Table of Contents

Foswiki::Attrs

package Foswiki::Attrs

Class of attribute sets, designed for parsing and storing attribute values from a macro e.g. `%MACRO{"joe" fred="bad" joe="mad"}%`

An attribute set is a hash containing an entry for each parameter. The default parameter (unnamed quoted string) is named `_DEFAULT` in the hash.

Attributes declared later in the string will override those of the same name defined earlier. The one exception to this is the _DEFAULT key, where the *first* instance is always taken.

As well as the default Foswiki syntax (parameter values double-quoted) this class also parses single-quoted values, unquoted spaceless values, spaces around the =, and commas as well as spaces separating values. The extended syntax has to be enabled by passing the `$friendly` parameter to `new`.

API version $Date: 2009-01-06 19:25:41 +0100 (Tue, 06 Jan 2009) $ (revision $Rev: 6075 (2010-01-17) $)

Since *date* indicates where functions or parameters have been added since the baseline of the API (TWiki release 4.2.3). The *date* indicates the earliest date of a Foswiki release that will support that function or parameter.

Deprecated *date* indicates where a function or parameters has been deprecated. Deprecated functions will still work, though they should *not* be called in new plugins and should be replaced in older plugins as soon as possible. Deprecated parameters are simply ignored in Foswiki releases after *date*.

Until *date* indicates where a function or parameter has been removed. The *date* indicates the latest date at which Foswiki releases still supported the function or parameter.

ClassMethod new ($string) => \%attrsObjectRef

- `$string` - String containing attribute specification

Parse a standard attribute string containing name=value pairs and create a new attributes object. The value may be a word or a quoted string. If there is an error during parsing, the parse will complete but $attrs->{_ERROR} will be set in the new object. $attrs->{_RAW} will always contain the full unprocessed $string.

ObjectMethod isEmpty() -> boolean

Return false if attribute set is not empty.

ObjectMethod remove($key) -> $value

- `$key` - Attribute to remove

Remove an attr value from the map, return old value. After a call to `remove` the attribute is no longer defined.

ObjectMethod stringify() -> $string

Generate a printed form for the map, using strict attribute syntax, with only the single-quote extension syntax

observed (no { } brackets, though).

Edit | Attach | Print version | History: %REVISIONS% | Backlinks | Raw View | More topic actions
Topic revision: r1 - 12 Sep 2009 - 04:10:27 - ProjectContributor

- System

- Log In

- **Toolbox**
- Users
- Groups
- Index
- Search
- Changes
- Notifications
- RSS Feed
- Statistics
- Preferences

- **User Reference**
- BeginnersStartHere
- TextFormattingRules
- Macros
- FormattedSearch
- QuerySearch
- DocumentGraphics
- SkinBrowser
- InstalledPlugins

- **Admin Maintenance**
- Reference Manual
- AdminToolsCategory
- InterWikis
- ManagingWebs
- SiteTools
- DefaultPreferences
- WebPreferences

- **Categories**
- Admin Documentation
- Admin Tools
- Developer Doc
- User Documentation
- User Tools

- **Webs**
- Public
- System

-