



# Odin Automation Premium

Office 365 Integration 17.2 Provider's Guide

Revision 19.23 (December 21, 2017)

Ingram Micro Inc.  
3351 Michelson Drive, Suite 100 Irvine, CA 92612 United States  
Tel: +1 714-566-1000

© 2017 Ingram Micro Inc. All rights reserved.

Distribution of this work or derivative of this work in any form is prohibited unless prior written permission is obtained from the copyright holder.

{ Patented technology protected by U.S. Patents 7,475,157; 7,509,369; 7,613,793; 8,171,385; 8,925,075.  
Patents pending in the U.S. }

Odin, Service Automation, the Odin logos are registered trademarks of Ingram Micro Inc. in the United States and other countries. All other trademarks are the property of their respective owners.

# Contents

|  |           |
|--|-----------|
| <b>Preface</b> .....                                 | <b>6</b>  |
| Documentation Conventions.....                       | 6         |
| Typographical Conventions .....                      | 6         |
| General Conventions .....                            | 7         |
| Feedback.....  | 7         |
| <b>Introduction</b> .....                            | <b>8</b>  |
| Audience .....                                       | 8         |
| Technologies.....                                    | 8         |
| Operations Available to Office 365 Subscribers ..... | 8         |
| Supported Languages.....                             | 9         |
| Abbreviations .....                                  | 9         |
| Helpful Links .....                                  | 9         |
| <b>Cloud Solution Provider Scenario</b> .....        | <b>11</b> |
| Integration Overview.....                            | 11        |
| General Architecture.....                            | 12        |
| Basic Partnership Scenario.....                      | 13        |
| Basic Object Model .....                             | 14        |
| Support for Account-Wide Services .....              | 14        |
| Prerequisites .....                                  | 15        |
| Deployment Architecture .....                        | 15        |
| Deployment Scheme .....                              | 16        |
| Preparing Server.....                                | 17        |
| Preparing SQL Server Databases .....                 | 17        |
| Allocating IP Addresses .....                        | 21        |
| Configuring Firewall .....                           | 21        |
| Adding DNS Records .....                             | 21        |
| Preparing CSP Account .....                          | 21        |
| Preparing Odin Automation Public API User .....      | 31        |
| Obtaining Application Package.....                   | 31        |
| Preparing Office 365 Welcome Message Templates.....  | 31        |

|   |     |
|---|-----|
| Deploying 'Office 365' Application .....  | 32  |
| Importing 'Office 365' Application.....   | 32  |
| Deploying Office 365 Application Endpoint.....  | 33  |
| Creating 'Office 365' Application Instance .....  | 41  |
| Setting CSP Admin User's Password to Never Expire .....                                     | 44  |
| Creating Domain for Office 365 Subdomains.....  | 45  |
| Configuring Billing .....   | 45  |
| Creating 'PBA' Resource Type.....   | 45  |
| Configuring OA Billing Attributes.....  | 46  |
| Configuring Internationalization Settings .....   | 47  |
| Creating 'Office 365' Service Terms .....   | 61  |
| Configuring Offers .....  | 62  |
| Restrictions and Limitations.....   | 66  |
| Creating License Types .....  | 66  |
| Creating Add-on Types .....   | 67  |
| Creating Office 365 Resource Types .....  | 69  |
| Creating 'Office 365' Service Template .....  | 74  |
| Creating 'Office 365' Service Plans .....   | 77  |
| Creating 'Office 365' Resource and Sales Categories .....                                   | 80  |
| Setting Up Units of Measure for 'Office 365 License' and 'Office 365 Add-on' Resources .... | 80  |
| Configuring Online Store.....   | 81  |
| Selecting Online Store .....  | 81  |
| Creating 'Office 365' Screen .....  | 81  |
| Configuring 'Checkout' Screen.....  | 82  |
| Creating 'Office 365' Purchase Scenario .....   | 82  |
| Synchronizing Online Store.....   | 82  |
| Configuring Office 365 Subdomain Form for OA Billing.....                                   | 83  |
| Configuring Offer Compatibility Checker.....  | 83  |
| Configuring Offers for Resellers.....   | 92  |
| Firewall Requirements .....   | 97  |
| Office 365 Application Endpoint Host .....  | 97  |
| OA Operations Management Node .....   | 99  |
| OA Billing Application Server .....   | 100 |
| OA Operations Branding Server .....   | 100 |
| Importing Tenant Subscriptions from Microsoft Cloud into Odin Automation .....              | 101 |

|  |            |
|--|------------|
| Preconditions .....  | 101        |
| Limitations .....  | 102        |
| Collecting Tenant Data .....   | 103        |
| Importing Tenant Subscriptions .....   | 106        |
| Validation of Imported Tenant Subscriptions.....                                     | 108        |
| Example .....  | 108        |
| Useful Information .....   | 110        |
| How to Upgrade Office 365 Subscriptions through Provider/Reseller Control Panel..... | 110        |
| Viewing Reconciliation Report .....  | 111        |
| Viewing Business Reports .....   | 113        |
| Synchronizing Changes from Office 365 Portal.....                                    | 114        |
| How to Update Expired SSL Certificates on Office 365 Application Endpoint Host.....  | 115        |
| Creating Office 365 Subscription On Behalf Of Customer.....                          | 116        |
| Registering New Customers with Existing Microsoft Accounts .....                     | 116        |
| Examples of Office 365 Service Plans.....  | 118        |
| <b>Conversion from Syndication Partner Program to CSP Program.....</b>               | <b>126</b> |
| Subscription Split Mode .....  | 126        |
| Overview (Subscription Split Mode) .....   | 126        |
| Limitations (Subscription Split Mode) .....  | 129        |
| Prerequisites (Subscription Split Mode) .....  | 129        |
| Preparing for Conversion (Subscription Split Mode).....                              | 130        |
| Conversion Procedure (Subscription Split Mode).....                                  | 133        |
| Subscription Preservation Mode .....   | 148        |
| Overview (Subscription Preservation Mode).....                                       | 148        |
| Limitations (Subscription Preservation Mode) .....                                   | 150        |
| Prerequisites (Subscription Preservation Mode).....                                  | 150        |
| Preparing for Conversion (Subscription Preservation Mode) .....                      | 151        |
| Conversion Procedure (Subscription Preservation Mode) .....                          | 154        |
| Advanced Subscription Split Mode .....   | 169        |
| Overview (Advanced Subscription Split Mode).....                                     | 169        |
| Limitations (Advanced Subscription Split Mode) .....                                 | 172        |
| Prerequisites (Advanced Subscription Split Mode).....                                | 172        |
| Preparing for Conversion (Advanced Subscription Split Mode) .....                    | 173        |
| Conversion Procedure (Advanced Subscription Split Mode) .....                        | 176        |
| <b>Index .....</b>   | <b>192</b> |

# Preface

## In This Chapter

|                                |   |
|--------------------------------|---|
| Documentation Conventions..... | 6 |
| Feedback.....                  | 7 |

## Documentation Conventions

Before you start using this guide, it is important to understand the documentation conventions used in it.

### Typographical Conventions

The following kinds of formatting in the text identify special information.

| Formatting convention | Type of Information   | Example  |
|-----------------------|---|--|
| <b>Special Bold</b>   | Items you must select, such as menu options, command buttons, or items in a list.   | Navigate to the <b>QoS</b> tab.  |
|                       | Titles of modules, sections, and subsections.   | Read the <b>Basic Administration</b> module.   |
| <i>Italics</i>        | Used to emphasize the importance of a point, to introduce a term or to designate a command line placeholder, which is to be replaced with a real name or value. | These are the so-called <i>shared VEs</i> .<br>To destroy a VE, type <code>vzctl destroy VEid</code> . |

**Important**

An important note provides information that is essential to the completion of a task. Users can disregard information in a note and still complete a task, but they should not disregard an important note.

**Important:** The device drivers installed automatically during Setup are required by your system. If you remove one of these drivers, your system may not work properly.

|                   |   |  |
|-------------------|---|--|
| <b>Note</b>       | A note with the heading “Note” indicates neutral or positive information that emphasizes or supplements important points of the main text. A note supplies information that may apply only in special cases—for example, memory limitations, equipment configurations, or details that apply to specific versions of a program. | <b>Note:</b> If Windows prompts you for a network password at startup, your network is already set up and you can skip this section. |
| Monospace         | The names of commands, files, and directories.  | Use <code>vzctl start</code> to start a VE.  |
| Preformatted      | On-screen computer output in your command-line sessions; source code in XML, C++, or other programming languages.   | Saved parameters for VE<br>101   |
| Preformatted Bold | What you type, contrasted with on-screen computer output.   | # <b>rpm -V virtuo-<br/>release</b>  |
| CAPITALS          | Names of keys on the keyboard.  | SHIFT, CTRL, ALT   |
| KEY+KEY           | Key combinations for which the user must press and hold down one key and then press another.  | CTRL+P, ALT+F4   |

## General Conventions

Be aware of the following conventions used in this book.

- Modules in this guide are divided into sections, which, in turn, are subdivided into subsections. For example, **Documentation Conventions** is a section, and **General Conventions** is a subsection.
- When following steps or using examples, be sure to type double-quotes (“), left single-quotes (‘), and right single-quotes (’) exactly as shown.
- The key referred to as RETURN is labeled ENTER on some keyboards.

Commands in the directories included into the PATH variable are used without absolute path names. Steps that use commands in other, less common, directories show the absolute paths in the examples.

## Feedback

If you spot a typo in this guide, or if you have an opinion about how to make this guide more helpful, you can share your comments and suggestions with us by completing the Documentation Feedback form at <http://www.odin.com/support/usersdoc/>

## CHAPTER 2

# Introduction

This guide provides detailed instructions on how to offer Microsoft Online Services (Office 365) to end-customers through Odin Automation. This includes deployment scenarios, hardware and software requirements, server roles and others. It also contains instructions on how to configure the Microsoft Online Services (Office 365) in Odin Automation.

### In This Chapter

|  |   |
|--|---|
| Audience.....  | 8 |
| Technologies.....                                    | 8 |
| Operations Available to Office 365 Subscribers ..... | 8 |
| Supported Languages.....                             | 9 |
| Abbreviations .....                                  | 9 |
| Helpful Links .....                                  | 9 |

## Audience

The primary audience of this guide are providers or system integrators who want to deploy Microsoft Online Services and people involved in administering resources of providers.

## Technologies

Before reading this guide, make sure you are familiar with the following technologies and solutions:

- Microsoft Online Services (administration)
- SQL Server (deployment and administration)
- Odin Automation (deployment and administration)
- APS 2 (deployment and administration)

## Operations Available to Office 365 Subscribers

After the deployment is finished, the following list of operations will become available to an Office 365 subscriber through the Odin Automation control panel:



- Adding, viewing, modifying, and removing Office 365 users.
- Assigning/revoking Office 365 licenses to/from Office 365 users.
- Changing limits of Office 365 licenses.
- Upgrading Office 365 licenses.
- Viewing usage of Office 365 licenses.
- Adding/removing Office 365 services to/from domains.
- Synchronizing changes from the Microsoft cloud.
- Logging in to Microsoft Online Services Portal as an administrator.

## Supported Languages

The languages supported by the **Office 365** application are listed in the KB article <https://kb.odin.com/en/130052>.

## Abbreviations

- *APSC* – APS Controller;
- *MOSI* – Microsoft Online Syndication Interface;
- *CSP* – Cloud Solution Provider;
- *Partner Center API* – API for CSP partners;
- *Graph API* – Azure Active Directory Graph API;

## Helpful Links

Below is the list of links you may find helpful during deployment and configuration:

- **Microsoft Cloud**
  - Partner Center – <https://partnercenter.microsoft.com/>
  - Office 365 Admin Center / Office 365 Portal – <https://portal.office.com/>
  - Azure Management Portal – <https://manage.windowsazure.com/>
- **Microsoft Cloud Germany**
  - Partner Center – <https://partnercenter.microsoft.com/>
  - Office 365 Admin Center / Office 365 Portal – <https://portal.office.de>
  - Azure Management Portal – <https://portal.microsoftazure.de/>
- **Office 365 operated by 21Vianet in China**

- Partner Center – <https://partner.partnercenter.microsoftonline.cn>
- Office 365 Admin Center / Office 365 Portal – <https://portal.partner.microsoftonline.cn>
- Azure Management Portal – <https://portal.azure.cn/>

## CHAPTER 3

# Cloud Solution Provider Scenario

This section is intended for providers who are going to use the Microsoft Cloud Solution Provider program (<https://mspartner.microsoft.com/en/us/pages/solutions/cloud-reseller-overview.aspx>) for selling Office 365 offers.

### In This Chapter

|  |     |
|--|-----|
| Integration Overview .....   | 11  |
| Prerequisites .....  | 15  |
| Deployment Architecture .....  | 15  |
| Deploying 'Office 365' Application .....                                       | 32  |
| Configuring Billing.....   | 45  |
| Configuring Offers .....   | 62  |
| Configuring Online Store.....  | 81  |
| Configuring Office 365 Subdomain Form for OA Billing.....                      | 83  |
| Configuring Offer Compatibility Checker.....                                   | 83  |
| Configuring Offers for Resellers.....  | 92  |
| Firewall Requirements .....  | 97  |
| Importing Tenant Subscriptions from Microsoft Cloud into Odin Automation ..... | 101 |
| Useful Information .....   | 110 |

## Integration Overview

Odin Automation can be integrated with Office 365. By taking advantage of the integrated solution from Ingram Micro and Microsoft, service providers can deliver cloud services – such as Office 365 – to global customers who access them over the Internet.

To offer cloud services based on Office 365, a service provider sets up a new CSP account with Microsoft, configures Odin Automation and creates new offerings. Customers can then purchase the services accessible for users with a web-based interface.

## General Architecture

The following scheme represents the general architecture of the Odin Automation and Microsoft Online Services integration.

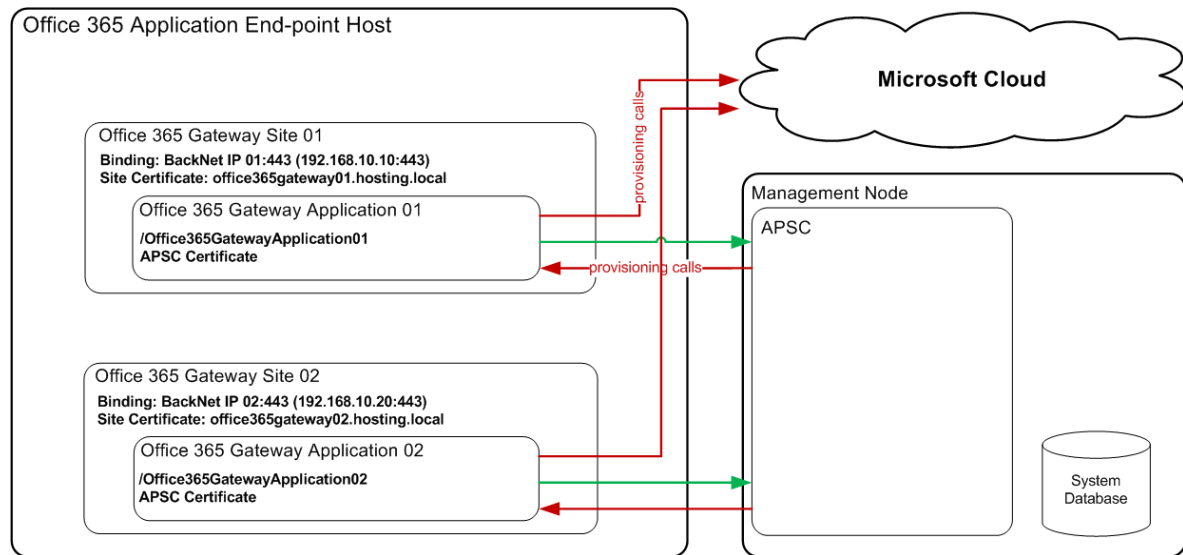


Figure 1: Integration with Office 365: General Architecture

Ingram Micro provides the **Office 365** application packaged according to the APS 2 standard. This package is deployed on your Odin Automation installation.

The Office 365 Application Endpoint Host provides communication between Odin Automation and the Microsoft cloud. Thus, operations performed by customers and end-users in Odin Automation control panels are propagated to their accounts in the Microsoft cloud.

The Office 365 Application Endpoint Host contains one or more Office 365 gateway sites depending on the required amount of Office 365 application instances. For example, it is possible to have one Office 365 application instance for the provider account and several Office 365 application instances for reseller accounts. An Office 365 gateway site contains an Office 365 gateway application. An Office 365 gateway application receives provisioning requests from the APSC. Then, the Office 365 gateway application calls methods of the Partner Center API and Graph API to view, create, update, or remove objects in the Microsoft cloud.

**Important:** One Odin Automation installation can be used for provisioning Office 365 services of different Office 365 clouds, such as Microsoft Cloud, Microsoft Cloud Germany, Office 365 operated by 21Vianet in China, but Office 365 application instances that belong to an Office 365 cloud must use their own Office 365 gateway sites with their own Office 365 gateway applications. In other words, Office 365 application instances of one Office 365 cloud and Office 365 application instances of another Office 365 cloud must not share the same Office 365 gateway sites and Office 365 gateway applications. For each Office 365 cloud and its Office 365 application instances, you must create separate Office 365 gateway sites with separate Office 365 gateway applications.

## Basic Partnership Scenario

The following scheme represents a basic partnership scenario.

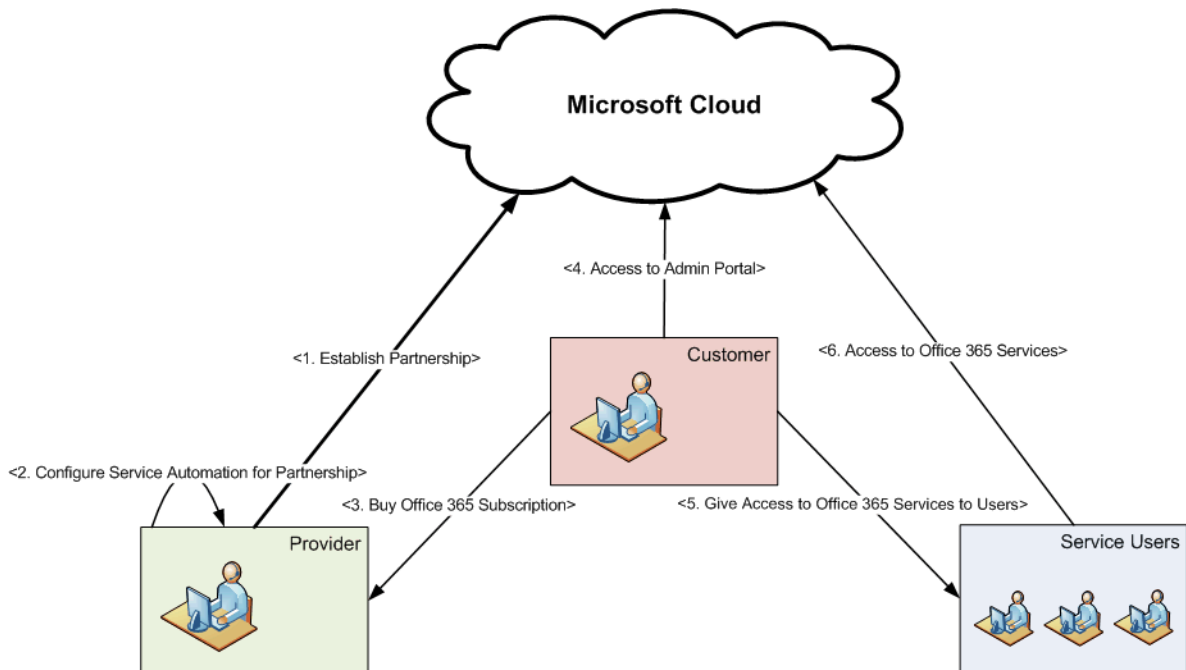


Figure 2: Integration with Office 365: Basic Partnership Scenario

A provider sets up a CSP account with Microsoft and configures Odin Automation to support partnership.

**Important:** Subscriptions that were canceled at Odin Automation side can be kept for some time in "Disabled" status at Microsoft side. Odin Automation stops charging a canceled subscription, but Microsoft may still keep charging for a subscription since it is "Disabled". To avoid undesired charges for canceled subscriptions, agree upon the time canceled subscriptions are kept at Microsoft side; for example zero time for such cases can be stated in your agreement with Microsoft.

A customer subscribes to Office 365 services and gets access to Administrative Portal. The customer gives access to Office 365 services to end-users.

## Basic Object Model

The following diagram shows a basic object model.

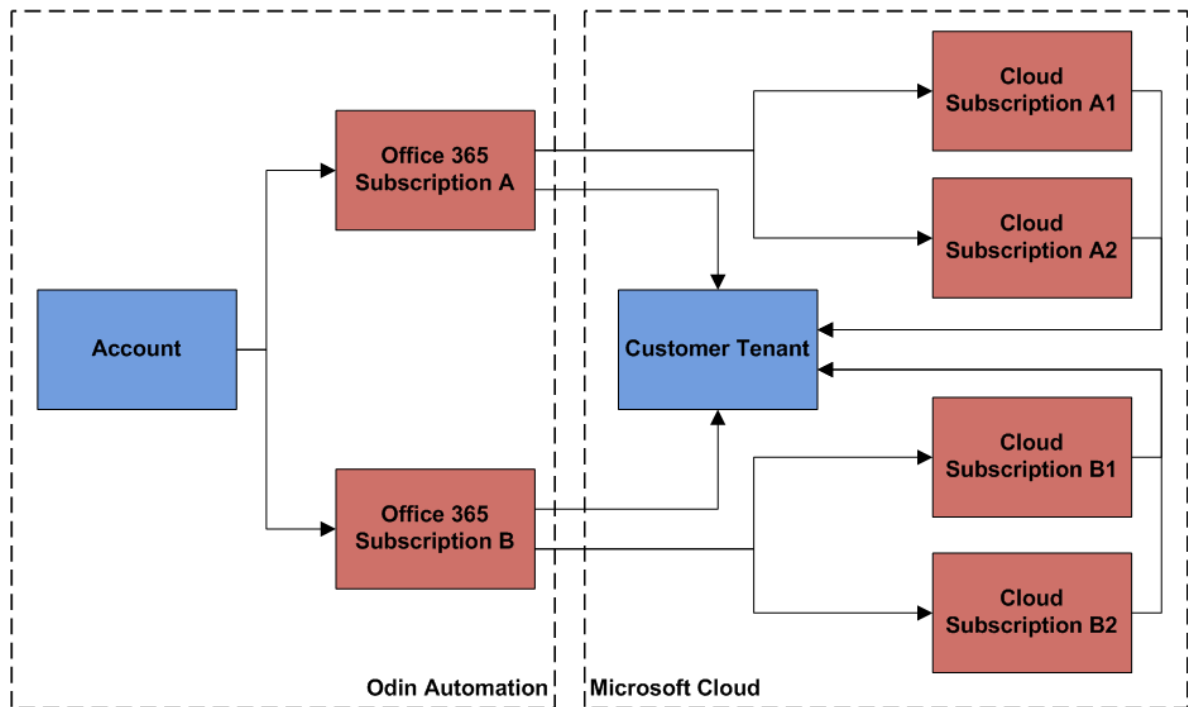


Figure 3: Basic Object Model

A customer account can have several Office 365 subscriptions in Odin Automation. The Office 365 subscriptions belong to one customer tenant in the Microsoft cloud. Each Office 365 subscription belongs to one or more cloud subscriptions depending on the offers included into the Office 365 subscription.

## Support for Account-Wide Services

The **Office 365** application can be provisioned in subscriptions where **Account-Wide Services** is set to **Yes**. This allows a service user of a customer account to have the **Office 365** service and other services from subscriptions where **Account-Wide Services** is also set to **Yes**.

## Prerequisites

Before deploying the **Office 365** application, make sure that the following prerequisites are satisfied:

- Odin Automation is installed:
  - 7.3.0 or a later 7.3.x version
  - 7.2.0 or a later 7.2.x version
  - 7.1.0 or a later 7.1.x version
  - 7.0.1\_hf130748 or a later 7.0.x version
- The **APS** module is installed in Odin Automation. Go to **Infrastructure** and click **Modules** in the **Packages** group. If the **APS** module is not installed, select the checkbox opposite the **APS** module and click on the **Install/Update** button.

Check the version of Python on the OA Operations Management Node. If it is less than **2.6.6**, you must prepare a separate host for running the Office 365 offer configuration script. The following requirements must be satisfied:

- Python 2.6.6 is installed on the host.
- The poaupdater RPM package is installed on the host. The RPM package must be taken from the Odin Automation distribution package/YUM repository that belongs to your version of Odin Automation (for example, see the `DISTRIBUTION_PACKAGE/modules/platform/OS/OS_VERSION/RPMS/` directory).
- The pyPgSQL-2.5.1 (or higher) Python package is installed on the host.
- The mx 3.1.1 (or higher) Python package is installed on the host.
- Connections are allowed from the host to the OA Operations Public API.
- Connections are allowed from the host to the OA Billing Public API.
- Connections are allowed from the host to the APSC.
- Connections are allowed from the host to the Odin Automation system database.

## Deployment Architecture

The goal of this section is to provide the conceptual architecture for the **Office 365** application.

This conceptual architecture should be used as a starting point for the envisioning process. Keep in mind that expected usage profiles, service level agreements, and cost control requirements are unique to each service provider and they can and will require changes to this architecture.

## Deployment Scheme

This scheme shows the deployment architecture of the **Office 365** application and provides the information about connections between integration components.

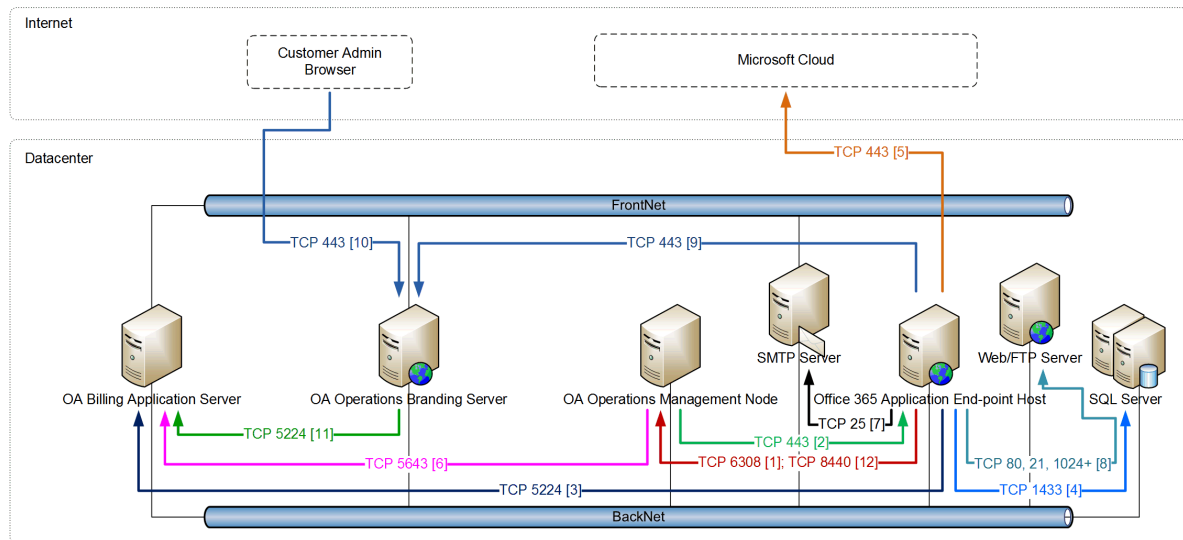


Figure 4: Office 365 Deployment Scheme

The following connections are shown:

- **[1]** – It is used for access to APSC from the application endpoint.
- **[2]** – It is used for access to the application endpoint from APSC.
- **[3]** – OA Billing Public API.
- **[4]** – It is used for access to Office 365 gateway application databases. This connection is used if an external SQL Server instance is deployed. See [Preparing SQL Server Databases](#) (page 17) for details.
- **[5]** – It is used for making provisioning calls from the application endpoint to the Microsoft cloud.
- **[6]** – It is used for access to the OA Billing application endpoint. See [Configuring Billing](#) (page 45) for details.
- **[7]** – It is used for sending welcome messages to newly created Office 365 users. See [Preparing Office 365 Welcome Message Templates](#) (page 31) for details.
- **[8]** – It is used for downloading welcome message templates. See [Preparing Office 365 Welcome Message Templates](#) (page 31) for details.
- **[9]** and **[10]** – It is used for validating offer compatibility. See [Configuring Offer Compatibility Checker](#) (page 83) for details.
- **[11]** – OA Billing Public API.



- [12] – OA Operations Public API.

## Preparing Server

For the deployment architecture, you will need to have 1 physical server. The following server role will be deployed on this server: **O365AEH** – Office 365 Application Endpoint Host.

Refer to the [Hardware Requirements for Office 365 Application Endpoint Host section](#) (page 33) for the detailed characteristics of the server.

**Note:** You can use an existing Office 365 Gateway Host of the **Office 365 APS 1.2** application to deploy the application endpoint of the **Office 365 APS 2** application. In this case you will need the following: increasing the hardware resources of the existing Office 365 Gateway Host; installing a BackNet NIC; installing .NET Framework 4.5.

## Preparing SQL Server Databases

Each Office 365 gateway application requires a SQL Server database to store configuration information. Before deploying the **Office 365** application, you must install SQL Server and prepare the required number of databases. SQL Server can be installed on the Office 365 Application Endpoint Host or on separate hosts if database high availability is required.

If database high availability is not required, the following SQL Server versions and editions can be used:

- SQL Server 2014 Express LocalDB (x64)

**Important:** If a SQL Server Express LocalDB database is in use by a running Office 365 gateway application, it is impossible to connect to the database with SQL Server Management Studio or sqlcmd.exe. To connect to the database with SQL Server Management Studio or sqlcmd.exe, you must perform the following actions: stop the corresponding Office 365 gateway site, stop the corresponding Office 365 gateway application, wait until the database is released.

- SQL Server 2012 Express (x64)
- SQL Server 2012 Web (x64)
- SQL Server 2012 Standard (x64)
- SQL Server 2012 Enterprise (x64)
- SQL Server 2014 Express (x64)
- SQL Server 2014 Web (x64)
- SQL Server 2014 Standard (x64)
- SQL Server 2014 Enterprise (x64)

If database high availability is required, the following SQL Server versions and editions must be used:

- SQL Server 2012 Standard (x64)
- SQL Server 2012 Enterprise (x64)
- SQL Server 2014 Standard (x64)
- SQL Server 2014 Enterprise (x64)

See the [SQL Server 2014 Express LocalDB \(x64\) section](#) (page 18) if you are going to use SQL Server 2014 Express LocalDB (x64). See the [SQL Server 2012/2014 Express/Web/Standard/Enterprise \(x64\) section](#) (page 18) if you are going to use SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64).

## SQL Server 2014 Express LocalDB (x64)

If you are going to use SQL Server 2014 Express LocalDB (x64) as a storage solution for Office 365 application endpoint data, you only need to install SQL Server 2014 Express LocalDB (x64) on the Office 365 Application Endpoint Host. You can use the instruction provided below to install it.

To install SQL Server 2014 Express LocalDB (x64), perform the following actions:

- 1 Log on to the Office 365 Application Endpoint Host as **Administrator**.
- 2 Download SQL Server 2014 Express LocalDB (x64) (<http://download.microsoft.com/download/E/A/E/EAE6F7FC-767A-4038-A954-49B8B05D04EB/LocalDB%2064BIT/SqlLocalDB.msi>).
- 3 Run the downloaded file.
- 4 Follow the wizard.

**Note:** You must not create any databases for Office 365 gateway applications. These databases will be created automatically during the installation of the Office 365 application endpoint.

## SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64)

If you are going to use SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64) as a storage solution for Office 365 application endpoint data, you must install SQL Server and prepare databases.

The following database requirements must be met for each database:

- A high-availability solution is configured for the database if it is required.
- The database storage device has enough disk space for the planned amount of Office 365 subscriptions (**5 MB** of disk space is required to store data of **1000** Office 365 subscriptions).
- The **Server authentication** property of the SQL Server instance of the database is set to **SQL Server and Windows Authentication mode**. You can use the instruction provided below to set this property.

- The **TCP/IP** protocol is enabled for the SQL Server instance of the database. You can use the instruction provided below to enable the protocol.
- The **Collation** property of the database is set to **Latin1\_General\_CI\_AS**. You can set this property when the database is created. Also, you can use the instruction provided below to set this property after creating the database.
- The following login is created for the database:
  - The **SQL Server authentication** option is selected.
  - The **Enforce password policy** property is turned off.
  - The database is specified in the **Default database** property.
  - The **sysadmin** server role is assigned.
  - There is a user mapping where a user is mapped to the database. The **Default Schema** property of the mapping is set to **dbo** and the **Database role membership** property of the mapping contains the **db\_owner** value.

You can use the instruction provided below to create this login.

- The following database information is prepared:
  - The name of the database. Examples: **Office365GatewayApplication01**, **Office365GatewayApplication02**.
  - The name of the login. Examples: **Office365GatewayApplication01**, **Office365GatewayApplication02**.
  - The password of the login.
  - The hostname or IP address of the SQL Server instance. For example: **o365sql.hosting.local**.
  - The name of the SQL Server instance if a non-default SQL Server instance is used. For example: **Office365Instance**.

**Notes:**

1. To obtain the information about high-availability solutions of SQL Server, refer to the <https://msdn.microsoft.com/en-us/library/ms190202.aspx> page.

2. To obtain the SQL Server installation and configuration instructions, refer to the <https://msdn.microsoft.com/en-us/library/bb500469.aspx> page.

To set the **Server authentication** property of the SQL Server instance of a database, perform the following actions:

- 1** Connect to the required SQL Server instance using **Microsoft SQL Server Management Studio**.
- 2** In the navigation area, right-click the SQL Server instance and select **Properties**.
- 3** Select the **Security** page.

**4** In the **Server authentication** group, select **SQL Server and Windows Authentication mode** and click **OK**.

**5** Restart the SQL Server instance.

To enable the **TCP/IP** protocol for the SQL Server instance of a database, perform the following actions:

**1** Open **SQL Server Configuration Manager**.

**2** In the navigation area, go to **SQL Server Network Configuration > <the required SQL Server instance>**.

**3** Right-click the **TCP/IP** protocol and select **Enable**.

To set the **Collation** property of a database, perform the following actions:

**1** Connect to the required SQL Server instance using **Microsoft SQL Server Management Studio**.

**2** In the navigation area, right-click the required database and select **Properties**.

**3** Select the **Options** page.

**4** Set the **Collation** property to **Latin1\_General\_CI\_AS** and click **OK**.

To create a login for a database, perform the following actions:

**1** Connect to the required SQL Server instance using **Microsoft SQL Server Management Studio**.

**2** In the navigation area, go to **Security > Logins** and right-click the **Logins** item.

**3** Select **New Login**.

**4** On the **General** page, specify the **Login name** property. Examples: **Office365GatewayApplication01**, **Office365GatewayApplication02**.

**5** On the **General** page, select the **SQL Server authentication** option and specify the **Password** and **Confirm password** properties.

**6** On the **General** page, make the **Enforce password policy** checkbox clear.

**7** On the **General** page, specify the database in the **Default database** property.

**8** Select the **Server Roles** page.

**9** On the **Server Roles** page, select the **sysadmin** server role.

**10** Select the **User Mapping** page.

**11** On the **User Mapping** page, create a user mapping by selecting the **Map** checkbox and setting the **Default Schema** property to **dbo** for the database.

**12** On the **User Mapping** page, select the user mapping and grant the **db\_owner** database role.

**13** Click **OK**.

## Allocating IP Addresses

The table below shows the number of IP addresses that must be configured on the BackNet NIC of the server.

| Server Role | IP Address   |
|-------------|--|
|             | BackNet  |
| O365AEH     | 192.168.10.10 (the first Office 365 gateway site)  |
|             | 192.168.10.20 (the second Office 365 gateway site) |

**Note:** Each Office 365 gateway site requires a separate BackNet IP address.

## Configuring Firewall

In the target data center, you will need to allow connections between the integration components. Refer to the [Firewall Requirements section](#) (page 97) for the detailed information.

**Note:** If you are going to use the existing Office 365 Gateway Host of the **Office 365 APS 1.2** application to deploy the application endpoint of the **Office 365 APS 2** application, make sure all the required firewall rules are configured for the Office 365 Gateway Host.

## Adding DNS Records

The following table provides the list of DNS records that must be created for the integration:

| Location     | Type | FQDN                             | IP address/FQDN | Maps to/Comments   |
|--------------|------|----------------------------------|-----------------|--|
| Internal DNS | A    | office365gateway01.hosting.local | 192.168.10.10   | The BackNet NIC of the Office 365 Application Endpoint Host. |
| Internal DNS | A    | office365gateway02.hosting.local | 192.168.10.20   | The BackNet NIC of the Office 365 Application Endpoint Host. |

**Note:** **hosting.local** is used here as an example of a provider's internal domain.

## Preparing CSP Account

You will need to prepare your CSP account as described in the subsequent sections. This includes the following:

- 1 Creating and configuring a separate administrative user.
- 2 Registering and configuring a separate native app for the Partner Center API.

- 3 Registering and configuring a separate native app for the Graph API.
- 4 Preparing information necessary for the deployment of the Office 365 app.

The user and native apps will be used to perform management operations in the Microsoft cloud.

**Note:** Countries where you can sell Office 365 services are limited by your agreement with Microsoft. In case your new customer specifies a country which breaks this limitation, the order will be placed in Odin Automation but the provisioning of the customer tenant will fail with the error 'Customer Country is currently not Supported'. See the <https://msdn.microsoft.com/en-us/library/partnercenter/dn974944.aspx> page to learn more about this limitation.

## Creating Administrative User

Log in to the Partner Center and create a separate admin user in your CSP account. When creating the admin user, set **Assists your customers as** to **Admin agent**.

Note that the login of the admin user must be based on the domain:

- **onmicrosoft.com** (for Microsoft Cloud)
- **onmicrosoft.de** (for Microsoft Cloud Germany)
- **partner.onmschina.cn** (for Office 365 operated by 21Vianet in China)

The **Office 365** application will use the domain part of the login to identify with which Microsoft cloud to interact.

We recommend that you use the following user name and login for the admin user:

- **Odin Automation** and [oa\\_account@<partnersubdomain>.onmicrosoft.com](mailto:oa_account@<partnersubdomain>.onmicrosoft.com) (for Microsoft Cloud)
- **Odin Automation** and [oa\\_account@<partnersubdomain>.onmicrosoft.de](mailto:oa_account@<partnersubdomain>.onmicrosoft.de) (for Microsoft Cloud Germany)
- **Odin Automation** and [oa\\_account@<partnersubdomain>.partner.onmschina.cn](mailto:oa_account@<partnersubdomain>.partner.onmschina.cn) (for Office 365 operated by 21Vianet in China)

To specify the **Assists your customers as** setting, you can use the following instruction:

- 1 Log in to the Partner Center.
- 2 Click the **View users** quick link.
- 3 Click on the required admin user.
- 4 Set **Assists your customers as** to **Admin agent**.
- 5 Click **Update**.

After creating the admin user, write down the admin user's login and password as they will be used in subsequent sections of the guide.

## Registering Native Apps for Partner Center and Graph APIs

This section describes how to register native apps for the Partner Center and Graph APIs.

- For Microsoft Cloud, use the procedure described in the [Microsoft Cloud section](#) (page 23).
- For Office 365 Operated by 21Vianet in China, use the procedure described in the [Office 365 Operated by 21Vianet in China section](#) (page 25).
- For Microsoft Cloud Germany, use the procedure described in the [Microsoft Cloud Germany section](#) (page 28).

### Microsoft Cloud

This section describes how to register native apps to enable access to the Partner Center API and the Graph API for the **Office 365** application. Note that you must register and configure two separate native apps: the first for the Partner Center API, the second for the Graph API.

#### Prerequisites

The following are the prerequisites:

- The steps in further sections require the user to have the **Global Admin** permissions in your CSP account's Azure Active Directory.
- A Windows client or server machine with **Azure Active Directory PowerShell Module Version 2** is installed.

**Note:** For the installation instructions, refer to <https://www.powershellgallery.com/packages/AzureAD/2.0.0.98>. Also note that the `Install-Module -Name AzureAD` command works on Windows 10. If your machine runs on Windows 8/8.1, you should install **PackageManagement PowerShell Modules Preview** (<https://www.microsoft.com/en-us/download/details.aspx?id=51451>) beforehand.

#### Registering Native Apps

Register native apps to enable access to the Partner Center API and the Graph API for the **Office 365** application:

- 1 Log in to the Partner Center.
- 2 Go to **Account settings > App management**.
- 3 Add a native app for the Partner Center API, and then write down its App ID.
- 4 Add a native app for the Graph API, and then write down its App ID.

#### Configuring API Access Permissions (Native App for Graph API)

Configure API access permissions for the native app that you created for the Graph API:

- 1 Log in to the Azure Management Portal.

- 2 In the left-hand navigation pane, choose **More services** and click **App registrations**.
- 3 On the **App registrations** blade, from the list, click on the native app.
- 4 On the blade of the native app, click **Settings**.
- 5 On the **Settings** blade, in **API Access**, click **Required permissions**.
- 6 On the **Required permissions** blade, click **Windows Azure Active Directory**.
- 7 On the **Enable Access** blade, in **Delegated permissions**, select these permissions:
  - **Read and write all groups**
  - **Access the directory as the signed-in user**Then, click **Save**.
- 8 On the **Required permissions** blade, click on **Add**.
- 9 On the **Add API access** blade, click **Select an API**.
- 10 On the **Select an API** blade, in the search box, type **Partner Center API** and select it from the results list. Then, click **Select**.
- 11 On the **Add API access** blade, click **Select permissions**.
- 12 On the **Enable Access** blade, select **Access Partner Center API**. Then, click **Select**.
- 13 On the **Add API access** blade, click **Done**.

#### **Granting Admin Consent (Native App for Graph API)**

Grant admin consent for the native app that you created for the Graph API:

- 1 Prepare the following URL:  
[https://login.windows.net/PARTNER\\_ID/oauth2/authorize?client\\_id=APPLICATION\\_ID&response\\_type=code&redirect\\_uri=http://localhost/&prompt=admin\\_consent&nonce=1234](https://login.windows.net/PARTNER_ID/oauth2/authorize?client_id=APPLICATION_ID&response_type=code&redirect_uri=http://localhost/&prompt=admin_consent&nonce=1234)

where

- **PARTNER\_ID** – Your partner subdomain (for example, **partner.onmicrosoft.com**).
  - **APPLICATION\_ID** – The App ID of the native app that you created.
- 2 Open the URL in your browser.
  - 3 On the page that you opened, click **Accept**.

#### **Configuring Pre-consent (Native App for Graph API)**

Configure pre-consent for the native app that you created for the Graph API:

- 1 Prepare the Object ID of the native app:
  - a Log in to the Azure Management Portal.
  - b In the left-hand navigation pane, choose **More services** and click **App registrations**.
  - c On the **App registrations** blade, click on the native app intended for the Graph API. Note that you can find the native app by using its App ID.



- d On the blade of the native app, copy and write down the Object ID of the native app.
- 2 On a machine having **Azure Active Directory PowerShell Module Version 2** installed, open a Windows PowerShell command prompt session.

- 3 Run the following command to connect to the Azure Active Directory:

```
Connect-AzureAD -AzureEnvironmentName "AzureCloud"
```

You will then be prompted for your credentials. Click **Work or School Account** and log in using the credentials of the user with the **Global Admin** permissions in your CSP account's Azure Active Directory.

- 4 Modify the PowerShell script given below by specifying the Object ID of the native app intended for the Graph API.

```
#Specify the Application Object Id of the Native App
$appObjectId = 'APPLICATION-OBJECT-ID-HERE'

Set-AzureADApplication -ObjectId $appObjectId -
AvailableToOtherTenants $true

$azureADApp = Get-AzureADApplication -ObjectId $appObjectId

#Get the Service Principal for the Application object
$servicePrincipal = Get-AzureADServicePrincipal -All $true | ?
{$_ .AppId -eq $azureADApp.AppId}

#Get the Admin Agents Group object
$adminAgentsGroup = Get-AzureADGroup -All $true | ?
{$_ .DisplayName -eq 'AdminAgents'}

#Run the below command to add preconsent for the application
Add-AzureADGroupMember -ObjectId $adminAgentsGroup.ObjectId -
RefObjectId $servicePrincipal.ObjectId
```

- 5 Execute the modified script to configure pre-consent for the native app.

### Getting Account ID of CSP Partner Account

Obtain and write down the Account ID of your CSP account:

- 1 Log in to the Partner Center.
- 2 Go to **Account settings > Organization profile**.
- 3 Use the value of **Microsoft ID**.

### Office 365 Operated by 21Vianet in China

This section describes how to register native apps to enable access to the Partner Center API and the Graph API for the **Office 365** application. Note that you must register and configure two separate native apps: the first for the Partner Center API, the second for the Graph API.

### Prerequisites

The following are the prerequisites:

- The steps in further sections require the user to have the **Global Admin** permissions in your CSP account's Azure Active Directory.
- A Windows client or server machine with the following components is installed:
  - **Microsoft Online Services Sign-in Assistant (64-bit version)**. The download link and installation instructions are provided here: <http://go.microsoft.com/fwlink/?LinkId=286152>. Note that you must select the **English** language on the download page and download the **msoidcli\_64.msi** file.
  - **Windows Azure Active Directory Module for Windows PowerShell (version 1.1.166.0)**. To install the module, do the following: make sure that the default language of your browser is set to **English**; open <http://connect.microsoft.com/site1164/Downloads/DownloadDetails.aspx?DownloadID=59185> in your browser; save and run the **AdministrationConfig-V1.1.166.0-GA.msi** file.

### Registering Native Apps

Register native apps to enable access to the Partner Center API and the Graph API for the **Office 365** application:

- 1 Log in to the Partner Center.
- 2 Go to **Account settings > App management**.
- 3 Add a native app for the Partner Center API, and then write down its App ID.
- 4 Add a native app for the Graph API, and then write down its App ID.

### Configuring API Access Permissions (Native App for Graph API)

Configure API access permissions for the native app that you created for the Graph API:

- 1 Log in to the Azure Management Portal.
- 2 In the left-hand navigation pane, choose **More services** and click **App registrations**.
- 3 On the **App registrations** blade, from the list, click on the native app.
- 4 On the blade of the native app, click **Settings**.
- 5 On the **Settings** blade, in **API Access**, click **Required permissions**.
- 6 On the **Required permissions** blade, click **Windows Azure Active Directory**.
- 7 On the **Enable Access** blade, in **Delegated permissions**, select these permissions:
  - **Read and write all groups**
  - **Access the directory as the signed-in user**

Then, click **Save**.

- 8 On the **Required permissions** blade, click on **Add**.
- 9 On the **Add API access** blade, click **Select an API**.
- 10 On the **Select an API** blade, in the search box, type **Partner Center operated by 21Vianet** and select it from the results list. Then, click **Select**.

- 11 On the **Add API access** blade, click **Select permissions**.
- 12 On the **Enable Access** blade, select **Access Partner Center Gallatin**. Then, click **Select**.
- 13 On the **Add API access** blade, click **Done**.

### Granting Admin Consent (Native App for Graph API)

Grant admin consent for the native app that you created for the Graph API:

- 1 Prepare the following URL:

[https://login.chinacloudapi.cn/PARTNER\\_ID/oauth2/authorize?client\\_id=APPLICATION\\_ID&response\\_type=code&redirect\\_uri=http://localhost/&prompt=admin\\_consent&nonce=1234](https://login.chinacloudapi.cn/PARTNER_ID/oauth2/authorize?client_id=APPLICATION_ID&response_type=code&redirect_uri=http://localhost/&prompt=admin_consent&nonce=1234)

where

- **PARTNER\_ID** – Your partner subdomain (for example, **partner.onmschina.cn**).
- **APPLICATION\_ID** – The App ID of the native app that you created.

- 2 Open the URL in your browser.
- 3 On the page that you opened, click **Accept**.

### Configuring Pre-consent (Native App for Graph API)

Configure pre-consent for the native app that you created for the Graph API:

- 1 On a machine having **Microsoft Online Services Sign-in Assistant (64-bit version)** and **Windows Azure Active Directory Module for Windows PowerShell (version 1.1.166.0)** installed, open a Windows PowerShell command prompt session.

- 2 Run the following command:

```
Connect-MsolService
```

You will then be prompted for your credentials. Log in using the credentials of the user with the **Global Admin** permissions in your CSP account's Azure Active Directory.

- 3 Run the following command:

```
Get-MsolGroup | Where-Object {$_.DisplayName -like "AdminAgents" }
```

From the command output, write down the Object ID of the **AdminAgents** group.

- 4 Run the following command:

```
Get-MsolServicePrincipal | Where-Object {$_.AppPrincipalId -like "APP_ID_OF_NATIVE_APP_FOR_GRAPH_API" }
```

From the command output, write down the Object ID of the native app.

- 5 Run the following command:

```
Add-MsolGroupMember -GroupObjectId  
OBJECT_ID_OF_ADMIN_AGENT_GROUP -GroupMemberType ServicePrincipal  
-GroupMemberObjectId OBJECT_ID_OF_NATIVE_APP_FOR_GRAPH_API
```

### Getting Account ID of CSP Partner Account

Obtain and write down the Account ID of your CSP account:

- 1 Log in to the Partner Center.
- 2 Go to **Account settings > Organization profile**.
- 3 Use the value of **Microsoft ID**.

### Microsoft Cloud Germany

This section describes how to register native apps to enable access to the Partner Center API and the Graph API for the **Office 365** application. Note that you must register and configure two separate native apps: the first for the Partner Center API, the second for the Graph API. The registration and configuration instructions are almost the same for both native apps.

#### Prerequisites

The following are the prerequisites:

- The steps in further sections require the user to have the **Global Admin** permissions in your CSP account's Azure Active Directory.
- A Windows client or server machine with **Azure Active Directory PowerShell Module Version 2** is installed.

**Note:** For the installation instructions, refer to <https://www.powershellgallery.com/packages/AzureAD/2.0.0.98>. Also note that the `Install-Module -Name AzureAD` command works on Windows 10. If your machine runs on Windows 8/8.1, you should install **PackageManagement PowerShell Modules Preview** (<https://www.microsoft.com/en-us/download/details.aspx?id=51451>) beforehand.

#### Registering Native Apps

Register two native application to enable access to the Partner Center API and Graph API. To register them, perform the following actions twice:

- 1 Log in to the Azure Management Portal.
- 2 In the left-hand navigation pane, choose **More services** and click **App registrations**.
- 3 On the **App registrations** blade, click **New application registration**.
- 4 On the **Create** blade, enter a name for the native app. For example: **Partner Center API Native App** (for the Partner Center API) or **Graph API Native App** (for the Graph API). Select **Native** in **Application type**. Specify <http://localhost/> in **Redirect URI**.

When you are done, click **Create**.

- 5 On the blade of the native app, copy and write down the following:
  - The Object ID of the native app.
  - The Application ID of of the native app.

These IDs will be used in subsequent sections of the guide.

### Configuring API Access Permissions (Both Native Apps)

Configure API access permissions for the native apps that you created in the previous section. Perform the following actions for each native app:

- 1 Log in to the Azure Management Portal.
- 2 In the left-hand navigation pane, choose **More services** and click **App registrations**.
- 3 On the **App registrations** blade, from the list, click on the native app.
- 4 On the blade of the native app, click **Settings**.
- 5 On the **Settings** blade, in **API Access**, click **Required permissions**.
- 6 On the **Required permissions** blade, click **Windows Azure Active Directory**.
- 7 On the **Enable Access** blade, in **Delegated permissions**, select these permissions:
  - **Read and write all groups**
  - **Access the directory as the signed-in user**Then, click **Save**.
- 8 On the **Required permissions** blade, click on **Add**.
- 9 On the **Add API access** blade, click **Select an API**.
- 10 On the **Select an API** blade, in the search box, type **Microsoft Partner Center** and select it from the results list. Then, click **Select**.
- 11 On the **Add API access** blade, click **Select permissions**.
- 12 On the **Enable Access** blade, select **Access Partner Center PPE**. Then, click **Select**.
- 13 On the **Add API access** blade, click **Done**.

### Granting Admin Consent (Both Native Apps)

Grant admin consent for the native apps created earlier. To do this, perform the following actions for each native app:

- 1 Prepare the following URL:  
[https://login.microsoftonline.de/PARTNER\\_ID/oauth2/authorize?client\\_id=APPLICATION\\_ID&response\\_type=code&redirect\\_uri=http://localhost/&prompt=admin\\_consent&nonce=1234](https://login.microsoftonline.de/PARTNER_ID/oauth2/authorize?client_id=APPLICATION_ID&response_type=code&redirect_uri=http://localhost/&prompt=admin_consent&nonce=1234)  
where
  - **PARTNER\_ID** – Your partner subdomain (for example, **partner.onmicrosoft.de**).
  - **APPLICATION\_ID** – The App ID of the native app that you created.
- 2 Open the URL in your browser.
- 3 On the page that you opened, click **Accept**.

### Configuring Pre-consent (Native App for Graph API)

**Note:** The steps of the section must be performed only for the native app intended for the Graph API.

Configure pre-consent:

- 1 On a machine having **Azure Active Directory PowerShell Module Version 2** installed, open a Windows PowerShell command prompt session.

- 2 Run the following command to connect to the Azure Active Directory:

```
Connect-AzureAD -AzureEnvironmentName "AzureGermanyCloud"
```

You will then be prompted for your credentials. Click **Work or School Account** and log in using the credentials of the user with the **Global Admin** permissions in your CSP account's Azure Active Directory.

- 3 Modify the PowerShell script given below by specifying the Object ID of the native app intended for the Graph API.

```
#Specify the Application Object Id of the Native App
$appObjectId = 'APPLICATION-OBJECT-ID-HERE'

Set-AzureADApplication -ObjectId $appObjectId -
AvailableToOtherTenants $true

$azureADApp = Get-AzureADApplication -ObjectId $appObjectId

#Get the Service Principal for the Application object
$servicePrincipal = Get-AzureADServicePrincipal -All $true | ?
{$_ .AppId -eq $azureADApp.AppId}

#Get the Admin Agents Group object
$adminAgentsGroup = Get-AzureADGroup -All $true | ?
{$_ .DisplayName -eq 'AdminAgents'}

#Run the below command to add preconsent for the application
Add-AzureADGroupMember -ObjectId $adminAgentsGroup.ObjectId -
RefObjectId $servicePrincipal.ObjectId
```

- 4 Execute the modified script to configure pre-consent for the native app.

#### Getting Account ID of CSP Partner Account

Obtain and write down the Account ID of your CSP account:

- 1 Log in to the Partner Center.
- 2 Go to **Account settings > Organization profile**.
- 3 Use the value of **Microsoft ID**.

## Preparing Odin Automation Public API User

You will need to create a separate staff member under your provider account for connecting to the OA Operations Public API and OA Billing Public API. This staff member must have the **Account Administrator** role in OA Operations and the **Full Access** role in OA Billing.

After creating the staff member, write down the listed below parameters. They will be used during the deployment.

- The login of the staff member;
- The password of the staff member;

Check the configuration of the OA Operations Public API (**System > Settings > Public API** in the **Integration** group). If **Accept connections** is set to **Only from allowed networks**, add the BackNet IP addresses of the Office 365 Application Endpoint Host to the **Allowed Networks** list.

## Obtaining Application Package

You will need to prepare the following application package: **Office 365** (the latest version). The **Office 365** application is packaged according to the APS 2 format. The application package contains the application and application endpoint files. The application package can be downloaded here: <http://download.automation.odin.com/pa/>. Download the application package to your workstation, the Office 365 Application Endpoint Host, and the target OA Operations Management Node.

## Preparing Office 365 Welcome Message Templates

It is possible to automatically send welcome messages to newly created Office 365 users. A welcome message can be used to provide a newly created Office 365 user with the following information:

- The URL of the Microsoft Online Services Portal
- The login of the user for logging in to the Microsoft Online Services Portal
- The temporary password of the user for logging in to the Microsoft Online Services Portal (the user must change this password after logging in to the portal)

If this ability is enabled, the administrator of an Office 365 organization can specify an e-mail address when a new Office 365 user is created. Then, the corresponding welcome message is sent to the specified e-mail address.

To enable this ability, you will need to prepare:

- A SMTP server which will be used for sending welcome messages to the Internet
- A Web/FTP server where welcome message templates will be placed

- The welcome message templates for required languages

To prepare the welcome message template for a language, perform the following actions:

- 1 Obtain the welcome message template for the English language. It is placed in the **Office 365** application package in the `doc\email-notification-templates` directory.
- 2 Obtain the language code of the language. To do this, log in to Provider Control Panel and go to **System > Settings > Locales**.
- 3 Create a copy of the `en.html` file (the default welcome message template for the English language), translate it into the required language.

**Important:** The name of the new welcome message template must consist of the first two letters of the language code and the `.html` extension. For example, the language code is `ja-JP`. In this case the name of the welcome message template must be `ja.html`.

- 4 Upload the welcome message template to the Web/FTP server.

**Important:** The welcome message template for the English language must also be uploaded to the Web/FTP server. It will be used as a default welcome message template. For example, if the welcome message template cannot be found for a non-English language.

When the SMTP server, Web/FTP server, and welcome message templates are prepared, write down the listed below parameters. They will be used during the deployment.

- The IP address or hostname of the SMTP server.
- The login and password for SMTP authentication (if SMTP authentication is required).
- The URL for downloading the welcome message templates. Examples:  
<http://provider.com/o365welcomemessagetemplates/>,  
<ftp://provider.com/o365welcomemessagetemplates/>, <http://192.168.10.100>,  
<ftp://192.168.10.100>.

## Deploying 'Office 365' Application

This section describes how to deploy the **Office 365** application.

### Importing 'Office 365' Application

Read the KB article <https://kb.odin.com/en/127961> and apply the KB article on your Odin Automation installation if necessary.

Import the **Office 365** application into Odin Automation. To import the application, use the instructions provided in the **OA Operations Application Hosting Deployment Guide >> Common Operations > Importing Application**.



## Deploying Office 365 Application Endpoint

This section describes how to deploy the Office 365 application endpoint.

### Hardware Requirements for Office 365 Application Endpoint Host

The Office 365 Application Endpoint Host must meet the following requirements:

|                          |   |  |
|--------------------------|---|--|
| <b>Server Names</b>      | <b>O365AEH</b>  |  |
| <b>Description</b>       | Office 365 Application Endpoint Host  |  |
| <b>Density</b>           | 500,000 Office 365 organizations  |  |
| <b>Quantity</b>          | 1   |  |
| <b>OS</b>                | <ul style="list-style-type: none"> <li>Windows Server 2008 R2 Web Edition SP1 (x64)</li> <li>Windows Server 2012 R2 Standard Edition (x64)</li> </ul> <p><b>Important:</b> Only English versions are supported.</p> |  |
| <b>Software</b>          | To be installed by the Customer: <ul style="list-style-type: none"> <li>OS</li> <li>IIS</li> <li>.NET Framework 4.5</li> </ul>  |  |
| <b>CPU</b>               | 2 core (2.3 GHz or higher)  |  |
| <b>RAM</b>               | 4 GB  |  |
| <b>Disks</b>             | Array1:   |  |
|                          | <ul style="list-style-type: none"> <li>OS and software</li> </ul>   | - 2 x 80 GB SATA, RAID1 (software impl.) |
| <b>Disk Partitioning</b> | Array 1:  |  |
|                          | <ul style="list-style-type: none"> <li>C: 80 GB</li> </ul>  | - for OS                                 |
| <b>NICs</b>              | BackNet   |  |

**Important:** You must use either a separate host or an existing Office 365 Gateway Host of the Office 365 APS 1.2 application for the Office 365 application endpoint deployment.

### Preparing Office 365 Application Endpoint Host

This section describes how to prepare the Office 365 Application Endpoint Host.

Install the following components on the Office 365 Application Endpoint Host:

- 1 Microsoft Online Services Sign-in Assistant (64-bit version).** The download link and installation instructions are provided here: <http://go.microsoft.com/fwlink/?LinkId=286152>. Note that you must select the **English** language on the download page and download the **msoidcli\_64.msi** file.

- 2 Windows Azure Active Directory Module for Windows PowerShell (version 1.1.166.0).** To install the module, do the following: make sure that the default language of your browser is set to **English**; open <http://connect.microsoft.com/site1164/Downloads/DownloadDetails.aspx?DownloadID=59185> in your browser; save and run the **AdministrationConfig-V1.1.166.0-GA.msi** file.

**Note:** If the existing Office 365 Gateway Host of the **Office 365 APS 1.2** application is used to deploy the application endpoint of the **Office 365 APS 2** application, make sure these components are installed on the Office 365 Gateway Host.

## Installing Office 365 Application Endpoint

This section describes how to install the Office 365 application endpoint.

The following options are available:

- If you are going to install the Office 365 application endpoint on the new host, use the instructions of the [Installing Office 365 Application Endpoint on New Host](#) section (page 34).
- If you are going to install the Office 365 application endpoint on the existing Office 365 Gateway Host of the **Office 365 APS 1.2** application, use the instructions of the [Using Office 365 APS 1.2 Gateway Host for Office 365 APS 2 Deployment](#) section (page 37).

### Installing Office 365 Application Endpoint on New Host

This section describes how to install the Office 365 application endpoint on the new host.

**Note:** To obtain the information about the Office 365 application endpoint architecture, see the [General Architecture](#) section (page 12).

### Preparing to Installation

- The installation is automatically performed by the `setup.cmd` script included in the **Office 365** application package. To obtain the script, perform the following actions:
  - a** Upload the **Office 365** application package to the Office 365 Application Endpoint Host.
  - b** Unpack the application package.
  - c** Unblock the content of the `0365-Web.zip` file. To do this, right-click the file in **Windows Explorer**, click **Properties**, click **Unblock**, click **OK**.
  - d** Extract the content of the `0365-Web.zip` file.
  - e** Open **Windows PowerShell Console**.
  - f** Go to the directory where the content of the extracted `0365-Web.zip` file is placed.
- Obtain BackNet IP addresses of the Office 365 Application Endpoint Host. These IP addresses will be used for Office 365 gateway sites. Note that each Office 365 gateway site requires a separate BackNet IP address.

- Prepare hostnames for Office 365 gateway sites. These hostnames must be resolved into corresponding BackNet IP addresses of the Office 365 Application Endpoint Host. Check it on the OA Operations Management Node and Office 365 Application Endpoint Host.
- Obtain the APSC SSL certificate from the OA Operations Management Node. The APSC SSL certificate is placed here:  
`/usr/local/pem/APS/certificates/controller.pem`. Upload the APSC SSL certificate to the Office 365 Application Endpoint Host. Open the certificate file in a text editor and delete the private part, which starts with "-----BEGIN RSA PRIVATE KEY-----" and ends with "-----END RSA PRIVATE KEY-----"; leave only the public part, which starts with "-----BEGIN CERTIFICATE-----" and ends with "-----END CERTIFICATE-----".
- Make sure you have prepared the SQL Server databases for your Office 365 gateway applications. See the [Preparing SQL Server Databases section](#) (page 17) for details.

### Creating Office 365 Gateway Sites

Run the script using the following parameters:

- **GatewaySiteName** – The name of the Office 365 gateway site. Note that the name must not contain dots and it must not be changed after the installation of the Office 365 application endpoint.
- **GatewayAppName** – The name of the Office 365 gateway application. Note that this name must not contain dots.
- **GatewayIPAddress** – The BackNet IP address of the Office 365 Application Endpoint Host. This IP address will be used by the Office 365 gateway site. Note that each Office 365 gateway site requires a separate BackNet IP address.
- **MOSIFactory** – The **CSP** value must be used.
- **GatewaySiteCertSubject** – The hostname of the Office 365 gateway site. The corresponding self-signed SSL certificate will be automatically generated. This SSL certificate will be used by the Office 365 gateway site.
- **APSCControllerCertPath** – The full path to the APSC SSL certificate file of the corresponding OA Operations Management Node. This SSL certificate will be used for securing communication between APSC and the Office 365 gateway application.
- **Force** – If this parameter is specified, the script does not ask a user to confirm actions.

If you are going to use **SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64)** as a storage solution, specify the following additional parameters:

- **SqlInstance** – The SQL Server instance of the Office 365 gateway application database. The following formats are possible here: **<hostname or IP address of SQL Server instance>** – if a default SQL Server instance is used; **<hostname or IP address of SQL Server instance>\<name of SQL Server instance>** – if a non-default SQL Server instance is used;
- **SqlDatabase** – The name of the Office 365 gateway application database.

- **SqlUser** – The login name of the Office 365 gateway application database.
- **SqlPassword** – The login password of the Office 365 gateway application database.

If you are going to use **SQL Server 2014 Express LocalDB (x64)** as a storage solution, specify the following additional parameter:

- **LocalDb** – If this parameter is specified, the Office 365 gateway application database is automatically created in SQL Server 2014 Express LocalDB (x64).

If you need to send welcome messages to newly created Office 365 users, specify the following additional parameters:

- **SMTPHost** – The IP address or hostname of the SMTP server which will be used for sending welcome messages to newly created Office 365 users.
- **SMTPPort** – The port of the SMTP server.
- **SMTPFrom** – The e-mail address which will be specified in the **From:** field of welcome messages.
- **SMTPFromPassword** – The password which will be used for SMTP authentication (the value of the **SMTPFrom** parameter will be used as a login). This parameter must be specified if SMTP authentication is required.
- **SMTPDoNotUseSsl** – If this parameter is specified, SSL will not be used.

For example, execute the following command to create the first Office 365 gateway site and Office 365 gateway application (**SQL Server 2014 Express LocalDB (x64)** is used as a storage solution):

```
./setup.cmd -GatewaySiteName Office365GatewaySite01 -GatewayAppName Office365GatewayApplication01 -GatewayIPAddress 192.168.10.10 -MOSIFactory CSP -GatewaySiteCertSubject office365gateway01.hosting.local -APSControllerCertPath C:\Temp\controller.pem -LocalDb -SMTPHost smtpserver.provider.com -SMTPPort 25 -SMTPFrom office365@provider.com -SMTPFromPassword smtppassword -Force
```

For example, execute the following command to create the second Office 365 gateway site and Office 365 gateway application (**SQL Server 2014 Express LocalDB (x64)** is used as a storage solution):

```
./setup.cmd -GatewaySiteName Office365GatewaySite02 -GatewayAppName Office365GatewayApplication02 -GatewayIPAddress 192.168.10.20 -MOSIFactory CSP -GatewaySiteCertSubject office365gateway02.hosting.local -APSControllerCertPath C:\Temp\controller.pem -LocalDb -SMTPHost smtpserver.provider.com -SMTPPort 25 -SMTPFrom office365@provider.com -SMTPFromPassword smtppassword -Force
```

For example, execute the following command to create the first Office 365 gateway site and Office 365 gateway application (**SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64)** is used as a storage solution):

```
./setup.cmd -GatewaySiteName Office365GatewaySite01 -GatewayAppName
Office365GatewayApplication01 -GatewayIPAddress 192.168.10.10 -
MOSIFactory CSP -GatewaySiteCertSubject office365gateway01.hosting.local
-APSControllerCertPath C:\Temp\controller.pem -SqlInstance
o365sql.hosting.local\Office365Instance -SqlDatabase
Office365GatewayApplication01 -SqlUser Office365GatewayApplication01 -
SqlPassword password01 -SMTPHost smtpserver.provider.com -SMTPPort 25 -
SMTPFrom office365@provider.com -SMTPFromPassword smtppassword -Force
```

For example, execute the following command to create the second Office 365 gateway site and Office 365 gateway application (**SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64)** is used as a storage solution):

```
./setup.cmd -GatewaySiteName Office365GatewaySite02 -GatewayAppName
Office365GatewayApplication02 -GatewayIPAddress 192.168.10.20 -
MOSIFactory CSP -GatewaySiteCertSubject office365gateway02.hosting.local
-APSControllerCertPath C:\Temp\controller.pem -SqlInstance
o365sql.hosting.local\Office365Instance -SqlDatabase
Office365GatewayApplication02 -SqlUser Office365GatewayApplication02 -
SqlPassword password02 -SMTPHost smtpserver.provider.com -SMTPPort 25 -
SMTPFrom office365@provider.com -SMTPFromPassword smtppassword -Force
```

The script will create the Office 365 gateway sites and Office 365 gateway applications.

#### Notes:

1. Replace the input parameter values with the values that belong to your Odin Automation installation.
2. The **SqlInstance**, **SqlDatabase**, **SqlUser**, **SqlPassword** parameters and the **LocalDb** parameter are mutually exclusive. Either the **SqlInstance**, **SqlDatabase**, **SqlUser**, **SqlPassword** parameters or the **LocalDb** parameter must be specified depending on the installed SQL Server edition/version.

## Using Office 365 APS 1.2 Gateway Host for Office 365 APS 2 Deployment

This section provides the instruction on how to use the existing Office 365 Gateway Host of the **Office 365 APS 1.2** application to deploy the application endpoint of the **Office 365 APS 2** application.

**Note:** To obtain the information about the Office 365 application endpoint architecture, see the [General Architecture section](#) (page 12).

To deploy the application endpoint of the **Office 365 APS 2** application on the existing Office 365 Gateway Host of the **Office 365 APS 1.2** application, perform the following actions:

- 1 Make sure the Office 365 Gateway Host satisfies the requirements of the Office 365 Application End-point Host. See the [Hardware Requirements for Office 365 Application End-point Host section](#) (page 33) and the [Preparing Office 365 Application End-point Host section](#) (page 33) for details.
- 2 Configure a BackNet IP address on the Office 365 Gateway Host. This BackNet IP address will be used by the Office 365 gateway site (APS 2).
- 3 Prepare a hostname for the Office 365 gateway site (APS 2). This hostname must be resolved into the BackNet IP address of the Office 365 Gateway Host. Check it on the OA Operations Management Node and Office 365 Gateway Host.
- 4 Prepare the installation script `setup.cmd`:
  - a Upload the **Office 365 APS 2** application package to the Office 365 Gateway Host.
  - b Unpack the application package.
  - c Unblock the content of the `0365-Web.zip` file. To do this, right-click the file in **Windows Explorer**, click **Properties**, click **Unblock**, click **OK**.
  - d Extract the content of the `0365-Web.zip` file.
  - e Open **Windows PowerShell Console**.
  - f Go to the directory where the content of the extracted `0365-Web.zip` file is placed.
- 5 Obtain the APSC SSL certificate from the corresponding OA Operations Management Node. It is placed here: `/usr/local/pem/APS/certificates/controller.pem`. Save the APSC SSL certificate to the Office 365 Gateway Host. Open the certificate file in a text editor and delete the private part, which starts with `-----BEGIN RSA PRIVATE KEY-----` and ends with `-----END RSA PRIVATE KEY-----`; leave only the public part, which starts with `-----BEGIN CERTIFICATE-----` and ends with `-----END CERTIFICATE-----`.
- 6 Prepare the SQL Server database for the Office 365 gateway application (APS 2). See the [Preparing SQL Server Databases section](#) (page 17) for details.
- 7 Run the `setup.cmd` script using the following parameters:
  - **GatewaySiteName** – The name of the Office 365 gateway site (APS 2). Note that this name must not contain dots.
  - **GatewayAppName** – The name of the Office 365 gateway application (APS 2). Note that this name must not contain dots.
  - **GatewayIPAddress** – The BackNet IP address of the Office 365 Gateway Host. This IP address will be used by the Office 365 gateway site (APS 2). Note that each Office 365 gateway site (APS 2) requires a separate BackNet IP address.
  - **MOSIFactory** – The **CSP** value must be used.
  - **GatewaySiteCertSubject** – The hostname of the Office 365 gateway site (APS 2). The corresponding self-signed SSL certificate will be automatically generated. This SSL certificate will be used by the Office 365 gateway site (APS 2).

- **APSCertPath** – The full path to the APSC SSL certificate file of the corresponding OA Operations Management Node. This SSL certificate will be used for securing communication between APSC and the Office 365 gateway application (APS 2).
- **Force** – If this parameter is specified, the script does not ask a user to confirm actions.

If you are going to use **SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64)** as a storage solution, specify the following additional parameters:

- **SqlInstance** – The SQL Server instance of the Office 365 gateway application (APS 2) database. The following formats are possible here: **<hostname or IP address of SQL Server instance>** – if a default SQL Server instance is used; **<hostname or IP address of SQL Server instance>\<name of SQL Server instance>** – if a non-default SQL Server instance is used;
- **SqlDatabase** – The name of the Office 365 gateway application (APS 2) database.
- **SqlUser** – The login name of the Office 365 gateway application (APS 2) database.
- **SqlPassword** – The login password of the Office 365 gateway application (APS 2) database.

If you are going to use **SQL Server 2014 Express LocalDB (x64)** as a storage solution, specify the following additional parameter:

- **LocalDb** – If this parameter is specified, the Office 365 gateway application (APS 2) database is automatically created in SQL Server 2014 Express LocalDB (x64).

If you need to send welcome messages to newly created Office 365 users, specify the following additional parameters:

- **SMTPHost** – The IP address or hostname of the SMTP server which will be used for sending welcome messages to newly created Office 365 users.
- **SMTPPort** – The port of the SMTP server.
- **SMTPFrom** – The e-mail address which will be specified in the **From:** field of welcome messages.
- **SMTPFromPassword** – The password which will be used for SMTP authentication (the value of the **SMTPFrom** parameter will be used as a login). This parameter must be specified if SMTP authentication is required.
- **SMTPDoNotUseSsl** – If this parameter is specified, SSL will not be used.

Example (**SQL Server 2014 Express LocalDB (x64)** is used as a storage solution):

```
./setup.cmd -GatewaySiteName Office365GatewaySite -GatewayAppName
Office365GatewayApplication -GatewayIPAddress 192.168.10.10 -
MOSIFactory CSP -GatewaySiteCertSubject office365gateway.hosting.local
-APSCertPath C:\Temp\controller.pem -LocalDb -SMTPHost
smtpserver.provider.com -SMTPPort 25 -SMTPFrom office365@provider.com -
SMTPFromPassword smtppassword -Force
```

Example (SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64) is used as a storage solution):

```
./setup.cmd -GatewaySiteName Office365GatewaySite -GatewayAppName Office365GatewayApplication -GatewayIPAddress 192.168.10.10 -MOSIFactory CSP -GatewaySiteCertSubject office365gateway.hosting.local -APSControllerCertPath C:\Temp\controller.pem -SqlInstance o365sql.hosting.local\Office365Instance -SqlDatabase Office365GatewayApplication -SqlUser Office365GatewayApplication -SqlPassword password -SMTPHost smtpserver.provider.com -SMTPPort 25 -SMTPFrom office365@provider.com -SMTPFromPassword smtppassword -Force
```

The script will create the new Office 365 gateway site/application (APS 2). Existing Office 365 dispatcher sites/applications (APS 1.2) and Office 365 gateway sites/applications (APS 1.2) will continue working.

**Important:** After performing the procedure described above, do not upgrade the Office 365 gateway site/application (APS 2) using files from the **Office 365 APS 1.2** application package. You must use the corresponding files from the **Office 365 APS 2** application package to upgrade the Office 365 gateway site/application (APS 2).

**Notes:**

1. Replace the input parameter values with the values that belong to your Odin Automation installation.
2. The **SqlInstance**, **SqlDatabase**, **SqlUser**, **SqlPassword** parameters and the **LocalDb** parameter are mutually exclusive. Either the **SqlInstance**, **SqlDatabase**, **SqlUser**, **SqlPassword** parameters or the **LocalDb** parameter must be specified depending on the installed SQL Server edition/version.

## Securing Office 365 Application Endpoint Host

This section provides instructions on how to improve the security of the Office 365 Application Endpoint Host.

- 1 Log on to the Office 365 Application Endpoint Host as **Administrator**.
- 2 Make sure the **IP and Domain Restrictions** role service is installed for the **Web Server (IIS)** Role. See the <http://www.iis.net/ConfigReference/system.webServer/security/ipSecurity> article for details.
- 3 Open **Internet Information Services (IIS) Manager**.
- 4 For each Office 365 gateway application, perform the following actions:
  - a In the navigation tree, select the Office 365 gateway application.



- b** Double-click the **IP Address and Domain Restrictions** icon.
- c** Click on the **Edit Feature Settings** link.
- d** Select **Deny** in the **Access for unspecified clients** field.
- e** Click **OK**.
- f** Click on the **Add Allow Entry** link.
- g** Select the **Specific IP address** option.
- h** In the field, specify the BackNet IP address of the OA Operations Management Node that corresponds to the Office 365 gateway application.
- i** Click **OK**.
- j** Validate that the <https://<Office 365 Gateway Site>/<Office 365 Gateway Application>/aps/> URL is accessible only from the OA Operations Management Node.

## Creating 'Office 365' Application Instance

To be able to use the **Office 365** application, first you should create the new application instance.

- 1** In Provider Control Panel, go to **Services > Applications** and click the **Office 365** package that you have imported before.
- 2** Open the **Instances** tab and click **Install**.
- 3** Specify the following parameters:
  - **Application API end-point URI** – The application API endpoint URI. This URI is constructed in the following way: <https://<Office 365 Gateway Site Hostname>/<Office 365 Gateway Application Name>/aps/>. Example: <https://office365gateway01.hosting.local/Office365GatewayApplication01/aps/>.
  - **Application Version** – Select the required version of the application.

Click **Next**.

### Important:

1. The hostname specified in **Office\_365\_Gateway\_Site\_Name** must be resolved from the OA Operations Management Node into the BackNet IP address of the Office 365 Application Endpoint Host.
2. Each Office 365 gateway site requires a separate BackNet IP address.

- 4** Specify the following application global settings:
  - **Instance Name** – The name of the application instance.
  - **URL for Microsoft Online Services Portal** – The URL of the Office 365 admin center.

- **Homepage** – The URL of your website. This parameter is optional. If it is specified, this URL is displayed in Customer Control Panel. For example: <https://provider.com>.
- **Business Automation OpenAPI URL** – The OA Billing Public API URL. For example: [https://ODIN\\_AUTOMATION\\_BILLING\\_APPLICATION\\_SERVER\\_BACKNET\\_IP:5224/RPC2](https://ODIN_AUTOMATION_BILLING_APPLICATION_SERVER_BACKNET_IP:5224/RPC2).
- **User name for connecting to Business Automation API** – The name of the prepared user for connecting to the OA Billing Public API.
- **User password for connecting to Business Automation API** – The password of the prepared user for connecting to the OA Billing Public API.
- **Administrator's login for connecting to Microsoft Online** – The login of the admin user of your CSP account. This parameter is mandatory. For example: [oa\\_account@<partnersubdomain>.onmicrosoft.com](mailto:oa_account@<partnersubdomain>.onmicrosoft.com) (for Microsoft Cloud), [oa\\_account@<partnersubdomain>.onmicrosoft.de](mailto:oa_account@<partnersubdomain>.onmicrosoft.de) (for Microsoft Cloud Germany), or [oa\\_account@<partnersubdomain>.partner.onmschina.cn](mailto:oa_account@<partnersubdomain>.partner.onmschina.cn) (for Office 365 operated by 21Vianet in China).
- **Administrator's password for connecting to Microsoft Online** – The password of the admin user of your CSP account. This parameter is mandatory.
- **Reseller tenant ID** – Specify the prepared Account ID / Commerce ID.
- **ID of Native Client for Partner Center SDK** – Specify the prepared App ID of the native app of the Partner Center API.
- **ID of Native Client for Azure AD Graph API** – Specify the prepared App ID of the native app of the Graph API.
- **Application reply URL prefix** – Leave the default value of the setting.
- **Billing ID** – Leave the default value of the setting.
- **Customer Deactivation Behavior** – Choose the way how to handle Microsoft Online accounts when the corresponding Odin Automation subscriptions are cancelled in Odin Automation. The following options are available:
  - **Production Account** – When an Odin Automation subscription is cancelled in Odin Automation, the corresponding Microsoft Online account and subscriptions are disabled. The corresponding Microsoft Online users are removed. Use this option for production environments. This option is selected by default.
  - **TIP (Test In Production) Account** – When an Odin Automation subscription is cancelled in Odin Automation, the corresponding Microsoft Online account, users, and subscriptions are removed. Use this option for test environments only.
- **Confirm Deactivation Behavior** – Confirm the **TIP (Test In Production) Account** customer deactivation behavior by selecting the **I do understand the difference between TIP and Production accounts and my account is a TIP account** option.
- **Verification DNS Record Type** – Choose the DNS record type which will be used for the verification of domains delegated to the **Office 365** application. The following options are available: **CNAME** (deprecated), **TXT** (recommended). The **TXT** value is selected by default.

- **Service Plans Compatibility Checker URL** – The URL of the offer compatibility checker. Leave empty. This URL will be specified later. See the [Configuring Offer Compatibility Checker section](#) (page 83) for details.
- **Sale Category List** – The list of the identifiers of the Office 365 sales categories. Leave empty. This list will be specified later. See the [Configuring Offer Compatibility Checker section](#) (page 83) for details.
- **"Secondary E-mail" is a mandatory Service User's details field** – Leave the default value. This global setting is reserved for future use.
- **Welcome Message Templates Download URL** – If you need to send welcome messages to newly created Office 365 users, specify the prepared URL.
- **Lync Partner Name and Plan** – Leave the default value of the setting.
- **Format Phone Numbers** – Make sure it is set to **No** (the checkbox is clear).
- **Prefix for Naming Subscriptions** – A prefix for generating names of customer default subdomains. The following options are available:
  - If this parameter is not specified, names of customer default subdomains must be manually specified in the online store or Customer Control Panel.
  - If this parameter is specified and names of customer default subdomains are not manually specified, Odin Automation automatically generates these names using the value of the parameter and subscription identifiers. For example, if the value of the parameter and a subscription identifier are **companyabc** and **1000000**, the name of the customer default subdomain will be **companyabc1000000.onmicrosoft.com** (for Microsoft Cloud), **companyabc1000000.onmicrosoft.de** (for Microsoft Cloud Germany), or **companyabc1000000.partner.onmschina.cn** (for Office 365 operated by 21Vianet in China). Note that you must use only the following symbols for this parameter: **a-z, A-Z, 0-9**.
- **Drivers environment** – Leave the default value of the setting.
- **Provider can change application settings** – Leave the default value of the setting.
- **T2 Client ID** – Leave the default value of the setting.
- **T2 Client Secret Key** – Leave the default value of the setting.
- **T2 MPN ID** – Leave the default value of the setting.
- **T2 Reseller tenant ID** – Leave the default value of the setting.
- **MPN ID** – Leave the default value of the setting.
- **Operations Automation OpenAPI URL** – The OA Operations Public API URL. This URL can be obtained at **System > Settings > Public API** in the **Integration** group.
- **User name for connecting to Operations Automation API** – The login of the prepared user for connecting to the OA Operations Public API.
- **User password for connecting to Operations Automation API** – The password of the prepared user for connecting to the OA Operations Public API.

- **Invitation URL** – This parameter will be available after creating the application instance. The value of the parameter is based on the value of the **Reseller tenant ID** parameter, so you do not need to set it manually. This parameter defines the URL of the page which is used for registering customers who already have Microsoft accounts. See the [Registering New Customers with Existing Microsoft Accounts section](#) (page 116) for details.

Click **Next**.

- 5 Validate the parameters carefully and click **Finish**.

## Setting CSP Admin User's Password to Never Expire

Configure your CSP account:

- 1 Log on to the Office 365 Application Endpoint Host as **Administrator**.
- 2 In the **Start** menu, click **Microsoft Online Services Module for Windows PowerShell**.
- 3 Run the `Connect-MsolService` command.
- 4 In the **Windows PowerShell Credential Request** window that opens, type the login and password of the created admin user  
[oa\\_account@<partnersubdomain>.onmicrosoft.com](#) (for Microsoft Cloud),  
[oa\\_account@<partnersubdomain>.onmicrosoft.de](#) (for Microsoft Cloud Germany), or  
[oa\\_account@<partnersubdomain>.partner.onmschina.cn](#) (for Office 365 operated by 21Vianet in China). Click **OK**.
- 5 Run the following command to set the password of the admin user to never expire:

### Microsoft Cloud

```
Set-MsolUser -UserPrincipalName  
oa\_account@<partnersubdomain>.onmicrosoft.com -  
PasswordNeverExpires $true
```

### Microsoft Cloud Germany

```
Set-MsolUser -UserPrincipalName  
oa\_account@<partnersubdomain>.onmicrosoft.de -  
PasswordNeverExpires $true
```

### Office 365 operated by 21Vianet in China

```
Set-MsolUser -UserPrincipalName  
oa\_account@<partnersubdomain>.partner.onmschina.cn -  
PasswordNeverExpires $true
```

**Important:** If your security policies do not allow setting passwords to never expire, you must regularly set a new admin user password in the Microsoft cloud and the global settings of the application instance before the old admin user password expires.

## Creating Domain for Office 365 Subdomains

Create the domain **onmicrosoft.com** (for Microsoft Cloud), **onmicrosoft.de** (for Microsoft Cloud Germany), or **partner.onmschina.cn** (for Office 365 operated by 21Vianet in China) under your provider account in Odin Automation. To create the domain, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Domains**.
- 2 Click **Add Existing Domain Registered Elsewhere**.
- 3 Specify the following parameters:
  - **Domain** – Specify **onmicrosoft.com** (for Microsoft Cloud), **onmicrosoft.de** (for Microsoft Cloud Germany), or **partner.onmschina.cn** (for Office 365 operated by 21Vianet in China).
  - **Hosting type** – Select **External (Use external name servers. In this case you are managing DNS records yourself)**.

Click **Next**.

- 4 Validate the specified parameters and click **Finish**.
- 5 Click on the just created domain.
- 6 Click **Edit**.
- 7 Make sure **Registrar Status** is set to **Undefined**.
- 8 Set **Allow parent and child domains** to **Yes** (the checkbox is selected).
- 9 Click **Submit**.

## Configuring Billing

Configure OA Billing using the instructions of this section.

### Creating 'PBA' Resource Type

**Note:** The **PBA** resource type must be created if you need integration between the **Office 365** application and OA Billing. This integration allows customers to adjust license limits and upgrade licenses from the application screen located at **Office 365 > Licenses**. If you do not need this integration, you must skip this section.

Create the **PBA** resource type. This resource type is used to communicate with OA Billing.

To create the **PBA** resource type, follow these steps:

- 1 In Provider Control Panel, go to **Services > Applications** and click the **pba** application in the list of applications.

- 2 Select the **Resource Types** tab and click **Create**.
- 3 Select the **Application Service Reference** resource class in the list of resource classes.
- 4 Specify the following parameters:
  - **Name** – The name of the resource type. For example: **PBA**.
  - **Description** – The description of the resource type. This parameter is optional.
- 5 Click **Next**.
- 6 Select the **Parallels Business Automation PBA API** APS type in the list of APS types.
- 7 Select the required instance of the **pba** application in the list of instances.
- 8 Check the resource type parameters carefully and click **Finish**.

## Configuring OA Billing Attributes

**Note:** These OA Billing attributes must be created if you need integration between the **Office 365** application and OA Billing. This integration allows customers to adjust license limits and upgrade licenses from the application screen located at **Office 365 > Licenses**. If you do not need this integration, you must skip this section.

Create the following attributes in OA Billing:

- **aps\_limits\_notification**
- **aps\_upsell\_returnViewResourceId**
- **aps\_upsell\_returnView**

To create the attributes, follow these steps:

- 1 In Provider Control Panel, go to **System > Settings** and click **Attributes** in the **Operations** group.
- 2 Click **Add New Attribute**.
- 3 Specify the following parameters:
  - **ID** – The identifier of the attribute. Make sure the identifier and name of the attribute are the same. Examples: **aps\_limits\_notification**, **aps\_upsell\_returnViewResourceId**, **aps\_upsell\_returnView**.
  - **Name** – The name of the attribute. Make sure the identifier and name of the attribute are the same. Examples: **aps\_limits\_notification**, **aps\_upsell\_returnViewResourceId**, **aps\_upsell\_returnView**.
  - **Value Type** – The type of the attribute. Make sure the **String** type is selected.
  - **Status** – The status of the attribute. Make sure the **Optional** status is selected.
  - **Applicable to** – The attribute applicability. Make sure the **Order** value is selected.

- **Active** – The state of the attribute. Make sure this checkbox is selected (**Yes**).

Do not change the default values of other parameters.

- 4 Check attribute parameters carefully and click **Save**.

## Configuring Internationalization Settings

You must configure internationalization settings in OA Billing to comply with Office 365 requirements for states and ZIP codes of various countries:

- Review the **State and ZIP Code Requirements** table. This table describes the address validation that is performed by the Microsoft cloud. Note the following:
  - If a country has **Is State Required == TRUE**:
    - If a state list is provided, the Microsoft cloud will validate that the supplied state matches one in the list.
    - If a state list is not provided, the Microsoft cloud will accept any text for the state.
  - **Postal Code Regex**: A set of expressions used in the validation process.
- Use the state codes provided in the **State Codes for Canada**, **State Codes for China**, **State Codes for Mexico**, and **State Codes for the United States** tables for Canada, China, Mexico, and the United States.

To manage countries, in Provider Control Panel go to **System > Settings > Internationalization > Countries**. To manage states, in Provider Control Panel go to **System > Settings > Internationalization > States / Provinces**.

### State and ZIP Code Requirements

| Country              | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex |
|----------------------|--------------|-------------------|------------|-------------------------|-------------------|
| Andorra              | AD           | FALSE             | <none>     | FALSE                   | <none>            |
| United Arab Emirates | AE           | FALSE             | <none>     | FALSE                   | <none>            |
| Afghanistan          | AF           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$     |
| Antigua and Barbuda  | AG           | FALSE             | <none>     | FALSE                   | <none>            |
| Anguilla             | AI           | FALSE             | <none>     | FALSE                   | <none>            |
| Albania              | AL           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$     |
| Armenia              | AM           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$     |
| Netherlands Antilles | AN           | FALSE             | <none>     | FALSE                   | <none>            |
| Angola               | AO           | FALSE             | <none>     | FALSE                   | <none>            |
| Antarctica           | AQ           | FALSE             | <none>     | FALSE                   | <none>            |

| Country                          | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex                    |
|----------------------------------|--------------|-------------------|------------|-------------------------|--------------------------------------|
| Argentina                        | AR           | FALSE             | <none>     | TRUE                    | ^[a-zA-Z][0-9]{4,4}[a-zA-Z]{3,3}\$   |
| American Samoa                   | AS           | FALSE             | <none>     | FALSE                   | <none>                               |
| Austria                          | AT           | FALSE             | <none>     | TRUE                    | ^[1-9][0-9]{3,3}\$                   |
| Australia                        | AU           | TRUE              | <none>     | TRUE                    | ^[0-9]{4,4}\$                        |
| Aruba                            | AW           | FALSE             | <none>     | FALSE                   | <none>                               |
| Azerbaijan                       | AZ           | FALSE             | <none>     | TRUE                    | ^([aA][zZ][ ])?[0-9]{4,4}\$          |
| Bosnia and Herzegovina           | BA           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$                        |
| Barbados                         | BB           | FALSE             | <none>     | TRUE                    | ^[a-zA-Z]{2,2}[0-9]{5,5}\$           |
| Bangladesh                       | BD           | FALSE             | <none>     | FALSE                   | ^[0-9]{4,4}\$                        |
| Belgium                          | BE           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                        |
| Burkina Faso                     | BF           | FALSE             | <none>     | FALSE                   | <none>                               |
| Bulgaria                         | BG           | FALSE             | <none>     | TRUE                    | ^[1-9][0-9]{3,3}\$                   |
| Bahrain                          | BH           | FALSE             | <none>     | TRUE                    | ^[0-9]{3,4}\$                        |
| Burundi                          | BI           | FALSE             | <none>     | FALSE                   | <none>                               |
| Benin                            | BJ           | FALSE             | <none>     | FALSE                   | <none>                               |
| Saint Barthelemy                 | BM           | FALSE             | <none>     | TRUE                    | ^[a-zA-Z]{2,2}[ ]?[a-zA-Z0-9]{2,2}\$ |
| Brunei                           | BN           | FALSE             | <none>     | TRUE                    | ^[a-zA-Z]{2,2}[0-9]{4,4}\$           |
| Bolivia                          | BO           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                        |
| Sint Bonaire, Eustatius and Saba | BQ           | FALSE             | <none>     | FALSE                   | <none>                               |
| Brazil                           | BR           | TRUE              | <none>     | TRUE                    | ^[0-9]{5,5}[-][0-9]{3,3}\$           |
| Bahamas                          | BS           | TRUE              | <none>     | FALSE                   | <none>                               |
| Bhutan                           | BT           | FALSE             | <none>     | FALSE                   | <none>                               |
| Bouvet Island                    | BV           | FALSE             | <none>     | FALSE                   | <none>                               |
| Botswana                         | BW           | FALSE             | <none>     | FALSE                   | <none>                               |
| Belarus                          | BY           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$                        |
| Belize                           | BZ           | TRUE              | <none>     | FALSE                   | <none>                               |



| Country                          | Country Code | Is State Required | State List   | Is Postal Code Required | Postal Code Regex                              |
|----------------------------------|--------------|-------------------|--|-------------------------|--|
| Canada                           | CA           | TRUE              | AB BC MB NB NL NT NS NU ON PE QC SK YT   | TRUE                    | ^[a-zA-Z][0-9][a-zA-Z][ ]?[0-9][a-zA-Z][0-9]\$ |
| Cocos (Keeling) Islands          | CC           | FALSE             | <none>   | FALSE                   | <none>   |
| Democratic Republic of the Congo | CD           | FALSE             | <none>   | FALSE                   | <none>   |
| Central African Republic         | CF           | FALSE             | <none>   | FALSE                   | <none>   |
| Republic of the Congo            | CG           | FALSE             | <none>   | FALSE                   | <none>   |
| Switzerland                      | CH           | FALSE             | <none>   | TRUE                    | ^[1-9][0-9]{3,3}\$                             |
| Ivory Coast                      | CI           | FALSE             | <none>   | FALSE                   | <none>   |
| Cook Islands                     | CK           | FALSE             | <none>   | FALSE                   | <none>   |
| Chile                            | CL           | FALSE             | <none>   | FALSE                   | ^([1-9]\d{2})[ \-]?(\d{4})\$                   |
| Cameroon                         | CM           | FALSE             | <none>   | FALSE                   | <none>   |
| China                            | CN           | TRUE              | AH BJ CQ FJ GD GS GX GZ HA HB HE HI HL HN JL JS JX LN NM NX QH SC SD SH SN SX TJ XJ XZ YN ZJ | TRUE                    | ^[0-9]{6,6}\$                                  |
| Colombia                         | CO           | FALSE             | <none>   | FALSE                   | <none>   |
| Costa Rica                       | CR           | TRUE              | <none>   | FALSE                   | ^[0-9]{4,5}?\$                                 |
| Cape Verde                       | CV           | FALSE             | <none>   | TRUE                    | ^[0-9]{4,4}\$                                  |
| Curacao                          | CW           | FALSE             | <none>   | FALSE                   | <none>   |
| Christmas Island                 | CX           | FALSE             | <none>   | FALSE                   | <none>   |
| Cyprus                           | CY           | FALSE             | <none>   | TRUE                    | ^[1-9][0-9]{3,3}\$                             |
| Czech Republic                   | CZ           | FALSE             | <none>   | TRUE                    | ^[0-9]{3,3}[ ][0-9]{2,2}\$                     |
| Germany                          | DE           | FALSE             | <none>   | TRUE                    | ^[0-9]{5,5}\$                                  |

| Country            | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex   |
|--------------------|--------------|-------------------|------------|-------------------------|---|
| Djibouti           | DJ           | FALSE             | <none>     | FALSE                   | <none>  |
| Denmark            | DK           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$   |
| Dominica           | DM           | FALSE             | <none>     | FALSE                   | <none>  |
| Dominican Republic | DO           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Algeria            | DZ           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Ecuador            | EC           | FALSE             | <none>     | TRUE                    | ^([eE][cC])?[0-9]{6,6}\$  |
| Estonia            | EE           | FALSE             | <none>     | TRUE                    | ^[1-9][0-9]{4,4}\$  |
| Egypt              | EG           | FALSE             | <none>     | FALSE                   | ^[1-9][0-9]{4,4}\$  |
| Eritrea            | ER           | FALSE             | <none>     | FALSE                   | <none>  |
| Spain              | ES           | TRUE              | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Ethiopia           | ET           | FALSE             | <none>     | FALSE                   | <none>  |
| Finland            | FI           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Fiji               | FJ           | FALSE             | <none>     | FALSE                   | <none>  |
| Falkland Islands   | FK           | FALSE             | <none>     | FALSE                   | <none>  |
| Micronesia         | FM           | FALSE             | <none>     | FALSE                   | <none>  |
| Faroe Islands      | FO           | FALSE             | <none>     | TRUE                    | ^[0-9]{3,3}\$   |
| France             | FR           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Gabon              | GA           | FALSE             | <none>     | FALSE                   | <none>  |
| United Kingdom     | GB           | FALSE             | <none>     | TRUE                    | ^([gG][iI][rR][ ]?0[aA]{2,2} [a-zA-Z]{0-9}[0-9a-zA-Z]? [a-zA-Z][0-9][0-9a-zA-Z]?[ ]?[0-9][a-zA-Z]{2,2})\$ |
| Grenada            | GD           | FALSE             | <none>     | FALSE                   | <none>  |
| Georgia            | GE           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$   |
| Grench Guiana      | GF           | FALSE             | <none>     | FALSE                   | <none>  |
| Guernsey           | GG           | FALSE             | <none>     | FALSE                   | <none>  |
| Ghana              | GH           | FALSE             | <none>     | FALSE                   | <none>  |
| Gibraltar          | GI           | FALSE             | <none>     | FALSE                   | <none>  |
| Greenland          | GL           | FALSE             | <none>     | FALSE                   | <none>  |
| Gambia             | GM           | FALSE             | <none>     | FALSE                   | <none>  |

| Country                                      | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex              |
|--|--------------|-------------------|------------|-------------------------|--------------------------------|
| Guinea                                       | GN           | FALSE             | <none>     | FALSE                   | <none>                         |
| Guadeloupe                                   | GP           | FALSE             | <none>     | FALSE                   | <none>                         |
| Equatorial Guinea                            | GQ           | FALSE             | <none>     | FALSE                   | <none>                         |
| Greece                                       | GR           | FALSE             | <none>     | TRUE                    | (^[0-8][0-9]{4,4}\$)(^90000\$) |
| South Georgia and the South Sandwich Islands | GS           | FALSE             | <none>     | FALSE                   | <none>                         |
| Guatemala                                    | GT           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$                  |
| Guam   | GU           | FALSE             | <none>     | FALSE                   | <none>                         |
| Guinea-Bissau                                | GW           | FALSE             | <none>     | FALSE                   | <none>                         |
| Guyana                                       | GY           | FALSE             | <none>     | FALSE                   | <none>                         |
| Hong Kong                                    | HK           | FALSE             | <none>     | FALSE                   | <none>                         |
| Heard Island and McDonald Islands            | HM           | FALSE             | <none>     | FALSE                   | <none>                         |
| Honduras                                     | HN           | TRUE              | <none>     | TRUE                    | ^[a-zA-Z]{2,2}[0-9]{4,4}\$     |
| Croatia                                      | HR           | FALSE             | <none>     | TRUE                    | ^[1-9][0-9]{4,4}\$             |
| Haiti  | HT           | FALSE             | <none>     | FALSE                   | <none>                         |
| Hungary                                      | HU           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                  |
| Indonesia                                    | ID           | FALSE             | <none>     | TRUE                    | ^[1-9][0-9]{4,4}\$             |
| Ireland                                      | IE           | FALSE             | <none>     | FALSE                   | <none>                         |
| Israel                                       | IL           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$                  |
| Isle of Man                                  | IM           | FALSE             | <none>     | FALSE                   | <none>                         |
| India  | IN           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$                  |
| British Indian Ocean Territory               | IO           | FALSE             | <none>     | FALSE                   | <none>                         |
| Iraq   | IQ           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$                  |
| Iceland                                      | IS           | FALSE             | <none>     | TRUE                    | ^([1-9][0-9]{2,2})[0-9]{5,5}\$ |

| Country               | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex   |
|-----------------------|--------------|-------------------|------------|-------------------------|---|
| Italy                 | IT           | TRUE              | <none>     | TRUE                    | ^([0-358]9 [0-589]8 [0-689][67]) ([0-9][0-5])[0-9]{3,3}\$ |
| Jersey                | JE           | FALSE             | <none>     | FALSE                   | <none>  |
| Jamaica               | JM           | TRUE              | <none>     | FALSE                   | ^[0-9A-Za-z]{0,5}\$                                       |
| Jordan                | JO           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Japan                 | JP           | TRUE              | <none>     | TRUE                    | ^[0-9]{3,3}[-]?[0-9]{4,4}\$                               |
| Kenya                 | KE           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Kyrgyzstan            | KG           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$   |
| Cambodia              | KH           | FALSE             | <none>     | FALSE                   | <none>  |
| Kiribati              | KI           | FALSE             | <none>     | FALSE                   | <none>  |
| Comoros               | KM           | FALSE             | <none>     | FALSE                   | <none>  |
| Saint Kitts and Nevis | KN           | FALSE             | <none>     | FALSE                   | <none>  |
| South Korea           | KR           | FALSE             | <none>     | TRUE                    | ^[0-9]{3,3}[-]?[0-9]{3,3}\$                               |
| Kuwait                | KW           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Cayman Islands        | KY           | FALSE             | <none>     | TRUE                    | ^[a-zA-Z]{2,2}[0-9]-[0-9]{4,4}\$                          |
| Kazakhstan            | KZ           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$   |
| Laos                  | LA           | FALSE             | <none>     | FALSE                   | <none>  |
| Lebanon               | LB           | FALSE             | <none>     | TRUE                    | ^[0-9]{8,8}\$   |
| Saint Lucia           | LC           | FALSE             | <none>     | FALSE                   | <none>  |
| Liechtenstein         | LI           | FALSE             | <none>     | TRUE                    | ^[1-9][0-9]{3,3}\$  |
| Sri Lanka             | LK           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Liberia               | LR           | FALSE             | <none>     | FALSE                   | <none>  |
| Lesotho               | LS           | FALSE             | <none>     | FALSE                   | <none>  |
| Lithuania             | LT           | FALSE             | <none>     | TRUE                    | ^([L][lT]-)?[0-9]{5,5}\$                                  |
| Luxembourg            | LU           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$   |
| Latvia                | LV           | FALSE             | <none>     | TRUE                    | ^([L][lV]-)?[0-9]{4,4}\$                                  |
| Libya                 | LY           | FALSE             | <none>     | FALSE                   | <none>  |
| Morocco               | MA           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |

| Country                  | Country Code | Is State Required | State List  | Is Postal Code Required | Postal Code Regex             |
|--------------------------|--------------|-------------------|---|-------------------------|-------------------------------|
| Monaco                   | MC           | FALSE             | <none>  | TRUE                    | ^[0-9]{5,5}\$                 |
| Moldova                  | MD           | FALSE             | <none>  | TRUE                    | ^(MD[ \-])?[0-9]{4,4}\$       |
| Montenegro               | ME           | FALSE             | <none>  | TRUE                    | ^[0-9]{5,5}\$                 |
| Madagascar               | MG           | FALSE             | <none>  | FALSE                   | <none>                        |
| Marshall Islands         | MH           | FALSE             | <none>  | FALSE                   | <none>                        |
| Macedonia                | MK           | FALSE             | <none>  | TRUE                    | ^[0-9]{4,4}\$                 |
| Mali                     | ML           | FALSE             | <none>  | FALSE                   | <none>                        |
| Mongolia                 | MN           | FALSE             | <none>  | TRUE                    | ^[0-9]{5,5}\$                 |
| Macau                    | MO           | FALSE             | <none>  | FALSE                   | <none>                        |
| Northern Mariana Islands | MP           | FALSE             | <none>  | FALSE                   | <none>                        |
| Martinique               | MQ           | FALSE             | <none>  | FALSE                   | <none>                        |
| Mauritania               | MR           | FALSE             | <none>  | FALSE                   | <none>                        |
| Montserrat               | MS           | FALSE             | <none>  | FALSE                   | <none>                        |
| Malta                    | MT           | FALSE             | <none>  | TRUE                    | ^[a-zA-Z]{3,3}[ ][0-9]{4,4}\$ |
| Mauritius                | MU           | FALSE             | <none>  | FALSE                   | <none>                        |
| Maldives                 | MV           | FALSE             | <none>  | FALSE                   | <none>                        |
| Malawi                   | MW           | FALSE             | <none>  | FALSE                   | <none>                        |
|                          |              |                   | AG BC BS CM CS CH CO CL DF DG MX GT GR HG JA MI MO NA NL OA PU QT QR SL SI SO TB TM TL VE YU ZA |                         |                               |
| Mexico                   | MX           | TRUE              |   | TRUE                    | ^[0-9]{5,5}\$                 |
| Malaysia                 | MY           | TRUE              | <none>  | TRUE                    | ^[0-9]{5,5}\$                 |
| Mozambique               | MZ           | FALSE             | <none>  | FALSE                   | <none>                        |
| Namibia                  | NA           | FALSE             | <none>  | FALSE                   | <none>                        |
| New Caledonia            | NC           | FALSE             | <none>  | FALSE                   | <none>                        |
| Niger                    | NE           | FALSE             | <none>  | FALSE                   | <none>                        |

| Country                   | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex                             |
|---------------------------|--------------|-------------------|------------|-------------------------|---|
| Norfolk Island            | NF           | FALSE             | <none>     | FALSE                   | <none>  |
| Nigeria                   | NG           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$                                 |
| Nicaragua                 | NI           | TRUE              | <none>     | TRUE                    | ^([0-9]{3,3}-[0-9]{3,3}-[0-9]) ([0-9]{5,5})\$ |
| Netherlands               | NL           | FALSE             | <none>     | TRUE                    | ^[1-9][0-9]{3,3}[ ]?[a-zA-Z]{2,2}\$           |
| Norway                    | NO           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                                 |
| Nepal                     | NP           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$                                 |
| Nauru                     | NR           | FALSE             | <none>     | FALSE                   | <none>  |
| Niue                      | NU           | FALSE             | <none>     | FALSE                   | <none>  |
| New Zealand               | NZ           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                                 |
| Oman                      | OM           | FALSE             | <none>     | FALSE                   | ^[0-9]{3,3}\$                                 |
| Panama                    | PA           | TRUE              | <none>     | FALSE                   | ^[0-9]{4,4}\$                                 |
| Peru                      | PE           | FALSE             | <none>     | FALSE                   | <none>  |
| French Polynesia          | PF           | FALSE             | <none>     | FALSE                   | <none>  |
| Papua New Guinea          | PG           | FALSE             | <none>     | FALSE                   | <none>  |
| Philippines               | PH           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                                 |
| Pakistan                  | PK           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$                                 |
| Poland                    | PL           | FALSE             | <none>     | TRUE                    | ^[0-9]{2,2}[-][0-9]{3,3}\$                    |
| Saint Pierre and Miguelon | PM           | FALSE             | <none>     | FALSE                   | <none>  |
| Pitcaim Islands           | PN           | FALSE             | <none>     | FALSE                   | <none>  |
| Puerto Rico               | PR           | FALSE             | <none>     | TRUE                    | (^[0-9]{5,5}[-]?[0-9]{4,4}\$) (^[0-9]{5,5}\$) |
| State of Palestine        | PS           | FALSE             | <none>     | FALSE                   | <none>  |
| Portugal                  | PT           | FALSE             | <none>     | TRUE                    | ^\d{4}(-\d{3})?\$                             |
| Palau                     | PW           | FALSE             | <none>     | FALSE                   | <none>  |
| Paraguay                  | PY           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                                 |
| Qatar                     | QA           | FALSE             | <none>     | FALSE                   | <none>  |
| Reunion                   | RE           | FALSE             | <none>     | FALSE                   | <none>  |

| Country                     | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex   |
|-----------------------------|--------------|-------------------|------------|-------------------------|---|
| Romania                     | RO           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$   |
| Serbia                      | RS           | FALSE             | <none>     | FALSE                   | ^[0-9]{5,6}\$   |
| Russia                      | RU           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$   |
| Rwanda                      | RW           | FALSE             | <none>     | FALSE                   | <none>  |
| Saudi Arabia                | SA           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Solomon Islands             | SB           | FALSE             | <none>     | FALSE                   | <none>  |
| Seychelles                  | SC           | FALSE             | <none>     | FALSE                   | <none>  |
| Sweeden                     | SE           | FALSE             | <none>     | TRUE                    | ^(3[013-9])4[0-7]9[0-8][125-8][0-9][0-9][1,1][0-9]{2,2}\$ |
| Singapore                   | SG           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$   |
| Saint Helena                | SH           | FALSE             | <none>     | FALSE                   | <none>  |
| Slovenia                    | SI           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$   |
| Svalbard                    | SJ           | FALSE             | <none>     | FALSE                   | <none>  |
| Slovakia                    | SK           | FALSE             | <none>     | TRUE                    | ^[0-9]{3,3}[ ]0-9]{2,2}\$                                 |
| Sierra Leone                | SL           | FALSE             | <none>     | FALSE                   | <none>  |
| San Marino                  | SM           | FALSE             | <none>     | FALSE                   | <none>  |
| Senegal                     | SN           | FALSE             | <none>     | FALSE                   | ^[0-9]{5,5}\$   |
| Somalia                     | SO           | FALSE             | <none>     | FALSE                   | <none>  |
| Suriname                    | SR           | FALSE             | <none>     | FALSE                   | <none>  |
| South Sudan                 | SS           | FALSE             | <none>     | FALSE                   | <none>  |
| Sao Tome and Principe       | ST           | FALSE             | <none>     | FALSE                   | <none>  |
| El Salvador                 | SV           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$   |
| Sint Maarten (Dutch part)   | SX           | FALSE             | <none>     | FALSE                   | <none>  |
| Swaziland                   | SZ           | FALSE             | <none>     | FALSE                   | <none>  |
| Turks and Caicos Islands    | TC           | FALSE             | <none>     | FALSE                   | <none>  |
| Chad                        | TD           | FALSE             | <none>     | FALSE                   | <none>  |
| French Southern Territories | TF           | FALSE             | <none>     | FALSE                   | <none>  |
| Togo                        | TG           | FALSE             | <none>     | FALSE                   | <none>  |

| Country                              | Country Code | Is State Required | State List   | Is Postal Code Required | Postal Code Regex         |
|--------------------------------------|--------------|-------------------|--|-------------------------|---------------------------|
| Thailand                             | TH           | FALSE             | <none>   | TRUE                    | ^[0-9]{5,5}\$             |
| Tajikistan                           | TJ           | FALSE             | <none>   | FALSE                   | <none>                    |
| Tokelau                              | TK           | FALSE             | <none>   | FALSE                   | <none>                    |
| Timor-Leste                          | TL           | FALSE             | <none>   | FALSE                   | <none>                    |
| Turkmenistan                         | TM           | FALSE             | <none>   | FALSE                   | <none>                    |
| Tunisia                              | TN           | FALSE             | <none>   | TRUE                    | ^[0-9]{4,4}\$             |
| Tonga                                | TO           | FALSE             | <none>   | FALSE                   | <none>                    |
| Turkey                               | TR           | TRUE              | <none>   | TRUE                    | ^[0-9]{5,5}\$             |
| Trinidad and Tobago                  | TT           | FALSE             | <none>   | FALSE                   | ^[0-9]{6,6}\$             |
| Tuvalu                               | TV           | FALSE             | <none>   | FALSE                   | <none>                    |
| Taiwan                               | TW           | FALSE             | <none>   | TRUE                    | ^[0-9]{3,3}  [0-9]{5,5}\$ |
| Tanzania                             | TZ           | FALSE             | <none>   | FALSE                   | <none>                    |
| Ukraine                              | UA           | FALSE             | <none>   | FALSE                   | ^[0-9]{5,5}\$             |
| Uganda                               | UG           | FALSE             | <none>   | FALSE                   | <none>                    |
| United States Minor Outlying Islands | UM           | FALSE             | <none>   | FALSE                   | <none>                    |
| United States                        | US           | TRUE              | AL AK AZ AR CA CO CT DE FL GA HI ID IL IN IA KS KY LA ME MD MA MI MN MS MO MT NE NV NH NJ NM NY NC ND OH OK OR PA RI SC SD TN TX UT VT VA WA WV WI WY DC | TRUE                    | ^\d{5}(-\d{4})?\$         |
| Uruguay                              | UY           | FALSE             | <none>   | TRUE                    | ^[0-9]{5,5}\$             |
| Uzbekistan                           | UZ           | FALSE             | <none>   | TRUE                    | ^[0-9]{6,6}\$             |
| Holy See (Vatican City)              | VA           | FALSE             | <none>   | FALSE                   | <none>                    |



| Country                          | Country Code | Is State Required | State List | Is Postal Code Required | Postal Code Regex                              |
|----------------------------------|--------------|-------------------|------------|-------------------------|--|
| Saint Vincent and the Grenadines | VC           | FALSE             | <none>     | FALSE                   | <none>   |
| Venezuela                        | VE           | TRUE              | <none>     | TRUE                    | ^[0-9]{4,4}\$                                  |
| British Virgin Islands           | VG           | FALSE             | <none>     | FALSE                   | <none>   |
| US Virgin Islands                | VI           | FALSE             | <none>     | TRUE                    | ^[a-zA-Z]{2,2}[0-9]{5,5}([0-9]{5}-[0-9]{4})?\$ |
| Vietnam                          | VN           | FALSE             | <none>     | TRUE                    | ^[0-9]{6,6}\$                                  |
| Vanuatu                          | VU           | FALSE             | <none>     | FALSE                   | <none>   |
| Wallis and Futuna                | WF           | FALSE             | <none>     | FALSE                   | <none>   |
| Samoa                            | WS           | FALSE             | <none>     | FALSE                   | <none>   |
| Yemen                            | YE           | FALSE             | <none>     | FALSE                   | <none>   |
| Mayotte                          | YT           | FALSE             | <none>     | FALSE                   | <none>   |
| South Africa                     | ZA           | FALSE             | <none>     | TRUE                    | ^[0-9]{4,4}\$                                  |
| Zambia                           | ZM           | FALSE             | <none>     | TRUE                    | ^[0-9]{5,5}\$                                  |
| Zimbabwe                         | ZW           | FALSE             | <none>     | FALSE                   | <none>   |

**State Codes for Canada**

| State Code | State                     |
|------------|---------------------------|
| AB         | Alberta                   |
| BC         | British Columbia          |
| MB         | Manitoba                  |
| NB         | New Brunswick             |
| NL         | Newfoundland and Labrador |
| NT         | Northwest Territories     |
| NS         | Nova Scotia               |
| NU         | Nunavut                   |
| ON         | Ontario                   |
| PE         | Prince Edward Island      |
| QC         | Quebec                    |
| SK         | Saskatchewan              |
| YT         | Yukon                     |

**State Codes for Mexico**

| <b>State Code</b> | <b>State</b>        |
|-------------------|---------------------|
| AG                | Aguascalientes      |
| BC                | Baja California     |
| BS                | Baja California Sur |
| CM                | Campeche            |
| CS                | Chiapas             |
| CH                | Chihuahua           |
| CO                | Coahuila            |
| CL                | Colima              |
| DF                | Federal District    |
| DG                | Durango             |
| MX                | Mexico State        |
| GT                | Guanajuato          |
| GR                | Guerrero            |
| HG                | Hidalgo             |
| JA                | Jalisco             |
| MI                | Michoacan           |
| MO                | Morelos             |
| NA                | Nayarit             |
| NL                | Nuevo Leon          |
| OA                | Oaxaca              |
| PU                | Puebla              |
| QT                | Queretaro           |
| QR                | Quintana Roo        |
| SL                | San Luis Potosi     |
| SI                | Sinaloa             |
| SO                | Sonora              |
| TB                | Tabasco             |
| TM                | Tamaulipas          |
| TL                | Tlaxcala            |
| VE                | Veracruz            |
| YU                | Yucatan             |
| ZA                | Zacatecas           |

**State Codes for the United States**

| <b>State Code</b> | <b>State</b>         |
|-------------------|----------------------|
| AL                | Alabama              |
| AK                | Alaska               |
| AZ                | Arizona              |
| AR                | Arkansas             |
| CA                | California           |
| CO                | Colorado             |
| CT                | Connecticut          |
| DE                | Delaware             |
| DC                | District of Columbia |
| FL                | Florida              |
| GA                | Georgia              |
| HI                | Hawaii               |
| ID                | Idaho                |
| IL                | Illinois             |
| IN                | Indiana              |
| IA                | Iowa                 |
| KS                | Kansas               |
| KY                | Kentucky             |
| LA                | Louisiana            |
| ME                | Maine                |
| MD                | Maryland             |
| MA                | Massachusetts        |
| MI                | Michigan             |
| MN                | Minnesota            |
| MS                | Mississippi          |
| MO                | Missouri             |
| MT                | Montana              |
| NE                | Nebraska             |
| NV                | Nevada               |
| NH                | New Hampshire        |
| NJ                | New Jersey           |
| NM                | New Mexico           |
| NY                | New York             |
| NC                | North Carolina       |
| ND                | North Dakota         |

| State Code | State          |
|------------|----------------|
| OH         | Ohio           |
| OK         | Oklahoma       |
| OR         | Oregon         |
| PA         | Pennsylvania   |
| RI         | Rhode Island   |
| SC         | South Carolina |
| SD         | South Dakota   |
| TN         | Tennessee      |
| TX         | Texas          |
| UT         | Utah           |
| VT         | Vermont        |
| VA         | Virginia       |
| WA         | Washington     |
| WV         | West Virginia  |
| WI         | Wisconsin      |
| WY         | Wyoming        |

**State Codes for China**

| State Code | State                            |
|------------|----------------------------------|
| BJ         | Beijing Municipality             |
| TJ         | Tianjin Municipality             |
| HE         | Hebei Province                   |
| SX         | Shanxi Province                  |
| NM         | Inner Mongolia Autonomous Region |
| LN         | Liaoning Province                |
| JL         | Jilin Province                   |
| HL         | Heilongjiang Province            |
| SH         | Shanghai Municipality            |
| JS         | Jiangsu Province                 |
| ZJ         | Zhejiang Province                |
| AH         | Anhui Province                   |
| FJ         | Fujian Province                  |
| JX         | Jiangxi Province                 |
| SD         | Shandong Province                |
| HA         | Henan Province                   |

| State Code | State                             |
|------------|-----------------------------------|
| HB         | Hubei Province                    |
| HN         | Hunan Province                    |
| GD         | Guangdong Province                |
| GX         | Guangxi Zhuang Autonomous Region  |
| HI         | Hainan Province                   |
| CQ         | Chongqing Municipality            |
| SC         | Sichuan Province                  |
| GZ         | Guizhou Province                  |
| YN         | Yunnan Province                   |
| XZ         | Tibet Autonomous Region           |
| SN         | Shaanxi Province                  |
| GS         | Gansu Province                    |
| QH         | Qinghai Province                  |
| NX         | Ningxia Hui Autonomous Region     |
| XJ         | Xinjiang Uyghur Autonomous Region |

## Creating 'Office 365' Service Terms

This section describes how to create a service terms set for the **Office 365** application. This service terms set will be used in 'Office 365' service plans.

To create the service terms set, perform the following actions:

- 1** Go to **System > Settings > Terms** in the **Finance** group.
- 2** Select the **Service Terms** tab.
- 3** Click **Add New Service Terms**.
- 4** Specify the following properties:

### General

- **Name: Office 365.**

### Regular Subscriptions Policy

- **Grace Period (days): 30.**
- **Hold Period (days): 90.**
- **Destroy Service after Hold Period: Yes.**

### Trial Subscriptions Policy

- **Grace Period: 30.**

- **Hold Period: 30.**
- **Destroy Service after Hold Period: Yes.**

5 Click **Save**.

## Configuring Offers

This section describes how to configure Office 365 offers.

**Important:** Before configuring Office 365 offers, read [Restrictions and Limitations](#) (page 66).

The **Office 365** application supports two types of offers: **License** and **Add-on**. The term **License** denotes an offer that can be ordered alone or in a combination with a compatible offer. The term **Add-on** denotes an offer that requires a subscription for another offer ('parent') and cannot be ordered alone.

There are two groups of offers in the Microsoft cloud: offers that support only monthly billing frequency; offers that support both monthly billing frequency and yearly billing frequency. For offers of the first group, you can create service plans with only monthly billing periods (the **Billing Period** setting is set to **Monthly**). For offers of the second group, you can create service plans with monthly billing periods (the **Billing Period** setting is set to **Monthly**) and yearly billing periods (the **Billing Period** setting is set to either **Yearly** or **Custom <a number of years> Year(s)**).

**Note:** Microsoft did not release new separate yearly offers. Microsoft just enabled yearly billing frequency for some existing CSP offers without changing their Offer IDs. You can see offers that support yearly billing frequency in your latest Cloud Reseller Offer Matrix.

### General Procedure

The offer configuration procedure consists of the following steps:

- 1 Creating license types. See [Creating License Types](#) (page 66) for details.
- 2 Creating add-on types. See [Creating Add-on Types](#) (page 67) for details.
- 3 Creating Office 365 resource types. See [Creating Office 365 Resource Types](#) (page 69) for details.
- 4 Creating Office 365 service template. See [Creating Office 365 Service Template](#) (page 74) for details.
- 5 Creating Office 365 service plans. See [Creating Office 365 Service Plans](#) (page 77) for details.
- 6 Creating Office 365 resource and sales categories. See [Creating 'Office 365' Resource and Sales Categories](#) (page 80) for details.

- 7 Setting up units of measure for 'Office 365 License' and 'Office 365 Add-on' resources. See [Setting Up Units of Measure for 'Office 365 License' and 'Office 365 Add-on' Resources](#) (page 80) for details.

## Automated Procedure

To configure offers automatically, perform the following actions:

- 1 Prepare the **CSP Deployment Assistant** tool:
  - a Upload the latest file the **CSP Deployment Assistant** tool from <https://www.dropbox.com/sh/flvexj79k532lma/AADVZXXgraLg7DbQilqYnf1a?dl=0> to your workstation (Office365DeploymentAssistant.x.x.x.xlsx for Microsoft Cloud, Office365DeploymentAssistant.DE.x.x.x.xlsx for Microsoft Cloud Germany, or Office365DeploymentAssistant.CN.x.x.x.xlsx for Office 365 operated by 21Vianet in China).
  - b Run the **CSP Deployment Assistant** tool.
- 2 Prepare the offer auto-configuration script `autoconf.py`:
  - a Upload the **Office 365** application package to the OA Operations Management Node.
  - b Unpack the package and extract the content of the `0365-Import.zip` file.
  - c Go to the directory where the content of the extracted `0365-Import.zip` file is placed.
  - d Unpack the `simplejson-2.3.2.tar` file.
  - e Go to the `simplejson-2.3.2` directory.
  - f Execute the `python setup.py install` command.
  - g Return to the parent directory.

**Important:** If the version of Python on the OA Operations Management Node is less than **2.6.6**, perform this step on the separate host. See the [Prerequisites section](#) (page 15) for details. Also, set the corresponding values for the `__dbhost`, `__dbname`, `__dbuser`, `__dbpass`, `__apihost` parameters in the `Utils.py` file.

- 3 Create the `CustomLicenseTypes.json` and `CustomServicePlans.json` files using the instruction provided by the **CSP Deployment Assistant** tool. Place these files in the directory where the offer auto-configuration script `autoconf.py` is prepared.

**Note:** We recommend that you check the files using a JSON validation tool. For example, you can use <http://jsonlint.com/>.

- 4 Update the `resourcecategories_CSP.json` and `salescategories_CSP.json` files using the information provided on the **CustomResourceCategories.json** and **CustomSalesCategories.json** tabs of the **CSP Deployment Assistant** tool. When updating the files, you must not change their names.

**Note:** We recommend that you check the files using a JSON validation tool. For example, you can use <http://jsonlint.com/>.

- 5 Obtain the application instance ID of the required instance of the **Office 365** application:
  - a In Provider Control Panel, go to **Services > Applications**.
  - b In the list of applications, click the **Office 365** application.
  - c Select the **Instances** tab.
  - d Write down the identifier of the required instance.

- 6 Specify the prepared application instance ID in the **CSP Deployment Assistant** tool.

- 7 Execute the first auto-configuration script command provided in the **CSP Deployment Assistant** tool to create the add-on types, license types, Office 365 resource types, and **Office 365** service template. For example:

```
python autoconf.py --license-types-json CustomLicenseTypes.json
--service-plans-json CustomServicePlans.json --o365-app-
instance-id 1
```

- 8 Prepare a service plan that will be used as a template for the Office 365 service plans. Make sure the following preconditions are satisfied:
  - The service plan is based on the **Office 365** service template.
  - The service plan must have one subscription period (**1 Year**).
  - The **Service Terms** property of the service plan is set to **Office 365**. See the [Creating 'Office 365' Service Terms section](#) (page 61) for details.
  - The **Auto-Renewal** property of the service plan is enabled and configured.

Write down the service plan ID of the service plan.

- 9 Specify the prepared service plan ID of the template service plan in the **CSP Deployment Assistant** tool.

- 10 Execute the second auto-configuration script command provided in the **CSP Deployment Assistant** tool to create the Office 365 service plans. For example:

```
python autoconf.py --license-types-json CustomLicenseTypes.json
--service-plans-json CustomServicePlans.json --o365-app-
instance-id 1 --pba-stub-plan-id 100000 --langs en,de,fr
```

- 11 Include the **PBA** resource type into the just created **Office 365** service template. The resource limit must be set to **Unlimited**.

**Note:** The **PBA** resource type must be included if you need integration between the **Office 365** application and OA Billing. This integration allows customers to adjust license limits and upgrade licenses from the application screen located at **Office 365 > Licenses**. If you do not need this integration, do not include the resource type into the **Office 365** service template.

- 12 Validate properties of the created Office 365 service plans.



**Notes:**

1. If the version of Python on the OA Operations Management Node is less than **2.6.6**, the offer auto-configuration script `autoconf.py` must be executed on the separate host. See the [Prerequisites section](#) (page 15) for details.
2. You can use the `--prefix` option to add a prefix to the names of Office 365 resource types, Office 365 service templates, and Office 365 service plans that are created by the script. For example, if the **ABC** prefix is specified, the names will start with **ABC:**.
3. The script also creates the `incompatibilitylist.json` file, which contains the offer compatibility information. You must keep this file because it will be used later in the [Configuring Offer Compatibility Checker section](#) (page 83).
4. You can create your custom license types, add-on types, Office 365 resource types, Office 365 service templates, and Office 365 service plans by using the `CustomLicenseTypes.json` and `CustomServicePlans.json` files as templates.
5. See the [Examples of Office 365 Service Plans section](#) (page 118) to obtain the complete example of the Office 365 offer configuration.

It is possible to run the script many times for existing installations of the **Office 365** application. For instance, you can create custom `.json` configuration files containing new offers and run the script for your installation containing already configured offers. In this case the script does the following:

- 1 It creates the license types for the new offers.
- 2 It creates the add-on types for the new add-on offers.
- 3 It updates the lists of possible transitions for all license types.
- 4 It updates the lists of parent offers for all add-on types.
- 5 It creates the corresponding resource types for the new offers and add-on offers.
- 6 It updates the corresponding **Office 365** service template.
- 7 It creates the corresponding Office 365 service plans for the new offers.
- 8 It updates the lists of allowed upgrades for all Office 365 service plans.

**Note:** If you have used a prefix for the initial offer configuration, you must use the same prefix for further offer reconfigurations.

**Manual Procedure**

To configure offers manually, use the instructions provided in the sections below.

**Note:** See the [Examples of Office 365 Service Plans section](#) (page 118) to obtain the complete example of the Office 365 offer configuration.

## Restrictions and Limitations

Take into account the following restrictions and limitations:

- A vendor selling Office 365 services to customers must use Office 365 resources of one application instance.
- It is not allowed to include Office 365 resources of different Office 365 application instances into the same service template/service plan.
- It is not allowed to subscribe a customer account to Office 365 service plans of different Office 365 application instances.

Note that your Office 365 application might not work properly due to violations of these restrictions and limitations.

## Creating License Types

This section describes how to create license types for offers.

You must create a separate license type for each offer that you are going to provide. To create a license type, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Instances** tab.
- 2 Select the required Office 365 application instance.
- 3 Select the **License Types** tab.
- 4 Click **New**.
- 5 Specify the following parameters:
  - **License Name** – The name of the license type. We recommend that you use offer names from your product offer catalogue.
  - **Offer ID** – The offer identifier of the required offer. You must use offer GUIDs from your product offer catalogue.
  - **Trial Offer ID** – Leave empty.
  - **Provisioning ID** – The provisioning identifier of the required offer. You must use the provisioning identifier from your offer catalogue.
  - **Can be assigned to users** – It defines whether the offer can be assigned to users (**Yes/No**). Set the parameter to **No** for 'Yammer Enterprise', 'Intune', 'Office 365 Exchange Online Protection', and 'Enterprise Mobility Suite' offers. Set the parameter to **Yes** for other offers.
  - **Description** – The description of the license type.

Specify the list of possible transitions for the license type. You must use your product offer catalogue to obtain information about allowed transitions from one offer to another offer.

**6** Click **Save Changes**.

For example, create the following license types:

- **Office 365 Business Essentials**
  - **License Name:** Office 365 Business Essentials
  - **Offer ID:** bd938f12-058f-4927-bba3-ae36b1d2501c
  - **Trial Offer ID:** empty
  - **Provisioning ID:** O365\_BUSINESS\_ESSENTIALS
  - **Can be assigned to users:** Yes
  - Possible transitions: **Office 365 Business Premium**
- **Office 365 Business Premium**
  - **License Name:** Office 365 Business Premium
  - **Offer ID:** 031c9e47-4802-4248-838e-778fb1d2cc05
  - **Trial Offer ID:** empty
  - **Provisioning ID:** O365\_BUSINESS\_PREMIUM
  - **Can be assigned to users:** Yes
  - Possible transitions: empty

**Note:** See the [Examples of Office 365 Service Plans section](#) (page 118) to obtain the complete example of the Office 365 offer configuration.

## Creating Add-on Types

This section describes how to create add-on types for add-on offers.

You must create a separate add-on type for each add-on offer that you are going to provide. To create an add-on type, perform the following actions:

- 1** In Provider Control Panel, go to **Services > Applications** and select the **APS Instances** tab.
- 2** Select the required Office 365 application instance.
- 3** Select the **Addon Types** tab.
- 4** Click **New**.
- 5** Specify the following parameters:

- **Addon Name** – The name of the add-on type. We recommend that you use offer names from your product offer catalogue.
- **Offer ID** – The offer identifier of the required offer. You must use offer GUIDs from your product offer catalogue.
- **Provisioning ID** – The provisioning identifier of the required offer. You must use the provisioning identifier from your offer catalogue.
- **Can be assigned to users** – It defines whether the add-on offer can be assigned to users (**Yes/No**). See your product offer catalogue for details.

**Note:** If in the description of an offer you see "Priced per gigabyte", it means the offer cannot be assigned to users.

- **Description** – The description of the add-on type.

In the list below, select parent offers of the add-on offer. See your product offer catalogue for details.

## 6 Click **Save Changes**.

For example, create the following add-on types:

- **Office 365 Extra File Storage**
  - **Add-on Name:** Office 365 Extra File Storage
  - **Offer ID:** 53fc25f7-6639-4f78-bb44-3c2dfec3ed40
  - **Provisioning ID:** SHAREPOINTSTORAGE
  - **Can be assigned to users:** No
  - **Parents offers:** Office 365 Business Essentials, Office 365 Business Premium
- **Exchange Online Archiving for Exchange Online**
  - **Add-on Name:** Exchange Online Archiving for Exchange Online
  - **Offer ID:** 2828be95-46ba-4f91-b2fd-0bef192ecf60
  - **Provisioning ID:** EXCHANGEARCHIVE\_ADDON
  - **Can be assigned to users:** Yes
  - **Parents offers:** Office 365 Business Essentials, Office 365 Business Premium
- **Exchange Online Advanced Threat Protection**
  - **Add-on Name:** Exchange Online Advanced Threat Protection
  - **Offer ID:** a2706f86-868d-4048-989b-0c69e5c76b63
  - **Provisioning ID:** ATP\_ENTERPRISE
  - **Can be assigned to users:** Yes
  - **Parents offers:** Office 365 Business Essentials, Office 365 Business Premium

**Note:** See the [Examples of Office 365 Service Plans](#) section (page 118) to obtain the complete example of the Office 365 offer configuration.

## Creating Office 365 Resource Types

This section describes how to create Office 365 resource types.

### Creating 'Office 365 Global Settings' Resource Type

'Office 365 Global Settings' resource types represent Office 365 application instances. The 'Office 365 Global Settings' resource type must be created for each Office 365 application instance.

To create an 'Office 365 Global Settings' resource type, perform the following steps:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.
- 2 In the list of applications, select the **Office 365** application.
- 3 Select the **Resource Types** tab.
- 4 Click **Create**.
- 5 In the list of resource classes, select the **Application Service Reference** resource class.
- 6 Specify the following parameters:
  - **Name** – The name of the resource type. For example: **Office 365 Global Settings**.
  - **Description** – The description of the resource type. This parameter is optional.Click **Next**.
- 7 In the list of APS types, select the **Office 365 global settings** APS type.
- 8 In the list of instances, select the required Office 365 application instance.
- 9 Check resource type parameters carefully before creation and click **Finish**.

### Creating 'Office 365 Tenant' Resource Type

The **Office 365 Tenant** resource type represents Office 365 organizations. One **Office 365 Tenant** resource type must be created.

To create the **Office 365 Tenant** resource type, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.
- 2 In the list of applications, select the **Office 365** application.
- 3 Select the **Resource Types** tab.
- 4 Click **Create**.

- 5 In the list of resource classes, select the **Application Service** resource class.
- 6 Specify the following parameters:
  - **Name** – The name of the resource type. For example: **Office 365 Tenant**.
  - **Description** – The description of the resource type. This parameter is optional.

Click **Next**.

- 7 In the list of services, select the **Tenant** service.
- 8 Specify the following parameters:
  - **Priority** – The priority of the resource type. This parameter is optional. Leave the default value.
  - **Automatically provision service** – It defines whether to automatically provision the service. Specify **On**.

Click **Next**.

- 9 Check resource type parameters carefully before creation and click **Finish**.

## Creating 'Office 365 User' Resource Type

The **Office 365 User** resource type represents Office 365 users of Office 365 organizations. One **Office 365 User** resource type must be created.

To create the **Office 365 User** resource type, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.
- 2 In the list of applications, select the **Office 365** application.
- 3 Select the **Resource Types** tab.
- 4 Click **Create**.
- 5 In the list of resource classes, select the **Application Service** resource class.
- 6 Specify the following parameters:
  - **Name** – The name of the resource type. For example: **Office 365 User**.
  - **Description** – The description of the resource type. This parameter is optional.

Click **Next**.

- 7 In the list of services, select the **User** service.
- 8 Specify the following parameters:
  - **Priority** – The priority of the resource type. This parameter is optional. Leave the default value.
  - **Automatically provision service** – It defines whether to automatically provision the service. Specify **Off**.

Click **Next**.

- 9 Check resource type parameters carefully before creation and click **Finish**.

## Creating 'Office 365 Domain' Resource Type

The **Office 365 Domain** resource type represents Office 365 domains of Office 365 organizations. One **Office 365 Domain** resource type must be created.

To create the **Office 365 Domain** resource type, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.
- 2 In the list of applications, select the **Office 365** application.
- 3 Select the **Resource Types** tab.
- 4 Click **Create**.
- 5 In the list of resource classes, select the **Application Service** resource class.
- 6 Specify the following parameters:
  - **Name** – The name of the resource type. For example: **Office 365 Domain**.
  - **Description** – The description of the resource type. This parameter is optional.

Click **Next**.

- 7 In the list of services, select the **Domain** service.
- 8 Specify the following parameters:
  - **Priority** – The priority of the resource type. This parameter is optional. Leave the default value.
  - **Automatically provision service** – It defines whether to automatically provision the service. Specify **Off**.

Click **Next**.

- 9 Check resource type parameters carefully before creation and click **Finish**.

## Creating 'Office 365 License' Resource Types

'Office 365 License' resource types represent Office 365 offers. A separate 'Office 365 License' resource type must be created for each license type you created earlier (see the [Creating License Types section](#) (page 66) for details).

To create an 'Office 365 License' resource type, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.
- 2 In the list of applications, select the **Office 365** application.
- 3 Select the **Resource Types** tab.

- 4 Click **Create**.
- 5 In the list of resource classes, select the **Application Service Reference** resource class.
- 6 Specify the following parameters:
  - **Name** – The name of the resource type.
  - **Description** – The description of the resource type. This parameter is optional.Click **Next**.
- 7 In the list of APS types, select the **License** APS type.
- 8 In the list of instances, select the required license type instance.
- 9 Check resource type parameters carefully before creation and click **Finish**.

For example, create the following 'Office 365 License' resource types:

- **Office 365 Business Essentials**
  - License type instance: **Office 365 Business Essentials**
- **Office 365 Business Premium**
  - License type instance: **Office 365 Business Premium**

**Note:** See the [Examples of Office 365 Service Plans](#) section (page 118) to obtain the complete example of the Office 365 offer configuration.

## Creating 'Office 365 Add-on' Resource Types

'Office 365 Add-on' resource types represent Office 365 add-on offers. A separate 'Office 365 Add-on' resource type must be created for each add-on type you created earlier (see the [Creating Add-on Types](#) section (page 67) for details).

To create an 'Office 365 Add-on' resource type, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.
- 2 In the list of applications, select the **Office 365** application.
- 3 Select the **Resource Types** tab.
- 4 Click **Create**.
- 5 In the list of resource classes, select the **Application Service Reference** resource class.
- 6 Specify the following parameters:
  - **Name** – The name of the resource type.
  - **Description** – The description of the resource type. This parameter is optional.Click **Next**.
- 7 In the list of APS types, select the **Addon** APS type.



- 8 In the list of instances, select the required add-on type instance.
- 9 Check resource type parameters carefully before creation and click **Finish**.

For example, create the following 'Office 365 Add-on' resource types:

- **Office 365 Extra File Storage (1 GB)**
  - Add-on type instance: **Office 365 Extra File Storage**
- **Exchange Online Archiving for Exchange Online**
  - Add-on type instance: **Exchange Online Archiving for Exchange Online**
- **Exchange Online Advanced Threat Protection**
  - Add-on type instance: **Exchange Online Advanced Threat Protection**

**Note:** See the [Examples of Office 365 Service Plans](#) section (page 118) to obtain the complete example of the Office 365 offer configuration.

## Creating Auxiliary Resource Types

Create the following auxiliary resource types:

- **Office 365 Subscription Provisioning**
- **Async Operations**
- **Office 365 Tenant Contact Data**
- **Total seats sold**
- **Total points sold**

To create the **Office 365 Subscription Provisioning**, **Async Operations**, and **Office 365 Tenant Contact Data** resource types, perform the following actions:

- 1 In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.
- 2 In the list of applications, select the **Office 365** application.
- 3 Select the **Resource Types** tab.
- 4 Click **Create**.
- 5 In the list of resource classes, select the **Application Service** resource class.
- 6 Specify the following parameters:
  - **Name** – The name of the resource type. For example: **Office 365 Subscription Provisioning**, **Async Operations**, **Office 365 Tenant Contact Data**.
  - **Description** – The description of the resource type. This parameter is optional.

Click **Next**.

**7** In the list of services, select the **Subscription** (for the **Office 365 Subscription Provisioning** resource type), **AsyncOperations** (for the **Async Operations** resource type), or **TenantContactData** (for the **Office 365 Tenant Contact Data** resource type) service.

**8** Specify the following parameters:

- **Priority** – The priority of the resource type. This parameter is optional. Leave the default value.
- **Automatically provision service** – It defines whether to automatically provision the service. Specify **Off**.

Click **Next**.

**9** Check resource type parameters carefully before creation and click **Finish**.

To create the **Total seats sold** and **Total points sold** resource types, perform the following actions:

**1** In Provider Control Panel, go to **Services > Applications** and select the **APS Packages** tab.

**2** In the list of applications, select the **Office 365** application.

**3** Select the **Resource Types** tab.

**4** Click **Create**.

**5** In the list of resource classes, select the **Application Counter (unit)** resource class.

**6** Specify the following parameters:

- **Name** – The name of the resource type. For example: **Total seats sold**, **Total points sold**.
- **Description** – The description of the resource type. This parameter is optional.

Click **Next**.

**7** In the list of resources, select the **Total number of seat limits of all corresponding MS subscriptions** (for the **Total seats sold** resource type) or **Total number of points of all corresponding MS subscriptions** (for the **Total points sold** resource type) resource.

**8** Check resource type parameters carefully before creation and click **Finish**.

## Creating 'Office 365' Service Template

This section describes how to create the **Office 365** service template. You must create only one **Office 365** service template.

To create the **Office 365** service template, perform the following actions:

**1** In Provider Control Panel, go to **Products > Service Templates**.

**2** Click **Add New Service Template**.

**3** Specify the following parameters:

- **Name** – The name of the service template. For example: **Office 365**.
- **Description** – The description of the service template. This parameter is optional.
- **Autoprovisioning** – It defines whether subscriptions based on the service template are auto-provisioned (**Yes/No**). Select **Yes**.
- **Type** – It defines the type of the service template. Select the **Custom** value from the drop-down list.

Click **Next**.

**4** Include the following resource types:

- 'Office 365 Global Settings' resource type. The resource limit must be set to **Unlimited**.
- **Office 365 Tenant**. The resource limit must be set to **1**.
- **Office 365 User**. The resource limit defines the maximum number of Office 365 users, which can be created in an Office 365 organization. The resource limit must be set to **Unlimited**.
- **Office 365 Domain**. The resource limit defines the maximum number of domains, which can be created in an Office 365 organization. The resource limit must be set to **Unlimited**.
- All 'Office 365 License' resource types. The resource limit defines the maximum number of Office 365 licenses of a particular type, which can be assigned to Office 365 users. The resource limit must be set to **0**. Note that the actual resource limit will be configured in the corresponding service plan. See [Creating 'Office 365' Service Plans](#) (page 77) for details.
- All 'Office 365 Add-on' resource types. The resource limit defines the maximum number of Office 365 add-on licenses of a particular type, which can be used in an Office 365 organization. The resource limit must be set to **0**. Note that the actual resource limit will be configured in the corresponding service plan. See [Creating 'Office 365' Service Plans](#) (page 77) for details.
- **PBA**. The resource limit must be set to **Unlimited**.

**Note:** The **PBA** resource type must be included if you need integration between the **Office 365** application and OA Billing. This integration allows customers to adjust license limits and upgrade licenses from the application screen located at **Office 365 > Licenses**. If you do not need this integration, do not include the resource type into the **Office 365** service template.

- **Office 365 Subscription Provisioning**. The resource limit must be set to **1**.
- **Async Operations**. The resource limit must be set to **Unlimited**.
- **Office 365 Tenant Contact Data**. The resource limit must be set to **1**.
- **Total points sold**. The resource limit must be set to **Unlimited**.
- **Total seats sold**. The resource limit must be set to **Unlimited**.

- **DNS Hosting.** The resource limit defines the maximum number of DNS zones. The resource limit must be set to **Unlimited**.

Click **Next**.

- 5 Specify the resource limits and click **Next**.
- 6 Validate service template parameters carefully and click **Finish**.
- 7 Activate the just created service template.
- 8 Go to **Products > Service Plans**.
- 9 Select the **Service Templates** tab.
- 10 Click on the just created service template.
- 11 Configure the **Destroy Service on Cancel** property of the service template:
  - a Select the **General** tab.
  - b Click **Edit**.
  - c Set **Destroy Service on Cancel** to **No** (the checkbox is clear).
  - d Click **Save**.
- 12 Select the **Service Parameters** tab.
- 13 Add the **aps\_defaults\_o365** service parameter to the service template:
  - a Click **Add New Parameter**.
  - b Specify the following properties:
    - **Parameter ID** – The **aps\_defaults\_o365** value must be specified.
    - **Name** – The **Office 365 Subdomain** value must be specified.
    - **Description** – Leave empty.
    - **Value Type** – The **String** value must be chosen.
    - **Required** – Leave the checkbox empty (**No**).
    - **Ask on Order placement** – Select the checkbox (**Yes**).
  - c Click **Save**.
- 14 Add the **o365\_based\_on\_email** service parameter to the service template:
  - a Click **Add New Parameter**.
  - b Specify the following properties:
    - **Parameter ID** – The **o365\_based\_on\_email** value must be specified.
    - **Name** – The **Office 365 Subdomain Based On Email** value must be specified.
    - **Description** – Leave empty.
    - **Value Type** – The **String** value must be chosen.
    - **Required** – Leave the checkbox empty (**No**).

- **Ask on Order placement** – Select the checkbox (**Yes**).

c Click **Save**.

**Notes:**

1. To obtain the detailed information about the **DNS Hosting** resource type, refer to **OA Operations Provider's Guide >> Managing Domains > Administering Domain > Managing DNS > DNS Hosting in OA Operations > Creating Resource Type for DNS Hosting**.

2. If the **Office 365** service template is intended for a Test in Production (TIP) account, you can speed up resource cleanup of the TIP account by setting up **Destroy Service on Cancel** to **Yes**.

For example, create the following service template:

**Office 365**

| Resource Type                                 | Limit     |
|---|-----------|
| 'Office 365 Global Settings'                  | Unlimited |
| Office 365 Tenant                             | 1         |
| Office 365 User                               | Unlimited |
| Office 365 Domain                             | Unlimited |
| Office 365 Business Essentials                | 0         |
| Office 365 Business Premium                   | 0         |
| Office 365 Extra File Storage (1 GB)          | 0         |
| Exchange Online Archiving for Exchange Online | 0         |
| Exchange Online Advanced Threat Protection    | 0         |
| PBA   | Unlimited |
| Office 365 Subscription Provisioning          | 1         |
| Async Operations                              | Unlimited |
| Office 365 Tenant Contact Data                | 1         |
| Total points sold                             | Unlimited |
| Total seats sold                              | Unlimited |
| DNS Hosting                                   | Unlimited |

## Creating 'Office 365' Service Plans

This section describes how to create 'Office 365' service plans.

You must create a separate service plan for each 'Office 365 License' resource. In some cases you may need more than one service plan for a particular 'Office 365 License' resource. See the [Examples of Office 365 Service Plans section](#) (page 118) to obtain the complete example of the Office 365 offer configuration.

A customer account can be subscribed to several 'Office 365' service plans and only one organization is created in the Microsoft cloud for the customer account. This allows customers to use offers from different subscriptions.

There are two groups of offers in the Microsoft cloud: offers that support only monthly billing frequency; offers that support both monthly billing frequency and yearly billing frequency. For offers of the first group, you can create service plans with only monthly billing periods (the **Billing Period** setting is set to **Monthly**). For offers of the second group, you can create service plans with monthly billing periods (the **Billing Period** setting is set to **Monthly**) and yearly billing periods (the **Billing Period** setting is set to either **Yearly** or **Custom <a number of years> Year(s)**). Note that Microsoft did not release new separate yearly offers. Microsoft just enabled yearly billing frequency for some existing CSP offers without changing their Offer IDs. You can see offers that support yearly billing frequency in your latest Cloud Reseller Offer Matrix.

**Important:** A customer account must not be subscribed to 'Office 365' service plans that contain incompatible offers. If it happens, the corresponding provisioning tasks will fail. To resolve this problem, you must cancel these tasks and cancel incorrectly created subscriptions. To obtain the offer compatibility rules, see your offer catalogue.

To create an 'Office 365' service plan for an 'Office 365 License' resource, perform the following actions:

- 1 Create a separate service plan on basis of the **Office 365** service template. The **Office 365** service terms set must be used in the service plan (see the [Creating 'Office 365' Service Terms section](#) (page 61) for details).
- 2 Add a resource rate for the required 'Office 365 License' resource in the corresponding service plan. Make sure a positive value is specified in the **Inc. Amount** field of the resource rate. Also, make sure **Show in Store** is set to **Yes**.
- 3 Add resource rates for required 'Office 365 Add-on' resources in the corresponding service plan. Make sure **Show in Store** is set to **Yes**. You must use your product offer catalogue to obtain information about allowed combinations of offers.

When all 'Office 365' service plans are created, specify allowed upgrade paths for these service plans. Note the following before configuring allowed upgrade paths:

- You must use your product offer catalogue to obtain information about allowed transitions from one offer to another offer.
- Upgrades between offers with different billing frequency are not allowed.

For example, create the following service plans:

### Office 365 Business Essentials

- **Service Template:** Office 365
- **Service Terms:** Office 365
- **Billing Period:** Monthly
- **Subscription Period:** Not Trial/1 Year/\$5 Recurring Fee
- **Resource Rates**
  - **Resource/Inc. Amount/Max Amount/Recurring Fee:** Office 365 Business Essentials/1/300/\$5
- **Upgrades:** Office 365 Business Essentials with Add-ons service plan, Office 365 Business Premium service plan

**Office 365 Business Essentials with Add-ons**

- **Service Template:** Office 365
- **Service Terms:** Office 365
- **Billing Period:** Monthly
- **Subscription Period:** Not Trial/1 Year/\$5 Recurring Fee
- **Resource Rates**
  - **Resource/Inc. Amount/Max Amount/Recurring Fee:** Office 365 Business Essentials/1/300/\$5
  - **Resource/Inc. Amount/Max Amount/Recurring Fee:** Office 365 Extra File Storage (1 GB)/0/10000000/\$0.20
  - **Resource/Inc. Amount/Max Amount/Recurring Fee:** Exchange Online Archiving for Exchange Online/0/10000000/\$3
  - **Resource/Inc. Amount/Max Amount/Recurring Fee:** Exchange Online Advanced Threat Protection/0/10000000/\$2
- **Upgrades:** Office 365 Business Essentials service plan

**Office 365 Business Premium**

- **Service Template:** Office 365
- **Service Terms:** Office 365
- **Billing Period:** Monthly
- **Subscription Period:** Not Trial/1 Year/\$13.5 Recurring Fee
- **Resource Rates**
  - **Resource/Inc. Amount/Max Amount/Recurring Fee:** Office 365 Business Premium/1/300/\$13.5
- **Upgrades:** empty

**Note:** To obtain the detailed instructions on how to configure OA Billing to sell resources, refer to [OA Billing Provider's Guide](#).

## Creating 'Office 365' Resource and Sales Categories

Create resource categories for your Office 365 resources:

- 1 Go to **Products > Resources** and select the **Resource Categories** tab.
- 2 Click **Add New Resource Category**.
- 3 Specify properties of the resource category and click **Add**.

After creating all necessary resource categories, distribute the Office 365 resources among the created resource categories.

Create sales categories for your Office 365 service plans:

- 1 Go to **Products > Service Plans** and select the **Sales Categories** tab.
- 2 Click **Add New Sales Category**.
- 3 Specify properties of the sales category and click **Save**.

After creating all necessary sales categories, distribute the Office 365 service plans among the created sales categories.

**Note:** See the [Examples of Office 365 Service Plans section](#) (page 118) to obtain the complete example of the Office 365 offer configuration.

## Setting Up Units of Measure for 'Office 365 License' and 'Office 365 Add-on' Resources

Set up units of measure for all the 'Office 365 License' and 'Office 365 Add-on' resources:

- 1 Go to **Products > Resources**.
- 2 Select the required 'Office 365' resource.
- 3 Click **Change Unit of Measure**.
- 4 Select **License** (for resources which are measured in items) or **GB** (for resources which are measured in gigabytes) and click **Save**.

**Note:** See the [Examples of Office 365 Service Plans section](#) (page 118) to obtain the complete example of the Office 365 offer configuration.



## Configuring Online Store

This section describes how to configure your online store for selling Office 365 offers.

### In This Chapter

|  |    |
|--|----|
| Selecting Online Store .....                 | 81 |
| Creating 'Office 365' Screen.....            | 81 |
| Configuring 'Checkout' Screen.....           | 82 |
| Creating 'Office 365' Purchase Scenario..... | 82 |
| Synchronizing Online Store.....              | 82 |

### Selecting Online Store

Select the required online store for editing:

- 1 In Provider Control Panel, go to **Products > Online Store**.
- 2 Click **Switch Store**.
- 3 In the list of online stores, click on the required online store.
- 4 Click **Select for editing**.

### Creating 'Office 365' Screen

Create the 'Office 365' screen in the required online store:

- 1 In Provider Control Panel, go to **Products > Online Store**.
- 2 Select the **Screens** tab.
- 3 Click **Add New Screen**.
- 4 Specify the following properties:
  - **Screen ID** – For example, specify **O365\_SCREEN**.
  - **Template Name** – Select the **HOSTING** template.
  - **Title** – For example, specify **Office 365**.
- 5 Click **Save**.
- 6 In the list of screens, click on the just created screen.
- 7 Click **Add Existing Category**.
- 8 Select sales categories of 'Office 365' service plans.

- 9 Click **Finish**.

## Configuring 'Checkout' Screen

Configure the 'Checkout' screen of the required online store in the following way:

- 1 In Provider Control Panel, go to **Products > Online Store**.
- 2 Select the **Screens** tab.
- 3 In the list of screens, click on the 'Checkout' screen.
- 4 Click **Parameters**.
- 5 In the list of parameters, click on the **REGFORM\_TYPE** parameter.
- 6 Click **Edit**.
- 7 Set the **Full** value and click **Save**.

## Creating 'Office 365' Purchase Scenario

Create the 'Office 365' purchase scenario in the required online store:

- 1 In Provider Control Panel, go to **Products > Online Store**.
- 2 Click **Add New Purchase Scenario**.
- 3 Specify the following properties:
  - **Title** – For example, specify **Office 365**.
  - **URL ID** – Specify **O365\_PATH**.
  - **Published in Store** – Select **Yes**.
- 4 Click **Save**.

Add the 'Office 365' and 'Checkout' screens to the 'Office 365' purchase scenario:

- 1 In the list of purchase scenarios, click on the purchase scenario.
- 2 Click **Add Existing Screen**.
- 3 In the list of screens, select the 'Office 365' and 'Checkout' screens and click **Add Screen(s)**.

## Synchronizing Online Store

Synchronize the required online store to apply the changes:

- 1 In Provider Control Panel, go to **Products > Online Store**.
- 2 Click **Synchronize Now**.

## Configuring Office 365 Subdomain Form for OA Billing

The Office 365 subdomain form is a form in which customers must specify their default subdomains (for instance, **customer.onmicrosoft.com** (for Microsoft Cloud), **customer.onmicrosoft.de** (for Microsoft Cloud Germany), or **customer.partner.onmschina.cn** (for Office 365 operated by 21Vianet in China)) on purchasing Office 365 services.

The form is displayed in the following places of OA Billing:

- the 'Add New Order' wizards of Provider, Reseller, and Customer Control Panels;
- the online store;

To configure the Office 365 subdomain form, use the instructions of the <https://kb.odin.com/en/130232> KB article.

Note the following:

- You must use the instruction which belongs to the target version of the **Office 365** application.
- You must use the instruction which belongs to the version of your Odin Automation system.
- Before configuring the Office 365 subdomain form, make sure that the **aps\_defaults\_o365** and **o365\_based\_on\_email** service parameters are added to the **Office 365** service template. See [Creating 'Office 365' Service Template](#) (page 74) to learn more about these service parameters.

## Configuring Offer Compatibility Checker

This section describes how to configure the offer compatibility checker. It allows preventing customers from ordering incompatible offers. Checking offers is performed in the following areas:

- Odin Automation Customer Control Panel, the screens of the **Buy More Services** wizard
- Odin Automation Customer Control Panel, the **Upgrade** screen of the **Office 365** application

To configure the offer compatibility checker, use the following instruction:

- 1 Prepare and write down the identifiers of sales categories containing your Office 365 service plans.
- 2 Configure OA Billing following these steps:

- a Log on to the OA Billing Application Server (**OABLINFE**) as **root**.
  - b Create the `Plan_CDB_BuyNewPlanListHead.xml` file in the `/usr/local/bm/customization` directory. The content of the file is provided below.
  - c Create the `/usr/local/bm/conf/html/o365` directory if it does not exist.
  - d Create the `script2.js` file in the `/usr/local/bm/conf/html/o365` directory. The content of the file is provided below.
  - e In the `script2.js` file replace **OFFICE\_365\_SALE\_CATEGORIES** with the list of the identifiers of the Office 365 sales categories. For example: **1, 2, 3**.
  - f Execute the `service pba restart` command.
  - g Execute the `service www restart` command.
- 3 On your branding site configure the offer compatibility checker:

**Important:** If your branding site resides on a web cluster, you must perform step 3 on each node of the web cluster.

- a Find the OA Operations Branding Server where the branding site resides. For instance: log in to Provider Control Panel; go to **System > Settings > Brands** in the **Branding** group; click on the brand; see the value of the **UI Node** property.
- b Log in to the OA Operations Branding Server as **root**.
- c Make sure PHP support is enabled on the OA Operations Branding Server where the site resides:
  - Check if the `php` and `php-xmlrpc` RPM packages are installed. If necessary, install them using the `yum install php php-xmlrpc` command and then execute the `service httpd reload` command.
  - If the operating system of the OA Operations Branding Server is CloudLinux, add the lines listed below to the `/etc/httpd/conf.d/brand_domain.tld.conf` configuration file and execute the `service httpd reload` command.

```
ScriptAlias /cgi-php/php-cgi /usr/bin/php-cgi
AddHandler php5-script .php .php5 .php4 .php3
Action php5-script /cgi-php/php-cgi
```

**Important:** The `/etc/httpd/conf.d/brand_domain.tld.conf` configuration file is managed by Odin Automation and the manual changes might get lost. In case the manual changes get lost, you must add them again.

- d In the site root directory (`/var/www/brands/<brand_domain_name>`) create the `o365` directory.
- e Create the `check.php` and `.htaccess` files in the `o365` directory. The contents of the files are provided below.
- f In the `check.php` file specify:

- **BA\_API** – The BackNet IP address of the OA Billing Application Server (**OABLINFE**). For example: **10.10.10.10**.
  - **BA\_VERSION** – The version of OA Billing. Set one of the following values depending on the version of OA Billing:
    - **0** – OA Billing version < **5.5.5**;
    - **1** – **5.5.5** <= OA Billing version < **6.0**;
    - **2** – OA Billing version >= **6.0**;
  - **BA\_USERNAME** – The name of the prepared user for connecting to the OA Billing Public API.
  - **BA\_PASSWORD** – The password of the prepared user for connecting to the OA Billing Public API.
- g** Create the `incompatibilitylist.json` file in the `o365` directory. If you have used the automated offer configuration procedure described in the [Configuring Offers section](#) (page 62), use the content of the generated file. Otherwise, fill in the file with the offer compatibility information in the following way:

```
{
  "ServicePlan1": ["IncompatibleServicePlan11",
    "IncompatibleServicePlan12", ..., "IncompatibleServicePlan1M"],
  "ServicePlan2": ["IncompatibleServicePlan21",
    "IncompatibleServicePlan22", ..., "IncompatibleServicePlan2M"],
  ...
  "ServicePlanN": ["IncompatibleServicePlanN1",
    "IncompatibleServicePlanN2", ..., "IncompatibleServicePlanNM"]
}
```

where

**ServicePlan1-ServicePlanN** – The identifiers of Office 365 service plans.

**IncompatibleServicePlan11-IncompatibleServicePlan1M** – The identifiers of Office 365 service plans that are incompatible with the **ServicePlan1** Office 365 service plan.

...

**IncompatibleServicePlanN1-IncompatibleServicePlanNM** – The identifiers of Office 365 service plans that are incompatible with the **ServicePlanN** Office 365 service plan.

For example:

```
{
  "38": ["35", "36", "37", "40", "41", "42", "43"],
  "39": ["35", "36", "37", "40", "41"]
}
```

}

Use your product offer catalogue to obtain the offer compatibility information.

**Important:** For each pair of incompatible service plans the first service plan must be specified in the incompatibility list of the second service plan and the second service plan must be specified in the incompatibility list of the first service plan. For example, the **A** and **B** service plans are incompatible. In this case the `incompatibilitylist.json` file must contain the following information: **"A": ["B", ...]** and **"B": ["A", ...]**.

- 4 Configure the required instance of the **Office 365** application:
  - a Log in to Provider Control Panel.
  - b Go to **Services > Applications > APS Instances** tab > **INSTANCE**.
  - c On the **Global Settings** tab, specify the following global settings:
    - **Service Plans Compatibility Checker URL** – The URL of the offer compatibility checker. This URL must be specified in the [https://BRANDING\\_SITE\\_HOSTNAME/o365/check](https://BRANDING_SITE_HOSTNAME/o365/check) format, where **BRANDING\_SITE\_HOSTNAME** is the hostname of the branding site. For example: <https://brand.com/o365/check>.
    - **Sale Category List** – The list of the identifiers of the Office 365 sales categories. For example: **1,2,3**.

**Note:** To learn more about customizing OA Billing screens, refer to the **OA Billing User Interface Customization Guide**.

### Plan\_CDB\_BuyNewPlanListHead.xml

```
<window id="Plan_CDB_BuyNewPlanListHead">
  <elements>
    <column type="ID" id="PlanID" label="ID" title=" Service plan ID"
visibility="vo" />
  </elements>
  <layout>
    <grid id="jsmysecret" before="body">
      <row id="jssecret">
        <cell>
          <html>
            <div id="alert" class="msg-
content"></div>
            <script
src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
            <script src="/bss-
static/o365/script2.js"></script>
          </html>
        </cell>
      </row>
    </grid>
  </layout>
</window>
```

### script2.js

```
//define office 365 sale categories. eg: var office365salescategories = [1, 15];
var office365salescategories = [OFFICE_365_SALE_CATEGORIES];
```

```

var salescategory = $("#input__StoreCategoryCategoryID").val();
if ($.inArray(parseInt(salescategory), office365salescategories) != -1) {
    //we must detect if this is customer CP or not
    var refreshButton = $("#input__refresh");
    if (refreshButton.length > 0) {
        //let's show the preloader
        $(".page-loading").css("display", "block");
        var href = refreshButton.attr("href");
        var pattern = href.match(/\/sudo([0-9]+)-/i);
        if (pattern.length > 1) {
            var accountId = parseInt(pattern[1]);
            $.when($.ajax(
"/o365/check?cat="+salescategory+"&acc="+accountId).then(function(myreturn) {
                var incompatiblePlansList = $.parseJSON(myreturn);
                var removetable = 0;
                var hasbeenremoved = 0;
                for (var index = 0; index <
incompatiblePlansList.length; ++index) {
                    var incompatiblePlanId =
incompatiblePlansList[index][0];
                    if (incompatiblePlanId == -1) {
                        console.log("We will hide ALL, need to
add html text explaining why");
                        $("#alert").addClass('OpResult-
wrapper').removeClass('text-value');
                        $("#alert").html("<div class='msg-box
msg-warning'
xmlns:global='http://www.swsoft.com/xsl/cp/global'><div><div><div><div><div
class='msg-content'>There are no available new Office 365 license types that are
compatible with your existing licenses. If you intended to increase / decrease
seats or switch those license types, please go to Office 365 >
License.</div></div></div></div></div></div>");
                        removetable = 1;
                        break;
                    }
                    else {
                        var idCell = $("#global_list").find("tr
td:nth-child(2)").filter(function() {return $(this).text().trim() ==
incompatiblePlanId;})
                        if (idCell.length == 1) {
                            var therow = idCell.parent();
                            hasbeenremoved = 1;
                            therow.remove();
                        }
                    }
                }
            })
            $(".page-loading").css("display", "none");
            if (removetable == 1) {
                $("#global_list").remove();
                $("#t1_title").remove();
            }
            if (hasbeenremoved == 1) {
                if (removetable != 1) {
                    $("#alert").addClass('OpResult-
wrapper').removeClass('text-value');

```

```

                $("#alert").html("<div
style='width:940px' class='msg-box msg-warning'
xmlns:global='http://www.swsoft.com/xsl/cp/global'><div><div><div><div><div
class='msg-content'>Here you can buy new Office 365 license types that don't
conflict with your existing licenses. We noticed that you already have existing
Office 365 licenses. If you intended to increase / decrease seats or switch those
license types, please go to Office 365 >
License.</div></div></div></div></div></div>");
            }
        }
        console.log(incompatiblePlansList);
    }, function(therror){
        if(therror.status == 500){
            console.log(therror);
            $("#alert").addClass('OpResult-
wrapper').removeClass('text-value');
            $("#alert").html("<div style='width:940px'
class='msg-box msg-warning'
xmlns:global='http://www.swsoft.com/xsl/cp/global'><div><div><div><div><div
class='msg-content'>An error occurred while checking the compatibility of
subscriptions</div></div></div></div></div></div>");
            $(".page-loading").css("display","none");
            //For security purposes, lets remove the table
in case of error
                $("#global_list").remove();
            }
            $(".page-loading").css("display","none");
        });
    }
    //$(".page-loading").css("display","none");
}
}
else{
    console.log("Request is not for office365");
}
}

```

## check.php

```

<?php
    #require_once 'XML/RPC.php';
    ###Definition of PBA Version, set:
    # 0 if PBA version is less then 5.5.5,
    # 1 if PBA version is 5.5.5 or higher, but less than 6.0,
    # 2 if PBA version is 6.0 or higher.
    # eg: $pbaversion = 0;
    $pbaversion = BA_VERSION;
    ### Define PBA API IP, eg: $PBAAPI = "10.10.10.10"
    $PBAAPI = "BA_API";
    ### Define PBA API Username and Password
    $PBAAPIUserName = "BA_USERNAME";
    $PBAAPIPassword = "BA_PASSWORD";
    ##### We define the incompatible list, format is PLAN => array of
incompatible plans
    ### this must be an array of arrays
    ### Format is: PBA_PLAN => array(list of plans that are incompatible with)
    $json = file_get_contents('./incompatibilitylist.json', true);
    $incompatibilitylist = json_decode($json, true);
    if(!isset($_GET['cat'])){
        throw new Exception ("Blocked");
    }
    if(!isset($_GET['acc'])){

```



```

        throw new Exception ("Blocked");
    }
    if(!is_numeric($_GET['cat'])){
        throw new Exception ("Blocked");
    }
    if(!is_numeric($_GET['acc'])){
        throw new Exception ("Blocked");
    }
    if(isset($_GET['subId']) && is_numeric($_GET['subId'])){
        $excludeSubsID = $_GET['subId'];
    }

    $methodName = "Execute";
    $accountId = $_GET['acc'];
    if ($pbaversion == 2){
        // In that case the $accountId is actually a staff member ID
        $methodParameters = array("Server" => "BM", "Method" =>
"UserDetailsGet_API", "Params" => array($accountId));
        $userDetails = sendRequest($methodName, $methodParameters);
        if(isset($userDetails["Result"][0])){
            $accountId = $userDetails["Result"][0][2];
        }
    }

    $methodParameters = array("Server" => "BM", "Method" =>
"AccountDetailsGet_API", "Params" => array($accountId));
    $vendorcall = sendRequest($methodName, $methodParameters);
    if(isset($vendorcall["Result"][0])){
        $vendor=$vendorcall["Result"][0][1];
    }
    else{
        $vendor=1;
    }

    $subscriptionList = array();
    class Subscription{
        public function __construct($id, $status, $planId) {
            $this->id = $id;
            $this->status = $status;
            $this->planId = $planId;
        }
        public $id;
        public $status;
        public $planId;
    }
    if ($pbaversion >= 1){
        $methodParameters = array("Server" => "BM", "Method" =>
"GetCustomerSubscriptionList_API", "Params" => array($accountId,0));
        $subscriptions = sendRequest($methodName, $methodParameters);
        if(isset($subscriptions["Result"][0])){
            foreach($subscriptions["Result"][0] as $thesub){
                $subscriptionList[] = new Subscription($thesub[0],
$thesub[9], $thesub[2]);
            }
        }
        else{
            throw new Exception("Error on retriving existing
subscriptions");
        }
    }
    }else{

```

```

        $methodParameters = array("Server" => "BM", "Method" =>
"SubscriptionGetCustomerSubscriptionList", "Params" => array($accountId,0));
        $subscriptions = sendRequest($methodName, $methodParameters);
        if(isset($subscriptions["Result"][0])){
            foreach($subscriptions["Result"][0] as $thesub){
                $methodParameters = array("Server" => "BM", "Method" =>
"SubscriptionDetailsGet_API", "Params" => array($thesub[0]));
                $subdetails = sendRequest($methodName,
$methodParameters);
                if(isset($subdetails["Result"][0])){
                    $subscriptionList[] = new
Subscription($subdetails["Result"][0][0], $subdetails["Result"][0][5],
$subdetails["Result"][0][3]);
                }
            }
        }else{
            throw new Exception("Error on retriving existing
subscriptions");
        }
    }

    $planidList = array();
    foreach($subscriptionList as $thesub){
        //10 - Ordered; 15 - Trial 30; - Active 40; - Graced; 50 - Expired;
60 - Terminated; 70 - Canceled; 80 - Administrative Hold; 85 - Credit Hold; 89 -
Credit and Administrative Hold
        if($thesub->status != 60 &&
(!isset($excludeSubsID) || $excludeSubsID != $thesub->id)){
            $planidList[] = $thesub->planId;
        }
    }

    $plancategorylist = array();
    if(count($planidList) != 0){
        $methodParameters = array("Server" => "BM", "Method" =>
"PlanListAvailable4SalesCategoryGet_API", "Params" => array($vendor, NULL,
$_GET['cat'], 0));
        $plancategorylist = sendRequest($methodName, $methodParameters);
    }

    $availableForSalePlansIds = array();
    if(isset($plancategorylist["Result"][0])){
        $position = 1;
        foreach($plancategorylist["Result"][0] as $plancat){
            $availableForSalePlansIds[$plancat[0]] = $position;
            $position++;
        }
    }

    $incompatiblePlansIdsPosDictionary = array();
    foreach($planidList as $planid){
        if(array_key_exists($planid, $incompatibilitylist)){
            foreach($incompatibilitylist[$planid] as $incompatPlanId){
                if(array_key_exists($incompatPlanId,
$availableForSalePlansIds) && is_numeric($incompatPlanId)){
                    $stemposition =
$availableForSalePlansIds[$incompatPlanId];
                    $newValue = array((int)$incompatPlanId,
$stemposition);
                    if(!in_array($newValue,
$incompatiblePlansIdsPosDictionary)){

```

```

        array_push($incompatiblePlansIdsPosDictionary, $newValue);
    }
}
}

###We must also add same plans as exists, since if not, customer may
overwrite limits in P1 for example...
foreach($planidList as $planid){
    if(array_key_exists($planid, $availableForSalePlansIds)){
        $tempposition = $availableForSalePlansIds[$planid];
        $newValue = array($planid, $tempposition);
        if(!in_array($newValue, $incompatiblePlansIdsPosDictionary)){
            array_push($incompatiblePlansIdsPosDictionary,
$newValue);
        }
    }
}

if(count($incompatiblePlansIdsPosDictionary) ==
count($availableForSalePlansIds)){
    if(count($incompatiblePlansIdsPosDictionary) != 0){
        array_unshift($incompatiblePlansIdsPosDictionary, array(-1, -
1));
        print json_encode($incompatiblePlansIdsPosDictionary);
    }
    else{
        print "{}";
    }
}
else{
    print json_encode($incompatiblePlansIdsPosDictionary);
}

function sendRequest($methodName, $methodParameters) {
    global $PBAAPI;
    global $PBAAPIUserName;
    global $PBAAPIPassword;
    $methodParameters["Username"] = $PBAAPIUserName;
    $methodParameters["Password"] = $PBAAPIPassword;
    $request = xmlrpc_encode_request("Execute", $methodParameters);
    $header = "Content-Type: text/xml";
    $context = stream_context_create(array('http' => array(
        'method' => "POST",
        'header' => $header,
        'content' => $request
    )));
    $file = file_get_contents("https://$PBAAPI:5224/RPC2",false,
$content);
    $response = xmlrpc_decode($file);
    return $response;
}
?>

```

## .htaccess

```

Options +FollowSymLinks +ExecCGI
<IfModule mod_rewrite.c>
    RewriteEngine On
    RewriteBase /o365

```

```
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_URI} !=/favicon.ico
RewriteRule ^check(|/.*)$ check.php?q=$1 [L,QSA]
</IfModule>
```

## Configuring Offers for Resellers

The Microsoft Cloud Solution Provider program provides the following means of selling Office 365 services:

- 1-Tier CSP Partners sell services to end-users directly. 1-Tier CSP Partners have access to the Partner Center API, which allows them to automate provisioning and billing tasks.
- 2-Tier CSP Distributors sell services to end-users through 2-Tier CSP Resellers only. 2-Tier CSP Resellers sell services to end-users directly. 2-Tier CSP Distributors have access to the Partner Center API, which allows them to automate provisioning and billing tasks. A 2-Tier CSP Reseller can have the Authorized Subscription Advisor status in Microsoft, which allows the 2-Tier CSP Reseller to receive additional bonuses from selling services.

Odin Automation supports the following business scenarios:

- A 1-Tier CSP Partner provides Office 365 services by using either the provider account or a reseller account. End-users are registered as client accounts under the account of the 1-Tier CSP Partner.
- A 2-Tier CSP Distributor provides Office 365 services by using either the provider account or a reseller account.
  - A 2-Tier CSP Reseller provides Office 365 services by using a reseller account. The reseller account is registered under the account of the 2-Tier CSP Distributor. End-users are registered as client accounts under the reseller account of the 2-Tier CSP Reseller.

If the 2-Tier CSP Reseller has the Authorized Subscription Advisor status, Odin Automation can pass the MPN ID of the 2-Tier CSP Reseller to Microsoft when Office 365 subscriptions are created.
  - A 2-Tier CSP Reseller having the Authorized Subscription Advisor status provides Office 365 services by using middleware integrated with Odin Automation, but at the same time the 2-Tier CSP Reseller does not have any account in Odin Automation. End-users of the 2-Tier CSP Reseller are registered as client accounts under the account of the 2-Tier CSP Distributor.

Odin Automation can pass the MPN ID of the 2-Tier CSP Reseller to Microsoft when Office 365 subscriptions are created.

Use the instructions provided below to configure Odin Automation for the business scenario you need.

**Important:** The customer control panel of reseller accounts is not intended for using Office 365 services. It is solely intended for performing customer management operations, such as upgrading customer subscriptions and viewing reconciliation reports. If a reseller needs Office 365 services with the ability to manage them through the customer control panel, the reseller must create a separate customer account and subscribe it to necessary Office 365 service plans.

### **1-Tier CSP Partner / 2-Tier CSP Distributor Uses Provider Account in Odin Automation**

This business scenario is a base business scenario of the guide. See the [Cloud Solution Provider Scenario section](#) (page 11) for details.

### **1-Tier CSP Partner / 2-Tier CSP Distributor Uses Reseller Account in Odin Automation**

This business scenario is a variation of the '1-Tier CSP Partner / 2-Tier CSP Distributor Uses Provider Account in Odin Automation' business scenario. You must use the same configuration instructions as described in the [Cloud Solution Provider Scenario section](#) (page 11) with a few exceptions:

- If you already have an installed Office 365 application endpoint, you do not need to deploy a new one.
- A separate instance of the **Office 365** application must be created for the 1-Tier CSP Partner / 2-Tier CSP Distributor. Note that the credentials of the 1-Tier CSP Partner / 2-Tier CSP Distributor must be specified in the following settings of the instance:
  - **Administrator's login for connecting to Microsoft Online**
  - **Administrator's password for connecting to Microsoft Online**
  - **Reseller tenant ID**
  - **ID of Native Client for Partner Center SDK**
  - **ID of Native Client for Azure AD Graph API**
- Separate add-on types, license types, Office 365 resource types, Office 365 service templates, and Office 365 service plans must be created for the 1-Tier CSP Partner / 2-Tier CSP Distributor. We recommend that you use a prefix in the names.

Perform the following additional steps to complete configuring the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor:

- 1** Delegate the Office 365 service plans of the 1-Tier CSP Partner / 2-Tier CSP Distributor from the provider account to the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor. To do this, follow these steps:
  - a** Log in to Odin Automation using the credentials of the provider account.
  - b** Go to **Products > Service Plans**.
  - c** Select the **Service Templates** tab.

- d** Click on the reseller service template (**Service Name** is **Resellers Management**) of the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor.
  - e** Select the **Delegated Plans** tab.
  - f** Add the Office 365 service plans of the 1-Tier CSP Partner / 2-Tier CSP Distributor to the list of delegated service plans.
- 2** Configure the online store under the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor. See the [Configuring Online Store section](#) (page 81) for details.
- 3** Configure the Office 365 subdomain form under the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor. See the [Configuring Office 365 Subdomain Form for OA Billing section](#) (page 83) for details.
- 4** Configure the offer compatibility checker for the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor using the instructions of the [Configuring Offer Compatibility Checker section](#) (page 83):
  - a** In the `script2.js` file, add the identifiers of the Office 365 sales categories obtained from the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor to the list of Office 365 sales categories.

**Important:** Do not remove existing identifiers from the list.

- b** On the branding site of the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor, configure the offer compatibility checker using the identifiers of the Office 365 service plans obtained from the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor.
  - c** Configure the instance of the **Office 365** application which was created for the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor:
    - Specify the **Service Plans Compatibility Checker URL** global setting. Use the offer compatibility checker of the reseller branding site of the 1-Tier CSP Partner / 2-Tier CSP Distributor.
    - Specify the **Sale Category List** global setting. Use the identifiers of the Office 365 sales categories obtained from the reseller account of the 1-Tier CSP Partner / 2-Tier CSP Distributor.

## 2-Tier CSP Reseller Uses Reseller Account in Odin Automation

Perform the following actions to configure a reseller account for a 2-Tier CSP Reseller:

- 1** Delegate the Office 365 service plans from the account of the required 2-Tier CSP Distributor to the reseller account of the 2-Tier CSP Reseller:
  - a** Log in to Odin Automation using the credentials of the account of the 2-Tier CSP Distributor.
  - b** Go to **Products > Service Plans**.
  - c** Select the **Service Templates** tab.



- c** Specify the following parameters:
  - **ID** – Use the **MPN\_ID** value.
  - **Owner** – Use the **1** value (the owner is the provider account).
  - **Name** – Use the **MPN\_ID** value.
  - **Value Type** – Use the **Integer** value.
  - **Status** – Use the **Optional** value.
  - **Applicable to** – Use the **Accounts** value.
  - **Active** – Use the **Yes** value (the checkbox is selected).

Leave the default values for other parameters.

- d** Click **Save**.

- 3** Specify the **MPN\_ID** attribute for the reseller account of the 2-Tier CSP Reseller:

- a** Go to **Operations > Resellers**.
- b** Select the reseller account of the 2-Tier CSP Reseller.
- c** Select the **Additional Information** tab and click **Edit**.
- d** Specify the corresponding value of the **MPN\_ID** attribute and click **Save**.

### **2-Tier CSP Reseller Does Not Have Account in Odin Automation. 2-Tier CSP Reseller is Authorized Subscription Advisor.**

Odin Automation can pass the MPN ID of such a 2-Tier CSP Reseller to Microsoft if the MPN ID is specified in the **MPN\_ID** attribute of client accounts. There are two options to enable this:

**The 2-Tier CSP Distributor of the 2-Tier CSP Reseller uses the provider account.** In this case you must perform the following actions:

- 1** Log in to Odin Automation using the credentials of the provider account.
- 2** Create the **MPN\_ID** attribute in OA Billing:
  - a** Go to **System > Settings** and click **Attributes** in the **Operations** group.
  - b** Click **Add New Attribute**.
  - c** Specify the following parameters:
    - **ID** – Use the **MPN\_ID** value.
    - **Owner** – Use the **1** value (the owner is the provider account).
    - **Name** – Use the **MPN\_ID** value.
    - **Value Type** – Use the **Integer** value.
    - **Status** – Use the **Optional** value.
    - **Applicable to** – Use the **Accounts** value.



- **Active** – Use the **Yes** value (the checkbox is selected).

Leave the default values for other parameters.

- d** Click **Save**.

**The 2-Tier CSP Distributor of the 2-Tier CSP Reseller uses a reseller account.** In this case you must perform the following actions:

- 1** Log in to Odin Automation using the credentials of the provider account.
- 2** Create the **MPN\_ID** attribute in OA Billing:
  - a** Go to **System > Settings** and click **Attributes** in the **Operations** group.
  - b** Click **Add New Attribute**.
  - c** Specify the following parameters:
    - **ID** – Use the **MPN\_ID** value.
    - **Owner** – Use the **1** value (the owner is the provider account).
    - **Name** – Use the **MPN\_ID** value.
    - **Value Type** – Use the **Integer** value.
    - **Status** – Use the **Optional** value.
    - **Applicable to** – Use the **Accounts** value.
    - **Active** – Use the **Yes** value (the checkbox is selected).Leave the default values for other parameters.
  - d** Click **Save**.
- 3** Set the **MPN\_ID** attribute of the reseller account of the 2-Tier CSP Distributor to **-1**:
  - a** Go to **Operations > Resellers**.
  - b** Select the reseller account of the 2-Tier CSP Distributor.
  - c** Select the **Additional Information** tab and click **Edit**.
  - d** Specify the corresponding value of the **MPN\_ID** attribute and click **Save**.

## Firewall Requirements

This section provides the firewall requirements for the **Office 365** application.

### Office 365 Application Endpoint Host

#### Internal Outgoing Connections

| From  | To   | Protocol | Port   |
|---|--|----------|--|
| Office 365 Application Endpoint Host ( <b>O365AEH</b> ) – BackNet NIC | DNS Resolver Server – BackNet NIC  | TCP, UDP | 53   |
|   | SMTP server – BackNet NIC  | TCP      | 25 (for sending welcome messages)                              |
|   | Web/FTP server – BackNet NIC   | TCP      | 80; 21, 1024-65535 (for downloading welcome message templates) |
|   | OA Operations Management Node on Linux ( <b>LINMN</b> ) – BackNet NIC                    | TCP      | 6308 (APSC)<br>8440 (OA Operations Public API)                 |
|   | OA Billing Application Server on Linux ( <b>OABLINF</b> ) – BackNet NIC                  | TCP      | 5224 (OA Billing API)  |
|   | SQL Server – BackNet NIC   | TCP      | 1433   |
|   | OA Operations Branding Server on Linux ( <b>LINUIBR/WSNG/HEWSL/HELB</b> ) – FrontNet NIC | TCP      | 443 (access to offer compatibility checker)                    |

### Internal Incoming Connections

| From  | To  | Protocol | Port |
|---|---|----------|------|
| OA Operations Management Node on Linux ( <b>LINMN</b> ) – BackNet NIC | Office 365 Application Endpoint Host ( <b>O365AEH</b> ) – BackNet NIC | TCP      | 443  |

### External Outgoing Connections

| From  | To   | Protocol | Port |
|---|--|----------|------|
| Office 365 Application Endpoint Host ( <b>O365AEH</b> ) – BackNet NIC | <p><b>Partner Center / Graph API (Microsoft Cloud):</b></p> <ul style="list-style-type: none"> <li>login.microsoftonline.com</li> <li>graph.windows.net</li> <li>login.windows.net</li> <li>partnerapi.store.microsoft.com</li> <li>api.cp.microsoft.com</li> <li>api.partnercenter.microsoft.com</li> </ul> <p><b>Partner Center / Graph API (Microsoft Cloud Germany)</b></p> <ul style="list-style-type: none"> <li>api.partnercenter.microsoft.com</li> <li>login.microsoftonline.de</li> <li>graph.cloudapi.de</li> </ul> <p><b>Partner Center / Graph API (Office 365 operated by 21Vianet in China)</b></p> <ul style="list-style-type: none"> <li>partner.partnercenterapi.microsoftonline.cn</li> <li>login.chinacloudapi.cn</li> <li>graph.chinacloudapi.cn</li> </ul> | TCP      | 443  |

**Note:** This section contains firewall rules that are related to the **Office 365** application. General rules are provided in the **Odin Automation Firewall Configuration Guide**.

## OA Operations Management Node

### Internal Outgoing Connections

| From  | To  | Protocol | Port |
|---|---|----------|------|
| OA Operations Management Node on Linux ( <b>LINMN</b> ) – BackNet NIC | Office 365 Application Endpoint Host ( <b>O365AEH</b> ) – BackNet NIC   | TCP      | 443  |
|   | OA Billing Application Server on Linux ( <b>OABLINF</b> ) – BackNet NIC | TCP      | 5643 |

### Internal Incoming Connections

| From  | To  | Protocol | Port   |
|---|---|----------|--|
| Office 365 Application Endpoint Host ( <b>O365AEH</b> ) – BackNet NIC | OA Operations Management Node on Linux ( <b>LINMN</b> ) – BackNet NIC | TCP      | 6308 (APSC)<br>8440 (OA Operations Public API) |

**Note:** This section contains firewall rules that are related to the **Office 365** application. General rules are provided in the **Odin Automation Firewall Configuration Guide**.

## OA Billing Application Server

### Internal Incoming Connections

| From   | To   | Protocol | Port                  |
|--|--|----------|-----------------------|
| OA Operations Management Node on Linux ( <b>LINMN</b> ) – BackNet NIC              | OA Billing Application Server on Linux ( <b>OABLINFE</b> ) – BackNet NIC | TCP      | 5643                  |
| OA Operations Branding Server on Linux ( <b>LINUIBR/WSNG/HEWSL</b> ) – BackNet NIC |  | TCP      | 5224 (OA Billing API) |
| Office 365 Application Endpoint Host ( <b>O365AEH</b> ) – BackNet NIC              |  | TCP      | 5224 (OA Billing API) |

**Note:** This section contains firewall rules that are related to the **Office 365** application. General rules are provided in the **Odin Automation Firewall Configuration Guide**.

## OA Operations Branding Server

### Internal Outgoing Connections

| From   | To   | Protocol | Port                  |
|--|--|----------|-----------------------|
| OA Operations Branding Server on Linux ( <b>LINUIBR/WSNG/HEWSL</b> ) – BackNet NIC | OA Billing Application Server on Linux ( <b>OABLINFE</b> ) – BackNet NIC | TCP      | 5224 (OA Billing API) |

### Internal Incoming Connections

| From  | To   | Protocol | Port  |
|---|--|----------|---|
| Office 365 Application Endpoint Host ( <b>O365AEH</b> ) – BackNet NIC | OA Operations Branding Server on Linux ( <b>LINUIBR/WSNG/HEWSL/HELB</b> ) – FrontNet NIC | TCP      | 443 (access to offer compatibility checker) |

**Note:** This section contains firewall rules that are related to the **Office 365** application. General rules are provided in the **Odin Automation Firewall Configuration Guide**.

## Importing Tenant Subscriptions from Microsoft Cloud into Odin Automation

This section describes how to import tenant subscriptions from the Microsoft cloud into Odin Automation. After importing tenant subscriptions, these subscriptions become manageable through Odin Automation.

Importing is performed by using scripts shipped with the **Office 365** application. The import procedure consists of the following steps:

- 1 Satisfying preconditions. See [Preconditions](#) (page 101) for details.
- 2 Collecting tenant data necessary to import tenant subscriptions. See [Collecting Tenant Data](#) (page 103) for details.
- 3 Importing tenant subscriptions. See [Importing Tenant Subscriptions](#) (page 106) for details.
- 4 Validation of imported tenant subscriptions. See [Validation of Imported Tenant Subscriptions](#) (page 108) for details.

### Preconditions

Before importing tenant subscriptions, make sure the following preconditions are satisfied on your Odin Automation installation:

- The host where you are going to run the import scripts meets the provided below requirements. You can use the OA Operations Management Node, but if the version of Python is less than **2.6.6**, use another host.
  - Python 2.6.6 is installed on the host.
  - The poaupdater RPM package is installed on the host. The RPM package must be taken from the Odin Automation distribution package/YUM repository that belongs to your version of Odin Automation (for example, see the `DISTRIBUTION_PACKAGE/modules/platform/OS/OS_VERSION/RPMS/` directory).

- The pyPgSQL-2.5.1 (or higher) Python package is installed on the host.
- The mx 3.1.1 (or higher) Python package is installed on the host.
- Connections are allowed from the host to the OA Operations Public API.
- Connections are allowed from the host to the OA Billing Public API.
- Connections are allowed from the host to the APSC.
- Connections are allowed from the host to the Odin Automation system database.
- The `readCSPAccounts.py` and `importSubsCSP.py` import scripts are placed on the host. The import scripts can be obtained here: `Office365-x.x-x.app.zip\0365-Import.zip`.
- Connections from the Office 365 Application Endpoint Host to the OA Operations Public API are allowed.
- The **Office 365** application is configured for the partner account whose tenants will be processed by the import scripts.
- The identifier of the required application instance of the **Office 365** application is prepared. To obtain the identifier, perform the following actions:
  - a In PCP, go to **Services > Applications**.
  - b Click on the **Office 365** application.
  - c Select the **Instances** tab.
  - d Use the **ID** column.
- The identifier of the required provider/reseller account is prepared.
- The corresponding customer account is created for each tenant. You can create customer accounts by using any suitable method. For instance, you can use the Odin Automation Public API or the Odin Automation control panel.
- Mappings between the customer accounts and their tenants are prepared.

## Limitations

The import scripts have the following limitations:

- For an Office 365 CSP subscription created with the `importSubsCSP.py` script, all resource downgrade orders placed within the current billing period will contain zero resource prices and will not include order details. As a result, the customer placing the orders will not receive any refunds. When the current billing period ends and the first billing order is automatically placed for the Office 365 subscription, further resource downgrade orders will work as expected.

To calculate a refund for a resource downgrade order of a subscription, Odin Automation uses orders placed earlier. Since the script creates an Office 365 subscription without placing any order, Odin Automation does not have the necessary information to calculate the refund for a resource downgrade order of the Office 365 subscription until the first billing order is automatically placed, which will happen when the current billing period ends.

- There are two groups of offers in the Microsoft cloud: offers that support only monthly billing frequency; offers that support both monthly billing frequency and yearly billing frequency. The import scripts do not take into account billing frequencies of cloud subscriptions and billing periods of Office 365 service plans when cloud subscriptions are mapped to Office 365 service plans.

We recommend that you plan the import of tenant subscriptions according to these specific circumstances.

## Collecting Tenant Data

Before importing tenant subscriptions, tenant data must be collected from the Microsoft cloud. It is done by using the `readCSPAccounts.py` script. The script retrieves information about tenants, such as tenant properties and lists of tenant subscriptions, tries to find the corresponding customer accounts for these tenants, and tries to find suitable Office 365 service plans for tenant subscriptions.

The `readCSPAccounts.py` script has the following input parameters:

- **csp-app-instance-id** – The prepared identifier of the required application instance.
- **vendor-id** – The prepared identifier of the required vendor account.
- **tenant-id-attribute-name** – The identifier of a custom attribute in OA Billing. This parameter is optional. Specify this input parameter if the prepared customer accounts have a custom attribute and the corresponding tenant identifiers are specified in the custom attribute. This will allow the `readCSPAccounts.py` script to find the corresponding customer account for each tenant. Do not specify the input parameter if the prepared customer accounts do not have such a custom attribute. In this case you will need to use another approach to find the corresponding customer account for each tenant.

The `readCSPAccounts.py` script produces the following output files:

- `customers_PARTNER_SUBDOMAIN.csv/customers_PARTNER_TENANT_ID.csv` – This file contains the list of tenants and their properties. Each tenant is represented in the following way:
  - **TenantId** – The identifier of the tenant in the Microsoft cloud.
  - **CspCustomerId** – The CSP customer identifier of the tenant in the Microsoft cloud.
  - **DefaultDomainName** – The default domain of the tenant.
  - **DisplayName** – The display name of the tenant.

- **OSACustomerId** – The identifier of the corresponding customer account in Odin Automation. It is set to **-1** if the customer account cannot be found.
  - **company\_name** – The name of the company of the tenant.
  - **culture** – The default locale of the tenant.
  - **address\_line1** – The first line of the address of the tenant.
  - **address\_line2** – The second line of the address of the tenant.
  - **address\_line3** – The third line of the address of the tenant.
  - **city** – The city of the tenant.
  - **country** – The country of the tenant.
  - **first\_name** – The first name of the tenant.
  - **last\_name** – The last name of the tenant.
  - **phone\_number** – The phone number of the tenant.
  - **postal\_code** – The postal code of the tenant.
  - **region** – The region of the tenant.
  - **email** – The email address of the tenant.
  - **language** – The language of the tenant.
  - **type** – The type of the tenant in the Microsoft cloud.
- `importSubs_PARTNER_SUBDOMAIN.csv/importSubs_PARTNER_TENANT_ID.csv` – This file contains tenant subscriptions which will be imported. This file will be used by the `importSubsCSP.py` script to perform importing. Each list of tenant subscriptions is represented in the following way:
    - **OSACustomerID** – The identifier of the corresponding customer account in Odin Automation. It is set to **-1** if the customer account cannot be found.
    - **OSAServicePlanId** – The identifier of the target Office 365 service plan which will be used to create an Office 365 subscription in Odin Automation for the tenant subscriptions listed in **MSSubscriptionIDs**.
    - **OSAPlanPeriodID** – The identifier of the target subscription period of the chosen Office 365 service plan.
    - **SubscriptionStartDate** – The start date of the Office 365 subscription which will be created in Odin Automation.
    - **ResourceRateLimits** – The list of resource rates and their limits. The format "**RR\_ID1-LIMIT,RR\_ID2-LIMIT**" is used for the list. For example: "**100-10,200-20**".
    - **CSPAccountId** – The CSP customer identifier of the tenant in the Microsoft cloud.
    - **MSSubscriptionIDs** – The list of tenant subscriptions. The format "**MS\_SUB\_ID1,MS\_SUB\_ID2,...**" is used for the list. For example: "**A49C6781-1557-42CA-A074-52D0553397EA,EBE5106C-B1ED-44FF-8CE7-A2746AF8BC42**".



- **DefaultAdminLogin** – The login of the default administrative user of the tenant.

**Notes:**

1. There might be several rows in the `importSubs_PARTNER_SUBDOMAIN.csv/importSubs_PARTNER_TENANT_ID.csv` file for different lists of tenant subscriptions of a tenant because one list of tenant subscriptions might belong to one Office 365 service plan and another list of tenant subscriptions might belong to another Office 365 service plan.

2. See [Creating 'Office 365' Service Plans](#) (page 77) to obtain information about the structure of Office 365 service plans.

3. Office 365 service plans with composite resources are supported by the `readCSPAccounts.py` script. If each composite resource of an Office 365 service plan contains one included resource and the amount of the included resource is equal to **1**, the Office 365 service plan is taken into account by the script to find suitable Office 365 service plans for tenant subscriptions.

4. Both active tenant subscriptions and suspended tenant subscriptions are taken into account by the `readCSPAccounts.py` script.

- `errorSubs_PARTNER_SUBDOMAIN.csv/errorSubs_PARTNER_TENANT_ID.csv` – This file contains tenant subscriptions which cannot be imported due to some reasons. Each list of tenant subscriptions is represented in the following way:
  - **OSACustomerID** – The identifier of the corresponding customer account in Odin Automation. It is set to **-1** if the customer account cannot be found.
  - **CSPAccountId** – The CSP customer identifier of the tenant in the Microsoft cloud.
  - **SubscriptionStartDate** – The start date of the Office 365 subscription which will be created in Odin Automation.
  - **MSSubscriptionIDs** – The list of tenant subscriptions.
  - **MSSubscriptionOfferIDs** – The list of tenant subscription offers.
  - **Comments** – Comments containing information which might be used for troubleshooting.

Analyse the

`errorSubs_PARTNER_SUBDOMAIN.csv/errorSubs_PARTNER_TENANT_ID.csv` file and fix problems. After doing this, you can run the `readCSPAccounts.py` script again.

- `readCSPAccountsYYYY-MM-DD.log` - The log file of the `readCSPAccounts.py` script.

**Notes:**

1. If the `readCSPAccounts.py` script fails with the error `The identity of the calling application could not be established`, use the KB article <https://kb.odin.com/en/128179> to resolve this problem.
2. To import tenant subscriptions of a specific tenant, use **--account-default-domain CUSTOMER\_TENANT\_SUBDOMAIN** with the `readCSPAccounts.py` script and then use the produced output file with the `importSubsCSP.py` script.

After running the `readCSPAccounts.py` script, make sure all tenant subscriptions are processed correctly:

- 1 Check that each tenant belongs to the corresponding customer account.
- 2 Check that each list of tenant subscriptions belongs to the appropriate Office 365 service plan; the correct resource rates and limits are specified for each list of tenant subscriptions; the billing frequencies of the tenant subscriptions belong to the billing periods of their Office 365 service plans. If required, update the `importSubs_PARTNER_SUBDOMAIN.csv/importSubs_PARTNER_TENANT_ID.csv` file.

## Importing Tenant Subscriptions

After collecting tenant information and preparing the final file `importSubs_PARTNER_SUBDOMAIN.csv/importSubs_PARTNER_TENANT_ID.csv`, you can run the `importSubsCSP.py` script which imports tenant subscriptions from the Microsoft cloud into Odin Automation. The `importSubsCSP.py` script imports tenant subscriptions specified in the file `importSubs_PARTNER_SUBDOMAIN.csv/importSubs_PARTNER_TENANT_ID.csv`.

The `importSubsCSP.py` script has the following input parameters:

- **csp-app-instance-id** – The prepared identifier of the required application instance.
- The name of the file you prepared earlier (`importSubs_PARTNER_SUBDOMAIN.csv/importSubs_PARTNER_TENANT_ID.csv`).

The `importSubsCSP.py` script creates the following files:

- `importCSPYYYY-MM-DD.log` – This file contains general messages, such as information messages and error messages.
- `errors_import_<input file name>` – This CSV file contains a list of tenant subscriptions which cannot be processed by the `importSubsCSP.py` script as well as reasons why the tenant subscriptions cannot be processed. The CSV file has the same format as the input file of the `importSubsCSP.py` script (except for the additional **Error** column, which contains error messages).

- `reimport_<input file name>` – This CSV file contains a list of tenant subscriptions which cannot be processed by the `importSubsCSP.py` script. The CSV file has the same format as the input file of the `importSubsCSP.py` script. You can use the file `reimport_<input file name>` to run the `importSubsCSP.py` script again only for affected tenant subscriptions.
- `result_<input file name>` – Each row of the CSV file contains the following data of a subscription created by the `importSubsCSP.py` script in Odin Automation:
  - **CustomerId** – The identifier of the customer account to which the subscription belongs.
  - **SubscriptionId** – The Odin Automation identifier of the newly created Office 365 CSP subscription.

To run the `importSubsCSP.py` script, perform the following actions:

- 1 Stop Office 365 synchronization:
  - a In Provider Control Panel, go to **Operations > Tasks**.
  - b Select the **Periodic** tab.
  - c Cancel all the **Office 365 \* Synchronization with Office 365 Portal** tasks.
- 2 Log in to the host where the import scripts are placed.
- 3 Run the `importSubsCSP.py` script. For example: `importSubsCSP.py --csp-app-instance-id 1 importSubs_example.onmicrosoft.com.csv`
- 4 After running the script, Office 365 import tasks will be scheduled in the task manager. Wait until they are processed.
- 5 Start Office 365 synchronization:
  - a In Provider Control Panel, go to **Operations > Tasks**.
  - b Select the **Periodic** tab.
  - c Run all the **Office 365 \* Synchronization with Office 365 Portal** tasks.

**Notes:**

1. To import tenant subscriptions of a specific tenant, use **--account-default-domain CUSTOMER\_TENANT\_SUBDOMAIN** with the `readCSPAccounts.py` script and then use the produced output file with the `importSubsCSP.py` script.
2. After importing all the tenant subscriptions of a customer tenant, do not use the Partner Center to manage the tenant subscriptions. All subscription management operations, such as adding new tenant subscriptions, changing tenant subscription limits, removing existing tenant subscriptions, must be performed through Odin Automation for the customer tenant.
3. When tenant subscriptions of a customer tenant are imported, the `importSubsCSP.py` script always creates new Office 365 subscriptions in Odin Automation. It means the script does not update limits of existing Office 365 subscriptions of the corresponding customer account in Odin Automation even though the Office 365 subscriptions contain a suitable set of resources to import the tenant subscriptions.
4. If the `importSubsCSP.py` script fails with the `A duplicated entry is found for the SA Customer #CustomerId and Service Plan #PlanId in the line LineNumber` error, consider using the **--allow-duplicates** option described at <https://kb.odin.com/en/128798>.
5. The **Office 365 \* Synchronization with Office 365 Portal** tasks facilitate tenant subscription import (for instance, users are synchronized by the tasks). To learn more about what the tasks synchronize and synchronization limitations, refer to the [Synchronizing Changes from Office 365 Portal section](#) (page 114).

## Validation of Imported Tenant Subscriptions

After importing tenant subscriptions, perform the following validation steps for each customer tenant:

- 1 Make sure the customer account of the customer tenant is subscribed to the required Office 365 service plans.
- 2 Make sure the limits of the Office 365 subscriptions of the customer account match the limits of the tenant subscriptions.
- 3 Make sure the billing frequencies of the tenant subscriptions belong to the billing periods of the Office 365 service plans.

## Example

This section shows how to import the tenant subscriptions of a customer tenant.

### Assumptions

- The customer tenant has the following tenant subscriptions in the Microsoft cloud:
  - **Office 365 Business Premium** (5 licenses)

- **Exchange Online Advanced Threat Protection** (5 licenses)
- **Exchange Online Archiving for Exchange Online** (5 licenses)
- **Office 365 Extra File Storage** (10 gigabytes)

All the tenant subscriptions have monthly billing frequency.

- The customer tenant does not have any customer account in Odin Automation.
- The identifier of the required Office 365 application instance is **10**.
- The Office 365 application instance and its Office 365 service plans belongs to the provider.
- The standard Office 365 service plan **Office 365 Business Premium** exists under the provider account. Its identifier is **100**. The service plan includes the following resource rates:
  - **Office 365 Business Premium** (the resource rate identifier is **1001**)
  - **Exchange Online Advanced Threat Protection** (the resource rate identifier is **1002**)
  - **Exchange Online Archiving for Exchange Online** (the resource rate identifier is **1003**)
  - **Office 365 Extra File Storage** (the resource rate identifier is **1004**)

**Billing Period** of the Office 365 service plan is set to **Monthly**.

- The customer tenant subdomain is **customer.onmicrosoft.com**.
- The partner subdomain is **provider.onmicrosoft.com**.

### Import Procedure

- 1 Execute the `readCSPAccounts.py --vendor-id 1 --csp-app-instance-id 10 --account-default-domain customer.onmicrosoft.com` command. The `customers_provider.onmicrosoft.com.csv` and `importSubs_provider.onmicrosoft.com.csv` files will be created.
- 2 Open the `importSubs_provider.onmicrosoft.com.csv` file and check the following:
  - The **OSAServicePlanId** field contains the identifier of the **Office 365 Business Premium** service plan.
  - The **ResourceRateLimits** field contains the correct resource rate identifiers and limits (**1001-5,1002-5,1003-5,1004-10**).
- 3 In Odin Automation under the provider account, create a customer account using the information from the `customers_provider.onmicrosoft.com.csv` file. After creating the account, write down its identifier.
- 4 Specify the identifier of the created customer account in the `importSubs_provider.onmicrosoft.com.csv` file in the **OSACustomerID** field.
- 5 In Provider Control Panel, stop Office 365 synchronization by canceling all the **Office 365 \* Synchronization with Office 365 Portal** tasks.

- 6 Execute the `importSubsCSP.py --csp-app-instance-id 10 importSubs_provider.onmicrosoft.com.csv` command. The customer account will be subscribed to the **Office 365 Business Premium** service plan.
- 7 After running the script, Office 365 import tasks will be scheduled in the task manager. Wait until they are processed.
- 8 In Provider Control Panel, start Office 365 synchronization by running all the **Office 365 \* Synchronization with Office 365 Portal** tasks.
- 9 Validate the limits of the subscription in Odin Automation. The limits of the **Office 365 Business Premium, Exchange Online Advanced Threat Protection, Exchange Online Archiving for Exchange Online, and Office 365 Extra File Storage** offer resources must be set to **5, 5, 5, and 10**. The limits of the other offer resources must be set to **0**.

## Useful Information

This section contains additional information that may be useful for providers.

### How to Upgrade Office 365 Subscriptions through Provider/Reseller Control Panel

This section describes how to upgrade Office 365 subscriptions from one service plan to another service plan using Provider/Reseller Control Panel.

To provide a reseller with the ability to perform upgrades from Reseller Control Panel, the reseller must be subscribed to the **Office 365 Customer Management** service plan. This service plan is created in the following way:

- It is based on the same **Office 365** service template which is used by Office 365 service plans delegated to the reseller.
- Zero prices are set in subscription periods of the service plan.
- It does not contain any resource rates.

Before upgrading subscriptions, pay attention to the following requirements:

- Each subscription must contain one 'License' resource with a non-zero limit. If a subscription includes additional 'License' and 'Add-on' resources, their limits must be equal to zero.
- The target service plan must contain one 'License' resource.

If the requirements are not satisfied for any of the subscriptions, the upgrade operation will not be allowed. See the <https://kb.odin.com/en/129264> KB article to obtain the instruction on how to process such subscriptions.

To upgrade subscriptions of a service plan, perform the following actions:

- 1 Log in to Provider/Reseller Control Panel.
- 2 In Provider Control Panel:
  - Go to **Services > Applications** and click the **Office 365** package.
  - Select the **Instances** tab and select the required application instance.
 In Reseller Control Panel:
  - Click **Customer CP**.
  - Select the **Office 365** tab.
- 3 Select the **Service Plans** tab.
- 4 Click on the name of the service plan you want to upgrade the subscriptions from.
- 5 Select the subscriptions you want to upgrade and click **Upgrade Subscriptions**.
- 6 Select the target service plan you want to upgrade the subscriptions to and click **Finish**.

The upgrade operations will be scheduled for the subscriptions. To view the progress of the upgrade operations, use **Refresh Statuses**.

You can cancel scheduled upgrade operations for subscriptions where **Upgrade Status** is **Pending**. To do this, select the required subscriptions and click **Cancel Upgrade**.

**Note:** Upgrade operations are performed in parallel for several subscriptions. The maximum number of simultaneous upgrade operations is **10**. You can change this number in the **MAX\_ALLOWED\_NB\_OF\_SIMULTANEOUS\_UPGRADES** setting of the `Settings.config` configuration file of the corresponding Office 365 gateway application. Note that after changing the configuration file, the application pool of the Office 365 gateway application must be restarted.

## Viewing Reconciliation Report

As a member of the Microsoft Cloud Solution Provider (CSP) program, you are regularly charged for selling Office 365 services. As a service provider, you charge your customers for using Office 365 services. To reconcile your charges with the charges of your customers, you can use reconciliation reports generated by Odin Automation.

A reconciliation report, which is generated for a specified period, contains information collected from completed orders of Office 365 subscriptions. Each item of a reconciliation report belongs to an order detail and contains the following columns:

- **OperatingUnit** – Reserved for future use. Ignore this column.
- **CustomerTenantId** – The identifier of the corresponding customer account in the Microsoft Partner Center.
- **CustomerName** – The name of the corresponding customer account in the Microsoft Partner Center.
- **MpnId** – The value of the **MPN\_ID** attribute. See the [Configuring Offers for Resellers section](#) (page 92) for details.

- **OrderId** – The identifier of the corresponding order in Odin Automation.
- **SubscriptionId** – The identifier of the corresponding Office 365 subscription in Odin Automation.
- **SyndicationPartnerSubscriptionNumber** – The identifier of the corresponding subscription in the Microsoft Partner Center (if the order detail belongs to an Office 365 resource). The identifiers of the corresponding subscriptions in the Microsoft Partner Center (if the order detail belongs to an Office 365 composite resource or an Office 365 subscription).
- **OfferId** – The identifier of the corresponding Office 365 resource in Odin Automation (if the order detail belongs to an Office 365 resource). The identifiers of the corresponding Office 365 resources in Odin Automation (if the order detail belongs to an Office 365 composite resource or an Office 365 subscription).
- **DurableOfferId** – The identifier of the corresponding offer from your partner price list (if the order detail belongs to an Office 365 resource). The identifiers of the corresponding offers from your partner price list (if the order detail belongs to an Office 365 composite resource or an Office 365 subscription).
- **OfferName** – The name of the corresponding Office 365 resource in Odin Automation (if the order detail belongs to an Office 365 resource). The names of the corresponding Office 365 resources in Odin Automation (if the order detail belongs to an Office 365 composite resource or an Office 365 subscription).
- **SubscriptionStartDate** – The start date of the corresponding Office 365 subscription in Odin Automation.
- **SubscriptionEndDate** – The end date of the corresponding Office 365 subscription in Odin Automation.
- **ChargeStartDate** – The value of the **Detail Start Date** property of the order detail.
- **ChargeEndDate** – The value of the **Detail End Date** property of the order detail.
- **ChargeType** – The type of customer charges (**PURCHASE FEE** for a setup fee, **CYCLE FEE** for a recurring fee).
- **UnitPrice** – The value of the **Unit Price** property of the order detail.
- **Quantity** – The value of the **Quantity** property of the order detail.
- **Amount** – It is calculated in the following way: <the value of the **Unit Price** property of the order detail> \* <the value of the **Quantity** property of the order detail> \* <the value of the **Duration** property of the order detail>.
- **TotalOtherDiscount** – The value of the **Discount** property of the order detail.
- **Subtotal** – It is calculated in the following way: <the value of the **Unit Price** property of the order detail> \* <the value of the **Quantity** property of the order detail> \* <the value of the **Duration** property of the order detail> - <the value of the **Discount** property of the order detail>.



- **Tax** – The value of the **Tax Total** property of the corresponding order in Odin Automation. Note that if an order contains several order details, **Tax** is only shown for the item of the first order detail.
- **TotalForCustomer** – The value of the **Extended Price** property of the order detail.
- **Currency** – The type of currency.

A reconciliation report can also be generated under a reseller account. To provide a reseller account with this ability, the reseller account must be subscribed to the **Office 365 Customer Management** service plan. This service plan is created in the following way:

- It is based on the **Office 365** service template.
- Zero prices are set in subscription periods of the service plan.
- It does not contain any resource rates.

To view a reconciliation report, perform the following actions:

- 1 Log in to Provider/Reseller Control Panel.
- 2 In Provider Control Panel:
  - Go to **Services > Applications**, and then select the **Office 365** application.
  - Select the **Instances** tab, and then select the application instance to generate the report for.In Reseller Control Panel
  - Click **Customer CP**.
  - Select the **Office 365** tab.
- 3 Select the **Reconciliation Report** tab.
- 4 Specify a period by selecting **Start Date** and **End Date**, and then click **Export Data**.

## Viewing Business Reports

The **Office 365** application provides graphical reports which you can use to monitor your business and track critical business numbers. Using these reports, you can:

- View active Office 365 subscriptions by customer business size;
- View active licenses by customer business size;
- View active Office 365 subscriptions by country;
- View active licenses by country;
- View how the total number of Office 365 subscriptions, the total number of sold licenses, the total number of active licenses changed over time for various countries;
- Export the data of a report to Excel;

To view a report, perform the following actions:

- 1 Log in to Provider Control Panel.
- 2 Go to **Services > Applications**, and then select the **Office 365** application.
- 3 Select the **Instances** tab, and then select the application instance to generate the report for.
- 4 Select the **Statistics** tab.

## Synchronizing Changes from Office 365 Portal

Changes which are directly made by customers in the Office 365 Portal are automatically synchronized from the portal to Odin Automation. Synchronization is performed by the periodic task **Synchronization with Office 365 Portal**, which is executed every 15 minutes.

Note the following:

- The number of the tasks depends on how many application instances exist on your installation.
- Customers can manually start synchronization from Customer Control Panel.
- Phone numbers of users are not synchronized.
- Synchronization ignores licenses that belong to cloud subscriptions obtained outside of Odin Automation. To make such cloud subscriptions manageable through Odin Automation, you can use one of these methods: cancel the cloud subscriptions in the Microsoft cloud and create necessary cloud subscriptions by subscribing the customer accounts of the customer tenants to Office 365 subscriptions through Odin Automation; use the import scripts to import the cloud subscriptions into Odin Automation (see [Importing Tenant Subscriptions from Microsoft Cloud into Odin Automation](#) (page 101) for details).

The following changes are synchronized from the portal and reflected in Odin Automation:

- A new user is added, modified, or removed.
- A license is assigned or unassigned.
- A domain is added or removed.

To monitor the status of synchronization, perform the following actions:

- 1 Log in to Provider Control Panel.
- 2 Go to **Operations > Tasks**, and then select the **Periodic** tab.
- 3 Click the task **Synchronization with Office 365 Portal**.
- 4 See task logs in the **Last task execution results** area.

Note that if a problem appears during synchronization, the task does not fail. The detailed information about the problem can be found in task logs.

## How to Update Expired SSL Certificates on Office 365 Application Endpoint Host

This section describes how to update expired SSL certificates on the Office 365 Application Endpoint Host.

To update an expired Office 365 Gateway site SSL certificate, perform the following actions:

- 1 Log on to the Office 365 Application Endpoint Host as **Administrator**.
- 2 Prepare a new site SSL certificate with the same hostname, which is used in the expired site SSL certificate.
- 3 Place the new site SSL certificate on the host.
- 4 Import the new site SSL certificate.

Open the **Microsoft Management Console** (MMC) and add the **Certificates** snap-in:

- a In the top menu, go to **File > Add/Remove Snap-in**.
- b Select the **Certificates** snap-in and click **Add**.
- c Select the **Computer account** option and click **Next**.
- d Select the **Local computer** option and click **Finish**.
- e Click **OK**.

Go to the **Personal** folder and choose the **All Tasks > Import...** command in the right-click pop-up menu. Follow the *Certificate Import Wizard* instructions to import the new site SSL certificate.

- 5 Go to the **Personal > Certificates** folder and remove the expired site SSL certificate.
- 6 Go to the **Personal > Certificates** folder. Right-click the new site SSL certificate and choose the **All Tasks > Manage Private Keys...** command in the pop-up menu.
- 7 Add the IIS AppPool\`<Office 365 application endpoint site name in IIS Manager console>` Application Pool Identity Account to the list using the **Add...** button.
- 8 Select the added Application Pool Identity Account and grant the `Read` permission by selecting the respective checkbox in the permissions list.
- 9 Open the **IIS Manager** console.
- 10 Go to **Sites > <your\_site>**, click **Bindings** in the right area.
- 11 Select the **https** binding and click **Edit**.
- 12 Make sure the new site SSL certificate is selected in the **SSL certificate** field.
- 13 Click **OK**. Then, click **Close**.

## Creating Office 365 Subscription On Behalf Of Customer

In some cases, you might need to create an Office 365 subscription on behalf of a customer. For instance, you have convinced a customer to buy Office 365 licenses from you, and the customer has asked you to create an account with an Office 365 subscription for him or her.

To do this, you can use the following guidelines:

- 1** Collect the following customer data:
  - the company name of the customer
  - the address of the customer
  - the contact e-mail address of the customer
  - the contact phone of the customer
  - the licenses the customer wants to buy
  - the Office 365 subdomain the customer wants to use
- 2** Create the respective account and Office 365 subscription in Odin Automation:
  - a** In your control panel, click **Billing**.
  - b** Go to **Operations > Orders**, and then click **Create Account And Place Order**.
  - c** Follow the wizard.

As a result, an order is created. If the customer has provided you with his or her billing data, such as a credit card number and the credit card's expiration date, you are able to make the corresponding order paid. Otherwise, there must be a pre-configured payment method that allows the customer to make the order paid later from the control panel of Odin Automation.

**Note:** To obtain information about configuring payment methods, refer to **OA Billing Provider's Guide**.

## Registering New Customers with Existing Microsoft Accounts

The **Office 365** application supports the multichannel and multipartner features of the CSP program. This allows you to grow your customer base by registering new customers who already have Microsoft accounts with users, domains, and licenses.

There are two scenarios of registering such customers:

- 1** A customer already has a Microsoft account which was created when the customer bought licenses from another service provider. The customer wants to buy licenses from you and use the control panel of Odin Automation for managing licenses, users, and domains.

- 2 A customer already has a Microsoft account which was created to evaluate Office 365 services using trial licenses. The customer wants to buy the same non-trial licenses from you and use the control panel of Odin Automation for managing licenses, users, and domains.

Both scenarios require the customer to accept an invitation allowing you to manage the customer's Microsoft account through Odin Automation. The invitation must be accepted interactively on an invitation page of the Microsoft Office 365 Portal by an eligible user. Note that this user must have the **Global administrator** role to the Microsoft account.

After the invitation is accepted, the customer must pay for the licenses which he or she wants to buy. As a result, an order is created in Odin Automation. When the order is processed, the existing users and domains are imported from the Microsoft account to the control panel of Odin Automation.

In the first scenario, the licenses bought from another service provider are not imported, so the customer needs to assign his or her users the licenses bought from you using the control panel of Odin Automation. In the second scenario, the trial licenses are converted into the paid ones and the license assignments are kept and visible in the control panel of Odin Automation.

The registration procedure consists of the steps described below, which are performed by a customer in the control panel/online store of Odin Automation:

- 1 The customer chooses the licenses which he or she wants to buy from you.
- 2 The customer selects the **Use Existing Microsoft Account** option.
- 3 The customer specifies the login and the password of a user with the **Global administrator** role to the Microsoft account.
- 4 The customer confirms that the Microsoft account will be managed by you. It is done on the invitation page, which is opened in a new browser window after clicking the **Sign In to Office 365 Portal** button. The login and password of the same user must be used at this step.

**Important:** The browser of the customer must allow pop-up windows for the online store and control panel.

- 5 The customer pays for the licenses.

If you want to register a customer with an existing Microsoft account on behalf of the customer, you need to do the following:

- 1 Ask the customer to provide you with the list of the licenses he or she wants to buy.
- 2 Ask the customer to provide you with the initial subdomain of the Microsoft account.
- 3 Ask the customer to provide you with information which is necessary to register the customer in Odin Automation, such as the first name and last name, e-mail, phone, address of the customer.

- 4 Provide the customer with the URL of the invitation page. You can obtain the URL from the **Invitation URL** global setting of the corresponding application instance.
- 5 Ask the customer to open the page and accept the invitation by signing in under a user with the **Global administrator** role to the Microsoft account.
- 6 Log in to your control panel and create the customer account and Office 365 subscription using the **Use Existing Microsoft Account** option.

As a result, an order is created. The default order flow configured in Odin Automation defers processing the order until the payment is received, so the customer is supposed to log in to the control panel of Odin Automation and pay the order. Note that this order flow can be altered by the Odin Automation provider in many ways. Refer to the **OA Billing Provider's Guide** for details.

## Examples of Office 365 Service Plans

You can view examples of Office 365 service plans in the **.json** files listed below.

The following **.json** files are used to create the standard set of Office 365 service plans:

- `CustomLicenseTypes.json` contains descriptions of offers. The file is generated with the **CSP Deployment Assistant** tool (see [Configuring Offers](#) (page 62) for details).
- `CustomServicePlans.json` contains descriptions of Office 365 service plans. The file is generated with the **CSP Deployment Assistant** tool (see [Configuring Offers](#) (page 62) for details).
- `resourcecategories_CSP.json` contains descriptions of Office 365 resource categories. The file is shipped with the **Office 365** application. See [Configuring Offers](#) (page 62) to obtain the location of the file.
- `salescategories_CSP.json` contains descriptions of Office 365 sales categories. The file is shipped with the **Office 365** application. See [Configuring Offers](#) (page 62) to obtain the location of the file.

### CustomLicenseTypes.json File

For instance, the **Office 365 Business Essentials** offer is described in the following way:

```
{
  "possibleTransitions": [ // The list of possible transitions of the offer.
    "031c9e47-4802-4248-838e-778fbld2cc05", // The identifier of the
Office 365 Business Premium offer.
    "91fd106f-4b2c-4938-95ac-f54f74e9a239", // The identifier of the
Office 365 Enterprise E1 offer.
    "796b6b5f-613c-4e24-a17c-eba730d49c02", // The identifier of the
Office 365 Enterprise E3 offer.
    "8909e28e-5832-42f4-9886-b0a5545f3645" // The identifier of the
Office 365 Enterprise E4 offer.
  ],
  "trialOfferId": null, // The identifier of the trial offer of this offer.
If trialOfferId is set to null, it means this offer has no trial offer.
}
```

```
"ResourceCategory": "XL", // The internal name of the Additional Licenses
resource category this offer belongs to. Internal names are only used in the
.json files.
"description": "For businesses with 1 to 300 users that need anywhere
access to email, file sharing, and online conferencing without a subscription to
the desktop version of Office.", // The description of the offer.
"provisioning_id": "O365_BUSINESS_ESSENTIALS", // The provisioning
identifier of the offer.
"offerId": "bd938f12-058f-4927-bba3-ae36b1d2501c", // The identifier of
the offer.
"possibleConflicts": [ // The list of offers which are incompatible with this
offer.
    "35a36b80-270a-44bf-9290-00545d350866", // The identifier of
the Exchange Online Kiosk offer.
    "22b61e04-b7eb-4405-9cb3-6a4407b9f95a", // The identifier of
the Exchange Online Kiosk (Government Pricing) offer.
    "195416c1-3447-423a-b37b-ee59a99a19c4", // The identifier of
the Exchange Online Plan 1 offer.
    "24cc266d-6fd3-4b85-b9c8-4d5f587521ac", // The identifier of
the Exchange Online Plan 1 (Government Pricing) offer.
    "2f707c7c-2433-49a5-a437-9ca7cf40d3eb", // The identifier of
the Exchange Online Plan 2 offer.
    "ec0d51df-916f-4455-a096-5ab08f58dbd2", // The identifier of
the Exchange Online Plan 2 (Government Pricing) offer.
    "031c9e47-4802-4248-838e-778fb1d2cc05", // The identifier of
the Office 365 Business Premium offer.
    "91fd106f-4b2c-4938-95ac-f54f74e9a239", // The identifier of
the Office 365 Enterprise E1 offer.
    "cfc69058-5106-40bd-81e4-44e0e29034b2", // The identifier of
the Office 365 Enterprise E1 (Government Pricing) offer.
    "796b6b5f-613c-4e24-a17c-eba730d49c02", // The identifier of
the Office 365 Enterprise E3 offer.
    "6b551829-de8c-41e5-8678-41d52c27aee8", // The identifier of
the Office 365 Enterprise E3 (Government Pricing) offer.
    "8909e28e-5832-42f4-9886-b0a5545f3645", // The identifier of
the Office 365 Enterprise E4 offer.
    "379dc555-ddcb-4ccb-bebf-37437333e278", // The identifier of
the Office 365 Enterprise E4 (Government Pricing) offer.
    "4f7ecaf1-e9d6-4cac-9687-e22eb3dfdd70", // The identifier of
the Office 365 Enterprise E5 without PSTN Conferencing offer.
    "c4158aa7-00e7-4ce1-9cf3-3cf8321f377a", // The identifier of
the Office 365 Enterprise E5 without PSTN Conferencing (Government Pricing)
offer.
    "6fbad345-b7de-42a6-b6ab-79b363d0b371", // The identifier of
the Office 365 Enterprise K1 offer.
    "6celccc8-8b18-4e1b-alc4-f89de9904389", // The identifier of
the Office 365 Enterprise K1 (Government Pricing) offer.
    "ff7a4f5b-4973-4241-8c43-80f2be39311d", // The identifier of
the SharePoint Online Plan 1 offer.
    "69c67983-cf78-4102-83f6-3e5fd246864f", // The identifier of
the SharePoint Online Plan 2 offer.
    "aca0c06c-890d-4abb-83cf-bc519a2565e5", // The identifier of
the Skype for Business Online (Plan 1) offer.
    "aeld0798-28fa-4460-8c67-c09b3ac7133d", // The identifier of
the Skype for Business Online (Plan 1) (Government Pricing) offer.
    "14c61739-b45a-42c0-832c-d330972d3173", // The identifier of
the Skype for Business Online (Plan 2) offer.
    "a6c260a7-545c-42f7-bb27-461cle131534", // The identifier of
the Skype for Business Online (Plan 2) (Government Pricing) offer.
    "a3f4ab4e-6239-4ecb-a859-77369dcalc08", // The identifier of
the Yammer Enterprise offer.
```

```
        "8aa7e78b-b265-4ac6-ada0-14900a8a3f94" // The identifier of
the Yammer Enterprise (Government Pricing) offer.
    ],
    "points": 3, // Reserved for future use.
    "isAddon": false, // This field defines if this offer is an add-on offer
(true/false). If this field is not specified, it means this offer is not an add-
on offer.
    "Measure": "License", // The unit of measure of the offer (License/GB).
    "prices": [ // The list of recommended prices of the offer for various
currencies.
        {
            "currency": "USD",
            "price": 5.0
        },
        {
            "currency": "AUD",
            "price": 6.4
        },
        {
            "currency": "CAD",
            "price": 6.1
        },
        {
            "currency": "CHF",
            "price": 4.7
        },
        {
            "currency": "DKK",
            "price": 31.5
        },
        {
            "currency": "EUR",
            "price": 4.2
        },
        {
            "currency": "GBP",
            "price": 3.1
        },
        {
            "currency": "INR",
            "price": 300.4
        },
        {
            "currency": "JPY",
            "price": 540.0
        },
        {
            "currency": "KRW",
            "price": 5510.0
        },
        {
            "currency": "NOK",
            "price": 35.3
        },
        {
            "currency": "NZD",
            "price": 6.1
        },
        {
            "currency": "RUB",
            "price": 312.4
        }
    ]
}
```



```

    },
    {
        "currency": "SEK",
        "price": 39.4
    },
    {
        "currency": "TWD",
        "price": 159.8
    }
],
"isUserSpecific": true, // This field defines if the offer can be assigned
to users (true/false).
"name": "Office 365 Business Essentials" // The name of the offer.
}

```

The following license type is created on the basis of the description:

### Office 365 Business Essentials

- **License Name:** Office 365 Business Essentials
- **Offer ID:** bd938f12-058f-4927-bba3-ae36b1d2501c
- **Trial Offer ID:** null
- **Provisioning ID:** O365\_BUSINESS\_ESSENTIALS
- **Can be assigned to users:** Yes
- **Possible transitions:** Office 365 Business Premium, Office 365 Enterprise E1, Office 365 Enterprise E3, Office 365 Enterprise E4

Also, the following resource type/resource is created on the basis of the description:

### Office 365 Business Essentials

- License type instance: Office 365 Business Essentials

The created resource is added to the **Additional Licenses** resource category. The unit of measure of the resource is set to **License**.

For instance, the **Office 365 Extra File Storage** add-on offer is described in the following way:

```

{
    "possibleTransitions": [], // The list of possible transitions of the add-
on offer. It is empty for add-on offers.
    "trialOfferId": null, // The identifier of the trial offer of this add-on
offer. It is set to null for add-on offers.
    "possibleParents": // The list of parent offers of the add-on offer.
    [
        "bd938f12-058f-4927-bba3-ae36b1d2501c", // The identifier if Office
365 Business Essentials parent offer.
        "031c9e47-4802-4248-838e-778fb1d2cc05", // The identifier if Office
365 Business Premium parent offer.
        "91fd106f-4b2c-4938-95ac-f54f74e9a239", // The identifier if Office
365 Enterprise E1 parent offer.
        "796b6b5f-613c-4e24-a17c-eba730d49c02", // The identifier if Office
365 Enterprise E3 parent offer.
    ]
}

```

```

        "ff7a4f5b-4973-4241-8c43-80f2be39311d", // The identifier if
SharePoint Online Plan 1 parent offer.
        "69c67983-cf78-4102-83f6-3e5fd246864f" // The identifier if
SharePoint Online Plan 2 parent offer.
    ],
    "ResourceCategory": "XS", // The internal name of the Extra Storage
resource category this add-on offer belongs to.
    "description": "Priced per gigabyte, Office 365 offers additional file
storage options to support an organization's file growth.", // The description of
the add-on offer.
    "provisioning_id": "SHAREPOINTSTORAGE", // The provisioning identifier of
the add-on offer.
    "multiplier": 1048576, // The multiplier of the add-on offer (1048576 for
the 'GB' UOM / 1 for the 'License' UOM).
    "offerId": "53fc25f7-6639-4f78-bb44-3c2dfec3ed40", // The identifier of the
add-on offer.
    "possibleConflicts": [], // The list of offers which are incompatible with
this add-on offer.
    "points": 0, // Reserved for future use.
    "isAddon": true, // This field defines if this offer is an add-on offer
(true/false). If this field is not specified, it means this offer is not an add-
on offer.
    "Measure": "GB", // The unit of measure of the add-on offer (License/GB).
    "prices": [ // The list of recommended prices of the offer for various
currencies.
        {
            "currency": "USD",
            "price": 0.2
        },
        {
            "currency": "AUD",
            "price": 0.3
        },
        {
            "currency": "CAD",
            "price": 0.2
        },
        {
            "currency": "CHF",
            "price": 0.2
        },
        {
            "currency": "DKK",
            "price": 1.3
        },
        {
            "currency": "EUR",
            "price": 0.2
        },
        {
            "currency": "GBP",
            "price": 0.1
        },
        {
            "currency": "INR",
            "price": 12.0
        },
        {
            "currency": "JPY",
            "price": 20.0
        },
    ],

```

```

    {
      "currency": "KRW",
      "price": 220.0
    },
    {
      "currency": "NOK",
      "price": 1.4
    },
    {
      "currency": "NZD",
      "price": 0.3
    },
    {
      "currency": "RUB",
      "price": 12.5
    },
    {
      "currency": "SEK",
      "price": 1.6
    },
    {
      "currency": "TWD",
      "price": 6.4
    }
  ]
  "isUserSpecific": false, // This field defines if the add-on offer can be
assigned to users (true/false).
  "name": "Office 365 Extra File Storage" // The name of the add-on offer.
}

```

The following add-on type is created on the basis of the description:

### Office 365 Extra File Storage

- **Add-on Name:** Office 365 Extra File Storage
- **Offer ID:** 53fc25f7-6639-4f78-bb44-3c2dfec3ed40
- **Provisioning ID:** SHAREPOINTSTORAGE
- **Can be assigned to users:** Yes
- **Parent offers:** SharePoint Online Plan 1, SharePoint Online Plan 2, Office 365 Enterprise E1, Office 365 Enterprise E3, Office 365 Business Essentials, Office 365 Business Premium

Also, the following resource type/resource is created on the basis of the description:

### Office 365 Extra File Storage

- **Add-on type instance:** Office 365 Extra File Storage

The created resource is added to the **Extra Storage** resource category. The unit of measure of the resource is set to **GB**.

### CustomServicePlans.json File

For instance, the **Office 365 Business Essentials** service plan is described in the following way:

```
"BE": { // The internal name of the Office 365 Business Essentials service plan.
Internal names are only used in the .json files.
    "RecurringFee": 5.0, // The recurring fee of the subscription period
of the service plan.
    "SegmentGroup": "E", // The internal name of the group this service
plan belongs to. Internal names are only used in the .json files.
    "Name": "Office 365 Business Essentials", // The name of the service
plan.
    "IncompatibleSegmentGroups": [], // The list of groups containing
service plans which are incompatible with this service plan.
    "PeriodType": "Y", // The type of the subscription period of the
service plan (M - month/Y - year).
    "IncompatiblePlans": [ // The list of service plans which are
incompatible with this service plan.
        "BE", // The internal name of the Office 365 Business
Essentials service plan.
        "BE+X" // The internal name of the Office 365 Business
Essentials with Add-ons service plan.
    ],
    "Period": "1", // The duration of the subscription period of the
service plan.
    "UpgradeTo": [ // The allowed upgrade paths of the service plan.
        "BE+X", // The internal name of the Office 365 Business
Essentials with Add-ons service plan.
        "BP", // The internal name of the Office 365 Business Premium
service plan
        "E1", // The internal name of the Office 365 Enterprise E1
service plan
        "E3", // The internal name of the Office 365 Enterprise E3
service plan
        "E4" // The internal name of the Office 365 Enterprise E4
service plan
    ],
    "Trial": 0, // It defines if this service plan is trial (0 - not
trial/1 - trial).
    "ShowPriority": "20", // It defines the listing priority of the
service plan in your online store.
    "Resources": { // The list of resource rates of the service plan.
        "bd938f12-058f-4927-bba3-ae36b1d2501c": { // The identifier
of the Office 365 Business Essentials offer.
            "Included": 1, // It defines the number of included
units of the resource rate.
            "RecurringFee": 5.0, // The recurring fee of the
resource rate.
            "Name": "Office 365 Business Essentials", // The name
of the resource type/resource of the Office 365 Business offer.
            "Maximum": 300 // It defines the maximum number of
units of the resource rate.
        }
    },
    "RegisterByDefault": 1, // Reserved for future use.
    "SalesCategories": [ // The list of sales categories of the service
plan.
        "B" // The internal name of the Office 365 Business sales
category this service plan belongs to.
    ],
}
```

```

    "Description": "<span class=\"officelogo\"/><br/><div
class=\"ShortDescription\"><b>Online versions of Office with email and video
conferencing</b></div><ul><li>Email with 50 GB mailbox</li>\n<li>1 TB file
storage and sharing</li>\n<li>HD video conferencing</li>\n<li>Office
Online</li>\n</ul>" // The description of the service plan.
}

```

The following service plan is created on the basis of the description:

### Office 365 Business Essentials

- **Service Template: Office 365**
- **Service Terms: Office 365**
- **Subscription Period: Not Trial/1 Year/\$5.00 Recurring Fee**
- **Resource Rates**
  - **Resource/Inc. Amount/Max Amount/Recurring Fee: Office 365 Business Essentials/1/300/\$5.00**
- **Upgrades: Office 365 Business Essentials with Add-ons, Office 365 Business Premium, Office 365 Enterprise E1, Office 365 Enterprise E3, Office 365 Enterprise E4**

The created service plan is added to the **Office 365 Business** sales category.

### resourcecategories\_CSP.json

For instance, the **Additional Licenses** resource category is described in the following way:

```

"XL": { // The internal name of the resource category. Internal names are only
used in the .json files.
    "name": "Additional Licenses", // The name of the resource category.
    "DisplayOrder": 10, // It defines the Sort Number property of the
resource category.
    "Description": "By default, the subscription contains just a single
license of selected type. You can order additional licenses here." // The
description of the resource category.
}

```

The **Additional Licenses** resource category is created on the basis of the description.

### salescategories\_CSP.json

For instance, the **Office 365 Business** sales category is described in the following way:

```

"B": { // The internal name of the sales category. Internal names are only used
in the .json files.
    "ShowInStore": 1, // Reserved for future use.
    "name": "Office 365 Business", // The name of the sales category.
    "Description": "Office 365 Business Plans" // The description of the
sales category.
}

```

The **Office 365 Business** sales category is created on the basis of the description.

# Conversion from Syndication Partner Program to CSP Program

This section describes how to perform conversion from the Syndication Partner Program to the Cloud Solution Provider (CSP) Program.

Conversion can be performed in one of the following modes:

- **Subscription split mode** (page 126). Existing Office 365 syndication subscriptions are cancelled, and new Office 365 CSP subscriptions are created. As a result, identifiers of Office 365 subscriptions are not preserved. If your Odin Automation system is not integrated with an external system which relies on identifiers of Office 365 subscriptions, you can perform conversion in this mode.
- **Subscription preservation mode** (page 148). Existing Office 365 subscriptions are switched from Office 365 syndication service plans to Office 365 CSP service plans. As a result, identifiers of Office 365 subscriptions are preserved. If your Odin Automation system is integrated with an external system which relies on identifiers of Office 365 subscriptions, you must perform conversion in this mode.
- **Advanced subscription split mode** (page 169). Existing Office 365 syndication subscriptions are switched to service plans including only non-Office 365 resources and new Office 365 CSP subscriptions are created. This mode allows you not only to convert Office 365 syndication subscriptions into Office 365 CSP subscriptions but also to split existing Office 365 subscriptions into subscriptions including only non-Office 365 resources and subscriptions including Office 365 resources.

## In This Chapter

|                                       |     |
|---------------------------------------|-----|
| Subscription Split Mode .....         | 126 |
| Subscription Preservation Mode.....   | 148 |
| Advanced Subscription Split Mode..... | 169 |

## Subscription Split Mode

### Overview (Subscription Split Mode)

Conversion from the Syndication Partner Program to the CSP Program is performed on both the Microsoft side and the Odin Automation side.

## Microsoft Side

The Microsoft side consists of two layers:

- **A service layer.** The service layer is responsible for the provisioning of Office 365 services. All service data, such as customer tenant settings, end-user settings, domain settings, the content of mailboxes, and so forth, is stored and managed at this level.
- **A commercial layer.** The commercial level is responsible for tracking financial information of Office 365 services. All financial information, such as cloud subscriptions of customer tenants, service prices, cloud subscription limits, and so forth, is stored and managed at this level. When a commercial operation is performed, corresponding changes are propagated to the service layer. There are many commercial systems at the commercial layer, including the commercial system of the Syndication Partner Program and the commercial system of the CSP Program.

Conversion on the Microsoft side is the transportation of financial information from the commercial system of the Syndication Partner Program to the commercial system of the CSP Program.

Key points of conversion:

- The service data of customer tenants is not modified during the conversion of the customer tenants. Daily operations of end-users of the customer tenants are not affected.
- The Office 365 admin center of the CSP Program is slightly different from the Office 365 admin center of the Syndication Partner Program.
- The offer catalogue of the CSP Program and the offer catalogue of the Syndication Partner Program are different:
  - Both programs have a set of common offers, and conversion is possible only within the set. This means that cloud subscriptions of other offers must be either replaced with cloud subscriptions of common offers or cancelled before conversion.
  - The CSP Program does not support trial offers. This means that trial cloud subscriptions cannot be converted. Trial cloud subscriptions must be either replaced with paid ones or cancelled before conversion.
- The programs provide different APIs for managing customer tenants and cloud subscriptions. The Syndication Partner Program provides the MOSI API. The CSP Program provides the Partner Center API.
- The CSP Program is globally divided into geographical regions (the regions can be seen at <https://www.microsoft.com/en-us/licensing/licensing-programs/open-regional.aspx>). CSP partners can sell Office 365 services only within their regions. For example, a CSP partner from the North America region can sell Office 365 services to customers only from the United States and Canada. This means that several CSP partner accounts might be required for a syndication partner who sells Office 365 services in various countries.
- **TIP partner accounts cannot be used to test conversion.**

To test the whole conversion procedure before performing actual conversion, create temporary customer tenants and temporary cloud subscriptions through Odin Automation connected to both a production syndication partner account and a production CSP partner account.

- Before performing conversion, a partner must contact a Microsoft representative to discuss and approve the volume and schedule of conversion.

### **Odin Automation Side**

A partner can be represented in one of the following ways:

- A syndication partner has one or more Office 365 APS 1.2 'Application' resource types depending on the number of organization units the partner uses for selling Office 365 services. Each organization unit has its own set of Office 365 APS 1.2 syndication service plans.
- A syndication partner is represented by one or more Office 365 APS 2 application instances depending on the number of organization units the partner uses for selling Office 365 services. Each organization unit has its own set of Office 365 service plans.
- A CSP partner is represented by an Office 365 APS 2 application instance. The CSP partner has one set of Office 365 service plans.

Conversion on the Odin Automation side is the replacement of Office 365 syndication subscriptions with Office 365 CSP subscriptions. Existing Office 365 syndication subscriptions are cancelled, and new Office 365 CSP subscriptions are created.

Key points of conversion:

- The Office 365 APS 1.2 and Office 365 APS 2 applications are supported.
- A syndication partner might have several organization units for selling Office 365 services. This means that a separate CSP partner account is required for each organization unit.
- Odin Automation supports the following configurations of Office 365 service plans:
  - **Consolidated.** All offers are available within an Office 365 service plan. A typical customer account has one subscription to such an Office 365 service plan.
  - **Partitioned.** All offers are partitioned into groups. A typical group contains one offer and its depended add-on offers. Offers constituting a group are available only within a separate Office 365 service plan. A typical customer account has several subscriptions to such Office 365 service plans.

Cloud subscriptions and their limits are controlled by setting limits of offer resources of Office 365 subscriptions. If the limit of an offer resource is equal to zero, the corresponding cloud subscription either does not exist or is suspended. If the limit of an offer resource is equal to a positive value, the corresponding cloud subscription exists and is active; the limit of the cloud subscription is equal to the limit of the offer resource.



It is possible to change one service plan configuration to another one during conversion. If there is a choice between 'consolidated' and 'partitioned' Office 365 APS 2 CSP service plans when Office 365 APS syndication subscriptions of a customer tenant are replaced with Office 365 APS 2 CSP subscriptions, the 'consolidated' service plans take precedence over the 'partitioned' service plans.

- Conversion is performed by using scripts shipped with the **Office 365** application.
- Customer accounts whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert must be split into groups. You must convert these groups one by one. This means that several groups must not be converted concurrently.

### Limitations (Subscription Split Mode)

The conversion procedure has the following limitations:

- Office 365 subscriptions are recreated during conversion, which means that their subscription identifiers are not preserved.
- For an Office 365 CSP subscription created with the `importSubsCSP.py` script, all resource downgrade orders placed within the current billing period will contain zero resource prices and will not include order details. As a result, the customer placing the orders will not receive any refunds. When the current billing period ends and the first billing order is automatically placed for the Office 365 subscription, further resource downgrade orders will work as expected.

To calculate a refund for a resource downgrade order of a subscription, Odin Automation uses orders placed earlier. Since the script creates an Office 365 subscription without placing any order, Odin Automation does not have the necessary information to calculate the refund for a resource downgrade order of the Office 365 subscription until the first billing order is automatically placed, which will happen when the current billing period ends.

We recommend that you plan conversion according to this specific circumstance.

- The conversion scripts can automatically process Office 365 subscriptions including non-Office 365 resources only if the limits of the resources are zero. In order not to affect customer data, non-Office 365 resources whose limits are either greater than zero or unlimited must be manually processed taking into account the specifics of the resources.
- The conversion scripts do not automatically process Office 365 subscriptions whose 'DNS Hosting' resources are in use by domains. In order not to affect customer data, 'DNS Hosting' resources which are in use by domains must be manually processed taking into account the specifics of the resources.

**Note:** Subdomains of the **onmicrosoft.com** domain are automatically processed by the scripts.

### Prerequisites (Subscription Split Mode)

Before performing conversion, the following prerequisites must be met on your Odin Automation installation:

- Odin Automation is installed:
  - 7.3.0 or a later 7.3.x version
  - 7.2.0 or a later 7.2.x version
  - 7.1.0 or a later 7.1.x version
  - 7.0.1\_hf130748 or a later 7.0.x version
- Office 365 (Syndication Partner Program) 16.8.3 or higher (APS 2) / 4.17 or higher (APS 1.2) is installed.

**Important:** For Office 365 (Syndication Partner Program) 4.17, the instructions of the KB article <https://kb.odin.com/en/130292> must be applied.

- Office 365 (CSP Program) 16.8.3 or higher is installed.
- All trial syndication Office 365 subscriptions are either converted to paid ones or cancelled.
- All legacy syndication Office 365 subscriptions are either converted to modern ones or cancelled.

## Preparing for Conversion (Subscription Split Mode)

This section describes how to prepare your Odin Automation installation for conversion.

The preparation procedure consists of the following steps:

- 1 [Preparing your installation \(Subscription Split Mode\)](#) (page 130).
- 2 [Configuring customer notification messages \(Subscription Split Mode\)](#) (page 131).
- 3 [Preparing conversion scripts \(Subscription Split Mode\)](#) (page 132).

## Preparing Your Installation (Subscription Split Mode)

On your Odin Automation installation, configure necessary Office 365 APS 2 CSP application instances and Office 365 APS 2 CSP service plans:

- Each Office 365 APS 1.2 'Application' resource type must be represented by an Office 365 APS 2 CSP application instance (if you have Office 365 APS 1.2 on your installation).
- Each Office 365 APS 2 syndication application instance must have an Office 365 APS 2 CSP counterpart (if you have Office 365 APS 2 on your installation).
- Each vendor account which has Office 365 syndication service plans must have necessary Office 365 APS 2 CSP service plans so that syndication offers can be replaced with CSP offers.

For example: if there are syndication service plans which include resources of the syndication offers **Office 365 Business Essentials** and **Office 365 Enterprise E3**, there must be CSP service plans which include resources of the CSP offers **Office 365 Business Essentials** and **Office 365 Enterprise E3**.

- All the Office 365 APS 2 CSP service plans must have one subscription period. This is necessary since the conversion scripts do not take into account Office 365 APS 2 CSP service plans with several subscription periods.
- In each Office 365 APS 2 CSP service plan, **Show in Store** must be set to **Yes** for all Office 365 resource rates.

## Notifying Customers (Subscription Split Mode)

We recommend that you notify customers whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert. You can use any suitable method to notify the customers.

You can use the instructions of the section as guidelines of how to configure notification.

### Configuring 'Upcoming Conversion to CSP' Notification Messages

- 1 Create a notification template using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75484.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75484.htm).

Set **Message Type** to **Account Notification**.

- 2 Create a distribution list using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75977.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75977.htm).

Do the following:

- a Specify the name of the notification template you created in the **Notification Template** field.
  - b Click the **Send to All** button.
- 3 After creating the distribution list, add a service plan criterion for each Office 365 syndication service plan:
    - a Set **Operator** to **Or**.
    - b Specify the Office 365 syndication service plan in the **Service Plan** field.

Send the notification messages to the distribution list when necessary.

### Configuring 'Conversion to CSP Has Started' Notification Messages

Depending on the `--on-hold` parameter of the conversion scripts, Office 365 syndication subscriptions of customer tenants can be put on administrative hold during conversion. If Office 365 syndication subscriptions are put on administrative hold, these events can be used to trigger sending notification messages.

- 1 Create a notification template using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75484.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75484.htm).

Do the following:

- a** Set **Message Type** to **Subscription Notification**.
  - b** Specify `@@Subscription_Plan_ID@~="^(ID1)|(ID2)|(ID3)...$"` in the **Condition** field.
  - c** Replace **IDx** with the identifiers of all Office 365 syndication service plans.
- 2** Add an event handler to the **Subscription Put On Administrative Hold** event using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75997.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75997.htm).

Do the following:

- a** Set **Object** to **MESSAGE**.
- b** Set **Method** to **SubscriptionNotification**.
- c** Set **Signature** to **integer subscriptionID, string TemplateName=NAME\_OF\_NOTIFICATION\_TEMPLATE**. Replace **NAME\_OF\_NOTIFICATION\_TEMPLATE** with the name of the notification template you created.
- d** Set **Protocol** to **Stellart**.
- e** Set **Max Attempts** to **1**.
- f** Set **Repeat interval (seconds)** to **1**.
- g** Set **Synchronous** to **No**.

## Preparing Conversion Scripts (Subscription Split Mode)

Conversion is performed by using the scripts `prepareMOSI2CSP.py`, `completeMOSI2CSP.py`, and `importSubsCSP.py`.

Before performing conversion, prepare the scripts and a host for running them:

- 1** Prepare a host which meets the following requirements:
  - Python 2.6.6 is installed.
  - The `poaupdater` RPM package is installed on the host. The RPM package must be taken from the Odin Automation distribution package/YUM repository that belongs to your version of Odin Automation (for example, see the `DISTRIBUTION_PACKAGE/modules/platform/OS/OS_VERSION/RPMS/` directory).
  - The `pyPgSQL-2.5.1` (or higher) Python package is installed.
  - The `mx 3.1.1` (or higher) Python package is installed.
  - Connections from the host to the OA Operations Public API are allowed.
  - Connections from the host to the OA Billing Public API are allowed.
  - Connections from the host to the Odin Automation system database are allowed.

- Connections from the host to the APSC are allowed.
- Connections from the host to the Office 365 APS 1.2 application gateway sites are allowed (if you have Office 365 APS 1.2 on your Odin Automation installation).

Note that the Odin Automation Management Node can also be used for running the scripts if it meets the requirements.

- 2 Upload the Office 365 application package to the host.
- 3 Extract the scripts from the package. The scripts are here: `Office365-xx.x.x-xx.app.zip\O365-Import.zip`.
- 4 Write down the location of the prepared scripts.

Also, we recommend you to check that the Office 365 Application Endpoint Host (CSP) has access to all the necessary APIs and system components (APSC, OA Operations Public API, OA Billing Public API, Partner Center API, and so on). See the following sections for details: [Deployment Scheme](#) (page 16), [Preparing Odin Automation Public API User](#) (page 31), and [Firewall Requirements](#) (page 97).

**Note:** In case the conversion scripts are not able to connect to the Odin Automation system database or the Odin Automation APIs, check the `__dbhost`, `__dbname`, `__dbuser`, `__dbpass`, `__apihost` parameters in the `Utils.py` file.

## Conversion Procedure (Subscription Split Mode)

This section describes how to perform conversion.

The conversion procedure consists of the following steps:

- 1 [Selecting customer accounts for conversion \(Subscription Split Mode\)](#) (page 133). This step is performed on the Odin Automation side.
- 2 [Preparing customer accounts for conversion \(Subscription Split Mode\)](#) (page 136). This step is performed on the Odin Automation side.
- 3 [Performing conversion \(Subscription Split Mode\)](#) (page 140). This step is performed on the Microsoft side.
- 4 [Completing conversion \(Subscription Split Mode\)](#) (page 140). This step is performed on the Odin Automation side.

### Selecting Customer Accounts (Subscription Split Mode)

Select customer accounts for conversion:

#### 1. Selecting Customer Accounts for Conversion

Prepare a group of customer accounts whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert. You can use any suitable method to prepare the group, but consider the following:

- All customer accounts of the group must belong to the same vendor account. For example, customer accounts of one reseller account and customer accounts of another reseller account must not be in the same group.
- All Office 365 subscriptions of all customer accounts constituting the group must belong to the same APS version. This means that customer accounts with Office 365 APS 1.2 subscriptions and customer accounts with Office 365 APS 2 subscriptions must not be in the same group.
- All Office 365 subscriptions of all customer accounts constituting the group must belong to the same syndication partner account. This means that customer accounts with Office 365 subscriptions of one syndication partner account and customer accounts with Office 365 subscriptions of another syndication partner account must not be in the same group.
- There might be several organization units for one syndication partner account:
  - Organization units might be represented by separate 'Application' APS 1.2 resource types.
  - Organization units might be represented by separate APS 2 application instances.

Customer accounts with Office 365 subscriptions of one organization unit and customer accounts with Office 365 subscriptions of another organization unit must not be in the same group.

- The number of customer accounts per group affects the performance and convenience of conversion. Use small groups of customer accounts (for example 100-500) until you find a suitable number of customer accounts.
- The conversion scripts have the following limitations:
  - The scripts can automatically process Office 365 subscriptions including non-Office 365 resources only if the limits of the resources are zero. We recommend you to find and process non-Office 365 resources whose limits are either greater than zero or unlimited before performing conversion.
  - The scripts do not automatically process Office 365 subscriptions whose 'DNS Hosting' resources are in use by domains. We recommend you to find and process such 'DNS Hosting' resources before performing conversion.

**Note:** Subdomains of the **onmicrosoft.com** domain are automatically processed by the scripts.

- The identifiers of the customer accounts constituting the group must be placed to a CSV file:
  - The customer account identifiers must be specified in the **CustomerId** column.
  - The CSV file can contain other auxiliary columns. These columns will be ignored by the scripts.
  - The file must be uploaded to the host prepared for running the scripts.

You can use the following instructions as examples of how to prepare a group of customer accounts:

- 1 Getting a list of customer accounts of a vendor:
  - a Log in to the control panel of the vendor.
  - b Go to **Operations > Customers** and select the **Direct Customers** tab.
  - c Click the **Select Columns** link and select necessary columns including the **Account ID** column.
  - d Click the **Export to Excel** link and save the file.
  - e Open the file in Excel and rename the **Account ID** column to **CustomerId**.
  - f Save the file as a CSV file.
- 2 Getting a list of customer accounts subscribed to a service plan:
  - a Log in to the control panel of a vendor.
  - b Go to **Products > Service Plans** and select the service plan.
  - c Select the **Subscriptions** tab.
  - d Click the **Select Columns** link and select necessary columns including the **Customer Account ID** column.
  - e Click the **Export to Excel** link and save the file.
  - f Open the file in Excel and rename the **Customer Account ID** column to **CustomerId**.
  - g Save the file as a CSV file.
- 3 Getting a list of customer accounts with subscriptions of a service template:
  - a Log in to the control panel of a vendor.
  - b Go to **Products > Service Plans** and select the **Service Templates** tab.
  - c Select the service template.
  - d Select the **Subscriptions** tab.
  - e Click the **Select Columns** link and select necessary columns including the **Customer Account ID** column.
  - f Click the **Export to Excel** link and save the file.
  - g Open the file in Excel and rename the **Customer Account ID** column to **CustomerId**.
  - h Save the file as a CSV file.

**Important:** If you need to prepare a list of customer accounts of a reseller account, you must open the reseller control panel. This can be done in the following way: log in to the provider control panel; go to **Operations > Resellers**; click on the reseller account; click on the **To Control Panel** link on the **General** tab.

## 2. Preparing Application Information

After preparing the group, prepare and write down the following:

- Office 365 APS 1.2 (Syndication Partner Program)

- The identifier of the source Office 365 APS 1.2 application to which the group belongs. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; see the **ID** column.
- The URL of the gateway application of the 'Application' resource type to which the group belongs. To obtain the URL, perform the following actions: log in to the provider control panel; go to **Products > Resources > APPLICATION\_RESOURCE\_TYPE > Activation Parameters** tab > **Global Settings** subtab; use the value of the **URL for MOSI Connector** global setting.
- The email address of an administrative user of the syndication partner account to which the group belongs. Note the following: users of the syndication partner account can be seen in the Partner Center; the user must have the **Global admin** role in the Partner Center.
- Office 365 APS 2 (Syndication Partner Program)
  - The identifier of the source application instance to which the group belongs. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; click on the **Office 365** application; select the **Instances** tab; see the **ID** column.
  - The email address of an administrative user of the syndication partner account to which the group belongs. Note the following: users of the syndication partner account can be seen in the Partner Center; the user must have the **Global admin** role in the Partner Center.
- Office 365 APS 2 (CSP Program)
  - The identifier of the target application instance. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; click on the **Office 365** application; select the **Instances** tab; use the **ID** column.
  - The Microsoft ID of the CSP partner account of the target application instance. The Microsoft ID can be obtained at <https://partnercenter.microsoft.com/en-us/pcv/accountsettings/organizationprofile>.

This information will be used in further steps of the conversion procedure.

## Preparing Customer Accounts (Subscription Split Mode)

The customer accounts you selected in the previous step of the conversion procedure must be prepared.

The following preparation activities are performed by the script `prepareMOSI2CSP.py` for each customer tenant of every customer account:

- 1 The customer tenant is validated against the following requirements:
  - The customer tenant must not have trial syndication cloud subscriptions.
  - The customer tenant must not have syndication cloud subscriptions which cannot be converted into CSP cloud subscriptions.



- There must not be pending or failed tasks related to the customer tenant.
- Office 365 subscriptions of the customer tenant can include non-Office 365 resources only if the limits of the resources are zero.
- None of the Office 365 subscriptions of the customer tenant must include 'DNS Hosting' resources which are in use by domains.

Validation is necessary to identify if a customer tenant is ready for conversion. If a customer tenant does not pass validation, this means that it is not ready for conversion and further preparation activities are not performed for it.

- 2** MOSI API communication is disabled for the customer tenant. This is needed to protect customer tenants and their cloud subscriptions from changes made in Odin Automation.
- 3** The MOSI Customer ID of the customer tenant is stored. The MOSI Customer ID is sent to Microsoft to convert the customer tenant and its cloud subscriptions on the Microsoft side.
- 4** The mapping between the customer account and its customer tenant is stored. This mapping will be used to find the customer account of the customer tenant in further steps of the conversion procedure.

### Parameters of prepareMOSI2CSP.py Script

Note that the `--mosi-app-id` and `--mosi-gateway-url` parameters and the `--mosi-app-instance-id` parameter are mutually exclusive.

- `--conversion-mode` – It defines the mode of conversion. You must specify the **resplitSubscriptions** value.
- `--on-hold` – It defines if Office 365 subscriptions are put on administrative hold. You must specify either **yes** or **no**. Note that the same value must be used in further steps of the conversion procedure.

We recommend using **yes** to prevent customers from placing upgrade and downgrade orders during conversion.

- `--mosi-app-id` – It defines the identifier of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--mosi-gateway-url` – It defines the URL of the gateway application of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--mosi-app-instance-id` – It defines the identifier of a source Office 365 APS 2 application instance (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the previous step of the conversion procedure.

- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer accounts which must be prepared for conversion. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.
- `--out-file` – It defines the name of a CSV file containing a list of customer tenants which are ready for conversion. The CSV file is created by the `prepareMOSI2CSP.py` script. The file is sent to Microsoft to convert the customer tenants on the Microsoft side.
- `--error-file` – It defines the name of a CSV file containing a list of customer tenants which are not ready for conversion. The CSV file is created by the `prepareMOSI2CSP.py` script. If the parameter is not specified, the default name `error_YYYYMMDD-HHMMSS.csv` is used.

### Input Files of `prepareMOSI2CSP.py` Script

The `prepareMOSI2CSP.py` script uses an input CSV file containing a list of customer accounts which must be prepared for conversion. Each row of the CSV file contains the following data of a customer account:

- **CustomerId** – The identifier of the customer account.
- other auxiliary columns (optional) – These columns are ignored by the script.

### Output Files of `prepareMOSI2CSP.py` Script

The `prepareMOSI2CSP.py` script creates an output CSV file containing a list of customer tenants which are ready for conversion. The name of the CSV file is defined by the `--out-file` parameter of the script. Each row of the CSV file contains the following data of a customer tenant:

- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.

The script creates an output CSV file containing a list of customer tenants which are not ready for conversion. The name of the CSV file is defined by the `--error-file` parameter of the script; if the parameter is not specified, the default name `error_YYYYMMDD-HHMMSS.csv` is used. Each row of the CSV file contains the following data of a customer tenant:

- **Aps** – The APS version of the Office 365 application to which the customer tenant belongs.
- **mosiAppldOrMosiAppInstanced** – It contains one of the following: the identifier of the Office 365 APS 1.2 application to which the customer tenant belongs; the identifier of the Office 365 APS 2 application instance to which the customer tenant belongs.
- **CustomerId** – The identifier of the customer account to which the customer tenant belongs.
- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.

- **MSTenantId** – The Microsoft Tenant ID of the customer tenant.
- **SubscriptionId** – The identifier of an Office 365 subscription which blocks the conversion of the customer tenant.
- **Error** – An error message.

### Log and Auxiliary Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` script creates the following log and auxiliary files:

- `prepareMOSI2CSP_YYYYMMDD-HHMMSS.log` – A log file containing general messages, such as information messages and error messages.
- `mosi2csp_mapping.csv` – A CSV file containing mappings between customer accounts and their customer tenants. The file is auxiliary and must not be modified or renamed. The file will be used by the `completeMOSI2CSP.py` script in further steps of the conversion procedure.

### Running prepareMOSI2CSP.py Script

Log in to the host where the conversion scripts are placed. Run the `prepareMOSI2CSP.py` script.

Example (Office 365 APS 1.2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitSubscriptions --on-hold yes --
mosi-app-id 100 --mosi-gateway-url
https://office365.provider.com/Office365GatewayApplication/HTTPEndpoint.aspx --
csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --error-
file errors.csv
```

Example (Office 365 APS 2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitSubscriptions --on-hold yes --
mosi-app-instance-id 999 --csp-app-instance-id 1000 --pba --input-file input.csv
--out-file out.csv --error-file errors.csv
```

After running the script, check the output CSV file containing a list of customer tenants which are not ready for conversion. If the file is not empty, you have the following options:

- Investigate and fix the errors. Run the script again using the same values of parameters. Repeat this until all customer tenants are ready for conversion.
- Postpone processing the customer tenants which are not ready for conversion and process them later. Note that you will need to start the conversion procedure from the beginning for the postponed customer tenants.

### Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

## Performing Conversion (Subscription Split Mode)

The customer tenants you prepared in the previous step of the conversion procedure must be converted on the Microsoft side:

- 1 Fill out a conversion request form:
  - a Download the conversion request form template provided at <https://kb.odin.com/en/130124>.
  - b Fill out the form using the information you prepared earlier:
    - The email address of the administrative user of the required syndication partner account.
    - The Microsoft ID of the required CSP partner account.
    - The CSV file containing the list of the customer tenants.
- 2 Send the file of the form to [csprans@microsoft.com](mailto:csprans@microsoft.com).
- 3 Microsoft will perform conversion on its side and provide you with two CSV files:
  - A CSV file containing a list of customer tenants which were successfully converted on the Microsoft side.
  - A CSV file containing a list of customer tenants which cannot be converted on the Microsoft side.
- 4 Check that the CSV files received from Microsoft have the **MosiCustomerId** column.
- 5 Upload both the CSV files to the host for running the conversion scripts.

The CSV files received from Microsoft will be used in further steps of the conversion procedure.

## Completing Conversion (Subscription Split Mode)

Complete the conversion of the customer tenants processed by Microsoft in the previous step of the conversion procedure:

- Use the instruction of the [Processing Customer Tenants Converted by Microsoft \(Subscription Split Mode\) section](#) (page 140) to process the customer tenants converted by Microsoft.
- Use the instruction of the [Processing Customer Tenants Rejected by Microsoft \(Subscription Split Mode\) section](#) (page 146) to process the customer tenants rejected by Microsoft.

### Processing Customer Tenants Converted by Microsoft (Subscription Split Mode)

The customer tenants converted on the Microsoft side in the previous step of the conversion procedure must be processed on the Odin Automation side.

The following activities are performed for every customer tenant:

- 1 Suitable Office 365 APS 2 service plans (CSP Program) are found for the customer account of the customer tenant. This means that all cloud subscriptions of the customer tenant will be represented by the corresponding Office 365 APS 2 subscriptions (CSP Program) in Odin Automation. This activity is performed by using the `completeMOSI2CSP.py` script.

Note the following:

- There might be several suitable Office 365 APS 2 service plans (CSP Program) for all cloud subscriptions (CSP Program) of a customer tenant. An Office 365 APS 2 service plan (CSP Program) containing suitable offer resources for all the cloud subscriptions (CSP Program) takes precedence over a set of Office 365 APS 2 service plans (CSP Program) containing suitable offer resources for all the cloud subscriptions (CSP Program).
  - If suitable Office 365 APS 2 service plans (CSP Program) are not found for the customer account of a customer tenant, further activities are not performed for the customer tenant.
  - Office 365 APS 2 service plans (CSP Program) containing composite resources are supported by the script. If each composite resource of an Office 365 APS 2 service plan (CSP Program) contains one included resource and the amount of the included resource is equal to **1**, the Office 365 APS 2 service plan (CSP Program) is taken into account by the script to find suitable Office 365 APS 2 service plans (CSP Program) for cloud subscriptions.
  - Both active cloud subscriptions and suspended cloud subscriptions are taken into account by the script.
- 2 All Office 365 subscriptions (Syndication Partner Program) of the customer tenant are cancelled. This activity is performed by using the `completeMOSI2CSP.py` script.

**Note:** All syndication subscriptions must be removed after the conversion procedure as it is prohibited to have subscriptions from different application instances within one account. You can set the **Destroy Service on Cancel** option of the **Office 365** service template in OA Billing for the conversion period. Take into account that the option will affect the cancellation of all the subscriptions based on the service template.

- 3 The customer account of the customer tenant is subscribed to the Office 365 APS 2 service plans (CSP Program) found earlier. This activity is performed by using the `importSubsCSP.py` script.

### Parameters of `completeMOSI2CSP.py` Script

- `--conversion-mode` – It defines the mode of conversion. You must specify the `resplitSubscriptions` value.
- `--on-hold` – Specify **yes** if Office 365 subscriptions are on administrative hold. Otherwise, specify **no**.

- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the first step of the conversion procedure.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer tenants converted by Microsoft. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.
- `--out-file` – It defines the name of a CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The CSV file is created by the `completeMOSI2CSP.py` script. The CSV file will be used by the `importSubsCSP.py` script.
- `--error-file` – It defines the name of a CSV file containing a list of customer tenants which cannot be processed. The CSV file is created by the `completeMOSI2CSP.py` script. If the parameter is not specified, the default name `errors_completion_YYYYMMDD-HHMMSS.csv` is used.

### Input Files of `completeMOSI2CSP.py` Script

The `completeMOSI2CSP.py` script uses an input CSV file containing a list of customer tenants converted by Microsoft. Each row of the CSV file contains the following data of a cloud subscription of a customer tenant:

- **CustomerName** – The Business Name of the customer tenant.
- **CustomerTenantId** – The Microsoft Tenant ID of the customer tenant.
- **FpoCaid** – The Microsoft Internal Commerce ID of the customer tenant.
- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.
- **MosiSubscriptionId** – The MOSI subscription identifier of the cloud subscription.
- **CspSubscriptionId** – The CSP subscription identifier of the cloud subscription.
- **CspSubscriptionOfferId** – The CSP Offer ID of the cloud subscription.
- **CspSubscriptionStartDate** – The start date of the cloud subscription.
- **CspSubscriptionEndDate** – The end date of the cloud subscription.

Note that the `completeMOSI2CSP.py` script uses the **MosiCustomerId** column and ignores the other columns.

### Output Files of `completeMOSI2CSP.py` Script

The `completeMOSI2CSP.py` script creates an output CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The name of the CSV file is defined by the `--out-file` parameter of the script. Each row of the CSV file contains the following data of a customer tenant:

- **OSACustomerID** – The identifier of the customer account of the customer tenant.
- **OSAServicePlanId** – The identifier of an Office 365 APS 2 service plan (CSP Program) to which the cloud subscriptions specified in the **MSSubscriptionIDs** column are mapped by the script.
- **OSAPlanPeriodID** – The identifier a subscription period of the Office 365 APS 2 service plan (CSP Program). The subscription period is chosen by the script.
- **SubscriptionStartDate** – The start date of an Office 365 APS 2 subscription (CSP Program) which is created for the cloud subscriptions specified in the **MSSubscriptionIDs** column.
- **ResourceRateLimits** – A list of resource rates and their limits. The resource rates are chosen by the script for the cloud subscriptions specified in the **MSSubscriptionIDs** column. The format "**RR\_ID1-LIMIT,RR\_ID2-LIMIT**" is used for the list. For example: "**100-10,200-20**".
- **CSPAccountid** – The Microsoft Tenant ID of the customer tenant.
- **MSSubscriptionIDs** – A list of cloud subscriptions of the customer tenant. The format "**MS\_SUB\_ID1,MS\_SUB\_ID2,...**" is used for the list. For example: "**A49C6781-1557-42CA-A074-52D0553397EA,EBE5106C-B1ED-44FF-8CE7-A2746AF8BC42**".
- **DefaultAdminLogin** – The login name of the default administrative user of the customer tenant.
- **DefaultDomainName** – The default domain name of the customer tenant.
- **SourceSubscriptionId** – This column is not used in this mode of conversion. Do not change the default value of the column.

Note that the CSV file might contain several rows for one customer tenant if the cloud subscriptions of the customer tenant are mapped to several suitable Office 365 APS 2 service plans (CSP Program).

The `completeMOSI2CSP.py` script creates an output CSV file containing a list of customer tenants whose cloud subscriptions cannot be mapped to Office 365 APS 2 service plans (CSP Program). The name of the CSV file is defined by the `--error-file` parameter of the script; if the parameter is not specified, the default name `errors_completion_YYYYMMDD-HHMMSS.csv` is used. Each row of the CSV file contains the following data of a customer tenant:

- **OSACustomerID** – The identifier of the customer account of the customer tenant.
- **CSPAccountid** – The Microsoft Tenant ID of the customer tenant.
- **SubscriptionStartDate** – The start date of the cloud subscriptions specified in the **MSSubscriptionIDs** column.
- **MSSubscriptionIDs** – A list of cloud subscriptions of the customer tenant. The format "**MS\_SUB\_ID1,MS\_SUB\_ID2,...**" is used for the list. For example: "**A49C6781-1557-42CA-A074-52D0553397EA,EBE5106C-B1ED-44FF-8CE7-A2746AF8BC42**".

- **MSSubscriptionOfferIDs** – The identifiers of the offers of the cloud subscriptions specified in the **MSSubscriptionIDs** column.
- **Comments** – An error message.

### Log and Auxiliary Files of completeMOSI2CSP.py Script

The `completeMOSI2CSP.py` script creates a log file with the name `completeMOSI2CSP_YYYYMMDD-HHMMSS.log`. The file contains general messages, such as information messages and error messages.

The `completeMOSI2CSP.py` script implicitly uses the `mosi2csp_mapping.csv` file created by the `prepareMOSI2CSP.py` script. The CSV file contains mappings between customer accounts and their customer tenants. The file is auxiliary and must not be modified or renamed.

### Running completeMOSI2CSP.py Script

- 1 Log in to the host where the conversion scripts are placed.
- 2 Run the `completeMOSI2CSP.py` script.
- 3 Check the output CSV file containing a list of customer tenants whose cloud subscriptions cannot be mapped to Office 365 APS 2 service plans (CSP Program). If the CSV file is not empty, investigate and fix the errors. Run the script again using the same values of parameters. Repeat this until all customer tenants are processed.
- 4 Check the output CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program):
  - a Make sure that mappings are correct.
  - b Make sure that resource rates and their limits are correct.

Modify the CSV file if necessary. For instance, you can specify your own mappings between Office 365 APS 2 service plans and cloud subscriptions.

Example:

```
python completeMOSI2CSP.py --conversion-mode resplitSubscriptions --on-hold yes -  
-csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --  
error-file error.csv
```

### Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

### Parameters of importSubsCSP.py Script

- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the first step of the conversion procedure.



- The name of a CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). You must use the CSV file created by the `completeMOSI2CSP.py` script.

### **Input Files of importSubsCSP.py Script**

The `importSubsCSP.py` script uses an input CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The CSV file is created by the `completeMOSI2CSP.py` script. The format of the CSV file is described above.

### **Output Files of importSubsCSP.py Script**

The `importSubsCSP.py` script creates a CSV file with the name `result_<input file name>`. Each row of the CSV file contains the following data of an Office 365 CSP subscription created by the `importSubsCSP.py` script:

- **CustomerId** – The identifier of the customer account to which the subscription belongs.
- **SubscriptionId** – The Odin Automation identifier of the newly created Office 365 CSP subscription.

### **Log and Auxiliary Files of importSubsCSP.py Script**

The `importSubsCSP.py` script creates a CSV file with the name `errors_import_<input file name>`. This file contains a list of cloud subscriptions which cannot be processed by the `importSubsCSP.py` script as well as reasons why the cloud subscriptions cannot be processed. The CSV file has the same format as the input file of the `importSubsCSP.py` script (except for the additional **Error** column, which contains error messages).

The `importSubsCSP.py` script creates a CSV file with the name `reimport_<input file name>`. This file contains a list of cloud subscriptions which cannot be processed by the `importSubsCSP.py` script. The CSV file has the same format as the input file of the `importSubsCSP.py` script. You can use the file `reimport_<input file name>` to run the `importSubsCSP.py` script again only for affected cloud subscriptions.

The `importSubsCSP.py` script creates a log file with the name `importCSPYYYY-MM-DD.log`. The file contains general messages, such as information messages and error messages.

### **Running importSubsCSP.py Script**

- 1 Stop Office 365 synchronization:
  - a In Provider Control Panel, go to **Operations > Tasks**.
  - b Select the **Periodic** tab.
  - c Cancel all the **Office 365 \* Synchronization with Office 365 Portal** tasks.

- 2 Log in to the host where the conversion scripts are placed.
- 3 Run the `importSubsCSP.py` script.
- 4 After running the script, Office 365 import tasks will be scheduled in the task manager. Wait until they are processed.
- 5 Start Office 365 synchronization:
  - a In Provider Control Panel, go to **Operations > Tasks**.
  - b Select the **Periodic** tab.
  - c Run all the **Office 365 \* Synchronization with Office 365 Portal** tasks.

Example:

```
python importSubsCSP.py --csp-app-instance-id 1000 input.csv
```

After running the script, perform the following validation steps for necessary customer tenants:

- 1 Check that the customer account of the customer tenant has correct Office 365 APS 2 subscriptions (CSP Program).
- 2 Check that the limits of the Office 365 APS 2 subscriptions match the limits of the cloud subscriptions of the customer tenant.

### Troubleshooting

- Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.
- If the `importSubsCSP.py` script fails with the `A duplicated entry is found for the SA Customer #CustomerId and Service Plan #PlanId in the line LineNumber` error, consider using the `--allow-duplicates` option described at <https://kb.odin.com/en/128798>.

### Processing Customer Tenants Rejected by Microsoft (Subscription Split Mode)

The customer tenants rejected by Microsoft in the previous step of the conversion procedure must be processed on the Odin Automation side.

The following activities are performed by the `prepareMOSI2CSP.py` script for every customer tenant:

- 1 If all Office 365 subscriptions of the customer tenant are on administrative hold, they are released from administrative hold.
- 2 MOSI API communication is enabled for the customer tenant.

### Parameters of `prepareMOSI2CSP.py` Script

Note that the `--mosi-app-id` and `--mosi-gateway-url` parameters and the `--mosi-app-instance-id` parameter are mutually exclusive.

- `--conversion-mode` – It defines the mode of conversion. You must specify the `resplitSubscriptions` value.

- `--rollback` – It defines if the script must roll back changes made for customer tenants. You must specify the parameter to process customer tenants rejected by Microsoft.
- `--mosi-app-id` – It defines the identifier of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--mosi-gateway-url` – It defines the URL of the gateway application of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--mosi-app-instance-id` – It defines the identifier of a source Office 365 APS 2 application instance (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer tenants rejected by Microsoft. The CSV file is received from Microsoft. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.

### Input Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` uses an input CSV file containing a list of customer tenants rejected by Microsoft. The `prepareMOSI2CSP.py` script uses the **MosiCustomerId** column and ignores other columns.

### Log and Auxiliary Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` script creates a log file with the name `prepareMOSI2CSP_YYYYMMDD-HHMMSS.log`. The log file contains general messages, such as information messages and error messages.

### Running prepareMOSI2CSP.py Script

Log in to the host where the conversion scripts are placed. Run the `prepareMOSI2CSP.py` script.

Example (Office 365 APS 1.2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitSubscriptions --rollback --
mosi-app-id 100 --mosi-gateway-url
https://office365.provider.com/Office365GatewayApplication/HTTPEndpoint.aspx --
pba --input-file input.csv
```

Example (Office 365 APS 2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitSubscriptions --rollback --
mosi-app-instance-id 999 --pba --input-file input.csv
```

After running the script, you can analyze the customer tenants rejected by Microsoft to find out the reason for rejection. Note that you will need to start the conversion procedure from the beginning for rejected customer tenants after fixing them.

## Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

# Subscription Preservation Mode

## Overview (Subscription Preservation Mode)

Conversion from the Syndication Partner Program to the CSP Program is performed on both the Microsoft side and the Odin Automation side.

### Microsoft Side

The Microsoft side consists of two layers:

- **A service layer.** The service layer is responsible for the provisioning of Office 365 services. All service data, such as customer tenant settings, end-user settings, domain settings, the content of mailboxes, and so forth, is stored and managed at this level.
- **A commercial layer.** The commercial level is responsible for tracking financial information of Office 365 services. All financial information, such as cloud subscriptions of customer tenants, service prices, cloud subscription limits, and so forth, is stored and managed at this level. When a commercial operation is performed, corresponding changes are propagated to the service layer. There are many commercial systems at the commercial layer, including the commercial system of the Syndication Partner Program and the commercial system of the CSP Program.

Conversion on the Microsoft side is the transportation of financial information from the commercial system of the Syndication Partner Program to the commercial system of the CSP Program.

Key points of conversion:

- The service data of customer tenants is not modified during the conversion of the customer tenants. Daily operations of end-users of the customer tenants are not affected.
- The Office 365 admin center of the CSP Program is slightly different from the Office 365 admin center of the Syndication Partner Program.
- The offer catalogue of the CSP Program and the offer catalogue of the Syndication Partner Program are different:
  - Both programs have a set of common offers, and conversion is possible only within the set. This means that cloud subscriptions of other offers must be either replaced with cloud subscriptions of common offers or cancelled before conversion.
  - The CSP Program does not support trial offers. This means that trial cloud subscriptions cannot be converted. Trial cloud subscriptions must be either replaced with paid ones or cancelled before conversion.

- The programs provide different APIs for managing customer tenants and cloud subscriptions. The Syndication Partner Program provides the MOSI API. The CSP Program provides the Partner Center API.
- The CSP Program is globally divided into geographical regions (the regions can be seen at <https://www.microsoft.com/en-us/licensing/licensing-programs/open-regional.aspx>). CSP partners can sell Office 365 services only within their regions. For example, a CSP partner from the North America region can sell Office 365 services to customers only from the United States and Canada. This means that several CSP partner accounts might be required for a syndication partner who sells Office 365 services in various countries.
- **TIP partner accounts cannot be used to test conversion.**

To test the whole conversion procedure before performing actual conversion, create temporary customer tenants and temporary cloud subscriptions through Odin Automation connected to both a production syndication partner account and a production CSP partner account.
- Before performing conversion, a partner must contact a Microsoft representative to discuss and approve the volume and schedule of conversion.

### **Odin Automation Side**

A partner can be represented in one of the following ways:

- A syndication partner has one or more Office 365 APS 1.2 'Application' resource types depending on the number of organization units the partner uses for selling Office 365 services. Each organization unit has its own set of Office 365 APS 1.2 syndication service plans.
- A syndication partner is represented by one or more Office 365 APS 2 application instances depending on the number of organization units the partner uses for selling Office 365 services. Each organization unit has its own set of Office 365 service plans.
- A CSP partner is represented by an Office 365 APS 2 application instance. The CSP partner has one set of Office 365 service plans.

Conversion on the Odin Automation side is the transformation of Office 365 syndication subscriptions into Office 365 CSP subscriptions. Existing Office 365 subscriptions are switched from Office 365 syndication service plans to Office 365 CSP service plans.

Key points of conversion:

- Since Office 365 subscriptions are only switched from one service plan to another one during conversion, subscription identifiers are preserved.
- Office 365 subscriptions can contain 'DNS Hosting' resources which are in use by domains.
- Office 365 subscriptions can contain non-Office 365 resources with any limits.
- The Office 365 APS 1.2 and Office 365 APS 2 applications are supported.
- A syndication partner might have several organization units for selling Office 365 services. This means that a separate CSP partner account is required for each organization unit.

- Conversion is performed by using scripts shipped with the **Office 365** application.
- Customer accounts whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert must be split into groups. You must convert these groups one by one. This means that several groups must not be converted concurrently.

### Limitations (Subscription Preservation Mode)

The conversion procedure has the following limitations:

- Office 365 subscriptions can be converted only if they meet the following requirements:
  - **Status** is **Active** (in OA Billing)
  - **Service Status** is **Running** (in OA Billing)

If Office 365 subscriptions are in other states, they must be manually made either **Active/Running** or **Terminated/Removed**.

- Odin Automation supports the following configurations of Office 365 service plans:
  - **Consolidated.** All offers are available within an Office 365 service plan. A typical customer account has one subscription to such an Office 365 service plan.
  - **Partitioned.** All offers are partitioned into groups. A typical group contains one offer and its depended addon offers. Offers constituting a group are available only within a separate Office 365 service plan. A typical customer account has several subscriptions to such Office 365 service plans.

Cloud subscriptions and their limits are controlled by setting limits of offer resources of Office 365 subscriptions. If the limit of an offer resource is equal to zero, the corresponding cloud subscription either does not exist or is suspended. If the limit of an offer resource is equal to a positive value, the corresponding cloud subscription exists and is active; the limit of the cloud subscription is equal to the limit of the offer resource.

Since Office 365 subscriptions are switched from Office 365 syndication service plans to Office 365 CSP service plans during conversion, it is not possible to change one service plan configuration to another one.

### Prerequisites (Subscription Preservation Mode)

Before performing conversion, the following prerequisites must be met on your Odin Automation installation:

- Odin Automation is installed:
  - 7.3.0 or a later 7.3.x version
  - 7.2.0 or a later 7.2.x version
  - 7.1.0 or a later 7.1.x version
  - 7.0.1\_hf130748 or a later 7.0.x version

- Office 365 (Syndication Partner Program) 16.8.3 or higher (APS 2) / 4.17 or higher (APS 1.2) is installed.

**Important:** For Office 365 (Syndication Partner Program) 4.17, the instructions of the KB article <https://kb.odin.com/en/130292> must be applied.

- Office 365 (CSP Program) 16.8.3 or higher is installed.
- All trial syndication Office 365 subscriptions are either converted to paid ones or cancelled.
- All legacy syndication Office 365 subscriptions are either converted to modern ones or cancelled.

## Preparing for Conversion (Subscription Preservation Mode)

This section describes how to prepare your Odin Automation installation for conversion.

The preparation procedure consists of the following steps:

- 1 [Preparing your installation \(Subscription Preservation Mode\)](#) (page 151).
- 2 [Configuring customer notification messages \(Subscription Preservation Mode\)](#) (page 152).
- 3 [Preparing conversion scripts \(Subscription Preservation Mode\)](#) (page 153).

## Preparing Your Installation (Subscription Preservation Mode)

On your Odin Automation installation, configure necessary Office 365 APS 2 CSP application instances and Office 365 APS 2 CSP service plans:

- Each Office 365 APS 1.2 'Application' resource type must be represented by an Office 365 APS 2 CSP application instance (if you have Office 365 APS 1.2 on your installation).
- Each Office 365 APS 2 syndication application instance must have an Office 365 APS 2 CSP counterpart (if you have Office 365 APS 2 on your installation).
- Each vendor account which has Office 365 syndication service plans must have necessary Office 365 APS 2 CSP service plans so that syndication offers can be replaced with CSP offers.
- Each Office 365 syndication service plan and its related Office 365 APS 2 CSP service plan must be configured in the following way:

- The Office 365 APS 2 CSP service plan includes all necessary Office 365 offers so that they can replace the Office 365 offers of the Office 365 syndication service plan.

For example: if a syndication service plan and its service template include a resource of the syndication offer **Office 365 Business Essentials**, its related CSP service plan and its service template must include a resource of the CSP offer **Office 365 Business Essentials**; the resource rate of the resource of the CSP service plan must have the same settings as the resource rate of the resource of the syndication service plan has (**Inc. Amount**, **Max Amount**, **Recurring Fee**, and so forth).

- If the Office 365 syndication service plan includes 'DNS Hosting' and non-Office 365 resources, the same resources are included into the Office 365 APS 2 CSP service plan; the limits of the resources are set to the same values.  
  
For example: if a syndication service plan and its service template include non-Office 365 resources, its related CSP service plan and its service template must include the same non-Office 365 resources; the resource rates of the resources of the CSP service plan must have the same settings as the resource rates of the resources of the syndication service plan have (**Inc. Amount, Max Amount, Recurring Fee**, and so forth).
- The Office 365 APS 2 CSP service plan has one subscription period (it is assumed that the Office 365 syndication service plan also has one subscription period). The subscription period of the Office 365 APS 2 CSP service plan has the same settings as the subscription period of the Office 365 syndication service plan has (**Number of Billing Periods, Recurring Fee**, and so forth).
- The Office 365 APS 2 CSP service plan has the same billing settings as the Office 365 syndication service plan has (**Billing Model, Billing Period**, and so forth).
- The Office 365 APS 2 CSP service plan is added to the allowed upgrade paths of the Office 365 syndication service plan.
- In the Office 365 APS 2 CSP service plan, **Show in Store** is set to **Yes** for all Office 365 resource rates.

### Notifying Customers (Subscription Preservation Mode)

We recommend that you notify customers whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert. You can use any suitable method to notify the customers.

You can use the instructions of the section as guidelines of how to configure notification.

#### Configuring 'Upcoming Conversion to CSP' Notification Messages

- 1 Create a notification template using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75484.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75484.htm).

Set **Message Type** to **Account Notification**.

- 2 Create a distribution list using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75977.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75977.htm).

Do the following:

- a Specify the name of the notification template you created in the **Notification Template** field.
- b Click the **Send to All** button.



- 3 After creating the distribution list, add a service plan criterion for each Office 365 syndication service plan:
  - a Set **Operator** to **Or**.
  - b Specify the Office 365 syndication service plan in the **Service Plan** field.

Send the notification messages to the distribution list when necessary.

### Configuring 'Conversion to CSP Has Started' Notification Messages

Depending on the `--on-hold` parameter of the conversion scripts, Office 365 syndication subscriptions of customer tenants can be put on administrative hold during conversion. If Office 365 syndication subscriptions are put on administrative hold, these events can be used to trigger sending notification messages.

- 1 Create a notification template using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75484.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75484.htm).

Do the following:

- a Set **Message Type** to **Subscription Notification**.
  - b Specify `@@Subscription_Plan_ID@~="^(ID1)|(ID2)|(ID3)...$"` in the **Condition** field.
  - c Replace **IDx** with the identifiers of all Office 365 syndication service plans.
- 2 Add an event handler to the **Subscription Put On Administrative Hold** event using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75997.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75997.htm).

Do the following:

- a Set **Object** to **MESSAGE**.
- b Set **Method** to **SubscriptionNotification**.
- c Set **Signature** to `integer subscriptionID, string TemplateName=NAME_OF_NOTIFICATION_TEMPLATE`. Replace **NAME\_OF\_NOTIFICATION\_TEMPLATE** with the name of the notification template you created.
- d Set **Protocol** to **Stellart**.
- e Set **Max Attempts** to **1**.
- f Set **Repeat interval (seconds)** to **1**.
- g Set **Synchronous** to **No**.

### Preparing Conversion Scripts (Subscription Preservation Mode)

Conversion is performed by using the scripts `prepareMOSI2CSP.py`, `completeMOSI2CSP.py`, and `importSubsCSP.py`.

Before performing conversion, prepare the scripts and a host for running them:

- 1 Prepare a host which meets the following requirements:
  - Python 2.6.6 is installed.
  - The poaupdater RPM package is installed on the host. The RPM package must be taken from the Odin Automation distribution package/YUM repository that belongs to your version of Odin Automation (for example, see the `DISTRIBUTION_PACKAGE/modules/platform/OS/OS_VERSION/RPMS/` directory).
  - The pyPgSQL-2.5.1 (or higher) Python package is installed.
  - The mx 3.1.1 (or higher) Python package is installed.
  - Connections from the host to the OA Operations Public API are allowed.
  - Connections from the host to the OA Billing Public API are allowed.
  - Connections from the host to the Odin Automation system database are allowed.
  - Connections from the host to the APSC are allowed.
  - Connections from the host to the Office 365 APS 1.2 application gateway sites are allowed (if you have Office 365 APS 1.2 on your Odin Automation installation).

Note that the Odin Automation Management Node can also be used for running the scripts if it meets the requirements.

- 2 Upload the Office 365 application package to the host.
- 3 Extract the scripts from the package. The scripts are here: `Office365-xx.x.x-xx.app.zip\O365-Import.zip`.
- 4 Write down the location of the prepared scripts.

Also, we recommend you to check that the Office 365 Application Endpoint Host (CSP) has access to all the necessary APIs and system components (APSC, OA Operations Public API, OA Billing Public API, Partner Center API, and so on). See the following sections for details: [Deployment Scheme](#) (page 16), [Preparing Odin Automation Public API User](#) (page 31), and [Firewall Requirements](#) (page 97).

**Note:** In case the conversion scripts are not able to connect to the Odin Automation system database or the Odin Automation APIs, check the `__dbhost`, `__dbname`, `__dbuser`, `__dbpass`, `__apihost` parameters in the `Utils.py` file.

## Conversion Procedure (Subscription Preservation Mode)

This section describes how to perform conversion.

The conversion procedure consists of the following steps:

- 1 [Selecting customer accounts for conversion \(Subscription Preservation Mode\)](#) (page 155). This step is performed on the Odin Automation side.
- 2 [Preparing customer accounts for conversion \(Subscription Preservation Mode\)](#) (page 158). This step is performed on the Odin Automation side.
- 3 [Performing conversion \(Subscription Preservation Mode\)](#) (page 161). This step is performed on the Microsoft side.
- 4 [Completing conversion \(Subscription Preservation Mode\)](#) (page 162). This step is performed on the Odin Automation side.

## Selecting Customer Accounts (Subscription Preservation Mode)

Select customer accounts for conversion:

### 1. Selecting Customer Accounts for Conversion

Prepare a group of customer accounts whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert. You can use any suitable method to prepare the group, but consider the following:

- All customer accounts of the group must belong to the same vendor account. For example, customer accounts of one reseller account and customer accounts of another reseller account must not be in the same group.
- All Office 365 subscriptions of all customer accounts constituting the group must belong to the same APS version. This means that customer accounts with Office 365 APS 1.2 subscriptions and customer accounts with Office 365 APS 2 subscriptions must not be in the same group.
- All Office 365 subscriptions of all customer accounts constituting the group must belong to the same syndication partner account. This means that customer accounts with Office 365 subscriptions of one syndication partner account and customer accounts with Office 365 subscriptions of another syndication partner account must not be in the same group.
- There might be several organization units for one syndication partner account:
  - Organization units might be represented by separate 'Application' APS 1.2 resource types.
  - Organization units might be represented by separate APS 2 application instances.

Customer accounts with Office 365 subscriptions of one organization unit and customer accounts with Office 365 subscriptions of another organization unit must not be in the same group.

- The number of customer accounts per group affects the performance and convenience of conversion. Use small groups of customer accounts (for example 100-500) until you find a suitable number of customer accounts.

- Office 365 subscriptions including 'DNS Hosting' and non-Office 365 resources are automatically processed by the conversion scripts only if the Office 365 CSP service plans to which the Office 365 subscriptions will be switched include the same 'DNS Hosting' and non-Office 365 resources. We recommend you to make sure that all Office 365 CSP service plans include all necessary resources.
- The conversion scripts process Office 365 subscriptions which are **Active/Running** (in OA Billing) and ignore Office 365 subscriptions which are **Terminated/Removed** (in OA Billing). We recommend you to make sure that all Office 365 subscriptions of the customer accounts constituting the group are either **Active/Running** or **Terminated/Removed**.
- The identifiers of the customer accounts constituting the group must be placed to a CSV file:
  - The customer account identifiers must be specified in the **CustomerId** column.
  - The CSV file can contain other auxiliary columns. These columns will be ignored by the scripts.
  - The file must be uploaded to the host prepared for running the scripts.

You can use the following instructions as examples of how to prepare a group of customer accounts:

- 1** Getting a list of customer accounts of a vendor:
  - a** Log in to the control panel of the vendor.
  - b** Go to **Operations > Customers** and select the **Direct Customers** tab.
  - c** Click the **Select Columns** link and select necessary columns including the **Account ID** column.
  - d** Click the **Export to Excel** link and save the file.
  - e** Open the file in Excel and rename the **Account ID** column to **CustomerId**.
  - f** Save the file as a CSV file.
- 2** Getting a list of customer accounts subscribed to a service plan:
  - a** Log in to the control panel of a vendor.
  - b** Go to **Products > Service Plans** and select the service plan.
  - c** Select the **Subscriptions** tab.
  - d** Click the **Select Columns** link and select necessary columns including the **Customer Account ID** column.
  - e** Click the **Export to Excel** link and save the file.
  - f** Open the file in Excel and rename the **Customer Account ID** column to **CustomerId**.
  - g** Save the file as a CSV file.
- 3** Getting a list of customer accounts with subscriptions of a service template:
  - a** Log in to the control panel of a vendor.

- b** Go to **Products > Service Plans** and select the **Service Templates** tab.
- c** Select the service template.
- d** Select the **Subscriptions** tab.
- e** Click the **Select Columns** link and select necessary columns including the **Customer Account ID** column.
- f** Click the **Export to Excel** link and save the file.
- g** Open the file in Excel and rename the **Customer Account ID** column to **CustomerId**.
- h** Save the file as a CSV file.

**Important:** If you need to prepare a list of customer accounts of a reseller account, you must open the reseller control panel. This can be done in the following way: log in to the provider control panel; go to **Operations > Resellers**; click on the reseller account; click on the **To Control Panel** link on the **General** tab.

## 2. Preparing Application Information

After preparing the group, prepare and write down the following:

- Office 365 APS 1.2 (Syndication Partner Program)
  - The identifier of the source Office 365 APS 1.2 application to which the group belongs. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; see the **ID** column.
  - The URL of the gateway application of the 'Application' resource type to which the group belongs. To obtain the URL, perform the following actions: log in to the provider control panel; go to **Products > Resources > APPLICATION\_RESOURCE\_TYPE > Activation Parameters** tab > **Global Settings** subtab; use the value of the **URL for MOSI Connector** global setting.
  - The email address of an administrative user of the syndication partner account to which the group belongs. Note the following: users of the syndication partner account can be seen in the Partner Center; the user must have the **Global admin** role in the Partner Center.
- Office 365 APS 2 (Syndication Partner Program)
  - The identifier of the source application instance to which the group belongs. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; click on the **Office 365** application; select the **Instances** tab; see the **ID** column.
  - The email address of an administrative user of the syndication partner account to which the group belongs. Note the following: users of the syndication partner account can be seen in the Partner Center; the user must have the **Global admin** role in the Partner Center.
- Office 365 APS 2 (CSP Program)

- The identifier of the target application instance. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; click on the **Office 365** application; select the **Instances** tab; use the **ID** column.
- The Microsoft ID of the CSP partner account of the target application instance. The Microsoft ID can be obtained at <https://partnercenter.microsoft.com/en-us/pcv/accountsettings/organizationprofile>.

This information will be used in further steps of the conversion procedure.

### Preparing Customer Accounts (Subscription Preservation Mode)

The customer accounts you selected in the previous step of the conversion procedure must be prepared.

The following preparation activities are performed by the script `prepareMOSI2CSP.py` for each customer tenant of every customer account:

- 1 The customer tenant is validated against the following requirements:
  - The customer tenant must not have trial syndication cloud subscriptions.
  - The customer tenant must not have syndication cloud subscriptions which cannot be converted into CSP cloud subscriptions.
  - There must not be pending or failed tasks related to the customer tenant.
  - Office 365 subscriptions of the customer tenant must be either **Active/Running** (in OA Billing) or **Terminated/Removed** (in OA Billing).

Validation is necessary to identify if a customer tenant is ready for conversion. If a customer tenant does not pass validation, this means that it is not ready for conversion and further preparation activities are not performed for it.

- 2 MOSI API communication is disabled for the customer tenant. This is needed to protect customer tenants and their cloud subscriptions from changes made in Odin Automation.
- 3 The MOSI Customer ID of the customer tenant is stored. The MOSI Customer ID is sent to Microsoft to convert the customer tenant and its cloud subscriptions on the Microsoft side.
- 4 The mapping between the customer account and its customer tenant is stored. This mapping will be used to find the customer account of the customer tenant in further steps of the conversion procedure.

### Parameters of `prepareMOSI2CSP.py` Script

Note that the `--mosi-app-id` and `--mosi-gateway-url` parameters and the `--mosi-app-instance-id` parameter are mutually exclusive.

- `--conversion-mode` – It defines the mode of conversion. You must specify the `preserveSubscriptionId` value.

- `--on-hold` – It defines if Office 365 subscriptions are put on administrative hold. You must specify either **yes** or **no**.

Note the following:

- The same value must be used in further steps of the conversion procedure.
- We recommend using **yes** to prevent customers from placing upgrade and downgrade orders during conversion.
- There is a drawback related to setting `--on-hold` to **yes**. This issue and its workaround are described at <https://kb.odin.com/en/131023>.
- `--mosi-app-id` – It defines the identifier of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--mosi-gateway-url` – It defines the URL of the gateway application of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--mosi-app-instance-id` – It defines the identifier of a source Office 365 APS 2 application instance (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer accounts which must be prepared for conversion. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.
- `--out-file` – It defines the name of a CSV file containing a list of customer tenants which are ready for conversion. The CSV file is created by the `prepareMOSI2CSP.py` script. The file is sent to Microsoft to convert the customer tenants on the Microsoft side.
- `--error-file` – It defines the name of a CSV file containing a list of customer tenants which are not ready for conversion. The CSV file is created by the `prepareMOSI2CSP.py` script. If the parameter is not specified, the default name `error_YYYYMMDD-HHMMSS.csv` is used.

### Input Files of `prepareMOSI2CSP.py` Script

The `prepareMOSI2CSP.py` script uses an input CSV file containing a list of customer accounts which must be prepared for conversion. Each row of the CSV file contains the following data of a customer account:

- **CustomerId** – The identifier of the customer account.
- other auxiliary columns (optional) – These columns are ignored by the script.

## Output Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` script creates an output CSV file containing a list of customer tenants which are ready for conversion. The name of the CSV file is defined by the `--out-file` parameter of the script. Each row of the CSV file contains the following data of a customer tenant:

- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.

The script creates an output CSV file containing a list of customer tenants which are not ready for conversion. The name of the CSV file is defined by the `--error-file` parameter of the script; if the parameter is not specified, the default name `error_YYYYMMDD-HHMMSS.csv` is used. Each row of the CSV file contains the following data of a customer tenant:

- **Aps** – The APS version of the Office 365 application to which the customer tenant belongs.
- **mosiAppldOrMosiApplInstanceid** – It contains one of the following: the identifier of the Office 365 APS 1.2 application to which the customer tenant belongs; the identifier of the Office 365 APS 2 application instance to which the customer tenant belongs.
- **CustomerId** – The identifier of the customer account to which the customer tenant belongs.
- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.
- **MSTenantId** – The Microsoft Tenant ID of the customer tenant.
- **SubscriptionId** – The identifier of an Office 365 subscription which blocks the conversion of the customer tenant.
- **Error** – An error message.

## Log and Auxiliary Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` script creates the following log and auxiliary files:

- `prepareMOSI2CSP_YYYYMMDD-HHMMSS.log` – A log file containing general messages, such as information messages and error messages.
- `mosi2csp_mapping.csv` – A CSV file containing mappings between customer accounts and their customer tenants. The file is auxiliary and must not be modified or renamed. The file will be used by the `completeMOSI2CSP.py` script in further steps of the conversion procedure.

## Running prepareMOSI2CSP.py Script

Log in to the host where the conversion scripts are placed. Run the `prepareMOSI2CSP.py` script.

Example (Office 365 APS 1.2 / Syndication Partner Program):



```
python prepareMOSI2CSP.py --conversion-mode preserveSubscriptionId --on-hold yes
--mosi-app-id 100 --mosi-gateway-url
https://office365.provider.com/Office365GatewayApplication/HTTPEndpoint.aspx --
csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --error-
file errors.csv
```

Example (Office 365 APS 2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode preserveSubscriptionId --on-hold yes
--mosi-app-instance-id 999 --csp-app-instance-id 1000 --pba --input-file
input.csv --out-file out.csv --error-file errors.csv
```

After running the script, check the output CSV file containing a list of customer tenants which are not ready for conversion. If the file is not empty, you have the following options:

- Investigate and fix the errors. Run the script again using the same values of parameters. Repeat this until all customer tenants are ready for conversion.
- Postpone processing the customer tenants which are not ready for conversion and process them later. Note that you will need to start the conversion procedure from the beginning for the postponed customer tenants.

### Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

### Performing Conversion (Subscription Preservation Mode)

The customer tenants you prepared in the previous step of the conversion procedure must be converted on the Microsoft side:

- 1 Fill out a conversion request form:
  - a Download the conversion request form template provided at <https://kb.odin.com/en/130124>.
  - b Fill out the form using the information you prepared earlier:
    - The email address of the administrative user of the required syndication partner account.
    - The Microsoft ID of the required CSP partner account.
    - The CSV file containing the list of the customer tenants.
- 2 Send the file of the form to [csptrans@microsoft.com](mailto:csptrans@microsoft.com).
- 3 Microsoft will perform conversion on its side and provide you with two CSV files:
  - A CSV file containing a list of customer tenants which were successfully converted on the Microsoft side.
  - A CSV file containing a list of customer tenants which cannot be converted on the Microsoft side.
- 4 Check that the CSV files received from Microsoft have the **MosiCustomerId** column.
- 5 Upload both the CSV files to the host for running the conversion scripts.

The CSV files received from Microsoft will be used in further steps of the conversion procedure.

### Completing Conversion (Subscription Preservation Mode)

Complete the conversion of the customer tenants processed by Microsoft in the previous step of the conversion procedure:

- Use the instruction of the [Processing Customer Tenants Converted by Microsoft \(Subscription Preservation Mode\) section](#) (page 162) to process the customer tenants converted by Microsoft.
- Use the instruction of the [Processing Customer Tenants Rejected by Microsoft \(Subscription Preservation Mode\) section](#) (page 168) to process the customer tenants rejected by Microsoft.

### Processing Customer Tenants Converted by Microsoft (Subscription Preservation Mode)

The customer tenants converted on the Microsoft side in the previous step of the conversion procedure must be processed on the Odin Automation side.

The following activities are performed for every customer tenant:

- 1** Suitable Office 365 APS 2 service plans (CSP Program) are found for the customer account of the customer tenant. This means that all cloud subscriptions of the customer tenant will be represented by the corresponding Office 365 APS 2 subscriptions (CSP Program) in Odin Automation. This activity is performed by using the `completeMOSI2CSP.py` script.

Note the following:

- If suitable Office 365 APS 2 service plans (CSP Program) are not found for the customer account of a customer tenant, further activities are not performed for the customer tenant.
  - Office 365 APS 2 service plans (CSP Program) containing composite resources are supported by the script. If each composite resource of an Office 365 APS 2 service plan (CSP Program) contains one included resource and the amount of the included resource is equal to **1**, the Office 365 APS 2 service plan (CSP Program) is taken into account by the script to find suitable Office 365 APS 2 service plans (CSP Program) for cloud subscriptions.
  - Both active cloud subscriptions and suspended cloud subscriptions are taken into account by the script.
- 2** If all Office 365 subscriptions of the customer tenant are on administrative hold, they are released from administrative hold. This activity is performed by a conversion task which is scheduled by the `completeMOSI2CSP.py` script.

- 3 All Office 365 subscriptions of the customer tenant are switched from Office 365 service plans (Syndication Partner Program) to the Office 365 APS 2 service plans (CSP Program) found earlier. This activity is performed by using the `importSubsCSP.py` script.

### Parameters of `completeMOSI2CSP.py` Script

- `--conversion-mode` – It defines the mode of conversion. You must specify the `preserveSubscriptionId` value.
- `--on-hold` – Specify **yes** if Office 365 subscriptions are on administrative hold. Otherwise, specify **no**.
- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the first step of the conversion procedure.
- `--mosi-gateway-url` – It defines the URL of the gateway application of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared earlier. Note that this parameter must be specified only if Office 365 APS 1.2 subscriptions are converted.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer tenants converted by Microsoft. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.
- `--out-file` – It defines the name of a CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The CSV file is created by the `completeMOSI2CSP.py` script. The CSV file will be used by the `importSubsCSP.py` script.
- `--error-file` – It defines the name of a CSV file containing a list of customer tenants which cannot be processed. The CSV file is created by the `completeMOSI2CSP.py` script. If the parameter is not specified, the default name `errors_completion_YYYYMMDD-HHMMSS.csv` is used.

### Input Files of `completeMOSI2CSP.py` Script

The `completeMOSI2CSP.py` script uses an input CSV file containing a list of customer tenants converted by Microsoft. Each row of the CSV file contains the following data of a cloud subscription of a customer tenant:

- **CustomerName** – The Business Name of the customer tenant.
- **CustomerTenantId** – The Microsoft Tenant ID of the customer tenant.
- **FpoCaid** – The Microsoft Internal Commerce ID of the customer tenant.
- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.
- **MosiSubscriptionId** – The MOSI subscription identifier of the cloud subscription.

- **CspSubscriptionId** – The CSP subscription identifier of the cloud subscription.
- **CspSubscriptionOfferId** – The CSP Offer ID of the cloud subscription.
- **CspSubscriptionStartDate** – The start date of the cloud subscription.
- **CspSubscriptionEndDate** – The end date of the cloud subscription.

Note that the `completeMOSI2CSP.py` script uses the **MosiCustomerId** column and ignores the other columns.

### **Output Files of completeMOSI2CSP.py Script**

The `completeMOSI2CSP.py` script creates an output CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The name of the CSV file is defined by the `--out-file` parameter of the script. Each row of the CSV file contains the following data of a customer tenant:

- **OSACustomerID** – The identifier of the customer account of the customer tenant.
- **OSAServicePlanId** – The identifier of an Office 365 APS 2 service plan (CSP Program) to which the cloud subscriptions specified in the **MSSubscriptionIDs** column are mapped by the script.
- **OSAPlanPeriodID** – The identifier a subscription period of the Office 365 APS 2 service plan (CSP Program). The subscription period is chosen by the script.
- **SubscriptionStartDate** – The start date of the Office 365 subscription specified in the **SourceSubscriptionId** column.
- **ResourceRateLimits** – A list of resource rates and their limits. The resource rates are chosen by the script for the cloud subscriptions specified in the **MSSubscriptionIDs** column. The format "**RR\_ID1-LIMIT,RR\_ID2-LIMIT**" is used for the list. For example: "**100-10,200-20**".
- **CSPAccountId** – The Microsoft Tenant ID of the customer tenant.
- **MSSubscriptionIDs** – A list of cloud subscriptions of the customer tenant. The format "**MS\_SUB\_ID1,MS\_SUB\_ID2,...**" is used for the list. For example: "**A49C6781-1557-42CA-A074-52D0553397EA,EBE5106C-B1ED-44FF-8CE7-A2746AF8BC42**".
- **DefaultAdminLogin** – The login name of the default administrative user of the customer tenant.
- **DefaultDomainName** – The default domain name of the customer tenant.
- **SourceSubscriptionId** – The identifier of the Office 365 subscription which will be switched from an Office 365 service plan (Syndication Partner Program) to the Office 365 APS 2 service plan (CSP Program) specified in the **OSAServicePlanId** column.

Note that the CSV file might contain several rows for one customer tenant if the customer tenant has several Office 365 subscriptions.

The `completeMOSI2CSP.py` script creates an output CSV file containing a list of customer tenants whose cloud subscriptions cannot be mapped to Office 365 APS 2 service plans (CSP Program). The name of the CSV file is defined by the `--error-file` parameter of the script; if the parameter is not specified, the default name `errors_completion_YYYYMMDD-HHMMSS.csv` is used. Each row of the CSV file contains the following data of a customer tenant:

- **OSACustomerID** – The identifier of the customer account of the customer tenant.
- **CSPAccountID** – The Microsoft Tenant ID of the customer tenant.
- **SubscriptionStartDate** – The start date of the cloud subscriptions specified in the **MSSubscriptionIDs** column.
- **MSSubscriptionIDs** – A list of cloud subscriptions of the customer tenant. The format `"MS_SUB_ID1,MS_SUB_ID2,..."` is used for the list. For example: `"A49C6781-1557-42CA-A074-52D0553397EA,EBE5106C-B1ED-44FF-8CE7-A2746AF8BC42"`.
- **MSSubscriptionOfferIDs** – The identifiers of the offers of the cloud subscriptions specified in the **MSSubscriptionIDs** column.
- **Comments** – An error message.

### **Log and Auxiliary Files of completeMOSI2CSP.py Script**

The `completeMOSI2CSP.py` script creates a log file with the name `completeMOSI2CSP_YYYYMMDD-HHMMSS.log`. The file contains general messages, such as information messages and error messages.

The `completeMOSI2CSP.py` script implicitly uses the `mosi2csp_mapping.csv` file created by the `prepareMOSI2CSP.py` script. The CSV file contains mappings between customer accounts and their customer tenants. The file is auxiliary and must not be modified or renamed.

### **Running completeMOSI2CSP.py Script**

- 1 Log in to the host where the conversion scripts are placed.
- 2 Run the `completeMOSI2CSP.py` script.
- 3 Check the output CSV file containing a list of customer tenants whose cloud subscriptions cannot be mapped to Office 365 APS 2 service plans (CSP Program). If the CSV file is not empty, investigate and fix the errors. Run the script again using the same values of parameters. Repeat this until all customer tenants are processed.
- 4 Check the output CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program):
  - a Make sure that mappings are correct.
  - b Make sure that resource rates and their limits are correct.

Modify the CSV file if necessary. For instance, you can specify your own mappings between Office 365 APS 2 service plans and cloud subscriptions.

Example (Office 365 APS 1.2 / Syndication Partner Program):

```
python completeMOSI2CSP.py --conversion-mode preserveSubscriptionId --on-hold yes
--mosi-gateway-url
https://office365.provider.com/Office365GatewayApplication/HTTPEndpoint.aspx --
csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --error-
file error.csv
```

Example (Office 365 APS 2 / Syndication Partner Program):

```
python completeMOSI2CSP.py --conversion-mode preserveSubscriptionId --on-hold yes
--csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --
error-file error.csv
```

### Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

### Parameters of importSubsCSP.py Script

- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the first step of the conversion procedure.
- The name of a CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). You must use the CSV file created by the `completeMOSI2CSP.py` script.

### Input Files of importSubsCSP.py Script

The `importSubsCSP.py` script uses an input CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The CSV file is created by the `completeMOSI2CSP.py` script. The format of the CSV file is described above.

### Output Files of importSubsCSP.py Script

The `importSubsCSP.py` script creates a CSV file with the name `result_<input file name>`. Each row of the CSV file contains the following data of an Office 365 CSP subscription processed by the `importSubsCSP.py` script:

- **CustomerId** – The identifier of the customer account to which the subscription belongs.
- **SubscriptionId** – The Odin Automation identifier of the Office 365 CSP subscription.

### Log and Auxiliary Files of importSubsCSP.py Script

The `importSubsCSP.py` script creates a CSV file with the name `errors_import_<input file name>`. This file contains a list of cloud subscriptions which cannot be processed by the `importSubsCSP.py` script as well as reasons why the cloud subscriptions cannot be processed. The CSV file has the same format as the input file of the `importSubsCSP.py` script (except for the additional **Error** column, which contains error messages).

The `importSubsCSP.py` script creates a CSV file with the name `reimport_<input file name>`. This file contains a list of cloud subscriptions which cannot be processed by the `importSubsCSP.py` script. The CSV file has the same format as the input file of the `importSubsCSP.py` script. You can use the file `reimport_<input file name>` to run the `importSubsCSP.py` script again only for affected cloud subscriptions.

The `importSubsCSP.py` script creates a log file with the name `importCSPYYYY-MM-DD.log`. The file contains general messages, such as information messages and error messages.

### Running importSubsCSP.py Script

- 1 Stop Office 365 synchronization:
  - a In Provider Control Panel, go to **Operations > Tasks**.
  - b Select the **Periodic** tab.
  - c Cancel all the **Office 365 \* Synchronization with Office 365 Portal** tasks.
- 2 Log in to the host where the conversion scripts are placed.
- 3 Run the `importSubsCSP.py` script.
- 4 After running the script, Office 365 import tasks will be scheduled in the task manager. Wait until they are processed.
- 5 Start Office 365 synchronization:
  - a In Provider Control Panel, go to **Operations > Tasks**.
  - b Select the **Periodic** tab.
  - c Run all the **Office 365 \* Synchronization with Office 365 Portal** tasks.

Example:

```
python importSubsCSP.py --csp-app-instance-id 1000 input.csv
```

After running the script, perform the following validation steps for necessary customer tenants:

- 1 Check that the customer account of the customer tenant has correct Office 365 APS 2 subscriptions (CSP Program).
- 2 Check that the limits of the Office 365 APS 2 subscriptions match the limits of the cloud subscriptions of the customer tenant.

### Troubleshooting

- Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.
- If the `importSubsCSP.py` script fails with the A duplicated entry is found for the SA Customer #CustomerId and Service Plan #PlanId in the line LineNumber error, consider using the **--allow-duplicates** option described at <https://kb.odin.com/en/128798>.

## Processing Customer Tenants Rejected by Microsoft (Subscription Preservation Mode)

The customer tenants rejected by Microsoft in the previous step of the conversion procedure must be processed on the Odin Automation side.

The following activities are performed by the `prepareMOSI2CSP.py` script for every customer tenant:

- 1 If all Office 365 subscriptions of the customer tenant are on administrative hold, they are released from administrative hold.
- 2 MOSI API communication is enabled for the customer tenant.

### Parameters of `prepareMOSI2CSP.py` Script

Note that the `--mosi-app-id` and `--mosi-gateway-url` parameters and the `--mosi-app-instance-id` parameter are mutually exclusive.

- `--conversion-mode` – It defines the mode of conversion. You must specify the `preserveSubscriptionId` value.
- `--rollback` – It defines if the script must roll back changes made for customer tenants. You must specify the parameter to process customer tenants rejected by Microsoft.
- `--mosi-app-id` – It defines the identifier of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--mosi-gateway-url` – It defines the URL of the gateway application of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--mosi-app-instance-id` – It defines the identifier of a source Office 365 APS 2 application instance (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer tenants rejected by Microsoft. The CSV file is received from Microsoft. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.

### Input Files of `prepareMOSI2CSP.py` Script

The `prepareMOSI2CSP.py` uses an input CSV file containing a list of customer tenants rejected by Microsoft. The `prepareMOSI2CSP.py` script uses the `MosiCustomerId` column and ignores other columns.

### Log and Auxiliary Files of `prepareMOSI2CSP.py` Script



The `prepareMOSI2CSP.py` script creates a log file with the name `prepareMOSI2CSP_YYYYMMDD-HHMMSS.log`. The log file contains general messages, such as information messages and error messages.

### Running prepareMOSI2CSP.py Script

Log in to the host where the conversion scripts are placed. Run the `prepareMOSI2CSP.py` script.

Example (Office 365 APS 1.2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode preserveSubscriptionId --rollback --
mosi-app-id 100 --mosi-gateway-url
https://office365.provider.com/Office365GatewayApplication/HTTPEndpoint.aspx --
pba --input-file input.csv
```

Example (Office 365 APS 2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode preserveSubscriptionId --rollback --
mosi-app-instance-id 999 --pba --input-file input.csv
```

After running the script, you can analyze the customer tenants rejected by Microsoft to find out the reason for rejection. Note that you will need to start the conversion procedure from the beginning for rejected customer tenants after fixing them.

### Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

## Advanced Subscription Split Mode

### Overview (Advanced Subscription Split Mode)

Conversion from the Syndication Partner Program to the CSP Program is performed on both the Microsoft side and the Odin Automation side.

#### Microsoft Side

The Microsoft side consists of two layers:

- **A service layer.** The service layer is responsible for the provisioning of Office 365 services. All service data, such as customer tenant settings, end-user settings, domain settings, the content of mailboxes, and so forth, is stored and managed at this level.

- **A commercial layer.** The commercial level is responsible for tracking financial information of Office 365 services. All financial information, such as cloud subscriptions of customer tenants, service prices, cloud subscription limits, and so forth, is stored and managed at this level. When a commercial operation is performed, corresponding changes are propagated to the service layer. There are many commercial systems at the commercial layer, including the commercial system of the Syndication Partner Program and the commercial system of the CSP Program.

Conversion on the Microsoft side is the transportation of financial information from the commercial system of the Syndication Partner Program to the commercial system of the CSP Program.

Key points of conversion:

- The service data of customer tenants is not modified during the conversion of the customer tenants. Daily operations of end-users of the customer tenants are not affected.
- The Office 365 admin center of the CSP Program is slightly different from the Office 365 admin center of the Syndication Partner Program.
- The offer catalogue of the CSP Program and the offer catalogue of the Syndication Partner Program are different:
  - Both programs have a set of common offers, and conversion is possible only within the set. This means that cloud subscriptions of other offers must be either replaced with cloud subscriptions of common offers or cancelled before conversion.
  - The CSP Program does not support trial offers. This means that trial cloud subscriptions cannot be converted. Trial cloud subscriptions must be either replaced with paid ones or cancelled before conversion.
- The programs provide different APIs for managing customer tenants and cloud subscriptions. The Syndication Partner Program provides the MOSI API. The CSP Program provides the Partner Center API.
- The CSP Program is globally divided into geographical regions (the regions can be seen at <https://www.microsoft.com/en-us/licensing/licensing-programs/open-regional.aspx>). CSP partners can sell Office 365 services only within their regions. For example, a CSP partner from the North America region can sell Office 365 services to customers only from the United States and Canada. This means that several CSP partner accounts might be required for a syndication partner who sells Office 365 services in various countries.
- **TIP partner accounts cannot be used to test conversion.**

To test the whole conversion procedure before performing actual conversion, create temporary customer tenants and temporary cloud subscriptions through Odin Automation connected to both a production syndication partner account and a production CSP partner account.
- Before performing conversion, a partner must contact a Microsoft representative to discuss and approve the volume and schedule of conversion.

### **Odin Automation Side**

A partner can be represented in one of the following ways:

- A syndication partner has one or more Office 365 APS 1.2 'Application' resource types depending on the number of organization units the partner uses for selling Office 365 services. Each organization unit has its own set of Office 365 APS 1.2 syndication service plans, which include not only Office 365 resources but also resources of other services (non-Office 365 resources).
- A syndication partner is represented by one or more Office 365 APS 2 application instances depending on the number of organization units the partner uses for selling Office 365 services. Each organization unit has its own set of Office 365 service plans, which include not only Office 365 resources but also resources of other services (non-Office 365 resources).
- A CSP partner is represented by an Office 365 APS 2 application instance. The CSP partner has one set of Office 365 service plans.

Conversion on the Odin Automation side is the transformation of Office 365 syndication subscriptions including both Office 365 resources and non-Office 365 resources into Office 365 CSP subscriptions and subscriptions including only non-Office 365 resources. Namely, an Office 365 syndication subscription is transformed in the following way: If the subscription has non-Office 365 resources with non-zero limits, the subscription is switched to a service plan including only non-Office 365 resources and then new Office 365 CSP subscriptions are created. If the subscription has only Office 365 resources or all non-Office 365 resources of the subscription have zero limits, the subscription is cancelled and then new Office 365 CSP subscriptions are created.

Key points of conversion:

- The Office 365 APS 1.2 and Office 365 APS 2 applications are supported.
- Office 365 syndication subscriptions can contain any non-Office 365 resources with any limits.
- A syndication partner might have several organization units for selling Office 365 services. This means that a separate CSP partner account is required for each organization unit.
- Odin Automation supports the following configurations of Office 365 service plans:
  - **Consolidated.** All offers are available within an Office 365 service plan. A typical customer account has one subscription to such an Office 365 service plan.
  - **Partitioned.** All offers are partitioned into groups. A typical group contains one offer and its depended addon offers. Offers constituting a group are available only within a separate Office 365 service plan. A typical customer account has several subscriptions to such Office 365 service plans.

Cloud subscriptions and their limits are controlled by setting limits of offer resources of Office 365 subscriptions. If the limit of an offer resource is equal to zero, the corresponding cloud subscription either does not exist or is suspended. If the limit of an offer resource is equal to a positive value, the corresponding cloud subscription exists and is active; the limit of the cloud subscription is equal to the limit of the offer resource.

It is possible to change one service plan configuration to another one during conversion. If there is a choice between 'consolidated' and 'partitioned' Office 365 APS 2 CSP service plans when Office 365 syndication subscriptions of a customer tenant are transformed, the 'consolidated' service plans take precedence over the 'partitioned' service plans.

- Conversion is performed by using scripts shipped with the **Office 365** application.
- Customer accounts whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert must be split into groups. You must convert these groups one by one. This means that several groups must not be converted concurrently.

### Limitations (Advanced Subscription Split Mode)

The conversion procedure has the following limitations:

- During conversion, a subscription including both Office 365 resources and non-Office 365 resources is switched to a service plan including only non-Office 365 resources. As a result, a change order with a non-zero refund is created. To disable refunds in such change orders, you can use the method described at <https://kb.odin.com/en/118910>. Note that this method is system-wide and affects all operations of switching service plans.
- For an Office 365 CSP subscription created with the `importSubsCSP.py` script, all resource downgrade orders placed within the current billing period will contain zero resource prices and will not include order details. As a result, the customer placing the orders will not receive any refunds. When the current billing period ends and the first billing order is automatically placed for the Office 365 subscription, further resource downgrade orders will work as expected.

To calculate a refund for a resource downgrade order of a subscription, Odin Automation uses orders placed earlier. Since the script creates an Office 365 subscription without placing any order, Odin Automation does not have the necessary information to calculate the refund for a resource downgrade order of the Office 365 subscription until the first billing order is automatically placed, which will happen when the current billing period ends.

We recommend that you plan conversion according to these specific circumstances.

### Prerequisites (Advanced Subscription Split Mode)

Before performing conversion, the following prerequisites must be met on your Odin Automation installation:

- Odin Automation is installed:
  - 7.3.0 or a later 7.3.x version
  - 7.2.0 or a later 7.2.x version
  - 7.1.0 or a later 7.1.x version
  - 7.0.1\_hf130748 or a later 7.0.x version

- Office 365 (Syndication Partner Program) 16.8.6 or higher (APS 2) / 4.17 or higher (APS 1.2) is installed.

**Important:** For Office 365 (Syndication Partner Program) 4.17, the instructions of the KB article <https://kb.odin.com/en/130292> must be applied.

- Office 365 (CSP Program) 16.8.6 or higher is installed.
- All trial syndication Office 365 subscriptions are either converted to paid ones or cancelled.
- All legacy syndication Office 365 subscriptions are either converted to modern ones or cancelled.

## Preparing for Conversion (Advanced Subscription Split Mode)

This section describes how to prepare your Odin Automation installation for conversion.

The preparation procedure consists of the following steps:

- 1 [Preparing your installation \(Advanced Subscription Split Mode\)](#) (page 173).
- 2 [Configuring customer notification messages \(Advanced Subscription Split Mode\)](#) (page 174).
- 3 [Preparing conversion scripts \(Advanced Subscription Split Mode\)](#) (page 175).

## Preparing Your Installation (Advanced Subscription Split Mode)

On your Odin Automation installation, configure necessary Office 365 APS 2 CSP application instances and service plans:

- Each Office 365 APS 1.2 'Application' resource type must be represented by an Office 365 APS 2 CSP application instance (if you have Office 365 APS 1.2 on your installation).
- Each Office 365 APS 2 syndication application instance must have an Office 365 APS 2 CSP counterpart (if you have Office 365 APS 2 on your installation).
- Under each vendor account selling Office 365 services, the following must be done for every Office 365 syndication service plan:
  - Necessary Office 365 APS 2 CSP service plans are created so that syndication offers can be replaced with CSP offers.

For example: if there are syndication service plans which include resources of the syndication offers **Office 365 Business Essentials** and **Office 365 Enterprise E3**, there must be CSP service plans which include resources of the CSP offers **Office 365 Business Essentials** and **Office 365 Enterprise E3**.

- A separate service plan including all the non-Office 365 resources of the Office 365 syndication service plan is created.

For example: if a syndication service plan and its service template include non-Office 365 resources, the separate service plan and its service template must include the same non-Office 365 resources; the resource rates of the non-Office 365 resources of the separate service plan must have the same settings as the resource rates of the non-Office 365 resources of the syndication service plan have (**Inc. Amount, Max Amount, Recurring Fee**, and so forth).

- The separate service plan including the non-Office 365 resources is added to the list of allowed upgrades of the Office 365 syndication service plan. Also, the list must not contain other service plans.
- All the Office 365 APS 2 CSP service plans must have one subscription period. This is necessary since the conversion scripts do not take into account Office 365 APS 2 CSP service plans with several subscription periods.
- In each Office 365 APS 2 CSP service plan, **Show in Store** must be set to **Yes** for all Office 365 resource rates.

### Notifying Customers (Advanced Subscription Split Mode)

We recommend that you notify customers whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert. You can use any suitable method to notify the customers.

You can use the instructions of the section as guidelines of how to configure notification.

#### Configuring 'Upcoming Conversion to CSP' Notification Messages

- 1 Create a notification template using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75484.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75484.htm).

Set **Message Type** to **Account Notification**.

- 2 Create a distribution list using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75977.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75977.htm).

Do the following:

- a Specify the name of the notification template you created in the **Notification Template** field.
  - b Click the **Send to All** button.
- 3 After creating the distribution list, add a service plan criterion for each Office 365 syndication service plan:
    - a Set **Operator** to **Or**.
    - b Specify the Office 365 syndication service plan in the **Service Plan** field.

Send the notification messages to the distribution list when necessary.

#### Configuring 'Conversion to CSP Has Started' Notification Messages

Depending on the `--on-hold` parameter of the conversion scripts, Office 365 syndication subscriptions of customer tenants can be put on administrative hold during conversion. If Office 365 syndication subscriptions are put on administrative hold, these events can be used to trigger sending notification messages.

- 1 Create a notification template using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75484.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75484.htm).

Do the following:

- a Set **Message Type** to **Subscription Notification**.
  - b Specify `@@Subscription_Plan_ID@~="^(ID1)|(ID2)|(ID3)...$"` in the **Condition** field.
  - c Replace **IDx** with the identifiers of all Office 365 syndication service plans.
- 2 Add an event handler to the **Subscription Put On Administrative Hold** event using the instructions at [https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing\\_providers\\_guide/index.htm?fileName=75997.htm](https://download.automation.odin.com/oa/7.0/oapremium/portal/en/billing_providers_guide/index.htm?fileName=75997.htm).

Do the following:

- a Set **Object** to **MESSAGE**.
- b Set **Method** to **SubscriptionNotification**.
- c Set **Signature** to **integer subscriptionID, string TemplateName=NAME\_OF\_NOTIFICATION\_TEMPLATE**. Replace **NAME\_OF\_NOTIFICATION\_TEMPLATE** with the name of the notification template you created.
- d Set **Protocol** to **Stellart**.
- e Set **Max Attempts** to **1**.
- f Set **Repeat interval (seconds)** to **1**.
- g Set **Synchronous** to **No**.

## Preparing Conversion Scripts (Advanced Subscription Split Mode)

Conversion is performed by using the scripts `prepareMOSI2CSP.py`, `completeMOSI2CSP.py`, and `importSubsCSP.py`.

Before performing conversion, prepare the scripts and a host for running them:

- 1 Prepare a host which meets the following requirements:
  - Python 2.6.6 is installed.

- The poaupdater RPM package is installed on the host. The RPM package must be taken from the Odin Automation distribution package/YUM repository that belongs to your version of Odin Automation (for example, see the `DISTRIBUTION_PACKAGE/modules/platform/OS/OS_VERSION/RPMS/` directory).
- The pyPgSQL-2.5.1 (or higher) Python package is installed.
- The mx 3.1.1 (or higher) Python package is installed.
- Connections from the host to the OA Operations Public API are allowed.
- Connections from the host to the OA Billing Public API are allowed.
- Connections from the host to the Odin Automation system database are allowed.
- Connections from the host to the APSC are allowed.
- Connections from the host to the Office 365 APS 1.2 application gateway sites are allowed (if you have Office 365 APS 1.2 on your Odin Automation installation).

Note that the Odin Automation Management Node can also be used for running the scripts if it meets the requirements.

- 2 Upload the Office 365 application package to the host.
- 3 Extract the scripts from the package. The scripts are here: `Office365-xx.x.x-xx.app.zip \ 0365-Import.zip`.
- 4 Write down the location of the prepared scripts.

Also, we recommend you to check that the Office 365 Application Endpoint Host (CSP) has access to all the necessary APIs and system components (APSC, OA Operations Public API, OA Billing Public API, Partner Center API, and so on). See the following sections for details: [Deployment Scheme](#) (page 16), [Preparing Odin Automation Public API User](#) (page 31), and [Firewall Requirements](#) (page 97).

**Note:** In case the conversion scripts are not able to connect to the Odin Automation system database or the Odin Automation APIs, check the `__dbhost`, `__dbname`, `__dbuser`, `__dbpass`, `__apihost` parameters in the `Utils.py` file.

## Conversion Procedure (Advanced Subscription Split Mode)

This section describes how to perform conversion.

The conversion procedure consists of the following steps:

- 1 [Selecting customer accounts for conversion \(Advanced Subscription Split Mode\)](#) (page 177). This step is performed on the Odin Automation side.
- 2 [Preparing customer accounts for conversion \(Advanced Subscription Split Mode\)](#) (page 179). This step is performed on the Odin Automation side.



- 3 [Performing conversion \(Advanced Subscription Split Mode\)](#) (page 183). This step is performed on the Microsoft side.
- 4 [Completing conversion \(Advanced Subscription Split Mode\)](#) (page 183). This step is performed on the Odin Automation side.

## Selecting Customer Accounts (Advanced Subscription Split Mode)

Select customer accounts for conversion:

### 1. Selecting Customer Accounts for Conversion

Prepare a group of customer accounts whose Office 365 subscriptions, customer tenants, and cloud subscriptions you need to convert. You can use any suitable method to prepare the group, but consider the following:

- All customer accounts of the group must belong to the same vendor account. For example, customer accounts of one reseller account and customer accounts of another reseller account must not be in the same group.
- All Office 365 subscriptions of all customer accounts constituting the group must belong to the same APS version. This means that customer accounts with Office 365 APS 1.2 subscriptions and customer accounts with Office 365 APS 2 subscriptions must not be in the same group.
- All Office 365 subscriptions of all customer accounts constituting the group must belong to the same syndication partner account. This means that customer accounts with Office 365 subscriptions of one syndication partner account and customer accounts with Office 365 subscriptions of another syndication partner account must not be in the same group.
- There might be several organization units for one syndication partner account:
  - Organization units might be represented by separate 'Application' APS 1.2 resource types.
  - Organization units might be represented by separate APS 2 application instances.

Customer accounts with Office 365 subscriptions of one organization unit and customer accounts with Office 365 subscriptions of another organization unit must not be in the same group.

- The number of customer accounts per group affects the performance and convenience of conversion. Use small groups of customer accounts (for example 100-500) until you find a suitable number of customer accounts.
- The identifiers of the customer accounts constituting the group must be placed to a CSV file:
  - The customer account identifiers must be specified in the **CustomerId** column.
  - The CSV file can contain other auxiliary columns. These columns will be ignored by the scripts.
  - The file must be uploaded to the host prepared for running the scripts.

You can use the following instructions as examples of how to prepare a group of customer accounts:

- 1** Getting a list of customer accounts of a vendor:
  - a** Log in to the control panel of the vendor.
  - b** Go to **Operations > Customers** and select the **Direct Customers** tab.
  - c** Click the **Select Columns** link and select necessary columns including the **Account ID** column.
  - d** Click the **Export to Excel** link and save the file.
  - e** Open the file in Excel and rename the **Account ID** column to **CustomerId**.
  - f** Save the file as a CSV file.
- 2** Getting a list of customer accounts subscribed to a service plan:
  - a** Log in to the control panel of a vendor.
  - b** Go to **Products > Service Plans** and select the service plan.
  - c** Select the **Subscriptions** tab.
  - d** Click the **Select Columns** link and select necessary columns including the **Customer Account ID** column.
  - e** Click the **Export to Excel** link and save the file.
  - f** Open the file in Excel and rename the **Customer Account ID** column to **CustomerId**.
  - g** Save the file as a CSV file.
- 3** Getting a list of customer accounts with subscriptions of a service template:
  - a** Log in to the control panel of a vendor.
  - b** Go to **Products > Service Plans** and select the **Service Templates** tab.
  - c** Select the service template.
  - d** Select the **Subscriptions** tab.
  - e** Click the **Select Columns** link and select necessary columns including the **Customer Account ID** column.
  - f** Click the **Export to Excel** link and save the file.
  - g** Open the file in Excel and rename the **Customer Account ID** column to **CustomerId**.
  - h** Save the file as a CSV file.

**Important:** If you need to prepare a list of customer accounts of a reseller account, you must open the reseller control panel. This can be done in the following way: log in to the provider control panel; go to **Operations > Resellers**; click on the reseller account; click on the **To Control Panel** link on the **General** tab.

## 2. Preparing Application Information

After preparing the group, prepare and write down the following:

- Office 365 APS 1.2 (Syndication Partner Program)
  - The identifier of the source Office 365 APS 1.2 application to which the group belongs. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; see the **ID** column.
  - The URL of the gateway application of the 'Application' resource type to which the group belongs. To obtain the URL, perform the following actions: log in to the provider control panel; go to **Products > Resources > APPLICATION\_RESOURCE\_TYPE > Activation Parameters** tab > **Global Settings** subtab; use the value of the **URL for MOSI Connector** global setting.
  - The email address of an administrative user of the syndication partner account to which the group belongs. Note the following: users of the syndication partner account can be seen in the Partner Center; the user must have the **Global admin** role in the Partner Center.
- Office 365 APS 2 (Syndication Partner Program)
  - The identifier of the source application instance to which the group belongs. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; click on the **Office 365** application; select the **Instances** tab; see the **ID** column.
  - The email address of an administrative user of the syndication partner account to which the group belongs. Note the following: users of the syndication partner account can be seen in the Partner Center; the user must have the **Global admin** role in the Partner Center.
- Office 365 APS 2 (CSP Program)
  - The identifier of the target application instance. To obtain the identifier, perform the following actions: log in to the provider control panel; go to **Services > Applications**; click on the **Office 365** application; select the **Instances** tab; use the **ID** column.
  - The Microsoft ID of the CSP partner account of the target application instance. The Microsoft ID can be obtained at <https://partnercenter.microsoft.com/en-us/pcv/accountsettings/organizationprofile>.

This information will be used in further steps of the conversion procedure.

## Preparing Customer Accounts (Advanced Subscription Split Mode)

The customer accounts you selected in the previous step of the conversion procedure must be prepared.

The following preparation activities are performed by the script `prepareMOSI2CSP.py` for each customer tenant of every customer account:

- 1 The customer tenant is validated against the following requirements:
  - The customer tenant must not have trial syndication cloud subscriptions.

- The customer tenant must not have syndication cloud subscriptions which cannot be converted into CSP cloud subscriptions.
- There must not be pending or failed tasks related to the customer tenant.
- All Office 365 subscriptions of the customer tenant can be switched from their Office 365 syndication service plans including Office 365 resources and non-Office 365 resources to service plans including only non-Office 365 resources.

Validation is necessary to identify if a customer tenant is ready for conversion. If a customer tenant does not pass validation, this means that it is not ready for conversion and further preparation activities are not performed for it.

- 2 MOSI API communication is disabled for the customer tenant. This is needed to protect customer tenants and their cloud subscriptions from changes made in Odin Automation.
- 3 The MOSI Customer ID of the customer tenant is stored. The MOSI Customer ID is sent to Microsoft to convert the customer tenant and its cloud subscriptions on the Microsoft side.
- 4 The mapping between the customer account and its customer tenant is stored. This mapping will be used to find the customer account of the customer tenant in further steps of the conversion procedure.

### Parameters of prepareMOSI2CSP.py Script

Note that the `--mosi-app-id` and `--mosi-gateway-url` parameters and the `--mosi-app-instance-id` parameter are mutually exclusive.

- `--conversion-mode` – It defines the mode of conversion. You must specify the **resplitAdvanced** value.
- `--on-hold` – It defines if Office 365 subscriptions are put on administrative hold. You must specify either **yes** or **no**.

Note the following:

- The same value must be used in further steps of the conversion procedure.
- We recommend using **yes** to prevent customers from placing upgrade and downgrade orders during conversion.
- There is a drawback related to setting `--on-hold` to **yes**. This issue and its workaround are described at <https://kb.odin.com/en/131023>.
- `--mosi-app-id` – It defines the identifier of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--mosi-gateway-url` – It defines the URL of the gateway application of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.

- `--mosi-app-instance-id` – It defines the identifier of a source Office 365 APS 2 application instance (Syndication Partner Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the previous step of the conversion procedure.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer accounts which must be prepared for conversion. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.
- `--out-file` – It defines the name of a CSV file containing a list of customer tenants which are ready for conversion. The CSV file is created by the `prepareMOSI2CSP.py` script. The file is sent to Microsoft to convert the customer tenants on the Microsoft side.
- `--error-file` – It defines the name of a CSV file containing a list of customer tenants which are not ready for conversion. The CSV file is created by the `prepareMOSI2CSP.py` script. If the parameter is not specified, the default name `error_YYYYMMDD-HHMMSS.csv` is used.

### Input Files of `prepareMOSI2CSP.py` Script

The `prepareMOSI2CSP.py` script uses an input CSV file containing a list of customer accounts which must be prepared for conversion. Each row of the CSV file contains the following data of a customer account:

- **CustomerId** – The identifier of the customer account.
- other auxiliary columns (optional) – These columns are ignored by the script.

### Output Files of `prepareMOSI2CSP.py` Script

The `prepareMOSI2CSP.py` script creates an output CSV file containing a list of customer tenants which are ready for conversion. The name of the CSV file is defined by the `--out-file` parameter of the script. Each row of the CSV file contains the following data of a customer tenant:

- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.

The script creates an output CSV file containing a list of customer tenants which are not ready for conversion. The name of the CSV file is defined by the `--error-file` parameter of the script; if the parameter is not specified, the default name `error_YYYYMMDD-HHMMSS.csv` is used. Each row of the CSV file contains the following data of a customer tenant:

- **Aps** – The APS version of the Office 365 application to which the customer tenant belongs.

- **mosiAppIdOrMosiAppInstanceid** – It contains one of the following: the identifier of the Office 365 APS 1.2 application to which the customer tenant belongs; the identifier of the Office 365 APS 2 application instance to which the customer tenant belongs.
- **CustomerId** – The identifier of the customer account to which the customer tenant belongs.
- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.
- **MSTenantId** – The Microsoft Tenant ID of the customer tenant.
- **SubscriptionId** – The identifier of an Office 365 subscription which blocks the conversion of the customer tenant.
- **Error** – An error message.

### Log and Auxiliary Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` script creates the following log and auxiliary files:

- `prepareMOSI2CSP_YYYYMMDD-HHMMSS.log` – A log file containing general messages, such as information messages and error messages.
- `mosi2csp_mapping.csv` – A CSV file containing mappings between customer accounts and their customer tenants. The file is auxiliary and must not be modified or renamed. The file will be used by the `completeMOSI2CSP.py` script in further steps of the conversion procedure.

### Running prepareMOSI2CSP.py Script

Log in to the host where the conversion scripts are placed. Run the `prepareMOSI2CSP.py` script.

Example (Office 365 APS 1.2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitAdvanced --on-hold yes --mosi-app-id 100 --mosi-gateway-url https://office365.provider.com/Office365GatewayApplication/HTTPEndpoint.aspx --csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --error-file errors.csv
```

Example (Office 365 APS 2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitAdvanced --on-hold yes --mosi-app-instance-id 999 --csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --error-file errors.csv
```

After running the script, check the output CSV file containing a list of customer tenants which are not ready for conversion. If the file is not empty, you have the following options:

- Investigate and fix the errors. Run the script again using the same values of parameters. Repeat this until all customer tenants are ready for conversion.
- Postpone processing the customer tenants which are not ready for conversion and process them later. Note that you will need to start the conversion procedure from the beginning for the postponed customer tenants.

## Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

## Performing Conversion (Advanced Subscription Split Mode)

The customer tenants you prepared in the previous step of the conversion procedure must be converted on the Microsoft side:

- 1 Fill out a conversion request form:
  - a Download the conversion request form template provided at <https://kb.odin.com/en/130124>.
  - b Fill out the form using the information you prepared earlier:
    - The email address of the administrative user of the required syndication partner account.
    - The Microsoft ID of the required CSP partner account.
    - The CSV file containing the list of the customer tenants.
- 2 Send the file of the form to [csprans@microsoft.com](mailto:csprans@microsoft.com).
- 3 Microsoft will perform conversion on its side and provide you with two CSV files:
  - A CSV file containing a list of customer tenants which were successfully converted on the Microsoft side.
  - A CSV file containing a list of customer tenants which cannot be converted on the Microsoft side.
- 4 Check that the CSV files received from Microsoft have the **MosiCustomerId** column.
- 5 Upload both the CSV files to the host for running the conversion scripts.

The CSV files received from Microsoft will be used in further steps of the conversion procedure.

## Completing Conversion (Advanced Subscription Split Mode)

Complete the conversion of the customer tenants processed by Microsoft in the previous step of the conversion procedure:

- Use the instruction of the [Processing Customer Tenants Converted by Microsoft \(Advanced Subscription Split Mode\) section](#) (page 184) to process the customer tenants converted by Microsoft.
- Use the instruction of the [Processing Customer Tenants Rejected by Microsoft \(Advanced Subscription Split Mode\) section](#) (page 189) to process the customer tenants rejected by Microsoft.

## Processing Customer Tenants Converted by Microsoft (Advanced Subscription Split Mode)

The customer tenants converted on the Microsoft side in the previous step of the conversion procedure must be processed on the Odin Automation side.

The following activities are performed for every customer tenant:

- 1 Suitable Office 365 APS 2 service plans (CSP Program) are found for the customer account of the customer tenant. This means that all cloud subscriptions of the customer tenant will be represented by the corresponding Office 365 APS 2 subscriptions (CSP Program) in Odin Automation. This activity is performed by using the `completeMOSI2CSP.py` script.

Note the following:

- There might be several suitable Office 365 APS 2 service plans (CSP Program) for all cloud subscriptions (CSP Program) of a customer tenant. An Office 365 APS 2 service plan (CSP Program) containing suitable offer resources for all the cloud subscriptions (CSP Program) takes precedence over a set of Office 365 APS 2 service plans (CSP Program) containing suitable offer resources for all the cloud subscriptions (CSP Program).
  - If suitable Office 365 APS 2 service plans (CSP Program) are not found for the customer account of a customer tenant, further activities are not performed for the customer tenant.
  - Office 365 APS 2 service plans (CSP Program) containing composite resources are supported by the script. If each composite resource of an Office 365 APS 2 service plan (CSP Program) contains one included resource and the amount of the included resource is equal to **1**, the Office 365 APS 2 service plan (CSP Program) is taken into account by the script to find suitable Office 365 APS 2 service plans (CSP Program) for cloud subscriptions.
  - Both active cloud subscriptions and suspended cloud subscriptions are taken into account by the script.
- 2 All Office 365 subscriptions of the customer tenant are processed in the following way:
    - If a subscription has non-Office 365 resources with non-zero limits, the subscription is switched to a service plan including only non-Office 365 resources.
    - If the subscription has only Office 365 resources or all non-Office 365 resources of the subscription have zero limits, the subscription is cancelled.

This activity is performed by using the `completeMOSI2CSP.py` script.

- 3 The customer account of the customer tenant is subscribed to the Office 365 APS 2 service plans (CSP Program) found earlier. This activity is performed by using the `importSubsCSP.py` script.

### Parameters of `completeMOSI2CSP.py` Script



- `--conversion-mode` – It defines the mode of conversion. You must specify the **resplitAdvanced** value.
- `--on-hold` – Specify **yes** if Office 365 subscriptions are on administrative hold. Otherwise, specify **no**.
- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the first step of the conversion procedure.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer tenants converted by Microsoft. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.
- `--out-file` – It defines the name of a CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The CSV file is created by the `completeMOSI2CSP.py` script. The CSV file will be used by the `importSubsCSP.py` script.
- `--error-file` – It defines the name of a CSV file containing a list of customer tenants which cannot be processed. The CSV file is created by the `completeMOSI2CSP.py` script. If the parameter is not specified, the default name `errors_completion_YYYYMMDD-HHMMSS.csv` is used.

### Input Files of completeMOSI2CSP.py Script

The `completeMOSI2CSP.py` script uses an input CSV file containing a list of customer tenants converted by Microsoft. Each row of the CSV file contains the following data of a cloud subscription of a customer tenant:

- **CustomerName** – The Business Name of the customer tenant.
- **CustomerTenantId** – The Microsoft Tenant ID of the customer tenant.
- **FpoCaid** – The Microsoft Internal Commerce ID of the customer tenant.
- **MosiCustomerId** – The MOSI Customer ID of the customer tenant.
- **MosiSubscriptionId** – The MOSI subscription identifier of the cloud subscription.
- **CspSubscriptionId** – The CSP subscription identifier of the cloud subscription.
- **CspSubscriptionOfferId** – The CSP Offer ID of the cloud subscription.
- **CspSubscriptionStartDate** – The start date of the cloud subscription.
- **CspSubscriptionEndDate** – The end date of the cloud subscription.

Note that the `completeMOSI2CSP.py` script uses the **MosiCustomerId** column and ignores the other columns.

### Output Files of completeMOSI2CSP.py Script

The `completeMOSI2CSP.py` script creates an output CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The name of the CSV file is defined by the `--out-file` parameter of the script. Each row of the CSV file contains the following data of a customer tenant:

- **OSACustomerID** – The identifier of the customer account of the customer tenant.
- **OSAServicePlanId** – The identifier of an Office 365 APS 2 service plan (CSP Program) to which the cloud subscriptions specified in the **MSSubscriptionIDs** column are mapped by the script.
- **OSAPlanPeriodID** – The identifier a subscription period of the Office 365 APS 2 service plan (CSP Program). The subscription period is chosen by the script.
- **SubscriptionStartDate** – The start date of an Office 365 APS 2 subscription (CSP Program) which is created for the cloud subscriptions specified in the **MSSubscriptionIDs** column.
- **ResourceRateLimits** – A list of resource rates and their limits. The resource rates are chosen by the script for the cloud subscriptions specified in the **MSSubscriptionIDs** column. The format "**RR\_ID1-LIMIT,RR\_ID2-LIMIT**" is used for the list. For example: "**100-10,200-20**".
- **CSPAccountid** – The Microsoft Tenant ID of the customer tenant.
- **MSSubscriptionIDs** – A list of cloud subscriptions of the customer tenant. The format "**MS\_SUB\_ID1,MS\_SUB\_ID2,...**" is used for the list. For example: "**A49C6781-1557-42CA-A074-52D0553397EA,EBE5106C-B1ED-44FF-8CE7-A2746AF8BC42**".
- **DefaultAdminLogin** – The login name of the default administrative user of the customer tenant.
- **DefaultDomainName** – The default domain name of the customer tenant.
- **SourceSubscriptionId** – This column is not used in this mode of conversion. Do not change the default value of the column.

Note that the CSV file might contain several rows for one customer tenant if the cloud subscriptions of the customer tenant are mapped to several suitable Office 365 APS 2 service plans (CSP Program).

The `completeMOSI2CSP.py` script creates an output CSV file containing a list of customer tenants whose cloud subscriptions cannot be mapped to Office 365 APS 2 service plans (CSP Program). The name of the CSV file is defined by the `--error-file` parameter of the script; if the parameter is not specified, the default name `errors_completion_YYYYMMDD-HHMMSS.csv` is used. Each row of the CSV file contains the following data of a customer tenant:

- **OSACustomerID** – The identifier of the customer account of the customer tenant.
- **CSPAccountid** – The Microsoft Tenant ID of the customer tenant.
- **SubscriptionStartDate** – The start date of the cloud subscriptions specified in the **MSSubscriptionIDs** column.

- **MSSubscriptionIDs** – A list of cloud subscriptions of the customer tenant. The format "MS\_SUB\_ID1,MS\_SUB\_ID2,..." is used for the list. For example: "A49C6781-1557-42CA-A074-52D0553397EA,EBE5106C-B1ED-44FF-8CE7-A2746AF8BC42".
- **MSSubscriptionOfferIDs** – The identifiers of the offers of the cloud subscriptions specified in the **MSSubscriptionIDs** column.
- **Comments** – An error message.

### Log and Auxiliary Files of completeMOSI2CSP.py Script

The `completeMOSI2CSP.py` script creates a log file with the name `completeMOSI2CSP_YYYYMMDD-HHMMSS.log`. The file contains general messages, such as information messages and error messages.

The `completeMOSI2CSP.py` script implicitly uses the `mosi2csp_mapping.csv` file created by the `prepareMOSI2CSP.py` script. The CSV file contains mappings between customer accounts and their customer tenants. The file is auxiliary and must not be modified or renamed.

### Running completeMOSI2CSP.py Script

- 1 Log in to the host where the conversion scripts are placed.
- 2 Run the `completeMOSI2CSP.py` script.
- 3 Check the output CSV file containing a list of customer tenants whose cloud subscriptions cannot be mapped to Office 365 APS 2 service plans (CSP Program). If the CSV file is not empty, investigate and fix the errors. Run the script again using the same values of parameters. Repeat this until all customer tenants are processed.
- 4 Check the output CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program):
  - a Make sure that mappings are correct.
  - b Make sure that resource rates and their limits are correct.

Modify the CSV file if necessary. For instance, you can specify your own mappings between Office 365 APS 2 service plans and cloud subscriptions.

Example:

```
python completeMOSI2CSP.py --conversion-mode resplitAdvanced --on-hold yes --csp-app-instance-id 1000 --pba --input-file input.csv --out-file out.csv --error-file error.csv
```

### Troubleshooting

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

### Parameters of importSubsCSP.py Script

- `--csp-app-instance-id` – It defines the identifier of a target Office 365 APS 2 application instance (CSP Program). You must use the value you prepared in the first step of the conversion procedure.
- The name of a CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). You must use the CSV file created by the `completeMOSI2CSP.py` script.

### Input Files of `importSubsCSP.py` Script

The `importSubsCSP.py` script uses an input CSV file containing mappings between Office 365 APS 2 service plans (CSP Program) and cloud subscriptions (CSP Program). The CSV file is created by the `completeMOSI2CSP.py` script. The format of the CSV file is described above.

### Output Files of `importSubsCSP.py` Script

The `importSubsCSP.py` script creates a CSV file with the name `result_<input file name>`. Each row of the CSV file contains the following data of an Office 365 CSP subscription created by the `importSubsCSP.py` script:

- **CustomerId** – The identifier of the customer account to which the subscription belongs.
- **SubscriptionId** – The Odin Automation identifier of the newly created Office 365 CSP subscription.

### Log and Auxiliary Files of `importSubsCSP.py` Script

The `importSubsCSP.py` script creates a CSV file with the name `errors_import_<input file name>`. This file contains a list of cloud subscriptions which cannot be processed by the `importSubsCSP.py` script as well as reasons why the cloud subscriptions cannot be processed. The CSV file has the same format as the input file of the `importSubsCSP.py` script (except for the additional **Error** column, which contains error messages).

The `importSubsCSP.py` script creates a CSV file with the name `reimport_<input file name>`. This file contains a list of cloud subscriptions which cannot be processed by the `importSubsCSP.py` script. The CSV file has the same format as the input file of the `importSubsCSP.py` script. You can use the file `reimport_<input file name>` to run the `importSubsCSP.py` script again only for affected cloud subscriptions.

The `importSubsCSP.py` script creates a log file with the name `importCSPYYYY-MM-DD.log`. The file contains general messages, such as information messages and error messages.

### Running `importSubsCSP.py` Script

- 1 Stop Office 365 synchronization:
  - a In Provider Control Panel, go to **Operations > Tasks**.



Note that the `--mosi-app-id` and `--mosi-gateway-url` parameters and the `--mosi-app-instance-id` parameter are mutually exclusive.

- `--conversion-mode` – It defines the mode of conversion. You must specify the **resplitAdvanced** value.
- `--rollback` – It defines if the script must roll back changes made for customer tenants. You must specify the parameter to process customer tenants rejected by Microsoft.
- `--mosi-app-id` – It defines the identifier of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--mosi-gateway-url` – It defines the URL of the gateway application of a source Office 365 APS 1.2 application (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--mosi-app-instance-id` – It defines the identifier of a source Office 365 APS 2 application instance (Syndication Partner Program). You must use the value you prepared in the first step of the conversion procedure.
- `--pba` – It defines if your Odin Automation installation includes OA Billing. You must specify the parameter if your Odin Automation installation includes OA Billing.
- `--input-file` – It defines the name of a CSV file containing a list of customer tenants rejected by Microsoft. The CSV file is received from Microsoft. You must specify the name of the CSV file you prepared in the previous step of the conversion procedure.

### Input Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` uses an input CSV file containing a list of customer tenants rejected by Microsoft. The `prepareMOSI2CSP.py` script uses the **MosiCustomerId** column and ignores other columns.

### Log and Auxiliary Files of prepareMOSI2CSP.py Script

The `prepareMOSI2CSP.py` script creates a log file with the name `prepareMOSI2CSP_YYYYMMDD-HHMMSS.log`. The log file contains general messages, such as information messages and error messages.

### Running prepareMOSI2CSP.py Script

Log in to the host where the conversion scripts are placed. Run the `prepareMOSI2CSP.py` script.

Example (Office 365 APS 1.2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitAdvanced --rollback --mosi-app-id 100 --mosi-gateway-url https://office365.provider.com/Office365GatewayApplication/HTTPEndpoint.aspx --pba --input-file input.csv
```

Example (Office 365 APS 2 / Syndication Partner Program):

```
python prepareMOSI2CSP.py --conversion-mode resplitAdvanced --rollback --mosi-app-instance-id 999 --pba --input-file input.csv
```

After running the script, you can analyze the customer tenants rejected by Microsoft to find out the reason for rejection. Note that you will need to start the conversion procedure from the beginning for rejected customer tenants after fixing them.

### **Troubleshooting**

Use the KB article <https://kb.odin.com/en/129840> for troubleshooting.

# Index

## A

Abbreviations - 9  
Adding DNS Records - 21  
Advanced Subscription Split Mode - 169  
Allocating IP Addresses - 21  
Audience - 8

## B

Basic Object Model - 14  
Basic Partnership Scenario - 13

## C

Cloud Solution Provider Scenario - 11  
Collecting Tenant Data - 103  
Completing Conversion (Advanced Subscription Split Mode) - 183  
Completing Conversion (Subscription Preservation Mode) - 162  
Completing Conversion (Subscription Split Mode) - 140  
Configuring API Access Permissions (Both Native Apps) - 29  
Configuring API Access Permissions (Native App for Graph API) - 23, 26  
Configuring Billing - 45  
Configuring 'Checkout' Screen - 82  
Configuring Firewall - 21  
Configuring Internationalization Settings - 47  
Configuring OA Billing Attributes - 46  
Configuring Offer Compatibility Checker - 83  
Configuring Offers - 62  
Configuring Offers for Resellers - 92  
Configuring Office 365 Subdomain Form for OA Billing - 83  
Configuring Online Store - 81  
Configuring Pre-consent (Native App for Graph API) - 24, 27, 30  
Conversion from Syndication Partner Program to CSP Program - 126

Conversion Procedure (Advanced Subscription Split Mode) - 176  
Conversion Procedure (Subscription Preservation Mode) - 154  
Conversion Procedure (Subscription Split Mode) - 133  
Creating Add-on Types - 67  
Creating Administrative User - 22  
Creating Auxiliary Resource Types - 73  
Creating Domain for Office 365 Subdomains - 45  
Creating License Types - 66  
Creating 'Office 365 Add-on' Resource Types - 72  
Creating 'Office 365' Application Instance - 41  
Creating 'Office 365 Domain' Resource Type - 71  
Creating 'Office 365 Global Settings' Resource Type - 69  
Creating 'Office 365 License' Resource Types - 71  
Creating 'Office 365' Purchase Scenario - 82  
Creating 'Office 365' Resource and Sales Categories - 80  
Creating Office 365 Resource Types - 69  
Creating 'Office 365' Screen - 81  
Creating 'Office 365' Service Plans - 77  
Creating 'Office 365' Service Template - 74  
Creating 'Office 365' Service Terms - 61  
Creating Office 365 Subscription On Behalf Of Customer - 116  
Creating 'Office 365 Tenant' Resource Type - 69  
Creating 'Office 365 User' Resource Type - 70  
Creating 'PBA' Resource Type - 45

## D

Deploying 'Office 365' Application - 32



Deploying Office 365 Application Endpoint - 33

Deployment Architecture - 15

Deployment Scheme - 16

Documentation Conventions - 6

## E

Example - 108

Examples of Office 365 Service Plans - 118

## F

Feedback - 7

Firewall Requirements - 97

## G

General Architecture - 12

General Conventions - 7

Getting Account ID of CSP Partner Account - 25, 28, 30

Granting Admin Consent (Both Native Apps) - 29

Granting Admin Consent (Native App for Graph API) - 24, 27

## H

Hardware Requirements for Office 365 Application Endpoint Host - 33

Helpful Links - 9

How to Update Expired SSL Certificates on Office 365 Application Endpoint Host - 115

How to Upgrade Office 365 Subscriptions through Provider/Reseller Control Panel - 110

## I

Importing 'Office 365' Application - 32

Importing Tenant Subscriptions - 106

Importing Tenant Subscriptions from Microsoft Cloud into Odin Automation - 101

Installing Office 365 Application Endpoint - 34

Installing Office 365 Application Endpoint on New Host - 34

Integration Overview - 11

Introduction - 8

## L

Limitations - 102

Limitations (Advanced Subscription Split Mode) - 172

Limitations (Subscription Preservation Mode) - 150

Limitations (Subscription Split Mode) - 129

## M

Microsoft Cloud - 23

Microsoft Cloud Germany - 28

## N

Notifying Customers (Advanced Subscription Split Mode) - 174

Notifying Customers (Subscription Preservation Mode) - 152

Notifying Customers (Subscription Split Mode) - 131

## O

OA Billing Application Server - 100

OA Operations Branding Server - 100

OA Operations Management Node - 99

Obtaining Application Package - 31

Office 365 Application Endpoint Host - 97

Office 365 Operated by 21Vianet in China - 25

Operations Available to Office 365 Subscribers - 8

Overview (Advanced Subscription Split Mode) - 169

Overview (Subscription Preservation Mode) - 148

Overview (Subscription Split Mode) - 126

## P

Performing Conversion (Advanced Subscription Split Mode) - 183

Performing Conversion (Subscription Preservation Mode) - 161

Performing Conversion (Subscription Split Mode) - 140

Preconditions - 101

Preface - 6

Preparing Conversion Scripts (Advanced Subscription Split Mode) - 175

Preparing Conversion Scripts (Subscription Preservation Mode) - 153

- Preparing Conversion Scripts (Subscription Split Mode) - 132
- Preparing CSP Account - 21
- Preparing Customer Accounts (Advanced Subscription Split Mode) - 179
- Preparing Customer Accounts (Subscription Preservation Mode) - 158
- Preparing Customer Accounts (Subscription Split Mode) - 136
- Preparing for Conversion (Advanced Subscription Split Mode) - 173
- Preparing for Conversion (Subscription Preservation Mode) - 151
- Preparing for Conversion (Subscription Split Mode) - 130
- Preparing Odin Automation Public API User - 31
- Preparing Office 365 Application Endpoint Host - 33
- Preparing Office 365 Welcome Message Templates - 31
- Preparing Server - 17
- Preparing SQL Server Databases - 17
- Preparing Your Installation (Advanced Subscription Split Mode) - 173
- Preparing Your Installation (Subscription Preservation Mode) - 151
- Preparing Your Installation (Subscription Split Mode) - 130
- Prerequisites - 15, 23, 25, 28
- Prerequisites (Advanced Subscription Split Mode) - 172
- Prerequisites (Subscription Preservation Mode) - 150
- Prerequisites (Subscription Split Mode) - 129
- Processing Customer Tenants Converted by Microsoft (Advanced Subscription Split Mode) - 184
- Processing Customer Tenants Converted by Microsoft (Subscription Preservation Mode) - 162
- Processing Customer Tenants Converted by Microsoft (Subscription Split Mode) - 140
- Processing Customer Tenants Rejected by Microsoft (Advanced Subscription Split Mode) - 189
- Processing Customer Tenants Rejected by Microsoft (Subscription Preservation Mode) - 168
- Processing Customer Tenants Rejected by Microsoft (Subscription Split Mode) - 146
- ## R
- Registering Native Apps - 23, 26, 28
- Registering Native Apps for Partner Center and Graph APIs - 23
- Registering New Customers with Existing Microsoft Accounts - 116
- Restrictions and Limitations - 66
- ## S
- Securing Office 365 Application Endpoint Host - 40
- Selecting Customer Accounts (Advanced Subscription Split Mode) - 177
- Selecting Customer Accounts (Subscription Preservation Mode) - 155
- Selecting Customer Accounts (Subscription Split Mode) - 133
- Selecting Online Store - 81
- Setting CSP Admin User's Password to Never Expire - 44
- Setting Up Units of Measure for 'Office 365 License' and 'Office 365 Add-on' Resources - 80
- SQL Server 2012/2014 Express/Web/Standard/Enterprise (x64) - 18
- SQL Server 2014 Express LocalDB (x64) - 18
- Subscription Preservation Mode - 148
- Subscription Split Mode - 126
- Support for Account-Wide Services - 14
- Supported Languages - 9
- Synchronizing Changes from Office 365 Portal - 114
- Synchronizing Online Store - 82
- ## T
- Technologies - 8
- Typographical Conventions - 6
- ## U
- Useful Information - 110

Using Office 365 APS 1.2 Gateway Host for  
Office 365 APS 2 Deployment - 37

**V**

Validation of Imported Tenant Subscriptions -  
108

Viewing Business Reports - 113

Viewing Reconciliation Report - 111