

cfdiv

Description

Creates an HTML div tag or other HTML container tag and lets you use asynchronous form submission or a bind expression to dynamically control the tag contents.

Category

Ajax

Implemented

From version 3.1.2.009

Usage Syntax

```
[bind="String"]  
[bindOnLoad="Boolean"]  
[id="String"]  
[onBindError="String"]  
[tagName="String"]  
/>
```

[] = Optional attribute

```
bind = "bind expression"  
bindOnLoad = "true|false"  
ID = "HTML tag ID"  
onBindError = "JavaScript function name"  
tagName = "HTML tag name"  
/>  
OR
```

```
ID = "HTML tag ID"  
tagName = "HTML tag name">  
Tag Output Content
```

Contents

- [cfdiv](#)
- [Description](#)
- [Category](#)
- [Implemented](#)
- [Usage Syntax](#)
- [Attributes](#)

- [Binding Syntax](#)

- [Example Usage](#)

- [Bind](#)
- [Refresh a div programmatically](#)

Attributes

Name	Type	Required	Default	Description
bind	String	No		A bind expression that returns the container contents. If you specify this attribute the cfdiv tag cannot have a body. <ul style="list-style-type: none"> • true: executes the bind attribute expression when first loading the tag • false: does not execute the bind attribute expression until the first bound event
bindOnLoad	Boolean	No		To use this attribute, you must also specify a bind attribute.
id	String	No		The HTML ID attribute value to assign to the generated container tag.
onBindError	String	No		The name of a JavaScript function to execute if evaluating a bind expression results in an error. The function must take two attributes: an HTTP status code and a message. To use this attribute, you must also specify a bind attribute.
tagName	String	No	DIV	The HTML container tag to create.

By default, the cfdiv tag creates a div HTML element. You can use standard HTML and CSS techniques to control the position and appearance of the element and its contents.

If a bind is used the value returned is expected to be plain text to be pushed into the cfdiv element body. If value returned represent a json or xml string these will not be parsed.

Binding Syntax

Visit [this page](#) for more about the supported binding syntax.

Example Usage

Bind

```
bind="url:file.cfm?name={myForm:name}&age={myForm:age}"
onBindError="onError" bindonload="false" id="mydiv"/>
```

A div with id mydiv is created and the content will be updated any time the elements indicated in

the bind expression change. Bindonload set to false will prevent the div to be filled on page load.

Refresh a div programmatically

Every tag that use a bind expression can be programmatically refreshed via javascript :

```
Railo.Ajax.refresh('mydiv')
```

This command will refresh the div evaluating the bind expression. The string passed must match the id use by the cfdiv tag.