Ontology primer

Shared meaning for the Web, enterprise data, and AI

A teaching edition with worked examples and Studio exercises

Core lesson: slides 1–38
Technical appendix and reading: slides 39–44
Studio workshop and exercises: W01–W10

What participants will learn

A progression from everyday language to an AI context workflow

01 Meaning

Explain concepts, relationships, instances, and formal statements.

02 The Web

Recognize how Semantic Web standards connect data and meaning.

03 AI context

Follow an ontology through retrieval, evidence, and an LLM response.

04 Practice

Scope a useful model and evaluate whether it improves a real task.

One word can hide several business meanings

Example: “Show me the client’s exposure”

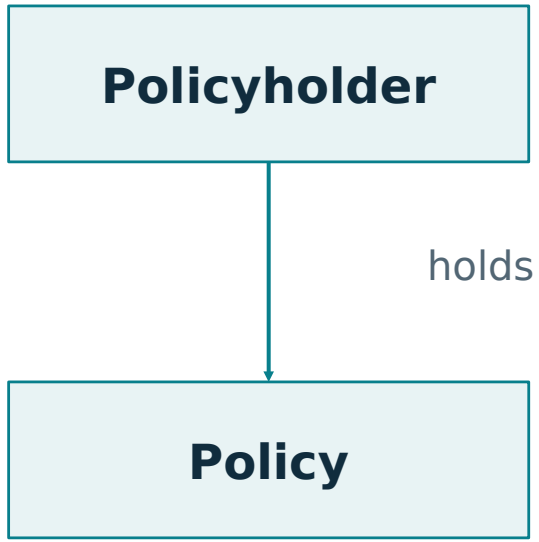
Team	Meaning of “client”	Missing context
Sales	The prospect or buying organization	Which legal entity?
Policy operations	The policyholder	Which policy and period?
Claims	The insured party or claimant	Which role in the claim?
Finance	The billing account	Which account hierarchy?

A shared term becomes useful when its identity, role, and scope are explicit

What an ontology is

A shared, explicit model of a domain and the meaning of its concepts

**An ontology describes types of things,
their relationships, and formal statements about their meaning.**



Domain example: a policyholder holds a policy.

People agree on meaning. Software can apply the model consistently.

The building blocks

A small vocabulary can describe a rich domain

Element	Plain meaning	Insurance example
Class	A kind of thing	Policy, Claim, Organization
Instance	A particular thing	Policy P-204
Property	A relationship or an attribute	hasPolicy, claimAmount
Axiom	A formal statement about meaning	AutoPolicy is a subclass of Policy
Identifier	A stable reference	An IRI for Policy P-204

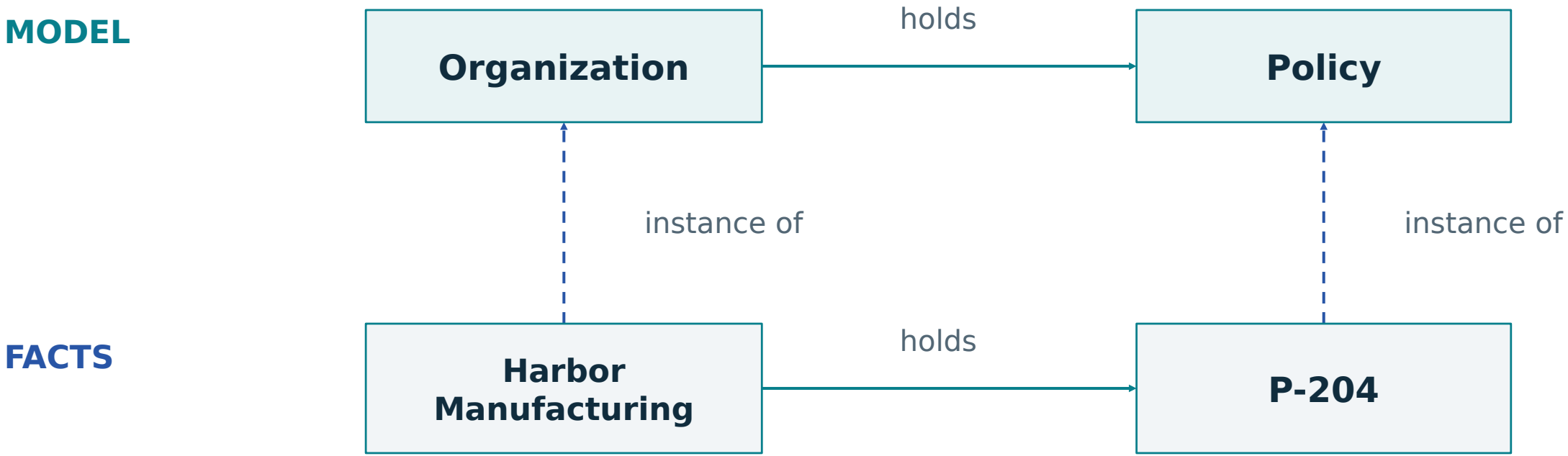
Related models serve different purposes

These artifacts can complement each other

Artifact	Primary purpose	Example
Glossary	Explain terms in language	Definition of policyholder
Taxonomy	Organize concepts by category	Policy categories
Ontology	Express meaning and relationships	Policyholder holds Policy
Database schema	Define a storage structure	policy_holder_id column
Knowledge graph	Represent connected facts	Harbor holds P-204

The ontology defines the model

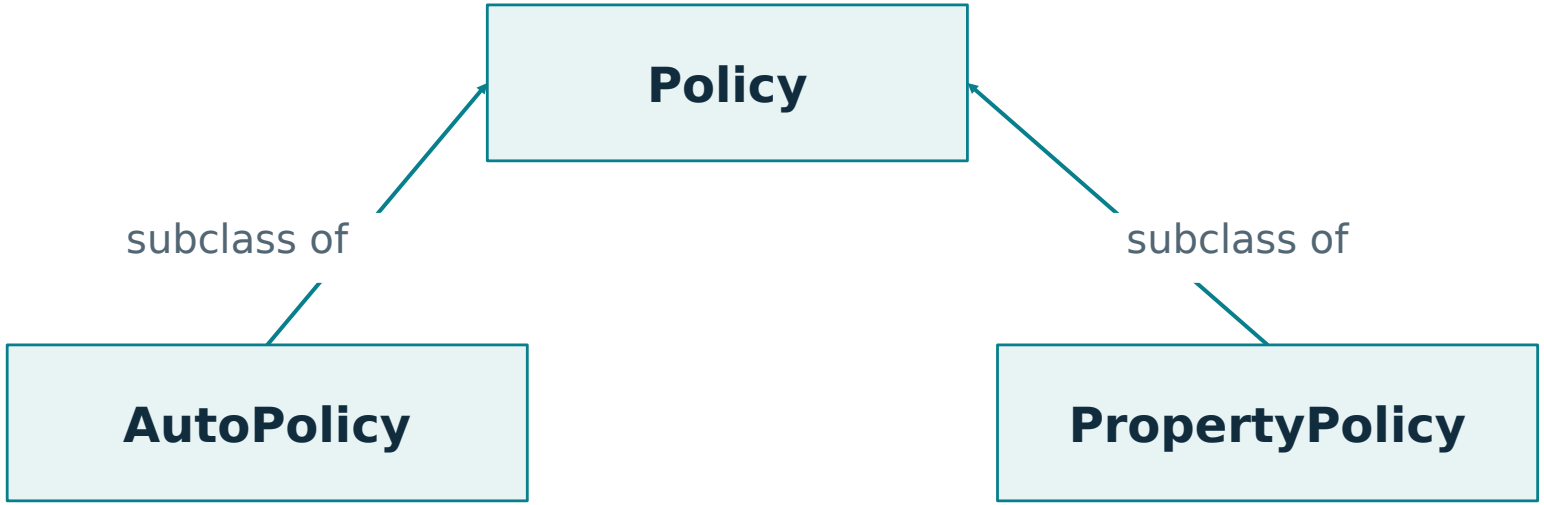
The knowledge graph records particular facts using that model



A model can stay stable while its facts change every day

Class hierarchies capture “is a”

A subclass inherits membership in its parent class

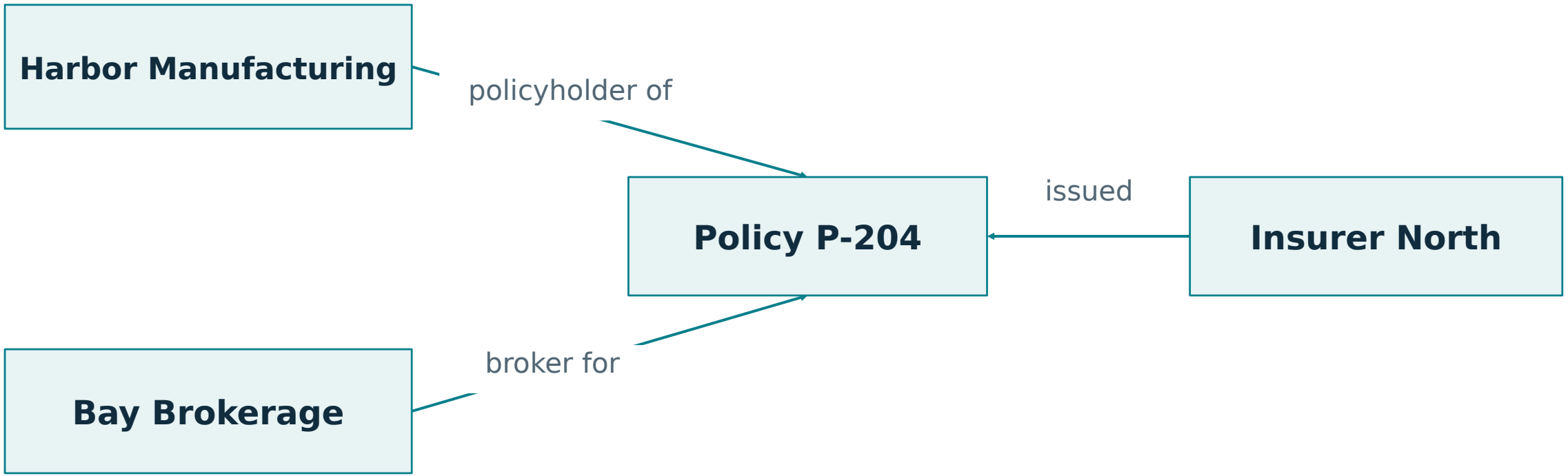


P-204 is an AutoPolicy.
A reasoner can also classify P-204 as a Policy.

“Harbor holds P-204” describes a relationship between instances

Relationships explain roles

The same organization can play different roles around a policy



For changing roles, model the role assignment and its effective period

Axioms support explicit inference

A reasoner derives consequences from the model and the facts

Model statement

Every AutoPolicy is a Policy.

Recorded fact

P-204 is an AutoPolicy.

Derived fact

P-204 is a Policy.

The inference is valid relative to the model and input facts

Inference and validation answer different questions

Missing information needs an explicit treatment

Task	Question	Example
OWL reasoning	What follows from these statements?	AutoPolicy implies Policy
Open-world interpretation	Could more facts exist elsewhere?	No recorded holder means unknown
SHACL validation	Does this graph meet a data requirement?	A Policy must record a holder
Application logic	What should the workflow do next?	Request the missing holder

An OWL restriction alone does not act like a required database field

Concept check

What does the statement describe?

Statement	Your classification
AutoPolicy is a kind of Policy	Class hierarchy
Harbor holds Policy P-204	Instance relationship
P-204 has an effective date of 1 January	Instance value
Every ingested Policy must record a holder	Data validation requirement

Discussion: which statement changes the model, and which changes the data?

The historical path

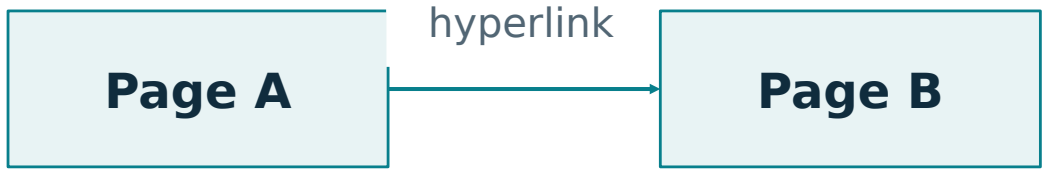
Ontologies predate the Web. The Web made shared meaning broadly exchangeable.

Period	Development	Contribution
Before the Web	Philosophy and AI knowledge representation	Conceptual models and formal meaning
1989–1991	World Wide Web at CERN	Linked documents and global identifiers
1999–2004	RDF and OWL standards	Machine-readable statements and ontologies
2006–2017	Linked Data, vocabularies, SHACL	Publication, shared terms, validation
2020–2024	RAG and graph-based retrieval research	External evidence as context for generation

The document Web and the data Web

Both use links. The link’s meaning changes.

DOCUMENT WEB



People interpret the text and decide what the linked page means.

SEMANTIC DATA



Software can interpret a named relationship using a shared vocabulary.

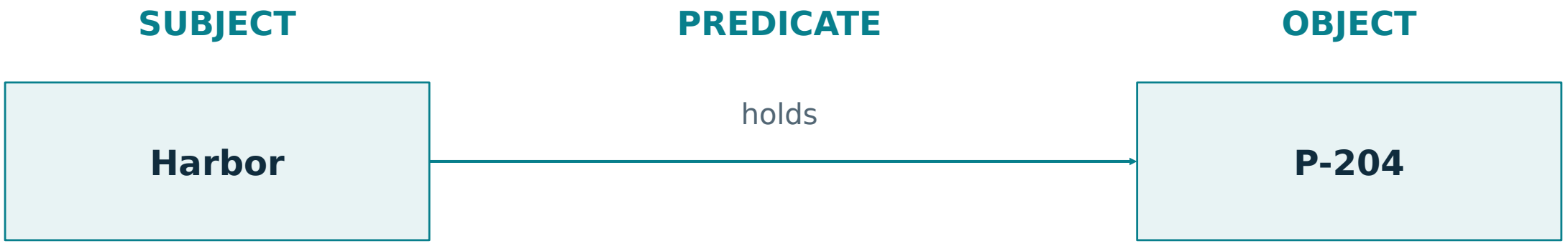
The Semantic Web toolkit

Each standard addresses a different job

Standard	Job	Plain example
RDF	Represent statements as triples	Harbor holds P-204
RDFS	Describe classes and basic relationships	AutoPolicy is a subclass of Policy
OWL	Express richer logical meaning	Two classes are disjoint
SPARQL	Query RDF data	Find Harbor’s open claims
SHACL	Validate RDF against requirements	A claim needs a claim ID

A triple is a statement

Subject, predicate, and object form a directed, labeled link



Identifiers refer to entities. Labels help people read them.
An object can also be a literal value, such as a date or amount.

Structured meaning on ordinary websites

Schema.org vocabularies and JSON-LD make web content easier to interpret

A page describes a business. Structured data identifies its type and names its properties.

```
{
 "@context": "https://schema.org",
 "@type": "Organization",
 "name": "Harbor Manufacturing",
 "url": "https://harbor.example"
}
```

Vocabulary: Schema.org
Representation: JSON-LD
Consumer: a search or other application

A vocabulary supplies terms. A serialization supplies the format.

Linked Data connects descriptions across sources

Shared identifiers and typed links support distributed meaning

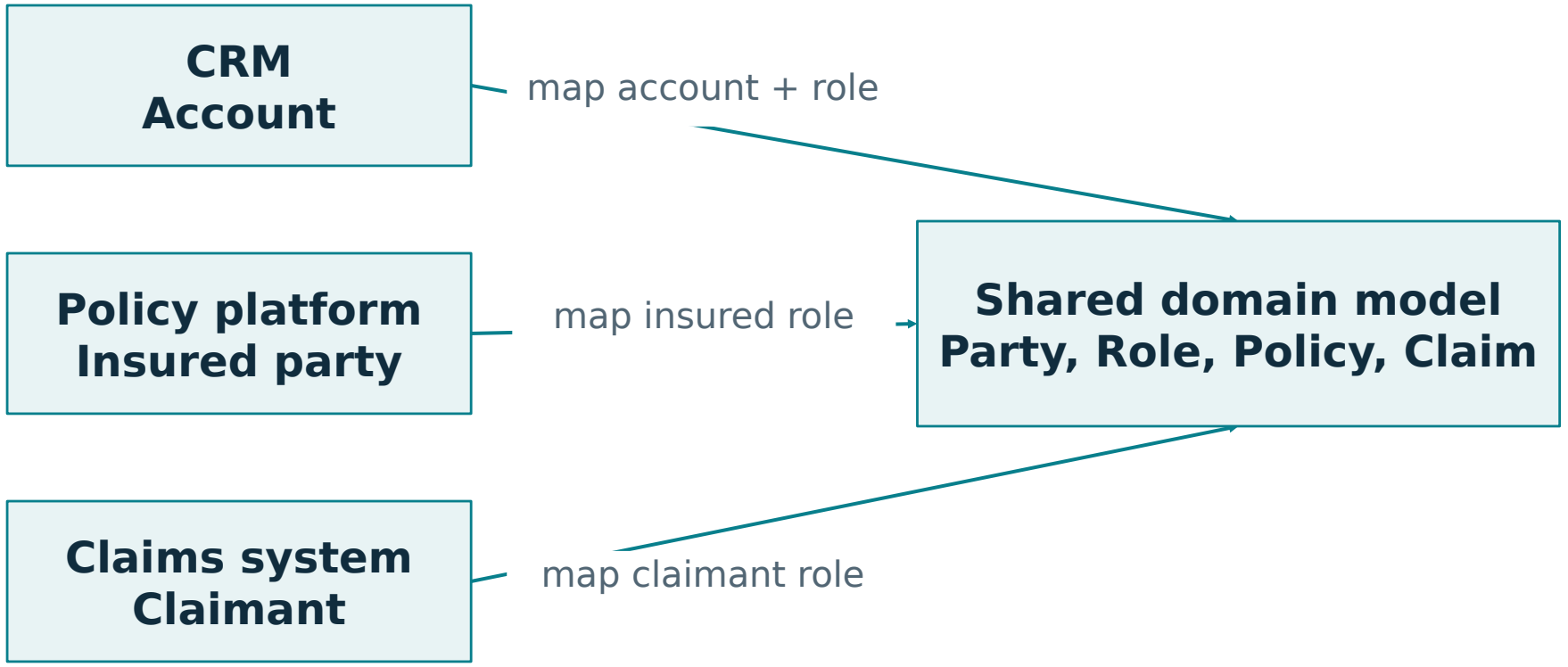


Data may remain in separate systems.
Mappings and identity checks establish which descriptions connect.

Linking records requires evidence of identity and clear relationship meaning

An enterprise ontology connects business language

Mappings translate each source into shared concepts



Source distinctions remain visible.

LLMs need task-specific evidence

Fluent language does not establish current enterprise facts

LLM capability	Additional context needed
Interprets a natural-language question	The organization’s meaning of client and exposure
Produces a coherent explanation	Current facts and relevant source passages
Suggests a plausible relationship	Verified entity IDs and approved relationships
Proposes a tool call	Allowed operations and validated parameters

An ontology can guide interpretation and retrieval. Evidence supports the answer.

What “context” means in an LLM workflow

The model receives a selected set of inputs for a specific task

Instructions

The task, expected output, and rules for handling uncertainty.

Domain meaning

Relevant definitions, roles, identifiers, and relationship descriptions.

Evidence

Current records, retrieved passages, and tool results with sources.

Conversation state

The user’s scope, date, and earlier clarifications.

Retrieval-augmented generation

RAG retrieves external material before generating an answer

Sources stay outside the model weights and can change over time.



RAG can improve grounding. Retrieval and generation still need evaluation.

Vector, graph, and hybrid retrieval

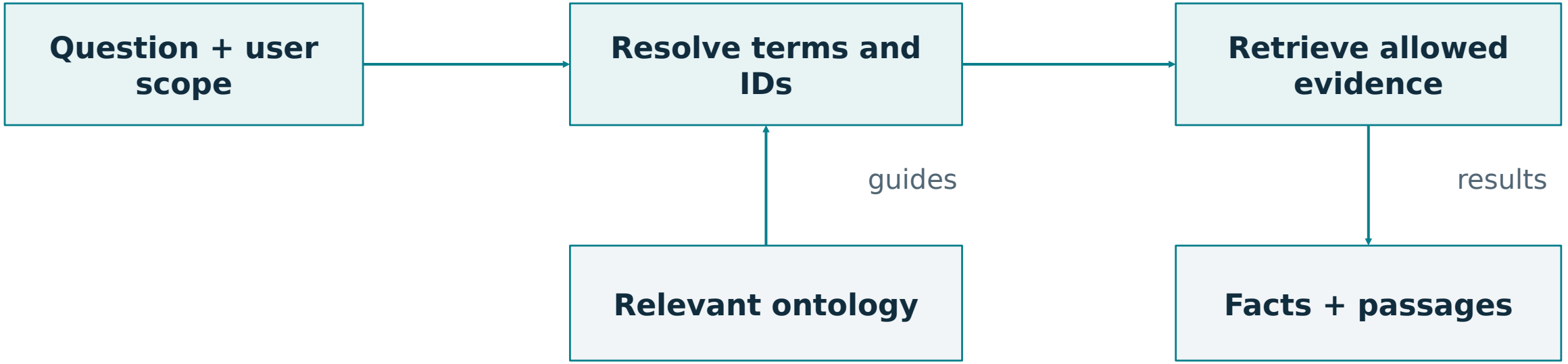
The question determines which retrieval method helps

Method	Useful question	Typical weakness
Vector search	Which passages resemble this issue?	Similarity may miss exact roles or joins
Graph query	Which claims belong to this policyholder?	The graph may lack a needed fact
Keyword or SQL	Find an ID or calculate a measure	Requires explicit search or query logic
Hybrid retrieval	Find related entities and supporting text	Adds orchestration and evaluation work

Explicit relationships and language similarity provide different kinds of evidence

An ontology-aware context architecture

Illustrative design pattern with controlled retrieval



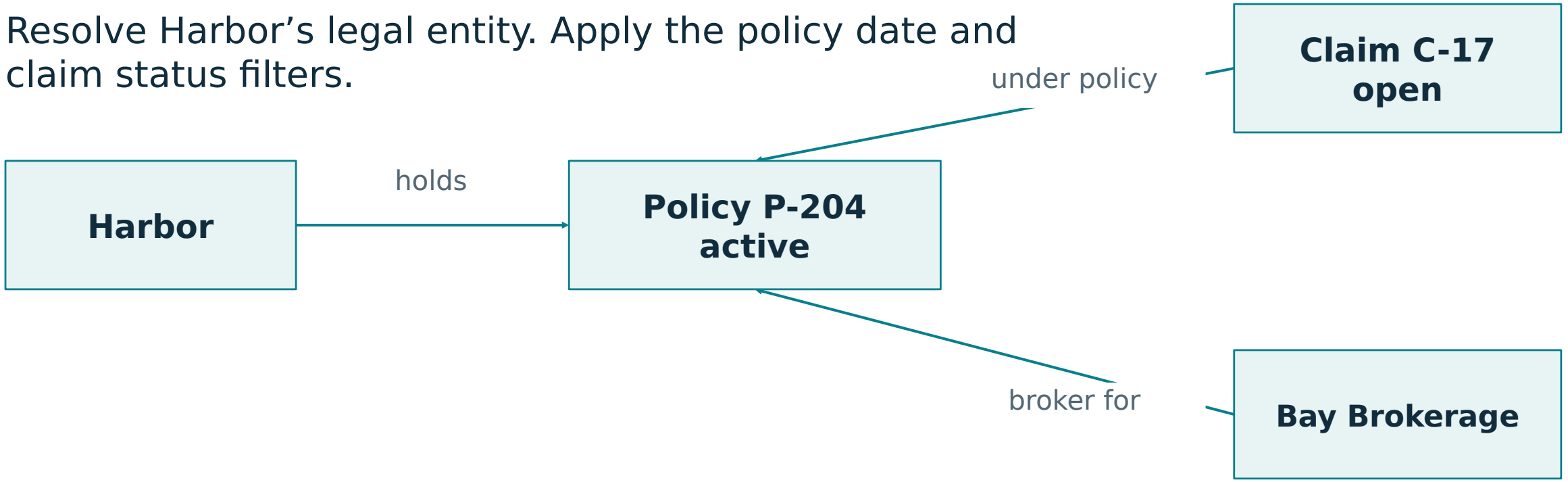
Selected evidence forms the context package.
The LLM explains it, and the application checks the response.

Authorization applies before retrieval and throughout downstream processing

Worked example: a question with several joins

“Which open claims relate to Harbor’s active policies, and who is the broker?”

Resolve Harbor’s legal entity. Apply the policy date and claim status filters.



The graph exposes the join path. The sources establish the facts.

The context package the LLM receives

A compact selection with meaning, facts, and evidence references

Context element	Illustrative content
Scope	Harbor entity ORG-18, as of 10 October 2026
Meaning	“Client” means policyholder for this question
Verified facts	P-204 active, C-17 open, Bay is effective broker
Evidence references	Policy record R1, claim record R2, broker assignment R3
Response instruction	Use supplied evidence, cite references, state missing data

An answer grounded in the selected evidence

The response distinguishes supported facts from missing information

As of 10 October 2026, Harbor has one open claim in the supplied records: C-17 under active policy P-204. Bay Brokerage is the effective broker. [R1, R2, R3]

This answer covers the supplied source systems.
Claim amounts were not included in the retrieved evidence.

A citation must support the claim beside it and respect the user’s permissions

GraphRAG covers several design approaches

A graph can help retrieve local facts or summarize a large corpus

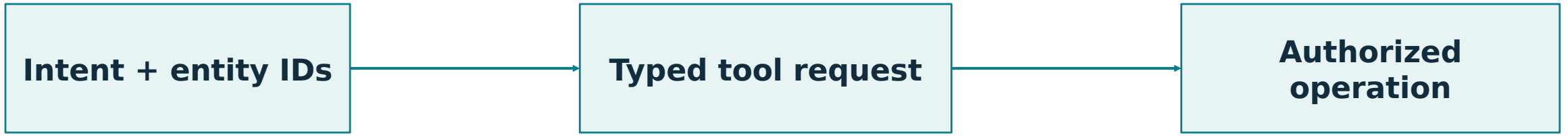
Approach	How it supplies context	Ontology role
Curated domain graph	Queries governed entities and relationships	Explicit model guides the query
Extracted document graph	Links entities found in source text	Optional model constrains extraction
Microsoft GraphRAG global search	Uses community summaries for broad questions	Formal OWL ontology is optional
Hybrid local retrieval	Combines entity neighborhoods and passages	Definitions can resolve terms and roles

GraphRAG is a retrieval approach. Graph presence alone does not ensure quality.

Agents can use meaning to choose tools

Execution still depends on application controls

Ontology context identifies the objects and relationships.



Example: retrieveClaims(policyId="P-204", status="open")

Tool schemas check structure. Authorization and workflow rules control the action.

Context needs provenance, time, and access

The same fact can have different validity and visibility

Dimension	Question	Implementation example
Provenance	Where did this statement come from?	Source record and extraction method
Time	When was it true and when observed?	Effective interval and ingestion time
Conflict	Do sources disagree?	Keep both assertions and authority rules
Access	May this user see this evidence?	Filter facts, passages, caches, and outputs

A semantic model expresses metadata. The application enforces access.

Business uses beyond a chatbot

The same shared meaning can support several applications

Use	Ontology contribution	Other components needed
Customer or party view	Distinguishes entities and roles	Identity matching and source integration
Knowledge discovery	Links topics, experts, and documents	Search and document indexing
Impact analysis	Connects assets and dependencies	Reliable lineage and current metadata
Analytics	Aligns concepts behind a measure	Metric definitions and calculation logic
AI assistance	Resolves terms and evidence paths	Retrieval, permissions, and evaluation

Building a useful ontology

A focused loop keeps the model tied to a task

Step	Work product
1 Scope a question	Clear domain boundary and intended users
2 Reuse and define concepts	Small vocabulary with definitions and examples
3 Model relationships	Class structure, roles, identity, and time
4 Connect real sources	Mappings, instance data, and validation
5 Test and revise	Reference questions, defects, and versioned changes

The first release can answer a narrow set of important questions

Competency questions define the scope

The model should support questions that people actually need answered

Question

Which open claims belong to Harbor’s active policies on a given date?

Required concepts

Organization, policyholder role, Policy, Claim, BrokerAssignment.

Required relationships

Holds policy, claim under policy, broker assigned to policy.

Required evidence

Entity identity, statuses, effective periods, and source references.

A model needs owners and a change process

Definitions become a product that teams maintain

Responsibility	Accountability
Domain owner	Approves meaning and resolves business ambiguity
Ontology steward	Maintains terms, examples, and model versions
Data and platform teams	Maintain mappings, IDs, validation, and runtime services
AI application owner	Tests task performance and context construction
Security and risk partners	Define access and review sensitive uses

Evaluation proves whether the model helps

Compare the same questions across retrieval approaches

Layer	Useful checks
Model and mappings	Definitions, role accuracy, identity matches, validation failures
Retrieval	Required evidence found, correct join path, permission compliance
Answer	Factual correctness, citation support, appropriate uncertainty
Operations	Latency, cost per task, freshness, maintenance effort

Compare a baseline with ontology-aware retrieval using a reviewed test set

Common failure modes

Each problem has a concrete repair

Problem	Likely consequence	Repair
Overly broad scope	Slow delivery and contested definitions	Limit the first competency questions
Ambiguous roles or IDs	Wrong entity or relationship joins	Define roles and verify identity
Unreviewed extraction	Plausible but unsupported graph facts	Keep evidence and review key facts
Stale facts or mappings	Wrong answer under current conditions	Track freshness and effective dates
Too much prompt context	Noise, higher cost, missed evidence	Select and test relevant subsets

Workshop: model one question in 10 minutes

Use your own domain or the fictional insurer

Minutes 0–2 Question

Choose one question. Agree on the user, date, and intended decision.

Minutes 2–5 Model

Name the concepts. Draw labeled relationships and one instance example.

Minutes 5–8 Evidence

Identify source records, IDs, missing facts, and permissions.

Minutes 8–10 Test

Describe the context package and what a supported answer would say.

The ideas to remember

Shared meaning becomes useful when it connects to evidence and a task

Ontology
Defines concepts, relationships, and statements about meaning.

Knowledge graph
Uses the model to connect particular facts.

The Web
Provides shared identifiers, vocabularies, and exchange standards.

AI context
Selects relevant meaning and evidence for the current question.

Glossary

APPENDIX Terms used in the core lesson

Term	Meaning
Class / instance	A kind of thing / a particular member of that kind
Axiom / inference	A formal statement / a consequence derived from statements
IRI / triple	A resource identifier / subject-predicate-object statement
Embedding	A numeric representation used for learned similarity
RAG / GraphRAG	Retrieval before generation / graph-assisted retrieval
Provenance	Information about the origin and production of a statement

RDF data and a SPARQL query

APPENDIX A minimal executable-style example

RDF data in Turtle

```
@prefix ex: <https://example.org/> .

ex:Harbor ex:holds ex:P204 .
ex:P204 ex:status "active" .
ex:C17 ex:underPolicy ex:P204 ;
      ex:status "open" .
```

SPARQL query

```
PREFIX ex: <https://example.org/>
SELECT ?claim WHERE {
  ex:Harbor ex:holds ?policy .
  ?policy ex:status "active" .
  ?claim ex:underPolicy ?policy ;
        ex:status "open" .
}
```

Result: ex:C17. A production query also needs date and access scope.

OWL inference and SHACL validation

APPENDIX Distinct formal statements for distinct tasks

OWL subclass axiom

```
ex:AutoPolicy
  rdfs:subClassOf ex:Policy .

ex:P204 a ex:AutoPolicy .
```

Inference: P204 is also a Policy.

SHACL data requirement

```
ex:PolicyShape a sh:NodeShape ;
  sh:targetClass ex:Policy ;
  sh:property [
    sh:path ex:hasHolder ;
    sh:minCount 1
  ] .
```

The shape requires a recorded holder for each targeted Policy

Choosing the right level of formality

APPENDIX The requirements determine the modeling investment

Requirement	Reasonable starting point
Explain terms consistently	Glossary with owners and examples
Organize and label content	Taxonomy or SKOS concept scheme
Integrate entities and relationships	Ontology, mappings, and a populated graph
Derive logical classifications	OWL and an appropriate reasoning profile
Check record completeness	SHACL or equivalent data contracts
Improve an LLM task	Relevant definitions plus tested retrieval and evidence

Reading: ontology and Web foundations

APPENDIX Primary sources for the lesson

Gruber, A Translation Approach to Portable Ontology Specifications (1993)

Noy and McGuinness, Ontology Development 101 (2001)

CERN, The birth of the Web

W3C, RDF 1.1 Primer

W3C, RDF Schema 1.1

W3C, OWL 2 Primer, Second Edition

Tim Berners-Lee, Linked Data design note

Schema.org, Getting Started

W3C, JSON-LD 1.1

Reading: validation, queries, and AI context

APPENDIX Standards, research papers, and implementation guidance

W3C, SPARQL 1.1 Query Language

W3C, Shapes Constraint Language

W3C, SKOS Primer

W3C, PROV-O

Lewis et al., Retrieval-Augmented Generation (2020)

Edge et al., From Local to Global: A Graph RAG Approach (2024)

Microsoft GraphRAG, Local Search

Liu et al., Lost in the Middle (2023)

Anthropic, Effective context engineering for AI agents (2025)

Studio practice lab

A 45–60 minute workshop after the foundation lesson

Build a small model for one question, using material you can explain and are entitled to use.

By the end	Evidence you can show
Scope meaning	One question, its intended user, date and exclusions
Review proposals	Definitions and relationships tied to source passages
Test an answer	The same question and evidence, scored against agreed criteria
Hand off safely	A selected context pack and a separate full backup

Teaching boundary: Studio is a Beta utility for business meaning and AI context. It is not a formal OWL reasoner, SHACL engine or enterprise approval system.

Choose a question before modeling

Financial services example: “What is our exposure to Harbor?”

Question to settle	Why it changes the model
Which Harbor?	Legal entity ORG-18, rather than every similarly named account
Which exposure?	A defined measure, not an interchangeable label for credit or market risk
As of when?	An agreed date and applicable hierarchy
For which decision?	The intended use determines the evidence and review needed

Participant task: rewrite the question so two people would select the same population. Name one fact the ontology cannot supply by itself.

Choose a bounded knowledge collection

Authorized source material supports proposals; it does not approve them

Source	What to record
Practice note or policy	Title, version, date, relevant passage and permitted use
PDF, DOCX, text, Markdown, CSV or HTML	Check extracted text; scanned PDFs may need text recognition first
Selected public page	Review its URL, retrieved content and currency
Conflicting definitions	Keep both source references and identify who can resolve the conflict

Participant task: choose two short, non-sensitive passages. State the scope each supports. Do not upload customer records, credentials or regulated data.

Review a proposed definition

An AI draft is a suggestion to check against the source

Review field	Example
Term and scope	Active policy: effective on the requested date and not ended under the agreed rule
Source support	Policy glossary v3, section 2; record the actual supporting passage
Example and counterexample	Effective yesterday / starts next month
Status and reviewer	Keep draft until reviewed; a typed name records a user statement

Participant task: find an unsupported clause in a proposal and remove it or request evidence. A fluent definition is not proof that its source supports it.

Model kinds, roles and changing facts

Reuse the distinction from the foundation slides

Statement	How to treat it
Organization is a kind of Party	Class hierarchy
Harbor is an Organization	A particular entity and its type
Harbor is policyholder of P-204	A role in a relationship
Bay is broker for P-204 during an interval	A time-bounded assignment with supporting records

Participant task: explain why “broker” should not become a permanent identity for every organization that has ever held that role. Keep valid time separate from when a record was observed.

PE and VC practice exercise

Agree a metric convention before calculating with supplied facts

Fictional exercise convention: ARR is twelve times active, contracted monthly recurring fees at the stated date. Exclude one-off services and uncontracted pipeline.

Supplied item	Monthly amount or fee
Active contracted recurring fees	$10,000 per month
One-off implementation fee	$2,000
Uncontracted pipeline	$3,000 per month

Participant task: calculate ARR under this convention. Explain the exclusions and identify the source facts needed to use it outside this exercise. This is a teaching convention, not a universal accounting definition.

Insurance practice exercise

An “as of” question needs evidence for that date

Question: which claims were open under Harbor’s active policies on 10 October?

Supplied record	Known fact
Policy P-204	Active on 10 October; Harbor is the policyholder
Claim C-17 event history	Opened 8 October; closed 11 October; under P-204
Claim C-18 current status only	Open when exported on 13 October; no earlier history

Participant task: distinguish the supported answer from the missing evidence. Do not treat a current status as proof of a historical status.

Evaluate the same question fairly

Keep the evidence fixed so the comparison is meaningful

Check	What a passing answer does
Meaning and scope	Uses the agreed entity, definition and date
Evidence	Supports each factual claim with the supplied records
Missing information	Asks, qualifies or stops instead of inventing a fact
Outcome	Meets the task criteria without a critical failure

Participant task: compare a baseline with a context-assisted answer. Keep model, question and evidence the same; record versions and date. Studio scores are user-entered assessments, not a certification or statistical study.

Decide where the information goes

Review each destination before using it

Action	Boundary
Local collection and practice work	Browser-local; include them in a full project backup
Chosen linked cloud model	Can include accepted private business-meaning excerpts
Optional OpenAI or Claude drafting	Selected evidence and your key pass through the hosted request service to the chosen provider
Context pack or hosted share	Inspect the selection and private-content filters before delivery

Beta exercise: use synthetic material. AI drafting needs a funded API key and confirmation of destination and possible charge; a personal chat subscription does not fund API requests. No key is needed for the local workshop.

Finish with a reviewable handoff

Choose the level of formality the next user needs

Need	Next step
Shared business meaning	Reviewed definitions, examples, stable IDs and source references
A supported AI answer	Selected context plus current authorized facts and evaluated retrieval
Formal inference or RDF validation	Use appropriate OWL or SHACL tooling outside Studio
Recovery and reuse	Download a full backup; test restoration separately from delivery exports

Exit check: a partner can explain a term, trace its source, identify a missing fact and restore a backup. Task help: Studio user guide.