

Structure 7.0 Release Notes



7th of June, 2021

Structure 7.0 introduces major enhancements to formulas and adds performance improvements

[Download the latest version of Structure and its Extensions](#)

Try It: [Structure Sandbox Server](#) (no installation required)

1. Version Highlights

Structure 7.0 introduces Expr 2 - a completely new level of sophistication and possibilities within formulas.

2. Changes in Detail

2.1. Formula Enhancements (Expr 2.0)

We've made major enhancements to Expr, Structure's formula language, allowing users to pull data from more places than ever before, including:

- issue links
- work logs
- related epic or stories
- parent task or sub-tasks (regardless of how the enclosing structure is built)
- versions – as a collection of values, and their properties
- sprints – as a collection of values, and their properties
- other multi-valued fields – as a collection of values
- properties of other fields – like users' full names

It's also now possible to embed queries (JQL and [Structured JQL](#)) within formulas.

There are multiple improvements in the Expr language, which support these new capabilities:

- Arrays - Formulas now supports arrays, or lists of values, such as Version fields with multiple values.
- Properties - Get the value of a particular property of an item using the following notation: `object.property`
- Chained Function Calls - Apply a sequence of functions to a value by using `value.function()` notation.
- User Functions - Define locally-used functions within a formula.
- Enhanced IF Expression - A convenient alternative to the `IF()` function, helping make formulas more readable.
- Concatenation Operator - Join text strings together.
- Text Snippets - Generate text using variables and expressions.
- New System Functions - Many new functions have been introduced to work with the new features.

Documentation: [Changes to Expr in Structure 7](#), [Formulas](#)

2.2. Additional Updates

- The new [Group by Attribute](#) generator replaces the old "Group by Text Attribute" generator. Now it supports formulas that return arrays and items.
- Extender performance improvements.
- Sprint field can be updated by Effectors.
- Fixed: Tree CustomField compatibility.

2.3. Important Considerations for Upgrade and Downgrade

2.3.1. Backward-incompatible changes in Expr

In making this new generation of the formula language, we had to make some backward-incompatible changes. Those changes are minor and they are very unlikely to break your formula, but it's possible.

If you have been using formulas before, please review the Backward Compatibility section of the [Changes to Expr in Structure 7](#).

2.3.2. Review S-JQL queries used in conjunction with formula-based Group by (Text) Attribute generators

After upgrade to Structure 7 some formulas may return array or item values instead of text. If such formula is used in a Group by Attribute generator, and the formula result is a user, a priority, a status or some other item, then instead of folders the generator will create group rows representing those items naturally – as users, priority values, status values. This may break some existing S-JQL queries if they use `"folder()"` constraints.

If you encounter this problem, the fix is to replace `"folder()"` constraint with the corresponding `"item()"` constraint.

2.3.3. Downgrading from Structure 7

If you use the new formula capabilities introduced with Expr 2, and then downgrade to Structure 6 or an earlier version, the new formulas will stop working. The result of the calculation would most likely be an error, but also might be an undefined value or an incorrect result in some rare cases.

For that reason, if you use the new formulas to configure generators, the downgrade may result in an incorrectly built structure. For example, if you use Group... menu to add grouping by Sprint Name, it would create a Group by Attribute generator, configured with the "sprint.name" formula, which will not work in the previous versions of Structure. To fix the problem, you will need to re-create the generator.

Should you decide to downgrade, please review the formulas added to your Jira while using Structure 7.

3. Supported Versions

Structure 7.0 and all extensions **support Jira versions 8.5 or later**. This release is for customers using Jira Server or Data Center (Jira Core, Jira Software, or Jira Service Management/Service Desk).

Compatible plugins versions:

- [Structure.Gantt](#) 2.7+
- [Structure.Pages](#) 1.6+
- [Structure.Testy](#) 2.5+

Cloud customers can learn more about our products on the "Cloud" tab of our marketplace listing.

4. Installation and Upgrade



Pick a Time

We strongly recommend that you install and upgrade your apps during off-peak hours or scheduled maintenance windows. There are known issues in the Jira plugin infrastructure that may cause performance degradation and impede app installation when your Jira instance is under heavy load.

4.1. Installing Structure

If your Jira server does not have Structure yet, the installation is simple:

1. Download and install Structure app, either from the Atlassian Marketplace or our [Download](#) page.
2. When Add-on Manager reports the successful installation, click Get Started to visit a page with important guidance for the Jira administrator. You may want to also check out the user's Get Started page, available under the "Structure" top-level menu.
3. Monitor `catalina.out` or `jira-application.log` for log messages from Structure.

4.2. Upgrading Structure

The upgrade procedure from versions 3.0–6.6 is simple:

1. Consider backing up Jira data. Use **Administration | System | Backup System**. (If you have a large instance and a proper backup strategy in place, you may skip this step.)
2. Back up Structure data. Use **Administration | Structure | Backup Structure** menu item. If you have a lot of structures and a large Jira, consider turning off the "Backup History" option to avoid a long backup process.
3. Install the new version of the plugin.
4. Monitor `catalina.out` or `jira-application.log` for warnings or errors.

5. Enterprise Deployment Notes

5.1. Extender Performance Improvements

We have improved the performance of Linked Issues, Stories under Epic and Sub-task extenders once again. This time the changes were made to the performance of incremental updates and to an invisible part of the Structure app's work - when Structure checks if changes should be applied to a structure or not. Actual numbers will depend on the structure configuration, amount of issues in Jira, amount of changed issue, database performance, etc. However, in our tests we saw a 15-20% speedup of incremental updates.

5.2. Event processing in Data Center

DATA CENTER

In the Data Center environment, Structure running on one node needs to let Structure running on other nodes know when an item (an issue or some other object) changes. This "change stream" is communicated to other nodes via the database, asynchronous caches and the occasional use of a global, one-per-cluster lock.

Normally, each change is written into the database immediately when it happens – in the "event listener". That code runs in the same execution thread as the change itself, typically as a response to a user's action. From time to time we received support requests where the Jira global locking subsystem failed and there were certain issues with the database. This made the writing to the change stream "hang", which, in turn, made user request threads hang, which led to Jira being unresponsive.

In Structure 5.1, an alternative implementation of this subsystem was introduced, which never blocks a user request thread. All global lock operations and writing to the database happen in a separate, dedicated thread of execution.

This feature was experimental and not enabled by default. We have fixed the only problem that was found since Structure 5.1 - *ConcurrentModificationException* that was happening in rare cases. In Structure 7.0 we made this feature **enabled by default**. There is still a dark feature that allows you to return to the synchronous event processing mechanism: set `structure.delegatingItemTracker.enableAsyncEventsQueue` to `false`. For more information, see [Advanced Configuration and Dark Features](#).

To test this change, you can make issue changes on one node while a user observes a structure with JQL-based automation on another node. In a few seconds, the user should see the most up-to-date information in Structure.

5.3. Fix for CDN based on Microsoft Azure

We recently worked on an incident where Jira was configured to use a CDN server based on Microsoft Azure. In such an environment, Structure breaks the loading of CSS resources on Structure Board, Issue pages, and some other pages. In Structure 7.0, this problem was fixed.

To test this fix, you can just open Structure Board and see that it looks as expected.

5.4. Expression Language Performance Aspects

Structure 7.0 introduces a new version of Expression Language that allows you to use Issue Links, Worklogs and other items and data types including arrays. It allows you to construct more complicated formulas with arrays, JQL filters and user functions. This can cause potential performance problems connected with the amount of data (array size) and formula execution steps count. Structure 7.0 applies bulk data load for links, worklogs and JQL. It also has limits for array size, call count and call depth in formulas. But it is still possible to create a formula complicated enough to be slow. So please keep this in mind.

5.5. Testing on Staging Environment

Apart from the changes and suggestions above, there are no particular special areas of interest for load testing and stress testing Structure 7.0. We advise running the same testing procedures as you've done for previous upgrades.



Need help or have questions? Contact [Tempo Support](#).