



INTEGO
GROUP

SDTM, ADaM and Database Normal Forms Implementation

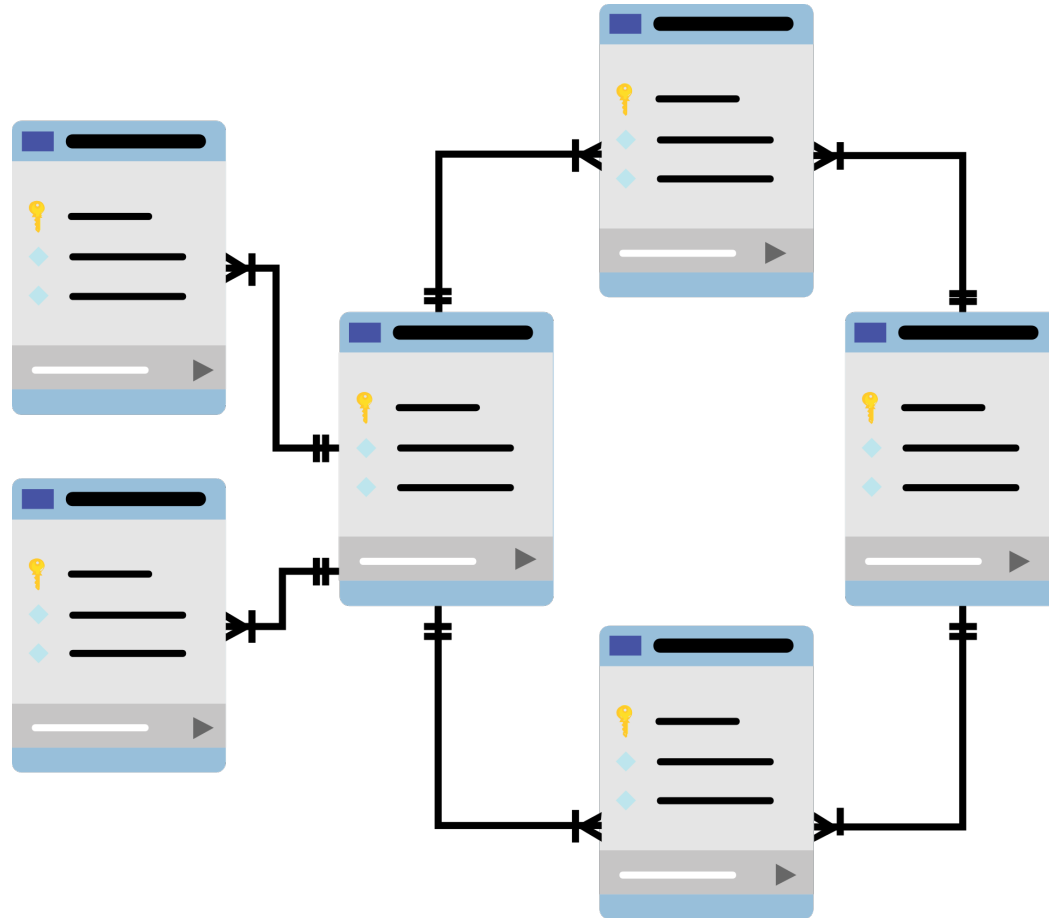
Andrii Borzenkov – Intego Group – Kharkiv – UKRAINE



March 25th, 2021



About Relational Database Theory





Database Normal Forms



1NF: corpuscular values only

Bad

Subject	AEDECOD
A	Headache, Cough

Better

Subject	AEDECOD
A	Headache
A	Cough

2NF: non-key attributes depend on inseparable primary key (+ 1NF)

Bad

<u>Subject</u>	<u>AESEQ</u>	<u>Sponsor ID</u>	AEDECOD
<u>A</u>	<u>1</u>	<u>A1</u>	Headache
<u>A</u>	<u>2</u>	<u>A2</u>	Cough

Better

<u>Subject</u>	<u>AESEQ</u>	Sponsor ID	AEDECOD
<u>A</u>	<u>1</u>	A1	Headache
<u>A</u>	<u>2</u>	A2	Cough

3NF: non-key attributes depend on primary key with no intermediary (+ 2NF)

Bad

Subject	HEIGHT	WEIGHT	BMI
A	181	96	29.3

Better

Subject	Parameter	Result
A	Height	181
A	Weight	96
A	BMI	29.3



Standards Implementation



Make sure values are inseparable

SDTM IG v3.3, section 4.2.8.1 says:

For example, if an adverse event term of "Headache and Nausea" or a concomitant medication of "Tylenol and Benadryl" is reported, sponsors will often split the original report into separate records and/or query the site for clarification.

We **should split**, otherwise datasets will fail 1NF.

“Other” & “Specify” values

It is important to split “Other” value and “Specify” values.
Otherwise it fails 1NF as well.

--SEQ Against Identical Records

(USUBJID, --SEQ) is primary key.

Suppose we have identical records, how to derive --SEQ?

If possible, use --SPID or put RAW.RecordId into SUPP--. It will be used as a part of the natural key of the domain.

Use Vertical Structure If Possible

Using vertical structure is a good way to fit 3NF as much as possible.

Having structure VAR1-VARn where each variable is a distinct value of one or more parameters is hard to read and analyse.

It makes sense only for long values that don't fit variable length.

Fortunately, SUPP-- helps a lot.



Project Setup



Store Functional Dependencies

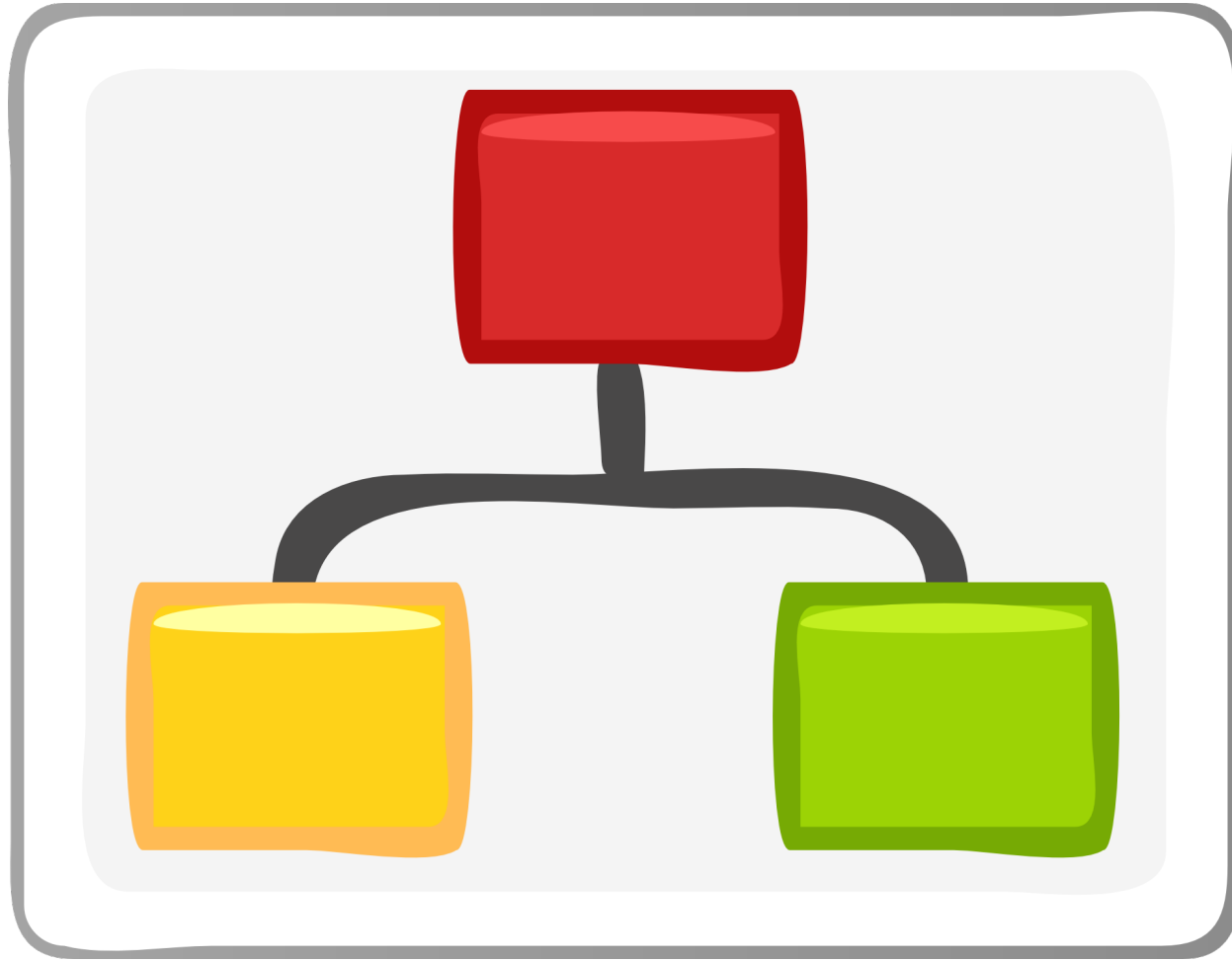
Units conversion

Parameter Codes -> Labels

Visit windowing

Grouping and Coding if applicable

Think Database Oriented





SDTM & ADaM vs Normal forms



ADaM fails 3NF

(USUBJID, ASEQ) -> AVAL -> AVALCATy

(USUBJID, ASEQ) -> BASE -> BASECATy

(USUBJID, ASEQ) -> (AVAL, AyLO) -> R2AyLO

(USUBJID, ASEQ) -> (AVAL, BASE) -> R2BASE

(USUBJID, ASEQ) -> (BASECATy, AVALCATy) -> SHIFTy

SDTM fails 3NF

(USUBJID, LBSEQ) -> LBTESTCD -> LBSTRESU

(USUBJID, LBSEQ) -> LBTESTCD -> LBTEST

(USUBJID, LBSEQ) -> LBTESTCD -> LBCAT

(USUBJID, LBSEQ) -> LBTESTCD/(LBTESTCD, DM.SEX, DM.AGE) ->
LBORNRLO/LBORNRHI/LBSTNRLO/LBSTNRHI

(USUBJID, LBSEQ) -> (LBTESTCD, LBSTRESC, LBSTNRLO, LBSTNRHI) -
> LBTOXGR

(USUBJID, LBSEQ) -> LBDTC -> LBDY

(USUBJID, LBSEQ) -> VISIT -> VISITNUM

Should they be updated?

ADaM **No need**

“ADaM datasets should have a structure and content that allow statistical analyses to be performed with minimal programming.” ADaM IG v1.1, section 2.1

SDTM **Probably**

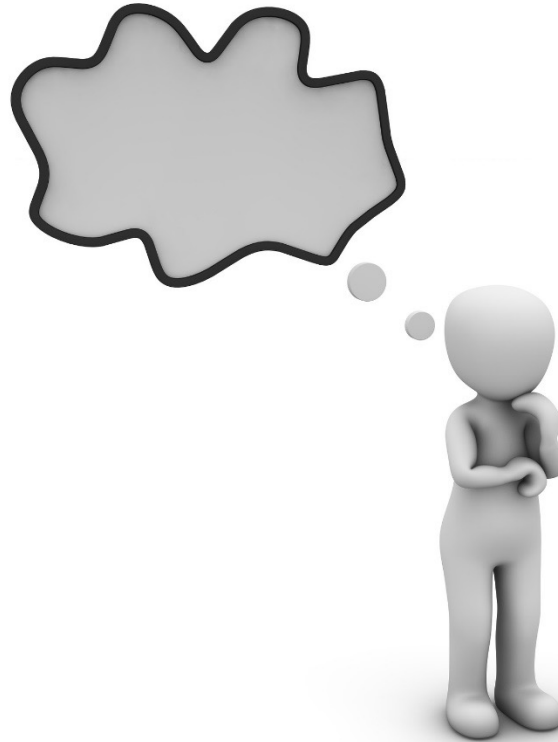
“In general, the SDTMIG v3.x Findings data models are closely related to normalized, relational data models in a vertical structure of one record per observation.” SDTM IG v3.3, section 3.2.1.2

Proposition

There are already Trial Design Model Datasets domains such as “Trial Visits” (TV), “Trial Inclusion/Exclusion Criteria” (TI)

Why not to create domains like “Trial Tests” that could contain all tests of the trial and their functional dependencies?

No mandatory rules, a topic for thought





Thank you for attention!

