



Automated generation of program templates with metadata based specification documents for integrated analysis

Thomas Wollseifen, Germany

Vienna, Oct 2015





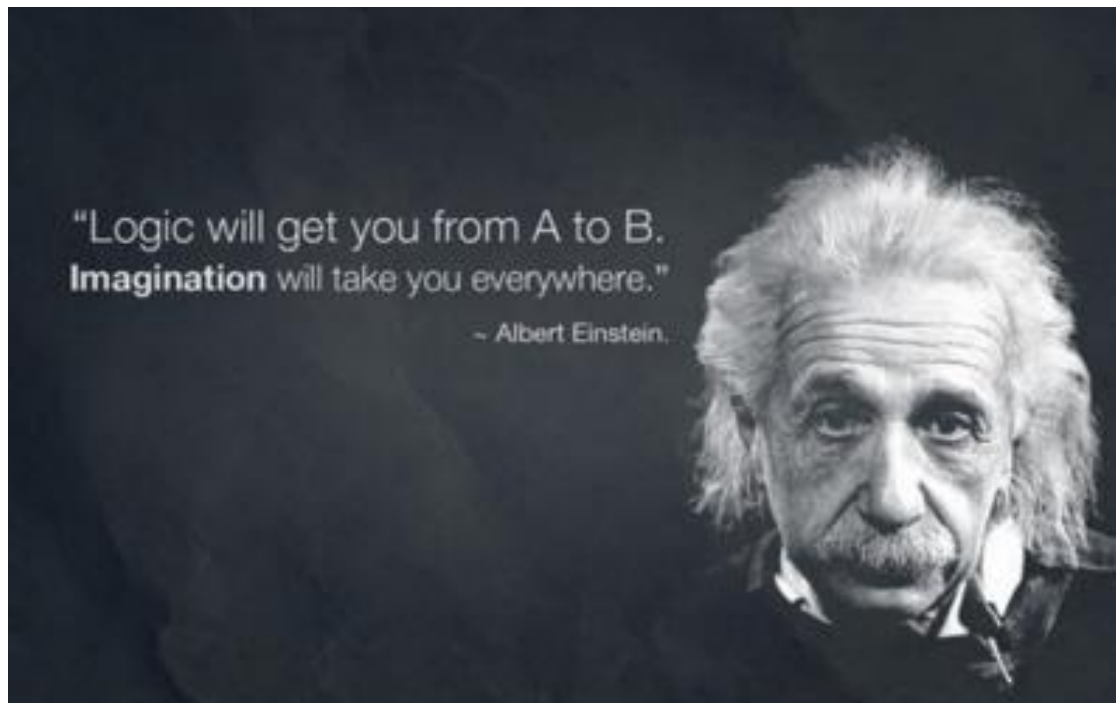
Content

1. Introduction to data integration process
2. Approach
 1. Concept of mapping data into an integrated database
 2. Specification generation
 3. Mapping program generation
3. Summary



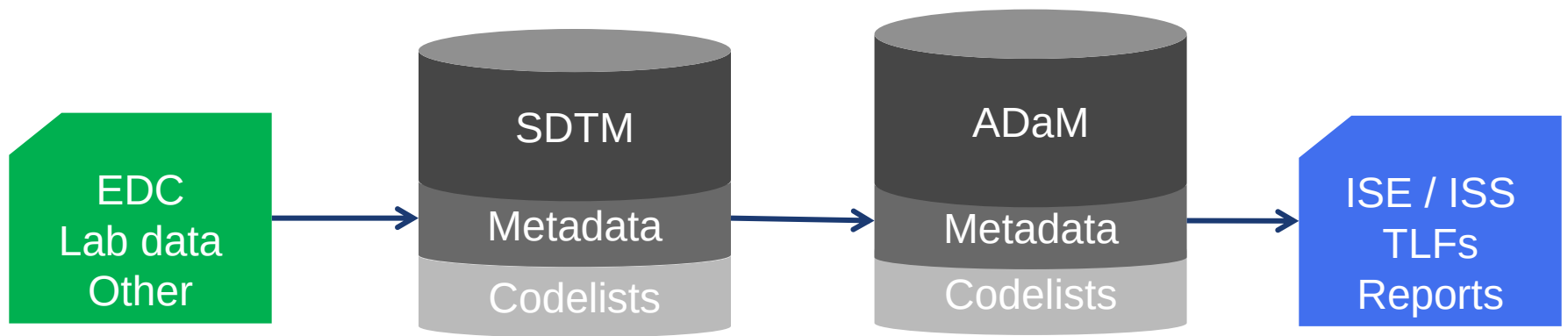
Content

1. Introduction to data integration process
2. Approach
 1. Concept of mapping data into an integrated database
 2. Specification generation
 3. Mapping program generation
3. Summary



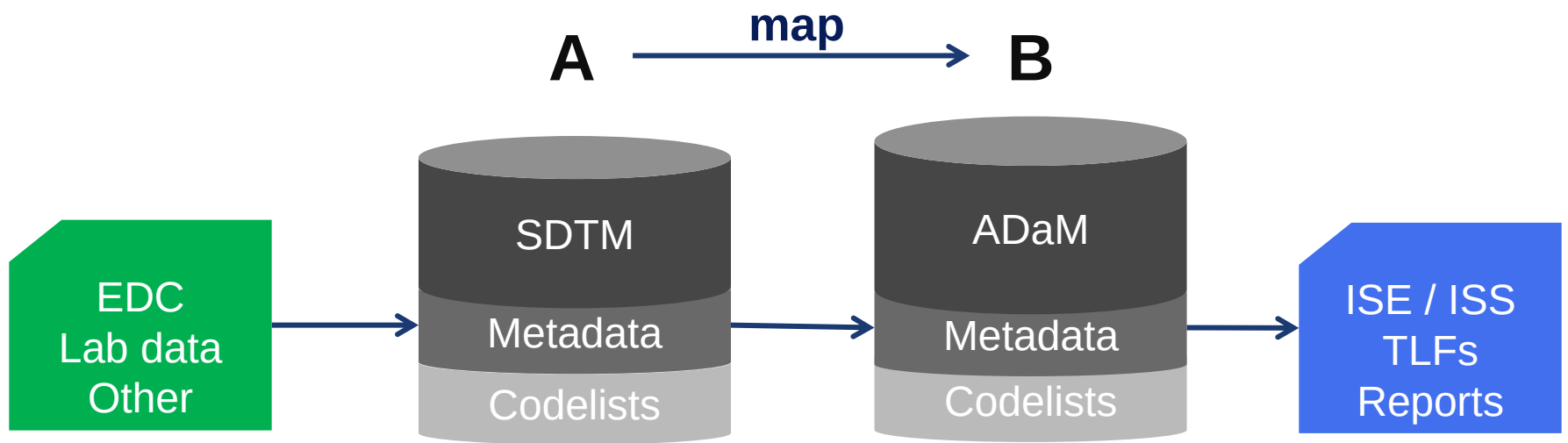


Introduction – Data integration (1)– data flow in clinical study



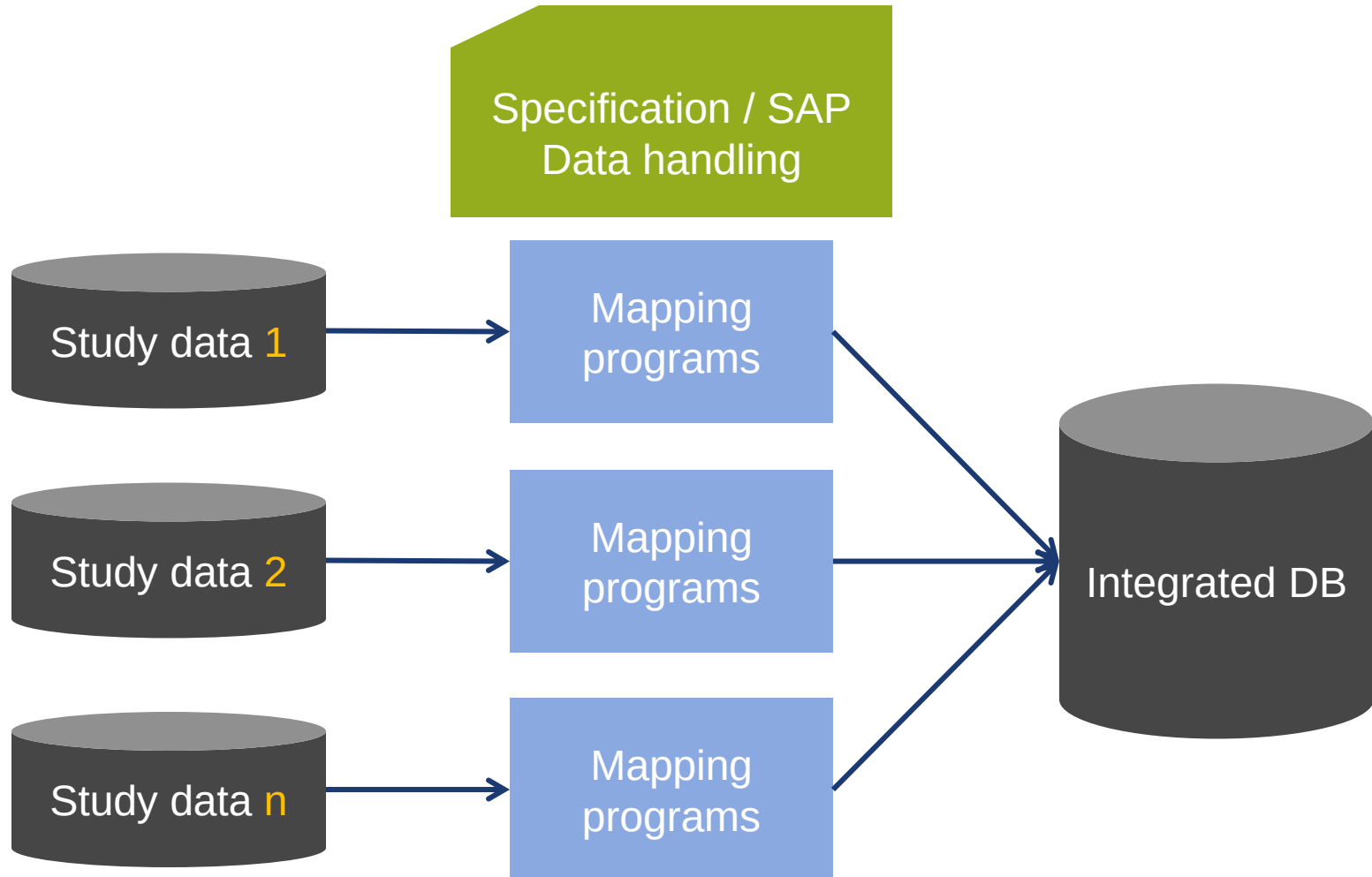


Introduction – Data integration (1)– data flow in clinical study





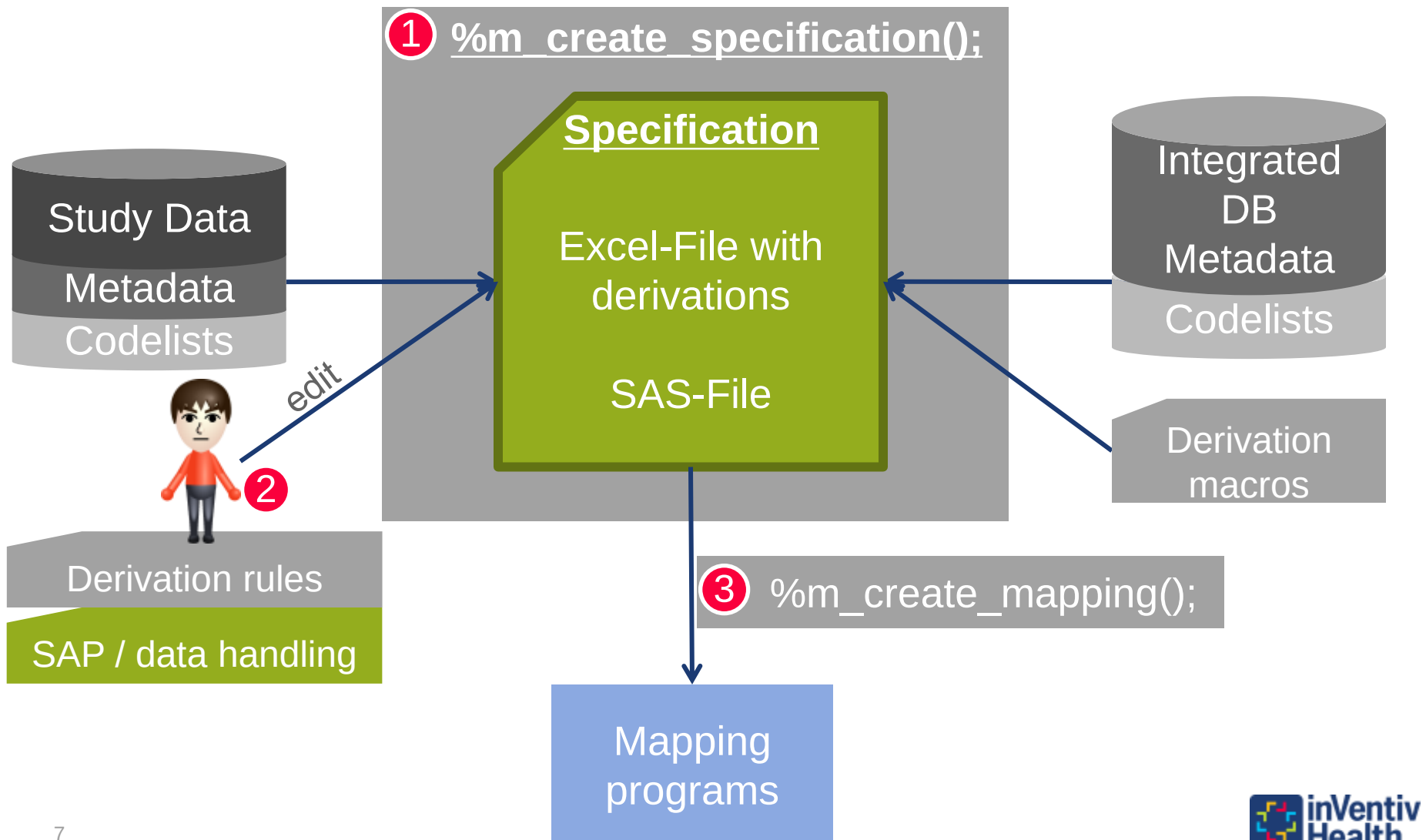
Introduction - Data integration process (2)





Approach:

Concept of mapping data into an integrated database

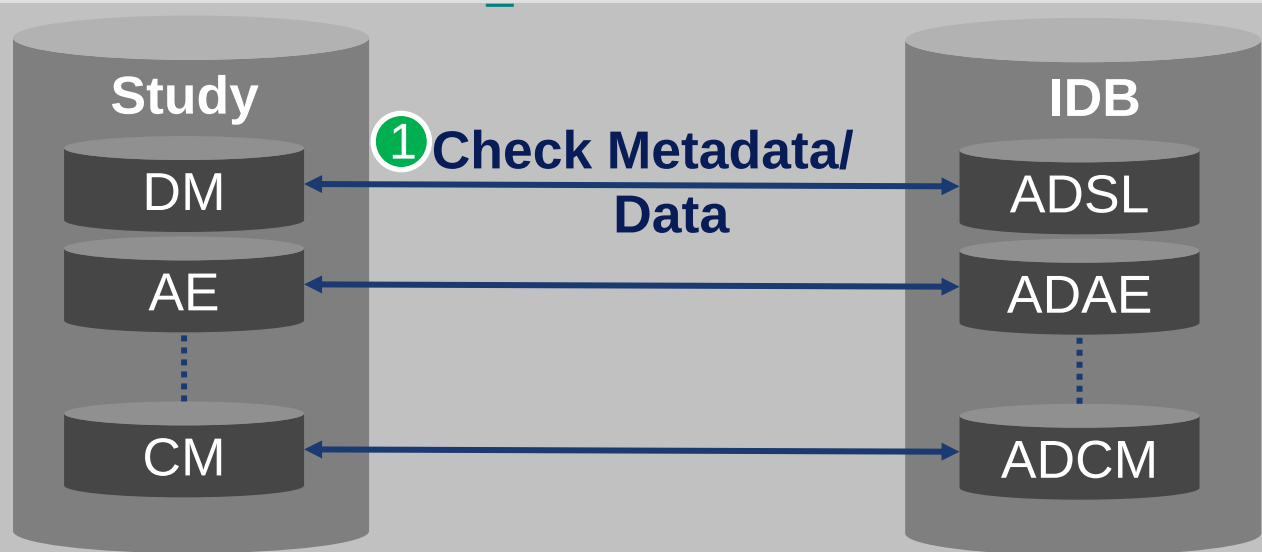




① Create specification from Study/IDB metadata

```
%m_create_specification();
```

```
%DO i=1 %TO &num_domains
```



↓ ② Write Specification

- domain description

- IDB metadata
- Study metadata
- Derivations

Derivation
macros

```
%END;
```



① Check study metadata/data against IDB metadata

```
%DO d=1 %TO &num_domains;  
  /*****  
  /*check study data structure against metadata of the IDB */  
  /*****  
  %mcheck_variables_metadata( indata  =&inlib.&sourcedomain  
                             , meta    =&metalib.&outdomain,  
                             , inadsl  =&inlib.&adslbase  
                             , outdata  =&outdomain._check);  
  ...  
%END;
```

- › Variable name
- › Labels
- › Type of a variable
- › Length of a variable
- › Codelist/Format
- › NotNull



② Write Specification

- domain description

1	domain	sort_order	keyvar	source	target
2	ADSL	STUDYID USUBJID	STUDYID USUBJID	DM	ADSL
3	ADAE	STUDYID USUBJID ASEQ	STUDYID USUBJID ASEQ	AE	ADAE
4	ADEX	STUDYID USUBJID ASEQ	STUDYID USUBJID ASEQ	EX	ADEX
5	ADLB	STUDYID USUBJID PARAMCD	STUDYID USUBJID ASEQ	LB	ADLB
6	ADVS	STUDYID USUBJID PARAMCD	STUDYID USUBJID ASEQ	VS	ADVS
7	ADCM	STUDYID USUBJID CMSEQ	STUDYID USUBJID CMSEQ	CM	ADCM
8	ADEGM	STUDYID USUBJID PARAMCD	STUDYID USUBJID ASEQ	EG	ADEGM
9	ADMH	STUDYID USUBJID MHSEQ	STUDYID USUBJID ASEQ	MH	ADMH
10	ADTA	STUDYID TSPARMCD	STUDYID ARMCD TAETORD	TA	ADTA
11	ADTE	STUDYID ELEMENT	STUDYID ETCD ELEMENT	TE	ADTE
12	ADTS	STUDYID TSPARMCD	STUDYID TSPARMCD	TS	ADTS
13	ADTV	STUDYID VISITNUM	STUDYID VISITNUM	TV	ADTV

- IDB metadata
- Study metadata
- Derivations

74	24	ADSL	BRTH_Y		4	N	Birth			M	Y	99999	DM	BRTH
75	25	ADSL	COUNTRY		3	C	Country	COUNTRY		M	RY	99999	DM	COUN
76	26	ADSL	ENRFL		15	C	Population			O	ENRFLN	99999		
Description ADSL ADAE ADEX ADLB ADVS ADCM ADEGM ADMH ADTA ADTE ADTS ADTV														



Example call: *%m_create_specification*

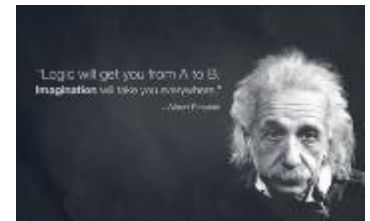
```
/* *****  
/* Example call SDTM --> ADAM */  
/* *****  
%m_create_specification( studyid  =99999  
                        , domain  =ae->adae#dm->adsl#ex->adex#lb->adlb  
                        , path    =&path  
                        , filename=9999_specification.xls  
                        , metalib =meta  
                        , inlib   =in  
                        , adslbase=dm);
```



Example call: *%m_create_specification*

```
/* *****  
/* Example call SDTM --> ADAM */  
/* *****  
%m_create_specification( studyid  =99999  
                        , domain  =ae->adae#dm->adsl#ex->adex#lb->adlb  
                        , path    =&path  
                        , filename=9999_specification.xls  
                        , metalib =meta  
                        , inlib   =in  
                        , adslbase=dm);
```

A  B





② Edit Specification / Approve





② Resulting specification + Manual changes

Target										Source							Derivation							
	A	B	C	D	E	F	G	H	I	K	L	M	N	O	P	Q	R	S	U	V	W	X		
1	Project: 999999 - Pool Metadata										Source Study Metadata							Derivation						
2	Varseq	Domain Target	sasname	Keyvar	Outform	Type	Label	Codelist	NotNull	Descript	Studyid	Domain Source	Var_study	Label Source	Type study	Format study	Length	Comment	Derivation	Derivation_order	Derivation_type	Derivation_flag		
56	6	ADSL	AGEGR1N		3	N	1 (N)	Z_AGEGRP			99999								%_adsl_agegr1(age);	1	[macro]	Y		
57	7	ADSL	AGEGR2		15	C	2			AGEGR2N	99999							[AGEGR2N]	AGEGR2=put(AGEGR2N, Z_AGEGRP.);			Y		
58	8	ADSL	AGEGR2N		3	N	2 (N)	Z_AGEGRP			99999								%_adsl_agegr2(age);		[macro]	Y		
59	9	ADSL	AGEGR3		15	C	3			AGEGR3N	99999							[AGEGR3N]	AGEGR3=put(AGEGR3N, Z_AGEGRP.);	3		Y		
60	10	ADSL	AGEGR3N		3	N	3 (N)	Z_AGEGRP			99999								%_adsl_agegr3(age);		[macro]	Y		
61	11	ADSL	AGEGR4		15	C	4			AGEGR4N	99999							[AGEGR4N]	AGEGR4=put(AGEGR4N, Z_AGEGRP.);			Y		
62	12	ADSL	AGEGR4N		3	N	4 (N)	Z_AGEGRP			99999								%_adsl_agegr4(age);		[macro]	Y		
63	13	ADSL	AGEU		6	C	Age Units	AGEU		DM AGEU	99999	DM	AGEU	Age Units	C	\$AGEU	6		ADSL AGEU					
64	14	ADSL	ARM		200	C	of Planned			ARMCD	99999	DM	ARM	of Planned	C	\$	200	[ARMCD]	ADSL ARM					
65	15	ADSL	ARMCD		20	C	Arm Code	ZARMCD		DM ARMCD	99999	DM	ARMCD	Arm Code	C	\$ZARMCD	20		ADSL ARMCD					
66	16	ADSL	ASR		8	C	ace				99999								asr=cat(scan(uasre,1,'/'), '/', scan(uas		[data step]	Y		
67	17	ADSL	BASEBMI		8.3	N	Body Mass				99999								%_adsl_bmi;		[macro]	Y		
68	18	ADSL	BASEHEIG		8.3	N	Height (cm)				99999								%_adsl_height;		[macro]	Y		
69	19	ADSL	BASEWEIG		8.3	N	Weight (kg)				99999								%_adsl_weight;		[macro]	Y		
70	20	ADSL	BRTHDT		DATE9.	N	Birth Date				99999								%_adsl_brthdt;		[macro]	Y		
71	21	ADSL	BRTHDTL		9	C	Birth Date			for Listings	99999								%_adsl_brthdtl;		[macro]	Y		
72	22	ADSL	BRTH_D		2	N	Birth			D	99999	DM	BRTH_D	Day of Birth	N		8		ADSL BRTH_D					
73	23	ADSL	BRTH_M		2	N	Birth			M	99999	DM	BRTH_M	Birth	N		8		ADSL BRTH_M					
74	24	ADSL	BRTH_Y		4	N	Birth			Y	99999	DM	BRTH_Y	Year of Birth	N		8		ADSL BRTH_Y					
75	25	ADSL	COUNTRY		3	C	Country	COUNTRY		RY	99999	DM	COUNTRY	Country	C	\$COUNTRY	3		ADSL COUNTRY					
76	26	ADSL	ENRFL		15	C	Population			ENRFLN	99999								ENRFL=put(ENRFLN, NY.);			Y		
77	27	ADSL	ENRFLN		3	N	Population	NY		any part of	99999								%_adsl_enrol		[macro]	Y		
110	60	ADSL	TR01EDT		DATE9.	N	Last			TRTEDT.	99999								%_adsl_trt_edt;		[macro]	Y		
111	61	ADSL	TR01ETM		TIME5.	N	Last				99999								%_adsl_trt_etm;		[macro]	Y		
112	62	ADSL	TR01SDT		DATE9.	N	First			TRTSDT.	99999								%_adsl_trt_sdt;		[macro]	Y		
113	63	ADSL	TR01STM		TIME5.	N	First				99999								%_adsl_trt_stm;		[macro]	Y		
114	64	ADSL	TRT01A		200	C	Treatment			TRT01AN	99999								TRT01A=put(TRT01AN, Z_TRT.);			Y		
115	65	ADSL	TRT01AN		4	N	Treatment	Z_TRT			99999								=TRTA;		[data step]	Y		
116	66	ADSL	TRT01P		200	C	Treatment			TRT01PN	99999								TRT01P=put(TRT01PN, Z_TRT.);			Y		
117	67	ADSL	TRT01PN		4	N	Treatment	Z_TRT			99999								=TRTP;		[data step]	Y		
124	74	ADSL	UASR		40	C	Subject				99999								uasr=cat(scan(uasre,1,'/'), '/', scan(uas	5	[data step]	Y		
125	75	ADSL	MDGROUPN		3	N	Grouping of	ZMGROUP			99999	DM	N	Grouping of	N		8		ADSL MDGROUPN					
126	76	ADSL	MEDGROUP		200	C	Grouping of				99999	DM	P	Grouping of	C	\$	200		ADSL MEDGROUP					
127	77	ADSL	BMIGRP		20	C	(C)			BMIGRPN	99999								BMIGRP=put(BMIGRPN, Z_BMIGR.);			Y		
128	78	ADSL	BMIGRPN		8.3	N	BMI Group	Z_BMIGR			99999								%_adsl_bmi_grp;		[macro]	Y		
129	79	ADSL	CNTYGR1		40	C	Country/Ge			CNTYGR1N	99999								CNTYGR1=put(CNTYGR1N, Z_CNTYGR.);			Y		
130	80	ADSL	CNTYGR1N		3	N	Country/Ge	Z_CNTYGR			99999											Y		

Derivation rules

SAP/data handling



edit

approve





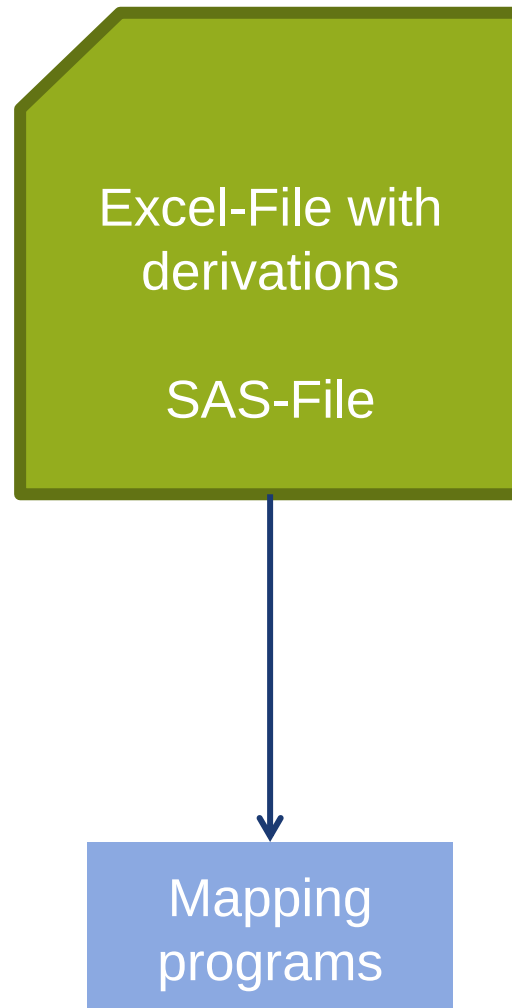
Specification features

Target										Source							Derivation	
Project: 999999 - Pool Metadata										Source Study Metadata							Derivation	
Varseq	Domain Target	sasname	Keyvar	Outform	Type	Label	Codelst	NotNull	Descript	Studyid	Domain Source	Var. study	Label Source	Type study	Format study	Length	Comment	Derivation
41	-1	SAFETY		.						99999	DM	SAFETY	Population	C	\$	2	[not in metadata]	
42	-1	SAFETYLT		.						99999	DM	SAFETYLT	Safety	C	\$	2	[not in metadata]	
43	-1	SAFETYTN		.						99999	DM	SAFETYTN	Population	N	NY	8	[not in metadata]	
44	-1	SAFSUR		.						99999	DM	SAFSUR	Population	C	\$	2	[not in metadata]	
45	-1	SAFSURN		.						99999	DM	SAFSURN	Population	N	NY	8	[not in metadata]	
46	-1	SFETYLTN		.						99999	DM	SFETYLTN	Safety	N	NY	8	[not in metadata]	
47	1	ADSL ADSNAME		8	C	Name	X_ADSNM			99999								
48	2	ADSL STUDYID		10	C	Identifier		Y	D	99999	DM	STUDYID	Identifier	C	\$	10		ADSL STUDYID
49	3	ADSL USUBJID		20	C	Subject		Y	D	99999	DM	USUBJID	Subject	C	\$	20		ADSL USUBJID
50	4	ADSL AGE		3	N	Age			DM AGE	99999	DM	AGE	Age	N		8		ADSL AGE
51	5	ADSL AGEGR1		15	C	1			AGEGR1N	99999							[AGEGR1N]	AGEGR1=put(AGEGR1N, Z_AGEGRP.);
52	6	ADSL AGEGR1N		3	N	1 (N)	Z_AGEGRP			99999								%_adsl_agegr1(age);
53	7	ADSL AGEGR2		15	C	2			AGEGR2N	99999							[AGEGR2N]	AGEGR2=put(AGEGR2N, Z_AGEGRP.);
54	8	ADSL AGEGR2N		3	N	2 (N)	Z_AGEGRP			99999								%_adsl_agegr2(age);
55	9	ADSL AGEGR3		15	C	3			AGEGR3N	99999							[AGEGR3N]	AGEGR3=put(AGEGR3N, Z_AGEGRP.);
56	10	ADSL AGEGR3N		3	N	3 (N)	Z_AGEGRP			99999								%_adsl_agegr3(age);
57	11	ADSL AGEGR4		15	C	4			AGEGR4N	99999							[AGEGR4N]	AGEGR4=put(AGEGR4N, Z_AGEGRP.);
58	12	ADSL AGEGR4N		3	N	4 (N)	Z_AGEGRP			99999								%_adsl_agegr4(age);
59	13	ADSL AGEU		6	C	Age Units	AGEU		DM AGEU	99999	DM	AGEU	Age Units	C	\$AGEU	6		ADSL AGEU
60	14	ADSL ARM		200	C	of Planned			ARMCD	99999	DM	ARM	of Planned	C	\$	200	[ARMCD]	ADSL ARM

- › Mapping from A to B
- › Includes already information from study data
- › Derivations could be adapted manually
- › Colors indicate:
 - Variable needs to be dropped/renamed
 - Variable needs to be derived
 - Variable is probably OK



③ Create mapping programs





3 Create mapping programs (1)

```
%macro m_create_mapping ( project=          /*Project name          */
                          , studyid=         /*Study ID                 */
                          , domain=          /*Domains (separated by '#' ) */
                          , path=            /*Path to output folder     */
                          , userid=          /*User ID                   */
                          , specification=    /*Specification filename     */
                          , spec_path=       /*Path to specification      */
                          , filenamepattern= /*Pattern of the filename    */
                          );
%DO d=1 %TO &num_domains;
    1. Read specification from Excel
    2. Create header of template file
    3. Insert derivations (Data Step / Macros) and comments (descriptions)
       into template file
    4. Post-processing
%END;
```



Mapping
programs



③ Create mapping programs (2)

```
%DO d=1 %TO &num_domains;
```

```
    %let _domain= %left(%qscan(&domain.,&d,"#"));
```

1. %read_meta_spec(domain=&_domain, path=&spec_path., filename=&specification);
2. %template_header(domain=&_domain, studyid=&studyid, project=&project, path=&path, userid=&userid, filenamepattern=&filenamepattern.);

```
    %template_set_ds(domain=&_domain);
```

```
    %template_set_keyvar(domain=&_domain);
```

```
    %template_in_data(domain=&_domain);
```

3. %template_derivation(domain=&_domain);

4. %template_sortorder(domain=&_domain);

```
    %template_drop_in_var(domain=&_domain);
```

```
    %template_add_dummy;
```

```
    %template_clean_lib;
```

```
    %template_eof;
```

```
%END;
```

› Sub-macros consist of **PUT** statements



Create header of template file

```
%MACRO template_header( domain          =          /*Domain          */
                        , studyid         =          /*Study ID          */
                        , project          =          /*Project/Therapeutic area/Compound */
                        , path             =          /*Path to output file          */
                        , validation       =double    /*Validation method <double/code review> */
                        , userid           =          /*User ID / Name of the programmer */
                        , filenamepattern= p-template- /*Filename pattern          */
                        );

data _null_;
  file _templ old;
  put /*-----*/ /*';
  put /* The Drug Company          */ /*';
  put /*-----*/ /*';
  put /* Studyname          : &studyid          */ /*';
  put /* Project           : &project          */ /*';
  put /* Program name      : %nrstr(%LET) prog = p-&targetdomain..sas; */ /*';
  put /* Purpose           : map dataset %upcase(&targetdomain.) to IDB */ /*';
  put /* Template          :          */ /*';
  put /* Validation method : &validation          */ /*';
  put /* Author name       : (&userid.) (%sysfunc(strip(&username.),char.)) */ /*';
  put /* Date completed    : &sysdate9.          */ /*';
  put /* Updated by       : (Name) - (Date):          */ /*';
  put /*-----*/ /*';
run;

%MEND;
```



Derivations with data step and macro calls for ADSL

L	M	N	O	P	Q	R	S	U	V	W	X
Source Study Metadata							Derivation				
Studyid	Domain Source	Var_study	Label Source	Type study	Format study	Length	Comment	Derivation	Derivation_n_order	Derivation_type	Derivation_flag
99999	DM	SAFETY	Population	C	\$	2	[not in metadata]		.		
99999	DM	SAFETYLT	Safety	C	\$	2	[not in metadata]		.		
99999	DM	SAFETYN	Population	N	NY	8	[not in metadata]		.		
99999	DM	SAFSUR	Population	C	\$	2	[not in metadata]		.		
99999	DM	SAFSURN	Population	N	NY	8	[not in metadata]		.		
99999	DM	SFETYLTN	Safety	N	NY	8	[not in metadata]		.		
99999						.			.		Y
99999	DM	STUDYID	Identifier	C	\$	10		ADSL.STUDYID	.		
99999	DM	USUBJID	Subject	C	\$	20		ADSL.USUBJID	.		
99999	DM	AGE	Age	N		8		ADSL.AGE	.		
99999						.	[AGEGR1N]	AGEGR1=put(AGEGR1N, Z_AGEGRP.);	2	[data step]	Y
99999						.		%_adsl_agegr1(age);	1	[macro]	Y
99999						.	[AGEGR2N]	AGEGR2=put(AGEGR2N, Z_AGEGRP.);	.		Y
99999						.		%_adsl_agegr2(age);		[macro]	Y
99999						.	[AGEGR3N]	AGEGR3=put(AGEGR3N, Z_AGEGRP.);	3		Y
99999						.		%_adsl_agegr3(age);	.	[macro]	Y
99999						.	[AGEGR4N]	AGEGR4=put(AGEGR4N, Z_AGEGRP.);	.		Y
99999						.		%_adsl_agegr4(aqe);	.	[macro]	Y

- › Data Step
- › Renames / drop
- › Decode/Code assignment
- › Macro calls



PUT 1) derivations and descriptions

```
%template_derivation(domain=&_domain);
```

```
data _null_;
  set _deriv01;
  file _templ mod;
  put '/*****';
  put '/*' sasname '(' label ')': 'descript '*/';
  put '/*****';
  put derivation;
run;
```

Derivation

[AGEGR1N]	AGEGR1=put(AGEGR1N, Z_AGEGRP.);	2	[data step]	Y
	%_adsl_agegr1(age);	1	[macro]	Y
[AGEGR2N]	AGEGR2=put(AGEGR2N, Z_AGEGRP.);	.		Y
	%_adsl_agegr2(age);	.	[macro]	Y
[AGEGR3N]	AGEGR3=put(AGEGR3N, Z_AGEGRP.);	3		Y
	%_adsl_agegr3(age);	.	[macro]	Y
[AGEGR4N]	AGEGR4=put(AGEGR4N, Z_AGEGRP.);	.		Y
	%_adsl_agegr4(age);	.	[macro]	Y
	ADSL.AGEU	.		
[ARMCD]	ADSL.ARM	.		
	ADSL.ARMCD	.		
	asr=cat(scan(uasre,1,'/'),'/',scan(uasre,2,'/'),'/',scan(uasre,3,'/'),'/',scan(uasre,4,'/'));	.	[data step]	Y

Mapping Program – Data steps

```
...
data &domain.;
  set &domain.;
  /*****
  /*AGEGR1 (Age Group 1 ): decode of AGEGR1N */
  /*****
  AGEGR1=put(AGEGR1N, Z_AGEGRP.);
  /*****
  /*UASR (Unique Subject Identifier/Age/Sex/Race ): */
  /*****
  uasr=cat(scan(uasre,1,'/'),'/',scan(uasre,2,'/'),'/',scan(uasre,3,'/'),'/',scan(uasre,4,'/'));
run;
```



PUT 2) macro calls

```
%template_derivation(domain=&_domain);
```

```
data _null_;
  set _deriv01;
  file _templ mod;
  put '/******';
  put '/*' sasname '(' label ')': ' descript '*/';
  put '/******';
  put derivation;
run;
```

Mapping Program – Macro calls

```
...
/*****
/*AGEGR2N (Age Group 2 (N) ): */
/*****
%_adsl_agegr2(age);
/*****
/*AGEGR3N (Age Group 3 (N) ): */
/*****
%_adsl_agegr3(age);
/*****
/*AGEGR4N (Age Group 4 (N) ): */
/*****
%_adsl_agegr4(age);
```

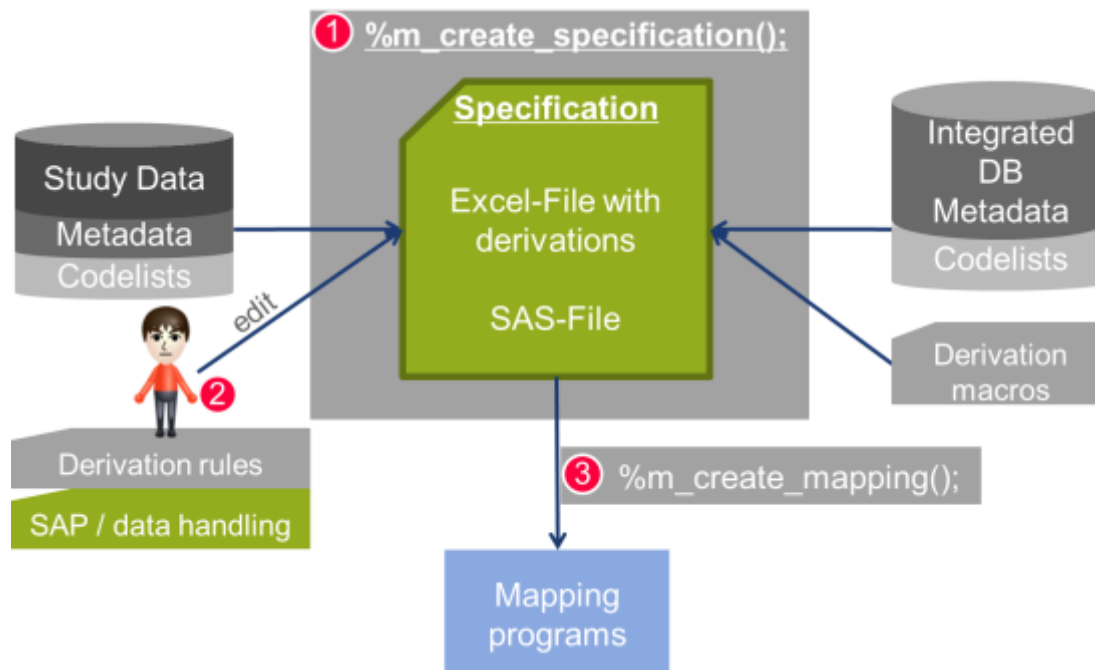
Derivation

[AGEGR1N]	AGEGR1=put(AGEGR1N, Z_AGEGRP.);	2	[data step]	Y
	%_adsl_agegr1(age);	1	[macro]	Y
[AGEGR2N]	AGEGR2=put(AGEGR2N, Z_AGEGRP.);	.		Y
	%_adsl_agegr2(age);	.	[macro]	Y
[AGEGR3N]	AGEGR3=put(AGEGR3N, Z_AGEGRP.);	3		Y
	%_adsl_agegr3(age);	.	[macro]	Y
[AGEGR4N]	AGEGR4=put(AGEGR4N, Z_AGEGRP.);	.		Y
	%_adsl_agegr4(age);	.	[macro]	Y
	ADSL.AGEU	.		
[ARMCD]	ADSL.ARM	.		
	ADSL.ARMCD	.		
	asr=cat(scan(uasre,1,'/'), '/', scan(uas	.	[data step]	Y



Summary

- › 1. Semi-automatic specification generation
 - Check study metadata/data vs. IDB metadata
 - Include predefined mapping macros
- › 2. Manual changes of derivations based on SAP/Data handling
- › 3. Automatic Mapping program generation





**Thank you
for your attention!**

Questions?

Email: thomas.wollseifen@inventivhealth.com