HCSS 2026 · HIGH CONFIDENCE SOFTWARE AND SYSTEMS

AI as an Enabler for Analyzing Business Logic Systems

Dr. Daniel Balasubramanian

Vanderbilt University

Institute for Software Integrated Systems

May 12, 2026

daniel.a.balasubramanian@vanderbilt.edu

Team

Dr. Daniel Balasubramanian

Symbolic Analysis

PI

Dr. Sandeep Neema

AI / ML

CO-PI

Mr. Mark Wutka

Analysis Tools

Dr. Himanshu Neema

Modeling & Simulation

CO-PI

Dr. Jason Scott

Model-Based Engineering

Mr. Ethan Owens

Testing

Objective

Develop algorithms and tools to proactively identify faults in business logic systems.

What Are Business Logic Systems?

Business logic is the **encoded reasoning** that drives business processes. The encoded process logic itself is vulnerable.

DOMINANT GLOBAL PROVIDERS

Microsoft, SAP, Oracle, Workday, Salesforce, IBM — a flawed rule can propagate errors silently at scale, undetected until failure.

Manufacturing

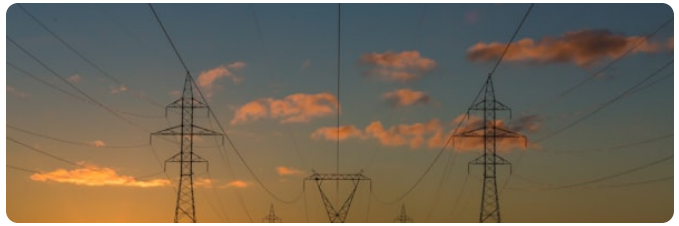
Manufacturing lines, production sequencing, accounts billing — tolerance specs in work instructions vs. MES records

E.G. Solumina MES · SAP PM · Windchill

Air Traffic Control

Sector transition rules, minimum separation distances, Notice to Air Missions (NOTAM) issuance and applicability

E.G. NOTAM · STARS · ERAM



Power & Water Infrastructure

Power grid protection, energy production dispatch, water purification and distribution process logic

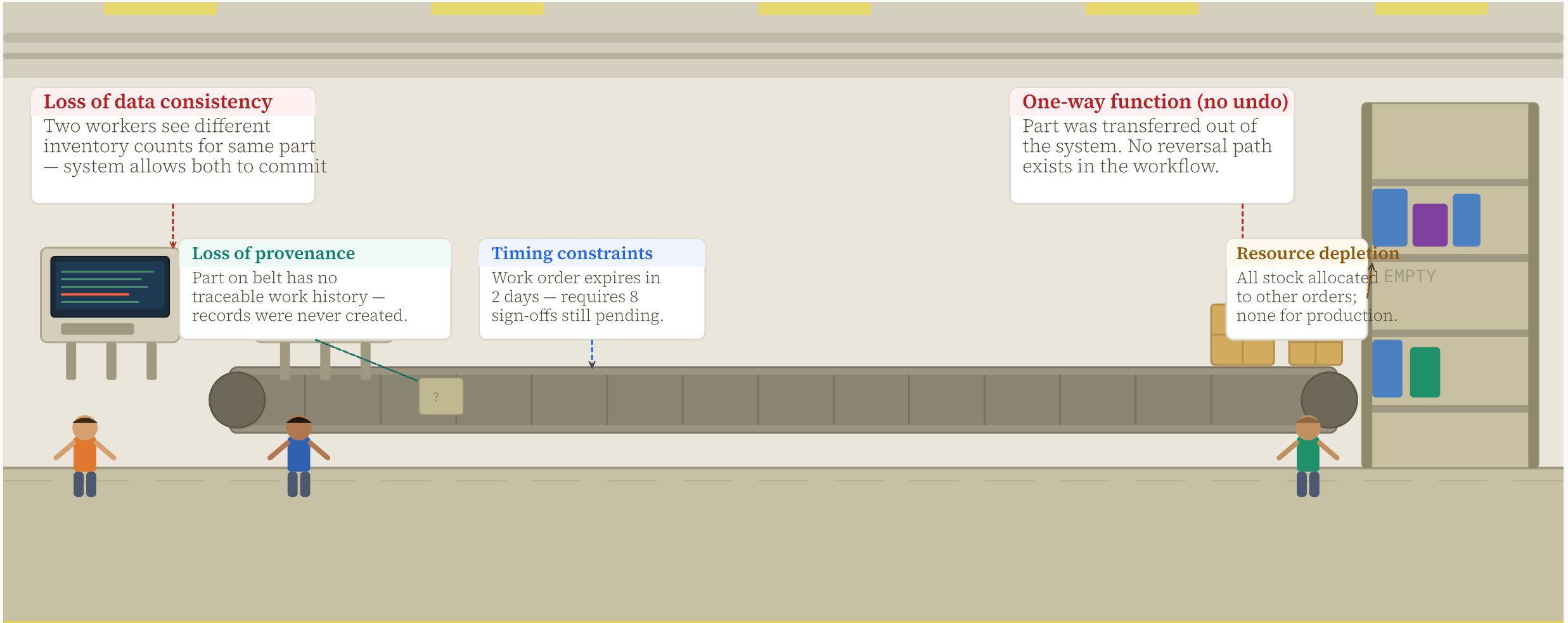
E.G. SCADA · OSIsoft PI · EPANET

Cargo & Maritime

Cargo handling workflows, shipping manifest rules, port sequencing and customs compliance logic

E.G. CargoWise · ShipNet · AWS Supply Chain

Business Logic Fault Examples



Although the BL system claims to represent the reality of the enterprise, it often does not

Manufacturing Fault Example

WORK INSTRUCTION — STEP 4.2.7 · FREE-FORM TEXT

Install the bearing assembly onto the main shaft. Verify that the outer diameter of the shaft is within **±0.005 in** of the nominal dimension per drawing 4412-B. Document completion in the work history log and notify QA for inspection hold.

DETECTED MISMATCH

SOLUMINA MES DATABASE · PART 4412-B

FIELD	VALUE	SOURCE
Part No.	4412-B	MES
Nominal OD	2.3750 in	Drawing
OD Tolerance	**±0.05 in**	MES
Torque spec	35 ft-lbs	MES
Inspection hold	Required	QA

Manufacturing Fault Example

WORK INSTRUCTION — STEP 4.2.7

Install the bea[illegible] the main shaft. Verify that the outer diameter is within ±0.005 in of the nominal per drawing 4412-B. Notify QA for inspection hold.

⚑ Stated tolerance

DETECTED MISMATCH

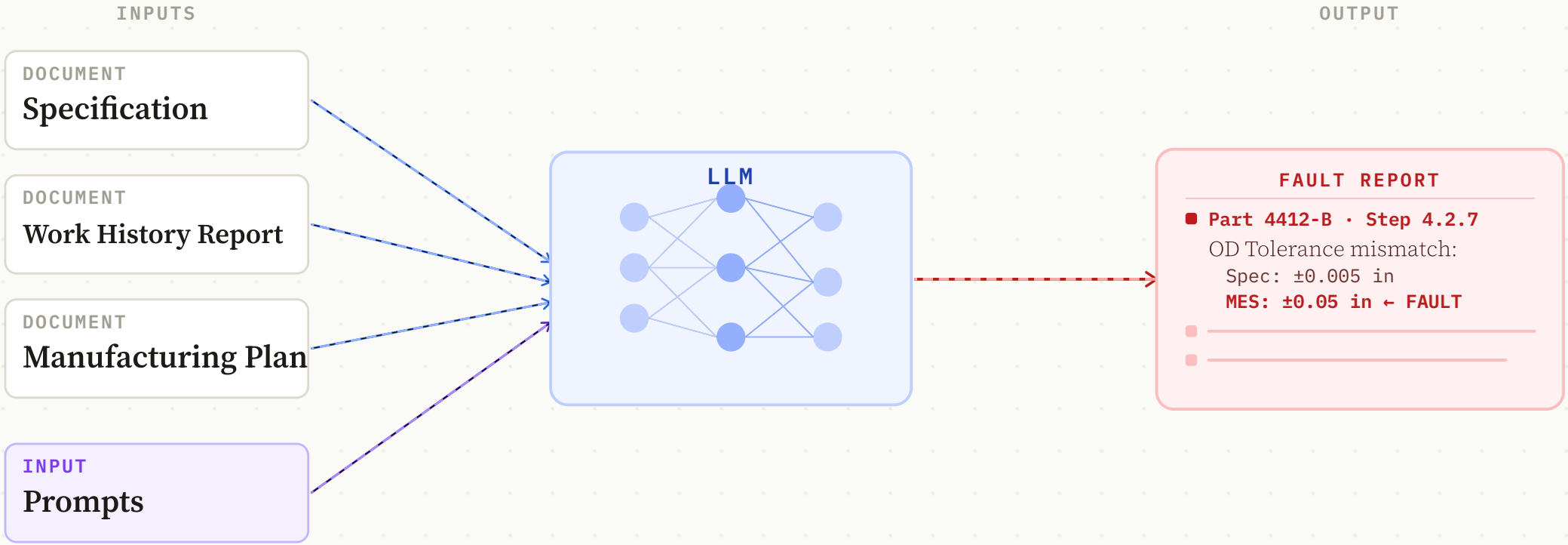
SOLUMINA MES DATABASE · PART 4412-B

FIELD	VALUE	SOURCE
Part No.	4412-B	MES
Nominal OD	2.3750 in	Drawing
OD Tolerance	±0.05 in FAULT	MES
Inspection hold	Required	QA

How can we identify these faults?

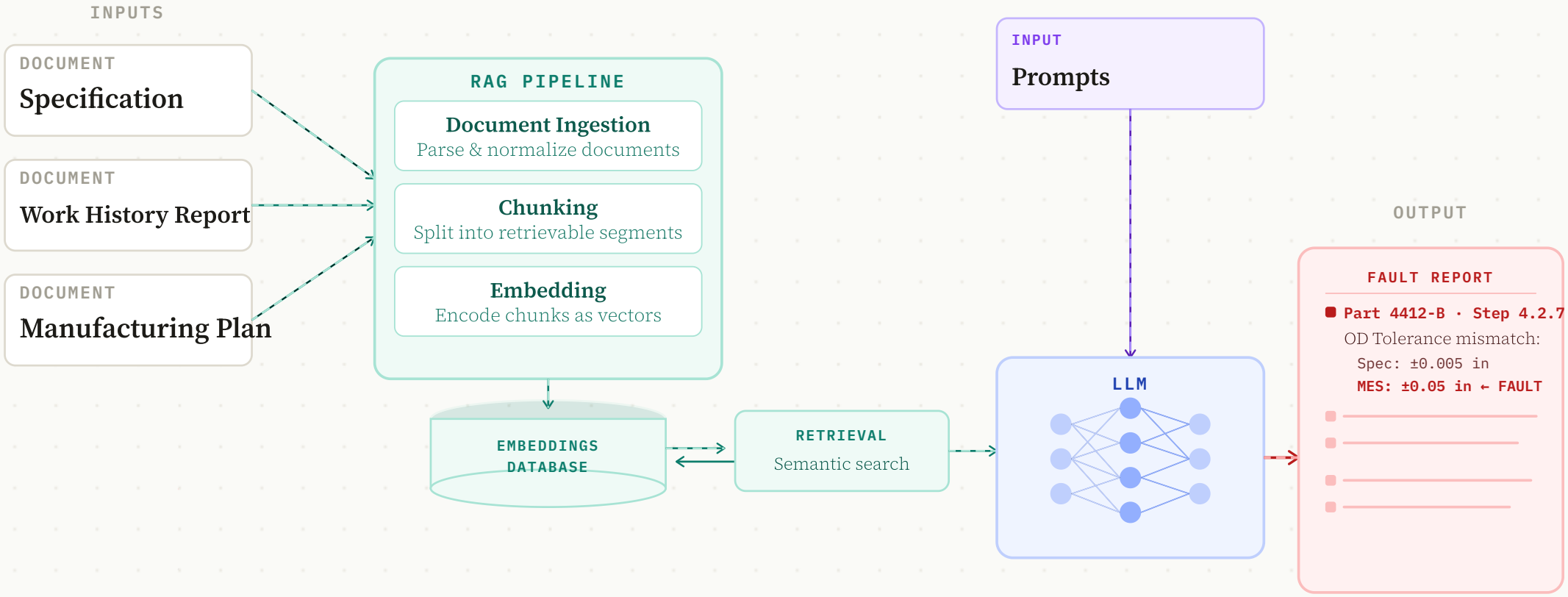
1. Identify the tolerance in the prose
2. Identify the corresponding database element
3. Compare the values

Approach 1: Use an LLM



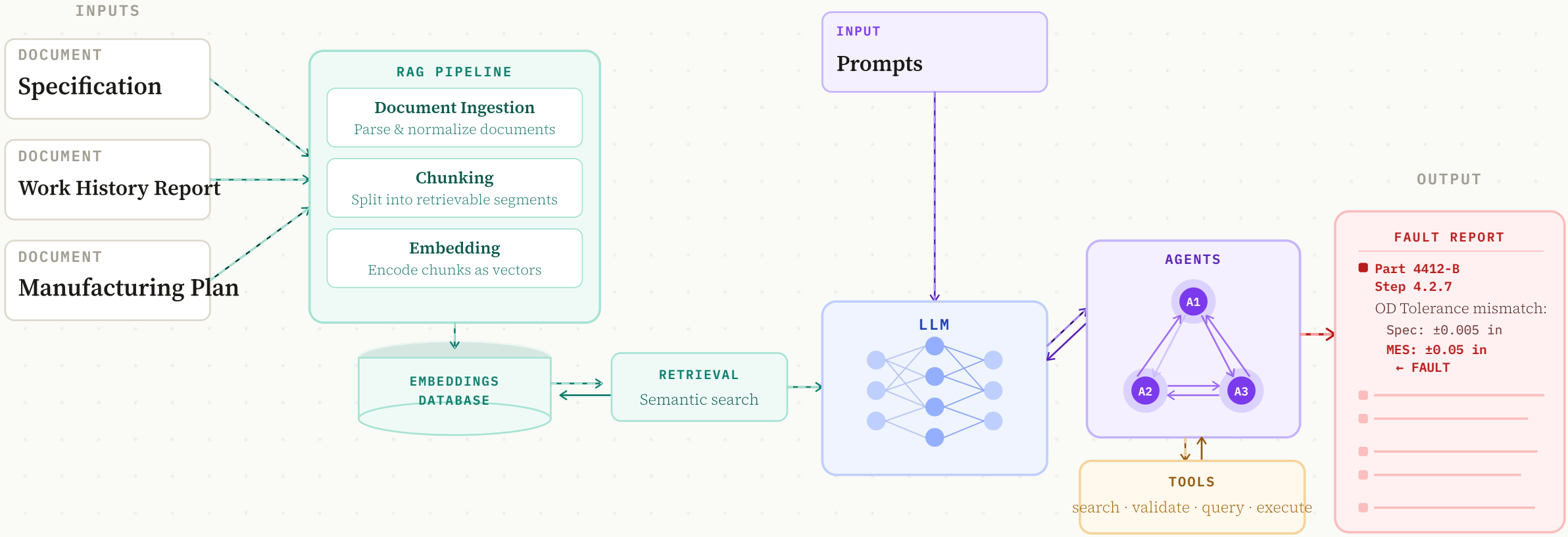
Drawback: hallucinations

Approach 2: Use RAG



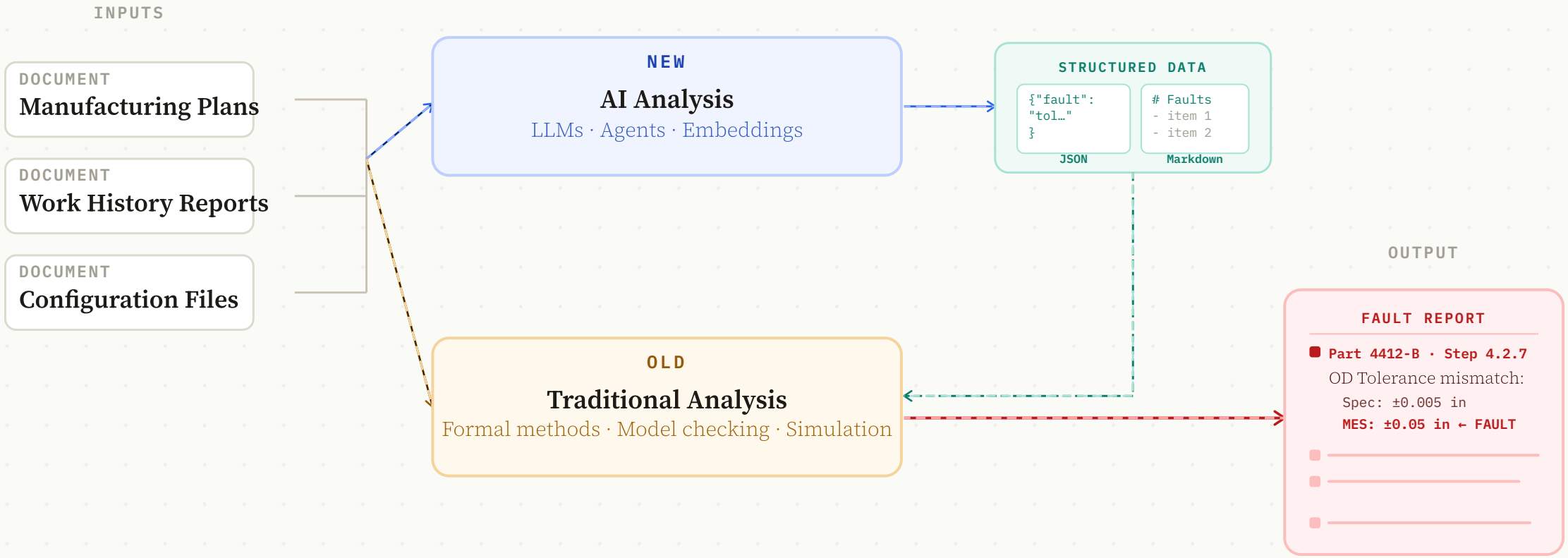
Drawback: hallucinations; tweaking RAG parameters

Approach 3: Use Agents

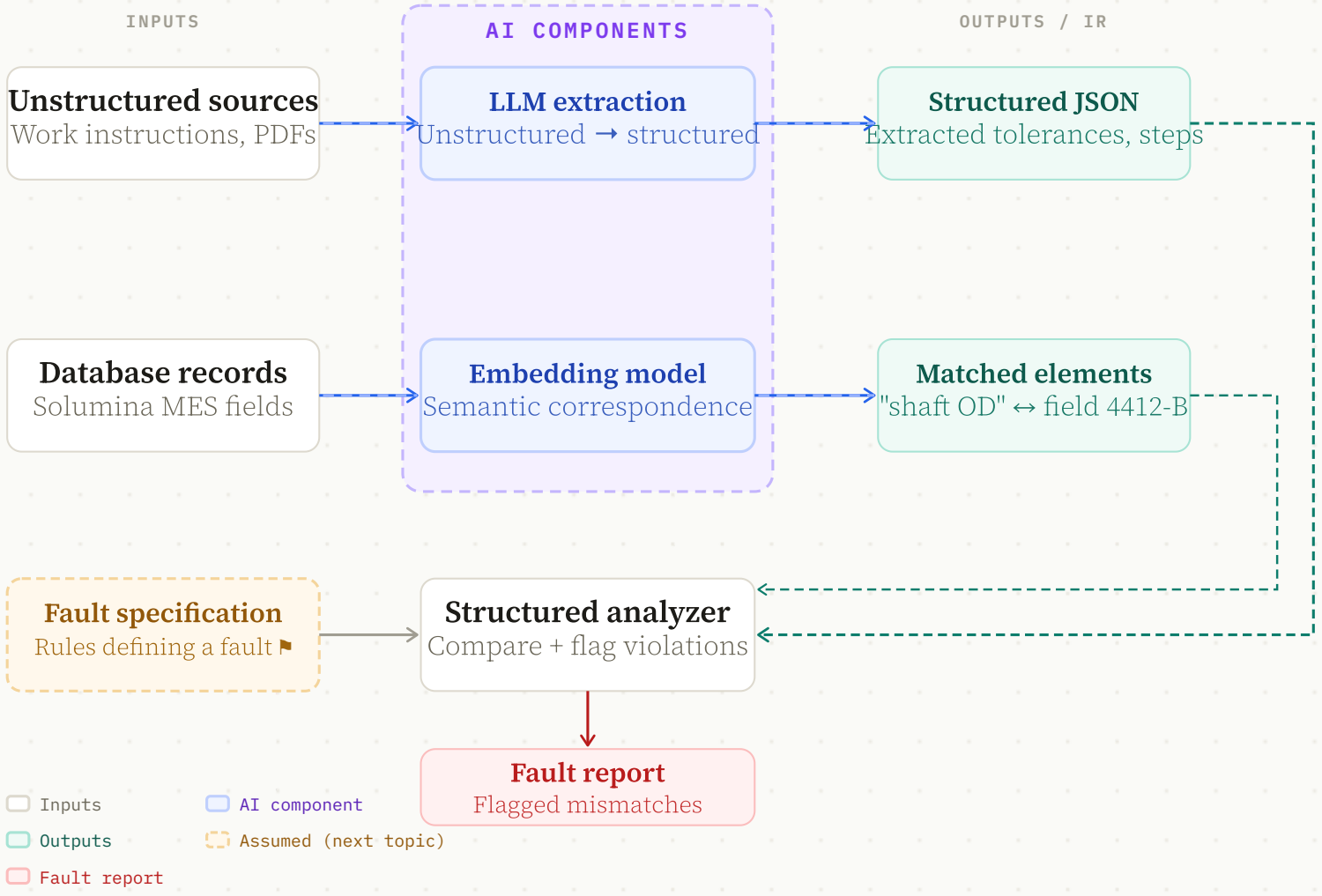


Drawback: efficiency with local models

Our Solution: Old Meets New



Where AI Does the Heavy Lifting: Targeted Extraction



THREE REQUIREMENTS FOR FAULT DETECTION

1 **Structured extraction**
Parse tolerances and constraints from free-form prose
`LLM`

2 **Element correspondence**
Match text references to the right database field
`Embedding model`

3 **Fault specification**
A rule that says *what* counts as a mismatch — rarely clean or structured in practice

Example: LLM Extraction

Unstructured text → structured JSON

INPUT — STEP 4.2.7

Install the **bearing assembly** onto the main shaft using **tool T-224**. Ensure the shaft surface is clean and free of burrs before pressing. Verify the outer diameter is within **±0.005 in** of the nominal per drawing **4412-B**. Apply thread-locking compound to all fasteners and torque to **35 ft-lbs** in a star pattern. **Steps must be completed in sequence.** Document completion in the work history log and notify QA for inspection hold.

→

LOCAL SERVER

- (•) gemma4
- ( ) gpt-oss
- ( ) llama3:8b

Constraint: sensitive data requires local LLMs only

→

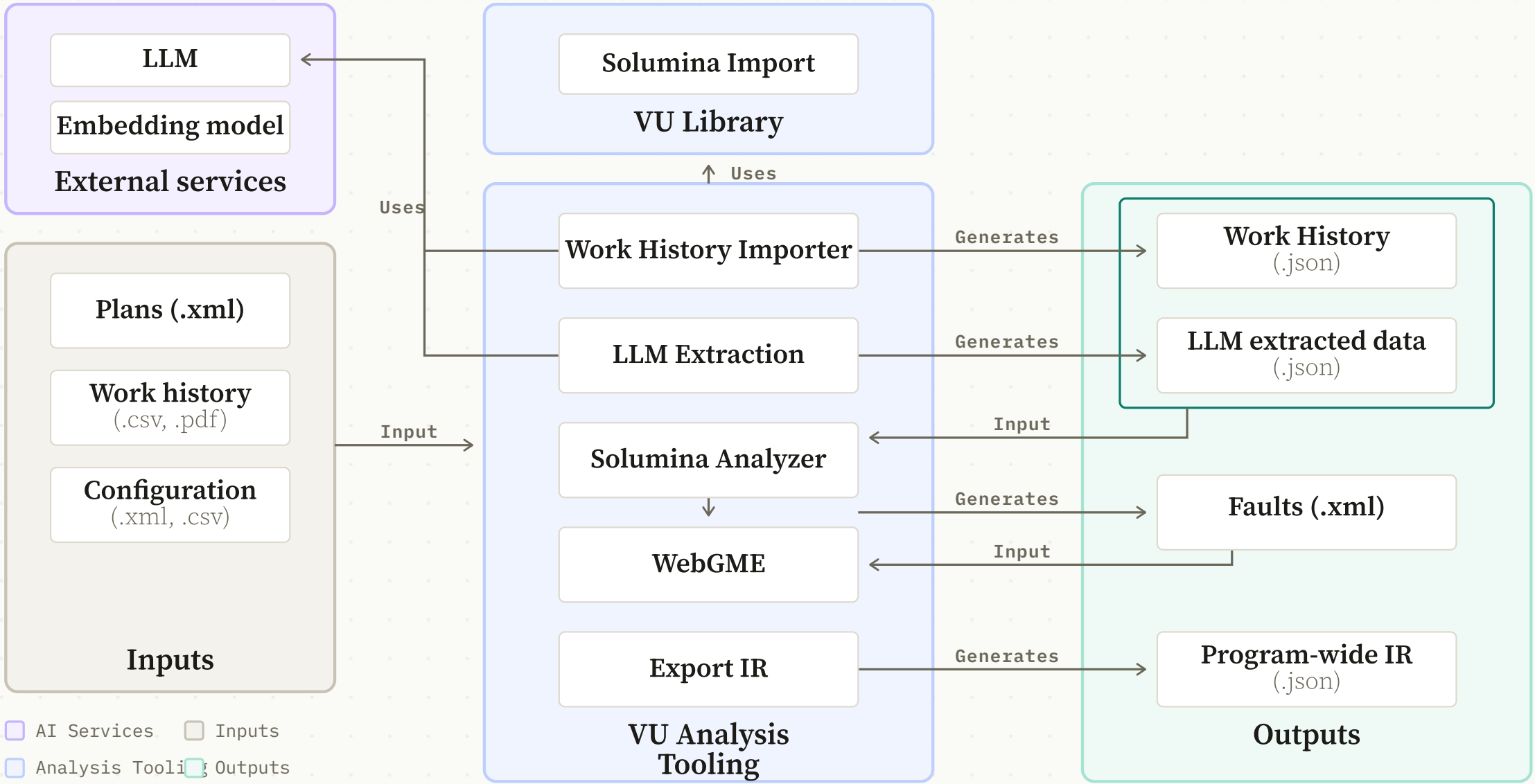
OUTPUT — JSON

```
{
  "step_id": "4.2.7",
  "part_reference": "4412-B",
  "component": "bearing assembly",
  "tool_required": "T-224",
  "tolerance": {
    "dimension": "outer_diameter",
    "value": 0.005,
    "unit": "in",
    "type": "bilateral"
  },
  "torque": { "value": 35, "unit": "ft-lbs" },
  "sequential": true,
  "inspection_hold": true,
  "buyoffs": ["QA"]
}
```

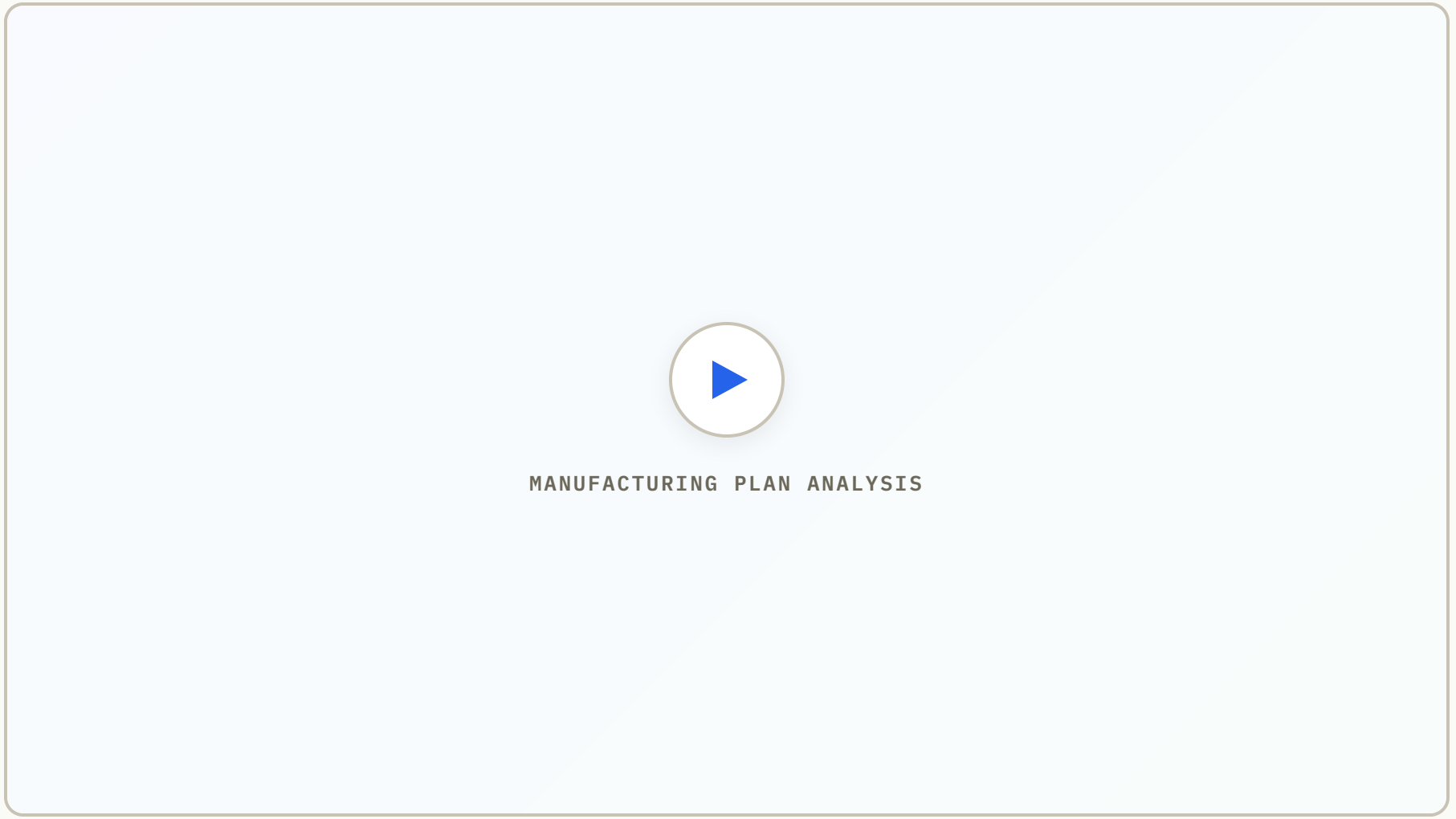
EXTRACTED FIELDS

Tolerances, tools, parts, torque specs, sequencing constraints, buyoff requirements

Structured Analysis Architecture



Demo



A STEP BACK

Where did the fault specification come from?

We've been assuming it exists.

The Specification Problem

01 — DISTRIBUTION

Spread across many documents

- Engineering drawings PDF / DXF
- Process standards PDF / Word
- Regulatory documents PDF / HTML
- MES configuration XML / CSV
- Engineering email Unstructured

02 — TEMPORALITY

Evolve over time

Regulatory updates and design revisions create version conflicts across documents.

Rev A · 2019
Drawing 4412 issued
OD tolerance: ±0.005 in

Rev B · 2021
Tolerance tightened after field failure
MES not updated for 6 months

Rev C · 2024
Major revision per AS9100 audit

03 — REPRESENTATION

Mixed modalities

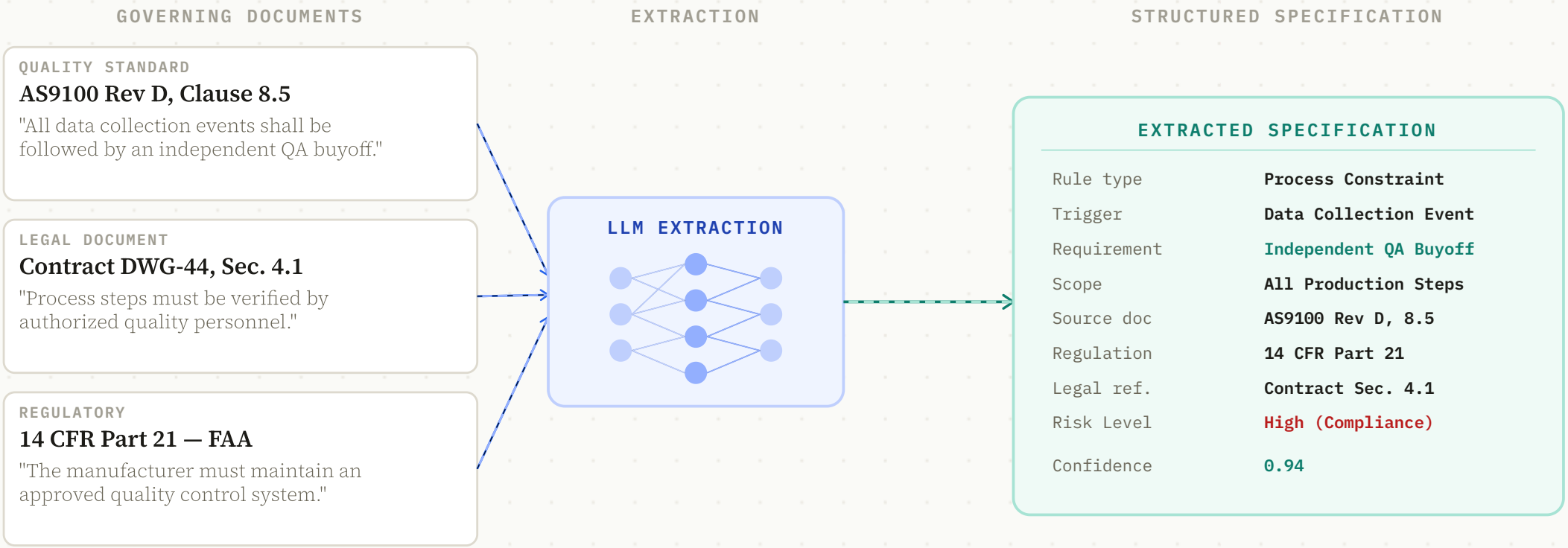
Requirements appear in many formats, including text, tables, and graphics.

TEXT (WORK INSTRUCTION)
"...verify OD is within **±0.005 in** of nominal..."

TABLE (DRAWING)
OD Tol. | `±0.005 in` | Rev C

GRAPHICS (ENGINEERING DRAWING)
⟵ Ø 2.500 **±0.005** ⟶ [GD&T callout, Rev C]

Specification Extraction: A Manufacturing Example



Challenges & Representation Choices

01 — DISTRIBUTION

Document heterogeneity

Specifications are split across incompatible formats, each requiring different parsing.

- Engineering drawings — PDF / DXF
- Process standards — PDF / Word
- Regulatory documents — PDF / HTML
- MES configuration — XML / CSV
- Engineering email — Unstructured

02 — TEMPORALITY

Version conflicts

Regulatory updates and design revisions create conflicts that are often invisible.

Rev A · 2019
Drawing 4412 issued
OD tolerance: ±0.005 in

Rev B · 2021
Tolerance tightened — MES not updated for 6 months

Rev C · 2024
Major revision per AS9100 audit

03 — REPRESENTATION

Representation choices

How a specification is encoded determines what analysis is possible.

- **Structured** — JSON / tables — queryable, LLM-friendly
- **Semi-formal** — DSLs — checkable, human-readable
- **Formal** — Logic / type systems — verifiable, provable

INFORMAL — FULLY FORMAL

Summary

01

Business logic systems contain systemic, deeply-buried faults

These systems harbor complex errors that elude conventional identification and review, creating significant risks that are often invisible until failure.

02

An "old-meets-new" approach enables systematic analysis

LLMs bridge the gap between natural language prose and structured data; when combined with traditional analysis, they enable a powerful new regime for fault detection.

03

Reliable extraction at scale is the defining challenge

Extracting specifications from vast, heterogeneous document corpora reliably enough for automated detection is a challenge.

Thank you.

Questions?

Dr. Daniel Balasubramanian

daniel.a.balasubramanian@vanderbilt.edu

Vanderbilt University · Institute for Software Integrated Systems