Data Collection

07/18/2016 • 6 minutes to read • Contributors

In this article

What is Data Collector?
Concepts
Terminology
Things you can do

THIS TOPIC APPLIES TO: SQL Server Azure SQL Database Azure SQL Data Warehouse Parallel Data Warehouse

The Data Collector is a component of SQL Server 2017 that collects different sets of data. Data collection either runs constantly or on a user-defined schedule. The data collector stores the collected data in a relational database known as the management data warehouse.

What is Data Collector?

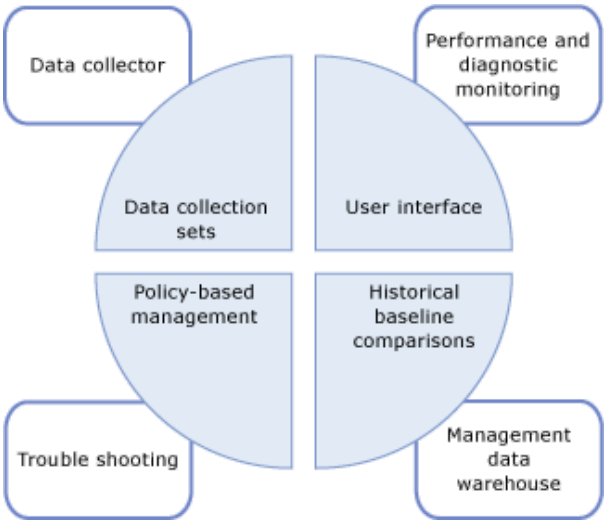
The data collector is a core component of the data collection platform for SQL Server 2017 and the tools that are provided by SQL Server. The data collector provides one central point for data collection across your database servers and applications. This collection point can obtain data from a variety of sources and is not limited to performance data, unlike SQL Trace.

The data collector enables you to adjust the scope of data collection to suit your test and production environments. The data collector also uses a data warehouse, a relational database that enables you to manage the data that you collect by setting different retention periods for your data.

The data collector supports dynamic tuning for data collection and is extensible through its API. For more information, see Data Collector Programming.

The following illustration shows how the data collector fits in the overall strategy for data collection and data management in SQL Server 2017.

FEEDBACK



Concepts

The data collector is integrated with SQL Server Agent and Integration Services, and uses both extensively. Before you work with the data collector, you should therefore understand certain concepts related to each of these SQL Server components.

SQL Server Agent is used to schedule and run collection jobs. You should understand the following concepts:

- Job
- Job step
- Job schedule
- Subsystem
- Proxy accounts

  For more information, see [Automated Administration Tasks (SQL Server Agent)](#).

  Integration Services (SSIS) is used to execute packages that collect data from individual data providers. You should be familiar with the following SSIS tools and concepts:

- SSIS package
- SSIS package configuration

  For more information, see [Integration Services (SSIS) Packages](#).

Terminology

target
An instance of the Database Engine in an edition of SQL Server that supports Data Collection. For more information about supported editions, see the "Manageability" section of Features Supported by the Editions of SQL Server 2016.

A *target root* defines a subtree in the target hierarchy. A *target set* is the group of targets that results from applying a filter to a subtree defined by a target root. A target root can be a database, an instance of SQL Server, or a computer instance.

target type
The type of target, which has certain characteristics and behavior. For example, a SQL Server instance target has different characteristics than a SQL Server database target.

data provider
A known data source, specific to a target type, that provides data to a collector type.

collector type
A logical wrapper around the SSIS packages that provide the actual mechanism for collecting data and uploading it to the management data warehouse.

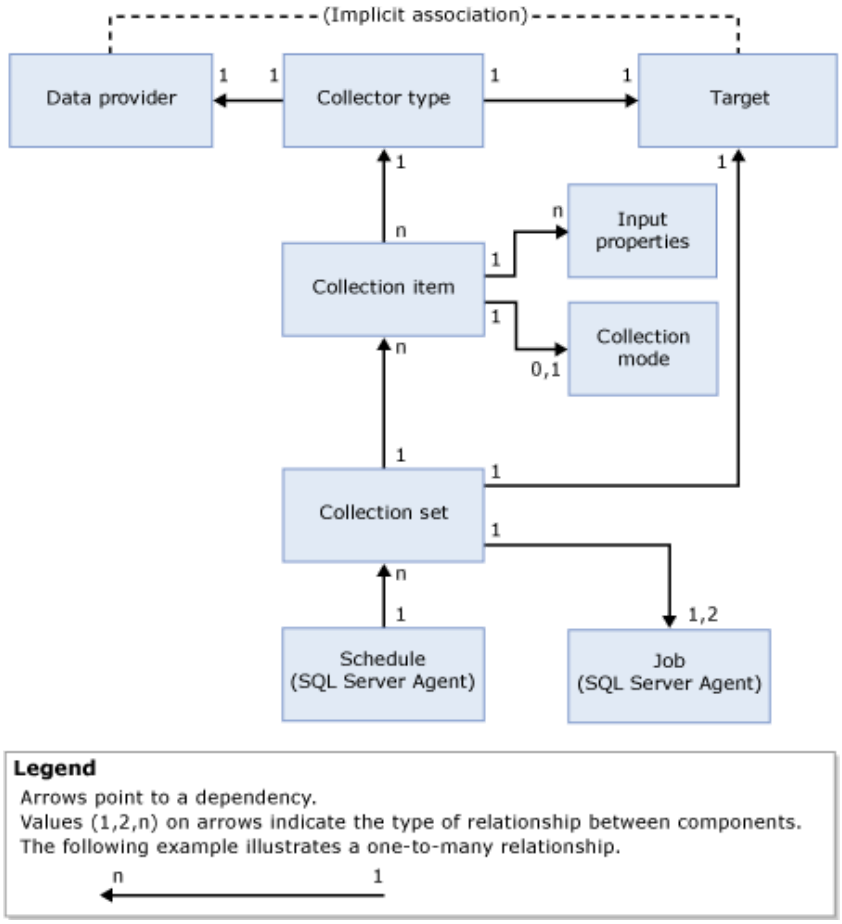
collection item
An instance of a collector type. A collection item is created with a specific set of input properties and a collection frequency.

collection set
A group of collection items. A collection set is a unit of data collection that a user can interact with through the user interface.

collection mode
The manner in which the data is collected and stored. Collection mode can be cached or non-cached. Cached mode supports continuous collection, whereas non-cached mode is intended for on-demand collection or a collection snapshot.

management data warehouse
A relational database used to store collected data.

The following illustration shows the dependencies and relationships between data collector components.



As shown in the illustration, the data provider is external to the data collector and by definition has an implicit relationship with the target. The data provider is specific to a particular target (for example, a SQL Server service such as the relational engine) and provides data such as system views in SQL Server, Performance Monitor counters, and WMI providers, that can be consumed by the data collector.

The collector type is specific to a target type, based on the logical association of a data provider to a target type. The collector type defines how data is collected from a specific data provider (by using schematized parameters) and specifies the data storage schema. The data provider schema and storage schema are required in order to store the data that is collected. The collector type also provides the location of the management data warehouse, which can reside on the computer running data collection or on a different computer.

A collection item, shown in the illustration, is an instance of a specific collector type, parameterized with input parameters, such as the XML schema for the collector type. All collection items must operate on the same target root or on an empty target root. This enables the data collector to combine collector types from the operating system or from a specific target root, but not from different target roots.

A collection item has a collection frequency defined that determines how often snapshots of values are taken. Although it is a building block for a collection set, a collection item cannot exist on its own.

Collection sets are defined and deployed on a server instance and can be run independently of each other. Each collection set can be applied to a target that matches the target types of all the collector

types that are part of a collection set. The collection set is run by a SQL Server Agent job or jobs, and data is uploaded to the management data warehouse on a predefined schedule.

All the data collected by different instances within the collection set is uploaded to the management data warehouse on the same schedule. This schedule is defined as a shared SQL Server Agent schedule and can be used by more than one collection set. A collection set is turned on or turned off as a single entity; collection items cannot be turned on or turned off individually.

When you create or update a collection set, you can configure the collection mode for collecting data and uploading it to the management data warehouse. The type of scheduling is determined by the type of collection: cached or non-cached. If the collection is cached, data collection and upload each run on a separate job. Collection runs on a schedule that starts when the SQL Server Agent starts and it runs on the frequency specified in the collection item. Upload runs according to the schedule specified by the user.

Under non-cached collection, data collection and upload both run on a single job, but in two steps. Step one is collection, step two is upload. No schedule is required for on-demand collection.

After a collection set is enabled, data collection can start, either according to a schedule or on demand. When data collection starts, SQL Server Agent spawns a process for the data collector, which in turn loads the Integration Services packages for the collection set. The collection items, which represent collection types, gather data from the appropriate data providers on the specified targets. When the collection cycle ends, this data is uploaded to the management data warehouse.

Things you can do

Description	Topic
Manage different aspects of data collection, such as enabling or disabling data collection, changing a collection set configuration, or viewing data in the management data warehouse.	Manage Data Collection
Use reports to obtain information for monitoring system capacity and troubleshooting system performance.	System Data Collection Set Reports
Use the Management Data Warehouse to collect data from a server that is a data collection target.	Management Data Warehouse
Exploit the server-side trace capabilities of SQL Server Profiler to export a trace definition that you can use to create a collection set that uses the Generic SQL Trace collector type	Use SQL Server Profiler to Create a SQL Trace Collection Set (SQL Server Management Studio)

ⓘ Note

The feedback system for this content will be changing soon. Old comments will not be carried over. If content within a comment thread is important to you, please save a copy. For more information on the upcoming change, we invite you to read our blog post.

0 Comments

Sign in

4 people listening

+ Follow

Post comment as...