



# **SM06: Multi-study programming**

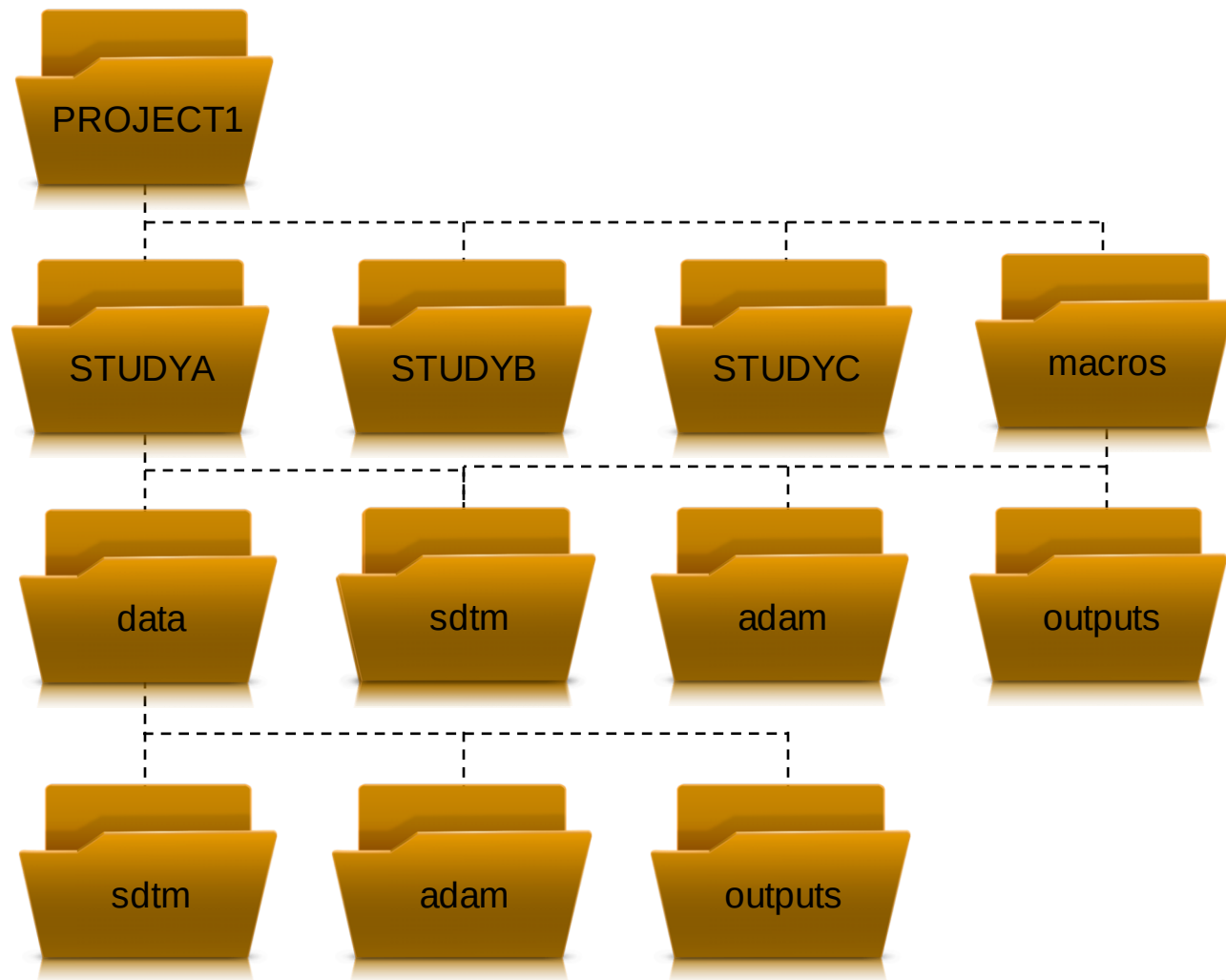
A best practice outline

Simon Purkess & Matt Metherell

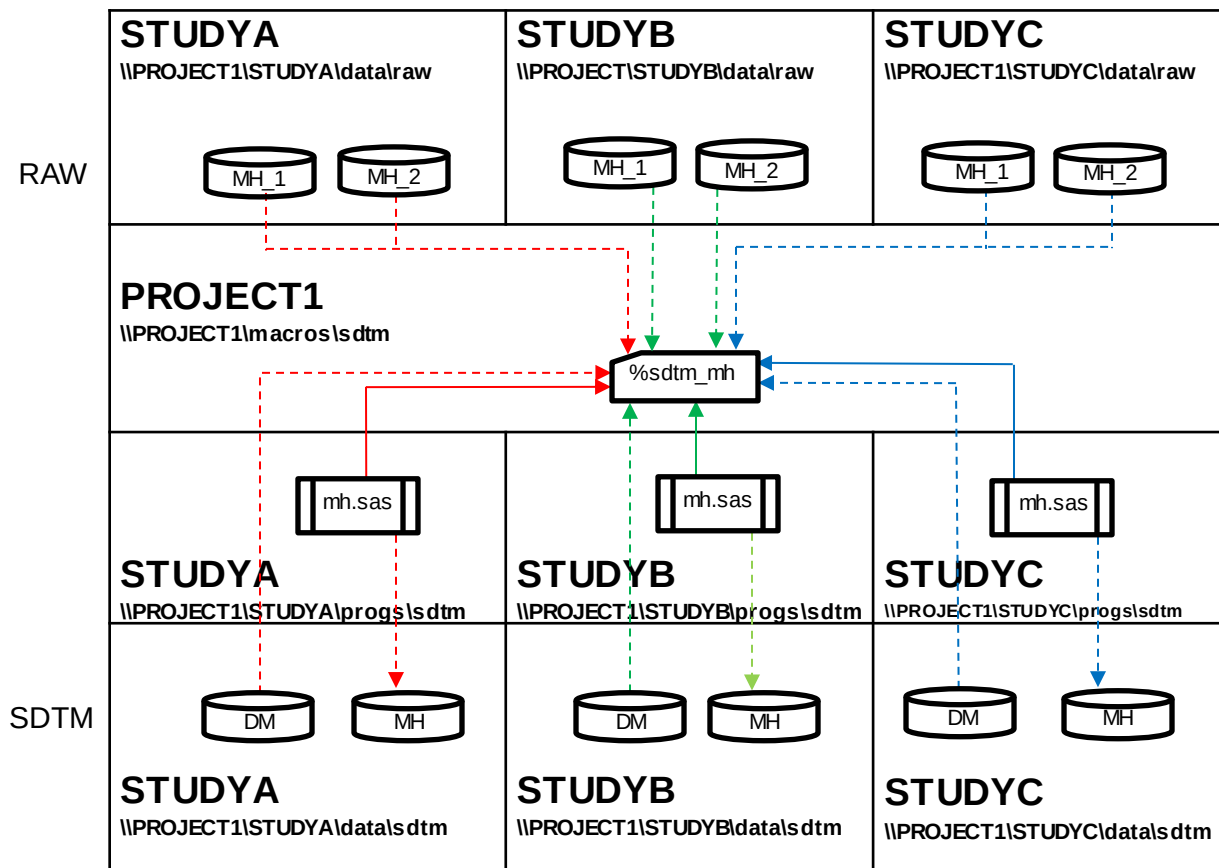
# Introduction

- When multiple studies are nearly identical and run in parallel, it can save a lot of effort if we can simply update one set of code that is used by all relevant studies.
- This can bring its own set of problems to overcome, so in this presentation we lay out some principles we recommend adhering to.

# Folder structure



# SDTM



# I Setting up for programming

A %setup\_program macro is used to assign several things:

- Useful macro variables such as the study, to allow the central macros to control the program flow based on the study.
- The libraries and macro folders that the programs can draw upon.
- Any global options you wish to declare.
- Applying format catalogues.

You could set these up explicitly in each area, or dynamically in a central area, but either way this will allow the central code to work across all areas.

# Setting up for programming

```
%macro setup_program;
```

```
  %*Define macro variables based on directory structure;
```

```
  %global project study;
```

```
  %let rc = %sysfunc(filename(fr,.));
```

```
  %let project = %scan(%sysfunc(pathname(&fr)),3,\);
```

```
  %let study = %scan(%sysfunc(pathname(&fr)),4,\);
```

```
  %let rc = %sysfunc(filename(fr));
```

```
  %*Macro library;
```

```
  filename sdtmmac "\\&project.\macros\sdtm";
```

```
  options mautosource
```

```
          sasautos = (sdtmmac sasautos);
```

```
  %*Study libraries;
```

```
  libname raw      "\\&project.\&study.\data\raw" access=readonly;
```

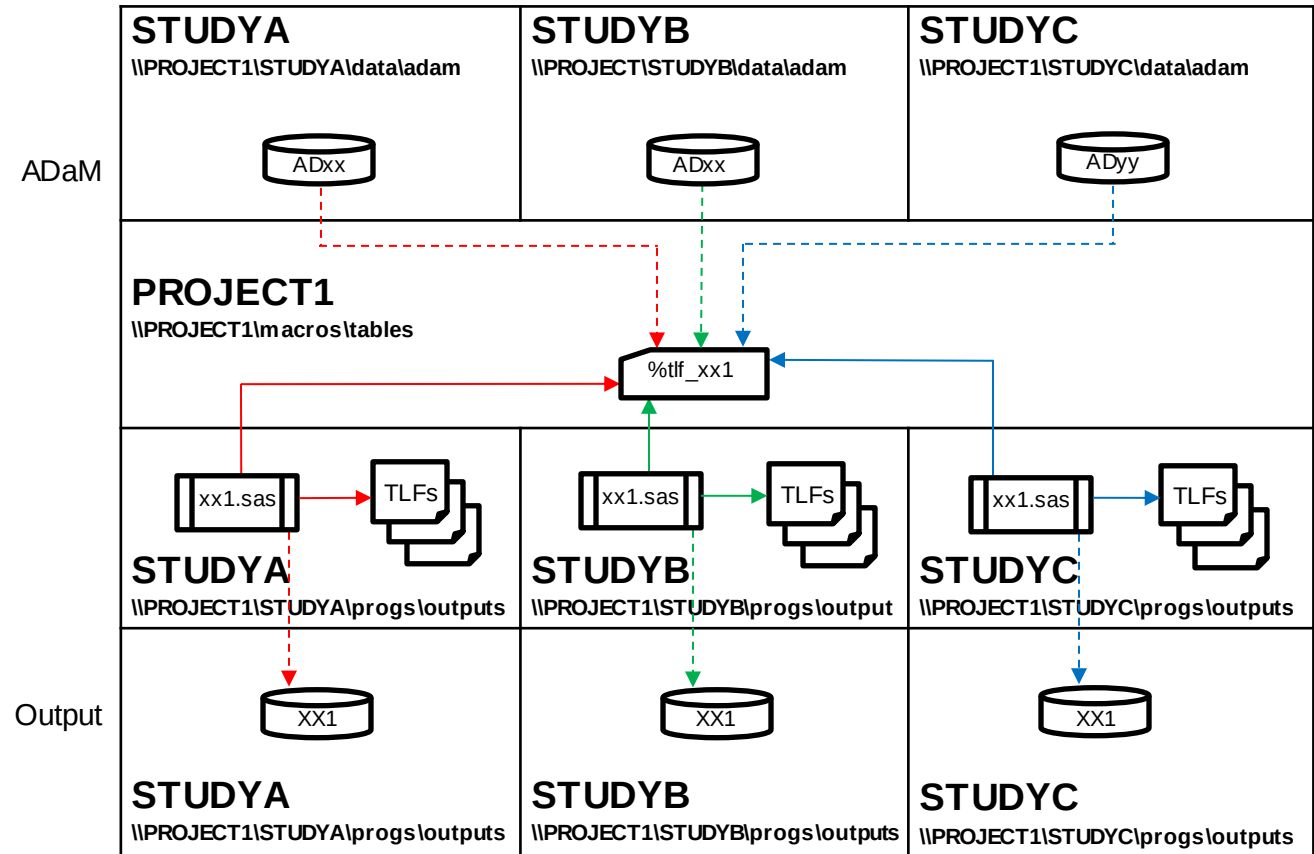
```
  libname sdtm     "\\&project.\&study.\data\sdtm";
```

```
%mend setup_program;
```

# Programs

| STUDYA                                                                                                                                                                                                                                                                                                                                                                                                              | STUDYB                                                                                                                                        | STUDYC                                                                                                                                        |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>\\PROJECT1\\STUDYA\\progs\\sdm  %*Set-up program; %setup_program;  %*Run central macro; %sdm_xx();  %*Close program; %end_program;</pre>                                                                                                                                                                                                                                                                       | <pre>\\PROJECT1\\STUDYB\\progs\\sdm  %*Set-up program; %setup_program;  %*Run central macro; %sdm_xx();  %*Close program; %end_program;</pre> | <pre>\\PROJECT1\\STUDYC\\progs\\sdm  %*Set-up program; %setup_program;  %*Run central macro; %sdm_xx();  %*Close program; %end_program;</pre> |
| <pre>PROJECT1 \\PROJECT1\\macros\\sdm  %*Macro to create XX dataset; %macro sdm_xx() / minoperator;  %*Read in raw dataset and derive variables; data xx;   length domain \$8 usubjid \$20;  %if &amp;study.=STUDYA %then %do;   set raw.xx_1; %end; %else %if &amp;study. in STUDYB STUDYC %then %do;   set raw.xx_2; %end;    domain   = "XX";   usubjid  = catx("-", studyid, subjid); run;  %mend sdm_xx;</pre> |                                                                                                                                               |                                                                                                                                               |

# TLFs



# I TLF Programming

- TLFs (Tables, Listings and Figures) for a project can be programmed using a similar program/macro structure
- The same macro can be called multiple times parsing different parameters, e.g. populations, subgroups for analysis
- %if, %else conditional blocks useful if TLFs are similar but not the same across studies (e.g. different number of study arms), or between outputs.
- Project may have a mix of study-specific and repeated TLFs – use shared macros only where appropriate

# TLF Programming

| STUDYA<br>\\PROJECT1\STUDYA\progs\sdm                                                                                                                                                                                                                                                                                                    | STUDYB<br>\\PROJECT1\STUDYB\progs\sdm                                                                                                                                                 | STUDYC<br>\\PROJECT1\STUDYC\progs\sdm                                                                                                                      |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> %*Set-up program; %setup_program;  %*Run central macro; %tlf_xx(num=1, pop=saffl); %tlf_xx(num=2, pop=fasfl);  %*Close program; %end_program; </pre>                                                                                                                                                                               | <pre> %*Set-up program; %setup_program;  %*Run central macro; %tlf_xx(num=1, pop=saffl); %tlf_xx(num=2, pop=fasfl); %tlf_xx(num=3, pop=ppsfl);  %*Close program; %end_program; </pre> | <pre> %*Set-up program; %setup_program;  %*Run central macro; %tlf_xx(num=1, pop=saffl); %tlf_xx(num=2, pop=fasfl);  %*Close program; %end_program; </pre> |
| PROJECT1<br>\\PROJECT1\macros\sdm                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                       |                                                                                                                                                            |
| <pre> %*Macro to create XX output; %macro tlf_xx(num=, ) / minoperator;  %*Create numbered dataset for each output; data output.xx_&amp;num.;  %if &amp;study. in STUDYA STUDYB %then %do;     set adam.adxx; %end; %else %if &amp;study. = STUDYC %then %do;     set adam.adyy; %end;  where &amp;pop.= "Y"; run;  %mend tlf_xx; </pre> |                                                                                                                                                                                       |                                                                                                                                                            |

# I Running Programs

Unfortunately, if we need to make an update for one study, we will often need to re-run the code across all of them, so it is important to make this as easy as possible.

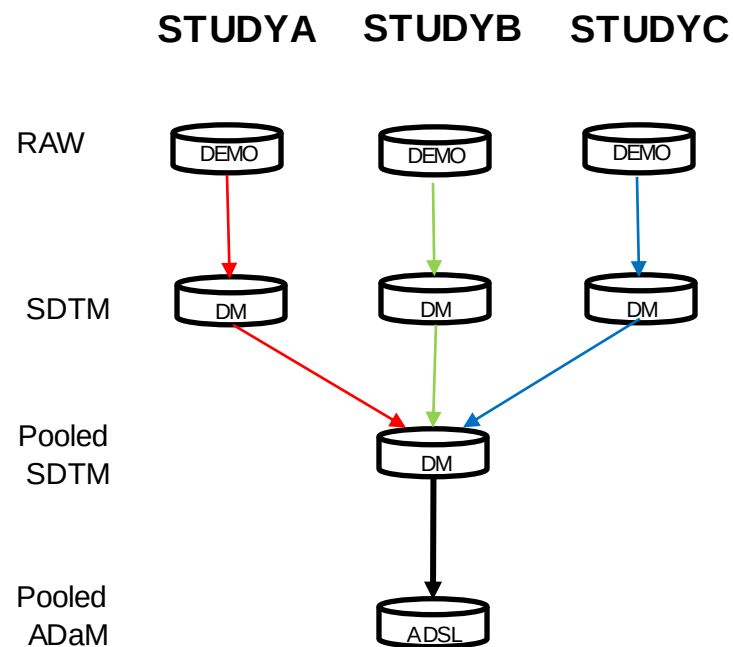
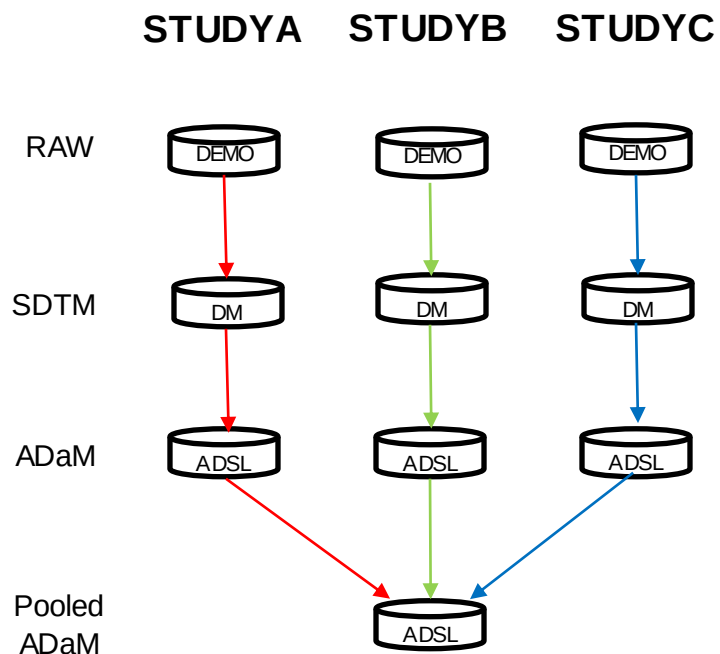
Therefore a macro that can run a program across all studies is really useful, especially if you can configure it to:

- Run both prod and qc when required
- Check both logs
- Check the compare
- Run all programs in a folder (eg all SDTMs)

# Pooling Studies

- Combined analysis across multiple studies (e.g. ISS/ISE) becomes easier if the study datasets are similar.
- A pooling step at SDTM or ADaM allows you to treat combined project datasets as if they were from a single study.
- Study-specific parameters can be added to ADaM BDS datasets if required

# Pooling at ADaM level vs Pooling at SDTM level



# Admin

## Possible Issues

- Any update to a central macro means all studies need to be re-QC'd.
- The more studies differ, the more complex macros become
- Studies progressing at different rates can make macro development more challenging
- More complex to budget as lines are blurred between work on each individual study
- Changes of scope if new studies are added or existing studies altered

# Keeping Track

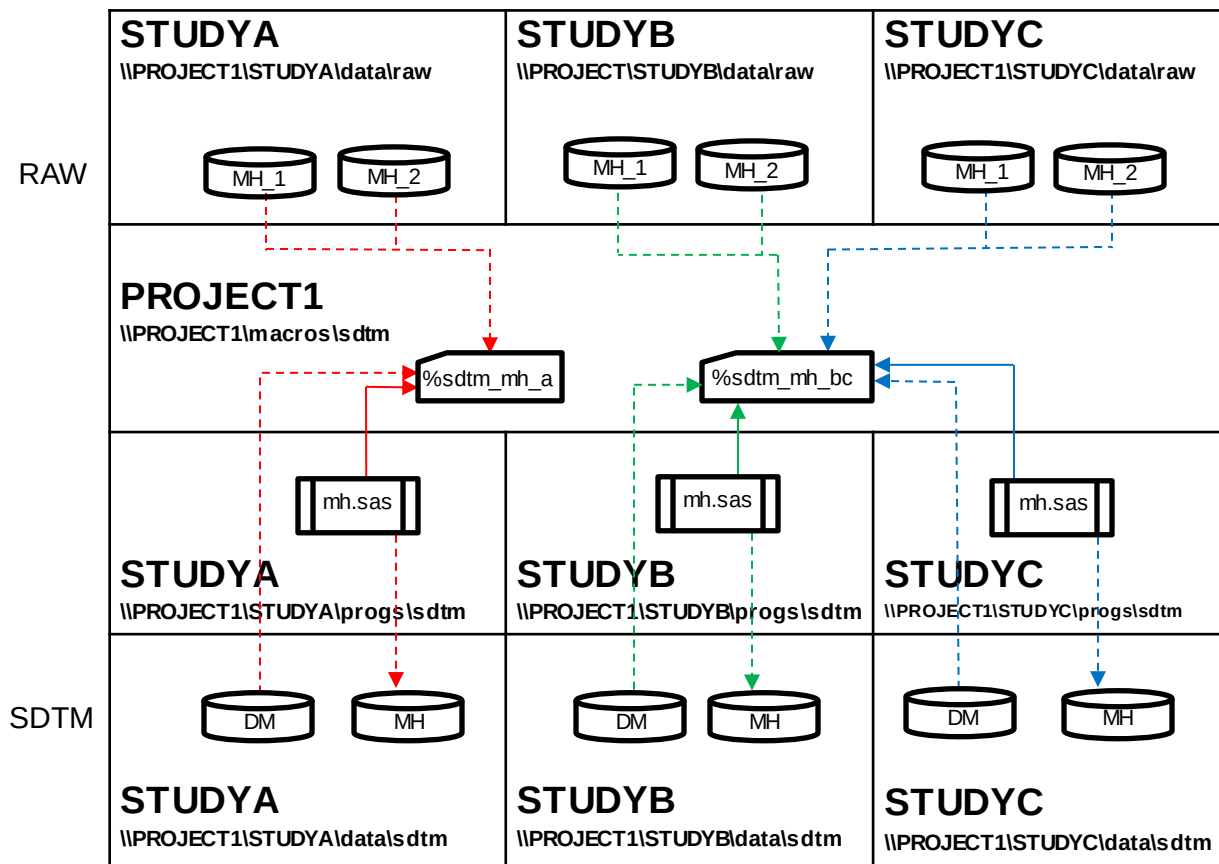
- As much as possible, maintain combined documents rather than multiple single-study versions
- Create archive of all programs and related documents after each study delivery

|              |                       |               | STUDYA          |          |             | STUDYB          |                      |             | STUDYC          |                        |             |
|--------------|-----------------------|---------------|-----------------|----------|-------------|-----------------|----------------------|-------------|-----------------|------------------------|-------------|
| Program Name | Production Programmer | QC Programmer | Completion Date | Comments | Status      | Completion Date | Comments             | Status      | Completion Date | Comments               | Status      |
| DM           | MM                    | SP            | 2021-07-01      |          | Completed   | 2021-07-01      |                      | Completed   |                 | Data not yet available | Not Started |
| SE           | MM                    | SP            | 2021-07-04      |          | Completed   | 2021-07-05      |                      | Completed   |                 | Data not yet available | Not Started |
| DS           | MM                    | SP            | 2021-07-06      |          | Completed   |                 | DSDECOD not matching | Started     |                 | Data not yet available | Not Started |
| EX           | MM                    | SP            |                 |          | Not Started |                 |                      | Not Started |                 | Data not yet available | Not Started |

# I Keeping Archives

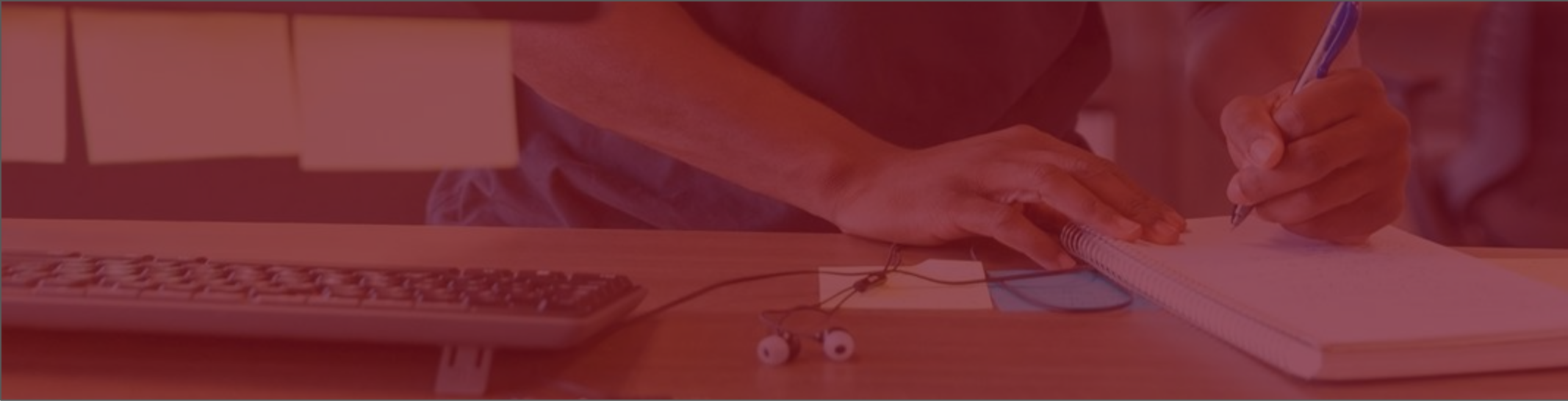
- Best practice to store a full copy of all programs at every delivery, either manually or using version control software
- Updates for one study could inadvertently affect other studies – comprehensive archive may help fix any issues
- Storing a working snapshot of the programs 'as they were' at delivery important for audit etc.

# Splitting off



# I Splitting Off

- If studies diverge over time, maintaining central macros can become harder to do efficiently. There may come a point where it makes more sense to split them. This could be either at an individual program level or for the whole studies
- Advantages:
  - Allows for more specialised code to be written without using lots of complex macro logic
  - Removes risk of an update for one study breaking others
- Disadvantages:
  - One-way process – very difficult to re-merge once split
  - Project-wide updates have to be individually applied to each study



Thank you

Questions?

[simon.purkess@phastar.com](mailto:simon.purkess@phastar.com)

[matthew.metherell@phastar.com](mailto:matthew.metherell@phastar.com)