From Study Intent to USDM-Compliant Digital Protocol

What we learned building a multi-agent AI protocol authoring system under 21 CFR Part 11

Shekhar Thumake

CDISC US Interchange 2026, Denver

Presenters : Achin Wadhwa | Prasanna Kakade



Agenda

1. Why the protocol is a data problem
2. The three standards that make it solvable
3. How the agents were wired
4. Four things we learned
5. What it means for DDF adoption

The protocol is the first data artifact of a trial



Written as unstructured text

- Design decisions live inside sentences.
- Arms, epochs and visit windows are described, not declared.



Rebuilt downstream

- EDC, SDTM and registry teams each re-derive the same design by hand, from the same document.



Amendments ripple

- One design change becomes many manual edits across many systems, each on its own timeline.

This is not a writing problem. It is a **structured data problem**.

Which is why the fix is a data standard, not a better template.

Two things called a protocol



THE DOCUMENT

A human reads it

Section 4 states that the trial is randomised, double blind, with two arms and a 12 week treatment period.

A machine sees one paragraph of text.



THE DATA

A machine reads it

StudyArm, StudyEpoch, StudyCell, StudyElement, and a blindingSchema carrying the coded term C15228.

The same facts, addressable as fields.

Same trial. Only one of them can be validated, compared across versions, or handed to another system.

cdisc®

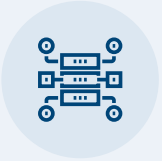
Three standards, three different jobs



ICH M11 (CeSHarP)

Step 4 Final, November 2025

Defines what a protocol must contain. Fourteen top level sections that cannot be deleted, renamed or reordered.



CDISC USDM 4.0

Unified Study Definitions Model

Defines how those contents are represented as data. Entity classes, relationships and controlled terminology.



CDISC CORE

Conformance rules engine

Decides whether the data is conformant. Over two hundred published DDF rules, executed deterministically.

M11 gives structure. USDM gives semantics. CORE gives proof.

How the agents were wired

Architecture, not magic

Four agents, four narrow jobs



Study Planning

Turns study intent and registry data into a design skeleton: arms, epochs and a first pass population.



Section Authoring

Drafts M11 section content against the template rules and the sponsor inputs it was given.



Data Architecture

Maps drafted content onto USDM entity classes and selects the coded terms.



Quality Validation

Runs conformance checks and flags what a human has to decide.

Plain asynchronous orchestration with a tool use loop. No agent framework was needed, and none was used.

Study intent to signed protocol

1. **Study intent**
   Sponsor inputs and public registry data
2. **Design skeleton**
   Arms, epochs, population
3. **M11 sections**
   Fourteen sections drafted
4. **USDM 4.0 JSON**
   Entity graph with coded terms
5. **Conformance gate**
   CORE rules, then signature

PROBABILISTIC

Language models draft. Every output is a proposal, never a record, and it carries a review flag.

DETERMINISTIC

Schema and CORE rules pass it or fail it.

ACCOUNTABLE

A named human signs. Nothing ships unsigned.

The pattern that mattered: every probabilistic step is followed by a deterministic one.

cdisc®

Four things we learned

The useful lessons were not the obvious ones

The hard part is the mapping

ONE CLAUSE OF STUDY INTENT

Randomised, double blind, two arms, 12 week treatment period.

Nothing here names an entity, a relationship or a code. All of it has to be inferred, then verified.

WHAT IT HAS TO BECOME IN USDM

- StudyArm (x2)
- StudyEpoch
- StudyCell
- StudyElement
- InterventionalStudyDesign.model
- blindingSchema Code C15228
- Duration quantity and unit
- Linked list pointers on each epoch

Six entity classes, one coded term, and pointer sets that have to stay bidirectional.

Fluent prose was never the bottleneck. A valid entity graph was. Cross entity references fail quietly: an epoch linked list with one bad pointer still reads perfectly as narrative.

cdisc®

Four guardrails that contained hallucination



Constrain the vocabulary

Coded fields select from CDISC controlled terminology and NCI C-codes. The model picks from a list instead of inventing a code, and every code is verified against NCI EVS rather than recalled.



Generate into a schema

Strict typed models reject a malformed entity before it can reach the database. A hallucinated attribute fails immediately instead of persisting quietly.



Let rules decide conformance

CORE evaluates the exported USDM against the published DDF rules. The model gets no vote on whether its own output is valid.



Separate author from approver

Every agent is a distinct system user with its own permissions. An agent that drafts content is not permitted to approve it.

Audit trail is a design input

Every generation is an audited event



Append only. Database triggers abort any attempt to update or delete a recorded event.

Where AI puts the most pressure on ALCOA+

- Attributable: which agent
- Contemporaneous: server clock
- Original: the event is the record
- Complete: middleware, not habit

What an AI event must record

- Which agent, and which model version, produced it
- The input context it was given
- Its reasoning chain and any tool calls
- A hash of the output content
- A processed_by attribution and a human review flag

ALCOA+ was not a checklist bolted on afterwards. **It decided the shape of the event schema.**

cdisc®

Human review is a property, not a phase



No AI generated element becomes part of an approved protocol without a named human signing for it.

LOW RISK

Timelines, headings, administrative content.

Logged and attributed. No signature.

MEDIUM RISK

Background sections, visit schedules.

Reviewer sign off before it can advance.

HIGH RISK

Endpoints, eligibility, dosing, schedule of activities.

Mandatory electronic signature. Hard stop.

An honest scorecard

Where it genuinely helped

- Structural completeness. All fourteen sections exist in every output, marked not applicable where that is the truth.
- Terminology consistency. Coded fields stay consistent in a way manual authoring rarely achieves at scale.
- Traceability by construction. Provenance is a by-product of the design, not an extra deliverable.
- First draft speed on the mechanical sections.

Where it did not

- Clinical judgement. Dose rationale and design trade-offs are not in the inputs and cannot be inferred from them.
- Novel design rationale. It reproduces patterns. It does not justify them.
- Anything requiring sponsor context outside the supplied inputs.
- Self assessment. Left unchecked it will produce a plausible, well formed, wrong entity.

Three things to take back

1

Design backwards from the model

If USDM is the deliverable, do not author a document and convert it later. Decide the entity graph first and let the narrative follow from it.

2

Put a deterministic gate after every probabilistic step

A language model should never be the last thing that touches a regulated record. Schemas, conformance rules and signatures come after it.

3

Budget for review, not only for generation

Generation turned out to be the cheap part. Adoption depends on whether reviewers trust what they are handed.

Questions?

