Content Services Toolkit

Updated: 9/26/2019

This document contains the steps to import the Content Services Toolkit, create a simple BAStudio application using the components from the toolkit and create an IBM Content Navigator desktop that displays your application.

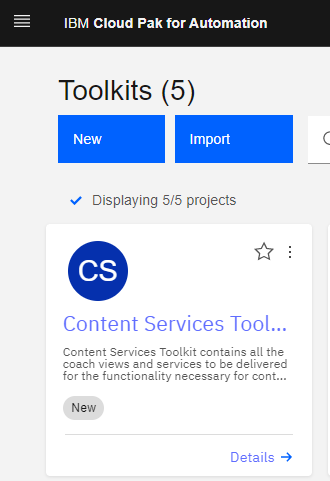
# Import the Content Services Toolkit

Retrieve the latest Content Services Toolkit from GitHub.

Location of Content\_Services\_Toolkit – 0.0.78.twx

[**https://github.com/icp4a/content-toolkit**](https://github.com/icp4a/content-toolkit)

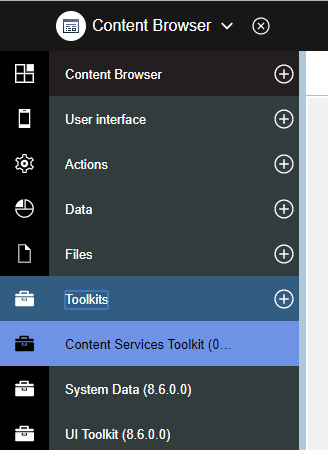
Import the toolkit into BAStudio



# Create BAStudio Application

Create your application and reference the Content Services Toolkit.

Add the “Content Services Toolkit” to your application. Find “Toolkits” in left panel and click “+” icon to add the toolkit.



Open your application’s “App Project Settings” view where you’ll define the location of the graphql application resource (server).

Select the “App Resource” tab and enter the following

Name: “GRAPHQL\_APP\_RESOURCE”

Type: “Defined Endpoint”

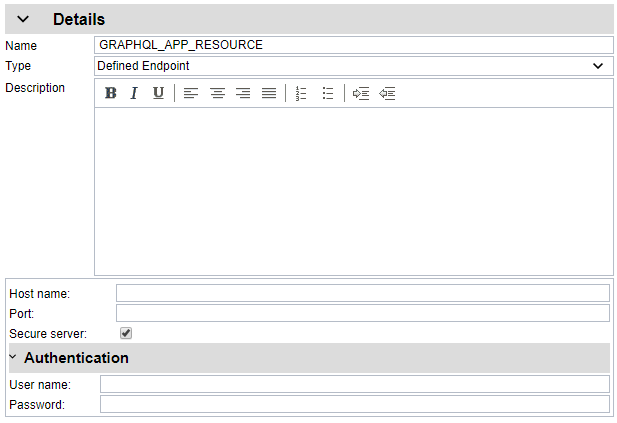
Host name: <URL to your graphql server>

Port: <port number>

Secure Server: checked

User name: <authenticating user for graphql>

Password: <password>

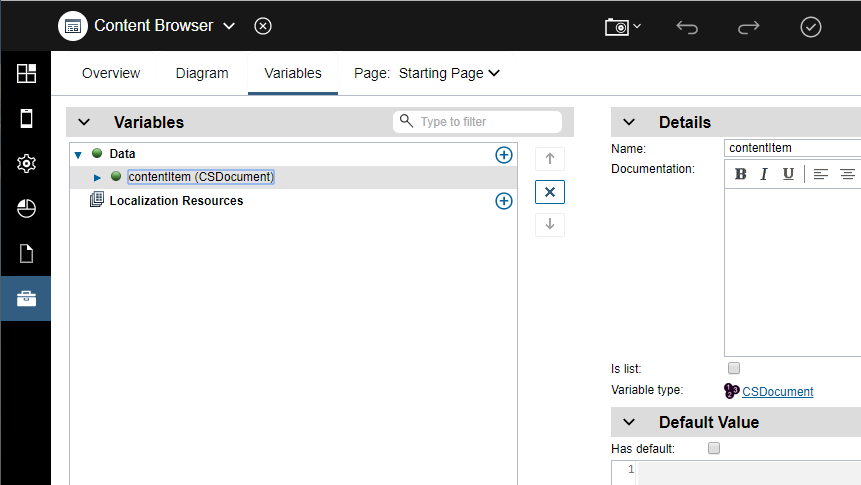


Re-select your application view and open the “Variables” tab

Create the “contentItem” variable with the following settings:

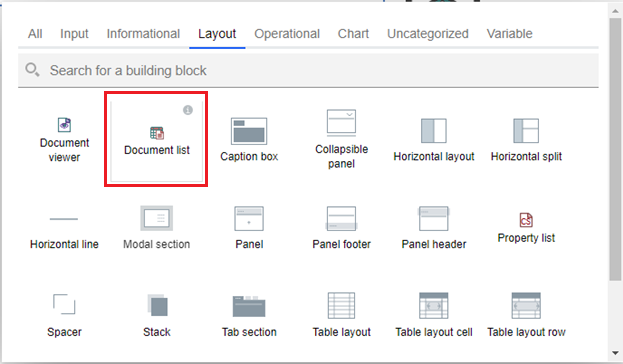
Is List = false

Variable type = CSDocument

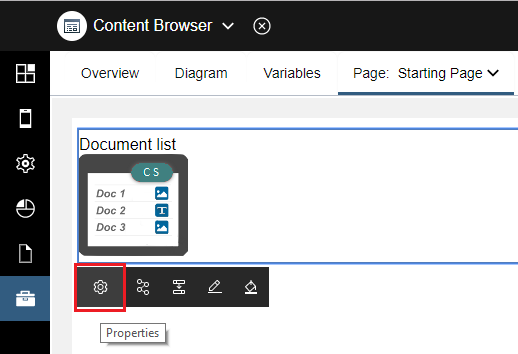


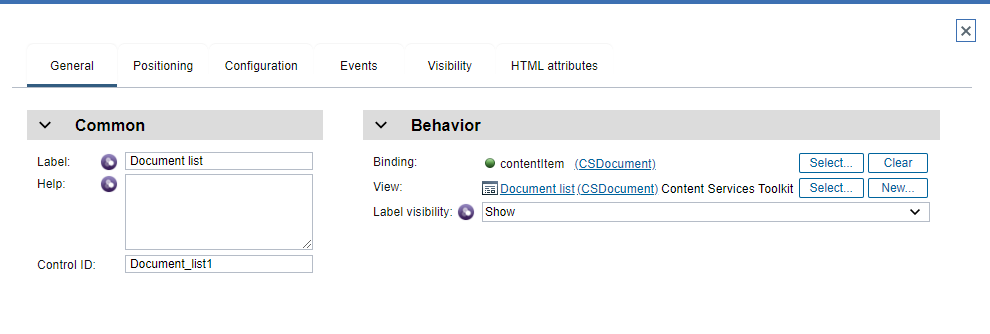
# Document list - Configuration

Open the component selection dialog and select the “Document list” component, located in the “Layout” category.



Open the components properties dialog (left click on component):





Select the “Configuration” tab

In the Server Configuration Name field, enter the graphql app resource name you created above.

e.g. “GRAPHQL\_APP\_RESOURCE”

In the Object store name field, enter the name of your content engine object store.

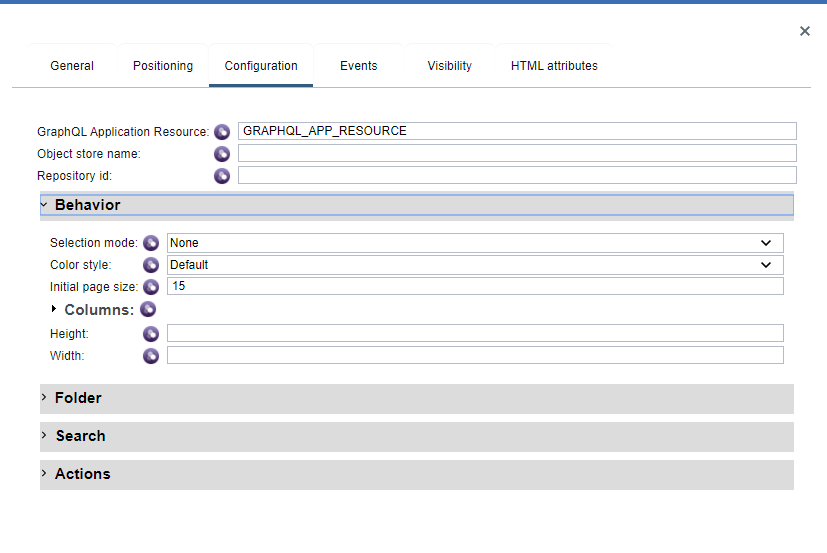
Note: The object store name can be retrieved from the Acce admin tool.

In the Repository Id field, enter the name of your desktop repository.

Note: The repository Id can be retrieved from the ICN admin desktop.

Open the Behavior section.

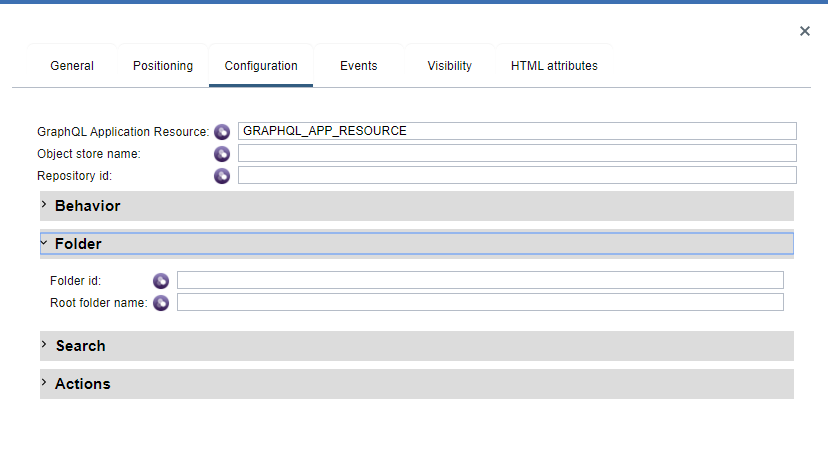
You can optionally enter a new paging size for the list view. The default is 20 items (if not specified).



If you want the list to display folder contents, open the Folder section and enter the ID (guid) of the folder you would like to display. Specifying a folder path instead of ID is also supported.

Note: The folder ID (guid) can be retrieved from Acce admin tool.

You can optionally enter the root folder name to display in the breadcrumb (in case your folder does not have a user-friendly name).



If you want to display search results, open the Search section and enter the property ID (symbolic name) you would like to search on, the operator, property type and optionally a default value. If no default value is entered, the list is initially displayed empty and the user must type in the search field to initiate a query. The hide search field flag can be used to lock down the query parameters if you don’t want the user to change them at runtime. Finally, you can specify a text search option (CBR), the default is not enabled.

Note: If both a Folder ID and Search criteria are specified, the Folder ID will have precedence.

We are still working through some “issues” with properties that have choice lists and multi-value properties.

If the property has a choice list and the display values are different than the actual values, you must type in the actual value. There currently isn’t any mechanism to resolve the display name to the actual value.

Multi-value property support is still being worked on and not fully functional at this point.

There are several special operators that are currently supported:

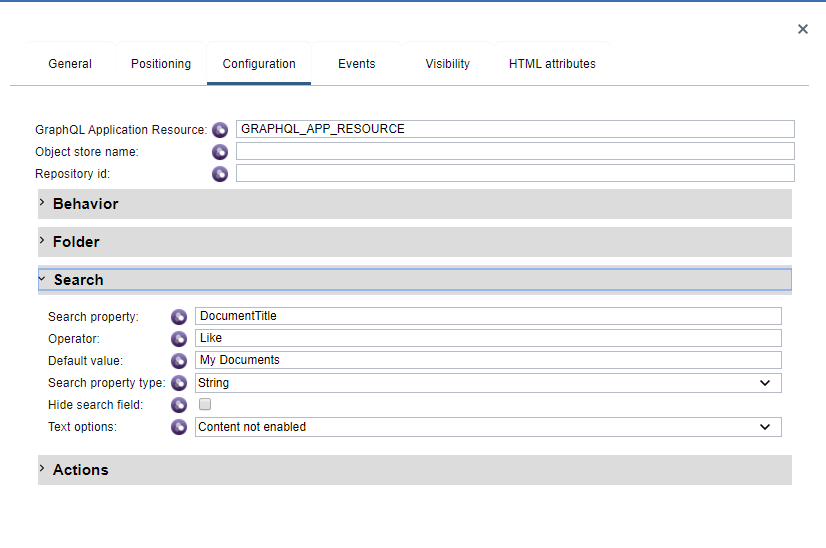
“Include All” – Multi value properties only

“In” – Single value properties with a choice list, this is the “Include Any” operator

“Intersects” – Multi value properties only, this is the “Include Any” operator

“Is Null” – Multi value properties only

“Is Not Null” – multi value properties only

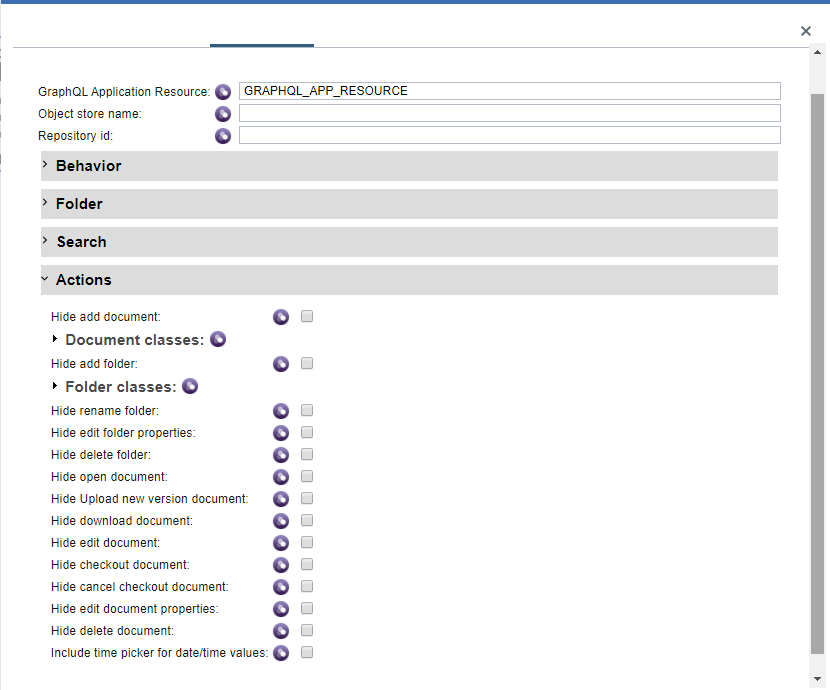


Open the Actions section.

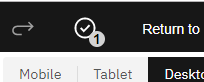
These settings allow the user to hide any of the listed actions from the context menu.

Users can also specify a list of classes that will be exposed from the Add Document and Add Folder modal dialogs (instead of retrieving the list from the repository).

Nothing needs to be set here.



At this point, you can save your settings (button on application banner bar):

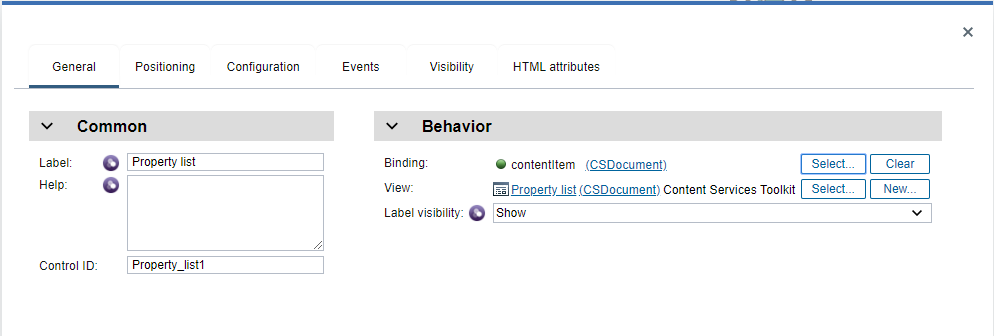


You can then run your application in test mode (carrot button on far right of banner). You should see the list view rendered with the results from the folder or search you specified.

# Property list – Configuration

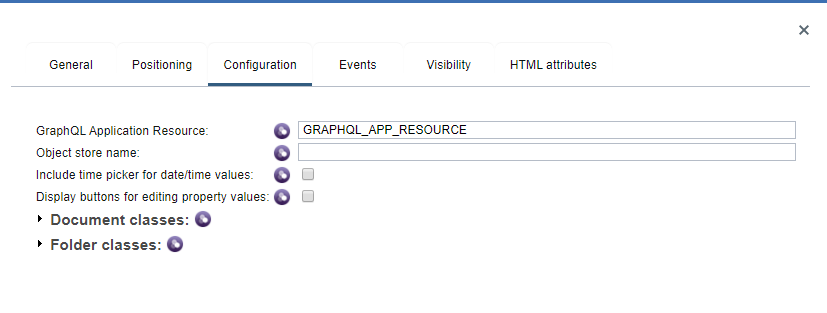
Select the Property list component from the palette, place below the list view or to the right (or where ever you’d like it) and open the properties dialog.

On the “General” tab, select the “Binding” property to be the variable called “contentItem”. This variable you created above as part of the Content list configuration.



Next, switch to the “Configuration” tab and enter the name of the graphql application resource you created above in the Server Application Resource field.

In the Object store name field, enter the name of your object store from the content engine (Acce).

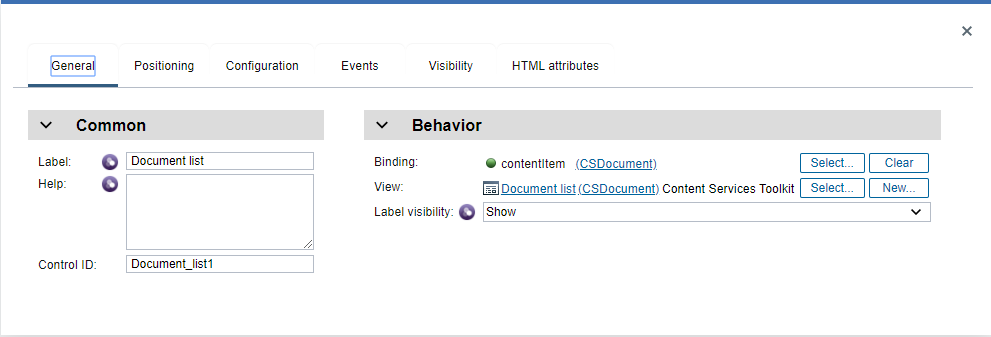


Close the Properties dialog.

Select the Content list and open its properties dialog.

On the “General” tab, select the “Binding” property to be the variable called “contentItem”. This variable you created above as part of the Content list configuration.

Both the Content list and the Property list must be bound to the same application-based variable to enable the event mechanism to work correctly between the two.



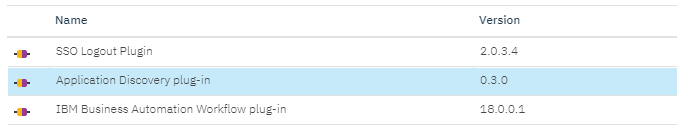
Close the Properties dialog and click the save button on the banner bar.

Run your application in “Test” mode (carrot button on far right). You should see the list view rendered and when you select a document from the list view, the property list component will render the document class and class properties.

# IBM Content Navigator (ICN) – Configuration

IBM Content Navigator should be running in “Platform” mode and logged into the admin desktop (Administration).

Make sure the application discovery plugin jar is installed in your IBM Content Navigator environment (it should already be installed). The plugin name is “Application Discovery plug-in”. Under “Plug-ins” check:



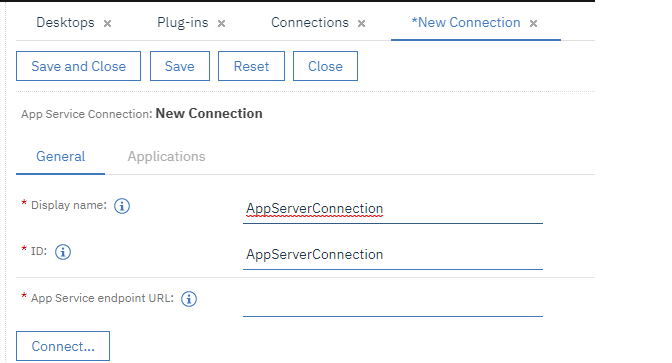
Open the “Connections” feature, click the New Connection button, give it a name:

“AppServerConnection”

Type in the URL to application server:

App Service endpoint URL: <URL to BAStudio application server>

Click Connect and enter a user name and password.



Click “Save” and then “Close”.

Open the “Repositories” tab (still in Connections admin feature area).

Note: if you already have your FileNet repository configured, you can skip this step.

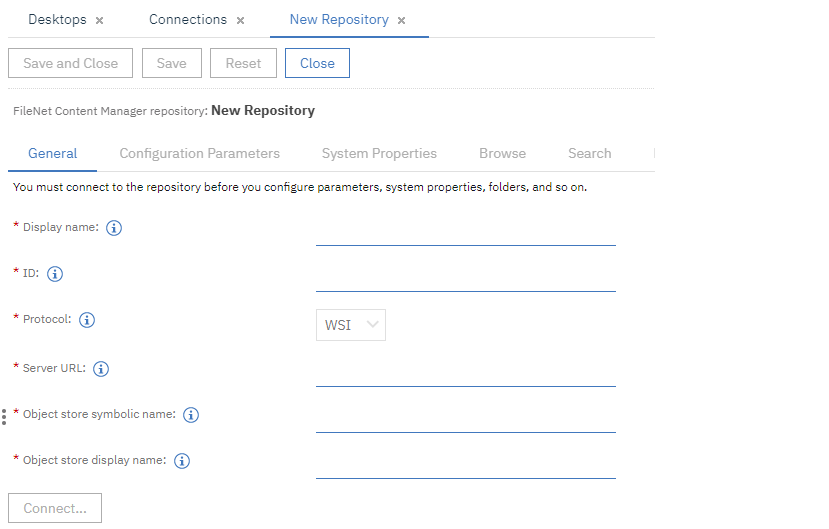
Click “New Repository”, create FileNet Content Manager repository, and give it a name:

Display name: “<your repo name>”

Type in the URL to the application server:

Server URL: <URL to CE content repository>

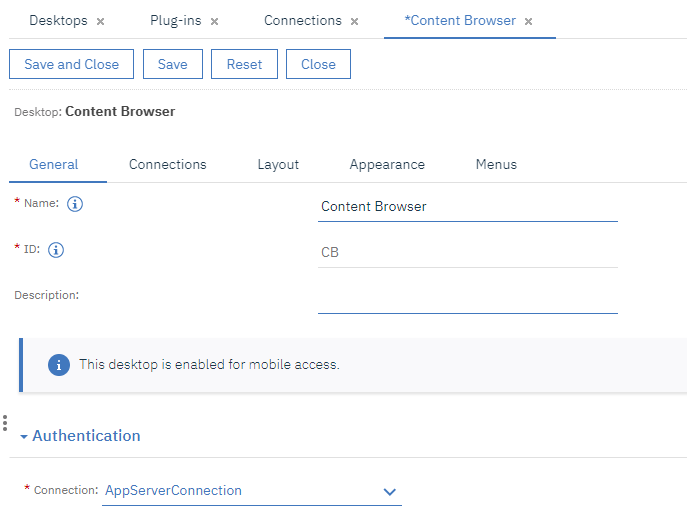
Click Connect and enter the user name and password.



Next, create a new desktop, select the “Desktops” admin feature.

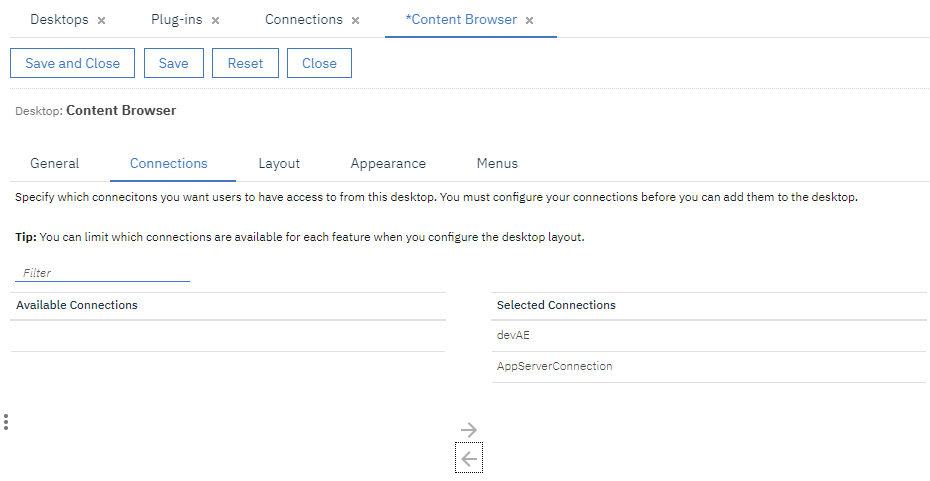
Click the “New Desktop” button, give it a name and ID.

Point the “Connection” of your desktop to the connection you just created.



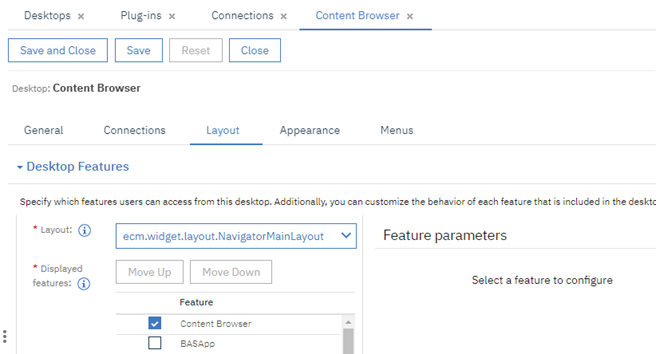
Next, click on the “Connections” tab

Select your Repository where your content will be located, you should have both a connection and repository:



Finally, select the “Layout” tab

In the Desktop Features selection area, select your BAStudio application as the feature. You can also have other IBM Content Navigator features enabled if you are running in “Platform and Content” mode.



Save and login to IBM Content Navigator using your newly created desktop… Enjoy 😊