Nuxeo Platform `8.1` Release Notes

Instant Share

A new feature allows to invite users to a given folder of the Nuxeo Platform with either Read or Edit permission, by email. Invited users do not have an account on the Nuxeo Platform instance. The access is temporary and any access remains tracked in the audit log, based on the used email address. Read more in the documentation.

Nuxeo Live Connect for Box

Nuxeo Live Connect has been improved and now integrates with Box.com. Nuxeo Live Connect allows to easily reference files that are stored in the cloud and have them participate to your business processes. Check the Nuxeo Live Connect documentation for more information.
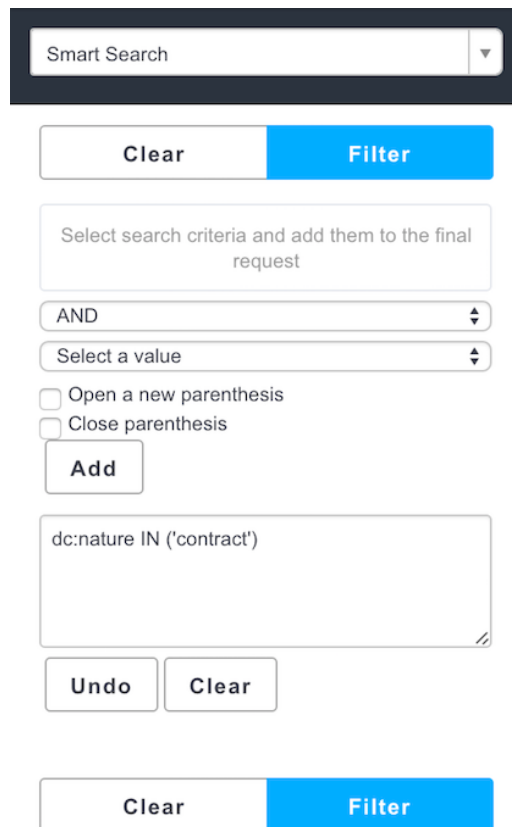

User Feedback on Permission Management Tab

Feedback using the Polymer element paper-toast has been implemented on the Permissions tab. It is used on the create, update and delete permission actions, as well as on the "send notification email" action.

Smart Search Has Been Re-Aligned

Smart Search, that allows users to build their own queries, has been re-integrated to the Search tab, consistently with other search options. Smart Search is delivered as a Nuxeo Package.

Confirmation Popup before Purge

In the Admin Center, when removing the permissions of a given user, you now have to confirm on a popup that you really want to purge all the permissions.
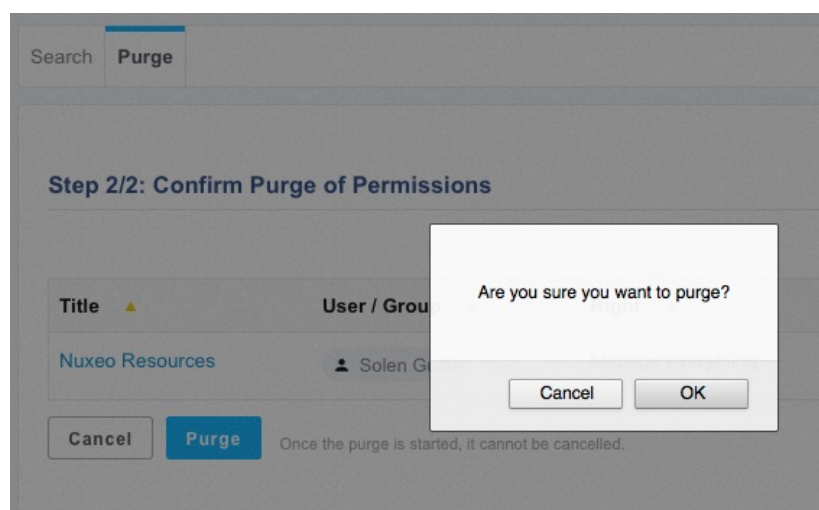
More Friendly "Http Transport returned a 0 status code" Error

When this error happens, the message is now displayed as a warning and translated in a more friendly way.

A C# Nuxeo Client

A C# Nuxeo Client has been added. A sample application is provided so as to understand how to make use of it. Read the C# client documentation for more information.

Configuring TinyMCE Editor via Tag Properties

Plugins and options have been extracted to control them by tag properties in order to display different options depending on the field.

Understanding The Sequence Of Listeners Calls

You can now add this on your log4j.xml:

```
<category name="org.nuxeo.common.logging">
<priority value="DEBUG" />
</category>
```

Then use the sequence.sh to generate a PNG with a sequence diagram of all threads and listeners.

Note that all events in the chart are in chronological order (not proportional) except for the Initiate link, where the origin points to the thread that has generated a work without telling when (earlier for sure).

HTML Content in Tipsy Tooltip

Tooltips generated with Tipsy now render HTML content properly.

S3 Configuration via Extension Point

Customers needing to connect to more than one S3 binary managers with different configurations are now able to do so using the new `configuration` extension point of `org.nuxeo.ecm.core.blob.BlobManager`. Previously only one bucket could be configured in nuxeo.conf.

```
<extension target="org.nuxeo.ecm.core.blob.BlobManager" point="configuration">
   <blobprovider name="default">
      <class>org.nuxeo.ecm.core.storage.sql.S3BinaryManager</class>
      <property name="awsid">your_AWS_ACCESS_KEY_ID</property>
      <property name="awssecret">your_AWS_SECRET_ACCESS_KEY</property>
      <property name="region">us-west-1</property>
      <property name="bucket">your_s3_bucket_name</property>
      <property name="bucket.prefix">myprefix/</property>
      <property name="directdownload">true</property>
      <property name="directdownload.expire">3600</property>
      <property name="cachesize">100MB</property>
      <property name="crypt.keystore.file">/my/keystore.jks</property>
      <property name="crypt.keystore.password">password</property>
      <property name="crypt.key.alias">mykey</property>
      <property name="crypt.key.password">password</property>
      <property name="connection.max">50</property>
      <property name="connection.retry">3</property>
      <property name="connection.timeout">50000</property>
      <property name="socket.timeout">50000</property>
   </blobprovider>
</extension>
```

Server-Side Encryption with S3

Support for Server-Side Encryption (SSE-S3) has been added as an alternative to Client-Side Encryption (CSE) as documented in the S3 Java SDK . This is enabled through the property `nuxeo.s3storage.crypt.serverside=true` .

GridFS as Part of the Default Distribution

GridFS is now part of the base distribution, in nuxeo-core. It will have to be activated explicitly using the gridfsbinaries template. There is no need to install an additional Nuxeo Package anymore. Read our MongoDB admin documentation for more information.

Elasticsearch Indexing Operations

Elasticsearch indexing operations are now available from Nuxeo Automation in addition to the Admin Center. Using a REST call you can now:

- Re-index an entire repository

- Re-index documents that match an NXQL query

- Re-index a document and its children

- Wait for indexing and perform an index refresh

Allow Separate Repository DDL Generation

Nuxeo can now start with a new `nuxeo.vcs.ddlmode` parameter that specifies how the DDL for repository initialization should be executed at startup. Depending on the chosen mode, the DDL may not be executed at all and instead dumped to the `log/ddl-vcs-default.sql` log file. Possible values for `nuxeo.vcs.ddlmode` are:

- ignore : no DDL detected or executed. Replaces `nuxeo.vcs.noddl=true`.

- compat : compatibility mode with previous version. Always executes DDL that recreates stored procedures (and does not attempt to detect existing ones). Replaces `nuxeo.vcs.noddl=false`.

- execute : executes the DDL and starts Nuxeo normally.

- dump : dumps the DDL (if any) to the file but still executes it and starts Nuxeo normally.

- dump,ignore : dumps the DDL (if any) to the file but does not execute it and starts Nuxeo normally (will likely result in errors).

- dump,abort : dumps the DDL (if any) to the file, and if not empty aborts startup.

For Nuxeo Platform 6.0 and Nuxeo Platform 7.10 the default was `nuxeo.vcs.ddlmode=compat`. For Nuxeo 8.1 the default is `nuxeo.vcs.ddlmode=execute`.

Note that all this applies to the DDL for the document repository, but not DDL used by other users of SQL in the platform (directories, quartz, JPA persistence).

Cluster Invalidation with Redis in nuxeo.conf

Using Redis to process cluster invalidation instead of database can be configured in nuxeo.conf using `repository.clustering.invalidation=redis` (it used to be in the repository contribution).