17. Migration reference

Source file: doc/eng-us/install/17-migration-reference.md Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/install/17-migration-reference.md

This chapter is the reference appendix for the three migration chapters before it. It holds the tables that those chapters point to rather than repeat: every table of the Exponential 6.0 kernel schema with its purpose and its name in eZ Platform and Ibexa; every kernel datatype with the columns it stores its value in and the matching field type; the INI settings that have a YAML counterpart; template constructs next to their Twig equivalents; the legacy PHP classes next to the Public API services; password hash types, URL alias tables and image storage paths; every upgrade, repair and conversion script in the installation; a list of data checks to run before moving content; and a glossary of the names that changed between generations. The old product names (eZ Publish 4.x and 5.x, eZ Platform, Ibexa DXP) appear only where they identify the system a site comes from or goes to.

Contents · Previous: 16. Migrating from eZ Platform and Ibexa

Contents of this chapter

- 17.1 How the tables were verified
- 17.2 Database schema reference
- 17.3 Datatypes and field types
- 17.4 Settings: INI and YAML
- 17.5 Templates: TPL and Twig
- 17.6 PHP API: legacy classes and Public API services
- 17.7 Passwords, URL aliases and image storage
- 17.8 Data checks before migrating
- 17.9 Migration and upgrade tooling
- 17.10 Glossary of renamed concepts
- References

How to use it: start from the chapter of your path (14 for 4.x, 15 for the 5.x stack, 16 for eZ Platform and Ibexa) and come here when a step needs a name. Before any move, run the checks of 17.8 on a copy of the database: they find the data the other side will refuse, while it is still cheap to fix. For a move to the platform the datatype table of 17.3 decides what has to be converted; for a move to Exponential the "Ibexa 4.6" and "Ibexa 5" columns of 17.2 tell you which of your tables Exponential reads and which it ignores.

17.1 How the tables were verified

Every name in this chapter was checked against code or data, in one of these places:

Source	What it was used for

Source	What it was used for
This repository	`share/db_schema.dba` and `kernel/sql/{mysql,postgresql,sqlite}/`, `kernel/classes/datatypes/*/`, `settings/*.ini`, `kernel/common/eztemplateautoload.php`, `lib/eztemplate/classes/eztemplateautoload.php`, `kernel/classes/*.php`, `bin/php/`, `update/`
An Ibexa DXP 4.6 installation on the same server (`ibexa/core` v4.6.28)	The Public API service methods, the Twig function names, the field type aliases (`fieldtypes.yml`) and the legacy storage schema (`schema.yaml`)
The Ibexa 5 media-site reference installation used to port the media site (read-only)	The `ibexa_*` table names and columns, the field type identifiers stored in its content, the password hash type in use, the image path layout, and real `ibexa.yaml` siteaccess, image variation, DFS and HTTP cache configuration
The vendor's migration pages and field type reference (References)	The documented field type mapping, the unsupported field types, custom tag attribute types and the common post-migration issues

The reference database (SQLite) was opened read-only and queried with `SELECT` statements only. Anything that could not be checked in one of those places is marked **(not verified here)**.

Ibexa DXP 4.6 still uses the legacy table names (`ezcontentobject`, `ezuser` ...) and the legacy field type identifiers (`ezstring`, `ezimage` ...). The Ibexa 5 database renames both: tables become `ibexa_*` with some columns renamed, and field types become `ibexa_*`. Both columns are given below where they differ.

17.2 Database schema reference

The Exponential 6.0 kernel schema has **128 tables**: `share/db_schema.dba`, `kernel/sql/mysql/kernel_schema.sql` and `kernel/sql/postgresql/kernel_schema.sql` list the same 128. `kernel/sql/sqlite/schema.sql` has three more, which belong to extensions but are created up front for SQLite: `ezgmaplocation` (ezgmaplocation), `ezstarrating` and `ezstarrating_data` (ezstarrating). The cluster tables (`ezdfsfile`, `ezdfsfile_cache`) are in the separate `kernel/sql/mysql/cluster_dfs_schema.sql`.

Legend for the last two columns: the table's name in the Ibexa DXP 4.6 legacy storage schema and in the Ibexa 5 reference database; "same" means the same name; "—" means the table does not exist there.

Content repository

Table	Purpose	Ibexa 4.6	Ibexa 5
`ezcontentobject`	One row per content object: class, owner, section, current version, language mask, remote id	same (adds `is_hidden`)	`ibexa_content` (`contentclass_id` is `content_type_id`; adds `is_hidden`)
`ezcontentobject_version`	Versions of an object, with status and creator	same	`ibexa_content_version`

Table	Purpose	Ibexa 4.6	Ibexa 5
ezcontentobject_attribute	Field values per object, version and language (data_text, data_int, data_float, sort_key_int, sort_key_string)	same	ibexa_content_field (contentclassattribute_id is content_type_field_definition_id)
ezcontentobject_name	Object name per version and language	same	ibexa_content_name
ezcontentobject_link	Relations between objects (relation_type 1 common, 2 embed, 4 link, 8 attribute)	same	ibexa_content_relation
ezcontentobject_tree	Nodes: the tree, with path_string, depth, sort field and order, visibility	same	ibexa_content_tree
ezcontentobject_trash	Nodes in the trash	same	ibexa_content_trash
eznode_assignment	Where a version will be placed when it is published	same	ibexa_node_assignment
ezcontent_language	Languages and their bit in the language masks	same	ibexa_content_language
ezcontentclass	Content classes per version	same	ibexa_content_type (version is status)
ezcontentclass_attribute	Class attributes with their datatype and settings (data_int1..4, data_text1..5, data_float1..4)	same (adds is_thumbnail)	ibexa_content_type_field_definition
ezcontentclass_name	Class names per language	same	ibexa_content_type_name
ezcontentclassgroup	Class groups	same (adds is_system)	ibexa_content_type_group
ezcontentclass_classgroup	Class to group links	same	ibexa_content_type_group_assignment
ezcobj_state, ezcobj_state_group, ezcobj_state_language, ezcobj_state_group_language, ezcobj_state_link	Object states, their groups, translations and the state of each object	same	ibexa_object_state, ibexa_object_state_group, ibexa_object_state_language, ibexa_object_state_group_language, ibexa_object_state_link
ezsection	Sections	same	ibexa_section
ezsite_data	Key and value pairs such as the schema version (ezpublish-version)	same	ibexa_site_data
ezpackage	Installed packages	same	ibexa_package

Table	Purpose	Ibexa 4.6	Ibexa 5
ezpending_actions	Deferred work such as index updates (action, param)	—	—
ezpublishingqueueprocesses	Asynchronous publishing queue	—	—

Datatype storage

Table	Purpose	Ibexa 4.6	Ibexa 5
ezbinaryfile	Files of ezbinaryfile attributes	same	ibexa_binary_file
ezmedia	Files and player settings of ezmedia attributes	same	ibexa_media
ezimagefile	Every file path an ezimage attribute uses (original and aliases)	same	ibexa_image_file
ezkeyword, ezkeyword_attribute_link	Keywords and their links to attributes	same (link adds version)	ibexa_keyword, ibexa_keyword_field_link (adds version)
ezurl, ezurl_object_link	External URLs and the attributes that use them	same	ibexa_url, ibexa_url_content_link
ezenumvalue, ezenumobjectvalue	ezenum options and chosen values	—	—
ezmultipricedata	Prices per currency of ezmultiprice attributes	—	—
ezisbn_group, ezisbn_group_range, ezisbn_registrant_range	ISBN-13 ranges for hyphenation and validation	—	—
ezproductcategory	Product categories (ezproductcategory datatype)	—	—
ezsubtree_notification_rule	Subtree subscriptions (ezsubtreesubscription datatype)	—	—
ezinfocollection, ezinfocollection_attribute	Collected information from forms	—	same (kept by the Netgen information collection bundle, not by Ibexa)

Users, roles and sessions

Table	Purpose	Ibexa 4.6	Ibexa 5
ezuser	Login, e-mail, password hash and hash type per user object	same (adds password_updated_at)	ibexa_user (adds password_updated_at)

Table	Purpose	Ibexa 4.6	Ibexa 5
ezuser_setting	Enabled flag and maximum concurrent logins	same	ibexa_user_setting
ezuser_accountkey	Account activation keys	same	ibexa_user_accountkey
ezuser_role	Role assignments with optional subtree or section limitation	same	ibexa_user_role
ezrole, ezpolicy, ezpolicy_limitation, ezpolicy_limitation_value	Roles, policies and their limitations	same	ibexa_role, ibexa_policy, ibexa_policy_limitation, ibexa_policy_limitation_value
ezpreferences	Per-user preferences	same	ibexa_user_preference
ezuservisit	Last visit, login count, failed login attempts	—	—
ezforgot_password	Password reset keys	—	—
ezsession	Database session storage	—	—
ezuser_discountrule	Discount rules per user (shop)	—	—
ezcontentbrowsebookmark	Bookmarks (Exponential adds folder_id, priority)	same (no folder columns)	ibexa_content_bookmark
ezcontentbrowserecent	Recently browsed nodes	—	—
expbookmark_folder	Bookmark folders (Exponential)	—	—

URLs and search

Table	Purpose	Ibexa 4.6	Ibexa 5
ezurlalias_ml	Multilingual URL aliases, one row per path element and language	same	ibexa_url_alias_ml
ezurlalias_ml_incr	Id sequence for ezurlalias_ml	same	ibexa_url_alias_ml_incr

Table	Purpose	Ibexa 4.6	Ibexa 5
ezurlalias	The pre-4.0 URL alias table, read only by the import in updateniceurls.php	same	ibexa_url_alias
ezurlwildcard	URL wildcards	same	ibexa_url_wildcard
ezsearch_word, ezsearch_object_word_link	The built-in search index	same	ibexa_search_word, ibexa_search_object_word_link
ezsearch_search_phrase	Search statistics	—	—

Workflows, notifications and collaboration

Table	Purpose	Ibexa 4.6	Ibexa 5
ezworkflow, ezworkflow_event, ezworkflow_group, ezworkflow_group_link, ezworkflow_assign, ezworkflow_process	Workflows, their events and running processes	—	—
eztrigger	Which workflow runs on which operation	—	—
ezoperation_memento, ezmodule_run	State of operations and modules interrupted by a workflow	—	—
ezwaituntildatevalue	Settings of the "wait until date" event	—	—
ezapprove_items	Items waiting for approval	—	—
ezcollab_group, ezcollab_item, ezcollab_item_group_link, ezcollab_item_message_link, ezcollab_item_participant_link, ezcollab_item_status, ezcollab_notification_rule, ezcollab_profile, ezcollab_simple_message	The collaboration system (approval messages)	—	—
eznotificationevent, eznotificationcollection, eznotificationcollection_item, ezgeneral_digest_user_settings	Notification events, mail queue and digest settings	—	—
ezmessage	Outgoing notification messages	—	—
ezscheduled_script	Progress of long-running scripts	—	—

Shop

Table	Purpose	Ibexa 4.6	Ibexa 5
ezbasket, ezwishlist	Baskets and wish lists	—	—
ezproductcollection, ezproductcollection_item, ezproductcollection_item_opt	Products in a basket or order	—	—

Table	Purpose	Ibexa 4.6	Ibexa 5
`ezorder`, `ezorder_item`, `ezorder_status`, `ezorder_status_history`, `ezorder_nr_incr`	Orders, their status and the order number sequence	—	—
`ezpaymentobject`	Payment gateway state	—	—
`ezcurrencydata`	Currencies and rates	—	—
`ezvattype`, `ezvatrule`, `ezvatrule_product_category`	VAT types and rules	—	—
`ezdiscountrule`, `ezdiscountsubrule`, `ezdiscountsubrule_value`	Discount rules	—	—

Other kernel features

Table	Purpose	Ibexa 4.6	Ibexa 5
`ezrss_export`, `ezrss_export_item`, `ezrss_import`	RSS feeds out and in	—	—
`ezrss_export_opml_item`	OPML export items (Exponential)	—	—
`ezpdf_export`	PDF exports	—	—
`eztipafriend_counter`, `eztipafriend_request`	Tip a friend	—	—
`ezview_counter`	Page view counts per node	—	—
`ezprest_clients`, `ezprest_authorized_clients`, `ezprest_authcode`, `ezprest_token`	OAuth clients and tokens of the REST API	—	—
`expaudit_event`, `expaudit_cursor`, `expaudit_file`	The audit trail (Exponential)	—	—
`expmail_category`, `expmail_preference`, `expmail_consent_log`, `expmail_pending`, `expmail_suppression`	Mail preferences, consent and suppression (Exponential)	—	—

Tables that exist only on the Symfony platform

Ibexa 4.6	Ibexa 5	Purpose
`ezcontentclass_attribute_ml`	`ibexa_content_type_field_definition_ml`	Translated field definition names, descriptions and settings
`eznotification`	`ibexa_notification`	Admin interface notifications
`ibexa_setting`	`ibexa_setting`	Platform settings
`ibexa_token`, `ibexa_token_type`	same	Tokens (for example password reset)
`ezdfsfile`	`ibexa_dfs_file`	DFS cluster metadata; in Exponential the same table comes from `kernel/sql/mysql/cluster_dfs_schema.sql`
`ezgmaplocation`	`ibexa_map_location`	Map locations; in Exponential the table belongs to the ezgmaplocation extension

Ibexa 4.6	Ibexa 5	Purpose
—	`ibexa_user_invitation`, `ibexa_user_invitation_assignment`	User invitations
—	`ibexa_messenger_messages`, `ibexa_messenger_lock_keys`	Symfony Messenger queue

Extensions bring their own tables, which keep their names on both sides when the extension has a platform counterpart: `eztags`, `eztags_keyword` and `eztags_attribute_link` (eztags and the Netgen Tags bundle) are present under those names in the Ibexa 5 reference database, and so is `sckenhancedselection` (enhancedselection2). The `ezm_block` and `ezm_pool` tables of `ezflow` (`extension/ezflow/sql/`) have no counterpart; see the Page field type in 17.3.

17.3 Datatypes and field types

The kernel registers 35 datatypes (`settings/content.ini` `[DataTypeSettings]` `AvailableDataTypes[]`). Each is implemented in `kernel/classes/datatypes/<name>/<name>type.php`, and the `data_type_string` below is that class's `DATA_TYPE_STRING`. The storage columns are those of `ezcontentobject_attribute`. The sort key is what `sortKeyType()` returns: `int` fills `sort_key_int`, `string` fills `sort_key_string`, and `false` fills neither (`kernel/classes/ezcontentobjectattribute.php`).

The Ibexa 4.6 identifier is the alias in `ibexa/core`'s `fieldtypes.yml` (or the bundle named); "Null" means 4.6 registers the identifier with the Null field type, which loads the stored value without understanding it. The Ibexa 5 identifier is the one in the field type reference or, where marked, the one stored in the reference database.

Datatype (`data_type_string`)	Stored in	Sort key	External tables	Ibexa 4.6	Ibexa 5	Notes
`ezauthor`	`data_text`: XML `<ezauthor><authors><author .../>`	—	—	`ezauthor`	`ibexa_author`	Same value: a list of name and e-mail pairs
`ezbinaryfile`	the `ezbinaryfile` row (attribute id and version)	—	`ezbinaryfile`	`ezbinaryfile`	`ibexa_binaryfile` (stored in the reference database)	Files under `<storage>/original/<mime group>/`; see 17.7
`ezboolean`	`data_int` 0 or 1	int	—	`ezboolean`	`ibexa_boolean` (stored)	Shown as "Checkbox" in the reference
`ezcountry`	`data_text`: comma-separated country codes	string	—	`ezcountry`	`ibexa_country`	Both store Alpha2 codes
`ezdate`	`data_int`: Unix timestamp	int	—	`ezdate`	`ibexa_date`	

Datatype (`data_type_string`)	Stored in	Sort key	External tables	Ibexa 4.6	Ibexa 5	Notes
`ezdatetime`	`data_int`: Unix timestamp	int	—	`ezdatetime`	`ibexa_datetime` (stored)	
`ezemail`	`data_text`	string	—	`ezemail`	`ibexa_email` (stored)	
`ezenum`	no attribute column; values in its tables	—	`ezenumvalue`, `ezenumobjectvalue`	Null	none	Convert to `ezselection` first with `bin/php/convertezenumtoezselection.php`
`ezfloat`	`data_float`	—	—	`ezfloat`	`ibexa_float`	
`ezidentifier`	`data_text` (formatted identifier), `data_int` (sequence number)	string	—	Null	none	
`ezimage`	`data_text`: XML `<ezimage>` with an `<original>` element and one `<alias>` per variation	—	`ezimagefile`	`ezimage`	`ibexa_image` (stored)	The reference database stores the same `<ezimage serial_number=... alias_key=...><original attribute_id=... /></ezimage>` XML; the paths differ, see 17.7
`ezinisetting`	`data_text` (value), `data_int` (empty-array flag)	—	writes INI files	Null	none	Has no meaning outside the legacy kernel
`ezinteger`	`data_int`	int	—	`ezinteger`	`ibexa_integer` (stored)	
`ezisbn`	`data_text`	—	`ezisbn_*` (ISBN-13 ranges)	`ezisbn`	`ibexa_isbn`	Convert ISBN-10 values first with `bin/php/ezconvert2isbn13.php` if they should be ISBN-13
`ezkeyword`	no attribute column; keywords in its tables	—	`ezkeyword`, `ezkeyword_attribute_link`	`ezkeyword`	`ibexa_keyword`	The platform link table has a `version` column the Exponential one lacks

Datatype (`data_type_string`)	Stored in	Sort key	External tables	Ibexa 4.6	Ibexa 5	Notes
`ezmatrix`	`data_text`: XML `<ezmatrix>` with columns and cells	—	—	`ezmatrix` (Null in core, implemented by `ibexa/fieldtype-matrix`)	`ibexa_matrix` (`ibexa/fieldtype-matrix`)	
`ezmedia`	the `ezmedia` row	—	`ezmedia`	`ezmedia`	`ibexa_media` (stored)	Same table columns, including the player settings
`ezmultioption`	`data_text`: XML `<ezmultioption>`	—	—	Null	none	Shop datatype
`ezmultioption2`	`data_text`: XML `<ezmultioption2>`	—	—	Null	none	Shop datatype
`ezmultiprice`	`data_text`: `"<VAT type>,<VAT included>"`; prices in its table	—	`ezmultipricedata`	Null	none	
`ezobjectrelation`	`data_int`: related object id	int	`ezcontentobject_link` (type 8)	`ezobjectrelation`	`ibexa_object_relation` (stored)	
`ezobjectrelationlist`	`data_text`: XML `<related-objects><relation-list><relation-item .../>`	—	`ezcontentobject_link` (type 8)	`ezobjectrelationlist`	`ibexa_object_relation_list` (stored)	The reference database stores the same `<related-objects><relation-list>` XML
`ezoption`	`data_text`: XML `<ezoption>`	—	—	Null	none	Shop datatype
`ezpackage`	`data_text`: package name	string	—	Null	none	
`ezprice`	`data_float` (price), `data_text` (`"<VAT type>,<VAT included>"`)	int (price × 100)	—	not registered	none	Convert to `ezmultiprice` with `bin/php/convertprice2multiprice.php` only if a multi-currency shop stays on Exponential; there is no platform target

Datatype (`data_type_string`)	Stored in	Sort key	External tables	Ibexa 4.6	Ibexa 5	Notes
`ezproductcategory`	`data_int`: category id	int	`ezproductcategory`	Null	none	
`ezrangeoption`	`data_text`: XML `<ezrangeoption>`	—	—	Null	none	Shop datatype
`ezselection`	`data_text`: selected option ids joined with `-`; options in the class attribute's `data_text5`	string	—	`ezselection`	`ibexa_selection`	
`ezstring`	`data_text`	string	—	`ezstring`	`ibexa_string` (stored)	"TextLine" in the reference
`ezsubtreesubscription`	`data_int` flag	—	`ezsubtree_notification_rule`	Null	none	
`eztext`	`data_text`	—	—	`eztext`	`ibexa_text` (stored)	"TextBlock" in the reference
`eztime`	`data_int`: seconds since midnight	int	—	`eztime`	`ibexa_time`	Both count seconds from the start of the day
`ezurl`	`data_int`: `ezurl.id`; `data_text`: link text	—	`ezurl`, `ezurl_object_link`	`ezurl`	`ibexa_url` (stored)	The reference database also keeps the URL id in `data_int`
`ezuser`	`ezuser` row; `data_text` holds a JSON draft only while the version is a draft	—	`ezuser`, `ezuser_setting`, `ezuser_accountkey`, `ezuservisit`	`ezuser`	`ibexa_user` (stored)	See the hash types in 17.7
`ezxmltext`	`data_text`: XML `<section xmlns:image=... xmlns:xhtml=... xmlns:custom=...>`	—	`ezurl`/`ezurl_object_link` for links, `ezcontentobject_link` types 2 and 4 for embeds and links	not registered	none	The platform uses RichText (`ezrichtext` in 4.6, `ibexa_richtext` in 5), stored as DocBook `<section xmlns="http://docbook.org/ns/docbook" ...>`; see the notes below

Extension datatypes found in this installation:

Datatype	Extension	Stored in	Ibexa 4.6	Ibexa 5 / platform package
`eztags`	eztags	tables `eztags`, `eztags_keyword`, `eztags_attribute_link`	Null in core; implemented by the Netgen Tags bundle	`eztags` (stored in the reference database, Netgen Tags bundle)
`ezgmaplocation`	ezgmaplocation	`data_int` (has location), table `ezgmaplocation`	`ezgmaplocation`	`ibexa_gmap_location`, table `ibexa_map_location`
`ezsrrating`	ezstarrating	`data_int` (disabled flag), tables `ezstarrating`, `ezstarrating_data`	not registered	none
`ezpage`	ezflow	`data_text`: XML <page>; tables `ezm_block`, `ezm_pool`	not registered	none in the open-source edition
`sckenhancedselection`	enhancedselection2	table `sckenhancedselection`; options in the class attribute's `data_text5`	not registered	`sckenhancedselection` (stored in the reference database, community bundle)
`ngenhancedlink`	exp_enhanced_link	`data_text` (JSON), `data_int` (internal object id or 0)	not registered	`ngenhancedlink` (stored in the reference database, Netgen bundle)
`ngclasslist`	ngclasslist	(not verified here)	not registered	`ngclasslist` (stored in the reference database)

Notes on the conversions the vendor documents

- **XmlText to RichText.** The vendor's migration page lists XmlText as replaced by RichText and as not supported on the platform; content is converted with `ezxmltext:convert-to-richtext` from `ezsystems/ezplatform-xmltext-fieldtype` (options `--dry-run`, `--export-dir`, `--export-dir-filter`, `--image-content-types`, `--concurrency`). RichText validates more strictly than XmlText, so run the dry run first. No such converter exists in this repository, in either direction; Exponential renders the DocBook elements `ezembed`, `para` and `literallayout` when they occur in imported `ezxmltext` (`kernel/classes/datatypes/ezxmltext/handlers/output/ezxhtmlxmloutput.php`), which keeps such text readable but does not convert it. The element mapping for a conversion of your own is in 15.8 and 16.6.3.
- **Custom tags.** Custom tag attribute types of the online editor (`extension/ezoe/design/standard/templates/ezoe/customattributes/`) map to RichText custom tag attribute types as the vendor documents:

ezoe attribute type	RichText type	Vendor note
`link`	`link`	supported
`number`, `int`	`number`	
`checkbox`	`boolean`	

ezoe attribute type	RichText type	Vendor note
`select`	`choice`	
`text`	`string`	
`textarea`, `email`, `hidden`, `color`, `htmlsize`, `csssize`, `csssize4`, `cssborder`	`string`	"use as workaround"

The tags themselves are declared in `settings/content.ini` `[CustomTagSettings]` `AvailableCustomTags[]` (`factbox`, `quote`, `strike`, `sub`, `sup` by default) and in extensions' `content.ini.append.php`.

- **Page (`ezflow`).** Replaced by the Page field type of the commercial editions; the open-source platform has no target. The vendor's `ezflow:migrate` command belongs to those editions.
- **Star rating (`ezsrrating`).** Listed by the vendor as unsupported: implement it or remove it from every class.
- **Custom datatypes.** Rewrite them as field types, or register the Null field type for their identifier so the stored values load without errors. Ibexa 4.6 already does this for the identifiers marked Null above.

17.4 Settings: INI and YAML

Only pairs that can be shown to describe the same thing are listed. The YAML keys were checked against the generated configuration reference and the `ibexa.yaml` files of the Ibexa 5 reference installation; the database and file pairs are also the ones the legacy bridge maps automatically (`doc/eng-us/specifications/6.0/legacy-bridge-bundle.md`). `<scope>` is a siteaccess, a siteaccess group or `default`, under `ibexa.system`.

Siteaccesses, languages and design

INI (file, section, key)	YAML	Notes
`site.ini [SiteAccessSettings] AvailableSiteAccessList[]`	`ibexa.siteaccess.list`	
`site.ini [SiteSettings] DefaultAccess`	`ibexa.siteaccess.default_siteaccess`	
(no equivalent; siteaccesses share settings through `settings/override`)	`ibexa.siteaccess.groups`	Shared configuration goes on the group
`site.ini [SiteAccessSettings] MatchOrder`, `URIMatchType=element`	`ibexa.siteaccess.match: { URIElement: 1 }`	
`site.ini [SiteAccessSettings] HostMatchType=map`, `HostMatchMapItems[]=host;siteaccess`	`ibexa.siteaccess.match: { Map\Host: { host: siteaccess } }`	
`site.ini [RegionalSettings] SiteLanguageList[]`	`ibexa.system.<scope>.languages`	Ordered, first is the main language
`site.ini [RegionalSettings] TranslationSA[]`	`ibexa.system.<scope>.translation_siteaccesses`	

INI (file, section, key)	YAML	Notes
site.ini [SiteAccessSettings] PathPrefix	ibexa.system.<scope>.content.tree_root.location_id	The legacy bridge derives one from the other
site.ini [SiteSettings] IndexPage, DefaultPage	ibexa.system.<scope>.index_page, default_page	
site.ini [DesignSettings] SiteDesign	ibexa.system.<scope>.design	
site.ini [DesignSettings] AdditionalSiteDesignList[], StandardDesign	ibexa_design_engine.design_list.<design>: [themes...]	The fallback order of themes inside one design
pagelayout.tpl of the design	ibexa.system.<scope>.page_layout	
override.ini blocks (Source, MatchFile, Match[class_identifier] ...)	ibexa.system.<scope>.content_view.<view type>.<name>: { template, controller, match }	See 17.5

Database, files and clustering

INI	YAML or environment	Notes
site.ini [DatabaseSettings] Server, Port, User, Password, Database, Socket	Doctrine connection: DATABASE_URL, or doctrine.dbal host, port, user, password, dbname, unix_socket	
site.ini [DatabaseSettings] DatabaseImplementation	Doctrine driver: ezmysqli = pdo_mysql, ezpostgresql = pdo_pgsql, ezoracle = oci8, sqlite3 = pdo_sqlite	The legacy bridge's driver map
site.ini [FileSettings] VarDir	ibexa.system.<scope>.var_dir	Exponential siteaccesses use var/site; so does the reference installation
site.ini [FileSettings] StorageDir	ibexa.system.<scope>.storage_dir	Default storage on both sides
(fixed original in the binary file and media datatypes)	ibexa.system.<scope>.binary_dir	Default original
file.ini [ClusteringSettings] FileHandler=eZDFSFileHandler with [eZDFSClusteringSettings] MountPointPath, DBBackend, DBHost, DBName ...	ibexa_io.metadata_handlers.dfs.legacy_dfs_cluster plus a doctrine.dbal.connections.dfs connection, and ibexa_io.binarydata_handlers.nfs (Flysystem) with the NFS path	
image.ini [ImageMagick] ExecutablePath, Executable	ibexa.imagemagick.path	
image.ini [AliasSettings] AliasList[] and one [<alias>] block each with Reference and Filters[]	ibexa.system.<scope>.image_variations.<alias>: { reference, filters: [{ name, params }] }	Filter names match: geometry/scaledownonly=100;100 becomes { name: geometry/scaledownonly, params: [100, 100] }

Users, sessions, URLs, caches and search

INI	YAML or environment	Notes
`site.ini [UserSettings] AnonymousUserID`	`ibexa.system.<scope>.anonymous_user_id`	
`site.ini [Session] SessionNamePrefix`, `SessionNamePerSiteAccess`	`ibexa.system.<scope>.session.name`	Use the `{siteaccess_hash}` token for a name per siteaccess
`site.ini [URLTranslator] TransformationGroup`	`ibexa.url_alias.slug_converter.transformation`	Exponential has `urlalias`, `urlalias_iri` and `urlalias_compat` in `settings/transform.ini`; the vendor names `urlalias_compat` and `urlalias_iri` for compatibility with legacy aliases
`site.ini [URLTranslator] WordSeparator`	`ibexa.url_alias.slug_converter.separator`	
`site.ini [SearchSettings] SearchEngine`	`ibexa.repositories.default.search.engine` (`legacy` or `solr`)	
`content.ini [VersionManagement] DefaultVersionHistoryLimit`	`ibexa.repositories.default.options.default_version_archive_limit`	
`site.ini [ContentSettings] ViewCaching`	`ibexa.system.<scope>.content.view_cache`	
`httpcache.ini [HttpCacheSettings] MaxAge`	`ibexa.system.<scope>.content.default_ttl`	Exponential's role-aware HTTP cache is its own; see HTTP caching
`httpcache.ini [HttpCacheSettings] TagHeader` (`xkey`, `Surrogate-Key`, `Cache-Tag`)	`ibexa.http_cache.purge_type` and `ibexa.system.<scope>.http_cache.purge_servers`	Different mechanisms: Exponential tags pages with Ibexa-style tags (`c<object>`, `l<node>`, `pl<parent>`, `p<path node>`, `ct<class>`, `s<section>`, `ez-all`) and purges its own store; the platform sends purges to Varnish or its local proxy
`site.ini [MailSettings] Transport`, `TransportServer`	`framework.mailer.dsn` (`MAILER_DSN`)	
`site.ini [DebugSettings] DebugOutput`	`APP_DEBUG` / `kernel.debug`	
`site.ini [ExtensionSettings] ActiveExtensions[]`	`config/bundles.php`	An extension and a bundle play the same role; neither loads the other

Settings without a platform counterpart include `site.ini [UserSettings] HashType` and `UpdateHash` (the platform stores the hash type per user, see 17.7), `[TemplateSettings]` (template compilation), `[SiteAccessSettings] RequireUserLogin`, and every shop, workflow and notification setting.

17.5 Templates: TPL and Twig

The TPL constructs below are defined in `lib/eztemplate/classes/eztemplateautoload.php` (language functions) and `kernel/common/eztemplateautoload.php` (kernel operators and functions). The Twig names are those of Ibexa 4.6 and 5, as used in the reference installation's

templates; eZ Platform used the same names with an `ez_` prefix (`ez_render_field`, `ez_path` ...), which Ibexa 4.6 still accepts.

TPL	Twig	Notes
`{def $x = 1}`, `{set $x = 2}`, `{undef $x}`	`{% set x = 1 %}`	Twig variables need no declaration
`{if $a}...{elseif $b}...{else}...{/if}`	`{% if a %}...{% elseif b %}...{% else %}...{% endif %}`	
`{foreach $items as $item}...{/foreach}`	`{% for item in items %}...{% endfor %}`	
`cond( $a, 'x', 'y' )`, `first_set( $a, $b )`	`a ? 'x' : 'y'`, `a ?? b`	
`{include uri='design:parts/menu.tpl' node=$node}`	`{% include '@ibexadesign/parts/menu.html.twig' with { node: node } %}`	`design:` resolves through the design fallback, `@ibexadesign` through `ibexa_design_engine.design_list`
`{$text|wash}`	`{{ text }}`	Twig escapes by default; `|raw` turns that off
`{'Read more'|i18n( 'design/site' )}`	`{{ 'Read more'|trans({}, 'design') }}`	Translation domains replace i18n contexts
`{$node.url_alias|ezurl}`	`{{ ibexa_path( location ) }}`	Relative URL of a node or location
`{$node.url_alias|ezurl( 'no', 'full' )}`	`{{ ibexa_url( location ) }}`	Absolute URL
`{'stylesheets/site.css'|ezdesign}`	`{{ asset( 'css/site.css' ) }}`	Symfony asset handling; `ezscript`/`ezcss` (ezjscore) have no direct counterpart
`{$node.name}`	`{{ ibexa_content_name( content ) }}`	
`{attribute_view_gui attribute=$node.data_map.title}`	`{{ ibexa_render_field( content, 'title' ) }}`	Field templates come from `field_templates`
`{$node.data_map.title.content}`	`{{ ibexa_field_value( content, 'title' ) }}`	
`{$node.data_map.title.has_content}`	`{{ not ibexa_field_is_empty( content, 'title' ) }}`	
`{$node.data_map.title.contentclass_attribute_name}`	`{{ ibexa_field_name( content, 'title' ) }}`	
`{$node.data_map.image.content['medium'].url}`	`{{ ibexa_image_alias( field, versionInfo, 'medium' ).uri }}`	Image alias = image variation
`{node_view_gui content_node=$child view='line'}`	`{{ render( controller( 'ibexa_content::viewAction', { contentId: child.contentInfo.id, location: child, viewType: 'line' } ) ) }}`	The template is chosen by `content_view` rules

TPL	Twig	Notes
`fetch( 'content', 'node', hash( 'node_id', 2 ) )`	No Twig function: load in a controller with `LocationService::loadLocation()`	
`fetch( 'content', 'list', hash( 'parent_node_id', $node.node_id, 'class_filter_type', 'include', 'class_filter_array', array( 'article' ), 'sort_by', $node.sort_array, 'limit', 10 ) )`	No Twig function: a query in a controller (`SearchService::findLocations()`) or a query type in the view configuration	The reference installation uses `queries:` with a `query_type` in `content_view` (Netgen Site API)
`fetch( 'user', 'current_user' )`	`app.user`	Symfony security user
`{cache-block keys=array( $uri_string ) subtree_expiry=$node.node_id}...{/cache-block}`	`{{ render_esi( controller( '...' ) ) }}` and HTTP cache tags (`ibexa_http_cache_tag_location( location )`)	No template-level cache on the platform: fragments are cached by the HTTP cache and expired by tags
`ezini( 'SiteSettings', 'SiteName' )`	A parameter or `ibexa.system.<scope>.twig_variables`	

`override.ini` and `content_view` express the same rule in two forms:

INI

```
# settings/siteaccess/site/override.ini.append.php
[full_article]
Source=node/view/full.tpl
MatchFile=full/article.tpl
Subdir=templates
Match[class_identifier]=article
```

YAML

```
ibexa:
    system:
        site_group:
            content_view:
                full:
                    article:
                        template: '@ibexadesign/full/article.html.twig'
                        match:
                            Identifier\ContentType: article
```

The legacy match keys for node views are set in `kernel/classes/eznodeviewfunctions.php` (`class_identifier`, `node`, `parent_node`, `section`, `section_identifier`, `depth`, `url_alias`, `state_identifier`, `parent_class_identifier`, `remote_id`, `node_remote_id`, among others). The platform matchers with the same meaning (classes in `Ibexa\Core\MVC\Symfony\Matcher\ContentBased`, Ibexa 4.6):

`override.ini` **key**	`content_view` **matcher**
`Match[class_identifier]`	`Identifier\ContentType`
`Match[class]`	`Id\ContentType`

override.ini **key**	content_view **matcher**
Match[node]	Id\Location
Match[object]	Id\Content
Match[parent_node]	Id\ParentLocation
Match[parent_class_identifier]	Identifier\ParentContentType
Match[section_identifier], Match[section]	Identifier\Section, Id\Section
Match[class_group]	Id\ContentTypeGroup
Match[remote_id], Match[node_remote_id]	Id\Remote, Id\LocationRemote
Match[depth]	Depth
Match[url_alias]	UrlAlias

17.6 PHP API: legacy classes and Public API services

The legacy methods are in `kernel/classes/` unless noted; the services are the interfaces in `Ibexa\Contracts\Core\Repository` (Ibexa 4.6, `ibexa/core` v4.6.28; on eZ Platform the namespace was `eZ\Publish\API\Repository`). Platform calls run as the current user set on `PermissionResolver`, where legacy code mostly ran with the logged-in user's rights and skipped checks in scripts.

eZContentObject and ContentService

Legacy	Public API
`eZContentObject::fetch( $id )`	`ContentService::loadContent( $id )`, `loadContentInfo( $id )`
`eZContentObject::fetchByRemoteID( $remoteID )`	`ContentService::loadContentByRemoteId()`, `loadContentInfoByRemoteId()`
`eZContentObject::fetchByNodeID( $nodeID )`	`LocationService::loadLocation( $id )->getContentInfo()`
`$object->dataMap()`	`$content->getFields()`, `$content->getField( 'identifier' )`
`eZContentFunctions::createAndPublishObject( $params )`	`ContentService::newContentCreateStruct()`, `createContent( $struct, [ $locationCreateStruct ] )`, `publishVersion( $versionInfo )`
`$object->createNewVersion()`	`ContentService::createContentDraft( $contentInfo )`
`eZContentFunctions::updateAndPublishObject( $object, $params )`	`ContentService::newContentUpdateStruct()`, `updateContent()`, `publishVersion()`
`eZOperationHandler::execute( 'content', 'publish', ... )` (`lib/ezutils/classes/ezoperationhandler.php`)	`ContentService::publishVersion()`
`eZContentObjectVersion::fetchVersion( $version, $objectID )`	`ContentService::loadVersionInfo( $contentInfo, $versionNo )`, `loadVersions()`
`$object->removeThis()`, `$object->purge()`	`ContentService::deleteContent( $contentInfo )`

Legacy	Public API
`eZContentOperationCollection::deleteObject( $ids, $moveToTrash )` (kernel/content/)	`TrashService::trash( $location )` or `ContentService::deleteContent()`
`$object->relatedObjects()` and `ezcontentobject_link`	`ContentService::loadRelationList()`, `loadReverseRelationList()`, `addRelation()`, `deleteRelation()`

The legacy class has no `publish()` method; publishing goes through the `publish` operation or `eZContentFunctions`.

eZContentObjectTreeNode and LocationService

Legacy	Public API
`eZContentObjectTreeNode::fetch( $nodeID )`	`LocationService::loadLocation( $id )`
`eZContentObjectTreeNode::fetchByRemoteID( $remoteID )`	`LocationService::loadLocationByRemoteId()`
`eZContentObjectTreeNode::fetchByContentObjectID( $objectID )`	`LocationService::loadLocations( $contentInfo )`
`eZContentObjectTreeNode::subTreeByNodeID( $params, $nodeID )`, `$node->children()`	`LocationService::loadLocationChildren()`; filtered lists with `SearchService::findLocations()`
`eZContentObjectTreeNode::subTreeCountByNodeID( $params, $nodeID )`	`LocationService::getLocationChildCount()`, `getSubtreeSize()`
`eZContentObjectTreeNodeOperations::move( $nodeID, $newParentNodeID )`	`LocationService::moveSubtree( $location, $newParent )`
`eZContentOperationCollection::swapNode( $nodeID, $selectedNodeID )`	`LocationService::swapLocation()`
`eZContentObjectTreeNode::hideSubTree( $node )`, `unhideSubTree( $node )`	`LocationService::hideLocation()`, `unhideLocation()`
`eZContentObjectTreeNode::removeSubtrees( $ids, $moveToTrash )`	`TrashService::trash()` or `LocationService::deleteLocation()`
`ezsubtreecopy.php`	`LocationService::copySubtree()`
`eZContentObjectTrashNode::trashList()`	`TrashService::findTrashItems()`, `recover()`, `emptyTrash()`

eZUser and UserService

Legacy (`kernel/classes/datatypes/ezuser/ezuser.php`)	Public API
`eZUser::fetch( $id )`	`UserService::loadUser( $id )`
`eZUser::fetchByName( $login )`	`UserService::loadUserByLogin( $login )`
`eZUser::fetchByEmail( $email )`	`UserService::loadUserByEmail()` (one user), `loadUsersByEmail()`
`eZUser::loginUser( $login, $password )`	`UserService::checkUserCredentials( $user, $password )`; signing in is done by Symfony security

Legacy (kernel/classes/datatypes/ezuser/ezuser.php)	Public API
eZUser::currentUser(), currentUserID()	PermissionResolver::getCurrentUserReference()
eZUser::setCurrentlyLoggedInUser($user, $userID)	PermissionResolver::setCurrentUserReference($user)
eZUser::create($objectID) with $user->setInformation(...)	UserService::newUserCreateStruct(), createUser($struct, $userGroups)
$user->hasAccessTo($module, $function)	PermissionResolver::hasAccess($module, $function), canUser()

eZContentClass and ContentTypeService

Legacy	Public API
eZContentClass::fetch($id)	ContentTypeService::loadContentType($id)
eZContentClass::fetchByIdentifier($identifier)	ContentTypeService::loadContentTypeByIdentifier( )
eZContentClass::fetchByRemoteID()	ContentTypeService::loadContentTypeByRemoteId()
eZContentClass::fetchList()	ContentTypeService::loadContentTypes($group), loadContentTypeList()
eZContentClass::create(), eZContentClassAttribute::create($classID, $dataTypeString), storeDefined()	ContentTypeService::newContentTypeCreateStruct(), newFieldDefinitionCreateStruct(), createContentType(), publishContentTypeDraft()
$class->instantiate()	ContentService::newContentCreateStruct($contentType, $language)

Search, URL aliases, roles and the rest

Legacy	Public API
eZSearch::search($text, $params)	SearchService::findContent(), findLocations(), findSingle()
eZSearch::addObject($object), removeObjectById($id)	No public method: the search engine is updated by the repository on publish and delete
eZURLAliasML::fetchByPath($uri), eZURLAliasML::translate($uri)	URLAliasService::lookup($url)
eZURLAliasML::fetchByAction('eznode', $nodeID)	URLAliasService::reverseLookup($location), listLocationAliases()
eZURLAliasML::storePath($path, $action)	URLAliasService::createUrlAlias(), createGlobalUrlAlias()
eZURLWildcard::fetchList(), translate()	URLWildcardService::loadAll(), translate(), create()
eZRole::fetch($id), fetchByName($name)	RoleService::loadRole(), loadRoleByIdentifier()
eZRole::create($name), $role->appendPolicy($module, $function, $limitations)	RoleService::createRole(), addPolicyByRoleDraft(), publishRoleDraft()

Legacy	Public API
`$role->assignToUser( $userID, $limitIdent, $limitValue )`	`RoleService::assignRoleToUser()`, `assignRoleToUserGroup()`
`$role->removeUserAssignment( $userID )`	`RoleService::removeRoleAssignment()`
`eZSection::fetch( $id )`, `fetchByIdentifier()`, `$section->applyTo( $object )`	`SectionService::loadSection()`, `loadSectionByIdentifier()`, `assignSection()`, `assignSectionToSubtree()`
`eZContentObjectStateGroup::newState()`, `eZContentObjectState::fetchByIdentifier()` (`kernel/private/classes/`)	`ObjectStateService::createObjectState()`, `loadObjectStateByIdentifier()`, `setContentState()`
`eZContentLanguage::fetchByLocale( $locale )`	`LanguageService::loadLanguage( $languageCode )`
`eZPersistentObject::fetchObjectList()` on kernel tables	No equivalent: the platform reads its tables only through the services; custom tables use Doctrine
`eZWorkflow`, `eZTrigger::runTrigger()`	No equivalent: the platform raises events (Symfony event subscribers) before and after each service call **(not verified here)**

17.7 Passwords, URL aliases and image storage

Password hash types

`ezuser.password_hash_type` holds one of the constants of `eZUser` (`kernel/classes/datatypes/ezuser/ezuser.php`); `eZUser::createHash()` computes each one:

Value	Constant	Identifier (`HashType`)	Hash	Exponential	Ibexa 4.6 / 5
0	`PASSWORD_HASH_EMPTY`	`empty`	none: the account cannot sign in	yes	no
1	`PASSWORD_HASH_MD5_PASSWORD`	`md5_password`	`md5( password )`	yes	no
2	`PASSWORD_HASH_MD5_USER`	`md5_user`	`md5( "login\npassword" )`	yes	no
3	`PASSWORD_HASH_MD5_SITE`	`md5_site`	`md5( "login\npassword\nsite" )`, site = `[UserSettings] SiteName`	yes	no
4	`PASSWORD_HASH_MYSQL`	`mysql`	MySQL `PASSWORD()`, removed in MySQL 8.0	yes	no
5	`PASSWORD_HASH_PLAINTEXT`	`plaintext`	the password itself	yes	no
6	`PASSWORD_HASH_BCRYPT`	`bcrypt`	`password_hash( PASSWORD_BCRYPT )`	yes	yes
7	`PASSWORD_HASH_PHP_DEFAULT`	`php_default`	`password_hash( PASSWORD_DEFAULT )`	yes (default)	yes (default)

Value	Constant	Identifier (HashType)	Hash	Exponential	Ibexa 4.6 / 5
256	—	—	invalid, set on purpose to block a login	no	PASSWORD_HASH_INVALID (Ibexa 5)

Ibexa 4.6 supports only 6 and 7 (SUPPORTED_PASSWORD_HASHES); Ibexa 5 adds 256. ezsystems/ezpublish-kernel 7.5 (eZ Platform 2.5) still declared 1, 2, 3, 5, 6 and 7, the MD5 and plain text types deprecated since kernel 6.13. All users in the Ibexa 5 reference database have type 7.

Exponential re-hashes on sign-in: with site.ini [UserSettings] UpdateHash=true (the default) and HashType=php_default (the default), eZUser::_loginUser() replaces any other stored type with a type 7 hash after a successful password check, updates password_hash and password_hash_type, and clears the user's cache. Users who have signed in since the site came to Exponential therefore already have a hash the platform accepts. To list the rest before migrating:

SQL

```sql
SELECT password_hash_type, COUNT(*) FROM ezuser GROUP BY password_hash_type;
```

Accounts still on types 0 to 5 cannot sign in on the platform; they need a password reset there. ezuser.password_hash is VARCHAR(255) in the Exponential schema (kernel/sql/mysql/kernel_schema.sql), as on the platform, so bcrypt and Argon2 hashes fit.

URL alias tables

Table	Holds
ezurlalias_ml	One row per path element and language: parent, text, text_md5, action (eznode:<node id>, module:<path> or nop:), action_type, link, id, is_original, is_alias, alias_redirects, lang_mask
ezurlalias_ml_incr	The id sequence
ezurlwildcard	Wildcards: source_url, destination_url, type
ezurlalias	The pre-4.0 format, only read by the import in updateniceurls.php

The platform tables have the same columns under the same names (Ibexa 4.6) or ibexa_url_alias_ml, ibexa_url_alias_ml_incr and ibexa_url_wildcard (Ibexa 5), so aliases move with the data. They are regenerated afterwards with ibexa:urls:regenerate-aliases (exponential:urls:regenerate-aliases on Exponential Platform v5, which registers no ibexa: alias). The slug transformation must match: Exponential's default is urlalias ([URLTranslator] TransformationGroup), the Ibexa 5 reference installation uses urlalias_lowercase.

Image and file storage paths

Item	Exponential	Platform
Storage root	<VarDir>/<StorageDir>: var/site/storage with the siteaccesses' VarDir=var/site	<var_dir>/<storage_dir>: var/site/storage in the reference installation

Item	Exponential	Platform
Published images	`<storage>/images/<node path with names>/<attribute id>-<version>-<language>/<name>.<ext>` (`image.ini [FileSettings] PublishedImages=images`)	`<storage>/images/<id digits reversed, one per directory>/<field id>-<version>-<language>/<name>.<ext>`, for example `var/site/storage/images/0/0/3/2/2300-11-eng-GB/...`
Draft images	`<storage>/images-versioned/<attribute id>/<version>-<language>/` (`VersionedImages=images-versioned`)	the same directory scheme as published images
Image aliases	`<basename>_<alias>.<ext>` beside the original	variations generated on demand, named by appending `_<variation>` to the file name
Paths of an image	`data_text` XML (`dirpath`, `url`, `<alias>`) plus every path in `ezimagefile.filepath`	`data_text` XML plus `ibexa_image_file.filepath`
Binary files and media	`<storage>/original/<mime group>/<md5 name>.<ext>`	`<storage>/original/...` (`binary_dir`)

The vendor's instructions copy `<old root>/web/var[/<site>]/storage` to the same place in the new installation; image paths stored in the XML and in `ezimagefile` must then point to files that exist, which the checks in 17.8 find.

17.8 Data checks before migrating

The vendor lists problems that appear after a migration. Each can be looked for in the Exponential database first. The SQL runs on MySQL, PostgreSQL and SQLite; run it on a copy.

Issue	How to detect it in Exponential	Fix
Nodes sorted by class identifier or class name, which the platform does not support	`SELECT node_id, parent_node_id, sort_field FROM ezcontentobject_tree WHERE sort_field IN (6, 7);` (6 = `SORT_FIELD_CLASS_IDENTIFIER`, 7 = `SORT_FIELD_CLASS_NAME` in `eZContentObjectTreeNode`)	Change the sort order of those nodes in the admin to name (9), published (2) or priority (8)
URL aliases out of date	`php bin/php/verify_aliases.php`	`php bin/php/verify_aliases.php --fix` and `php bin/php/updateniceurls.php --update-nodes`; on the platform, `ibexa:urls:regenerate-aliases`
Image paths outside the storage directory or with unusual characters	`php update/common/scripts/5.1/fiximagesoutsidevardir.php --dry-run`; `SELECT filepath FROM ezimagefile WHERE filepath NOT LIKE 'var/site/storage/%';` (use your `VarDir`)	`fiximagesoutsidevardir.php` without `--dry-run`, `update/common/scripts/5.3/recreateimagesreferences.php`; on the platform, `ezplatform:images:normalize-path`

Issue	How to detect it in Exponential	Fix
Relations with type 0 ("Unknown relation type 0")	`SELECT COUNT(*) FROM ezcontentobject_link WHERE relation_type = 0;` (valid values are 1, 2, 4, 8 and their sums)	On the platform, `ezpublish:update:legacy_storage_clean_up_relation_type_eq_zero`. In Exponential the 5.1 update file already deleted such rows; rows that appeared later are ignored by every relation query and can be removed on a backed-up database with `DELETE FROM ezcontentobject_link WHERE relation_type = 0;` (15.20)
"Always available" set on fields of every language, not only the main one	`SELECT COUNT(*) FROM ezcontentobject_attribute a JOIN ezcontentobject o ON o.id = a.contentobject_id WHERE (a.language_id & 1) = 1 AND (a.language_id & ~1) <> (o.initial_language_id & ~1);` (bit 1 of a language id is the always-available flag; Oracle: `bitand()`)	On the platform, `ezpublish:update:legacy_storage_fix_fields_always_available_flag`
Empty `sort_key_string`	`SELECT data_type_string, COUNT(*) FROM ezcontentobject_attribute WHERE sort_key_string = '' AND data_text <> '' AND data_type_string IN ('ezstring', 'ezemail', 'ezcountry', 'ezidentifier', 'ezselection', 'ezpackage') GROUP BY data_type_string;`	On the platform, `ezpublish:update:legacy_storage_update_sort_keys`
Datatypes the platform cannot store	`SELECT data_type_string, COUNT(*) FROM ezcontentclass_attribute WHERE version = 0 GROUP BY data_type_string;` and compare with 17.3	Convert (`ezenum`, `ezxmltext`), remove, or register a Null field type
Orphaned URL links	`php update/common/scripts/5.4/fixremovedezurlobjectlinks.php` (without `--fix` it only reports)	the same script with `--fix`
Duplicate remote ids (the platform loads by remote id and expects them unique)	`SELECT remote_id, COUNT(*) FROM ezcontentobject GROUP BY remote_id HAVING COUNT(*) > 1;` (the same for `ezcontentobject_tree` and `ezcontentclass`)	`update/common/scripts/4.1/fixobjectremoteid.php`, `fixnoderemoteid.php`, `fixclassremoteid.php`: `--mode=a` fixes automatically, `--mode=d` shows the details, no `--mode` asks per remote id
Character set not UTF-8	`site.ini [DatabaseSettings] Charset` and the table collations	`php bin/php/ezconvertdbcharset.php`

The platform command names are those the vendor documents; their prefix changes between platform versions.

17.9 Migration and upgrade tooling

Every `bin/php/<name>.php` is also `./console exp:<name>`; the code of most of them is in `kernel/private/classes/commands/`. The standard options (`-s/--siteaccess`, `-q`, `-v`, `-d`, `--allow-root-user`) apply to all eZScript-based scripts. Chapter 11 explains the order to run

them in.

Database update files

`update/database/<engine>/<version>/dbupdate-<from>-to-<to>.sql`, for `mysql`, `postgresql` and `sqlite`. MySQL and PostgreSQL carry the chain from `4.0/dbupdate-3.10.0-to-4.0.0.sql` through the 5.x steps (with `unstable/` alpha, beta and rc steps for 4.0 to 5.0, and `dbupdate-cluster-*` files for the MySQL cluster at 4.3, 4.7, 5.2 and 5.4), then the 6.0 directory with the step from 5.4 (`dbupdate-5.4.0-6.0.0.sql` on MySQL, `dbupdate-5.4-to-6.0.0.sql` on PostgreSQL) and `dbupdate-6.0.0-6.0.15.sql`. SQLite has only `6.0/dbupdate-6.0.0-6.0.15.sql`. Every MySQL file opens with `SET default_storage_engine=InnoDB;` (accepted by MySQL from 5.5.3 and by MariaDB); copies from before October 2026 said `SET storage_engine`, which current servers reject.

The `6.12/`, `7.2/` (PostgreSQL only) and `7.3/` directories come from the upstream legacy line of 2016 to 2018 and are numbered after the Symfony kernel version they ran beside; never apply them as files. Their three schema changes (`ezuser.password_hash` widened to 255, the PostgreSQL sequences renamed to `<table>_<column>_seq`, `ezcontentobject_trash.trashed`) are made by the 5.4 to 6.0.0 file and again by the 6.0.0 to 6.0.15 file, each only where the database lacks it (11.3). Oracle has no 6.0 file: run the `password_hash` and `trashed` statements of 11.5 by hand. `php bin/php/console exp:checkdbfiles --no-verify-branches` checks that the tree is complete (table below).

update/common/scripts

Script	Purpose	Options
`cleanup.php`	Deletes data by name: `session`, `expired_session`, `preferences`, `browse`, `tipafriend`, `shop`, `forgotpassword`, `workflow`, `collaboration`, `collectedinformation`, `notification`, `searchstats`, `all`	`--db-host`, `--db-user`, `--db-password`, `--db-database`, `--db-type`, `--sql`, name
`updatecontentobjectname.php`	Rebuilds every object name	`-h`, `-q`
`updatenbxmlcontents.php`	Fixes non-breaking space encoding in XML text	`--dry-run`, `-n`, `-v`, `--iteration-sleep`, `--iteration-limit`
`4.0/updatebinaryfile.php`	Adds the file suffix to stored binary files	none
`4.0/updatetipafriendpolicy.php`	Updates the tip a friend policy	siteaccess, user
`4.0/updatevatcountries.php`	Updates VAT countries	none
`4.1/addlockstategroup.php`	Adds the `ez_lock` state group	none
`4.1/correctxmlalign.php`	Moves embed and custom align attributes to `align`	`--skip-embed-align`, `--skip-custom-align`, `--custom-align-attribute`, `--db-*`
`4.1/fixclassremoteid.php`, `fixnoderemoteid.php`, `fixobjectremoteid.php`	Make remote ids unique	`--mode` (d detailed, a automatic)

Script	Purpose	Options
4.1/fixezurlobjectlinks.php	Re-links URLs for every version and translation	--fix, --fetch-limit
4.1/initurlaliasmlid.php	Initialises ezurlalias_ml_incr	none
4.1/updateimagesystem.php	Moves image attributes to the 3.3 image system	none
4.2/fixorphanimages.php	Repairs images left by an alias handler bug	-n
4.3/updatenodeassignment.php	Updates node assignments	-q
4.4/updatesectionidentifier.php (same file in 4.5/)	Fills section identifiers	-q
4.6/removetrashedimages.php	Removes images of trashed content	-n, -q
4.6/updateordernumber.php	Updates shop order numbers	-q
5.0/deduplicatecontentstategrouplanguage.php	Removes duplicate state group translations	-q
5.0/disablesuspicioususers.php	Lists, or disables, logins containing < or >	--disable
5.0/restorexmlrelations.php	Restores missing XML text relations	none
5.1/fiximagesoutsidevardir.php	Fixes image references outside VarDir	--dry-run
5.2/cleanupdfscache.php	Deletes cache rows from the DFS table	--iteration-sleep, --iteration-limit
5.3/recreateimagesreferences.php	Recreates ezimagefile rows	--dry-run
5.3/updatenodeassignmentparentremoteids.php	Fills eznode_assignment.parent_remote_id	-n, --iteration-sleep, --iteration-limit
5.4/cleanuntranslatablerelations.php	Removes stale relations of untranslatable relation attributes	--dry-run, --iteration-sleep, --iteration-limit
5.4/cleanupfieldvaluerelations.php	Removes relations to missing content from field values	-n, -v, --iteration-sleep, --iteration-limit
5.4/fixremovedezurlobjectlinks.php	Removes orphaned ezurl_object_link rows	--fix, --fetch-limit
5.4/fixtrashedimagereferences.php	Fixes image references of trashed content	--dry-run, --iteration-sleep, --iteration-limit
6.0/createaudittables.php	Creates the expaudit_* tables on any engine and indexes the audit files; safe to repeat	--dry-run, --no-index

Script	Purpose	Options
`6.0/movetrashrecords.php`	Copies who trashed what from `<VarDir>/trash/trashed.json` into `ezcontentobject_trash.trashed_by` and `trashed_via`; safe to repeat; exit 2 when the file is kept, 1 when the columns are missing or the file holds no JSON object	`--dry-run`, `--remove-file`, `-s <siteaccess>`

bin/php: schema, data and conversion

Script	Purpose	Options
`checkdbfiles.php`	Checks the update files against the upgrade path: every directory under `update/database/` (4.0 to 7.3 on MySQL and PostgreSQL, 6.0 on SQLite), with the cluster files and the differently named 6.0 files; `?` a file or directory it does not know, `!` a missing file. Without `--no-verify-branches` it also compares 4.3 to 5.4 with the old SVN branches, which needs `svn` and the retired svn.ez.no repository and so reports one `C` line. The exports go to a new directory it makes inside `--export-path` (default `var/tmp`) and removes again; nothing else there is touched	`--no-verify-branches`, `--export-path`
`ezsqldiff.php`	Compares two schemas (database or `.dba` file); exit status says whether they differ	`--source-type`, `--source-host`, `--source-user`, `--source-password`, `--source-socket`, `--match-type`, `--match-host`, `--match-user`, `--match-password`, `--match-socket`, `-t/--type`, `--host`, `-u/--user`, `-p/--password`, `--socket`, `--lint-check`, `--reverse`, `--check-only`
`ezsqldumpschema.php`	Dumps schema or data as `.dba`, serialized or SQL	`--type`, `--user`, `--host`, `--password`, `--port`, `--socket`, `--output-array`, `--output-serialized`, `--output-sql`, `--diff-friendly`, `--meta-data`, `--table-type`, `--table-charset`, `--compatible-sql`, `--no-sort`, `--format`, `--output-types`, `--allow-multi-insert`, `--schema-file`, database, file
`ezsqlinsertschema.php`	Creates a schema and data from a `.dba` file	`--type`, `--user`, `--host`, `--password`, `--port`, `--socket`, `--table-type`, `--table-charset`, `--insert-types`, `--allow-multi-insert`, `--schema-file`, `--clean-existing`, file, database

Script	Purpose	Options
`ezimportdbafile.php`	Imports a datatype's `.dba`	`--datatype`
`ezconvertdbcharset.php`	Converts every table to UTF-8	`--extra-xml-attributes`, `--extra-xml-data`, `--extra-serialized-data`, `--collation`, `--skip-class-translations`, `--iconv-character-set`, `--log-filename`
`ezconvertmysqltabletype.php`	Changes the MySQL storage engine	`--host`, `--user`, `--password`, `--database`, `--list`, `--newtype`, `--usecopy`
`convertezenumtoezselection.php`	Converts an `ezenum` attribute to `ezselection`	`--preview`, attribute id
`convertprice2multiprice.php`	Converts `ezprice` to `ezmultiprice`	none
`ezconvert2isbn13.php`	Converts ISBN-10 values to ISBN-13	`--class-id`, `--attribute-id`, `--all-classes`, `-f/--force`
`updateisbn13.php`	Updates the ISBN range tables	`--url`, `--db-*`
`ezsqldumpisbndata.php`	Dumps the ISBN range tables	`--stdout-sql`, `--stdout-dba`, `--filename-sql`, `--filename-dba`, `--db-*`
`updateniceurls.php`	Imports pre-4.0 URL aliases and rebuilds node aliases	`--import`, `--no-import`, `--import-nodes`, `--import-aliases`, `--import-redirections`, `--import-wildcards`, `--update-nodes`, `--no-update-nodes`, `--verify-data`, `--interactive`, `--backup-tables`, `--column-width`, `--fetch-limit`, `--db-*`, `--sql`
`verify_aliases.php`	Checks `ezurlalias_ml`	`--fix`, `--verbose`, `--sql`
`updatesearchindex.php`	Rebuilds the search index	`--clean`, `--db-*`, `--sql`
`adddefaultstates.php`	Gives every object the default object states (dry run unless `--apply`)	`--apply`
`cleanuppolicies.php`	Removes policies of modules that no longer exist (dry run unless `--apply`)	`--apply`, `-n`
`cleanupversions.php`	Removes archived versions beyond the limits	`-n`
`flatten.php`	Removes unused data: `contentobject`, `contentclass`, `workflow`, `role`, `all`	`--db-*`, `--sql`, name
`trashpurge.php`	Empties the trash	`--iteration-sleep`, `--iteration-limit`, `--memory-monitoring`, `--trashed-days`

Script	Purpose	Options
`ezsubtreecopy.php`	Copies a subtree	`--src-node-id`, `--dst-node-id`, `--all-versions`, `--keep-creator`, `--keep-time`
`ezsubtreeremove.php`	Removes subtrees	`--nodes-id`, `--ignore-trash`
`expcontentjob.php`	Large copy and remove jobs in batches	`--all`, `--json`, `--background`, `--trash`, `--delete`, `--all-versions`, `--keep-creator`, `--keep-time`
`ezcsvimport.php`, `ezcsvexport.php`	Import objects from CSV; export a subtree to CSV	`--class`, `--creator`, `--storage-dir`, node, file
`clusterize.php`	Moves files into or out of the cluster	`-u`, `--skip-binary-files`, `--skip-media-files`, `--skip-images`, `-r`, `-n`
`dfscleanup.php`	Compares the DFS table with the files and deletes what is missing	`-S`, `-B`, `-D`, `--path`, `--iteration-limit`
`ezflowupgrade.php`, `ezwebinupgrade.php`	Upgrade the ezflow and ezwebin packages	`--to-version`, `--repository`, `--package`, `--package-dir`, `--url`, `--auto-mode`

Extension scripts: `extension/eztags/bin/php/convertezkeyword.php` moves `ezkeyword` attributes to `eztags` (`--from-attr-id`, `--to-attr-id`, `--parent-tag-id`); `extension/enhancedselection2/bin/php/migrate_to_database.php` and `updateenhancedselection.php` update enhanced selection storage.

bin/php: checks after an upgrade

Script	Purpose	Options
`checkmanifest.php`	Compares the files with `share/filelist.md5`; no database needed	`--all`, `--staged`, `--fix`
`checkclasses.php`	Loads every declared class and reports failures	`--kernel`, `--tests`, `--quiet-ok`
`ezpgenerateautoloads.php`	Regenerates the autoload arrays	`-e`, `-k`, `-o`, `-s`, `-n`/`--dry-run`, `-t`/`--target`, `--exclude`, `-p`, `-v`, `-q`
`ezcache.php`	Clears caches	`--clear-all`, `--clear-tag`, `--clear-id`, `--list-tags`, `--list-ids`, `--purge`, `--expiry`, `--iteration-sleep`, `--iteration-max`
`eztemplatecheck.php`	Checks template syntax	files
`maintenance.php`	Turns maintenance mode on or off	`on`, `off`, `status`, `--message`, `--until`, `--allow-ip`, `--allow-admin`
`audit.php`	Shows and verifies the audit log	many; see the audit trail

Shell helpers in `bin/shell/`: `checkdbschema.sh`, `checkdbupdate.sh` (`--check-stable`, `--check-previous`), `verifyfiles.sh` (`--quiet`) and `generatefilelist.sh`. In the admin, Setup, Upgrade check runs the same file check against `share/filelist.md5` and compares the live schema with `share/db_schema.dba` merged with the active extensions' schemas (

`kernel/private/classes/views/setup/systemupgrade.php`).

Moving to or from the platform

This repository has no data converter to or from eZ Platform or Ibexa. The legacy bridge runs the Exponential kernel inside a platform installation on a shared database (branches for eZ Platform 2.5, Platform 3.3, Ibexa 4.6 and Platform 5), with the console commands `exponential:legacy:init`, `configure`, `install-extensions`, `symlink`, `assets-install` and `script` on the `3.x` line from `v3.0.0.30`, the `4.x` line from `v4.0.0.2` and every `5.x` tag, where the old `ezpublish:*` names remain as deprecated aliases. The 2.5 line (`master`, `v2.1.10` to `v2.1.12`) and the earlier 3.x and 4.x tags know only the old names (`ezpublish:legacy:init` ...). On the platform side, the vendor documents `ezxmltext:convert-to-richtext`, `ibexa:urls:regenerate-aliases`, `ezplatform:images:normalize-path` and the `ezpublish:update:legacy_storage_*` commands listed in 17.8.

17.10 Glossary of renamed concepts

Exponential (and eZ Publish 4.x and 5.x legacy)	eZ Platform and Ibexa	Notes
content object	content item (`Content`, `ContentInfo`)	
object version	version (`VersionInfo`)	
object attribute	field (`Field`)	
content class	content type (`ContentType`)	
class attribute	field definition (`FieldDefinition`)	
class group	content type group	
datatype	field type	
node	location (`Location`)	
main node	main location	
section, object state, role, policy, limitation	the same names	
user (an object of a user class)	user (`User`), still a content item	
siteaccess	SiteAccess	Matching rules become `siteaccess.match`; settings move from INI files to `ibexa.system.<scope>`
siteaccess settings in `settings/siteaccess/<name>/`	SiteAccess or SiteAccess group configuration	
extension	bundle	
module and view (`/content/view/full/2`)	route and controller	
template override (`override.ini`)	view configuration (`content_view` with matchers)	
view mode (`full`, `line`, `embed`)	view type	

Exponential (and eZ Publish 4.x and 5.x legacy)	eZ Platform and Ibexa	Notes
pagelayout (`pagelayout.tpl`)	page layout (`page_layout`)	
design and its fallback list	design and theme list (`ibexa_design_engine`)	
template operator or function	Twig function or filter	
`cache-block`, view cache	HTTP cache with tags, ESI fragments	
image alias	image variation	
`ezxmltext` (XML text)	RichText (DocBook)	
`ezflow` page	Page field type (commercial editions)	
eztags	Netgen Tags	Taxonomy in the commercial editions
clustering (`eZDFSFileHandler`)	IO handlers (`legacy_dfs_cluster` metadata, Flysystem binary data)	
workflow, trigger, operation	events and subscribers	(not verified here)
information collection	form builder (commercial) or the Netgen information collection bundle	The reference installation keeps `ezinfocollection` with the Netgen bundle
`ezfind` (Solr)	Solr search engine bundle	
`ezsite_data` `ezpublish-version`	the same row; the platform tracks schema changes with migrations	
eZ Publish 4.x	Exponential 6.0 continues this line	
eZ Publish 5.x (Platform stack beside legacy)	eZ Platform 1.x to 3.x, then Ibexa DXP 3.3 onward	Exponential keeps the legacy half and can run beside the platform through the legacy bridge

References

In this repository:

- 11. Upgrading: the order to run the update files and scripts in.
- Upgrading, Content model and editing, Templates and design, Operating a site.
- Legacy bridge, legacy bridge bundle specification, platform package map, platform site bundles, platform package forks and command renames.
- HTTP caching, file consistency check, audit trail, glossary.
- Code: `share/db_schema.dba`, `kernel/sql/`, `kernel/classes/datatypes/`, `kernel/classes/datatypes/ezuser/ezuser.php`, `kernel/classes/ezcontentobjecttreenode.php`, `kernel/classes/datatypes/ezimage/ezimagealiashandler.php`, `lib/eztemplate/classes/eztemplateautoload.php`,

`kernel/common/eztemplateautoload.php`, `settings/site.ini`, `settings/image.ini`, `settings/file.ini`, `settings/httpcache.ini`, `update/`, `bin/php/`.

External:

- Vendor migration pages: Migrate from eZ Publish, Migrate from eZ Publish Platform, Common migration issues, Migrating from eZ Publish (eZ Platform 2.5 version) (the 1.13 version of the page is no longer published; its old address redirects to the current page).
- Field types: field type reference, Image, User, Null, MapLocation, Matrix.
- Source: ezsystems/ezpublish-legacy, ezsystems/ezpublish-kernel (`User` hash constants in 7.5), ezsystems/ezplatform-xmltext-fieldtype, ezsystems/LegacyBridge, ibexa/core.
- PHP: password_hash, password_verify.
- Symfony and Twig: Configuration, Doctrine DBAL connection URLs, Mailer DSN, ESI, Twig for template designers, Twig `include`.

Contents · Previous: 16. Migrating from eZ Platform and Ibexa

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.