View modes

For CMS developers, how one piece of content is shown in many contexts. It covers the full, line, embed and other shipped view modes, how a mode is chosen, a template per class and mode, your own view modes for cards...

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

This page is for developers who need the same article to look like a page, a list item, a card and an embedded box. Exponential separates **what** is shown (the object) from **how** (the view mode): every context asks for a view mode, and each class can have its own template per mode. Templates that list content never need to know which classes they list.

Contents

1. The idea
2. The shipped view modes
3. How a mode is chosen
4. A template per class and mode
5. Your own view modes
6. View modes in embeds and layouts
7. Caching
8. Planning the modes of a site

The idea

```
article "Opening of the new hall"
  full   -> the article page                (node/view/full.tpl -> full/article.tpl)
  line   -> one row in the news list         (node/view/line.tpl -> line/article.tpl)
  card   -> a card on the front page         (node/view/card.tpl -> card/article.tpl)
  embed  -> a box inside another article      (content/view/embed.tpl -> embed/article.tpl)
```

A list template draws each child with `node_view_gui` and a mode name; the right template per class is found by override rules. Adding a class later needs templates for that class, not changes to every list.

The shipped view modes

Mode	Used for
`full`	the page of the content itself
`line`	an item in a list
`embed`	content embedded as a block in rich text
`embed-inline`	content embedded inside a line of text

Mode	Used for
`text_linked`, `text`, `tiny`, `plain`	short forms in lists, relations and the administration
`listitem`, `thumb`, `sitemap`, `print`, `pdf`	other shipped modes

Templates: View modes, Full view, Line view, Embed view.

How a mode is chosen

Context	How
A URL	`/content/view/<mode>/<node id>`; a URL alias opens `full`
A template	`{node_view_gui content_node=$child view='line'}`, `{content_view_gui content_object=$object view='embed'}`
Rich text	the editor picks embed or another mode allowed by `content.ini [embed] AvailableViewModes[]`
A Layouts block	the block's view type and the item view type chosen in the editor

A template per class and mode

`node/view/<mode>.tpl` is the default for every class. A rule gives one class its own template for a mode:

INI

```ini
[exp_mysite_line_event]
Source=node/view/line.tpl
MatchFile=line/exp_event.tpl
Subdir=templates
Match[class_identifier]=exp_event
```

Template

```
{* extension/exp_mysite/design/exp_mysite/override/templates/line/exp_event.tpl *}
<article class="exp-line exp-line--event">
    <h3><a href={$node.url_alias|ezurl}>{$node.name|wash()}</a></h3>
    <p>{$node.data_map.start_date.content.timestamp|l10n( 'shortdate' )}</p>
</article>
```

Rules and their order: Templates and overrides for CMS developers.

Your own view modes

A mode is just a name for which templates exist. To add `card`:

1. Create the default `design/exp_mysite/templates/node/view/card.tpl` for every class; a mode without a default template anywhere in the design chain fails.
2. Add rules and templates for the classes that need their own card.
3. Use it: `{node_view_gui content_node=$child view='card'}`.
4. Clear the template override cache (`php bin/php/ezcache.php --clear-id=template-override --allow-root-user`).

Avoid the names the kernel already uses. Prefix your own when they could clash with a theme you build on (`exp_card`). The designers' recipe: Custom view modes.

View modes in embeds and layouts

Editors choose among the embed view modes that `content.ini [embed] AvailableViewModes[]` lists; add your own there, in your extension's `content.ini.append.php`, with a template per class (For editors' embeds). Exponential Layouts draws the items of a block in a view type of the block definition; block templates can draw items with your view modes too (Add a view type, Block view templates).

Caching

`site.ini [ContentSettings] CachedViewModes=full;sitemap;pdf`: these modes, opened as pages, are stored in the view cache. Modes drawn inside another template (`line`, `card`, `embed`) are cached as part of the page that draws them, so a change to an article reaches a front page card when that page's cache is cleared (Caching for CMS developers).

Planning the modes of a site

Write a table per class: which modes it needs, with a sketch of each. A typical site needs `full`, `line`, one card or teaser mode and `embed` for the classes editors embed. Fewer modes, used everywhere, are easier to keep consistent than one mode per page (Planning the view modes of a design).

Related pages

- Fetching content in templates
- View modes for builders
- Node templates by class
- 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.