

Nuxeo Issue Tracker

JQL Query: project = "NXP" AND resolution = Fixed AND fixVersion = "10.3" AND ("Impact type" = "API change" OR "Upgrade notes" is not EMPTY) ORDER BY component ASC, key ASC

Sorted by: Component/s ascending, then Key ascending

1–54 of 54 as at: 11/12/18 16:36

| Key | Summary | Impact type | Upgrade notes |
|-----------|---|----------------------|--|
| NXP-25110 | Allow enabling GWT annotations | | <p>The GWT annotations are enabled when using the old JSF preview that can be enabled nuxeo.old.jsf.preview property to true.</p> <pre><extension target="org.nuxeo.runtime.ConfigurationService" point="configurati <property name="nuxeo.old.jsf.preview">true</property> </extension></pre> |
| NXP-24734 | Stateless Authentication based on a JSON Web Token (JWT) | Configuration Change | <p>There is a new Java API to acquire a JWT token to authorize a user:</p> <pre>JWTService service = Framework.getService(JWTService.class); String token = service.newBuilder().build();</pre> <p>The builder can also be used to add specific claims in the token (only CLAIM_SUBJECT to the authenticator for now) and a TTL.</p> <p>The token should then be propagated and passed by a third-party service wishing to do using the Authorization: Bearer <token> request header. As a compatibility fall request query parameter access_token can also be used.</p> <p>IMPORTANT</p> <p>For the JWTService to work, all the cluster servers must be configured with a common in nuxeo.conf:</p> <pre>nuxeo.jwt.secret=...</pre> |
| NXP-25062 | Expose create and update flags through contribution for UserMapperBasedResolver | | <p>Two new parameters are now available when configuring SAML authentication plugin:</p> <ol style="list-style-type: none"> userResolverCreateIfNeeded to create the user if it does not exist in the repository value is true) userResolverUpdate to update the user if present in the repository (default is va <p>When set to true, both parameters require a user directory that is not read-only.</p> |
| NXP-25887 | Allow PORTAL_AUTH to use a different digest than MD5 | Configuration Change | <p>To change the digest algorithm used by PORTAL_AUTH, use a contribution like:</p> <pre><require>org.nuxeo.ecm.platform.ui.web.auth.WebEngineConfig</require> <require>org.nuxeo.ecm.platform.login.Portal</require> <extension target="org.nuxeo.ecm.platform.ui.web.auth.service.PluggableAuthenticationSer point="authenticators"> <authenticationPlugin name="PORTAL_AUTH"> <loginModulePlugin>Trusting_LM</loginModulePlugin> <parameters> <parameter name="secret">...</parameter> <!-- shared secret between t and Nuxeo server --> <parameter name="maxAge">...</parameter> <!-- in seconds --> <parameter name="digestAlgorithm">SHA-512</parameter> </parameters> </authenticationPlugin> </extension></pre> <p>The algorithm used must be one of those described in https://docs.oracle.com/javase/8/docs/technotes/guides/security/StandardNames.html#</p> <p>For compatibility with previous Nuxeo versions, the default is MD5.</p> |
| NXP-25975 | Transform OAuth 1 pre-filter into a regular Authentication plugin | Configuration Change | <p>To activate OAuth 1, an <authenticationChain> must be defined to include <plugin>OAUTH1_AUTH</plugin>.</p> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|---|----------------------------------|--|
| NXP-25598 | Do not remove attachments from Document when ignore Blob property in PUT requests | | <p>Since 10.3, it is now possible to update blob(s) through the REST API when PUTting a c Nuxeo reads the JSON object for a blob property and tries to resolve a blob from the dc in the data field.</p> <p>The format of the JSON object is:</p> <pre>{ "data": "http://SERVER/nuxeo/nxfile/repository/docid/xpath" }</pre> <p>All other fields are ignored.</p> <p>Nuxeo does the following when trying to resolve a blob:</p> <ul style="list-style-type: none"> fetch the document fetch the referenced blob property check the download permissions <p>If one of the above fails (null, not found, no permission, ...), you get back a Bad Requ (400) when saving the document.</p> <p>On 9.10-HF19, it is not enabled by default to avoid any behavior changes. To enable it, contribute the following configuration property:</p> <pre><require>org.nuxeo.ecm.core.automation.core.properties</require> <extension point="configuration" target="org.nuxeo.runtime.ConfigurationSer <property name="nuxeo.automation.object.blob.decoder.enabled">true</propert </extension></pre> |
| NXP-25075 | Add new Nuxeo AWS service to get credentials and other config | API change, Configuration Change | <p>A new template <code>aws</code> is available, to define the AWS configuration. When activated (whi automatically done by the <code>marketplace-amazon-s3</code> package), the following nuxeo. properties are available:</p> <ul style="list-style-type: none"> <code>nuxeo.aws.accessKeyId</code> <code>nuxeo.aws.secretKey</code> <code>nuxeo.aws.region</code> <p>They are optional, and if not present the default AWS SDK mechanism for configuring tl used (environment variables, Java system properties, local AWS profile, container-spec configuration (ECS/EC2)). See https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/credentials.html#credentials-default and https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/java-dg-region-selection.html#default-region-provider-chain for information.</p> <p>Instead of using the template, the service can also be configured manually:</p> <pre><extension target="org.nuxeo.runtime.aws.AWSConfigurationService" point="configuration"> <configuration> <accessKeyId>MY_ACCESS_KEY_ID</accessKeyId> <secretKey>MY_SECRET_KEY</secretKey> <region>MY_REGION</region> </configuration> </extension></pre> <p>From Java there are two new APIs to get this information:</p> <ul style="list-style-type: none"> <code>NuxeoAWSCredentialsProvider.getInstance()</code> <code>NuxeoAWSRegionProvider.getInstance().getRegion()</code> <p>These credentials and region providers will default to standard AWS SDK behavior if no configuration is provided (see above).</p> |
| NXP-25525 | Expose the "pathStyleAccess" s3 client option within a new configuration key | Configuration Change | <p>To configure path-style access, use the configuration property:</p> <pre>nuxeo.s3storage.pathstyleaccess=true</pre> <p>Or if configuring from an extension point:</p> <pre><extension target="org.nuxeo.ecm.core.blob.BlobManager" point="configuratic <blobprovider name="default"> <class>org.nuxeo.ecm.core.storage.sql.S3BinaryManager</class> <property name="pathstyleaccess">true</property> ... </blobprovider> </extension></pre> |
| NXP-25540 | Use more secure encryption algorithm for AESBinaryManager | Data Persistence Change | <p>Before Nuxeo 10.3, the encryption used was based on <code>AES/CBC/PKCS5Padding</code> which found to be insecure (susceptible to padding oracle attacks).</p> <p>Starting with Nuxeo 10.3, a more secure encryption algorithm, <code>AES/GCM/NoPadding</code>, i you're migrating data from an earlier version and therefore still need to use the old inse algorithm, you must add the option <code>useInsecureCipher=true</code> to the <code>nuxeo.core.binarymanager_key</code> in <code>nuxeo.conf</code>.</p> <p>See https://doc.nuxeo.com/nxdoc/implementing-encryption/#filesystem-aes-encryption</p> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|--|-------------|---|
| NXP-25910 | Use NuxeoPrincipal instead of Principal in most APIs | API change | <p>The general upgrade rule is the Nuxeo code should not deal with a <code>java.security.Principal</code> anymore (except for internal authentication-related classes).</p> <p>API changes</p> <p>NuxeoPrincipal is now used instead of Principal in the methods and constructors classes:</p> <ul style="list-style-type: none"> ▪ CoreSession ▪ DocumentEventContext ▪ EventContext ▪ EventContextImpl ▪ InlineEventContext ▪ OperationContext ▪ QueryFilter ▪ SQLQuery.Transformer ▪ SecurityPolicy ▪ SecurityPolicy.QueryTransformer ▪ SecurityService ▪ TrashService ▪ UnboundEventContext ▪ UserWorkspaceService ▪ WebContext <p>Removed APIs</p> <p>These APIs were previously deprecated or internal and have been removed.</p> <ul style="list-style-type: none"> ▪ CoreInstance.getInstance() ▪ CoreInstance.getNumberOfSessions() ▪ CoreInstance.getRegistrationInfos() ▪ CoreInstance.openCoreSession(repositoryName, Map context) ▪ CoreInstance.openCoreSession(repositoryName, Principal principal) ▪ DetachedNuxeoPrincipal ▪ SimplePrincipal (use UserPrincipal instead) |
| NXP-25543 | Relocate nuxeo-platform-collections in nuxeo-service | | <ul style="list-style-type: none"> ▪ Collection Core types (Collection, Favorites, etc.), schema and facets have been merged into <code>CoreExtensions.xml</code> ▪ nuxeo-platform-collections-jsf has disappeared. Resources and classes have been moved to nuxeo-jsf/nuxeo-platform-webapp-base and nuxeo-dm/nuxeo-platform-webapp-dm ▪ All Collection-related automation operations have been moved to nuxeo-automation ▪ Collection and Favorites services have been kept but moved under nuxeo-services/collections ▪ UserWorkspaces service now contributes where should be located User's collections by implementing <code>CollectionLocationService.java</code> ▪ You can use <code>CollectionFeature.java</code> to test a feature relying on Collections, need to add the following pom dependency: <pre><dependency> <groupId>org.nuxeo.ecm.platform</groupId> <artifactId>nuxeo-platform-collections-core</artifactId> <type>test-jar</type> <scope>test</scope> </dependency></pre> |
| NXP-25429 | Handle Comment migration | | <p>When upgrading from a previous Nuxeo version, if you want to keep (and later migrate) comments, you need to add the contribution:</p> <pre><require>org.nuxeo.ecm.platform.comment.manager.migrator</require> <extension target="org.nuxeo.runtime.migration.MigrationService" point="configuration"> <migration id="comment-storage"> <defaultState>relation</defaultState> </migration> </extension></pre> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|---|----------------------|--|
| NXP-25840 | DWG file upload generates 500 error | | <p>Since 9.10-HF22</p> <p>A new configuration property <code>nuxeo.preview.allowGenericHTMLConverter</code> is available to disable the generic HTML converter used by the preview when no converter is found to handle the input blob. Disable it if you run into an issue when previewing formats that LibreOffice can't handle. Defaults to <code>true</code> to keep backward compatibility.</p> <p>To disable the generic HTML converter, use the following contribution:</p> <pre><require>org.nuxeo.ecm.platform.preview.properties</require> <extension point="configuration" target="org.nuxeo.runtime.ConfigurationService" <property name="nuxeo.preview.allowGenericHTMLConverter">false</property> </extension></pre> |
| | | | <p>Since 10.3</p> <p>When calling a converter directly through its name, a check is done on its source MIME type to ensure it can handle the input blob. A <code>ConversionException</code> is thrown if the converter can't handle the input blob. A new configuration property <code>nuxeo.convert.enforceSourceMimeTypeCheck</code> is available to disable this behavior which defaults to <code>true</code>.</p> <p>To disable the check, use the following contribution:</p> <pre><require>org.nuxeo.ecm.core.convert.configuration</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configuration" <property name="nuxeo.convert.enforceSourceMimeTypeCheck">false</property> </extension></pre> |
| NXP-24725 | Annotation type should extend Comment type | | <p>A new facet <code>ExternalEntity</code> has been created to handle Annotations or Comments created by an external service like <code>ARender</code>.</p> <p>This facet allows to store the serialized entity.</p> |
| NXP-25279 | Make the raw binary text available for processing | API change | <p>The <code>binaryTextUpdated</code> event now contains two properties of interest to know what was updated exactly:</p> <ul style="list-style-type: none"> ▪ <code>systemProperty</code> contains the name of the property updated ▪ <code>systemPropertyValue</code> contains the value |
| NXP-25716 | Simplify fulltext extraction | Configuration Change | <p>Fulltext maximum size is now 128 KB by default. To change this, the repository configuration must be updated to use another <code>fieldSizeLimit</code> value:</p> <pre><fulltext ... fieldSizeLimit="1048576"> ... </fulltext></pre> <p>A value of 0 means no limit.</p> <p>See https://doc.nuxeo.com/nxdoc/repository-configuration/#full-text for more</p> |
| NXP-25761 | Fix Trash migration | | <p>The <code>org.nuxeo.isTrashed.from.deleteTransition</code> introduced in 10.2 has been replaced by <code>org.nuxeo.isTrashed.from.deleteUndeleteTransition</code>. If you have migrated your Nuxeo instance to use new dedicated <code>isTrashed</code> property, calls to <code>CoreSession#followTransition</code> with <code>delete/undelete</code> transition are forwarded to <code>deleteUndeleteTransition</code> without following the transition.</p> <p>Also, these transitions are deprecated as we DO NOT follow them anymore. As a consequence, on a migration or on a fresh instance, documents can't have <code>deleted</code> state anymore.</p> <p>As a consequence of this backward mechanism, following these transitions on proxy will not work.</p> |
| NXP-25617 | Update mongodb template to handle KeyValueStore and LockManager | | <p>Since 10.3 the mongodb template set the implementation of the <code>KeyValueStore</code> and <code>LockManager</code> to <code>Redis</code>.</p> <p>If you want to keep or use a <code>Redis</code> implementation you need to be explicit in your <code>nuxeo.properties</code>:</p> <pre>nuxeo.keyvalue.provider=redis nuxeo.lock.manager=redis</pre> |
| NXP-25928 | Repository LIKE queries should be anchored on MongoDB | Configuration Change | <p>For MongoDB queries, the LIKE queries are now anchored by default on master. This changes the behavior to the old behavior using:</p> <pre><require>org.nuxeo.ecm.core.storage.mongodb.configuration</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configuration" <documentation> Property to disable the fact that on MongoDB a LIKE match anchors its (like in SQL databases), i.e., absent wildcards, it does not match in the middle of a string. When true, does not match in the middle of a string. When false, works as if there was a % on each side of the LIKE expression. @since 10.3 </documentation> <!-- default is false for 9.10 and earlier --> <property name="nuxeo.mongodb.like.anchored">false</property> </extension></pre> <p>Conversely, this can be set to <code>true</code> on 9.10 if anchoring is required.</p> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|---|----------------------|--|
| NXP-26072 | Enable use of a custom MongoDB Client TrustStore for in-flight encryption | Configuration Change | <p>The following nuxeo.conf properties can be set to define appropriate TLS/SSL config MongoDB:</p> <ul style="list-style-type: none"> ▪ nuxeo.mongodb.ssl=true ▪ nuxeo.mongodb.truststore.path ▪ nuxeo.mongodb.truststore.password ▪ nuxeo.mongodb.truststore.type ▪ nuxeo.mongodb.keystore.path ▪ nuxeo.mongodb.keystore.password ▪ nuxeo.mongodb.keystore.type <p>If more fine-grained configuration is needed than properties, the following extension poi instead:</p> <pre><require>org.nuxeo.mongodb.connection</require> <extension target="org.nuxeo.runtime.mongodb.MongoDBComponent" point="conne <connection id="default"> ... <ssl>true</ssl> <trustStorePath>/path/to/cacerts.jks</trustStorePath> <trustStorePassword>changeit</trustStorePassword> <trustStoreType>jks</trustStoreType> <keyStorePath>/path/to/keystore.jks</keyStorePath> <keyStorePassword>changeit</keyStorePassword> <keyStoreType>jks</keyStoreType> </connection> </extension></pre> |
| NXP-17479 | Create MySQL fulltext table with InnoDB | | <p>Since Nuxeo 10.3, fulltext tables stored in MySQL/MariaDB now use InnoDB engine instead of MyISAM. InnoDB engine is more performant and allow to add foreign constraints on table.</p> <p>You can change your current engine by following this documentation.</p> <p>Note: As there weren't any foreign constraints on these tables, you must first purge table converting them to new InnoDB engine with the constraint.</p> <p>You need to run:</p> <pre>DELETE FROM `fulltext` WHERE id not in (SELECT id FROM hierarchy);</pre> |
| NXP-25219 | CSV Importer can't import Date properties with times | | <p>Dates in CSV files must be formatted using the W3C format. More information on the W3C documentation.</p> <p>It is possible to use the old legacy date format (without time information) MM/dd/yyyy by configuration property nuxeo.csv.import.legacyDateFormat to true:</p> <pre><require>org.nuxeo.ecm.csv.core.properties</require> <extension point="configuration" target="org.nuxeo.runtime.ConfigurationService" <property name="nuxeo.csv.import.legacyDateFormat">true</property> </extension></pre> |
| NXP-25208 | hardcoded blacklisted mime types list prevents customizations in nuxeo-diff | | <p>Deprecated ContentDiffHelper#getHtmlConversionBlackListedMimeTypes</p> <p>Added:</p> <pre>org.nuxeo.ecm.diff.content.MimeTypesDescriptor org.nuxeo.ecm.diff.content.MimeTypeDescriptor ContentDiffAdapterManager#getHtmlConversionBlacklistedMimeTypes()</pre> <p>Added the htmlConversionBlacklistedMimeTypes extension point to the org.nuxeo.ecm.diff.content.adapter.ContentDiffAdapterManagerComponent component:</p> <pre><extension-point name="htmlConversionBlacklistedMimeTypes"> <documentation> @since 10.10 Allows to contribute the list of blacklisted mime types for HTML conversion. By default, contributing a list merges its mime types with the existing ones. To remove a mime type, use enabled=false. <code> <extension target="org.nuxeo.ecm.diff.content.adapter.ContentDiffAdapterManagerComponent" point="htmlConversionBlacklistedMimeTypes"> <mimeTypes> <MimeType>application/msword</MimeType> <MimeType>application/rtf</MimeType> <MimeType enabled="false">application/pdf</MimeType> </mimeTypes> </extension> </code></pre> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|------------------------------------|-------------|--|
| NXP-19262 | Improved query API for directories | API change | <p>New directory query APIs using a QueryBuilder:</p> <ul style="list-style-type: none"> ▪ <code>Session.query(QueryBuilder, fetchReferences)</code> ▪ <code>Session.queryIds(QueryBuilder)</code> |
| NXP-19264 | Improved query API for UserManager | API change | <p>New UserManager query APIs using a QueryBuilder:</p> <ul style="list-style-type: none"> ▪ <code>UserManager.searchUsers(QueryBuilder)</code> ▪ <code>UserManager.searchGroups(QueryBuilder)</code> <p>New MultiTenantUserManager query APIs like above but with a context:</p> <ul style="list-style-type: none"> ▪ <code>MultiTenantUserManager.searchUsers(QueryBuilder, context)</code> ▪ <code>MultiTenantUserManager.searchGroups(QueryBuilder, context)</code> <p>New GroupComputer API using a QueryBuilder:</p> <ul style="list-style-type: none"> ▪ <code>GroupComputer.searchGroups(QueryBuilder)</code> |
| NXP-23863 | Switch logger backend to Log4j 2 | | <p>Nuxeo now uses Log4j 2 as its logging backend instead of Log4j. Two logging APIs are generally available in Nuxeo:</p> <ul style="list-style-type: none"> ▪ Commons Logging ▪ Log4j 2 API <p>SLF4J is still available, but Log4j 2 API are preferred by reason of its lambda support.</p> <p>Breaking change</p> <ul style="list-style-type: none"> ▪ <code>Log4jHelper.setQuiet</code> and <code>Log4jHelper.setDebug</code> was removed due to its u API which are now impossible with Log4j 2. These APIs were replaced by <code>Log4jHelper.setLevel</code>. Compared to previous implementation, <code>setLevel</code> API d appender's threshold, it just acts on logger's levels. ▪ <code>LogCaptureFeature.Filter</code> has also changed without backward compatibility because Log4j internal class. This has been replaced by corresponding Log4j 2 internal class. ▪ File rollover has changed. Files are now compressed using gzip and pattern has been to <code>NAME-YYYY-MM-dd.log.gz</code>. For instance: <ul style="list-style-type: none"> ▪ <code>server.log.2018-01-01</code> becomes <code>server-2018-01-01.log.gz</code> ▪ Watch log4j file for reload has been removed in favor of built-in logic in Log4j2. These properties are no longer read: <ul style="list-style-type: none"> ▪ <code>org.nuxeo.runtime.log4jwatch.disabled</code> ▪ <code>org.nuxeo.runtime.log4jwatch.delay</code> <p>See <i>Automatic Reconfiguration</i> in Log4j 2 documentation to have watch file be</p> <p>Code behavior change</p> <ul style="list-style-type: none"> ▪ LogCaptureFeature behavior has changed. Now log Result is cleaned between eac have an isolated result to assert for @Before/@After/test methods. ▪ You may need to add the slf4j dependency if you were using it in your source code: <pre><dependency> <groupId>org.slf4j</groupId> <artifactId>slf4j-api</artifactId> </dependency></pre> <p>New</p> <ul style="list-style-type: none"> ▪ LogCaptureFeature.FilterOn now accepts a logger class instead of a logger name ▪ You can now use Log4j 2 API in your code |

| Key | Summary | Impact type | Upgrade notes |
|-----------|--|----------------------|--|
| NXP-26074 | Enable use of a custom Elasticsearch RestClient Truststore | Configuration Change | <p>The following nuxeo.conf properties can be set to define appropriate TLS/SSL config Elasticsearch RestClient:</p> <ul style="list-style-type: none"> ▪ <code>elasticsearch.restClient.truststore.path</code> ▪ <code>elasticsearch.restClient.truststore.password</code> ▪ <code>elasticsearch.restClient.truststore.type</code> ▪ <code>elasticsearch.restClient.keystore.path</code> ▪ <code>elasticsearch.restClient.keystore.password</code> ▪ <code>elasticsearch.restClient.keystore.type</code> <p>The following properties are deprecated (they were misnamed and are actually referring trustStore, not the keyStore):</p> <ul style="list-style-type: none"> ▪ <code>elasticsearch.restClient.keystorePath</code> ▪ <code>elasticsearch.restClient.keystorePassword</code> ▪ <code>elasticsearch.restClient.keystoreType</code> <p>If more fine-grained configuration is needed than properties, the following extension point instead:</p> <pre><require>org.nuxeo.elasticsearch.defaultConfig</require> <extension target="org.nuxeo.elasticsearch.ElasticSearchComponent" point="elasticSearchClient"> <elasticSearchClient class="org.nuxeo.elasticsearch.client.ESRestClientFa ... <option name="trustStorePath">/path/to/cacerts.jks</option> <option name="trustStorePassword">changeit</option> <option name="trustStoreType">jks</option> <option name="keyStorePath">/path/to/keystore.jks</option> <option name="keyStorePassword">changeit</option> <option name="keyStoreType">jks</option> </elasticSearchClient> </extension></pre> |
| NXP-25791 | Make an Operation that launches Image Recompute | | Added the <code>Picture.RecomputeViews</code> operation for administrators. |
| NXP-26183 | New <code>KeyValueStoreProvider.keyStream(prefix)</code> API | API change | New API: <code>KeyValueStoreProvider.keyStream(prefix)</code> |
| NXP-24885 | Document Nuxeo Drive operations | API change | <p>Deprecated the following operations:</p> <pre>GetRepositoriesOperation NuxeoDriveAddToLocallyEditedCollection NuxeoDriveCanMove NuxeoDriveGenerateConflictedItemName NuxeoDriveGetClientUpdateInfo NuxeoDriveWaitForAsyncCompletion</pre> <p>Deprecated the following operation parameters:</p> <pre>NuxeoDriveGetChangeSummary#lastSyncActiveRootDefinitions</pre> |
| NXP-25797 | Drive error when uploading zip file to server | API change | <p>Added to <code>FileManager</code>:</p> <pre>DocumentModel createDocumentFromBlob(CoreSession documentManager, Blob input, path, boolean overwrite, String fullName, boolean noMimeTypeCheck, boolean excludeOneToMany) throws IOException;</pre> <p>to allow excluding "one-to-many" importers (ie. the ones creating more than one document given blob, typically <code>CSVZipImporter</code> or <code>ExportedZipImporter</code>) when selecting them. This is used by Drive to not try to explode a CSV import ZIP for instance.</p> <p>Added to <code>FileImporter</code>:</p> <pre>boolean isOneToMany();</pre> <p>to flag an importer as "one-to-many", ie. it creates more than one document for the given blob, typically <code>CSVZipImporter</code> or <code>ExportedZipImporter</code>.</p> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|--|----------------------|---|
| NXP-25844 | Remove usage of deprecated code and code deprecated since 7.10 or less in nuxeo-drive-server | API change | <p>Removed as deprecated since 7.10 or less:</p> <pre> FileSystemItemFactory#getDocumentByFileSystemId(String id, Principal principal) FileSystemItemManager#getSession(String repositoryName, Principal principal) NuxeoDriveManager#setChangeFinder(FileSystemChangeFinder changeFinder) NuxeoDriveActions#UPDATE_SITE_URL_PROP_KEY NuxeoDriveActions#SERVER_VERSION_PROP_KEY NuxeoDriveCreateFile#name NuxeoDriveGetChangeSummary#lastSyncDate NuxeoDriveGetTopLevelChildren </pre> <p>Removed as should have been deprecated since 7.10 or less, see NXP-14826:</p> <pre> FileSystemChangeFinder#getFileSystemChangesIntegerBounds(CoreSession session, lastActiveRootRefs, SynchronizationRoots activeRoots, Set<String> collectionSyncRootMemberIds, long lowerBound, long upperBound, int limit) FileSystemChangeFinder#getCurrentDate(); NuxeoDriveManager#getChangeSummaryIntegerBounds(Principal principal, Map<String, Set<IdRef>> lastSyncRootRefs, long lowerBound) </pre> <p>and changed</p> <pre> FileSystemChangeFinder#getFileSystemChanges(CoreSession session, Set<IdRef> lastActiveRootRefs, SynchronizationRoots activeRoots, long lastSuccessfulSync syncDate, int limit) NuxeoDriveManager#getChangeSummary(Principal principal, Map<String, Set<IdRef> lastSyncRootRefs, long lastSuccessfulSync) </pre> <p>to</p> <pre> FileSystemChangeFinder#getFileSystemChanges(CoreSession session, Set<IdRef> lastActiveRootRefs, SynchronizationRoots activeRoots, Set<String> collectionSyncRootMemberIds, long lowerBound, long upperBound, int limit) NuxeoDriveManager#getChangeSummary(Principal principal, Map<String, Set<IdRef> lastSyncRootRefs, long lowerBound) </pre> |
| NXP-26155 | Improve query-related classes: Predicates/QueryBuilder/AST | API change | <p>Predicates has many more useful static methods. OrderByList now has an empty constructor and a copy constructor. The QueryBuilder API has been clarified, and a QueryBuilder can internally hold a or OR-based MultiExpression. It can also hold a countTotal flag.</p> |
| NXP-26073 | Allow configuring Redis in SSL mode and with custom TrustStore | Configuration Change | <p>The following nuxeo.conf properties can be set to define appropriate TLS/SSL configuration for Redis:</p> <ul style="list-style-type: none"> ▪ nuxeo.redis.ssl=true ▪ nuxeo.redis.truststore.path ▪ nuxeo.redis.truststore.password ▪ nuxeo.redis.truststore.type ▪ nuxeo.redis.keystore.path ▪ nuxeo.redis.keystore.password ▪ nuxeo.redis.keystore.type <p>If more fine-grained configuration is needed than properties, the following extension point can be used instead:</p> <pre> <require>org.nuxeo.ecm.core.redis.config</require> <extension target="org.nuxeo.ecm.core.redis" point="configuration"> <server> ... <ssl>true</ssl> <trustStorePath>/path/to/cacerts.jks</trustStorePath> <trustStorePassword>changeit</trustStorePassword> <trustStoreType>jks</trustStoreType> <keyStorePath>/path/to/keystore.jks</keyStorePath> <keyStorePassword>changeit</keyStorePassword> <keyStoreType>jks</keyStoreType> </server> </extension> </pre> |

| Key | Summary | Impact type | Upgrade notes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------------|--|-------------|---|----------------|-----------|-----------------------|----------------------|----------------------|---------------------|------------------|-----------------|----------------------|---------------------|------------------------------|-----------------------------|---------------------|--------------------|----------------------|-------------------------|--------------------|-------------------|------------------|-----------------|----------------------|---------------------|---------------------------------|-------------------|---------------------|--------------------|-----------------------|------------------|---------------|--------------|---------------------|-----------------|--------------------------|---------------------|-----------------------|------------------|--------------------------|-------------------|
| NXP-25036 | Make REST API endpoints consume and produce 'application/json' as content type | API change | <p>The REST API now produces and consumes only <code>application/json</code> as content type</p> <ul style="list-style-type: none"> <code>application/json+nxentity</code> is now never returned as content type response; th not expect it as content type request. <code>application/json+nxrequest</code> content type should not be used anymore. It still deprecated server side. <code>application/json</code> should be used instead when POSTING automation. <code>application/json+esentity</code> which was never used was also removed. <p>Breaking change:</p> <ul style="list-style-type: none"> the <code>org.nuxeo.ecm.automation.jaxrs.io.documents.JsonESDocumentWr</code> in <code>nuxeo-automation-io</code> has moved to <code>org.nuxeo.elasticsearch.io.JsonESDocumentWriter</code>, located in <code>nuxeo-elasticsearch-core</code>. The <code>JsonESDocumentWriter</code> does not implement <code>anymc MessageBodyWriter<DocumentModel></code>, you may need to update your code. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NXP-26126 | Make the "blob" enricher a BlobProperty enricher | | <p>Enrichers can now apply to the <code>blob</code> type.</p> <p>For instance, to get the links to open a LiveConnect blob in all applications associated to type, you can use the <code>enrichers-blob: appLinks</code> enricher.</p> <p>Any blob property, e.g. <code>file:content</code>, will then be enriched in the following way:</p> <pre> { "file:content": { "name": "...", "mime-type": "...", ..., "appLinks": [{ "appName": "...", "icon": "...", "link": "..." }, ...] } } </pre> <p>Note: since such a blob enricher applies to a <code>BlobProperty</code>, the backing class should <code>AbstractJsonEnricher<BlobProperty></code> as in the example below:</p> <pre> public class BlobAppLinksJsonEnricher extends AbstractJsonEnricher<BlobProperty> { @Override public void write(JsonGenerator jg, BlobProperty blobProperty) throws IOException { ... } } </pre> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NXP-25402 | Handle Duration deserialisation in descriptor | | <p>You can now use <code>Duration</code> type into your descriptors.</p> <p>Nuxeo handles the JDK Duration format (PT10M, PT200S, PT0.500S) or a specific form hours, minutes, seconds and milliseconds (10m, 200s, 500ms).</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NXP-24766 | Introduce clock skew in SAML authenticator | | <p>This introduces a new <code>nuxeo.sam12.skewTimeMs</code> configuration property to control th in milliseconds. Default value is <code>60 * 1000</code> (1 minute).</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| NXP-22232 | Rename operations WebUI.* related to JSF | API change | <table border="1"> <thead> <tr> <th>Previous names</th> <th>New names</th> </tr> </thead> <tbody> <tr><td>WebUI.AddErrorMessage</td><td>Seam.AddErrorMessage</td></tr> <tr><td>WebUI.AddInfoMessage</td><td>Seam.AddInfoMessage</td></tr> <tr><td>WebUI.AddMessage</td><td>Seam.AddMessage</td></tr> <tr><td>WebUI.ClearClipboard</td><td>Seam.ClearClipboard</td></tr> <tr><td>WebUI.ClearSelectedDocuments</td><td>Seam.ClearSelectedDocuments</td></tr> <tr><td>WebUI.ClearWorklist</td><td>Seam.ClearWorklist</td></tr> <tr><td>WebUI.ShowCreateForm</td><td>Seam.CreateDocumentForm</td></tr> <tr><td>WebUI.DownloadFile</td><td>Seam.DownloadFile</td></tr> <tr><td>WebUI.NavigateTo</td><td>Seam.NavigateTo</td></tr> <tr><td>WebUI.AddToClipboard</td><td>Seam.AddToClipboard</td></tr> <tr><td>WebUI.PushDocumentToSeamContext</td><td>Seam.PushDocument</td></tr> <tr><td>WebUI.AddToWorklist</td><td>Seam.AddToWorklist</td></tr> <tr><td>WebUI.RaiseSeamEvents</td><td>Seam.RaiseEvents</td></tr> <tr><td>WebUI.Refresh</td><td>Seam.Refresh</td></tr> <tr><td>WebUI.SetJSFOutcome</td><td>Seam.SetOutcome</td></tr> <tr><td>WebUI.DestroySeamContext</td><td>Seam.DestroyContext</td></tr> <tr><td>WebUI.InitSeamContext</td><td>Seam.InitContext</td></tr> <tr><td>WebUI.RunOperationInSeam</td><td>Seam.RunOperation</td></tr> </tbody> </table> | Previous names | New names | WebUI.AddErrorMessage | Seam.AddErrorMessage | WebUI.AddInfoMessage | Seam.AddInfoMessage | WebUI.AddMessage | Seam.AddMessage | WebUI.ClearClipboard | Seam.ClearClipboard | WebUI.ClearSelectedDocuments | Seam.ClearSelectedDocuments | WebUI.ClearWorklist | Seam.ClearWorklist | WebUI.ShowCreateForm | Seam.CreateDocumentForm | WebUI.DownloadFile | Seam.DownloadFile | WebUI.NavigateTo | Seam.NavigateTo | WebUI.AddToClipboard | Seam.AddToClipboard | WebUI.PushDocumentToSeamContext | Seam.PushDocument | WebUI.AddToWorklist | Seam.AddToWorklist | WebUI.RaiseSeamEvents | Seam.RaiseEvents | WebUI.Refresh | Seam.Refresh | WebUI.SetJSFOutcome | Seam.SetOutcome | WebUI.DestroySeamContext | Seam.DestroyContext | WebUI.InitSeamContext | Seam.InitContext | WebUI.RunOperationInSeam | Seam.RunOperation |
| Previous names | New names | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.AddErrorMessage | Seam.AddErrorMessage | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.AddInfoMessage | Seam.AddInfoMessage | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.AddMessage | Seam.AddMessage | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.ClearClipboard | Seam.ClearClipboard | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.ClearSelectedDocuments | Seam.ClearSelectedDocuments | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.ClearWorklist | Seam.ClearWorklist | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.ShowCreateForm | Seam.CreateDocumentForm | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.DownloadFile | Seam.DownloadFile | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.NavigateTo | Seam.NavigateTo | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.AddToClipboard | Seam.AddToClipboard | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.PushDocumentToSeamContext | Seam.PushDocument | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.AddToWorklist | Seam.AddToWorklist | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.RaiseSeamEvents | Seam.RaiseEvents | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.Refresh | Seam.Refresh | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.SetJSFOutcome | Seam.SetOutcome | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.DestroySeamContext | Seam.DestroyContext | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.InitSeamContext | Seam.InitContext | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WebUI.RunOperationInSeam | Seam.RunOperation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Key | Summary | Impact type | Upgrade notes |
|-----------|--|----------------------|---|
| NXP-25780 | Revise the implementation of <code>EditableModelImpl.getRowData</code> | | <p>Since 10.3 and 9.10-HF19, a new configuration property is exposed to allow an alternate management of a missing row. Use the following contribution to enable it:</p> <pre><require>org.nuxeo.ecm.platform.ui.web.configuration.default</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configuration" > <property name="nuxeo.jsf.skipMissingRow">true</property> </extension></pre> |
| NXP-26043 | Reset selection when navigating away | Configuration Change | <p>From 10.3, the list <code>CURRENT_SELECTION</code> is reset when switching between main tabs. In 9.10, the following contribution is needed to reset the selection:</p> <pre><require>org.nuxeo.ecm.webapp.documentsLists.DefaultDocumentsLists</require> <extension target="org.nuxeo.ecm.webapp.documentsLists.DocumentsListsService" point="list"> <documentsList name="CURRENT_SELECTION"> <events> <event>folderishDocumentSelectionChanged</event> <event>searchPerformed</event> <event>mainTabsChanged</event> </events> <isSession>false</isSession> </documentsList> </extension></pre> |
| NXP-25680 | Use a condition to protect Nuxeo from a CSRF attack from local filesystem | Configuration Change | <p>On 10.3, the behavior is still disallowed but will be allowed in dev mode.</p> <p>On 7.10, 8.10 and 9.10, the behavior is now again allowed by default. To prevent local from being allowed to POST, use the following contribution:</p> <pre><require>org.nuxeo.ecm.platform.ui.web.cors</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configuration" > <property name="nuxeo.cors.allowNullOrigin">false</property> </extension></pre> |
| NXP-25828 | Improve external permissions experience dealing with multiple items | | <p>Since 8.10-HF40 and 9.10-HF24</p> <p>A new configuration property <code>nuxeo.transient.username.unique</code> is available to control if a computed transient username should be unique no matter what base username is provided. It should be always the same for a given base username. Having always the same transient username for a given base username allows to generate a token for a given email when giving permission to an external user: that means if you invite an external user on 2 documents, he won't have to log out from the first document to see the second one. It defaults to <code>true</code> for backward compatibility.</p> <p>To disable the uniqueness of the transient username computation, use the following contribution:</p> <pre><require>org.nuxeo.ecm.core.api.properties</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configuration" > <property name="nuxeo.transient.username.unique">false</property> </extension></pre> <p>Since 10.3</p> <p>A computed transient username is now always the same for a given base username, so a token is generated for a given email when giving permission to an external user: that means if you invite the same external user on 2 documents, he won't have to log out from the first document to see the second one.</p> <p>To enable back the uniqueness of the transient username computation, use the following contribution:</p> <pre><require>org.nuxeo.ecm.core.api.properties</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configuration" > <property name="nuxeo.transient.username.unique">true</property> </extension></pre> |
| NXP-25035 | Add configuration property to be able to reduce tag sanitisation on faceted implementation | | <p>A new configuration property <code>nuxeo.tag.sanitization.enabled</code> has been added to enable/disable tag sanitization (true by default)</p> |
| NXP-25481 | Make the UI reflect the changes when a tag is sanitized | | <p>New global function <code>String.sanitize(boolean)</code> to cleanup tags</p> |
| NXP-24717 | Thumbnail should have a better resolution for non image files | API change | <p>In <code>ThumbnailDocumentConverter</code>, removed unnecessary constants:</p> <pre>THUMBNAI_ CONVERTER_NAME THUMBNAI_SIZE_PARAMETER_NAME // use ThumbnailConstants#THUMBNAI_SIZE_PARAMETER_NAME instead THUMBNAI_DEFAULT_SIZE // use ThumbnailConstants#THUMBNAI_DEFAULT_SIZE instead</pre> |
| NXP-25966 | Upgrade Restlet libraries to latest 2.4.1 | API change | <p>Some Restlet-related imports have slightly changed packages. The Restlet maven artifacts also have been renamed slightly.</p> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|--|----------------------|--|
| NXP-25385 | Review all Web UI dynamic and static resources http caching strategy | | <p>Note that <code>filterConfig <pattern></code> contributions now match the full path and query the URI (previously the query parameter was not included).</p> <p>Thumbnails, blobs and nxfile resources can have the doc changeToken appended as a for optimized http caching.</p> <pre><filterConfig name="cachedBPR" cached="true" private="true" cacheTime="31 transactional="true"> <!-- if url contains doc changeToken: approximately one year --> <pattern>\${org.nuxeo.ecm.contextPath}/.*@(blob preview rendition).*\?.*char </pattern> </filterConfig> <filterConfig name="cachednxfile" transactional="false" synchronize="false cached="true" private="true" cacheTime="31536000"> <!-- if url contains doc changeToken: approximately one year --> <!-- transaction is manually handled in DownloadServlet --> <pattern>\${org.nuxeo.ecm.contextPath}/nxfile.*\?.*changeToken=.*+</pat </filterConfig></pre> <p>Otherwise, we let eTag play.</p> |
| NXP-25903 | CSRF Token for Platform | Configuration Change | <p>Activation</p> <p>To activate CSRF Token verification, use the following configuration:</p> <pre><extension target="org.nuxeo.runtime.ConfigurationService" point="configur <property name="nuxeo.csrf.token.enabled">true</property> </extension></pre> <p>When this is activated, all clients accessing Nuxeo will need to get a token and provide requests that are not GET/HEAD.</p> <p>Note that even when the CSRF Token is not activated, other CSRF checks not using a token are still being done (using the Origin/Referer headers).</p> <p>Getting the token initially</p> <p>The client must use the following request with the header <code>CSRF-Token: fetch</code>:</p> <pre>GET /nuxeo CSRF-Token: fetch</pre> <p>The response will contain the token in the header:</p> <pre>200 OK CSRF-Token: uNTIwv3oEImb3singqJKSuJDNjM9ldVOjnwtxmFh</pre> <p>Passing the token</p> <p>Then on every request that is not a GET/HEAD (so this applies to POST/PUT/DELETE) must provide the same token, either in the <code>CSRF-Token</code> request header or in the <code>csrf</code> request parameter:</p> <pre>POST /nuxeo/something CSRF-Token: uNTIwv3oEImb3singqJKSuJDNjM9ldVOjnwtxmFh</pre> <p>or</p> <pre>POST /nuxeo/something?csrf-token=uNTIwv3oEImb3singqJKSuJDNjM9ldVOjnwtxmFh</pre> <p>Missing, expired or invalid token</p> <p>If the token is missing, expired or invalid, the client will get a 403 Forbidden error, an <code>Token: invalid</code> header will be set:</p> <pre>403 Forbidden CSRF-Token: invalid</pre> <p>Skipping certain endpoints</p> <p>Some authentication endpoints need to be available with a POST without CSRF token and can be done using for example:</p> <pre><extension target="org.nuxeo.runtime.ConfigurationService" point="configur <property name="nuxeo.csrf.token.skip" list="true"/>/login</property> </extension></pre> |
| NXP-26060 | Fix merge of DocumentsList descriptor | Configuration Change | <p>Since 10.3, the events in a new contribution overrides the one from the default contribution. Previously they were ignored.</p> |

| Key | Summary | Impact type | Upgrade notes |
|-----------|--|-------------|---|
| NXP-24880 | Make it possible to disable aggregation computations on initial search | | <p>A nuxeo search form result layout can dynamically tell whether aggregations should be instance, on the <code>nuxeo-default-search-form.html</code>, if aggregations should not be compute parameter is not set, you can add the following</p> <pre> skipAggregates: { type: Boolean, notify: true, computed: '_skipAggregates(params.*)' } </pre> <p>where</p> <pre> _skipAggregates : function() { return !this.params !this.params.ecm_fulltext this.params.ecm_fulltext.length === 0; }, </pre> |
| NXP-25204 | Nuxeo instance useless after default-domain is removed | | <p>User's workspaces stored under a trashed domain are no longer used. Trashing the def will kind of trashed the User's workspace. But they can also be restored if the default dc restored.</p> |
| NXP-25464 | Upgrade paper-input to include fixes | | <p>Unlocking paper-input version. At the time ticket is done, we pull `paper-input` `2.3.3` of inner native input does not have the `nativeInput` html tag id anymore. See diff</p> <p>This will impact elements using `paper-input` and manipulating the inner native input wi `#nativeInput` css selector.</p> |
| NXP-25512 | WebUI: Allow to override/extend fetch properties and enrichers used to browse a document | | <p>You can now add new enrichers used when serializing the current document by contrib</p> <pre> <require>org.nuxeo.web.ui.properties.contrib</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configur <property name="org.nuxeo.web.ui.enrichers">myEnricher</property> </extension> </pre> <p>You can now override the fetched properties when loading the current document (defau still resolving all properties):</p> <pre> <require>org.nuxeo.web.ui.properties.contrib</require> <extension target="org.nuxeo.runtime.ConfigurationService" point="configur <property name="org.nuxeo.web.ui.fetchedProperties">dc:creator,dc:nature, </property> </extension> </pre> |