



A modular approach to automating the program promotion process

On an existing standards library

Lino Jardim, MSD, Zurich, Switzerland

Carl Herremans, MSD, Brussels, Belgium

David Maher-McWilliams, MSD, London, UK

For presentation at PHUSE SDE Copenhagen 2023



Agenda

- 1 Introduction
- 2 Manual process
- 3 Automated process
- 4 Conclusion

Introduction

Standards libraries

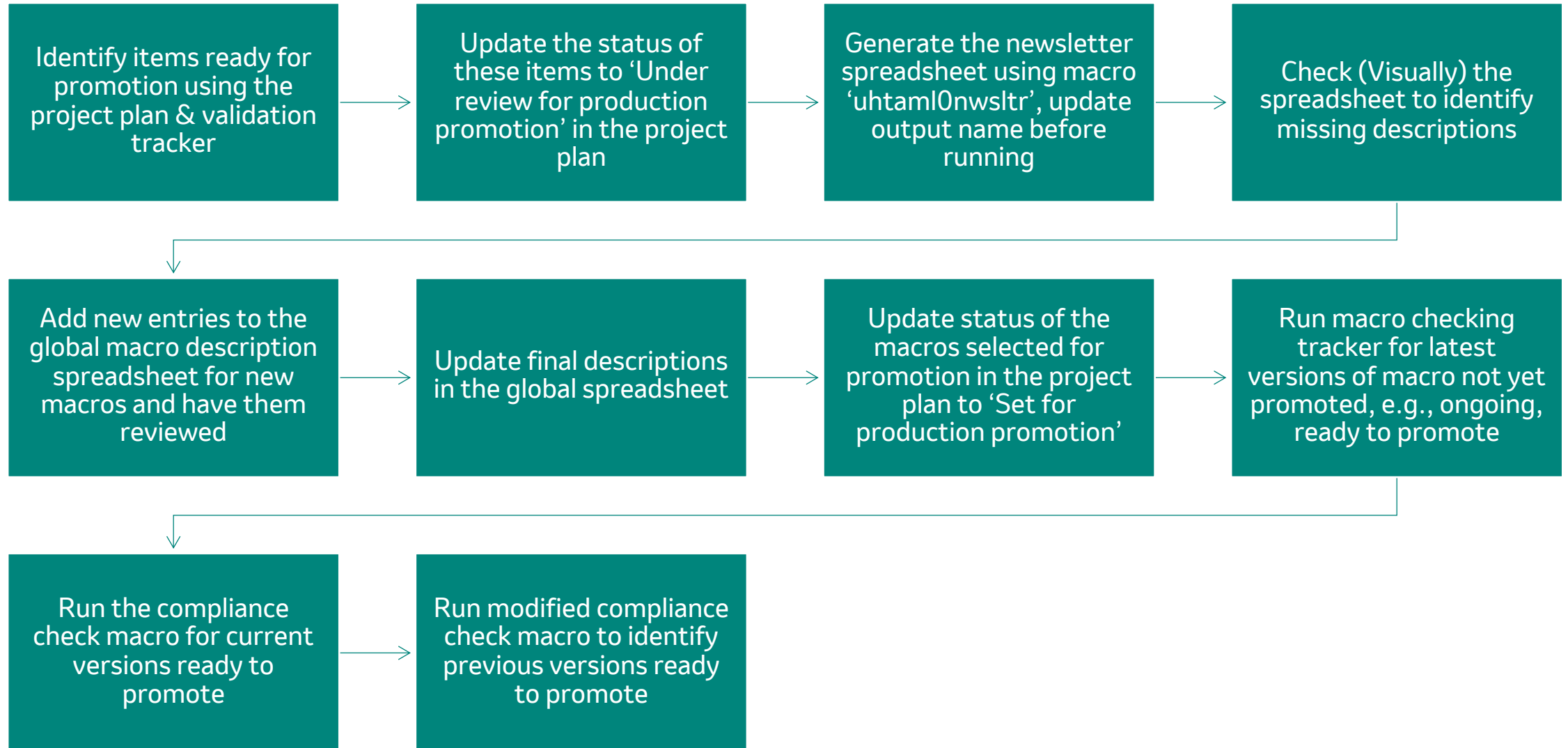
Standards libraries are used to ensure cost-effective reproducibility, consistency and compliance across deliveries and projects.

Before releasing standard macros into production, various due diligence checks need to be conducted to ensure that these macros are ready for deployment.

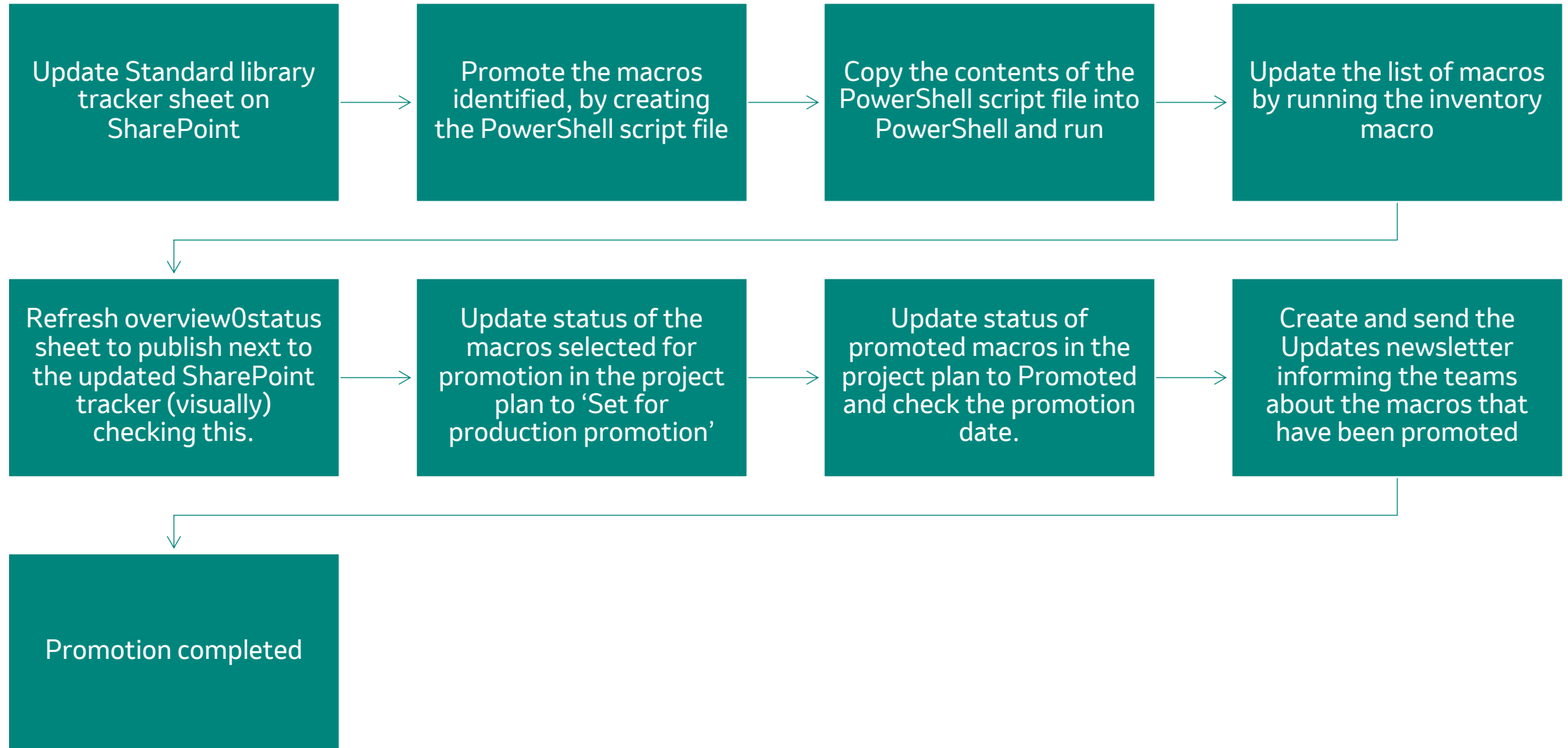


Manual Process

Manual promotion process part 1



Manual promotion process part 2

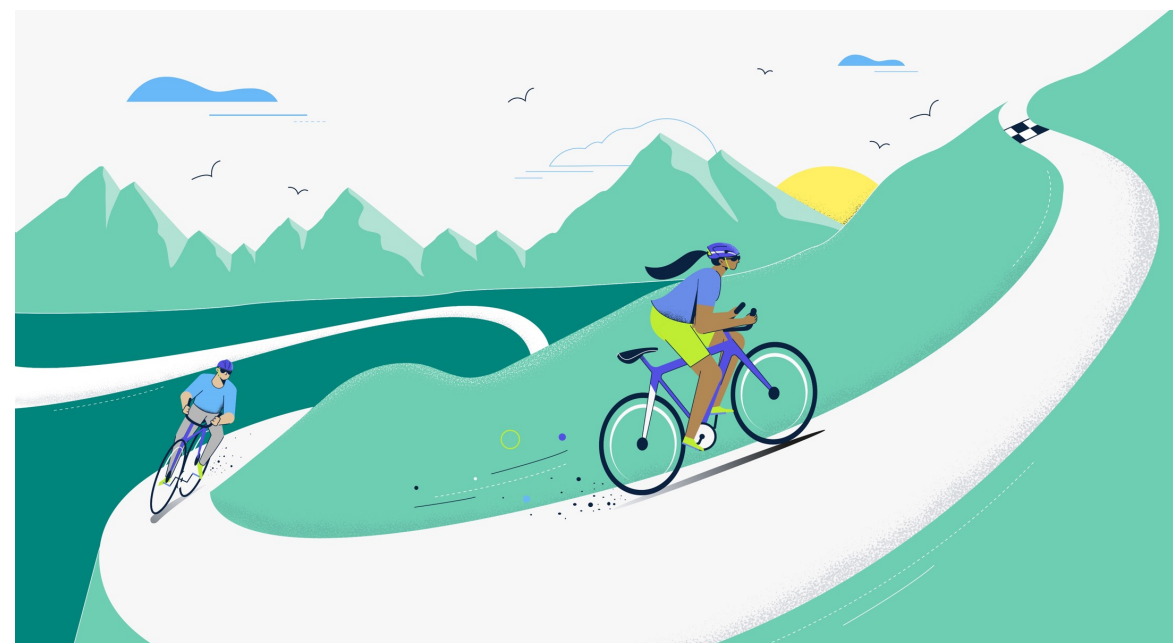


Manual promotion process ... Conclusion

As illustrated by the exhaustive list, the promotion process has been utilizing multiple stand-alone macros, sheets and trackers taking care of specific tasks to facilitate promotion.

There has however been no clean link between these steps to form an end-to-end streamlined promotion process, that didn't rely heavily on human intervention.

This has been an ongoing uphill battle faced during each promotion cycle.



Automated process

Key considerations to facilitate automation

Leverage on existing documentation, tracking the tasks on the standards library.

Leverage on existing macros, enhancing them where necessary. Creating new macros only when needed.

The automation focuses on the steps after the macros have been developed, independent tested and the example packages created. Focusing on the promotion process.

SharePoint site used to track macro versions has been retired, eliminating ~20% of the steps.



What do we need and how do we get it..?

Identification and Compliance

Create a list of the macros to be promoted

Check the entries of these items in the project plan as well as the validation change request log

Do compliance checks on each of the macros to ensure that all the manual process steps have been followed

If any of the items have failed compliance, highlight these and provide details on why they have failed

Information on updates to share

Leveraging on the list of macros, create a newsletter sharing the description of the macro, type of update and information detailing the update

Check that all the macros identified have a description in the global macro spreadsheet. If not prompt the user to update the global spreadsheet

Generate an rtf version of the newsletter to share with staff

Archive it

Lock down the current version of the macros, including the documentation, validation artifacts and example packages

Copy to a “holding area” in a date stamped folder containing all items pertaining to the promotion, including validation artifacts

Deploy it

Copy Relevant items including documentation ready for promotion to the production areas respective folders

Once copy is complete send out the newsletter detailing the macros and the associated updates

Update the project plan indicating that the macros have been promoted

Identification and compliance: Check

Once the development, independent testing and example package of a macro is complete the project plan's status for this item is updated to '07_ready for production_promotion'

Deliverable category	Deliverable	Programming_Status	Status flag	PRS Assigned to	PRS assigned	PRS Target date	PRS Actual delivery date	PRS approved	Programmer Assigned to	Assigned on	Start date	Target date	Actual Delivery Date
macrolib	a0example0macro	07_ready for production_promotion	New	Lino	14/Jul/2023	15/Jul/2023	17/Jul/2023	17/Jul/2023	Lino	18/Jul/2023	18/Jul/2023	19/Jul/2023	19/Jul/2023
macrolib	hta0example0macro	07_ready for production_promotion	New	David	18/Jul/2023	19/Jul/2023	19/Jul/2023	19/Jul/2023	David	20/Jul/2023	20/Jul/2023	21/Jul/2023	25/Jul/2023
macrolib	hta0example0macro1	07_ready for production_promotion	New	Carl	02/Aug/2023	03/Aug/2023	03/Aug/2023	04/Aug/2023	Carl	04/Aug/2023	04/Aug/2023	07/Aug/2023	07/Aug/2023

Deliverable category	Deliverable	Programming_Status	Status flag	Indep testing Programmer Assigned	Indep testing assigned	Indep testing target start date	Indep testing Target date	Actual Indep testing date	QC Sample code Programmer Assigned	QC Sample code Assigned on	QC Sample code Target date	QC Sample code Actual delivery date
macrolib	a0example0macro	07_ready for production_promotion	New	David	19/Jul/2023	19/Jul/2023	20/Jul/2023	20/Jul/2023	Carl	21/Jul/2023	21/Jul/2023	21/Jul/2023
macrolib	hta0example0macro	07_ready for production_promotion	New	Carl	25/Jul/2023	26/Jul/2023	27/Jul/2023	28/Jul/2023	Lino	28/Jul/2023	31/Jul/2023	31/Jul/2023
macrolib	hta0example0macro1	07_ready for production_promotion	New	Lino	08/Aug/2023	09/Aug/2023	11/Aug/2023	11/Aug/2023	David	14/Aug/2023	15/Aug/2023	15/Aug/2023

The hta0sl0promote0list macro is called:

```

*Call to macro to create the list of macros that are ready for promotion;
%hta0sl0promote0list(deli_v_cat=macrolib
,prog_status=07_ready for production_promotion
,location=Usermanuals0Macros);
    
```

Identification and compliance: Results

Examples of outputs produced by running the hta0sl0promote0list macro

Log identifies issues by means of user generated error:

```
hta0example0macro1
***** Problem in Macro HTA0SL0PROMOTE0LIST call *****
ERROR: The following macros have not passed the compliance checks: hta0example0macro1
ERROR: Check the lptda.faillist and lptda.pdvtfail datasets for additional information
ERROR: Please correct these issues and rerun or update the project plan to remove these items
*****
```

Additional information on what checks have failed:

MACRO	MESSAGE	TYPE
hta0example0macro1	File devtest0hta0example0macro1.log is expected in folder /validation/dev-test/output	DEVTEST
hta0example0macro1	File devtest0hta0example0macro10bulletproof.log is expected in folder /validation/dev-test/output	DEVTEST
hta0example0macro1	File devtest0hta0example0macro10bulletproof.sas is expected in folder /validation/dev-test/pgm	DEVTEST
hta0example0macro1	File has no round defined in program validation tracker	DEVTEST
hta0example0macro1	File indtest0hta0example0macro1.log is expected in folder /validation/ind-test/output	INDTEST
hta0example0macro1	File indtest0hta0example0macro10bulletproof.log is expected in folder /validation/ind-test/output	INDTEST
hta0example0macro1	File indtest0hta0example0macro10bulletproof.sas is expected in folder /validation/ind-test/pgm	INDTEST
hta0example0macro1	File has no round defined in program validation tracker	INDTEST
hta0example0macro1	File call0hta0example0macro1.log is expected in folder /sample/output	SAMPLE
hta0example0macro1	File call0hta0example0macro1.sas is expected in folder /sample/pgm	SAMPLE
hta0example0macro1	File has no round defined in program validation tracker	SAMPLE

Compliance report is written out showing the compliant macros:

	DELIVERABLE	VERSION	PROM_VER
1	a0example0macro	1	1
2	hta0example0macro	1	1

Identification and compliance: Acknowledge

The macro also creates a contributor spreadsheet.

The sheet contains all the information identifying each employee's contribution and the durations of the tasks are also calculated. This information is used in 2 keyways.

Firstly, to acknowledge and recognize the employees for their contributions to the advancement of standards

And

Secondly, the summary of these durations can be used as a benchmark to determine average development and Independent testing times.

Below is an example of the contributor sheet produced during this demo.

CATEGORY	DELIVERABLE	PRS_PROG	DEV_PROG	IND_PROG	SAM_PROG	PROM_VER	PROD_DUR	DEV_START	DEV_ACT	IND_DUR	IND_START	IND_ACT
macrolib	a0example0macro	Jardim, Lino	Jardim, Lino	Maher-Mcwilliams, David	Herremans, Carl	1	2	18JUL2023	19JUL2023	2	19JUL2023	20JUL2023
macrolib	hta0example0macro	Maher-Mcwilliams, David	Maher-Mcwilliams, David	Herremans, Carl	Jardim, Lino	1	4	20JUL2023	25JUL2023	3	26JUL2023	28JUL2023
macrolib	hta0example0macro1	Herremans, Carl	Herremans, Carl	Jardim, Lino	Maher-Mcwilliams, David	1	2	04AUG2023	07AUG2023	3	09AUG2023	11AUG2023

Information on updates to share: Verify

With the list of compliant macros, start putting together the information.

By running the u0htaml0nwsltr macro

```
*Call to macro to create the newsletter to be shared with the staff detailing the macros promoted;
%u0htaml0nwsltr(htaml_project_plan= Path to project plan
,htaml_macro_desc= path to the global macro description file
,output_filename=Outputpath/filename
);
```

If there are new macros without descriptions in the global description file, the following error is presented:

```
*===== Problem in Macro UOHTAML0NWSLTR call =====*
ERROR: The descriptions for te following macro(s): a0example0macro hta0example0macro
       not present in the Macro Description document
       Please add an entry in the Macro Description document located at:
       path to document
*=====*
```

Once the global description file has been updated with the entries this error will be resolved.

The newsletter containing the details of the macros will be created.

Archive it

Next the u0htasl0copy0promotion macro is run.

```
*Call to macro that copies the macros ready for production to the promotion area;
%u0htasl0copy0promotion(htaslroot= Path to the standards library development area root directory
                        ,prompath= Path to the holding area where the promotion is copied to
                        ,promdate= Date of the promotion, will use system date if left blank
                        ,createprev=Y/N Flag to lock down the current version
                        );
```

The macro takes care of the following:

Copies the macro, Documentation, Validation artifacts and example package to a holding area

Creates a locked down version copy for the promoted macros

hta0example0macro > priorversion >

Name
hta0example0macro0v01

a0example0macro > priorversion

Name
a0example0macro0v01

<< 20230815 > macros

20230815

- macros
- usermanuals
 - a0example0macro
 - sample
 - hta0example0macro
- validation
 - a0example0macro
 - validation
 - dev-test
 - ind-test
 - input
 - output
 - pgm
 - hta0example0macro

Name

- a0example0macro.sas
- hta0example0macro.sas

Deploy it

Once macros have been copied deploy the newsletter.

BARDS HTA Standards Library

Updates

15 August 2023

Dear Team,

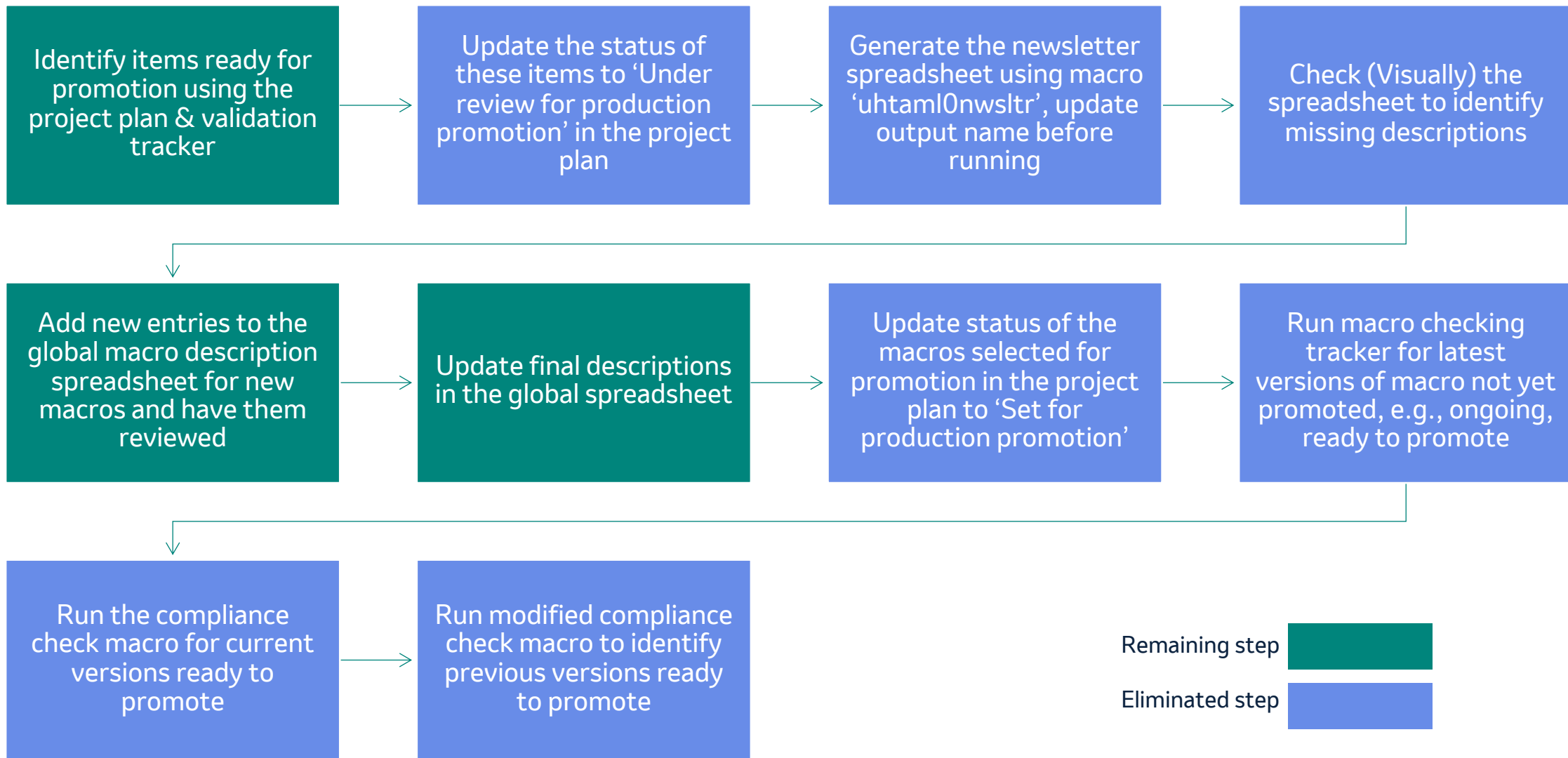
HTA Standard Library on BARDS-CPI has just been updated with new or improved HTA macros and associated metadata. Please use this platform to download the code you need in your assigned HTA projects. A List of Library Entries is available on BARDS CPI TA Production area to guide you across the available macros.

You will find details on the library updates in the following table:

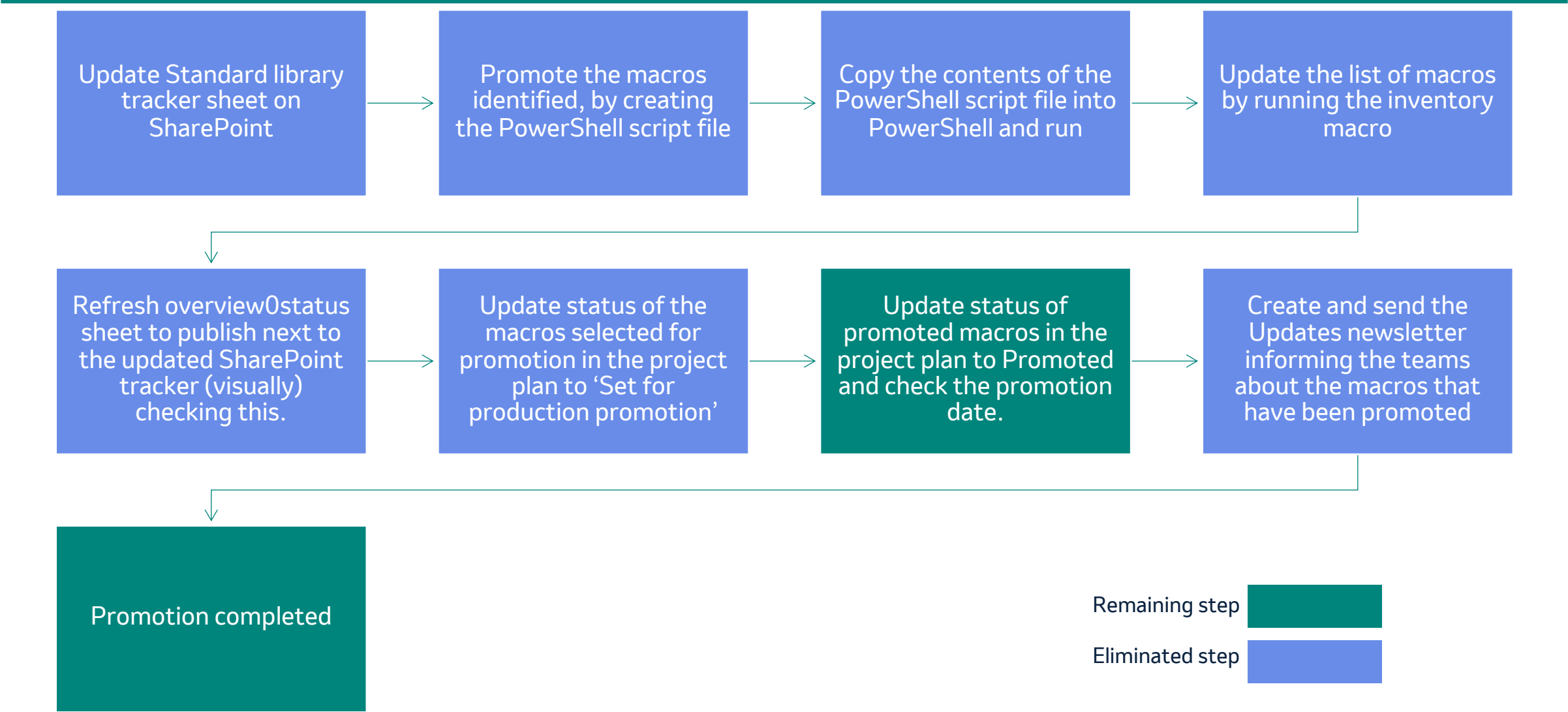
HTA Macro Name	Functionality	Update	Macro Type
a0example0macro	Simple analysis macro example	***New macro***	Statistical/Analysis
hta0example0macro	Simple reporting macro example	***New macro***	Reporting

Conclusion

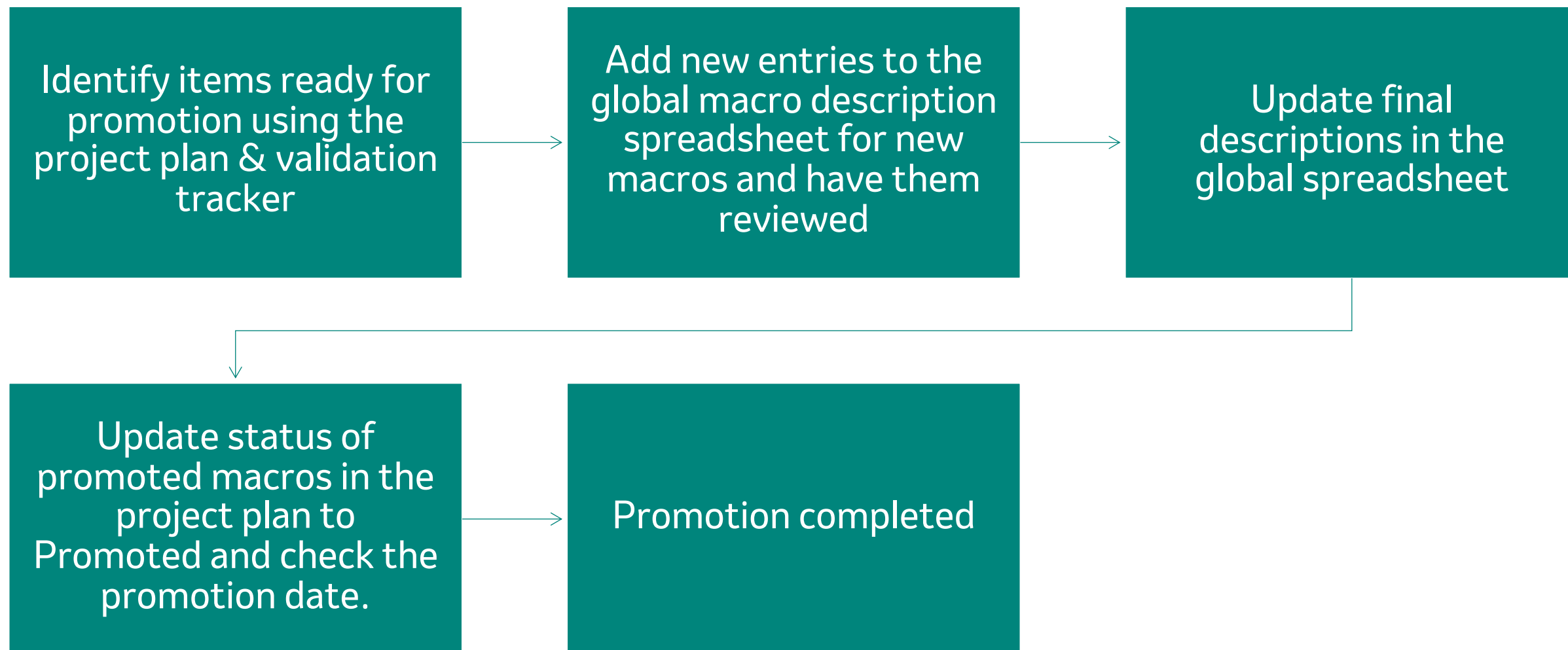
Manual promotion process recap



Manual promotion process recap.... continued



Automated process



Closing remarks

The Automation of previously manual steps has:

- reduced the promotion time form days to hours,
- reduced the complexity of the process,
- increased the traceability.





Thank you



We welcome any suggestions or observations:

Lino Jardim, lino.jardim@msd.com

Carl Herremans, carl_herremans@msd.com

David Maher-McWilliams, david.maher-mcwilliams@msd.com