

SōNet

social networking CMS

Advanced Features

OMEGANETRESEARCH
social networking systems

www.omeganetresearch.com

Table of Contents

PHP File Components	7
<i>Description</i>	7
<i>Syntax</i>	7
Parameters	7
CMS Components	8
<i>Description</i>	8
<i>Syntax</i>	8
Parameters	8
Authenticated Content Block	9
<i>Description</i>	9
<i>Syntax</i>	9
Parameters	9
Group Content Block	10
<i>Description</i>	10
<i>Syntax</i>	10
Parameters	10
Smart Links	11
<i>Description</i>	11
<i>Syntax</i>	11
Parameters	11
Togglers	12
<i>Description</i>	12
<i>Syntax</i>	12
Parameters	12
Document Attributes	13
<i>Description</i>	13
<i>Syntax</i>	13

<i>Attribute Types</i>	13
Pre-configured Scripts	14
<i>Description</i>	14
rss.php	15
<i>Description</i>	15
contact.php	16
<i>Description</i>	16
<i>Accepted URL Parameters</i>	16
<i>contact.properties file</i>	16
processForm.php	17
<i>Description</i>	17
<i>Accepted URL Parameters</i>	17
<i>processForm.properties file</i>	17
Configuration File (config.properties)	18
<i>Description</i>	18
<i>displaySignUpAgreementLink</i>	18
<i>signUpAgreementLinkUrl</i>	18
<i>signUpAgreementLinkLabel</i>	18
<i>requireSignUpApproval</i>	18
<i>contentCreationNotifications</i>	18
<i>masterNotification</i>	18
<i>adminNotification</i>	18
<i>editorNotification</i>	18
<i>findPeopleAuthentication</i>	19
<i>findGroupsAuthentication</i>	19
<i>viewGroupsAuthentication</i>	19
<i>enableRssFeed</i>	19

<i>siteEmailAddress</i>	19
<i>profileFieldsEnabled</i>	19
CMS Rich Content Components	20
<i>Description</i>	20
<i>Syntax</i>	20
Type Options	20
<i>Rich Content Component Types</i>	21
<i>Document Component</i>	21
Parameters	21
Document Component CSS Classes and DIV structure	23
<i>Blog Component</i>	25
Parameters	25
Blog Component CSS Classes and DIV Structure	27
<i>Event Component</i>	28
Parameters	28
Event Component CSS Classes and DIV Structure	29
<i>Announcement Component</i>	30
Parameters	30
Announcement Component CSS Classes and DIV Structure	31
<i>Conversation Component</i>	32
Parameters	32
Conversation Component CSS Classes and DIV Structure	33
<i>Member Component</i>	34
Parameters	34
Member Component CSS Classes and DIV Structure	36
<i>Comment Component</i>	37
Parameters	37
Comment Component CSS Classes and DIV Structure	38

<i>Group Component</i>	39
Parameters	39
Group Component CSS Classes and DIV structure	40
Table 1 - Date Formatting Options	41

PHP File Components

Description

PHP file components allow you to import external PHP files into your document to be run in place and have their output displayed where they were imported into the document at. PHP file components can contain references to other non-PHP components in their output and will be rendered accordingly.

Syntax

```
[[component file="the/path/to/any_valid_filename_here.php"]]
```

Parameters

1. *file* is the path and filename of the PHP code you want to run and display.

Note: It is a good idea to include your PHP code inside a function or class to avoid variable name conflicts.

If you decide to include your PHP code inside a class, please keep in mind that the following class names are reserved by the CMS and using them within your own code will likely causes errors or other problems:

- ComponentLoader
- ConfigReader
- SiteContainer
- UserGroupValidator

CMS Components

Description

CMS Component Blocks are a handy way to help you modularize your design by creating reusable design elements. These reusable design elements can then be imported into other documents within the CMS using the special reference code shown below.

An good example of why you'd want to use such a feature would be for menus. Very often you'll find that the same menu needs to appear at the top or on the side of multiple (if not all) pages throughout your website. While the CMS itself generates menus based on your categories automatically, sometimes you'll want to create links to various documents within other documents and then either assign the documents as categories, use them as subareas, or a variety of other reasons. It's really up to you how they're put to use - they're certainly not limited to just menus.

Syntax

```
[[component id="shortcut"]]
```

Parameters

1. **component id** is the shortcut that was assigned to the document. The document must be registered as a component and published.

You can deactivate a component on all pages where it is referenced by unpublishing it or by unchecking the "This document is a component" checkbox. When a component has been deactivated it is not displayed.

Note: Documents marked as components cannot be accessed directly by non-administrative users. They can only be imported into other documents by the CMS

Authenticated Content Block

Description

Authenticated content allows you to hide blocks of content in your document from people who are not logged.

You could use this feature to hide links to documents that have the *Require authentication to view this document* option enabled.

Syntax

```
[[authenticated_content]]  
mixed content  
mixed content  
mixed content  
[[/authenticated_content]]
```

Parameters

1. *n/a* - this component doesn't have any options. Just place content between the authenticated content tags and it will not be added to the document unless the user is logged in.

Group Content Block

Description

Group content allows you to hide blocks of content in your document from people who are not members of the groups you define within the block's parameters.

You could use this if you've created content in a category that's assigned to a group, but would like to link to that content from a document that is not restricted by a group and/or is in a different category.

Syntax

```
[[group_content groups=""]]  
mixed content  
mixed content  
mixed content  
[[/group_content]]
```

Parameters

1. **groups** are a comma separated list of groups that can view the content contained between the group content tags.

Note: If the groups specified do not have *Allow Viewing* access enabled, the content will not be added to the document when accessed by users in those groups.

Smart Links

Description

Smart Links allow you to dynamically style your links using CSS classes based on document shortcut names. This is accomplished by supplying several parameters that tell the link what to do when the link is located within a specific document.

Syntax

```
[[smartlink activeDocument="shortcut(s)" cssClass="css1" activeCssClass="css2" url="url" linkOnActive="link_active"]]text[/smartlink]
```

Parameters

1. **activeDocument** should contain either a single document shortcut or a comma separated list of document shortcuts. When a smartlink is included in a document whose shortcut matches any of the shortcuts listed within the activeDocument attribute, it will be rendered inside a <div> with the CSS class specified in the **activeCssClass** attribute. If the smartlink is not inside a document specified in **activeDocument** it will be rendered inside a <div> with the CSS class specified in the **cssClass** attribute.
2. **cssClass** is the CSS class that the link will use while displayed inside a <div> when it is not located in a document specified in the activeDocument attribute.
3. **activeCssClass** is the CSS class that the link will use while displayed inside a <div> when it is located in a document specified in the activeDocument attribute.
4. **url** is the URL the link should point to.
5. **linkOnActive** is a true or false attribute that tells the CMS whether or not to make the text a clickable link that points the browser to the URL specified by the url attribute.
6. **text** is the clickable text that is displayed on the page as a link. It can also be displayed as just plain, unclickable text if the linkOnActive attribute is set to false.

Togglers

Description

Togglers allow you to create hidden sections of content whose visibility can be toggled by clicking a link.

Syntax

```
[[togger text="toggle text" id="togger_id" activeDocument="id(s)" cssClassLink="css1"  
activeCssClassLink="css2" cssClassContent="css3" togglerStyle="fade/standard"]]  
mixed content  
mixed content  
mixed content  
[[/togger]]
```

Parameters

1. **text** is the text displayed as a clickable link for displaying and hiding the toggler's content.
2. **id** is a unique identifier for this toggler and its content. The identifier should start with a letter and not contain any non-alphanumeric characters other than a dash or an underscore.
3. **activeDocument** should contain either a single document shortcut or a list of comma separated document shortcuts. This attribute is used to automatically display the contents of the toggler (rather than hide them), based on which document is loaded.
4. **cssClassLink** is the CSS class that the toggler's clickable link will use while displayed inside a <div> when it is not selected and its contents are not displayed.
5. **activeCssClassLink** is the CSS class that the toggler's clickable link will use while displayed inside a <div> when it is selected and its contents are displayed.
6. **cssClassContent** is the CSS class that the toggler content will use while displayed inside a <div>.
7. **togglerStyle** allows you to specify whether the content fades in or "pops" open by supplying "fade" or "standard".
8. **mixed content** is the content that is displayed or hidden when the toggler's clickable link is clicked.

Document Attributes

Description

The attribute tag allows you to display a variety of different attributes related to the currently loaded document. This tag will always display the attributes of the "root document" (i.e. the attribute tag will not display the attributes of a component that's contained inside another document)

HTML entities are always encoded.

Syntax

```
[[attribute type="attribute_type"]]
```

Attribute Types

- **usernameCreated** displays the username of the user who originally created the document.
- **usernameUpdated** displays the username of the user who last updated the document.
- **documentType** displays the document's chosen document type. This will not work with events.
- **category** displays a link to the document list using the current document's category.
- **subcategory** displays a link to the document list using the current document's subcategory.
- **subject** displays a link to the document list using the current document's subject.
- **ratingGraphic** displays the rating in a div using the class **rating_graphic** with each rating option assigned a number appended to **rating** (e.g. if the third rating option was selected, the document would render `<div class="rating_graphic rating3"></div>` when using the **rating_graphic** attribute)
- **ratingText** displays the rating text inside a div using the **rating_text** class (e.g. `<div class="rating_text">1.5 Stars</div>`)
- **dateCreated, FORMAT** displays the date the document was created. Use the **FORMAT** option to choose how to display the date and time. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
- **datePublished, FORMAT** displays the last date that the document was switched to published state. Use the **FORMAT** option to choose how to display the date and time. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
- **dateUpdated, FORMAT** displays the date that the document was last updated. Use the **FORMAT** option to choose how to display the date and time. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
- **startDate, FORMAT** will display the start date if the document is an event. Use the **FORMAT** option to choose how to display the date and time. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
- **expireDate, FORMAT** will display the expiration date if the document is an event. Use the **FORMAT** option to choose how to display the date and time. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
- **title** displays the document's title.
- **author** displays a link to the document list using the current document's author if the document is not an event.

Pre-configured Scripts

Description

This CMS comes with several PHP scripts that perform common functions that can be used to add additional value to your website:

1. rss.php - a simple RSS aggregator that can be accessed by an RSS reader.
2. contact.php - a simple, configurable contact form that can e-mail the contact form's data to the specified e-mail recipients.
3. processForm.php - a forms processor that can e-mail the field and values from any form to the specified e-mail recipients.

The three scripts are detailed in the following sections.

rss.php

Description

The RSS script simply serves up your published documents in an XML format that is compatible with RSS readers that support RSS v2.0. By default it pushes the content out by publish data in descending order.

rss.php?filter=

Returns the 50 most recent documents or blogs that are published, do not have "do not syndicate" set, are not set as a "component types", and are not under a category that is assigned to a user group.

rss.php?filter=documents

Returns only the 50 most recent documents that are published, do not have "do not syndicate" set, are not set as a "component types", and are not under a category that is assigned to a user group.

rss.php?filter=blogs

Returns only the 50 most recent blogs that are published and are not under a category that is assigned to a user group.

contact.php

Description

This simple contact form is designed to collect common contact information and be easy to implement - you literally just have to link to it (but you're free to customize the layout of the page or adjust the settings in the properties file, if you like) The contact form script features several different parameters which allow you to customize how it behaves:

Accepted URL Parameters

subject - prefill the subject.

category - allows you to fake the category that the contact form is under.

returnUrl - a URL to redirect the user to when the cancel button is clicked, or when the form has been successfully submitted.

contact.properties file

enableContactForm - is a true or false setting that determines whether or not the contact form can be viewed and/or process information submitted to it. Setting this to false disables the contact form.

Required Field Settings

name - is a true or false setting that determines whether or not the name field is required.

email - is a true or false setting that determines whether or not the email field is required.

phone - is a true or false setting that determines whether or not the phone field is required.

subject - is a true or false setting that determines whether or not the subject field is required.

body - is a true or false setting that determines whether or not the body field is required.

fromEmailAddress - the e-mail address that the contact form's data will be sent from if the user does not supply an e-mail address, otherwise the e-mail address that the user supplied is used as both the from and reply-to address.

contactFormRecipients - e-mail recipients (comma separated) of the contact form's data.

processForm.php

Description

This is a general purpose script that will process any form data that is submitted to it and e-mail the data in the form of key/value pairs to the formProcessorRecipients email addresses defined in the processForm.properties file.

Accepted URL Parameters

subject - this value will be used as the subject for the e-mailed data. (optional)

returnURL - the URL to return the user to after submitting the form successfully.

processForm.properties file

enableFormsProcessor - is a true or false setting that determines whether or not the forms processor can process information submitted to it. Setting this to false disables the forms processor.

fromEmailAddress - the e-mail address that the forms processor will send its data from.

formsProcessorRecipients - the e-mail recipients (comma separated) of the forms processor's data.

Configuration File (config.properties)

Description

The configuration file allows you to control the behavior of the CMS in ways not accessible via the control panel.

displaySignUpAgreementLink

Options: *true/false*

Function: If this attribute is set to true, visitors are presented with an agreement checkbox at the bottom of the sign-up form and a link to `signUpAgreementLinkUrl` is given. User will not be able to sign-up unless they check the agreement checkbox.

signUpAgreementLinkUrl

Options: *provide a link to the agreement file (defaults to termsandconditions.html)*

Function: This is the file that will be linked to if `displaySignUpAgreementLink` is set to true.

signUpAgreementLinkLabel

Options: *provide the text used to link to the agreement (defaults to "the terms and conditions")*

Function: This is the text that is displayed as a clickable link to the right of the checkbox on the sign-up form when `displaySignUpAgreementLink` has been set to true.

requireSignUpApproval

Options: *true/false*

Function: If this attribute is set to true, visitors are allowed to sign-up, but the system then displays the "`memberApprovalPending.php`" content and e-mails a notification to the master account and all site admin accounts (provided they have set their profiles to allow e-mail notifications) that a new member is pending approval. New users cannot access restricted areas of the site until an admin approves their account.

contentCreationNotifications

Options: *true/false*

Function: If set to true, the system sends notifications when a user creates new content.

masterNotification

Options: *true/false*

Function: Notifies the master account that new content has been created if `contentCreationNotifications` is set to true.

adminNotification

Options: *true/false*

Function: Notifies any admin accounts that new content has been created if `contentCreationNotifications` is set to true.

editorNotification

Options: *true/false*

Function: Notifies any editor accounts that new content has been created if `contentCreationNotifications` is set to true.

findPeopleAuthentication

Options: *true/false*

Function: The find people option will be hidden and not function unless the user is logged in, if this attribute is set to true.

findGroupsAuthentication

Options: *true/false*

Function: The group search will be hidden and not function unless the user is logged in, if this attribute is set to true.

viewGroupsAuthentication

Options: *true/false*

Function: Setting this attribute to true will require a user to be logged in to view a group.

enableRssFeed

Options: *true/false*

Function: Setting this attribute to true will cause the rss.php file to return valid XML data.

siteEmailAddress

Options: *Provide a single e-mail address.*

Function: This is the e-mail address that is used in all auto-generated outbound e-mail messages.

profileFieldsEnabled

Options: *race,gender,height,bodytype,orientation,religion,vices,lookingfor*

Function: Removing options removes them from view in the user profiles, profile editors, and profile searches.

CMS Rich Content Components

Description

CMS Rich Content Component Blocks are a great way to tap into the unique architecture of our CMS by automatically displaying pieces of content based on their type, category assignments, and popularity or date published.

Syntax

```
[[rc_component type="types" parameter="" parameter="" parameter="" etc etc etc]]
```

Type Options

1. **type** is the name of the component type you want to display. There are currently 6 types to choose from:
 - document
 - blog
 - event
 - announcement
 - conversation
 - member

Each Rich Component type has its own set of parameters which are outlined in the following sections.

Rich Content Component Types

Document Component

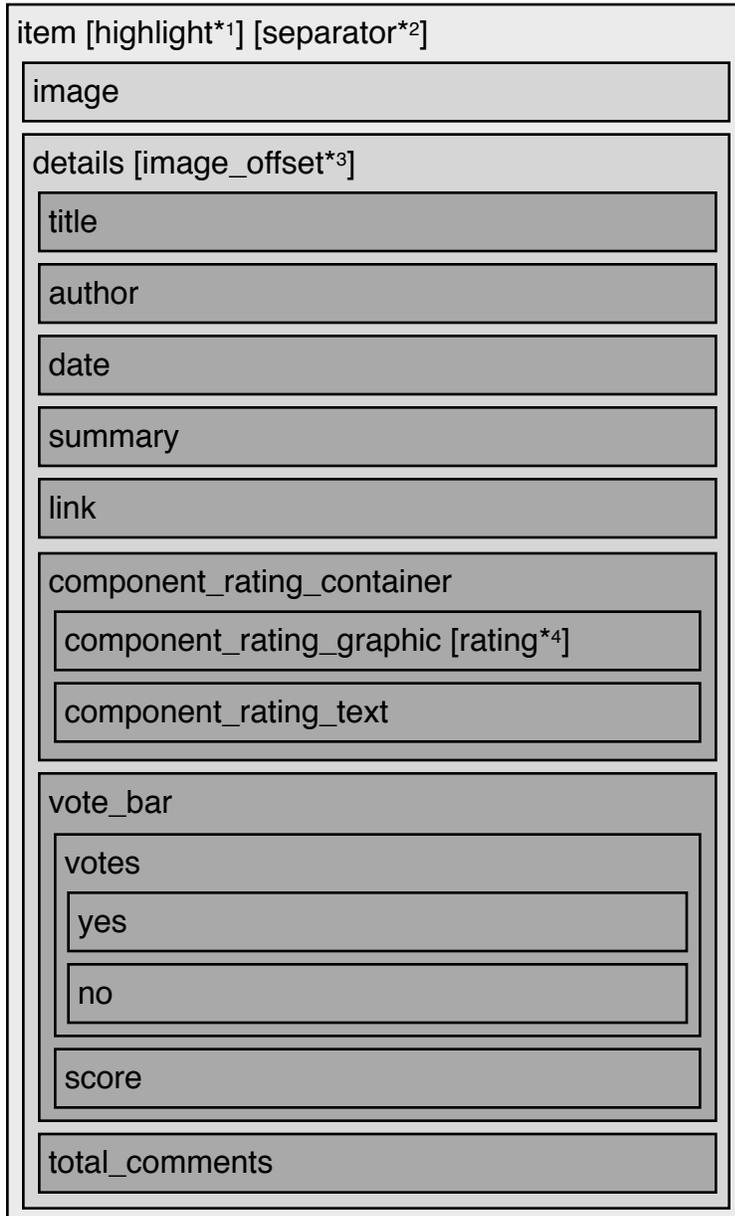
```
[[rc_component type="document" documentType="" category="" subcategory="" subject="" showTitle="" showAuthor="" showDate="" showSummary="" maxCharCount="" showSummaryLink="" showRatingGraphic="" showRatingText="" showVotes="" showScore="" showTotalComments="" organizedBy="" showImage="" imageWidth="" imageHeight="" maxDisplay="" startAt="" maxPerRow="" highlight="" skip="" noContent=""]]
```

Parameters

1. **documentType** is a comma separated list that filters the output using the specified document types. Specifying `$_this` as a value will use the current document's value.
2. **category** is a comma separated list that filters the output using the specified categories. Specifying `$_this` as a value will use the current document's value.
3. **subcategory** is a comma separated list that filters the output using the specified subcategories. Specifying `$_this` as a value will use the current document's value.
4. **subject** is a comma separated list that filters the output using the specified subjects. Specifying `$_this` as a value will use the current document's value.
5. **showTitle** is a true or false setting that determines whether or not to show the title.
6. **showAuthor** is a true or false setting that determines whether or not to show the author.
7. **showDate, FORMAT** is a true or false setting that determines whether or not to show the date published. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
8. **showSummary** is a true or false setting that determines whether or not to show the summary.
9. **maxCharCount** is the maximum number of characters to display from the summary.
10. **showSummaryLink** is a true or false setting that determines whether or not to show the summary link.
11. **showRatingGraphic** is a true or false setting that determines whether or not the rating graphics (if a rating was selected) are displayed.
12. **showRatingText** is a true or false setting that determines whether or not the rating text (if a rating was selected) is displayed.
13. **showVotes** is a true or false setting that determines whether or not to show the total + and - votes. This option will only display the total + and - votes if the corresponding document's option to show a toolbar has also been selected.
14. **showScore** is a true or false setting that determines whether or not to show overall % score. This option will only display the overall % score if the corresponding document's option to show a toolbar has also been selected.
15. **showTotalComments** is a true or false setting that determines whether or not to show the total comment count. This option will only display the total comment count if the corresponding document's option to show comments has also been selected.
16. **organizedBy** can be set to either popular or latest:
 - 16.1. **popular,nn** displays items by most popular on the current day, week, month, year, or all time depending on the number of items available from each time span relative to the request count, which is set by providing a value for *nn*. The request count indicates how many items must be available to trigger the algorithm. For example, if you specify 4, then there must be at least 4 items published during any one of the available time spans (i.e. 24 hours, 1 week, 2 weeks, 30 days, or entire year) If there are not at least 4 in any one of the time spans, then the algorithm will use any available items from any time.
 - 16.2. **latest** (default) displays items by published date from newest to oldest.
 - 16.3. **featured** displays items that have been added to the featured documents list, have not expired, and are listed as active.

- 16.4. **focused** displays items that have been added to the focused documents list, have not expired, and are listed as active.
- 16.5. **accessed** displays items in order of most recently accessed.
17. **showImage** is a true or false setting that determines whether or not to show the summary image. If the `imageWidth` and `imageHeight` values are not supplied then the cached thumbnails are used. In that case the width and height will be 170 pixels. Using the cached thumbnails will result in faster load times, reduced CPU usage and overall better performance.
18. **imageWidth**: is the maximum width to display the summary image in.
19. **imageHeight** is the maximum height to display the summary image in.
20. **maxDisplay** is the maximum number of items to display.
21. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
22. **maxPerRow** is the maximum number of items to display before inserting the row separator.
23. **highlightCurrent** is a true or false setting that determines whether or not the highlight class is added to the item when the ID of the item currently being displayed matches the ID of the document that contains this component. This is useful when displaying a list of documents in a sidebar, for example, and you want to indicate to the user which document in list is currently being viewed.
24. **skip** can be used to tell this component to skip all documents marked as featured and/or focused by adding `$_featured` and/or `$_focused` respectively as values in this parameter. Choosing to skip featured items when you've set `organizedBy` to `featured` will cause skip to be ignored, the same is true when `organizedBy` is set to `focused`. You can't skip these items if you're using them as the primary output. `$_this` can be used to prevent the current document in which this component is used from showing up in the component's output. This parameter is useful when you're displaying combinations of featured, focused, latest, and/or accessed on the same page but you want to avoid duplication.
25. **noContent** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Document Component CSS Classes and DIV structure



*1: The highlight class is appended to each item when the highlight parameter is set to true and the currently displayed item's ID matches the ID of the document that contains this component.

*2: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.

*3: The image_offset class is appended to the end of the details class when images are enabled and a summary image is used in the document. This could be used to apply a margin (equal to the width of the image plus its margins and/or padding) to the details section, creating a column effect.

*4: The rating[nn] class is appended to the end of the component_rating_graphic class. Its intended use is to adjust the width of the div to display a set of horizontal rating star images. The [nn] is a number

representing the selected array value for a particular rating. (e.g. *rating_graphic rating3*) The array is stored in `./assets/core/config/part_ratings.php`

row_separator

no_content

Blog Component

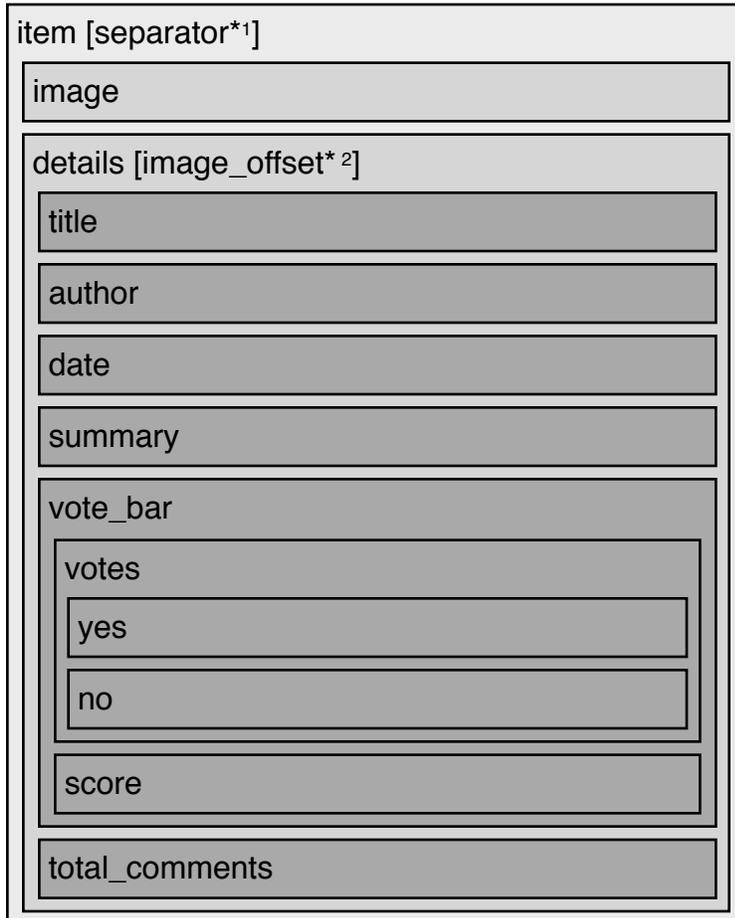
```
[[rc_component type="blog" author="" documentType="" category="" subcategory="" subject=""
showTitle="" showAuthor="" showDate="" showSummary="" maxCharCount="" showVotes=""
showScore="" showTotalComments="" organizedBy="" showImage="" imageWidth="" imageHeight=""
maxDisplay="" startAt="" maxPerRow="" noContent=""]]
```

Parameters

1. **author** filters the output using the specified username.
2. **documentType** is a comma separated list that filters the output using the specified document types. Specifying `$_this` as a value will use the current document's value.
3. **category** is a comma separated list that filters the output using the specified categories. Specifying `$_this` as a value will use the current document's value.
4. **subcategory** is a comma separated list that filters the output using the specified subcategories. Specifying `$_this` as a value will use the current document's value.
5. **subject** is a comma separated list that filters the output using the specified subjects. Specifying `$_this` as a value will use the current document's value.
6. **showTitle** is a true or false setting that determines whether or not to show the title.
7. **showAuthor** is a true or false setting that determines whether or not to show the author.
8. **showDate, FORMAT** is a true or false setting that determines whether or not to show the date published. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
9. **showSummary** is a true or false setting that determines whether or not to show the summary.
10. **maxCharCount** is the maximum number of characters to display from the summary.
11. **showVotes** is a true or false setting that determines whether or not to show the total + and - votes. This option will only display the total + and - votes if the corresponding document's option to show a toolbar has also been selected.
12. **showScore** is a true or false setting that determines whether or not to show overall % score. This option will only display the overall % score if the corresponding document's option to show a toolbar has also been selected.
13. **showTotalComments** is a true or false setting that determines whether or not to show the total comment count. This option will only display the total comment count if the corresponding document's option to show comments has also been selected.
14. **organizedBy** can be set to either popular or latest.
 - 14.1. **popular,nn** displays items by most popular on the current day, week, month, year, or all time depending on the number of items available from each time span relative to the request count, which is set by providing a value for *nn*. The request count indicates how many items must be available to trigger the algorithm. For example, if you specify 4, then their must be at least 4 items published during any one of the available time spans (i.e. 24 hours, 1 week, 2 weeks, 30 days, or entire year) If there are not at least 4 in any one of the time spans, then the algorithm will use any available items from any time.
 - 14.2. **latest** (default) displays items by published date from newest to oldest.
 - 14.3. **accessed** displays items in order of most recently accessed.
15. **showImage** is a true or false setting that determines whether or not to show the summary image. If the `imageWidth` and `imageHeight` values are not supplied then the cached thumbnails are used. In that case the width and height will be 170 pixels. Using the cached thumbnails will result in faster load times, reduced CPU usage and overall better performance.
16. **imageWidth**: is the maximum width to display the summary image in.
17. **imageHeight** is the maximum height to display the summary image in.
18. **maxDisplay** is the maximum number of items to display.
19. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
20. **maxPerRow** is the maximum number of items to display before inserting the row separator.

21. ***skip*** is a list of ids separated with commas used to indicate specific items to not display in the output of this component.
22. ***noContent*** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Blog Component CSS Classes and DIV Structure



*1: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.

*2: The image_offset class is appended to the end of the details class when images are enabled and a summary image is used in the document. This could be used to apply a margin (equal to the width of the image plus its margins and/or padding) to the details section, creating a column effect.

row_separator

no_content

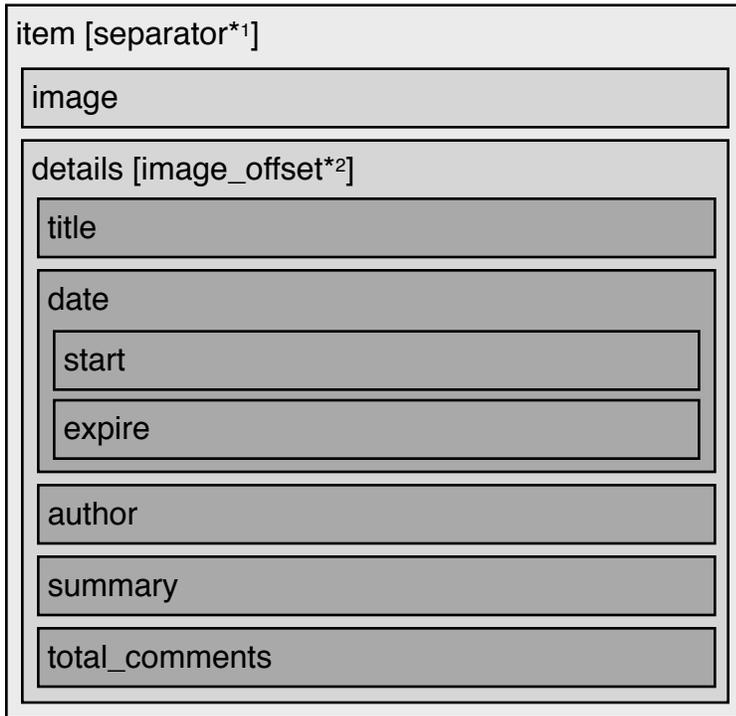
Event Component

```
[[rc_component type="event" category="" subcategory="" subject="" showTitle="" showDate="" showAuthor="" showSummary="" maxCharCount="" showImage="" imageWidth="" imageHeight="" showTotalComments="" organizedBy="" maxDisplay="" startAt="" maxPerRow="" noContent=""]]
```

Parameters

1. **category** is a comma separated list that filters the output using the specified categories. Specifying `$_this` as a value will use the current document's value.
2. **subcategory** is a comma separated list that filters the output using the specified subcategories. Specifying `$_this` as a value will use the current document's value.
3. **subject** is a comma separated list that filters the output using the specified subjects. Specifying `$_this` as a value will use the current document's value.
4. **showTitle** is a true or false setting that determines whether or not to show the title.
5. **showAuthor** is a true or false setting that determines whether or not to show the author.
6. **showDate, FORMAT** is a true or false setting that determines whether or not to show the start and end dates. When set to true, use the **FORMAT** option to choose how to display the start and end date and time for the event. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
7. **showSummary** is a true or false setting that determines whether or not to show the summary.
8. **maxCharCount** is the maximum number of characters to display from the summary.
9. **showImage** is a true or false setting that determines whether or not to show the summary image. If the `imageWidth` and `imageHeight` values are not supplied then the cached thumbnails are used. The default maximum width and height is 170 pixels and cannot be modified. Using the cached thumbnails will result in faster load times, reduced CPU usage and overall better performance.
10. **imageWidth** is the maximum width to display the summary image in.
11. **imageHeight** is the maximum height to display the summary image in.
12. **showTotalComments** is a true or false setting that determines whether or not to show the total comment count. This option will only display the total comment count if the corresponding document's option to show comments has also been selected.
13. **organizedBy** can be set to either popular or latest.
 - 13.1. **unexpired** (default) displays items by start date from next event to start to events further out .
 - 13.2. **accessed** displays items in order of most recently accessed.
14. **maxDisplay** is the maximum number of items to display.
15. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
16. **maxPerRow** is the maximum number of items to display before inserting the row separator.
17. **noContent** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Event Component CSS Classes and DIV Structure



*1: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.

*2: The image_offset class is appended to the end of the details class when images are enabled and a summary image is used in the document. This could be used to apply a margin (equal to the width of the image plus its margins and/or padding) to the details section, creating a column effect.

row_separator

no_content

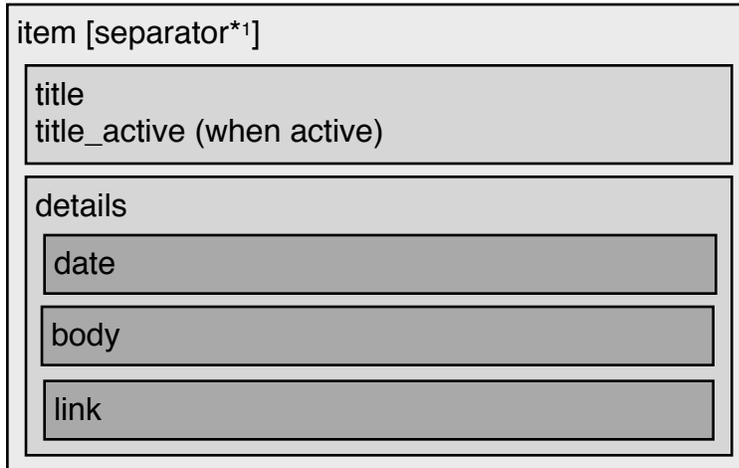
Announcement Component

```
[[rc_component type="announcement" id="" showDate="" showLink="" togglerStyle="" maxDisplay="" startAt="" maxPerRow="" noContent=""]]
```

Parameters

1. **id** is the ID that will be used to label the togglers with the HTML. It must be unique.
2. **showDate, FORMAT** is a true or false setting that determines whether or not to show the date published. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
3. **showLink** is a true or false setting that determines whether or not to show the link.
4. **togglerStyle** sets the visual behavior when the togglers opened or closed and can be set to either fade or standard.
5. **maxDisplay** is the maximum number of items to display.
6. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
7. **maxPerRow** is the maximum number of items to display before inserting the row separator.
8. **noContent** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Announcement Component CSS Classes and DIV Structure



*1: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.

row_separator

no_content

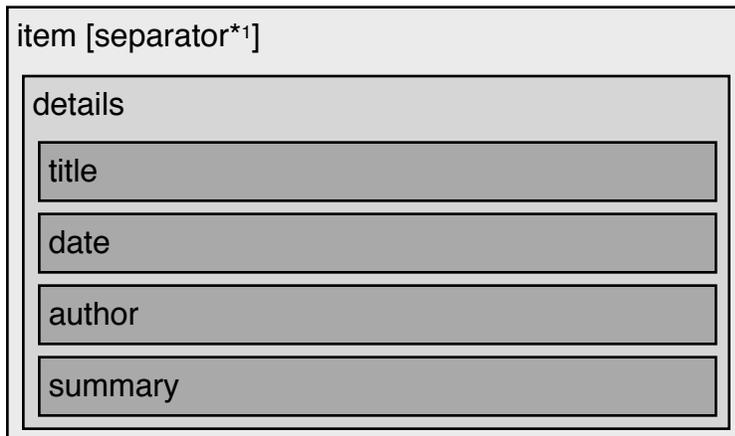
Conversation Component

```
[[rc_component type="conversation" showTitle="" showDate="" showAuthor="" showBody=""  
maxCharCount="" maxDisplay="" startAt="" maxPerRow="" noContent=""]]
```

Parameters

1. **showTitle** is a true or false setting that determines whether or not to show the title.
2. **showDate, FORMAT** is a true or false setting that determines whether or not to show the date published. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
3. **showAuthor** is a true or false setting that determines whether or not to show the author.
4. **showBody** is a true or false setting that determines whether or not to show the post's body.
5. **maxCharCount** is the maximum number of characters to display from the post's body.
6. **maxDisplay** is the maximum number of items to display
7. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
8. **maxPerRow** is the maximum number of items to display before inserting the row separator.
9. **noContent** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Conversation Component CSS Classes and DIV Structure



*1: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.



Member Component

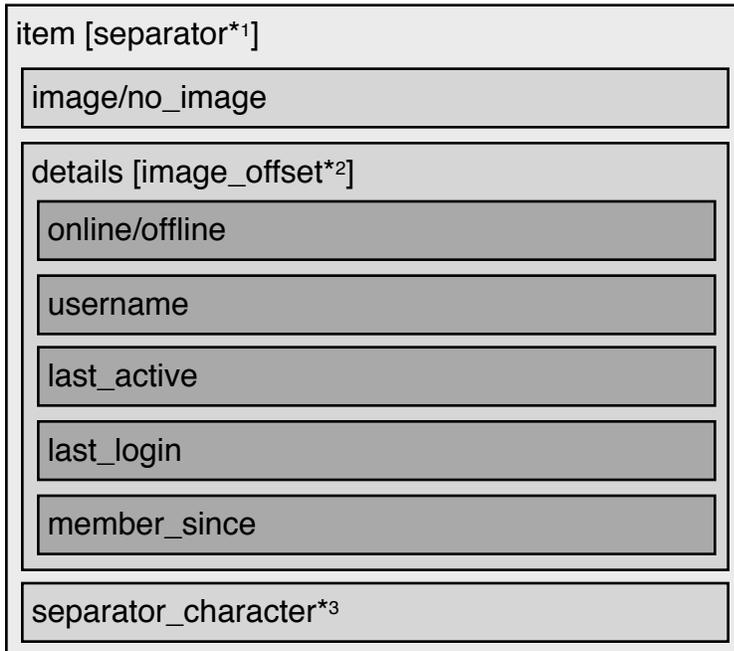
```
[[rc_component type="member" showImage="" imageWidth="" imageHeight="" showOnlineNow=""  
onlineLabel="" offlineLabel="" showUsername="" showLastActive="" lastActiveLabel="" showLastLogin=""  
lastLoginLabel="" showMemberSince="" memberSinceLabel="" hasImage="" noImageUrl=""  
organizedBy="" maxDisplay="" startAt="" maxPerRow="" separatorCharacter="" noContent=""]]
```

Parameters

1. **showImage** is a true or false setting that determines whether or not to display the user's default image. If the imageWidth and imageHeight values are not supplied then high performance cached thumbnails are used; in which case the default width and height is 170 pixels.
2. **imageWidth** is the maximum width to display the summary image in.
3. **imageHeight** is the maximum height to display the summary image in.
4. **showOnlineNow** is a true or false setting that determines whether or not to display the user's online status.
5. **onlineLabel** is the text displayed when the user is online.
6. **offlineLabel** is the text displayed when the user is offline.
7. **showUsername** is a true or false setting that determines whether or not to display the user's username.
8. **showLastActive, FORMAT** is a true or false setting that determines whether or not to display the user's last activity date/time. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
9. **lastActiveLabel** is the text displayed to the left of the user's last activity date/time.
10. **showLastLogin, FORMAT** is a true or false setting that determines whether or not to display the user's last login date/time. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
11. **lastLoginLabel** is the text displayed to the left of the user's last login date/time.
12. **showMemberSince, FORMAT** is a true or false setting that determines whether or not to display when the user signed up for an account. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
13. **memberSinceLabel** is the text displayed to the left of when the user signed up for an account.
14. **hasImage** is a true or false setting that when set to true causes the component to only display members that have a default image selected.
15. **noImageUrl** is the URL path to an image to use when no image is available and showImage is set to true. The image is not automatically resized to the imageWidth and imageHeight parameters.
16. **organizedBy** can be set to either active, loggedIn, or signedUp.
 - 16.1. **active** (default) lists users by those who have recently been active on the site, starting with those who have most recently had activity. Activity is based on how often the user clicks around on the site. When a user signs out, activity is no longer available, and in that case the last log in date and time is used.
 - 16.2. **loggedIn** list users in order of those who have most recently logged in, starting with the most recently logged in.
 - 16.3. **signedUp** lists users in order of when they signed up, starting with the most recently signed up users.
 - 16.4. **accessed** lists users in order of most recently accessed.
 - 16.5. **random** lists users randomly.
17. **maxDisplay** is the maximum number of items to display. The default is one.
18. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
19. **maxPerRow** is the maximum number of items to display before inserting the row separator.

20. ***separatorCharacter*** is an optional character (contained in its own div) that is appended to the bottom/end of the details for each member returned in the results of the Member Component.
21. ***noContent*** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Member Component CSS Classes and DIV Structure



*1: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.

*2: The image_offset class is appended to the end of the details class when images are enabled and a summary image is used in the document. This could be used to apply a margin (equal to the width of the image plus its margins and/or padding) to the details section, creating a column effect.

*3: The separator_character div is only rendered if there is more than one item returned and the separatorCharacter parameter is provided.

row_separator

no_content

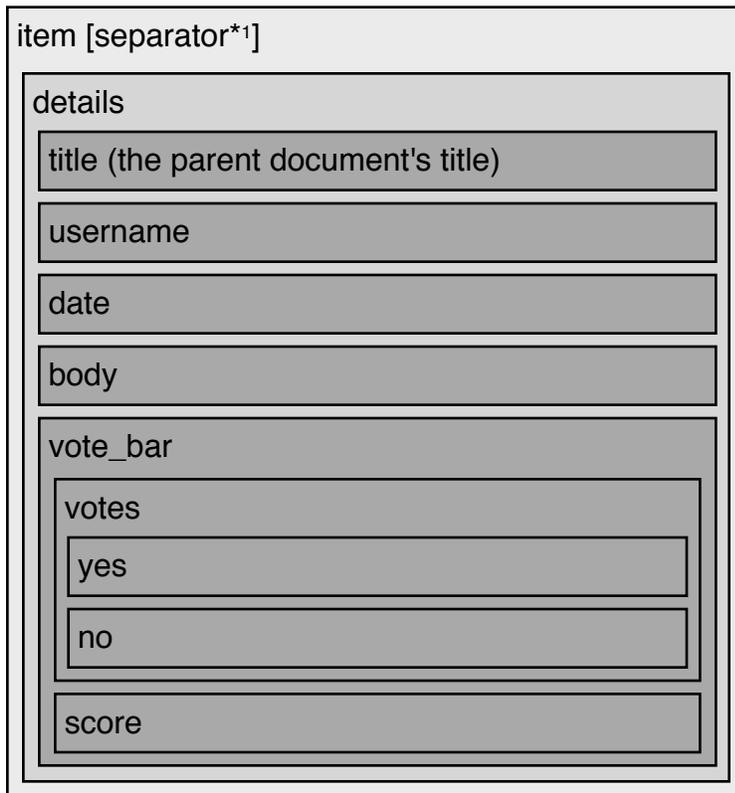
Comment Component

```
[[rc_component type="comment" category="" subcategory="" subject="" showTitle="" showDate="" showUsername="" showBody="" maxCharCount="" showVotes="" showScore="" organizedBy="" maxDisplay="" startAt="" maxPerRow="" noContent=""]]
```

Parameters

1. **category** filters the output using the specified category. Specifying `$_this` as the value will use the current document's value.
2. **subcategory** filters the output using the specified subcategory. Specifying `$_this` as the value will use the current document's value.
3. **subject** filters the output using the specified subject. Specifying `$_this` as the value will use the current document's value.
4. **showTitle** is a true or false setting that determines whether or not to show the title of the parent document. Note: you need to set this to true if you want your users to be able to click to get to the parent document.
5. **showDate, FORMAT** is a true or false setting that determines whether or not to show the date published. When set to true, use the **FORMAT** option to choose how to display the published date and time for the document. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
6. **showUsername** is a true or false setting that determines whether or not to show the author.
7. **showBody** is a true or false setting that determines whether or not to show the post's body.
8. **maxCharCount** is the maximum number of characters to display from the post's body.
9. **showVotes** is a true or false setting that determines whether or not to show the total + and - votes. This option will only display the total + and - votes if the corresponding document's option to show a toolbar has also been selected.
10. **showScore** is a true or false setting that determines whether or not to show overall % score. This option will only display the overall % score if the corresponding document's option to show a toolbar has also been selected.
11. **organizedBy** can be set to either popular or latest:
 - 11.1. **latest** displays items by published date from newest to oldest.
 - 11.2. **oldest** (default) displays items by published date from oldest to newest.
 - 11.3. **popular** displays items by score from most highest score and most up votes, to lowest score.
 - 11.4. **unpopular** displays items by score from most lowest score and most down votes, to highest score.
12. **maxDisplay** is the maximum number of items to display
13. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
14. **maxPerRow** is the maximum number of items to display before inserting the row separator.
15. **noContent** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Comment Component CSS Classes and DIV Structure



*1: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.

row_separator

no_content

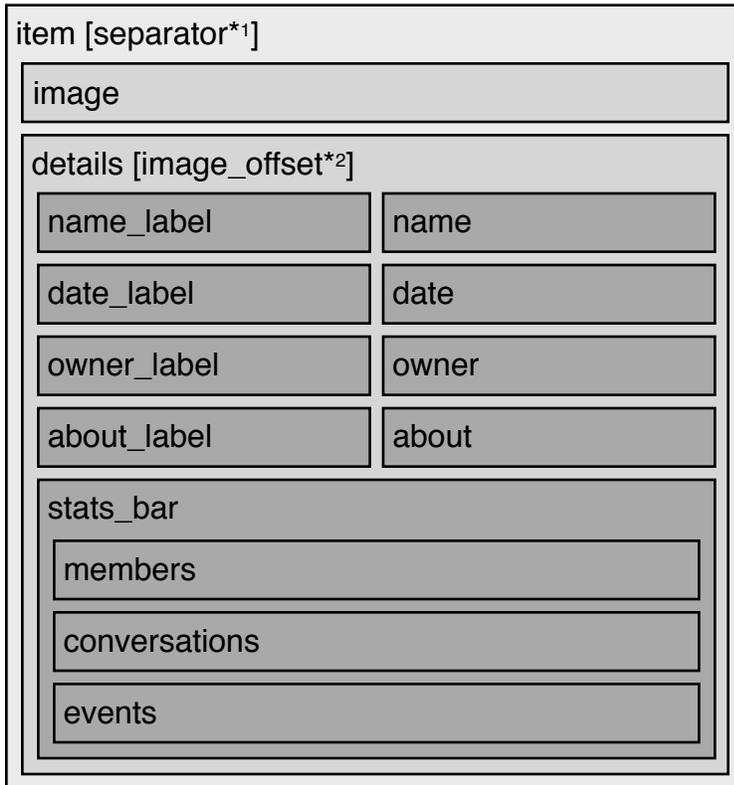
Group Component

```
[[rc_component type="group" showName="" showOwner="" showDate="" showAbout=""  
maxCharCount="" showLabels="" showTotalMembers="" showTotalConversations="" showTotalEvents=""  
showImage="" imageWidth="" imageHeight="" organizedBy="" maxDisplay="" startAt="" maxPerRow=""  
noContent=""]]
```

Parameters

1. **showName** is a true or false setting that determines whether or not to show the group's name.
2. **showOwner** is a true or false setting that determines whether or not to show the owner of the group.
3. **showDate, FORMAT** is a true or false setting that determines whether or not to show the date the group was established. When set to true, use the **FORMAT** option to choose how to display the established date and time for the group. The default value is: F jS, Y h:i A. Refer to [Table 1](#) for a list of possible values.
4. **showAbout** is a true or false setting that determines whether or not to show the group's summary.
5. **maxCharCount** is the maximum number of characters to display from the group's summary.
6. **showLabels** is a true or false setting that determines whether or not to display labels for the name, owner, date established, and about summary sections. (displayed as: *Name*, *Owner*, *Established*, and *About* respectively)
7. **showTotalMembers** is a true or false setting that determines whether or not to show how many users are registered members of the group.
8. **showTotalConversations** is a true or false setting that determines whether or not to show the total conversations this group contains.
9. **showTotalEvents** is a true or false setting that determines whether or not to show the total number of events this group has created.
10. **showImage** is a true or false setting that determines whether or not to show the summary image. If the **imageWidth** and **imageHeight** values are not supplied then the cached thumbnails are used. In that case the width and height will be 170 pixels. Using the cached thumbnails will result in faster load times, reduced CPU usage and overall better performance.
11. **imageWidth**: is the maximum width to display the summary image in.
12. **imageHeight** is the maximum height to display the summary image in.
13. **organizedBy** can be set to either popular or latest:
 - 13.1. **latest** (default) displays groups by date established from newest to oldest.
 - 13.2. **oldest** displays groups by date established from oldest to newest.
 - 13.3. **accessed** displays groups in order of most recently accessed.
 - 13.4. **alphabetical** displays groups in alphabetical order.
14. **maxDisplay** is the maximum number of items to display.
15. **startAt** determines where to start the list at (0 or empty is the beginning of the list; 1 would be the second item)
16. **maxPerRow** is the maximum number of items to display before inserting the row separator.
17. **noContent** is designed to contain a brief message to display when there is no content to display based on the criteria supplied in the rest of the component's parameters.

Group Component CSS Classes and DIV structure



*1: The separator class is appended to each item as long as the item is not the last item in the row, as defined by the maxPerRow value.

*2: The image_offset class is appended to the end of the details class when images are enabled and a summary image is used in the document. This could be used to apply a margin (equal to the width of the image plus its margins and/or padding) to the details section, creating a column effect.

row_separator

no_content

Table 1 - Date Formatting Options

	Description	Examples
Day		
d	Day of the month, 2 digits with leading zeros	01 to 31
D	A textual representation of a day, three letters	Mon through Sun
j	Day of the month without leading zeros	1 to 31
l	A full textual representation of the day of the week	Sunday through Saturday
N	ISO-8601 numeric representation of the day of the week	1 (for Monday) through 7 (for Sunday)
S	English ordinal suffix for the day of the month, 2 characters	st, nd, rd or th. Works well with j
w	Numeric representation of the day of the week	0 (for Sunday) through 6 (for Saturday)
z	The day of the year (starting from 0)	0 through 365
Week		
W	ISO-8601 week number of year, weeks starting on Monday	Example: 42 (the 42nd week in the year)
Month		
F	A full textual representation of a month, such as January or March	January through December
m	Numeric representation of a month, with leading zeros	01 through 12
M	A short textual representation of a month, three letters	Jan through Dec
n	Numeric representation of a month, without leading zeros	1 through 12
t	Number of days in the given month	28 through 31
Year		
o	ISO-8601 year number. This has the same value as Y, except that if the ISO week number (W) belongs to the previous or next year, that year is used instead.	Examples: 1999 or 2003
Y	A full numeric representation of a year, 4 digits	Examples: 1999 or 2003
y	A two digit representation of a year	Examples: 99 or 03

Time		
a	Lowercase Ante meridiem and Post meridiem	am or pm
A	Uppercase Ante meridiem and Post meridiem	AM or PM
g	12-hour format of an hour without leading zeros	1 through 12
G	24-hour format of an hour without leading zeros	0 through 23
h	12-hour format of an hour with leading zeros	01 through 12
H	24-hour format of an hour with leading zeros	00 through 23
i	Minutes with leading zeros	00 to 59
s	Seconds, with leading zeros	00 through 59
Timezone		
e	Timezone identifier	Examples: UTC, GMT, Atlantic/Azores
T	Timezone abbreviation	Examples: EST, MDT ...