



Nuxeo

WebUI

Bulk Download Specifications

Last update: 27/06/2017

Table of Content

Feature Overview	3
Goals	3
Audience	3
Scope	3
Reference Documents	3
Glossary	3
Functional Requirements	4
User Stories	4
Acceptance Criteria	4
Tests Cases	4
Non Functional Requirements/ Software Attributes	5
Scalability	5
Privacy	5
Performance	5
Features to Implement	6
Requirements	6
Feature(s) description	6
Mockups	6
Limitations	7

Feature Overview

Goals

Be able to download **one** or a **set of assets** in a relevant and meaningful way.
Download folderish content, collections and selections aggregated in a zip file.

Problems to solve

Currently, there is no functionality to download a set of assets. Folders have a rendition that exports all their content, but in a non-usable format outside of Nuxeo platform.

Even a very basic scenario, if a user needs to download a set of photos to send via email, to use on another software tool, there aren't any option currently.

In a bulk download, understanding which is the correct content and format for each asset type is also an issue. It can also depend on user profile or other variables.

Some specific content types are only usable if accompanied by related files. Some examples are 3D files that can have several attachments or, as an example, inDesign file types that requires other related assets. How these assets and relations gain meaning is also an identified problem.

Audience

Every user.

Scope

Downloading set of assets in a straightforward fashion.

Reference Documents

Glossary

No remarks.

Functional Requirements

<https://app.prodpad.com/ideas/6/userstories>

Non Functional Requirements/ Software Attributes

Scalability

No remarks.

Privacy

No remarks.

Performance

Not being asynchronous, the response time can be longer than timeout, that will result in error, but server will still process all zipping with unused and unnecessary effort and cost.

Features to Implement

Requirements

Must work with current rendition system.

Feature(s) description

When doing bulk download, the platform has to understand what to download for each asset, concatenate and export. "What to download" can be different for different users and contexts, so a system is needed to process it all in bulk automatically with proper flexibility.

Main rendition contribution

Renditions are a CMIS standard. They serve as an alternative representation of an asset and can provide a preview, a different format or a view with a specific goal.

As so, they provide a good solution for asset download as different renditions can serve for different use cases and different contexts.

One of the identified problems is how to download an asset that has all significant data and context: renditions as flexible and can resolve this issue.

For this topic, a new concept: main renditions should be produced. Main rendition is the identification of the canonical rendition that is used for export / download.

For now, the platform allows definition of renditions and renditions providers, that can define renditions dynamically. But there is no concept of main rendition of an asset according to a context.

A new contribution type should allow a dynamic definition of main rendition for different assets. It should provide the identification of one rendition per asset (in case the system isn't able to provide only one main rendition name - because of overlapping filters -, they should be ordered so there is always one main rendition identified).

The main rendition contribution should be dynamic and allow something like an EL expression that uses the current user and context to decide if it should be applied.

It is important that this filter mechanism has access to both user information, document - including its current lifecycle state - and generic platform variables already in Filters.

The user can then provide one or several contributions to setup different main renditions to different user profiles and different doc types or facets.

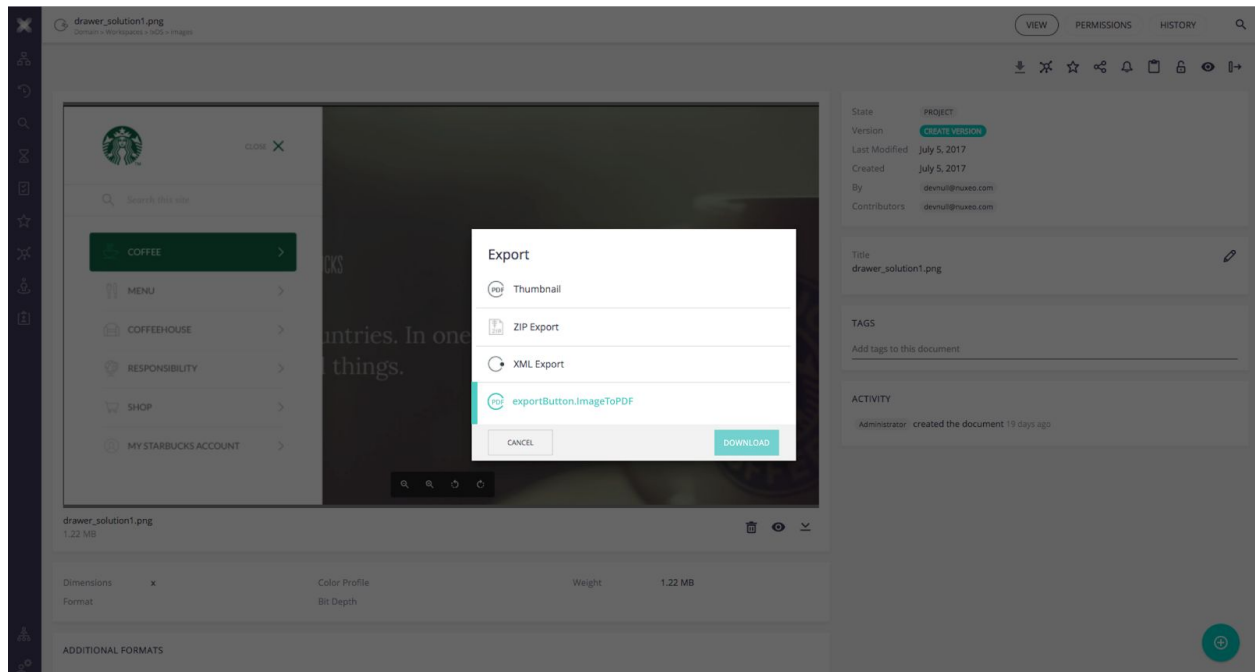
The contribution might be as simple as a id parameter - that indicates rendition name -, an order/priority number and a Filter.

Export improvements

In order to take advantage of this new concept export popup will be improved.

Instead of links, the export list should be selectable with a download button.

When there is a main rendition, this rendition should always be present in this list and selected by default.



Actions and Feedback

To initiate the download process, the user will have a bulk download action button. These should be displayed on assets, folderish containers and collections.

Also, in any selection, a similar action is presented to download only the select items.

The screenshot shows a web application interface with a dark theme. At the top, it indicates "2 item(s) selected" and "Display selection". The main content area features a table with columns for Title, Modified, Last Contributor, and State. Two items are selected, highlighted in green:

Title	Modified	Last Contributor	State
Nuxeo Technical Brief Architecture for ...	June 7, 2016	devnull@nuxeo.com	Project
Nuxeo Security Technical Brief.pdf	June 7, 2016	devnull@nuxeo.com	Project
Nuxeo DAM Digital Workplace.pdf	June 7, 2016	devnull@nuxeo.com	Project
Nuxeo API.pdf	June 7, 2016	devnull@nuxeo.com	Project

On the right side, there is a sidebar with metadata for the selected items, including State (PROJECT), Last Modified (June 7, 2016), Created (June 7, 2016), By (devnull@nuxeo.com), and Contributors (devnull@nuxeo.com). Below this, there are sections for Title (Reports), TAGS (Add tags to this document), and ACTIVITY.

At the bottom left, a notification tray shows two items: "Case studies (4 assets) preparing download" and "Reports (2 assets) preparing download", each with a close button (x). A green circular button with a plus sign is located at the bottom right.

This screenshot is identical to the one above, showing the same document list and sidebar. However, the notification tray at the bottom left now shows two items: "Case studies (4 assets) preparing download" and "Reports (2 assets) preparing download", each with a close button (x). A green circular button with a plus sign is located at the bottom right.

Download is synchronous, but its request should be in background and allow the user to use Web UI while it is processing. This will allow cases of multiple downloads. In this case, data about each pending download should be kept to allow proper feedback.

This feature will require a multi-toast notifying system.

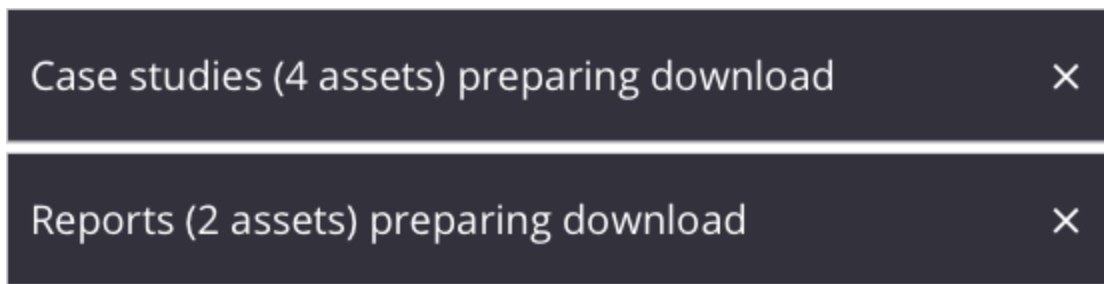
Notification improvement

The new system should allow for multiple toast, that should appear vertically, ordered by event date. It should be possible to have permanent toasts that only close on user action, or by other events.

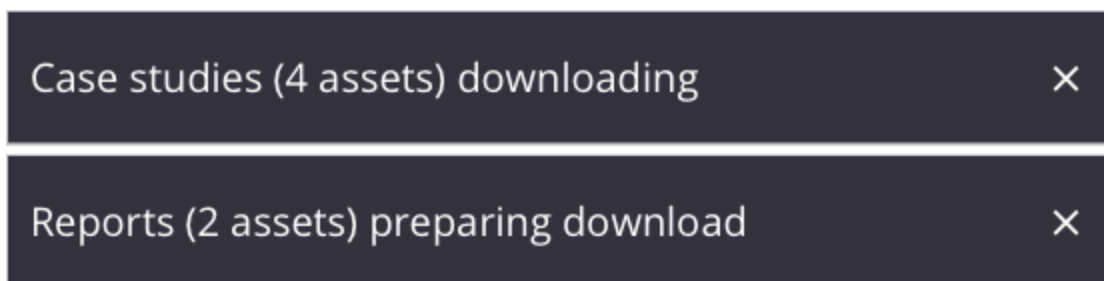
Toast feedback

When the user start a download the present toast system should present:

- “Preparing download” : A toast should appear with “Preparing download” and a new line for each bulk download with the name and the number of assets. Eg. “Test Folder (24 assets) preparing download”



- “Downloading” : When a bulk download finish processing and starts downloading, its corresponding line should change to “downloading” and disappear after some brieve seconds.



- “Error” : Whenever an error happens on a bulk download preparation, on its line, it should appear “An error occurred” or the error info, if we have a summary. It should not disappear and have a closing action to be sure user is aware of this exception.

Reports (2 assets) an error occurred



User flow

Next is presented an user flow while downloading. These illustrate download icons and feedback.

User navigates to a folder and taps on bulk download. It is presented a toast with Preparing Download with only one download line. This toast only disappears on user action.

If user navigates to a collection and bulk downloads, another line will appear.

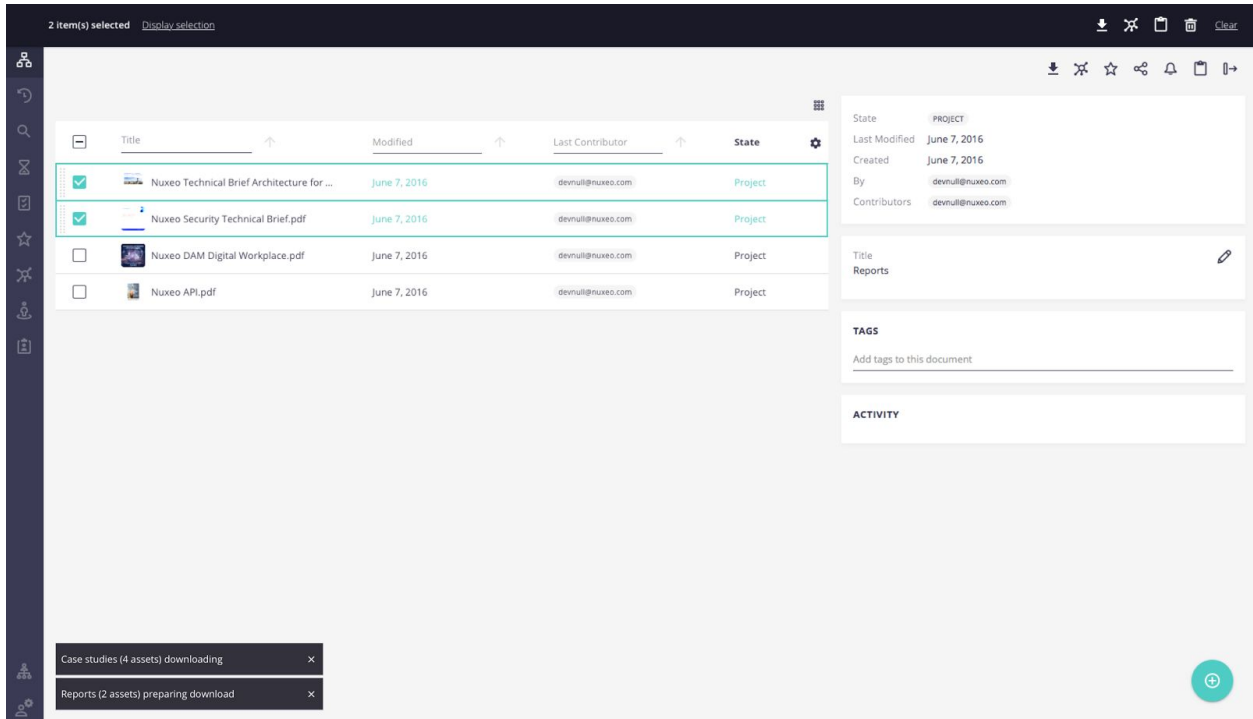
The screenshot displays a document management interface. At the top, a dark toast notification reads "Reports (2 assets) an error occurred" with a close button. Below, a table lists four assets:

	Title	Modified	Last Contributor	State
<input checked="" type="checkbox"/>	Nuxeo Technical Brief Architecture for ...	June 7, 2016	devnull@nuxeo.com	Project
<input checked="" type="checkbox"/>	Nuxeo Security Technical Brief.pdf	June 7, 2016	devnull@nuxeo.com	Project
<input type="checkbox"/>	Nuxeo DAM Digital Workplace.pdf	June 7, 2016	devnull@nuxeo.com	Project
<input type="checkbox"/>	Nuxeo API.pdf	June 7, 2016	devnull@nuxeo.com	Project

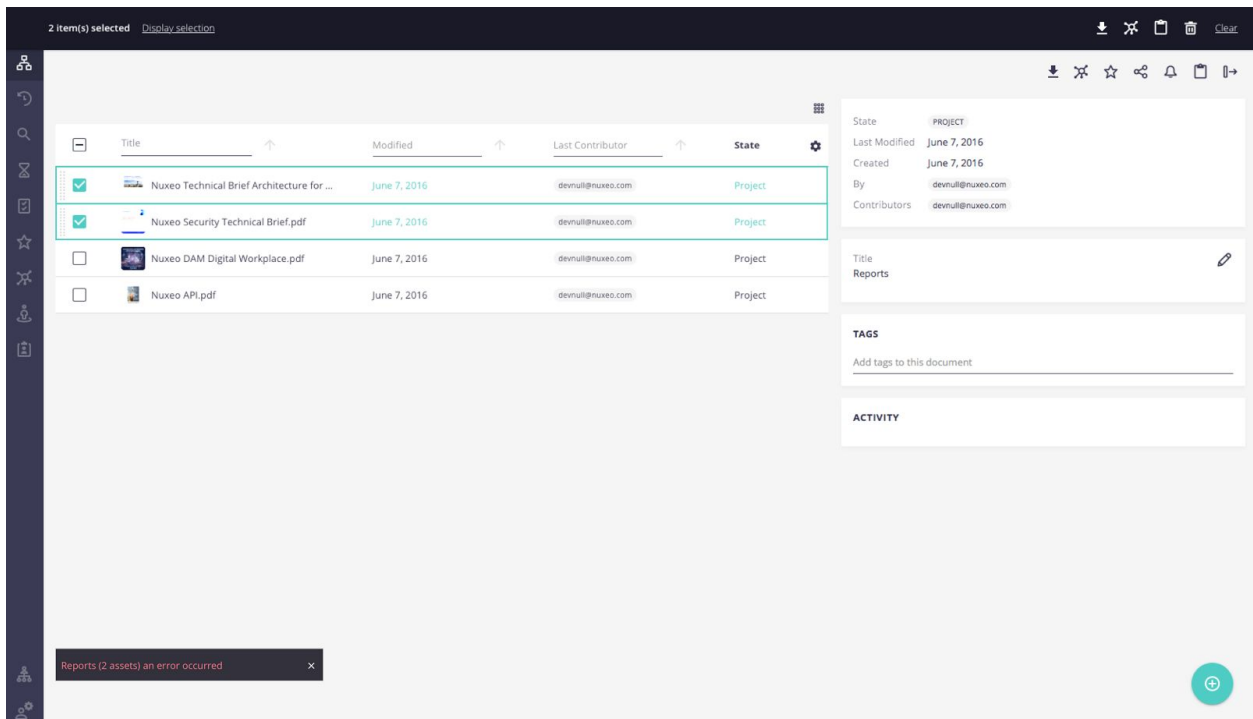
On the right, a metadata panel shows details for the selected items, including State (PROJECT), Last Modified (June 7, 2016), Created (June 7, 2016), By (devnull@nuxeo.com), and Contributors (devnull@nuxeo.com). Below this are sections for Title (Reports), TAGS (Add tags to this document), and ACTIVITY.

At the bottom, a toast notification shows two lines: "Case studies (4 assets) preparing download" and "Reports (2 assets) preparing download", each with a close button. A green circular button with a plus sign is visible in the bottom right corner.

When one of the bulk downloads preparation is finished and starts downloading, the corresponding toast will disappear and a new one will appear temporarily with updated info "downloading".



Finally, if one of the preparations results in error, corresponding toast should disappear and a new toast should appear with error message. This toast should be permanent until user closes it.



Download Naming

On **folderish** documents, the download naming should be:

<document-name>-<timestamp>.zip

On **collections**, the download naming should be:

<collection-name>-<timestamp>.zip

For **selections** the download naming should be:

<folderish|collection|Search-name>-selection-<timestamp>.zip

Limitations

Download should not be recurring for this first effort. When downloading a folder with collections/folderish documents, download should not act on these document children.

Also, the download process should be synchronous, server side.

Gang team feedback

- Download adapter in TL
- Not clear folderish and collections have bulk download action (at least from presentation)
- Export and bulk download action may be redundant