Implementation of an Actuarial Reserves System using SAS

Ignacio Viggers, database consultant

iviggers@yahoo.com

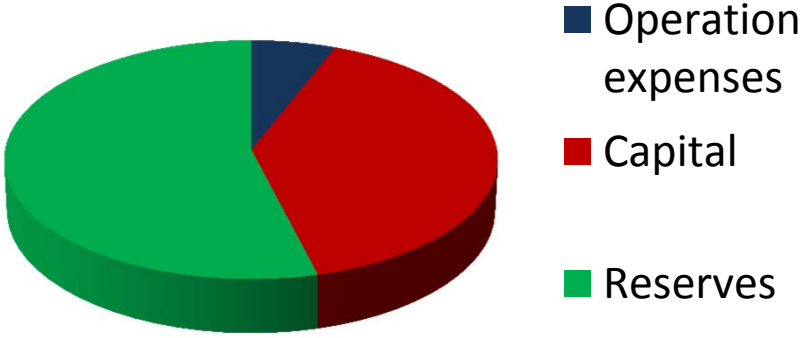
Presentation at Michigan SAS User Group, Feb. 2013

Contents

- Background: Insurance company
- Outline of implementation and analysis of results
  - Flowchart
  - Intermediate steps, snapshots, analysis
  - Program results
- Remarks on performance, integration
- Conclusions
- References

Background: Insurance company

Allocation of earnings*



Operation expenses	Capital mgmt	Reserves
Payroll	Dividends	Cash
Rent, maintenance, taxes	Capital growth	Liquid assets
Marketing	Illiquid assets	Low-risk investments
Re-insurance	High-risk investments	Hedging instruments

*Not to confuse with the concept of Earned Premiums

Outcomes from adequacy of Reserves, per impact of contingencies

- Profitable
  Nearly optimal investment strategy and projection of claims.
- Solvent
  Acceptable projection of claims.
- Risky
  Uncertain adequacy. Approach & policies are questionable.
- Regulatory scrutiny
  Exhaustive review and restructuring with auditors.
- Insolvent
  Need for financial rescue.
- Bankrupt

Outline of implementation and analysis of results

- SAS “grouping” features
- Summarized flowchart
- Intermediate steps
- Tail-factor scenarios
- Snapshots and program results

Claim details*, record layout for grouping

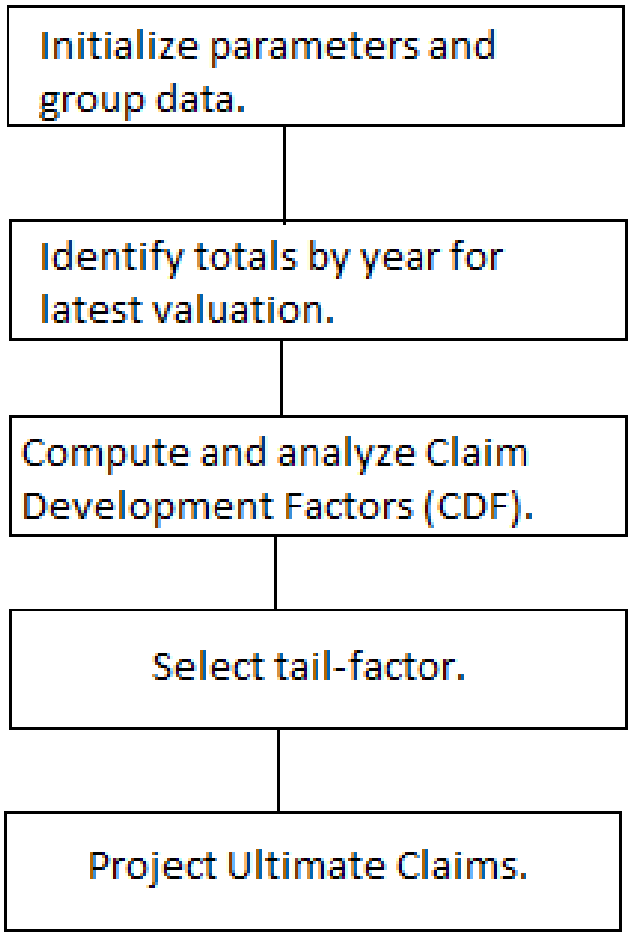
Table 5 – Detailed Example – Claims Transaction Data

			2005 Transactions		2006 Transactions		2007 Transactions		2008 Transactions	
Claim ID	Accident Date	Report Date	Total Payments	Ending Case O/S	Total Payments	Ending Case O/S	Total Payments	Ending Case O/S	Total Payments	Ending Case O/S
1	Jan-5-05	Feb-1-05	400	200	220	0	0	0	0	0
2	May-4-05	May-15-05	200	300	200	0	0	0	0	0
3	Aug-20-05	Dec-15-05	0	400	200	200	300	0	0	0
4	Oct-28-05	May-15-06			0	1,000	0	1,200	300	1,200
5	Mar-3-06	Jul-1-06			260	190	190	0	0	0
6	Sep-18-06	Oct-2-06			200	500	0	500	230	270
7	Dec-1-06	Feb-15-07					270	420	0	650
8	Mar-1-07	Apr-1-07					200	200	200	0
9	Jun-15-07	Sep-9-07					460	390	0	390
10	Sep-30-07	Oct-20-07					0	400	400	400
11	Dec-12-07	Mar-10-08							60	530
12	Apr-12-08	Jun-18-08							400	200
13	May-28-08	Jul-23-08							300	300
14	Nov-12-08	Dec-5-08							0	540
15	Oct-15-08	Feb-2-09								

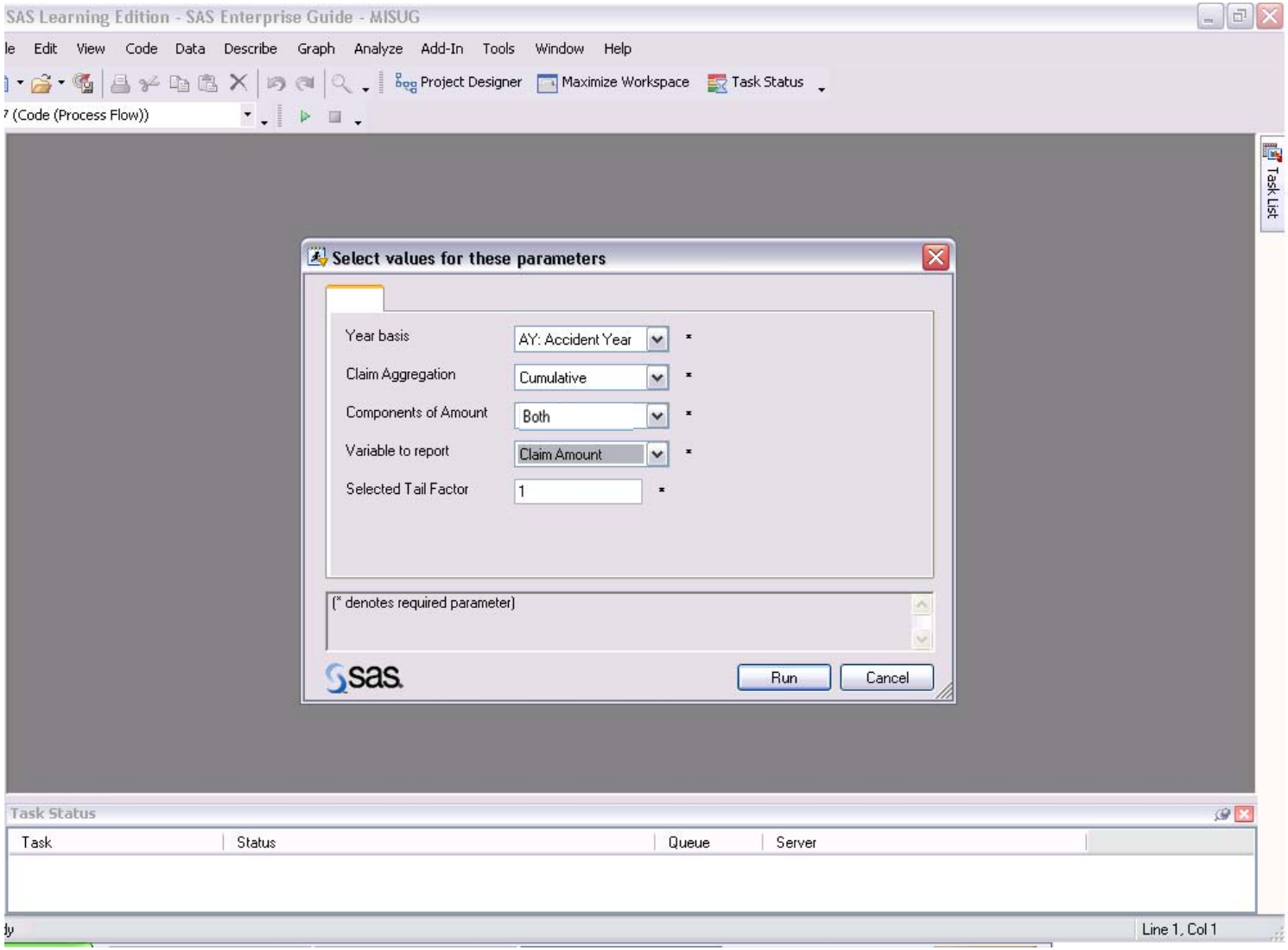
```
data claims;
input cl_id a_date mmddyy10. +1 r_date mmddyy10. payment case_os tx_year;
...
cards;
1 01/05/2005 02/01/2005 400 200  2005
...
run;
```

* Data excerpts copied/adapted from [1]. Used by permission of the Casualty Actuarial Society. Reprinted from "Estimating Unpaid Claims Using Basic Techniques," Jacqueline Frank Friedland, 2009.

Chain Ladder implementation: Summarized flow-chart



Running the application …



Data organized in forms useful to the actuary …

Development Triangle

	age			
	1	2	3	4
a_date	total	total	total	total
2005	1500	2420	2720	3020
2006	1150	1840	2070	.
2007	1650	2640	.	.
2008	1740	.	.	.

Valuation per Period

valuation_period	total_or_count
2005	1500
2006	3570
2007	6210
2008	9470

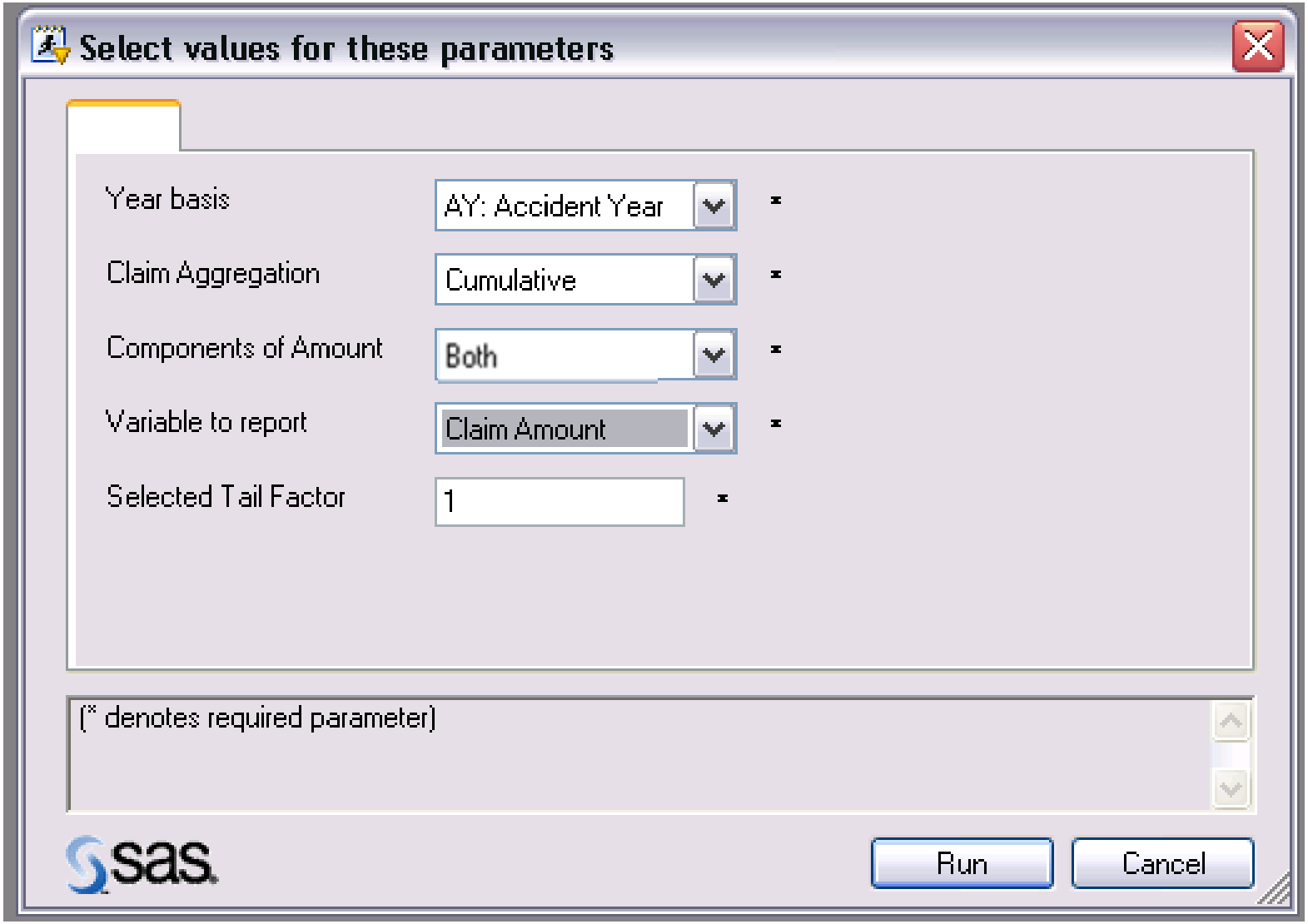
Age-to-age-factor Triangle

	age		
	2	3	4
a_date	age_to_age_factor	age_to_age_factor	age_to_age_factor
2005	1.6133	1.1240	1.1103
2006	1.6000	1.1250	.
2007	1.6000	.	.

Selection of tail-factor

- Tail factor: short- or long-tail.
- Assess the arguments for each assumption:
  - Knowledge of business.
  - Amount of data available.
  - Progression of Claim Development Factors.

Short-tail scenario



Claim Development Factors and Claim Estimates

Average and Cumulative CDF

age	avg_cdf	cml_cdf
1	1.6044	2.0031
2	1.1245	1.2485
3	1.1103	1.1103
4	1.0000	1.0000

Projection of Ultimate Claims

a_date	proj_ult_claims
2005	3020
2006	2298.3088
2007	3296.0591
2008	3485.4726

Long-tail assumption: one possible strategy for derivation of tail-factor

- Compute the ratio of similar, consecutive CDFs:

  $CDF_4 / CDF_3 = 1.1245 / 1.1103 = 1.01279$

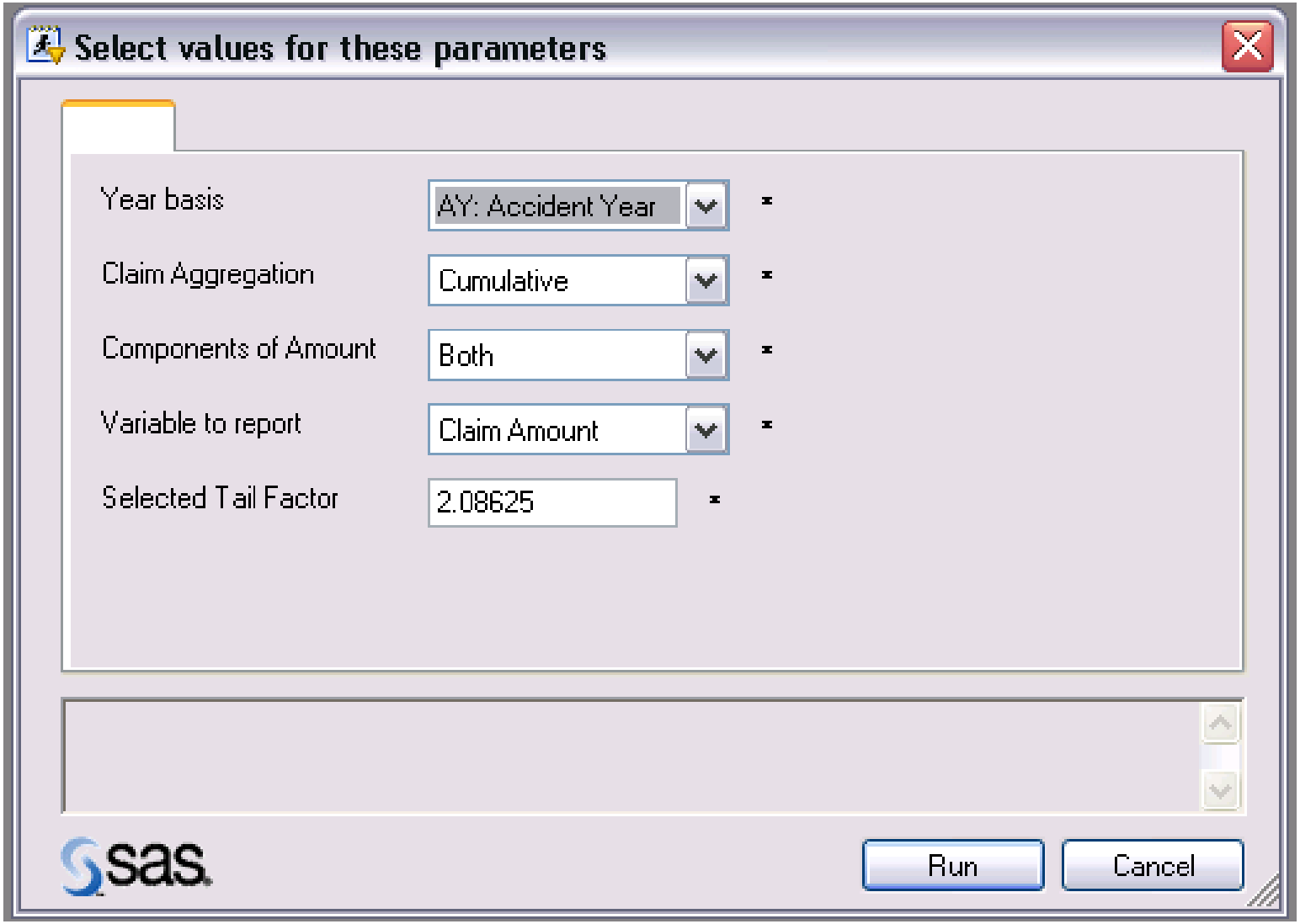
- Assuming constant progression of subsequent CDFs, find “n” s.t.

  $1.1103 / (1.01279)^n \approx 1$ (n = 8)

- Project and multiply “subsequent” CDFs:

  $tail\_factor = (1.1103 / 1.01279)^n = 2.0865$

Long-tail scenario



Claim Development Factors and Claim Estimates

Average and Cumulative CDF

age	avg_cdf	cml_cdf
1	1.6044	4.1791
2	1.1245	2.6047
3	1.1103	2.3164
4	2.0863	2.0863

Projection of Ultimate Claims

a_date	proj_ult_claims
2005	6300.475
2006	4794.8468
2007	6876.4034
2008	7271.5672

Remarks on performance and integration

- "*pure/native Base SAS […] much better for large data volumes, whereas PROC SQL is usually better for small data volumes*" [2].
- Advantage on big data, if compared to open-source R [3].
- Export aggregates, mid-results for use with external packages (e.g. ChainLadder, R language).

Conclusions

- Base/macro SAS language “self-contained” for actuarial stand-alone projections.
- By default, convenient grouping at multiple levels.
- Straight-forward adjustments of data granularity.
- Useful for computing Reserve Estimates, Loss Triangles, other aggregates and input to external functionality.

References

[1] *Friedland, Jacqueline*. Estimating Unpaid Claims Using Basic Techniques. Casualty Actuarial Society, 2010.

[2] *Holland, Philip R*. Efficient SAS Coding. Retrieved from http://www.hollandnumerics.co.uk/pdf/Efficient_SAS_Coding(paper)3.pdf

[3] *Yun, Steven; Rickert, Joseph*. Start Small Before Going Big. Retrieved from http://cdn.oreillystatic.com/en/assets/1/event/85/Start Small Before Going Big Presentation.ppt