FirstSpirit™

Unlock Your Content

Release Notes+

FirstSpirit™ 2020-09
(from January 2020)

Status	**RELEASED**
Department	FS-Core
Copyright	2020 e-Spirit AG
File name	Releasenotes_2020_09_EN_yearly

e-Spirit

Table of contents

8   FirstSpirit Content Experience Tools (CXT)..70

1   Administration

1.1   Browser compatibilities 2020-05

FirstSpirit is an enterprise content management system that was designed for use in complex IT landscapes and which supports numerous operating systems, Java Runtime Environments, and databases. To guarantee high-quality software and ensure that FirstSpirit can run in all supported configurations, regular quality assurance tests are carried out. However, the large number of third-party components that are supported, the short release cycles, and the combination and configuration options for third-party components mean that testing is extremely complex.

As a result, FirstSpirit generally supports the latest version of a third-party component. These are regularly tested by e-Spirit AG and measures are taken to rectify any errors or incompatibilities. When looking at and selecting third-party components to be supported for operation with FirstSpirit, e-Spirit also gives special consideration to components that cover the largest possible market shares and that are used successfully by a large number of customers.

This includes the browsers that are used to access the FirstSpirit web apps: Because **Microsoft Internet Explorer** has been gradually replaced by Microsoft Edge, First Spirit will no longer support Microsoft Internet Explorer in future. Instead, Google Chrome and Microsoft Edge Chromium are recommended. The development of the new ContentCreator was already strongly focused on these two browsers: The new ContentCreator is therefore no longer compatible with Microsoft Internet Explorer for technical reasons.

Cessation of support for Microsoft Internet Explorer for all FirstSpirit web apps is planned for FirstSpirit 2020-07. We also plan to retire the "old" ContentCreator with this release.

The point of reference is always the latest version of the **Chrome** browser available on the operating system in question. Regular tests are performed on this browser and steps are taken to eliminate errors based on error messages.
Browsers that are similar to Chrome (such as Apple Safari) are also supported and tested, but functionality may be restricted (e.g., limited SVG support).

An overview of planned browser compatibilities from FirstSpirit 2020-07:

Recommended browsers/reference:

- Google Chrome

- Microsoft Edge

Compatible browsers:
- Mozilla Firefox

- Apple Safari (subject to restrictions)

Not compatible:

- Microsoft Internet Explorer

- Opera and others

The "FirstSpirit Technical Data Sheet" has been adapted accordingly (https://docs.e-spirit.com/odfs/edocs/admi/technical-requi/index.html).

1.2   The FirstSpirit session cookie now supports the attribute SameSite

2020-05

FirstSpirit uses several standard web applications (fs5root, fs5webedit, fs5webmon, fs5preview, fs5staging) as well as possibly additional, project-local web applications (fs5webedit_PROJECTID and fs5preview_PROJECTID). For authentication of the web applications (WebAuthentication) on the FirstSpirit server, a technology is used which uses randomly generated **session cookies**. Through the use of session cookies, the user's login data only has to be transferred once from the web browser to the FirstSpirit server. After the successful login, the web browser uses exclusively the unique session cookie, which is valid for a limited period of time, and which is then sent from the web browser to the server with every further server enquiry, instead of the login data, in order to authenticate the user there.

The session cookie is an integral part of the Servlet API and is used for many other use cases besides authentication.

Many browsers (including Chrome 80) have massively restricted third-party cookies since the beginning of the year. All cookies that do not have the attribute `SameSite=None` are automatically restricted to first-level domains. **This behavior leads to problems when FirstSpirit web applications are integrated into other web applications (e.g. as IFrame).** In this case the new standard browser behavior ensures

that the FirstSpirit session cookies are blocked by the browser and that the users in the embedded FirstSpirit web application can no longer be authenticated via the session cookie, for example.

For this use case with the current FirstSpirit release the session cookie can be extended by **the attribute `SameSite=None`**. `SameSite` is a standard that is intended to prevent cookies from being automatically sent by the browser with so-called cross-site requests and thus offers protection against cross-site request forgery (CSRF). In addition to this security aspect, the `SameSite` attribute allows you to define which cookies can be read in which context.

> **!** The `SameSite` attribute requires a "Secure" flag. Cookies with `SameSite=None` without a "Secure" flag are rejected by the browser. The "Secure" flag defines that a cookie is always sent over a secure HTTPS connection.

The value for the FirstSpirit Session Cookies can be set via the configuration file `fs-server.conf`, both globally via the parameter `servletSessionCookieSameSite` and individually for specific WebApp paths, e.g. `servletSessionCookieSameSite.fs5webmon=None` for FirstSpirit ServerMonitoring. A WebApp-specific configuration overwrites the global setting `servletSessionCookieSameSite` for this WebApp path.

In the default configuration (of the fs-server.conf config file) no global value is set for `servletSessionCookieSameSite`.

```
# Servlet engine session cookie SameSite attribute. If left empty, the SameSite
# attribute for the session cookie is not
# set and the servlet engine defaults apply.
# Supported values: None, Strict, Lax
servletSessionCookieSameSite=
# Servlet engine session cookie SameSite attribute for a specific webapp context path.
# 'ROOT' is the reserved name for the root webapp context path.
# servletSessionCookieSameSite.ROOT=None
# servletSessionCookieSameSite.fs5webmon=None
# servletSessionCookieSameSite.webappContextPath=None
```

Possible values for the `SameSite` attibute:

- Strict:
  - The session cookie is only sent in the first-party context (i.e. only if the page for the cookie matches the URL in the browser) and

  - not together with cross-site requests initiated by third-party websites.

- Lax:
  - The session cookie is only sent in the first-party context (i.e. only if the page for the cookie matches the URL in the browser) and

  - only together with cross-site requests that are considered "safe". This applies to secure HTTP methods (GET, HEAD, OPTIONS, and TRACE) and top-level navigation (actions that cause the URL in the browser address bar to change, such as links). `SameSite=Lax` is the default setting in modern browsers.

- None:
  - The session cookie is sent in all contexts (i.e. also in the third-party context), i.e. sending is allowed across origins.

  - In this setting, the attribute does not provide any additional protection against CSRF. However, this setting can be useful if a FirstSpirit web application is to be integrated into another web application.

- Value not set (FirstSpirit default):
  - If the value is not set, the default settings of the Servlet Engine are used.

  - If no value is configured here for the `SameSite` attribute, the browser default setting is used. Modern browsers interpret an unset SameSite attribute as `SameSite=Lax`.

> **!** e-Spirit recommends retaining the default settings of the `fs-server.conf` configuration file. In most cases the default setting covers both security aspects (good protection against cross-site request forgery) and user concerns (good user experience).
> Only in exceptional cases (e.g. FirstSpirit as an embedded web app) a change is necessary.

1.3 Change the order of the project languages by drag-and-drop 2020-02

All languages used in a project are listed via the project properties in the FirstSpirit Server Manager ("FirstSpirit ServerManager / Project / Properties / Languages"). The defined project languages are displayed in the defined order in SiteArchitect. In addition to using the context menu, the order of the project languages can now also be changed by using the mouse pointer (drag-and-drop).

The following applies: The master language always remains at the top of the sequence.

1.4 The support for the Windows installation program and the tar archive (Unix) is discontinued 2020-06

With the end of Java 8 support, the support for the Windows installation program (.exe) (as announced with 2018-06) and at the same time the support for the previous tar archive (.tgz) for all Unix systems will also be discontinued.

The previous installation options will be replaced by new installation options for the FirstSpirit server and update options for the FirstSpirit backend, which were released with FirstSpirit 2019-06:

- `fs-install-[version].tar.gz` to install the FirstSpirit server and

- `fs-update-[version].tar.gz` to update the FirstSpirit backend.

The new archives can be used flexibly for different requirements. They are compatible with all supported operating systems and can be used for servers in isolated and legacy mode.

Another benefit of the new installation and update options is the automatic update of the (Java) version-specific start parameters for the server start. For all new installations based on the installation archive `fs-install-[version].tar.gz`, a configuration file `fs-wrapper-vendor.conf` is created, which contains the recommended manufacturer start options for starting the FirstSpirit server. This file is automatically updated when the FirstSpirit backend is updated via the FirstSpirit update archive `fs-update-[version].tar.gz` - so it always contains the current start options for the respective Java version.

Further documentation:

- Installing FirstSpirit

- Updating FirstSpirit

- Java VM and Wrapper (fs-wrapper*.conf)

- FirstSpirit ServerMonitoring / FirstSpirit / Configuration / Startup options

1.5   End of support: Java Web Start 2020-06

Up to now, the FirstSpirit desktop apps (FirstSpirit SiteArchitect and ServerManager) could be started and updated (roll-out) via Java Web Start or via the FirstSpirit Launcher.

The Java Web Start functionality is included in Oracle Java 8, but is not supported from Oracle Java 11 on. OpenJDK does not offer a comparable technology. Therefore, since FirstSpirit does no longer support Oracle Java 8, **the option of starting and rolling out the FirstSpirit desktop applications via Java Web Start will no longer be supported** (as announced with 2019-10). The FirstSpirit JNLP files, which were required for Java Web Start, will also be dropped.

In most cases it is possible to switch to the new FirstSpirit Launcher without having to modify the configuration. The FirstSpirit Launcher supports a fully automatic update (of both the Launcher itself and the JRE). When using the launcher, the launcher version available on the FirstSpirit server is determined and an update is started if the version on the server is higher. What's more, the new Launcher offers a tray application that gives the user a convenient overview of the FirstSpirit Java applications that are currently active and that have been used recently.

Further documentation:

- Installation FirstSpirit Launcher

1.6   FirstSpirit is currently not compatible with Tomcat 10 2020-03

Apache Tomcat can be used as an external web server with FirstSpirit, and can be operated either on the same host as the FirstSpirit server or on its own dedicated host (standalone / remote).

FirstSpirit is currently compatible with the following Tomcat versions:

- Apache Tomcat 9.0.31 or higher

- Apache Tomcat 8.5.51 or higher

With the release of the milestone 1 version of Tomcat 10 on 20.02.2020, there was a change from Java EE to Jakarta EE (see http://tomcat.apache.org/).

This changed the namespace (package) for all implemented APIs from `javax.*` to `jakarta.*`. This change requires extensive adjustments to the FirstSpirit web applications (FirstSpirit Start Page, ContentCreator, etc.) to enable migration from Tomcat 9.0 and lower to Tomcat 10 and higher.

At the current time FirstSpirit is incompatible with Tomcat 10 and the use of a Tomcat version higher than 9.0 is also not recommended.

1.7 Starting FirstSpirit using Linux has been further improved with the new "fs-server start script". 2020-07

To further secure operation of the FirstSpirit Server using Linux and to simplify the central configuration, a further, optional configuration file has been introduced.

This new `fs-server-custom.shconf` file can be used

- to define the user with which the FirstSpirit Server is to be started.

- to define the path to the Java version.

The file is not updated automatically when the FirstSpirit backend is updated and is therefore not included in `fs-update*.tar.gz`. For servers that have already been installed, the file can be extracted from `fs-install*.tar.gz`.

In the file, the corresponding lines must then be uncommented (# at the start of the line removed) and the required values entered

```
RUN_AS_USER=<Username>
export FS_JAVA_HOME=<path to java>
```

1.8 Generation directories of deleted orders are cleaned up. 2020-04

If a schedule with a generation action is deleted and there is a corresponding directory for this schedule on the application server on which the Staging Web application is running, the corresponding directory on the application server is also deleted.

1.9 Automatically update Java version-specific wrapper files 2020-04

The FirstSpirit Java VM is started and stopped via the Tanuki Java Service Wrapper.

Previous behavior:

The start parameters of the Java VM were defined with the help of a corresponding configuration file for the wrapper (fs-wrapper[.isolated].conf).

When changing the Java version, the start parameters were adjusted. The changed configuration was directly available for new FirstSpirit installations. For existing installations, the changed configuration of the start parameters had to be adapted manually.

New behavior:

This behavior is improved with the current FirstSpirit release. For this purpose, the startup options have been split into two separate configuration files:

- New: `fs-wrapper-vendor.conf` contains the recommended vendor startup options

- Changed: `fs-wrapper[.isolated].conf` now only contains the startup options that should be adjusted server specific
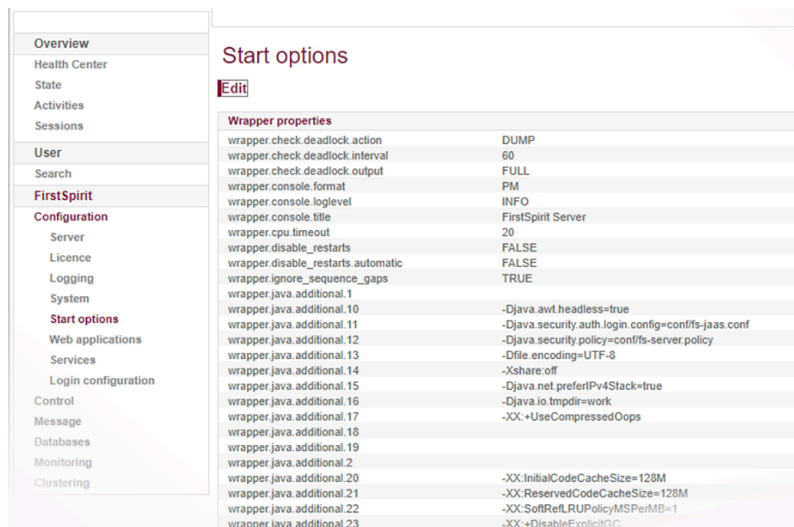
Objective: The new configuration file `fs-wrapper-vendor.conf` is part of the FirstSpirit update archive `fs-update-[version].tar.gz`. When updating the FirstSpirit server via the update archive `fs-update-[version].tar.gz` the startup options are automatically updated. Thus, the configuration adjustments are also available for existing FirstSpirit installations without any manual adjustments.

> ! The configuration file `fs-wrapper-vendor.conf` should not be changed. Should it be necessary in individual cases to adjust the recommended vendor startup options, the corresponding startup parameters should be redefined in the configuration file `fs-wrapper[.isolated].conf` (and not overwritten in `fs-wrapper-vendor.conf`).

Display via FirstSpirit ServerMonitoring

The start options from both configuration files are displayed via FirstSpirit ServerMonitoring (at "FirstSpirit" / "Configuration" / "Startup options"). Here the values from both files are combined in one view:

Display of the startup options (ServerMonitoring)

Editing via FirstSpirit ServerMonitoring

By clicking on "Edit", the start options can be edited via the FirstSpirit ServerMonitoring (at "FirstSpirit" / "Configuration" / "Startup options"). Only the content of the configuration file `fs-wrapper[.isolated].conf` is offered for editing.

The start parameters of the configuration file `fs-wrapper-vendor.conf` are displayed read-only at the bottom, with the note that the values for overwriting should be redefined ("manufacturer properties: Redefine property to overwrite the recommended value").

Clicking "Save" saves the changes to the configuration file. If invalid configuration settings are made, the error is indicated in the ServerMonitoring when saving the configuration and the save operation will not be completed.

By transferring and changing the values in the configuration file `fs-wrapper[.isolated].conf` all parameters of the configuration file `fs-wrapper-vendor.conf` can be overwritten (not only the general Java parameters `wrapper.java.additional.#`)

Editing of the startup options (ServerMonitoring)

Migration guide

To convert to the new behavior, the following steps should be carried out for all existing FirstSpirit installations:

1. Download the update archive `fs-update-[version].tar.gz`. The changes are available for update archives **from version 1.0.5** on. The update archive is available for download (if you need login data, please contact the Technical Support).

2. Stop the FirstSpirit server.

3. First, decompress the update archive `fs-update-[version].tar` (e.g., via g(un)zip) and then unpack it to the desired FirstSpirit installation directory. Example call under Linux:
   `~/firstspirit5$ tar xvfz fs-update-[version].tar.gz`

4. Compare the configuration files

   `<FirstSpiritROOT>/conf/fs-wrapper[.isolated].conf`

   and

   `<FirstSpiritROOT>/conf/conf-all/fs-wrapper[.isolated].conf.default.txt`

   Then, adjust the parameters with different values in the following file:

   `<FirstSpiritROOT>/conf/conf-all/fs-wrapper[.isolated].conf.default.txt`

   Especially, the following parameters have to be taken care of:

   `wrapper.java.command`

   `wrapper.java.maxmemory`

   `wrapper.java.initmemory` (set to the value of `wrapper.java.maxmemory`)

   `wrapper.ntservice.name` (only at Windows))

   `wrapper.umask`

   If known, apply self-defined parameters to `<FirstSpiritROOT>/conf/conf-all/fs-wrapper[.isolated].conf.default.txt`

   (ideally, no adjustments are required).

5. After the adjustment, rename the file

   `<FirstSpiritROOT>/conf/fs-wrapper[.isolated].conf`

   to

   `<FirstSpiritROOT>/conf/fs-wrapper[.isolated].old`.

6. Then rename the file

   `<FirstSpiritROOT>/conf/conf-all/fs-wrapper[.isolated].conf.default.txt`

   to

   `<FirstSpiritROOT>/conf/fs-wrapper[.isolated].conf`

7. Start the FirstSpirit server.

8. Optional: If no problems occurred after several days of operation, the file `<FirstSpiritROOT>/conf/fs-wrapper[.isolated].old` can be deleted.

Further documentation:

- Java-VM and -Wrapper (fs-wrapper*.conf)

- Java Wrapper Parameters

- FirstSpirit ServerMonitoring / FirstSpirit / Configuration / Startup Options

- Updating FirstSpirit

1.10   Metaspace: Note on the configuration of the Java VM 2020-02

The metaspace (another native memory area next to the heap) is automatically managed by the Java VM. A different configuration of this memory area, for example using the parameters `MetaspaceSize` or `MaxMetaspaceSize,` is **no longer recommended**.

These parameters can be configured in the configuration file `fs-wrapper*.conf` and were previously set automatically, provided that the installation or update of the FirstSpirit server was carried out <u>via an old installation path</u>, e.g. the Windows installation file (.exe): for Microsoft Windows Server (discontinued with 2018-06).

Please check if your configuration conforms to this recommendation. If necessary, you can manually remove the `MetaspaceSize` or `MaxMetaspaceSize` parameters from the `fs-wrapper*.conf.`

If you install or upgrade using the new Tar Gzip archives (released with 2019-06), <u>no adjustment is required</u>:

- `fs-install-[version].tar.gz` for installing the FirstSpirit server and

- `fs-update-[version].tar.gz` for updating the FirstSpirit server

Further documentation:

- for the configuration of the Java VM and Wrapper (fs-wrapper*.conf)

- to common Java parameters for the Java VM

- to external application server requirements

1.11 New default value: Attribute SameSite=Lax for the FirstSpirit Session Cookie   2020-09

FirstSpirit uses several standard web applications (fs5root, fs5webedit, fs5webmon, fs5preview, fs5staging) as well as possibly additional, project-local web applications (fs5webedit_PROJECTID and fs5preview_PROJECTID). For (user) authentication FirstSpirit, like most other web applications, uses randomly generated **session cookies** (see information on the use of cookies).

Many browsers (including Chrome 80) have massively restricted third-party cookies since the beginning of the year. All cookies that do not have the attribute `SameSite` are causing warning messages. `SameSite` is a standard that is intended to prevent cookies from being automatically sent by the browser with so-called cross-site requests and thus offers protection against cross-site request forgery (CSRF). In addition to this security aspect, the `SameSite` attribute allows you to define which cookies can be read in which context.

The FirstSpirit Session Cookie supports the `SameSite` attribute from **FirstSpirit 2020-05** on with the option of configuration (see configuration servletSessionCookieSameSite). The previously recommended default setting (SameSite attribute is not set) can now cause the following log output in the JavaScript console in some browsers:

```
Cookie "FS52ID" will be soon treated as cross-site cookie (...)
```

The default setting for the SameSite attribute is changed with FirstSpirit 2020-09 to `SameSite=Lax`.

With this setting the session cookie is only sent in the first-party context (i.e. only if the page for the cookie matches the URL in the browser) and only together with cross-site requests that are considered "safe". This applies to secure HTTP methods (GET, HEAD, OPTIONS, and TRACE) and top-level navigation (actions that cause the URL in the browser address bar to change, such as links).

e-Spirit recommends retaining the default settings of the `fs-server.conf` configuration file. In most cases the default setting covers both security aspects (good protection against cross-site request forgery) and user concerns (good user experience).
Only in exceptional cases (e.g. FirstSpirit as an embedded web app) a change is necessary.

The value for the FirstSpirit Session Cookies can be set via the configuration file `fs-server.conf`, both globally via the parameter `servletSessionCookieSameSite` and individually for specific WebApp paths, e.g. `servletSessionCookieSameSite.fs5webmon=None` for FirstSpirit ServerMonitoring. A WebApp-specific configuration overwrites the global setting `servletSessionCookieSameSite` for this WebApp path.

1.12 Problems starting the server after updating from FirstSpirit 2020-08

2020-09

When switching from FirstSpirit 2020-08 to another FirstSpirit version, the FirstSpirit server may not be accessible. The web applications log an error message of the type

```
de.espirit.firstspirit.common.ConnectError: Wrong client version
```

In this case the web applications should be deleted manually. The following directories can be affected:

- `~/tomcat/webapps`

- `~/firstspirit/web`

- `~/firstspirit/data/modules/FirstSpirit  Jetty  Server.JettyService/webapps`

The FirstSpirit server can then be restarted.

If you require support, please contact e-Spirit Technical Support (https://help.e-spirit.com).

1.13 ServerManager: Deleting template sets 2020-03

Template sets (e.g., "Intranet", "RSS feed") are defined for the project from the presentation channels (e.g., "json", "html"). Template sets for a project are created and edited via "FirstSpirit ServerManager / Project properties / Template sets".

Template sets can now be removed from the project via "FirstSpirit ServerManager / Project properties / Template sets / Delete" (previously it was only possible to deactivate them). To prevent accidental deletion, an additional dialog box is shown before deletion ("Do you really want to delete this template set?").

> **!** The deletion of a template set cannot be undone!

If a template set is accidentally removed from a project, please contact our Technical Support team immediately to restore the template set.

For more information on template sets, see the Documentation for administrators.

2   Administration / Modules

2.1   Improved stability of the system when running module updates during live operation   2020-08

Further improvements have been made for better system stability when running module updates during live operation. In most cases, corresponding updates can be completed without having to restart the FirstSpirit Server.

Note: The increased stability of the system has been achieved through available module resources that are in use outside the module remaining online.
For example, if the classes of a module that is being updated are being used by another module, this service should be restarted subsequently.

Note: A server restart must be performed to ensure that resources from old module versions are no longer being used. A server restart is still recommended after a module update.

3   General

3.1   Code signing certificate update   2020-08

In the current FirstSpirit release, the content of FirstSpirit jar files is signed with a new certificate. An update of the FirstSpirit Server is not necessary.

3.2   Announcement: FirstSpirit Experience Accelerator   2020-02

e-Spirit has so far provided demo projects with sample basic content and configurations. These will be replaced in the future by the contemporary **"FirstSpirit Experience Accelerator"**. The FirstSpirit Experience Accelerator is a new, important building block in the FirstSpirit ecosystem. It contains reference implementations for important content modules (including Shoppable Image and Video, Touchpoint Editing, Teaser Carousel, and Banner) and can be used as a quick entry into a new FirstSpirit project. The objective of the FirstSpirit Experience Accelerator is a faster product launch in the enterprise and eCommerce environment.

The "Mithras Energy" demo project (for ContentCreator and SiteArchitect) will be dropped in the future and will no longer be distributed with FirstSpirit.

3.3   Browser compatibilities (Micorsoft Edge)   2020-01

In general FirstSpirit always works with the latest version of a third-party component. They are regularly tested by e-Spirit AG and measures are taken to eliminate any errors and incompatibilities. This also applies to browsers for using FirstSpirit ServerMonitoring, ContentCreator and FragmentCreator.

The recommended browsers for **FirstSpirit FragmentCreator** are currently:

- Google Chrome

- Microsoft Edge

Microsoft Edge was recently moved to a new technological basis (Chromium) by the manufacturer. This new version will be automatically delivered to customers as planned with a Windows update from mid-January 2020.
Also in this case FirstSpirit supports the latest version of Microsoft Edge. In particular, MicroApps (MicroApp Framework) do not work with "old" versions of Microsoft Edge, so the new (chromium-based) version of Microsoft Edge should be used for working with FirstSpirit.

3.4   End of Java 8 support   2020-06

With the current FirstSpirit release, Oracle Java 8 and OpenJDK 8 are no longer supported by FirstSpirit.

Background: FirstSpirit is operated successfully by customers and partners with a wide range of different third-party components (operating systems, JDKs, servlet engines, application servers, and databases) and external libraries. Some of these components / libraries are no longer compatible with Java 8 in their current version. This applies, for example, to the internal Apache Derby database supplied with FirstSpirit. As a consequence, updates can no longer be carried out for those third-party components / libraries, which is particularly critical for security-relevant updates.

Oracle has not provided any public updates for Java 8 since February 2019 and recommends that developers and end users transition to a higher Java version in order to continue receiving public updates and security enhancements.

Also, with the end of Java 8 support, as announced before:

- The support for the **Windows installer** and the legacy **tar archive (.tgz) for all Unix systems** is dropped. We advise our customers to use the new installation method via the install archive (see Installing FirstSpirit).

- The option of starting the FirstSpirit desktop applications via **Java Web Start is no longer available**. (The Java Web Start functionality is included in Oracle Java 8, but is not support from Oracle Java 11 on. OpenJDK does not offer a comparable technology.) Therefore, SiteArchitect and ServerManager can now be started solely via the FirstSpirit Launcher (see FirstSpirit Launcher documentation). The FirstSpirit JNLP files, which were required for Java Web Start, will also be dropped.

The ramp-down phase for Java 8:

- started as planned in the second quarter of 2020. First, the Java 8 compatibility tests were gradually reduced.

- **With the current FirstSpirit version**, Java 8 functionality has been removed from FirstSpirit:
  - Wrapper configuration: All definitions for Java 8 have been removed from the standard configuration files for the Java Service Wrapper (`fs-wrapper[.isolated].conf`, `fs-wrapper-vendor.conf`) provided by e-spirit.
    The standard configuration files are provided via the FirstSpirit installation archive or the FirstSpirit update archive and can be found in the folder `<FirstSpiritROOT>/`

`conf/conf-all/` after a new installation or after updating the FirstSpirit backend. The adjustments are included from wrapper version 1.0.8 onwards.
A Java version 11 or higher is now required to start the FirstSpirit server.

▪ Launcher JRE module: Oracle Java 8 and AdoptOpenJDK 8 are no longer included in the FirstSpirit Launcher JRE module. The FirstSpirit Launcher JRE module currently offers the versions AdoptOpenJDK 11.0.7+10 (default) and AdoptOpenJDK 14+36.

▪ In 2020Q4 the byte code level will be switched from Java 8 to Java 11. **From this point on, all Java versions prior to Java 11 (LTS) will be incompatible with FirstSpirit.**

This means that FirstSpirit will no longer be executable with Oracle Java 8 or OpenJDK 8 and that e-Spirit AG will no longer implement any measures to eliminate errors or problems associated with Java 8. We recommend that you switch to a higher Java version soon.

! On the server side, a change from Java 8 to Java 11 (and higher) should be tested extensively on a test system, especially when using third-party components and individual customer extensions. Java 8 and Java 11 in parts differ considerably, which may result in existing libraries and implementations being incompatible with Java 11 (and higher). In such cases, updating the respective libraries, if possible, may help. When changing from Java 8 to Java 11 (and higher) we recommend changing the FirstSpirit server to Isolated Mode at the same time, in order to avoid having to test modules and implementations twice. For this topic, see Switching to Isolated Mode.

3.5   Cessation of support for Microsoft Internet Explorer 2020-07

With the latest FirstSpirit release, we are planning to cease support for Microsoft Internet Explorer as a browser for using the FirstSpirit web apps. The manufacturer is also gradually replacing Microsoft Internet Explorer with Microsoft Edge.

The point of reference for using FirstSpirit is always the latest version of the **Chrome** browser available on the operating system in question. Regular tests are performed on this browser and steps are taken to eliminate errors based on error messages.
Browsers that are similar to Chrome (such as Apple Safari) are also supported and tested, but functionality may be restricted (e.g., limited SVG support).

Overview of current browser compatibilities:

Recommended browsers / reference:

- Google Chrome

- Microsoft Edge

Compatible browsers:
- Mozilla Firefox

- Apple Safari (subject to restrictions)

Not compatible:

- Microsoft Internet Explorer

- Opera and others

The "FirstSpirit Technical Data Sheet" has been adapted accordingly (https://docs.e-spirit.com/odfs/ edocs/admi/technical-requi/index.html).

3.6   FirstSpirit documentation: PDF format discontinued   2020-04

Documentation for all FirstSpirit products is now provided in the HTML format.

Until recently, we have provided documentation in the PDF format as well, but as of the current release, we will discontinue those efforts.

More specifically:

- The navigation entry "Documentation" has been removed from the FirstSpirit online documentation interface.

- The DVD index, that depends on the beforementioned navigation structure of the online documentation, has now been detached and will keep the state it had in FirstSpirit Release 2020-03.

- The conversion of the FirstSpirit release notes to the HTML format is currently ongoing (for the duration, release notes will be provided in the PDF format).

- Currently, the documentation for some of our FirstSpirit modules is only available in the PDF format, and may still be obtained from https://docs.e-spirit.com/.

3.7   Java: Current status and outlook   2020-04

For operating FirstSpirit, e-Spirit AG supports:

- OpenJDK: the current Java version as well as the latest LTS (Long-Term-Support) variant.

- Oracle Java: only the latest LTS (Long-Term-Support) variant.

This applies to both the FirstSpirit Server and the FirstSpirit desktop applications.

Deviating from this rule, e-Spirit AG continues to support Oracle Java and OpenJDK Version 8 for the time being, and will discontinue support for Java 8 in the future (see below for Outlook Java 8 support).

As of FirstSpirit 2020-04, **OpenJDK 14** (non-LTS; release date 2020/03) is officially approved for use with FirstSpirit (i.e, the FirstSpirit Server and the FirstSpirit desktop applications).

AdoptOpenJDK 14 is included in the "FirstSpirit Launcher JRE" module (see module "FirstSpirit Launcher JRE": Updating Java versions). After appropriate configuration, FirstSpirit desktop applications can be started via the "FirstSpirit Launcher JRE" module using the FirstSpirit Launcher with AdoptOpenJDK 14. (For documentation on the "FirstSpirit Launcher JRE" module, see the corresponding documentation).

With support for OpenJDK 14, support for **OpenJDK 13** (non-LTS; release date was 2019/07) expires.

FirstSpirit is expected to remain functional with OpenJDK 13. However, e-Spirit AG will neither carry out tests with OpenJDK 13 nor implement any measures designed to eliminate errors or problems that are exclusively associated with the use of OpenJDK 13.

Current status: The following JDKs are therefore compatible for FirstSpirit 2020-04:

- Oracle Java 11 (in its current version) (LTS)

- Oracle Java 8 (in its current version) (LTS; outdated)

- OpenJDK 14 (in its current version) (non-LTS) (**)

- OpenJDK 11 (in its current version) (LTS) (**)

- OpenJDK 8 (in its current version) (LTS; outdated) (*) (**)

(*) Version 8u242 (Outdated) contained in the "FirstSpirit Launcher JRE" module will not be updated to version 8u252 (Outdated) in April 2020.

(**) Recommended: Use of the OpenJDK distribution AdoptOpenJDK (HotSpot JVM).

A complete list of the supported Java versions can be found in Technical requirements and recommendations (Documentation for administrators).

Outlook Java 8 support:

As part of product maintenance, discontinuation of support for Java 8 was first announced with FirstSpirit 2019-05 (**).
FirstSpirit is scheduled to no longer support Oracle Java 8 and OpenJDK 8 from June 2020.

(**) Oracle has not provided any public updates for Java 8 since February 2019 and recommends that developers and end users transition to a higher Java version in order to continue receiving public updates and security enhancements.

Background: FirstSpirit is operated successfully by customers and partners with a wide range of different third-party components (operating systems, JDKs, servlet engines, application servers, and databases) and external libraries. Some of these components / libraries are no longer compatible with Java 8 in its current version. This applies, for example, to the internal "Apache Derby" database supplied with FirstSpirit. As a consequence, updates can no longer be carried out for those third-party components / libraries, which is particularly critical for security-relevant updates.

Should security gaps or security-relevant problems occur under Java 8 in the future, e-Spirit AG reserves the right to terminate Java 8 support for FirstSpirit prematurely (before June 2020).

Also, with the end of Java 8 support, as announced before:

- The support for the Windows installer will be dropped. We advise our customers to use the new installation method via the install archive (see Installing FirstSpirit).

- The option of starting the FirstSpirit desktop applications via Java Web Start will no longer be available. (The Java Web Start functionality is included in Oracle Java 8, but is not support from Oracle Java 11 on. OpenJDK does not offer a replacement.) Therefore, SiteArchitect and ServerManager may only be started via the FirstSpirit Launcher (see FirstSpirit Launcher documentation). The FirstSpirit JNLP files, which were required for Java Web Start, will also be dropped.

The ramp-down phase for Java 8 started as planned in the second quarter of 2020 and will gradually reduce the Java 8 compatibility tests. In 2020Q4 the byte code level will be switched from Java 8 to Java 11. **From this point on, all Java versions prior to Java 11 (LTS) will be incompatible with FirstSpirit.**

This means that FirstSpirit will no longer be executable with Oracle Java 8 or OpenJDK 8 and that e-Spirit AG will no longer implement any measures to eliminate errors or problems associated with Java 8. **We recommend that you switch to a higher Java version soon.**

> **!** On the server side, a change from Java 8 to Java 11 (and higher) should be tested extensively on a test system, especially when using third-party components and individual customer extensions. Java 8 and Java 11 in parts differ considerably, which may result in that existing libraries and implementations are not compatible with Java 11 (and higher). In such cases, updating the respective libraries may help, if possible. When changing from Java 8 to Java 11 (and higher) we recommend changing the FirstSpirit server to Isolated Mode at same time, in order to avoid having to test modules and implementations twice. For this topic see Switching to Isolated Mode.

3.8 New JSON format version 1.1 2020-02

With the current FirstSpirit release a new version of the JSON output format is released. The **new JSON format version 1.1** contains several improvements over version 1.0 (see below). Starting with FirstSpirit 2020-02 the new format version will be used (default setting).

The previous format version 1.0 remains supported but is to be replaced in the medium term.

The format version can be set explicitly via the JSON settings:

```
$CMS_SET(#global.json.formatVersion, "1.1")$
```

> **!** If a stable JSON output is needed even after FirstSpirit updates have been performed, the format version should be set explicitly in the project settings, under "Global Settings".

If the format version is not set explicitly, the latest JSON format version (now 1.1) will always be used.

Overview of the changes:

1) New attribute "page" for page references:

When rendering a page reference, the corresponding page is now rendered as a child attribute "page".

As a result, the page reference attributes only contain values of the page reference. All values that belong to the page (for example, the attribute "formData") are now in the page's JSON object. (Previously, when a page reference was rendered, the values from both the page and the page reference were merged to keep the JSON object structure as simple as possible).

2) The attribute "metaFormData" is always rendered by default:

Any object's meta data will now be rendered, unless rendering is explicitly disabled via `$CMS_SET(#global.json.metaDataRendering, false)$`.

Before the behavior was opposite - metadata rendering explicitly had to be activated with `$CMS_SET(#global.json.metaDataRendering, true)$`.

3) Rendering of display names (attribute "displayname"):

The display name is rendered in the current language, in the attribute "display name", for all nodes that can have display names.

When no display name was defined, the display name is taken from the master language. If that display name has not been set either, the node's reference name is rendered.

4) The attribute "valueSchema" has been removed:

The attribute "valueSchema" is no longer being rendered.

5) The attribute "identifier" has been removed (rendering of input components):

When rendering the data of input components only the attribute "name" is being rendered. The attribute "identifier" has been removed, since both attributes contained the same information.

Rendering of a page reference in format version 1.1 (simplified representation):

```
{
    "fsType": "PageRef",
    "name": "pt_metadata_selective_1",
    "displayName": "pt_metadata_selective",
    "identifier": "15ff7b8d-9059-4721-b92b-e697e153fc85",
    "uid": "pt_metadata_selective_1",
    "uidType": "SITESTORE_LEAF",
    "metaFormData": {
        "meta_field1": {
            "fsType": "CMS_INPUT_TEXT",
            "name": "meta_field1",
            "identifier": "meta_field1",
            "value": null,
            "valueSchema": {
                "type": "null"
            }
        },
        "meta_field2": {
            "fsType": "CMS_INPUT_TEXT",
            "name": "meta_field2",
            "identifier": "meta_field2",
            "value": "value_field2_set",
            "valueSchema": {
                "type": "string"
            }
        },
        "meta_field3": {
            "fsType": "CMS_INPUT_DATE",
            "name": "meta_field3",
            "identifier": "meta_field3",
            "value": null,
            "valueSchema": {
                "type": "null"
            }
        }
    },
    "page": {
        "fsType": "Page",
```

```
        "name": "pt_metadata_selective_1",
        "displayName": "pt_metadata_selective",
        "identifier": "76a0b7c2-3ec8-4908-88e8-53c2f43d9902",
        "uid": "pt_metadata_selective_1",
        "uidType": "PAGESTORE",
        "template": {
            "fsType": "PageTemplate",
            "name": "pt_empty_page",
            "displayName": "pt_empty_page",
            "identifier": "7c020f2f-719c-4334-8efc-f44015333d2a",
            "uid": "pt_empty_page",
            "uidType": "TEMPLATESTORE"
        },
        "formData": {
        },
        "metaFormData": {
            "meta_field1": {
                "fsType": "CMS_INPUT_TEXT",
                "name": "meta_field1",
                "identifier": "meta_field1",
                "value": "value_field_1",
                "valueSchema": {
                    "type": "string"
                }
            },
            "meta_field2": {
                "fsType": "CMS_INPUT_TEXT",
                "name": "meta_field2",
                "identifier": "meta_field2",
                "value": null,
                "valueSchema": {
                    "type": "null"
                }
            },
            "meta_field3": {
                "fsType": "CMS_INPUT_DATE",
                "name": "meta_field3",
                "identifier": "meta_field3",
                "value": null,
                "valueSchema": {
                    "type": "null"
                }
            }
        },
        "children": [
```

```
            ]
        }
    }
}
```

3.9   External database support via the DatasetDataAccessPlugin   2020-04

The DatasetDataAccessPlugin was extended in such a way that external data sources configured in FirstSpirit, on which FirstSpirit does not have write permissions (e.g. read-only referenced databases), can now also be referenced via the FS_INDEX input component.

In this case a coded variant of the primary key of the respective table is used as identifier.

4   ContentCreator

4.1   Workflows: Selecting a deadline time   2020-05

For workflows, a deadline can be specified by which the entire workflow must be completed. The deadline, which can be selected by the editor, is shown to the next editor in the corresponding action window. Furthermore, the deadline is shown in the report area for tasks.

Previously, it was only possible to select a date as the deadline. In the current release, a time can also be selected.
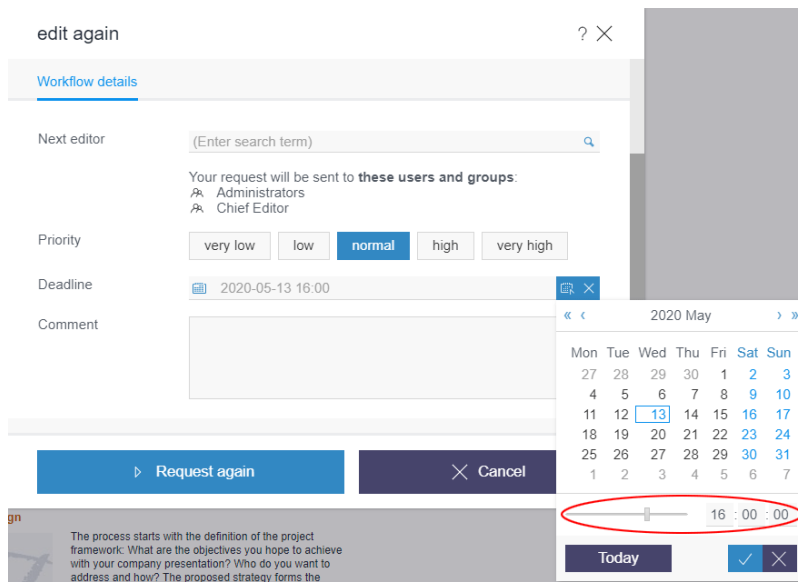
4.2 User prompting/UX: Display of check boxes, radio buttons and toggles 2020-05

The input elements "check box" (CMS_INPUT_CHECKBOX), "radio button" (CMS_INPUT_RADIOBUTTON) and "toggle" (CMS_INPUT_TOGGLE) allow the selection of values specified by the project developer. "Radio buttons" and "toggles" allow the selection of one value; a "check box" allows the selection of multiple values. With "radio buttons" and "toggles", activating a field deactivates any previously selected field; with a "check box", this is not the case.

In the new FirstSpirit ContentCreator, these input elements are now once again easier to distinguish from one another. The editor can more easily see the status of these input elements in a form and how they are operated: "Radio buttons" are now represented by circles, while "check boxes" are boxes.

This new visualization also affects ContentCreator dialogs, such as the "Create new page" dialog:

4.3   ContentCreator re-design: Release 2020-01

The design and technical basis of the FirstSpirit ContentCreator have been revised. The use of the revised ContentCreator (also "FirstSpirit CXT ContentCreator") is officially released with the current FirstSpirit release, the ramp-up / EAP phase is completed.

In principle, the new design does not involve any functional changes **for editors**. At a few central points, the user guidance has been additionally optimized.
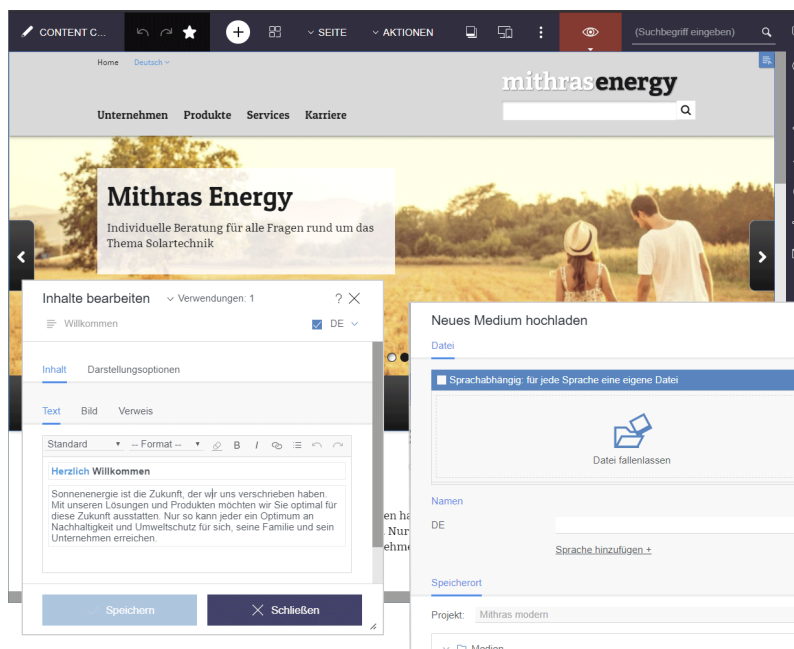
In addition to the new design with fresher colours and modern design language, the redesigned menu bar is particularly striking: The status display has been moved to the right-hand edge of the ContentCreator and actions for creating new content can now be found centrally and prominently under the + icon. Another new feature is that not only pages and datasets are now displayed in the working area, but also media: in contrast to the dialog/window variant of Media administration previously used, this brings with it completely new viewing and editing options: for example, it is now also possible to crop images there. For easier orientation, the menu bar now also shows the element level on which the user is currently located. The report area has been extended to display recently visited elements ("Visited lately"). Additionally, the so-called "Dashboard" now displays the information of the most important reports: the status of the current element, possibilities to start or switch workflows on the element, open tasks and latest visited elements.

For developers, a change to the revised ContentCreator does not in principle involve any migration effort either. Only in very rare cases can there be any compatibility problems with customer-specific modules. In this case please contact e-Spirit Technical Support (https://help.e-spirit.com).

How to activate the revised "FirstSpirit CXT ContentCreator"
The revised ContentCreator is automatically available with the current FirstSpirit release in the form of a module. In future releases the module will receive new features and bug fixes - like other modules - automatically through updates of the FirstSpirit server.

By default, the classic ContentCreator with the "old" design is used for all projects.
In order to use the revised "FirstSpirit CXT ContentCreator", it must be activated (server-wide or project-wise, as required). This is done in the FirstSpirit ServerManager, in each case using the "Use new design" checkbox. This must be activated in the following places:

1) in "Server properties / Web applications" / "ContentCreator" tab:
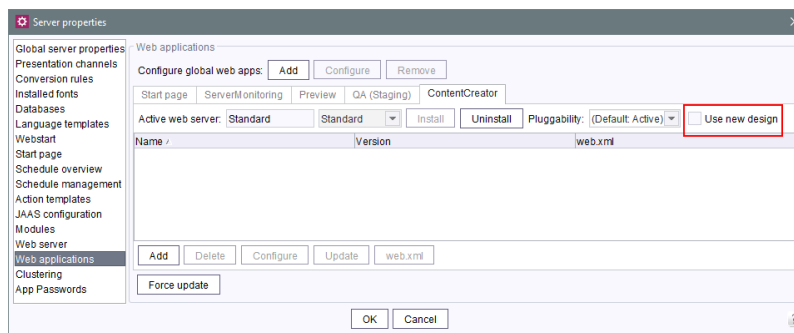
This setting affects all projects that use the global ContentCreator instance. ("Global ContentCreator instance": No web server is configured on the "ContentCreator" tab in the "Project properties / Web components": "[None]").

2) in "Project properties / Web components" / "ContentCreator" tab:
If there are **project-local ContentCreator instances**, the checkbox must also be activated for these. ("Project-local ContentCreator instance": A different web server is configured on the "ContentCreator" tab in "Project properties / Web components".)
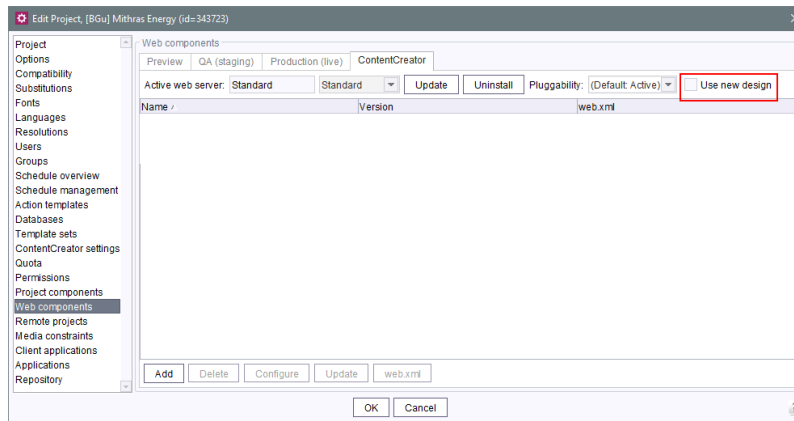
> ! **Important:** The "Pluggability" option must always be set to "Active".
>
> If "Jetty Web Server" is to be used as web server, FirstSpirit must be used at least in version 2019-12.

The modifications must be deployed ("Update" or "Install" button).

With these settings, **all** projects on the server use the ContentCreator in the new design.

For example, if you want to test the new design for **one** project only, a local ContentCreator instance must exist for this project. The new design must be activated for this instance using the repective checkbox. In the server properties, the checkbox must then be deactivated.

If necessary, you can switch between the two ContentCreator versions by activating / deactivating the checkbox (and then deploying).

Documentation

Currently, the user documentation for the ContentCreator is being revised and adapted to the new design; other documentation (e.g. "FirstSpirit Online Documentation" (ODFS)) will follow. The presentation in the documentation therefore does not yet correspond in places to the interface in the new design and may differ from the ContentCreator that you are using.

Further planning

- *FirstSpirit 2020-02/03*: The revised ContentCreator will be used by default for all projects (first for cloud customers, then also for on-premise customers). If necessary, you can switch back to the "old" ContentCreator by deactivating the checkbox.

■ *FirstSpirit 2020-07*: Omission of the "old" ContentCreator. Then only the revised ContentCreator will be available to all customers. Until then the functionality and compatibility of customer-specific modules should be checked and - if necessary with the help of e-Spirit - adapted to the new technological basis.

4.4   Retirement of the "old" ContentCreator and further design improvement 2020-07

The design and technical basis of FirstSpirit ContentCreator were revised. With FirstSpirit 2020-01, the revised ContentCreator was officially released for use. As of FirstSpirit 2020-03, the revised ContentCreator is used by default for all projects. In addition to the internal quality assurance measures, feedback on specific operational scenarios from customers and partners has since been promptly incorporated into the development.

If necessary, it was possible to easily switch between the old and new ContentCreator on a project-by-project basis. We plan to **retire the "old" ContentCreator** with the current release; at this point, all customers will only be able to use the revised Content Creator.

The checkbox which previously enabled users to switch between the old and new ContentCreator (FirstSpirit ServerManager / "Server properties / Web applications" or "Project properties / Web components" / "ContentCreator", option "Use new design") has been removed in this context. When switching from previous FirstSpirit versions to 2020-07, settings which were made for the server and/or project in those versions will no longer be taken into account. From FirstSpirit Server 2020-07, only the revised ContentCreator is used.
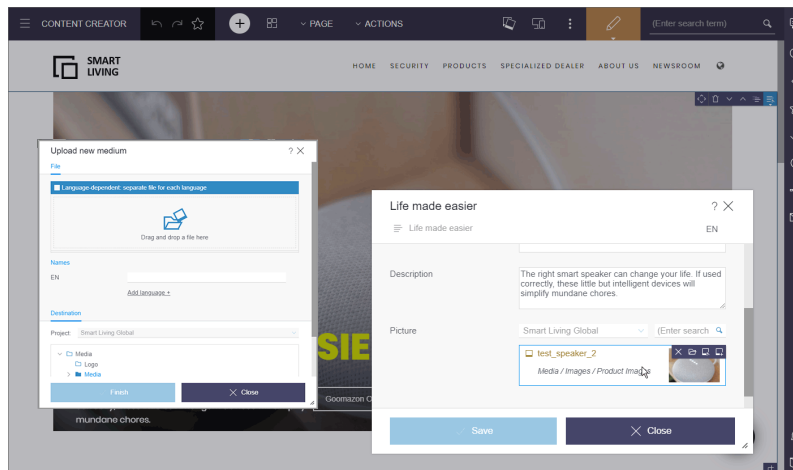
For editors, no functional changes are associated with the new design in principle. At just a few central locations, the user prompting and guidance has additionally been optimized.

As of FirstSpirit 2020-07 a number of design and user experience optimizations were made based on customer feedback, including

- "ACTIONS" menu:
  Here you can now visually differentiate between scripts, (contextless) workflows and plugIns, again.

- FS_CATALOG/FS_INDEX:
  New display option "Grid" (compared to "Symbols", the thumbnails are not adjusted to the existing space)

- Media management:
  The last row of entries is no longer stretched over the entire page width.

- Adaptation of dialog buttons for a uniform cross-client UI

- Optimized layout for integrated help texts and tooltips

- In release messages, which are temporarily displayed in the upper left corner below the menu bar, the relevant names are now also displayed for media folders and the top level of media management ("Media").

For **developers and administrators**, the switch to the revised ContentCreator does not involve any migration work. Only in very rare cases, there may be compatibility problems with customer-specific modules. In such cases, please contact e-Spirit Technical Support (https://help.e-spirit.com/).

Information on using the new FirstSpirit ContentCreator

- **Resource requirements:** The change in technologies used for ContentCreator ("MicroApp Framework") increases the resource and system requirements. For this reason, you should

check and, if necessary, increase resources, in particular for the application server (e.g., Apache Tomcat) and specifically the **hard drive capacity** and **Java Heap**.

Where possible, ContentCreator should be used as a **global web application**. ContentCreator should only be run as its own project instance ("projectlocal") where it is absolutely necessary to do so.

- **Working with Apache Tomcat:** If Apache Tomcat is used as the application server, we recommend that Apache Tomcat is **restarted** after each update.

- **Pluggability:** The **Pluggability** option has to be activated for unrestricted use of the new FirstSpirit Content Creator (FirstSpirit ServerManager / "Server properties / Web applications" or "Project properties / Web components" / "ContentCreator"). When using ContentCreator, the system automatically activates pluggability mode (log entry: `WARN    03.02.2020    15:58:20.902 (de.espirit.firstspirit.server.module.WebServerManagerImpl):    Force enabled pluggability for 'fs5webedit' in project '-1'.`). This has no effect on the interface itself; the option "Pluggability" may still show as being set to "Inactive" even though the option is activated.

- **Working in the browser:** The new ContentCreator is not compatible with Microsoft Internet Explorer. We recommend using Google Chrome and Microsoft Edge Chromium for the new ContentCreator. See also chapter "Cessation of support for Microsoft Internet Explorer".

Manual

The manual has already largely been adapted to the new design. In just a few places, the interfaces illustrated do not match the interface in the new design and may differ from the version of ContentCreator you are using.

4.5   Restoring deleted pages   2020-08

Deleted pages can now be restored in FirstSpirit ContentCreator via the status menu.

To do this, the deleted page must first be searched for in the project history. By clicking on the corresponding entry, the page is displayed in the state in which it was deleted.

By clicking on the menu item "Revert to this version" in the status menu
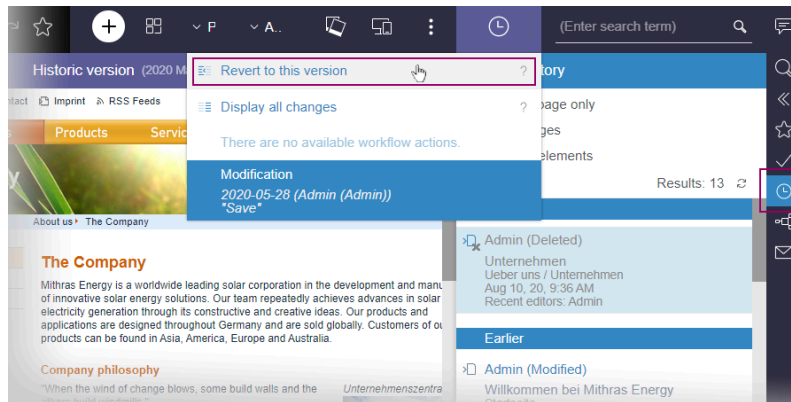
the page can be restored.

Both the page and the menu item are restored (if a menu item was available).

In combination with "BasicWorkflows" (module file *basicworkflows.fsm*), problems may occur when restoring.

4.6 Navigation view and evaluation of permissions: Hiding invisible elements 2020-06

Permission management in FirstSpirit makes it possible to assign permissions to users and groups. Permissions can be assigned with a high level of granularity for each FirstSpirit element in a project. Permissions are assigned in FirstSpirit SiteArchitect using the "Extras / Change permissions" context menu function on the required element or subtree of a project.

Elements of the Site Store (menu levels, page references, etc.)

that are not visible to the current user ("Visible" checkbox is deactivated), such as
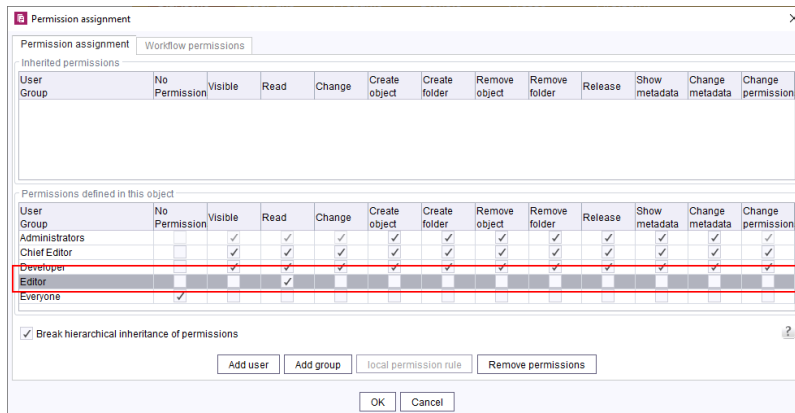
are now no longer shown on the left-hand edge of the screen in the navigation view.

Previously, *** No access *** was shown in these cases:
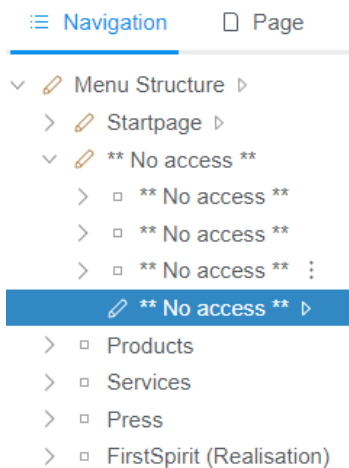
If a user or a group should not be able to **edit** the content of a page, the permissions for the page in question need to be assigned in the **Page Store**.

For more information about permission assignment in FirstSpirit, see https://docs.e-spirit.com/odfs/edocs/fsar/permissions-sit/editorial-permi/index.html.

First Spirit™
Unlock Your Content

4.7 New navigation view in the ContentCreator combines hierarchical structure display, context information and editing options 2020-04

In order to make work for editors particularly simple and intuitive, the **preview** is the focus of the FirstSpirit ContentCreator's operating concept: Content is edited directly on the website in the browser. This allows the editor to see immediately how changes will later affect the website. Previously, only the preview itself and reports (e.g. search, Related elements) were used to navigate to the pages to be edited.

Due to increased requirements for editing projects using the ContentCreator, the current release now also displays the project or website structure in a **structure tree**.

Among other things, it permits the following:

- a quick overview of structural relationships within the website structure (**hierarchy**)

- an uncomplicated, direct **navigation** to pages, even if they are not accessible via the website navigation (typical for landing pages or technical project pages)

- a quick overview of **contextual information** on individual pages (e.g. release status, presence of metadata, etc.)

- **editing** vof properties and contents of pages and sections

- simple and intuitive **moving** of menu items

Synchronization and highlighting bridge the gap between preview and structure information: a click on an element in the structure shows the corresponding page in the preview and highlights it in the preview. In turn, the new navigation view also follows the editor's mouse pointer in the preview.

Additionally, the new navigation view allows switching between different **languages** of the project.
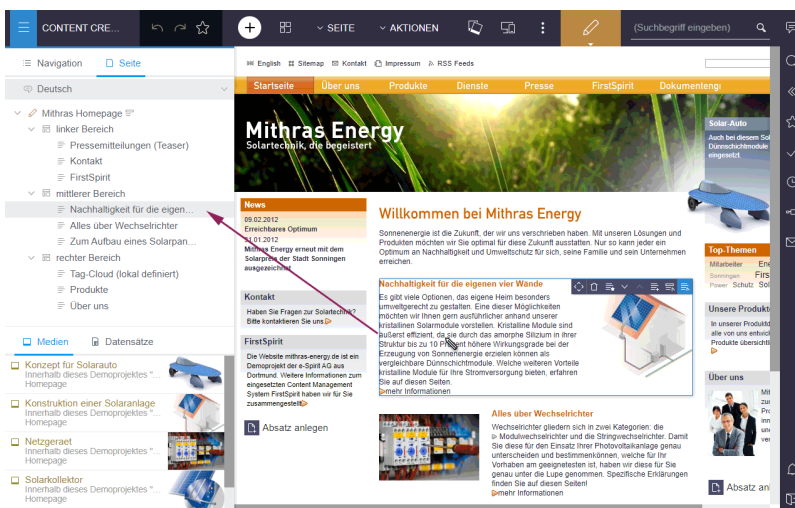
The corresponding function is located behind the icon on the far left of the menu bar:

In this context, the dialog for editing the navigation, which was previously available under the menu entry "Edit navigation" in the "ACTIONS" menu, has been removed, since the same functions are now offered by the new navigation view. Instead of the dialog, the "Edit navigation" menu entry now opens the navigation view.

It is planned to include new functions in this new navigation view in future releases. The new navigation view is **not** available in the old ContentCreator.

Note for developers and administrators: The new navigation view displays the **entire** site store of a project by default. Unlike before, editors can now navigate to any page. And thus, probably also to more "technical" pages that may not have any output that is readable by the editor. This should be taken into account during project development and nodes in the site store that are not relevant for editors should be hidden by appropriate permission assignment.

4.8 New ContentCreator: Rollout for all customers   2020-03

The design and technical basis of FirstSpirit ContentCreator were revised:

- With **FirstSpirit 2020-01**, the revised ContentCreator was officially released for use. Since this point, all customers have been able to use the revised design of the ContentCreator as long as the appropriate settings have been configured accordingly in server/project settings.

- With **FirstSpirit 2020-02**, the "new" ContentCreator was automatically rolled out to all cloud customers.

With the **current release**, the "new" ContentCreator is now automatically used **for all** projects started with ContentCreator (for cloud **and** on-premises customers).
If necessary, the customer can switch back to the "original" ContentCreator for a project by deactivating the checkbox "Use new design" on the "ContentCreator" tab in FirstSpirit ServerManager, under "Server properties / Web applications" or "Project properties / Web components". See also *FirstSpirit Manual for Administrators*, FirstSpirit ServerManager / Server properties / Web applications or FirstSpirit ServerManager / Project properties / Web components.
With FirstSpirit 2020-07, we are planning to remove the "old" ContentCreator design, so we recommend that customers start to use the "new" ContentCreator for all projects as soon as possible.

Why and how was the ContentCreator redesigned? – Check out our blog for a glimpse behind the scenes of the design and development process for the new ContentCreator: www.e-spirit.com/us/blog/contentcreator_redesign.html.

Interactive help and documentation

As part of the redesign, we incorporated more help messages and tips into the standard ContentCreator interface to help users understand the changes. These messages can be switched off in a message dialog ("Deactivate all tips").
Large-scale updates were made to the user documentation for ContentCreator and the documentation for developers ("FirstSpirit Online Documentation" (ODFS)) based on the new design; other documentation will follow. In some places, the interfaces shown in the documentation do not match the interface in the new design and may differ from the version of ContentCreator you are using.

Information on using the new FirstSpirit ContentCreator

- Up to and including FirstSpirit 2020-01, the **Pluggability** option had to be set to "Active" for unrestricted use of the new FirstSpirit ContentCreator. Since the rollout of FirstSpirit 2020-02, pluggability is force-enabled for the new FirstSpirit ContentCreator. This means that if the new FirstSpirit ContentCreator is used, the system

automatically activates pluggability mode (log entry: `WARN 03.02.2020 15:58:20.902 (de.espirit.firstspirit.server.module.WebServerManagerImpl): Force enabled pluggability for 'fs5webedit' in project '-1'.`). This has no effect on the interface itself; the option "Pluggability" may still show as being set to "Inactive" even though the option is activated.

For further information on the option "Pluggability", see also *FirstSpirit Manual for Administrators*, page Web applications or Web components.

- The change in technologies used for the ContentCreator ("MicroApp Framework") increases the resource and **system requirements**. This is particularly noticeable during the currently ongoing transition period, in which the "new" and the "old" versions of ContentCreator can be run in parallel.

  - For this reason, you should check and, if necessary, increase **resources**, in particular for the application server (e.g., Apache Tomcat) and specifically the **hard drive capacity** and **Java Heap**.

  - Where possible, ContentCreator should be used as a **global web application**. ContentCreator should only be run as its own project instance ("projectlocal") where it is absolutely necessary to do so.

  - If Apache Tomcat is used as the application server, we recommend that Apache Tomcat is **restarted** after each update.

- If "Jetty Web Server" is used as a web server, you must use at least FirstSpirit version 2019-12.

4.9   Project selection `2020-03`

With the current release, the handling of project selection has been improved.:

If only **one project** is available to the editor on a server, this project is started directly when ContentCreator is started via the FirstSpirit start page using the following icon:
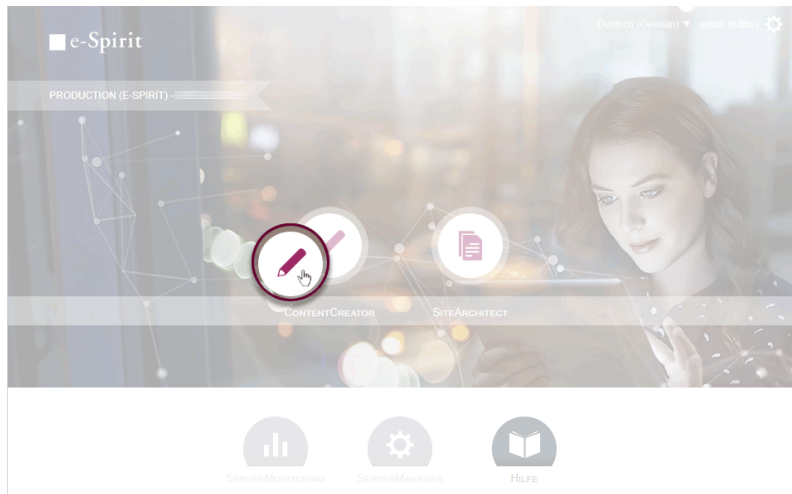
The project selection dialog is no longer displayed.

If **several projects** are available to the editor on a server, he can now easily switch between the projects using the new function "Change project" in the "Further functions" menu (3 points):
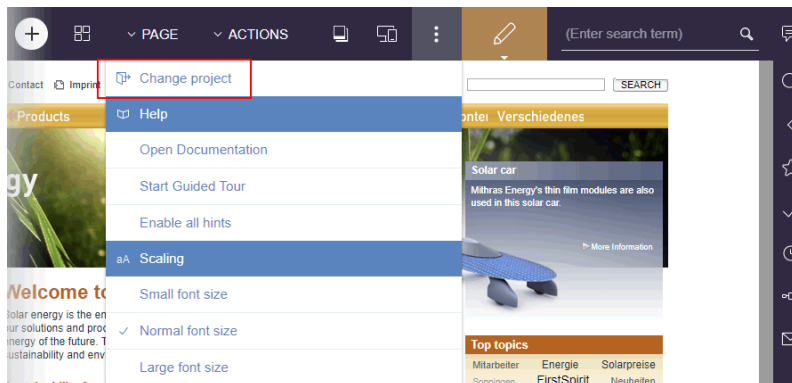
This function logs off the user and opens the project selection dialog from which the user can select the desired project for editing (via ContentCreator):
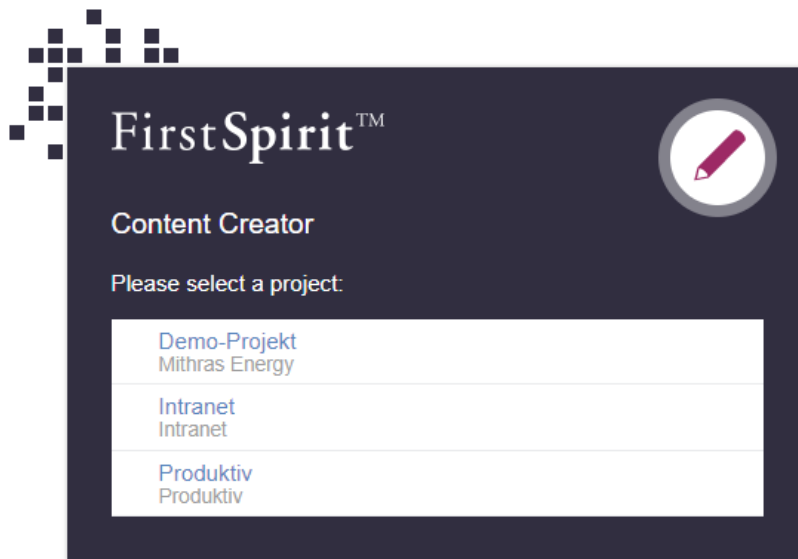
4.10   Ramp-up: Less manual image cropping thanks to "Smart Cropping"

2020-07

The concept of "image resolutions" in FirstSpirit has always enabled the uniform use of image sizes in online presences. The resolutions required for a project are automatically calculated and generated by the system as standard.

However, depending on the image material and project settings, these automatically generated resolutions must often be checked by the editor and manually adjusted: In some project setups (e.g. when concrete widths and heights are specified for images) the resolutions are compressed, the proportions of the delivered image material do not match the proportions and sizes required for the internet presence. In other cases, the editor may want to select a different focus for some images and move a different area to the center of attention.

Smart Cropping largely eliminates the need to check the individual resolutions and manual adjustment by the editor: thanks to intelligent technology, a suitable section can be selected manually for all resolutions of one image at the same time.
For this purpose, a suitable cropping is automatically calculated for each resolution on the basis of an area that the editor can determine in the image as the main motif using a selection frame ("focus area"). In this process the "Golden ratio" is applied: This ensures that the image composition appears organically and harmoniously proportioned instead of static and constructed.

By default, this is done from the center of the image, i.e. the cutouts are extended towards the center of the image. However, the editor can also use the "angle of view" function to determine the direction of the original image in which the cutouts should be extended (top, bottom, right, left...). With this angle of view you can control which area of the image is relevant for the resolutions and should be considered for the automatic cropping.

In the case of images with people, for example, the viewing angle can ideally be selected so that the person and his or her viewing direction as well as the target is in the focus area.
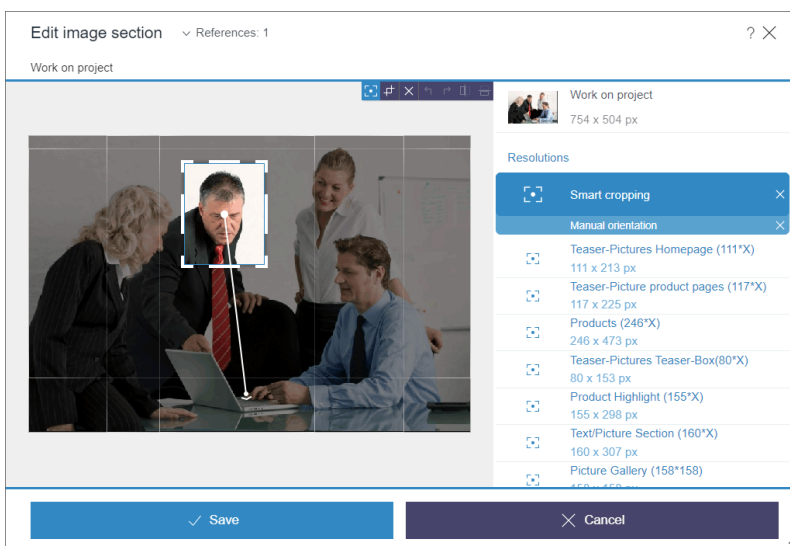
In this context, existing APIs were extended and new ones were created, for example to define focus areas without user interaction.

The new feature can be tested with FirstSpirit 2020-07 in the scope of a ramp-up phase and will be made available to some selected customers and partners for trying out. Feedback from this phase is promptly incorporated into the development. Are you also interested in testing the new functionality? Then please contact your responsible customer or partner manager.

An official release for all customers is planned for the 4th quarter of 2020.

4.11   Rollout of the new ContentCreator   2020-02

Design and technical basis of the FirstSpirit ContentCreator have been revised. With FirstSpirit 2020-01 the use of the revised ContentCreator was officially released. Since then, all customers have been able

to use the ContentCreator in the revised design, as long as this has been configured accordingly in the server/project settings.

As scheduled, the "new" ContentCreator will be automatically rolled out to all **cloud customers** with the current release: For all projects started with the ContentCreator, the revised design will be used as of this release.

If required, you can switch back to the "classic" ContentCreator for individual projects by deactivating the "Use new design" checkbox in the FirstSpirit ServerManager under "Project properties / Web components" / "ContentCreator" tab.

For **all customers**, the "new" ContentCreator will be used as planned with the upcoming release (FirstSpirit 2020-03): Then the revised ContentCreator will automatically be used for all projects. If any problems occur, the "old" can be easily switched back to if necessary. With FirstSpirit 2020-07, however, the ContentCreator in the "old" design will be discontinued as planned, so the recommendation is to use the new ContentCreator for all projects as early as possible.

Why was the ContentCreator revised at all and how? - Our blog article will give you a look behind the scenes of the design development for the new ContentCreator: https://www.e-spirit.com/en/blog/contentcreator_redesign.html.

Documentation

Currently, the user documentation for the ContentCreator is being revised and adapted to the new design; other documentation (e.g. "FirstSpirit Online Documentation" (ODFS)) will follow. The presentation in the documentation therefore does not yet correspond in places to the interface in the new design and may differ from the ContentCreator that you are using.

In addition, you can now find the user documentation for the ContentCreator as an HTML version (instead of PDF) on the central documentation server docs.e-spirit.com.

Notes on operating the new FirstSpirit ContentCreator

- Up to now, for unrestricted use of the new FirstSpirit ContentCreator, the **Pluggability** option had to be set to "Active". With the current release, pluggability is enforced for the new FirstSpirit ContentCreator. This means: If the new FirstSpirit ContentCreator is used, the Pluggability mode is automatically activated by the system (log entry: `WARN 03.02.2020 15:58:20.902 (de.espirit.firstspirit.server.module.WebServerManagerImpl):   Force enabled pluggability for 'fs5webedit' in project '-1'.`). This has no effect

on the user interface, so the "Pluggability" option may be set to "Inactive" even though the option is enabled.

For more information about the "Pluggability" option please see *FirstSpirit Manual for Administrators*, page Web applications or Web components.

- The technology change of the ContentCreator ("MicroApp Framework") leads to an increased resource requirement and **increased system requirements**. This is particularly noticeable during the current transition period, during which the "new" and the "old" ContentCreator can be operated in parallel.

  - For this reason, **resources** should be checked and, if necessary, increased especially in the area of the application server used (e.g. Apache Tomcat), especially the **hard disk space** and the **Java heap**.

  - If possible, the ContentCreator should be used as a **global Web application**. The ContentCreator should only be operated as a separate project instance ("project local") if this is absolutely necessary.

  - When using Apache Tomcat as an application server, it is generally recommended that Apache Tomcat be **restarted** after each update of FirstSpirit.

4.12 Tables: Vertical alignment of content in cells ("valign") 2020-05

The **vertical alignment** can now be determined for content in tables (CMS_INPUT_DOM / "Inline tables" and CMS_INPUT_DOMTABLE) in FirstSpirit ContentCreator. To do this, the new attribute `valign` can be used in a similar way to the existing attribute `align` (for horizontal alignment). The predefined keywords for the different alignments are

- *top*

- *middle* (default setting)

- *bottom*

1) Example – CMS_INPUT_DOM / Inline tables:

Style templates can be used to define table layout properties for inline tables, e.g., background color, text alignment, etc. The form for entering vertical text alignment in a style template (e.g., in the "standard style template") must contain the identifier `valign`.

Example:

```
<CMS_INPUT_RADIOBUTTON name="valign" gridHeight="1" useLanguages="no">
    <ENTRIES>
      <ENTRY value="top">
        <LANGINFOS>
          <LANGINFO lang="*" label="Top"/>
          <LANGINFO lang="DE" label="Oben"/>
        </LANGINFOS>
      </ENTRY>
      <ENTRY value="middle">
        <LANGINFOS>
          <LANGINFO lang="*" label="Centre"/>
          <LANGINFO lang="DE" label="Zentriert"/>
        </LANGINFOS>
      </ENTRY>
      <ENTRY value="bottom">
        <LANGINFOS>
          <LANGINFO lang="*" label="Bottom"/>
          <LANGINFO lang="DE" label="Unten"/>
       </LANGINFOS>
      </ENTRY>
    </ENTRIES>
    <LANGINFOS>
      <LANGINFO lang="*" label="Vertical alignment"/>
      <LANGINFO lang="DE" label="Vertikale Ausrichtung"/>
    </LANGINFOS>
</CMS_INPUT_RADIOBUTTON>
```

Display in ContentCreator:

The alignment that is defined by the editor in the corresponding dialog "Edit cell properties" can be output in the HTML template set of the style template, for example, and therefore in the preview and on the generated page, e.g.,

```
...
style='
    ...
    $CMS_VALUE(if(!valign.isEmpty, "vertical-align:" + valign + ";"))$
    ...
        '
```

2) Example – CMS_INPUT_DOMTABLE:

For CMS_INPUT_DOMTABLE, cell properties are defined by the editor via a dialog that can be configured via the attribute `propertyConfig`. The vertical alignment of content can also be activated via the keyword *valign*, e.g.,

```
<CMS_INPUT_DOMTABLE name="table"
        propertyConfig="
        valign:Vertical alignment:
           [top:Top|middle:Middle|bottom:Bottom],
        align:Horizontal alignment:
           [left:Left|center:Center|right:Right]">
    <LANGINFOS>
      <LANGINFO lang="*" label="Table"/>
      <LANGINFO lang="DE" label="Tabelle"/>
    </LANGINFOS>
</CMS_INPUT_DOMTABLE>
```

Display in ContentCreator:

The properties of a cell of a CMS_INPUT_DOMTABLE, defined by the editor, can be evaluated and output via the HTML template set of the standard format template "Table cell". The system object `#cell` is available for this purpose.

Example:

```
<td..
    ...
    $CMS_VALUE(if(!#cell.valign.isEmpty, " valign='" + #cell.valign + "'"))$
    $CMS_VALUE(if(!#cell.align.isEmpty, " align='" + if(#cell.align == "block",
        "justify", #cell.align) + "'"))$
    ...
</td>
```

Note: There are **no** plans to introduce corresponding support for the attribute `valign` for FirstSpirit SiteArchitect.

For more information, see

- Templates for inline tables: https://docs.e-spirit.com/odfs/templates-basic/composition-tem/inline-tables/index.html

- CMS_INPUT_DOMTABLE: https://docs.e-spirit.com/odfs/template-develo/forms/input-component/domtable/index.html

- `#style` system object: https://docs.e-spirit.com/odfs/template-develo/template-syntax/system-objects/style/index.html

- `#cell` system object: https://docs.e-spirit.com/odfs/template-develo/template-syntax/system-objects/cell/index.html

4.13   Restoring a version in the "Show modifications" dialog   2020-09

In the "Comparison View" for a page, changes that have been made to the page can be visualized in the "Show modifications" dialog. This is opened via the "Display differences" icon:
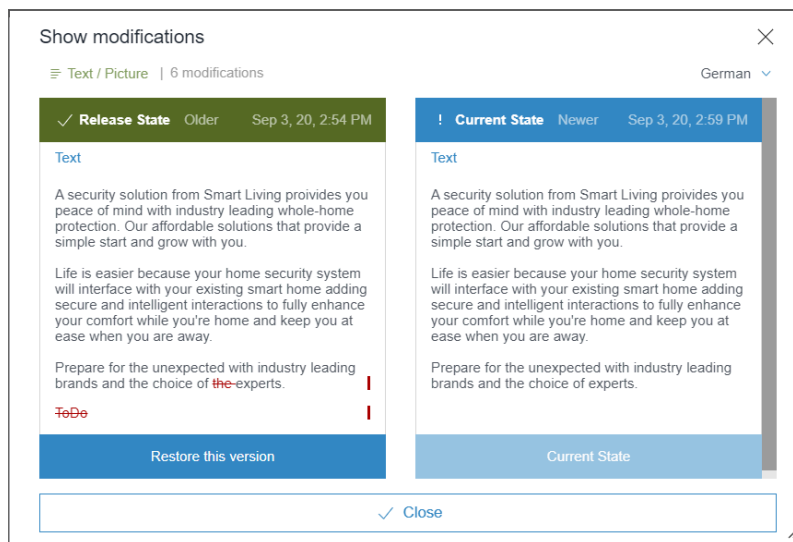
The user guidance when restoring a version has been modified in this dialog (via the "Restore this version" button or icon

Previously, the dialog closed immediately after restoring the version, but now it remains open until the "Restoration succeeded. The current version will now be loaded" information dialog is confirmed with "OK":
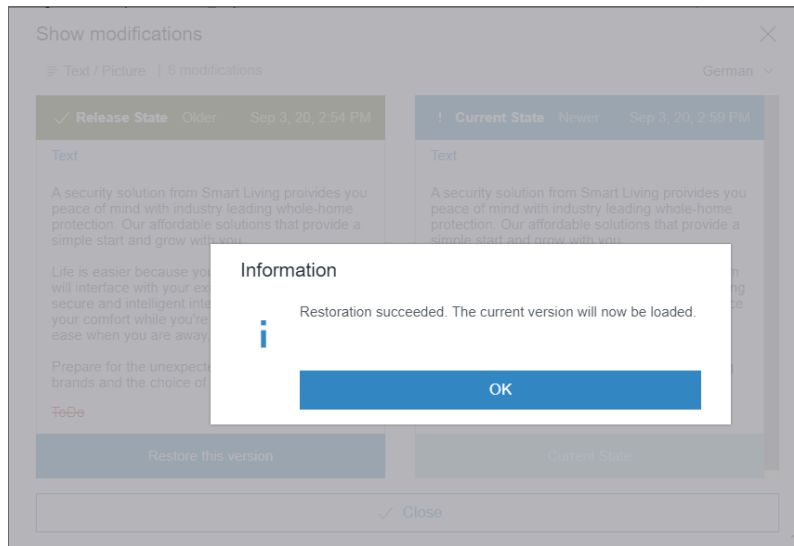
In this way, it is now possible – when using the API (`ComparisonDialogOperation` interface, `de.espirit.firstspirit.webedit.server` package, FirstSpirit Developer API) – to hold off on closing the dialog until another action is carried out.

4.14   Use of the "old" ContentCreator 2020-04

The design and technical basis of FirstSpirit ContentCreator were revised.

By default, the "new" ContentCreator is automatically used for all projects that are started with the ContentCreator. However, you can switch back to the "classic" ContentCreator for individual projects if required.

With FirstSpirit 2020-07, the ContentCreator in the "old" design will be eliminated as planned. The recommendation is therefore to use the "new" ContentCreator for all projects as early as possible.

If one or more web applications are configured to run on the "old" of a FirstSpirit server, the start page of the FirstSpirit server will show a warning to users with the administrator role:

```
Legacy ContentCreator usage detected. Web application: ...
```

For further information please see https://docs.e-spirit.com/odfs/edocs/admi/firstspirit-ser/server-properti/web-application/index.html.

> **!** Note on the **new** ContentCreator: It is recommended that you use Google Chrome or Microsoft Edge on chromium basis for the new ContentCreator. The new ContentCreator is not compatible with Microsoft Internet Explorer.

5 Input components

5.1 The output behavior when accessing the CMS_INPUT_PERMISSION input component via the API has been corrected. 2020-07

If no metadata has been defined for an element or if the box for the "Define permissions" option in the CMS_INPUT_PERMISSION input component has not been checked, an empty list is now returned for the input component.

Before querying the content, a check should be carried out to see whether metadata has been defined for this element and whether these permissions have been applied or inherited there.

```
( <IDProvider>.hasMeta() && !<IDProvider>.getMetaFormData().get(null,
    <Name of the input component>).get().isInherited() )
```

If the above query returns "false", the correct values are located on the node in the parent chain for which this query returns "true".

5.2 FS_LIST: Ramp-down phase complete: input component no longer included 2020-07

As of FirstSpirit 2020-07, the FS_LIST input component is no longer included. The component has been removed from FirstSpirit and is no longer available for use. This means it is no longer possible to edit forms with FS_LIST and expected content will no longer be generated on the output side. This concludes the ramp-down phase.

Overview: Process of ramp-down phase for input component FS_LIST

To ensure that the FS_LIST component was phased out with minimum risk, it was removed in multiple stages:

- (Complete) As of FirstSpirit 5.2R3, the input component for creating and managing lists FS_LIST was **discontinued**.

- (Complete) As of FirstSpirit 2019-11, a **deprecation warning** was logged as a first step in all projects, when data was generated from an FS_LIST input component, or when data from an FS_LIST component was previewed.

- (Complete) As of FirstSpirit 2020-01, the FS_LIST input component **stopped receiving support**. It was no longer possible to use the component from this point.

- To minimize the risk of production problems, particularly in cases involving customers with longer update cycles, there was a six-month transition period during which it was possible to reactivate FS_LIST temporarily.

- (Complete) As of FirstSpirit 2020-07, the input component has been **removed** from FirstSpirit and can no longer be used in any projects.

The FS_CATALOG, FS_INDEX, and CMS_INPUT_SECTIONLIST input components are available as substitutes. They also provide content bundling functions, which means that they largely cover the same functions as FS_LIST.

- FS_LIST, type DATABASE can be substituted with an input component of type FS_INDEX.

- FS_LIST, type INLINE can be substituted with an input component of type FS_CATALOG.

- FS_LIST with MEDIAMODE tag can be substituted with an input component of type FS_INDEX, using the `DatasetDataAccessPlugin` (`SOURCE` tag) supplied as standard.

- FS_LIST, type PAGE can be substituted with the input component CMS_INPUT_SECTIONLIST.

- FS_LIST, type SERVICE can be substituted with an input component of type FS_INDEX, using a corresponding database access plug-in.

Detailed guidelines for performing conversion in ongoing projects are provided in the FirstSpirit documentation.

> ! Once a conversion to the new input components has been performed, it will no longer be possible to revert back to FS_LIST, even if the server is downgraded.

If you have any questions or feedback on migrating FS_LIST, please contact e-Spirit Technical Support.

5.3 FS_LIST: Ramp-down phase: The input component is no longer supported 2020-01

With FirstSpirit 2020-01, the input component FS_LIST is no longer supported. The component can no longer be used from this point on. That means forms with FS_LIST can no longer be edited and the expected output will no longer be generated.

Overview: steps of the ramp down phase for the FS_LIST input component

In order to make the phasing out of FS_LIST as low-risk as possible, the removal is carried out in several steps:

1. (Done) With FirstSpirit version 5.2R3 the input component FS_LIST, for creating and maintaining lists, has been deprecated.

2. (Done) With FirstSpirit 2019-11 in all projects, when generating or previewing data from the FS_LIST input component a **deprecation warning** will be logged.

3. (Done) With FirstSpirit 2020-01 the input component is **no longer supported**. The component can no longer be used from this point on.

4. In order to minimize the risk of production problems, particularly for customers with longer update cycles, during a six months transition period it will be possible to temporarily re-active FS_LIST. In order to do so please contact e-Spirit Technical Support.

5. **With FirstSpirit 2020-07** the input component will be **removed** from FirstSpirit, and can no longer be used in any project from that moment on.

Migration: switch to alternative FirstSpirit input components

> **!** e-Spirit recommends developers and end users to switch to alternative FirstSpirit input components at an early stage.
> Before updating to FirstSpirit 2020-01 or higher, all FS_LIST uses must be converted to one of the successor components.

As replacements the input components FS_CATALOG, FS_INDEX, and CMS_INPUT_SECTIONLIST are available, likewise providing functionality for the bundling of contents and thus largely taking over the FS_LIST use cases.

- FS_LIST, type DATABASE can be replaced by an FS_INDEX input component.

- FS_LIST, type INLINE can be replaced by an FS_CATALOG input component.

- FS_LIST with tag MEDIAMODE can be replaced by an FS_INDEX input component, when using the `DatasetDataAccessPlugin` (tag `SOURCE`), which is delivered with FirstSpirit by default.

- FS_LIST, type PAGE can be replaced by the input component CMS_INPUT_SECTIONLIST.

- FS_LIST, type SERVICE can be replaced by an FS_INDEX input component, when using a corresponding data access plug-in.

A detailed guide for the transition of existing projects is provided via the FirstSpirit Documentation.

> **!** After the transition to the new input components a return to FS_LIST is not possible - not even after a server downgrade.

Please contact e-Spirit Technical Support if you have questions or feedback regarding the migration away from FS_LIST.

5.4 Improved Deprecation Warnings 2020-01

Starting with **FirstSpirit 2019-11** when generating or previewing data from input components with the status "deprecated" a deprecation warning is logged. This message is meant to make it easier for

users to locate and migrate "deprecated" input components (e.g. in order to facilitate the FS_LIST input component's ramp-down phase).

With the current FirstSpirit release this service is improved. This was achieved by consolidating all deprecation warnings of a project, and writing them to the first section of the log file (in the case of generation in the generation log file, and in the case of a preview under "Show preview errors").

Note: Now when invoking "Show preview errors" on a page, the deprecation warnings for input components without any connection to that page will also be rendered.

"Deprecated" input components:

- are no longer maintained by e-Spirit

- are still functional and can be used in FirstSpirit projects

- will be dropped in the future

The following input components are currently "deprecated":

- CMS_INPUT_CONTENTAREALIST (should be migrated to FS_CATALOG)

- CMS_INPUT_CONTENTLIST (should be migrated to FS_INDEX)

- CMS_INPUT_FILE (should be migrated to FS_REFERENCE)

- CMS_INPUT_LINKLIST (should be migrated to FS_CATALOG)

- CMS_INPUT_OBJECTCHOOSER (should be migrated to FS_INDEX)

- CMS_INPUT_PAGEREF (should be migrated to FS_REFERENCE)

- CMS_INPUT_PICTURE (should be migrated to FS_REFERENCE)

- CMS_INPUT_TABLIST (should be migrated to FS_INDEX)

When using a deprecated input component and

- generating or

- calling "Show preview errors" on a page

all deprecation warnings of a project, with the location of the usage (i.e. the respective template), are logged. Example:

```
Usage of deprecated form element CMS_INPUT_PICTURE for variable 'ttPicture' in
template 'testschema.testtable' (ID: 455559)
```

This applies to all components used in a form, even if they do not render contents via the template.

The deprecation warning can be disabled by

- commenting out the input component in the form with CMS_COMMENT or by

- migrating the input component in the project to a different input component.

Contact e-Spirit Technical Support if you have questions or feedback regarding the migration.

6   External Synchronization

6.1   Transporting the Activity Status of Schedule Entries 2020-08

When importing new and existing schedule entries using the "External Synchronization" function, the **activity status of the schedule entries** can now be transferred from the source project to the target projects.

The activity status can be managed via the command line tool FSDevTools (fs-cli) or via the FirstSpirit Developer API.

Via the command line tool FSDevTools (fs-cli):

With FSDevTools, the command line call is triggered by the `--import-schedule-entry-active-state` parameter.

Example:

```
D:\fs-cli\bin> fs-cli -h example.com [connectionparameters] import
--import-schedule-entry-active-state
```

If the parameter is set, the activity status of new and existing schedule entries is transferred from the source project to the target projects. For existing schedule entries, the value in the target project may be overwritten by the value from the source project.

If the parameter is not set, the activity status of new schedule entries is set to inactive (active=false) by default when you import them. Existing schedule entries remain unchanged in the target projects.

Via the FirstSpirit Developer API:

The new method `setImportScheduleEntryActiveState(boolean)` of the ImportOperation interface (FirstSpirit Developer API,

`package   de.espirit.firstspirit.store.access.nexport.operations`) is used to manage the activity status of the schedule entries in the target projects.

Possible values:

- Default case, `false`: If the value `false` is set or the method is not called (standard case), the activity status is set to inactive (active=false) by default when importing new schedule entries. Existing schedule entries remain unchanged in the target projects.

- `true`: If the value `true` is set, the activity status of new and existing schedule entries is transferred from the source project to the target projects. For existing schedule entries, the value in the target project may be overwritten by the value from the source project.

Example:

```
import de.espirit.firstspirit.agency.OperationAgent;
import de.espirit.firstspirit.store.access.nexport.operations.ImportOperation;
import de.espirit.firstspirit.io.FileSystemsAgent;
importOp =
context.requireSpecialist(OperationAgent.TYPE).getOperation(ImportOperation.TYPE);
fileSystem =
context.requireSpecialist(FileSystemsAgent.TYPE).getOSFileSystem("c:/fs-cli/exam/");
importOp.setImportScheduleEntryActiveState(true);
importOp.perform(fileSystem);
```

Further information:

- Interface ImportOperation (FirstSpirit Developer-API,
  package `de.espirit.firstspirit.store.access.nexport.operations`)

- Import via FSDevTools

7   External Synchronization / Content Transport

7.1   External Synchronization and Content Transport: Improved support for shared access to a database 2020-05

Using the "FirstSpirit External Synchronization" and "FirstSpirit Content Transport" mechanisms, database content (datasets) and the associated database structures (schemas, tables, columns) of a FirstSpirit project (source project) can be exported and imported into other FirstSpirit projects (target projects). The `dbnames` of column names and tables are recalculated during synchronization. This behavior can, under the following conditions, lead to the problem that columns and tables in the target projects cannot be found in the database:

- the FirstSpirit schemas of the participating projects reference the same database tables (and do not point to their own database tablespaces) and

- the `dbnames` do not match the automatically calculated `dbnames`

For this special use case an additional mapping file (`Mapping.xml`) can now be used when exporting and importing database schemas. In this file the database names of the source project are stored together with FirstSpirit's own unique identifiers (UUIDs), e. g.:

```
<xs:gid dbName="j_categories_column_x" uuid="66843bbe-da0d-46e3-a136-76bd93d782bb"/>
<xs:gid dbName="j_categories_column_y" uuid="4ba29641-26af-4edd-bac0-5a352913974d"/>
...
```

Using this unique assignment, it is now possible to synchronize the database schemas of read-only-connected databases ("No Schema Sync") via "External Synchronization" or "Content Transport" as follows:

- If a schema is changed in the source project and synchronized with other FirstSpirit projects using one of the two mechanisms, the names of database tables and database columns are transferred from the source project to the target projects.

- If a new database table or database column is created in the target project, the corresponding `dbname` of the source schema is now preferred.

When using "FirstSpirit External Synchronization" the mapping file (`Mapping.xml`) can be added to the export file via a new option in the interface ExportOperation.SchemaOptions (`setExportGidMapping=true`) as follows:

```
ExportOperation exportOperation = ... // get the export operation via agent;
ExportOperation.SchemaOptions schemaOptions = exportOperation.addSchema(schema);
schemaOptions.setExportGidMapping(true);
ExportOperation.Result result = exportOperation.perform(...);
```

The previous export format did not have to be extended for the new functionality. The **export format in version 3.2** can still be used.

> **!** The command line tool "FSDevTools" will only support the new option **from version 2.6 on**. This version is not yet available. By default, no mapping file is created when using "FSDevTools" (`setExportGidMapping=false`).

Without a mapping file (default behavior when using "FirstSpirit External Synchronization") the `dbnames` of the column names and tables are recalculated as before during synchronization. For the regular use cases there will be no deviation during recalculation, so that the use of a mapping file is not necessary.

When using the "FirstSpirit Content Transport" functionality, the mapping file (`Mapping.xml`) is automatically generated and used.

8 FirstSpirit Content Experience Tools (CXT)

8.1 "SameSite" attribute for cookies 2020-08

Since the beginning of this year, many browsers have been restricting third-party cookies ("Cross site") for security and data protection reasons: If cookies could previously be used in a third-party context by default, the attribute `SameSite=None` must now be set explicitly. Cookies without a `SameSite` attribute no longer allow third-party use.

`SameSite` is a cookie attribute that prevents the browser from automatically sending cookies with cross-site requests and thus offers protection against cross-site request forgery (CSRF). The attribute also allows you to define which cookies can be read in which context.

The current release responds to this request for the MicroApp Framework: The MicroApp Framework uses cookies for communication between MicroApps and FragmentCreator, for example. Without the attribute `SameSite=None`, problems can occur if the REST API is to be used (for example, for the integration of customer-specific MicroApps), but this is on a different domain than the one the user has just opened in the browser.

Now the attribute `SameSite` of the CSRF token can be set for theMicroApp Framework using the parameter

```
cxt.platform.csrf-samesite
```

in the *web.xml* file of the "FragmentCreator" module or in the *cxt-fragment-creator.properties* file.

By default, the value of this attribute is*Unset*, i.e.

```
cxt.platform.csrf-samesite=Unset
```

With this configuration, the attribute for the cross-site cookie is **not** set. This corresponds to the previous behavior and as well to the state if `cxt.platform.csrf-samesite` is not defined.

To ensure that the MicroApp Framework can be fully used even if the REST API is used, one of the following values must be set:

- *None*: If no value was previously specified, the browser automatically recognized this as a third-party cookie. This enabled users to be identified across different web pages, for example. The MicroApp Framework also benefits from this functionality in some places.
  In order for this behavior to continue to be possible (e.g. with Google Chrome from version 80), the attribute `SameSite=None` must now be explicitly set.
  Important: `SameSite=None` requires a "Secure" flag. This defines that a cookie is always sent over a secure HTTPS connection.

- *Lax*: This mode provides an appropriate balance between security and usability for websites: The cookie is only sent if the URL in the browser matches the URL stored in the cookie. However, cross-site requests are also possible if top-level navigation is performed and secure HTTP methods are used.

- *Strict*: The cookie is only sent if the URL in the browser matches the URL stored in the cookie. No login information is then passed on to other websites, for example, even if the user follows a link from another page or from an e-mail. This mode offers good protection against cross-site request forgery and is useful for websites with sensitive content, for example.

8.2   Latest module versions 2020-01

FirstSpirit 2020-01 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.24
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	1.32
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.20
`fragmentdap-[version].fsm`	
FirstSpirit Media DAP	1.20
`mediadap-[version].fsm`	
FirstSpirit Markdown Editor	1.17*
`markdown-editor-[version].fsm`	
FirstSpirit Tagging Editor	1.17
`tagging-editor-[version].fsm`	
FirstSpirit CXT FragmentCreator - CaaS Integration	1.15

```
caas-integration-[version].fsm
```

* The module "FirstSpirit Markdown Editor" requires from version 1.17 FirstSpirit 2019-12 or higher.

8.3   Latest module versions   2020-02

FirstSpirit 2020-02 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.25
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	2.0
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.21
`fragmentdap-[version].fsm`	
FirstSpirit Media DAP	1.21
`mediadap-[version].fsm`	
FirstSpirit Markdown Editor	1.18*
`markdown-editor-[version].fsm`	
FirstSpirit Tagging Editor	1.18
`tagging-editor-[version].fsm`	
FirstSpirit CXT FragmentCreator - CaaS Integration	1.17
`caas-integration-[version].fsm`	

* The module "FirstSpirit Markdown Editor" requires from version 1.17 FirstSpirit 2019-12 or higher.

8.4   Latest module versions   2020-03

FirstSpirit 2020-03 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.26
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	2.1
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.22

```
fragmentdap-[version].fsm
```

FirstSpirit Media DAP   1.22*

```
mediadap-[version].fsm
```

FirstSpirit Markdown Editor   1.19*

```
markdown-editor-[version].fsm
```

FirstSpirit Tagging Editor   1.19

```
tagging-editor-[version].fsm
```

FirstSpirit CXT FragmentCreator - CaaS   1.18
Integration

```
caas-integration-[version].fsm
```

* These modules require FirstSpirit 2020-03 or higher from this version.

8.5   Latest module versions   2020-04

FirstSpirit 2020-04 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.27
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	2.2
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.23
`fragmentdap-[version].fsm`	
FirstSpirit Media DAP	1.23*
`mediadap-[version].fsm`	
FirstSpirit Markdown Editor	1.20*
`markdown-editor-[version].fsm`	
FirstSpirit Tagging Editor	1.20
`tagging-editor-[version].fsm`	
FirstSpirit CXT FragmentCreator - CaaS Integration	1.19
`caas-integration-[version].fsm`	

* These modules require FirstSpirit 2020-03 or higher from this version.

8.6   Latest module versions 2020-05

FirstSpirit 2020-05 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.28
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	2.4
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.25
`fragmentdap-[version].fsm`	
FirstSpirit Media DAP	1.23*
`mediadap-[version].fsm`	
FirstSpirit Markdown Editor	1.21*
`markdown-editor-[version].fsm`	
FirstSpirit Tagging Editor	1.21
`tagging-editor-[version].fsm`	
FirstSpirit CXT FragmentCreator - CaaS Integration	1.21
`caas-integration-[version].fsm`	

* These modules require FirstSpirit 2020-03 or higher from this version.

8.7   Latest module versions 2020-06

FirstSpirit 2020-06 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.29
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	2.6
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.26
`fragmentdap-[version].fsm`	
FirstSpirit Media DAP	1.23*
`mediadap-[version].fsm`	
FirstSpirit PageRef DAP	1.0

```
pagerefdap-[version].fsm
```

FirstSpirit Markdown Editor	1.21*

```
markdown-editor-[version].fsm
```

FirstSpirit Tagging Editor	1.21

```
tagging-editor-[version].fsm
```

FirstSpirit CXT FragmentCreator - CaaS Integration	1.21

```
caas-integration-[version].fsm
```

* These modules require FirstSpirit 2020-03 or higher from this version.

8.8   Latest module versions   2020-07

FirstSpirit 2020-07 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.30

```
dataservice-[version].fsm
```

FirstSpirit CXT FragmentCreator	2.8

```
fragment-creator-[version].fsm
```

FirstSpirit Fragment DAP	1.27

```
fragmentdap-[version].fsm
```

FirstSpirit Media DAP	1.24*

```
mediadap-[version].fsm
```

FirstSpirit PageRef DAP	1.1

```
pagerefdap-[version].fsm
```

FirstSpirit Markdown Editor	1.22*

```
markdown-editor-[version].fsm
```

FirstSpirit Tagging Editor	1.22

```
tagging-editor-[version].fsm
```

FirstSpirit CXT FragmentCreator - CaaS Integration	1.23

```
caas-integration-[version].fsm
```

* These modules require FirstSpirit 2020-03 or higher from this version.

8.9 Latest module versions 2020-08

FirstSpirit 2020-08 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.31
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	2.9
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.27
`fragmentdap-[version].fsm`	
FirstSpirit Media DAP	1.24*
`mediadap-[version].fsm`	
FirstSpirit PageRef DAP	1.1
`pagerefdap-[version].fsm`	
FirstSpirit Markdown Editor	1.22*
`markdown-editor-[version].fsm`	
FirstSpirit Tagging Editor	1.22
`tagging-editor-[version].fsm`	
FirstSpirit CXT FragmentCreator - CaaS Integration	1.23
`caas-integration-[version].fsm`	

* These modules require FirstSpirit 2020-03 or higher from this version.

8.10 Latest module versions 2020-09

FirstSpirit 2020-09 supports the following module versions for "FirstSpirit Content Experience Tools":

Module / file name	Version number
FirstSpirit CXT DAP Bridge	1.32
`dataservice-[version].fsm`	
FirstSpirit CXT FragmentCreator	2.10
`fragment-creator-[version].fsm`	
FirstSpirit Fragment DAP	1.28
`fragmentdap-[version].fsm`	
FirstSpirit Media DAP	1.25*

```
mediadap-[version].fsm
```
FirstSpirit PageRef DAP   1.2
```
pagerefdap-[version].fsm
```
FirstSpirit Markdown Editor   1.23*
```
markdown-editor-[version].fsm
```
FirstSpirit Tagging Editor   1.23
```
tagging-editor-[version].fsm
```
FirstSpirit CXT FragmentCreator - CaaS   1.24
Integration
```
caas-integration-[version].fsm
```

* These modules require FirstSpirit 2020-03 or higher from this version.

8.11   Labeling input elements in FragmentCreator ("display language")

2020-06

FirstSpirit provides consistent support for the multi-language concept:

- Via the start page, it is possible to select the language for the labeling on the menu bar, dialogs, etc. (**interface language**).
  The interface languages that are currently available are English, German, Spanish, French, Italian, Dutch, and Russian.

- However, the project developer uses the **editorial language** (also called the **display language**) to control the language for displaying input elements, tooltips, etc.

The lang attribute (in the LANGINFOS / LANGINFO tag) is now taken into account for the editorial language, e.g.:

```
<CMS_INPUT_TEXT name="headline" useLanguages="no">
    <LANGINFOS>
      <LANGINFO lang="*" label="Headline"/>
      <LANGINFO lang="DE" label="Überschrift"/>
    </LANGINFOS>
</CMS_INPUT_TEXT>
```

If the interface language selected is "German", the input element labeling (`label` attribute) also takes "German" into account ("DE" in the example above):
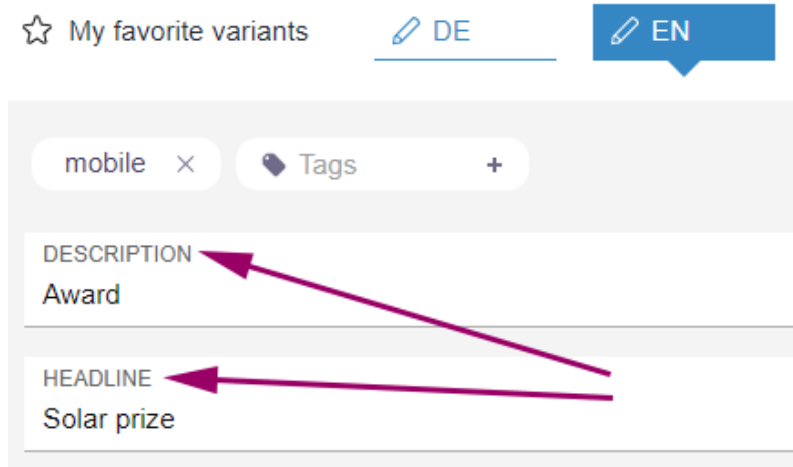
In the case of interface languages that are not included in the editorial languages available, the **master language** is used as a fallback option.

8.12 Extension of MicroApps API: Embedding of MicroApps into a page ("widget") 2020-05

Previously, MicroApps in MicroApp Framework could be called up as modal dialogs via the MicroApps API (`CxtMicroApps.getButtons(context)`).

In the current release, the MicroApps API has been expanded to include the method `CxtMicroApps.embedMicroApp`, which can be used to embed MicroApps directly into an HTML page ("widget").

The API also allows handlers to be added and used, and enables responses to events.

Note: The API is currently intended for internal use only and is being tested internally and by partners. In the future, it will also be made available to external customers.

8.13 Error analysis for MicroApp Framework 2020-04

An incorrect configuration is a common source of errors. It can result in the software not functioning as expected or not functioning at all.

The latest release makes it possible to set up an error page for the MicroApp Framework which presents the current configuration of the server clearly and lists error messages.

The error page is accessible via

```
/status
```

for administrators, appended to the URL of the FragmentCreator, e.g.

```
http://myServer:8080/fragments/status
```


Along with providing information about the server, it also runs a test on the configuration and lists the most recent HTTP requests and log messages.

In order to receive log messages, the `logging.file.name` parameter must be specified with the desired log file (e.g., `/tmp/fc.log`) in the `web.xml` file, e.g.

```
logging.file.name=/home/tomcat/logs/fragments.log
```

or alternatively in the properties file `/config/cxt-fragment-creator.properties`.

See also the manual for the MicroApp Framework, https://docs.e-spirit.com/odfs/edocs/cxt/installation-mo/installing-modu/fragmentcreator/index.html.

8.14   Fragment DAP: filter fragments in report by category 2020-05

Previously, the Fragment DAP `<SOURCE       name="FirstSpiritFragmentAccess/ FSFAConnector">` report allowed to filter for fragments based on a remote project filter and search terms. With this release, a new filter adds the option to limit the listed fragments based on their category.

The default filter settings are unrestricted, which allows to view all available fragments from every remote project.
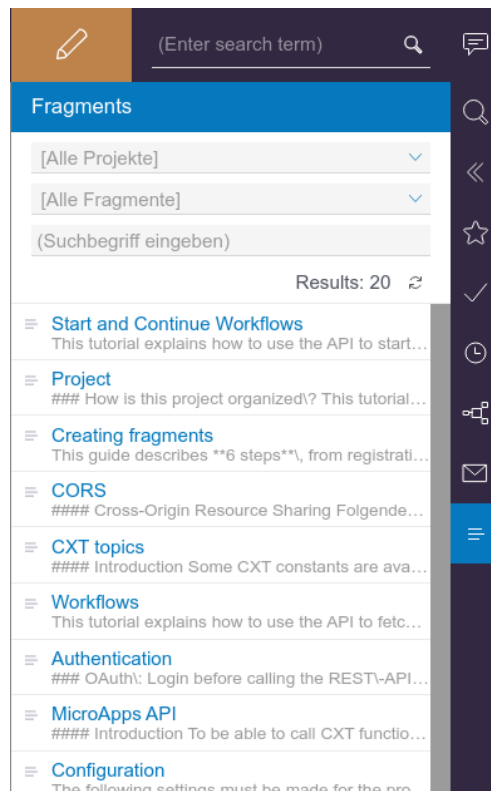
Fragment DAP report with no filters applied

In addition to the previous functionality that allowed searching for search terms or limiting the list to fragments from a specific project, the new drop-down menu for category filtering is added. It contains all categories of the remote projects and selecting any category will filter for fragments that belong to that category.

In case that the same category name is available in more than one project, selecting that category filter displays all fragments belonging to that category from any project. To filter the results, a specific project or search terms be applied, just as before.

8.15 FragmentCreator: Easier selection of other medium 2020-01

Media in an input component FS_INDEX with *DataAccessPlugin* component "FirstSpiritMediaAccess/Connector" can now be replaced more easily. The following icon was introduced for this purpose:
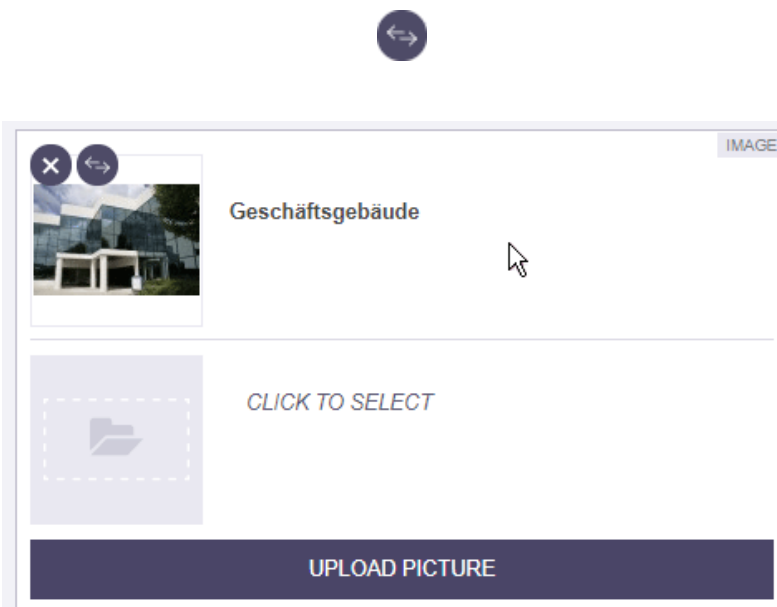
A click on it opens the selection dialog from which the desired medium can be selected.

8.16 FragmentCreator: Design revision 2020-02

As part of the revision and consolidation of the technical basis for FirstSpirit ContentCreator and FragmentCreator, the graphic interface of FragmentCreator was revised and adapted to the design principles of ContentCreator (see also the "Rollout of the new ContentCreator" chapter).
In principle, no functional changes for editors or developers are associated with the design update. Generally, all functions in the user interfaces are available in their familiar positions. The design update will not give rise to any additional development or operational work.

The design changes relate to the color and shape of the icons, fonts, and background colors:
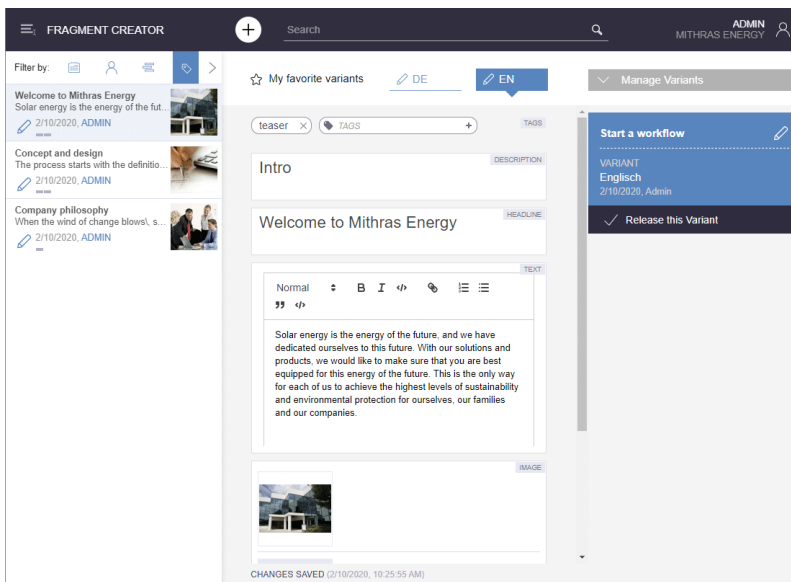
For better orientation, the currently selected fragment is now highlighted more clearly in the **fragment list** (left side of the screen). The current user is now displayed in the fragment list in blue font, all other users in grey font.

In the **variant overview** (right side of the screen, "Manage Variants") you can now select more than one variant with a click and start a workflow for them.

The **detailed information** that was previously displayed for the selected fragment or variant underneath the task area in the bottom right-hand corner is the same as the information in the status display. Therefore, it has now been removed to make the display clearer.

The user guidance when **uploading and selecting pictures** in an input element (button "Upload Picture") has also been improved in this context:
In principle, all pictures that are dragged into the upload dialog by drag-and-drop will now be referenced in the respective input element after the upload (visualized by check mark symbols). Previously, the pictures to be referenced in the input element had to be actively selected.
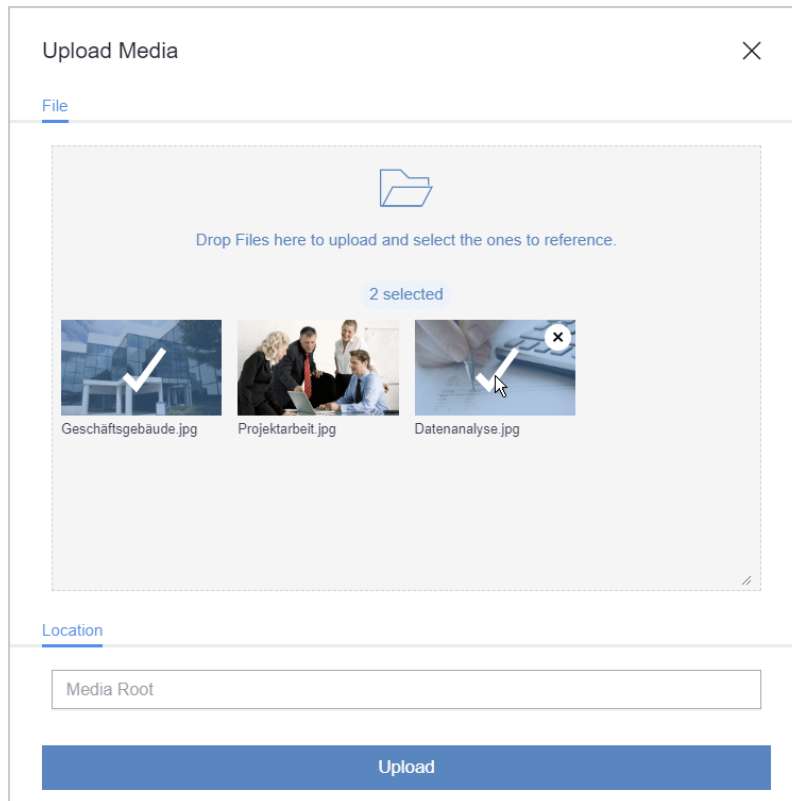
By clicking on the desired picture you can still select/unselect it. This allows you to control which pictures are to be referenced in the relevant input element.

After a successful upload (button "Upload") the upload dialog now closes automatically and no longer needs to be closed manually. Exception: If an error occurs, the dialog remains open, shows a corresponding error message and the affected pictures are not uploaded.

The adaptation of the documentation with regard to the design and the revised functionalities is successive, therefore the presentation in the documentation potentially no longer corresponds to the current software status.

8.17 FragmentCreator: Revision of the design (input fields) 2020-05

As part of the revision and merging of the technical basis for FirstSpirit ContentCreator and FragmentCreator, the graphic interface of FragmentCreator was revised and adapted to the design principles of ContentCreator.

The design update does not involve any functional changes for editors or developers. All functions in the interfaces are available in principle at the known positions. Neither on the development side nor on the operational side are there any costs associated with the design update.

The modifications of the current release primarily affect the input fields for text and selection as well as the tags (CMS_INPUT_TEXT, CMS_INPUT_CHECKBOX, CMS_INPUT_COMBOBOX, CMS_INPUT_RADIOBUTTON, FS_TAGGING and others).
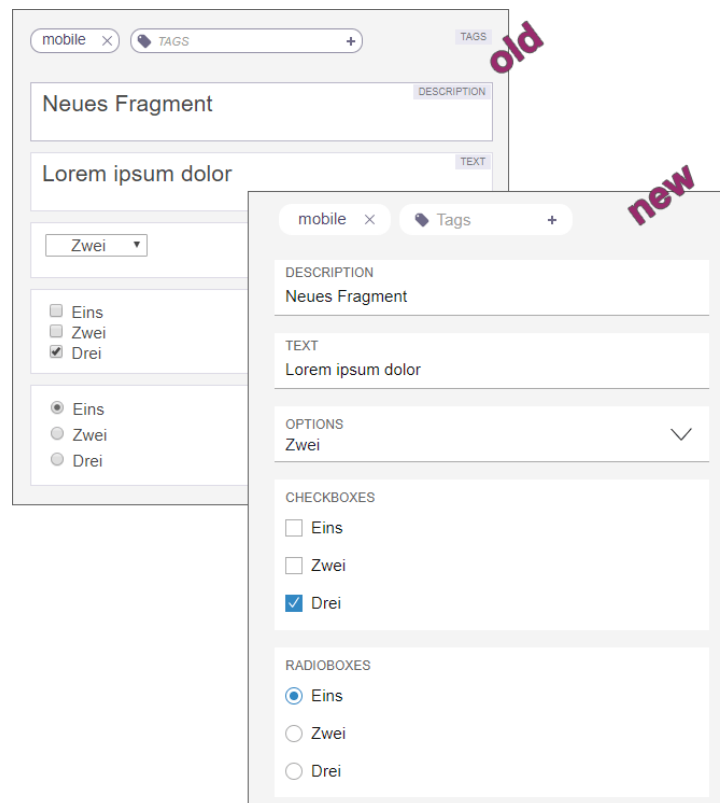
The adaptation of the documentation with regard to the design is successive, therefore the presentation in the documentation potentially no longer corresponds to the current software status.

8.18   FragmentCreator: Design revision (status bar) 2020-07

As part of the revision and consolidation of the technical basis for FirstSpirit ContentCreator and FragmentCreator, FragmentCreator's graphical interface has been revised and adapted in line with the design principles of ContentCreator.

The design update does not include any functional changes for editors or developers. All functions in the interfaces are generally available in their familiar positions. The design update will not give rise to any additional development or operational work.

The changes in the latest release concern the status bar along the bottom edge of the screen.

Now:
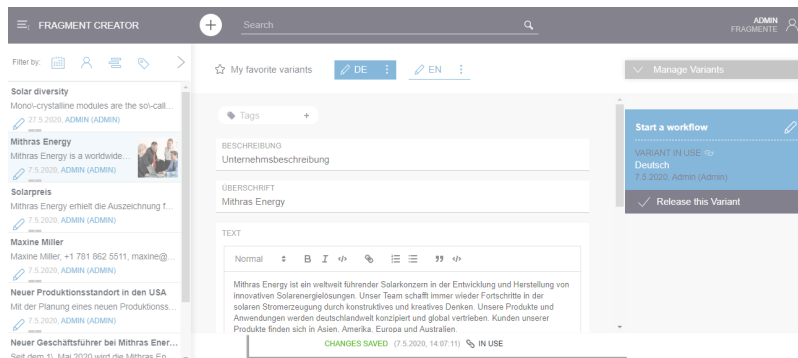
Previously:
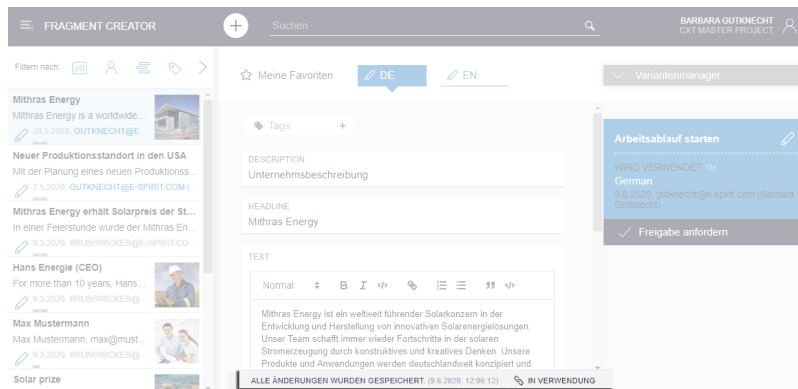
Adjustment of the manual regarding the design is carried out gradually. The presentation in the manual may therefore no longer correspond to the current software.

8.19   FragmentCreator: Improved handling for media upload 2020-08

The component "FS_INDEX" with module "Media DAP" is used for uploading and selecting media in FragmentCreator. As of the current release, a number of improvements have been made to the user interface:

Maximum file size

Up to now, an upload was limited to a file size of up to 10 MB. Error message in the browser console:
`CxtElement request firstspirit-media-upload-widget/uploadMedia failed: 500.`
As of the current release, the maximum file size allowed for an upload is **64 MB** by default.
If media that are larger than the configured values are dropped into the corresponding dialog, this is indicated in the dialog after the drop and the media in question are not uploaded into the project when the upload process is started using the "UPLOAD" button.

If necessary, this value can be adjusted using the following parameters in the *web.xml* file of the "FragmentCreator" module or in the *cxt-fragment-creator.properties* file:

- `spring.servlet.multipart.max-file-size`: Allowed file size for a medium uploaded via upload dialog.
  The value must not be greater than the value for `spring.servlet.multipart.max-request-size`.
  For possible size specifications, see class DataSize (Spring Framework).
  Default value: *64MB*

- `spring.servlet.multipart.max-request-size`: Each file is transferred in a separate HTTP request. Therefore, the value specified here must be at least as large as `spring.servlet.multipart.max-file-size`.
  For possible size specifications, see class DataSize (Spring Framework).
  Default value: *64MB*

Further configuration options regarding file size for media uploads are (still) available via the FirstSpirit ServerManager under "Project properties / Media restrictions". Settings for media sizes can also be specified there for the remote project into which media are to be uploaded using FragmentCreator, for example.
These can compete with those for the upload dialog of the FragmentCreator.

Upload via file selection dialog

Until now, media to be uploaded to the project were dragged and dropped into a corresponding dialog. As of the current release, the media to be uploaded can now also be selected via file selection dialog. The file selection dialog can be opened by clicking into the grey area.
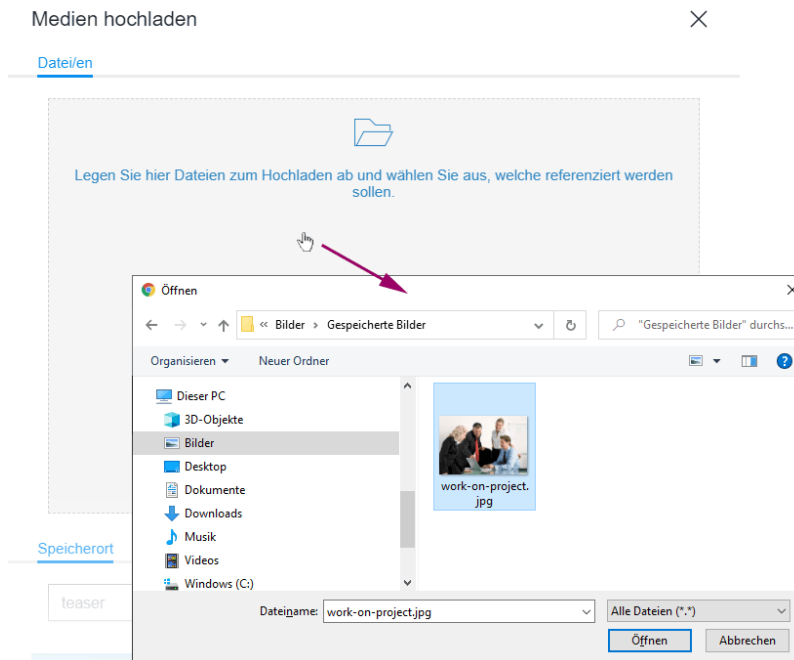
Upload from another browser window/tab

In addition to an upload from the workstation, images can now be dragged into the upload dialog from other browser windows/tabs.

To enable this, the server that delivers the image must allow cross-origin requests.

For more information please see

- for developers:
  https://docs.e-spirit.com/odfs/edocs/cxt/templates/access-media/index.html

- for editors:
  https://docs.e-spirit.com/odfs/edocs/cxt/fragment-creato/input-elements/index.html#upload

- for the file `web.xml`:
  https://docs.e-spirit.com/odfs/edocs/cxt/installation-mo/installing-modu/fragmentcreator/index.html

8.20   FS_INDEX now also supports the creation of fragments of several different categories (tag CATEGORY) `2020-04`

With the fragment DAP, fragments from one fragment project can be referenced in another fragment project, edited there and new ones can be created. The input component FS_INDEX with DataAccessPlugin component *FirstSpiritFragmentAccess/FSFAConnector* used for this.

The `CATEGORY` tag was previously used to restrict the selection to certain categories. If **only one** category was specified, a button was added to FS_INDEX, via which new fragments of the relevant category could be created.

As of the current release, the creation of fragments of different categories is now also supported if **several** categories are specified.
Example syntax:

```
<FS_INDEX name="fragments">
    <LANGINFOS>
      <LANGINFO lang="*" label="Fragments"/>
      <LANGINFO lang="DE" label="Fragmente"/>
    </LANGINFOS>
    <SOURCE name="FirstSpiritFragmentAccess/FSFAConnector">
      <PROJECT remote="fragmentproject"/>
        <CATEGORY uid="contact"/>
        <CATEGORY uid="teaser"/>
    </SOURCE>
</FS_INDEX>
```

The behavior regarding the `CATEGORY` tag is now as follows:

- **No** specification of `CATEGORY`: Only a selection of fragments from the project specified by the `PROJECT` tag is possible, no fragments can be created.

- Specification of **one** `CATEGORY` tag: Fragments of the specified category can be selected and created. Clicking on the button for creating new fragments opens the corresponding form.

- Specification of **more than one** `CATEGORY` tag: Fragments of the specified categories can be selected and created. Clicking on the button for creating new fragments opens a list of available categories:
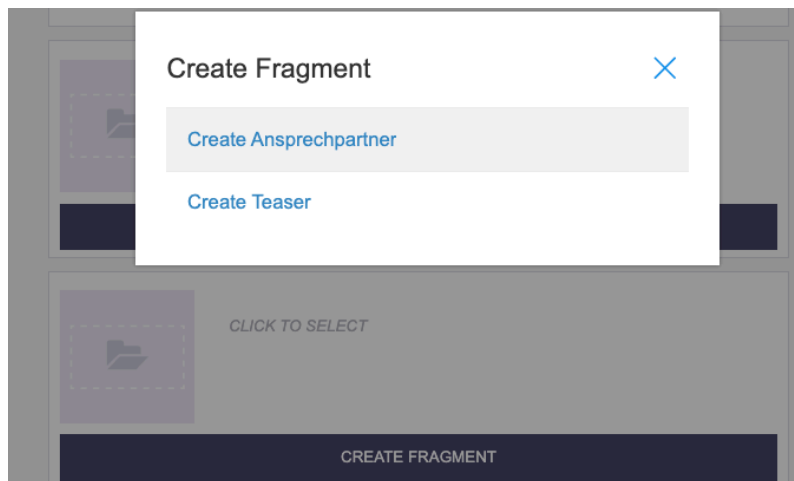
The referenced fragments can be edited, too.

> ❗ In projects in which FS_INDEX with DataAccessPlugin component *FirstSpiritFragmentAccess/FSFAConnector* and tag `CATEGORY` is already used, the configuration should be checked due to the new behavior: If more than one `CATEGORY` tag was specified, only a **selection** of fragments by editors was previously possible. Now editors can also **create** new fragments of the types specified by the `CATEGORY` tags.

8.21   Configuration of the Content-Security-Policy header 2020-04

Using the configuration property `cxt.platform.microapps.csp-origins` allows for modifying the `Content-Security-Policy` header. This might be necessary when access to MicroApps should be allowed from different domains.

The default value is `'self'`.

8.22   Markdown: Creating links using "DataAccessPlugins" 2020-03

Formatted texts are entered in the FragmentCreator using the input component *FS_MARKDOWN*. Example syntax:

```
<FS_MARKDOWN name="text" useLanguages="no">
    <LANGINFOS>
       <LANGINFO lang="*" label="Content"/>
       <LANGINFO lang="DE" label="Inhalt"/>
    </LANGINFOS>
</FS_MARKDOWN>
```

Up to now, when using the markdown editor "Quill" (see file `web.xml`, parameter *markdownEditorType*, vaue: *quill* or empty), it was possible to link text in addition to formatting, using the icon

As of the current release, further referencing options can be made available via this icon. To do this, a corresponding *DataAccessPlugin* (DAP) must be specified, via which the references are made available in the project. This *DataAccessPlugin* must implement the interface `UrlSupporting` (FirstSpirit Developer API, package `de.espirit.firstspirit.client.plugin.dataaccess.aspects`, also new with FirstSpirit 2020-03, see Chapter 5.2: "DAP: New aspect 'UrlSupporting' for URL generation").

Currently, for example, the "Media Data Access Plugin" can be connected, which is included in the standard scope of delivery of the MicroApp Framework / FirstSpirit Content Experience Tools. With this plugin, images and files of other formats can be referenced in the Markdown Editor.

Example syntax FS_MARKDOWN

```
<FS_MARKDOWN name="text" useLanguages="no">
    <DATALINKSOURCES>
      <DATALINKSOURCE name="files" mode="file"
          plugin="FirstSpiritMediaAccess/Connector">
       <LANGINFOS>
         <LANGINFO lang="*" label="Files"/>
         <LANGINFO lang="DE" label="Dateien"/>
      </LANGINFOS>
      </DATALINKSOURCE>
      <DATALINKSOURCE name="pictures" mode="picture"
          plugin="FirstSpiritMediaAccess/Connector">
        <LANGINFOS>
          <LANGINFO lang="*" label="Pictures"/>
          <LANGINFO lang="DE" label="Bilder"/>
        </LANGINFOS>
      </DATALINKSOURCE>
    </DATALINKSOURCES>
    <LANGINFOS>
      <LANGINFO lang="*" label="Content"/>
      <LANGINFO lang="DE" label="Inhalt"/>
    </LANGINFOS>
</FS_MARKDOWN>
```

Referencing in the Markdown Editor is enabled by specifying the tag

```
DATALINKSOURCES
```

Definition of the *DataAccessPlugin* to be used

Which *DataAccessPlugin* should be available for referencing in the respective Markdown component is determined by the mandatory tag

```
DATALINKSOURCE
```

The following mandatory attributes are required:

```
name
plugin
```

`name` is a component-wide unique identifier for the reference type. You can use this identifier to access the value stored for the reference type in the output / in JSON.

Via the attribute `plugin` the desired *DataAccessPlugin*, for the "Media Data Access Plugin" for example

```
plugin="FirstSpiritMediaAccess/Connector"
```

You can also specify several different *DataAccessPlugins*.

Use the `label` attribute to implement a language-dependent label for the *DataAccessPlugin* iin the list of referencing types to be selected in the markdown editor.

Configuration of the selection (via "Media Data Access Plugin")

Using the optional attribute

```
mode
```

you can restrict when using the "Media Data Access Plugin" (analogous to the definition in FS_INDEX with "Media Data Access Plugin") whether only images (`mode="picture"`) or only files of other formats (`mode="file"`) should be selectable. By default (without specifying `mode`) all element types can be selected as reference.

By default, when using the "Media Data Access Plugin", a selection of references from the **entire** media management of the **local** project can be made.

The selection can be limited to special folders by specifying the tags `PROJECT` and `FOLDER`. The reference name of the desired folder is specified via the *uid* attribute, for example

```
<PROJECT>
  <FOLDER uid="contact"/>
</PROJECT>
```

If media should be selected from a remote project, this can also be defined via the `PROJECT` tag. The mandatory parameter `remote` must be used to specify the symbolic project name of the remote project, for example

```
<PROJECT remote="media">
```

Example syntax:

```
...
<DATALINKSOURCES>
        <DATALINKSOURCE name="pictures" mode="picture"
            plugin="FirstSpiritMediaAccess/Connector">
          <LANGINFOS>
            <LANGINFO lang="*" label="Pictures"/>
            <LANGINFO lang="DE" label="Bilder"/>
          </LANGINFOS>
          <PROJECT>
            <FOLDER uid="contact"/>
          </PROJECT>
        </DATALINKSOURCE>
</DATALINKSOURCES>
...
```

Appearance in FragmentCreator
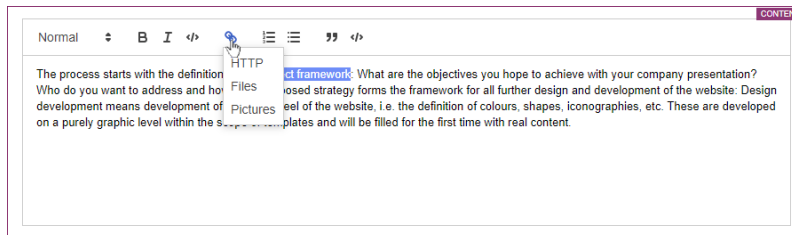
Inserting a link
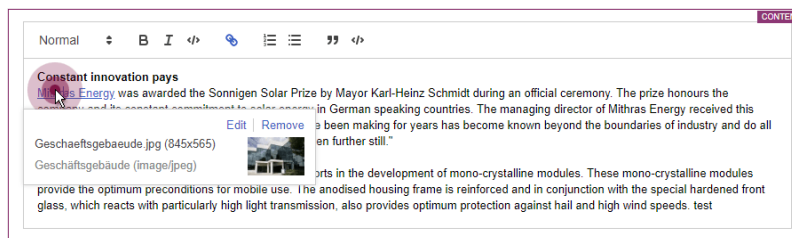
Inline view of an image referencing

Output / JSON

The JSON output of the Markdown Editor has been extended accordingly:

Attribute `dataLinks`: This attribute is included in JSON output whenever there is a reference in the Markdown component via *DataAccessPlugin*.

The link or reference is mapped using a special link schema that is used internally:

```
dap://{linkSource}/{objectId}
```

- `dap://`: URI schema/protocol

- `{linkSource}`: Identifier of the DAP configuration in the Markdown component (attribute `name` in the `DATALINKSOURCE` tag)

- `{objectId}`: Identifier of the DAP object selected by the editor

The values in curly brackets are URL-encoded. An example of a resolved version could look like this:

```
[link text](dap://pictures/id1234)
```

Where `[link text]` is the text in the Markdown component that is linked.

Attributes `identifier, url, value`: These attributes in the `dap` object come from the implementation of the *DataAccessPlugin* used:

- `identifier`: Identifier of the DAP object selected by the editor

- `url`: resolved URL of the DAP object selected by the editor

- `value`: JSON output that the *DataAccessPlugin* provides for the DAP object selected by the editor

Attributes `htmlExt, markdownExt, textExt`: If one of the DAP links contained in the Markdown Editor can be converted to an external URL due to the implementation of the *DataAccessPlugin* used, these attributes are created in addition to the attributes used for internal representation without the suffix `Ext`. In the case of media, an external URL can then be a URL under which the image can be viewed and/or downloaded, for example.

Example JSON output of a Markdown component:

```
{
   "html": "<p><a href=\"dap:\/\/linkSource\/id1234\">link text<\/a><\/p>\n",
   "markdown": "[link text](dap:\/\/linkSource\/id1234)",
   "text": "\"link text\" (dap:\/\/linkSource\/id1234)",
   "htmlExt": "<p><a href=\"#1234\">link text<\/a><\/p>\n",
   "markdownExt": "[link text](#1234)",
   "textExt": "\"link text\" (#1234)",
   "dataLinks": {
     "dap:\/\/linkSource\/id1234": {
       "identifier": "id1234",
       "url": "#1234",
       "value": {
         "store": "mediastore",
         "id": 1234
       }
     }
   }
}
```

Access in the template

The access to contents with resolved references can be done for example via the generated JSON:

```
$CMS_VALUE(json(#this).get("markdown").get("value").get("htmlExt"))$
```

> **!** For the markdown editor "SimpleMDE" no referencing via *DataAccessPlugin* is currently supported.

Further information

- Markdown Editor module

- Configuration of the input component FS_MARKDOWN

- General information about JSON output

8.23   Media DAP: Type-dependent upload of files   2020-04

Up to now only images could be uploaded in the FirstSpirit FragmentCreator via the FS_INDEX input component with Media DAP (`<SOURCE name="FirstSpiritMediaAccess/Connector">`). The current release now also supports the upload of files of other formats. In addition, the `mode` attribute in the `SOURCE` tag can be used to control which media type may be uploaded. As before, the filter also refers to the selection of media types:

- `PICTURE`: Only images can be uploaded and selected.

- `FILE`: Only non-image files can be uploaded and selected.

- `ANY` (default) All media (both images and files) can be uploaded and selected.

Example syntax:

```
<FS_INDEX name="files">
    <LANGINFOS>
      <LANGINFO lang="*" label="Files"/>
      <LANGINFO lang="DE" label="Dateien"/>
    </LANGINFOS>
    <SOURCE name="FirstSpiritMediaAccess/Connector" mode="file">
      <PROJECT remote="media"/>
    </SOURCE>
</FS_INDEX>
```

If media are dropped into the upload dialog that do not match the type specified by `mode`, a corresponding warning is displayed and these files are not uploaded to the project.

For further information see https://docs.e-spirit.com/odfs/edocs/cxt/templates/access-media/index.html.

8.24   Media DAP: Uploading directory structures   2020-09

To upload and select media in FragmentCreator, the "FS_INDEX" component is used with the "Media DAP" module. In the current release, entire directory structures (folders), including any images and files of different formats they contain, can be selected for upload instead of just files from the workstation PC.

To do this, the required folder(s) must be moved to the upload dialog using drag-and-drop.

Once the folders have been dropped, all media contained in the folders are displayed in the dialog. This enables the media that are to be referenced in the component to be selected.

Potential restrictions regarding file size and format that have been implemented by the project developer are taken into account directly. Only files which meet the size and format requirements are transferred to the dialog.

The folder path for the media to be uploaded is displayed underneath the media, along with the file name.
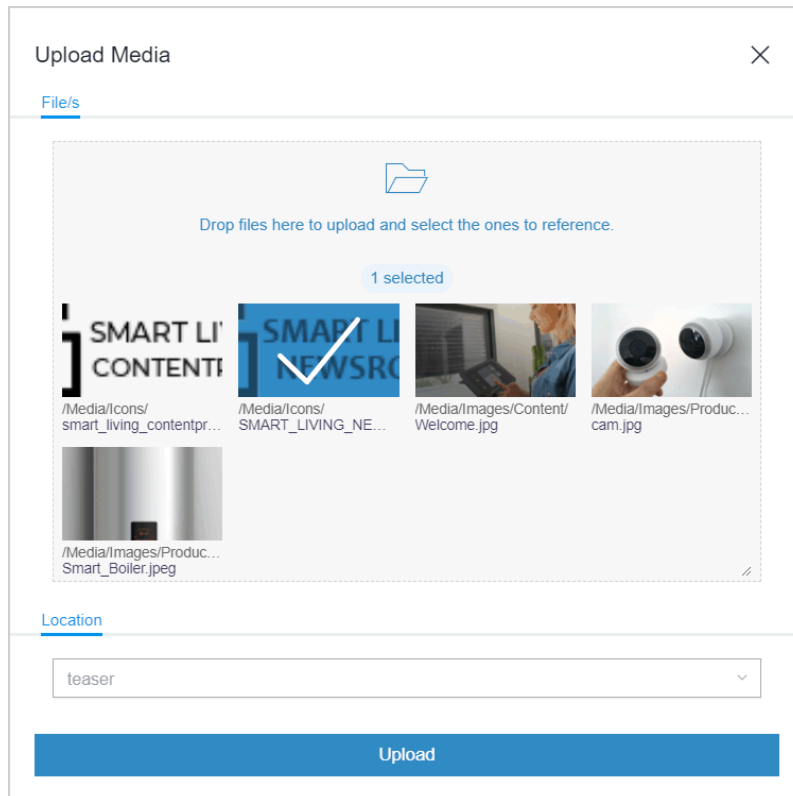
Once all the required media have been selected, the upload can be started by clicking "Upload". The upload dialog closes automatically once the media have been uploaded successfully. Exception: In the event of an error, the dialog stays open, displays a corresponding error message, and the media affected are not uploaded.

In the Media Store in the associated FirstSpirit project, the uploaded media and folders are mapped in the file structure of the workstation PC. Empty folders are not taken into account, however.

For both the media and the folders, the names of the folders and files are used by the workstation PC in each case.

8.25   MicroApps API: Providing variants with actions from MicroApps

2020-07

In FragmentCreator, favored variants can now be provided with actions from customer-specific MicroApps. The variant tabs are being expanded to include 3 dots that can be used to access editing options for MicroApps. For example:
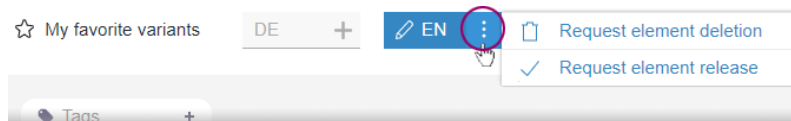
When an action is selected, it affects the selected variant.

By default, the standard workflows for release ("Request release" / *request_release*) and deletion ("Request deletion" / *delete_fragment_wf*) are started there.
See also https://docs.e-spirit.com/odfs/edocs/cxt/workflows/index.html.
Only actions for which the user has the appropriate permissions are shown.

To provide this UI interface with actions from **customer-specific** MicroApps, the MicroApp must support the new `VariantActionContext`. This contains

- the *projectId*
  ID of the project in question

- a *documentId*
  ID of the variant in question

- a *command*
  the required action, e.g., *'EditVariant'*

- the *application* (optional)
  Currently, the value *'fragmentcreator'* can be used here to specify that MicroApps are to be displayed at this point.

Example:

```
{ command: 'EditVariant', documentId: '4757cadc-d71e-4a58-9f3e-5cfbe86a9caa',
    projectId: 1, application: 'fragmentcreator'}
```

Note: The API is currently intended for internal use only and is being tested internally and by partners. In the future, it will also be made available to external customers.

8.26   MicroApps: Cancel fragment creation   2020-08

The MicroApp of the FragmentCreator "Create fragment" is used by the input component FS_INDEX with Fragment DAP and Tag `CATEGORY`, for example

```
<FS_INDEX name="contact" useLanguages="no">
    <LANGINFOS>
      <LANGINFO lang="*" label="Contact"/>
      <LANGINFO lang="DE" label="Ansprechpartner"/>
    </LANGINFOS>
    <SOURCE name="FirstSpiritFragmentAccess/FSFAConnector">
      <PROJECT remote="fragments">
        <CATEGORY uid="news"/>
      </PROJECT>
    </SOURCE>
</FS_INDEX>
```

(See also Documentation for "FirstSpirit Content Experience Tools (CXT)", Templates / Using fragments.)

The "CREATE" button opens a window with the input elements of the fragment to be created.
If no content is entered in the input elements and the dialog is closed using the X, an (empty) fragment has been created so far (folders and pages in the Page Store).
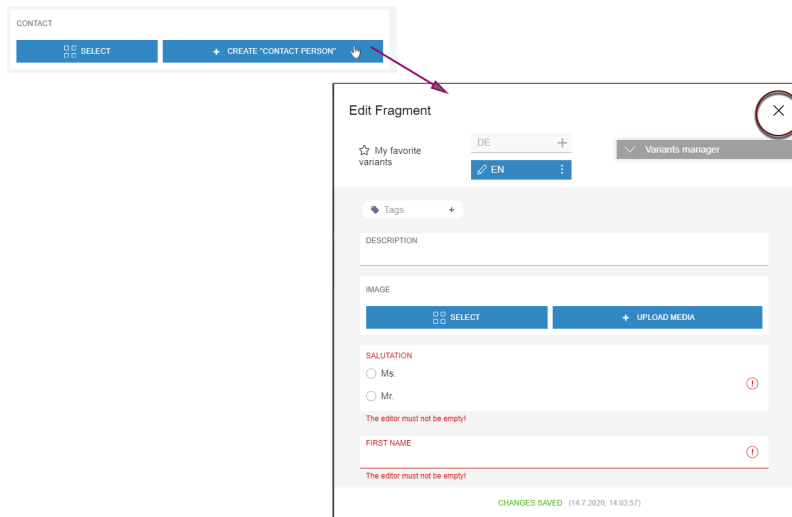
As of the current release, you can now choose whether the (empty) fragment should actually be created in the project or not.

With a click on the X the following dialog opens:
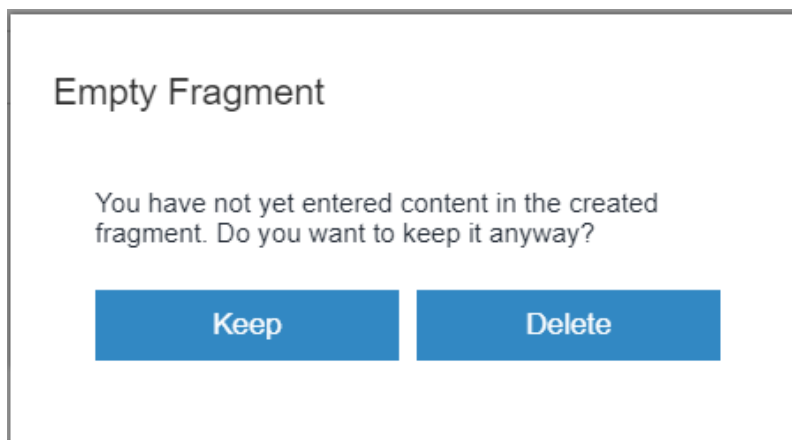
- If the user selects "Delete", no data is stored in the project.

- If the user selects "Keep", an empty fragment is created and stored in the project.
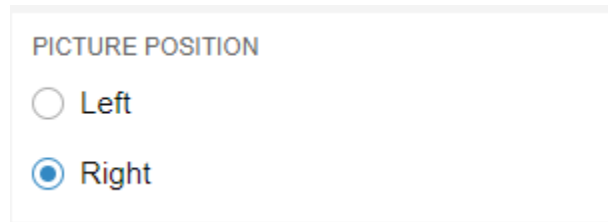
8.27   New input component for switching between two states: CMS_INPUT_TOGGLE 2020-09

In the current release, FragmentCreator also supports CMS_INPUT_TOGGLE, the input component for switching between two states.

Example:

```
<CMS_INPUT_TOGGLE name="picture_position" useLanguages="yes">
        <LANGINFOS>
          <LANGINFO lang="*" label="Picture position"/>
          <LANGINFO lang="DE" label="Ausrichtung des Bildes"/>
        </LANGINFOS>
        <OFF>
          <LANGINFO lang="*" label="Right"/>
          <LANGINFO lang="DE" label="Rechts"/>
        </OFF>
        <ON>
          <LANGINFO lang="*" label="Left"/>
          <LANGINFO lang="DE" label="Links"/>
        </ON>
</CMS_INPUT_TOGGLE>
```

PICTURE POSITION

◯ Left

◉ Right

CMS_INPUT_TOGGLE in FragmentCreator

The various states are mapped via the mandatory tags `<ON>` and `<OFF>`. A label must be defined for each state via the `<LANGINFO>` tag.

The radio buttons for the component are displayed one below the other with the `<ON>` state at the top and the `<OFF>` state at the bottom. This order cannot be changed.

Note: Unlike in FirstSpirit ContentCreator and SiteArchitect, only the "Radio button" type is supported in FragmentCreator (corresponds to `type="RADIO"`). The `type` parameter is not taken into account in FragmentCreator; the input component is always presented as a radio button.

For more information, see https://docs.e-spirit.com/odfs/edocs/cxt/templates/input-component/index.html.

8.28 New input component for the input of numerical values: CMS_INPUT_NUMBER 2020-08

In the current release, FragmentCreator also supports CMS_INPUT_NUMBER, the input component for the input of numerical values.

Example:

```
<CMS_INPUT_NUMBER name="number" useLanguages="no">
    <LANGINFOS>
      <LANGINFO lang="*" label="Number"/>
      <LANGINFO lang="DE" label="Zahl"/>
    </LANGINFOS>
</CMS_INPUT_NUMBER>
```

NUMBER

3.4

Integer numbers (including negatives) can be input in this input component by default. The exponential symbol `E` or. `e` is also possible, e.g., *1.7976931348623057e+308*.

In addition, the `type` attribute can be used to control whether it should be possible to input floating-point/decimal numbers. The `min` and `max` attributes can be used to specify a value range. Invalid inputs are not saved; they are rejected and a corresponding message is displayed.

User operation varies depending on the browser being used: Google Chrome and Microsoft Edge do not permit the input of letters in the component (with the exception of the exponential symbol `E` or `e`); Mozilla Firefox and Apple Safari, on the other hand, permit this. In Microsoft Edge and Google Chrome, how decimal separators (, or .) are displayed depends primarily on the operating system language and not (only) on the browser language, whereas Mozilla Firefox takes the browser language into account.

These differences are due to the implementation framework behind each individual browser and as such are out of the direct control of e-Spirit.

Moreover, the `format` attribute that can be used for FirstSpirit ContentCreator and SiteArchitect is not supported in FragmentCreator.

For more information, see https://docs.e-spirit.com/odfs/edocs/cxt/templates/input-component/index.html.

8.29   Optimizations for multi-user mode   2020-06

The editing area or MicroApps did not always show the current state of a variant after another user had made changes to the content, for example.
It is now only necessary to perform a manual update if another user is editing the content simultaneously. In this case, all the input elements of the variant ("form") are locked for editing and the following message is output:

```
The changes can not be saved - the element was modified on server.
```

Clicking on

```
CONFLICTING CHANGES! Click here to reload.
```

updates the display; the variant can now be edited again.

Note: When FS_INDEX is in use, referenced fragments are **not** automatically updated.

8.30   PageRef Access Plugin (PageRefDAP): Referencing pages   2020-06

As of the current release, references to pages can now be created using the new *PageRef Access Plugin* module (PageRef DAP).
The references to pages selected by the user can be output as HTML links.

Provision and Installation
The corresponding module file (file name extension: *.fsm*) can be requested from Technical Support:

```
pagerefdap-[version].fsm
```

The module is installed using the FirstSpirit ServerManager, in the "Server properties / Modules" dialog using the "Install" button.

Remote configuration

To enable references to pages from other FirstSpirit projects, the desired project must be configured as a "Remote project" in the project properties (FirstSpirit ServerManager):
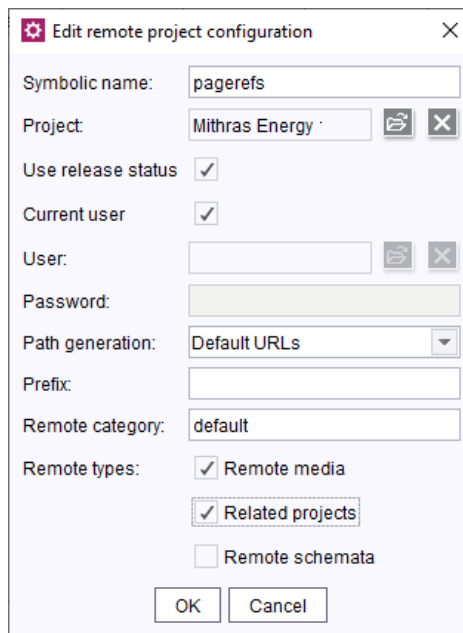
Web component "PageRef DAP for ContentCreator"

The plugin can be used in ContentCreator using the Web component "PageRef DAP for ContentCreator". To do this, the web component must be added to the relevant project in the "ContentCreator" area ("FirstSpirit ServerManager / Project properties / Web components").

Use in projects

If the module has been installed, it can be addressed from the component for selecting references, "FS_INDEX".

The editor can select one or more page references from the configured remote project(s).

Syntax example for FS_INDEX, tab "Form":

```
<FS_INDEX name="links">
    <LANGINFOS>
      <LANGINFO lang="*" label="Page referencing / links"/>
      <LANGINFO lang="DE" label="Seitenreferenzen / Links"/>
    </LANGINFOS>
    <SOURCE name="FirstSpiritPageRefAccess/PageRefConnector">
      <PROJECT remote="pagerefs">
        <FOLDER uid="products"/>
      </PROJECT>
    </SOURCE>
</FS_INDEX>
```

- `SOURCE`: This tag is used to specify the FirstSpirit module and the DataAccessPlugin component via the mandatory parameter `name`. In this case: "FirstSpiritPageRefAccess/ PageRefConnector".

- `PROJECT`: This tag is used to specify the remote project from which references are to be made available. The symbolic project name of the remote project must be specified via the mandatory parameter `remote`.

- `FOLDER`: The selection can be restricted to folders of the referenced project by using the `FOLDER` tag. If several folders are to be taken into account, a `FOLDER` tag must be specified for each folder. The name of the folder in question is entered using the `uid` parameter (reference name).

If the configuration was made correctly, the editor will be able to select page references:
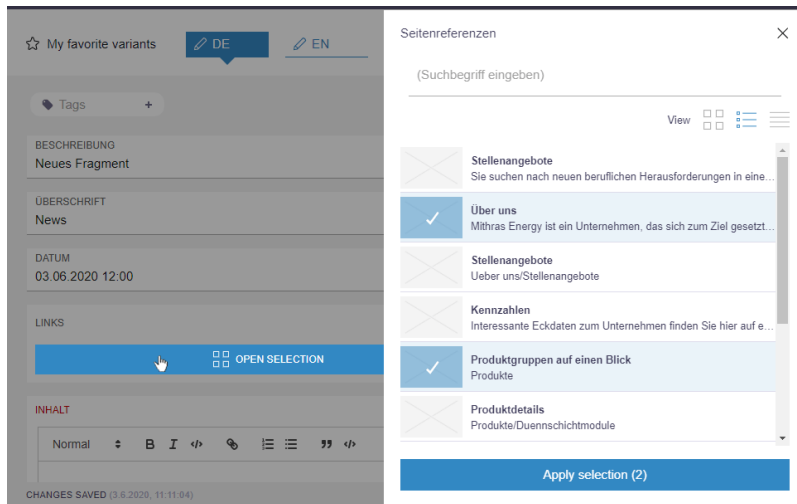
Selection
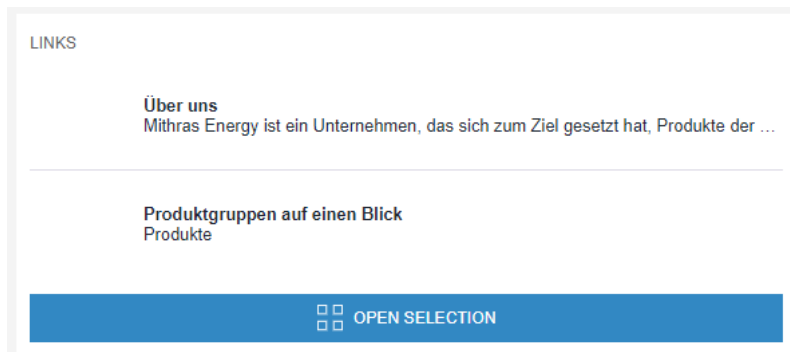
FS_INDEX with selected page references

Representation in ContentCreator:
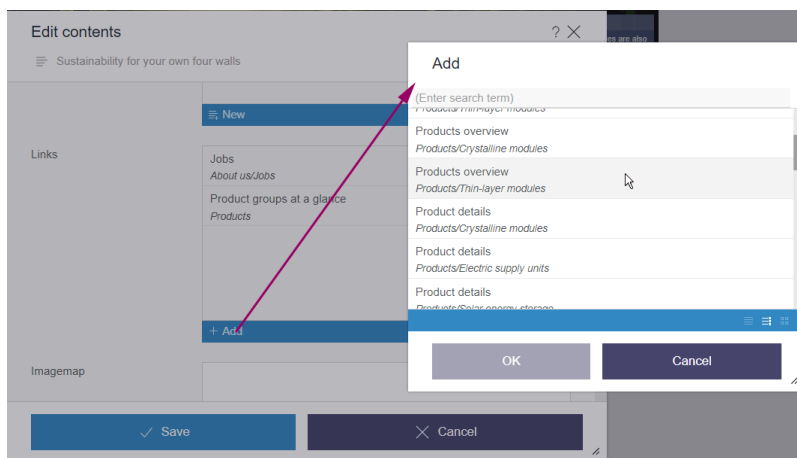
Output in FirstSpirit projects

The page references selected by the editor can be output in a FirstSpirit project as follows:

```
$CMS_FOR(pageref, links.values())$<br />
     <a href="$CMS_REF(pageref)$")$">
      $CMS_VALUE(pageref.uid)$  <br/>
     </a>
$CMS_END_FOR$
```

8.31   Status page shows registered MicroApps   2020-08

The status page that can be set up for the MicroApp Framework (under `~/status`) now lists all currently registered MicroApps with reachability, name and URL.

Note: To test accessibility, the status page sends an empty context to each MicroApp for verification. This can lead to warning messages like this one in the log:

```
de.espirit.cxt.microapps.stereotype.UnsupportedContextException: Context is not
    supported
```

See also the manual for the MicroApp Framework, https://docs.e-spirit.com/odfs/edocs/cxt/installation-mo/installing-modu/fragmentcreator/index.html.

9   Compatibility

9.1   New layer for Oracle Database 19c   2020-06

The available database layer for Oracle Database 11g/12c is, due to changes in database and driver, not compatible with Oracle Database 19c.

In order to establish FirstSpirit compatibility with Oracle Database 19c a new database layer (Oracle19cLayer) thus had to be implemented.

Generally, you can either simply use a new database layer, or you can update the existing database layer.

Use the new database layer:

In order to use the new database layer (Oracle19cLayer), you simply have to add it in the server properties (ServerManager) in the database section, and configure it according to your local requirements (see Server properties / Databases)

Update existing database layer:

In order to update an Oracle Database 12c used with FirstSpirit to Oracle Database 19c, you can,

- export a FirstSpirit project, and import it as a new project with Oracle Database 19c or

- change database layers by pausing the server, updating the database and manually switching database layers.

Migrating from Oracle Database 11g/12c to 19c without changing the database the new features of Oracle Database 19c can not be used, and we thus do not advise doing so!

Warning! When updating an Oracle Database 12c to 19c you should follow the vendor guidelines.

> **!** **Note:** Oracle 19c support has already undergone extensive testing, but currently still is in "EAP" status ("Early Access Program"). Not all use cases and migration scenarios could be tested. Thus it is important that you extensively check your use cases in a test environment!

The EAP phase will be accompanied through the FirstSpirit Community (community.e-spirit.com). This way questions and problems can be dealt with in a timely manner.
Extensive documentation regarding the use of Oracle Database 19c with FirstSpirit will also be published in the Community soon (initially in German only). The recommendations contained in this documentation for the configuration of the Oracle database and the FirstSpirit server should always be adhered to.

Final release of the new layer is planned for Q3/2020.

9.2 Postponed: Release of the new database layer Oracle Database 19c

2020-09

The release of the new database layer Oracle Database 19c, planned for 2020Q3, has been postponed and is now scheduled for 2020Q4.

Currently, the Oracle Database 19c layer is still in the EAP (Early Access Program) phase. The EAP phase will be accompanied through the FirstSpirit Community (https://community.e-spirit.com). Detailed documentation regarding the use of Oracle Database 19c with FirstSpirit is also published in the Community.

10 Module Development, Scripts, API

10.1 API extension for generating references (ReferenceConstructionAgent and ValueReferencesJournal) 2020-04

Essential FirstSpirit functions are based on a project's reference graph which is used to identify dependencies of objects within complex projects.

To generate references via the FirstSpirit Developer API, an agent of the type `ReferenceConstructionAgent` is required. A `ReferenceConstructionAgent` can then be used to generate instances of the type `ReferenceHolder`.

The `ReferenceConstructionAgent` (FirstSpirit Developer API, package `de.espirit.firstspirit.client.access.editor`) was expanded with the latest FirstSpirit release to include methods which enable references to be generated on the basis of meta information (via the unique identifier (UID) and `IDProvider.UidType` of a FirstSpirit object or via the global identifier (GID) for references to datasets). Previously, an instance of the type `IDProvider` was required in order to generate references.

New methods in the `ReferenceConstructionAgent`:

```
IDProviderReferenceHolder create(String uid, IDProvider.UidType uidType,...);
```

```
IDProviderReferenceHolder create(String schemaUid, String entityTypeName,...);
```

The `ValueReferencesJournal` interface (FirstSpirit Access API, package `de.espirit.firstspirit.client.plugin.dataaccess.aspects`) was also optimized at the same time. In the latest FirstSpirit release, methods that were specifically provided for adding defective references (broken references) are marked as "deprecated". Instead, the interface has been expanded with new methods which can be used to add defective as well as regular references (e.g., in a DataAccessProvider). The desired object is simply transferred when adding the reference – the system will check by itself whether the target object is accessible or not and evaluate the reference accordingly.

The new method in the `ValueReferencesJournal` interface

```
addReferenceToElement(final String uid, final IDProvider.UidType uidType,...);
```

replaces the previous method, for example:

```
addBrokenReferenceToElement(String uid, IDProvider.UidType uidType,...);
```

10.2   Accessing generated data via schedule script   2020-09

In a generation process, data is generated from FirstSpirit elements and is stored in the file system. After one or more generation processes of this nature, it is often necessary to carry out post-processing steps on the generated data. Examples include:

- replacing placeholder media with media from a MAM (controlled via metadata)

- training a search engine using metadata stored for the individual FirstSpirit elements

The new method `getOutput()` in the `GenerationAgent` interface (FirstSpirit Developer API, `de.espirit.firstspirit.scheduling.agency` package) can be used to allocate the generated files to the corresponding FirstSpirit objects after a generation process. This information can be used in a downstream schedule step in order to change the generated data or to forward the information about this data to third-party systems.

> **!** In order to use the functionality, a generation action must be executed first with the "Use ACL database" option activated.

Example code (Beanshell):

```
import de.espirit.firstspirit.scheduling.agency.GenerationAgent;
import de.espirit.firstspirit.agency.LanguageAgent;
import de.espirit.firstspirit.scheduling.agency.FileEntry;
import de.espirit.firstspirit.access.store.IDProvider;
generationAgent = context.requireSpecialist(GenerationAgent.TYPE);
ml = context.requireSpecialist(LanguageAgent.TYPE).getMasterLanguage();
Collection<FileEntry> output = generationAgent.getOutput();
for (entry : output) {
 uid = entry.getElementUid();
 IDProvider.UidType uidType = entry.getElementUidType();
 path = entry.getPath();
 element = entry.getElement(context, false);
 context.logInfo("+ Generated: " + uid + " (" + uidType + ") = " +
   element.getDisplayName(ml) + " at: " + path);
}
```

The method `getOutput()` outputs all data that is located in the generation directory when the script is executed (i.e., the data could potentially be deployed). Multiple previous generations of the schedule are taken into account automatically.

In addition to the output allocation, the `PermissionsAgent` (FirstSpirit Developer API, `de.espirit.firstspirit.access.editor` package) has been introduced in order to determine existing access and execution permissions for the individual FirstSpirit objects. These permissions are special user permissions which are specified by the template developer via a project-specific metadata template using the CMS_INPUT_PERMISSION input component and are then defined in the metadata tab of the FirstSpirit objects (they are not editorial permissions which are defined in FirstSpirit). The `PermissionsAgent` also takes into account the inheritance of these permissions, i.e., permissions which have been inherited from a higher-level node are also read out.

10.3   During delta generation, media changes can now be forwarded to datasets that reference the medium in question. 2020-09

The new rule `PROPAGATE_MEDIA_CHANGES_TO_CONTENTSTORE` in the `DeltaGeneration` interface (FirstSpirit Developer API, `de.espirit.firstspirit.access.schedule` package) can now be used to forward media changes to datasets.

When the rule is used, releasing a medium means that all datasets that reference this medium in the release state are regarded as "changed" for the delta generation.

10.4   DAP: New aspect "UrlSupporting" for URL generation   2020-03

You can use the input component FS_INDEX to connect external components or modules that are to make data available to the input component. The data is made available using "DataAccessPlugins". This data can then be selected (that is, referenced) using FS_INDEX. Function and appearance of "DataAccessPlugins" depends to a great extent on the individual implementation of the plug-in. The implementation of a "DataAccessPlugin" uses the packages `de.espirit.firstspirit.client.plugin.dataaccess` and `de.espirit.firstspirit.client.plugin.dataaccess.aspects` of the FirstSpirit Developer API and others.

With the current release the package `de.espirit.firstspirit.client.plugin.dataaccess.aspects` has been extended by the aspect `UrlSupporting`.

This can be used to determine a URL string for an object that originates from a "DataAccessPlugin". For this purpose, an `UrlGenerationContext` is passed (new interface with this name in the same package). Among other things, this context grants access to the URL creator (provider) to be used, the language and the status of the object (released or not released).

The JSON generation of the FS_INDEX input component and its entries was extended by an attribute `url` (if the DAP implements this aspect):

```
{
   "identifier": "...",
   "fsType": "...",
   "value": {...},
   "valueSchema": {...},
   "url": "..."
}
```

For a medium from the "Media Data Access Plugin" (Media DAP), the value could be

```
"/media/Solar_panel.png"
```

An exemplary use of the new aspect could look like this:

```
final UrlSupporting<T> urlSupporting = dapSession.getAspect(UrlSupporting.TYPE);
if (urlSupporting != null) {
  UrlGenerationContext context = UrlGenerationContext.builder().from
              (jsonOrGenerationContext).create();
  Optional<String> dapObjectUrl = urlSupporting.getUrl(context, dapObject);
}
```

Note: If an error occurs when generating the URL in the DAP (e.g. `java.lang.NullPointerException`), this is only recorded in the log as a warning and there will be no JSON output.

Example "Media Data Access Plugin" (Media DAP)

With FirstSpirit 2020-03, the new aspect has been incorporated into the "Media Data Access Plugin" (Media DAP) module, which is included in the standard scope of delivery of the MicroApp Framework / FirstSpirit Content Experience Tools. This plugin can be used to reference images and files of other formats in the Markdown Editor. See also Chapter 3.2: "Markdown: Creating links using 'DataAccessPlugins'".

10.5   The classes from the Apache PDFBox open-source library have been removed from fs-[isolated-]webrt.jar   `2020-07`

The classes from the "Apache PDFBox" open-source library have been removed from `fs[-isolated-]webrt.jar`.

If these classes were used during module development (which was only possible for modules that contain web applications in legacy mode), they are now missing and must be added explicitly to the web application. This means the "Apache PDFBox" library must be provided in the module as a "web-resource" by being added to the `module[-isolated].xml` file, e.g.:

```
<web-resources>
   <resource name="org.apache.pdfbox:pdfbox" version="2.0.17">
       lib/pdfbox-2.0.17.jar
   </resource>
</web-resources>
```

10.6   Introduction of a new Convenience.Method "isEmpty()"   2020-08

The new Convenience.Method `isEmpty()` has been introduced for `de.espirit.common.util.Listable`.

With

```
de.espirit.common.util
Interface Listable<T>
```

it is possible to use `.isEmpty()` to check directly whether the corresponding object is empty (return value `true`) or not (return value `false`).

10.7   Global module resources: Improved stability when resolving dependencies   2020-05

Module resources have a specific **scope**. This means that the corresponding jars and directories can either only be reached within the module (locally within the module) or server-wide (globally). **Global resources (scope="global")** are also available to other modules on the server. However, all classes are in a namespace and each class can therefore only exist once. It is for this reason that different versions of a class cannot be used.

Classloading and versioning of resources: If the same global resources are provided by several modules in different versions, only one of these global resources can be used by all modules. To prevent conflicts between different modules, the classes are – as far as possible – loaded via individual module Classloaders that are separated from each other. In the case of global resources, one of these classes is then made available via the combined, global Classloader.

If names are identical, an attempt is made to ascertain whether resources are **compatible** with one another. In order to do this, the unique identifier ("name") and the **version** of the resource supplied ("version") are required in the resource definition (`module.xml`) and it is necessary to specify the minimum compatible version ("minVersion") (optional) and the maximum compatible version ("maxVersion"), e.g.:

```
<web-resources>
 <resource name="..." version="4.4" minVersion="4.4" maxVersion="4.5.2">
    lib/httpclient-compatibility.jar
 </resource>
</web-resources>
```

Previous behavior

Previously, it was not possible to ensure that the latest version of a global resource was always used. Resources locally within a module were loaded via the relevant module Classloader and one of these resources was then made available via the combined global Classloader. Although this process ensured that the versions were compatible with the other modules (based on "name", "maxVersion", and "minVersion"), the latest version (i.e., the version with the highest version number) was not always used.

This behavior could cause problems when modules were updated; for example, because services that access these global resources no longer functioned with a different version.

New behavior

The latest FirstSpirit release ensures that in normal operation, the latest version of a global resource is always used (i.e., the version with the highest version number), even if these were previously loaded via various module Classloaders. If several modules provide the same version of a global resource, in this case, too, the global resource is always loaded from the same module. The compatibility of the versions with the other modules is still taken into account (based on "name", "maxVersion", and "minVersion").

The following applies: If several modules provide the same global resource, and if the versions are compatible with each other, the version with the highest version number is always used.

This increases the stability when it comes to resolving dependencies after a module update. There is now a greater probability that modules containing global resources which are used by different services will remain functional after a module update. In isolated cases, however, restarting the FirstSpirit Server may still be necessary.

Additional documentation:

- Module development (versioning resources)

- Module definition

■ Module update

10.8 JSON function: Improved support for FS_REFERENCE 2020-05

In the latest FirstSpirit release, the JSON function also supports referenced objects of the input component FS_REFERENCE.

No new version of the JSON output format will be published for this change; the changes will be made in the **current format version 1.1**.

Overview of changes in format version 1.1

The input component FS_REFERENCE is used to record any reference. The type of referenced object is not predetermined: References to media (images and files), page references, folders, sections on the current page, and many other objects can be defined.

Using the JSON function, the referenced FirstSpirit objects of the input component FS_REFERENCE can now be transferred to a JSON object structure (previously the referenced objects were not rendered). In this process, the object itself is not rendered; the function only renders the data that is required for referencing.

Example: An input component of type FS_REFERENCE refers, via the `sections="only"` mode, to the sections on the current page.

The output of the page reference (or the page) in format 1.1 therefore contains the new area "section": which holds data on the referenced sections (`"fsType": "Section"`)

```
"st_reference_pageref": {
   "fsType": "FS_REFERENCE",
   "name": "st_reference_pageref",
   "value": {
      "fsType": "PageRef",
      "name": "test",
      "identifier": "c88bc52c-aab9-4f66-870c-1354ca73fcb1",
      "uid": "test",
      "uidType": "SITESTORE_LEAF",
      "url": "http:\/\/www.myurl.de\/external_sync_DE\/test.json",
      "section": {
            "fsType": "Section",
            "name": "cms_input_text",
            "displayName": "CMS_INPUT_TEXT",
            "identifier": "c56cbc16-e70d-426d-af0f-16de05578922"
         }
      }
   }
}
```

Additional documentation:

- JSON configuration (ServerManager)

- JSON output (templates)

- FS_REFERENCE input component

Notes on the format version

The format version can be explicitly configured via the JSON settings:

```
$CMS_SET(#global.json.formatVersion, "1.1")$
```

If an unchanged JSON output is required outside the framework of FirstSpirit updates, the format version should be explicitly set in the project settings under Global Settings.

If the format version is not explicitly set, the latest JSON format version (1.1) will always be used.

10.9 New Aspect "FormReferenceContaining" (provide a form element's model reference) 2020-03

The (entered) values of FirstSpirit input components can contain references to other objects (so-called "value references").
An input component of type FS_INDEX (with DatasetDataAccessPlugin), for example, holds references to all datasets that are referenced via this input component.

Aside from these value references an input component can hold further references regarding the input component's GOM representation (FirstSpirit GUI Object Model (GOM)) (so-called "model references"). For an input component of type FS_INDEX (with DatasetDataAccessPlugin), for example, this is the reference to the desired table template (that is defined in the *TEMPLATE* tag under the template's form tab).

An input component developer can now provide the form element's (e.g. input component's) model references via the aspect `FormReferenceContaining`, with the method `List<Reference> collectFormReferences(F formElement)` (FirstSpirit Developer API, package `de.espirit.firstspirit.client.access.editor`).

The value references of a form element can be provided, with the aspect `ReferenceContaining<T>` (FirstSpirit Developer API, package `de.espirit.firstspirit.client.access.editor`), just like before.

10.10 Creating a pop-up dialog as a warning message 2020-05

Pop-up dialogs created using the interface `RequestOperation` (FirstSpirit Developer-API, Package `de.espirit.firstspirit.ui.operations`) can now also be created as warning messages via `Enum RequestOperation.Kind` (FirstSpirit Developer-API, Package `de.espirit.firstspirit.ui.operations`):

Example:

```
import de.espirit.firstspirit.agency.OperationAgent;
import de.espirit.firstspirit.ui.operations.RequestOperation;
import de.espirit.firstspirit.ui.operations.RequestOperation.Kind;
opAgent = context.requireSpecialist(OperationAgent.TYPE);
if (opAgent != null) {
      RequestOperation requestOp = opAgent.getOperation(RequestOperation.TYPE);
      requestOp.setKind(Kind.WARN);
      requestOp.setTitle("Whoa, there!");
      requestOp.perform("Once you delete this content, there's no getting it back.
        Make sure you want to do this!");
  } else {
   context.logInfo("...");
  }
```

Possible types (`RequestOperation.Kind`):

- WARN (new)

- INFO

- QUESTION

- ERROR

In SiteArchitect and ServerManager, WARNING-type pop-up dialogs are displayed with an exclamation mark, and INFO-type pop-up dialogs are displayed with an "i" symbol.

In ContentCreator, there is no visual difference between WARNING-type and INFO-type pop-up dialogs. Both types are displayed in exactly the same way with an exclamation mark.

10.11 Improved configuration options for user-specific URLs (AdvancedUrlCreator) 2020-05

FirstSpirit provides API interfaces and a reference implementation ("Advanced URL Creator") in order to integrate user-specific path strategies for URL generation in FirstSpirit. All configuration parameters (including the standard parameters evaluated by FirstSpirit as well as user-defined parameters) must either be defined via a schedule script that is executed before the actual generation schedule or via the configuration settings in the `module.xml` file.

The FirstSpirit framework evaluates some predefined standard parameters directly. These include:

- useWelcomeFileNames (for configuring start page references)

- stripWelcomeFileNames (removes the extension /index.* from the Advanced URL)

For more standard parameters, see the FirstSpirit manual: Standard configuration parameters.

The following applies to all parameters: The capitalization of parameters is not relevant, i.e., useWelcomeFileNames, USEWELCOMEFILENAMES, or usewelcomefilenames are all valid designations for the same parameter.

The parameters "useWelcomeFileNames" and "stripWelcomeFilenames", which are used for the configuration of user-specific path strategies for URL generation, offer new configuration options in the latest FirstSpirit release.

Change relating to the parameter "useWelcomeFileNames"

The "useWelcomeFileNames" parameter can be used to configure start page references.

```
context.setProperty("#urlCreatorSettings",
Collections.singletonMap("usewelcomefilenames", "true"));
```

Previous behavior:

Possible values:

- `true` or `yes` or value not set (default value)

- `false` or `no`

`true` or `yes` (default value): If the parameter is passed with the value "true" (default value), the file name `index.*` is always provided for page references which are marked as the start page of a folder in the Site Store when an Advanced URL is generated (regardless of the display name or the file name from the properties dialog).

In standard URL generation mode,

```
../en/startpage/mithras_home.html
```

becomes the following in advanced mode with "useWelcomeFileNames" (true):

```
../Startpage/index.html
```

For page references which are not marked as start pages, on the other hand, the display name continues to be used when the Advanced URL is generated.

`false` or `no`: If the parameter is passed with the value "false", regardless of whether a page reference is a start page or not, the URLs are generated based on the display name of the page reference (with the blank space being replaced by a "-" character):

```
../Startpage/Mithras-Homepage.html
```

New behavior:

Possible values:

- `true` or `yes` or value not set (default value): Only the first HTML template set uses Welcomefilenames.

- `false` or `no`: No Welcomefilenames are used.

- `all`: All HTML template sets use Welcomefilenames. (This type of configuration can lead to identical URLs, see (*)).

- Comma-separated list of template sets: All listed channels use Welcomefilenames. (This type of configuration can lead to identical URLs, see (*)).

`true` or `yes`: If the parameter is passed with the value "true" (default value), the file name `index.*` is provided for page references which are marked as the start page of a folder in the Site Store when an Advanced URL is generated (regardless of the display name or the file name from the properties dialog) – but this only applies to the first HTML template set (e.g., "html"), whereas for all other template sets (e.g., "php") the URLs are still generated based on the display name for the page reference (with the blank space being replaced by a "-" character).

```
../Startpage/index.html
```

`false` or `no`: If the parameter is passed with the value "false", regardless of whether a page reference is a start page or not, the URLs are generated based on the display name of the page reference (with the blank space being replaced by a "-" character):

```
../Startpage/Mithras-Homepage.html
```

`all`: If the parameter is passed with the value "all", the file name `index.*` is provided for page references which are marked as the start page of a folder in the Site Store when an Advanced URL is generated (regardless of the display name or the file name from the properties dialog) – for all HTML template sets contained in the project. (This type of configuration can lead to identical URLs, see (*)).

List of template sets: If a list of template sets is passed to the parameter, the file name `index.*` is provided for page references which are marked as the start page of a folder in the Site Store when an Advanced URL is generated (regardless of the display name or the file name from the properties dialog) – for all HTML template sets that are included in the list. The name of the template set is given here ("template set name": see FirstSpirit product manual under Template sets for a project). (This type of configuration can lead to identical URLs, see (*)).

> **!** (*) With a configuration that uses Welcomefilenames for all or multiple template sets, a folder can contain several index.* files (e.g., "/index.html" and "/index.php"). If the /index.* extensions are then also removed via stripWelcomeFileNames, this will result in identical URLs. It is strongly advised not to use this type of configuration.

Change relating to the parameter "stripWelcomeFileNames"

The "stripWelcomeFileNames" parameter is only relevant if the URL path strategy used also uses the "useWelcomeFileNames" configuration parameter.

The "stripWelcomeFileNames" parameter can be used to remove the "/index*" extension added by "useWelcomeFileNames" from the Advanced URL (but not from the file name under which the page is stored in the file system).

```
factorySettings = new HashMap();
factorySettings.put("usewelcomefilenames", "true");
factorySettings.put("stripwelcomefilenames", "true");
context.setProperty("#urlCreatorSettings", factorySettings);
```

Previous behavior:

Possible values:

- `true` or `yes` or value not set (default value)

- `false` or `no`

If the parameters "useWelcomeFileNames" and "stripWelcomeFileNames" are passed with the value "true" (default value), the display name of the start page is removed from the Advanced URL for page references which are marked as the start page of a folder in the Site Store.

In standard URL generation mode,

```
../en/startpage/mithras_home.html
```

becomes the following (in the file system) in advanced mode with "useWelcomeFileNames" and "stripWelcomeFileNames":

```
../Startpage/index.html
```

and the Advanced URL becomes:

```
../Startpage
```

As a result, although the start page of the "Startpage" folder is created in the file system with the extension "/index.*", it can (if the web server has been configured accordingly) be called via the "/Startpage/" URL.

If the project contains more than one template set, "stripWelcomeFileNames" only removes "index.{master extension}". "Master extension" is the file extension from the first template set (usually "html").

If the "stripWelcomeFileNames" parameter is passed with the value "false", the extension "/index.*" is retained for all start page references both in the file system and in the Advanced URL.

New behavior:

Possible values:

- `true` or `yes` or value not set (default value): "/index.*" (usually "/index.html") is shortened.

- `false` or `no`: URL is not shortened

- List of extensions: All listed extensions (e.g., "/index.html" and "/index.php") are shortened.

`true` or `yes`: If the "stripWelcomeFileNames" parameter is passed with the value "true" (default value), the start page of the "Startpage" folder is created in the file system with the extension "/index.*", but can (if the web server has been configured accordingly) be called via the "/Startpage/" URL.

`false` or `no`: If the "stripWelcomeFileNames" parameter is passed with the value "false", the extension "/index.*" is retained for all start page references both in the file system and in the Advanced URL.

List of extensions: If a list of extensions is passed to the parameter, the extension "/index.*" is shortened for page references which are marked as the start page of a folder in the Site Store when an Advanced URL is generated and, with the right web server configuration, can be called via the relevant URL (in a similar way to the behavior with the value "true"). The names of the extensions are given here ("Target file extension:" see FirstSpirit product manual under Template sets for a project).

10.12 Determining and returning previous store elements 2020-08

With the new method `getPreviousSibling()` in the Interface `StoreElement` (package `de.espirit.firstspirit.access.store`, FirstSpirit Access API) the previous StoreElement can now be determined and returned starting from a StoreElement.

11 Repository

11.1 Improved archiving function 2020-02

Pictures/media can exist in various resolutions in the project. If necessary, the system automatically creates image data in the correct resolution, when a medium is being requested in a certain resolution for the first time. Such image data (for one resolution) is saved on the server in an image cache (MEDIA_STORE_CACHED_PICTURES).

This method speeds up generation times, as generating large numbers of media in many different resolutions is very time-consuming. However, the image cache increases the project's data volume. Up to now, it has only been possible to remove automatically calculated resolutions from the project's data inventory again via a project export (by deactivating the "Export automatically calculated image resolutions" option).

This method has been improved in the latest FirstSpirit release. Using project archiving (the "Archive old project states" schedule), all calculated media/pictures whose resolutions are no longer available in the project can now be removed from the image cache. These resolutions were previously removed via "Project properties" – "Resolutions" – "Delete".

To use project archiving, the "Content, media, and data content" option must be activated under "Objects":
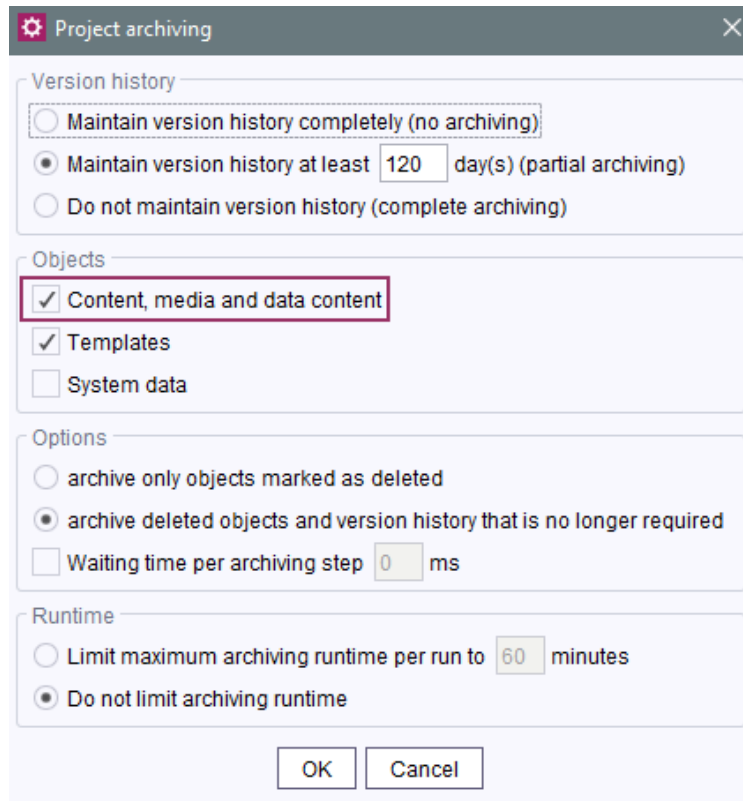
ServerManager – Project-based schedule entries – Project archiving

During archiving, the automatically calculated resolutions that are no longer required are subsequently removed from the project and are no longer carried over to the archive file.

The automatically calculated resolutions that are no longer required are removed regardless of the additional settings in place for the project archiving process (e.g., for "Version history" and "Options").

Advantages:

- The project size is reduced.

- Loading times are reduced.

- The FirstSpirit Server's overall performance is improved.

For more documentation, see:

- Archive old project states

12 Security

12.1 Apache Tomcat AJP: Vulnerability (Ghostcat) 2020-03

A Chinese security company has discovered a vulnerability in the binary communication protocol Apache JServ Protocol (AJP) used by Tomcat. This vulnerability has been named "Ghostcat" and is referenced as CVE-2020-1938.

The following Tomcat version series, which have been recommended by e-Spirit to date, are also demonstrably affected:

- Apache Tomcat 8.x before version 8.5.51 and

- Apache Tomcat 9.x before version 9.0.31

> **!** The vulnerability is classified as critical. e-Spirit therefore recommends a prompt update to a secure Tomcat version (8.5.51 or 9.0.31 see below).
> Furthermore, we recommend checking the AJP Connector configuration in each installation and adjusting it if necessary, as the AJP Connector was activated by default in the `server.xml` configuration file before the above-mentioned versions.

In the meantime, Apache Tomcat versions 8.5.51 and 9.0.31 have been released, which are considered secure and fix the underlying configuration problem in the default configuration.

With these versions, the default behavior of the AJP Connector has changed:

- By default, the AJP Connector only listens to the loopback address (previously all).

- Additionally, a secret must now be specified for the connection. (If required, the secret obligation can also be altered) by the parameter `secretRequired`.

Please consult the Apache Tomcat migration guide for the necessary adjustments:

- for version 8.5 or

- for version 9.0

Further information can also be found in the Ghostcat notes of the individual Linux distributions, e.g. for RHEL: access.redhat.com/solutions/4851251

13   SiteArchitect / ContentCreator

13.1   Workflow: Changing the order of transitions on an activity   2020-05

A workflow is a sequence of tasks that is completed according to a fixed, predefined structure. In FirstSpirit, this is made up of state, activities, and transitions. Transitions form the link between activities and states.

Project-specific workflows can be created using a graphical editor in FirstSpirit SiteArchitect ("Templates" / "Workflows"). Instances of these workflows can then be started in context on each element in a FirstSpirit project (SiteArchitect: context menu in the tree view / "Workflow" and in ContentCreator via the element state) or, if there is no context, using the FirstSpirit menu bar ("Tasks" / "Workflow" or in ContentCreator via "Actions"). Every instance of a workflow has to run according to the rules set in the workflow. For each activity, a dialog opens that offers the editor the option to advance the workflow by choosing a transition (via a button in the dialog).

In the latest FirstSpirit release, the order for displaying the transitions on an activity in a workflow can be configured.

The configuration is managed on the workflow in FirstSpirit SiteArchitect "(Templates" / "Workflows") in the "State diagram" tab. If a manual activity is selected in the graphical editor, the new property "Transitions" will appear in the Properties view (right) under "General". This will show all outgoing transitions for the selected activity. The transitions can be opened in a separate window via the

icon that appears after the property. The order of the transitions can be changed in this view using drag-and-drop or the buttons on the right-hand side of the window (Move up / Move down).
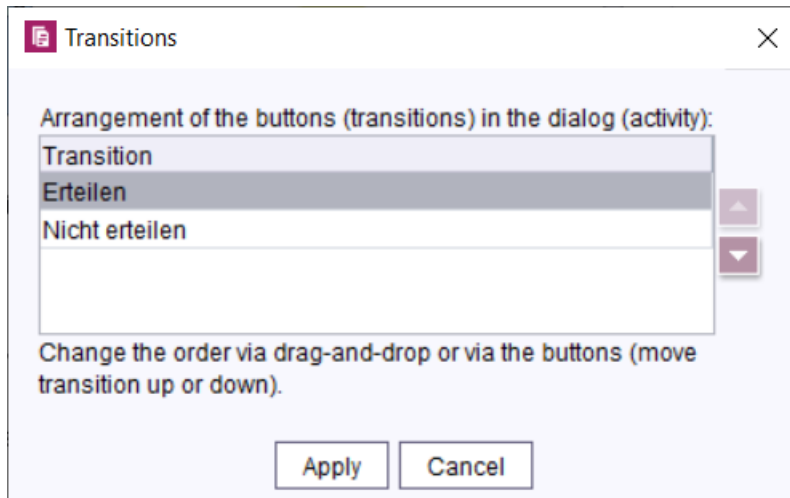
Changing the order of outgoing transitions

The changed order affects the display of buttons in the dialog.
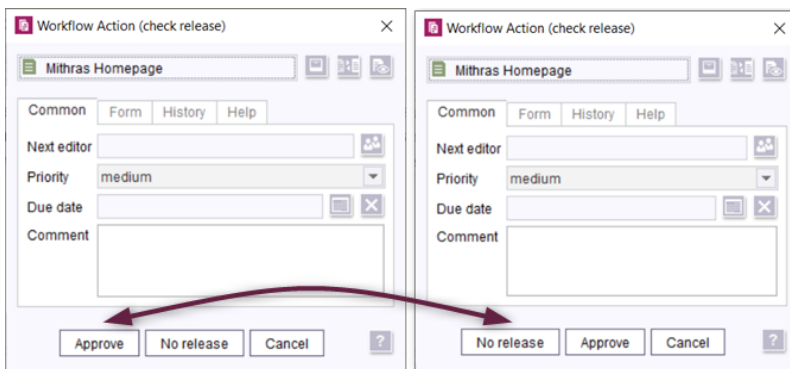
Changed order of transitions in dialog

Further information:

- Configuring workflows (templates)

- Workflows in SiteArchitect

- Workflows in ContentCreator

- Workflows in FragmentCreator

13.2   The Mithras Energy demo project is no longer available   2020-06

With the current FirstSpirit release the Mithras Energy demo project for ContentCreator and SiteArchitect is no longer available and will no longer be distributed with FirstSpirit (as announced with FirstSpirit 2020-02).

e-Spirit replaces the demo project with the contemporary **FirstSpirit Experience Accelerator** (see FirstSpirit Experience Accelerator documentation).
The FirstSpirit Experience Accelerator is a new, important building block in the FirstSpirit ecosystem. It contains reference implementations for important content modules (including Shoppable Image and Video, Touchpoint Editing, Teaser Carousel, and Banner) and can be used as a quick entry into a new FirstSpirit project. The objective of the FirstSpirit Experience Accelerator is a faster product launch in the enterprise and eCommerce environment.

13.3   New attributes for rules: ID, UID, GID, template   2020-07

The following properties are now available in the rules for corresponding elements:

- *ID*: determining the ID of a FirstSpirit element

- *UID*: determining the reference name/UID of a FirstSpirit element
  Sections do not technically have a UID, which is why the display name is returned in this case.

- *GID*: determining the global ID (GID) of a FirstSpirit element

- *TEMPLATE*: determining the template of the current FirstSpirit element
  The reference name of the template is returned (*UID*).

Code snippet:

```
<PROPERTY name="ID" source="#global"/>
```

```
<PROPERTY name="UID" source="#global"/>
```

```
<PROPERTY name="GID" source="#global"/>
```

```
<PROPERTY name="TEMPLATE" source="#global"/>
```

On a **page**, for example, the information for the page is returned (even if the rule is executed in a section within an FS_CATALOG input component, for instance).

Metadata returns information about the element on which the metadata has been defined.

14   SiteArchitect / ServerManager

14.1   Update of internally used software 2020-08

As of the current release, the following internally used software has been updated:

- **Java** (JRE with which the FirstSpirit Launcher is operated)
  The new versions are:
  - 11.0.8+10 64bit AdoptOpenJDK

  - 14.0.2+12 64bit AdoptOpenJDK

14.2   Database connection: support for the parameter
jdbc.MAXSTRINGLENGTH 2020-07

The database integration of the FirstSpirit Server is configured via the ServerManager (Server properties / Databases). All connected database layers on the server, i.e. all connections to a database from the server, are listed here. The optional JDBC parameter `jdbc.MAXSTRINGLENGTH` can be used in the database layer to define the maximum number of characters of a VARCHAR column when creating a new database table. If a higher value is specified for a string attribute than the one defined via parameter `jdbc.MAXSTRINGLENGTH`, this string attribute is stored as a different data type (e.g. as BLOB, CLOB or TEXT) in the database. If no value is set here, a default value is set depending on the database used.

New behavior:

With the current FirstSpirit release the parameter `jdbc.MAXSTRINGLENGTH` is supported for the following databases:

- PostgreSQL

- Oracle 19c

The following applies to PostgreSQL:

- Default value: If no value is defined using `jdbc.MAXSTRINGLENGTH`, the default value `255` is used.

- Maximum value: It is not recommended to define values higher than `65534`.

The following applies to Oracle 19c:

- Default value: If no value is defined using `jdbc.MAXSTRINGLENGTH`, the default value `2000` is used.

- Recommendation: Depending on the character set (parameter `jdbc.oracle.CHARACTERSET`), it is useful to adjust the character length. For Oracle 19c, for example, the character set `AL32UTF8` with character length `8191` is recommended. The recommended value for the respective character set can be found in the FirstSpirit Documentation (see "Values to be used for Oracle 19c (depending on the character set)").

- Maximum value: If values are specified for the parameter that are higher than the recommended value of the specified character set (parameter `jdbc.oracle.CHARACTERSET`), a corresponding message is logged in the server log. The maximum value for the respective character set can be found in the FirstSpirit Documentation (see "Values to be used for Oracle 19c (depending on the character set)").
  - For values that are higher than the maximum value of the specified character set, an ERROR is logged, and the value is reduced to the maximum value.

  - For values higher than the recommended value and smaller than the maximum value of the specified character set, a corresponding WARN message is logged, in which the recommended value is displayed.

For all other recommended and compatible database systems, the parameter `jdbc.MAXSTRINGLENGTH` is ignored. Instead, the hard-coded default values of the respective database are used.

I use the parameter `jdbc.MAXSTRINGLENGTH` in my existing layer configuration. Do I have to modify the configuration?

> **!** e-Spirit recommends that customers who use the `jdbc.MAXSTRINGLENGTH` parameter in an existing PostgreSQL database configuration remove the parameter from the configuration before updating to FirstSpirit 2020-07. This recommendation also applies if an existing Oracle database configuration (with the parameter `jdbc.MAXSTRINGLENGTH`) is converted to an Oracle 19c configuration.

Background: The parameter has been ignored so far. After an update to FirstSpirit 2020-07 the parameter is then used for existing PostgreSQL or Oracle 19c database configurations. Depending on the value that is set for the parameter in the configuration, the data types of the string attributes can change during the first sync to the database, and data can therefore possibly be lost.

More information:

- Server properties - Databases

- Mandatory parameters for the database connection

- Optional parameters (including jdbc.MAXSTRINGLENGTH)

14.3 External users can now be used for password configurations in ServerManager. 2020-09

In the current FirstSpirit release, external users can also be used for password configurations in ServerManager. These passwords can be used both for scripts and for remote connections.

The extension affects all passwords which were explicitly set in remote connections or script tasks in ServerManager.

- In the Schedule management area – in the "Script properties" dialog,
  see Documentation for administrators – FirstSpirit ServerManager / Schedule entry planning / Server-based actions / Execute script

■ In the Remote projects area – in the "Edit remote project configuration" dialog,
see Documentation for Administrators – FirstSpirit ServerManager / Project properties / Remote projects

Note: For security reasons, server administrators cannot be used for the password configuration.

To expand the users, the password format of the *PasswordAgent* has been changed. This does not affect existing data; the old format can continue to be read. A new authentication only takes place if the password is re-entered in the configuration dialogs. The old login implementation continues to be used with old persistent data.

Note: Forward compatibility is ensured. Once the FirstSpirit Server has been updated to the latest version, all set passwords continue to be valid. It is not necessary to reset the passwords.

Note: If the FirstSpirit Server is downgraded to FirstSpirit version 2020-08 or earlier, encrypted passwords are now no longer compatible via the *PasswortAgent* and must be reset.

14.4   "FirstSpirit Launcher JRE" module: update to Java versions 2020-06

The Java versions included in the "FirstSpirit Launcher JRE" module were updated in the latest FirstSpirit release:

In addition to the version supplied with the FirstSpirit Launcher, **AdoptOpenJDK 11.0.7+10.2 (default)**, the following Java variant is currently available:

■ AdoptOpenJDK 14.0.1+7.1

For documentation on the "FirstSpirit Launcher JRE" module, see the corresponding manual.

14.5   Certificate update FirstSpirit Launcher 2020-09

The FirstSpirit Launcher is digitally signed. Thanks to the signature, the customer knows where the application comes from and that it was not manipulated during the transfer (from the manufacturer to the recipient).

In the current FirstSpirit version, the certificate of the FirstSpirit Launcher is updated.

15 System

15.1 Updates to integrated third-party software 2020-03

The following internal software has been updated in the latest FirstSpirit release:

Java

The JRE, which is used as standard to run FirstSpirit SiteArchitect and ServerManager, was updated from **version 11.0.5** to **version 11.0.6**.
This is the Java version recommended by e-Spirit. When using the "FirstSpirit Launcher JRE" module, this is the "Default" configuration point.

The Java versions in the "FirstSpirit Launcher JRE" module were updated: AdoptOpenJDK 13.0.1 to version 13.0.2, AdoptOpenJDK 8u232 to version 8u242 ("outdated").

Install4j (used for FirstSpirit Launcher)

Updated from **version 8.0.2** to **version 8.0.4**.

ASM (library for analyzing and modifying Java byte code)

Updated from **version 7.2** to **version 7.3.1**.

Apache FOP (Formatting Objects Processor)

Via the module fs-fop.fsm (for the output of FirstSpirit content in PDF format).

Updated from **version 2.2** to **version 2.4**.

Pack200 removed (jar packing tool)

The Pack200 compression of jar files transferred from the server to the user's computer when the FirstSpirit desktop applications are launched was expanded in preparation for upcoming Java14 support. If the FirstSpirit desktop applications are still started via Java Web Start, the uncompressed jars are now transferred. (The Java Web Start function is still present in Oracle Java 8 but has been removed in Oracle Java 11. OpenJDK does not offer a comparable technology).

Java Service Wrapper (configuration and monitoring of FirstSpirit Java VM)

Updated from **version 3.5.39** to **version 3.5.42**.

Note: When the FirstSpirit server is updated (auto-update, in-place upgrade), the Java Service Wrapper files are not automatically updated. The Java Service Wrapper can be updated via the tar gzip update archive (file fs-update-5.2.*.tar.gz) as described in the FirstSpirit installation instructions.

Additional documentation:

- on Java Service Wrapper (manual for administrators)

- on Updating the Java Service Wrapper (installation instructions)

15.2   Update of internally used software   2020-04

As of the current release, the following internally used software has been updated:

Java (JRE which is used to operate the FirstSpirit Launcher)
The Java versions included in the "FirstSpirit Launcher JRE" module (see "FirstSpirit Launcher JRE" documentation) were updated in the latest FirstSpirit release. In addition to the version supplied with the FirstSpirit Launcher, AdoptOpenJDK 11.0.6 (default), the following Java variants are currently available:

- AdoptOpenJDK 14 (new)

- AdoptOpenJDK 8u242 (outdated)

AdoptOpenJDK 13.0.2 was removed from the FirstSpirit Launcher JRE module.

> **!** After updating the module, it may be necessary to restart the ServerManager to perform the appropriate configuration.

Spring Boot (used for MicroApp Framework)
Update from version 2.2.2 to version 2.2.5

Spring Cloud Hoxton (used for MicroApp Framework)
Update from version Hoxton.SR1 to version Hoxton.SR3

Eclipse Jetty (contained in the module `fs-jetty.fsm` / "FirstSpirit Jetty Service")
Update from version 9.4.23 to version 9.4.27

> **!** As before, use of Eclipse Jetty for production purposes is not recommended!

15.3 Updates to integrated third-party software 2020-05

The following internal software has been updated in the latest FirstSpirit release:

ASM (library for analyzing and modifying Java byte code)

Updated from **version 7.3.1** to **version 8.0.1**.

15.4 Update of internally used software 2020-06

As of the current release, the following internally used software has been updated:

Eclipse Jetty (contained in the module `fs-jetty.fsm` / "FirstSpirit Jetty Service")
Update from version 9.4.27 to version 9.4.28

> **!** As before, use of Eclipse Jetty for production purposes is not recommended!

Library com.fasterxml.jackson.core:jackson-databind
(contained in the DataAccessPlugin implementation example module for accessing web services using the standardized Open Data Protocol (OData))
Update from version 2.4.2 to version 2.11.0
For general information on the module see: https://docs.e-spirit.com/ (Examples)

15.5 Update of internally used software 2020-02

As of the current release, the following internally used software has been updated:

- **Spring Boot** (used for MicroApp Framework)
  Update from version 2.2.1 to version 2.2.2

15.6   Update of internally used software   2020-07

As of the current release, the following internally used software has been updated:

- **Spring Boot** (used for MicroApp Framework)
  Update from version 2.2.5 to version 2.3.1

- **Spring Cloud Hoxton** (used for MicroApp Framework)
  Update from version Hoxton.SR3 to version Hoxton.SR5

15.7   FirstSpirit Server: Release of new control files for Windows and Linux   2020-08

The current FirstSpirit release now includes the definitive release of the new control files for Windows and Linux operating systems. The use of these files with immediate effect is therefore recommended in live environments.

Java and the Java Service Wrapper are required to operate a FirstSpirit Server. The Java Service Wrapper and corresponding control files are provided by e-Spirit via:

- the installation archive (for new installations) or

- the update archive (for existing installations)

Both archives are available for download (please contact Technical Support for login details).

The new control files are included in both archives. They support a range of functions that were not covered by the old control files (including `install` under Linux). The functional scope of the new control files can be called up in the command line via the integrated help (Usage, Command syntax, Commands), e.g., using the `fs-server` call (under Linux):

```
test@fs_example:~$ fs-server
Usage: /home/fs_example/firstspirit/bin/fs-server [ console | start | stop | restart |
 condrestart | status | install | installstart | remove | dump ]
Commands:
 console      Launch in the current console.
 start        Start in the background as a daemon process.
 stop         Stop if running as a daemon or in another console.
 restart      Stop if running and then start.
 condrestart  Restart only if already running.
 status       Query the current status.
 install      Install to start automatically when system boots.
 installstart Install and start running as a daemon process.
 remove       Uninstall.
 dump         Request a Java thread dump if running.
```

Control files in the installation or update archive:

- New: fs-server (for Linux)

- New: fs-server.bat (for Windows)

- Old: fs5 (for Linux)

- Old: fs5.cmd (for Windows)

*The old control files are being **removed**. Support is scheduled to cease with FirstSpirit 2021-02.*

Requirement for using the new control files:

To use the new control files, the Tanuki Java Service Wrapper must be updated to version 3.5.42. This version of the wrapper is included in the latest installation and update archive.

Additional documentation:

- Update of the FirstSpirit backend

- Installation via installation archive

- Controlling the FirstSpirit Server (Linux)

■ Controlling the FirstSpirit Server (Windows)

15.8 FirstSpirit server: New control files for Windows and Linux 2020-03

Java and the Java Service Wrapper are prerequisites for operating a FirstSpirit server. The Java Service Wrapper and corresponding control files are provided by e-Spirit via:

■ the installation archive (for new installations) or

■ the update archive (for existing installations)

Both archives can be downloaded (for access data, please contact Technical Support).

The control files for Windows and Linux operating systems, which are included in both archives, have been enhanced by two new control files, with the current FirstSpirit release.

The new control files support a range of functions that were not available in the previous control files (including `install` under Linux). The scope of functions of the new control files can be called via the integrated help (Usage, Command syntax, Commands) in the command line, for example via the call `fs-server` (under Linux):

```
test@fs_example:~$ fs-server
Usage: /home/fs_example/firstspirit/bin/fs-server [ console | start | stop | restart |
       condrestart | status | install | installstart | remove | dump ]
Commands:
  console      Launch in the current console.
  start        Start in the background as a daemon process.
  stop         Stop if running as a daemon or in another console.
  restart      Stop if running and then start.
  condrestart  Restart only if already running.
  status       Query the current status.
  install      Install to start automatically when system boots.
  installstart Install and start running as a daemon process.
  remove       Uninstall.
  dump         Request a Java thread dump if running.
```

This new development's result can be tested as part of an Early Access Program (EAP) starting with FirstSpirit 2020-03. An early, constructive feedback helps us to identify problems faster and more easily,

thus allowing us to add missing functionality. For feedback please consult our Technical Support (for partners) or your designated Account Manager (for customers).

> **!** Since these new control files are still in the EAP phase, we do not recommend using them in production environments.

Control files in the installation archive or in the update archive:

- new: fs-server (for Linux)

- new: fs-server.bat (for Windows)

- fs5 (for Linux)

- fs5.cmd (for Windows)

Prerequisite for using the new control files:

An update of the Tanuki Java Service Wrapper to version 3.5.42 is necessary.

Further documentation:

- Updating the FirstSpirit backend

- Installing via install archive

- Controlling the FirstSpirit Server in Unix operating systems

- Controlling the FirstSpirit Server in Windows operating systems

16 Template development

16.1 Workflows: More configuration options for context-based information in e-mails 2020-05

In e-mails sent by workflows, special placeholders may be used that are automatically replaced by the system based on the context. For example, this can be used to include a link to preview the page on which the workflow is active or to provide a wide range of other information in the e-mail.

The placeholders can be configured based on the properties of an activity or a transition in the "E-mail" tab "(Template store"/"Workflows"/"State diagram"). When a transition or an activity is switched, e-mails containing the relevant information are then sent automatically.

In the current FirstSpirit release, the following options have been added to the placeholders:

- `%DATE_TIME%` = due date (when set) with time. The format is localized, e.g., in the English version: "February 25, 1986 at 11:04:37 AM CET".

- `%HISTORY_SHORT%` = History of the workflow instance showing only the last three entries in ascending order (e.g., entry: 8, 9, 10).

- `%HISTORY_LAST%` = History of the workflow instance showing only the last entry (e.g., entry: 10).

- `%HISTORY_DESC%` = History of the workflow instance showing all entries in descending order (e.g., entry: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1).

- `%HISTORY_SHORT_DESC%` = History of the workflow instance showing only the last three entries in descending order (e.g., entry: 10, 9, 8).

- `%LAST_USER_FULLNAME%` = The last editor; the full name of the person is displayed (if this is not known, the login name is shown).

- `%NEXT_USER_FULLNAME%` = The names of the persons permitted to edit the workflow at the next stage. The full names are displayed (if these are not known, the login names are shown).

All previous placeholders are still supported:

- `%FIRSTspiritURL%` = HTTP connection mode (standard mode; master language)

- `%FIRSTspiritSOCKETURL%` = SOCKET connection mode (master language)

- `%PAGESTORE_PREVIEW_URL%` = Preview URL for a page from the Page Store (master language)

- `%SITESTORE_PREVIEW_URL%` = Preview URL for a page reference from the Site Store (master language)

- `%WF_NAME%` = Name of the workflow

- `%CREATOR%` = The name of the person who created the workflow. The full name is displayed (if this is not known, the login name is shown).

- `%LAST_USER%` = The name of the person who last edited the workflow. The login name is shown.

- `%LAST_COMMENT%` = Last comment

- `%NEXT_USER%` = The names of the persons permitted to edit the workflow at the next stage. The login names are shown.

- `%PRIORITY%` = Priority

- `%DATE%` = Due date (only if set)

- `%HISTORY%` = History of the workflow instance showing all entries in ascending order (e.g., entry: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10).

- `%WEBeditURL%` = WebEdit link to the preview of the page (master language)

Additional documentation:

- Workflows/Properties of an activity/E-mail tab

- Workflows/Properties of a transition/E-mail tab

16.2   Extended output options for snippets   2020-03

At many points in the FirstSpirit clients the presentation of objects (pages, sections, datasets, media, etc.) can be influenced, for example in search result lists, reports, selection dialogs, etc. The so-called "Snippets" are used for this. The configuration is done on the "Snippet" tab of the respective template, for media on the "Snippet" tab of the metadata template selected for the project.

As of the current release, the output options for snippets have been extended. For example, the display name or the UID of the desired element can be output. For this purpose, you can now use the variable

```
#item
```

in the fields "Label" and "Extract" on the "Snippet" tab.
It returns the respective tree element. Object-specific methods from the interface IDProvider (FirstSpirit Access API, package `de.espirit.firstspirit.access.store`) can in turn be applied to this.

Examples

```
#item.getDisplayName(#language)
```

Outputs the display name of the respective object in the respective language.

```
#item.Uid
```

Outputs the UID of the respective object.

Example

```
#item.getDisplayName(#language)
"[UID:" + " " +italic(#item.getUid) + "]"
```

For **datasets**, the use of `#item.getDisplayName()` is not allowed, as this will output the snippet itself. Instead, the default value for the snippet will be displayed (data source name and ID of the dataset, separated by a # character. For example, the GID of a dataset can be specified using

```
#item.getEntity().getGid()
```

Furthermore, `#item` is not available in the data type **Card**.

For **media** (snippets of the Metadata Template) the following variables are additionally available:

- *#medium*: provides information on the medium in XML format (PICTURE or FILE)

- *#mediumtype*: identifies whether the medium is an image or a file (PICTURE or FILE)

- *#extension*: identifies the file extension

- *#filename*: identifies the file name

- *#filesize*: determines the file size in bytes

- *#mimetype*: identifies the MIME type

- *#resolution*: determines the dimensions of the image in pixels (only for `PICTURE`)

Example
Label:

```
#filename + "." + #extension + if(#mediumtype == "Picture", " (" + #resolution
    + "px)", "")
```

Extract:

```
#item.displayName(#item.project.masterLanguage) + " (" + #mimetype + ")"
```

Gives as title the file name with file extension and the resolution of the selected medium, as text extract the display name and the MIME type.
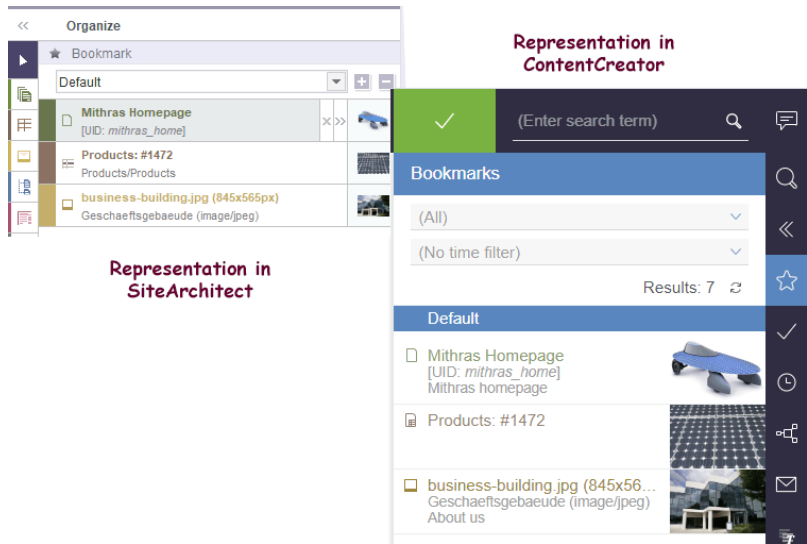
> **!** In exceptional cases, for example when creating objects, `#item` is not available and a warning of the type `java.lang.UnsupportedOperationException` or `java.lang.RuntimeException` will be logged ("Expression failed"). This should be taken into account during template development.

For further information about snippets please see *FirstSpirit Online Documentation*, https://docs.e-spirit.com/odfs/template-develo/snippets/index.html.

17 Deprecations

For e-Spirit, an important goal in software development is to avoid introducing incompatibilities and migration expenditures related to updating from one FirstSpirit release to the next as much as possible or to compensate for these within the software. FirstSpirit updates should generally be deployable with little effort or able to be carried out in a fully automated fashion.

However - not least in order to ensure maintainability and to future-proof the software - e-Spirit cannot fully avoid replacing existing functionality with new mechanisms. In the future, functionality that will be removed from the software will be listed in this section, including the date at which time the functionality will be removed.

Functionality	Deprecated as of	Will be removed/ Was removed as of
Input component FS_LIST	5.2R3	2020-07
Input component CMS_INPUT_CONTENTAREALIST	5.2R3	
Input component CMS_INPUT_CONTENTLIST	5.2R3	
Input component CMS_INPUT_FILE	5.2R3	
Input component CMS_INPUT_LINKLIST	5.2R3	
Input component CMS_INPUT_OBJECTCHOOSER	5.2R3	
Input component CMS_INPUT_PAGEREF	5.2R3	
Input component CMS_INPUT_PICTURE	5.2R3	
Input component CMS_INPUT_TABLIST	5.2R3	
FirstSpirit Developer API: de.espirit.firstspirit.agency.GroupsAgent	5.2R15	
FirstSpirit Access API: delete (de.espirit.firstspirit.access.AccessUtil)	5.2R18	
Windows installation program	2018-06	2020-06
FirstSpirit Access API: release (de.espirit.firstspirit.access.AccessUtil)	2018-06	
FirstSpirit Developer API: getLastLoginAsDate (de.espirit.firstspirit.agency.UserStatisticsAgent)	2018-07	
FirstSpirit Developer API: remainingDurationOfCurrentStageInMillis (de.espirit.firstspirit.server.MaintenanceModeInfo)	2018-07	
FirstSpirit Developer API: getStartingTimeOfStageAsDate (de.espirit.firstspirit.server.MaintenanceModeInfo)	2018-07	
FirstSpirit Access API: getSelectedWebserverConfiguration (de.espirit.firstspirit.access.serverConfiguration)	2018-10	

Functionality	Deprecated as of	Will be removed/ Was removed as of
FirstSpirit Access API: setSelectedWebserverConfiguration (de.espirit.firstspirit.access.serverConfiguration)	2018-10	
FirstSpirit Access API: getSelectedWebServer (de.espirit.firstspirit.access.project.Project)	2018-10	
FirstSpirit Access API: setSelectedWebServer (de.espirit.firstspirit.access.project.Project)	2018-10	
FirstSpirit Developer API: getLostAndFoundStoreNodes(); (de.espirit.firstspirit.feature.FeatureInstallResult)	2018-10	
FirstSpirit Developer API: getDeletedStoreNodes(); (de.espirit.firstspirit.feature.FeatureInstallResult)	2018-10	
FirstSpirit Access API: de.espirit.firstspirit.access.store.Previewable	2019-01	
Java 8 support for FirstSpirit	2019-05	2020-06
WebSphere Application Server support for FirstSpirit	2019-05	
Legacy mode for the FirstSpirit server and module development	2019-06	
Java Web Start support for FirstSpirit	2019-10	2020-06
FirstSpirit ContentCreator (old design / old technology)	2019-11	2020-07
Mithras Energy demo project	2020-02	2020-06
Support for Microsoft Internet Explorer for FirstSpirit Web Apps	2020-05	2020-07
Control files for Windows and Linux operating systems (old)	2020-08	February 2021

18 Overview

ID	Description	Categories	Version
CORE-3312	Via "FirstSpirit ServerManager / Project properties / Template sets / Delete", you can now delete template sets for a project. This action cannot be undone. Further information can be found in chapter "Administration: ServerManager: Deleting template sets".	FirstSpirit Administrator, ServerManager, Template Store	2020-03
CORE-5816	Corrected a bug that could cause a `java.lang.StackOverflowError` in rare cases when using the `toJSON()` function in the output channel.	Developer, Support for JSON, Template Development	2020-04
CORE-7234	External users can now be used for password configurations in ServerManager. Further information can be found in chapter "SiteArchitect / ServerManager: External users can now be used for password configurations in ServerManager.".	ServerManager	2020-09
CORE-7678	In e-mails sent by workflows, special placeholders may be used that are automatically replaced by the system based on the context. In the current FirstSpirit release, these placeholders have been expanded to include further configuration options. Further information can be found in chapter "Template development: Workflows: More configuration options for context-based information in e-mails".	Developer, FirstSpirit Administrator, Release, Workflow	2020-05
CORE-8756	In addition to using the context menu, the order of the project languages ("FirstSpirit ServerManager / Project / Properties / Languages") can now also be changed by using the mouse pointer (drag-and-drop). Further information can be found in chapter "Administration: Change the order of the project languages by drag-and-drop".	FirstSpirit Administrator, Languages, ServerManager	2020-02

ID	Description	Categories	Version
CORE-9572	In the current release, a deadline time can now also be selected for workflows, in addition to the deadline date. Further information can be found in chapter "ContentCreator: Workflows: Selecting a deadline time".	ContentCreator, Editor, Workflow	2020-05
CORE-9677	With the new method `getPreviousSibling()` in the Interface `StoreElement` (package `de.espirit.firstspirit.access.store`, FirstSpirit Access API) the previous StoreElement can now be determined and returned starting from a StoreElement. Further information can be found in chapter "Module Development, Scripts, API: Determining and returning previous store elements".	Developer, FirstSpirit API	2020-08
CORE-10457	Syntax errors in BeanShell scripts for script visibility are now handled appropriately and no longer cause menus to be hidden, for example.	ContentCreator, Developer, Editor, FirstSpirit Administrator	2020-09
CORE-10498 CORE-11555	The output options for snippets have been extended (`#item` and others). Further information can be found in chapter "Template development: Extended output options for snippets".	Developer, Editor, FirstSpirit API, Snippets	2020-03

ID	Description	Categories	Version
CORE-10776	With the current FirstSpirit release the Java VM's startup parameters can be automatically updated. For this purpose, the startup options have been split into two separate configuration files. The new file `fs-wrapper-vendor.conf` contains the recommended vendor startup options. These are automatically updated when the FirstSpirit server is updated via the update archive `fs-update-[version].tar.gz`. The existing file `fs-wrapper[.isolated].conf` only contains those startup options which have been modified server-specifically. The changes are available for update archives **from version 1.0.5** on. The update archive is available for download (if you need login data, please contact the FirstSpirit Technical Support). Further information can be found in chapter "Administration: Automatically update Java version-specific wrapper files".	FirstSpirit Administrator, Java, Migration, Server Administrator, ServerMonitoring	2020-04
CORE-10797	When transporting datasets via "External Synchronization" or "Content Transport", updates were occasionally not recognized. The error only occurred if an input component was set to the value `NULL`. In this case, the value was ignored during the transport and an existing value in the target project was not updated. This error has been corrected.	Content Transport, External synchronization	2020-02
CORE-10832	**In the latest FirstSpirit release**, the order for displaying transitions on a workflow activity can be configured. The configuration is managed on the workflow in FirstSpirit SiteArchitect ("Templates" / "Workflows") in the State diagram tab via the new property "Transitions". The altered order affects the display of the buttons that are shown when a workflow is run through via the dialog box for the corresponding activity. Further information can be found in chapter "SiteArchitect / ContentCreator: Workflow: Changing the order of transitions on an activity".	ContentCreator, FragmentCreator, SiteArchitect, Template Store, Workflow	2020-05

ID	Description	Categories	Version
CORE-10834	The configuration options for the parameters "useWelcomeFileNames" and "stripWelcomeFilenames", which are used for the configuration of user-specific path strategies for URL generation, have been extended in the latest FirstSpirit release. Further information can be found in chapter "Module Development, Scripts, API: Improved configuration options for user-specific URLs (AdvancedUrlCreator)".	Advanced URL, Developer, Generation, Migration, SiteArchitect	2020-05
CORE-10836	When deleting a schedule with a generation action, the corresponding directory on the application server on which the Staging Web application is running is now also deleted. Further information can be found in chapter "Administration: Generation directories of deleted orders are cleaned up.".	FirstSpirit Administrator	2020-04
CORE-11017	The new rule `PROPAGATE_MEDIA_CHANGES_TO_CONTENTSTORE` in the `DeltaGeneration` interface can now be used to forward media changes to datasets that reference the medium in question. Further information can be found in chapter "Module Development, Scripts, API: During delta generation, media changes can now be forwarded to datasets that reference the medium in question.".	Delta generation, Developer	2020-09
CORE-11046	In very rare cases, restoring deleted objects may result in an error. If the cause of this error is that the target folder cannot be found (e.g., because it is deleted at the same time as objects are restored), there will now be a specific `RestoreFailedException` which provides further information to identify the cause.	Editor, FirstSpirit Administrator, SiteArchitect	2020-07
CORE-11051	If sections are being used as "section references" in multiple locations, this is displayed in the edit dialog by the term "Usages". The number indicates in how many locations the section is being referenced/reused. The source section is now added to the number of usages.	ContentCreator, Editor	2020-08

ID	Description	Categories	Version
CORE-11079	An error has been corrected which made it difficult to analyze problems when displaying input masks. In such cases, the exception that was actually causing the problem is now output instead of the unhelpful NullPointerException.	FirstSpirit Administrator, Input Components, SiteArchitect	2020-07
CORE-11179	The `reload` parameter, which can be forwarded to the `NewSection` executable via the `fsbutton(...)` function, is now correctly evaluated again.	ContentCreator, Developer, Editor, FirstSpirit API	2020-09
CORE-11224	An error has been corrected which, in very rare cases, may have resulted in a `java.util.ConcurrentModificationException` when editing datasets.	Content Store, Editor, FirstSpirit Administrator, Sessions	2020-08
CORE-11351	Changing from `IllegalArgumentException` to `InvalidSessionException`, when trying to get a *UserService* with expired session.	Debugging, Developer, Module development, Services	2020-06
CORE-11395	Performance improvement when using the input component FS_CATALOG in projects with many languages.	ContentCreator, Input Components, Performance	2020-01
CORE-11407	Behavior has been corrected which, in very rare cases, may have resulted in erroneous behavior when elements were deleted.	Editor, SiteArchitect	2020-07
CORE-11462	The "FirstSpirit External Synchronization" and "FirstSpirit Content Transport" mechanisms now provide an additional mapping file (`Mapping.xml`) for exporting database content (schemas, entities), in which the database names of the source project are stored together with FirstSpirit's own unique identifiers (UUIDs). Further information can be found in chapter "External Synchronization / Content Transport: External Synchronization and Content Transport: Improved support for shared access to a database".	Content Transport, Database, Developer, External synchronization, FirstSpirit API	2020-05

ID	Description	Categories	Version
CORE-11471	A bug was fixed that could cause that after updating a module used in schedules, the corresponding action could not be used until the server was restarted. **Note:** The problem may still occur once within the current execution for schedules that were running at the time of the update.	FirstSpirit Administrator, Isolated mode, Modules, Server Administrator, Tasks	2020-04
CORE-11516	When copying elements using the `AccessUtil.copyStoreElement` method, contents were not copied in rare cases.	Developer, FirstSpirit API	2020-03
CORE-11522	An error was fixed that could cause folder structures from the file system not to be transferred to the FirstSpirit Media Store when using the Media Import Wizard.	Media Store, SiteArchitect	2020-03
CORE-11540	A problem has been rectified which could lead to unnecessary waiting times when copying datasets in FirstSpirit SiteArchitect (from the second copy of a dataset).	Content Store, Database, Performance, SiteArchitect	2020-09
CORE-11592	The error tolerance when implementing customer-specific reports has been increased so that reports can now also be displayed if there are problems using icons (e.g., error message `WARN 18.11.2019 09:41:07.595 (de.espirit.firstspirit.webedit.server.sidge` Web client report 'xyz' couldn't be loaded!).	Data Access Plug-In, Debugging, Developer, Editor, FirstSpirit Administrator, Module development, Reports	2020-09
CORE-11607	Project archiving has been improved in the latest FirstSpirit release. Using project archiving (the "Archive old project states" schedule), all automatically calculated media/pictures <u>whose resolutions have already been deleted in the project</u> can now be removed from the image cache. Further information can be found in chapter "Repository: Improved archiving function ".	Archiving, FirstSpirit Administrator, Performance, ServerManager	2020-02
CORE-11691	Error handling when uploading reporting data has been improved. Reporting is only applied if it has been explicitly activated for the corresponding instance.	Debugging, FirstSpirit Administrator, Server Administrator, Tasks	2020-01

ID	Description	Categories	Version
CORE-11699	The use of the revised ContentCreator (also "FirstSpirit CXT ContentCreator") is officially released as of the current FirstSpirit release. Further information can be found in chapter "ContentCreator: ContentCreator re-design: Release".	ContentCreator, Developer, Editor, FirstSpirit Administrator, ServerManager	2020-01
CORE-11711	If the same global resources are provided by several modules in different versions, only one of these global resources can be used by all modules. The latest FirstSpirit release ensures that in normal operation, the latest version (i.e., the version with the highest version number) of a global resource is always used. This increases the stability when it comes to resolving dependencies after a module update. Further information can be found in chapter "Module Development, Scripts, API: Global module resources: Improved stability when resolving dependencies".	Developer, Isolated mode, Module development, Modules	2020-05
CORE-11722	With the current FirstSpirit release all deprecation warnings of a project are consolidated and logged in the first section of the log file (in the case of generation in the generation log file and in the case of a preview under "Show preview errors"). Further information can be found in chapter "Input components: Improved Deprecation Warnings".	Developer, Input Components, Template Development	2020-01
CORE-11740		Sessions	2020-02
CORE-11741	**With FirstSpirit 2020-01**, the input component FS_LIST is no longer supported. The component can no longer be used from this point on. That means forms with FS_LIST can no longer be edited and the expected output will no longer be generated. Further information can be found in chapter "Input components: FS_LIST: Ramp-down phase: The input component is no longer supported".	ContentCreator, Developer, Input Components, Migration, SiteArchitect, Template Development	2020-01

ID	Description	Categories	Version
CORE-11772 CORE-11786	Internal software has been updated in the latest FirstSpirit release. Further information can be found in chapter "System: Updates to integrated third-party software".	FirstSpirit Administrator, Integrated software, Java, Launcher, Modules	2020-03
CORE-11784 CXT-1433	The design and technical basis of FirstSpirit ContentCreator were revised. The "old" ContentCreator retires with the latest release. Further information can be found in chapter "ContentCreator: Retirement of the "old" ContentCreator and further design improvement".	ContentCreator, Editor, FirstSpirit Administrator	2020-07
CORE-11796	The metaspace (another native memory area next to the heap) is automatically managed by the Java VM. A different configuration of this memory area, for example using the parameters `MetaspaceSize` or `MaxMetaspaceSize`, is no longer recommended. Further information can be found in chapter "Administration: Metaspace: Note on the configuration of the Java VM".	Server Administrator	2020-02
CORE-11797	If the FS_CATALOG component was being used in a nested manner, it was possible that content from the translation help would not transfer correctly. The corresponding error has been rectified.	Editor, Input Components, SiteArchitect	2020-07
CORE-11800	With the current FirstSpirit release a new version of the JSON output format is released. From FirstSpirit 2020-02 the new format version 1.1 is used (standard setting). The previous format version 1.0 remains supported but is to be replaced in the medium term. Further information can be found in chapter "General: New JSON format version 1.1".	Developer, Support for JSON	2020-02

ID	Description	Categories	Version
CORE-11808 CXT-652	New interface `UrlSupporting` for "DataAccessPlugins", which can be used to generate URLs for data objects. Further information can be found in chapter "Module Development, Scripts, API: DAP: New aspect "UrlSupporting" for URL generation".	Data Access Plug-In, Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Media	2020-03
CORE-11816	The `RenderingAgent` interface (package: `de.espirit.firstspirit.agency`, FirstSpirit Developer API) can be used to control generation for specific dynamic FirstSpirit content directly via the API. In conjunction with the json() function, the output was incomplete here in rare cases. This error has been resolved.	Developer, FirstSpirit API, Support for JSON	2020-04
CORE-11827 CORE-12141 CXT-1393	Update of internally used software Further information can be found in chapter "System: Update of internally used software".	FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FirstSpirit Jetty Web Server, Integrated software, Java, Launcher	2020-04
CORE-11835	An error was fixed that could cause data not to be displayed when migrating the input component FS_LIST (discontinued with 2020-01) to FS_INDEX. The problem only occurred in rare cases in connection with inconsistent data.	Developer, Input Components, Migration	2020-02
CORE-11857	An input component developer can now provide the form element's (e.g. input component's) model references via the aspect `FormReferenceContaining`, with the method `List<Reference> collectFormReferences(F formElement)` (FirstSpirit Developer API, package `de.espirit.firstspirit.client.access.editor`). Further information can be found in chapter "Module Development, Scripts, API: New Aspect "FormReferenceContaining" (provide a form element's model reference)".	Developer, FirstSpirit API, Template Development	2020-03

ID	Description	Categories	Version
CORE-11858	The `ReferenceConstructionAgent` (FirstSpirit Developer API, package `de.espirit.firstspirit.client.access.editor`) was expanded with the latest FirstSpirit release to include methods which enable references to be generated on the basis of meta information (via the unique identifier (UID) and IDProvider.UidType of a FirstSpirit object or via the global identifier (GID) for references to datasets). At the same time, the `ValueReferencesJournal` interface (FirstSpirit Access API, package `de.espirit.firstspirit.client.plugin.dataacce` was expanded with new methods which can be used to add defective references as well as regular references (e.g., in a DataAccessProvider). Further information can be found in chapter "Module Development, Scripts, API: API extension for generating references (ReferenceConstructionAgent and ValueReferencesJournal)".	Data Access Plug-In, Developer, FirstSpirit API, Module development	2020-04
CORE-11859	During start-up of a FirstSpirit server, information about the current runlevel is provided. An error has been fixed, which caused the runlevel of the FirstSpirit server to be incorrect (after a restart) when using an external web server, unless the external web server was also restarted. This error had no effect on the functionality of the web applications. Now the correct runlevel is reached at the latest 15 minutes after starting the FirstSpirit server.	Server Administrator, Web server	2020-03
CORE-11866	In very rare cases, an error occurred while using the ContentCreator, which was caused by a "not authorized" message. This error has been fixed.	ContentCreator	2020-02
CORE-11870 CXT-11809	Rollout of the new ContentCreator Further information can be found in chapter "ContentCreator: Rollout of the new ContentCreator".	ContentCreator, FirstSpirit Administrator, Performance	2020-02

ID	Description	Categories	Version
CORE-11871	The "Mithras Energy" demo project (for ContentCreator and SiteArchitect) will be dropped in the future and will no longer be distributed with FirstSpirit. Further information can be found in chapter "General: Announcement: FirstSpirit Experience Accelerator".	ContentCreator, Developer, Editor, SiteArchitect	2020-02
CORE-11878	A potential memory leak in ContentCreator has been fixed.	ContentCreator, FirstSpirit Administrator	2020-03
CORE-11906	If a particular key term (e.g. FS_LIST) is selected in a FirstSpirit template and the "F1" key (or optionally the "Help" button) is pressed at the same time, the FirstSpirit documentation for this subject area is opened. For deprecated input components, the FirstSpirit documentation refers to alternative input components (e.g. FS_REFERENCE instead of CMS_INPUT_PICTURE). For the input component FS_LIST (that is no longer available) and all deprecated input components replaced by FS_LIST (e.g. CMS_INPUT_CONTENTLIST), the documentation refers to the corresponding migration guide.	Developer, Input Components, Migration	2020-02
CORE-11911	The DatasetDataAccessPlugin now also supports the referencing of records from external databases. Further information can be found in chapter "General: External database support via the DatasetDataAccessPlugin".	Data Access Plug-In, Database	2020-04
CORE-11913	An error was fixed that could cause information in languages that are only present in the target project to be lost when transporting FS_LIST data migrated to FS_CATALOG. This problem affected all converted FS_LIST input components (discontinued with 2020-01) if they were converted with FirstSpirit version 2019-12 or 2020-01.	Content Transport, External synchronization, Input Components, Migration	2020-02

ID	Description	Categories	Version
CORE-11919	The control files for Windows and Linux operating systems have been enhanced with the current FirstSpirit release. The new control files support a range of functions that were not available in the previous control files (including `install` under Linux). This new development's result can be tested as part of an Early Access Program (EAP). The use in productive environments is not recommended at this time. Further information can be found in chapter "System: FirstSpirit server: New control files for Windows and Linux".	Launcher, Server Administrator	2020-03
CORE-11932	Rollout of the new ContentCreator for all customers Further information can be found in chapter "ContentCreator: New ContentCreator: Rollout for all customers".	ContentCreator, Editor, FirstSpirit Administrator, Modules, ServerManager	2020-03
CORE-11992	A vulnerability has been discovered in the Apache JServ Protocol (AJP) binary communication protocol used by Tomcat (Ghostcat CVE-2020-1938). The vulnerability is classified as critical. e-Spirit therefore recommends a prompt update to a secure Tomcat version (8.5.51 or 9.0.31). We also recommend checking the AJP Connector configuration in each installation and adjusting it if necessary, as the AJP Connector was activated by default in the `server.xml` configuration file before the above-mentioned versions. Further information can be found in chapter "Security: Apache Tomcat AJP: Vulnerability (Ghostcat) ".	FirstSpirit Administrator, Security, Server Administrator, Web server	2020-03
CORE-11993	Due to the change from Java EE to Jakarta EE and the associated change of namespace (package) in Tomcat 10, FirstSpirit is currently not compatible with Tomcat 10. Further information can be found in chapter "Administration: FirstSpirit is currently not compatible with Tomcat 10".	FirstSpirit Administrator, Server Administrator, Web server	2020-03
CORE-12006	An error has been corrected which, in very rare cases, may have resulted in a `StackOverflowError` when classes were loaded.	FirstSpirit Administrator, Modules, SiteArchitect	2020-07

ID	Description	Categories	Version
CORE-12016	The output behavior when accessing the CMS_INPUT_PERMISSION input component via the API has been corrected. Further information can be found in chapter "Input components: The output behavior when accessing the CMS_INPUT_PERMISSION input component via the API has been corrected.".	Metadata, Permissions	2020-07
CORE-12018	**Support Open JDK 14:** As of FirstSpirit 2020-04, OpenJDK 14 (non-LTS; release date 2020/03) is officially approved for use with FirstSpirit (i.e, the FirstSpirit Server and the FirstSpirit desktop applications). With support for OpenJDK 14, support for **OpenJDK 13** (non-LTS; release date was 2019/07) expires. **Discontinuation of Java 8:** As part of product maintenance, the discontinuation of support for Java 8 was announced with FirstSpirit 2019-05: FirstSpirit is scheduled to no longer support Oracle Java 8 and OpenJDK 8 from June 2020. Further information can be found in chapter "General: Java: Current status and outlook".	Developer, FirstSpirit Administrator, Java, Server Administrator	2020-04
CORE-12023 CXT-1527	The ID, UID, GID, and the corresponding template of FirstSpirit elements can now be determined using rules. Further information can be found in chapter "SiteArchitect / ContentCreator: New attributes for rules: ID, UID, GID, template".	ContentCreator, Developer, Dynamic Forms, Editor, Input Components, Template Development	2020-07
CORE-12024	When using the translation help, it was possible that an error would occur if the data had been created with an input component which is no longer available (e.g., FS_LIST). The error has been corrected, so the current source data now displays properly.	Editor, Input Components, Migration, SiteArchitect	2020-07
CORE-12031	An error could occur when using services from your own services if the components involved were not yet fully started. This dependency scenario has been corrected.	FirstSpirit Administrator, Server Administrator	2020-04

ID	Description	Categories	Version
CORE-12038	With the current FirstSpirit release the FirstSpirit session cookie can be extended by the attribute `SameSite`. The setting `SameSite=None` may be necessary in individual cases if FirstSpirit web applications are integrated into other web applications (e.g. as an IFrame). Further information can be found in chapter "Administration: The FirstSpirit session cookie now supports the attribute SameSite".	FirstSpirit Administrator, Security, Server Administrator, ServerMonitoring	2020-05
CORE-12043	An error has been resolved which, in rare cases, could previously lead to problems when using X.509 certificates in conjunction with the `fs-jetty.fsm` module ("FirstSpirit Jetty Service"). (Note: Using Eclipse Jetty for production purposes is still not recommended!)	FirstSpirit Jetty Web Server, Server Administrator, Web server	2020-04
CORE-12060	FirstSpirit documentation which we provide in the HTML format will no longer be provided as PDF. Further information can be found in chapter "General: FirstSpirit documentation: PDF format discontinued".	Documentation	2020-04
CORE-12068	Pop-up dialogs created using the interface `RequestOperation` (FirstSpirit Developer API, Package `de.espirit.firstspirit.ui.operations`) can now also be created as warning messages via `Enum RequestOperation.Kind` (FirstSpirit Developer API, Package `de.espirit.firstspirit.ui.operations`). Further information can be found in chapter "Module Development, Scripts, API: Creating a pop-up dialog as a warning message".	ContentCreator, Developer, FirstSpirit API, Module development, ServerManager, SiteArchitect	2020-05
CORE-12079	In rare cases, datasets could not be edited via EasyEdit due to incorrect evaluation of the relevant `editorId`.	Content Store, ContentCreator, Developer, Editor, Table templates	2020-04

ID	Description	Categories	Version
CORE-12085	Using the JSON function, the referenced FirstSpirit objects of the input component FS_REFERENCE can now be transferred to a JSON object structure (previously the referenced objects were not rendered). In this process, the object itself is not rendered; the function only renders the data that is required for referencing. No new version of the JSON output format will be published for this change; the changes will be made in the current format version 1.1. Further information can be found in chapter "Module Development, Scripts, API: JSON function: Improved support for FS_REFERENCE".	Developer, Support for JSON	2020-05
CORE-12091	If one or more web applications are configured to run on the "old" ContentCreator of a FirstSpirit server, the start page of the FirstSpirit server will show a warning to users with the administrator role. Further information can be found in chapter "ContentCreator: Use of the "old" ContentCreator".	ContentCreator, Developer, FirstSpirit Administrator	2020-04
CORE-12129	An error was rectified that could have meant that the Classloading for a service did not function as expected.	Developer, FirstSpirit Administrator, Modules, Server Administrator	2020-05
CORE-12161	In rare cases, saving an FS_CATALOG input component may have resulted in erroneous behavior.	Editor, Input Components, SiteArchitect	2020-07
CORE-12171	The vertical alignment of content in tables can now also be defined by the editor in FirstSpirit ContentCreator (CMS_INPUT_DOM / "Inline tables" and CMS_INPUT_DOMTABLE). Further information can be found in chapter "ContentCreator: Tables: Vertical alignment of content in cells ("valign")".	ContentCreator, Developer, Editor	2020-05

ID	Description	Categories	Version
CORE-12205	The classes from the "Apache PDFBox" open-source library have been removed from `fs-[isolated-]webrt.jar`. If these classes were used during module development (which was only possible for modules that contain web applications in legacy mode), they are now missing and must be added explicitly to the web application. Further information can be found in chapter "Module Development, Scripts, API: The classes from the Apache PDFBox open-source library have been removed from fs-[isolated-]webrt.jar".	Developer, Module development, Modules	2020-07
CORE-12209	Ramp-up phase for new feature: "Smart Cropping" reduces the effort for manual image cropping thanks to automatic operation Further information can be found in chapter "ContentCreator: Ramp-up: Less manual image cropping thanks to "Smart Cropping"".	ContentCreator, Editor, FirstSpirit Administrator, Media, Media Store	2020-07
CORE-12223	The Java versions included in the "FirstSpirit Launcher JRE" module were updated in the latest FirstSpirit release. Further information can be found in chapter "SiteArchitect / ServerManager: "FirstSpirit Launcher JRE" module: update to Java versions".	Java, Launcher	2020-06
CORE-12229	Internal software has been updated in the latest FirstSpirit release. Further information can be found in chapter "System: Updates to integrated third-party software".	FirstSpirit Administrator, Integrated software	2020-05

ID	Description	Categories	Version
CORE-12240	When adding a new database layer via the ServerManager (Server Properties / Databases / Add), problems could occur if special connection parameters (`jdbc.property.`) were required for the database connection (e.g. for TNS entries, SSL connections, etc.). These connection parameters were not properly parsed. As a result, the connection test (button "Test connection") failed and the database layer could not be saved although it was configured correctly. This problem has been fixed.	Database, ServerManager	2020-07
CORE-12245 CXT-12321	When updating the "SpellService" module (file `fs-spell.fsm`) to a higher version, existing configurations were overwritten in rare cases. This could cause, for example, that self-created dictionaries were removed or not taken into account. In addition, a bug was fixed which caused the FirstSpirit server with "SpellService" module to no longer start after an update (error message in `fs-server.log` like `ERROR ... (de.espirit.firstspirit.server.module.Service cannot install service, module=SpellService, version=..., componentClass=de.espirit.firstspirit.opt.spe`	Modules, Spell Checker	2020-05
CORE-12246	An error was rectified which was causing the `installed` method to be called up for all services when ServiceManager started up. Executing the `installed` method may have meant that global configuration files were rewritten, for example.	FirstSpirit Administrator, Modules, ServerManager, Services	2020-05
CORE-12249	The stability of the system when running module updates during live operation has been further improved. Further information can be found in chapter "Administration / Modules: Improved stability of the system when running module updates during live operation".	Modules	2020-08

ID	Description	Categories	Version
CORE-12250 CORE-11119	Update of internally used software. Further information can be found in chapter "System: Update of internally used software".	Developer, FirstSpirit Administrator, FirstSpirit Jetty Web Server, Integrated software	2020-06
CORE-12261	Fixed a bug that could cause a freeze of the application in very rare cases (deadlock in `DefaultStoreElement#_permissionCache`).	Developer, FirstSpirit API	2020-06
CORE-12262	Changes to an object which have taken place from one revision number to another revision number can be displayed in FirstSpirit using differential visualization (via context menu "Version History" / Button: "Compare"). An error in the display of this view in combination with dropped input components (such as FS_LIST) has been corrected.	Input Components, SiteArchitect	2020-07
CORE-12281	The API (FirstSpirit Access API, `PackagePoolItem` interface, `de.espirit.firstspirit.access.store` package) has been stabilized so that a `NullPointerException` will no longer occur when an attempt is made to remove a PackagePool element from a package that does not exist.	Corporate Content, Developer, FirstSpirit API	2020-08
CORE-12289	The configuration of permitted applications ("FirstSpirit ServerManager - Project properties - Applications") is now taken into account for export and import when using the "External Synchronization" functionality.	Developer, External synchronization	2020-06
CORE-12297	Element caching when using FirstSpirit connections has been corrected. In rare cases, data changes to store elements in service implementations have not been correctly detected or forwarded.	Developer, FirstSpirit API	2020-09
CORE-12298	A FirstSpirit server is installed via the installation archive `fs-install-[version].tar.gz`. When executing the commands `FS_HOME/bin/fs-server installstart` or `FS_HOME/bin/fs-server install` (under Linux operating systems) an error could occur because the `run` folder was not available. This folder is now included in the installation archive.	Server Administrator	2020-05

ID	Description	Categories	Version
CORE-12322	In FirstSpirit ServerMonitoring, external groups of users are displayed in the "User" area only for server administrators, no longer for project administrators.	FirstSpirit Administrator, Permissions, Server Administrator, ServerMonitoring	2020-06
CORE-12333	The Convenience.Method `isEmpty()` has been introduced for `de.espirit.common.util.Listable`. It can be used to check whether the corresponding object is empty or not. Further information can be found in chapter "Module Development, Scripts, API: Introduction of a new Convenience.Method "isEmpty()" ".	FirstSpirit API	2020-08
CORE-12336	As previously announced, with the end of Java 8 support, the support for the Windows installation program (.exe) and, at the same time, the support for the previous tar archive (.tgz) for all Unix systems will also be discontinued. The previous installation options will be replaced by new installation options for the FirstSpirit server and update options for the FirstSpirit backend. Further information can be found in chapter "Administration: The support for the Windows installation program and the tar archive (Unix) is discontinued".	FirstSpirit Administrator, Java, Server Administrator	2020-06
CORE-12337	**End of Java 8 support:** As part of product maintenance, support for Java 8 with FirstSpirit version 2020-06 will be discontinued. Further information can be found in chapter "General: End of Java 8 support".	Developer, FirstSpirit Administrator, Java, Server Administrator	2020-06

ID	Description	Categories	Version
CORE-12338	Up to now, the FirstSpirit desktop apps (FirstSpirit SiteArchitect and ServerManager) could be started and updated (roll-out) via Java Web Start or via the FirstSpirit Launcher. With the current FirstSpirit version, FirstSpirit no longer supports the option of starting and rolling out the FirstSpirit desktop applications via Java Web Start (as announced with 2019-10). Further information can be found in chapter "Administration: End of support: Java Web Start".	FirstSpirit Administrator, Server Administrator, ServerManager, SiteArchitect	2020-06
CORE-12339	With the current FirstSpirit release the Mithras Energy demo project for ContentCreator and SiteArchitect is no longer available and will no longer be distributed with FirstSpirit. e-Spirit replaces the demo project with the contemporary FirstSpirit Experience Accelerator. Further information can be found in chapter "SiteArchitect / ContentCreator: The Mithras Energy demo project is no longer available".	ContentCreator, Developer, Editor, FirstSpirit Administrator, SiteArchitect	2020-06
CORE-12343	New layer for Oracle Database 19c Further information can be found in chapter "Compatibility: New layer for Oracle Database 19c".	Database	2020-06
CORE-12362	An error was fixed that could cause that saving an FS_INDEX input component that references datasets was not possible.	Content Store, Developer, Editor, Input Components	2020-06
CORE-12368	Starting FirstSpirit using Linux has been further improved with the new "fs-server start script". Further information can be found in chapter "Administration: Starting FirstSpirit using Linux has been further improved with the new "fs-server start script".".	Server Administrator	2020-07

ID	Description	Categories	Version
CORE-12388	With the current FirstSpirit release, the parameter `jdbc.MAXSTRINGLENGTH` is supported for PostgreSQL and Oracle 19c databases. Please note the recommendations for configuring the parameter in this document and in the "FirstSpirit documentation". Customers who use the `jdbc.MAXSTRINGLENGTH` parameter in an existing PostgreSQL database configuration are advised by e-Spirit to remove the parameter from the configuration before updating to FirstSpirit 2020-07, in order to avoid data loss. Further information can be found in chapter "SiteArchitect / ServerManager: Database connection: support for the parameter jdbc.MAXSTRINGLENGTH ".	Database, Migration	2020-07
CORE-12414	Erroneous behavior could occur when creating remote references with the `ReferenceConstructionAgent` interface (FirstSpirit Developer API, `de.espirit.firstspirit.client.access.editor` package).	Developer, FirstSpirit API	2020-07

ID	Description	Categories	Version
CORE-12416	The new method `getOutput()` in the `GenerationAgent` interface (FirstSpirit Developer API, `de.espirit.firstspirit.scheduling.agency` package) can be used to allocate the generated files to the corresponding FirstSpirit objects after a generation process. This information can be used in a downstream schedule step in order to change the generated data or to forward the information about this data to third-party systems. In addition to the output allocation, the `PermissionsAgent` (FirstSpirit Developer API, `de.espirit.firstspirit.access.editor` package) has been introduced in order to determine existing access and execution permissions for the individual FirstSpirit objects. Further information can be found in chapter "Module Development, Scripts, API: Accessing generated data via schedule script".	Developer, FirstSpirit API, Generation, Module development	2020-09
CORE-12435 CXT-1704	In the current FirstSpirit release, the content of FirstSpirit jar files is signed with a new certificate. An update of the FirstSpirit Server is not necessary. Further information can be found in chapter "General: Code signing certificate update".	Server Administrator, ServerManager, SiteArchitect	2020-08
CORE-12460	As of FirstSpirit 2020-07, the FS_LIST input component is no longer included. The component is no longer available for use. This means it is no longer possible to edit forms with FS_LIST and expected content will no longer be generated on the output side. Further information can be found in chapter "Input components: FS_LIST: Ramp-down phase complete: input component no longer included".	ContentCreator, Developer, Input Components, Migration, SiteArchitect, Template Development	2020-07

ID	Description	Categories	Version
CORE-12473	During module updates, it was not possible to end and restart an associated service correctly with certain module configurations. This has been resolved by correcting the update process.	FirstSpirit Administrator, Module development, Modules	2020-07
CORE-12489	A problem was fixed where **SVG** and **WebP** graphics formats could not be imported into projects using `fs-cli` (FSDevTools). This was caused by a missing component in the `fs-isolated-runtime.jar` normally used by fs-cli.	Developer, FSDevTools	2020-08
CORE-12504	When importing new and existing schedule entries using the "External Synchronization" function, the activity status of the schedule entries can now be transferred from the source project to the target projects using the command line tool FSDevTools (fs-cli) or using the FirstSpirit Developer API. Further information can be found in chapter "External Synchronization: Transporting the Activity Status of Schedule Entries".	Developer, External synchronization, FirstSpirit API, Tasks	2020-08
CORE-12561	Update of internally used software. Further information can be found in chapter "SiteArchitect / ServerManager: Update of internally used software".	Integrated software, Java, Launcher	2020-08
CORE-12567	A problem has been solved when closing remote connections to other projects. This problem in some cases had negative effects on the main connection.	Developer	2020-08
CORE-12598	The current FirstSpirit release includes the definitive release of the new control files for Windows and Linux operating systems. The old control files are being **removed**. Further information can be found in chapter "System: FirstSpirit Server: Release of new control files for Windows and Linux".	Launcher, Server Administrator	2020-08

ID	Description	Categories	Version
CORE-12623	When using an FS_CATALOG input component with multiple FS_INDEX input components which can each be used to select the data from a DataAccess plug-in via drag-and-drop, the transfer type is now evaluated correctly in ContentCreator. Previously, erroneous behavior could occur in FS_CATALOG when dragging and dropping data.	ContentCreator, Developer, Editor, Input Components, Module development, Reports	2020-09
CORE-12651	With the current FirstSpirit release the FirstSpirit Session Cookie is extended by the attribute `SameSite=Lax` (previously no value was set). Further information can be found in chapter "Administration: New default value: Attribute SameSite=Lax for the FirstSpirit Session Cookie".	FirstSpirit Administrator, Security, Server Administrator, ServerMonitoring	2020-09
CORE-12677	When switching from FirstSpirit 2020-08 to another FirstSpirit version, the FirstSpirit server may not be accessible. Further information can be found in chapter "Administration: Problems starting the server after updating from FirstSpirit 2020-08".	FirstSpirit Administrator, Server Administrator	2020-09
CORE-12701	Modified user guidance when restoring a version in the "Show modifications" dialog. Further information can be found in chapter "ContentCreator: Restoring a version in the "Show modifications" dialog".	ContentCreator, Developer, Editor, FirstSpirit API	2020-09
CORE-12728	The performance when using the Navigation view has been improved. Previously, particularly in the event of larger volumes of referenced datasets, increased numbers of server requests could occur in certain cases, resulting in higher loads.	ContentCreator, Developer, FirstSpirit Administrator, Performance	2020-09

ID	Description	Categories	Version
CORE-12816	The release of the new database layer **Oracle Database 19c**, planned for 2020Q3, has been postponed and is now scheduled for 2020Q4. Currently, the Oracle Database 19c layer is still in the "EAP" ("Early Access Program") phase. Further information can be found in chapter "Compatibility: Postponed: Release of the new database layer Oracle Database 19c".	Database	2020-09
CORE-15515	In the current FirstSpirit version, the certificate of the FirstSpirit Launcher is updated. Further information can be found in chapter "SiteArchitect / ServerManager: Certificate update FirstSpirit Launcher".	Launcher	2020-09
CXT-177	As of the current release, references to pages can now be created using the new *PageRef Access Plugin* module. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): PageRef Access Plugin (PageRefDAP): Referencing pages".	Developer, Editor, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules, Remote access, ServerManager, Template Development	2020-06
CXT-193	Workflow permissions (context menu: "Extras" / "Change Permissions" / "Workflow permissions" tab), which should apply to all users, ("All" checkbox activated) are now correctly evaluated in FragmentCreator. Previously, a 500 or 403 error message appeared.	Developer, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Permissions, Template Development, Workflow	2020-07

ID	Description	Categories	Version
CXT-578	The ability for several editors to edit variants has been optimized. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Optimizations for multi-user mode".	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Sessions	2020-06
CXT-580	Improved user guidance when selecting another medium (FS_INDEX). Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): FragmentCreator: Easier selection of other medium".	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components, Media	2020-01
CXT-652 CORE-11808	As of the current release, project-specific referencing options can be made available in the markdown editor *Quill*. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Markdown: Creating links using "DataAccessPlugins"".	Data Access Plug-In, Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Media	2020-03
CXT-711	The language defined using the `lang` attribute is now taken into account when input element labeling is displayed. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Labeling input elements in FragmentCreator ("display language")".	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Languages	2020-06
CXT-726	As of the current release, the creation of empty fragments via MicroApp can now be prevented. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): MicroApps: Cancel fragment creation".	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-08

ID	Description	Categories	Version
CXT-742	Content from the FragmentCreator can also be edited in the ContentCreator if configured accordingly. Modifications are now visible in the preview directly when the corresponding form is closed, not only after a manual update of the preview page.	ContentCreator, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-03
CXT-760	In FragmentCreator, favored variants can now be provided with actions from customer-specific MicroApps. The standard workflows for release and deletion are started there by default. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): MicroApps API: Providing variants with actions from MicroApps".	Developer, FirstSpirit API, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Module development, Modules, Workflow	2020-07
CXT-846	FS_INDEX with DataAccessPlugin component *FirstSpiritFragmentAccess/FSFAConnector* now also supports the creation of fragments of several different categories (tag `CATEGORY`). Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): FS_INDEX now also supports the creation of fragments of several different categories (tag CATEGORY)".	Developer, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components	2020-04
CXT-864	The Fragment DAP report now features filtering fragments by category. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Fragment DAP: filter fragments in report by category".	Data Access Plug-In, Editor, FirstSpirit Content Experience Tools (CXT), Reports	2020-05
CXT-879	When using the module "FirstSpirit CXT FragmentCreator - CaaS Integration", releases and other switching of workflows are now also taken into account during the event-based generation of fragments on the preview CaaS server. In addition, an error has been fixed that prevented the generation by task from being completed.	FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FirstSpirit Content as a Service (CaaS), Tasks, Workflow	2020-02

ID	Description	Categories	Version
CXT-1111	When starting MicroApps in FragmentCreator (creating/editing fragments or uploading images), the error message `An Error Occurred` no longer appears erroneously.	FirstSpirit Content Experience Tools (CXT)	2020-01
CXT-1122 CXT-1149 CXT-1173 CXT-1218 CXT-1219 CXT-1250	FragmentCreator's graphical interface is currently being revised. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): FragmentCreator: Design revision".	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-02
CXT-1158	When the "FirstSpirit CXT FragmentCreator - CaaS Integration" is being used, all affected fragments are now transferred correctly to the required CaaS instance again, even if they have been generated for each schedule by the server.	FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FirstSpirit Content as a Service (CaaS), Tasks	2020-02
CXT-1162	A problem relating to the FirstSpirit Omnichannel Manager, the `WE_API.Preview.reload` method, and the `WE_API.Preview.getWindow` method (FirstSpirit Developer API, `de.espirit.firstspirit.webedit.client.api` package, `Preview` interface) has been resolved.	FirstSpirit API, Omnichannel Manager / Third Party Preview	2020-02
CXT-1189	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules	2020-01
CXT-1200	The new (chromium-based) version of Microsoft Edge should be used for working with FirstSpirit. Further information can be found in chapter "General: Browser compatibilities (Micorsoft Edge)".	Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-01

ID	Description	Categories	Version
CXT-1205	The Media DAP now allows the type-dependent upload of files. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Media DAP: Type-dependent upload of files".	Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Media, Modules	2020-04
CXT-1206	Information for analysis and debugging Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Error analysis for MicroApp Framework".	Debugging, Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT)	2020-04
CXT-1209	Update of internally used software Further information can be found in chapter "System: Update of internally used software".	FirstSpirit Administrator, Integrated software	2020-02
CXT-1215	In rare cases, creating a new variant or starting workflows had to be executed twice in succession.	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Workflow	2020-02
CXT-1221	Language selection lists now have a scroll bar if there are large numbers of entries.	ContentCreator, Editor, Languages	2020-04
CXT-1224	With the current release the information on the current project, the FirstSpirit version used, and the FirstSpirit license can be copied again in the new ContentCreator using the "Copy to clipboard" icon. Previously an error message of the type `UncaughtTypeError` could occur when using Google Chrome.	ContentCreator, Developer, Editor	2020-03
CXT-1243	FragmentCreator: Revision of the design (input fields) Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): FragmentCreator: Revision of the design (input fields)".	Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components	2020-05

ID	Description	Categories	Version
CXT-1245	With the current release, the handling of project selection has been improved. Further information can be found in chapter "ContentCreator: Project selection".	ContentCreator, Editor, FirstSpirit home page, Project switch	2020-03
CXT-1262	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules	2020-02
CXT-1268 CXT-1473	In the current release, MicroApps can now be integrated directly into HTML pages using the MicroApps API ("widget"). Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Extension of MicroApps API: Embedding of MicroApps into a page ("widget")".	Developer, FirstSpirit API, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Module development	2020-05
CXT-1290	The security of the MicroApp Framework and the FragmentCreator has been increased, particularly with regard to cross-site request forgery attacks.	Editor, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Security	2020-03
CXT-1296	In rare cases, pages that were newly created in the ContentCreator were not displayed directly in the preview.	ContentCreator, Editor	2020-03
CXT-1300	`WE_API.Common.getDisplayLanguage()` (interface `Common`, package `de.espirit.firstspirit.webedit.client.api`, FirstSpirit Developer API) now also returns the display language in the new ContentCreator (instead of previously *undefined*).	ContentCreator, Developer, Languages	2020-03

ID	Description	Categories	Version
CXT-1301	The language display in the new ContentCreator has been corrected (interface language).	ContentCreator, Developer, Editor, Workflow	2020-03
CXT-1302 CXT-803 CXT-1297	The input component CMS_INPUT_DATE is used to enter the date and/or time. Display and persistence in the FragmentCreator now work correctly. For example, the `mode` is now correctly taken into account and the selected time is now also displayed.	Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components	2020-03
CXT-1306 CXT-1495 CXT-1549	Some improvements have been made in the FragmentCreator media upload for the user: For example, the maximum file size that should be allowed for a media upload has now been increased to 64 MB, and the size can be configured by the administrator. In addition, files can now be selected via file selection dialog and drag-and-drop of images from another browser window/tab is possible. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): FragmentCreator: Improved handling for media upload".	Developer, Editor, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Media, Media Store	2020-08
CXT-1308	Improved behavior of the new ContentCreator when using modules with incompatible libraries.	ContentCreator, Developer, FirstSpirit API, Module development	2020-03
CXT-1315	Browser compatibilities Further information can be found in chapter "Administration: Browser compatibilities".	ContentCreator, Developer, Editor, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-05

ID	Description	Categories	Version
CXT-1319	The standard workflow for deleting fragments and variants – "Request deletion" – now once again removes entire fragments from live CaaS instances rather than just individual variants. In existing projects which use this workflow, the relevant script must be updated accordingly. You will find this in the latest version of the blueprint project.	FirstSpirit Content Experience Tools (CXT), FirstSpirit Content as a Service (CaaS)	2020-04
CXT-1330	In rare cases no preview image was displayed after dragging and dropping images into the corresponding dialog of FS_INDEX (error message: `Refused to load the image 'blob:http:// host.meinserver.de:8050/2796eb8b-643e-4791- bc37-42140696915b' because it violates the following Content Security Policy directive: "img-src 'self' data:").`	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Media	2020-04
CXT-1334	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules	2020-03
CXT-1338	The display of buttons in dialogs in the "new" FirstSpirit ContentCreator has been optimized so that labels in dialogs for switching on workflows are now easier to read.	ContentCreator, Developer, Editor	2020-04
CXT-1339	Nested frames are now prevented in the new ContentCreator: Instead of another instance of ContentCreator being opened in an existing ContentCreator window (due to a corresponding template configuration in the project), an empty page is now displayed. (Error message in the browser console: `(index):35 Skipping startup of nested ContentCreator, please contact your administrator.`)	ContentCreator, Developer, Editor	2020-05

ID	Description	Categories	Version
CXT-1343	A bug has been fixed that could cause unwanted flickering of icons in the new ContentCreator in rare cases (e.g. bookmark icon in the menu bar, icons in the image cropping dialog.	ContentCreator, Developer, Editor	2020-04
CXT-1354	LiveEdit now works again with the "new" ContentCreator.	ContentCreator, Developer, LiveEdit	2020-04
CXT-1359	An error in the browser "Chromium" in combination with FirstSpirit Omnichannel Manager could lead to drag-and-drop not being possible in the new ContentCreator (e.g. when moving menu levels via "ACTIONS / Edit navigation").	ContentCreator, Editor, FirstSpirit Omnichannel Manager	2020-03
CXT-1360	Indents/outdents in lists which are generated in FragmentCreator in the input element for formatted text entries (FS_MARKDOWN input component) using the Tab or the Shift + Tab are now saved permanently.	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-07
CXT-1376	When uploading files by drag-and-drop in the new ContentCreator (via FS_REFERENCE), the MIME type is now correctly recognized by the file extension	ContentCreator, Editor, Media	2020-04
CXT-1378	The MicroApp Framework now works even more stably with native DataAccess plug-ins which are included in the standard scope of delivery of FirstSpirit, such as the *DatasetDataAccessPlugin*.	Data Access Plug-In, Developer, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-04
CXT-1405	In addition to the preview, the FirstSpirit ContentCreator now offers navigation and editing options via a structure tree. Further information can be found in chapter "ContentCreator: New navigation view in the ContentCreator combines hierarchical structure display, context information and editing options".	ContentCreator, Editor	2020-04
CXT-1413	In rare cases, data could not be edited because of timing problems. Instead, a loading animation was displayed.	ContentCreator, Editor	2020-05

ID	Description	Categories	Version
CXT-1419	It is now possible to configure the `Content-Security-Policy` header. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Configuration of the Content-Security-Policy header".	FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT)	2020-04
CXT-1421 CXT-1446	After having made modifications to pages, the view is now directly updated again, so that, among other things, the correct status (editing status/work flow) in combination with FirstSpirit Omnichannel Manager is displayed again.	ContentCreator, Developer, Editor, FirstSpirit Administrator, Omnichannel Manager / Third Party Preview, Workflow	2020-04
CXT-1436	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules	2020-04
CXT-1464	Although thumbnails/preview images are displayed for images in the Upload dialog for media or in the FS_INDEX input component, for example, a standardized placeholder image is now used for displaying files (e.g., PDF, HTML, TXT).	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components, Media	2020-07
CXT-1465	Workflows can be started and advanced with the flyout menu via the element state in the right-hand menu bar. Due to a timing problem, an incomplete or outdated view was sometimes displayed. The available workflow actions are now displayed directly, without the need to manually update the page.	ContentCreator, Editor	2020-05

ID	Description	Categories	Version
CXT-1466	No datasets could be created or edited via FS_INDEX with *DatasetDataAccessPlugin* if this input component was used in a metadata form. (Error message: `...java.lang.IllegalArgumentException: Unknown language: §...`)	Content Store, ContentCreator, Data Access Plug-In, Developer, Editor, Input Components, Template Development	2020-05
CXT-1477	FragmentCreator: Design revision (status bar) Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): FragmentCreator: Design revision (status bar)".	Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator	2020-07
CXT-1482	For security reasons, access to the error page for the MicroApp Framework introduced in FirstSpirit 2020-04 (accessible via `~/status`) is now limited – as intended – solely to FirstSpirit administrators. Furthermore, the error pages now contain more information on the cause of the error.	Debugging, Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), Security	2020-05
CXT-1483	When the "BasicWorkflows" (`basicworkflows.fsm` module) were in use, in ContentCreator it was not always possible to open all the navigation view elements in the preview (error message in the browser console: `CxtElement request cc-tree/getChildren failed: null`, **serverseitig:** `java.lang.UnsupportedOperationException: de.espirit.firstspirit.store.access.pagestore`): `unique identifier not supported!`).	ContentCreator, Developer, Editor, Workflow	2020-06
CXT-1490	In the new ContentCreator, the user prompting/UX for the input elements "check box" (CMS_INPUT_CHECKBOX), "radio button" (CMS_INPUT_RADIOBUTTON) and "toggle" (CMS_INPUT_TOGGLE) have been improved. Further information can be found in chapter "ContentCreator: User prompting/UX: Display of check boxes, radio buttons and toggles".	ContentCreator, Developer, Editor, Input Components	2020-05

ID	Description	Categories	Version
CXT-1503	The preview is now reloaded automatically following changes to the structure of the navigation view (moving pages) so that the changes are visible immediately in the preview. Previously the preview had to be updated manually in some cases.	ContentCreator, Editor	2020-08
CXT-1525	FirstSpirit Content Experience Tools: Latest module versions

Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions". | Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules | 2020-05 |
| CXT-1528 CXT-1500 | Temporarily, when using the "FirstSpirit CXT FragmentCreator - CaaS Integration" module when releasing fragments, the content was no longer transferred to live CaaS instances, but only to preview CaaS instances. | Developer, FirstSpirit Content as a Service (CaaS), FragmentCreator, Release | 2020-05 |
| CXT-1547 | The "hidden" attribute is now also correctly evaluated in the metadata for corresponding input components in the FragmentCreator. | Developer, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components | 2020-08 |
| CXT-1558 | The input component for switching between two states – CMS_INPUT_TOGGLE – is now also supported in FragmentCreator.

Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): New input component for switching between two states: CMS_INPUT_TOGGLE". | Developer, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components | 2020-09 |
| CXT-1560 | In the FS_INDEX selection dialog, entries (media or fragments) are now sorted according to change date, unless another explicit search term has been specified. | Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components | 2020-07 |

ID	Description	Categories	Version
CXT-1561	The navigation view no longer shows elements for which the user does not have "Visible" permissions. Further information can be found in chapter "ContentCreator: Navigation view and evaluation of permissions: Hiding invisible elements".	ContentCreator, Developer, Editor, Permissions	2020-06
CXT-1571	In the case of external references, opening the navigation view in ContentCreator may have resulted in a *NullPointerException* (error message: `de.espirit.firstspirit.webedit.server.report` `$null` `$2(ContextBarReferenceProvider.java:88)`).	ContentCreator, Editor	2020-07
CXT-1572	New input component for the input of numerical values: CMS_INPUT_NUMBER Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): New input component for the input of numerical values: CMS_INPUT_NUMBER".	Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Input Components	2020-08
CXT-1596	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules, ServerManager	2020-06
CXT-1603	The language in which the preview page is displayed could not be changed in the preview itself after opening the navigation view (e.g. via a language switch on the preview page).	ContentCreator, Editor, Languages	2020-06

ID	Description	Categories	Version
CXT-1607	The status page under `~/status` now lists all currently registered MicroApps with accessibility, name and URL. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Status page shows registered MicroApps".	Debugging, Developer, FirstSpirit Content Experience Tools (CXT)	2020-08
CXT-1622	Support for Microsoft Internet Explorer has ceased. Further information can be found in chapter "General: Cessation of support for Microsoft Internet Explorer".	ContentCreator, Developer, Editor, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Module development, ServerMonitoring	2020-07
CXT-1627	HTTP requests with verbs other than *GET* were previously answered with a generic 500 message in the event of an error. The correct error code is now being provided.	Debugging, Developer, FirstSpirit Content Experience Tools (CXT), Module development	2020-07
CXT-1628	Update of internally used software Further information can be found in chapter "System: Update of internally used software".	FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Integrated software	2020-07
CXT-1640	The registration for MicroApps now reacts more flexibly to URLs with and without final slash.	Developer, FirstSpirit Content Experience Tools (CXT), Module development	2020-07

ID	Description	Categories	Version
CXT-1654	Deleted pages can now be conveniently restored in FirstSpirit ContentCreator via the status menu Further information can be found in chapter "ContentCreator: Restoring deleted pages".	ContentCreator, Editor	2020-08
CXT-1672	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules, ServerManager	2020-07
CXT-1678	The `ComparisonDialogOperation` interface (`de.espirit.firstspirit.webedit.server` package, FirstSpirit Developer API) is now available again in FirstSpirit ContentCreator. This can be used to activate the version comparison dialog in ContentCreator via the API.	ContentCreator, Developer, FirstSpirit API, Module development, Version history	2020-07
CXT-1682	If a large number of groupings (CMS_GROUP) are defined for a form, navigation using the arrows now works again in all cases.	ContentCreator, Editor, Input Components	2020-09
CXT-1691	In some cases, error messages displayed in FirstSpirit ContentCreator were not meaningful enough. Now the error text shows the actual cause of the error.	ContentCreator, Debugging, Developer, Editor	2020-08
CXT-1717	When using the module "FirstSpirit CXT FragmentCreator - CaaS Integration" , an `IllegalStateException` could occur when deploying fragments to the CaaS instance (error message: `ERROR 15.07.2020 06:05:06.322{dNR=}` `(de.espirit.firstspirit.server.scheduler.Scr:` `error during script execution :` `java.lang.IllegalStateException: not` `connected`) and no data was transferred. This problem has been fixed.	FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FirstSpirit Content as a Service (CaaS)	2020-07

ID	Description	Categories	Version
CXT-1746	Uploading media to directory structures using drag-and-drop Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Media DAP: Uploading directory structures".	Developer, Editor, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Media, Media Store	2020-09
CXT-1749	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules, Server Administrator, ServerManager	2020-08
CXT-1755	The sessions in the DAP Bridge are now closed correctly when logging out of FragmentCreator.	FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules, Remote access, Sessions	2020-09
CXT-1757	As of the current release, the attribute `SameSite` of the CSRF token can be set using the parameter `cxt.platform.csrf-samesite`. Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): "SameSite" attribute for cookies".	Developer, FirstSpirit Administrator, FirstSpirit Content Experience Tools (CXT), FragmentCreator, Modules, Security	2020-08
CXT-1760	Entries from the *WebeditStatusNotePlugins* (FirstSpirit Developer API, package `de.espirit.firstspirit.webedit.plugin.status` are now displayed again in the status menu.	ContentCreator, Developer, FirstSpirit API, Module development	2020-09

ID	Description	Categories	Version
CXT-1773	Validations in link templates within a CMS_INPUT_LINK input component now function as expected in FirstSpirit ContentCreator.	ContentCreator, Developer, Dynamic Forms, Editor, Input Components	2020-09
CXT-1786	In rare cases, tooltips were not hidden in FirstSpirit ContentCreator when the mouse pointer was moved to a different area. This problem has been rectified.	ContentCreator, Editor	2020-09
CXT-1793	Previously, when using the "Undo" or "Restore" function in the menu bar, the function "Display differences" (page state / "Display all changes") or when resetting a section via the `ComparisonDialogOperation` interface (`de.espirit.firstspirit.webedit.server` package, FirstSpirit Developer API), the associated page was erroneously not locked for editing and the page was not marked as changed.	ContentCreator, Developer	2020-09
CXT-1800	Fixed update problems in the tree view of the Navigation view.	ContentCreator, Editor	2020-09
CXT-1832	FirstSpirit Content Experience Tools: Latest module versions Further information can be found in chapter "FirstSpirit Content Experience Tools (CXT): Latest module versions".	Developer, FirstSpirit Administrator, FirstSpirit Content as a Service (CaaS), FragmentCreator, Modules	2020-09
CXT-1837	When cropping the original resolution using the ContentCreator's media management ("Edit image section"), incorrect cropping and error messages could occur when saving.	ContentCreator, Editor, Media, Media Store	2020-09

ID	Description	Categories	Version

DEVEX-140	The current FSDevTools release (release 2.5.10) supports the execution of project and server schedules by means of the "schedule start" command. The parameters required for execution are transferred via the command line using the following schema:
	
	`fs-cli [-p|-project <ProjectName>] schedule start [-n|-name <scheduleName>]`
	
	Additionally, the "schedule list" command can be used to list all the server schedules or project schedules for a certain project:
	
	`fs-cli [-p|-project <ProjectName>] schedule list`
	
	Parameters:
	
	`-project`: (optional) project name (short form `-p`)
	
	    ■ If the parameter is not specified, a server schedule is started.
	
	    ■ If the parameter is transferred with a valid project name, a project schedule for this project is started.
	
	`-name`: (mandatory parameter) schedule name (short form: `-n`)
	
	    ■ If the parameter is transferred with a valid schedule name (`ScheduleEntry#getName`), the schedule is executed.
	
	Example: `bin\fs-cli -h example.com -port 4242 -p "SmartLiving" schedule start -n test` executes the "test" schedule for the "SmartLiving" project.
	
	Information about the schedule and the execution duration is provided in the log.
	
	Note: The command line tool should not be exited before the schedule is finished.
		Developer, FSDevTools	2020-06

Information about the individual commands is available via the `fs-cli help <command>` call.

ID	Description	Categories	Version
DEVEX-378	When using "FirstSpirit External Synchronization", an additional mapping file (`Mapping.xml`) can be added to the export file using an option in the ExportOperation.SchemaOptions interface (`setExportGidMapping=true`) for the purpose of exporting database content (schemas, entities). This mapping file stores the database names of the source project plus FirstSpirit's proprietary unique identifiers (UUIDs). The command line tool FSDevTools supports the new option as of version 2.6. Without a mapping file (default behavior when using "FirstSpirit External Synchronization"), the dbnames of the column names and tables are optimized in line with the database during synchronization, as before. In standard application cases (without the additional `Mapping.xml` file), this ensures that the schema is compatible with the target database.	Database, Developer, External synchronization	2020-07
DEVEX-395	As of FSDevTools release 2.5.10, the execution of project and server schedules by means of the `schedule start` command is supported. The execution is started via the command line using the following schema: `fs-cli [-p|-project <ProjectName>] schedule start [-n|-name <scheduleName>]` In order to correctly detect incorrect executions, the `RunState` of the schedule is now also evaluated following execution (ABORTED / ERROR; FINISHED_WITH_ERRORS; SUCCESS) and logged accordingly. The changes are provided with **FSDevTools release 2.6.2**.	Developer, FSDevTools	2020-09