

gtbu /
Typesetter5.2

<> Code

Issues 4

Pull requests

Actions

Projects

Wiki

Security

Addon Dev – Hooks

Edit

New page

[Jump to bottom](#)gtbu edited this page on May 26, 2021 · [6 revisions](#)

Addon hooks are defined in the Addon.ini file of a plugin or theme and registered during its installation. During addon development, to add or remove hooks, the addon must be updated (or uninstalled / reinstalled) via Typesetter's admin menu.

Filters: Filter hooks are designed to manipulate certain values that are used in the execution of Typesetter. Depending on the hook, the output, the response to a certain request or the behavior of the system can be changed. Usually one or more parameters are passed to the hook function / method as an array or as an array of arrays. The called filter method will normally return the first passed parameter – in a modified form.

Actions: Contrary to filters, action hook calls do not expect a return value and they do not necessarily pass any parameters. They may, for example, be used to simply output content at the point in the CMS execution stack where the hook is called, e.g. when the <head> section of a page is created. Action hook calls may also trigger the execution of custom actions (hence the name) such as creating a custom thumbnail from a just uploaded image.

Both filter and action hooks are commonly used to process global variables (e.g. `$config`), global objects (e.g. `$page`) or superglobals (e.g. `$_REQUEST`). It largely depends on the point where the hook is called if a global variable, object or object property is already available. For optimal performance, it is recommended to implement early exit conditions in hook methods to avoid unnecessary code execution. If a hook is used by several addons, its calls will be executed in the order of addon installation. This fact usually has no noticeable effect but may play a role in plugin design under rather special circumstances, e.g. when addons themselves call other addon hooks.

All Current Plugin Hooks

A

- [AdminLinkLabel](#) (filter) (new as of ver. 5.2)
Replace an Admin Link Label (of a plugin or theme) e.g. for internationalization, based on `$config('language')`
- [AllowedTypes](#) (filter)
Change the file type extensions allowed to upload via finder
- [AntiSpam_Form](#) (filter)
Append form elements to the contact form, e.g. for anti-spam measures
- [AntiSpam_Check](#) (filter)
Check the submitted input values of the contact form, e.g. to verify anti-spam measures added via AntiSpam Form

added via `AntispamForm`

- [AvailableClasses](#) (filter) (new as of ver. 5.2)

Dynamically filters the Available Classes array before loaded into the Manage Sections editor.
Does not affect the saved classes via configuration.

B

C

- [CKEditorConfig](#) (filter)

Allows to change the CKEditor configuration in order to add / remove toolbar elements and others

- [CKEditorPlugins](#) (filter)

Allows to add CKEditor plugins

- [CleanText](#) (filter)

Allows to modify strings after being filtered by the CleanText method

- [contact_form_check](#) (filter)

May be used to prevent sending an e-mail by returning `false`

- [contact_form_pre_captcha](#) (action)

May be used to echo html / text before the output of a CAPTCHA in the contact form

D

E

- [edit_layout_cmd](#) (action)

No description

- [ExecArea](#) (filter)

May be used to modify parameters before they are passed to the
`\gp\tool\Output::ExecInfo` function

F

- [FileDeleted](#) (action)
Allows to take actions after a file was deleted (via finder or gallery editor)
- [FileUploaded](#) (action)
Allows to take actions after a file was uploaded (via finder or gallery editor)
- [FinderOptionsClient](#) (filter)
May be used to modify the client options of finder

G

- [GenerateContent_Admin](#) (action)
Add to all pages in case a user adds a gallery
- [GetAdminLink](#) (action)
May be used to add output to the admin link (login / logout link)
- [GetContent_After](#) (action)
Gets called after the content has been sent, so you can add your own stuff
- [GetDefaultContent](#) (filter)
Get the default content for the specified content type (section type)
- [GetHead](#) (action)
May be used to add output to the `<head>` section. Last chance to modify the page object before `template.php` is executed
- [GetMenuArray](#) (filter)
Allows to modify a menu before rendering
- [GetUrl](#) (filter)
May be used to filter urls generated by the `\gp\tool::GetUrl` function

H

- [HeadContent](#) (action)
Gets called before `ob_start()` so plugins can get buffer content
- [Html_Output](#) (filter)
Allows to filter the DOM array of the html output before saving it, e.g. to remove elements that shouldn't be saved

that shouldn't be saved

I

- [InlineEdit_Scripts](#) (filter)

Define the script content sent to the editor component of a section type (via AJAX)

J

K

L

- [LoggedIn](#) (filter)

May be used to modify the boolean returned by the `\gp\tool\LoggedIn` function. Use it wisely.

M

- [MenuCommand](#) (filter)

Allows to modify the `$cmd` value used by menu editing operations

- [MenuPageOptions](#) (action)

Allows to add options (e.g. controls) to the menu editing callout in Page Manager

- [MenuPageTrashed](#) (action)

May be used to take custom actions after a menu item (page) was moved to the trash

- [MetaTitle](#) (filter)

Allows to modify the content of the `<title>` tag in the `<head>` section

N

- [NewSections](#) (filter)

Used to define new section types that can be added to a page via 'Sections' (formerly 'Page') mode of the content editor

- [Notifications](#) (action) (new as of ver. 5.2)

Allows to use the methods of the `$notifications` object e.g. to add custom ones before output

O

P

- [PageCreated](#) (action)

Allows to take additional actions when a new page was created

- [PageRunScript](#) (filter)

A mighty hook, called relatively early, that allows to execute own functions based on the passed cmd parameter in a request. It has complete control of the type and content of the response / output

- [PageSetVars](#) (action)

A hook called relatively early to modify variables of the `$page` object

- [PostedSlug](#) (filter)

May be used to modify a slug posted by the user. Called after cleaning / filtering a slug by the `\gp\admin\Tools::PostedSlug` function

Q

R

- [RenameFileDone](#) (action)

May be used to take custom actions once a file was deleted (via finder)

- [ReplaceContentVars](#) (filter) (new as of ver. 5.2)

Allows to change existing or implement additional processing / replacement of variables used in content

S

- [SaveSection](#) (filter)

- [saveSection](#) (filter)
Used to process posted data of certain / custom section types before saving them
- [Search](#) (action)
Used to add custom content to be queried by Typesetter's built-in search 'engine'
- [SectionIsHidden](#) (filter)
Allows to change a content section's hidden state programmatically
- [SectionToContent](#) (filter)
This filter is used to return formatted content built from the `$section_data` array
- [SectionTypes](#) (filter)
Used with plugins to introduce / register new section types to the CMS
- [SimilarTitles](#) (filter) (new as of ver. 5.2)
Hook to filter/remove similar page links suggested on the 'Missing' (Error 404) page and used for auto-redirection

T

U

V

W

- [WhichPage](#) (filter)
The WhichPage filter hook is one of the first hooks called hence it can be used for quite a lot of 'early stuff'.

X

Y

Z



+ Add a custom footer

▼ Pages 37

Find a page...

[Home](#)

▸ [Addon Dev – 1st Steps](#)

▸ [Addon Dev – Addon.ini](#)

▼ [Addon Dev – Hooks](#)

All Current Plugin Hooks

[A](#)

[B](#)

[C](#)

[D](#)

[E](#)

[F](#)

[G](#)

[H](#)

[I](#)

[J](#)

[K](#)

[L](#)

[M](#)

[N](#)

[O](#)

[P](#)

[Q](#)

[R](#)

[S](#)

[T](#)

[U](#)[V](#)[W](#)[X](#)[Y](#)[Z](#)

► [Addon Dev – Plugin Pages](#)

[Ckeditor](#)[Combine.php](#)[Constants](#)[Content](#)[CSS Styling](#)[Custom Sections](#)[Debugging](#)[Email](#)[Extra Areas](#)[Gadgets](#)[Show 22 more pages...](#)

+ Add a custom sidebar

Clone this wiki locally

<https://github.com/gtbu/Typesetter5.2.wiki.git>

