Choosing datatypes

For CMS developers modelling content, which datatype to choose for each kind of field. It covers the 35 kernel datatypes and the extension datatypes of the default distribution, the choices that are hard to undo, and…

Source file: doc/eng-us/help/cms-developers/choosing-datatypes.md Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/cms-developers/choosing-datatypes.md

This page is for developers who turn a list of fields into class attributes. A datatype decides storage, validation, the edit form, the view template, search and how imports read values, so the choice is worth a minute per field. Changing a datatype later means a new attribute and a migration script (Content classes and attributes).

Contents

1. Where the list comes from
2. The kernel datatypes by purpose
3. Datatypes from the default distribution
4. Decisions that are hard to undo
5. Datatypes to avoid in new models
6. When to write a datatype of your own

Where the list comes from

The datatypes offered in the class editor are `settings/content.ini [DataTypeSettings] AvailableDataTypes[]`: 35 in the kernel (`kernel/classes/datatypes/`). Active extensions add theirs in their own `content.ini.append.php`. To see what your installation offers:

Shell

```
./console exp:ini get content.ini/DataTypeSettings/AvailableDataTypes --allow-root-user
```

The kernel datatypes by purpose

The names in brackets are what the class editor shows.

Purpose	Datatype	Notes
A title, a name, a short line	`ezstring` (Text line)	maximum length per attribute; usually the source of the object name
Plain text of several lines	`eztext` (Text block)	no markup; good for teasers, addresses, meta descriptions
Formatted text: headings, lists, links, embeds	`ezxmltext` (XML block)	edited in the Online Editor; see Rich text

Purpose	Datatype	Notes
Whole numbers, decimals	`ezinteger` (Integer), `ezfloat` (Float)	minimum and maximum per attribute
Yes or no	`ezboolean` (Checkbox)	
Dates and times	`ezdate`, `ezdatetime`, `eztime`	`ezdatetime` for anything you sort or filter by time of day
One or more options from a fixed list	`ezselection` (Selection)	single or multiple; the options live in the class
Free keywords	`ezkeyword` (Keywords)	flat, comma separated; fetch by keyword with `fetch( 'content', 'keyword', ... )`
An image	`ezimage` (Image)	scaled by image aliases; alternative text; see Images and image aliases
A downloadable file	`ezbinaryfile` (File)	the file is kept in the storage directory, outside the database
Audio or video files	`ezmedia` (Media)	
A link to another site	`ezurl` (URL)	URL and text; the link list checks links (Links and URL aliases)
An e-mail address	`ezemail` (Email)	validated
References to other content	`ezobjectrelationlist` (Object relations), `ezobjectrelation` (Object relation)	see Relations and embeds
A small table of values	`ezmatrix` (Matrix)	columns defined in the class
Authors as names and e-mail addresses	`ezauthor` (Authors)	
A country	`ezcountry` (Country)	
An ISBN	`ezisbn` (ISBN)	
A sequential number	`ezidentifier` (Identifier)	prefix, suffix and digits per class
A user account	`ezuser` (User account)	only in user classes
Prices and product options	`ezprice`, `ezmultiprice`, `ezoption`, `ezmultioption`, `ezmultioption2`, `ezrangeoption`, `ezproductcategory`	for the built-in shop (The built-in shop)
Settings and packages from content	`ezinisetting`, `ezpackage`	special purposes in the setup tree
Notification subscription to a subtree	`ezsubtreesubscription`	

Each datatype's template and edit form are explained in Lesson 4.2: Datatypes and their templates; the value format an import uses (`fromString()`) is in Import and export tools, section 4.

Datatypes from the default distribution

Exponential 6 installs a set of extensions through its `composer.json`. Those that matter for content models:

Need	Datatype	Extension
Hierarchical tags, synonyms, a controlled vocabulary	`eztags`	`se7enxweb/eztags`
A map location	`ezgmaplocation`	`se7enxweb/ezgmaplocation`
A star rating by visitors	`ezsrrating`	`se7enxweb/ezstarrating`
Search engine metadata (title, keywords, description)	`xrowmetadata`	`se7enxweb/xrowmetadata`
A link with an internal or external target	`ngenhancedlink`	`se7enxweb/exp_enhanced_link`
Selections with more options than `ezselection`	`sckenhancedselection`	`se7enxweb/enhancedselection2`
A file with more options than `ezbinaryfile`	`enhancedezbinaryfile`	`se7enxweb/enhancedezbinaryfile`
Spam protection on forms	`recaptcha`, `hcaptcha`	`se7enxweb/recaptcha`, `se7enxweb/hcaptcha`

The same list, mapped from fields of other systems, is in Mapping content to Exponential. Check that an extension is active on your siteaccesses before you use its datatype (`./console exp:ini get site.ini/ExtensionSettings/ActiveExtensions --allow-root-user`).

Decisions that are hard to undo

- **Text line or rich text.** A heading that later needs a link or italics needs rich text. If in doubt for a body text, choose `ezxmltext`; for a title, `ezstring`.
- **Selection or tags or related objects.** A fixed short list that only you change: `ezselection`. A vocabulary editors extend: `eztags`. Categories that have pages of their own: related objects or locations.
- **Image attribute or image object.** An `ezimage` attribute belongs to one object. An image used in several places, or one that editors replace once for all pages, belongs in an image object in the media library, related or embedded.
- **Single or list relation.** Prefer `ezobjectrelationlist` even for one target: it limits the classes, has a default location for new targets and can hold more later.
- **Translatable or not.** A price, an e-mail address or a date often should not differ per language: untick *translatable* so translators do not have to copy it.

Datatypes to avoid in new models

Datatype	Why	Use instead
`ezenum`	an old selection datatype; the kernel ships `bin/php/convertezenumtoezselection.php` to move away from it	`ezselection`

Datatype	Why	Use instead
`ezobjectrelation`	one target only, fewer options	`ezobjectrelationlist`
Many small text fields for paragraphs	editors fight the form	one `ezxmltext` field

When to write a datatype of your own

Write one only when the value has its own storage, validation and editing that no existing datatype or combination of attributes covers: a value from an external service, a structured value with rules of its own. First try a combination of attributes, a child class, or a custom tag in rich text. When a datatype is the answer, the book has the full path: Datatypes and the tutorials a datatype by hand and with the RAD wizard.

Related pages

- Content classes and attributes
- Designing a content model
- The eZDataType API
- 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.