

Haystack is a Document Intelligence Framework, not an Agent Framework

Even without retrieval or LLMs, companies need:

- Ingesting millions of PDFs/technical manuals
- Extracting content from formal documents
- Cleaning, normalizing, chunking
- Metadata extraction
- Data versioning & schema control
- Pipeline Serialization (YAML) and DocumentStore
- Document pipelines (DAGs)
- Text indexing & transformations
- Regulatory compliance workflows

Haystack excels at “document → knowledge → structured pipeline” transformation.

LangGraph and ADK do *not* solve this domain at all.

Companies like BMW and Airbus have **huge technical document ecosystems**,

So, they prioritize document infrastructure, not just agents.

Haystack offers DAG Pipeline Orchestration (deterministic, predictable)

Unlike agent frameworks (dynamic, stochastic, unpredictable), Haystack provides:

- **deterministic, rule-based pipeline graphs**
- **explicit node execution**
- **traceable inputs/outputs per node**
- **repeatable execution every time**
- **no hallucinating tool choices**

Enterprises prefer:

“Predictable pipelines > Unpredictable agents”

Because:

- compliance reasons,
- safety reasons,
- maintenance reasons,
- and reliability reasons.

LangGraph and ADK are powerful, but **agents are not deterministic**.

Haystack has production-grade evaluation tools

Haystack gives:

- dataset-based evaluation
- pipeline-level validation
- structured scoring
- regression detection
- unit-level component evaluation
- semantic similarity scoring
- robustness checks

Large companies require *measurable, testable, repeatable* evaluation before every deployment.
LangGraph and ADK have agent-eval tooling but **lack robust document/pipeline evaluation**.
Evaluation is a first-class citizen in Haystack.

Haystack supports multi-step, non-LLM NLP pipelines

Companies still use non-LLM steps such as:

- classification models
- entity extraction models
- summarization models
- sentiment/intent models
- rule-based models
- hybrid ML pipelines

Haystack can orchestrate:

OCR → NER → Classifier → Normalizer → Rule Engine → Summarizer

LangGraph and ADK are centered around **LLM agent** reasoning.

Haystack is centered around **text-processing workflow** reasoning.

Haystack integrates with enterprise search and content systems

Companies like BMW and Airbus run:

- ElasticSearch clusters
- OpenSearch deployments
- FAISS/Milvus indexing pipelines
- metadata-driven filtering
- internal document stores
- compliance archives

Haystack works **natively** with these systems.

LangGraph and ADK do not.

This makes Haystack a **backend data-first choice**, not an agent-first one.

Haystack is deeply DevOps-friendly

Enterprise engineering teams love it because it is:

- container-friendly
- Kubernetes-native
- CI/CD-friendly
- has config-based pipeline definitions
- supports directory-based pipeline loading
- easily logged and monitored
- deterministic in production

LangGraph and ADK are powerful agent SDKs,
but do not provide **full data-engineering DevOps layers**.

Haystack emphasizes Auditability, Traceability, Explainability

Companies still need:

- per-step logs
- provenance of pipeline transformations
- reproducible results
- deterministic behavior
- exact origin of processed data
- low variability between runs

Haystack guarantees:

- each node → trace
- inputs, outputs → logged
- reproducible state
- pipeline-level versioning

LangGraph/ADK → dynamic agent behavior changes per run.

Enterprises cannot deploy untraceable, non-deterministic flows.

Haystack supports Regulatory & Safety Requirements

Industries like automotive, aerospace, and semiconductors must comply with:

- ISO documentation standards
- safety guidelines
- internal GDPR requirements
- EU AI Act readiness
- traceability audits

Haystack's deterministic pipelines, data lineage, and versioned workflows make it safer.

Agent-first systems (LangGraph/ADK) are not audit-oriented.

Haystack is mature, stable, low-risk, and time-tested

Enterprises prefer:

- stable APIs
- minimal breaking changes
- multi-year support
- long software lifespan
- predictable roadmaps

Haystack has existed for years and is already used in enterprise production.

LangGraph and ADK are newer, evolving fast, and may change frequently.

Companies avoid operational risk.

Haystack is the Knowledge Foundation while LangGraph/ADK are Agent Layers

This is the most important concept:

- **Haystack = Data + Document + Knowledge foundation**
- **LangGraph = Agent orchestration on top**
- **ADK = Agent/tool lifecycle & evaluation layer**

Large companies **do not replace Haystack**.

They **extend** it with agent frameworks.

Haystack is the “database + index + pipeline engine”.

Agent SDKs are the “business logic brain”.

You need BOTH.

Summary

Companies use Haystack because it solves the **enterprise-critical problems of document pipelines, deterministic workflows, evaluation, auditability, and long-term stability** — things that LangGraph and ADK do NOT specialize in, even though they excel at agent behavior.