📂 Deasy Labs

This section covers the fundamental building blocks of Deasy. Understanding these concepts is essential for effectively using the platform.

---

1. Data Connectors

Data Connectors are connections to your document repositories where unstructured content resides. They serve as the entry point for all documents that you want to enrich with Deasy's metadata.

What is a Data Connector?

A Data Connector establishes a secure connection between Deasy and your document storage. Once connected, the platform can:

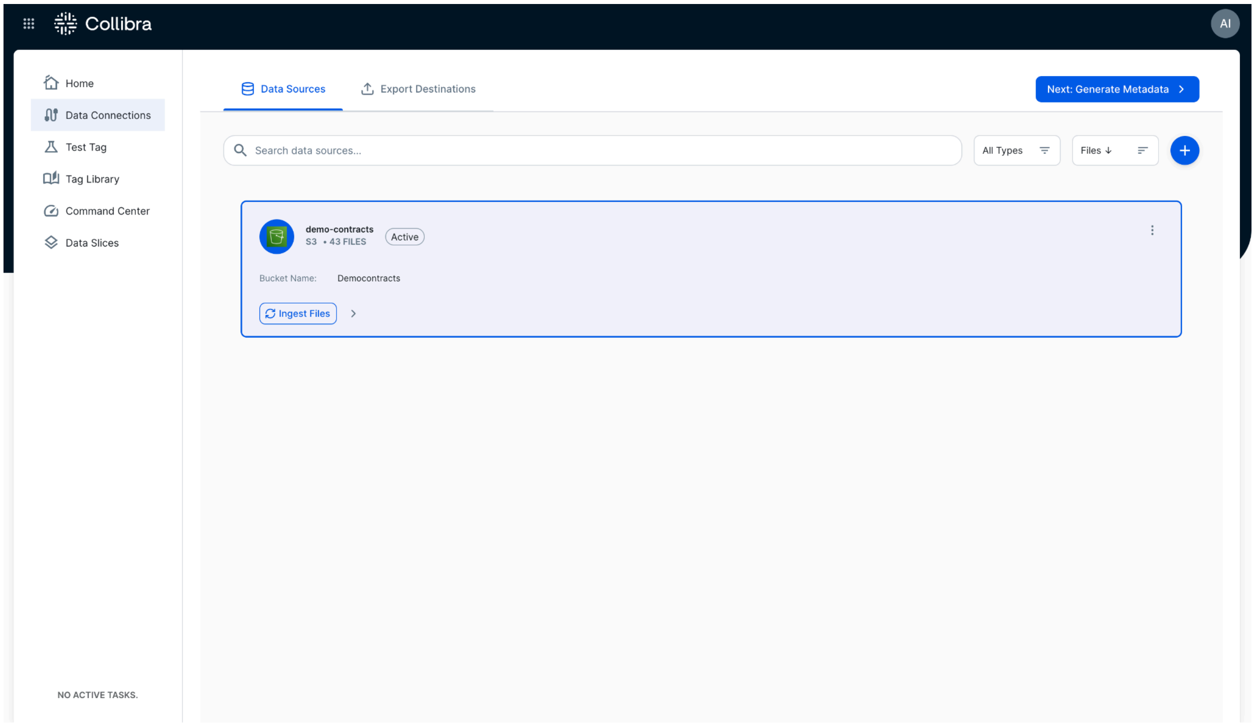
- **Discover** all documents within the repository
- **Read** document content for metadata extraction
- **Write** enriched metadata back to the source

Supported Data Connectors

Source Type	Description	Key Configuration	Ideal Use Case
Amazon S3	AWS cloud object storage	Bucket name, Access Key, Secret Key	Large-scale document archives, cloud-native workflows
SharePoint	Microsoft 365 document management	Client ID, Client Secret, Tenant ID, Site Name	Enterprise document libraries, Office 365 environments
PostgreSQL	Relational database with pgvector extension	Host URL, Database name, User credentials, Port	Structured + unstructured hybrid data, existing database workflows
Qdrant	Purpose-built vector database for AI	API Key, Collection name, URL	Semantic search applications, RAG pipelines

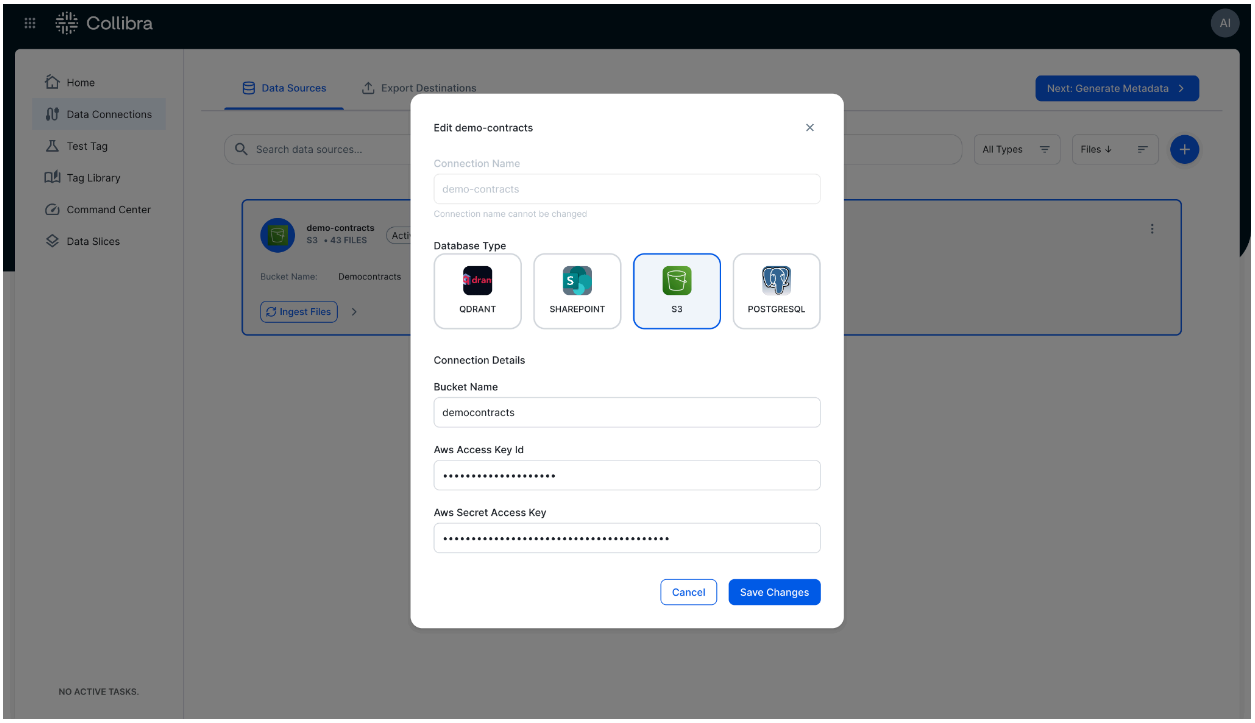
Key Features

- **Multiple Profiles** — Create and manage multiple Data Connector connections
- **Connection Testing** — Validate credentials before saving
- **Active Profile Selection** — Switch between Data Connectors with one click
- **Schema Configuration** — Customize field mappings (filename key, text key, tags key)



demo-contracts
S3 • 43 FILES
Active

Bucket Name: Democontracts



Edit demo-contracts

Connection Name: demo-contracts

Connection name cannot be changed

Database Type: S3

Connection Details

Bucket Name: democontracts

Aws Access Key Id: ••••••••••••••••••••

Aws Secret Access Key: ••

2. Destinations

Destinations are target systems where enriched metadata and document data can be exported. They enable you to send processed data from Deasy to external databases, vector stores, and document management systems.

What is a Destination?

A Destination establishes an outbound connection from Deasy to an external storage system. While Data Connectors bring documents *into* the platform, Destinations push enriched data *out* to:

- Vector databases for semantic search applications
- Document management systems with updated metadata
- Databases for downstream analytics and applications

Supported Destinations

Destination Type	**Description**	**Key Configuration**	**Ideal Use Case**
Qdrant	Vector database for AI applications	Collection name, URL, API Key	Semantic search, RAG pipelines
PostgreSQL	Relational database with vector support	Host, Database, Collection, Credentials	Structured analytics, hybrid search
Azure SQL	Microsoft cloud database	Server, Database, Table, Credentials	Enterprise data warehouses
SharePoint	Microsoft 365 document management	Client ID/Secret, Tenant ID, Site Name	Enriching original documents with metadata columns

Export Options

When exporting to a destination, you can configure:

Option	**Description**	**Values**
Export Level	What data granularity to export	`file` (document-level), `chunk` (segment-level), `both`
Export Tags	Specific metadata tags to include	List of tag names, or empty for all
Export Nodes	Include vector embeddings	`true` / `false`
Export Metadata	Include extracted metadata	`true` / `false`
Metadata Format	How metadata is stored	`column_store` (separate columns), `json_store` (single JSON column)

Export Process

1. Small Exports (< 100 files): Processed synchronously with immediate results
1. Large Exports (≥ 100 files): Processed in the background with progress tracking
    - Returns a `tracker_id` for monitoring
    - Batched processing for reliability

Destination-Specific Features

SharePoint Destination:

- Creates site columns for each metadata tag
- Supports choice columns with predefined values
- Updates file properties directly in document library
- Maintains column-to-tag mapping

Vector Database Destinations (Qdrant, PostgreSQL):

- Exports embeddings with metadata payloads

- Configurable vector dimensions
- Collection/table creation options
- Supports dense and sparse vectors

📷 IMAGE NEEDED: Destinations Configuration Panel

📷 IMAGE NEEDED: Export Progress Tracker

3. Projects

Projects are organizational containers that group related work together — including Data Connectors, taxonomies, and sensitivity detection settings.

What is a Project?

Think of a Project as a workspace for a specific initiative. For example:

- "Contract Analysis 2024" — for processing legal contracts
- "HR Document Compliance" — for employee document PII detection
- "Financial Reports Q4" — for extracting financial metrics

Project Components

Component	**Description**	**Required**
Name	Unique identifier for the project	✅ Yes
Description	Optional notes about the project purpose	❌ No
Data Connectors	One or more connected data repositories	✅ At least one
Data Slice	Optional filtered subset of data	❌ No (defaults to "All Data")
Taxonomies	One or more tag hierarchies to apply	❌ No
Sensitive Data Detection	Enable PII/PHI/PCI scanning	❌ No

Sensitivity Detection Options

When creating a project, you can enable automatic sensitive data detection:

Type	Full Name	Examples
PII	Personal Identifiable Information	Names, emails, addresses, phone numbers, SSN
PHI	Protected Health Information	Medical records, diagnoses, prescriptions, insurance IDs
PCI	Payment Card Industry Data	Credit card numbers, bank accounts, payment details

📷 IMAGE NEEDED: Projects Dashboard

📷 IMAGE NEEDED: Create Project Side Panel

4. Taxonomies & Tags

Tags are the metadata attributes you want to extract or classify from your documents. **Taxonomies** organize tags into hierarchical structures that define parent-child relationships.

What is a Tag?

A Tag defines a specific piece of information you want to capture from documents. Each tag has:

Property	**Description**	**Example**
Name	The tag identifier	Contract Type
Description	Instructions for the AI on what to extract	"Identify the type of legal agreement (NDA, MSA, SOW, etc.)"
Output Type	How values are returned	Word, Number, Date
Max Values	How many values get returned	1 to however many relevant values an AI can find
Available Values	Predefined options (for classification)	["NDA", "MSA", "SOW", "Employment Agreement"]
Strategy	Extraction method	LLM (AI), Regex (Pattern), Rule-based

Tag Types Explained

Tag Type	How It Works	When to Use	Example
Classification Tags	AI chooses from predefined list of values	When you have a known set of categories	Document Type: Contract, Invoice, Report
Extraction Tags	AI extracts open-ended values from text	When the value is unpredictable	Contract Value: $1,500,000, €2.3M
Pattern Tags	General: AI + Regex Pattern Sensitivity: NLP detection of PII/PHI/PCI	For compliance, scaleable cheap classification/extraction, keyword-search, etc.	SSN: XXX-XX-XXXX, Email: user@domain.com

What is a Taxonomy?

A Taxonomy is a structure of different tags. It enables:

- **Hierarchical organization** - Child tags only get generated when parent conditions are met
- **Conditional extraction** - Extract "Contract Value" only when "Document Type" = "Contract"
- **Efficient processing** - Skip irrelevant branches to save processing time

Taxonomy Example

- **Document Type** (Classification: Contract | Invoice | Report)
    - 📄 **Contract**
        - 💰 Contract Value (Extraction)
        - 👥 Parties Involved (Extraction)
        - 📅 Effective Date (Extraction)
        - ⏰ Termination Date (Extraction)
    - 📄 **Invoice**
        - 💵 Invoice Amount (Extraction)
        - 📅 Due Date (Extraction)
        - 🏢 Vendor Name (Extraction)
    - 📄 **Report**
        - 📊 Report Category (Classification: Financial | Operational | Compliance)
        - 📅 Report Period (Extraction)

In this taxonomy:

- First, the AI classifies the document as Contract, Invoice, or Report
- Then, it only extracts the relevant child tags for that document type
- A Contract won't have "Invoice Amount" extracted, saving time and cost

📷 **IMAGE NEEDED: Taxonomy Graph View**

📷 **IMAGE NEEDED: Tag Editor Side Panel**

---

4. Metadata

Metadata represents the actual extracted values that result from applying Tags to your documents. While Tags define *what* to extract, Metadata is the *extracted data* itself.

What is Metadata?

Metadata in Deasy is the structured output generated when Tags are applied to documents. Each piece of metadata includes:

Property	**Description**	**Example**
Values	The extracted or classified value(s)	`["NDA", "Non-Disclosure Agreement"]`
Evidence	Text snippet supporting the extraction	`"This Non-Disclosure Agreement is entered into..."`
Confidence	AI confidence score (0-1)	`0.95`

Metadata Levels

Metadata exists at two levels:

Level	Description	Use Case
File-Level	Aggregated metadata for the entire document	Document classification, search filters
Chunk-Level	Granular metadata per text segment	Precise evidence location, RAG retrieval

Metadata Standardization

Deasy includes AI-powered standardization to clean and normalize extracted values:

Feature	Description
Deduplication	Merge similar values (e.g., "Inc." and "Incorporated")
Normalization	Standardize formats (dates, currencies, names)
Bulk Standardization	Apply standardization across multiple tags

5. Data Slices

Data Slices are filtered subsets of your data based on metadata. They allow you to focus on specific segments of your document repository without affecting the entire dataset.

What is a Data Slice?

A Data Slice applies filter conditions to your Data Connector, creating a "view" of documents that match specific criteria. The original data is unchanged — you're simply defining which documents to work with.

How Data Slices Help

Benefit	**Description**
Targeted Processing	Run extraction on only relevant documents (e.g., just 2024 contracts)
Focused Analysis	View and analyze specific document categories
Efficient Workflows	Avoid reprocessing already-enriched documents
Team Collaboration	Share team specific data slices with team members
Controllability	Run your downstream applications (Chatbot, Agent, Feature Engineering pipeline) on controlled data

Creating a Data Slice

Data Slices are created based on metadata filters.

Data Slice Properties

Property	**Description**
Name	User-defined identifier
Description	Optional notes about what's included
Data Connector	Which Data Connector it's derived from
Document Count	Number of files matching the conditions

Last Updated	When the slice was created or refreshed
Conditions	The filter rules that define the slice based on the metadata

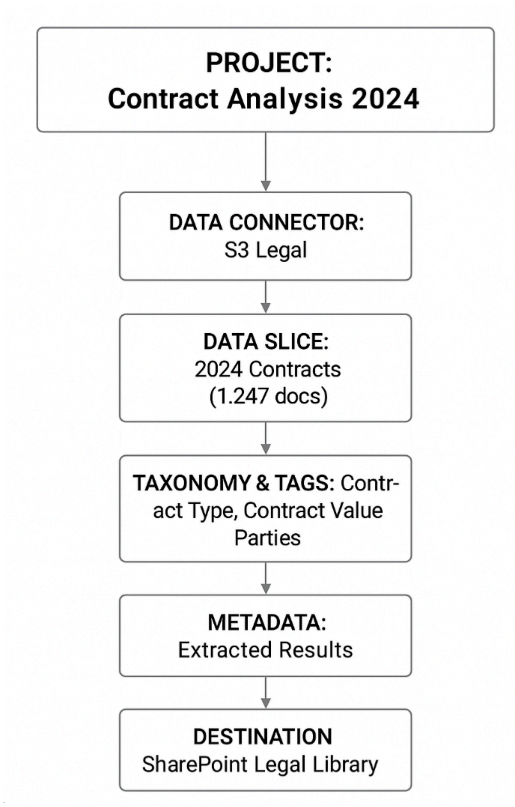
Example Data Slices

Slice Name	**Filter Conditions**	**Document Count**
"2024 Contracts"	Document Type = Contract AND Year = 2024	1,247
"High-Value Invoices"	Invoice Amount > $10,000	89
"Documents with PII"	SSN_Detected = Yes OR Email_Detected = Yes	3,521
"Unprocessed Files"	Document Type = null	5,892

📷 IMAGE NEEDED: Data Slice Selection Screen

📷 IMAGE NEEDED: Create Data Slice Modal/Flow

Concept Relationships Diagram



Summary Table

Concept	**What It Does**	**Think of It Like...**
Data Connector	Connects to where your documents live	Plugging in an external drive
Project	Only work with the data sources and taxonomies you want to work with	A project folder
Tag	Defines what info to extract	A question on a form
Taxonomy	Organizes tags into a structure	An outline or checklist
Metadata	The actual extracted information	The filled-out answers

Data Slice	Filters to specific documents	A saved search
Destination	Sends enriched data to other systems	Exporting to share