**Azure AI Search**

Azure AI Search ([formerly known as "Azure Cognitive Search"](https://learn.microsoft.com/en-us/azure/search/whats-new#new-service-name)) is an enterprise-ready search and retrieval system, with a comprehensive set of advanced search technologies, built for high-performance applications at any scale.

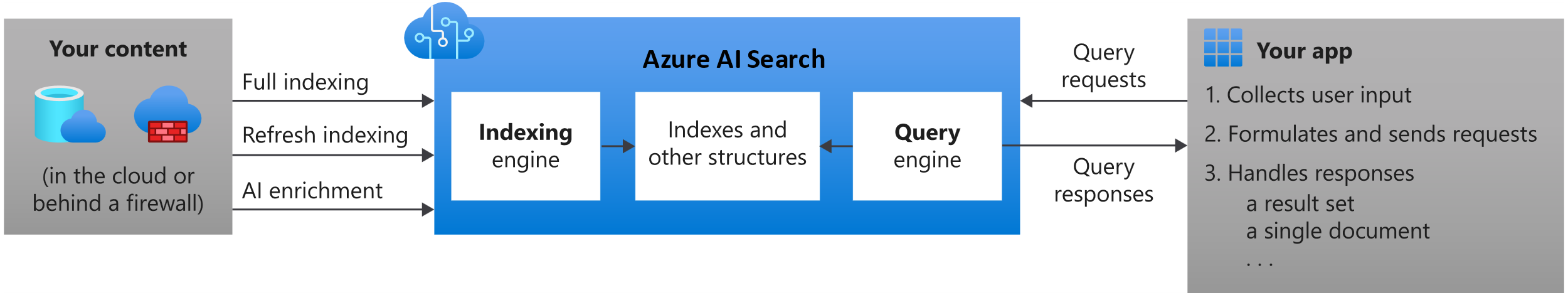
Azure AI Search is the recommended retrieval system for building RAG-based applications on Azure, with native LLM integrations between Azure OpenAI Service and Azure Machine Learning, and multiple strategies for relevance tuning.

Azure AI Search can be used in both traditional and GenAI scenarios. Common use cases include knowledge base insights (catalog or document search), information discovery (data exploration), retrieval-augmented generation (RAG), and indexing automation.

When you create a search service, you work with the following capabilities:

* A search engine for [vector search](https://learn.microsoft.com/en-us/azure/search/vector-search-overview) and [full text](https://learn.microsoft.com/en-us/azure/search/search-lucene-query-architecture) and [hybrid search](https://learn.microsoft.com/en-us/azure/search/hybrid-search-overview) over a search index
* Rich indexing with [integrated data chunking and vectorization](https://learn.microsoft.com/en-us/azure/search/vector-search-integrated-vectorization), [lexical analysis](https://learn.microsoft.com/en-us/azure/search/search-analyzers) for text, and [optional applied AI](https://learn.microsoft.com/en-us/azure/search/cognitive-search-concept-intro) for content extraction and transformation
* Rich query syntax for [vector queries](https://learn.microsoft.com/en-us/azure/search/vector-search-how-to-query), text search, [hybrid queries](https://learn.microsoft.com/en-us/azure/search/hybrid-search-how-to-query), fuzzy search, autocomplete, geo-search and others
* Relevance and query performance tuning with [semantic ranking](https://learn.microsoft.com/en-us/azure/search/semantic-search-overview), [scoring profiles](https://learn.microsoft.com/en-us/azure/search/index-add-scoring-profiles), [quantization for vector queries](https://learn.microsoft.com/en-us/azure/search/vector-search-how-to-quantization), and parameters for controlling query behaviours at runtime
* Azure scale, security, and reach
* Azure integration at the data layer, machine learning layer, Azure AI services and Azure OpenAI

Architecturally, a search service sits between the external data stores that contain your un-indexed data, and your client app that sends query requests to a search index and handles the response.



In your client app, the search experience is defined using APIs from Azure AI Search, and can include relevance tuning, semantic ranking, autocomplete, synonym matching, fuzzy matching, pattern matching, filter, and sort.

Across the Azure platform, Azure AI Search can integrate with other Azure services in the form of *indexers* that automate data ingestion/retrieval from Azure data sources, and *skillsets* that incorporate consumable AI from Azure AI services, such as image and natural language processing, or custom AI that you create in Azure Machine Learning or wrap inside Azure Functions.

**Inside a search service**

On the search service itself, the two primary workloads are *indexing* and *querying*.

* [**Indexing**](https://learn.microsoft.com/en-us/azure/search/search-what-is-an-index) is an intake process that loads content into your search service and makes it searchable. Internally, inbound text is processed into tokens and stored in inverted indexes, and inbound vectors are stored in vector indexes. The document format that Azure AI Search can index is JSON. You can upload JSON documents that you've assembled or use an indexer to retrieve and serialize your data into JSON.

[Applied AI](https://learn.microsoft.com/en-us/azure/search/cognitive-search-concept-intro) through a [skillset](https://learn.microsoft.com/en-us/azure/search/cognitive-search-working-with-skillsets) extends indexing with image and language models. If you have images or large unstructured text in source document, you can attach skills that perform OCR, analyze and describe images, infer structure, translate text and more. Output is text that can be serialized into JSON and ingested into a search index.

Skillsets can also perform [data chunking and vectorization during indexing](https://learn.microsoft.com/en-us/azure/search/vector-search-integrated-vectorization). Skills that attach to Azure OpenAI, the model catalog in Azure AI Foundry portal, or custom skills that attach to any external chunking and embedding model can be used during indexing to create vector data. Output is chunked vector content that can be ingested into a search index.

* [**Querying**](https://learn.microsoft.com/en-us/azure/search/search-query-overview) can happen once an index is populated with searchable content, when your client app sends query requests to a search service and handles responses. All query execution is over a search index that you control.

[Semantic ranking](https://learn.microsoft.com/en-us/azure/search/semantic-search-overview) is an extension of query execution. It adds secondary ranking, using language understanding to reevaluate a result set, promoting the most semantically relevant results to the top.

[Integrated vectorization](https://learn.microsoft.com/en-us/azure/search/vector-search-integrated-vectorization) is also an extension of query execution. If you have vector fields in your search index, you can submit raw vector queries or text that's vectorized at query time.

**Why use Azure AI Search?**

Azure AI Search is well suited for the following application scenarios:

* Use it for traditional full text search and next-generation vector similarity search. Back your generative AI apps with information retrieval that leverages the strengths of both keyword and similarity search. Use both modalities to retrieve the most relevant results.
* Consolidate heterogeneous content into a user-defined and populated search index composed of vectors and text. You maintain ownership and control over what's searchable.
* [Integrate data chunking and vectorization](https://learn.microsoft.com/en-us/azure/search/vector-search-integrated-vectorization) for generative AI and RAG apps.
* [Apply granular access control](https://techcommunity.microsoft.com/t5/azure-ai-services-blog/access-control-in-generative-ai-applications-with-azure/ba-p/3956408) at the document level.
* Offload indexing and query workloads onto a dedicated search service.
* Easily implement search-related features: relevance tuning, faceted navigation, filters (including geo-spatial search), synonym mapping, and autocomplete.
* Transform large undifferentiated text or image files, or application files stored in Azure Blob Storage or Azure Cosmos DB, into searchable chunks. This is achieved during indexing through [AI skills](https://learn.microsoft.com/en-us/azure/search/cognitive-search-concept-intro) that add external processing from Azure AI.
* Add linguistic or custom text analysis. If you have non-English content, Azure AI Search supports both Lucene analyzers and Microsoft's natural language processors. You can also configure analyzers to achieve specialized processing of raw content, such as filtering out diacritics, or recognizing and preserving patterns in strings.

**Compare search options**

Customers often ask how Azure AI Search compares with other search-related solutions. The following table summarizes key differences.

A screenshot of a computer

Description automatically generated

**How to get started**

Functionality is exposed through the Azure portal, simple [REST APIs](https://learn.microsoft.com/en-us/rest/api/searchservice/), or Azure SDKs like the [Azure SDK for .NET](https://learn.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk). The Azure portal supports service administration and content management, with tools for prototyping and querying your indexes and skillsets.

**Use the Azure portal**

An end-to-end exploration of core search features can be accomplished in four steps:

1. [**Decide on a tier**](https://learn.microsoft.com/en-us/azure/search/search-sku-tier) and region. One free search service is allowed per subscription. All quickstarts can be completed on the free tier. For more capacity and capabilities, you'll need a [billable tier](https://azure.microsoft.com/pricing/details/search/).
2. [**Create a search service**](https://learn.microsoft.com/en-us/azure/search/search-create-service-portal) in the Azure portal.
3. [**Start with Import data wizard**](https://learn.microsoft.com/en-us/azure/search/search-get-started-portal). Choose a built-in sample or a supported data source to create, load, and query an index in minutes.
4. [**Finish with Search Explorer**](https://learn.microsoft.com/en-us/azure/search/search-explorer), using a portal client to query the search index you just created.

**Use APIs**

Alternatively, you can create, load, and query a search index in atomic steps:

1. [**Create a search index**](https://learn.microsoft.com/en-us/azure/search/search-what-is-an-index) using the portal, [REST API](https://learn.microsoft.com/en-us/rest/api/searchservice/indexes/create), [.NET SDK](https://learn.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk), or another SDK. The index schema defines the structure of searchable content.
2. [**Upload content**](https://learn.microsoft.com/en-us/azure/search/search-what-is-data-import) using the ["push" model](https://learn.microsoft.com/en-us/azure/search/tutorial-optimize-indexing-push-api) to push JSON documents from any source, or use the ["pull" model (indexers)](https://learn.microsoft.com/en-us/azure/search/search-indexer-overview) if your source data is of a [supported type](https://learn.microsoft.com/en-us/azure/search/search-indexer-overview#supported-data-sources).
3. [**Query an index**](https://learn.microsoft.com/en-us/azure/search/search-query-overview) using [Search explorer](https://learn.microsoft.com/en-us/azure/search/search-explorer) in the portal, [REST API](https://learn.microsoft.com/en-us/azure/search/search-get-started-rest), [.NET SDK](https://learn.microsoft.com/en-us/dotnet/api/azure.search.documents.searchclient.search), or another SDK.

**Use accelerators**

Or try solution accelerators:

* [**Chat with your data** solution accelerator](https://github.com/Azure-Samples/chat-with-your-data-solution-accelerator) helps you create a custom RAG solution over your content.
* [**Conversational Knowledge Mining** solution accelerator](https://github.com/microsoft/Customer-Service-Conversational-Insights-with-Azure-OpenAI-Services) helps you create an interactive solution to extract actionable insights from post-contact center transcripts.
* [**Document Knowledge Mining accelerator**](https://github.com/microsoft/Document-Knowledge-Mining-Solution-Accelerator) helps you process and extract summaries, entities, and metadata from unstructured, multimodal documents.
* [**Build your own copilot** solution accelerator](https://github.com/microsoft/Build-your-own-copilot-Solution-Accelerator), leverages Azure OpenAI Service, Azure AI Search and Microsoft Fabric, to create custom copilot solutions.
  + [Generic copilot](https://github.com/microsoft/Generic-Build-your-own-copilot-Solution-Accelerator) helps you build your own copilot to identify relevant documents, summarize unstructured information, and generate Word document templates using your own data.
  + [Client Advisor](https://github.com/microsoft/Build-your-own-copilot-Solution-Accelerator/blob/main/ClientAdvisor/README.md) all-in-one custom copilot empowers Client Advisor to harness the power of generative AI across both structured and unstructured data. Help our customers to optimize daily tasks and foster better interactions with more clients
  + [Research Assistant](https://github.com/microsoft/Build-your-own-copilot-Solution-Accelerator/blob/main/ResearchAssistant/README.md) helps build your own AI Assistant to identify relevant documents, summarize and categorize vast amounts of unstructured information, and accelerate the overall document review and content generation.

**Pricing of Azure AI Search**

<https://azure.microsoft.com/en-in/pricing/details/search>/

**How to create/use Azure AI Search for Blob Storage?**

**Task 1: Setup AI Search Service**

* Navigate to the Azure Portal, and search for the **AI Search service**.
* Create a new search service and fill in the required details

A screenshot of a service

Description automatically generated

* Try to choose a location as close as possible to your Blob storage location.
* In the Free trial, we can create only**1** AI Search Service and scaling is not allowed.

**Task 2: Connect the Azure Blob Storage Account to the AI Search Service**

* Once the AI Search service is created, click on **Import Data.**

A screenshot of a chat

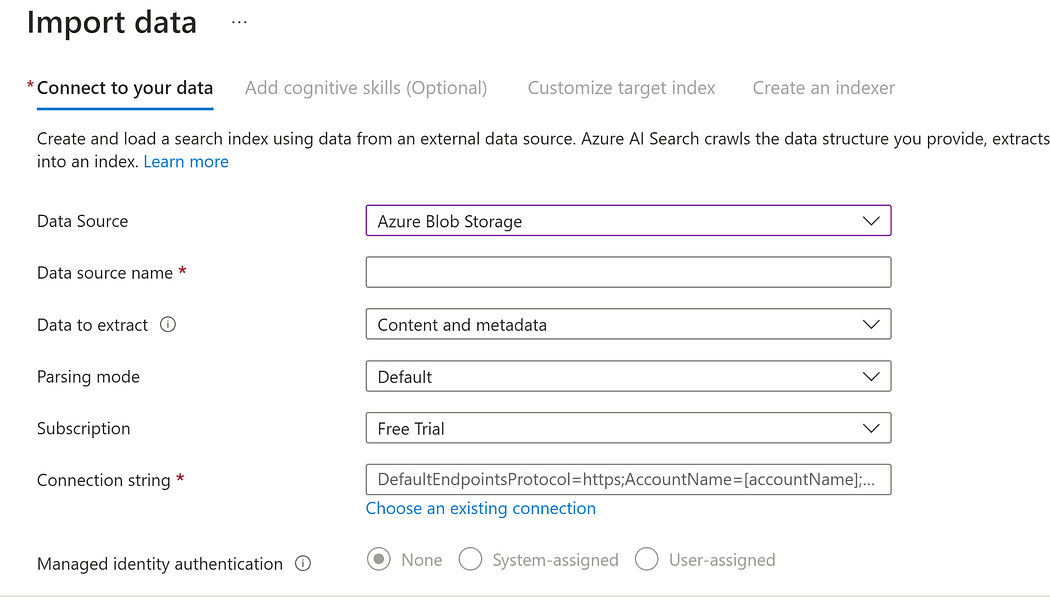
Description automatically generated

* Select Azure Blob Storage as the data source.

A screenshot of a computer

Description automatically generated

* Provide the required details



* Cognitive skills are optional and are mainly used for images and unstructured data.

**Task 3: Create an index on Azure blob files**

* Once the data source is connected, we can move to the Index creation.

A screenshot of a computer

Description automatically generated

* Provide a unique Index name.
* Default Key is the **metadata\_storage\_path**which is unique for every Blob file.
* You can select any other **Key**as well which is uniquely generated by the data source.

A screenshot of a survey

Description automatically generated

* If the files contain a smaller number of fields, then the search service can auto-detect the schema.
* But if there are more fields, you can manually write the fields based on which you want to do the indexing.
* You can select the type of field like Object, String, Number, Boolean, Array.
* Also, you can select the features applicable to the fields like searchable, filterable, retrievable, sortable, and facetable.
* You can remove the extra fields which are not required.
* Once the Index is created, the next step is to create an Indexer.

A screenshot of a computer

Description automatically generated

* In the Indexer, you can select the scheduler running frequency.
* In the advanced option, you can exclude/include Blob files with specific extensions from the indexing.
* You can specify how many failed records are allowed during indexing. -1 (No Limit), 0 (No Failed Record Allowed).
* Batch size refers to how many files to index in one run.
* Data to extract: It refers to what type of data should be extracted from the source by the Indexer.
* Parsing Mode: It is specified as per the format of data inside the files.

A screenshot of a computer

Description automatically generated

**Task 4: Search files using Azure AI Search Service**

* Once the Indexer is ready, it will run and index the files.

A screenshot of a computer

Description automatically generated

* Now, we can search the files based on the fields specified during Indexing using **REST APIs or Search Explorer**.
* Navigate to the search explorer tab in the Azure AI Search and select the Index you want to search in.

A screenshot of a computer

Description automatically generated

A long black line on a white background

Description automatically generated

* Write the value, you want to search and hit the search button

A screenshot of a computer

Description automatically generated

**Search using the REST APIs of Azure AI Search:**

* For searching using the REST APIs, you need the URL and API-Key of the search service.
* URL can be found in the overview page.
* API-Key can be found in the settings->keys page

A screenshot of a computer

Description automatically generated

Now, you can use the REST APIs of Azure AI Search:

A screenshot of a computer

Description automatically generated

**Key points to remember while using Azure AI Search:**

* **Resource Group:** It is a container that holds related resources in Azure solution.
* **Tags:** It can be used by the Azure admin to track the billing of related services having the same tag.
* Only the HOT and COOL access tier of Azure Blob Storage can be used in Indexing in Azure AI Search.
* **Track Deletions** must need to be turned on once the index is created. So, if a Blob file is deleted, then it will be automatically removed from the index.

A screenshot of a computer

Description automatically generated

* The free tier can have only 3 Index and 3 Indexers.
* It is recommended to create 1 indexer per Index.
* You can create an Index on sub-folders inside the Blob container.
* The blob container is a mandatory field in an Index.
* Index and Indexer can be created using the JSON body.
* Once the Index is created, we cannot change the field values and types.