

TARGIT

InMemory Database

Installation and Configuration Guide

Version 1.0.162

Contents

<i>TARGIT</i>	1
<i>InMemory Database</i>	1
Introduction	3
Architecture	3
Database:	3
Upgrading from an earlier version of the TARGIT InMemory Database	4
Installation	4
Installing the TARGIT InMemory Database Engine	4
Installing the TARGIT InMemory ETL Studio	5
Installing the TARGIT InMemory Scheduler Service	5
Installing the TARGIT InMemory Data Drivers	5
Installing the TARGIT InMemory SSIS Tasks	5
Configuring the InMemory Database for use with TARGIT Decision Suite	6
Getting Started with TARGIT InMemory Database from the command line.	6
Troubleshooting Tips	8
Checking whether the port is used by another application	8

Introduction

TARGIT InMemory Database is offered for use with TARGIT Decision Suite and is in nature similar to other column-store database running in-memory like the SSAS tabular model.

Column-store databases allow for large data set storage in a small space when loaded in-memory. Rather than repeating all the data values for each row like a traditional relational database would do, a column-store database only keeps a distinct list of values for each field and replaces the values of the transactions with highly indexed integer values so it can quickly look up the values and present the data set.

The TARGIT InMemory Database can be used almost any data source and does not require 3rd party components or licensing to e.g. SQL Server. Data can even be loaded directly from the source systems into the TARGIT InMemory Database using the TARGIT InMemory ETL Studio.

This guide is designed for helping to get the product installed, for a detailed user guide please visit <https://doc.targit.com/en/inmemory>

Architecture

The TARGIT InMemory Database will require the port tcp/5060 to be open to allow the TARGIT Server to connect to the database.



Requirements

The following requirements must be present for successful installation of the TARGIT InMemory Database:

- TARGIT Decision Suite 2015 SR1 or newer
- License for the TARGIT InMemory Database module.
- Windows Server 2008 x64 or higher
- .NET Framework 4.6 or newer
- Enough memory to contain the data in database (16GB recommended for a base install)
- It is recommended that the in-memory database files are stored on an SSD drive if frequent updates are performed to minimize the load time of repopulated data stores.

Upgrading from an earlier version of the TARGIT InMemory Database

1. Ensure you have a backup of your TARGIT InMemory Database executables folder and data folders. The default paths are:

Executables: C:\Program Files\TARGIT\TARGIT InMemoryDB
Data: C:\ProgramData\TARGIT\TARGIT InMemoryDB

Always keep a copy of the installer in a known place, thereby ensuring that you can roll back if necessary.

2. Run the TARGIT InMemory Database Installer.

Installation

The TARGIT InMemory Database is installed from a single installer that can be obtained from the Download Center.

Additionally, the TARGIT InMemory Database comes with range of tools.

The full download should therefore be (Actual version numbers are specified as X):

TARGITInMemoryDB-X.X.X.msi	Installer for the TARGIT InMemory Database Engine.
TARGIT InMemory ETL Studio X.X.X.exe	Installer for the ETL Studio that can be used to design individual .imp files or to deploy an ETL project to the TARGIT InMemory Scheduler
TARGIT InMemory Scheduler Service X.X.X.exe	Installer for the scheduler service that allows you to perform scheduled automatic updates.
TARGIT InMemory Data Drivers X.X.X.exe	Contains drivers that can be used with either the Scheduler Service or Standalone drivers for TARGIT InMemory Database.
TARGITInMemorySSISTasks 1.00.msi	Installer for customers wanting to integrate InMemory Database updates into an existing SQL Server Environment.

Installing the TARGIT InMemory Database Engine

Run the installer and choose the location for the data files and the database log file, also you will be prompted to enter the default password for the InMemory Database Server. Make sure to write the

password down in a safe place. It is recommended that the data folder is placed on a fast disk array e.g. an SSD drive.

The installer will also create an icon in the Windows Start Menu with a link to the “TARGIT InMemory Query Tool” (tiQuery).

Note: The log file is a text file capturing the commands execute by the server and does therefore not need to be on its own drive.

Installing the TARGIT InMemory ETL Studio

When executing the installer, you can basically just accept all settings. It will place the files in \Program Files and create an icon for “TARGIT InMemory ETL Studio” in the Windows Start Menu.

Installing the TARGIT InMemory Scheduler Service

Installing this component requires you to specify where it is to be installed. Please note that it is highly recommended to install the service to the same drive where you have located the database files for the TARGIT InMemory Database since the scheduling service will write the temporary in-memory store to this folder before moving it to the database server and reloading it.

The installer will create an icon for the TARGIT InMemory Scheduler Management application which will be placed in the Windows Start Menu.

Installing the TARGIT InMemory Data Drivers

The data drivers are a separate install and gives you access to a range of data sources.

The drivers will be installed in the folder where the InMemory Scheduler Service is located and will register themselves for use with the scheduler service and with the “server option” that is default when creating new data sources in TARGIT InMemory ETL Studio.

If you want to change the registration of the assemblies (drivers) this is done by modifying them in TARGIT InMemory Scheduler Management.

Installing the TARGIT InMemory SSIS Tasks

This is not required to run TARGIT InMemory Database, however we deliver this as an add-on for customers running SQL Server 2012 and SQL Server 2014 and it can replace the need for creating a .bat (Batch file) to run updates.

The installer for this is available in the TARGIT Download Center under TARGIT InMemory Database.

To install simply run the installer and make the appropriate selections and the TARGIT InMemory Database import task will be available in SSDT/BIDS.

In case the task isn't available for your version of SQL Server please submit a task in the TARGIT Support System and we will review whether we will provide support for your version.

Configuring the InMemory Database for use with TARGIT Decision Suite

This guide will help you connect to the default “Northwind” database that accompanies the InMemory Database.

1. Launch the TARGIT Management for the TARGIT Decision Suite.
2. Click license and verify that the option “TARGIT InMemory Database” is present. If this entry is missing, you will not be able to add the connection.
3. Go to connections, right-click and choose “Add”, pick the category “Relational Databases” and choose “TARGIT InMemory Database” (if it’s greyed out you don’t have the correct license)
4. Give the database a name e.g. “Sample Northwind”
5. Enter the following settings
 - a. Specify the name of server hosting your TARGIT InMemory Database (ensure the firewall allows connection on this machine)
 - b. Enter “default” as the username
 - c. Enter the password you chose and wrote down during the installation of the TARGIT InMemory Database engine.
 - d. Choose the Northwind database in the database dropdown
 - e. Click the “Next” button
 - f. Toggle the Data consistency option to allow inconsistent data (since this is an unchecked Northwind database then this is the best option)
6. Save the connection
7. Right-click the new connection and choose to edit.
8. Find a single table and drag it onto the workspace
9. Click the cube tab at the cube and create a new cube from the toolbar and choose the table you selected as fact table
10. Drag relevant text columns from the Northwind table to the Dimensions folder on the cube pane to the left from the table in the workspace also find a numeric column and drag it in as a measure.
11. When done, click the save button and launch TARGIT.
12. Build you analysis and you’re now working with your first TARGIT InMemory Database – data model.

Getting Started with TARGIT InMemory Database from the command line.

The TARGIT InMemory Database can load data from a wide range of database. In the example below we will focus on loading information from a Microsoft SQL Server Database that in this example is called TrainingDatabase and resides on the default SQL server on the local machine using the import scripts.

1. Ensure that the tiServer (TARGIT InMemory Database Server) is running. This you can always see in the Windows Services control panel.
2. Create a folder to store the script. Where the script will be located is where the in-memory datastore will be created when executing the import script (.imp). In this example c:\InMemory is used.
3. Use Notepad++ (you can with benefit turn on the language of the file to SQL) or just plain notepad with the following lines:

```
DATASOURCE sqlldb=SQLSERVER
'Server=localhost;Database=TrainingDatabase;Trusted_Connection=True;'

IMPORT InvoiceLines=sqlldb.{SELECT * FROM ERPInvoiceLines}
IMPORT Customers=sqlldb.{SELECT * FROM ERPCustomers}

SAVE
```

Save this file as ERPData.imp

This will create two new tables in the in-memory data store called InvoiceLines and Customers. The tables will have the content of the SELECT-statement executed on the Microsoft SQL Database that is defined in the top DATASOURCE definition.

After loading the data, the SAVE command is issue which writes the in-memory datastore file and will have the name ERPData.targitdb

4. Execute the command in the Windows command prompt:

```
"C:\Program Files\TARGIT\TARGIT InMemoryDB\tiImport.exe"
"c:\inmemory\ERPDData.imp"
```

This will execute the ERPData.imp script and create the in-memory data store ERPData.targitdb

5. Copy the in-memory data store file to the servers Data folder. Below the default path is used:

```
copy c:\inmemory\erpdata.targitdb "c:\programdata\targit\targit
inmemorydb\data\"
```

6. Execute the command tiLoad thereby requesting the server to reload the ERPData.targitdb file and use that one for queries on the server.

```
"C:\Program Files\TARGIT\TARGIT InMemoryDB\tiLoad.exe"
database=ERPDData;pwd=PASSWORD_FROM_INSTALL /reload
```

Note: Item 4-6 can with benefit be combined into a single

7. Launch TARGIT Management, go to Connections, right-click the white space and choose Add. Choose the type Relational connection and find the TARGIT InMemory Database.

Name the connection e.g. InMemory ERPData and setup it up with the following parameters:

Server: localhost

Port: 5060 (this is the default port)

Username: default (This is important!)

Password: PASSWORD_FROM_INSTALL (this is the password you entered at install time. Choose the database: ERPData in the dropdown database dropdown.

Test the connection. If it doesn't connect verify the server is running and check if port 5060 is occupied by another application (see the troubleshooting appendix in this guide)

8. Right-click the functioning connection and choose "Edit datamodel". Drag the two tables InvoiceLines and Customers to the work space and make sure to defined the join correctly by dragging FROM the fact table TO the dimension table. In this example a relation is defined from CustomerNo in InvoiceLines to CustomerNo in the customers table.
9. Click the create cube icon in the toolbar and choose InvoiceLines as the fact table.

Next drag CustomerName from the Customers table to the dimensions folder in the tree, ad do the same with ItemNo from the InvoiceLines-table, do the same process for the columns InvoiceDate, InvoiceNo and ProductDescription. Next define measures by dragging LineAmount, LineDiscount, LineCost to the measures folder in the cube and click save.

10. Launch the TARGIT client and create a crosstab with the data – and you have a BI solution with blistering performance – even if you combine every single dimension attribute in the cube. This applied even if you 200 million rows in your fact table.

For more information, examples and much more visit the documentation portal at <https://doc.targit.com/en/inmemory>

Troubleshooting Tips

Checking whether the port is used by another application

In case port tcp/5060 is used by another application the command "netstat -anob" can be executed in the command prompt and be used to find the process number of the application listening to the port.

