# How to Install and Use the ASTrace Utility for SQL Server Analysis Services 2012



**Author:**Karan Gulati{MCSE, MCDBA, MCTS, OCAP, OCP}|DS Escalation Services

[Facebook BI Page](http://www.facebook.com/#!/pages/Data-Warehouse-Business-Intelligence/189141584512231)|[Twitter](https://twitter.com/karangspeaks)|[Linked In](http://in.linkedin.com/in/karanspeaks)|[Personal Site](http://karanspeaks.com/)|[Blog Space](http://blogs.msdn.com/karang)

**V2 Updates:** Greg Galloway (Artis Consulting)

Contents

[How to Install and Use the ASTrace Utility for SQL Server Analysis Services 2012 1](#_Toc360054761)

[Introduction 2](#_Toc360054762)

[Overview of the ASTrace Service 2](#_Toc360054763)

[Prerequisites 2](#_Toc360054764)

[Out of Scope 3](#_Toc360054765)

[Setup 3](#_Toc360054766)

[Removing ASTrace 6](#_Toc360054767)

[Removing the Service 6](#_Toc360054768)

[Usage 6](#_Toc360054769)

[Exceptions for the ASTrace Utility 6](#_Toc360054770)

[List of Events in Standard.tdf 6](#_Toc360054771)

[List of Events in “Capture AS Queries.tdf” 8](#_Toc360054772)

[Customizing the Events Collection Template 8](#_Toc360054773)

[More Information about SSAS Events 9](#_Toc360054774)

## Introduction

SQL Server ships with a tool called SQL Server Profiler that is able to capture trace events from a SQL Server Analysis Services and write those events to a SQL Server table. However, the Profiler GUI consumes unnecessary memory and processor power capturing the trace events and displaying them on the screen. For constant monitoring and logging, the ASTrace tool will capture a Profiler trace and write it to a SQL Server table without requiring a GUI. ASTrace also runs as a Windows service allowing it to restart automatically when the server reboots.

Alternatives to ASTrace include [server side traces](http://blogs.msdn.com/b/karang/archive/2012/09/10/9916124.aspx) (and then loading those .trc files into SQL Server with a [PowerShell script](http://www.bp-msbi.com/2012/02/counting-number-of-queries-executed-in-ssas/) after the fact) and Xevents.

ASTrace is a great way to capture the text and duration of MDX queries, to log details about processing failures, and to troubleshoot SSAS issues such as locking.

ASTrace 2.0 adds the ability for one ASTrace service to capture events from multiple Analysis Services instances at once.

## Overview of the ASTrace Service

To set up monitoring of Analysis Services, use the ASTrace tool to enable tracing of SQL Analysis Services events. The service can run on any computer, as long as that computer has SQL Server Profiler installed.

The ASTrace tool collects the information based on a preconfigured Analysis Services Profiler template, and stores the results in SQL Server tables in a logging database. You can change the logging database used by the utility, and you can customize the templates for event collection based on your own requirements for either monitoring or auditing or both.

The ASTrace utility was originally created for SQL Server 2005 by former Analysis Services Program Manager Edward Melomed. This documentation applies to an update of the utility created by Microsoft support engineer Karan Gulati and community members Greg Galloway (Artis Consulting) and Andrew Calvett (UBS). An ASTrace2008 version is provided for SQL Server Analysis Services 2008 and 2008R2. An ASTrace2012 version is provided for SQL Server Analysis Services 2012. It is necessary that each service use the version that matches the version of Analysis Services that you intend to monitor.

Please see [CodePlex](http://sqlsrvanalysissrvcs.codeplex.com/) for related samples, including updates to the ASTrace utility released by members of the community, whose contributions and support we gratefully acknowledge.

### Prerequisites

* Microsoft SQL Server 2012, including the following components:
* Database Engine
* Analysis Services
* SQL Server Management Studio
* Microsoft.NET Framework 4.0

### Out of Scope

This release supports running the utility only as a service; there is no standalone application. The reason is that for monitoring, you need to run it on a more or less continuous basis. If you want to create a console application or other utility for monitoring SSAS you are welcome customize the ASTrace source code to suit your purposes. The latest source code is available from [Codeplex](https://sqlsrvanalysissrvcs.codeplex.com/SourceControl/latest).

## Setup

The following steps describe how to install the ASTrace utility as a service.

#### Copy executable files

* + Copy the following the entire ASTrace2012 folder an appropriate folder such as the root of the E: drive.
  + Unblock the ASTrace.exe file by right clicking on the file and clicking Unblock:  
    

#### Register the service

* + Right click on the install.bat file in the ASTrace2012 directory and choose **Run As Administrator**. Wait for the script to complete and ensure you see a success message.
  + Windows Key-R… services.msc to open the Services management applet.
  + Scroll down to the ASTrace service and double click it.
  + If desired, set the service to start automatically, but do not start it.

|  |  |
| --- | --- |
| C:\Users\jeannt\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\Z5HEO183\MC900433883[1].png | **IMPORTANT!**  Do not start the service yet. |

#### Configure security

* On the **Logon** tab of the ASTrace service in the Services management applet, set the account that runs the ASTrace service to an appropriate local or domain user account. This will minimize the surface attack area. The account that runs the ASTrace service must have the following rights:
* Log on as Service to the computer.
* Membership in the role **SQL db\_owner** on for the **TraceDB** database.
* Administrator rights on the instances of SQL Server Analysis Services that the service will be monitoring.

#### Configure the ASTrace service

1. Open the ASTrace.exe.config file in Notepad.
2. Type values for the following fields as shown in the following example below. Note that these values should be customized to suit your environment, and are just examples.
   * **AnalysisServerNames** – a comma separated list of one or more Analysis Services instances. For example, “localhost,localhost\tabular” will monitor the default instance and the instance named TABULAR on the local server.
   * **SQLServer** – the SQL Server instance to store the trace events. The instance can be “servername” if it is the default instance or “servername\instancename” for a named instance.
   * **SQLServerDatabase** – The SQL Server database name in which to log the trace events.
   * **TraceTableName** - The name of the table in which to store the trace events. Just specify the name of the table. The dbo schema is assumed.
     1. If PreserveHistory=True, then an equivalently named table suffixed with “History” will be created each time the trace must be restarted.
     2. If multiple instances are specified in AnalysisServerNames, then the instance name will be appended to the TraceTableName automatically. For example, the table table to capture the events from the localhost\tabular instance will be written to [dbo].[ASTraceTable\_localhost\tabular].
   * **TraceDefinition** – The filename of a trace template. Create your own or use one of the two trace templates provided. Standard.tdf captures many typical events. “Capture AS Queries.tdf” is a lightweight trace to just capture MDX queries and durations.
   * **PreserveHistory** – If set to True, each time the ASTrace service starts (or each time the SQL or SSAS services restart), the current rows in TraceTableName will be moved into a History table to preserve them. If set to False, each time the ASTrace service starts, each time the SSAS instance or the SQL instance restarts, the trace table will be overwritten.
   * **RestartRetries** – If the SQL or SSAS instance goes down for maintenance or restarts, this integer setting determines how many times ASTrace will attempt to reconnect the trace before stopping the ASTrace service. If monitoring multiple SSAS instances, when even one instance exceeds the allowed number of retries, the whole ASTrace instance will stop.
   * **RestartDelayMinutes** – If the SQL or SSAS instance goes down for maintenance or restarts, this integer setting will determine how many minutes ASTrace waits between each attempt to retry starting the trace.

<applicationSettings>

<ASTrace.Properties.Settings>

<setting name="AnalysisServerNames" serializeAs="String">

<value>localhost,localhost\tabular</value>

</setting>

<setting name="SQLServer" serializeAs="String">

<value>localhost\SQLInstance</value>

</setting>

<setting name="SQLServerDatabase" serializeAs="String">

<value>TraceDB</value>

</setting>

<setting name="TraceTableName" serializeAs="String">

<value>ASTraceTable</value>

</setting>

<setting name="TraceDefinition" serializeAs="String">

<value>Standard.tdf</value>

</setting>

<setting name="PreserveHistory" serializeAs="String">

<value>True</value>

</setting>

<setting name="RestartRetries" serializeAs="String">

<value>3</value>

</setting>

<setting name="RestartDelayMinutes" serializeAs="String">

<value>1</value>

</setting>

</ASTrace.Properties.Settings>

</applicationSettings>

#### Configure the SQL Server instance used for logging

1. Connect to the instance of SQL Server where you will create the **TraceDB** table that serves as repository.
2. Once connected, execute the following command:

CREATE DATABASE TraceDB

GO

ALTER DATABASE TraceDB SET RECOVERY SIMPLE

1. Grant permissions to the ASTrace security account assigned in step 3 “Configure security”.

#### Start the service

1. In **Control Panel**, select **Administrative Tools**, and then click **Services**.
2. Locate the ASTrace utility and start the service.
3. Alternately, to start the service automatically with a DOS batch file, run the following:  
   net start ASTrace

#### Ensure the service is functioning properly

1. View ASTraceService.log in Notepad and ensure there are no errors reported.
2. View the SQL Server tables and ensure events are being captured properly.

## Removing ASTrace

### Removing the Service

1. Right click on uninstall.bat and choose **Run As Administrator**.

## Usage

This section briefly describes the type of information captured by the ASTrace utility, and provides links to additional resources where you can learn about the events provided by Analysis Services.

### Exceptions for the ASTrace Utility

All exceptions generated by the ASTrace utility are stored in the following log file:

ASTraceService.log

This log file is generated in the ASTrace root directory each time the ASTrace service starts up.

By default, the file is appended each time the service starts.

### List of Events in Standard.tdf

By default, the following events are collected for Analysis Services, using the **Standard.tdf** template in SQL Profiler.

|  |  |  |
| --- | --- | --- |
| **Event Class** | **EventName** | **Columns collected** |
| CommandEvents | Command Begin | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, , ,DatabaseName, , ,ClientProcessID, SPID, ,NTDomainName, RequestParameters, RequestProperties |
| Command End | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, CurrentTime, Duration, DatabaseName, ,Error, ClientProcessID, SPID, CPUTime, NTDomainName, , |
| Discover Events | Discover Begin | , , , , , , , , , , , , , ,NTDomainName, RequestParameters, RequestProperties |
| Discover End | , , , , , , , , , , , , , ,NTDomainName, ,RequestProperties |
| Discover Server State Events | Server State Discover Begin | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, CurrentTime, , , , ,ClientProcessID, SPID, ,NTDomainName, ,RequestProperties |
| Server State Discover End | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, CurrentTime, Duration, , , ,ClientProcessID, SPID, CPUTime, NTDomainName, , , |
| Errors and Warnings | Error | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, , ,DatabaseName, ,Error, ClientProcessID, SPID, ,NTDomainName, , , |
| File Load and Save | (none) |  |
| Locks | (none) |  |
| Notification Events | Notification | TextData, ConnectionID, NTUserName, ,IntegerData, StartTime, CurrentTime, Duration, DatabaseName, ObjectName, , ,SPID, ,NTDomainName, ,RequestProperties |
| User Defined | TextData, ConnectionID, NTUserName, ,IntegerData, ,CurrentTime, ,DatabaseName, , , ,SPID, ,NTDomainName, , , |
| Progress Reports | Progress Report Begin | TextData, ConnectionID, , , , , , , , , , , , , , , , |
| Progress Report End |  |
| Queries Events | Query Begin | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, , ,DatabaseName, , ,ClientProcessID, SPID, ,NTDomainName, RequestParameters, RequestProperties |
| Query End | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, ,Duration, DatabaseName, ,Error, ClientProcessID, SPID, CPUTime, NTDomainName, , , |
| Query Processing | Query Subcube | TextData, ConnectionID, , , ,StartTime, CurrentTime, Duration, DatabaseName, , , ,SPID, CPUTime, , , , |
| Security Audit | Audit Admin Operations Event | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, , ,DatabaseName, ,Error, ClientProcessID, SPID, ,NTDomainName, , , |
| Audit Login | ,ConnectionID, NTUserName, ApplicationName, , ,CurrentTime, Duration, , ,Error, ClientProcessID, , ,NTDomainName, , , |
| Audit Logout | ,ConnectionID, NTUserName, ApplicationName, , ,CurrentTime, , , , ,ClientProcessID, ,CPUTime, NTDomainName, , , |
| Audit Object Permission Event | TextData, ConnectionID, NTUserName, ApplicationName, , , , ,DatabaseName, ObjectName, Error, ClientProcessID, , ,NTDomainName, , , |
| Audit Server Starts and Stops | TextData, , , , , ,CurrentTime, , , ,Error, ,SPID, , , , , |
| Session Events | Existing Connection | ,ConnectionID, NTUserName, ApplicationName, ,StartTime, CurrentTime, , , , ,ClientProcessID, SPID, ,NTDomainName, , , |
| Existing Session | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, CurrentTime, Duration, DatabaseName, , ,ClientProcessID, SPID, CPUTime, NTDomainName, ,RequestProperties |
| Session Initialize | TextData, ConnectionID, NTUserName, ApplicationName, ,StartTime, CurrentTime, ,DatabaseName, , ,ClientProcessID, SPID, ,NTDomainName, ,RequestProperties |

### List of Events in “Capture AS Queries.tdf”

The following events are collected for Analysis Services, using the **Capture AS Queries.tdf** template in SQL Profiler.

|  |  |  |
| --- | --- | --- |
| **Event Class** | **EventName** | **Columns collected** |
| Errors and Warnings | Error | EventSubclass, TextData, ConnectionID, NTUserName, ApplicationName, StartTime, , ,DatabaseName, Error, ClientProcessID, SPID, ,NTDomainName, , , |
| Queries Events | Query Begin | EventSubclass, TextData, ConnectionID, NTUserName, ApplicationName, StartTime, CurrentTime, ,DatabaseName, ,ClientProcessID, SPID, ,NTDomainName, RequestParameters, RequestProperties, SessionID |
| Query End | EventSubclass, TextData, ConnectionID, NTUserName, ApplicationName, StartTime, CurrentTime, Duration, DatabaseName, Error, ClientProcessID, SPID, CPUTime, NTDomainName, , , SessionID |

### Customizing the Events Collection Template

Although this utility uses the default template for Analysis Services event collection, you can specify any template that you like. You can also create custom templates for collecting additional events (or fewer events). For example, you might set up different custom templates for auditing security events as opposed to events used for analyzing specific queries or general system performance.

For more information about how to generate a template file, see:

**Export an Existing Trace to Use as Template**

<http://msdn.microsoft.com/en-us/library/ms187482.aspx>

Note that if you change the name of the template, you must modify the configuration file for the utility, as described previously in the section on **Setup**.

### More Information about SSAS Events

A full explanation of how to interpret trace events is out of the scope of this documentation. However, there are several excellent resources that you can use to interpret the trace information:

**Performance Guide on TechNet Wiki**

The SQL Server Performance Guide is a community hub for all kinds of articles about tuning and performance, including Analysis Services.

<http://social.technet.microsoft.com/wiki/contents/articles/5957.sql-server-performance-survival-guide.aspx>

**SQL CAT Guides for Analysis Services**

The SQL Customer Advisory Team has compiled a series of best practices and guides for performance tuning.

<http://sqlcat.com/sqlCat/b/whitepapers/archive/tags/Analysis+Services/default.aspx>

Be sure to read the performance and operations guides for SQL Server 2008 Analysis Services.

**Books Online**

SQL Server Books Online includes information about DMVs, trace events and XEvents that can be used for monitoring.

Analysis Services Instance Management

<http://technet.microsoft.com/en-us/library/hh230806>

Monitoring Analysis Services

<http://technet.microsoft.com/en-us/library/bb500198>