Built-in fetch functions by module

Every fetch function the kernel's modules declare, listed from their function_definition.php, with the ones used most and how fetch(module, function, hash()) relates to PHP.

Source file: doc/eng-us/help/extension-development/part-8-reference/fetch-functions-reference.md
Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/extension-development/part-8-reference/fetch-functions-reference.md

Part 8, reference 25. A fetch function is a read operation a module offers to templates: `fetch( 'content', 'list', hash( 'parent_node_id', 2 ) )`. Each module declares its fetch functions in `function_definition.php`; this page lists the kernel's, read from those files in October 2026. Writing your own is in Fetch functions; using them in templates is in Fetch for builders and the designer help (Fetch basics).

The ones used most

Call	Returns
`fetch( 'content', 'node', hash( 'node_id', 2 ) )`	a node
`fetch( 'content', 'object', hash( 'object_id', 57 ) )`	an object
`fetch( 'content', 'list', hash( 'parent_node_id', 2, 'limit', 10, 'sort_by', array( 'published', false() ) ) )`	children of a node
`fetch( 'content', 'list_count', hash( 'parent_node_id', 2 ) )`	how many
`fetch( 'content', 'tree', hash( 'parent_node_id', 2, 'class_filter_type', 'include', 'class_filter_array', array( 'article' ) ) )`	a subtree
`fetch( 'user', 'current_user' )`	the current user
`fetch( 'user', 'has_access_to', hash( 'module', 'content', 'function', 'edit' ) )`	whether the user may do it
`fetch( 'content', 'class', hash( 'class_id', 'article' ) )`	a class

The list parameters are the lower-case forms of `eZContentObjectTreeNode::subTreeByNodeID()`'s (eZContentObjectTreeNode).

Every module

Module	Fetch functions
content	object, version, node, locale_list, locale, prioritized_languages, prioritized_language_codes, translation_list, language_states, non_translation_list, class, class_attribute_list, class_attribute, calendar, list, list_count, tree, tree_count, search, trash_count, trash_object_list, draft_count, draft_version_list, pending_count, pending_list, version_count, version_list, can_instantiate_class_list, class_list, can_instantiate_classes, contentobject_attributes, bookmarks, bookmark_rows, bookmark_folders, recent, section_list, tipafriend_top_list, view_top_list, collected_info_count, collected_info_count_list, collected_info_collection, collected_info_list, object_by_attribute, object_count_by_user_id, same_classattribute_node

Module	Fetch functions
user	current_user, is_logged_in, logged_in_count, anonymous_count, logged_in_list, logged_in_users, user_role, member_of, has_access_to, can_open, account_status, deactivation_offers
section	object, list, object_list, object_list_count, roles, user_roles
class	list, list_by_groups, latest_list, attribute_list, override_template_list
role	role, policies, policy_count
search	list, list_count
url	list, list_count
infocollector	collected_info_count, collected_info_count_list, collected_info_collection, collected_info_list
shop	basket, best_sell_list, related_purchase, wish_list, wish_list_count, current_wish_list, order, order_status_history_count, order_status_history, currency_list, currency, preferred_currency_code, user_country, product_category_list, product_category
package	list, list_count, maintainer_role_list, can_create, can_edit, can_import, can_install, can_export, content_export_access, can_read, can_list, can_remove, item, dependent_list, repository_list
notification	handler_list, digest_handlers, digest_items, event_content, subscribed_nodes, subscribed_nodes_count
collaboration	participant, participant_list, participant_map, message_list, item_list, inbox, inbox_row, item_count, group_tree, tree_count
setup	version, alias, major_version, minor_version, release, state, is_development, database_version, database_release, edition, extension_counts, cronjob_pauses, log_files
audit	events, count, event, chain_status, can_read, hidden_menu_links
apikey	can_create, counts
rss	has_export_by_node, export_by_node
workflow	workflow_statuses, workflow_type_statuses
layout	sitedesign_list
uiswitch	toolbar
exphelp	context, license, context_menu

Extensions add their own modules' fetch functions; each module's `function_definition.php` is the authority for its parameters.

In PHP

A fetch function is a method of the module's function collection class, named in `function_definition.php`. From PHP, call the underlying kernel class directly (for content, `eZContentObjectTreeNode::subTreeByNodeID()` and friends) rather than going through the template layer.

Fetch aliases

`fetch_alias( 'name', hash( ... ) )` calls a fetch function configured in `fetchalias.ini`, so a site can change what a named fetch does without editing templates.

Related pages

- Previous: Built-in template functions
- Next: INI files an extension author meets
- Fetch functions
- Fetch performance
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

> **Become an Exponential Partner**
>
> Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.
>
> Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.