



AD05

Don't be a Mutt

Use SAS MUTT

PHUSE EU CONNECT 2022
NOVEMBER 13-16

MICHAEL HEDEGAARD THOMSEN
MHZT@NOVONORDISK.COM



Disclaimer

- Views and opinions expressed are those of the speaker and not necessarily Novo Nordisk

Why use Unit Test

What is MUTT

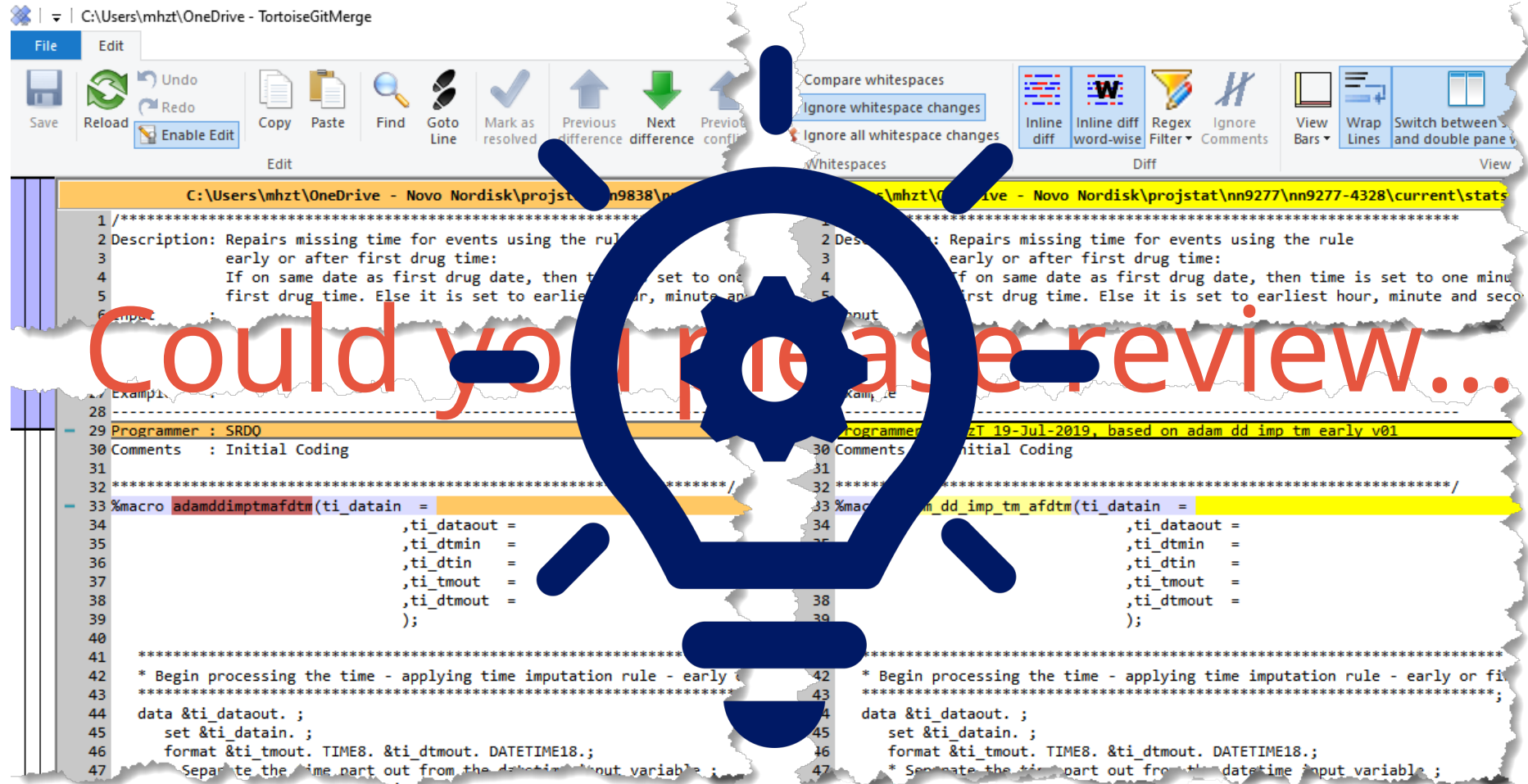
Demo

Future perspectives

Conclusion

Agenda

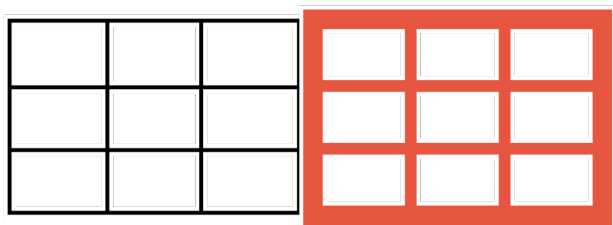
Macro Unit Test Tool



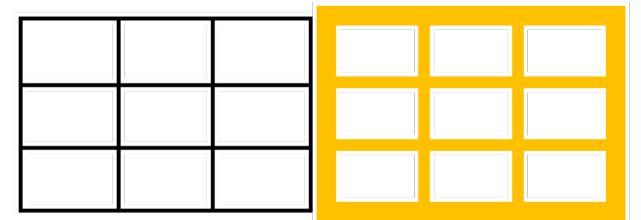
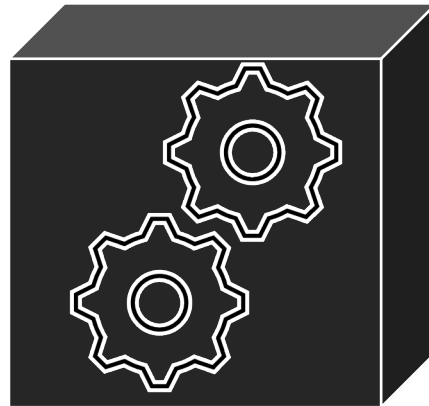
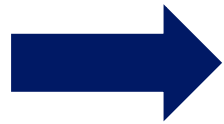
Macro Unit Test Tool

Consider macro to be a black box

- Give input data
- Run macro
- Pre-specify expected results
- Compare results



Expected results
(pre-specified)



Actual results
(added columns)

Why use Unit Testing

- Build trust in your code
- Develop specification
- Simple testing
- Re-usable testcases
- Prevent errors or side effects
 - when fixing bugs or
 - extending functionality
- Automated tests
- Continuous Integration



What is MUTT

MUTT (Macro Unit Test Tool) is

- A framework for testing SAS macros
- A SAS program that can execute unit tests
- Test Data (specification) created in Excel
 - Create test data without coding
 - Separate code and tests
- Creates feedback on quality
 - SAS log
 - Output dataset

How does it work?

- Create specification (Excel)
 - Store in **data > unittest**
- Create macro
 - Store in **program > macros**
- Execute MUTT
 - Check results in SAS log
 - Results stored in **data > unittest > result > [macroname]**

nn9932 > general > current > stats > data > unittest		
Name	Date modified	
result	28-Mar-22 12:44 PM	
adamimptdlastvisit.xlsx	01-Apr-22 8:55 AM	
adamimptmafdtm.xlsx	10-Mar-22 11:38 AM	
adamimptmearly.xlsx	10-Mar-22 12:10 PM	
misccalculations.xlsx	01-Apr-22 11:09 AM	

nn9932 > general > current > stats > program > macros		
Name	Date modified	Type
adamimptdlastvisit.sas	25-Feb-22 1:05 PM	SAS File
adamimptmafdtm.sas	24-Feb-22 1:08 PM	SAS File
adamimptmearly.sas	25-May-21 1:12 PM	SAS File
adammkadur.sas	21-Mar-22 9:00 AM	SAS File
adampktail.sas	19-Feb-22 3:05 PM	SAS File
comparev01.sas	17-Feb-22 3:57 PM	SAS File
misccalculations.sas	28-Mar-22 4:16 PM	SAS File

nn9932 > general > current > stats > data > unittest > result >		
Name	Date modified	
adamimptdlastvisit	01-Apr-22 11:09 AM	
adamimptmafdtm	01-Apr-22 11:09 AM	
adamimptmearly	10-Mar-22 12:10 PM	
misccalculations	01-Apr-22 11:09 AM	

The Specification

Specification is done in Excel

- Test Suite ([Setup] pane)

nn9932 > general > current > stats > data > unittest

Macro signature

```

macro misccalculations(
    input_ds =
    , input_var = x
    , output_ds = &input_ds._out
);
  
```

testsuite	Data	Parameters	keys	OUTPUTDATA	SupplementData	Enable
misccalc_small_numbers	input	input_ds = input , output_ds = misccalc_result		misccalc_result		TRUE
misccalc_large_numbers	linput	input_ds = linput , output_ds = misccalc_lresult		misccalc_lresult		TRUE

setup input linput +

A	B	C	D	E
TEST_NAME	x	always_two	sum_self	Run_testcase
Value two	2	2	4	TRUE
Value three	3	2	6	TRUE
Value four	4	2	8	TRUE

setup input linput +

- Test cases (other panes)

Demo

miscalculations.sas - * MUTT_DEMO - SAS Enterprise Guide

FileEditViewProgramToolsHelp

misccalculations.... x

RunCancel

CodeLog

33 %macro misccalculations(
34 input_ds =
35 , input_var = x
36 , output_ds = &input_ds._out
37);
38 data &output_ds.;
39 set &input_ds.;
40 always_two = 2;
41 sum_self = &input_var. + &input_var.;
42 run;
43 %mend;
44 %testmacros();
45

Output Data (0)

Run code to generate output data.

Project: MUTT_DEMO C:\Users\mhzt\OneDrive - Novo Nordisk\projstat\nn9932\general+ cur... Line 33, Col 25 Zoom: 190% Ready 4 9

Will Unit Testing Save the Day?

WARNING: Garbage in – Garbage out

- Verify test cases and test data
 - Boundary conditions
 - Special cases
 - Extended functionality
- Currently only for macros adding data to input

Future perspectives

- (Re-)introduction of Standard Macros
 - Less review
 - Higher quality
 - Fewer resources
- Automation: True continuous integration
 - Automatic testing when committing code
 - Reject of commits if tests fail

Future perspectives

- Test driven design and development process
 - Think test cases while preparing functionality
 - will trigger further detailing
 - Documentation
 - Sharing knowledge

Conclusion

- Use MUTT to unit test macros without coding
 - Easy to use
 - Higher quality
 - Safer to update
 - Automated testing
- Beware of tests cases
 - can be a complex discipline
 - needs to be maintained/updated

Thank you

Questions or comments?

Michael Hedegaard Thomsen
mhzt@novonordisk.com