Syn2Real

Traceable, AI-generated synthetic SDTM and ADaM for automation testing

Thibaud Coroller, PhD - on behalf of the winning team

5 October 2026, Denver CO

CDISC AI Innovation Challenge 2026 · Use Case 1 · Winning submission

NOVARTIS

The team



Giuseppe Hannen

AWS
Delivery Consultant - AI&ML



Srikrishna Chaitanya Konduru

AWS
Sr. Delivery Consultant - AI&ML



Peter Campbell

Novartis
Date Engineering Lead, AI & DE



Peter Krusche

Novartis
Global Head, AI & Data Engineering

THE PROBLEM

Why code built on synthetic data breaks

Code is written and tested against synthetic data › **The real study data arrives** › **The code breaks**

Because three things were missing from the data all along

Not conformant

Values that the specification would never allow: wrong codelist, wrong type, out of range.

Not coherent

A subject's records contradict each other, and domains don't line up with one another.

Not traceable

When something is wrong there is no way to see why the value is what it is.

Make it conformant, coherent and traceable, and the code carries over.

THE PROBLEM

What today's approaches can and cannot give you

APPROACH	WHAT IT GIVES YOU	WHAT IT CANNOT GIVE YOU
Hand-building data	Conformant and coherent records, fully under your control.	Coverage of a whole study, or any record of why a value was chosen.
Rules and simulation	Scale, repeatability, and coherent patient histories.	SDTM and ADaM without writing new rules for every study.
Statistical and machine learning	Realism at scale, learned from real studies.	Anything at all without access to real patient data.
Large language models, used directly	Fast, flexible output over any domain you describe.	Consistency across a large domain, or values that stay inside the specification.

Each gives you part of it. None gives you all three at study scale — without using real patient data.

THE ANSWER

Build all three in, from the standards up

INPUT	GROUNDING	GENERATION	DERIVATION	OUTPUT
Standards	**Metadata**	**Synthetic SDTM**	**Admiral R**	**Synthetic ADaM**
SDTM · ADaM · USDM	resolved and pinned	one agent per record	reusable on real SDTM	validated

CONFORMANT

Specifications are resolved into pinned rules first, and every generated value is checked against them as it is produced.

COHERENT

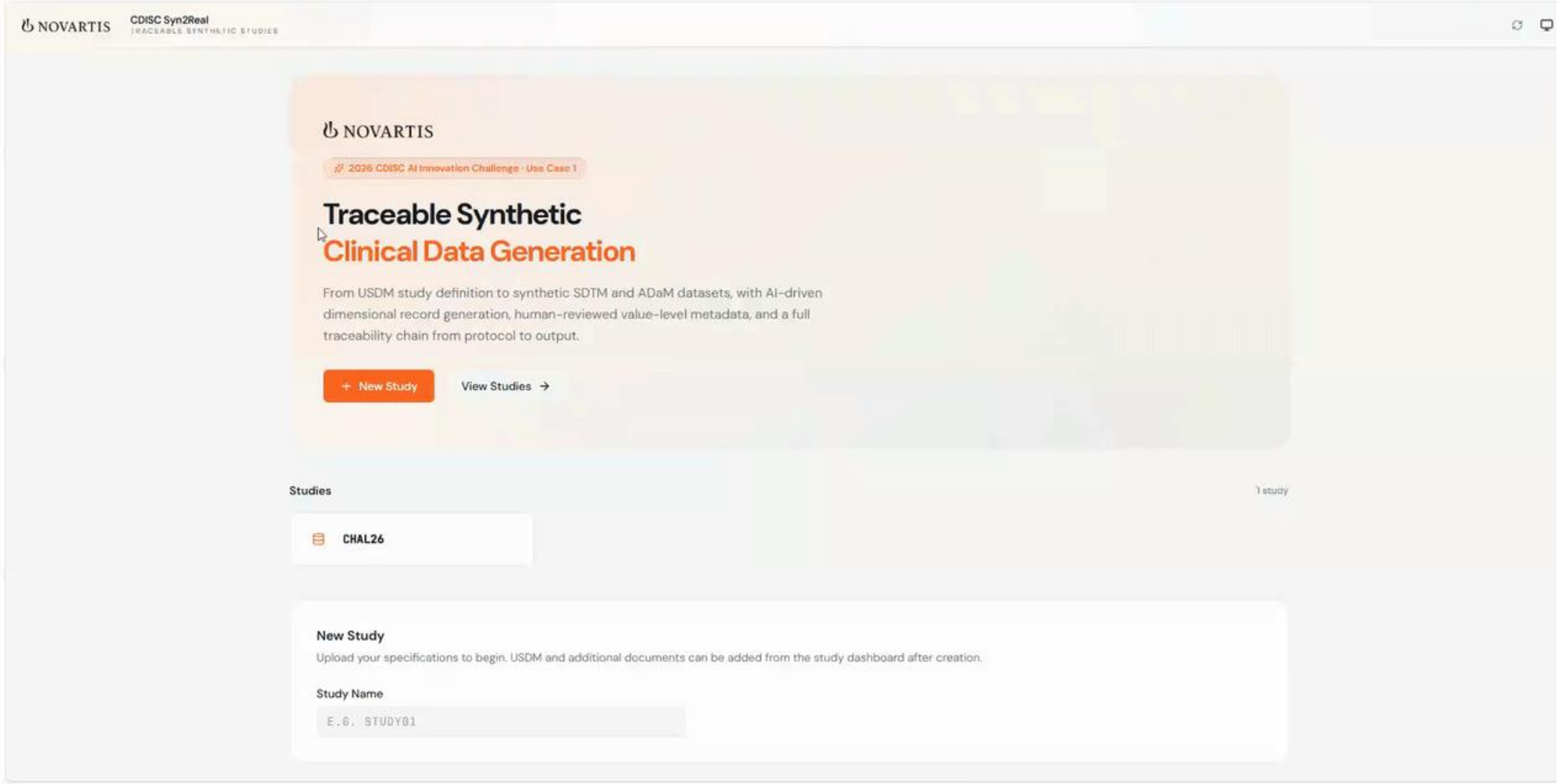
Every record is written to fit one subject's clinical story, so domains agree with each other and with the patient.

TRACEABLE

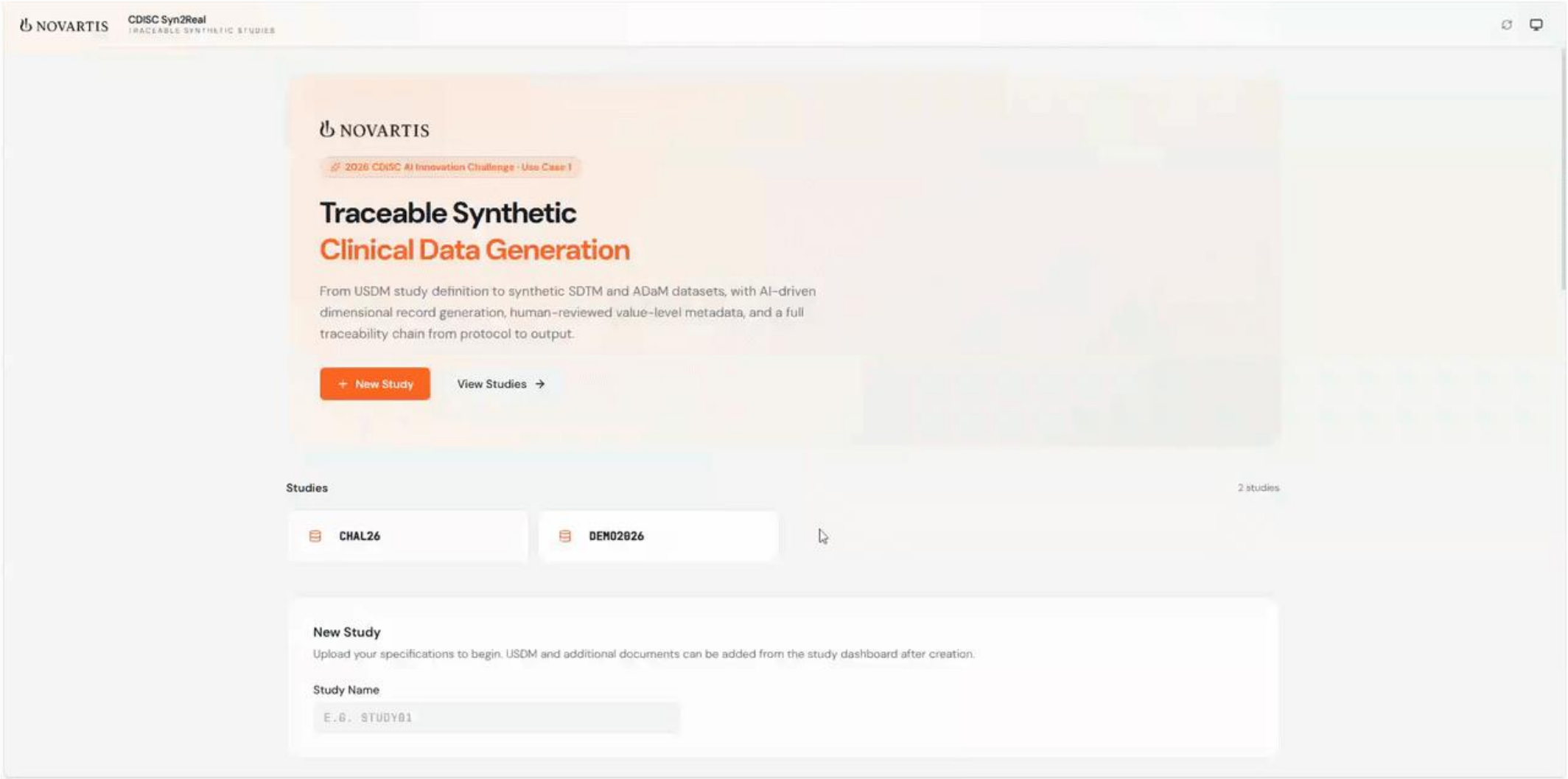
Provenance is recorded as data is produced, and ADaM is computed from SDTM by code rather than invented.

No real patient data at any point — the only inputs are the study's own standards.

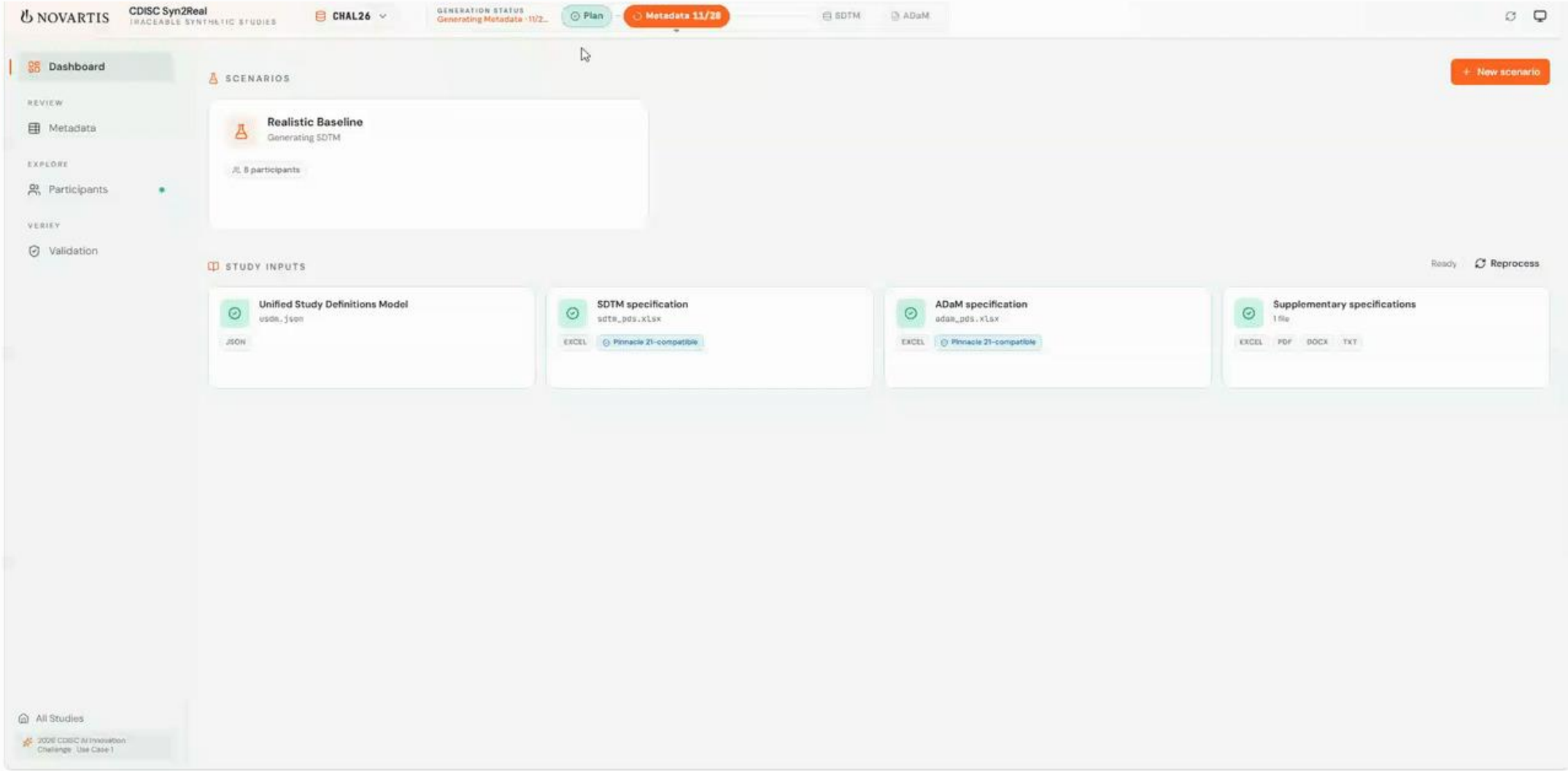
Creating a study in Syn2Real



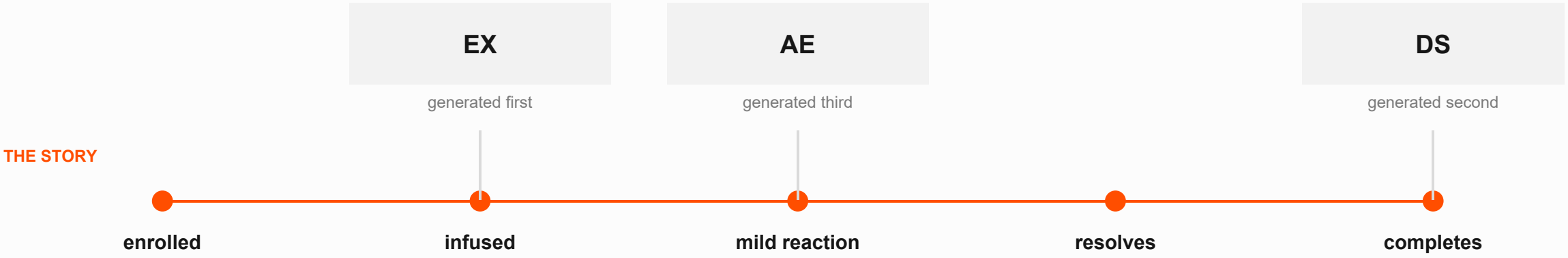
Planned scenarios that synthetic data will be conditioned on



Planned patient narratives that synthetic data will be conditioned on



Every record is written to fit the patient narratives



Written during generation

Each entry is dated and links back to the records it summarises.

Read in both directions

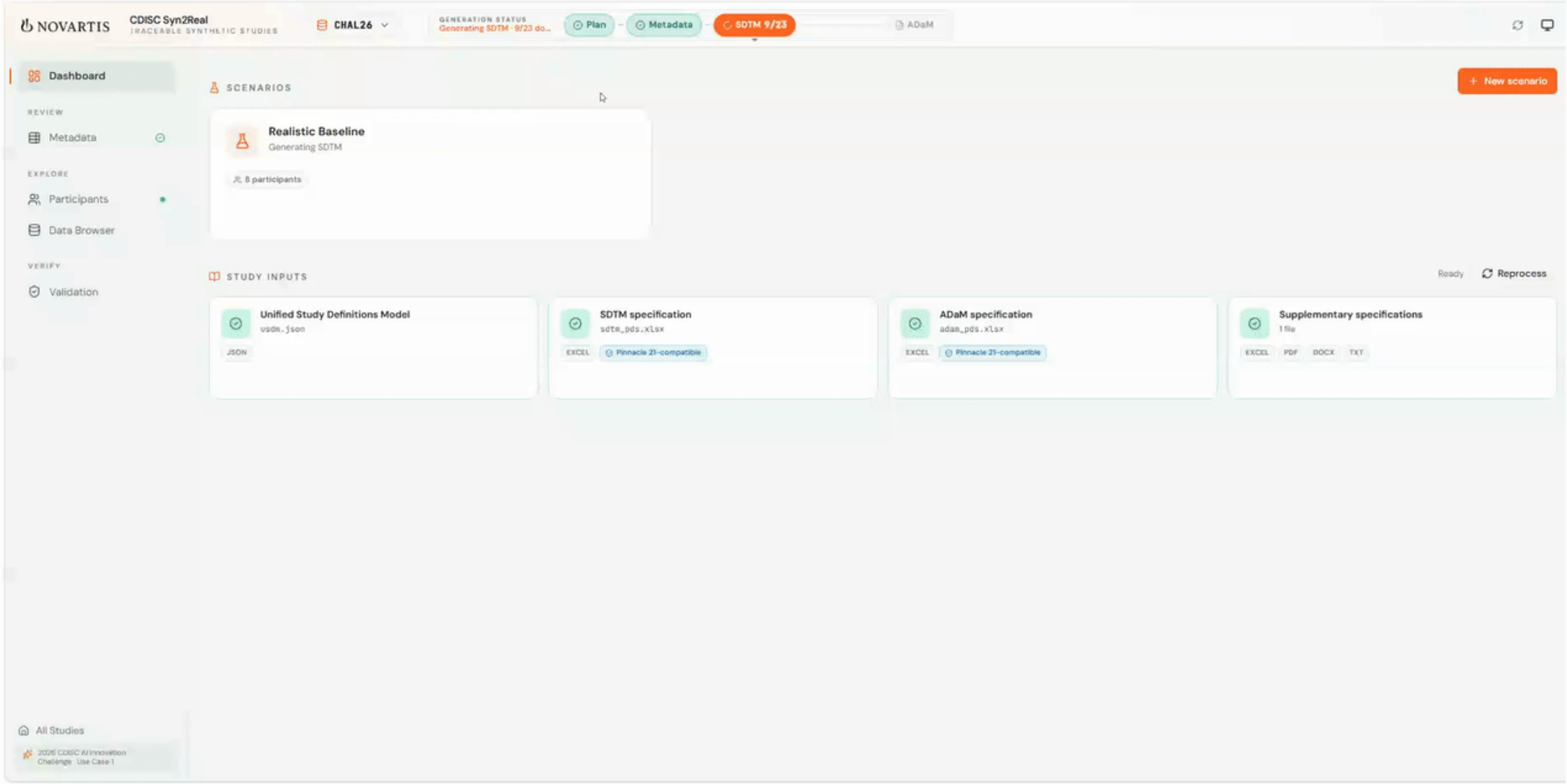
A record generated late is fitted to a story that already has an ending.

Bounded context

A thousand records compress to fifteen or twenty entries.

CONFORMANT · IN THE APPLICATION

Metadata assumptions made explicit before generation



Enforcing the metadata assumptions globally using validators

Specifications are resolved into pinned, checkable rules before any data is generated.

WHAT THE SPECIFICATION SAYS for AE.TOXGR

"Toxicity grade codelist"

▼

WHAT THE SYSTEM RESOLVES IT TO

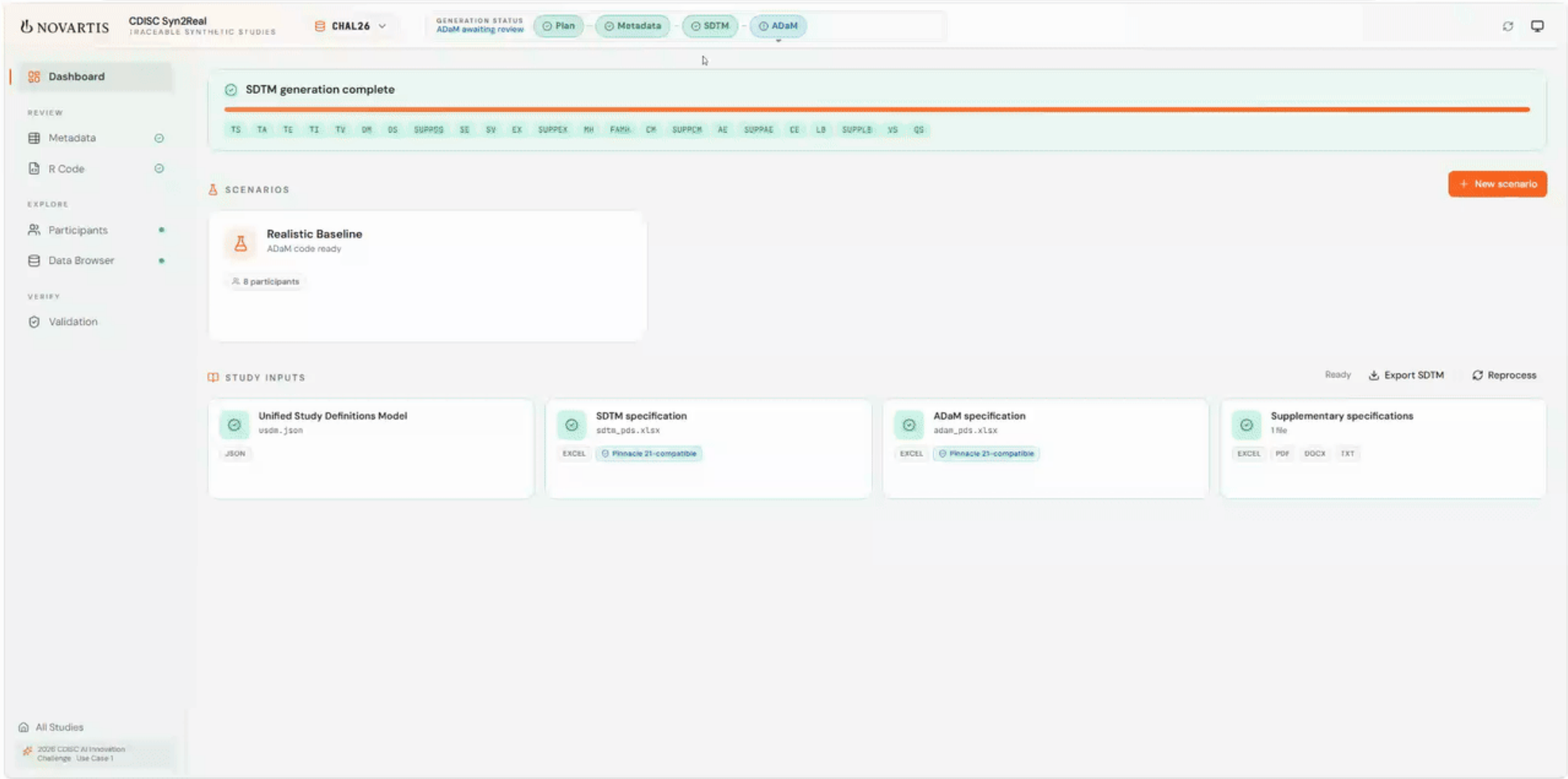
Categorical(1, 2, 3, 4, 5)

▼

WHAT HAPPENS AT GENERATION

Agent proposes: "G1" → **Rejected** → Regenerated as: 1

The synthetic SDTM data



Define the record set per domain, then generate one record at a time

STUDYID › SUBJID › VISITNUM › LBTEST › **One record**

The key that makes a row unique, resolved one level at a time

992

lab records in this study, each one its own bounded call

23

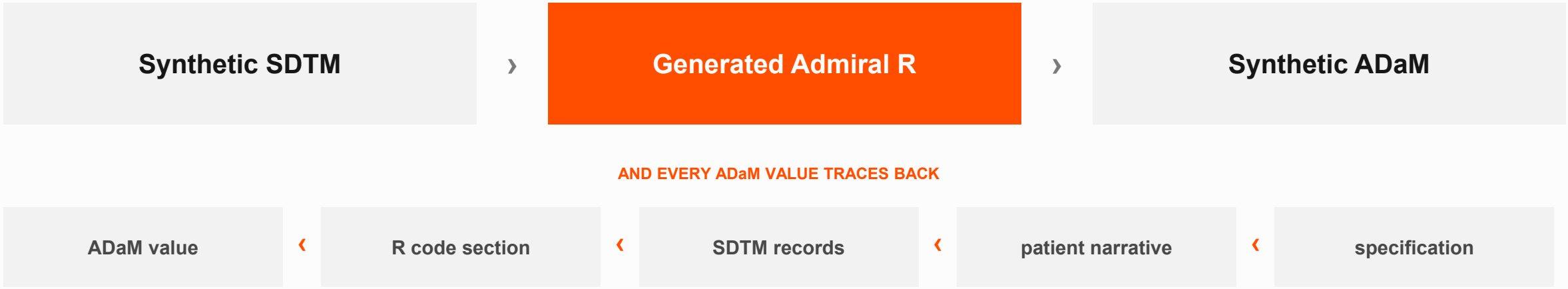
SDTM domains covered by this same recursion, with no per-domain rules

Parallel Generation

independent records generate concurrently, so volume costs time, not quality

ADaM is computed from SDTM

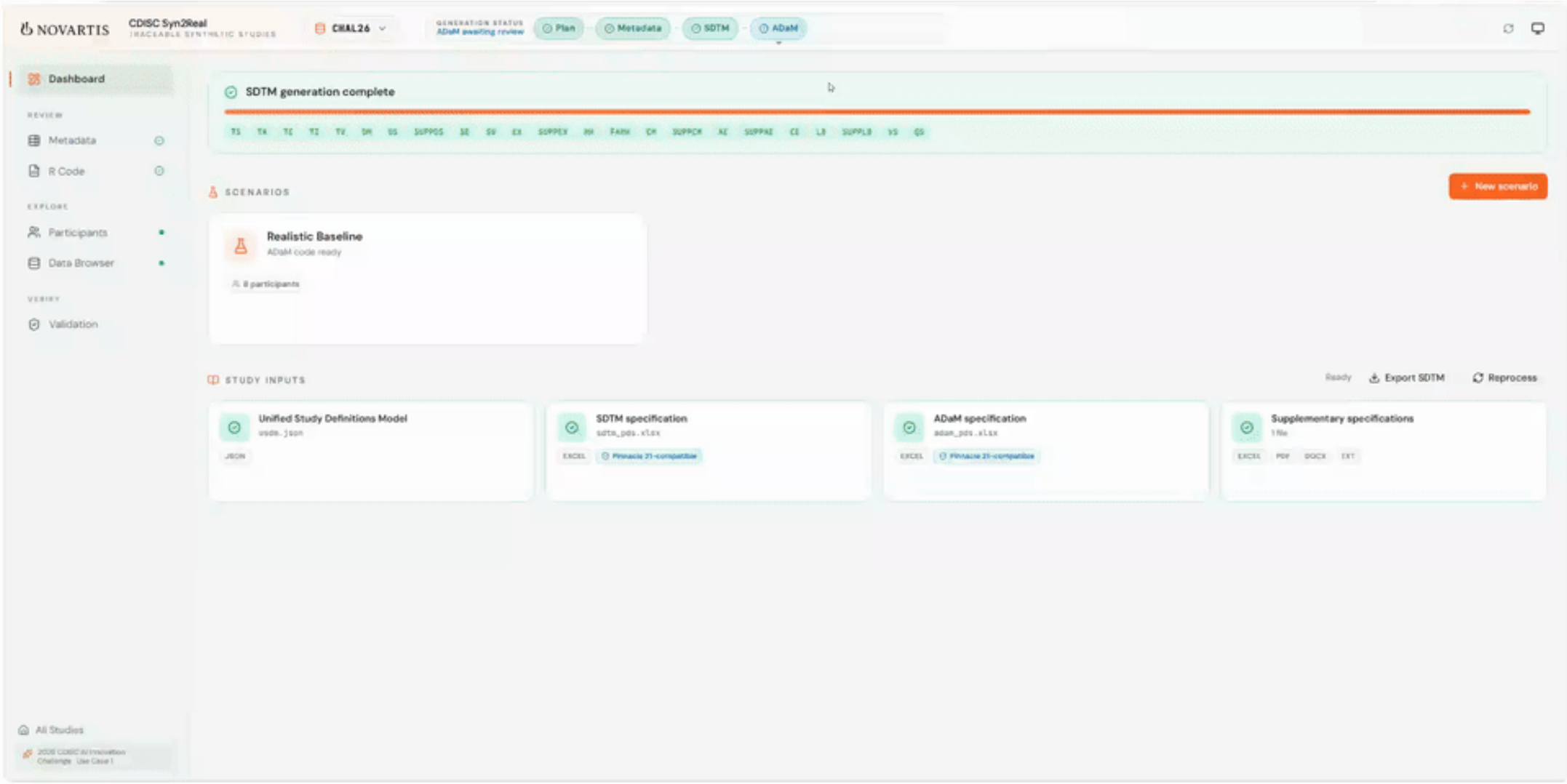
Generating the derivation code instead of the values is what makes ADaM traceable.



Because ADaM is computed from SDTM, the two cannot drift apart.
The same code runs on the real study.

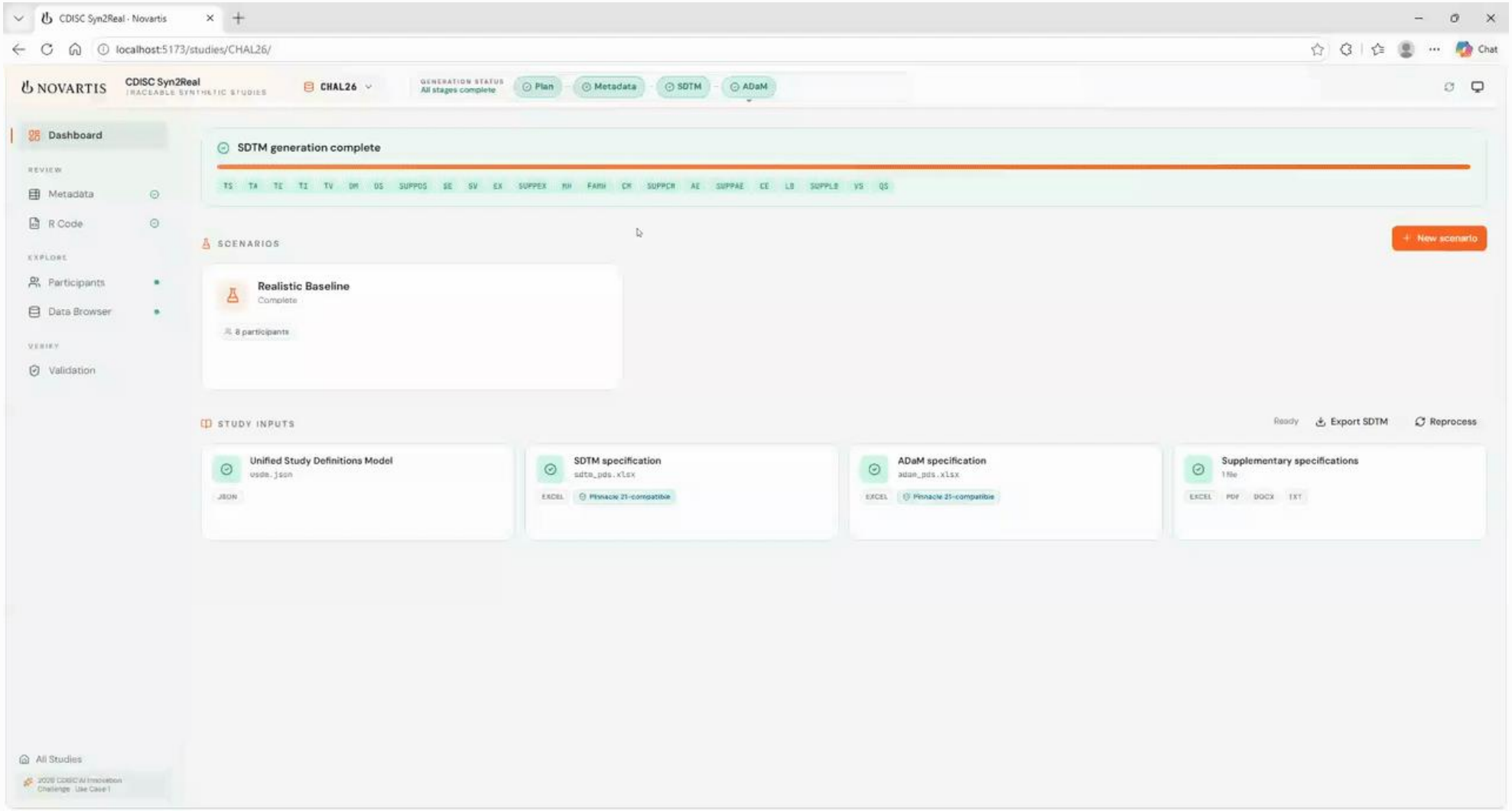
TRACEABLE · IN THE APPLICATION

The ADaM derivation code



CONFORMANT · IN THE APPLICATION

Validating synthetic data using natural language rules



WHAT IT TAKES TO RUN

Built to run at study scale

SCALE

- Records that share context are batched into one conversation, so the specs and the narrative are explored once and reused.
- The remaining records are independent, so they generate in parallel — capped at 18 concurrent sessions.

CONTROL

- Every dependency is pinned: CT snapshot, Admiral version, model identifier.
- Runs inside Novartis-controlled AWS — no data leaves the boundary.
- The model is never trusted as a computational component; every value is checked against a pinned specification.

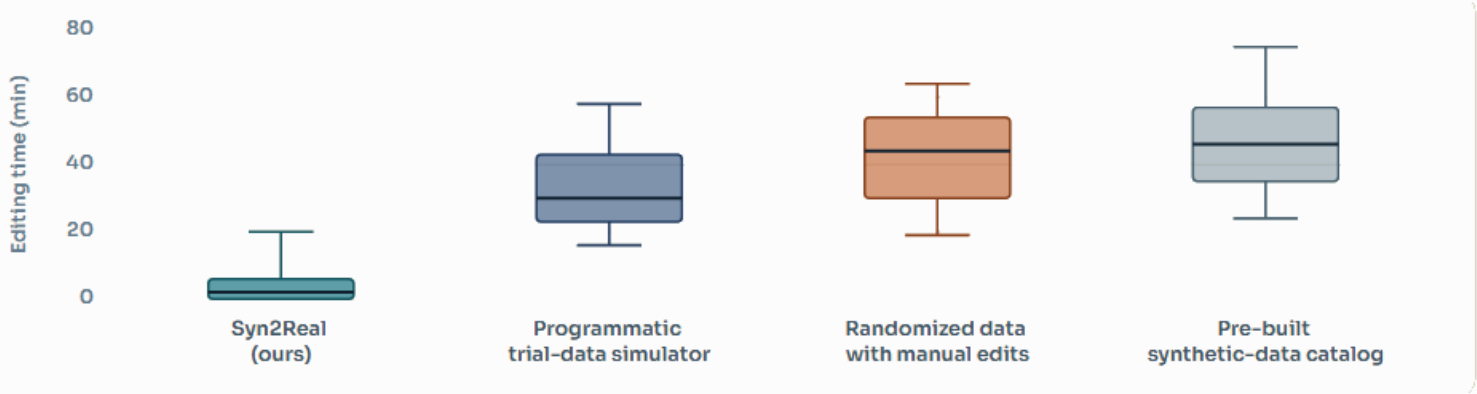
RELIABILITY

- Hung calls time out and retry automatically, so one stalled request can't hold up a run.
- Long runs survive credential expiry, and resume from the last completed step.

Next: a full-scale run with published cost and wall-clock figures, and formal qualification documentation.

WHAT IT DELIVERS

How much editing before your code runs



Same hypothetical TFL programs implemented against each data source, then run against held-out reference data. Edits counted until each program ran.

THE RESULT

Syn2Real needed the least editing.

The spread is tighter too: predictable effort, not an occasional nasty surprise.

A large, consistent gap, measured in an early internal pilot with AI-generated reference data. A full controlled study is the natural next step.