



Documentation: **Advanced Sync**

Document version: 28.0.1

Date: 19/10/2026

Author: Dynavision Product Team

1	Introduction.....	2
2	Installation	2
2.1	Extension	2
3	Configuration	3
3.1	Service-to-Service Authentication	3
3.2	PULL – Slave pulls data from Master	4
3.2.1	Create App Registration in master tenant	4
3.2.2	Create an Entra Application in the master Business Central tenant and grant consent.....	7
3.2.3	Create Authentication Setup in slave tenant.....	8
3.2.4	Set up API URL in slave company.....	9
3.3	PUSH – Master pushed data to Slave	11
3.3.1	Create App Registration in slave tenant	11
3.3.2	Create an Entra Application in the slave Business Central tenant and grant consent 14	
3.3.3	Create Authentication Setup in master tenant.....	15
3.3.4	Set up API URL in master company	16
3.4	Synchronization Setup	18
3.4.1	Create the setup using the wizard	18
3.4.2	Creating the setup manually	19
3.4.3	Enable/disable.....	20
3.5	Synchronization Mappings	21
3.6	Synchronization job queue entry	22
3.7	Retention policy.....	22
3.8	Export feature on Synchronization Setup.....	23
3.8.1	What's included in the export?.....	23

1 Introduction

Welcome to the official manual of the Dynavision Advanced Sync module.

This extension provides functionalities for syncing information between Business Central and another endpoint.

2 Installation

2.1 Extension

The Dynavision Advanced Sync module is a separate Business Central extension.

1. Choose the search icon, enter **Extension Management**, and then choose the related link.
2. Choose **Manage** in the Action bar on the page and choose action **Extension Marketplace**.
3. In the search bar, enter **Dynavision Advanced Sync** and install this app.
4. Activate the trial as described in following [Get Started guide](#).

3 Configuration

3.1 Service-to-Service Authentication

Service-to-Service authentication is used in scenarios where there is no user interaction required. This authentication enables access to API's using the identity of an application instead of a user.

In order to set up this authentication, the following steps need to be performed:

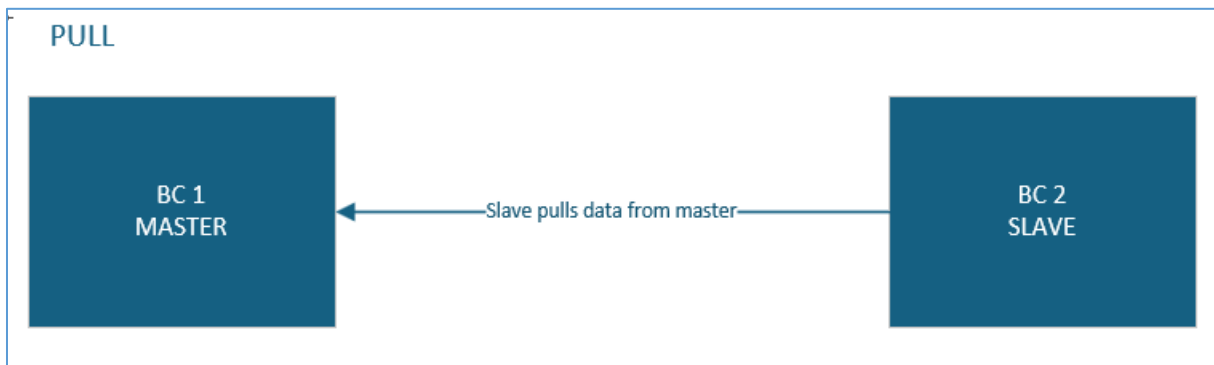
1. Create an App Registration in Azure.
2. Create an Entra Application in Business Central and grant consent.

When it comes to licensing, this authentication method also requires an additional license:

If an application connects to Business Central with a single account, then that's considered multiplexing. In itself, this is not prohibited. However, it does not reduce the number of required licenses. All users accessing Business Central need to be properly licensed. No matter if they access Business Central with their own credentials or by sharing a single user.

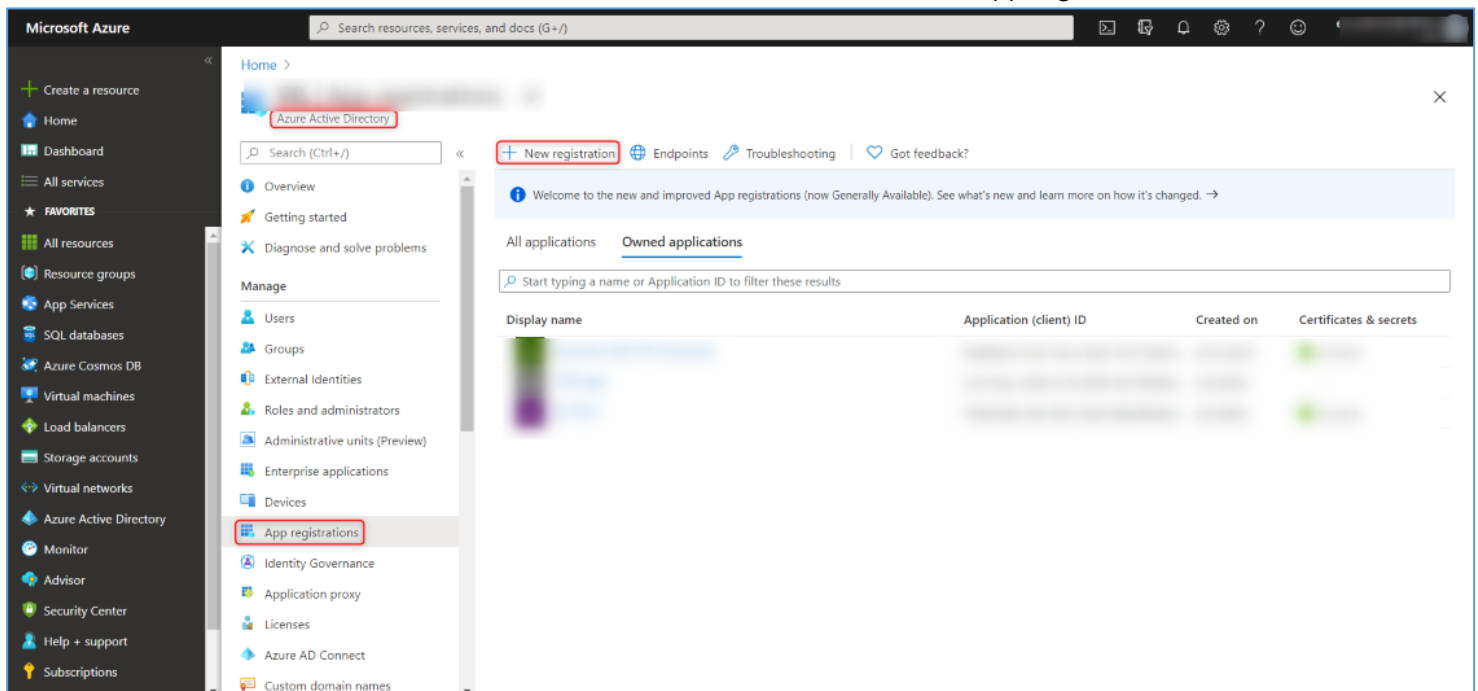
See the following [blog](#) for more info about licensing.

3.2 PULL – Slave pulls data from Master



3.2.1 Create App Registration in master tenant

1. Navigate to the Azure portal.
2. Search and open the **App Registrations** page.
3. Choose the action **New Registration**.
 - a. *Make sure that the master tenant ID is noted in the app registration.*



4. Choose one of the following supported account types:
 - a. Accounts in this organizational directory only ([organization] – Single tenant) : Select if the external application will only be used inside your organization.
 - b. Accounts in any organizational directory (Any Azure AD directory – Multitenant) : Select if other organizations should access the external application.

Microsoft Azure

Home > VIL | App registrations >

Register an application

* Name

The user-facing display name for this application (this can be changed later).

TEST

Supported account types

Who can use this application or access this API?

☒ Accounts in this organizational directory only (VIL only - Single tenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

Help me choose...

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web e.g. https://myapp.com/auth

By proceeding, you agree to the Microsoft Platform Policies

Register

5. Choose the action **Register** to create the application.
6. Copy the **Application (Client) ID**, and save this information for later use.
7. Navigate to the Tab **Authentication**, in the navigation menu and add a Platform.

Microsoft Azure

Home > VIL | App registrations >

TEST | Authentication

Search (Ctrl+/)

Save Discard Got feedback?

Overview

Quickstart

Integration assistant (preview)

Manage

Branding

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

Owners

Roles and administrators (Preview)

Manifest

Support + Troubleshooting

Troubleshooting

New support request

Platform configurations

Depending on the platform or device this application is targeting, add redirect URIs, specific authentication settings, or fields specific to the platform

+ Add a platform

Supported account types

Who can use this application or access this API?

☒ Accounts in this organizational directory only (VIL only - Single tenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multi

Help me decide...

Due to temporary differences in supported functionality, we don't recommend accounts for an existing registration. If you need to enable personal accounts, [Learn more about these restrictions.](#)

Advanced settings

Default client type

Treat application as a public client. Required for the use of the following flows where a redirect URI is not used

Configure platforms

Web applications

Web
Build, host, and deploy a web server application. .NET, Java, Python

Single-page application
Configure browser client applications and progressive web applications. Javascript.

Mobile and desktop applications

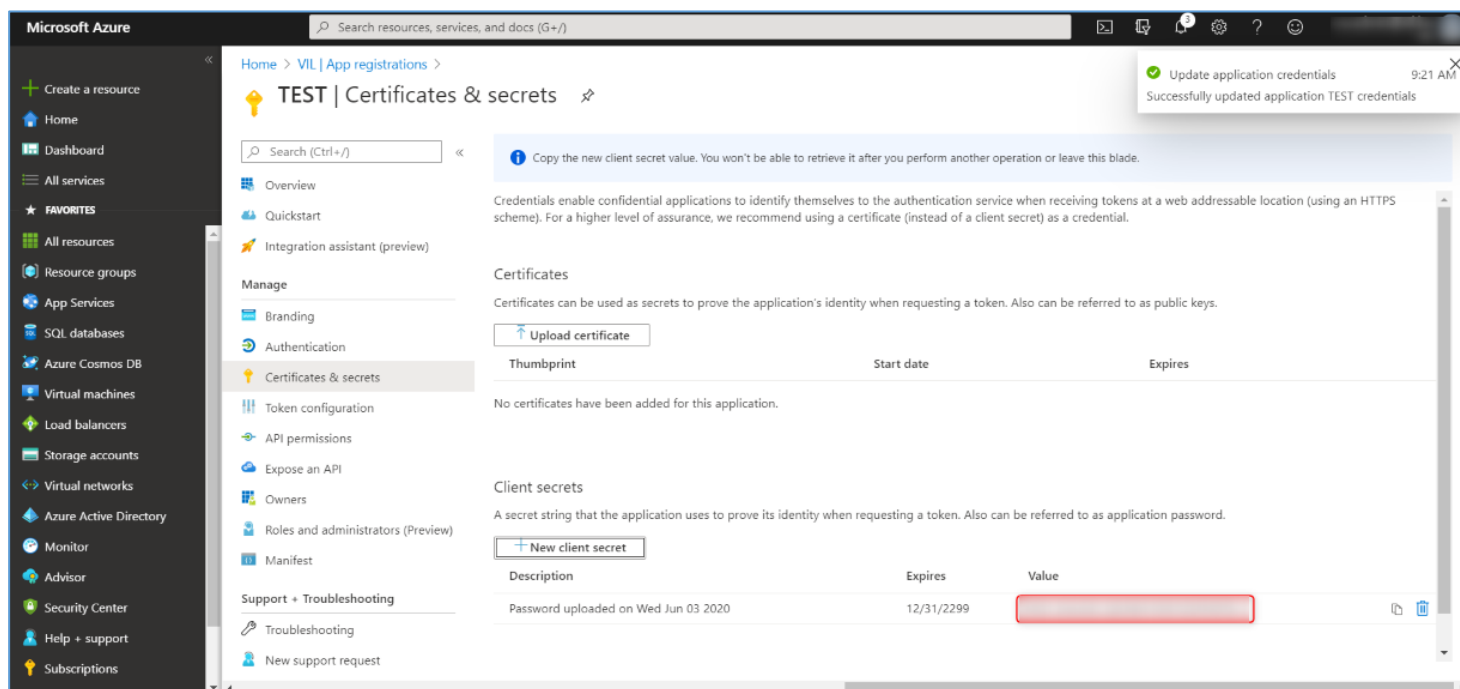
iOS / macOS
Objective-C, Swift, Xamarin

Android
Java, Kotlin, Xamarin

Mobile and desktop applications
Windows, UWP, Console, IoT & Limited-entry Devices, Classic iOS + Android

8. Copy the **Redirect URL** and save this information for later use.
 - a. In case **API** is used for communication, use <https://businesscentral.dynamics.com/OAuthLanding.htm> as redirect URL.
9. Navigate to the Tab **Certificates & secrets**.

10. Assign the correct permissions to the application.
 - a. Open the created application and navigate to **API Permissions**, in the navigation pane.
 - b. Choose the action **Add a permission** and select the **Dynamics 365 Business Central API**.
 - c. Select **Application Permissions** (not delegated permissions) when asked for the type of permissions to configure.
 - d. Check the **API.ReadWrite.All** permission.
 - e. Choose the action **Add permissions**.
 - f. Select the **API.Readwrite.All** permission and choose **Grant admin consent for [organization]**.
11. Create a **New Client Secret** that will be used in the slave tenant to connect to the master tenant. This is not needed when a *mobile and desktop application* is created as platform. Save the secret immediately, because this will only be displayed once.
 - a. Navigate to **Certificates & secrets** in the app registration.
 - b. Choose the action **New client secret**.
 - c. Enter a **Description** and select an **Expiration period**.
 - d. Choose **Add**.
 - e. Copy the Value and store somewhere safely. This will be needed later to perform the API calls from slave to master.



3.2.2 Create an Entra Application in the master Business Central tenant and grant consent

1. Navigate to the page **Microsoft Entra Applications** in the master Business Central tenant.
2. Choose the action **New**.
3. Fill in the **Client ID**. This can be found on the Overview page of the Azure App Registration, in the field Application (Client) ID.
4. Choose the action **Grant Consent**.
5. Enter the **Login Credentials** in the login page that is shown. Only users with the following permissions will be able to login to grant consent.
 - a. Global Administrator,
 - b. Application Administrator, and
 - c. Cloud Application Administrator.
6. **Accept** the request permissions and enable the application.
7. Add the system permission **D365 Automation**. This permission set included the necessary rights for the table and table data.
8. Add a permission set that exposes the required API pages. *Example given: in case of the Dynavision Message API, add the ESCA CORE, EDIT permission set.*

Remark: it is important to add the **minimum permissions required for the scenario to work**. See the following resources for more info:

 - a. [Microsoft Learn](#).
 - b. [Kauffmann documentation](#).

3.2.3 Create Authentication Setup in slave tenant

1. In the slave Business Tenant, choose the search icon, enter **Authentication Setup List**, and then choose the related link.
2. Choose the action **New**, to create a new Authentication Setup.
3. Fill in the information needed according to the **Authorization Type**. For the communication between Business Central tenants, use **Business Central OAuth2**.

3.2.3.1 Communication between Business Central tenants

a. Authorization Type: Business Central OAuth2

i. OAuth2 Grant Type

Specifies the OAuth2 Grant Type.

For communication between Business Central Tenants, select **Client Credentials**.

ii. Tenant

Specifies the tenant for which the Authentication is set up. This is the tenant ID of the slave Business Central tenant. *Can be found in the URL of the environment.*

iii. Client Id

Specifies the Client ID of the App registration. This is the ID that is copied from the App registration Setup.

iv. Client Secret

Specifies the Client Secret of the App registration. This is the secret that is copied when the App registration secret was created.

v. Authentication Provider URL

Specifies the Authentication Provider URL. This is the URL that will be used to authenticate the communication. *This URL will be filled in automatically and has e.g. following format*
https://login.microsoftonline.com/TenantID/oauth2/v2.0/token.

vi. Redirect URL

Specifies the Redirect URL. This is the Redirect URL that was copied from the App registration.

vii. Scope

Specifies the Scope. This is the scope that contains the API's that will be used to communicate, using the authenticated connection. *This will be automatically filled in.*

viii. Access Token expires at

Shows when the Access Token expires. If this is blank, a new token is requested each time. This can be updated using the action **Get authentication String**.

ix. Initialized

Indicates if additional steps (grant consent,...) are required before the authentication setup can be used.

- b. For the **Authentication Type Business Central OAuth2**, the supplied information can be checked by using the action **Test Connection**.

3.2.3.2 Other types of communication

- c. **Authorization Type: Basic**
 - i. **Web Service Username**
Specifies the Web Service Username.
 - ii. **Web Service Access Key**
Specifies the Web Service Access Key.
- d. **Authorization Type: OAuth2**
 - i. See the setup of Business Central OAuth2 for the explanation of the fields.
- e. For the **Authentication Type OAuth2** the supplied information can be checked by using the action **Test Connection**.

3.2.4 Set up API URL in slave company

Set up the API URL's that will be needed to pull the information from the master tenant into the slave tenant.

1. In the slave Business Central tenant, choose the search icon, enter **API URL**, and choose the related link.
2. Either add the line for the API URL manually, or choose the action **Create API URL...** to start up the wizard to create the API URL.
3. Fill in the information requested on the list page or the wizard. Below is described what is asked for in the wizard.
 - a. **API Code**
Specifies the code that will be used to identify the API URL.
 - b. **Authentication Code**
Specifies the Authentication code, used to authenticate the communication with the master tenant that is executed with the API URL. This has been set up in the previous section.
 - c. **TenantId**
Specifies the Tenant ID of the Business Central Tenant that will communicate with the set up API URL. The current tenant ID will be filled in by default.
 - d. **Environment**
Enter the name of the Tenant with which the communication is set up. *E.g. the name of the other tenant is DynavisionSlave, then this is the Environment that will be filled in in this field.*
 - e. **Company Name**
Specifies the name of the company in the other tenant, to which this company will be "connected". Using the lookup function, all companies in the previously supplied environment are displayed to choose from.
 - f. **Page Id**
Specifies the API Page ID. *Attention: the presumption here is that the API pages on both environments are the same. It is also possible to view all API pages that can be selected using the lookup functionality in the field.*
 - g. **Page Name**
Displays the name of the selected API Page ID.

h. API URL

Displays the API URL that will be created based on the entered parameters in the setup wizard.

Remark: if this API URL is filled in, the process has been completed successfully.

4. After filling in the information in the wizard and choosing **Finish**, the line is created for the API URL.

a. Source Table No.

Specifies the source table of the API page.

b. CultureInfo

Specifies the culture that is used to format the date values. This will be filled in with the set **Region** on the tenant.

This is used to correctly format the information that is synced between the different tenants.

5. Once this information is set up, this can be used in the synchronization setup.

3.2.4.1 Change environment

It is possible to change the environment name for selected API URL's. This is possible by selecting the API url lines, and choosing **Change environment**. For each found environment name, a line is listed to change the name.

- **Old Environment**

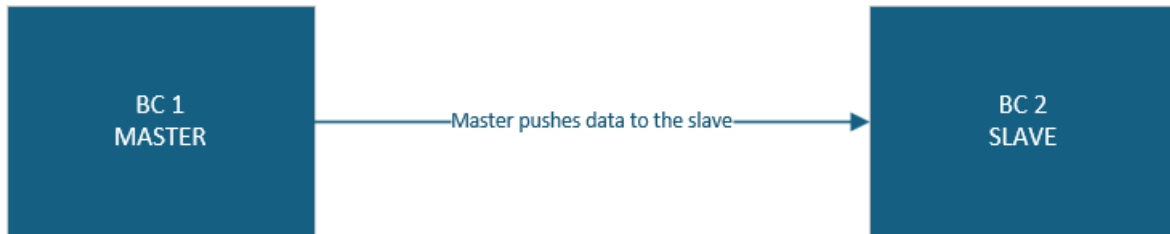
Displays the current environment name on the selected API url(s).

- **New Environment**

Type the name of the new environment.

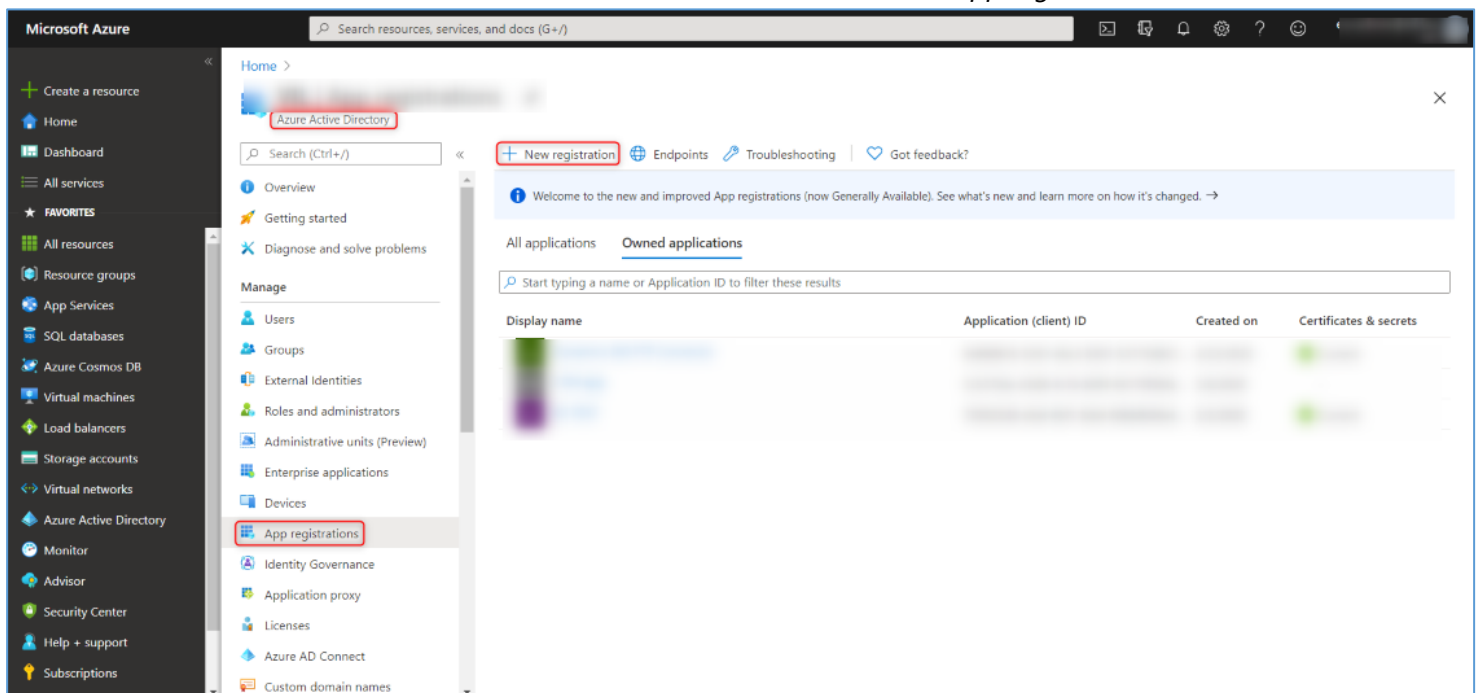
3.3 PUSH – Master pushed data to Slave

PUSH



3.3.1 Create App Registration in slave tenant

1. Navigate to the Azure portal.
2. Search and open the **App Registrations** page.
3. Choose the action **New Registration**.
 - a. *Make sure that the slave tenant ID is noted in the app registration.*



4. Choose one of the following supported account types:
 - a. Accounts in this organizational directory only ([organization] – Single tenant) :
Select if the external application will only be used inside your organization.

- b. Accounts in any organizational directory (Any Azure AD directory – Multitenant) : Select if other organizations should access the external application.

Microsoft Azure

Search resources, services, and docs (G+J)

Home > VIL | App registrations >

Register an application

* Name

The user-facing display name for this application (this can be changed later).

TEST

Supported account types

Who can use this application or access this API?

☒ Accounts in this organizational directory only (VIL only - Single tenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

Help me choose...

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web e.g. https://myapp.com/auth

By proceeding, you agree to the Microsoft Platform Policies

Register

5. Choose the action **Register** to create the application.
6. Copy the **Application (Client) ID**, and save this information for later use.
7. Navigate to the Tab **Authentication**, in the navigation menu and add a Platform.

Microsoft Azure

Search resources, services, and docs (G+J)

Home > VIL | App registrations >

TEST | Authentication

Search (Ctrl+J)

Save Discard Got feedback?

Overview

Quickstart

Integration assistant (preview)

Manage

Branding

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

Owners

Roles and administrators (Preview)

Manifest

Support + Troubleshooting

Troubleshooting

New support request

Platform configurations

Depending on the platform or device this application is targeting, add redirect URIs, specific authentication settings, or fields specific to the platform

+ Add a platform

Supported account types

Who can use this application or access this API?

☒ Accounts in this organizational directory only (VIL only - Single tenant)

☐ Accounts in any organizational directory (Any Azure AD directory - Multi)

Help me decide...

Due to temporary differences in supported functionality, we don't recommend accounts for an existing registration. If you need to enable personal accounts, [Learn more about these restrictions.](#)

Advanced settings

Default client type

Treat application as a public client.

Required for the use of the following flows where a redirect URI is not used

Configure platforms

Web applications

Web

Build, host, and deploy a web server application. .NET, Java, Python

Single-page application

Configure browser client applications and progressive web applications. Javascript.

Mobile and desktop applications

iOS / macOS

Objective-C, Swift, Xamarin

Android

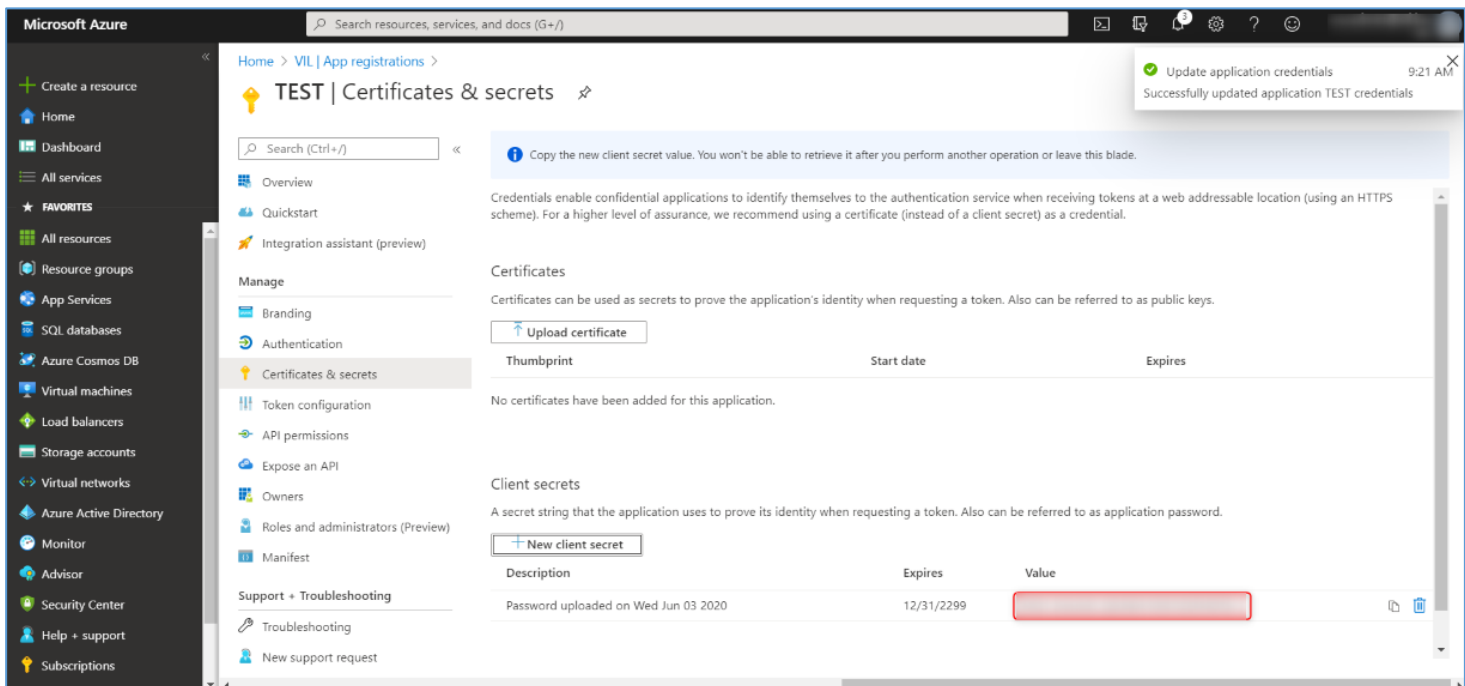
Java, Kotlin, Xamarin

Mobile and desktop applications

Windows, UWP, Console, IoT & Limited-entry Devices, Classic iOS + Android

8. Copy the **Redirect URL** and save this information for later use.
 - a. In case **API** is used for communication, use <https://businesscentral.dynamics.com/OAuthLanding.htm> as redirect URL.
9. Navigate to the Tab **Certificates & secrets**.
10. Assign the correct permissions to the application.
 - a. Open the created application and navigate to **API Permissions**, in the navigation pane.

- b. Choose the action **Add a permission** and select the **Dynamics 365 Business Central API**.
 - c. Select **Application Permissions** (not delegated permissions) when asked for the type of permissions to configure.
 - d. Check the **API.ReadWrite.All** permission.
 - e. Choose the action **Add permissions**.
 - f. Select the **API.Readwrite.All** permission and choose **Grant admin consent for [organization]**.
11. Create a **New Client Secret** that will be used in the master tenant to connect to the slave tenant. This is not needed when a *mobile and desktop application* is created as platform. Save the secret immediately, because this will only be displayed once.
 - a. Navigate to **Certificates & secrets** in the app registration.
 - b. Choose the action **New client secret**.
 - c. Enter a **Description** and select an **Expiration period**.
 - d. Choose **Add**.
 - e. Copy the Value and store somewhere safely. This will be needed later to perform the API calls from master to slave.



3.3.2 Create an Entra Application in the slave Business Central tenant and grant consent

1. Navigate to the page **Microsoft Entra Applications** in the slave Business Central tenant.
2. Choose the action **New**.
3. Fill in the **Client ID**. This can be found on the Overview page of the Azure App Registration, in the field Application (Client) ID.
4. Choose the action **Grant Consent**.
5. Enter the **Login Credentials** in the login page that is shown. Only users with the following permissions will be able to login to grant consent.
 - a. Global Administrator,
 - b. Application Administrator, and
 - c. Cloud Application Administrator.
6. **Accept** the request permissions and enable the application.
7. Add the system permission **D365 Automation**. This permission set included the necessary rights for the table and table data.
8. Add a permission set that exposes the required API pages. *Example given: in case of the Dynavision Message API, add the ESCA CORE, EDIT permission set.*

Remark: it is important to add the **minimum permissions required for the scenario to work**. See the following resources for more info:

- a. <https://learn.microsoft.com/en-us/dynamics365/business-central/dev-itpro/administration/automation-apis-using-s2s-authentication#task-2-set-up-the-azure-ad-application-in->
- b. <https://www.kauffmann.nl/2021/07/02/service-to-service-authentication-in-business-central-18-3-usage-and-license-terms/>

3.3.3 Create Authentication Setup in master tenant

2. In the master Business Tenant, choose the search icon, enter **Authentication Setup List**, and then choose the related link.
3. Choose the action **New**, to create a new Authentication Setup.
4. Fill in the information needed according to the **Authorization Type**. For the communication between Business Central tenants, use **Business Central OAuth2**.

3.3.3.1 Communication between Business Central tenants

a. Authorization Type: Business Central OAuth2

i. OAuth2 Grant Type

Specifies the OAuth2 Grant Type.

For communication between Business Central Tenants, select **Client Credentials**.

ii. Tenant

Specifies the tenant for which the Authentication is set up. This is the tenant ID of the master Business Central tenant. *Can be found in the URL of the environment.*

iii. Client Id

Specifies the Client ID of the App registration. This is the ID that is copied from the App registration Setup.

iv. Client Secret

Specifies the Client Secret of the App registration. This is the secret that is copied when the App registration secret was created.

v. Authentication Provider URL

Specifies the Authentication Provider URL. This is the URL that will be used to authenticate the communication. *This URL will be filled in automatically and has e.g. following format*
<https://login.microsoftonline.com/TenantID/oauth2/v2.0/token>.

vi. Redirect URL

Specifies the Redirect URL. This is the Redirect URL that was copied from the App registration.

vii. Scope

Specifies the Scope. This is the scope that contains the API's that will be used to communicate, using the authenticated connection. *This will be automatically filled in.*

viii. Access Token expires at

Shows when the Access Token expires. If this is blank, a new token is requested each time. This can be updated using the action **Get authentication String**.

ix. Initialized

Indicates if additional steps (grant consent,...) are required before the authentication setup can be used.

- b. For the **Authentication Type Business Central OAuth2**, the supplied information can be checked by using the action **Test Connection**.

3.3.3.2 Other types of communication

- c. **Authorization Type: Basic**
 - i. **Web Service Username**
Specifies the Web Service Username.
 - ii. **Web Service Access Key**
Specifies the Web Service Access Key.
- d. **Authorization Type: OAuth2**
 - i. See the setup of Business Central OAuth2 for the explanation of the fields.
- e. For the **Authentication Type OAuth2** the supplied information can be checked by using the action **Test Connection**.

3.3.4 Set up API URL in master company

Set up the API URL's that will be needed to push the information from the master tenant to the slave tenant.

1. *In the master Business Central tenant*, choose the search icon, enter **API URL**, and choose the related link.
2. Either add the line for the API URL manually, or choose the action **Create API URL...** to start up the wizard to create the API URL.
3. Fill in the information requested on the list page or the wizard. Below is described what is asked for in the wizard.
 - a. **API Code**
Specifies the code that will be used to identify the API URL.
 - b. **Authentication Code**
Specifies the Authentication code, used to authenticate the communication with the slave tenant that is executed with the API URL. This has been set up in the previous section.
 - c. **TenantId**
Specifies the Tenant ID of the Business Central Tenant that will communicate with the set up API URL. The current tenant ID will be filled in by default, but will need to be changed to the tenant ID of the slave company if this differs.
 - d. **Environment**
Enter the name of the Tenant with which the communication is set up. *E.g. the name of the other tenant is DynavisionSlave, then this is the Environment that will be filled in in this field.*
 - e. **Company Name**
Specifies the name of the company in the other tenant, to which this company will be "connected". Using the lookup function, all companies in the previously supplied environment are displayed to choose from.
 - f. **Page Id**
Specifies the API Page ID. *Attention: the presumption here is that the API pages on both environments are the same. It is also possible to view all API pages that can be selected using the lookup functionality in the field.*
 - g. **Page Name**
Displays the name of the selected API Page ID.

h. API URL

Displays the API URL that will be created based on the entered parameters in the setup wizard.

Remark: if this API URL is filled in, the process has been completed successfully.

4. After filling in the information in the wizard and choosing **Finish**, the line is created for the API URL.

a. Source Table No.

Specifies the source table of the API page.

b. CultureInfo

Specifies the culture that is used to format the date values. This will be filled in with the set **Region** on the tenant.

This is used to correctly format the information that is synced between the different tenants.

5. Once this information is set up, this can be used in the synchronization setup.

3.3.4.1 Change environment

It is possible to change the environment name for selected API URL's. This is possible by selecting the API url lines, and choosing **Change environment**. For each found environment name, a line is listed to change the name.

- **Old Environment**

Displays the current environment name on the selected API url(s).

- **New Environment**

Type the name of the new environment.

3.4 Synchronization Setup

The setup of the tables that has to be synced and the API URL that needs to be used can be specified in the **Synchronization setup**.

In case the Slave Pulls the data from the master tenant, create the synchronization setup in the slave tenant using **Operation Get: Filter**.

In case the Master tenant is Pushing the data: create the synchronization setup in the Master tenant, using Operation **Post / Update** or **Post & Update**.

1. Choose the search icon, enter **Synchronization Setup**, and then choose the related link.
2. Either choose the action **Create Synchronization Setup...** to [use a wizard](#) for setting up the sync information, [or fill the information in manually](#).

3.4.1 Create the setup using the wizard

1. Choose the action **Create Synchronization Setup** (which can be found by choosing the "arrow" next to the New action).
2. Fill in the information requested in the wizard.
 - a. **Authentication Code**
Choose the authentication setup to be used to authenticate for the communication.
 - b. **TenantId**
This is the unique identifier of the tenant from which the connection is set.
This can be found in the URL of the currently opened Dynavision tenant after the after the first / until the next /. (So after businesscentral.dynamics.com/.)
 - c. **Environment**
Fill in the environment to which the connection needs to be made.
 - d. **Company Name**
Select the company to which the connection needs to be made in the specified environment.
3. After filling in the information on the screen, choose next to move on to the next step. Select the **API page(s)** that need to be added as a synchronization setup entry. This can be done by selecting the line, and choosing to **Include** the API.
4. Choose **Finish** to close the wizard. The synchronization setup entries will be added.

3.4.2 Creating the setup manually

Fill in or complete the information for the Synchronization.

- 1. Code**
Specifies the code of the Synchronization Setup.
- 2. Table Caption**
Specifies the table that is being Synchronized.
- 3. Enabled**
Specifies if the Synchronization Setup is enabled.
- 4. Requires Context**
Specifies if the Synchronization setup is executed via code and cannot be executed individually.
- 5. API URL Code**
Specifies the Code of the API URL used for synchronization.
- 6. API URL**
Specifies the API URL used for synchronization.
- 7. Operation**
Specifies the type of operation for the synchronization.
The options are:
 - a. Post & update**
Information will be posted from the current environment to the other endpoint. This will create or update records.
 - b. Post**
Information will be posted from the current environment to the other endpoint. This will create records.
 - c. Update**
Information will be posted from the current environment to the other endpoint. This will update records.
 - d. Get: Filter**
Will fetch information from the other endpoint, using the specified **OData Filter** in the field **Filter**. Using the fetched information, records will be created, updated or deleted in the current environment, based on the values set in the next three columns.
***Remark:** The filter will be automatically formatted to OData filter format. More information can be found by clicking following link: [Using Filter Expressions in OData URIs - Business Central | Microsoft Learn](#).*
- 8. Insert**
Specifies if new external records are inserted locally.
- 9. Modify**
Specifies if external modified records are modified locally.
- 10. Delete**
Specifies if external deleted records are deleted locally.
- 11. Last Synchronization On**
Specifies when the last synchronization occurred.
- 12. Only Process Modified Data**
Specifies that only modified records will be processed. This requires the Modified At

field to be included in the mappings when data is imported. When exporting data, the value of the Source Date Field No. is used.

13. Skip Token

Specifies the Skip Token, that is used for server side paging.

14. Filter

Specifies the filter applied to local records before sending, or applied to records when fetching information.

15. Pre-processor

Specifies how the records are pre-processed before they are synchronized.

16. No. Of Templates

Specifies the number of linked templates. These templates are applied before the post-processor is executed.

17. Re-Apply Template

Specifies if the template has to be re-applied for existing records.

18. Post-Processor

Specifies how the records are post-processed after they are synchronized.

19. Error Message

Specifies the error message when an error has occurred during processing.

3.4.3 Enable/disable

Use the function **Enable** or **Disable** to enable or disable the selected synchronization setup lines.

3.5 Synchronization Mappings

The synchronization mappings need to be set up for the **Synchronization Setup**, in order to map the fields that are synchronized between the two endpoints.

1. Choose the action **Synchronization Mappings** on the page **Synchronization Setup**. This will open the **Synchronization Mappings** page for the selected **Synchronization Setup**.
2. Add all API Fields that need to be mapped.
 - a. **API Field Name**
Specifies the name of the field in the external API.
 - b. **Enabled**
Specifies if the mapping is enabled.
 - c. **Is Master System Id**
If enabled, this field will be used as OData key field instead of the primary key fields. This also needs to be activated when delete actions are used.
 - d. **Skip field Validation**
Specifies if the validation of the field should be skipped.
 - e. **Field Caption**
Specifies the field that needs to be mapped with the API Field.
 - f. **Update Field From Response**
Specifies if the field should be updated with the corresponding value in the API response.
 - g. **Calculated Value**
Specifies a calculated value that will be used to fill the field.
The options are:
 - i. Street,
 - ii. House No.,
 - iii. VAT No.,
 - iv. ISO Language Code,
 - v. Contact Account ID,
 - vi. Feature Property Type,
 - vii. Opportunity Account ID,
 - viii. Opportunity Contact ID,
 - ix. Opportunity Property ID,
 - x. Customer default Contact ID,
 - xi. Next Quotation Number,
 - xii. Contact Last Name,
 - xiii. Production BOM Description,
 - xiv. Production BOM Unit of Measure,
 - xv. Production BOM Sales Price,
 - xvi. Quotation Property Value,
 - xvii. Property Value,
 - xviii. Ship-To Address Customer No.,
 - xix. Ship-To Address Code.

h. Fix Value

Specifies a fix value that will be used to fill the field.

i. No. of Mappings

Specifies the number of the configured **Sync Field Value Mappings** for this Synchronization mapping field. The From Value and To Value fields currently only support text fields.

j. Example Data

Specifies example data for the mapping.

k. Validation Failed

Specifies if the example data was correctly validated. The example data validation can be triggered by choosing the action **Validate Example Data**.

Remarks:

- *When fields are exported using Synchronization Mappings, you can either use Calculated value OR Fixed value OR sync field mappings. It is not possible to use a combination of these values in the export of values.*
- *When data is imported, following logic is executed: first the sync field mapping that is configured is executed, and on this value, the calculated value can be applied. This information will then be saved in the field. If a fixed value is used, this value will be saved in the field.*

3.6 Synchronization job queue entry

The synchronization of records can be set up using a job queue entry report **71177401**.

In this job queue entry, it is possible to Synchronize all entries, without looking at the last synced date for the entries.

1. Navigate to the **Job Queue Entry Card** of the Execute Synchronization report.
2. Choose **Job Queue, Report Request Page**.
3. Enable or disable the option **Synchronize All Records**. This Specifies if all records are synchronized. The last synchronization date will be ignored.

3.7 Retention policy

To make sure that the entries will be cleaned up after a certain period of time, the retention policy needs to be set up.

1. Navigate to the page **Retention Policies**, using the search function of Business Central.
2. Search for the Table ID **71176855** Entry To Synchronize and open the retention policy card page.
3. Choose the **Retention Period** and **Enable** the retention policy to have a regular cleanup of the entries.

3.8 Export feature on Synchronization Setup

A new **Export Advanced Sync Data** action has been added to the **Synchronization Setup** page. It exports all relevant Advanced Sync configuration and diagnostics into a single .zip file that can be shared with support for troubleshooting, without needing direct database access.

Important Remark: *the exported data may contain sensitive/confidential data and should be treated with the necessary care.*

1. Choose the search icon, enter **Synchronization Setup**, and then choose the related link.
2. Choose the action **Support > Export Advanced Sync Data**.
 - a. *The action is only available for users that have Modify permission on the ESCY Sync Setup table.*
3. Enable or disable the option **Export Web Request Logs**.
 - a. This option specifies whether to export web request log entries.
The logs are automatically filtered from the earliest last synchronization date across all sync setups. *These can be large, so can be disabled if only the configuration data needs to be exported.*
4. Choose **OK** to export the information into a ZIP file.

3.8.1 What's included in the export?

The .zip file contains the following JSON files:

File	Contents
Synchronization Setup.json	All sync setup records with their full configuration (API URL code, operation, filters, processors, enabled state, error messages, error callstacks, last sync dates, etc.). Also includes environment info: tenant URL, environment name, whether it's production/SaaS, who downloaded the export, and the export timestamp.
Synchronization Mappings.json	All field mapping lines across all sync setups (BC field to API field name, key field flags, enabled state, update-from-response flags).
Synchronization Field Value Mappings.json	All value translation mappings (how BC field values map to API values and vice versa).
Synchronization Config Templates.json	Configuration templates applied to imported records during sync.
API URLs.json	The API URL configurations referenced by the sync setups (URL, authorization type, culture info, source table). Only includes API URLs actually used by a sync setup.
Web Request Log.json (optional)	HTTP request/response logs filtered to Advanced Sync entries only. Automatically scoped to the date range of the earliest to latest Last Synchronization On across all sync setups, so only relevant logs are included.

Every record also includes SystemModifiedAt (when it was last changed) and SystemModifiedByUser (who changed it, shown as the username rather than a GUID).