

Fresh upgrade guide

Upgrade to v4.2

Updated 28 April 2022



FRESH

Content and Code Ltd. | 130 Fenchurch Street, London EC3M 5DJ
contentandcode.com | freshintranet.com

1 Prerequisites

1.1 Version of Fresh

Make sure the previous version of Fresh (**Fresh 4.1**) is installed in your tenant.

You can check which version of Fresh you are on in the App catalog following the steps described in [9.1 Check which version of Fresh you are on](#)

If you do not have the latest version of Fresh deployed in your tenant, ask the Service Owner for the previous upgrade packages.

1.2 Install PnP.PowerShell module

The PnP.PowerShell module provide commands which are required by the upgrade script.

If not already installed, navigate to the following URL and install the latest PnP.PowerShell version.

<https://pnp.github.io/powershell/>

1.3 Azure CLI

Only if the provisioning module is deployed in your tenant, you will need to install Azure CLI to deploy the resources to Azure:

<https://docs.microsoft.com/en-us/cli/azure/install-azure-cli-windows?view=azure-cli-latest>

1.4 Set PowerShell execution policy

An execution policy is part of the PowerShell security strategy. Execution policies determine whether you can load configuration files, such as your PowerShell profile, or run scripts. In order to run scripts, you will need to set this policy and then, after the deployment is completed, set it back to the original value.

Open a new instance of the SharePoint Online Management Shell as an administrator (right-click, 'Run as Administrator'). Enter the following command:

```
Get-ExecutionPolicy
```

Take note of the current execution policy as it should be returned to this value after the deployment is complete.

Enter the following command:

```
Set-ExecutionPolicy "Unrestricted"
```

When prompted to change the execution policy type "Yes"

This will allow PowerShell to execute scripts including those which have not been signed by a trusted publisher

2 Deployment account

The deployment steps in this document must be actioned using an account that has the required attributes. The deployment account must:

- The account should have Cloud authentication (i.e. not Federated authentication).
 - The deployment scripts will run faster if MFA is disabled for this user.
- Be a SharePoint administrator. This is provided by either of the following administrator roles:
 - 'Global administrator'
 - 'Customized administrator' with 'SharePoint administrator' rights
['Office 365 admin center' > USERS > Active Users > **select a user** > EDIT USER ROLES]
- Be a SharePoint term store administrator
['Office 365 admin center' > ADMIN CENTERS > SharePoint > term store > Term Store Administrators]
- Have English as the primary language
['About me' > 'Update profile' > 'Additional information' > 'Edit' > '...' > 'Language and Region']
- Have permissions to add solutions in the App Catalog.
['App catalog' > 'Site settings'].
- If the Provisioning module is deployed, the deployment account must have contributed permissions in the Resource group in Azure

Recommendation: Create a dedicated deployment account.

- Create a cloud-only account on the *onmicrosoft* domain associated with the tenant
- Do not assign a license
- Configure the account as a SharePoint administrator, term store administrator, and App catalog administrator.
- Once the deployment is complete disable sign-on for the account

Note that an account with higher privileges will be needed to approve SharePoint API permission requests and to grant admin consent for the permission that app registrations need.

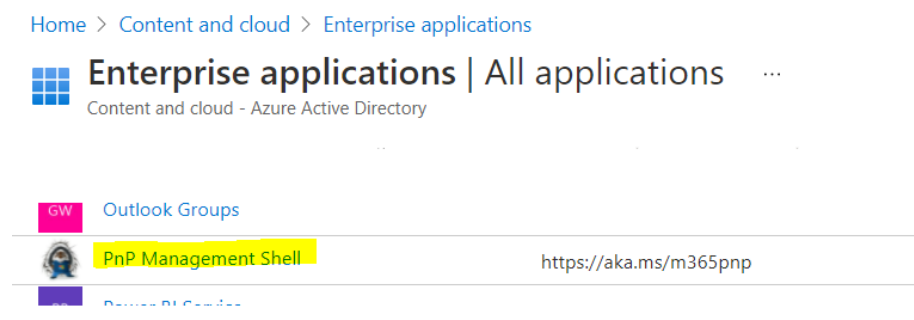
3 PnP.PowerShell authentication

PnP.PowerShell allows you to authenticate with credentials to your tenant. However, due to changes in the underlying SDKs this PowerShell commands require to register an Azure AD Application which will allow to authenticate.

This action will create an Enterprise Application in the Azure Active Directory that will allow the PnP PowerShell script run the commands in the tenant.

*If this has been already done in the environment, it doesn't need to be done again.
The account that is used in the deployment needs to be added in the Enterprise Application.*

This is the Enterprise Application that will be created in the Azure Active Directory



You can read more about the PnP.PowerShell authentication in the following page:
<https://pnp.github.io/powershell/articles/authentication>

3.1.1 Run the script

1. Open a new Windows PowerShell console as an administrator (right-click, 'Run as Administrator').
2. Enter the following command: **Register-PnPManagementShellAccess**

The Enterprise Application can be removed after the deployment if there are any concerns about security.

4 Configure deployment environment script

NOTE: Do not edit any of the deployment scripts beyond what is specifically mentioned here

Copy the provided deployment scripts folder locally.

At the root of the provided Scripts folder, find the folder called Config.

You should have a file for every environment. Use **dev.ps1**, for the development environment or duplicate it and rename it with the corresponding environment name (like uat.ps1 or prod.ps1).

If you don't have a config file from a previous deployment, open the provided file in a text editor such as VS Code or Notepad and update the following variables:

SharePoint settings:

- **\$global:tenantUrl** The value assigned should be the URL of the root site collection and should end with ".sharepoint.com". The root site collection will not be a deployment target – this value is only used to build other URLs. **Make sure the URL has no trailing slash.**
- **\$global:tenantAdminUrl** The value assigned should be the URL of the SharePoint Admin Centre. This value should end with "-admin.sharepoint.com"
- **\$global:defaultSiteOwner** The value assigned should be the username of the account under which the deployment script will run.
- **\$global:termStoreLanguages** Array with the codes of the languages.

Authentication settings:

- **\$global:useWebLogin** Indicate if web login should be used. This should be true for Multi-factor authentication. If it is set to false, your credentials should be stored in a file.
- **\$global:storedCredentialsPath** Only if **\$useWebLogin** is set to false. Indicate the path of the credentials file. After editing and saving this configuration file, execute the script **Tools/Save-Credentials.ps1** to generate the credentials file.

Azure settings (not needed for a deployment without Azure components):

- **\$global:subscriptionId** You can get the subscription id from <https://portal.azure.com> > Cost Management + Billing > Subscriptions.
- **\$global:tenantDomain** Domain of the tenant. <https://portal.azure.com> > Azure Active Directory. (not needed for a deployment without Azure components)
- **\$global:azureAssetPrefix** Azure Assets Prefix that was used in the Fresh deployment
- **\$global:resourceGroupName** Resource group where the Fresh artefacts are located excluding the azureAssetPrefix
- **\$global:resourceGroupLocation** Location of the resource group, like "uksouth".

Provisioning settings (Not needed for a deployment without Provisioning module):

- **`$global:provisioningStorageAccountName`** the name of the storage account used for provisioning
 - Go to portal.azure.com
 - In the top search bar, write "storage" and then click "Storage account".
 - Get the name of the storage account used in the provisioning process
- **`$global:provisioningFunctionAppName`** the name of the function used in the provisioning.
 - Go to portal.azure.com
 - In the top search bar, write "function app" and then click "Function app".
 - Get the name of the function for provisioning.

5 Configure instances script

If you have the file **Files\Instances.psm1** from a previous deployment, you can overwrite the provided file with it in the Files folder.

If you don't have the file **Files\Instances.psm1** from a previous deployment, edit the provided one and make sure all the Fresh instances are listed in the `$instances` array. For every instance, at least the properties `InstanceStamp` and `InstancePrefix` should be populated. For environments with only one instance the provided file should be good to use.

6 Microsoft 365 CDN issue

This only applies if the Microsoft 365 CDN is activated in your tenant.

To get general information about the Microsoft 365 CDN, [read this article: Use the Office 365 Content Delivery Network \(CDN\) with SharePoint Online.](#)

We have found an issue with the Microsoft CDN that only affects some users directly after deployment is done, where the error message "Something went wrong" is displayed to the user. This is related to the synchronization between the updated SPFx packages in the App Catalog and the CDN. The error message is removed if the user does a hard refresh in the browser (Control + F5).

This seems to be caused by the Microsoft 365 CDN taking extra time to refresh the reference to the new files that has been deployed and this cause an error in the pages.

To avoid this, we are deactivating the CDN in the tenant as the first step in the deployment. We recommend this action to avoid the errors. We are actively researching about this problem and when we have more information, and the issue is solved we will contact you to activate the CDN again. The script to enable the CDN in the tenant is included in the Tools folder: **Enable-Office365Cdn.ps1**

If you feel comfortable deploying the upgrade with the CDN activated, this step can be skipped

To disable the CDN (public and private types):

6.1.1 Execute script

3. Update the first line of the **Environment.ps1** file to point to the corresponding environment file.
4. Open a new instance of the SharePoint Online Management Shell as an administrator (right-click, 'Run as Administrator').
5. Navigate to the local location of the Upgrade deployment scripts folder. E.g., cd C:\Deployment\Scripts\Tools
6. Enter the following command: **.\Disable-Office365Cdn.ps1**
7. If you are using WebLogin, when prompted, enter the username and password.

7 Deployment

7.1 Mandatory updates

The upgrade script will perform the following steps:

- a. Update the following packages in the app catalog
 - candc-fresh-globalstructure.sppkg
 - candc-fresh-contentwebparts.sppkg
 - candc-fresh-intranetwebparts.sppkg
 - candc-fresh-coredwp.sppkg
 - candc-fresh-governanceadmin.sppkg
 - candc-fresh-usercentricwebparts.sppkg
 - candc-fresh-toolsadmin.sppkg
 - candc-fresh-corelibrary.sppkg
 - candc-fresh-dashboard.sppkg
- b. Install the new package in the app catalog
 - candc-fresh-coreweb.sppkg
 - candc-fresh-media.sppkg
 - candc-fresh-governance.sppkg
- c. Depending on the modules that are deployed, these packages will be updated as well
 - candc-fresh-provisioningwebparts.sppkg
 - candc-fresh-searchwebparts.sppkg
 - candc-fresh-appchecklist.sppkg
 - candc-fresh-webanalytics.sppkg
- d. Create new managed properties
 - FreshContentOwner
 - FreshReviewDate
- e. Re-map FreshSortableSiteTitle managed property to SiteName instead of SiteTitle

7.1.1 Prepare the scripts

1. Update the first line of the `Environment.ps1` file to point to the corresponding environment file.
2. Open the `Upgrade\Upgrade-Fresh4.2-deployapps.ps1` file
3. Edit the line 25 to indicate the optional modules deployed in your environment:
 - Add `-Provisioning` parameter if the Provisioning is deployed in the environment
 - Add `-WebAnalytics` parameter if the Analytics solution is deployed in the environment
 - Add `-CheckList` parameter if the Checklist solution is deployed in the environment
 - Add `-Search` parameter if the Search module solution is deployed in the environment

The line 25 should be like this (in case the 4 packages are deployed in your tenant):

`Update-Solutions -Provisioning -WebAnalytics -CheckList -Search`

If none of these modules are installed in the environment the line 25 should be: `Update-Solutions`

7.1.2 Execute script

1. Open a new Windows PowerShell console as an administrator (right-click, 'Run as Administrator').
2. Navigate to the local location of the `Upgrade` deployment scripts folder. E.g. `cd C:\Deployment\Scripts\Upgrade`
3. Enter the following command: `.\Upgrade-Fresh4.2-deployapps.ps1`
4. If you are using WebLogin, when prompted, enter the username and password.

Note that after the script is complete it is expected to see the version **4.2.0.0** in the Fresh packages in the app catalog.

7.2 Update Fresh site collections

For this step the correct Instance file is needed in the Files folder. See the section 5 of this document.

The upgrade script will perform the following steps in every site:

- a. Update existing apps.
- b. Install the new "Fresh - Core website features" app.
- c. Install the new "Fresh - Media features" app.
- d. Install the new "Fresh - Governance features" app.
- e. Create the new "Review date" site column and add it to the "Fresh document" and "Site page" content types.
- f. Create the new "Content owner" site column and add it to the "Fresh document" and "Site page" content types
- g. Trigger a reindex of the site so that FreshSortableSiteTitle will be refreshed.

7.2.1 Execute script

Note that if the WebLogin option is being used, the password would need to be introduced for each site collection. In order to avoid this, the MFA for the user that is executing the script could be temporarily deactivated and use the deployment with WebLogin to false (use the credentials file)

This script takes around 40 seconds per site.

1. Open a new Windows PowerShell console as an administrator (right-click, 'Run as Administrator').
2. Navigate to the local location of the **Upgrade** deployment scripts folder. E.g. `cd C:\Deployment\Scripts\Upgrade`
3. Enter the following command: `.\Upgrade-Fresh4.2-updatesites.ps1`
4. If you are using WebLogin, when prompted, enter the username and password.

7.3 Provisioning module

Follow these steps only if the provisioning module is deployed in your tenant.

In this release 2 new columns are added to all the content types:

- Review date
- Content owner

Also, new solutions have been created that, by default, are added to all the sites.

- Media
- Governance

And some webparts of the Core Digital workplace solution have been moved to a new solution called Core website.

These changes affect the site templates that are deployed. The upgrade script will perform the following steps:

- Upload apps file where the apps to install are defined.
- Upload template files where the content types are defined for the following entities:
 - Content site
 - Departments
 - Locations
 - Communities

7.3.1 Preparing the scripts

Note that the templates where the content types are defined will be overridden. If you have customized any of the above content types, you will need to merge the changes before running the script. If you have not made any changes in the content types, you can directly execute the script.

1. Access to your current templates as indicated in the section 9.3
2. Click in ProvisioningTemplates and then click in the customized entity, like CC.ContentSite.PnP, for example. Click the file that ends with "-contenttype.xml" and click "Download" in the opening panel.
3. Merge your changes with the file provided in **Scripts\Upgrade\4.2**
 - a. Note the 2 new columns that have been introduced:

```
<pnp:FieldRef ID="f0b78a7d-5887-4c20-9df0-678d1e4309a3" Name="CandC_ReviewDate" />
<pnp:FieldRef ID="a0eb6a43-aed9-4c3f-a01d-6afa3dc2a7e6" Name="CandC_ContentOwner" />
```

7.3.2 Execute script

1. Open a new Windows PowerShell console as an administrator (right-click, 'Run as Administrator).
2. Navigate to the local location of the **Upgrade** deployment scripts folder. E.g. **cd C:\Deployment\Scripts\Upgrade**
3. Enter the following command: **.\Upgrade-Fresh4.2-provisioning.ps1**
4. If you are using WebLogin, when prompted, enter the username and password.

8 Rollback plan

To come back to the previous version, please follow these steps

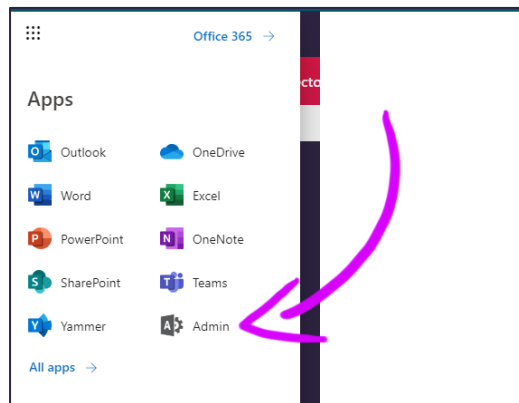
1. Find the previous version of the packages and deploy them to the App catalog.

9 Troubleshooting

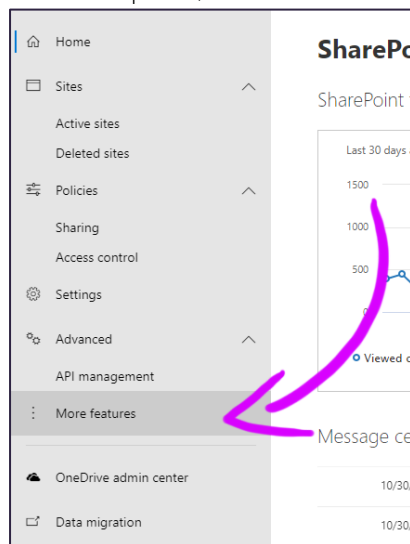
9.1 Check which version of Fresh you are on

You can check the version of Fresh in the App Catalog. Please follow these steps:

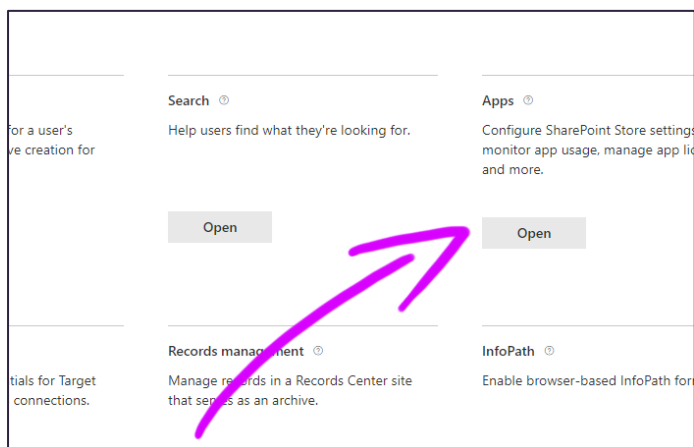
1. To go to the app catalog:
 - a. Click the waffle and then click Admin



- b. In the left panel, expand "Show all" and click "SharePoint" under "Admin centers"
 - c. In the left panel, click "More features"



- d. Click the "Open" button under "Apps"



- e. Click the link "App catalog".
- f. In the left menu, click "Apps for SharePoint"
2. Check the "App version" column of any of the Fresh solutions

Home

Apps for SharePoint ⓘ

Recent

- Apps for SharePoint
- Apps for Office
- App Requests
- Site contents

[New](#)
[Upload](#)
[Sync](#)
[Share](#)
[More](#)

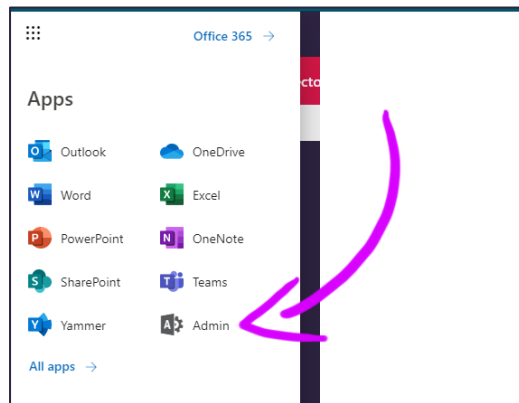
[All Apps](#)
[Featured Apps](#)
[Unavailable Apps](#)
[Find a file](#)

✓	Title	Name	App Version	Edit	Product ID
Product ID : {CDE2E4BE-97A0-49B9-A8C8-0794C07F6307} (1)					
	candc-fresh-globalstructure	candc-fresh-globalstructure	3.3.0.0		{CDE2E4BE-97A0-49B9-A8C8-0794C07F6307}
Product ID : {AA76022B-F3EB-4B27-9545-1B4CBA749354} (1)					
	candc-fresh-contentwebparts	candc-fresh-contentwebparts	3.3.0.0		{AA76022B-F3EB-4B27-9545-1B4CBA749354}
Product ID : {BB76022B-F3EB-4B27-9545-1B4CBA7493BB} (1)					
	candc-fresh-usercentricwebparts	candc-fresh-usercentricwebparts	3.3.0.0		{BB76022B-F3EB-4B27-9545-1B4CBA7493BB}

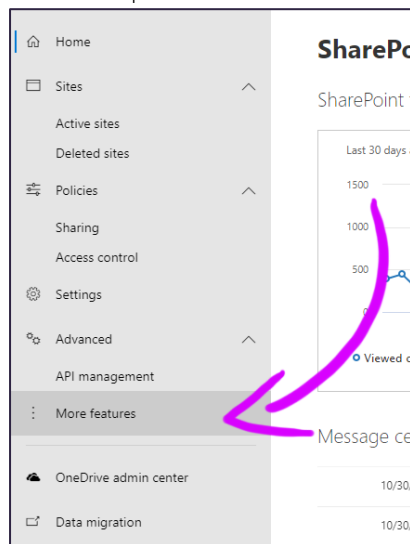
9.2 Manually upload the packages

If for some reason there is a problem uploading the packages to the App Catalog, this step can be done manually following these steps:

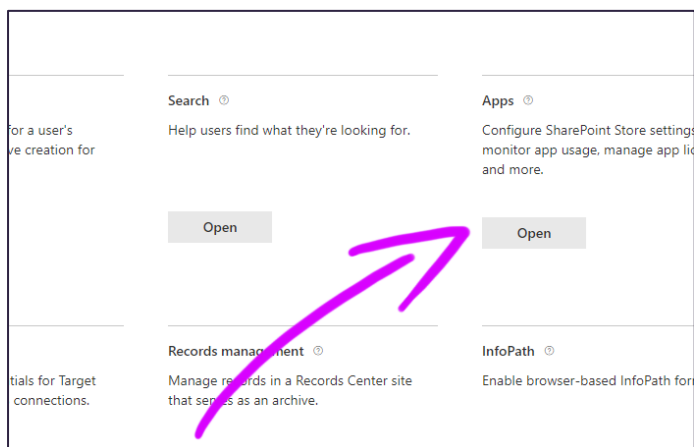
1. Go to the app catalog:
 - a. Click the waffle and then click Admin



- b. In the left panel, expand "Show all" and click "SharePoint" under "Admin centers"
 - c. In the left panel, click "More features"



- d. Click the "Open" button under "Apps"

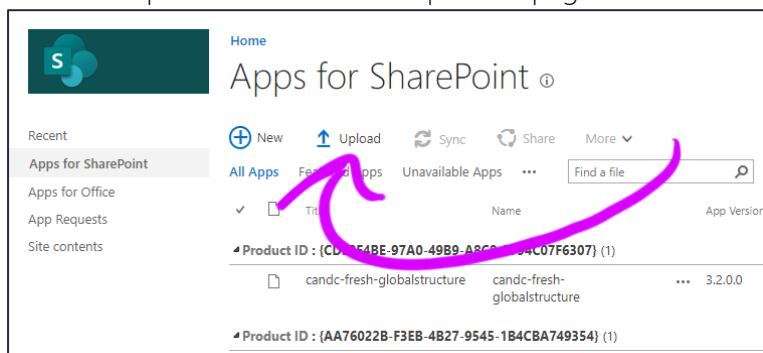


- e. Click the link "App catalog".
- f. In the left menu, click "Apps for SharePoint"

2. Upload the 6 solutions provided in the folder Scripts\SolutionPackages.

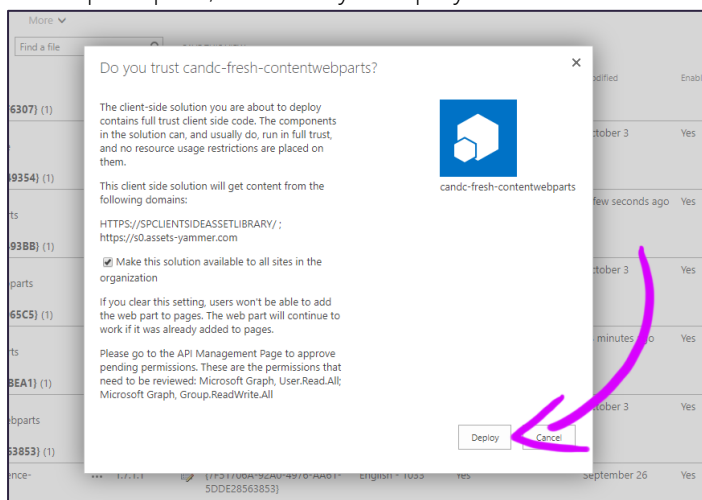
For every solution:

- a. Click the "Upload button" at the top of the page.



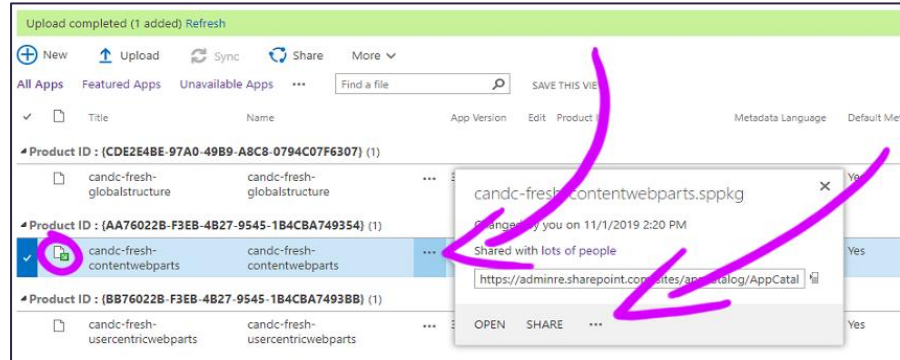
- b. In the pop-up, select the solution and click "OK".

- c. When prompted, click always "Deploy".



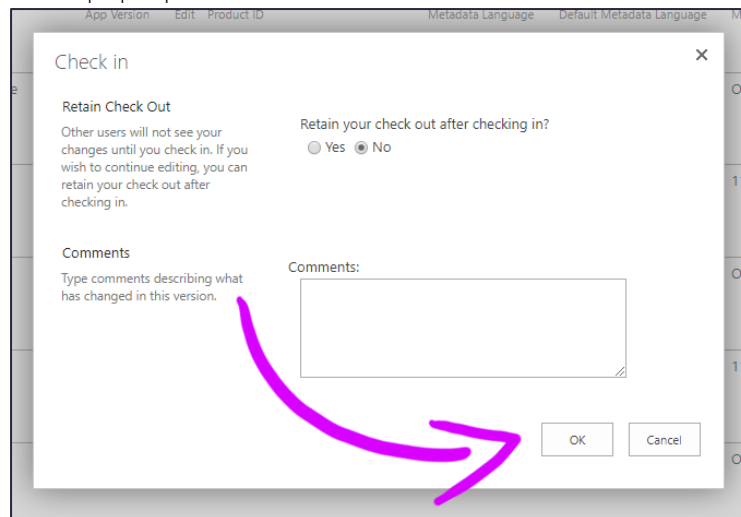
d. If the solution appears with a green box with an arrow inside (check out status), you need to check it in. To do that:

i. Open the contextual menu by clicking in the ellipsis "...". In the dialog that appears, click the ellipsis "...".



ii. In the menu select "Advanced" and then "Check-in".

iii. In the pop-up, click "OK"

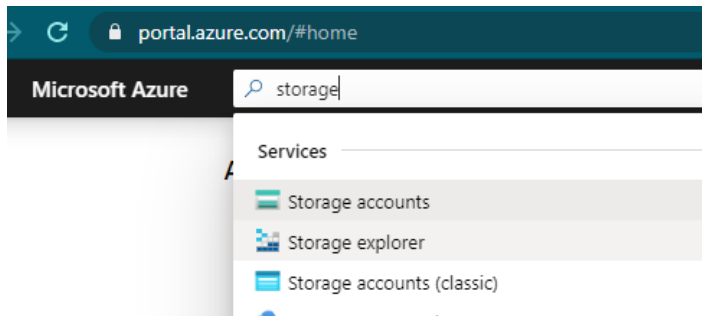


9.3 Access to the Provisioning files

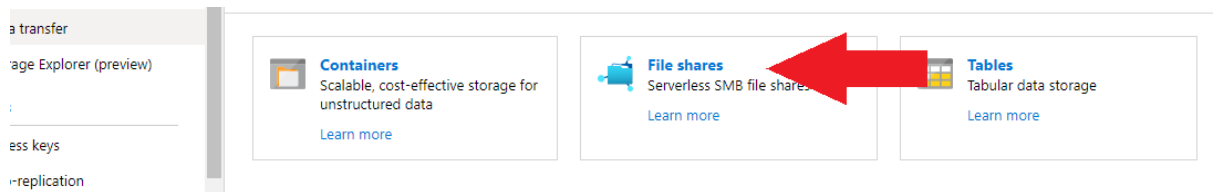
This section describes how to access files to the Provisioning engine through the Azure portal. This method does not require that you install any additional tool on your computer. Alternatively, you could consider using [Azure Storage Explorer](#).

Please follow these steps to access to the Provisioning engine files through the Azure portal:

1. Log in to the Azure portal in the Test tenant: <https://portal.azure.com/>
2. In the top search box write "storage" and then click in "Storage Accounts":



3. Click in the provisioning storage account
4. Click "File shares":



5. Click the first item in the list.
6. Click on the "site" folder. Then the "wwwroot" folder.
 - a. Note that the **bin** folder contains the libraries (dll) that are needed to execute the provisioning engine.
 - b. Note that the **ProvisioningTemplates** folder contains the PnP templates that can be customized.

9.4 Manually upload the Provisioning files

Follow the steps in the previous section to access to the provisioning files.

1. Copy the 2 libraries that are the bin folder in the update scripts to the **bin** folder
 - CandC.FunctionHelpers.dll
 - CandC.ProvisioningFunctions.dll
2. The PnP templates that contain the content types have been updated in this upgrade. These templates need to be overwritten in the storage. In the **wwwroot** folder, click in **ProvisioningTemplates** and overwrite the following files:
 - a. In the **CC.Community.PnP** overwrite the community-contenttype.xml file
 - b. In the **CC.ContentSite.PnP** overwrite the contentsite-contenttype.xml file
 - c. In the **CC.Department.PnP** overwrite the department-contenttype.xml file
 - d. In the **CC.FreshCommon.PnP** overwrite these files
 - apps.pnp.xml
 - sitecolumns.pnp.xml
 - sitecontenttypes.pnp.xml
 - sitelibraries.pnp.xml