

Developing an end to end Standard & Automated Risk Based Monitoring Process.

AR01

Giuseppe Di Monaco,

UCB BioSciences GmbH, Monheim, Germany

Tim Williams,

UCB BioSciences, Raleigh, USA



Inspired by **patients.**
Driven by **science.**

INTRODUCTION

Risk Based Monitoring (RBM)

- Ensure quality of clinical trials by
 - Identifying, assessing, monitoring, mitigating risks which could affect quality or safety of a study.

Key Risk Indicator (KRI)

- A metric used to assess risk associated with an activity

Central Data Surveillance Oversight activities

- Identify issues & risks
- Relevant signals are monitored

Three major areas of Data Surveillance at UCB

- Data Quality & Validity
- Patient Safety
- Site Compliance and Performance

RBM Programming Team

- Creates KRI analysis datasets from SDTM/ADAM datasets

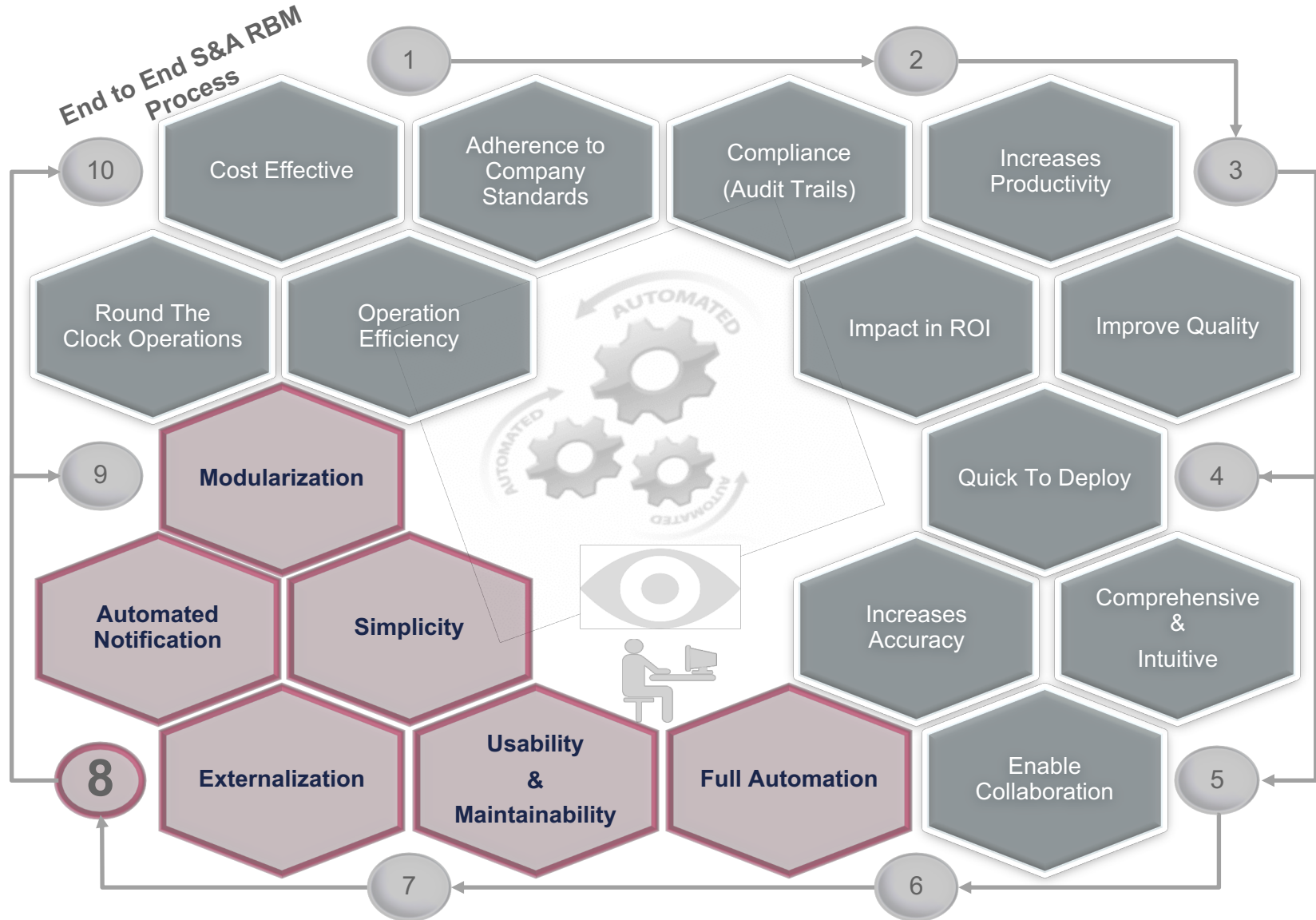
Design Problems

- **3 areas** of data surveillance are **broad** and **complex**
- **Lifespan** of the RBM Programs
- **Total Number** of RBM Programmers
- **Lifecycle** of KRI programs
- **Monolithic** programs **focused** on KRIs Functional requirements (**FR**)
 - **FR** describes what the program generates
 - **NFR** describes how the program is built

Impacts

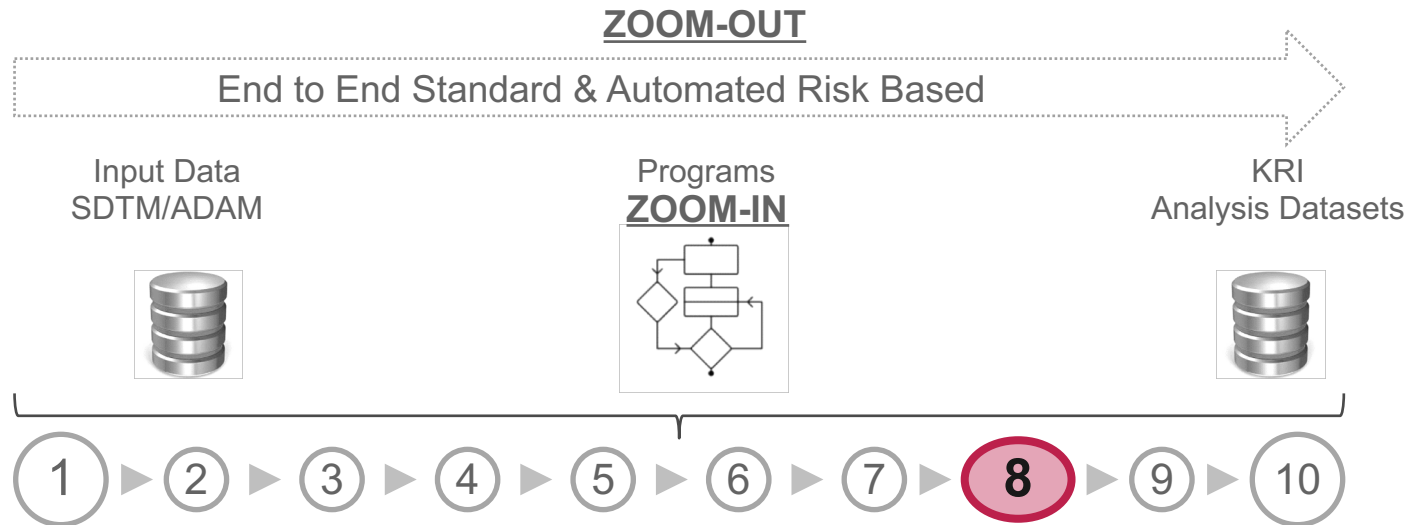
- **Maintainability**
 - Is it easy to understand the program code?
- **Usability**
 - Is it easy to use?
- **Scalability**
 - Can it handle a growing amount of work?
- **Communicability**
 - Is it easy to share information?

ASPECTS COVERED IN THIS PRESENTATION



End to End S&A

- RBM Data Surveillance process and adjacent areas **re-engineered/improved/automated**



STEP 1: Autoconfiguration

STEP 2: Check data

STEP 3: Copy data

STEP 4: Email notification

STEP 5: Close previous runs

STEP 6: Adapt to standards

STEP 7: Load study settings

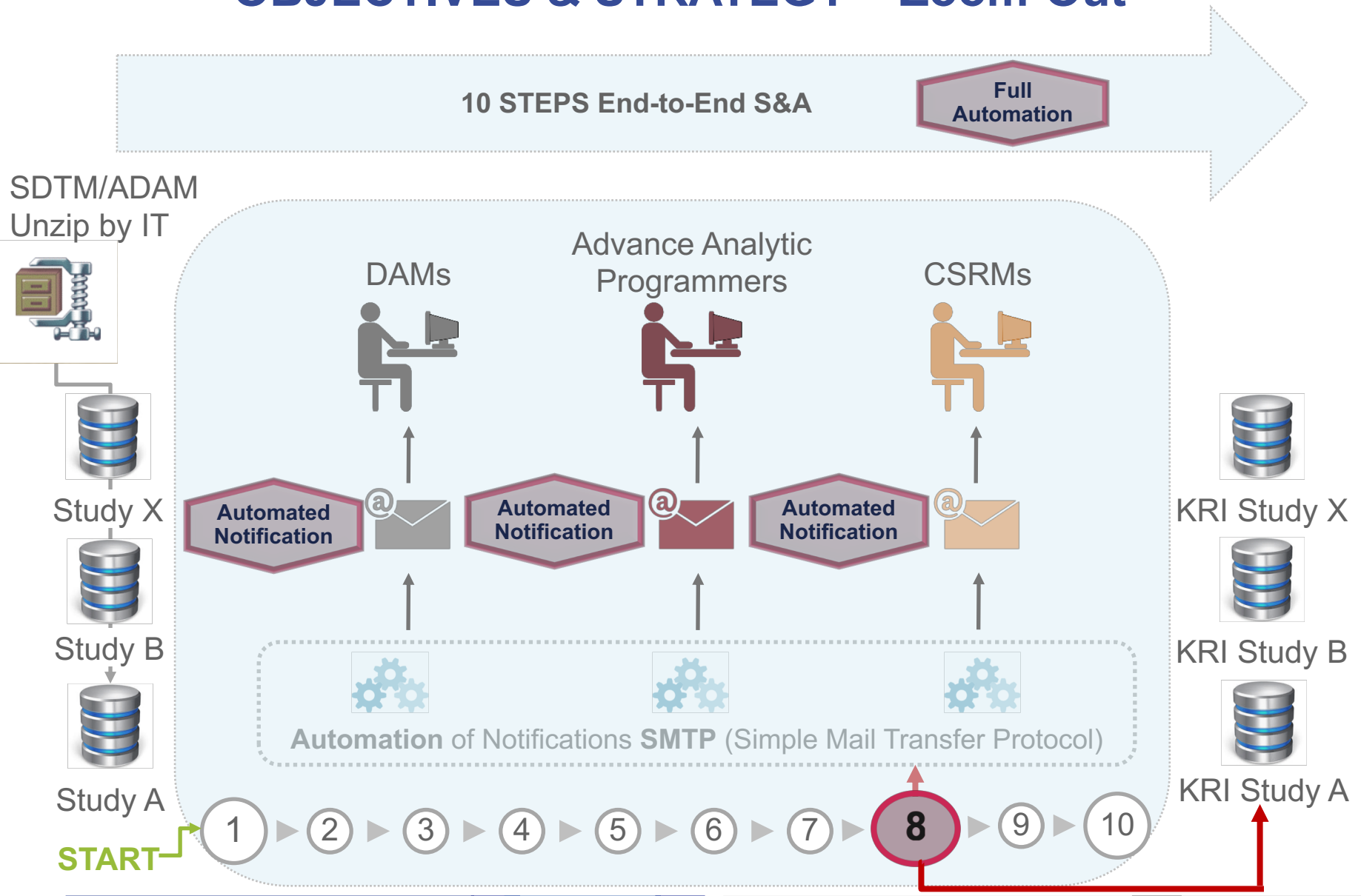
STEP 8: Run, compare, validate

STEP 9: Archive

STEP 10: Quality, sanity checks

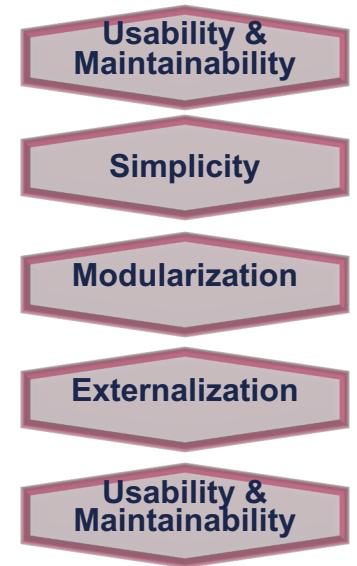
10 major steps. We focus on step number **8**

OBJECTIVES & STRATEGY – Zoom-Out



OBJECTIVES & STRATEGY – Zoom-In

7



Modular Programming

- **MODULES**: collection of data & function **abstractions**
- **ZOOM-IN** in to the **NFR** (Non-Functional Requirements)

Types of modules:

- Process support (utilities), functional modules (KRI analysis creation)

Benefits

- **Maintainability**: minimal effort to repair
- **Usability**: Debugging automation
- **Scalability**: Able to handle a growing amount of work
- **Communicability**: Automated email notifications
- **Traceability**: Same filenames for → .sas, .txt, .log, .sas7bdat, .xls

Example

- **ut_tstcd_mahap_dat_u.sas** → Utility
- **v_dva_mahap_dat_u.sas** → KRI “Low Variability”

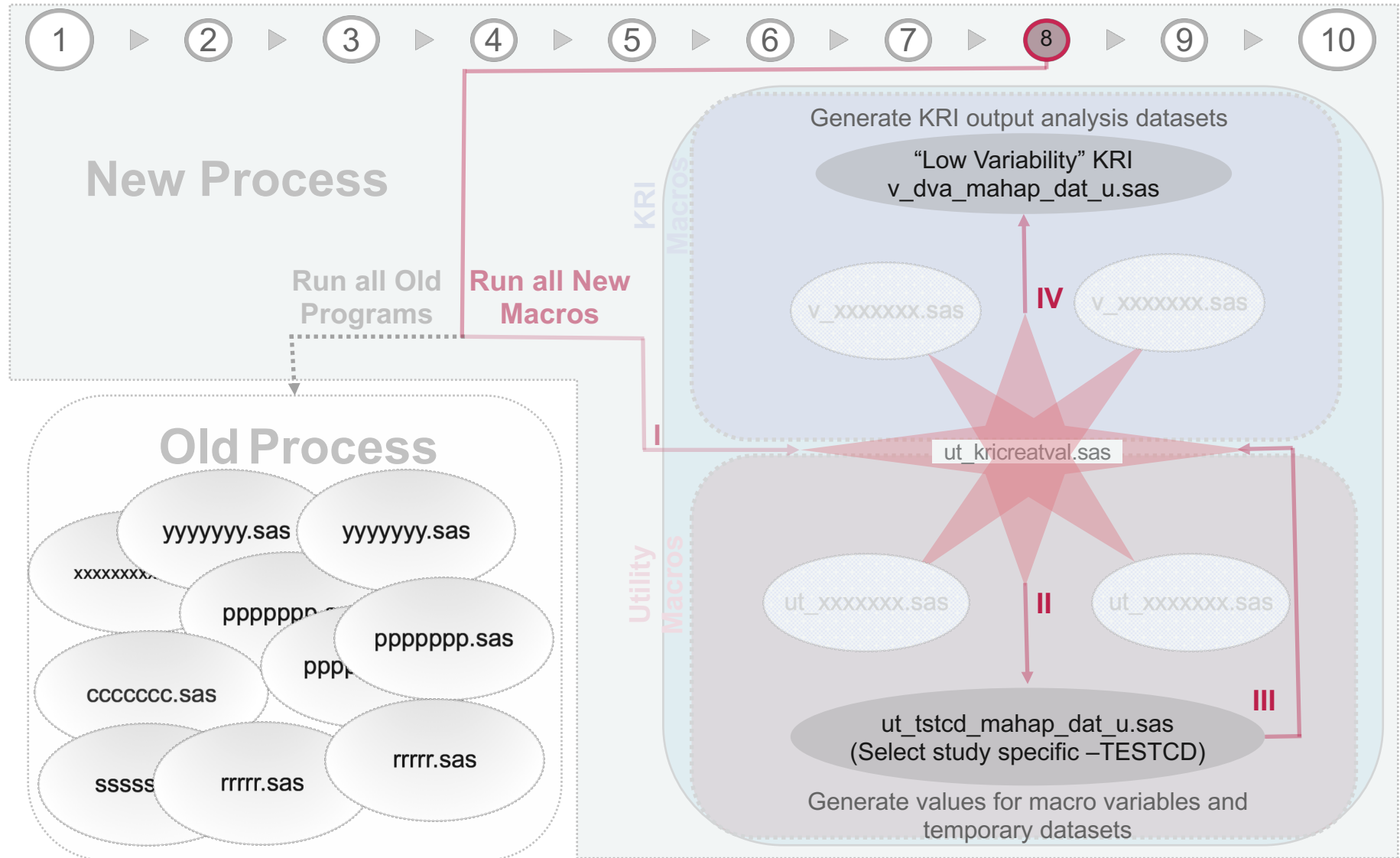
DESIGN & IMPLEMENTATION STAR SCHEMA

Full
Automation

Modularization

Externalization

8

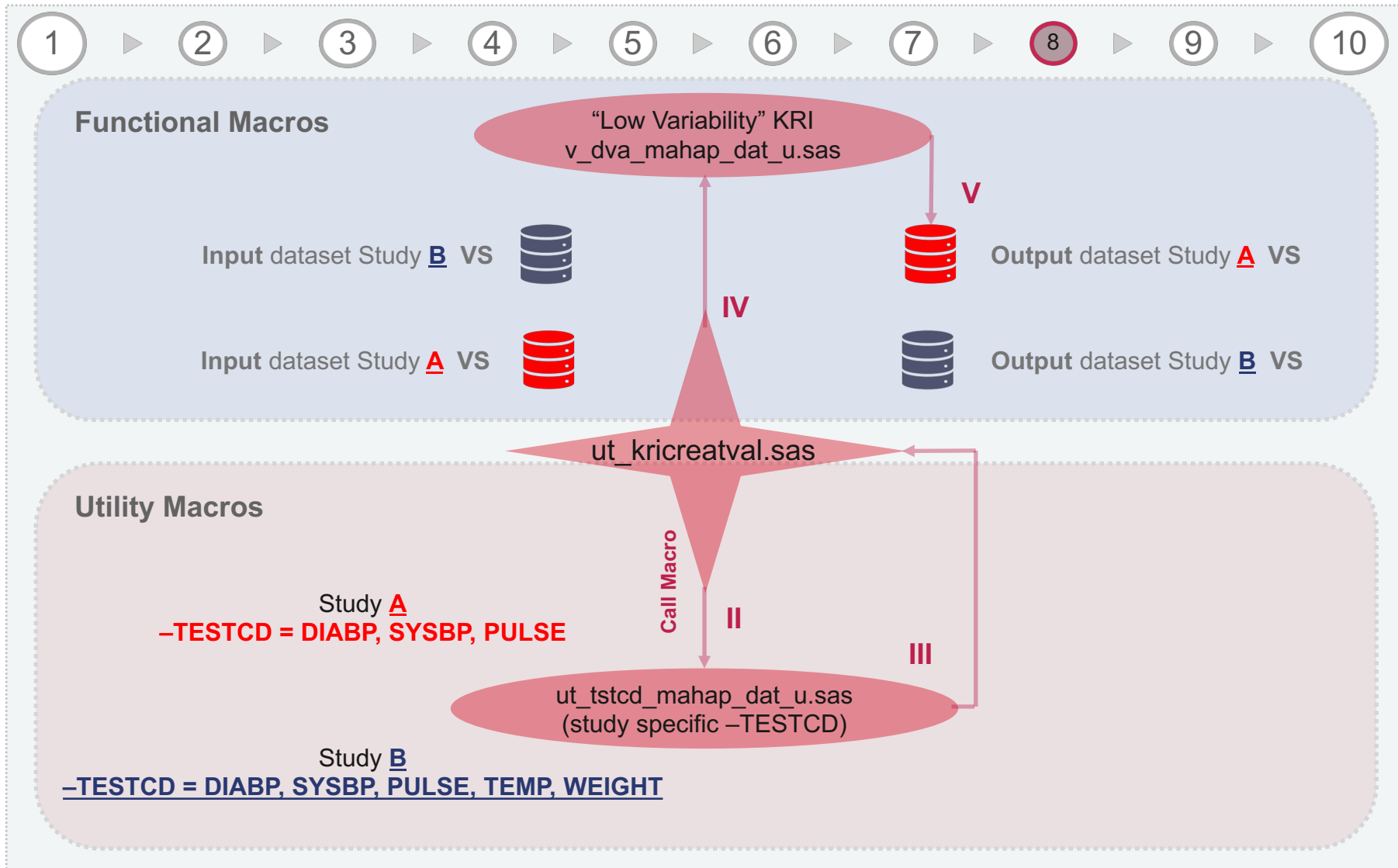


DESIGN & IMPLEMENTATION - STAR SCHEMA

9

Modularization

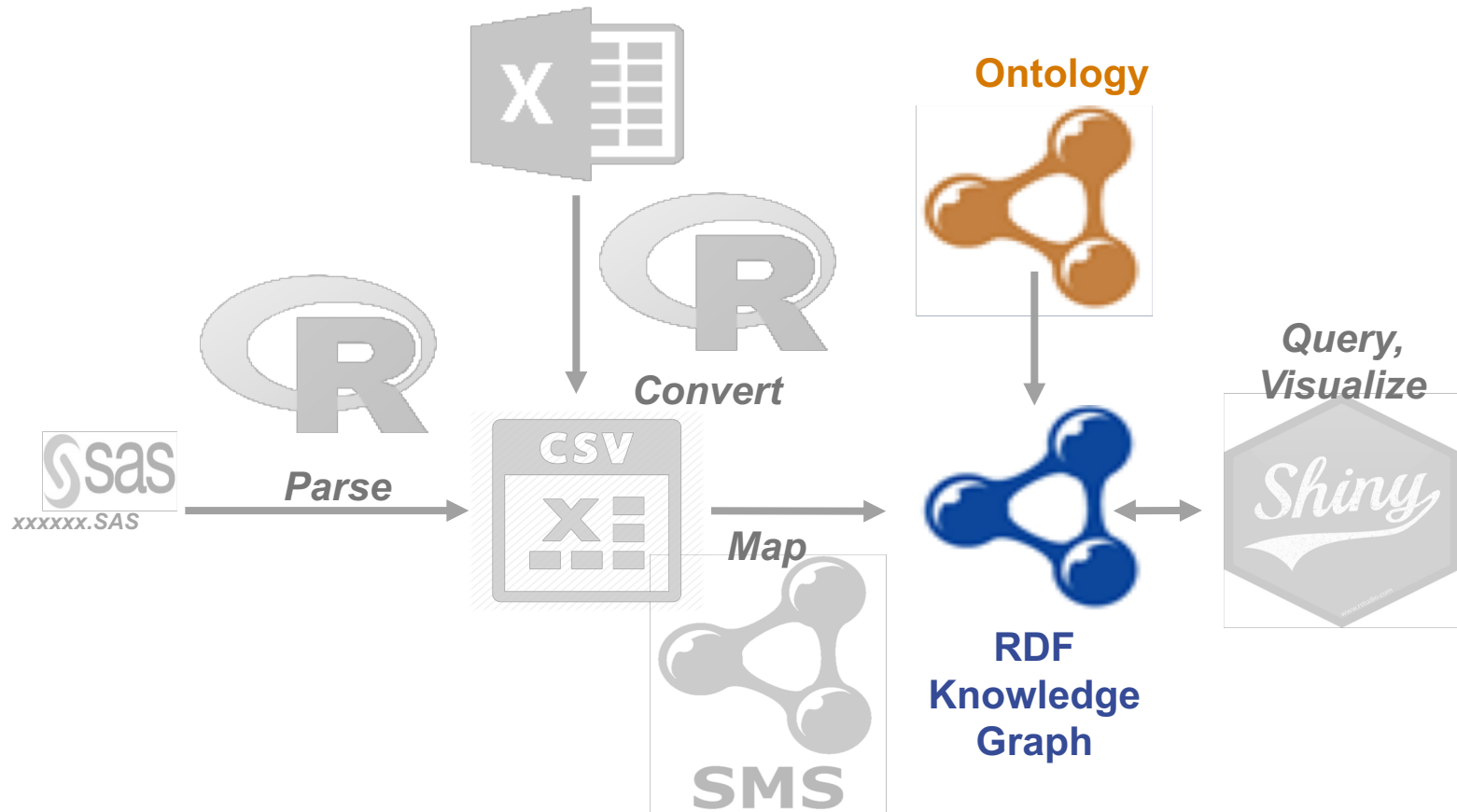
Externalization



10 STEPS S&A PROCESS

1. DATA QUALITY CHECKS & OVERSIGHT
2. PROCESS LOG
3. USABILITY

CODE MANAGEMENT WITH A KNOWLEDGE GRAPH



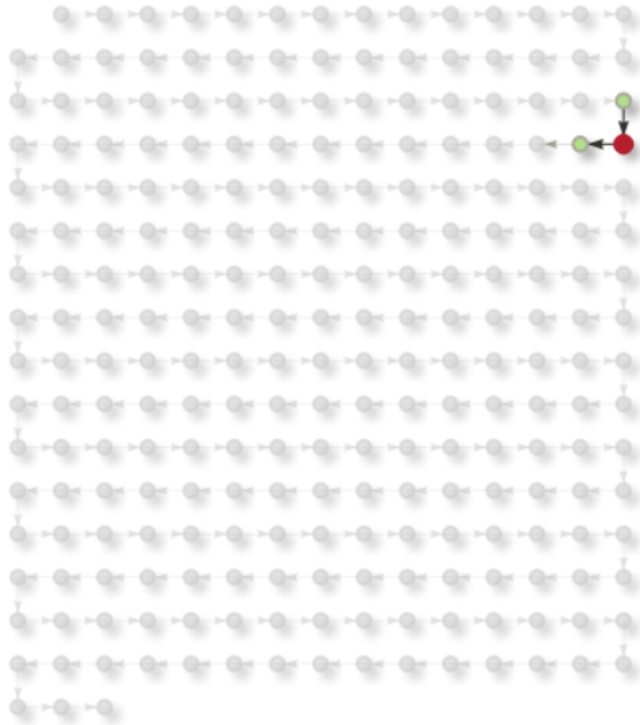
Conversion to Knowledge Graph

CODE MANAGEMENT WITH A KNOWLEDGE GRAPH

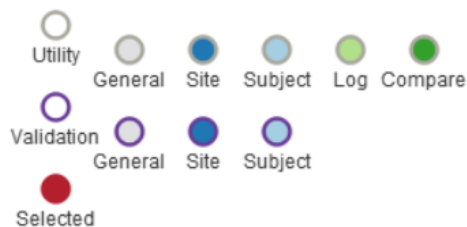
12

Macro call order in program: `ut_kricreatval.SAS`

Call type ▾



LEGEND



`&v_dva_mahap_dat_u()`

Called at line: 177

Purpose: To create v_dva_mahap_dat_u

Primary Author: Giuseppe Di Monaco

Documentation: [Manual](#), [DDT](#)

Key Risk Indicators

ID	KRI	Category
DQV002	Equal Pairs	Quality
DQV002	Equal Pairs	Data
DQV002	Equal Pairs	Validity

Parameter=Value data

parameter	value
_debug	N
dstlib	dpst1_v
outdset	v_dva_mahap_datvs_u
outslib	dva_mahap_datvs_u
precision	0.0001
principlst	&_ut_testcdlist_VS.
sourcedset1	v_cen_usub
sourcedsetspecial	VS
srclib	inraw1

Modularization

Externalization

Automated
Notification

Simplicity

Full Automation

Usability &
Maintainability

CODE MANAGEMENT WITH A KNOWLEDGE GRAPH

13

Enforcing rules

Modularization

Simplicity

Externalization

Usability
&
Maintainability

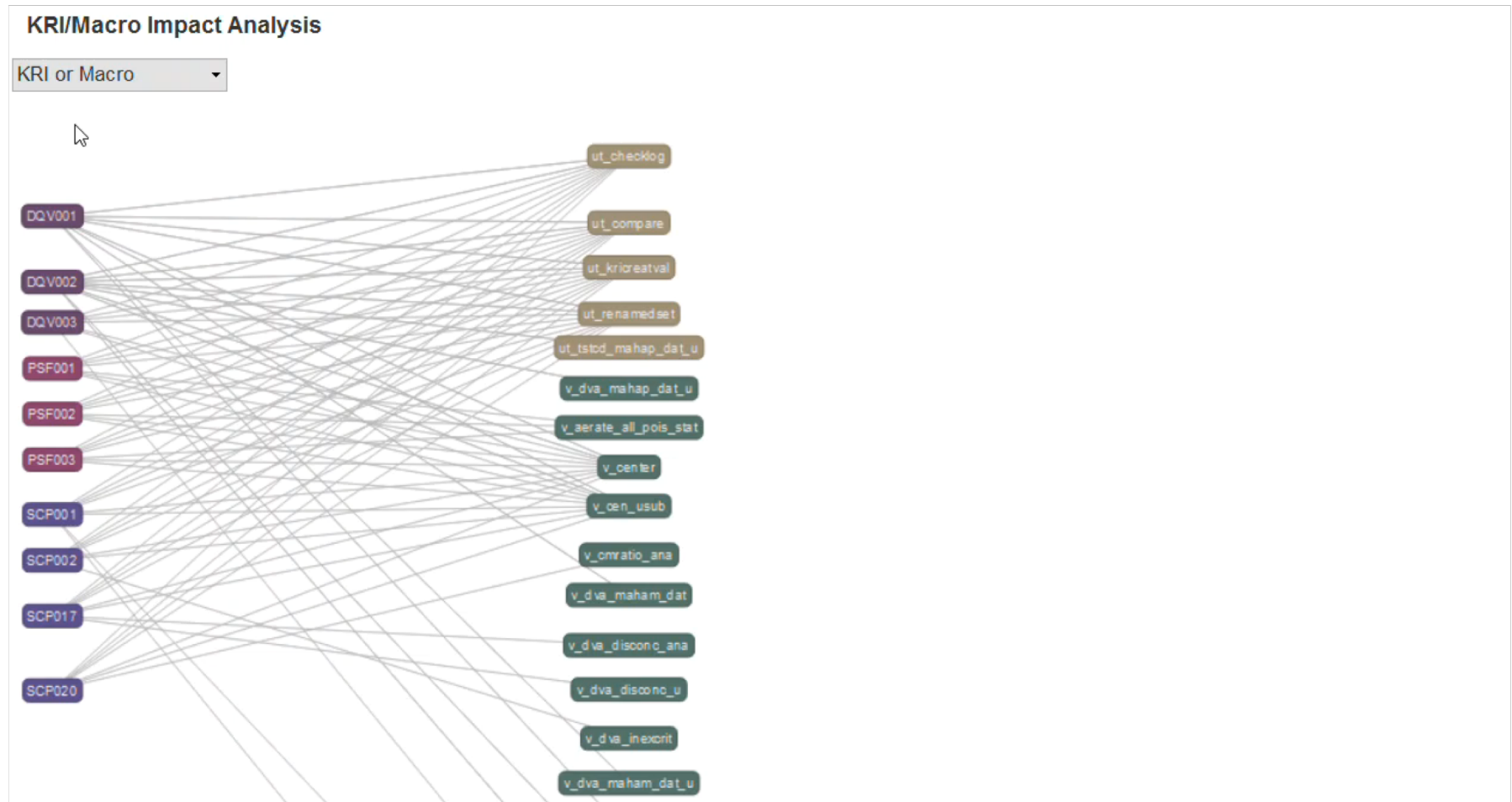
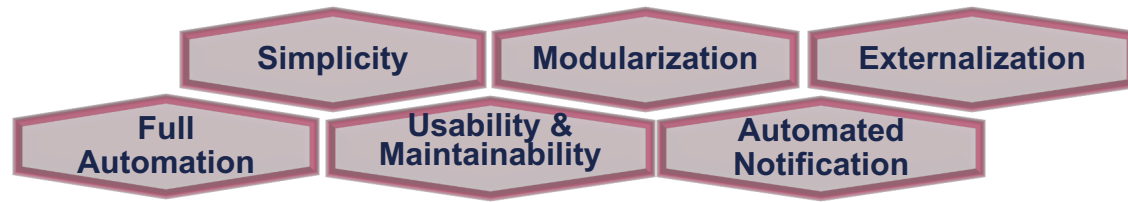
Macro calls in program `ut_kricreatval.sas`

Line	Macro	Parameter1	Value1	Parameter2	Value2	Parameter3	Value3	Parameter4	Value4
36	ut_checklog	macroname	v_cen_usub	logfilename	v_cen_usub	_debug	N	folder	&loglocati...
43	ut_checklog	folder	&loglocati...	macroname	v_center	logfilename	v_center	_debug	N
53	ut_checklog	macroname	v_dva_priocon_...	logfilename	v_dva_priocon_...	folder	&loglocati...	_debug	N
63	ut_checklog	macroname	v_dva_priocon_...	logfilename	v_dva_priocon_...	folder	&loglocati...	_debug	N
73	ut_checklog	logfilename	v_cmratio_ana	macroname	v_cmratio_ana	folder	&loglocati...	_debug	N
83	ut_checklog	logfilename	v_dva_disconc_...	macroname	v_dva_disconc_...	_debug	N	folder	&loglocati...
93	ut_checklog	logfilename	v_dva_disconc_...	macroname	v_dva_disconc_...	folder	&loglocati...	_debug	N
106	ut_checklog	macroname	ut_tstcd_mahap...	logfilename	ut_tstcd_mahap...	_debug	N	folder	&loglocati...
119	ut_checklog	logfilename	v_dva_mahap_da...	macroname	v_dva_mahap_da...	folder	&loglocati...	_debug	N
126	ut_checklog	logfilename	v_dva_mahap_da...	macroname	v_dva_mahap_da...	folder	&loglocati...	_debug	N
Line	Macro	Parameter1	Value1	Parameter2	Value2	Parameter3	Value3	Parameter4	Value4
	v_dva_m	All	/	All	/	All	All	All	All
117	v_dva_mahap_dat_u	precision	0.0001	principlst	&_ut_testc...	outdset	v_dva_mahap_da...	where_filter	(LBCAT in ("HE...
177	v_dva_mahap_dat_u	precision	0.0001	principlst	&_ut_testc...	outdset	v_dva_mahap_da...	where_filter	

CODE MANAGEMENT WITH A KNOWLEDGE GRAPH

Impact Analysis Demo

14



CONCLUSIONS

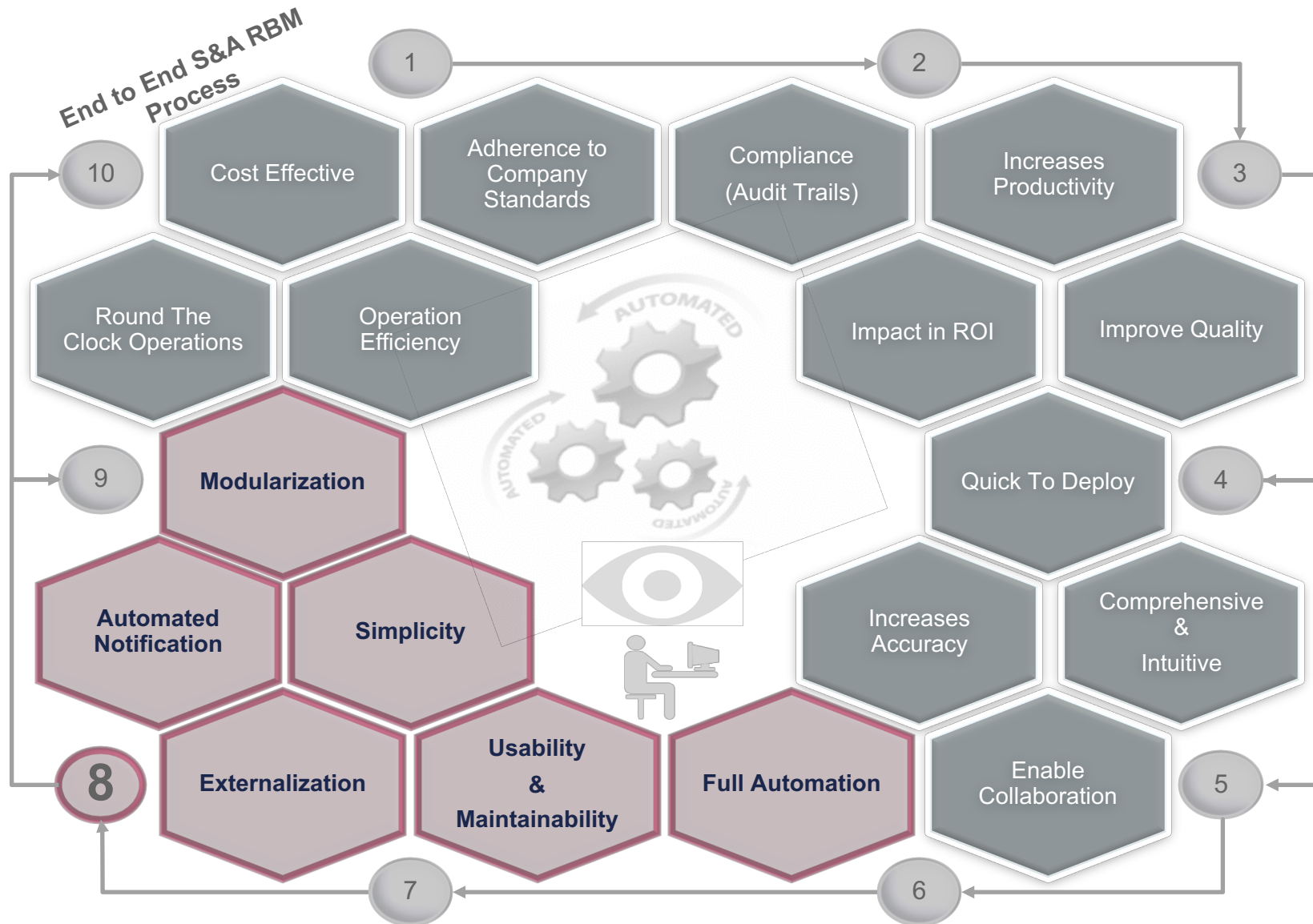
SUMMARY

- **Design Problems**
 - **Data Surveillance** areas are **broad** and **complex**
 - **KRI Monolithic** programs are **difficult** and **expensive** to maintain
 - **High lifespan** and **Large scale** system
 - **High Number** of RBM Programmers
- **Solution**
 - **S&A** Process covering **RBM Data Surveillance** process and other areas
- **Strategy**
 - **ZOOM-OUT** → 10 Steps process
 - **ZOOM-IN** → Macros Functionalities

CONSIDERATIONS

- **S&A** is an Organizational attitude → Need for support of managers and colleagues
- **Caution** → Do not fall into the "prototype trap"

CONCLUSION - CONTINUED



ACKNOWLEDGMENTS

The authors would like to thank

Brigitte Bernhoff

Without her contribution all this would have not been possible.

Questions?

Thanks!