
# i-doit pro OCS Inventory AddOn

**Version 1.6.0**

# Table of Contents

# 1. Introduction

OCS Inventory (Open Computer and Software Inventory Next Generation) is an open-source software for automatic inventory of hardware and software components as well as entire networks. The software is available on the official website at http://www.ocsinventory-ng.org/ for various operating systems.

The OCS Inventory add-on offers an import interface to import both hardware and software objects from the OCS Inventory System directly into the i-doit CMDB.

# 2. System requirements

The current version of the OCS add-on requires at least i-doit version 1.19.
The interface is tested and supported for OCSInventory Server 2.12.x.

> (**i**) Unless otherwise declared, this documentation refers to the current version (1.5.0) of the OCS Inventory add-on. The paths for opening the masks are based on min. i-doit 25 and may differ in older versions.

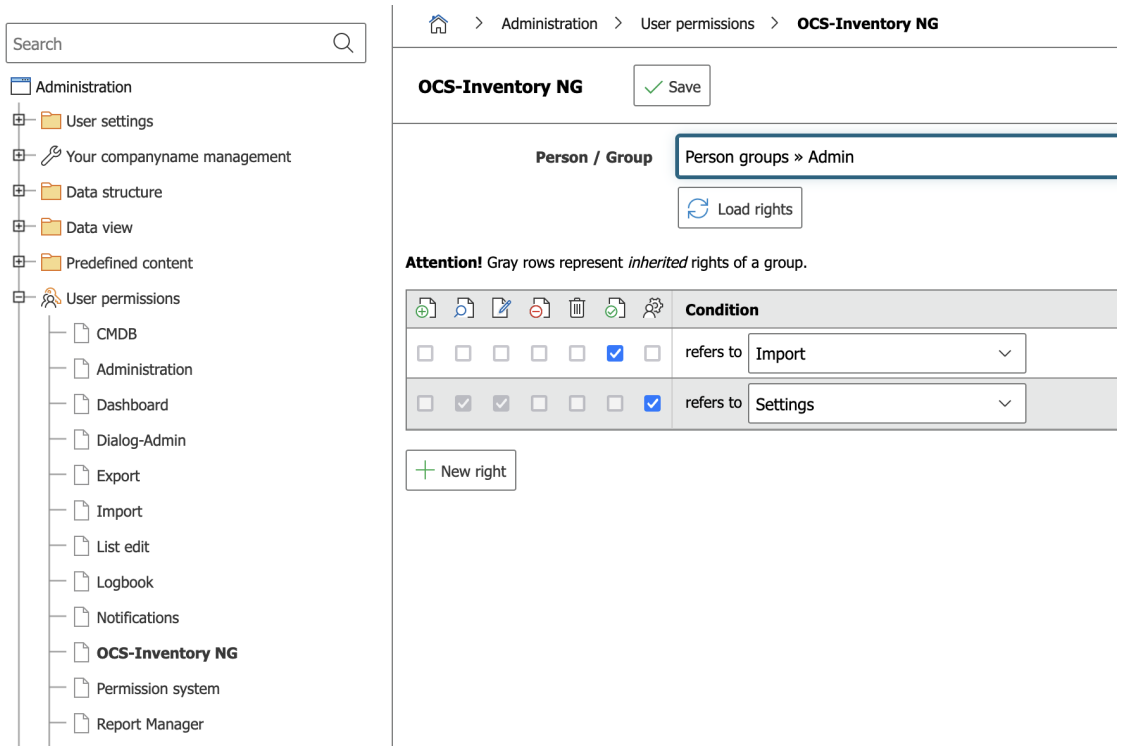
# 3. Installation

The installation of the OCS Inventory add-on follows the standard procedure for the installation of i-doit add-ons:

- Log in to the i-doit Admin Center.
- Go to the “Add-ons” tab
- Click on the “Install/update Add-on” button
- Select the ZIP package of the add-on
- Click the “Upload and install” button
- Ready

# 4. Rights

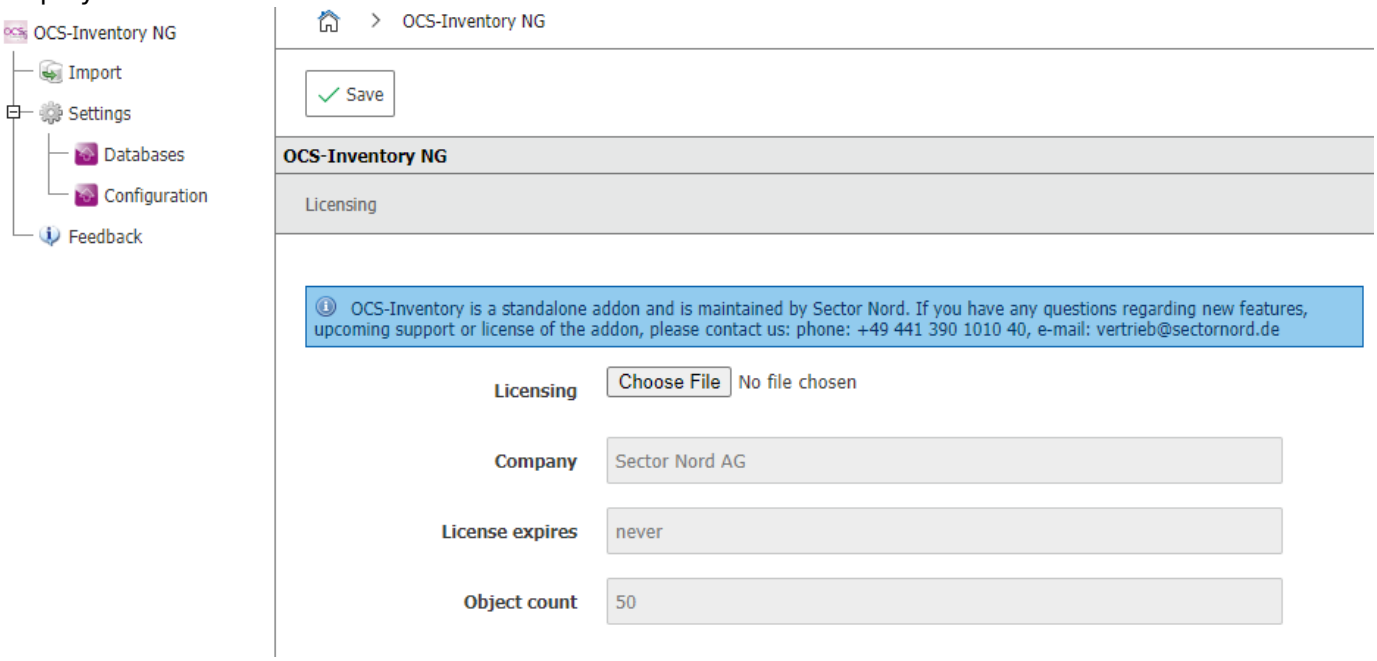
To be able to use the OCS import, the permissions for the OCS Add-on must be set under **Administration → User permissions → OCS-Inventory**




# 5. Setup

## 5.1. Licensing

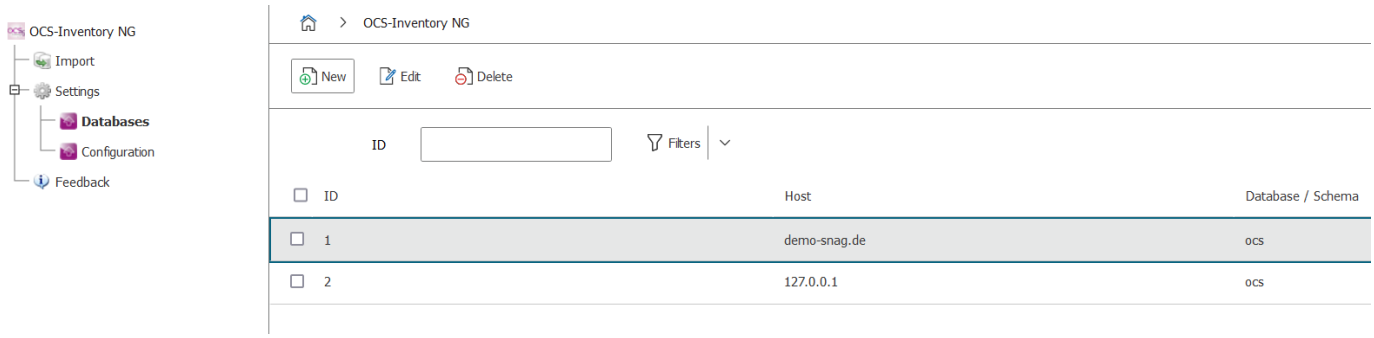
The licensing of the module can be done under **Add-ons → OCS-Inventory → Settings → Licensing**. After importing the license file, the expiration date and the number of licensed objects are displayed.




Note: As the license is provided by Sector Nord AG, it is not imported via the i-doit admin portal and is not displayed in the list there.

## 5.2. OCS databases

Access to the OCS databases can be configured under **Add-ons → OCS-Inventory → Settings → Databases**. Multiple OCS instances can be specified:




Create/edit a new database connection:

| Database setting | Description | Example |
|---|---|---|
| Host | IP/DNS address of the OCS server | 127.0.0.1 |
| URL | Specifying the URL | https://ocs.example.com/ocsreports/ |
| Port | Port to access the database | 3306 |
| Database / Schema | Name of the database | ocs |
| User | Database user with access rights to the OCS database | ocs |
| Password | User password | ****** |

The database configuration can be tested before saving using the “Test connection to OCS” button. The following error messages may occur:

| Error Message | Cause |
|---|---|
| Can access the OCS server but can not connect to the OCS database. | invalid database/username/password |
| Can not connect to the OCS server. Check host and port. | incorrect IP |

## 5.3. Configuration

Under **Add-ons → OCS-Inventory → Settings → Configuration** some parameters for the handling of objects during import can be set.

| Setting | Description | Example |
|---|---|---|
| Default Database | Sets the database that will be used as the default for import in the user interface and console. | 127.0.0.1 - ocs |
| Ignore object types for import (Blacklist) | Provides a dropdown where object types can be selected that should not be considered for import. The object type after the eventual redetermination is used for the calculation. | Printer, Client ... |
| Ignore inactive OCS objects (days) | Devices for which the last synchronization to OCS was longer ago than the specified time (days) are not imported. With no entry or a 0, no limit is set | 0 |

| Setting | Description | Example |
|---|---|---|
| Ignore objects with CMDB status | Provides a dropdown to select the CMDB states where a device should not be imported. Selection only accesses already existing objects | inoperative, defect |
| Update archived OCS objects | Provides a dropdown menu where the CMDB statuses and conditions can be selected, which an object receives when it is archived in OCS. | Archive, CMDB status: inoperative |
| Object type mapping | Allows defining a regex search pattern. The mapping checks both tags and the object title and assigns the matching object type during import. The rules are executed in order. A rule can contain several regex filters one below the other. | /srv.*/<br>/clt.*/<br>.... |
| Default import object type | Sets the default object type if no object type is selected during import and/or none of the mapping entries apply. | Client |
| Only import known applications | Specifies whether the imported objects should only be linked with applications that are already present in the i-doit environment.<br>If this option is set to No, all applications not yet in i-doit will be newly created and linked to the corresponding object. | Yes/No |
| Default object type for imported applications | Offers a selection of object types, with the Specific category “Applications”, that can be chosen as the default object type. | Application, Custom Application ... |
| Remove existing application ssignments | This option allows removing existing application assignments to already created software objects. The assignment is cleaned before import and filled with the new application links. The application objects are not deleted. | Yes/No |
| Ignore specific application names | Definition of regex strings to not import matched applications. Multiple regexes can be defined (one regex per line).<br>**Note:** see advice under section 5.3.1 | /lib\|kernel/ |
| Regex to edit application names | Definition of regex strings, for example, to remove the version number from the OCS application name. Multiple regexes can be defined (one regex per line).<br>**Note:** see advice under section 5.3.2 | /(\d+\.)+\d+/ |
| Ignore specific monitors | Allows the definition of regex strings to not import monitors with certain serial numbers. One regex per line. | /LED Monitor.*/ |
| Filter layer 3 networks | Only import monitors where the connected client is assigned to a specific Layer 3 network. Provides a selection list of the stored Layer 3 networks. Empty = No filter | 192.168.1.0/24 |
| Cable connections for monitors | Determines the mode for importing monitors and their cable connections. Three modes can be selected.<br>**Note:** see advice under section 5.5 | Import Monitors Only |
| Logbook active while importing | Activates entering all object changes in the logbook. | Yes/No |
| Object matching profile | The Object Matching Profile parameter specifies how already documented objects are to be identified. Identified objects are updated during data import. | Default |

| Setting | Description | Example |
|---|---|---|
| Do not overwrite Serial numbers with values | Definition of values that don't overwrite serial numbers . Multiple values can be stored. | unknown,- |
| Ignore host addresses from unknown layer 3 networks | Host addresses for which there are no existing Layer 3 networks are not imported | Yes/No |
| License expires | Shows the expiration date of the license. Provides a link to the licensing | 2022-12-31 |

### 5.3.1 Ignore specific application names

By using the filter, any application that matches will be ignored during import. To match an application, only substrings need to match, not the complete name.
Multiple filters are additive. Thus, all filters are applied one after the other to the software list. The remaining applications will be imported.

A whitelist can also be defined instead of a blacklist by using a negative filter. For example, a "negative lookahead" can be used:
`/^(?!MyApp1|MyApp2).*$/i`
During import, only applications whose name begins with *MyApp1* or *MyApp2* will be imported.
By specifying the 'i' option, the filter is performed case-insensitively.

### 5.3.2 RegEx to edit application name

The configuration of the regular expression always replaces the first found match. If there are multiple matches for the expression in the application name, only the first is considered. Multiple filters work additive. Thus, all filters are applied one after the other to the software list. The remaining strings will be used for the application name.

Additionally, the first found group is used to supplement possible missing version numbers. If the version number includes a string like 'unavailable', it will be replaced with the extracted version.

Here are some examples of possible expressions for cleaning up application names:

| RegEx | Application Name | Result |
|---|---|---|
| `/((\d+\.)+\d+/)` | MyApp 4.0.1 | MyApp |
| `/V((\d+\.)+\d+)/` | MyApp V4.0.1 | MyApp |
| `/(\d+/)` | MyApp 2019 | MyApp |

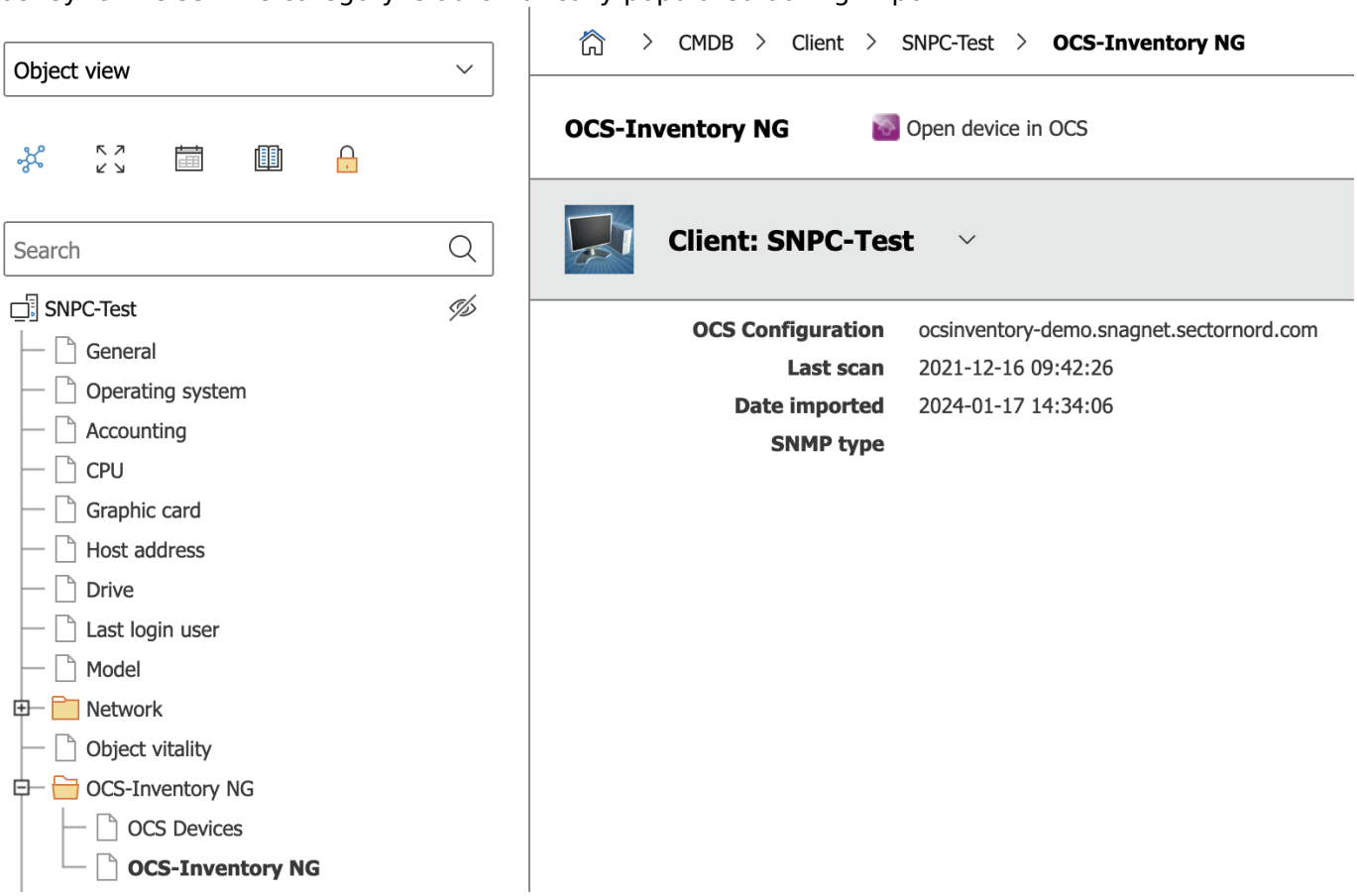
With the last example, it should be considered that this removes the first sequence of numbers from every application name (this may not necessarily be the version number, e.g., in "Microsoft Office 365 Apps…").
These filters can also be refined with OR conditions `|` or by using case-insensitive matching. The filter `/(?:v|)((\d+\.)+\d+)/i` would transform the string `MyApp 4.0.1`, as well as `MyApp V4.0.1` or `MyApp v4.0.1` to `MyApp`, and create or assign this application. By using a non-capturing group `(?:)`, it can be ensured that the first group contains the correct version number for replacement.
For creating advanced regular expressions, tools like Regex101 should be used before importing to ensure data quality.

## 5.4 OCS Categories

For an overview of relevant data from the OCS import, the category folder “OCS-Inventory” can be activated for desired object types. Activation is done by default in the object type configuration or via “Edit data structure”.
Two categories are provided in the folder.

### 5.4.1 Category - OCS-Inventory

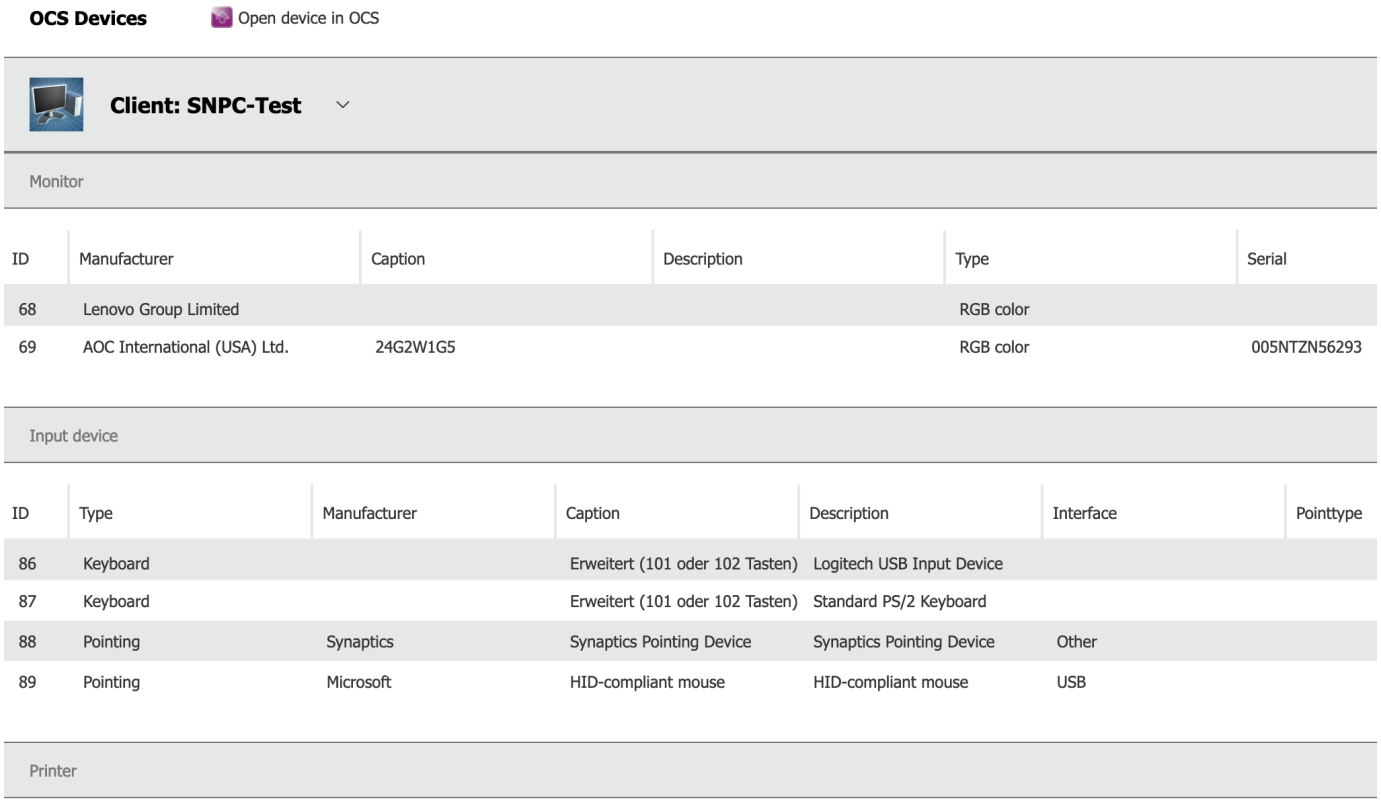
The folder’s own category ‘OCS-Inventory’ provides information about the import into i-doit and the last sync in OCS. The category is automatically populated during import.




By clicking the “Open Device in OCS” button, the device can be directly accessed in OCS. For this, the URL must be correctly entered in the Database Configuration.

### 5.4.2 Category - OCS Devices

The ‘OCS Devices’ category provides an overview of connected input devices, monitors, and printers. This category is only populated if the ‘OCS Devices’ category is selected during Import. The data is analogous to the overview in OCS.



**OCS Devices** Open device in OCS

**Client: SNPC-Test**

Monitor

| ID | Manufacturer | Caption | Description | Type | Serial |
|---|---|---|---|---|---|
| 68 | Lenovo Group Limited | | | RGB color | |
| 69 | AOC International (USA) Ltd. | 24G2W1G5 | | RGB color | 005NTZN56293 |

Input device

| ID | Type | Manufacturer | Caption | Description | Interface | Pointtype |
|---|---|---|---|---|---|---|
| 86 | Keyboard | | Erweitert (101 oder 102 Tasten) | Logitech USB Input Device | | |
| 87 | Keyboard | | Erweitert (101 oder 102 Tasten) | Standard PS/2 Keyboard | | |
| 88 | Pointing | Synaptics | Synaptics Pointing Device | Synaptics Pointing Device | Other | |
| 89 | Pointing | Microsoft | HID-compliant mouse | HID-compliant mouse | USB | |

Printer

In addition to the overall view in the ‘OCS Devices’ category, the individual categories ‘OCS Monitor’, ‘OCS Input Device’, and ‘OCS Printer’ can be activated separately. These three categories can also be activated and used independently of the folder. The categories can also be evaluated in the Report Manager.

## 5.5 Monitors

There are three different options for importing monitors. For a monitor to be imported as a separate object, the serial number is mandatory. If this field is not provided by OCS, no monitor object will be created or linked.
If monitors are not to be imported or linked, the ***Monitor*** category must be deselected in the Category Selection.

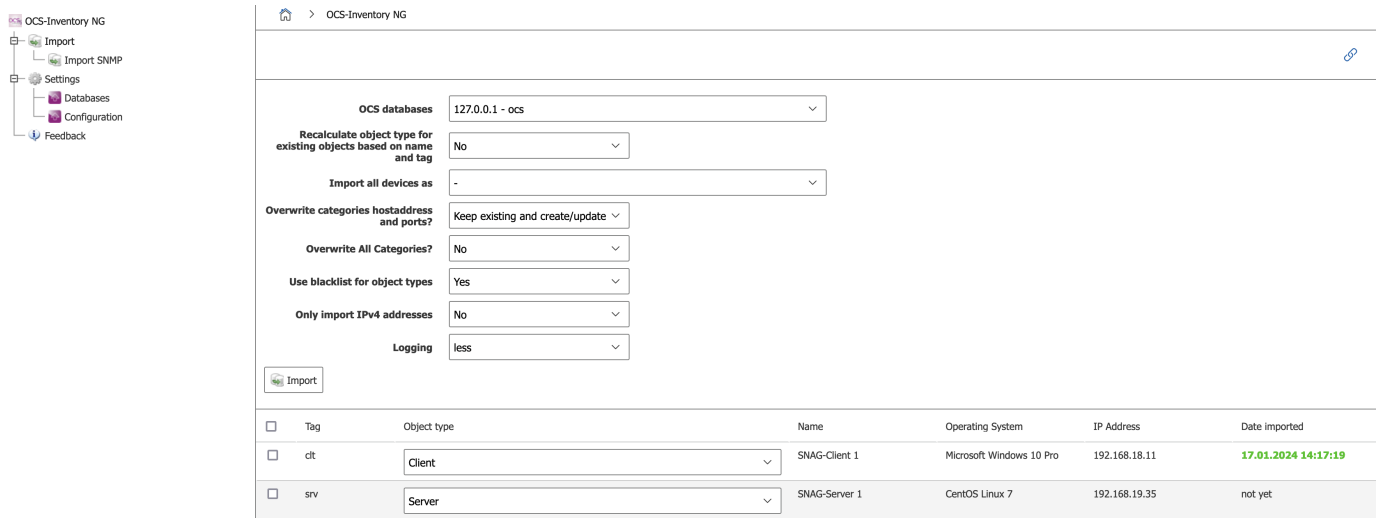
| Mode | Explanation |
|---|---|
| Import monitors and create cable connections | All valid monitors will be imported as monitor objects and linked to the client via the cabling category. Existing monitors and links will be matched and updated. |
| Import monitors only | All valid monitors will be imported as monitor objects. No connection to the client will be made. |
| Create and update cable connections only | Only existing monitor objects will be matched and cabling updated. |

Matching of existing monitors is done exclusively via the serial number field from the model category. The name of the monitor may differ from the name in OCS and can also be freely adjusted afterward. For the cable connections, a new interface “Monitor connector”/ “Client connector” is created on the client/monitor with connection type “Monitor” and port type “Video Port”.

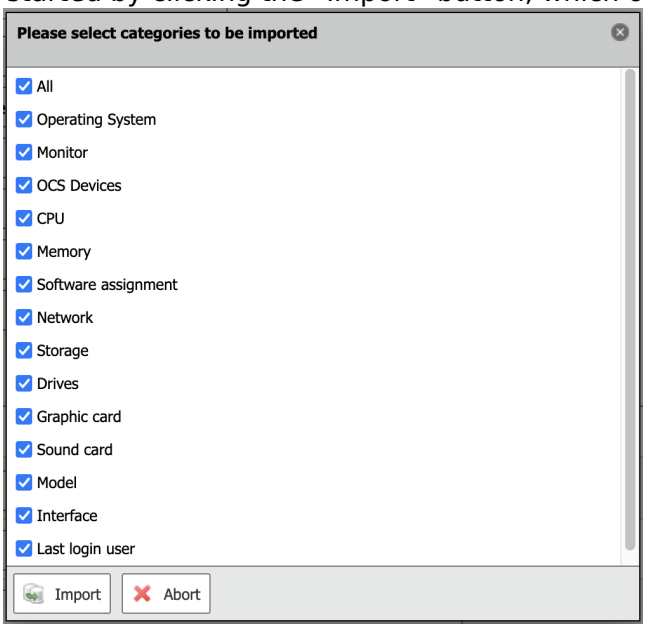
# 6. Import

Under **Add-ons → OCS-Inventory → Import** one or more objects can be imported from the OCS database into the i-doit CMDB. The list of displayed objects can be sorted alphabetically by clicking on

the individual column headers.




The import of the selected objects is started by clicking the “Import” button, which opens another window for selecting the categories:




> When the import is finished, you will receive a status message with information about the individual steps of the import.

# 6.1 Import settings

| Import setting | Description | Example |
|---|---|---|
| OCS databases | A list selection with the configured databases is provided here | 127.0.0.1 - ocs |
| Recalculate object type for existing objects based on tags | Determines whether the object type for existing objects should be redefined based on the tag and tag-prefix matching. The determination can be manually adjusted afterwards. | Yes/No |
| Import all devices as | Determines the object type with which all objects are to be imported. The determination can be adjusted manually afterwards. | Client |

| Import setting | Description | Example |
|---|---|---|
| Overwrite categories hostaddress and ports? | Handling for the multivalue categories. Similar to the CSV import, there are three options to choose from: Add only and do not delete / Delete and refill / Fill empty categories only. | Keep existing and create/update any from OCS |
| Overwrite All Categories? | Purge all categories included by the import. A list of all categories can be viewed when running the import. Other categories, such as Accounting, are not deleted. | Yes/No |
| Use blacklist for object types | Decides whether the defined blacklist should be used for the import | Yes/No |
| Only import IPv4 adresses | Defines if Ipv6 addresses should be ignored during import. | Yes/No |
| Logging | Sets the log level for the created log files.<br>Less: Does not create a log file.<br>Detailed: Writes a log file with import information to /i-doit-PATH/log/ .<br>Detailed+Debug: Adds debug info to logging | less<br>detailed (slower)<br>detailed+debug (very slow & memory intensive) |

# 7. Console import

Using the i-doit console, the import can be executed automatically, for example via a cronjob. The responsible command is called **import-ocs**. The –help parameter can be used to display the following options:

| Parameter (short version) | Parameter (long version) | Example |
|---|---|---|
| | --ipPortOverwrite= IPPORTOVERWRITE | Determines if hostaddresses and ports should be deleted first for each imported device: 0 = Keep existing and create/update any from OCS (Default); 1 = Delete existing and create from OCS; 2 = Keep existing and don’t update any from OCS |
| | --allCatsOverwrite= ALLCATSOVERWRITE | Determines if all cats selected for import should be purged before new import. 1 = Active; 0 = Inactive (Default) |
| | --databaseSchema= DATABASESCHEMA | Selection of the database schema used for the import |
| | --onlyIPv4=ONLYIPV4 | Imports IPv4 addresses only. 1 = Active; 0 Inactive (default) |
| | --useBlacklist= USEBLACKLIST | Determines if the configured blacklist of object types should be used (1) or ignored (0) |
| | --objectType=OBJECTTYPE | Default objecttype constant from the object type configuration. This objecttype will be used if no objecttype can be determined. |
| | --objectTypeWhitelist=OBJECTTYPE | Only import defined object types. E.g. C__OBJTYPE__SERVER. Use –usage to see all possible object types. Multiple values can be separated by comma. |
| | --objectTypeBlacklist=OBJECTTYPE | Skip defined object types during import. E.g. C__OBJTYPE__CLIENT. Use –usage to see all possible object types. Multiple values can be separated by comma. |
| | --recalculateObjectType= RECALCULATEOBJECTTYPE | Recalculation of object types for already existing objects based on tags. [default: false] |
| | --file=FILE | Option for a source file which contains hostnames which will be imported/updated. |
| | --hosts=HOSTS | Comma separated list of Hostnames which will be imported/updated. |
| | --snmpDevices= SNMPDEVICES | Switch if snmp device should be imported. |
| | --categories=CATEGORIES | Comma separated list of categories to import. Possible Values: drive, ui, sound, application, memory, model, graphic, net, stor, operating_system, cpu, last_login_user |
| | --logging=LOGGING | Activate file logging. Possible log levels: 1 = Normal Log; 2 = Debug Log |
| | --listObjectTypes | Lists all possible object types |
| | --listCategories | Lists all possible categories |
| | --usage | Show more helping information |

| Parameter (short version) | Parameter (long version) | Example |
|---|---|---|
| | --hardwareID=HARDWAREID | Import only by hardware/snmp ID from the ocs database. |
| -u | --user=USER | User who is authorized to execute |
| -p | --password=PASSWORD | Password to authenticate the specified user |
| -i | --tenantId=TENANTID | Client ID of the client to be used (default: 1) |
| -c | --config=CONFIG-FILE | Specification of the path to the configuration file |
| -h | --help | Help message for displaying further information |
| -q | --quiet | Quiet mode to disable the return |
| -V | --version | Output of the i-doit Console version |
| | --ansi | Forces output in ANSI format |
| | --no-ansi | Disables the output in ANSI format |
| -n | --no-interaction | Disables all interaction questions of the i-doit Console |
| -v / -vv / -vvv | --verbose | Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug |

## 7.1. Example

```
sudo -u apache php console.php import-ocs --user admin --password admin --tenantId 1 --databaseSchema ocs --hosts=device1,device2 --logging=2 --objectType C__OBJTYPE__CLIENT --objectTypeWhitelist=C__OBJTYPE__SERVER,C__OBJTYPE__CLIENT --categories=cpu,memory,net
```

Explanation:
**--databaseSchema:** Retrieves the OCS configuration from i-doit via schema name which will be used as import source. **--hosts:** Comma-separated list of hosts which will be searched and imported from the OCS database.
**--logging:** Specifies the log level of the import.
**--objectType:** All newly imported devices that could not be automatically identified are being created with the specified object type. Default from the configuration will be used if not specified.
**--objectTypeWhitelist:** Comma-separated list of object types that may be imported. **--categories:** Comma-separated list of categories that will be imported.

▎When importing, only one OCS database can be imported at a time.

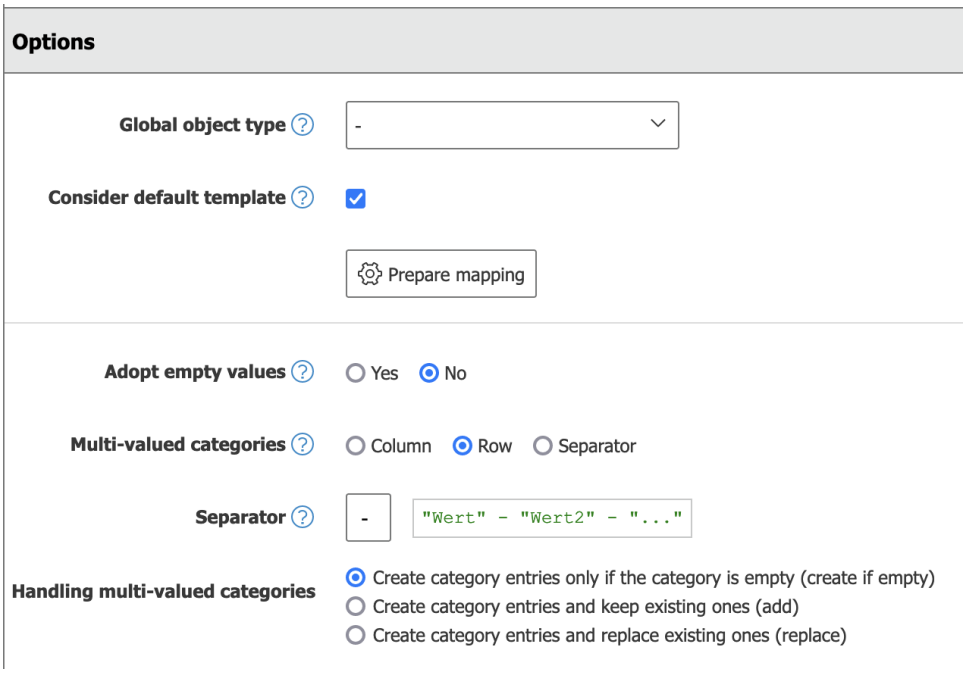
# 8. Import of SNMP data

Under **Extras → OCS-Inventory → Import SNMP/Custom** there is the possibility to import all objects from different SNMP types into the i-doit environment.

The mask shows the available SNMP types of the selected database configuration.

The import for the selected SNMP type can be started via the “Use for Import” button.

## 8.1. Prepare Mapping



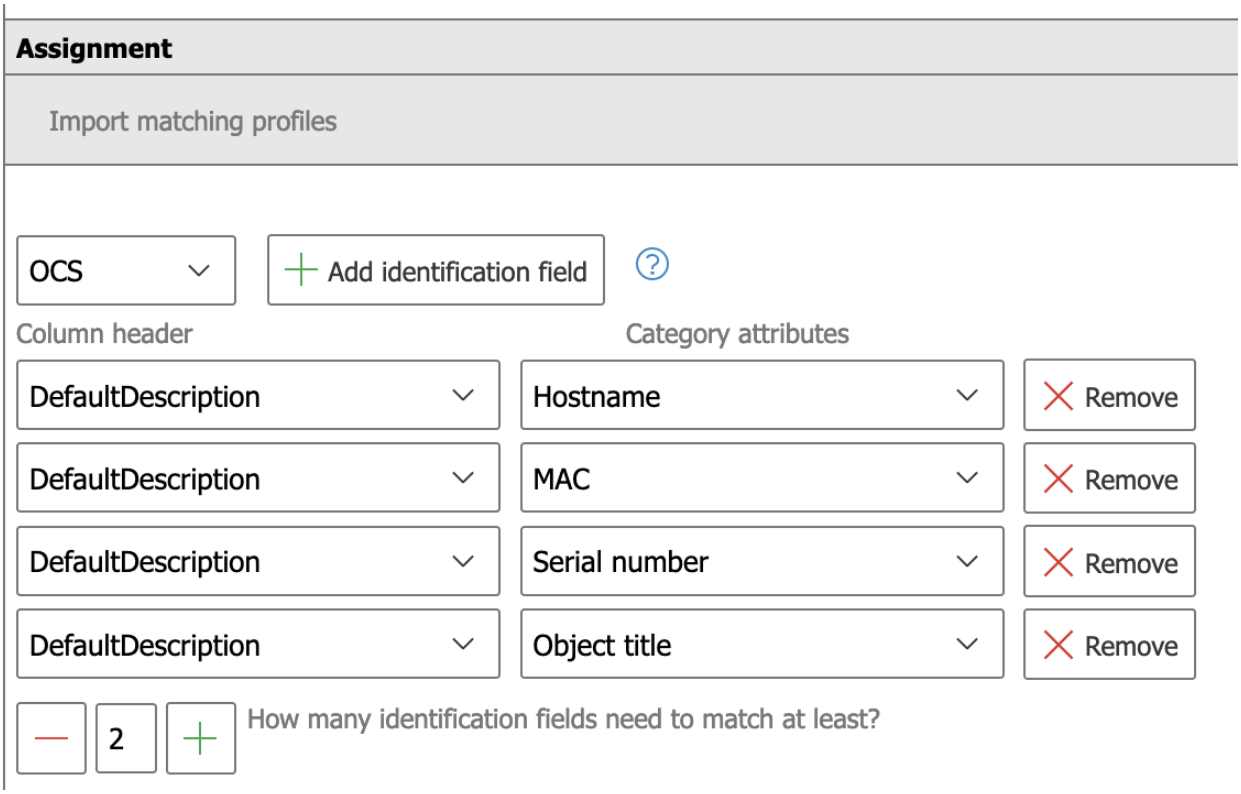

At the beginning of the import, some options must be set.

| Option | Description | Example |
|---|---|---|
| Global object Type | When you select an object type via the drop-down menu, all objects will be imported as this type of object. | Printer |
| Consider default template | The objects are created with the data from the default template. | 1/0 |
| Adopt Empty Values | When updating existing objects, this option defines how blank cells are handled.<br>**Yes**: Blank cells will be processed and overwrite existing attributes (if available).<br>**No**: Blank cells are ignored so that existing attributes (if available) are preserved. | Yes |
| Multi-valued categories | Defines how values for multi-valued categories are found. If there are no multi-value categories affected during import, the following options can be ignored.<br>**Column**: Values are present in multiple columns, e.g., “Title;IP-Address1;IP-Address2”<br>**Row**: Values are distributed across multiple rows, e.g.<br>“Title;IP-Address1”<br>“Title;IP-Address2”<br>**Separator**: Values are separated by a separator within a cell.<br>“Title;IP-Address1 - IP-Address2” | separator |
| Separator | Defines the separator used for multi-value columns. The field on the right provides a preview of how the data structure looks with the separator. The OCS Agent uses ” - ” as the default separator. If you wish to change this, the source files of the OCS Agent must be adjusted. | „ - „ |
| Handling multi-valued categories | Defines the handling for categories with multi-valued entries. If no categories of this type are involved in the data import, following options can be ignored:<br>Create category entries only if the category is empty (create if empty)<br>Create category entries and keep existing ones (add)<br>Create category entries and replace existing ones (replace) | create if empty |

By clicking the ‘Prepare mapping’ button you can reach the next step. ## 8.2. Assignment

### 8.2.1. Import matching profiles

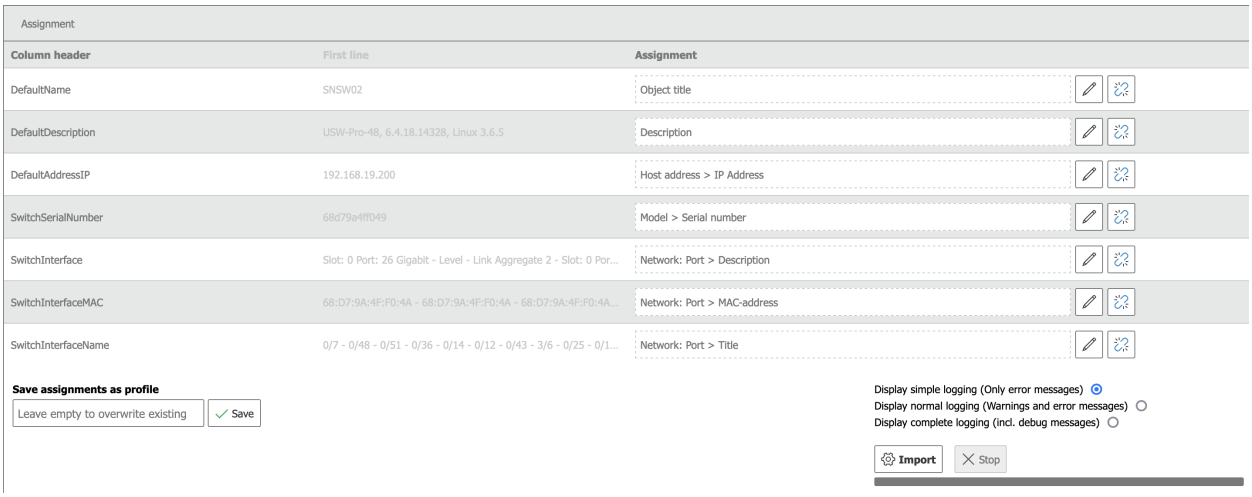
Provides a selection with the import matching profiles. The profile is used to update existing objects based on matching attributes instead of creating a new object.




objects-identifying

### 8.2.2. Assignment of Columns to Attributes

Each table column has its own row in the mapping. In this way, each column of the SNMP table can be linked to an attribute from i-doit. The pencil symbol activates the input field for selecting the associated attribute. The attribute can either be determined by selecting it from the drop-down menu or by entering its name directly in the field to activate the suggestion function. The entry is confirmed with the “Apply” button. To remove an assignment, use the chain symbol. Columns without an assignment are ignored during import.




Assignment of Attributes

## 8.3. Mandatory Attributes

Both the object title and the object type must be specified. If a global object type is defined in the data import options, it is only necessary to link the object title to a column in the assignment. If the global object type is not set, a link using the import data is also required. Otherwise, it is not possible

to import the data. The object type is set via its database constant (e.g. **C_OBJTYPE_SERVER**). Setting the name of the object type (e.g. **Server**) is not sufficient.

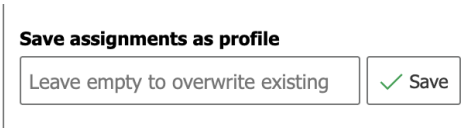
## 8.4. Start of Data Import

The level of detail for logging the import can be set beneath the mapping. The more extensive the logging is, the more time and resources are needed for the import. The logging of debug messages can be helpful for possibly required troubleshooting.

The Import button beneath the mapping is used to start the import. The time needed for the import depends on the extent of the information as well as the selected level of logging.

Once the import has been completed, information regarding the import as well as a confirmation of its completion will be indicated. The imported or updated objects are linked directly. The content of these objects can be changed manually anytime if needed.

## 8.5. Import Profiles




Finally, the import configuration can be saved in a profile by specifying a name. This avoids recurring work steps.
A saved profile can be selected and loaded or deleted in the profile selection in the top right-hand area.
If an import profile has already been loaded, it can be edited by clicking on the “Save” button without specifying a name.

# 9. Import from user-defined tables or views

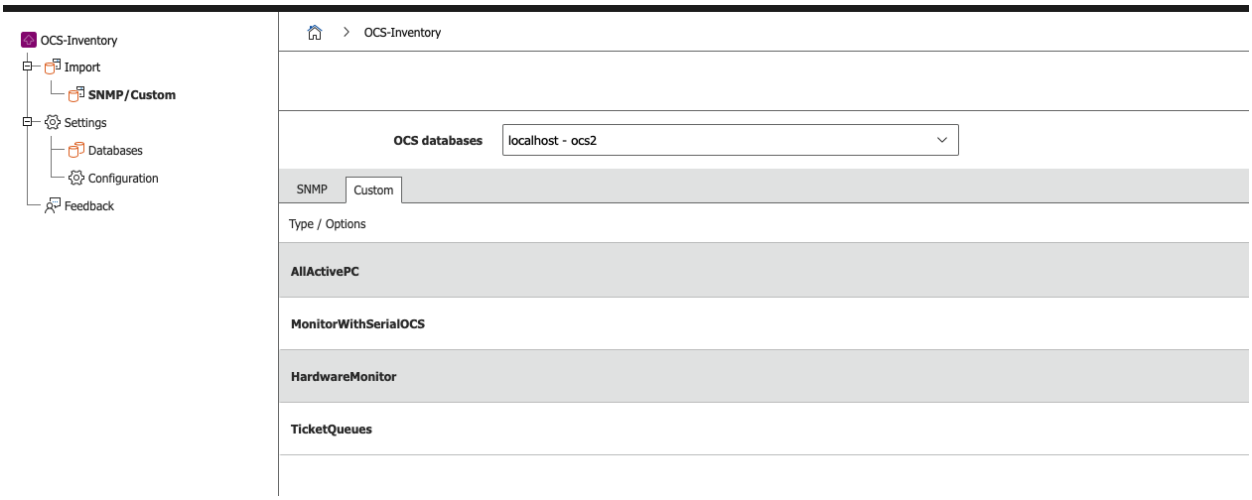
Similar to SNMP import, user-defined import allows you to import data from user-defined database tables or views.

This feature can be particularly helpful when information cannot be imported via the standard method using OCS add-ons. By creating appropriate views in the database, this information can be imported.

The procedure is exactly the same as for SNMP import.

## 9.1 Requirements

For tables or views to appear in the “Custom” tab, the following condition must be met: The name of the table/view must have the prefix “*custom_idoit_*”. In the overview of available tables/views, the name after the prefix appears:




Select Custom Table

## 9.2 Example View

The following statement creates a view that is valid for import. It lists all connected monitors that are linked to non-archived hardware.

```
CREATE VIEW `custom_idoit_MonitorWithSerialOCS`
AS
SELECT CONCAT('Display-', m.ID) AS MonName,
    m.serial AS SerialNumber,
    h.name AS HwName,
    h.id AS HwID
FROM monitors m
JOIN hardware h on h.id = m.hardware_id
WHERE h.archive IS NULL
```

The name consists of static text plus an internal ID to ensure uniqueness.

## 9.3 Import

The import procedure is the same as for SNMP, as described in 8.1 to 8.5.

# 10. SNMP / Custom Console Import

The import can be automated via the i-doit console, for example via a cron job. The relevant command is **import-ocs-snmp**. The following options can be displayed using the **–help** parameter:

| Parameter (short version) | Parameter (long version) | Description |
|---|---|---|
| | --databaseSchema= DATABASESCHEMA | Import from selected database schema name or ID. If not set default database schema will be used in the configuration. |
| | --tableName= TABLENAME | Import from selected table. Use OCS TYPE_NAME not TABLE_TYPE_NAME. |
| | --importProfileId= IMPRTPROFILEID | Profile which should be used to map import |
| | --multiValueMode= MULTIVALUEMODE | Multivalue mode. Possible modes are “row”, “column”, “separator” |

| Parameter (short version) | Parameter (long version) | Description |
|---|---|---|
| | --Separator= SEPARATOR | TSeparator of multivalue option |
| | --multi-value-update-mode= MULTI-VALUE-UPDATE-MODE | Multivalue update mode. Possible modes are “create-if-empty”, “add”, “replace” |
| | --default-template | Define if the default template should be applied |
| | --empty-values | Define if empty values should be applied |
| -u | --user=USERNAME | User |
| -p | --password=PASSWORD | Password |
| -i | --tenantId=TENANTID | Tenant ID [default: 1] |
| -c | --config=CONFIG-FILE | Config File |
| -h | --help | Display help for the given command. When no command is given display help for the list command |
| -q | --quiet | Do not output any message |
| -V | --version | Display this application version |
| | --ansi | Erzwingt die Ausgabe im ANSI-Format |
| | --no-ansi | Force (or disable –no-ansi) ANSI output |
| -n | --no-interaction | Do not ask any interactive question |
| -v / -vv / -vvv | --verbose | Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug |

To output a list of available SNMP import profiles, the command **import-ocs-snmpprofiles** can be used.

**The command import-ocs-snmp is also used to import user-defined tables!**

## 10.1 Example

```
sudo -u apache php console.php import-ocs-snmp --user admin --password admin --tenantId 1 --databaseSchema ocs --tableName="custom_idoit_MonitorWithSerialOCS" --importProfileId=6 --multivalueMode=separator --Separator "," -v
```

# 11. Contact & Support

Sector Nord AG
Edewechter Landstr. 123
26131 Oldenburg

Software maintenance includes support via phone (+49 441 39010 42), fax (+49 441 39010 11), remote (via Teamviewer and MS-Teams) and e-mail (service@sectornord.de) from Monday - Friday, 9:00h -17:00h, (4 hours response time - except on public holidays in the federal state of Lower Saxony).

Current information like prices for the subscription or current release notes around our OCS-AddOn for i-doit can be found at https://www.sectornord.de/en/ocs-inventory.html.

All information about support contracts can be found at:

https://www.sectornord.de/de/supportvertraege.html. (GERMAN)

# Changelog

1.0.0
---
- Moved OCS from idoit core to an own package

1.0.1
---
- Added sortable import table (#1)
- Bugfix: Import filter with specific categories does work correctly now (#2)
- Bugfix: Added error message again if the connection settings are incorrect (#3)
- Bugfix: Fixed link to database configuration in error message (#9)
- Optimized column-width of import table (#1)
- Added Feedback link (#5)
- Resized popup to select categories (#4)
- Added licensing (#6)

1.0.2
---
- Improved import button position (#17)
- Added page for licensing key (#27)

1.1.0
---
- Added option to change software name during import via Regular Expression (#33)
- Added option to prevent updating existing host addresses (#19)
- Added option to set object types that should not be imported (#7)
- Added option to set object cmdb states that should not be updated (e.g. scrapped) (#29)
- Added option to prevent importing IPv4 addresses (#31)
- Added free and used space for disks (#28)
- Fixed several bugs that caused too many logbook entries to be created (#26)
- Fixed bug with wildcards for tags to define the category (#13)
- Added support for i-doit 1.19 and php8 (#35, #39)
- More small improvements for better user experience (#34, #36)

1.1.1
---
- Added option to prevent updating existing serial numbers

1.2.0
---
- Added OCS category to identify i-doit objects imported from OCS
- Display new object type in dropdown if selecting 'recalculate object type'
- Import category 'Last logged user' from OCS
- Option to use custom software object types
- Fixed bug with cutting version number from software name
- Fixed bug when importing clients with loopback address
- Fixed bug if OCS delivers incorrect CPU speed
- Fixed layout issue if object has multiple IP addresses

1.2.1
---
- Bugfix: Changed type of category to improve settings
- Improved documentation

1.2.2
---
- Added support for i-doit 25

1.3.0
---
- Added SNMP import (OCS 2.0.7+ required)
- Bugfix: Fixed double import of graphic cards with no memory (#84)
- Added white- and blacklist for import objects with console (#85)
- Removed old SNMP import (#88)
- Bugfix: Fixed assigned software with different versions (#87)
- Improved logic for cutting out version number (#86)

1.3.1
---
- Fixed bug with blacklist from configuration (#96)
- Fixed creation of multiple version relations for applications (#97)

1.3.2
---
- Support for i-doit 27

- Bugfix: overwrite empty values (#98)

1.4.0
---
- Bugfix: Software with new version number now keeps the old license (#99)
- Fixed error message if no objects in ocs found (#101)
- Added new ocs category for connected devices: monitor, printer & input devices (keyboard, mouse...) (#32)
- Renamed old selectable category 'Devices' to 'Storage' (#32)
- Added folder structure for categories (#32)
- Added dynamic configuration for object type mapping (#70)
- Reorganized OCS configuration page (#70)
- Added option to ignore host addresses from unknown layer 3 networks (#71)
- Added option to skip configured applications by name during import (#69)
- Added import of monitors (#51)
- Added logbook source filter for OCS entries (#102)

1.4.1
---
- Added widget for OCS license warning (#104)
- Added option to filter monitor import from specific layer 3 networks (#103)
- Added option to configure preselected categories for import (#100)
- Fixed wrong sorting of import table (#105)
- Fixed net import if only empty category should be filled (#106)
- Fixed static image paths in configuration (#110)

1.4.2
---
- Fixed bug with blacklist during monitor import (#111)
- Fixed bug with SNMP tablesdesc containing whitespaces (#112)

1.4.3
---
- Fixed drag and drop of license file (#115)
- Updated icons to new i-doit design (#117)

1.4.4
---
- Fixed missing CMDB Event for creating IP or Host addresses (#124)
- Improved error handling if no ocs database is configured (#123)
- Added button to database config to check the connection to the OCS database (#119)
- Fixed internal config of ocs categories (#122)
- Fixed timeout during installation if many logbook entries exist (#126)
- Fixed combination of deleting network category without importing new network category (#132)
- Added option to control archived OCS objects (#125)
- Fixed permissions for snmp import profile (#133)
- Improved error handling for empty database config attributes (#118, #140)
- Renamed addon to 'OCS-Inventory' (#141)

1.5.0
---
- Use standard template for object creation if configured (#72)
- Show calculated objecttype for new objects in import list (#76)
- Do not overwrite objecttype if recalculation is selected but there is no rule configured (#76)
- Import OCS tag to the OCS category (#151)
- Added option to skip host addresses from DHCP (#138)
- Improved object type mapping from wildcards to regex filter (#142)
- Added button on the import page to scroll to the top (#18)
- Ignore hosts with addresses from archieved layer 3 networks if configured (#143)
- Improved validation for ocs ip address and subnet (#144)
- Moved OCS device categories to the OCS folder (#149)
- Refactoring AJAX handling for better performance (#147)
- Added missing language constant (#148)
- Fixed filter for datetime fields in OCS category in Report Manager (#150)
- Refactoring installation routine (#146)

1.6.0
---
- Added option to import custom tables (#155)
- Updated functions to php 8.4 (#156)
- Possible import of tables that are not connected to OCS (#159)
- Linked monitor objects to the ocs monitor category (#152)
- Fixed wrong displayed title in SNMP Import (#163)