Public



Accelerating TLF Generation in Clinical Trials Using Domain-Tuned Generative AI

Presenter – Niharikka Tyagi

Public

Author & Biography



Niharikka Tyagi

Software Developer at Sycamore Informatics with over 2 years of experience advancing the DSW product, focusing on applying software solutions and AI to streamline pharmaceutical processes and accelerate clinical trial outcomes.

I am passionate about building tools that positively impact healthcare and research.

Public

Agenda

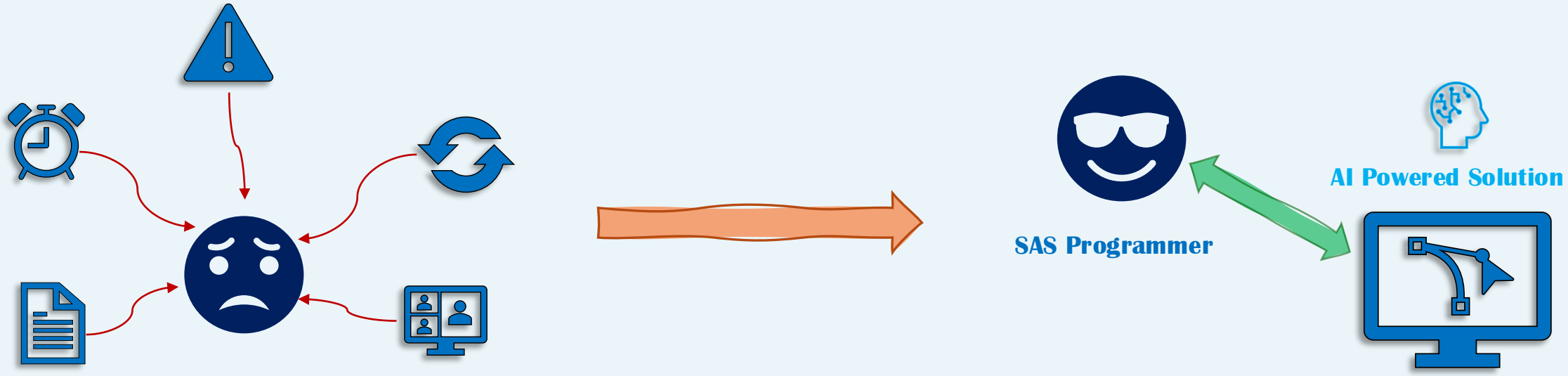
1. The Problem
2. The Solution
3. How it Works
4. Validation
5. The Impact
6. Conclusion

Public

Beyond the Bottleneck

A New Workflow for TLF Generation

From Manual Coding to AI-Assisted Review
A 'Glass-Box' Approach



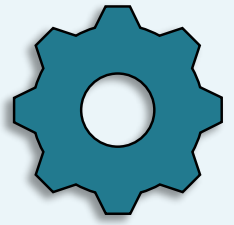
Public

The Problem

Slow and Manual TLF Generation

Understanding the common hurdles in our current workflow & the opportunities they present for a new approach






Public

The Problem

Slow and Manual TLF Generation

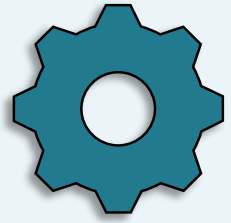


A disproportionate amount of expert programmer/statistician time is spent on routine, boilerplate outputs.

May delay high-priority analyses and slowing down the decision process.

Time Intensive Process







Public

The Problem

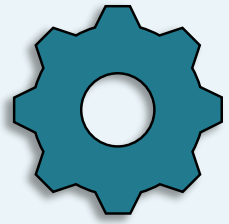
Slow and Manual TLF Generation

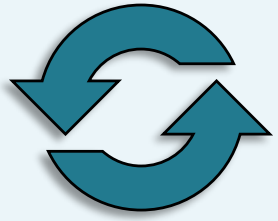


Every line of manually-written code has a possibility of containing human error.

A simple typo can have a cascading effect, jeopardizing timelines and creating re-validation burden.

High Risk & Error Prone



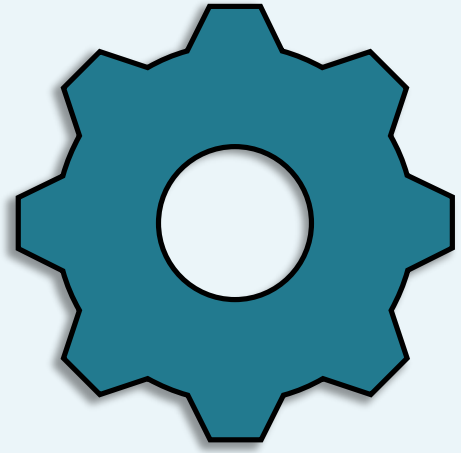




Public

The Problem

Slow and Manual TLF Generation



In the absence of a single, enforced framework, each programmer “style” becomes the *de facto* standard.

QC is forced into forensics of interpreting logic of each program, not just verifying it.

Gap in Standardization







Public

The Problem

Slow and Manual TLF Generation



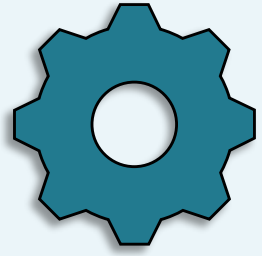
A minor spec change often triggers a full-scale manual rework and re-validation cycle.

This repetitive churn is a primary source of project delays and team frustration.

Repetitive & Rigid





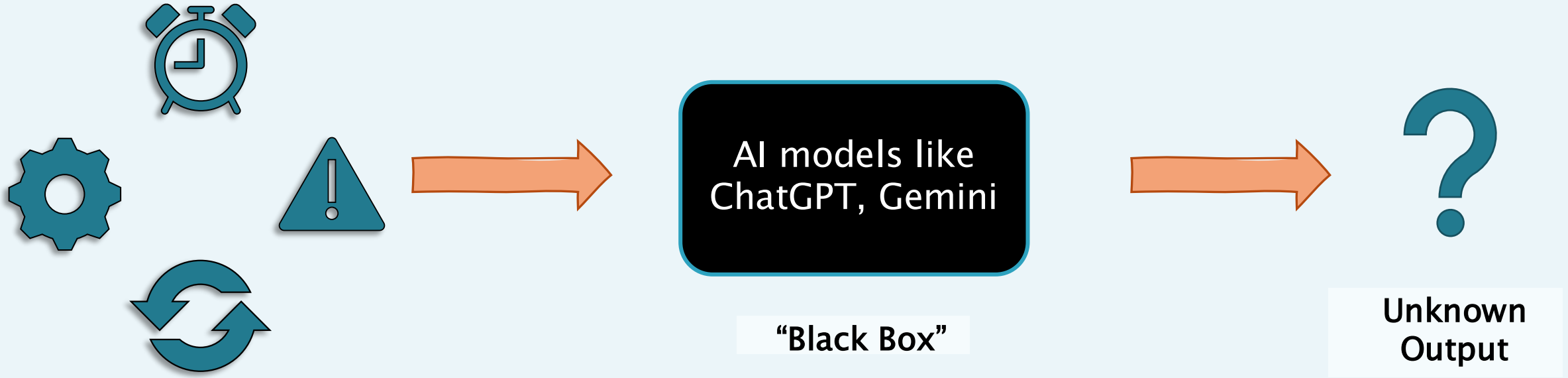


Public

The Problem

Why a General-Purpose AI is not suitable

We have these problems and we're tempted to feed them into a "magic" AI box, but the result is a total unknown



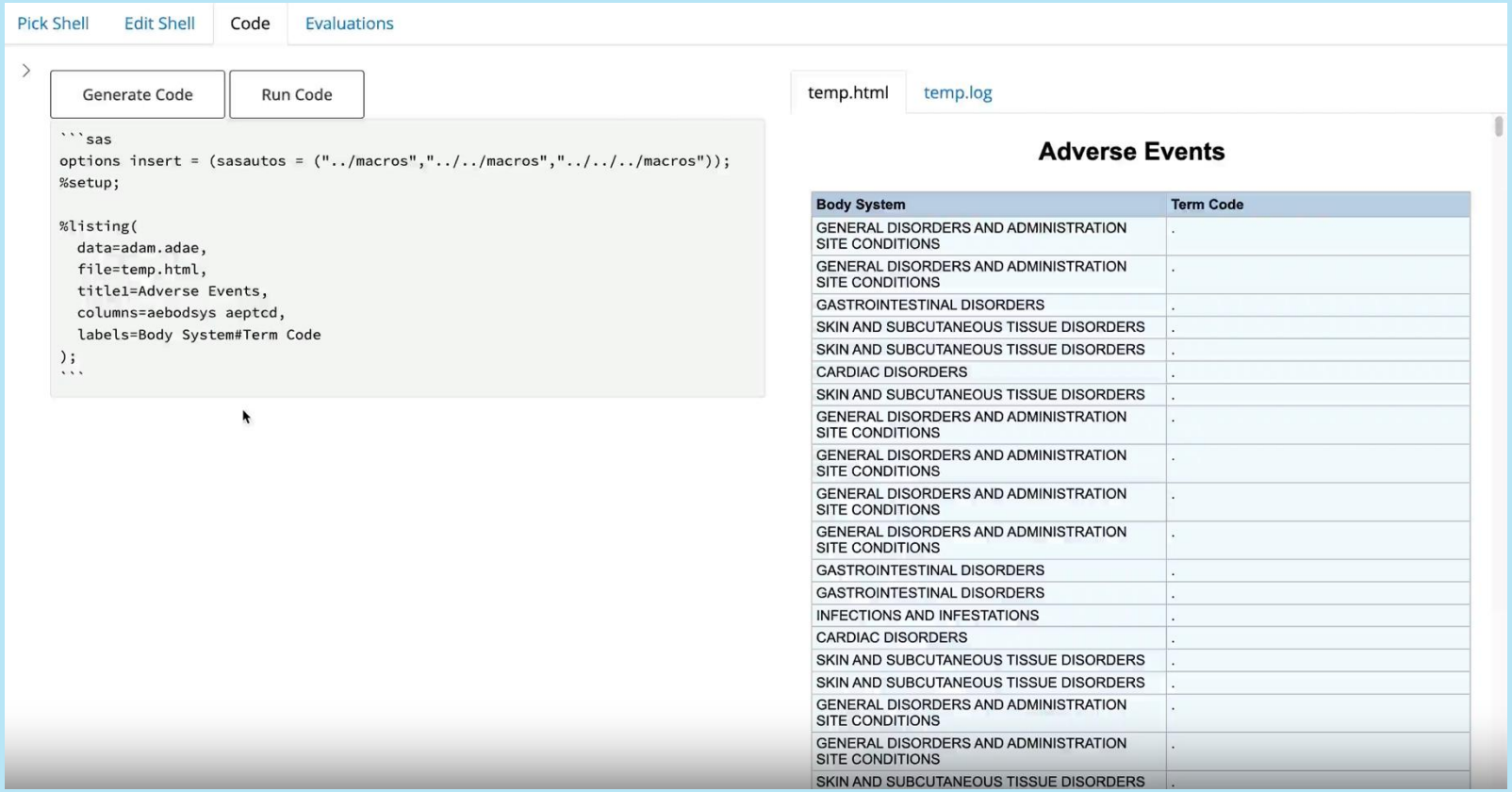
Public

The Solution

Generative AI for TLF Automation

Time Efficient

Secure Env



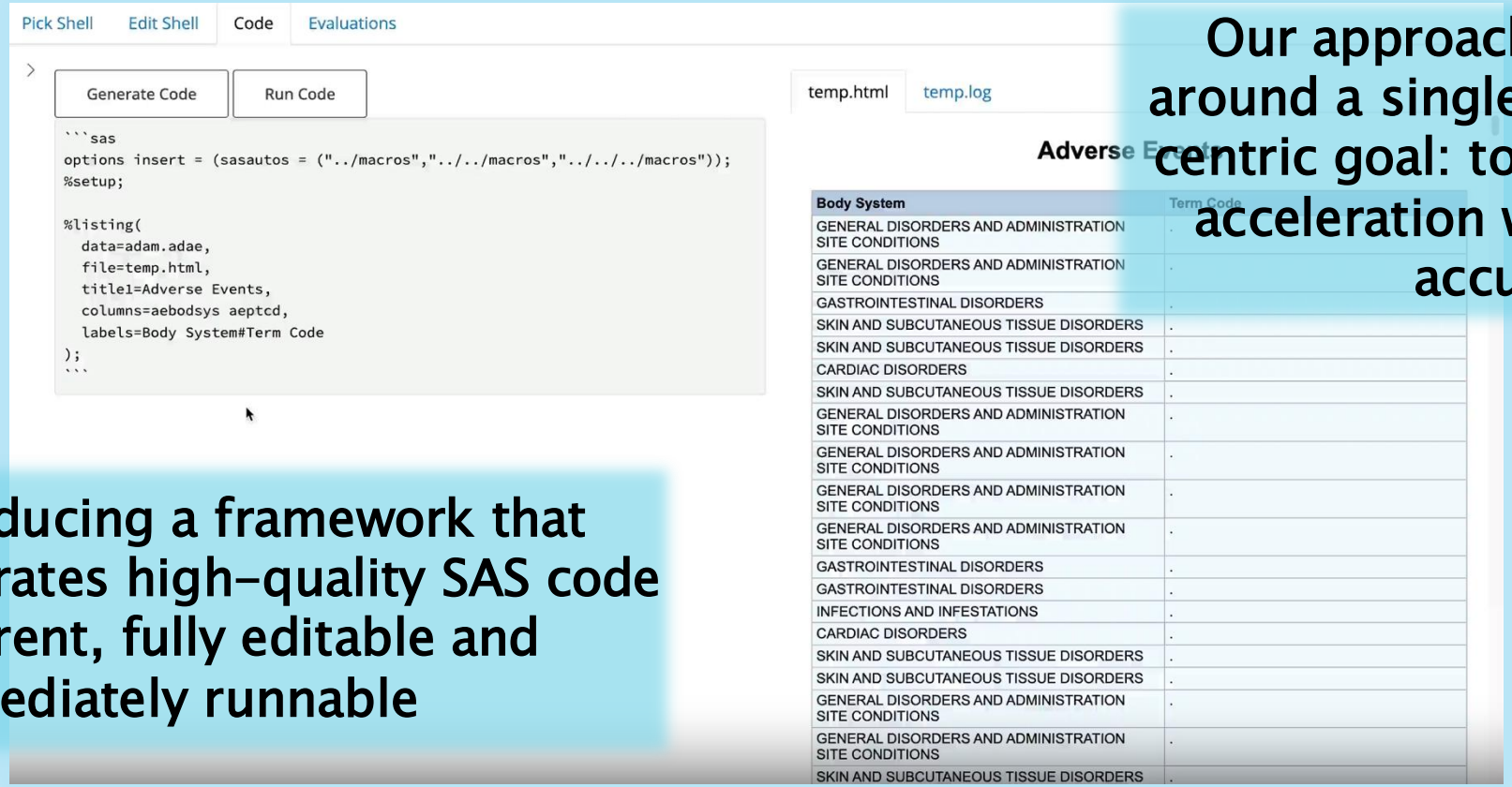
Transparent

GxP Compliant

Public

The Solution

Generative AI for TLF Automation



Our approach is designed around a single, programmer-centric goal: to combine rapid acceleration with verifiable accuracy

We're introducing a framework that instantly generates high-quality SAS code – transparent, fully editable and immediately runnable

Public

The Solution

Generative AI for TLF Automation

Secure

A private, sandboxed env for GxP Compliance

Smart

An AI engine with Clinical context

Safe

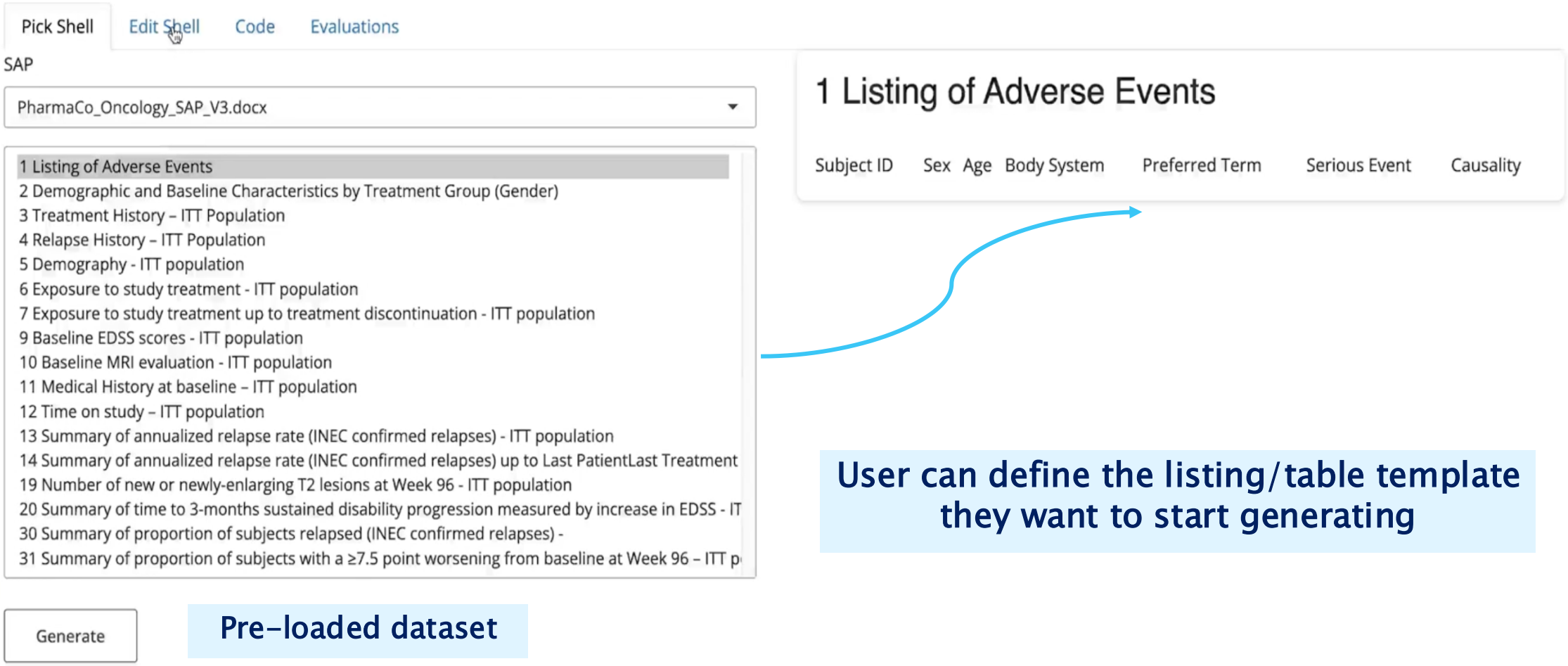
By Validation

Not “reinventing the wheel”

Public

How it Works: Spec to Code Workflow

Step 1: User picks an SAP



Pre-loaded dataset

User can define the listing/table template they want to start generating

Public

How it Works: Spec to Code Workflow

Step 2: Generate & review the code

System intelligently retrieves the required Macro

Generate Code | Run Code

```sas
options insert = (sasautos = ("../macros","../
%setup;

%listing(
  data=adam.adae,
  file=temp.html,
  title1=Adverse Events,
  columns=aebodsys aeptcd,
  labels=Body System#Term Code
);
```

temp.html | temp.log

Generated code

Adverse Events

Body System	Term Code
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GASTROINTESTINAL DISORDERS	.
SKIN AND SUBCUTANEOUS TISSUE DISORDERS	.
SKIN AND SUBCUTANEOUS TISSUE DISORDERS	.
CARDIAC DISORDERS	.
SKIN AND SUBCUTANEOUS TISSUE DISORDERS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GASTROINTESTINAL DISORDERS	.
GASTROINTESTINAL DISORDERS	.
INFECTIONS AND	.

Final Output

Public

How it Works: Spec to Code Workflow

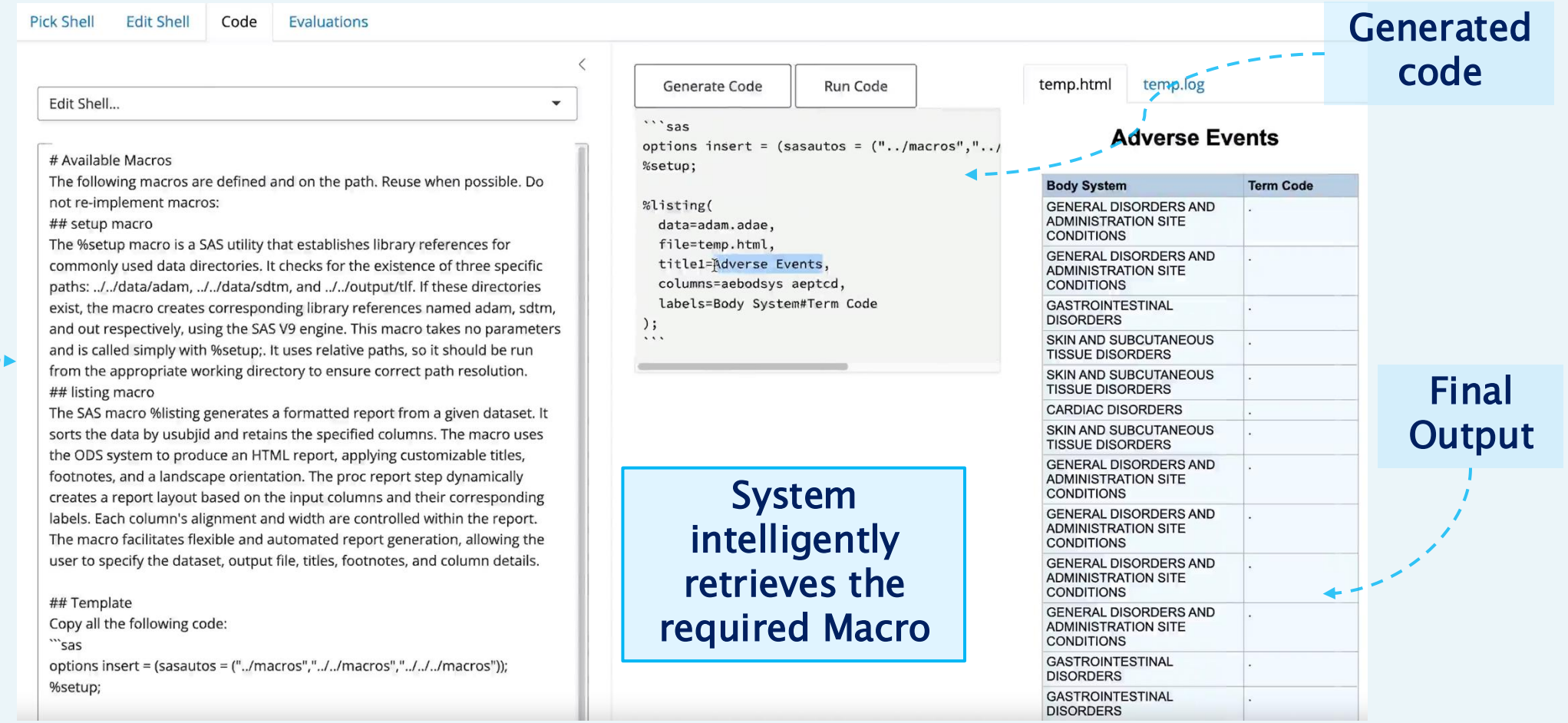
Step 2: Generate & review the code

Macro description and template

Generated code

Final Output

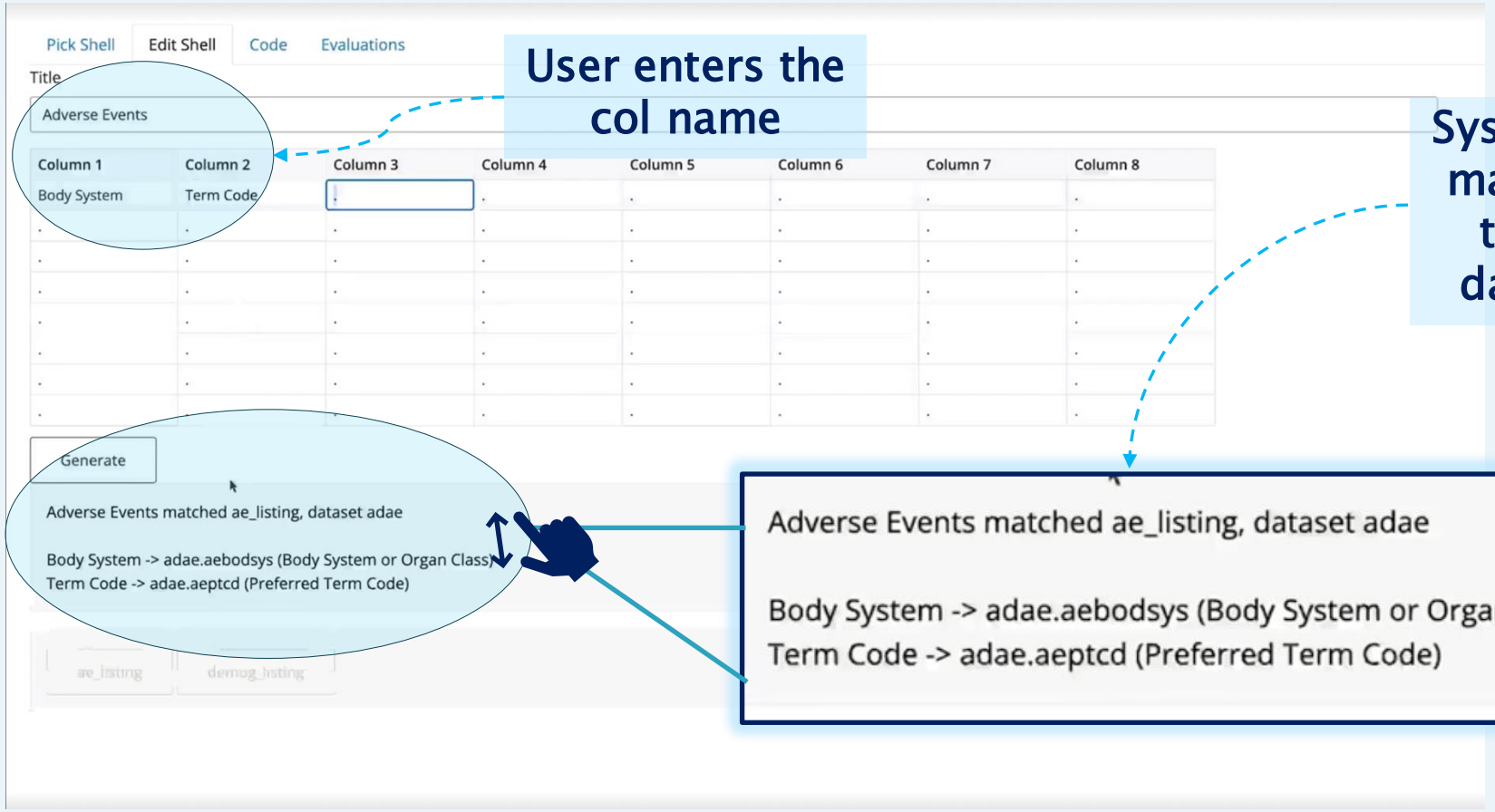
System intelligently retrieves the required Macro



Public

How it Works: Spec to Code Workflow

User can edit the Shell



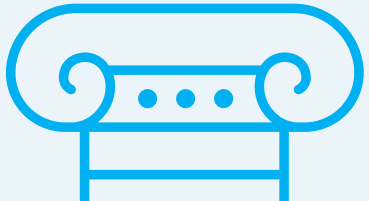
Public

Validation: 3 Pillars of Verification

The Burning question:
How do we make sure its GxP-Compliant

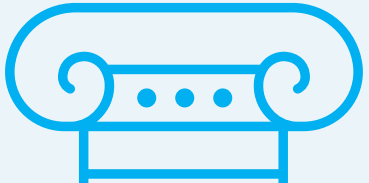


Macro Library



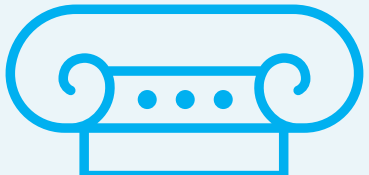


Secure AI env





Human in Loop



Public

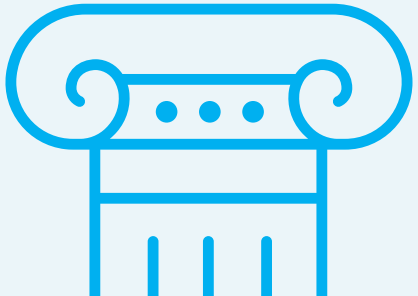
Validation: 3 Pillars of Verification



Macro Library

The macro library is not a simple collection of individual programs.

It is a robust, metadata-driven SAS framework architected to be the single, GxP compliant engine for all TLF Generation



Secure AI env



Human in Loop

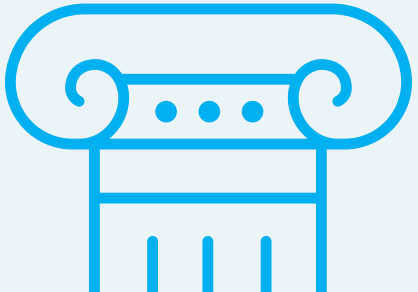


Public

Validation: 3 Pillars of Verification



Macro Library



Edit Shell...

Available Macros
The following macros are defined and on the path. Reuse when possible. Do not re-implement macros:
setup macro
The %setup macro is a SAS utility that establishes library references for commonly used data directories. It checks for the existence of three specific paths: ../../data/adam, ../../data/sdtm, and ../../output/tlf. If these directories exist, the macro creates corresponding library references named adam, sdtm, and out respectively, using the SAS V9 engine. This macro takes no parameters and is called simply with %setup;. It uses relative paths, so it should be run from the appropriate working directory to ensure correct path resolution.
listing macro
The SAS macro %listing generates a formatted report from a given dataset. It sorts the data by usubjid and retains the specified columns. The macro uses the ODS system to produce an HTML report, applying customizable titles, footnotes, and a landscape orientation. The proc report step dynamically creates a report layout based on the input columns and their corresponding labels. Each column's alignment and width are controlled within the report. The macro facilitates flexible and automated report generation, allowing the user to specify the dataset, output file, titles, footnotes, and column details.

Template
Copy all the following code:
```sas
options insert = (sasautos = ("../macros","../../macros","../../../macros"));
%setup;
```

Generate Code | Run Code

```sas
options insert = (sasautos = ("../macros","../
%setup;

%listing(
  data=adam.adae,
  file=temp.html,
  title1=Adverse Events,
  columns=aebodsys aeptcd,
  labels=Body System#Term Code
);
```

AI Generated detailed Macro descriptions

temp.html | temp.log

Adverse Events

Body System	Term Code
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GASTROINTESTINAL DISORDERS	.
SKIN AND SUBCUTANEOUS TISSUE DISORDERS	.
SKIN AND SUBCUTANEOUS TISSUE DISORDERS	.
CARDIAC DISORDERS	.
SKIN AND SUBCUTANEOUS TISSUE DISORDERS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	.
GASTROINTESTINAL DISORDERS	.
GASTROINTESTINAL	.

Public

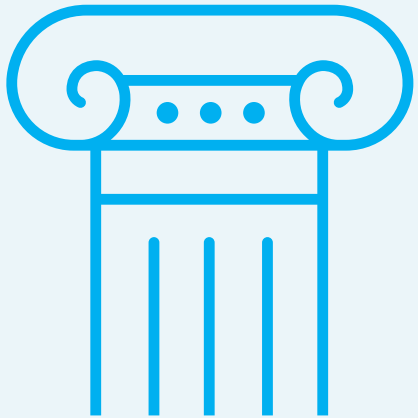
Validation: 3 Pillars of Verification



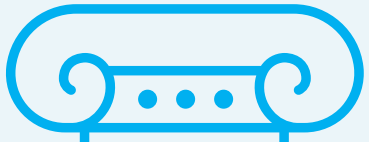
Secure AI env

The entire system, including the domain-tuned LLM is hosted within a secure, private and traceable environment

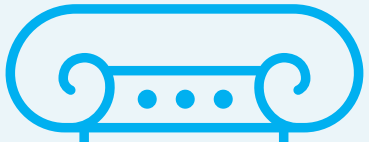
The platform adheres to strict industry standards for access control & data encryption



Human in Loop



Macro Library

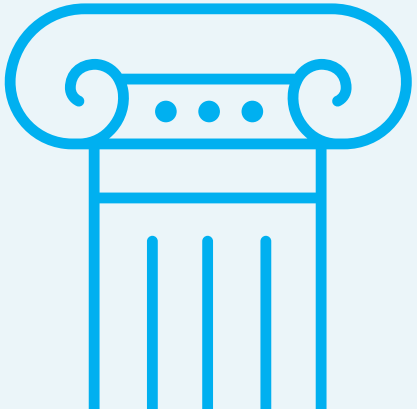


Public

Validation: 3 Pillars of Verification



Human in Loop



The tool is designed as a programmer's co-pilot, not an autonomous system

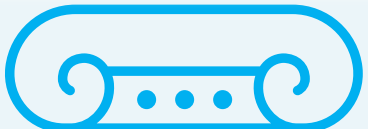
The "Code" tab is the built-in, mandatory QC checkpoint

The programmer performs final verification before execution

Macro Library

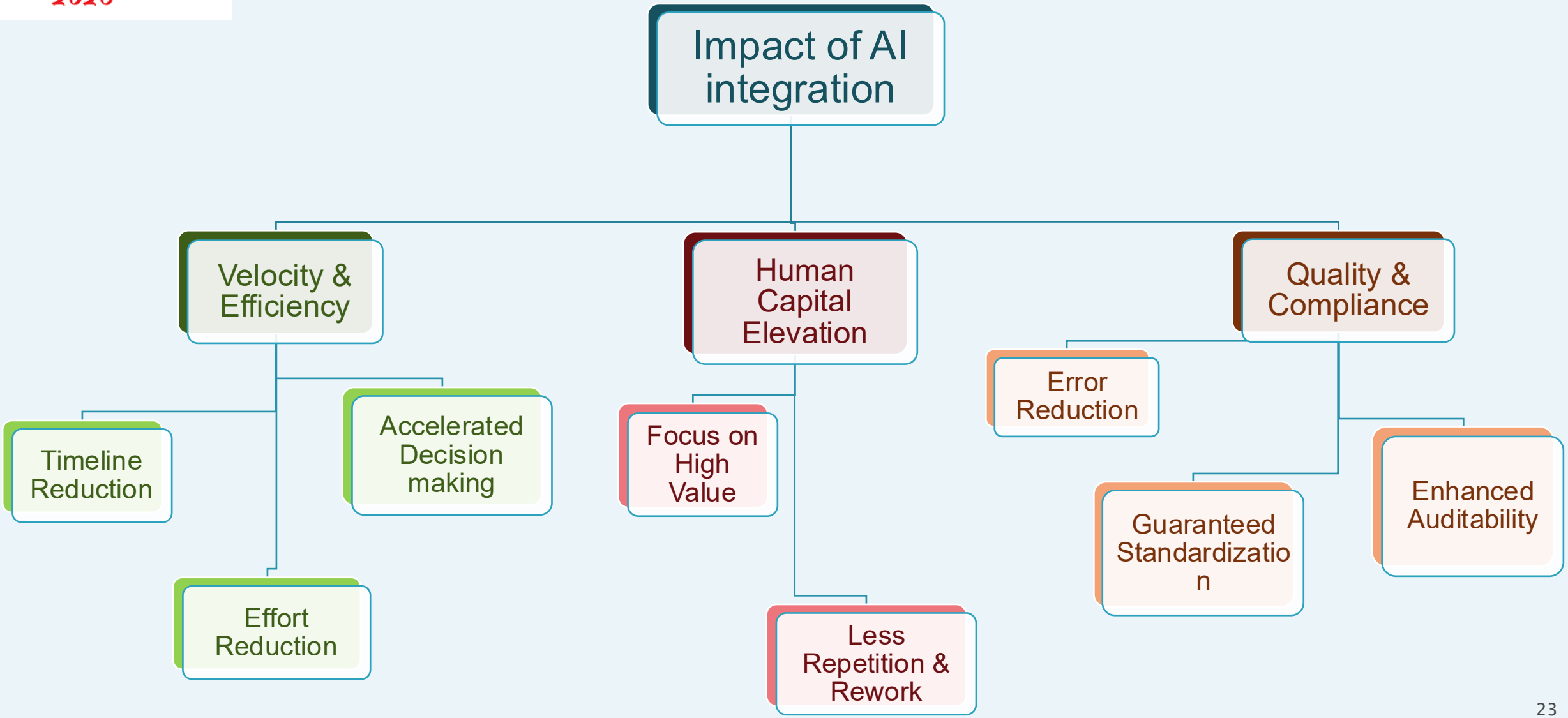


Secure AI env



Public

The Impact: Weeks to Hours



Public

The Impact: Weeks to Hours

From Manual Bottleneck to Strategic Accelerator

The Programmer

Eliminates the 80% of repetitive, boilerplate work.

This frees you to focus on the 20% of complex, novel analyses where your expertise truly matters.

Impact:
Empowerment & Focus



Impact:
Speed & Iteration

The Statistician

The "spec-to-output" cycle:

Reduces the timeline from weeks to hours. This enables rapid, data-driven decisions

The Submission

Guaranteed Quality & standardization:

Enforces 100% consistency by using a validated macro framework.

Produces error-free, transparent, editable code with the ability to run and produce outputs rapidly within a much smaller timeline.

Impact:
Quality & Velocity

Public

Conclusion & Final Thoughts

"The challenge is not *if* we should use Generative AI, but *how* we can leverage it responsibly.

We have demonstrated a way to bridge innovation with compliance, moving the industry from manual programming to AI-assisted review."

Public



Thank You!

Q&A