

PDF Multi-level Bookmarks Via SAS®

Steve Griffiths
PhUSE - Berlin 2010

Contents

- The Problem
- Proposed Methodology
- Initial Approach
- Secondary Approach
- Third (& Final) Approach
- Final Thoughts
- References

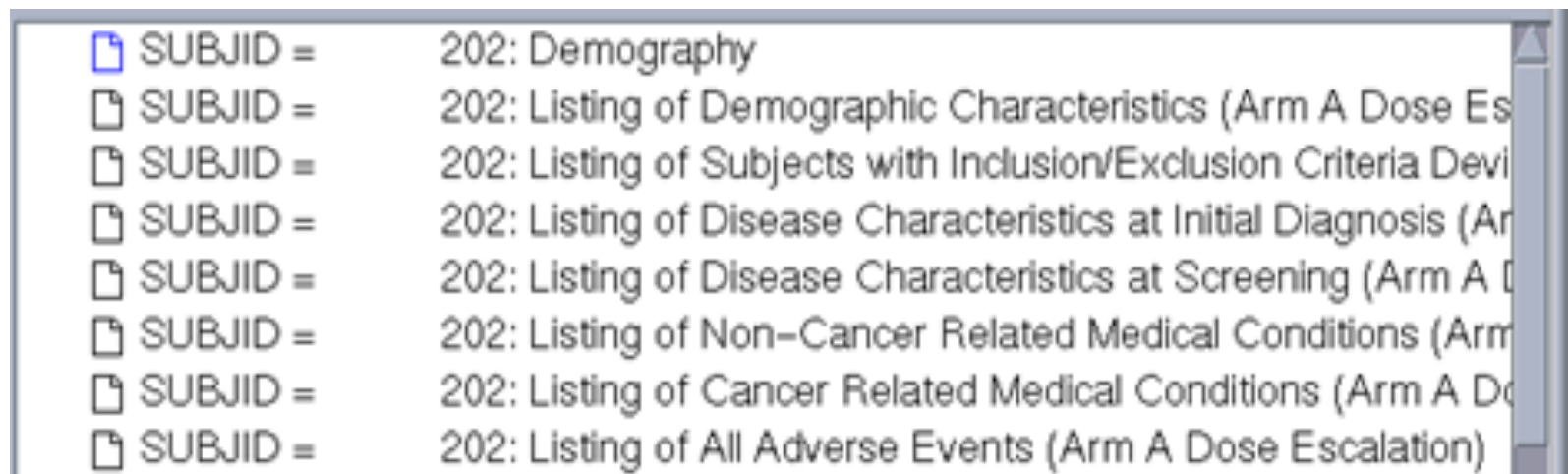
The Problem

One Level Bookmarks

- Title/text altered via ODS PROCLABEL statement

ODS noptitle;

ODS proclabel "SUBJID = 202: Demography";



A screenshot of a document's table of contents or bookmark list. The first entry is highlighted with a blue icon and text, while the others are greyed out. The highlighted entry is 'SUBJID = 202: Demography'. The other entries are 'SUBJID = 202: Listing of Demographic Characteristics (Arm A Dose Es...', 'SUBJID = 202: Listing of Subjects with Inclusion/Exclusion Criteria Devi...', 'SUBJID = 202: Listing of Disease Characteristics at Initial Diagnosis (Ar...', 'SUBJID = 202: Listing of Disease Characteristics at Screening (Arm A D...', 'SUBJID = 202: Listing of Non-Cancer Related Medical Conditions (Arm...', 'SUBJID = 202: Listing of Cancer Related Medical Conditions (Arm A D...', and 'SUBJID = 202: Listing of All Adverse Events (Arm A Dose Escalation)'.

	SUBJID =	202: Demography
	SUBJID =	202: Listing of Demographic Characteristics (Arm A Dose Es
	SUBJID =	202: Listing of Subjects with Inclusion/Exclusion Criteria Devi
	SUBJID =	202: Listing of Disease Characteristics at Initial Diagnosis (Ar
	SUBJID =	202: Listing of Disease Characteristics at Screening (Arm A D
	SUBJID =	202: Listing of Non-Cancer Related Medical Conditions (Arm
	SUBJID =	202: Listing of Cancer Related Medical Conditions (Arm A D
	SUBJID =	202: Listing of All Adverse Events (Arm A Dose Escalation)

The Problem

Two Level Bookmarks

- Possible to add via CONTENTS= statement within PROC REPORT
- `proc report data = temp contents='Table1'...;`



The Problem

Two Level Bookmarks

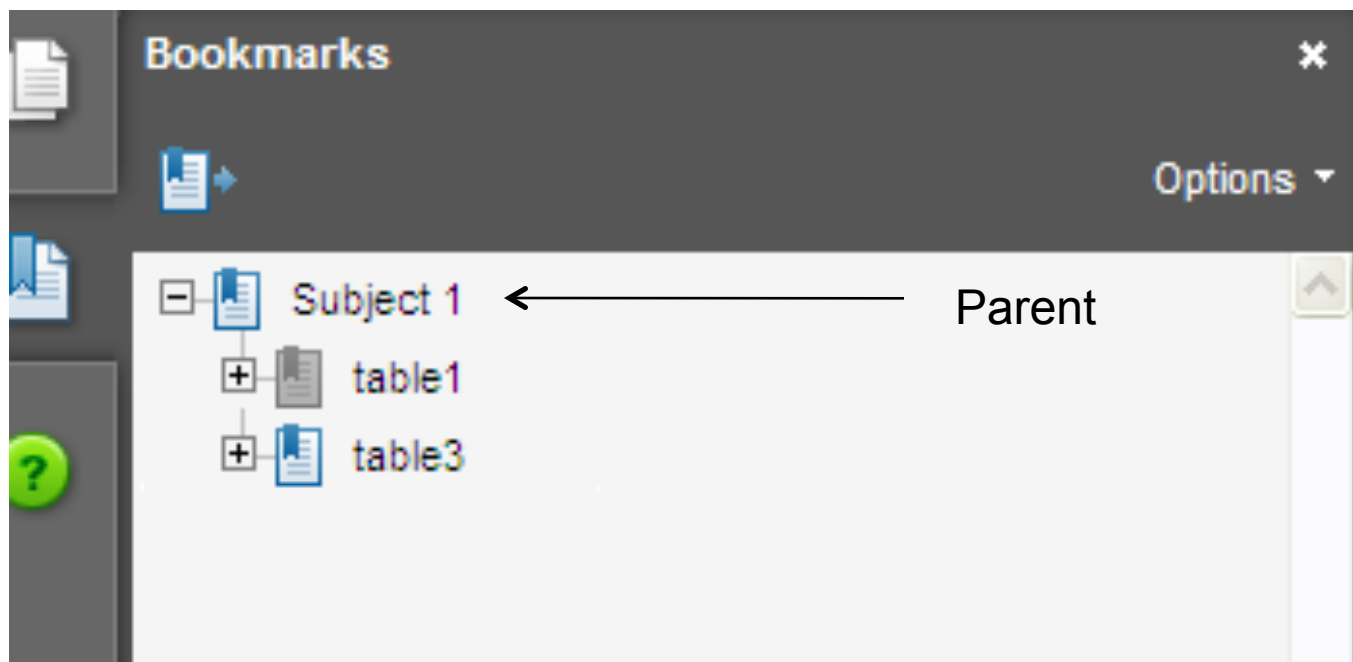
- Possible to add via CONTENTS= statement within PROC REPORT
- `proc report data = temp contents='Table1'...;`



The Problem

Two Level Bookmarks

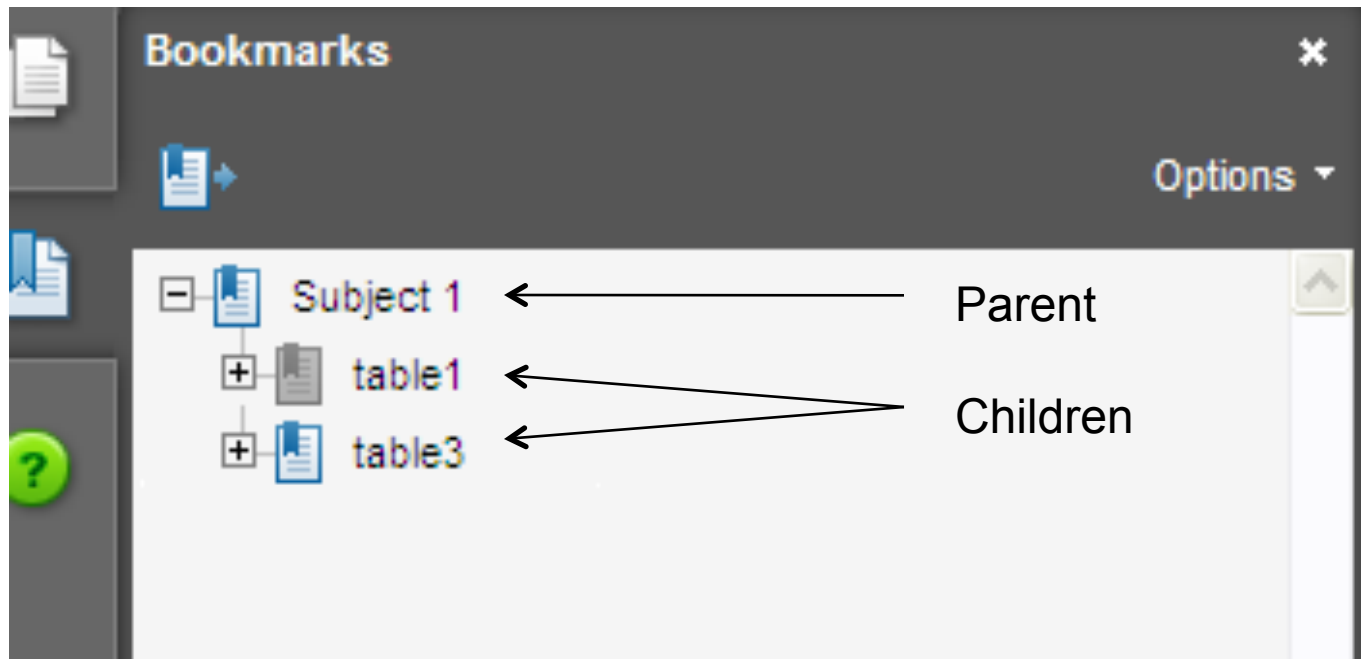
- Possible to add via CONTENTS= statement within PROC REPORT
- `proc report data = temp contents='Table1'...;`



The Problem

Two Level Bookmarks

- Possible to add via CONTENTS= statement within PROC REPORT
- `proc report data = temp contents='Table1'...;`



Proposed Methodology

- How is this achieved?
 - Direct manipulation of PDF file not possible
 - Append 'bookmark' information to Postscript file in form of PS language statements
 - Convert file using PS2PDF

Proposed Solution

- ‘Parent’ Language Statement
[/TITLE (*PARENT TITLE*)
/COUNT **NUMBER OF CHILDREN**
/PAGE **FIRST PAGE**
/VIEW [/XYZ *VALUE VALUE VALUE*]
/OUT PDFMARK
- ‘Child’ Language Statement
[/TITLE (*CHILD TITLE*)
/PAGE **FIRST PAGE**
/VIEW [/XYZ *VALUE VALUE VALUE*]
/OUT PDFMARK

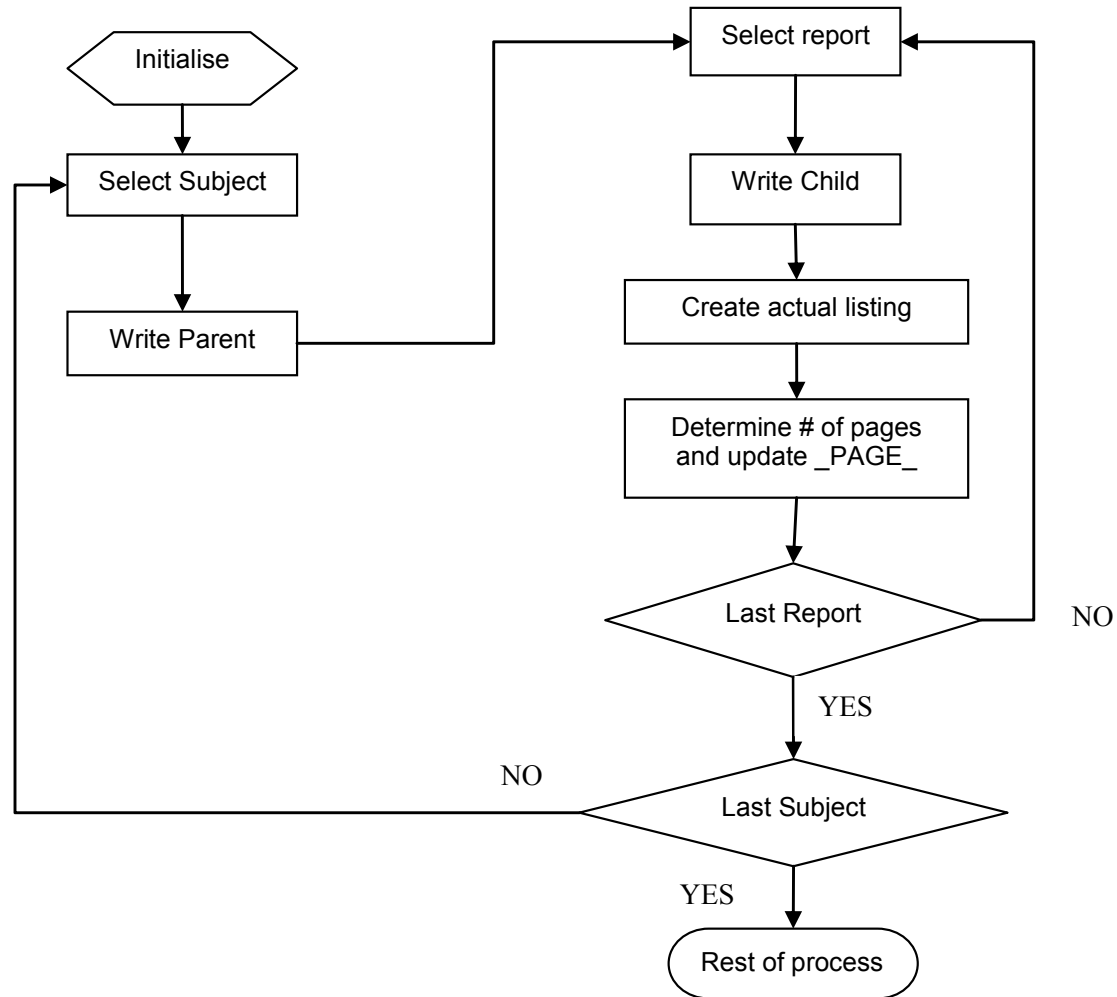
First Approach

- Restricted to small group of outputs.
- Initially needed to determine number of pages generated per report.
- Used differing versions of 'Page X of Y' macros using PROC REPORT and COMPUTE blocks.
- Update 'page count' variable.
- Due to ODS formatting not long before bookmark and associated page failed to correspond to each other.

Second Approach

- Restricted to small group of outputs.
- Extract number of pages produced from individual PS file
- Produce summary report twice
 - As part of main [ods ps (id=main)...]
 - Individually [ods ps (id=temp) ...]
- Search temporary output file
 - filename pages pipe “grep -i %%pages: outfile.PS”;
- Maintain file with Postscript language statements
 - SAS catalog source entry used
- Append to main postscript file

Second Approach

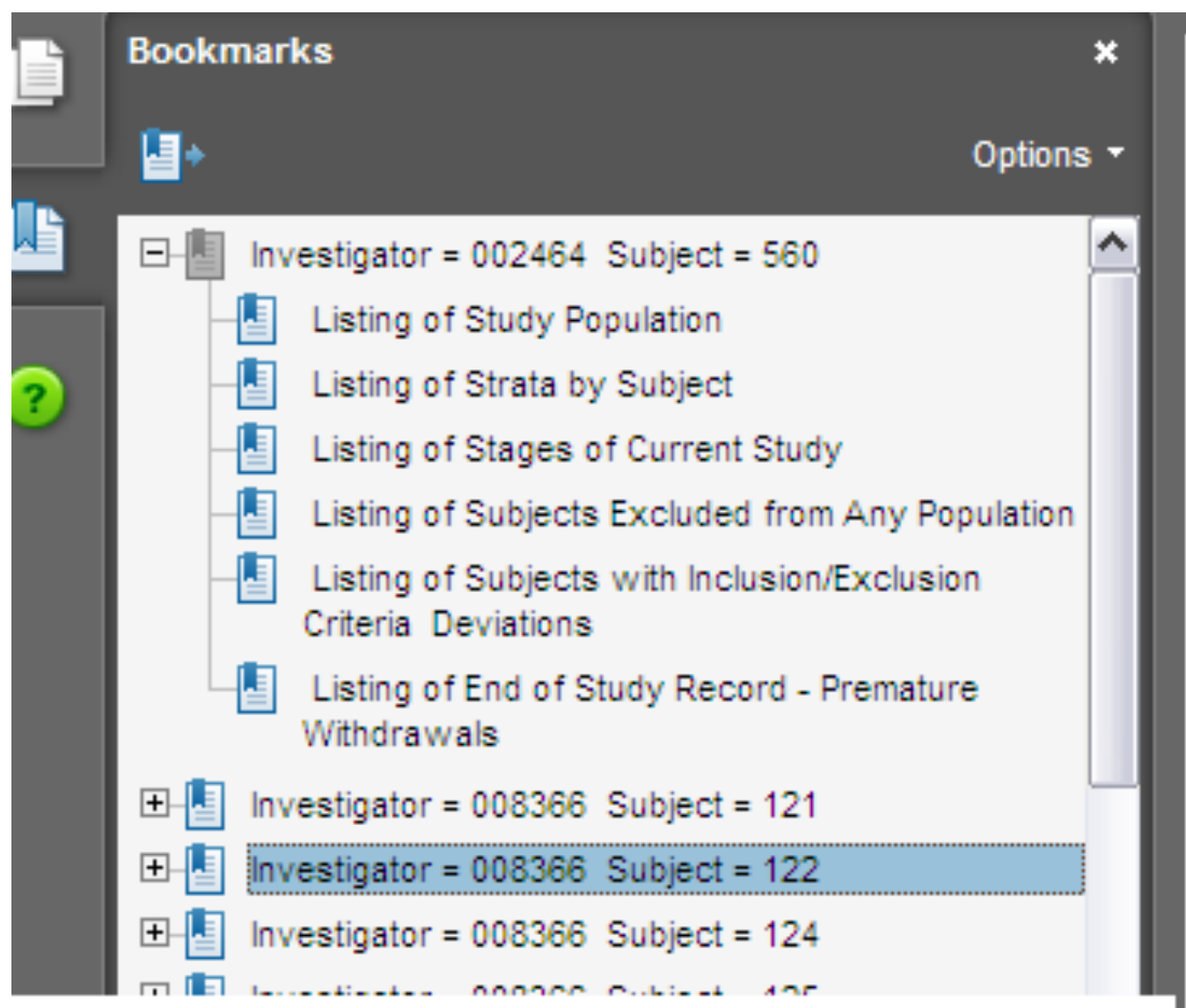


Third (& Final) Approach

- Main requirement – multiple outputs per page.
 - Question: How to determine how many reports can be placed on page.
 - Solution: Get SAS to sort it out.
- Produce entire Postscript file.
 - Contains page, subject & title information
 - `%let txt_string = egrep -l 'page:|subject:|##' outfile.ps;`
 - Filename pages pipe "&txt";
- Manipulate to create 'table of contents'.

Third (& Final) Approach

- Make sure all titles correct and not truncated.
- Determine number of 'child' bookmarks
 - NOTE: Having negative count value will collapse all bookmarks on file opening.
- Create output txt file with bookmark statements
- Append to Postscript file.
- Convert and perform housekeeping.



Final Thoughts

- Limitations

- Time
- Potential memory issues

- Conclusions

- Possible to create two-level bookmarks using SAS.
- Alternative methods available to manipulate PDF file directly
- Could become easier within SAS once PROC DOCUMENT supports REPORT.

References

- Chung, Chang Y. and Dunn, Toby (2005), “Page X of Y with Proc Report.” Paper CC31. *Proceedings of the Pharmaceutical Industry SAS Users Group Conference 2005*
<http://www.lexjansen.com/pharmasug/2005/coderscorner/cc31.pdf>
- Jansen, Lex. (2001), “Creating PDF Documents including Links, Bookmarks and a Table of Contents with the SAS® Software.” Paper FDA05. *Proceedings of the Pharmaceutical Industry SAS Users Group Conference 2001*.
 - http://www.lexjansen.com/pharmasug/2001/proceed/fdacomp/fda05_jansen.pdf
- Karunasundera, Tikiri. (2006), “Creating and Modifying PDF Bookmarks” *SAS Conference Proceedings: Western Users of SAS Software 2006*
 - http://www.lexjansen.com/wuss/2006/data_presentation_and_business_intelligence/DPR-Karunasundera.pdf

Thank you.

Any questions?

stephen.j.griffiths@gsk.com