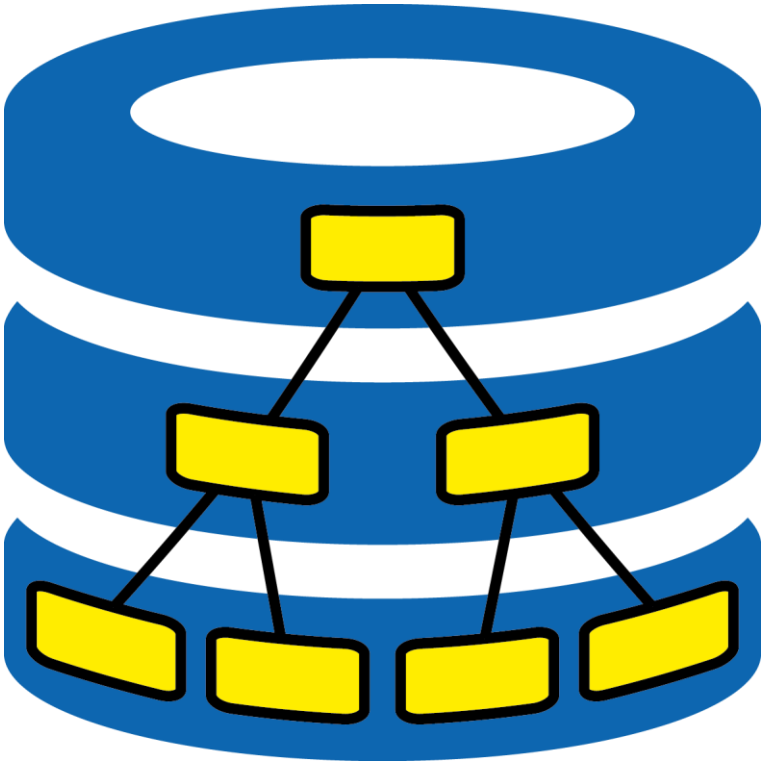
A.I.M. Automated Index Management

For **Microsoft® SQL Server®**



Version 4.1

ObviuZ
DATA SOLUTIONS

1 Table of contents

2 Preface

2.1 Target Audience

This user manual is intended for IT management and database administrators who are responsible for installing, configuring, and managing the software. IT management can use this manual to gain an understanding of the software's features and functions, and to plan for its deployment. Database administrators will use this manual to learn how to install and configure the software, and how to use it.

2.2 Trademarks

Microsoft®, **Windows®**, **MSDTC®** and **SQL Server®** are registered trademarks of **Microsoft Corporation**.

2.3 About this manual

IT management is advised to read at least *this* chapter; ***2 Preface***.

References inside this document are shown as a number with text in ***Bold and Italic*** as right above.

To get A.I.M. up and running the database administrator needs to perform all actions in chapter ***4 Getting Started***.

Buttons are presented like this: [Save].

Tips are presented in this format.

2.3.1 Legal notice

2.3.2 Glossary of terms

Term	Meaning
Console	The software program that is used to control A.I.M.
DBA	Database administrator
Instance or SQL Instance	A specific **SQL Server** installation on a Windows Server, a Windows Server can host multiple SQL Instances.
Master (instance)	An instance that is hosting the A.I.M. repository, an organization *can* have multiple A.I.M. Masters. A Master can also be a Target at the same time.
Member (database)	Any database on a Target that is marked as Member and therefore managed by A.I.M.
MSDTC	Microsoft Distributed Transaction Coordinator
Server	Window Server, can host zero or more SQL Instances.
Target (instance)	Any instance that hosts databases managed by A.I.M.
Repository	The database that A.I.M. uses to store its data, which can be a Member at the same time.

2.3.3 We would like to hear from you

Whether you have questions, doubts, ideas about improvements, complements or complaints don't hesitate to contact us, email: office@obviuz.com.

3 Introduction

3.1 Purpose of A.I.M.

A.I.M. stands for Automated Index Management and it does a few important things that other tools do not:

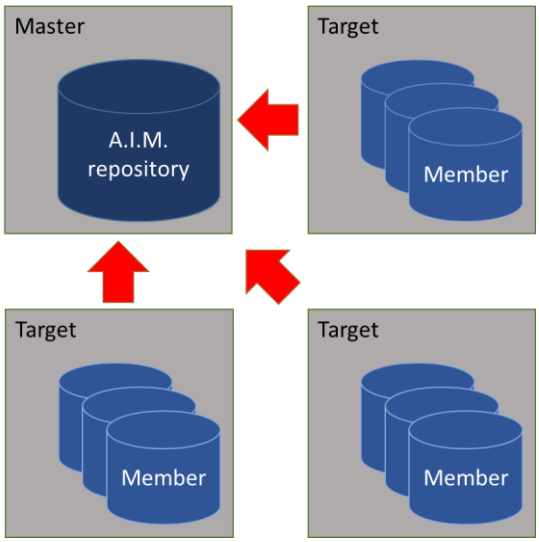
- Propose new indexes that would be useful but do not exist yet (Missing indexes).
- Identify indexes that are no longer useful (Obsolete indexes).
- Automatically execute the necessary actions to deal with both categories or help you doing that manually if preferred.

A.I.M. intelligently reduces routine, time-consuming tasks for database administrators (DBAs) while providing them with comprehensive insights and optimizations, ensuring peak performance, reduced storage requirements and improved productivity.

3.2 Overview

A.I.M. establishes connections with servers designated as Targets by the DBA. On these servers, it gathers information regarding existing and missing indexes in databases identified as Members by the DBA. This data is stored and maintained in a repository.

While a single A.I.M. instance can effectively manage an organization's databases, deploying multiple A.I.M. Masters can offer additional benefits in terms of security, scalability, and compliance. The choice between a single-Master or multi-Master deployment depends on the specific requirements of each organization, a typical setup:



Note: The Master can also serve as a Target with its own set of Members. The A.I.M. repository is typically also included as a Member.

A.I.M. periodically re-evaluates indexes and updates their categorization based on usage patterns and retention policies. This dynamic process ensures that the index collection remains optimized for the current workload and helps maintain optimal performance.

3.3 Key features

1. Automated Index Management: Automate the creation and removal of indexes, freeing up DBA time and effort for more strategic database optimization tasks.
2. Performance Enhancement: Maintain an optimal index collection that is tailored to the actual usage patterns of your databases, ensuring peak performance and minimizing query slowdowns.
3. Data-Driven Insights: Gain valuable insights into your indexes with a comprehensive categorization system that provides clear insights into their effectiveness and usage.
4. Effortless Index Recovery: Restore dropped indexes with ease if you see fit.
5. Overall performance improvement.

3.4 General information

For a further introduction of A.I.M. and more information about its purpose, we refer to the website aim.obviuz.com.

4 Getting Started

4.1 Introduction

Completing this entire configuration chapter is crucial to ensure A.I.M.'s full functionality.

Although an incomplete setup may not cause immediate problems, it must be said that A.I.M. will not be fully functional until you complete the basic configuration.

After requesting a License, you need to wait for it to arrive before you can proceed. We'll try to provide that as soon as possible.

Once you have received the License data, enter it and complete the basic configuration to enable A.I.M. to function fully.

4.2 System requirements

- .NET Framework 4.8 on computers running A.I.M. Console.
- **SQL Server** 2014 Developer or Standard Edition with: Service Pack 1 applied and up and running for all Instances involved. A higher version is recommended.
- .NET Framework 3.5 on the Master instance.
- **MSDTC** running on all servers involved, this is essential to transport the necessary data from Target to Master.
- Around ¼ GB storage per member database, of course this strongly depends on the usage of the specific database and the (potential) number of indexes in it.
- The Instance running the A.I.M. Repository must have CLR enabled, the config procedure takes care of that.
- The provided A.I.M. database needs to be trusted after installation; the config procedure takes care of that.
- A.I.M. uses SQL Server Authentication Mode on Targets, make sure that it is enabled.
- Make sure that the appropriate ports between program and ***all database instances*** (Master and Targets) are not blocked by any firewall rule.
- Database Mail must already be configured and running, please refer to the appropriate documentation here.

4.3 What's in the package

Excellence can be compact; this is the complete set:

- AIM.mdf; primary data file.
- AIM_log.ldf; transaction log file.
- AIM_Console.exe; program file.
- ODSCrypt.dll; encryption routines file.

4.4 Installation

1. Copy the supplied database files to the designated folders.
2. Attach the database. Note that you will be the default owner. It is recommended to change the ownership to the appropriate user after attaching the database. If you accept the default (your current login), the Configuration will handle it later.

3. Copy the supplied program file and ODSCrypt.dll to the designated folder. Registration of ODSCrypt.dll is not necessary; simply place it in the same folder as the program file.

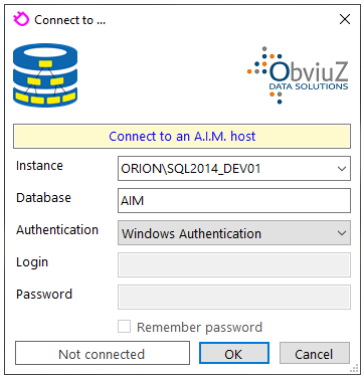
4.5 Configuration

4.5.1 License request

Without a valid license, the Console will be limited to enabling license request or license entry, and no other functionality will be available.

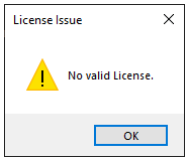
The License can be a free evaluation license or a paid regular license:

- Start the Console and enter the name of the instance where the A.I.M. Repository is located:

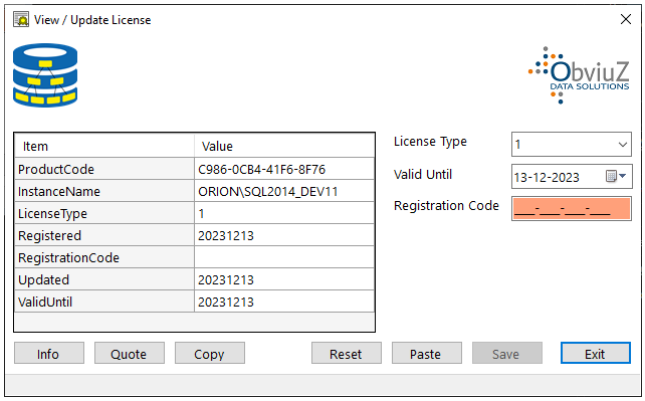


Choose Windows Authentication or enter your (sysadmin) credentials and click [OK].

- After you connect for the first time, you'll see this popup because there is no License configured yet:



- Click [OK] and you'll see this form:



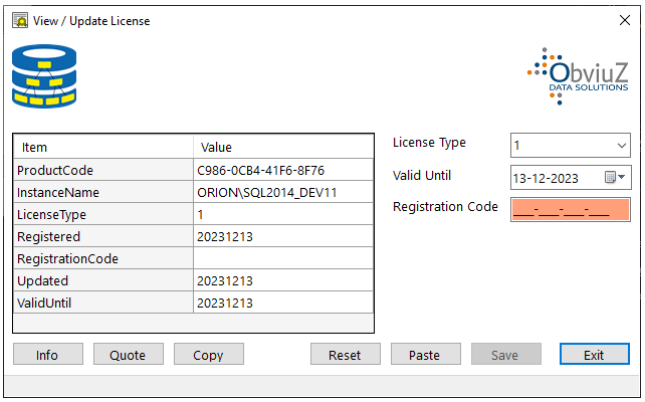
- Using the appropriate buttons, you can either:
  a) Read the info (click [Info])
  b) have A.I.M. open a new mail with a quotation request an all data in it (click [Quote])
  c) copy all necessary data to the clipboard so you can save it and send it in an email from another computer (click [Copy]).

Send us the request. Once the License data has been received you can finish this chapter.

4.5.2 License configuration

Upon receiving your request, we will provide you with the License data. Please note that the registration code is generated exclusively from this data combination. Any alteration will render the license invalid.

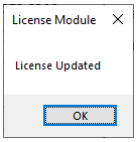
Once the License data has been received, start the Console again as described in the previous chapter and enter the License data in this form:



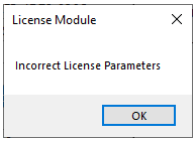
Enter the following data:

- Type in the *License Type* of select it from the ComboBox.
- Type in the *Valid Until* or select from the DateTimePicker.
- Type in the *Registration Code* or use [Paste].

Click [Save] to validate and save the License data. A valid combination results in this pop-up and the License is activated:



Should the combination of License data not be correct, then you'll see this:



In case you have difficulty with the License, contact us.

4.5.3 Database configuration

Several database server options require configuration, and A.I.M. Jobs must be created. The subsequent procedure addresses both tasks.

Open a query window and execute…

```
USE AIM;
exec hlp.up_Config;
```

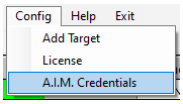
You'll get results *similar* to this:

DtCfg	Result
2023-12-17 10:39:07.830	A.I.M. by ObviuZ © 2018/2023 self configuration:
2023-12-17 10:39:07.847	- parameters already created
2023-12-17 10:39:07.847	- 0 parameter(s) updated
2023-12-17 10:39:07.847	- 0 parameter(s) deleted
2023-12-17 10:39:07.877	- CLR is already enabled
2023-12-17 10:39:07.877	- trustworthy is already ON
2023-12-17 10:39:07.877	- db-owner is 'ORION\SqlService' (unchanged)
2023-12-17 10:39:07.923	- JobCategory 'A.I.M. - Automated Index Manageme...
2023-12-17 10:39:07.940	- Job 'A.I.M. - Inventory Indexes': added
2023-12-17 10:39:08.003	- JobStep 'A.I.M. - Inventory Indexes Steps': added
2023-12-17 10:39:08.033	- JobSchedule 'A.I.M. - Inventory Indexes Schedule'...
2023-12-17 10:39:08.143	- Finalization for Job 'A.I.M. - Inventory Indexes': Done
2023-12-17 10:39:08.143	- Job 'A.I.M. - Drop Indexes': added
2023-12-17 10:39:08.160	- JobStep 'A.I.M. - Drop Indexes Steps': Done!
2023-12-17 10:39:08.160	- JobStep 'A.I.M. - Drop Indexes Steps': added
2023-12-17 10:39:08.160	- Finalization for Job 'A.I.M. - Drop Indexes': Done
2023-12-17 10:39:08.160	- Job 'A.I.M. - Create Indexes': added
2023-12-17 10:39:08.160	- JobStep 'A.I.M. - Create Indexes Steps': Done!
2023-12-17 10:39:08.160	- JobStep 'A.I.M. - Create Indexes Steps': added
2023-12-17 10:39:08.173	- Finalization for Job 'A.I.M. - Create Indexes': Done
2023-12-17 10:39:08.173	- Job 'A.I.M. - Maintenance': added
2023-12-17 10:39:08.173	- JobStep 'A.I.M. - Maintenance Steps': added
2023-12-17 10:39:08.173	- JobSchedule 'A.I.M. - Maintenance Schedule': ad...
2023-12-17 10:39:08.173	- Finalization for Job 'A.I.M. - Maintenance': Done
2023-12-17 10:39:08.173	- Job 'A.I.M. - Summary Report': added
2023-12-17 10:39:08.173	- JobStep 'A.I.M. - Summary Report Steps': added
2023-12-17 10:39:08.190	- JobSchedule 'A.I.M. - Summary Report Schedule': ...
2023-12-17 10:39:08.190	- Finalization for Job 'A.I.M. - Summary Report': Done
2023-12-17 10:39:08.190	- License was initialized, now it needs to be registered

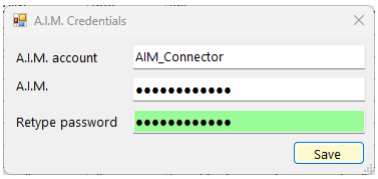
4.5.4 Default A.I.M. Credentials

To streamline the registration process of instances (Targets), A.I.M. offers the option of utilizing a default set of credentials stored for subsequent Target additions. This approach eliminates the need for repetitive credential entry and enhances user convenience. Alternatively, A.I.M. supports the manual entry of credentials for each Target at the time of registration.

To enter the default credentials now, select this option from the menu:



… the following form will appear:



Just enter the Credentials and click [Save].

Please note that credential validation and checking is not performed at the time of saving the default credentials. Instead, validation occurs when attempting to connect to a Target using the stored credentials.

Additionally, modifying the default credentials does not affect the previously saved credentials for already registered Targets.

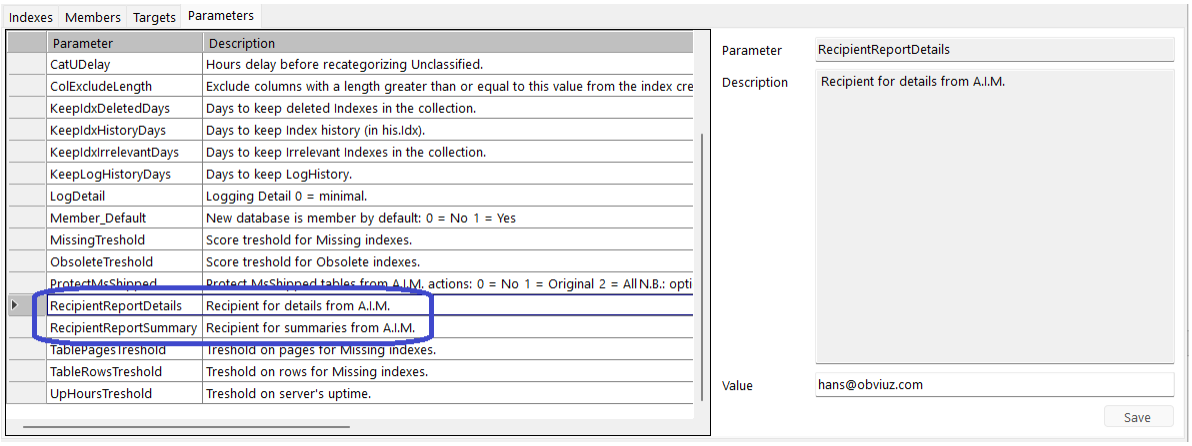
4.5.5 Email addresses

A.I.M. sends Reports:

- A detailed report after every Inventory run but only when there is something new or changed.
- A summary report every week.

For that to work, you need to configure the relevant email addresses. As indicated in paragraph ***4.2 System requirements***, Database Mail should already be configured and working.

Go to the Parameter tab and enter valid email addresses for these two Parameters:



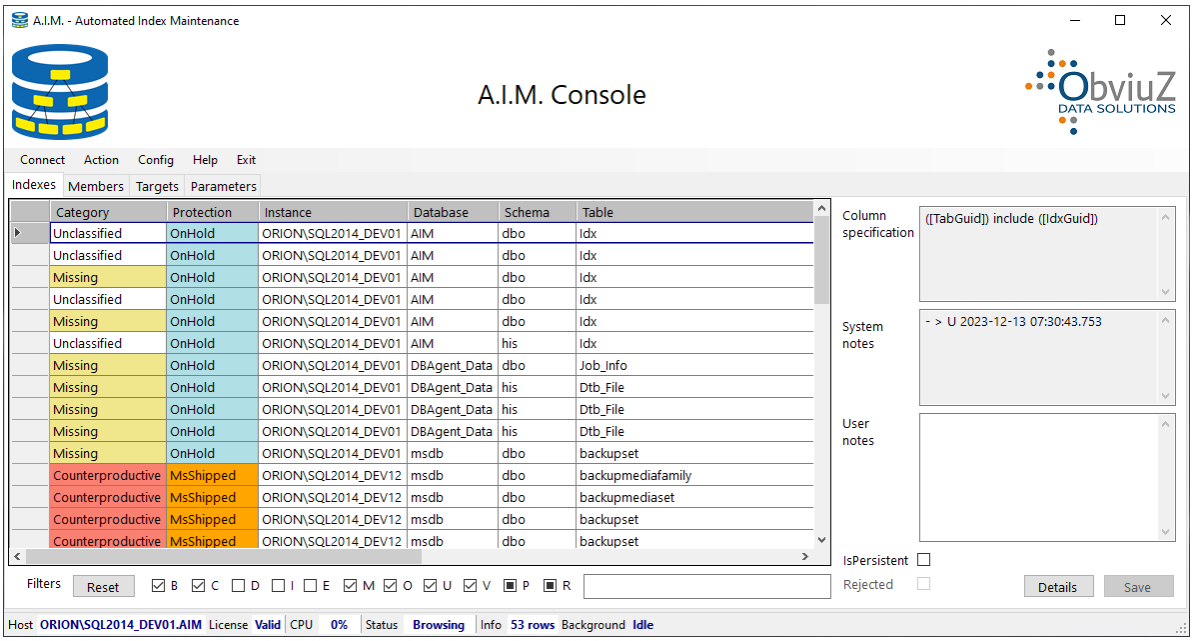
With the completion of the configuration, A.I.M. is technically ready for deployment.

However, before you can observe the results, you will need to establish at least one Target with one or more Members. This process is described in detail in Sections ***6.3.2 Add Target*** and ***6.2 Members*** respectively.

Nevertheless, we highly recommend to keep on reading the entire manual chapter by chapter to gain a comprehensive understanding of A.I.M.'s features and functionality.

5 User Interface

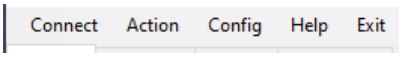
5.1 Overview of the user interface



This shows the initial window with indexes and default filtering, of course your Console will initially show no indexes until you registered at least one Target.

5.2 Description of the main components of the user interface

5.2.1 Menu bar

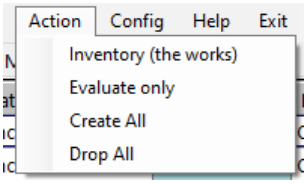


The Menu functions are described below.

5.2.1.1 Connect

Use this to connect to an A.I.M. host, just like you did in paragraph ***4.5.1 License***.

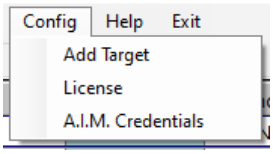
5.2.1.2 Action



- *Inventory (the works)* will perform a full inventory and evaluate it all.
- *Evaluate only* will (re)evaluate, which can be useful to see the changes after changing Parameters.
- *Create All* will create all appropriate indexes.
- *Drop All* will drop all appropriate indexes.

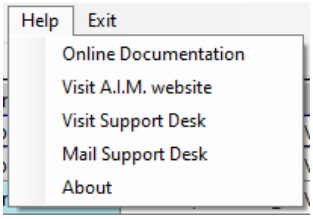
Note: Any of these choices will perform just the same actions as the corresponding jobs, but if you don't want to wait for those to run, you can run these from the Menu.

5.2.1.3 Config



- *Add Target* opens a window to specify new Targets, described in ***6.3.2 Add Target*.**
- *License* opens a window to view or update License info, see ***4.5.1 License request, 4.5.2 License configuration*** and ***6.5 License***.
- A.I.M. *Credentials*, see ***6.6*** A.I.M. ***Credentials***.

5.2.1.4 Help

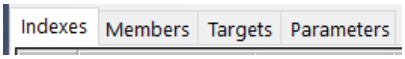


- *Online Documentation* speaks for itself.
- *Visit A.I.M. website* speaks for itself.
- *Visit Support Desk* takes you to the support site where you can create a ticket for support.
- *Mail Support Desk* creates a support ticket.
- *About* shows information about the product.

5.2.1.5 Exit

Closes the application, you'll be notified of any pending changes and have the opportunity to save them.

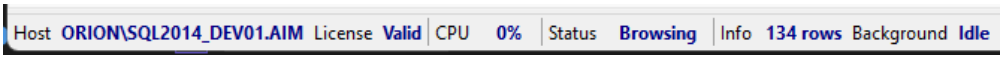
5.2.2 Tab control



These tabs are organized based on the frequency they probably will be used.
Logically, the initial use is from right to left where it concerns the collection: Targets first, then Members and finally Indexes.

Use this tab control to navigate, all tabs have master and detail panels and one or more buttons, these Tabs are described in detail in chapter ***6 Using the Software***.

5.2.3 Status bar



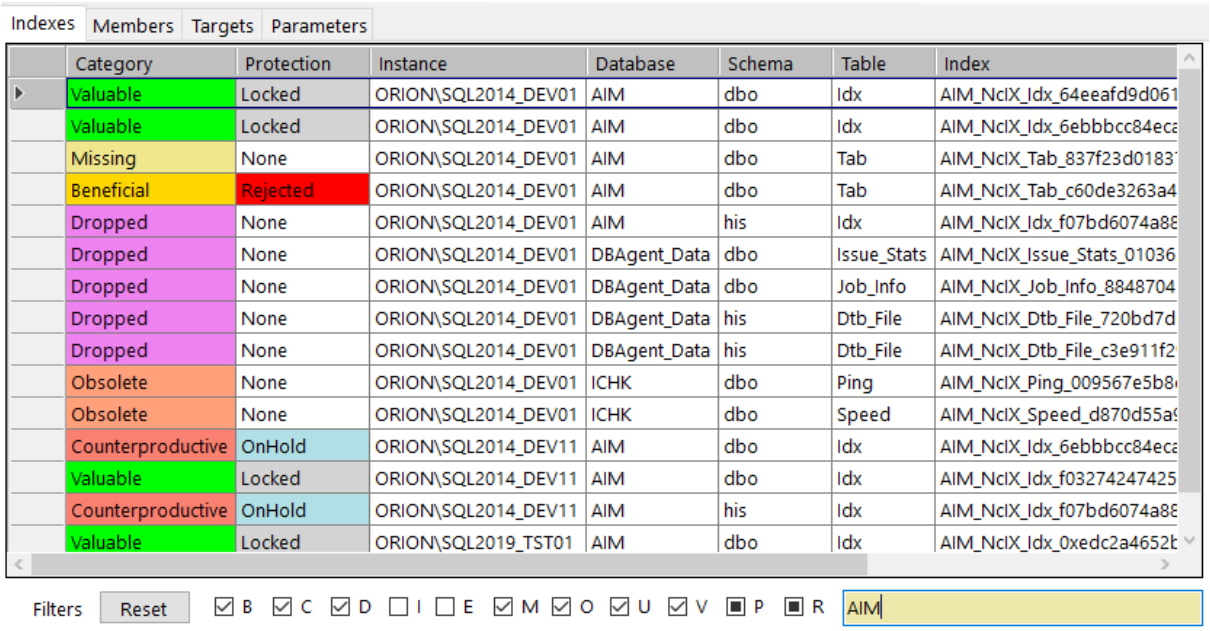
Showing:

- The Host (or Master instance)
- License status
- CPU usage of the Host
- Current Status of the Console
- Number of rows currently displayed
- Anything running in the background

6 Using the Software

To get A.I.M. to work, you need to specify which Instances (Targets) it needs to inventory, and which Databases (Members) on those Instances should be included. Only after that is done, Indexes are inventoried.

6.1 Indexes



What you may immediately notice are the colors, what do they mean? They are added to give a quick impression of the different types / situations displayed, for details about colors, see ***8.1 Category*** and ***8.2 Protection***.

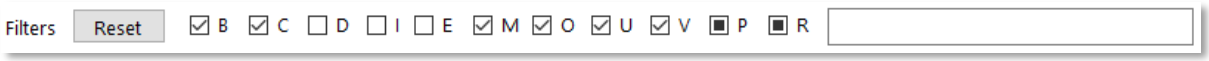
Both Category and Protection colors are passed on to the Details form.

6.1.1 Left panel

6.1.1.1 Index list

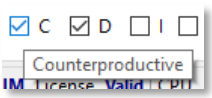
Category and Protection in the Index list are colored so you can easily distinguish the different qualifications.

6.1.1.2 Filters pane



The Filters can be used to filter on Category, characters **B** through **V** (include or exclude), and **P**ersistent and **R**ejected which have 3 states: Include, Exclude or indifferent (as shown above).

Hints are enabled so you can see what is what when hovering over it:



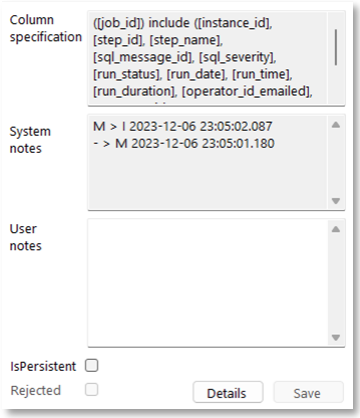
The textbox (in the example in paragraph ***6.1 Indexes*** currently filtering on *AIM*) allows you to enter a wildcard to filter on Instance, Database, Schema, Table or Index names.

Deleted, **I**rrelevant and/or **E**ssential are deselected by default as they are generally not considered to be very relevant in the context of this program.

Including **D**eleted, **I**rrelevant and/or **E**ssential indexes can result in a relatively large number of Indexes listed, hence slow down the Console.

Double-click on an index, or clicking on [Details] opens the Details form (see ***6.1.3 Details***).

6.1.2 Right panel



This panel displays essential details of the currently selected index, allowing you to add or modify *User notes*, designate the index as *Persistent* or *Rejected* where applicable.

6.1.2.1 Column Specification

ReadOnly Column specification listed in the format as used by the create statement.

6.1.2.2 System Notes

ReadOnly System notes, like it says: notes added by the system. These show all Category modifications, latest on top.

6.1.2.3 User notes

Add or edit notes.

6.1.2.4 IsPersistent

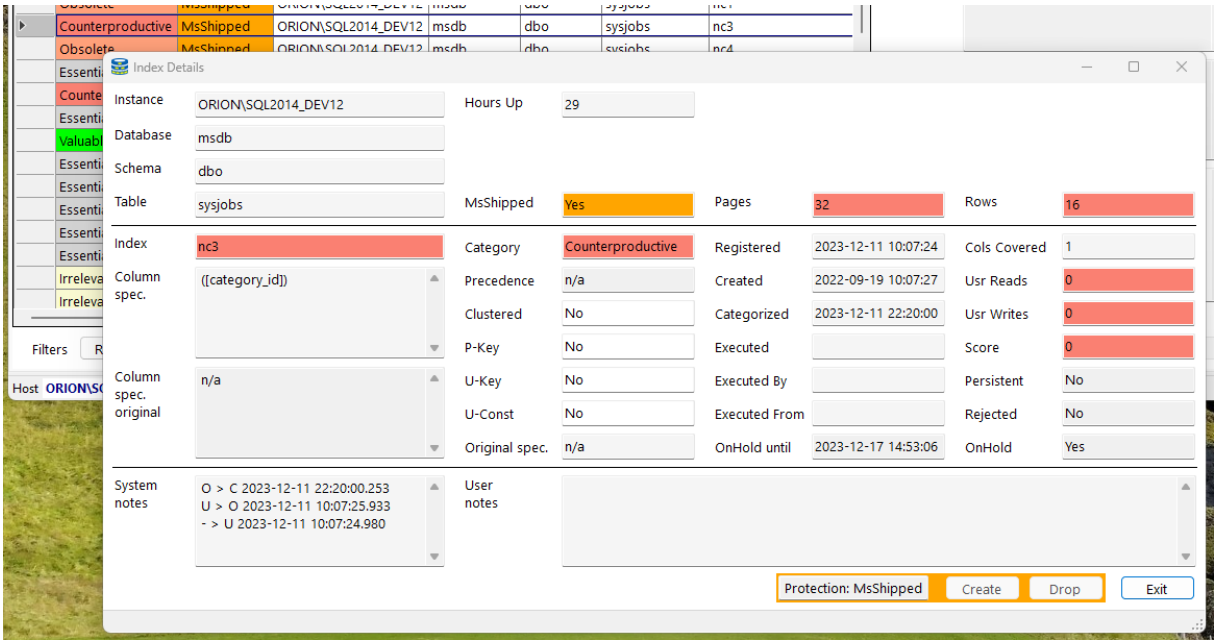
Initially off. If you always want to keep a certain index, you can switch this option on. This will make sure it will not be dropped even if it qualifies to be dropped and the system is configured for automated management. Not available on Missing or Beneficial indexes.

6.1.2.5 Rejected

Initially off. If an index is qualified as Missing but you do not want it to be created, you can reject it by checking this box.

The [Details] button opens the comprehensive *Details form* while the [Save] button saves any changes to the 3 previously mentioned items.

6.1.3 Details



Some Details fields adopt the Category colors from the Index list; those are the Index name and Category and the fields upon which the Category is based.

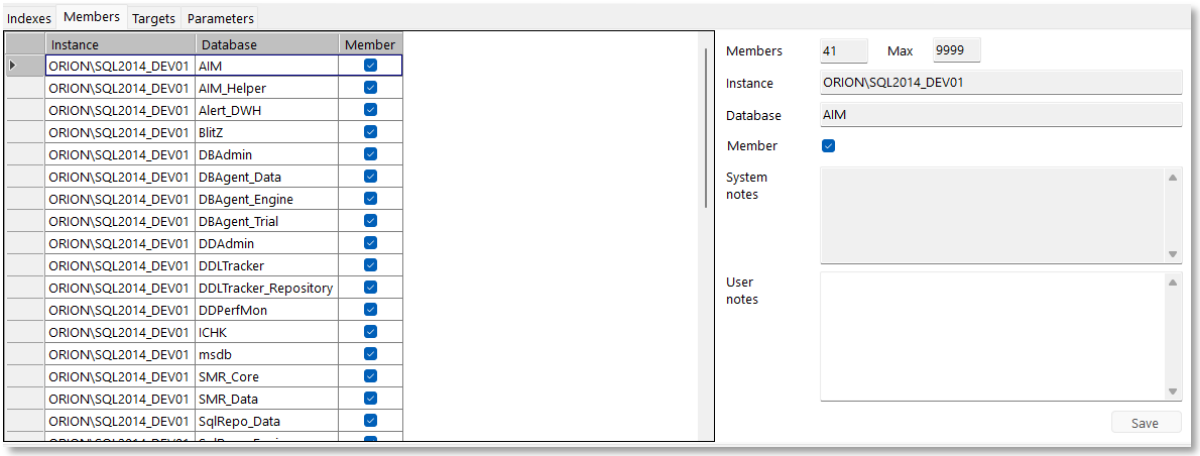
The protection fields adopt the protection color from the Index list. For detailed info on Protection, see ***8.2.6 How Protection works in the Console***.

Most columns are self-explanatory or explained earlier, the ones that are not are described next:

Column	Description
Hours Up	Number of hours since this Target started.
Column spec. original	Only applicable for Missing and Beneficial indexes. This is the set of columns that SQL Server originally proposed for the Missing index. Depending on settings, the used Column specification can be adopted, or some columns can be filtered out.
Precedence	Only applicable for Missing indexes, when there are multiple Missing indexes per table, one is marked as Precedent, the others will not become Beneficial and not Created.
Original spec.	Only applicable for Missing indexes; indicates if *Column spec.* is identical to *Column spec. original*.
Created	DateTime that the index was created as far as known, this is not reliable because regular index management can recreate indexes.
Executed	Only for Created and Dropped indexes, the DateTime when this happened.
Executed from	The process that executed it, Console or Agent.

Column	Description
OnHold until	Applicable for specific Categories and based on Parameters that define retention times. This DateTime indicates when recategorization can take place, or an index can be created or dropped.
Cols Covered	Number of columns covered in the index, only applicable on Existing and Missing / Beneficial indexes.
Usr Reads	Number of read actions (estimated for Missing / Beneficial).
Usr Writes	Number of write actions, existing indexes only.
Score	• Existing indexes: Reads – Writes • Missing indexes: based on estimates.
OnHold	Yes if *OnHold until* lies in the future.

6.2 Members



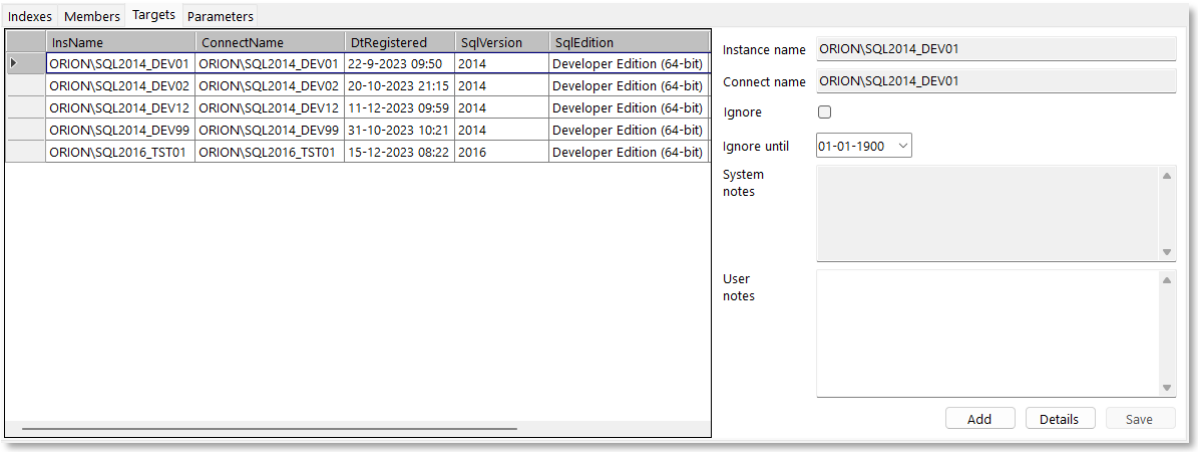
The **master**, **model** and **tempdb** databases are excluded from being Members. Only the **msdb** system database is eligible to be a Member; the philosophy behind that stems from the fact that depending on certain settings some tables can get quite large, think about history tables for back-ups or jobs.

The left panel shows a list of all (potential) members, i.e., all databases of all Targets, column *Member* shows if it actually is a Member.

The right panel shows the current number of Members and the max. Licensed number of Members on top, and some details about the currently selected Database. If offers the ability to change *Member*ship (Check/Uncheck), enter or edit *User notes* and save any changes.

After adding or deleting a Member, there will be no automatic re-inventory.

6.3 Targets

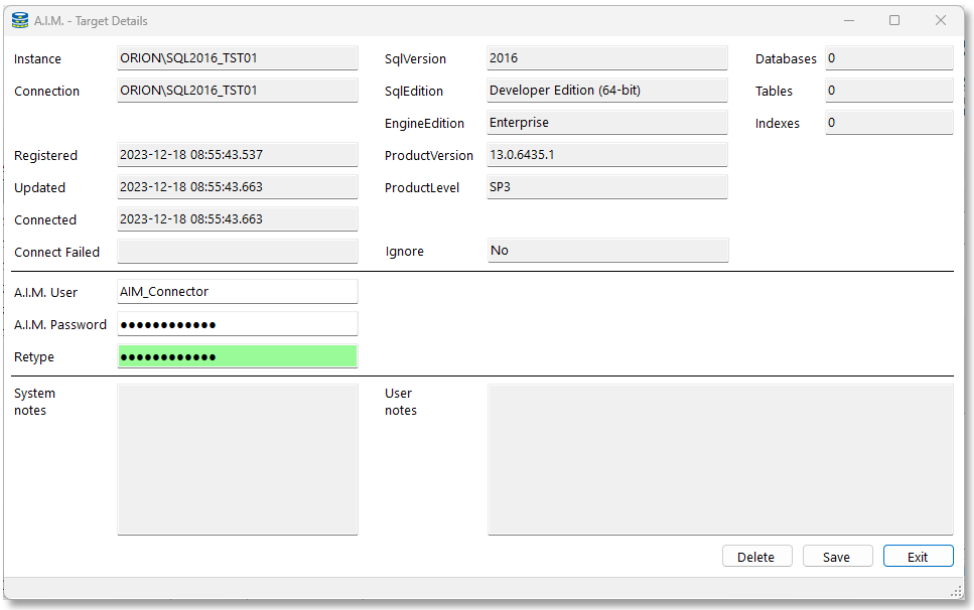


Left panel shows the list of currently defined Targets. Right-clicking shows all the details for the currently selected Target in the *Target Details* form.

Right panel shows some details for the currently selected Target and offers the option to temporarily *Ignore* it (excluding it from being inventoried), enter or edit *User notes* and save any changes using the [Save] button.

Finally, there's an [Add] button to open the *Add Target* form which is also available from the *Config* menu, see ***6.3.2 Add Target*** and a [Details] button to also open the *Target Details* form.

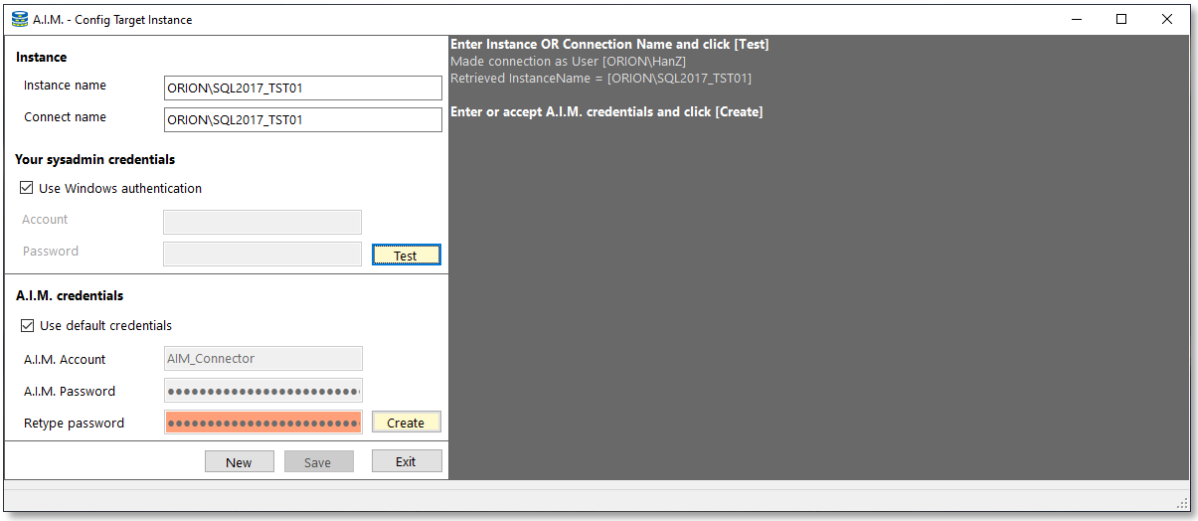
6.3.1 Target Details



All details are shown, and you have the option to change the Credentials or Delete the Target.

You cannot change *Ignore* or *User notes*, these you can only change through the parent form.

6.3.2 Add Target



This form provides a guided process for adding new Targets, with instructions in bold-white and results in grey or bold-yellow for emphasis:

6.3.2.1 Instance pane

If you can use the **SQL Server** instance name to connect to the Target, enter it as *Instance name* and you can leave *Connect name* empty, it will be copied.

However, if you need to use a different string for the connection instead of the instance name (e.g. `TCP:192.168.1.101,8002`), leave *Instance name* empty and enter that string as *Connect name*, the Instance name will be retrieved and filled in.

6.3.2.2 Your sysadmin credentials pane

As already indicated, you need to have Sysadmin privileges on all Instances involved.

If you use Windows authentication, you can leave the sysadmin credentials as they are.
Otherwise, enter your **SQL Server** credentials.

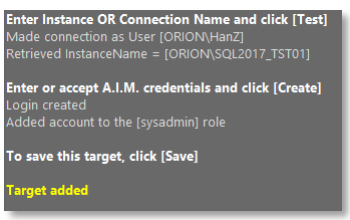
Then press the [Test] button to confirm they are valid. The form should then appear as shown above.

6.3.2.3 A.I.M. Credentials pane

If you did not enter the default A.I.M. credentials before **or** you want to use non-default credentials, you need to uncheck the *Use default …* checkbox and enter the correct credentials here.

Next, press the [Create] button.

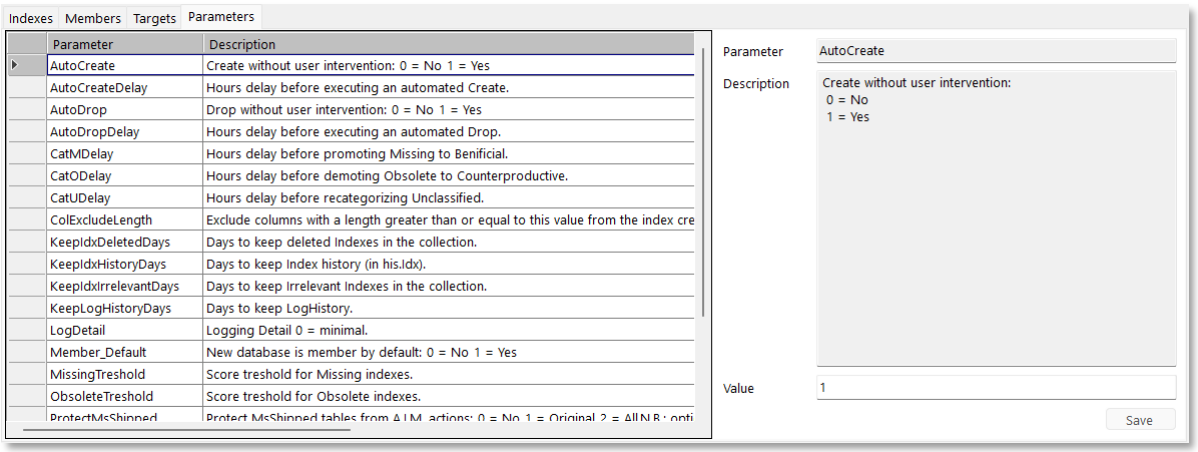
Finally, press the [Save] button to store the Target, right panel should be similar to this:



6.3.2.4 *Inventorying new Targets*

Upon closing this form, new Targets will be automatically inventoried instantaneously, providing immediate visibility into all Target's indexes in one Details report and Instance Details, including Version, Edition, and other relevant information in the Target list.

6.4 Parameters



Parameters come preconfigured, except for the mail addresses used for the reports.

In this form you can change any of the listed values but be sure you know what the consequences are because a lot of functions depend on them and some of them are used in conjunction with each other.

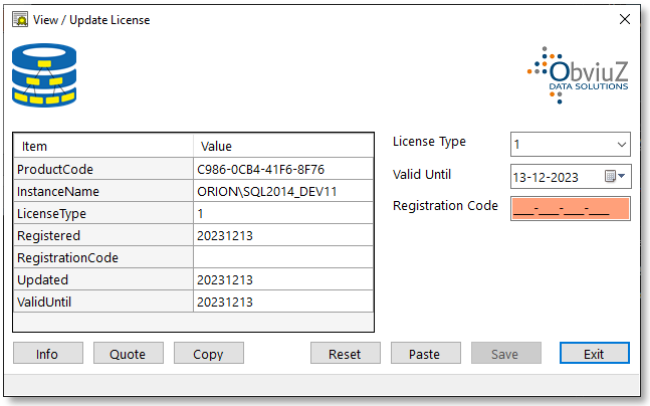
If you want to reset all parameters (except mail addresses) to their factory settings, see ***10.1 Reset Parameter values***.

6.5 License

The process of entering License data is already explained in section ***4.5.2 License configuration***.

This paragraph serves as a more in-depth elaboration on the topic, providing additional insights and clarifications.

Every A.I.M. Repository deployment requires a valid License. Licenses are issued for one year, unless otherwise specified, and are bound to the specific Master (or Host) instance where the initial installation was performed.
Copying or moving the repository to a different instance will render the copied or moved License invalid; in such cases, a new License will be required. If you encounter any issues related to License management, please contact Support, and we'll assist you in resolving the matter.



Displayed items:

- *ProductCode*; a unique code for this Product installation.
- *InstanceName*; the instance in which this License was configured.
- *LicenseType*; max. number of Members for this license.
- *Registered*; date this License was activated.
- *RegistrationCode*; the currently registered Registration Code.
- *Updated*; date this configuration was last updated.
- *ValidUntil*; License end date.

Editable items:

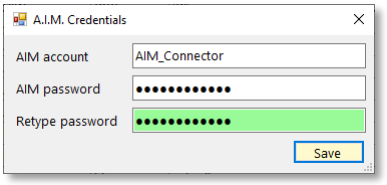
- *License Type*; max. number of Members for the new license.
- *Valid Until*; new License end date.
- *Registration Code*; new License's Registration Code.

Buttons:

- [Info]; shows information about getting a (new) License.
- [Quote]; opens your email client with a Quotation request.
- [Copy]; copies quotation data to the clipboard.
- [Reset]; deletes the current License data.
- [Paste]; paste clipboard content into Registration Code field.
- [Save]; save (and verify) the entered values.

- [Exit]; exits the form, notifies you of any pending changes.

6.6 A.I.M. Credentials



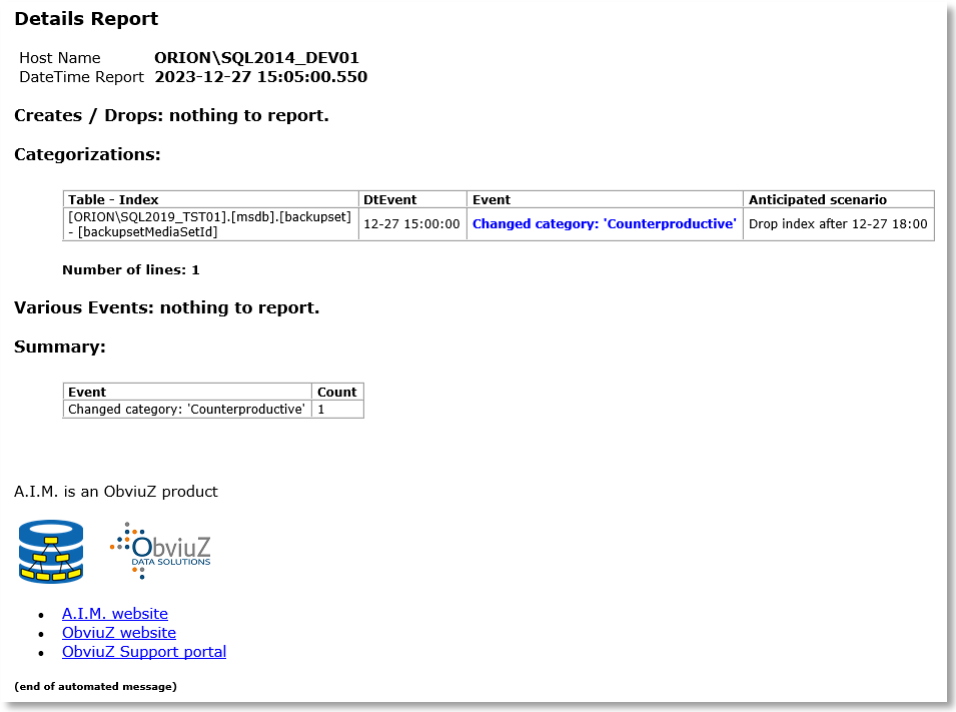
The credentials you enter here are the default credentials for future use. These credentials will be used to configure new Targets.

Existing Targets have their own credentials saved separately, which correspond with the credentials used for their respective accounts. Changing the default credentials will not affect the credentials saved for existing Targets.

If you need to change the credentials for an existing Target, you will need to update the login credentials directly at the Target and also update the stored credentials in the Target details form (see ***6.3.1 Target Details***).

7 Reports

7.1 Details report



Details Report

Host Name **ORION\SQL2014_DEV01**
DateTime Report **2023-12-27 15:05:00.550**

Creates / Drops: nothing to report.

Categorizations:

Table - Index	DtEvent	Event	Anticipated scenario
[ORION\SQL2019_TST01].[msdb].[backupset] - [backupsetMediaSetId]	12-27 15:00:00	**Changed category: 'Counterproductive'**	Drop index after 12-27 18:00

Number of lines: 1

Various Events: nothing to report.

Summary:

Event	Count
Changed category: 'Counterproductive'	1

A.I.M. is an ObviuZ product

- A.I.M. website
- ObviuZ website
- ObviuZ Support portal

(end of automated message)

Whenever an Inventory is performed, any changes in index categorization are reported.
Additionally, any modifications made to parameter values or changes in the collection (Targets, Members, Tables, Indexes) are also reported.
In a frequently changing environment, it is expected that a report will be generated whenever a full Inventory is run.

Upon adding one or more Targets, the next report generated will be comprehensive due to the initial addition of items to the collection. Subsequent reports will only contain information about changed categories and minor collection modifications.

The events that are (potentially) significant and may warrant action are **highlighted** as shown above, these are changes in the categories *Missing*, *Beneficial*, *Obsolete*, and *Counterproductive*. Before these indexes are created or dropped, there is an opportunity to intervene.
The other events are merely reported for completeness.

7.2 Summary report

Details are summarized and reported every Monday at 7:00 (default schedule).

Of course you can change the time and frequency of this schedule, see also ***8.4.6 Reports***.

8 In-depth information

8.1 Category

Different Index categories:

Index Category	Description	Existing	Explanation
Beneficial	An index that is likely to improve query performance.	No	Previously Missing but has been evaluated and found to be Beneficial, should be created.
Counterproductive	An index that is negatively impacting query performance.	Yes	Previously Obsolete and has been evaluated and found to be Counterproductive, should be dropped.
Dropped	An index that no longer exists in the database.	No	Previously found in the database but no longer present.
Essential	Indexes that are essential for the database's operation and should not be changed.	Yes	Clustered indexes, Primary keys, and Unique keys/constraints.
Irrelevant	Indexes that are suggested by **SQL Server** to be created but can be ignored.	No	Missing indexes that do not meet the criteria to be considered Beneficial or don't have Precedence.
Missing	Indexes that are suggested by **SQL Server** to be created but have not yet been evaluated by A.I.M.	No	Indexes that are not currently in the database but could potentially be Beneficial.
Obsolete	Indexes that are no longer used.	Yes	Previously created indexes that are no longer used and are not essential for the database's operation. These indexes can negatively impact query performance so are likely to be counterproductive.
Unclassified	Indexes that have not yet been evaluated by A.I.M.	Yes	Indexes that are currently in the database but have not been categorized yet.
Valuable	Existing indexes that are found to have sufficient added value.	Yes	Valuable indexes that should be kept.

8.1.1 Assignment of categories

Immediate assignments (upon discovery):

- **D**ropped; can only change into **U**nclassified if the same index reappears.
- **E**ssential; can only change into **U**nclassified or **I**rrelevant if the index is redefined with certain specific parameters, or into **D**ropped if the index is dropped *outside* A.I.M. (A.I.M. will never drop an **E**ssential index).
- **M**issing; can evolve into **B**eneficial or **I**rrelevant.
- **O**bsolete; can evolve into **C**ounterproductive or **V**aluable.
- **U**nclassified; can evolve into **O**bsolete or **V**aluable.

Delayed assignments (after an evaluation period):

- **B**eneficial is assigned if a **M**issing index appears to be Beneficial.
- **C**ounterproductive is assigned if an **O**bsolete index appears to be Counterproductive
- **I**rrelevant is assigned if a **M**issing index is not found Missing anymore and also not in the collection as something else.
- **V**aluable is assigned if an **U**nclassified index appears to be Valuable

More possible transitions between categories:

- **B**eneficial can devolve back into **M**issing if it is no longer considered Beneficial.
- **C**ounterproductive can devolve back into **O**bsolete if it is no longer considered Counterproductive.
- **V**aluable can devolve into **O**bsolete or vice versa.
- Any existing index can transition into **U**nclassified if the specification changes.

Additional notes:

- Missing means that **SQL Server** suggested that index would be beneficial, A.I.M. will decide if that really is the case.
- Obsolete means that A.I.M. suggests that index could be Counterproductive.
- Missing indexes can appear, disappear and reappear. It all depends on "what's happening in there" in terms of queries. To prevent excessive dynamics in the collection, a disappeared Missing index is categorized as Irrelevant but kept *in* the collection. Only after a specified period where it does not reappear, it's registration will be deleted.
- Missing indexes that do not disappear, but do not meet certain criteria to make them interesting, are also categorized as Irrelevant.
- With Irrelevant indexes it's not obvious if they are currently being seen as Missing by **SQL Server** or if they have been suggested as such but are not at the moment.
  Important thing here is that they are Irrelevant so you should not worry about them. By handling them as described above, they don't keep reappearing in Reports distracting you from more important cases.
  However, if an Irrelevant index is suggested Missing again by **SQL Server**, its DtUpdated timestamp will be updated to reflect the latest recommendation from **SQL Server** so you actually *can* conclude if it is just hanging in there or has been recently suggested as Missing.

8.1.2 Category colors

There's a fixed color schema for Categories which might be user-definable in future versions:

Index Category	Color
Beneficial	Gold
Counterproductive	Salmon
Dropped	Violet
Essential	LightGray
Irrelevant	LightGoldenrodYellow
Missing	Khaki
Obsolete	LightSalmon
Unclassified	None
Valuable	Lime

8.2 Protection

There are 6 types of protection levels for indexes in A.I.M.:

There are several types / levels of protection, sometimes more than one applies in which case the most important one will prevail:

1. None; This is the default protection level. Indexes with this protection level can be created or dropped at any time, both manually and automatically.
2. OnHold; Indexes with this protection level can be created or dropped at any time manually, but it is strongly advised to wait for A.I.M. to take action, as it will automatically create or drop indexes based on its analysis and after specified time thresholds.
3. MsShipped; Indexes with this protection level could be created at any time, both manually and automatically, but to drop indexes that were not created by A.I.M. will require a specific setting see ***8.3.17 ProtectMsShipped***.
4. Persistent (user imposed); indexes with this protection level will not be dropped by A.I.M.
5. Rejected (user imposed); indexes with this protection level will not be created by A.I.M.
6. Locked (system imposed); indexes with this protection level will not be dropped by A.I.M.

To summarize the behavior:

- Every Drop or Create from the console will ask for conformation
- **Persistent and Locked** offer the only real protection, in fact they act the same, but is Persistent is User imposed while Locked is System imposed. All other protection types (other than **None**) just add a layer of extra confirmation.

Next, each of those will be described in detail.

8.2.1 None

Obviously, you can drop existing indexes, or create non-existing ones.

8.2.2 OnHold

There can be actions that need to be performed, but we're just not doing that *yet* for some reason; that could be a threshold or some other index that's more important. You can however overrule this and perform the action through A.I.M. Console.

8.2.3 MsShipped

This indicates if a table has the *is_ms_shipped* attribute set to True, meaning that it was created by an internal **SQL Server** component or simply is a standard (system) table provided by **Microsoft**, by default A.I.M. will protect the standard indexes on that table from being dropped but indexes created by A.I.M. are not subject to this protection.

This protection feature can be switched off by setting Parameter ProtectMsShipped to 0.

8.2.4 Persistent

The user can label any Index that is registered in A.I.M. as Persistent, which means that A.I.M. will never attempt to drop it, no matter what.

To support this for the categories Beneficial, Dropped and Missing doesn't appear to make sense because they do not exist… at the moment! But any of those could be (re)created through A.I.M. so it actually does make sense, therefore A.I.M. allows you mark them Persistent to prevent it to be dropped *after* they are (re)created, and it gets even better: when you mark a Dropped index Persistent, it will be recreated during the next round!

Essential indexes are inherently persistent due to their *Locked* protection, and it's unlikely but not impossible for them to change category in the future. Therefore, these indexes *can* still be explicitly marked as persistent if desired.

Finally, indexes on MsShipped tables can be made Persistent. When configuration is already protecting MsShipped tables (through the Parameter *ProtectMsShipped*), then it has no immediate effect but when this configuration changes, it actually does! So, keep this in mind.

8.2.5 Locked

Several Categories have **Locked** protection, which means that A.I.M. will not create or drop those indexes, nor will it allow you to do so through the Console for obvious reasons.

- Essential; Primary Keys, Unique Keys etc. will be retained
- Irrelevant; will not be created
- Valuable; will be retained

Of course, A.I.M. cannot prevent you from dropping it manually (outside the Console) but A.I.M. does not support those actions.

8.2.6 How Protection works in the Console

Let's go back to the *Index Details* form and look at the *Protection* section:

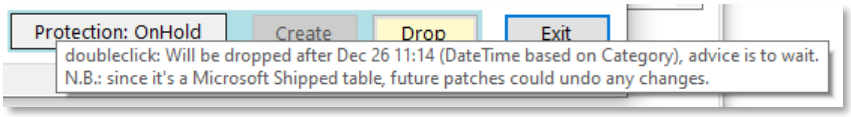
In some cases, the [Create] *or* [Drop] button is enabled, based on Category and Protection:

Category	Button
Beneficial	Create
Counterproductive	Drop
Dropped	Create
Missing	Create
Obsolete	Drop

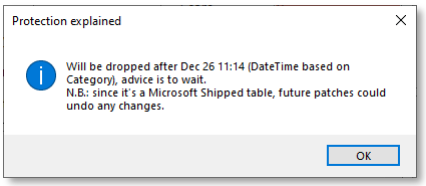
For other Categories these buttons remain disabled.

Protection determines if the button in case is really enabled; protections *None* and *OnHold* allow the use of those, but others do not.

If the protection panel behind the Create and Drop buttons displays any background, it indicates that even if one of the buttons is enabled, clicking it and executing the corresponding action is not in line with A.I.M.'s principles. Instead, it is recommended to wait for A.I.M. to take the necessary action. Hovering the *Protection* field reveals a detailed explanation of the situation in effect:



Double-clicking it opens a dialog box with the explanation:



In the event that an enabled button is clicked a pop-up will appear to ask confirmation before proceeding, if that happens while protection is active, first an extra pop-up will appear to re-emphasize the current protection status and advise against proceeding with the action. However, you still have the option to override the system behavior and proceed with the corresponding action at your own discretion.

8.2.7 Protection colors

There's a fixed color schema for Protection which might be user-definable in future versions:

Description	Color
Locked	LightGray
MsShipped	Orange
None	
OnHold	PowderBlue
Persistent	DeepSkyBlue
Rejected	Red

8.3 Parameters

Certain Parameters collaborate synergistically, exerting substantial influence over the timing and occurrence of events. Some are used to calculate OnHold, see ***8.2 Protection*** item ***2***.

Before modifying any Parameter values, exercise caution and consider the potential consequences.

Overview of Parameters and their values after initial configuration:

Parameter	Description	Value	Check
AutoCreate	Create without user intervention: 0 = No 1 = Yes	1	Should be 0 or 1
AutoCreateDelay	Hours delay before executing an automated Create.	12	Should be an integer >= 0
AutoDrop	Drop without user intervention: 0 = No 1 = Yes	1	Should be 0 or 1
AutoDropDelay	Hours delay before executing an automated Drop.	24	Should be an integer >= 0
CatMDelay	Hours delay before promoting Missing to Benificial.	12	Should be an integer >= 0
CatODelay	Hours delay before demoting Obsolete to Counterproductive.	12	Should be an integer >= 0
CatUDelay	Hours delay before recategorizing Unclassified.	24	Should be an integer >= 0
ColExcludeLength	Exclude columns with a length greater than or equal to this value from the index creation DDL, even though **Microsoft** did suggest them.	500	Should be an integer >= 0
KeepIdxDroppedDays	Days to keep Dropped Indexes in the collection.	14	Should be an integer >= 0
KeepIdxHistoryDays	Days to keep Index history (in his.Idx).	14	Should be an integer >= 0
KeepIdxIrrelevantDays	Days to keep Irrelevant Indexes in the collection.	5	Should be an integer >= 0
KeepLogHistoryDays	Days to keep LogHistory.	14	Should be an integer >= 0
LogDetail	Logging Detail 0 = minimal.	1	Should be an integer >= 0

Parameter	Description	Value	Check
Member_Default	New database is member by default: 0 = No 1 = Yes	1	Should be 0 or 1
MissingTreshold	Score treshold for Missing indexes.	100	Should be an integer >= 0
ObsoleteTreshold	Score treshold for Obsolete indexes.	0	Should be an integer
ProtectMsShipped	Protect MsShipped tables from A.I.M. actions: 0 = No 1 = Original 2 = All Note: option 1 will only create and drop new indexes and option 2 will not create or drop any index.	1	Should be between 0 and 2
RecipientReportDetails	Recipient for details from A.I.M.	hans@obviuz.com	Should be a valid email format
RecipientReportSummary	Recipient for summaries from A.I.M.	hans@obviuz.com	Should be a valid email format
TablePagesTreshold	Treshold on pages for Missing indexes.	1000	Should be an integer >= 0
TableRowsTreshold	Treshold on rows for Missing indexes.	100000	Should be an integer >= 0
UpHoursTreshold	Treshold on server's uptime.	48	Should be an integer >= 0

Each of those will be explained in detail in the next paragraphs.

8.3.1 AutoCreate

Determines whether A.I.M. should create an index when appropriate, if you set this to 0, you need to create them manually.

Works in conjunction with *AutoCreateDelay* and *ProtectMsShipped*.

8.3.2 AutoCreateDelay

Determines the OnHold datetime for Creates, *after* they have been categorized as Beneficial.

Only has effect if *AutoCreate* = 1.

8.3.3 AutoDrop

Determines if A.I.M. should drop an index when appropriate, if you set this to 0, you need to drop them manually.

Works in conjunction with *AutoDropDelay* and *ProtectMsShipped*.

8.3.4 AutoDropDelay

Determines the OnHold datetime for Drops, *after* they have been categorized as Counterproductive.

Only has effect if *AutoDrop* = 1.

8.3.5 CatMDelay

Determines the OnHold datetime for recategorization as Beneficial, after they have been categorized as Missing.

8.3.6 CatODelay

Determines the OnHold datetime for recategorization as Counterproductive, after they have been categorized as Obsolete.

8.3.7 CatUDelay

Determines the OnHold datetime for recategorization after they have been found and categorized as Unclassified.

8.3.8 ColExcludeLength

In general, it is very questionable if indexes on large columns are useful. That's why this setting was introduced; you can force A.I.M. to exclude columns with a length greater than or equal to this value from the index creation DDL, even though **Microsoft** did suggest them.

8.3.9 KeepIdxDroppedDays

Dropped indexes are retained in the collection for a specific period so that you can recreate them if you see fit. This parameter determines the retention period.

8.3.10 KeepIdxHistoryDays

Table *his.Idx* contains a copy of each row from dbo.Idx, plus previous versions of it. It allows you to inspect the chain of events for any Index.

This parameter determines the retention period for this history.

8.3.11 KeepIdxIrrelevantDays

Irrelevant indexes will be kept low profile: they don't appear in the index list (by default), nor will they be reported. However, they could become relevant again due to circumstances. That's why they are retained for some period so that they are not new each time and the existing registration is reused.

This parameter determines the retention period for these registrations.

8.3.12 KeepLogHistoryDays

There's a lot of logging going on, so we need to cleanup regularly. This parameter determines the retention days for Log records.

8.3.13 LogDetail
In some cases, it can be desired to (temporarily) raise the level of logging, that's where we can use this parameter for.

8.3.14 Member_Default
Determines whether newly discovered databases should be automatically categorized as Members. If this setting is 1 (recommended), new databases will be automatically marked as Members. If this setting is 0, you will need to manually mark databases as Members.

8.3.15 MissingTreshold
This value is used to determine if an index that **SQL Server** proposes as Missing, will be treated as Missing by A.I.M. also. It's evident that the lower this value is set, the more indexes will be created.

8.3.16 ObsoleteTreshold
This value is used to determine if an index is to be treated as Obsolete. If the score gets below this value, it qualifies for Obsolete.

8.3.17 ProtectMsShipped
Indicates MsShipped tables should be protected from creating and dropping indexes, values:

0. No, treat MsShipped tables like user tables.
1. Yes, keep the original indexes but create or drop non-original indexes.
2. Yes, do not create or drop any indexes.

8.3.18 RecipientReportDetails
Provide the email address of the user who should receive the Details report.

8.3.19 RecipientReportSummary
Provide the email address of the user who should receive the Summary report.

8.3.20 TablePagesTreshold
Tables with limited pages are considered not to benefit from indexes.
Missing indexes on tables with a page count below this value, will not become Beneficial.

This parameter is used in conjunction with *TableRowsTreshold*.

8.3.21 TableRowsTreshold
Tables with limited rows are considered not to benefit from indexes.
Missing indexes on tables with a row count below this value, will not become Beneficial.

This parameter is used in conjunction with *TablePagesTreshold*.

8.3.22 UpHoursTreshold
After a restart, Missing index information is blank. **SQL Server** needs some runtime to build up useful data. This parameter indicates after how many up time we start using the data.

8.4 Default schedules

8.4.1 Introduction

While we have provided default schedules for the jobs as a courtesy to save you time, it is important to note that these are based on general assumptions, and these assumptions may not align with your specific business needs.

To ensure optimal performance and alignment with your specific workload requirements, we strongly recommend reviewing and fine-tuning the default schedules immediately after configuration. Additionally, we advise reevaluating and adjusting the schedules after utilizing the product for a week or two to refine the settings based on your actual usage patterns.

8.4.2 Dropping Counterproductive indexes

This task can be performed at any time without performance penalties, so we have scheduled it to run *"every day every 2 hour(s) between 01:00:00 and 23:59:59"*.

8.4.3 Creating Beneficial indexes

We assume that these indexes can be created overnight, so the default schedule is *"every day every 1 hour(s) between 20:20:00 and 05:59:59"*.

8.4.4 Inventory

The frequency of inventory runs depends on various factors such as the nature of the business, data usage, and dataset types. The default schedule is *"every day every 4 hour(s) between 03:05:00 and 23:59:59"*.

8.4.5 Maintenance

Default schedule for maintenance is *"every day at 22:30:00"*.

8.4.6 Reports

The Details Report is generated as part of the Inventory process and does not have a separate schedule.

The Summary Report is generated *"every week on Monday at 07:00:00"*.

8.4.7 E-mail notification

Set up e-mail notification!

It is generally recommended to set up email notification for failed jobs. When a job fails, it's important for the database administrator to be informed. Therefore, we advise setting up email notification for all jobs, including A.I.M. jobs.

See this article for help.

9 Deinstalling A.I.M.

In case you want to stop using A.I.M., here's the procedure to deinstall.

1. Drop the A.I.M. jobs
2. Delete Job Category 'A.I.M. – Automated Index Management'
3. Drop the A.I.M. database
4. Remove the installed program files (see 4.4 Installation item ***3***)

10 Various procedures

10.1 Reset Parameter values

To reset all Parameters (except mail addresses) to the defaults, execute the following procedure on the A.I.M. host:

```
use [AIM];
exec [hlp].[up_Config_Parameter] 1, 1;
```

10.2 Change A.I.M. Credentials for an existing Target

1. Change the registered Credentials using the *Target Details form*, see ***6.3.1 Target Details***.
2. Change the account and/or password for the Login on the Target instance accordingly.

10.3 Change Parameters that influence OnHold

There are several Parameters that influence OnHold:

- CatMDelay
- CatODelay
- CatUDelay
- AutoCreateDelay
- AutoDropDelay
- UpHoursTreshold

When (one of those) are modified, that could affect the OnHold datetime and status, and Protection. These values are calculated dynamically so a simple refresh of the display will suffice to see that effect.

However, it could also affect the (timing of) determination of Category, for that effect to show, you need to re-run *Evaluation* from the Menu, see ***5.2.1.2 Action***.

10.4 Change Parameters that influence Category

There are several Parameters that influence the determination of Category:

- MissingTreshold
- ObsoleteTreshold
- ProtectMsShipped
- TablePagesTreshold
- TableRowsTreshold

To see the effect of changing those, you need to re-run *Inventory* from the Menu, see ***5.2.1.2 Action***.