# 4.x feature comparison

This page represents a comparison between **Legacy Stack** (aka "*4.x*" stack) vs **Platform Stack** (previously referred to as "*Symfony*", "*new*" or "*6. x*" stack); it will be updated as new features are added. Note: Features mentioned on the black list are still available in 5.x, but via legacy kernel so you will need to use "pure legacy" or a mixed setup using "legacy fallbacks".

For a list of Symfony features supported out of the box, see here.

For additional related topics see Legacy code and features and Intro for eZ Publish 4.x/3.x developers.

## Known existing integration (white list):

| 5.x Feature | 4.x Feature | Note/Limitation |
| --- | --- | --- |
| **Content Model** | | 5.x aimed at not adding **any** features in the content/data model that would lead to backwards compatibility breaks, new features will instead be introduced when we reach "eZ Platform 1.0" and beyond. |
| Content* | (Content) Object | |
| (Content) Relation | (Object) Relation | |
| Field | (Object) Attribute | Missing FieldTypes will cause eZ Publish to return exception, this can be avoided by either implementing the FieldType or configuring it as "NullType" if you want to just quickly ignore it for now in your front end. |
| Field Type | Data Type | See FieldTypes reference for list of supported types, known unsupported FieldTypes are mentioned in Known issues |
| Locations | Node | |
| Section | | |
| Content Type | (Content) Class | |
| Field Definition | (Content) Class Attribute | |
| Field Groups | (Content) Class Attribute Grouping | |
| Object State | ObjectState | |
| Url Alias | | When following a virtual url alias ("nop:") in legacy, it will return an "Object not found" page. On Plaform it will redirect to the home page. |
| Url Wildcard | | |
| Language | | |
| User | | |
| User Group | | Can by design *(and intention)* only have one location in 5.x when using Public API UserService |
| Policy | | |
| Policy Limitation | | |
| Role | | |
| Role Limitation | | |
| Role Assignment | | Until eZ Platform 2015.09.01 there was no exposed unique id for Role Assignments making it hard to delete a specific assignment. |

|  |  |  |
|---|---|---|
| **Core** |  |  |
| Public API | n/a | Legacy/4.x does not have a official public API, almost all internal functions are exposes publicly |
| REST API |  | 5.1 and higher includes a working read/write REST v2 API, 4.x only had a readonly REST v1 API as of 4.5 |
| Development environment | Development mode | In 5.x this is provided by Symfony, and web debugger is extended by eZ Publish. |
|  |  |  |
| **Web stack** |  | 5.x web stack **is** Symfony with several eZ features on top |
| Controllers | Modules |  |
| Bundles | Extensions |  |
| (Twig) Template system | (eZ) Template system |  |
| View Provider | Override system | Restricted to *viewLocation* in 5.0, *viewContent added* in 5.1, for changing presentation logic. |
| Controller override | n/a | 5.2 added feature for being able to also use override system of View Provider to also be able to override content controller under your own conditions to be able to modify behavior. |
| Theme system | Design system | Basic template/ bundle inheritance supported in Symfony, full siteaccess aware design overrides like in legacy planned for future release. |
| HTTP Cache | View- / Static- Cache | For anonymous users with content aware caching since 5.0, also for logged in users with context aware caching since 5.2 |
| Site Access | Site access | Custom Url, host, port and combination matching to a specific set of settings |
| Site Access Group | n/a | New in 5.x, grouping of related SiteAccesses and common settings for them, typically used to denote a "Site" |
| CSRF | eZ FormToken | Provided by Symfony |
| Render + Query / API use | fetch functions | Combining the render functionality of Symfony with Override system and API you are able to create reusable views of your content similarly to how you would use fetch functions in the past, but getting cache and view / logic separation handled in a native way. |
|  |  |  |
| **Data** |  |  |
| Persistence cache | Cluster- / Class- / ClassAttribute- / var- / "Global"- cache | Introduced in 5.1, multi repository (aka multi-db) setups support added in 5.2. |
| IO | Cluster (-ing of files) | 5.0-5.3 uses legacy cluster system. New improved clustering system using Flysystem introduced in 5.4 using same paths as legacy for compatibility, path layout improved in eZ Platform 15.09 and higher. |
| Multi repository | Multi-db | Supported, however see note on "Persistence cache", as of 5.3 with own "repository" configuration. |
|  |  |  |

\* When referring a specific instance of any of the content model entries, either "object" or "item" is normal to append depending on context. In our BDD specifications the domain names have "object" appended to them forming "Content object(s)", "Location object(s)" and so on.


## Known missing in Platform stack (black/grey list):

Note: Missing integration doesn't mean the system can not use the feature in a new installation relying on the new stack, but if it needs, it shall rely on a fallback to the legacy stack or features available in Symfony bundles. For instance: a website with a shop system could be done in twig and rely on legacy only for the fallback for shop parts.

| Feature | Note/Limitation |
|---|---|
| **Core** | |
| Rating | User Rating not currently supported by API (the service to deal with ratings, rating field type already exists), planned to be implemented later when some convention for FieldType provided API's are established.<br><br>Solutions from community exists for the missing functionality: https://packagist.org/packages/makingwaves/ezstarratingbundle |
| Shop | No shop or field(/data) types related to shop currently supported by API, see Known issues. This is planned for "eZ Platform 1.x" or a release shortly thereafter, it will most likely use parts of Sylus for this functionality. |
| eZComment | ezcomments was replaced in 5.2 with a CommentsBundle that will provide an agnostic system to integrate 3rd party comment services. Out of the box supporting Disqus and Facebook integration but can via custom integration also work with other 3rd party systems (example: service, content model based comments or even ezcomments) |
| Content Editing | Content editing module is not implemented yet, but custom needs can for instance be solved with Public API+Symfony controllers or use of javascript REST client + REST API. |
| Module Permissions | Controller / Bundle Permissions is not fully integrated with eZ Publish Repository permission system yet, currently standard Symfony firewall rules need to be used. Public API however has full support for permissions and limitations, and is as of 5.2 known to support legacy most edge cases. |
| UrlAlias Routing | UrlAlias has the same model as before and hence does not support routing directly to a bundle:controller:action |
| Workflow | Planned for future version, see Shop. Simple workflows already possible by using Symfony evens and Public API "Signals". |
| Information Collector | Planned as a Form builder UI basically using Symfony Forms + swiftmailer for mail transport, might reuse content types similarly to how Information Collector did it. Also see Form tutorial [share.ez.no], for how to create forms with Symfony and eZ Publish 5.x. |
| Package system | Planned to return as a import/export system which uses either API or SPI directly to replicate some or full amount of content from one repository to another, preferred format: XML using XML reader/writer. |
| Menu management | Menu system in Platform will use KnpMenu bundle, a light version is scheduled for 5.3 while a more deep integration is planned for 5.4. |
| PDF Export | Not currently planned, could be re-implemented outside of Core |
| RSS feeds | Currently no API for managing RSS feeds, REST view can be used as basis of feeds. |
| Frontend Editing | See "Content Editing".<br><br>Be aware that eZ Flow timeline feature is not available in the legacy admin (only in front end editing), hence not usable on the new stack. |
| Preview | Comes with 5.3, versions before that uses legacy for preview. |
| Search | Search API exists, but front end is currently using legacy fallback (ezfind & ezsearch) and 5.x Search API uses SQL for now while Solr implementation will be added in 2014. |
| User module | User login, login handlers and SSO handlers uses Symfony authentication system as of 5.3, change-password, forgot-password, register, profile and editing currently uses legacy fallback. |
| oAuth 2.0 | oAuth 1.0 supported in REST v1 in legacy, planned for new stack and REST v2 for future 5.x version. |
| Fetch alias | Planned for future version as a way to specify a query with support for dynamic variables given to it. Currently also possible by means of OOP extending existing query functionality. |
| | |
| **Extensions** | (aka Bundles) |
| eZ Tags | Planned for future 5.x version, still community supported in legacy stack on 5.x. |
| eZ SI | eZ SI is purely legacy, not required in Symfony as it support ESI natively. ez SI is still supported in legacy stack on 5.x. |
| eZ Content staging | No equivalent yet on the new stack, still supported in legacy stack on 5.x. |
| eZ Style editor | No equivalent on the new stack, still supported in legacy stack on 5.x up until 5.1. |
| | |

| | |
|---|---|
| eZ Survey | No equivalent yet on the new stack, still supported in legacy stack on 5.x. |
| * | For other extensions still supported on legacy, see http://doc.ez.no/List-of-eZ-Publish-extensions-and-their-support-options |

For additional comparison between 4.x/legacy stack vs 5.x stack, see the following community provided spreadsheet provided "as is" with no guarantee on accuracy of content: https://docs.google.com/spreadsheet/ccc?key=0AgGKn8SQ_akMdFQ3VU0wMlR3c ENxTDZ2dkhUcVN2SWc#gid=0