also: spinX, spinY, spinZ, startspin

Tag name: **spinX**

Description: Specifies the rotation speed along the X-axis (horizontal axis) in degrees per second. The default is 0 (no spin)

Syntax: **spinX={degrees per second}**

See also: `spinfps, spinY, spinZ, startspin`

Tag name: **`spinY`**

Description: Specifies the rotation speed along the Y-axis (vertical axis) in degrees per second. The default is 30

Syntax: **`spinY={degrees per second}`**

See also: `spinfps, spinX, spinZ, startspin`

Tag name: **`spinZ`**

Description: Specifies the rotation speed along the Z-axis in degrees per second. This is the axis that "comes out of the screen" toward the user. The default is 0 (no spin)

Syntax: **`spinZ={degrees per second}`**

See also: `spinfps, spinX, spinY, startspin`

Tag name: **`src`**

Description: Specifies what file to open and display. This can be a MOL (MDL molecule) file, a PDB (Protein Databank) file, or an XYZ animation file.

In general, if you use the **`src`** tag, you do not need to also specify an embed **`type`**. The embed type is determined automatically

Syntax: **`src={file name}`**

See also: `structure`

Tag name: **`ssbonds`**

Description: Specifies whether, and how, to display disulphide bonds.

Syntax: **`ssbonds={off|on|number}`**

`off` - do not display disulphide bonds

`on` - display disulphide bonds as dashed lines

`number` - display disulphide bonds as cylinders whose diameter is proportional to the number.

See also: `hbonds`

Tag name: **`startanim`**

Description: Set to `true` if you want an animation in a concatenated XYZ file to begin immediately. If `false`, you can start the animation using the RasMol script command `"animation on"`.

Syntax: **`startanim={false|no|true|yes}`**

See also: `animfps, AnimFrameCallback, animmode`

Tag name: **`startspin`**

Description: Set to `true` if you want the structure to start spinning immediately upon loading. If `false`, you can start the spin with the RasMol script command `"spin"`.

Syntax: **`startspin={true|yes|false|no}`**

See also: `spinfps, spinX, spinY, spinZ`

Tag name: **`structure`**

Description: Used instead of **`src`** to load an in-line compressed molfile. The compressed molfile is a compressed, URL-safe encoded version of the structure file. It is generated by Chemscape Server. You must specify the mime type of the plug-in via the **`type`**= tag

Syntax: **`structure={compressed molfile}`**

See also: `src, type`

Tag name: **`target`**

Description: In a Chime button, specifies which Chime structure window should execute the attached RasMol or CSML script. The other Chime is referenced according to its **`name`**.

Syntax: **`target={target Chime name}`**

See also: `button, csml, immediate, name, script`

Tag name: **`type`**

Description: Used to specify the mime type of the plug-in when the **`structure`** = option is used.

Syntax: **`type={Chime_mime_type}`**

See also: `structure`

Tag name: **`display2d`**

Description: Forces a 2D rendering of a molecule that would be rendered in 3D by default. Any molecule with 3D coordinates is automatically rendered in 3D unless this tag is used.

Syntax: **`display2d=true`**

See also: `color2d, display3d`

Tag name: **`exportV3000`**

Description:

Set to `true` if you want to export a Chime structure into a V3000 molfile. By default, when you save a Chime structure to a file, copy it to the clipboard, or transfer it to ISIS/Draw, Chime(or Chime Pro) exports the structure

into a V2000 molfile. The `exportV3000` tag only applies to V3000 molfiles. It does not convert V2000 molfiles into V3000 molfiles.

A V3000 molfile is an extended molfile format that consists of a regular molfile "no structure", followed by a single molfile appendix that contains the body of the connection table (Ctab). For more information about the extended molfile format or the V3000 molfile, see MDL File Formats.

A newer version of the V3000 molfile uses new tags which are not supported by the existing versions of ISIS/Draw (version 2.3 and earlier). Teh new V3000 molfile tags include the `COLLECTION` tag which is used for new features such as structure highlighting. From the Chime (or Chime Pro) plug-in, you can save a V3000 molfile that contains a new V3000 tags by setting exportV3000 to `true`. However, if you transfer or open this V3000 molfile in an existing version of ISIS/Draw, ISIS/Draw cannot disply it. A future release of ISIS/Draw will support these new V3000 tags.

The V2000 molfile is the format that is used in the existing versions of ISIS/Draw. If you set `exportV3000` to `false` or if you do not use `exportV3000`, you can display a V3000 molfile in the Chime(or Chime Pro) plug-in, and transfer the structure to an existing version of ISIS/Draw. If you set `exportV3000` to `true` for a V2000 molfile, the structure remains as a V2000 molfile.

Syntax:

`exportV3000={true|false}`

`true` - Export the structure into a V3000 molfile

See also: (none)
