


Monitor operations and activity of Azure Cognitive Search

06/30/2020 • 5 minutes to read •  +2

In this article

[Built-in monitoring](#)

[Add-on monitoring with Azure Monitor](#)

[Monitor user access](#)

[Next steps](#)

This article is an overview of monitoring concepts and tools for Azure Cognitive Search. For holistic monitoring, you can use a combination of built-in functionality and add-on services like Azure Monitor.

Altogether, you can track the following:

- Service: health/availability and changes to service configuration.
- Storage: both used and available, with counts for each content type relative to the quota allowed for the service tier.
- Query activity: volume, latency, and throttled or dropped queries. Logged query requests require [Azure Monitor](#).
- Indexing activity: requires [diagnostic logging](#) with Azure Monitor.

A search service does not support per-user authentication, so no identity information will be found in the logs.

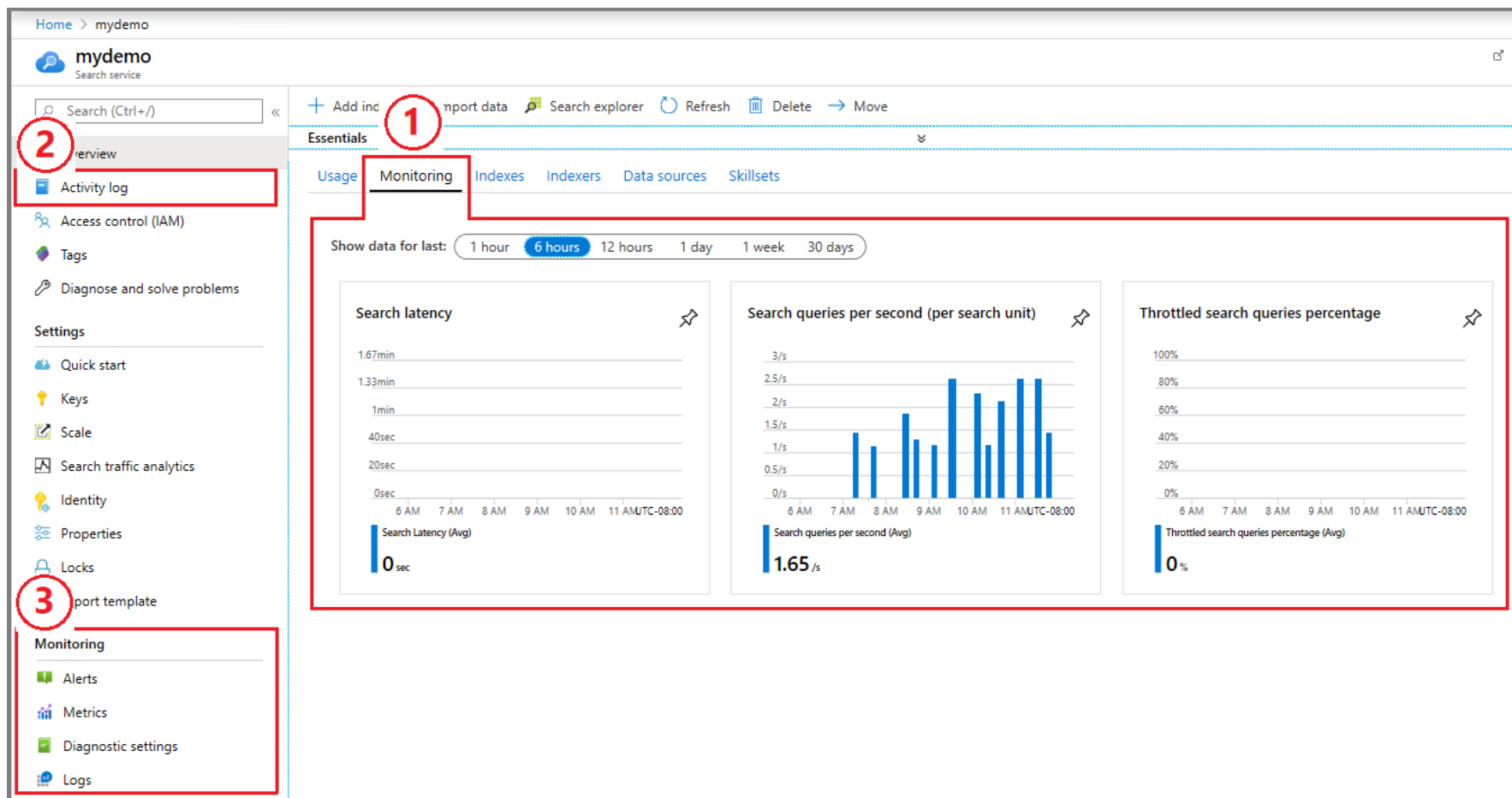
Built-in monitoring

Built-in monitoring refers to activities that are logged by a search service. With the exception of diagnostics, no configuration is required for this level of monitoring.

Azure Cognitive Search maintains internal data on a rolling 30-day schedule for reporting on service health and query metrics, which you can find in the portal or through these [REST APIs](#).

The following screenshot helps you locate monitoring information in the portal. Data becomes available as soon as you start using the service. Portal pages are refreshed every few minutes.

- **Monitoring** tab, on the main Overview page, shows query volume, latency, and whether the service is under pressure.
- **Activity log**, in the left navigation pane, is connected to Azure Resource Manager. The activity log reports on actions undertaken by Resource Manager: service availability and status, changes to capacity (replicas and partitions), and API key-related activities.
- **Monitoring** settings, further down, provides configurable alerts, metrics, and diagnostic logs. Create these when you need them. Once data is collected and stored, you can query or visualize the information for insights.



📌 Note

Because portal pages are refreshed every few minutes, the numbers reported are approximate, intended to give you a general sense of how well your system is servicing requests. Actual metrics, such as queries per second (QPS) may be higher or lower than the number shown on the page. If precision is a requirement, consider using APIs.

APIs useful for monitoring

You can use the following APIs to retrieve the same information found in the Monitoring and Usage tabs in the portal.

- [GET Service Statistics](#)
- [GET Index Statistics](#)
- [GET Document Counts](#)
- [GET Indexer Status](#)

Activity logs and service health

The [Activity log](#) page in the portal collects information from Azure Resource Manager and reports on changes to service health. You can monitor the activity log for critical, error, and warning conditions related to service health.

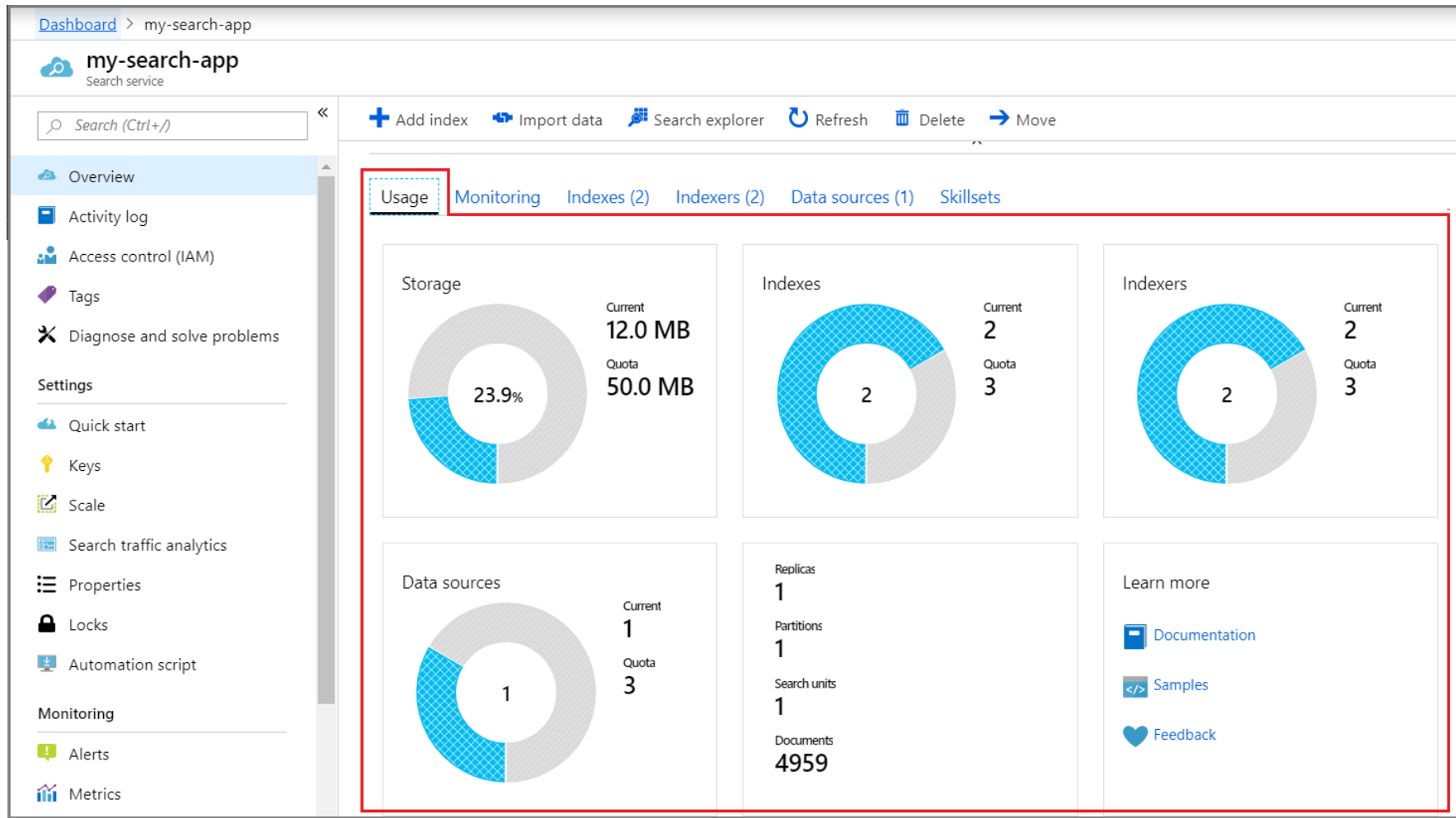
Common entries include references to API keys - generic informational notifications like *Get Admin Key* and *Get Query keys*. These activities indicate requests that were made using the admin key (create or delete objects) or query key, but do not show the request itself. For information of this grain, you must configure diagnostic logging.

You can access the **Activity log** from the left-navigation pane, or from Notifications in the top window command bar, or from the **Diagnose and solve problems** page.

Monitor storage in the Usage tab

For visual monitoring in the portal, the **Usage** tab shows you resource availability relative to current **limits** imposed by the service tier. If you are finalizing decisions about **which tier to use for production workloads**, or whether to **adjust the number of active replicas and partitions**, these metrics can help you with those decisions by showing you how quickly resources are consumed and how well the current configuration handles the existing load.

The following illustration is for the free service, which is capped at 3 objects of each type and 50 MB of storage. A Basic or Standard service has higher limits, and if you increase the partition counts, maximum storage goes up proportionally.



Alerts related to storage are not currently available; storage consumption is not aggregated or logged into the **AzureMetrics** table in Azure Monitor. To get storage alerts, you would need to **build a custom solution** that emits resource-related notifications, where your code checks for storage size and handles the response.

Add-on monitoring with Azure Monitor

Many services, including Azure Cognitive Search, integrate with [Azure Monitor](#) for additional alerts, metrics, and logging diagnostic data.

[Enable diagnostic logging](#) for a search service if you want control over data collection and storage. Logged events captured by Azure Monitor are stored in the **AzureDiagnostics** table and consists of operational data related to queries and indexing.

Azure Monitor provides several storage options, and your choice determines how you can consume the data:

- Choose Azure Blob storage if you want to [visualize log data](#) in a Power BI report.
- Choose Log Analytics if you want to explore data through Kusto queries.

Azure Monitor has its own billing structure and the diagnostic logs referenced in this section have an associated cost. For more information, see [Usage and estimated costs in Azure Monitor](#).

Monitor user access

Because search indexes are a component of a larger client application, there is no built-in methodology for controlling or monitoring per-user access to an index. Requests are assumed to come from a client application, for either admin or query requests. Admin read-write operations include creating, updating, deleting objects across the entire service. Read-only operations are queries against the documents collection, scoped to a single index.

As such, what you'll see in the activity logs are references to calls using admin keys or query keys. The appropriate key is included in requests originating from client code. The service is not equipped to handle identity tokens or impersonation.

When business requirements do exist for per-user authorization, the recommendation is integration with Azure Active Directory. You can use \$filter and user identities to [trim search results](#) of documents that a user should not see.

There is no way to log this information separately from the query string that includes the \$filter parameter. See [Monitor queries](#) for details on reporting query strings.

Next steps

Fluency with Azure Monitor is essential for oversight of any Azure service, including resources like Azure Cognitive Search. If you are not familiar with Azure Monitor, take the time to review articles related to resources. In addition to tutorials, the following article is a good place to start.

[Monitoring Azure resources with Azure Monitor](#)

Is this page helpful?

 Yes  No
