How Modern AI Systems Actually Work

A field guide to LLMs, retrieval, agents, MCP, memory, tools, deterministic code, and trust - without the LinkedIn mythology.

PREPARED FOR JEFF FRANZEN · JULY 2026 · INDEPENDENT LEARNING GUIDE

The simple diagrams are useful, but incomplete. An LLM is not the whole system. A dependable AI product layers a language model over approved information, controlled tools, ordinary code, permissions, and evidence that the result is correct.

> *The model handles ambiguity. Code handles certainty. Retrieval supplies evidence. Tools change the world. A human still owns the outcome.*

The corrected five-part stack

INTERPRET	GROUND	CONTROL	ACT	CONNECT
LLM	**Context**	**Code**	**Tools**	**MCP / APIs**
Understands requests, reasons, drafts language, and chooses among available options.	RAG, files, APIs, maps, databases, and current organizational knowledge.	Exact calculations, joins, business rules, validation, permissions, and repeatable workflows.	Search, GIS queries, email, browsers, databases, file writes, deployments, and other actions.	Standard or direct interfaces that expose data and tools to the AI application.

What wraps the stack

- Identity and least-privilege access
- Human approval for consequential actions
- Source citations and provenance
- Testing, monitoring, and recovery

The central design rule

If the task can be drawn as a stable flowchart, implement that part in code. Use the model where interpretation, judgment, or natural language is genuinely required.

RAG: giving the model an open book

Retrieval-Augmented Generation does not permanently teach the model your documents. It searches an external knowledge source at question time, places selected passages into the model's working context, and asks the model to answer from that evidence.

1. Prepare
Parse, clean, split, label, and permission the source material.

>

2. Retrieve
Translate the question into a search and find likely evidence.

>

3. Refine
Filter, rerank, deduplicate, and verify relevance.

>

4. Answer
Give the evidence to the model; return citations and uncertainty.

What the eight-step graphic gets right

1 **Documents must become searchable**
Collection, chunking, embeddings, and storage are common preparation steps for semantic retrieval.

2 **The question becomes a search**
The system compares the user's need with stored content, retrieves candidate passages, and adds them to the prompt.

3 **The evidence should travel with the answer**
A serious system returns source links or document references so the human can verify the claim.

RAG can fail when...

- The source is wrong, stale, incomplete, or unauthorized.
- Chunk boundaries separate the fact from its meaning.
- The search misses the right passage.
- The model ignores or misreads retrieved evidence.
- The question requires calculation or action, not document lookup.

Production RAG adds...

- Metadata and access-control filters
- Keyword plus semantic search
- Reranking and result thresholds
- Citations and abstention behavior
- Freshness checks and ingestion logs
- Evaluation questions with known answers

RAG reduces unsupported guessing. It does not eliminate hallucinations, and it cannot repair a bad source of truth.

Dragon's test

Ask four questions before trusting a RAG answer: **What source was searched? What exact evidence was found? Is it current and authorized? Can I reproduce the answer without trusting the prose?**

Chunk size and vector databases are design choices, not universal rules. Production retrieval should be tuned and evaluated against the actual documents and questions.

Agents act. MCP connects. They are not the same thing.

An agent is a model operating in a loop: understand the goal, make a plan, use a tool, observe the result, adjust, and continue until finished or blocked. MCP is one standardized way for an AI application to discover and use external tools, resources, and prompt templates.

CONCEPT	WHAT IT ACTUALLY DOES	WHAT IT DOES NOT GUARANTEE
LLM	Interprets, reasons, generates language, and makes probabilistic choices.	Current facts, persistent memory, permission, or correctness.
Agent	Lets a model direct multiple steps and tool calls toward a goal.	Good judgment, safe actions, completion, or recovery.
Tool	Performs a bounded operation: query a layer, read a file, send a request, write a record.	That the model selected the right tool or supplied valid arguments.
Memory	Stores selected state beyond the immediate prompt or session.	Automatic truth, good routing, freshness, or privacy.
MCP	Standardizes context exchange between an AI host and servers exposing tools, resources, or prompts.	Reasoning, orchestration, RAG quality, security policy, or successful action.

Use a workflow when...

- The steps are known in advance.
- Consistency matters more than flexibility.
- Failures must be easy to reproduce.
- The action is consequential or regulated.

Example: validate a monthly workbook, join counties by FIPS, preview totals, then publish only after approval.

Use an agent when...

- The path cannot be fully predicted.
- The system must interpret vague requests.
- Different tools may be needed each time.
- It can safely stop, explain, and ask.

Example: investigate why a map is wrong by reading the registry, handoff, repo docs, live service, and deployment evidence.

The safe agent loop

Goal Clear outcome and limits > **Plan** Choose the next bounded step > **Act** Call an allowed tool > **Observe** Read real output and errors > **Verify** Test, cite, stop, or repeat

> *The useful autonomy is not "do anything." It is "make bounded decisions, show evidence, and stop before crossing the authority line."*

MCP is often compared with USB-C: a common connection standard. Direct APIs and SDKs remain valid, and many dependable agents use no MCP at all.

Dragon's practical architecture

Your advantage is not knowing every framework. It is knowing the humanitarian and GIS domain well enough to define truth, spot bad geography, demand provenance, and decide where automation must stop.

JOB	BEST MECHANISM	DRAGON EXAMPLE
Interpret	LLM	Understand a loosely worded operational question and explain the result clearly.
Know	Retrieval / APIs	Read the canonical layer catalog, project handoff, current service, or approved source document.
Calculate	Deterministic code	Join Red Cross geography by FIPS, ECODE, RCODE, and DCODE; compute exact counts and totals.
Act	Bounded tools	Run validation, update a file, query ArcGIS, deploy an app, or prepare a draft.
Connect	MCP or direct API	Expose GitHub, databases, files, browsers, or ArcGIS functions through documented interfaces.
Prove	Tests + human review	Verify source totals, citations, live custom domains, access controls, and the visible UI.

Eight questions to study and reuse

- ☐ **What is the exact job?**
  One bounded outcome beats a universal assistant.
- ☐ **What is the source of truth?**
  Name the file, service, database, or registry.
- ☐ **What requires judgment?**
  Give ambiguity and language to the model.
- ☐ **What requires certainty?**
  Put rules, joins, and calculations in code.
- ☐ **What may the system access?**
  Use least privilege and protect sensitive data.
- ☐ **What may it change?**
  Require approval before costly or irreversible actions.
- ☐ **How will we know it worked?**
  Choose one operational metric and test cases.
- ☐ **Who owns it after launch?**
  Monitor failures, freshness, cost, and recovery.

Pocket glossary

GROUNDING	Constraining an answer with approved evidence relevant to the current task.
EMBEDDING	A numeric representation used to compare semantic similarity; not knowledge or proof.
CONTEXT WINDOW	The working material visible to the model during a request; finite and temporary.
GUARDRAIL	A control that limits inputs, outputs, permissions, actions, or policy violations.
EVALUATION	A repeatable test of answer quality, tool behavior, safety, latency, cost, or completion.

Primary reading

1. Anthropic, “Building Effective Agents” - workflows, agents, tools, retrieval, and simplicity.
2. Model Context Protocol, “Architecture Overview” - official scope, participants, tools, resources, and prompts.
3. Microsoft Learn, “Build Advanced RAG Systems” - why real retrieval needs more than the naive pipeline.
4. Anthropic, “Demystifying Evals for AI Agents” - testing multi-step behavior before production.

One-sentence version: use AI for interpretation, retrieval for evidence, code for certainty, tools for action, and human accountability for the outcome.