



SPA MDR CDR SCE

ML26

Validating AI in GxP: Test Harnesses, Guardrails, and Compliance Controls

MAR 2026



About the Presenters



Bill Qubeck

Senior Director of
Services



bill.qubeck@sycamoreinformatics.com



North Carolina
USA



<https://www.linkedin.com/in/williamqubeck/>

For more than 30 years, Bill has helped lead digital transformation in clinical analysis & reporting, driving automation, interoperability, and technology adoption to improve speed, quality, and efficiency. He is particularly focused on the role of AI, reuse, and metadata driven solutions in reshaping how the industry designs protocols, executes studies, and delivers reporting. Through next generation analytics and automation, he helps pharma companies and CROs accelerate clinical development and improve outcomes.



Heather Gorr

Associate Director -
Technology
Development



Heather.gorr@sycamoreinformatics.com



Boston
USA



<https://www.linkedin.com/in/heather-gorr-phd/>

Leads AI product strategy at Sycamore Informatics, where she drives the development of scalable AI solutions. A physicist by training, she specializes in the challenging "last mile" of data science: taking models from research notebooks to production-grade deployment. Previously, Heather spent over a decade consulting on multilingual interoperability in AI applications and near real-time deployment focusing on architect robust AI and deep learning workflows.



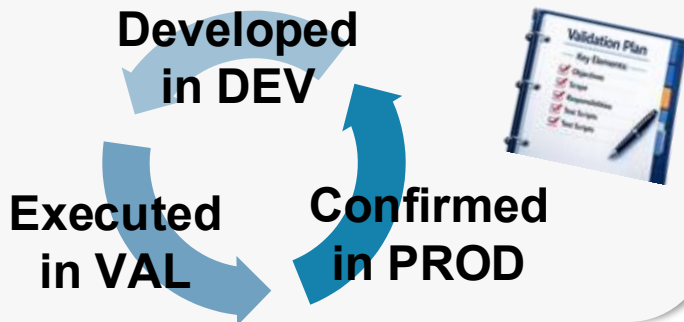
Agenda

- The AI Validation Challenge
- Validation Framework
- Technical Guardrails
- Core Principle: Test Harness
- Regulatory Requirements and Expectations
- A Key Technical Components and Considerations



GxP Validation Process

Author a Validation Plan and a set of test scripts (e.g., IQs, OQs)



Expectation



LLMs



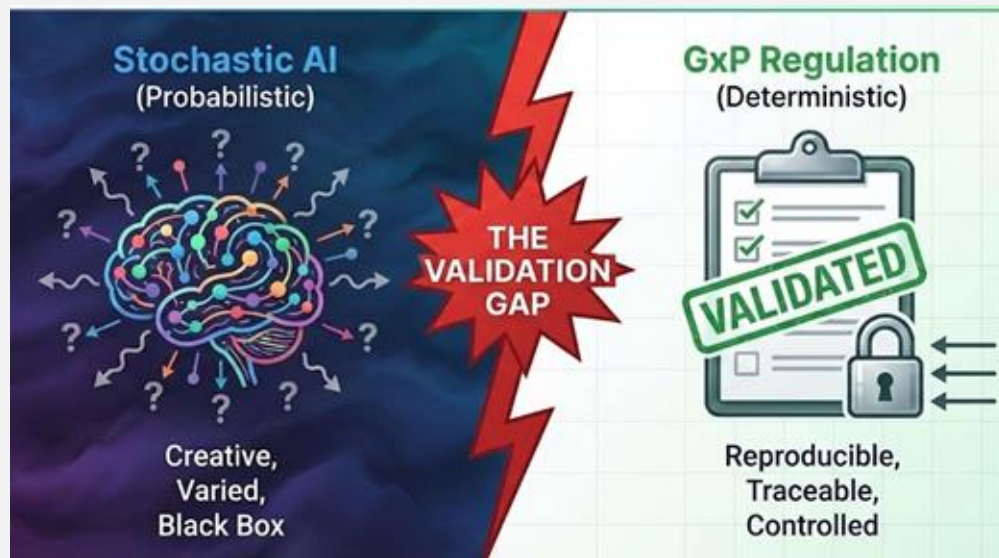
But can also be...

LLM results can vary, sometimes significantly, they are stochastic by design



The AI Validation Challenge

- **The Conflict:** Historical approaches to GxP expect determinism (exact reproducibility), but Large Language Models (LLMs) are stochastic (probabilistic)
- **The Solution:** We validate the Intended Use, not the model itself
- **The Method:** Wrapping the AI in a deterministic "Test Harness" with strict guardrails



Why Most AI Programs Fail in Regulated Submission Workflows

“Business Wants”

- Faster delivery
- Reduced manual effort
- Faster content generation
- Lower operational effort
- Improved throughput
- More consistent outputs

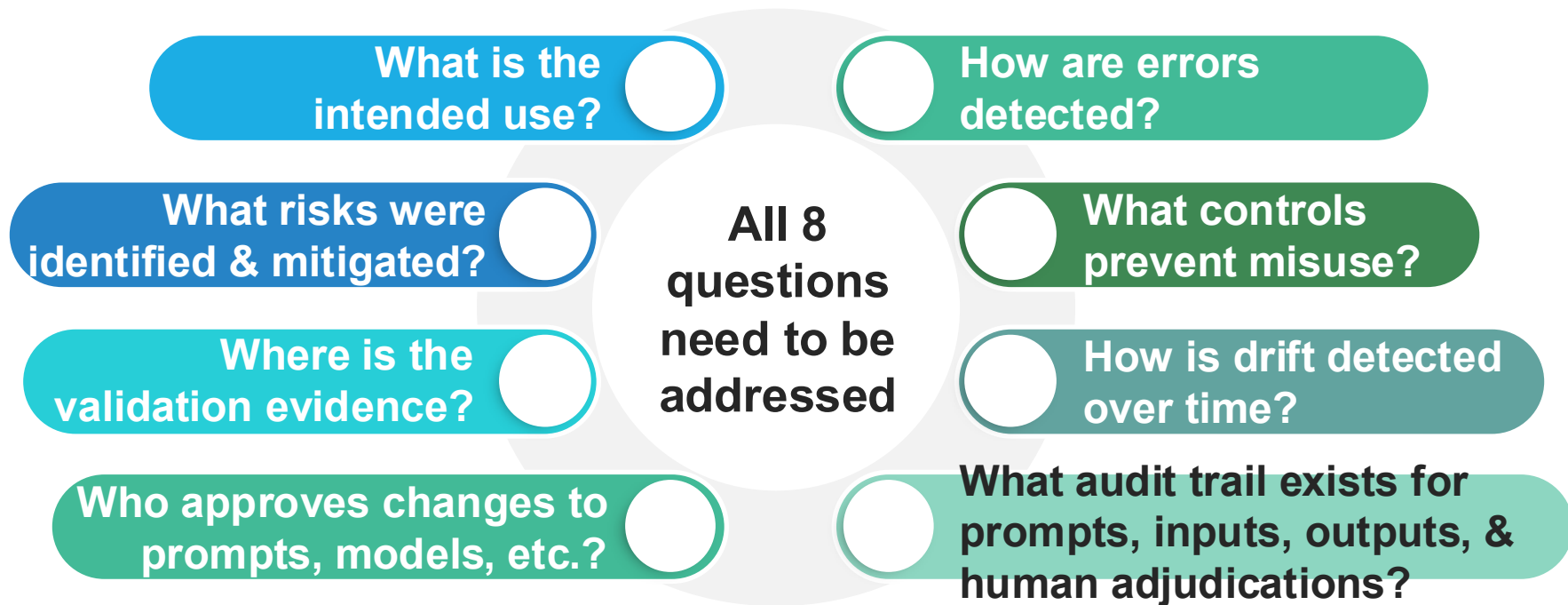
“QA Expectations”

- Inspectable system
- Narrow intended use
- Explicit responsibilities
- Enforceable controls
- Objective evidence
- Audit trail and traceability

An impressive pilot is not enough, it must be controlled, traceable, auditable, and monitored over time. The real requirement is not model performance alone, it is an inspection ready control of the end-to-end process across the lifecycle

The Inspection Readiness Gap:

Is the system inspectable, if these questions are unanswered?



The Bounded Assistant Model:

AI is an Assistant, Not an Approver or Decision Maker



What the AI is Enabled To Do

- Draft content using approved source materials
- Summarize analysis outputs, narratives, etc.
- Compare artifacts for consistency
- Flag discrepancies, omissions, and anomalies
- Support controlled quality checks across analysis and reporting assets



What the AI Should Not Do

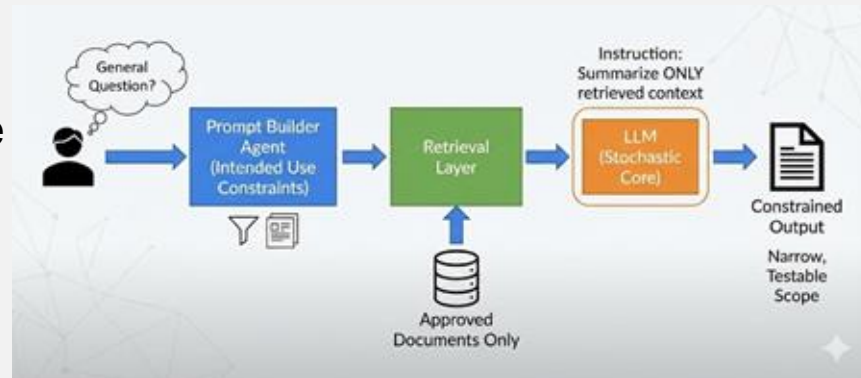
- Approve regulatory content
- Release submission artifacts
- Make decisions with scientific or regulatory authority
- Modify datasets unless explicitly permitted and separately controlled
- Bypass human review or adjudication
- Own accountability

Qualified individuals are accountable, not the system

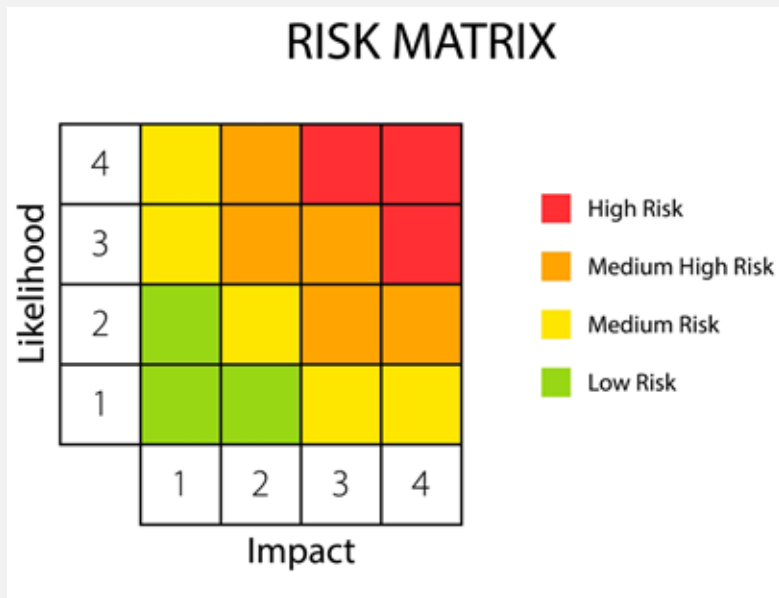


Intended Use as the Validation Boundary

- **Validation Boundary:** We define strict limits on what the AI is allowed to do.
- **The Architecture:**
 - Prompt Builder Agent: Constructs the query.
 - Retrieval Layer: Fetches approved documents.
 - LLM: Generates the response based only on retrieved data.
- **Result:** The AI is reduced from a "Creative Writer" to a "Summarization Engine."



Risk Assessment for LLMs



➤ Key Risks Identified:

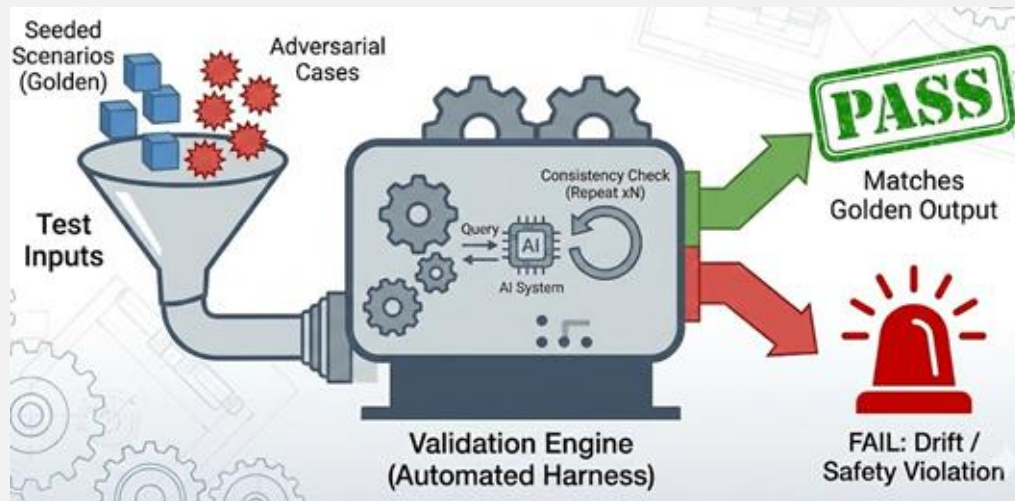
- **Hallucinated Efficacy:** Inventing facts or citations
- **Prompt Injection:** Users manipulating the AI to reveal hidden data
- **Stochastic Variation:** Getting different answers to the same question

➤ Mitigation Strategy: Fit-for-purpose controls based on the specific risk of the task (Analysis vs. Search)



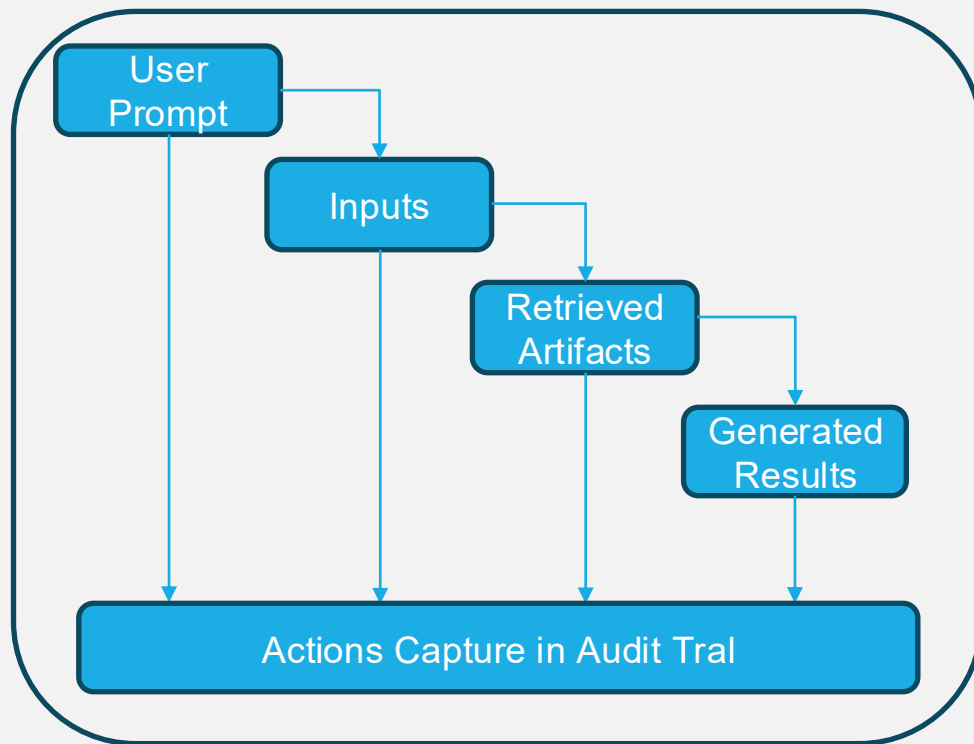
The Validation Test Harness

- **Seeded Scenarios:** A library of fixed inputs with known "Golden" outputs.
- **Adversarial Testing:** specifically designing inputs to break the system (e.g., tricking the AI into giving medical advice).
- **Consistency Checking:** Running the same seeded scenario multiple times to prove the output does not drift.



Traceability and Audit Trails

- **Complete Audit Trail:** Capturing not just the answer, but the process.
 - The exact Prompt sent.
 - The exact "Context" (documents) retrieved.
- **Citation Enforcement:** The AI is required to cite the reference source for every claim.
- **Verification:** Auditors can trace any output back to the specific document version it came from.



What “Good” Looks Like

Five Key Properties



Narrow intended use



Role based controls



**Audit trail for prompts,
inputs, outputs,
adjudications**



**Repeatable test
harness**



**Continuous
verification and drift
monitoring**

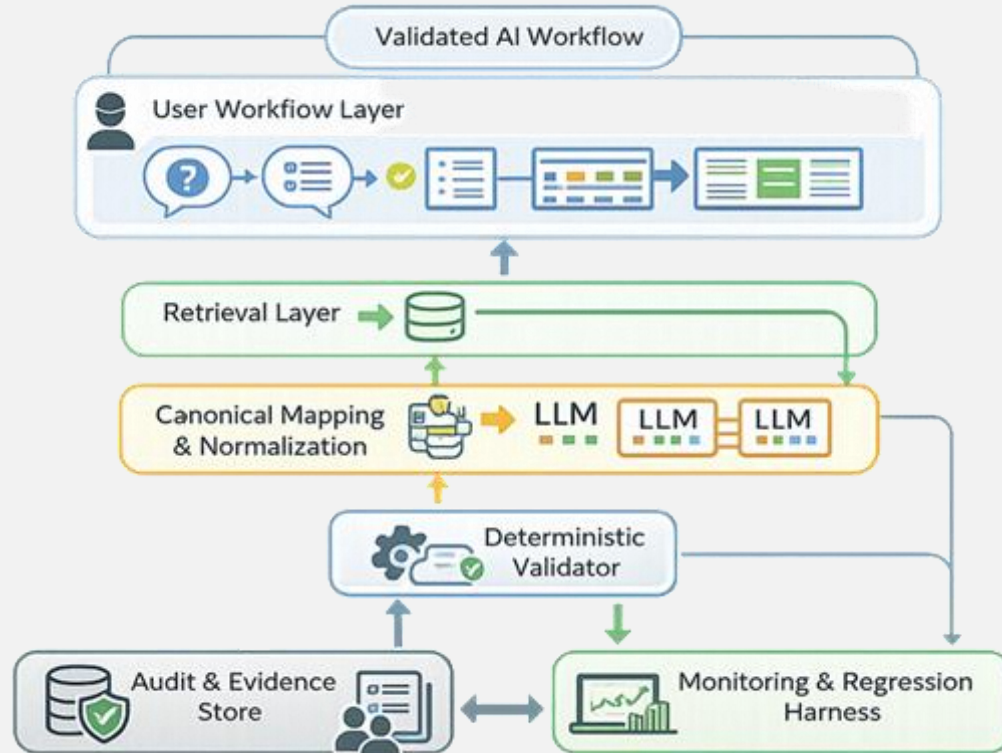


Reference Architecture for Validated AI Workflow

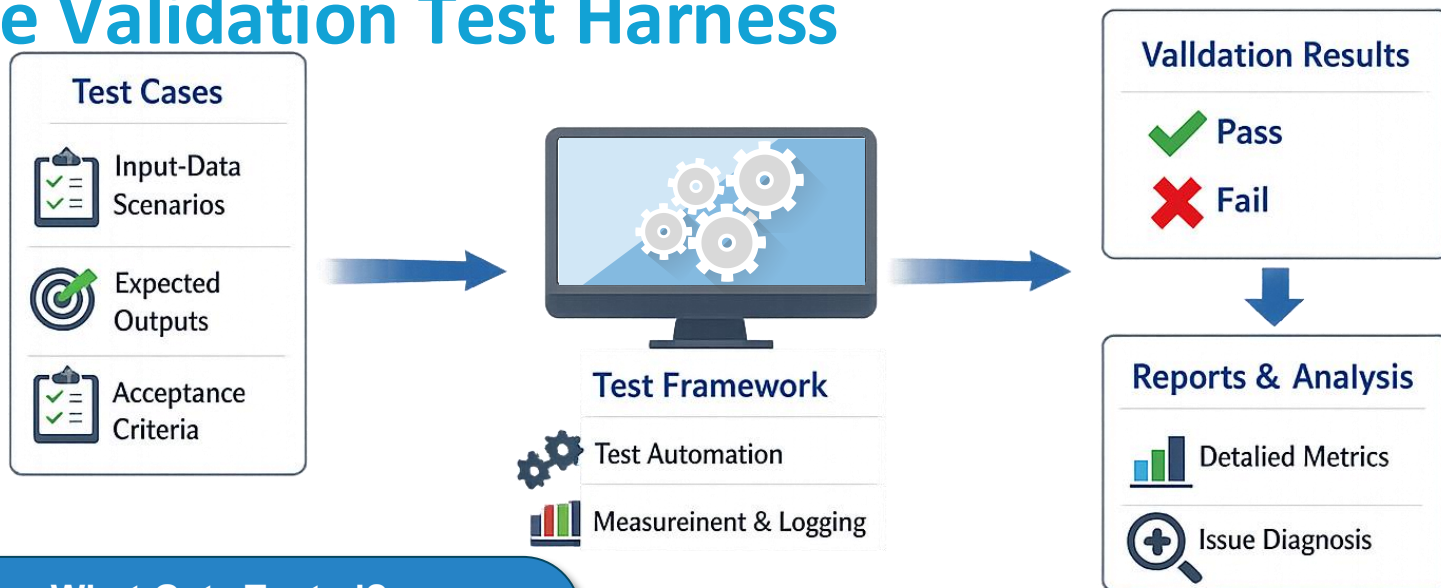
LLM a configurable component inside a larger controlled system

Architecture Elements

- User workflow layer
- Retrieval layer
- Canonical mapping and normalization
- LLM comparison or drafting layer
- Deterministic validator
- Review and adjudication workflow
- Audit and evidence store
- Monitoring and regression harness



The Validation Test Harness



What Gets Tested?

- Behavior under different conditions
- Failure detection, adverse scenarios
- Boundary enforcement
- Traceability of outputs to source evidence
- Stability across repeated runs

With and Without a Test Harness

- | | | |
|--|-----------|--|
| <ul style="list-style-type: none">• Behavior is measurable• Evidence is consistent• Failures are detectable• Validation is repeatable | vs | <ul style="list-style-type: none">• Testing is anecdotal• Lack of reproducibility• Drift is invisible• Validation is indefensible |
|--|-----------|--|



Technical Guardrails (Control Settings)

➤ Enforced Parameters:

- **Temperature = 0:** Eliminates randomness in token selection.
- **Top_p = 1.0:** Ensures the model considers the full probability distribution before selecting the most likely token.
- **Max Tokens:** Caps the output length (400–900 tokens) to prevent rambling.

➤ Output Schema Enforcement: Forcing the AI to reply in a specific format (e.g., JSON) to ensure machine readability.



Security and Data Privacy

- **Data Leakage Prevention:**
The Agent operates within the user's existing permission scope.
- **PII Protection:** Preventing the model from training on or exposing Patient Health Information (PHI).
- **Prompt Injection Defense:**
Using the "Prompt Builder" to sanitize inputs before they reach the LLM.



Continuous Verification and Drift Detection:

Validation is not an event. It is an operating model.

- **Drift Detection:** Monitoring the system post-deployment to ensure the model's behavior hasn't changed.
- **Routine Execution:** Automatically running the "Seeded Scenarios" on a schedule.
- **Version Control:** If the underlying model version changes, the full validation suite is re-triggered.



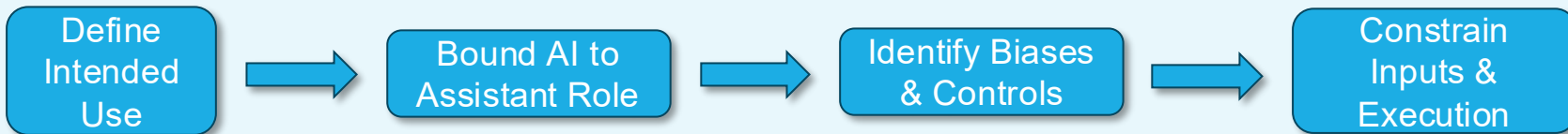
A Key Technical Components and Considerations: Explicit enforced LLM control settings

Control Category	Details	Description
Sampling and determinism controls	temperature = 0 top_p = 1.0	These settings suppress creative variation and constrain the model to the most likely completion paths
Reproducibility and drift visibility	seed = fixed per validation scenario	Seed-based determinism is treated as best effort, not a guarantee
Output length and termination	max_tokens = 400 to 900 explicit stop sentinel enforced	Token limits are sized to the expected volume of structured findings and prevent uncontrolled or truncated output
Penalty configuration	frequency_penalty = 0 presence_penalty = 0	Novelty is explicitly undesirable in regulated review tasks
Output channel enforcement	Strict JSON schema enforcement	The system does not permit narrative prose intended for direct consumption

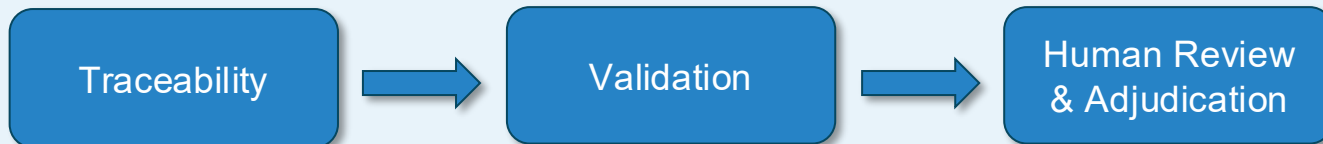


Validate the Process, not the model alone

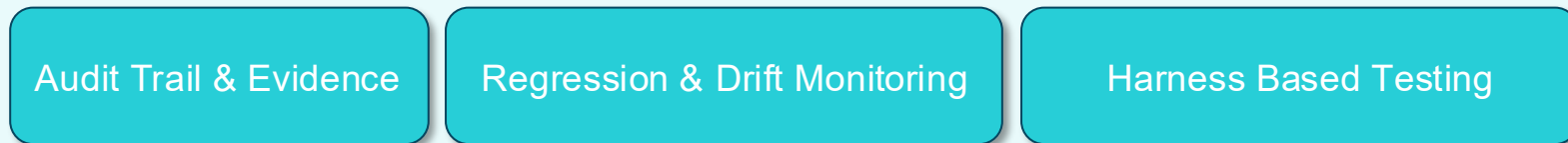
Scope and Governance



Controlled Execution



Oversight and Lifecycle



Wrap-up

Summary: Do not trust the model. Trust a controlled process that constrains, tests, monitors, and evidences the model's role.

5 Practical Actions

1. Bound one use case narrowly
2. Write intended use precisely
3. Build the harness early
4. Treat prompts and retrieval as configuration
5. Design for drift from day one

Final Takeaway

When AI is embedded inside a controlled process with clear boundaries and objective evidence, it behaves like any other validated computerized system.

Thank You

<https://www.sycamoreinformatics.com>

Come see what is possible today, Booth #31



Bill Qubeck

Bill.Qubeck@SycamoreInformatics.com
<https://www.linkedin.com/in/williamqubeck/>



Heather Gorr

Heather.Gorr@SycamoreInformatics.com
<https://www.linkedin.com/in/heather-gorr-phd/>

