

Creating Connectors

The biggest change in Aspire 3.1 is related to the way connectors work. They now use an external database (MongoDB) to hold all of the crawling information such as document urls, status, statistics, snapshots (for incrementals), logs, etc. This allows the connectors to work distributed from the architectural design.

All of the connectors run under the same principles, using the same logic, so that each connector is more like a Repository Access Provider. We keep them as simple as possible, rather than a complex (multi-threaded) crawling application. The complexity of distributed crawling and multi-threading relies on the Connector Framework.

Responsibilities that the Connector developers implement:

- **Scan** the repository document containers to discover new documents to process
- **Populate** document metadata
- **Fetch** document content

If you want to create your connector right away, go to [Write Your Own Connector from Scratch](#)

Responsibilities of the Connector Framework (you don't have to worry about this):

- Multi-threading processing
- Distribute the crawl processing
- Store and fetch documents from the database.
- Maintain a snapshot for incremental crawling (adding, updating or deleting documents)
- Handle statistics
- Start, Pause, Stop, Resume the crawl
- Send the documents to the respective workflows for processing and search engine indexing

If you want to learn more about the Connector Framework check out [NoSQL Connector Framework Overview](#).

What's next?

- [NoSQL Connector Framework Overview](#)
 - [NoSQL Connector Framework Implementation](#)
 - [NoSQL Connector Framework Security Implementation](#)
 - [MongoDB Collections Description](#)
 - [Failed Documents Handling](#)
- [Write Your Own Connector from Scratch](#)

