verisian

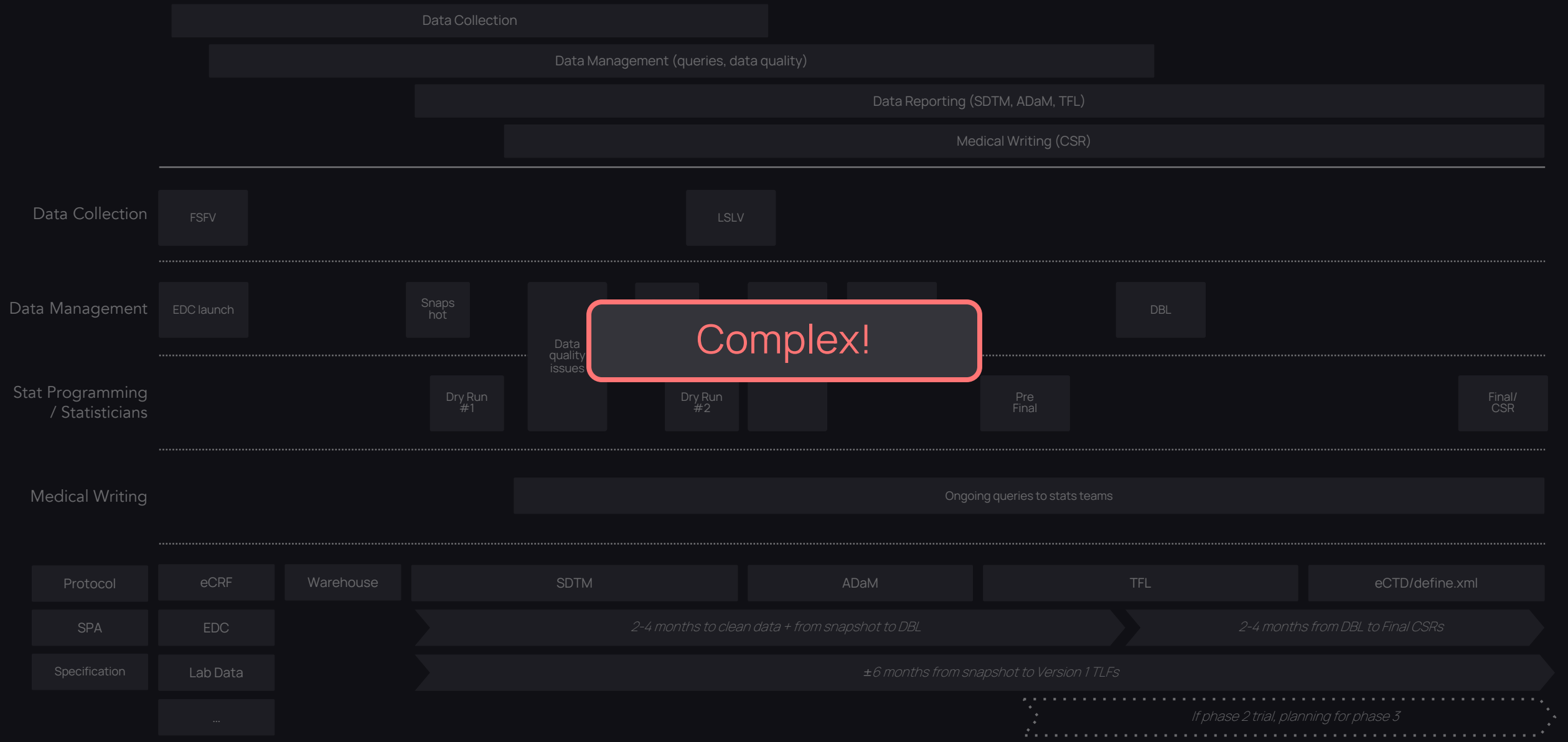
Traceability and AI for Better Understanding, Communication, and QC of Trials

Tomás Sabat Stöfsel

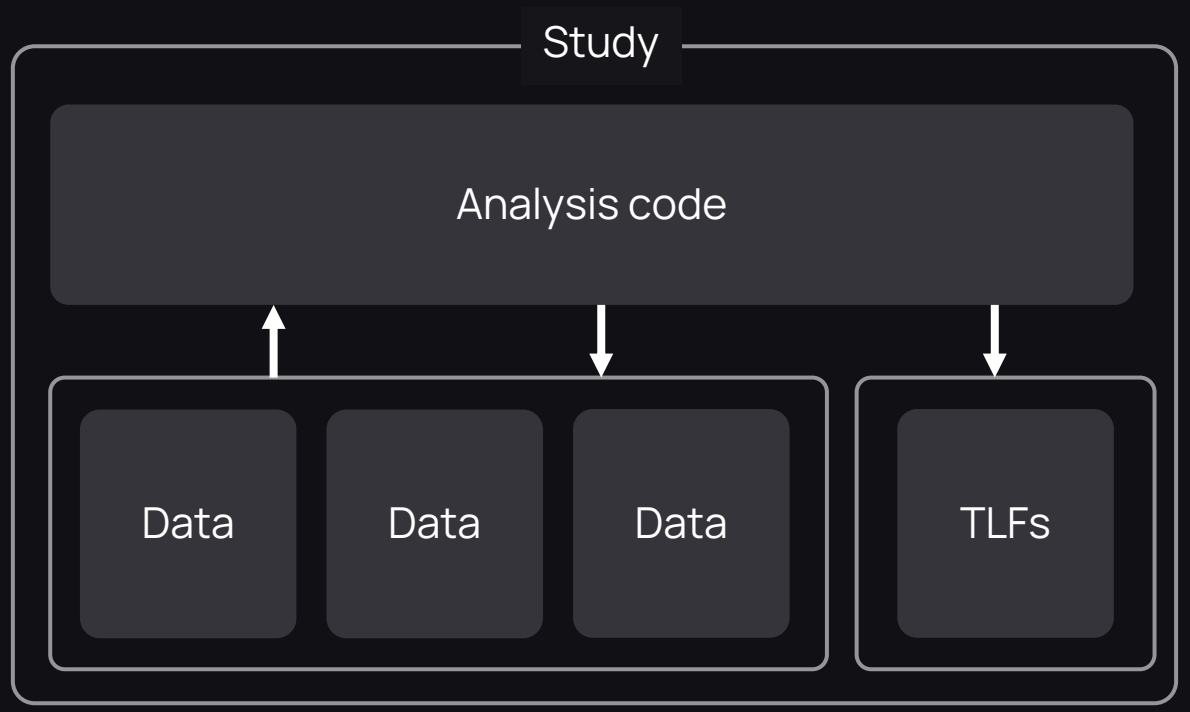
CEO & Co-Founder, Verisian

Why can’t we release a drug immediately after the last pivotal trial?

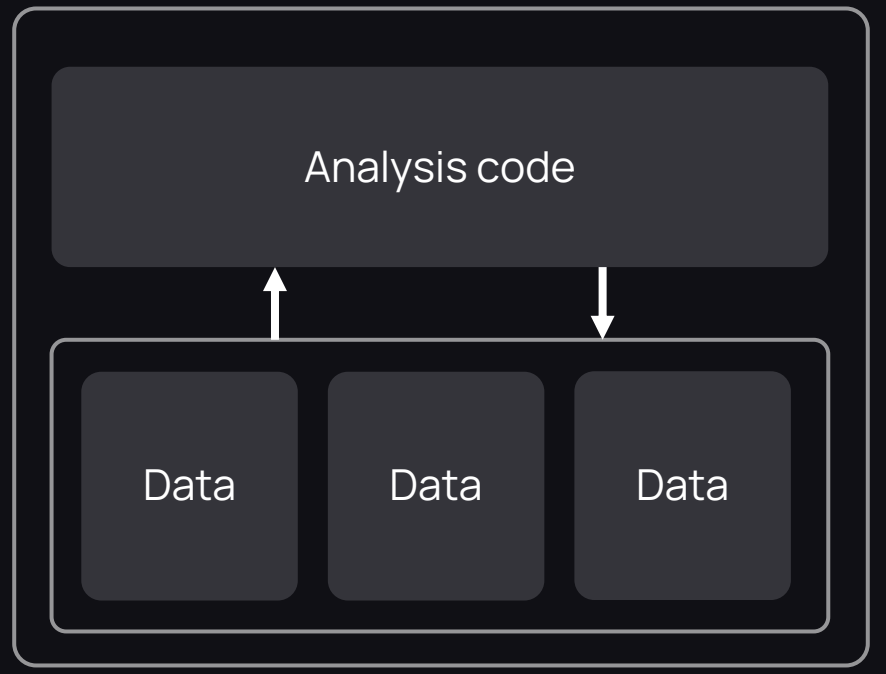
Drug submission data workflow



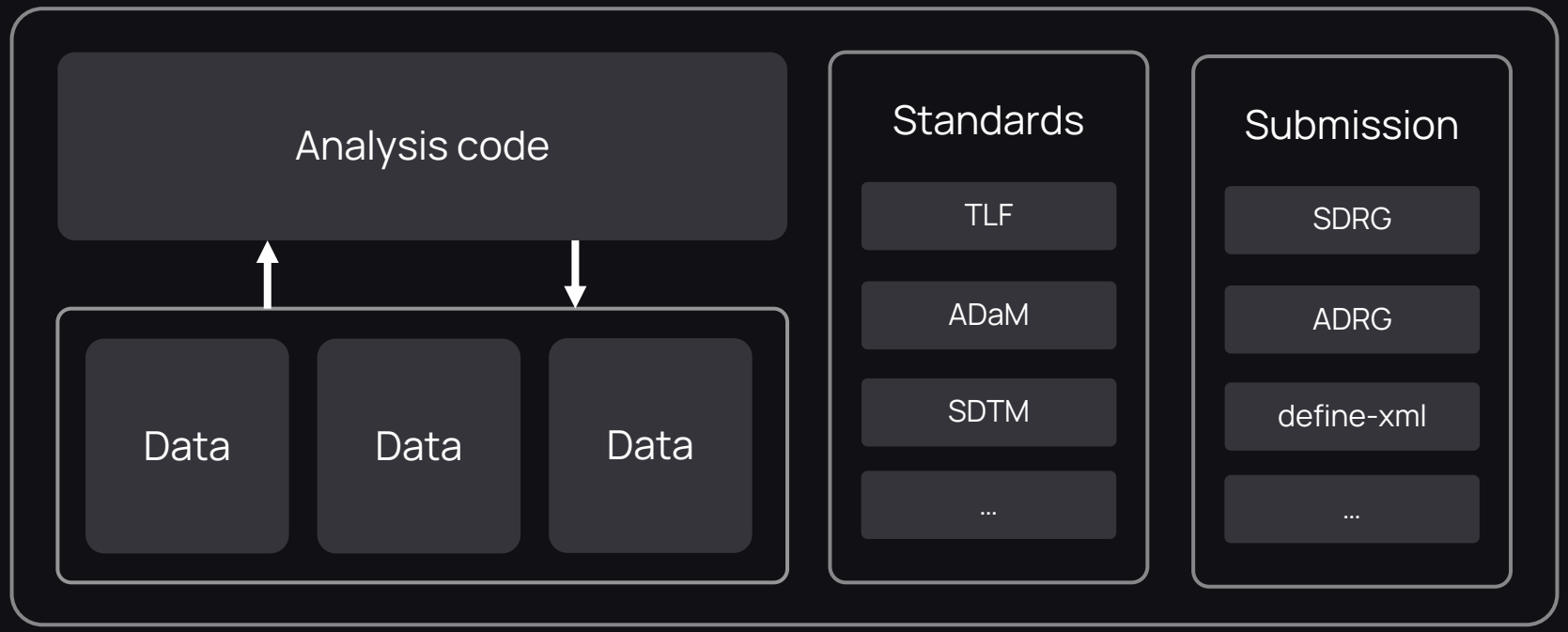
The challenge is to understand data and code, and how they interact



Once data are collected: what is a clinical trial?



Once data are collected: what is a clinical trial?

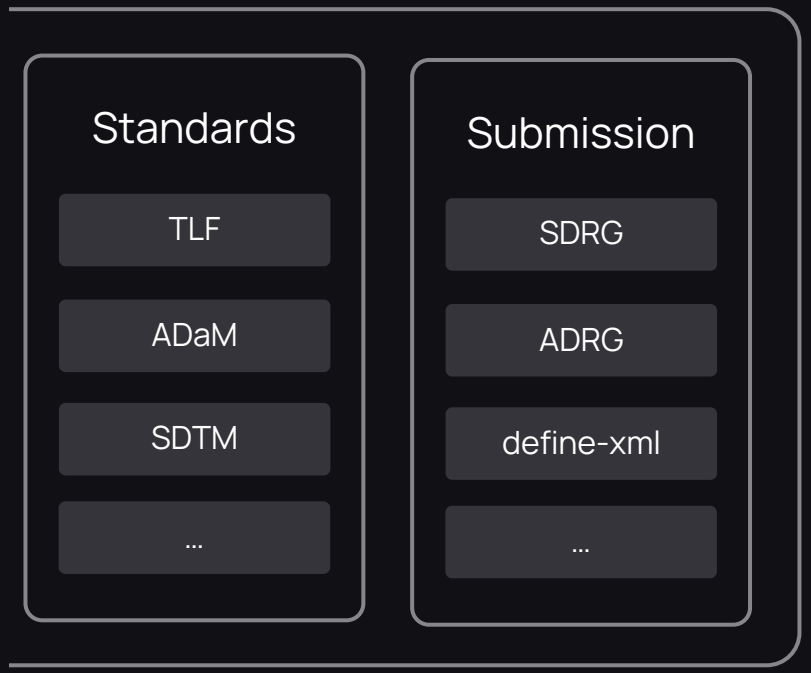


Why data standards?

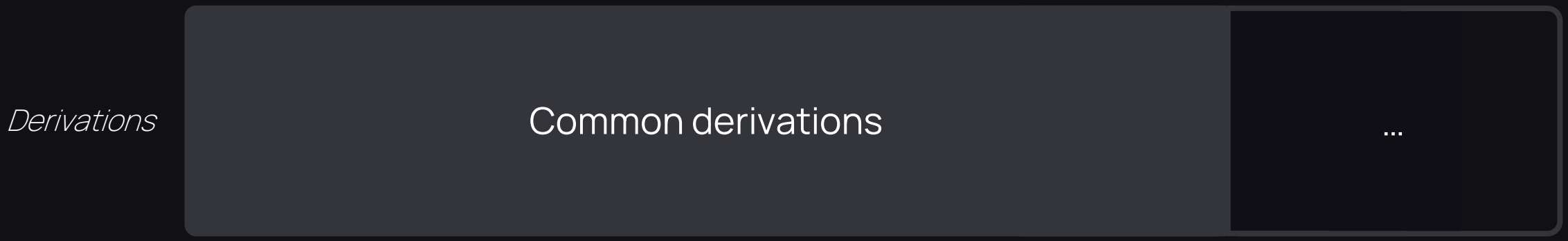
Standardize data structure

Structural study understanding

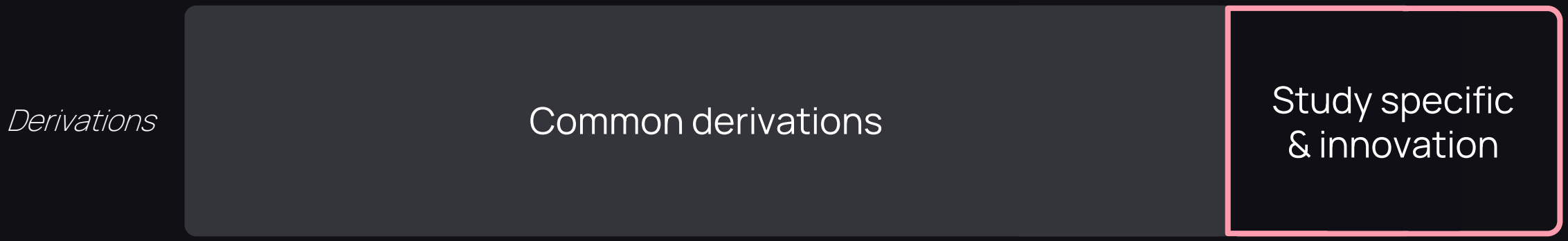
Manual metadata traceability



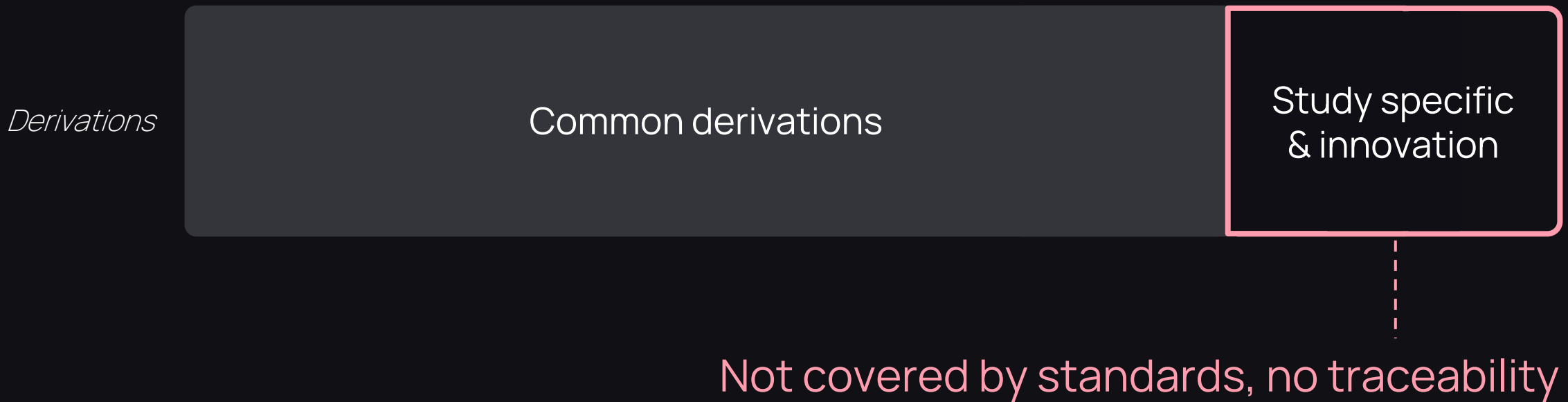
Most derivations in a study are standardized



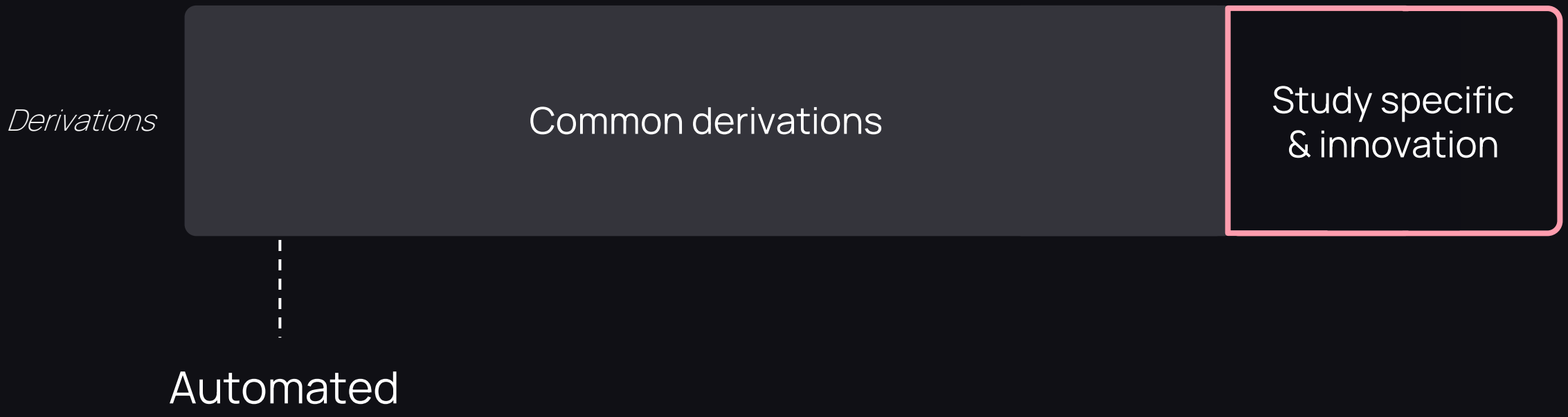
Most derivations in a study are standardized



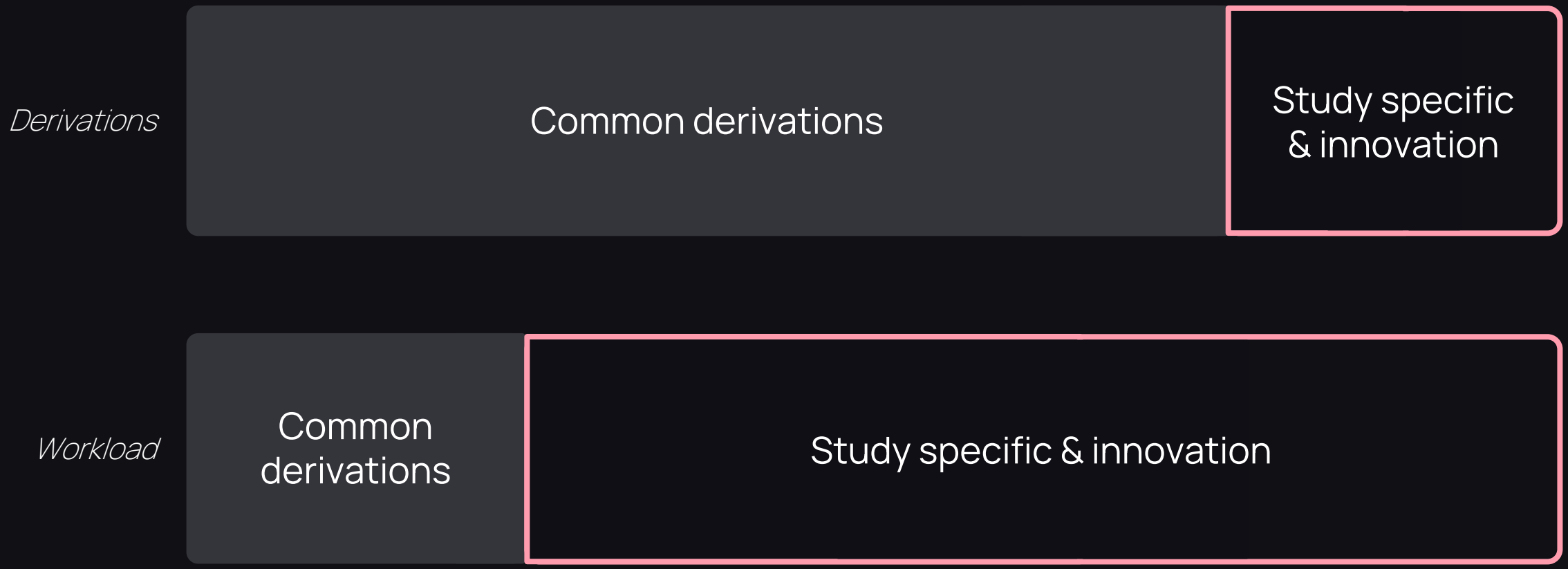
Most derivations in a study are standardized



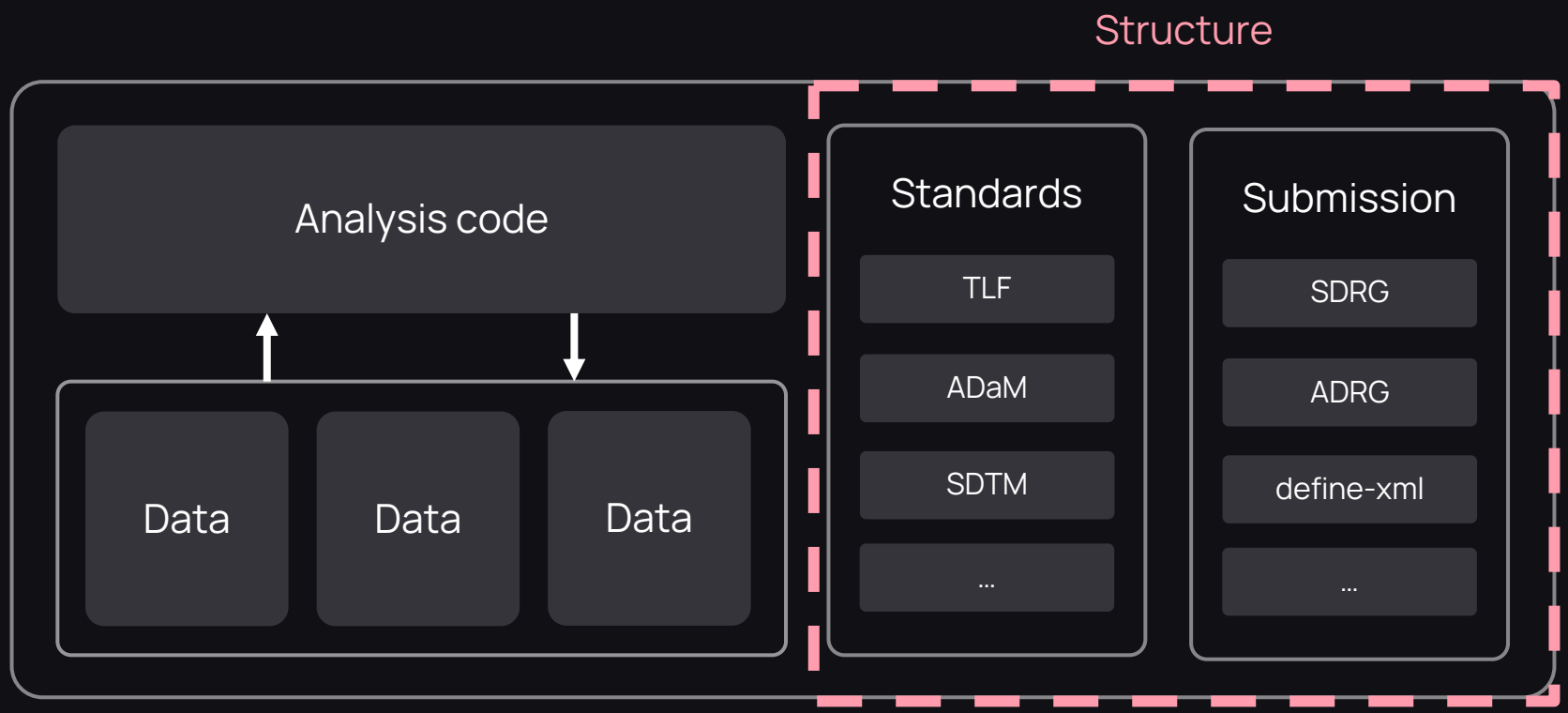
Most derivations in a study are standardized



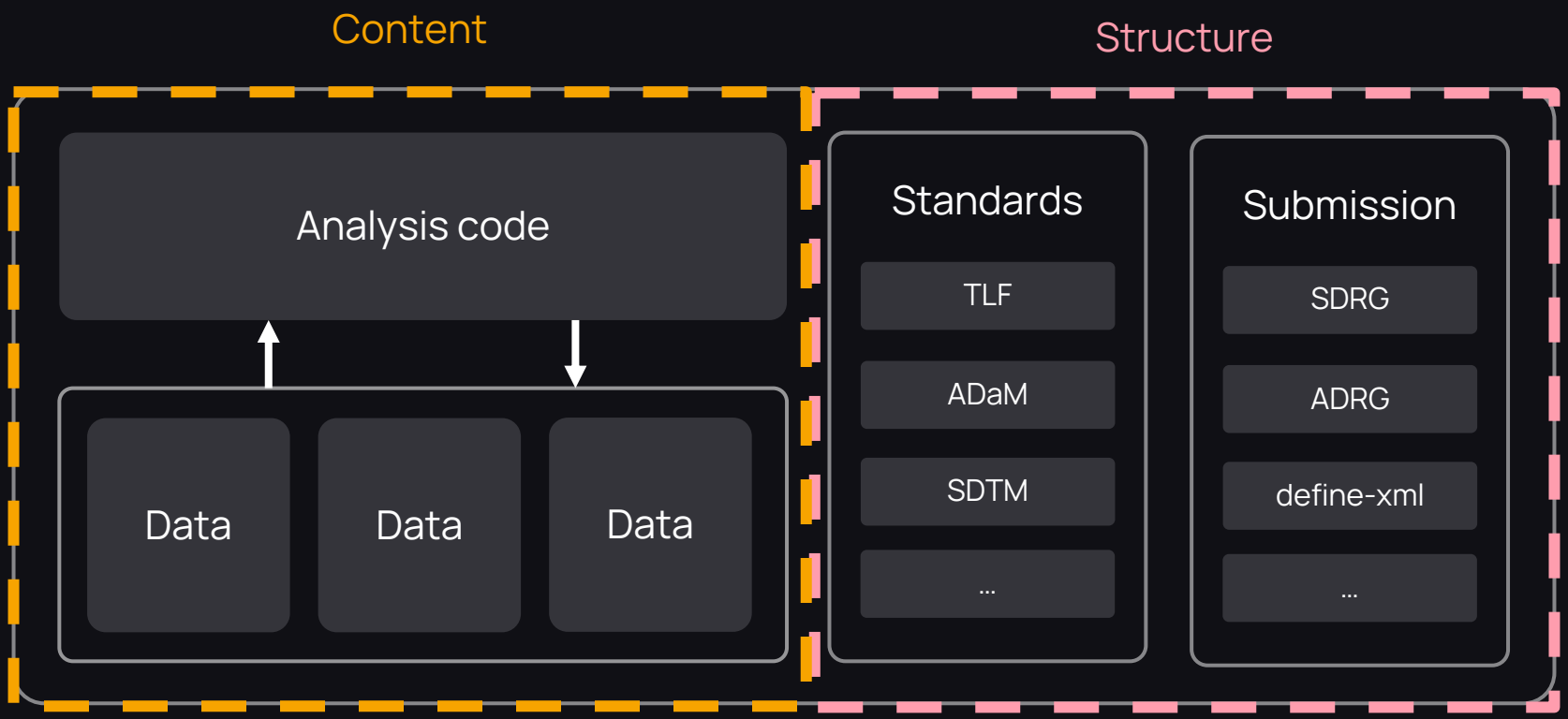
Most derivations in a study are standardized



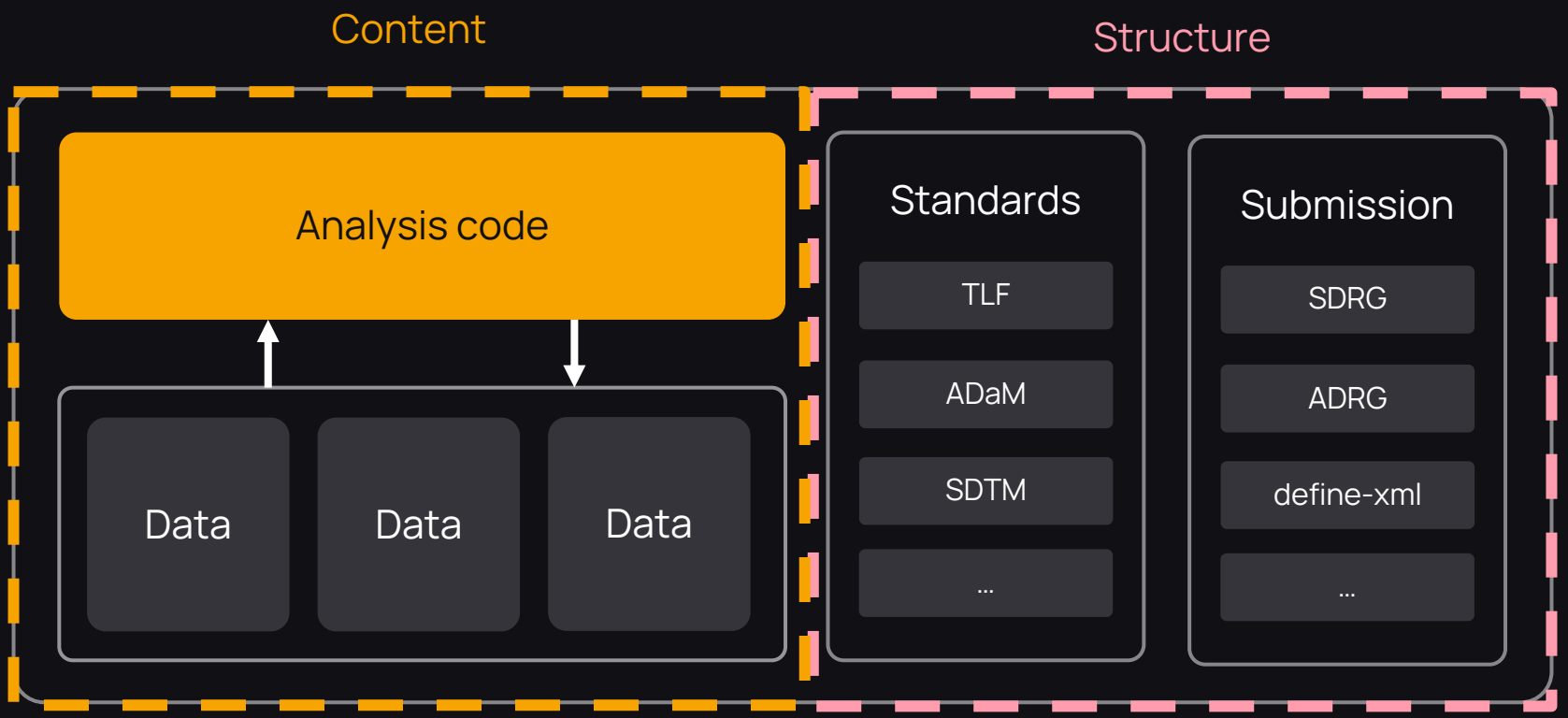
What is a clinical trial?



What is a clinical trial?



The source of truth of content of a trial analysis is its code



What matters is the content of the study, not the structure

Non-standard, study-specific, scientific innovation

Code traceability

Tracing the content interactions between data, code and results

Code traceability

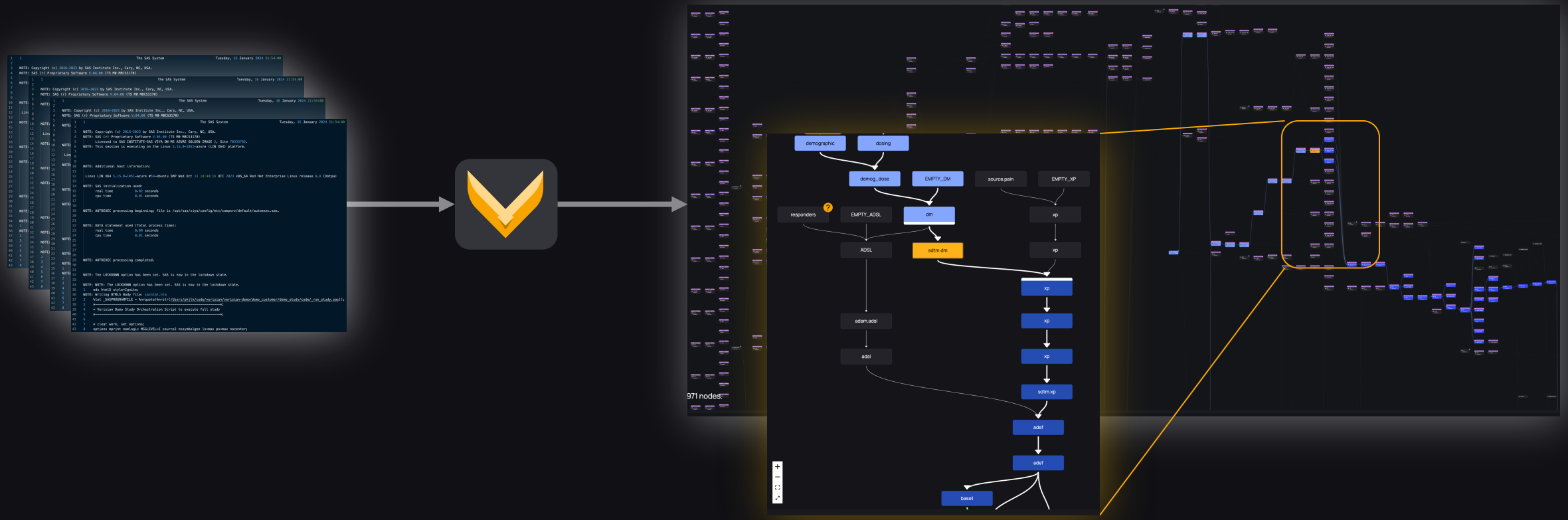
Tracing the content interactions between data, code and results

Independent of standards and languages

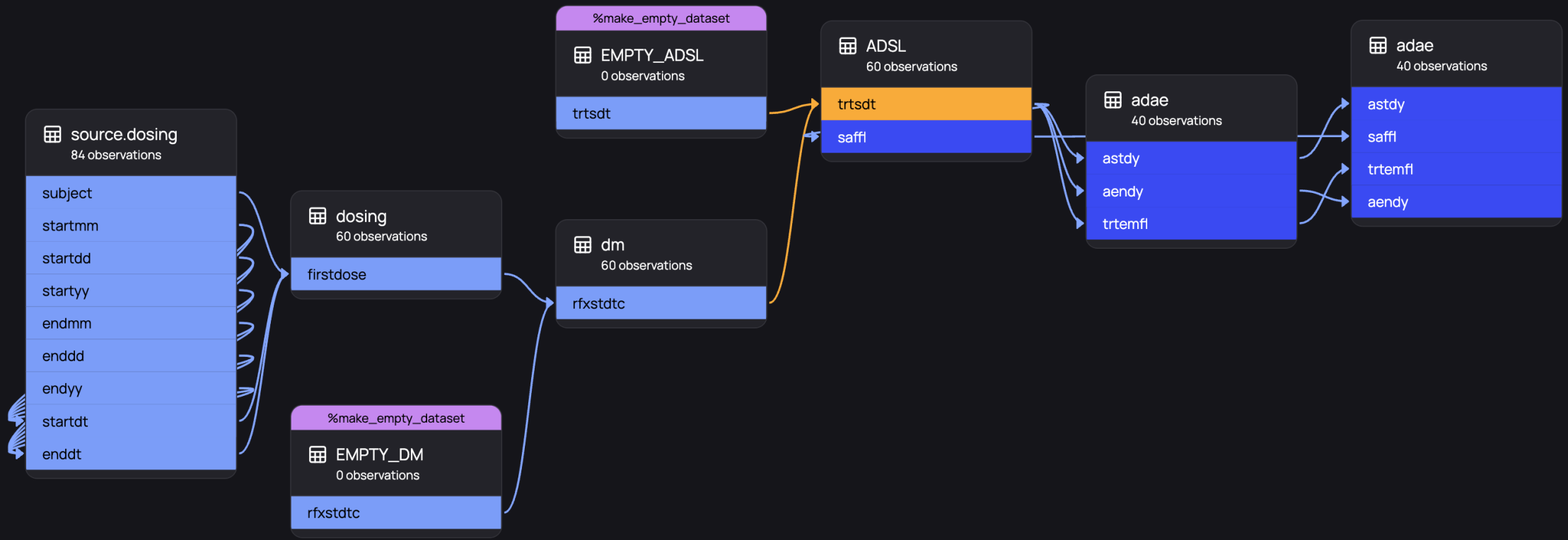
Code traceability

verisian

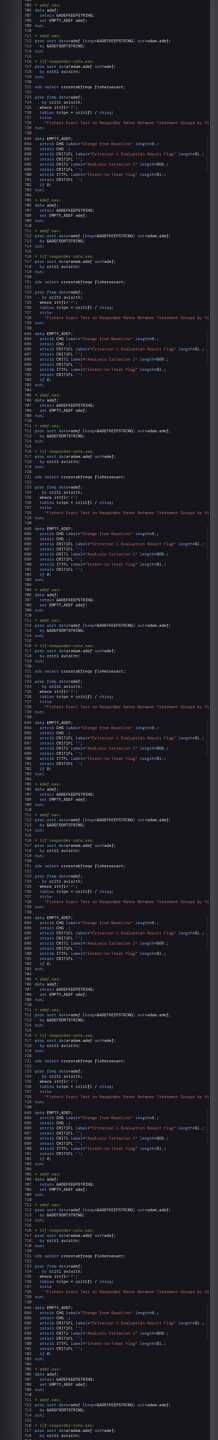
All you need are logs...



Dataset Graph | Overview Graph | **Variable Graph**



8 nodes

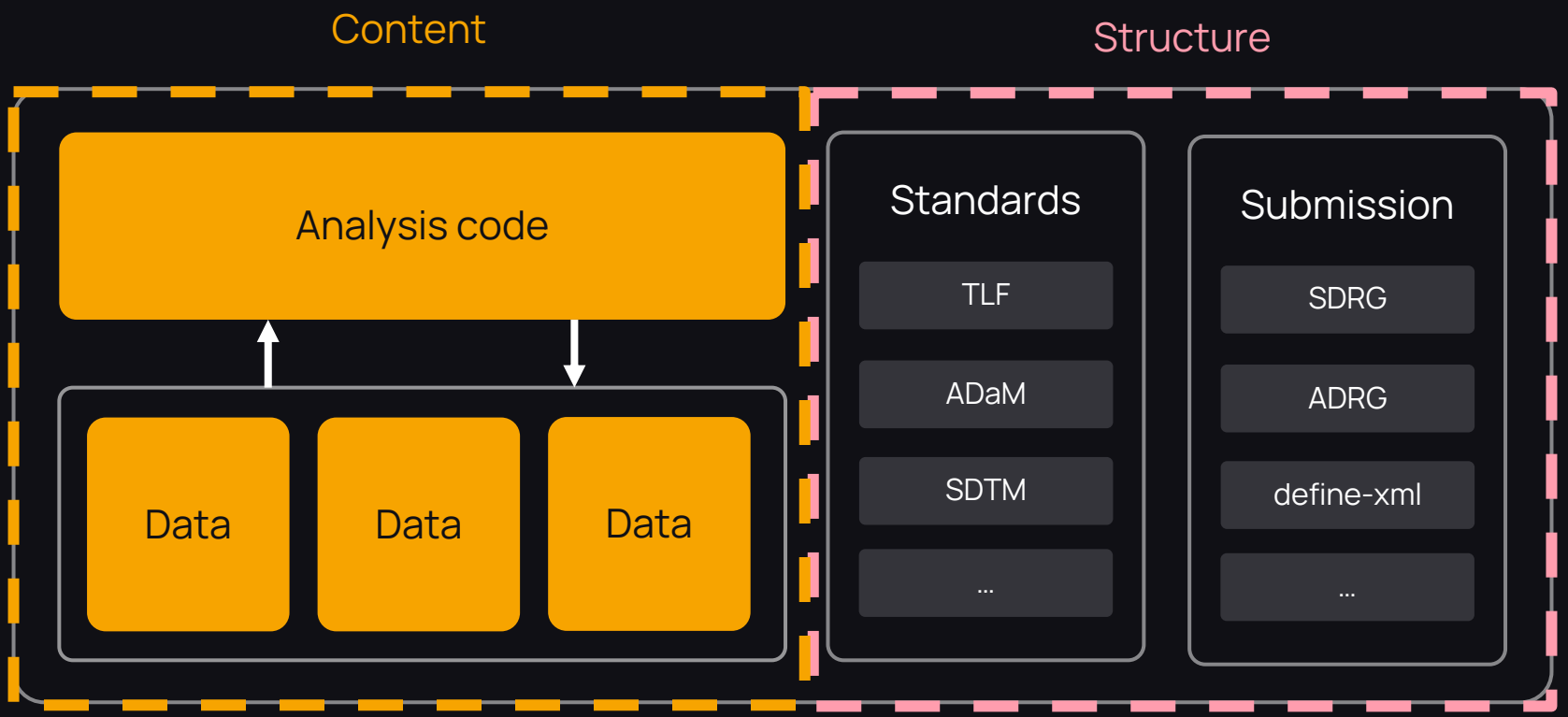


From 700+ lines to 37 lines

```sas
* demographic.sas;
data source.demographic;
 . label trt="Treatment";
  input trt 5;
  *...;
run;

* DM.sas;
data dm;
  *...;
  arm=put(trt, arm_demographic_trt.);
 . *...;
run;

* adsl.sas;
data ADSL;
  *..;
  trt01p=ARM;
  *...;
  trt01pn=input(put(trt01p, $trt01pn.), best.);
  *...;
run;

* adef.sas;
data adef;
  * ...;
  rename trt01pn=trtpn;
  *...;
run;

proc freq data=adef;
  by crit1 avisitn;
  where ittfl='Y';
  tables trtpn * crit1fl / chisq;
  title
    "Fishers Exact Test on Responder Rates Between Treatment Groups by Vi
run;

```

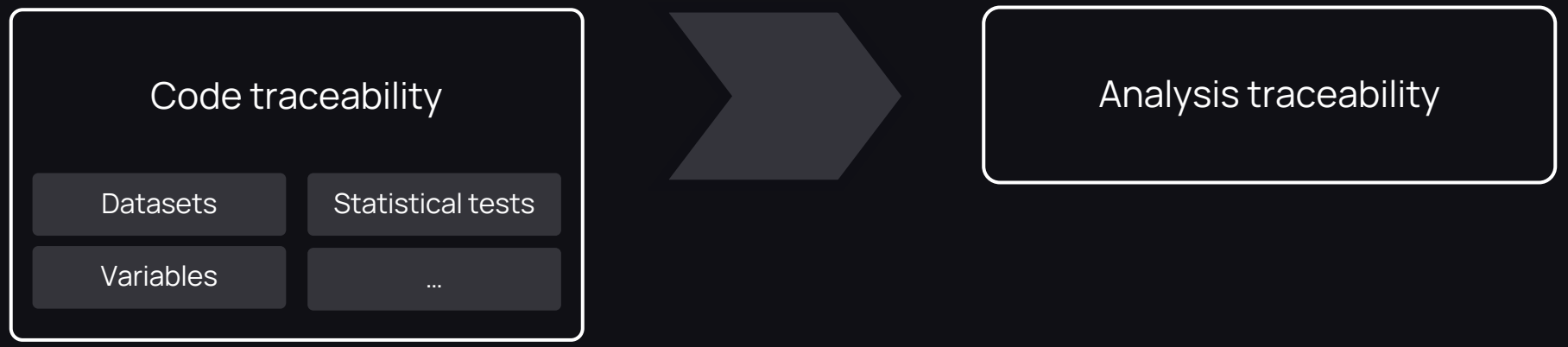
95% time savings on ADaM-level inquiries

The source of truth of content of a trial is its code and data



How to go from code traceability to analysis traceability?

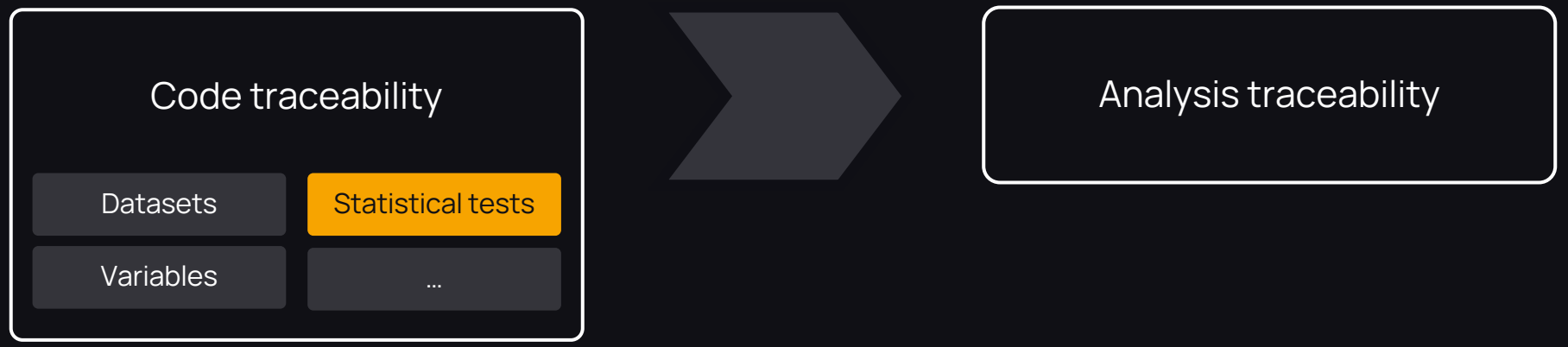
How to go from code traceability to analysis traceability?



How to go from code traceability to analysis traceability?

How to go from code traceability to analysis traceability?

What does code traceability give us for analysis traceability?



Validate a result's statistical assumptions

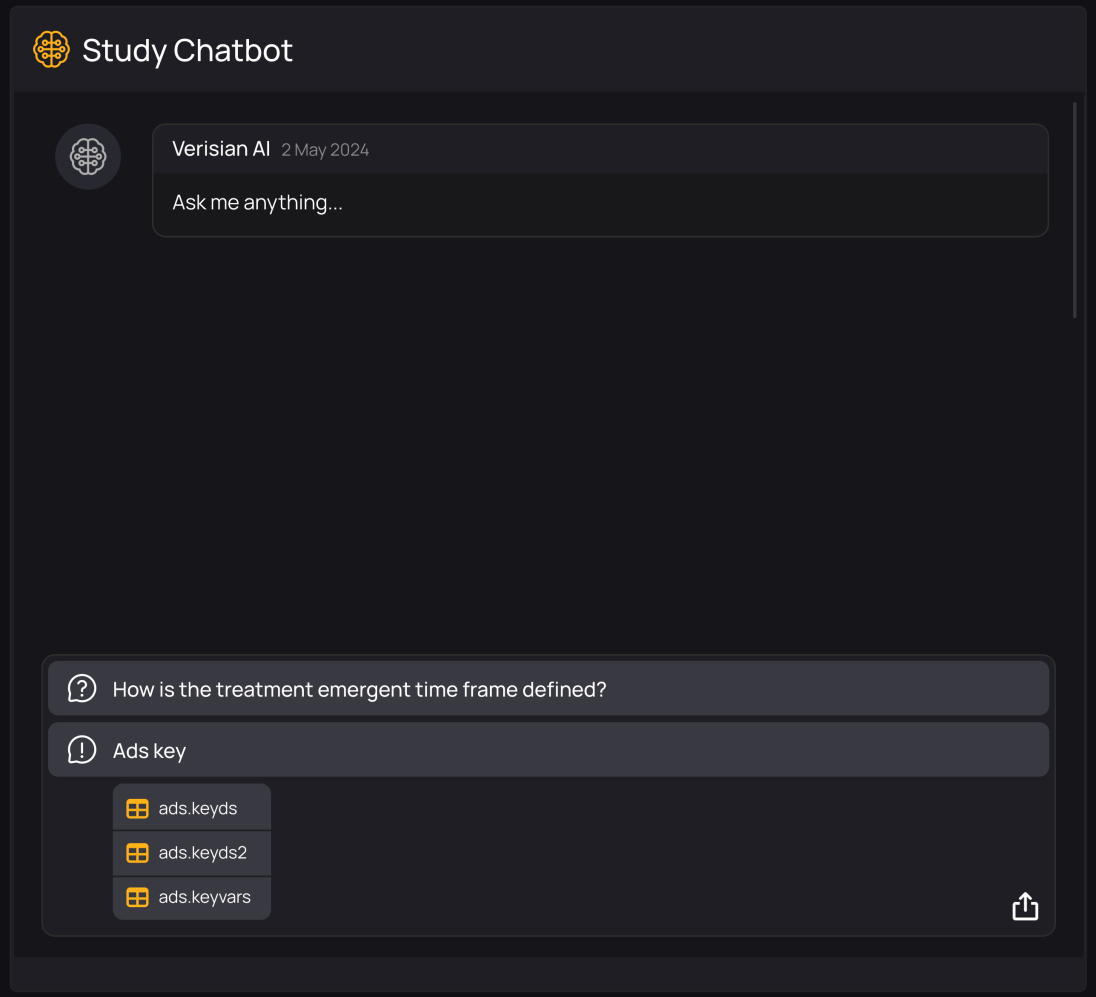
Know variables and their role in the test

Know the assumptions of the test

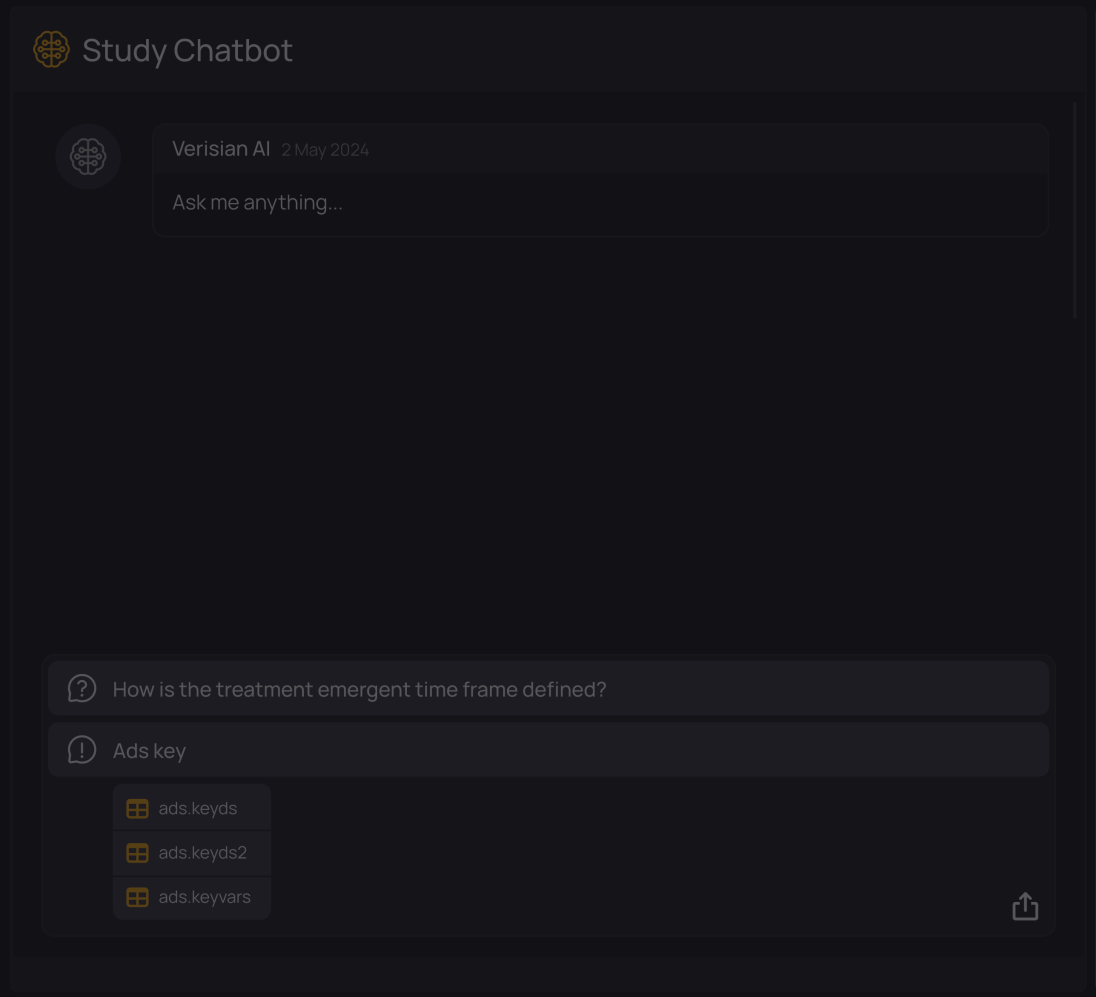
Validate that test assumptions are met in data

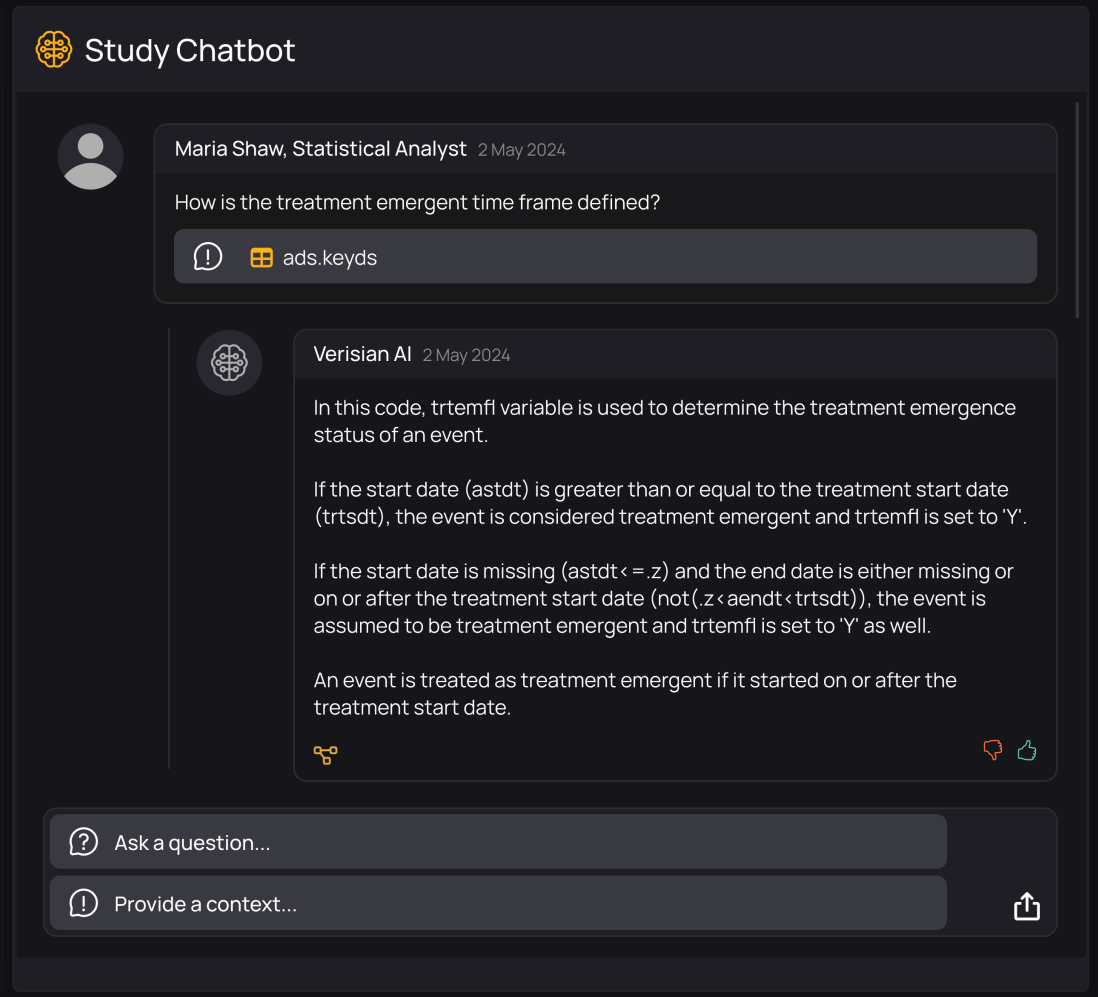
Explore studies using natural language, starting anywhere

Verisian AI: Traceability-aware & Evidence-based



Verisian AI: Traceability-aware & Evidence-based

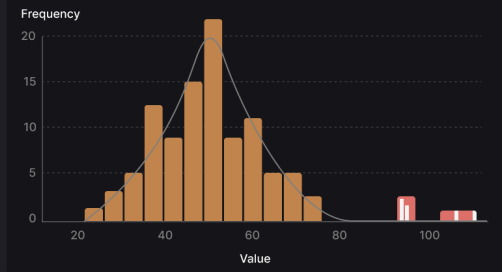




Verisian AI: Traceability-aware & Evidence-based

Data ADaM > adef > CONTVAR

ADaM.CONTVAR



Summary

Demographic Data Input (demographic.sas)
- Reads treatment data into the dataset source.demographic and labels the variable trt as "Treatment".

Demographic Transformation (DM.sas)
- Converts the trt variable into a formatted string arm using a format arm_demographic_trt..

Assign Treatment Codes (adsl.sas)
- Checks if arm is missing; if so, assigns it a value of 0.
- Assigns the value of arm to trt01p.
- Converts the formatted string trt01p to a numeric variable trt01pn using a format Strt01pn..

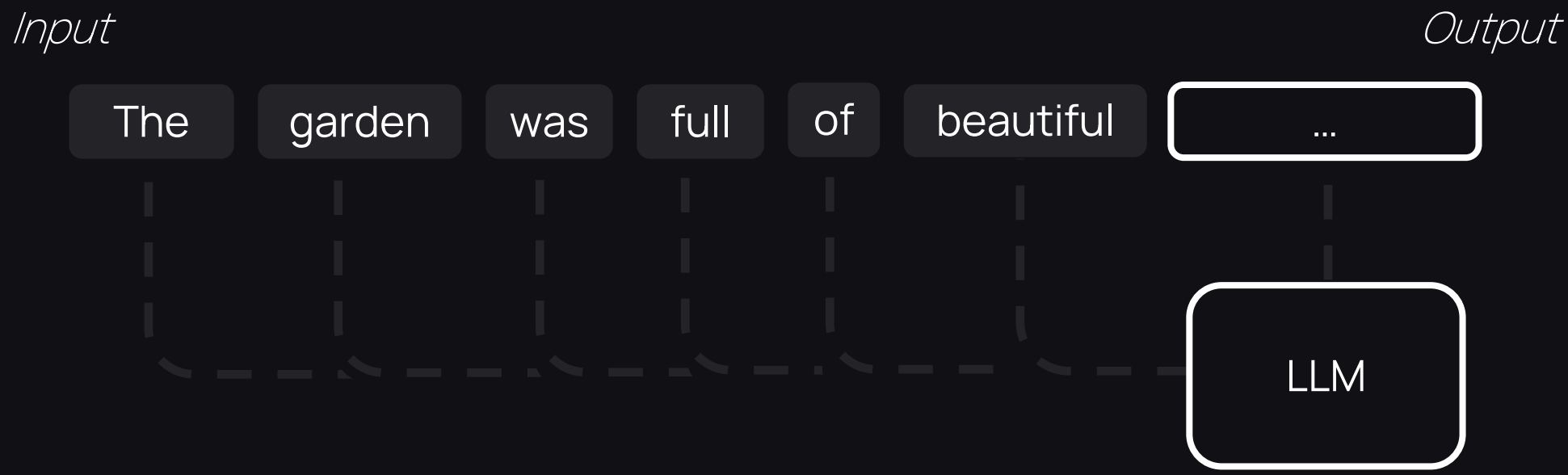
Rename Variable for Analysis (adef.sas)
- Renames the variable trt01pn to trtpn in the dataset adef to standardize the variable name for further analysis.

Derivation

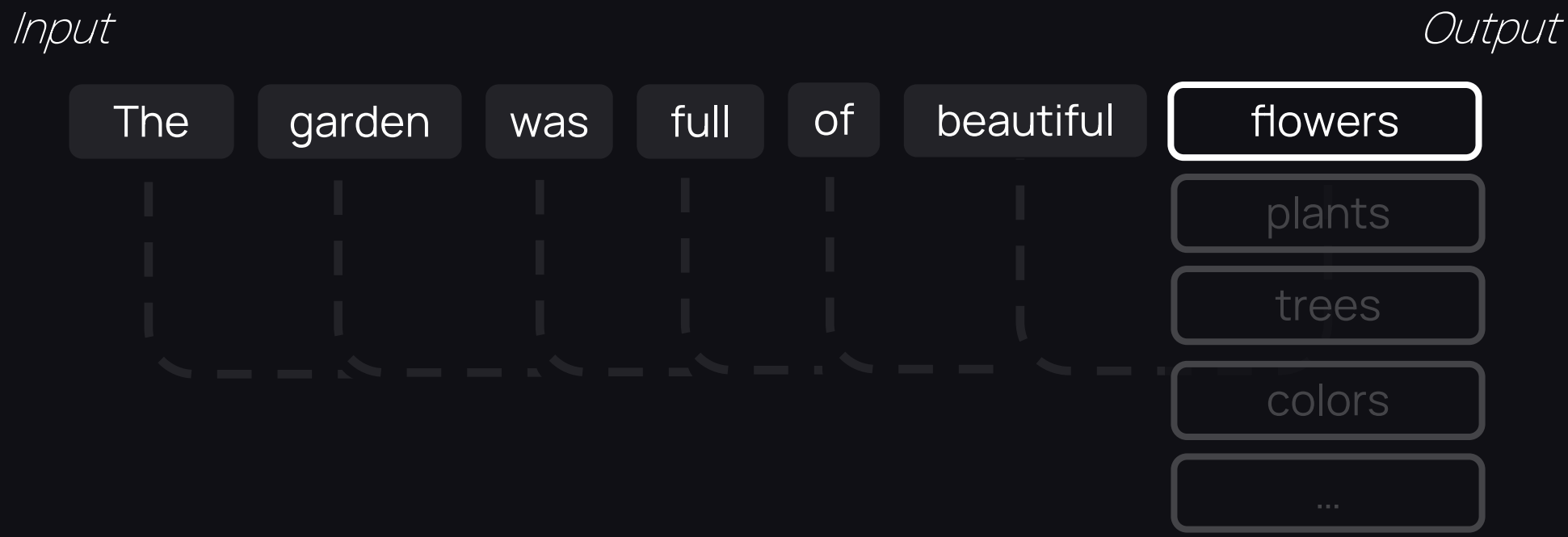
(Macro-resolved)

```
1  * demographic.sas;
2  data source.demographic;
3   . label trt="Treatment";
4    input trt 5;
5    *...;
6  run;
7
8  * DM.sas;
9  data dm;
10   *...;
11   arm=put(trt, arm_demographic_trt.);
12  . *...;
13 run;
14
15 * adsl.sas;
16 data ADSL;
17   *.;
18   trt01p=ARM;
19   *...;
20   trt01pn=input(put(trt@1p, $trt01pn.), best.);
21   *...;
22 run;
23
24 * adef.sas;
25 data adef;
26   * ...;
27   rename trt01pn=trtpn;
28   *...;
29 run;
```

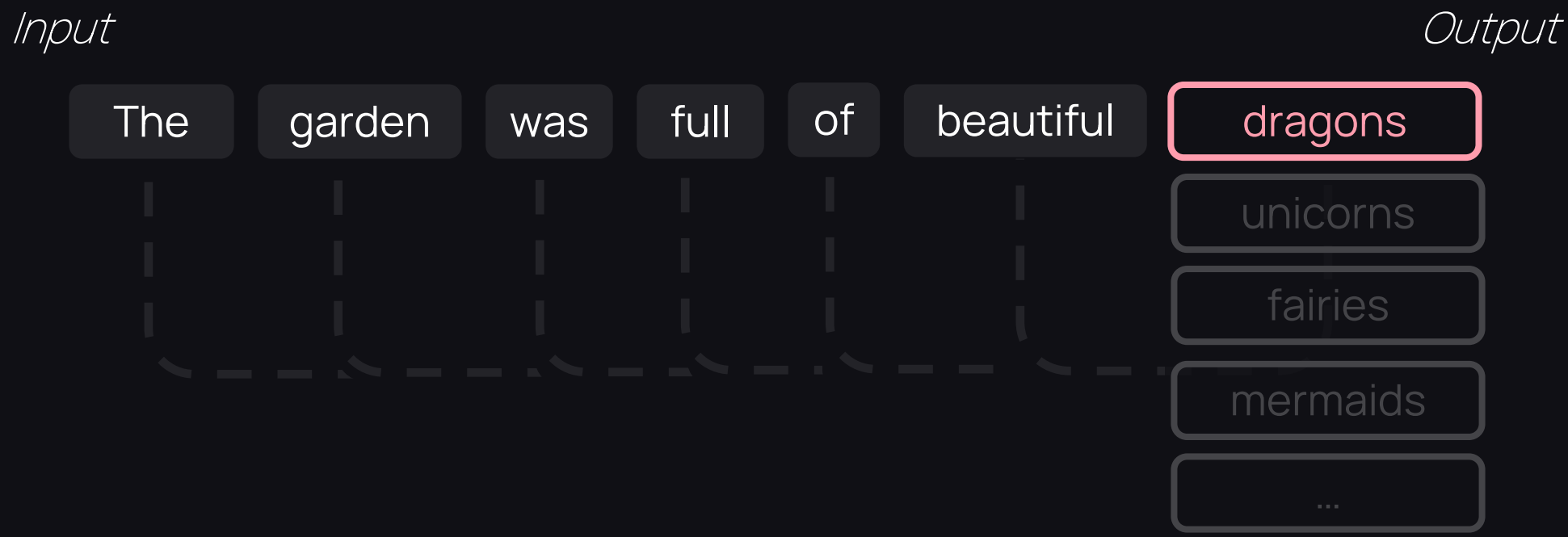
LLMs are really sophisticated autocompletes



LLMs are really sophisticated autocompletes



The bigger the question, the likelier the hallucination



Ethics and Information Technology (2024) 26:38
https://doi.org/10.1007/s10676-024-09775-5

ORIGINAL PAPER

ChatGPT is bullshit

Michael Townsen Hicks[1] · James Humphries[1] · Joe Slater[1]

Abstract
Recently, there has been considerable interest in large language models: machine learning systems which produce human-like text and dialogue. Applications of these systems have been plagued by persistent inaccuracies in their output; these are often called "AI hallucinations". We argue that these falsehoods, and the overall activity of large language models, is better understood as *bullshit* in the sense explored by Frankfurt (On Bullshit, Princeton, 2005): the models are in an important way indifferent to the truth of their outputs. We distinguish two ways in which the models can be said to be bullshitters, and argue that they clearly meet at least one of these definitions. We further argue that describing AI misrepresentations as bullshit is both a more useful and more accurate way of predicting and discussing the behaviour of these systems.

Keywords Artificial intelligence · Large language models · LLMs · ChatGPT · Bullshit · Frankfurt · Assertion · Content

So how can we gain confidence in AI?

manual review of answers, but...

We solve the same problem for AI that we solve for you

dataset_derivation.sas U ×

variable-level-traceability › POC › dataset_derivation.sas › ...

```
645 data EMPTY_ADEF;
694   attrib CHG label="Change from Baseline" length=8.;
695   retain CHG .;
696   attrib CRIT1FL label="Criterion 1 Evaluation Result Flag" length=$1.;
697   retain CRIT1FL '';
698   attrib CRIT1 label="Analysis Criterion 1" length=$60.;
699   retain CRIT1FL '';
700   attrib ITTFL label="Intent-to-Treat Flag" length=$1.;
791   retain CRIT1FL '';
702   if 0;
703 run;
704
705 * adef.sas;
706 data adef;
707   retain &ADEFKEEPSTRING;
708   set EMPTY_ADEF adef;
709 run;
710
711 * adef.sas;
712 proc sort data=adef (keep=&ADEFKEEPSTRING) out=adam.adef;
713   by &ADEFSORTSTRING;
714 run;
715
716 * tlf-responder-rate.sas;
717 proc sort data=adam.adef out=adef;
718   by crit1 avisitn;
719 run;
720
721 ods select crosstabfreqs fishersexact;
722
723 proc freq data=adef;
724  . by crit1 avisitn;
725   where ittfl='Y';
726   tables trtpn * crit1fl / chisq;
727   title
728     "Fishers Exact Test on Responder Rates Between Treatment Groups by Vi
729 run;
730
```

variable_derivation_trtpn_VALUE.sas U ×

variable-level-traceability › POC › trtpn › variable_derivation_trtpn_VALUE.sas › DATA

```
1  * demographic.sas;
2  data source.demographic;
3   . label trt="Treatment";
4    input trt 5;
5    *...;
6  run;
7
8  * DM.sas;
9  data dm;
10   *...;
11   arm=put(trt, arm_demographic_trt.);
12  . *...;
13 run;
14
15 * adsl.sas;
16 data ADSL;
17   *..;
18   trt01p=ARM;
19   *...;
20   trt01pn=input(put(trt@1p, $trt01pn.), best.);
21   *...;
22 run;
23
24 * adef.sas;
25 data adef;
26   * ...;
27   rename trt01pn=trtpn;
28   *...;
29 run;
30
31 proc freq data=adef;
32   by crit1 avisitn;
33   where ittfl='Y';
34   tables trtpn * crit1fl / chisq;
35   title
36     "Fishers Exact Test on Responder Rates Between Treatment Groups by Vi
37 run;
38
```

Traceability-enabled AI

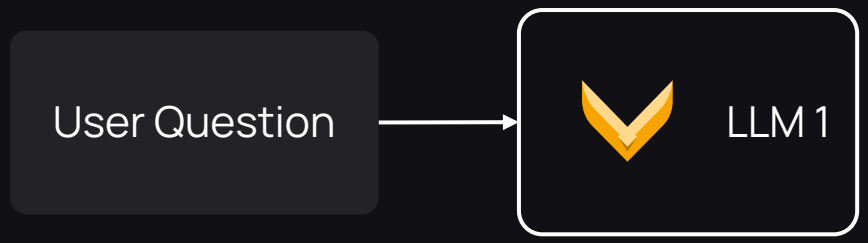
Reproducible, evidence based

Given any natural language prompt...

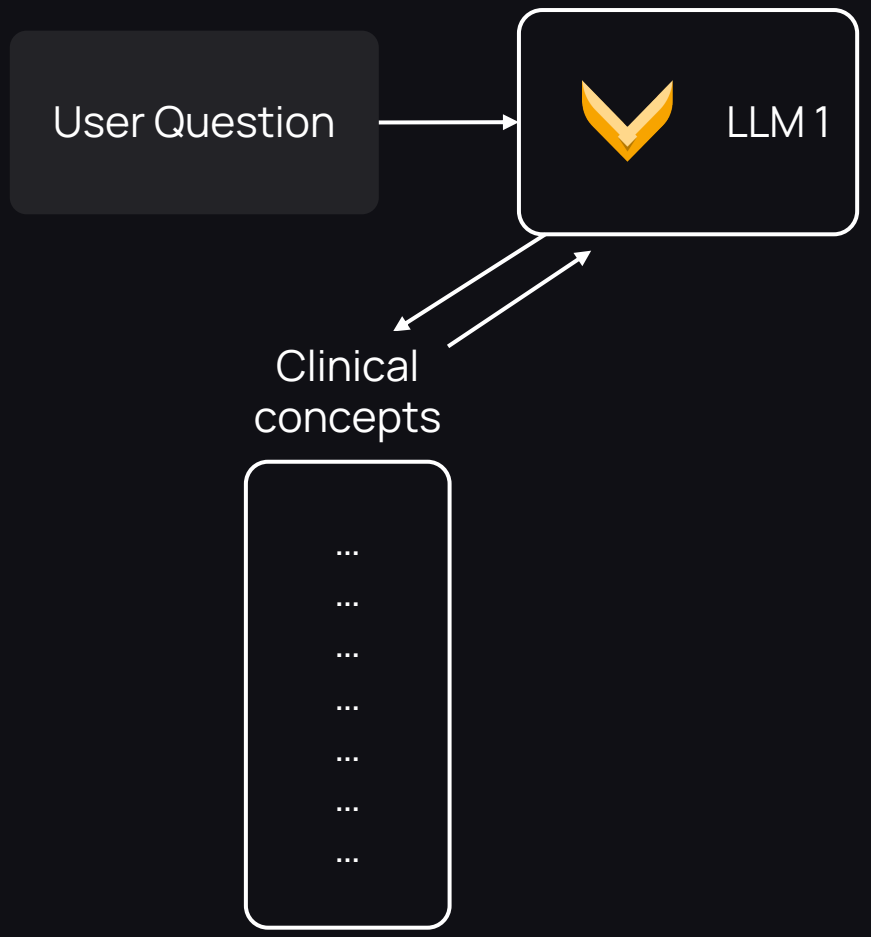
User Question

```
How is time to event
determined in this study?
```

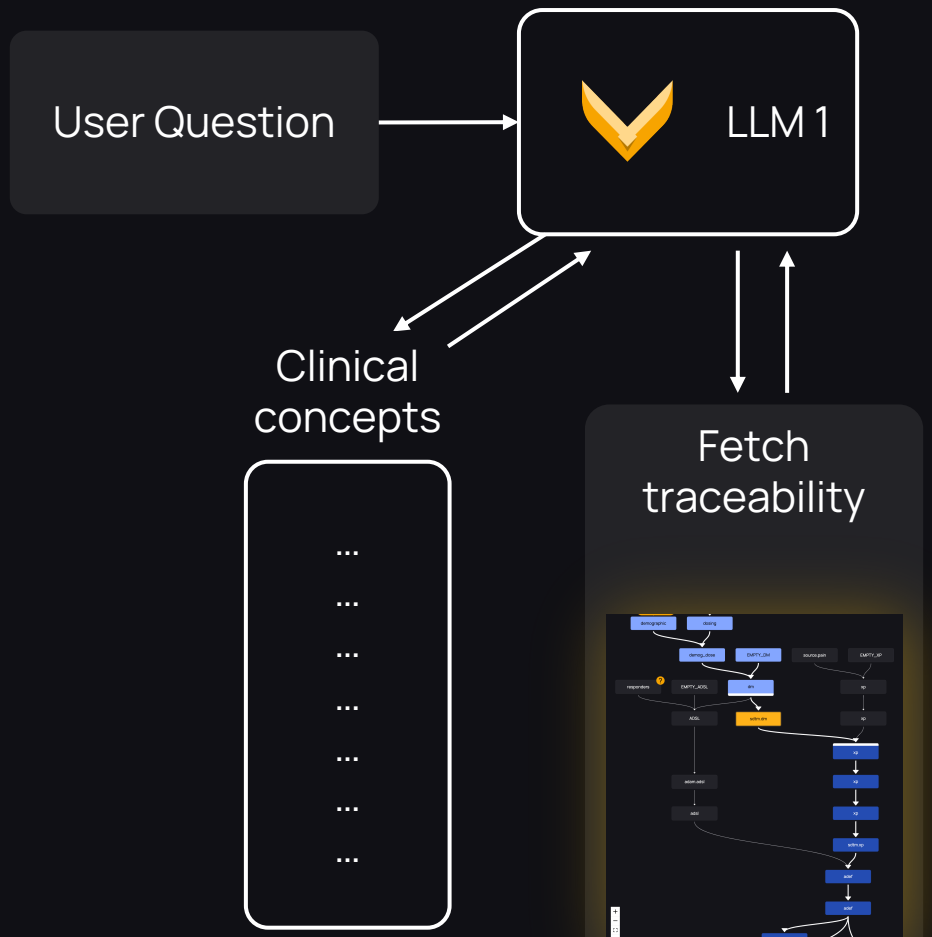
...and provide evidence to troubleshoot AI outputs.



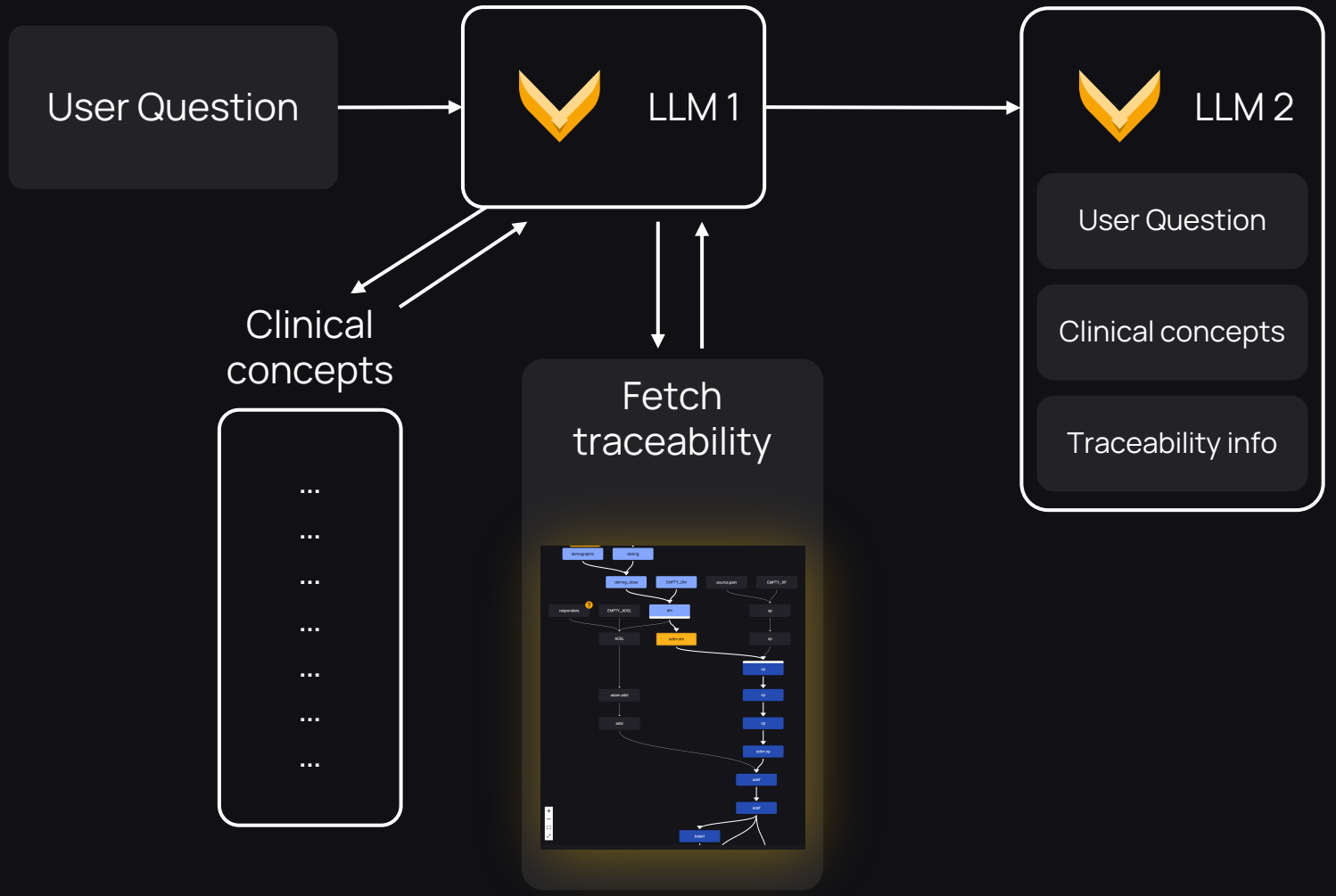
...and provide evidence to troubleshoot AI outputs.



...and provide evidence to troubleshoot AI outputs.



...and provide evidence to troubleshoot AI outputs.



...and provide evidence to troubleshoot AI outputs.

User Question

LLM 1

Clinical concepts

...

...

...

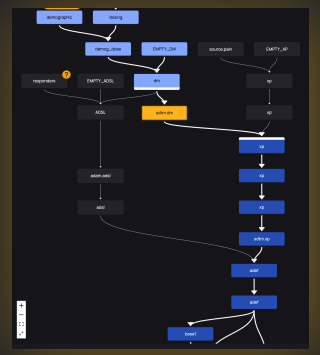
...

...

...

...

Fetch traceability



LLM 2

User Question

Clinical concepts

Traceability info

Output

Natural language

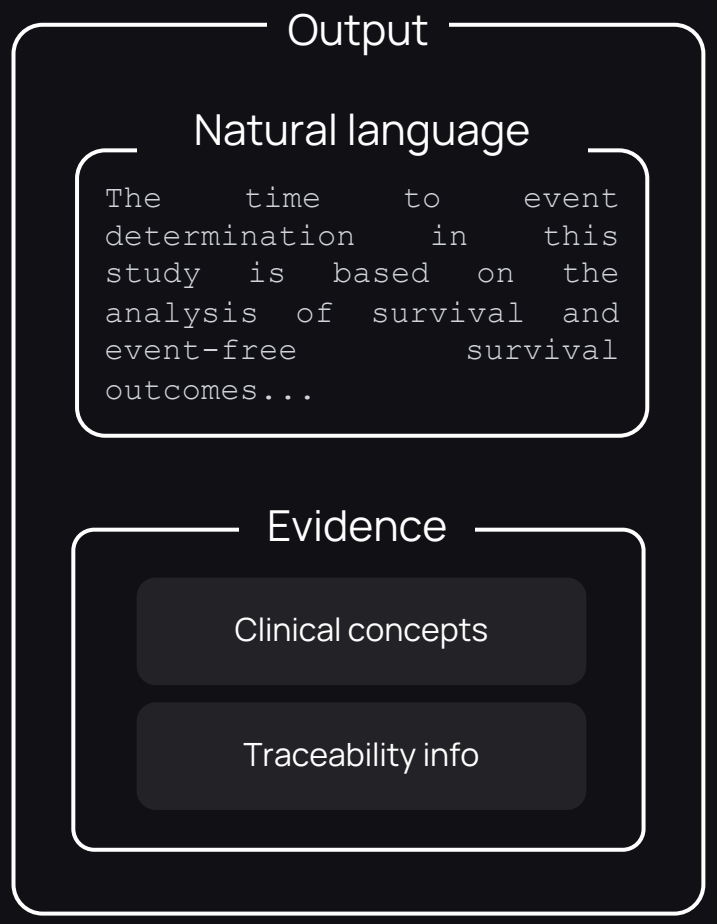
The time to event determination in this study is based on the analysis of survival and event-free survival outcomes...

Evidence

Clinical concepts

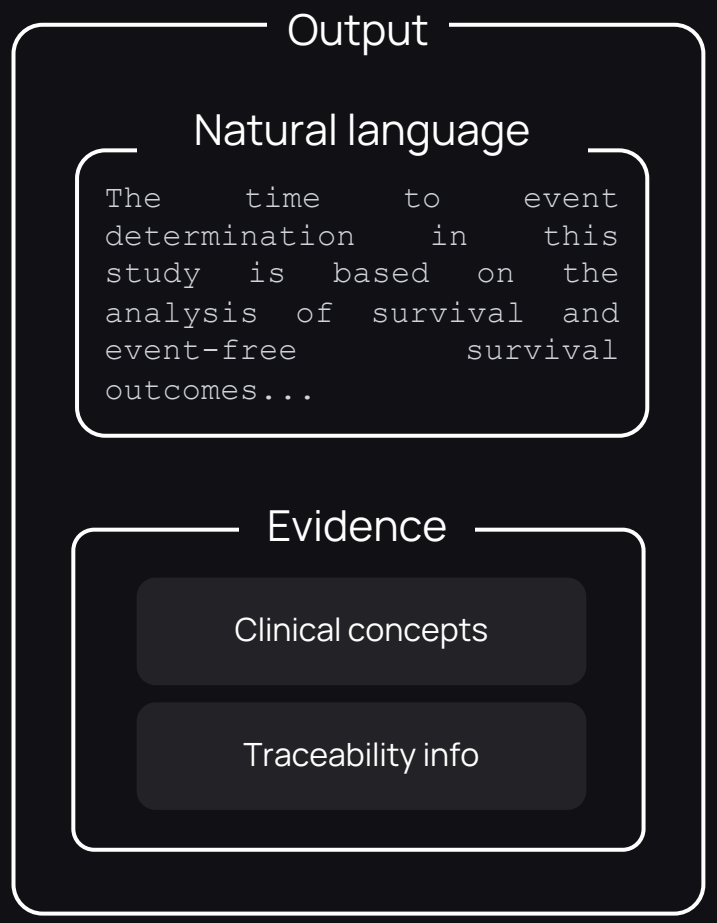
Traceability info

...and provide evidence to troubleshoot AI outputs.



Deterministic	Evidence	Study specific
We provide source of truth traceability	Always provided to review AI outputs	Retrieving directly from a study

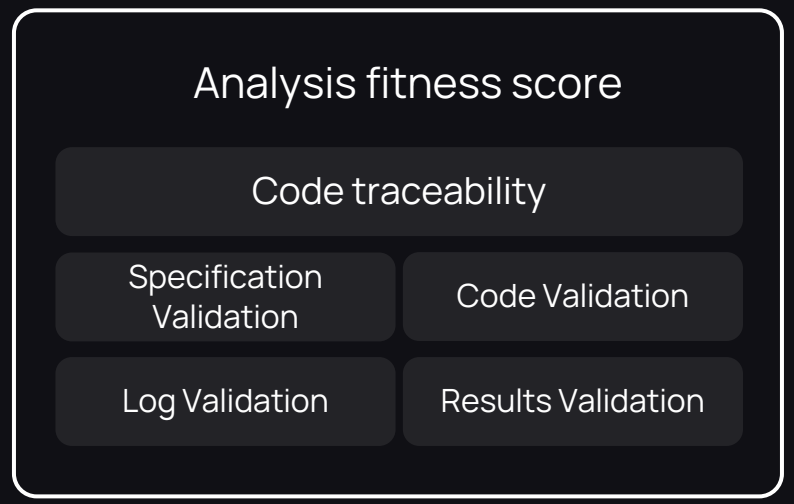
...and provide evidence to troubleshoot AI outputs.



Deterministic	Evidence	Study specific
We provide source of truth traceability	Always provided to review AI outputs	Retrieving directly from a study

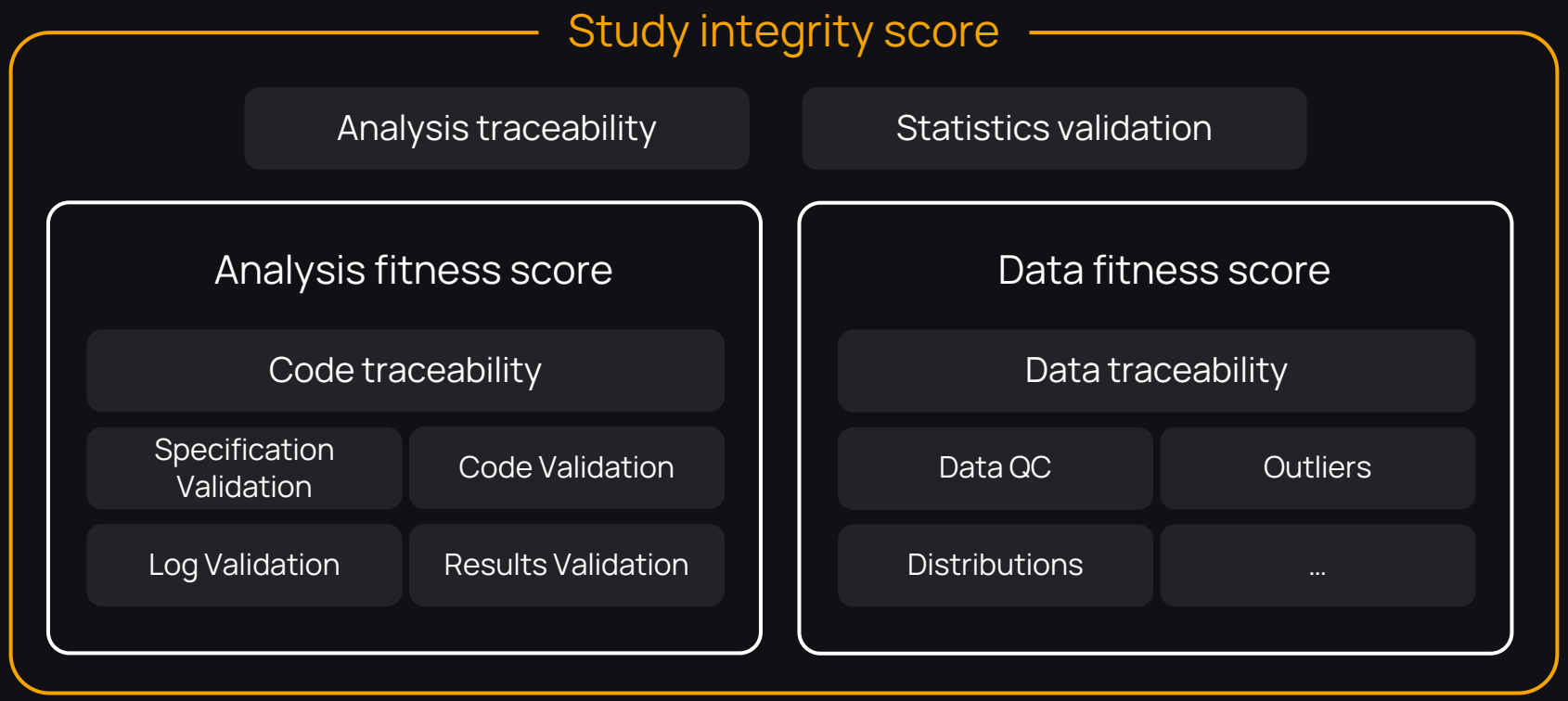
Confidence in output means confidence in your review

Validated by Verisian



Validated by Verisian

Validated by Verisian



Verisian Team

Strategic Team

Tomás Sabat Stöfsel
CEO & Co-Founder

Dr. Henning Kuich
CTO & Co-Founder

Sunil Gupta
Strategic Advisor

Richard Vonk
Strategic advisor

Product Team

Aaron Hutton
Lead Engineer

Andrei Bostan
Software Engineer

Davis Fredrick
Software Engineer

Maksim Vorobev
DevOps Engineer

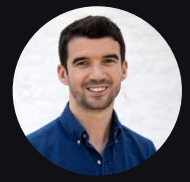
Vikas Surera
Software Engineer

Vishwanath Pawar
Software Engineer

Investors

Armen Vidian
Founder Recode Health

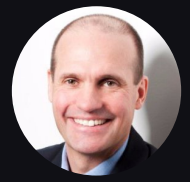
Vishal Gulati, MD
Founder Recode Health



Tom Wilson
Partner Seedcamp

Kate McGinn
Investor Seedcamp

Mads Jensen
Managing Partner, Superseed



Paul Forster
Founder indeed.com

Naud van der Ven
20 years CFO

Naud van der Ven
Founder of Grabyo & Fonix Mobile

Leng Lee
Founder Oui Therapeutics

Endre Sagi
Founder Inference Partners

Loic Veillard
Founder Blitz