

Konwersja własnych kodów

Zbigniew Wyszomierski
Data Integration Product Manager

Agenda

- Przygotowanie kodów do konwersji
- Programy standardowe
 - SQL
 - SAS 4GL
- Makroprogramy
- Przetwarzanie równoległe

Przygotowanie kodów do konwersji

- Procedura SCAPROC (SAS Code Analyzer)
 - Komentarze standardowe
 - Przetwarzanie równoległe
- SAS Data Integration Studio
 - Kod bez komentarzy (automatyczne użycie procedury SCAPROC)
 - Kod wyjściowy z procedury SCAPROC

Przygotowanie kodów do konwersji

- Rezultat działania SAS Code Analyzer
 - Zestaw słów kluczowych
 - Wejście \ wyjście
 - Tryb przetwarzania
 - Przetwarzanie równoległe

slajd_01.sas *

```
data tmp.agd_2005;  
  set demo.agd_complex;  
  where rok=2005;  
  Koszt=Koszt_zm+Koszt_st;  
run;
```

```
proc means data=tmp.agd_2005;  
  var Koszt;  
  class Kod;  
  output out=Koszt_Prod;  
run;
```

slajd_01_sca.sas

```
/* JOBSPLIT: DATASET INPUT SEQ DEMO.AGD_COMPLEX.DATA */  
/* JOBSPLIT: LIBNAME DEMO "D:\Dane\AGD" */  
/* JOBSPLIT: DATASET OUTPUT SEQ TMP.AGD_2005.DATA */  
/* JOBSPLIT: LIBNAME TMP "D:\Dane\TMP_BASE" */  
/* JOBSPLIT: SYMBOL GET SYS_IOUSEEE */  
/* JOBSPLIT: ELAPSED 10905 */  
/* JOBSPLIT: PROCNAME DATASTEP */  
/* JOBSPLIT: STEP SOURCE FOLLOWS */
```

```
data tmp.agd_2005;  
  set demo.agd_complex;  
  where rok=2005;  
  Koszt=Koszt_zm+Koszt_st;  
run;
```

```
/* JOBSPLIT: DATASET INPUT SEQ TMP.AGD_2005.DATA */  
/* JOBSPLIT: LIBNAME TMP "D:\Dane\TMP_BASE" */  
/* JOBSPLIT: DATASET OUTPUT SEQ WORK.KOSZT_PRODUKT_AGREGAT.DATA */  
/* JOBSPLIT: LIBNAME WORK V9 'C:\DOCUME~1\sp1zbn\LOCALS~1\Temp\SAS Temporary Files\_TD7320' */  
/* JOBSPLIT: ITEMSTORE INPUT SASUSER.TEMPLAT */  
/* JOBSPLIT: ITEMSTORE INPUT SASHELP.TMPLMST */  
/* JOBSPLIT: FILE OUTPUT D:\sasdocs\SASForum\2009\ImportCode\slajd_01.lst */  
/* JOBSPLIT: FILE OUTPUT D:\sasdocs\SASForum\2009\ImportCode\slajd_01_sca.sas */  
/* JOBSPLIT: SYMBOL GET SYS_IOUSEEE */  
/* JOBSPLIT: SYMBOL GET SYSSUMTRACE */  
/* JOBSPLIT: ELAPSED 421 */  
/* JOBSPLIT: PROCNAME MEANS */  
/* JOBSPLIT: STEP SOURCE FOLLOWS */
```

```
proc means data=tmp.agd_2005 NWAY;  
  var Koszt;  
  class Kod;  
  output out=Koszt_Produkt_Agregat sum=Suma mean=Srednia;  
run;
```

```
/* JOBSPLIT: END */
```

Konwersja kodów w SAS DI Studio

SAS Data Integration Studio 4.21 - User (splzbw)

File Edit View Check Outs Actions Debug Tools Window Help

Transformations
Folders Inventory

My Folder
Products
Shared Data
Demo Environment
Jobs
Import
SAS Co
SAS Co
SAS Co
Libraries
Tables
SASApp - OLAF
SASApp - SAS
System
Users

Properties
Collapse
New
Check Out
History
Copy Ctrl+C
Paste Ctrl+V
Paste Special...
Delete
Rename
Copy to Folder...
Move to Folder...
Register Tables...
Import
Export
Refresh F5

SAS Package...
Import SAS Code
Metadata...

Import SAS Code

Select the SAS program files to import:

Add
Remove

Import options

Convert and import files
 Convert files only
 Import already converted files

Location: /Shared Data/Demo Environment/Jobs/Import Browse

OK Cancel Help

Basic Properties

Name	Value
Name	Import
Description	
Folder Location	/Shared Data/Demo Envi...
Checked Out By	

Import (Folder)

SASApp
sasdemo@splzbw as SAS Demo User
splzbw : 8561

Programy standardowe

- Kod w SAS 4GL nie zawierający makr

Programy standardowe

- Kod w SQL nie zawierający makr

Makroprogramy

- Mechanizm pozwalający na parametryzację istniejących programów

```
Editor - Untitled2 *  
proc means data=sashelp.prdsale SUM MEAN MIN;  
  var actual;  
  class country;  
run;
```

```
Editor - Untitled2 *  
1 %LET input=sashelp.prdsale;  
2 %LET stat=SUM MEAN MIN;  
3 proc means data=&input &stat;  
4   var actual;  
5   class country;  
6 run;
```

Makroprogramy

- Automatyczna generacja części lub całości programów

```
Editor - Untitled2 *
1 data AGORA AMICA APATOR ... WOLCZANKA ZREW ZYWIEC;
2 set notowania;
3 select (spoln)
4 when("AGORA")
5 output AGO
6 when("AMICA")
7 output AMI
8 when("APATC")
9 output API
10 ...
11 when("ZYWIEC")
12 output ZY
13 otherwise
14 put "Niepoprawna nazwa spólki";
15 end;
16 run;
```

```
Editor - Untitled2 *
1 data %do i=1 %to &ilosc;
2 &&nazwa&i
3 %end;;
4 set &zbiór;
5 select (&zmienna);
6 %do i=1 %to &ilosc;
7 when("&&nazwa&i") output &&nazwa&i;
8 %end;
9 otherwise put "Niepoprawna nazwa spólki";
10 end;
11 run;
```

Makroprogramy

- Definicja makra i wywołanie w jednym programie
 - Dla wybranego zbioru i podanej kolumny należy sprawdzić ile jest unikalnych wartości w kolumnie
 - Tablica kontrolna zawierająca listę unikalnych wartości (wyjście)
 - Na podstawie tablicy kontrolnej z poprzedniego kroku należy wygenerować odpowiedni tekst programu
 - Tablica kontrolna zawierająca listę unikalnych wartości (wejście)
 - Lista tablic, których liczba jest równa liczbie unikalnych wartości (wyjście)

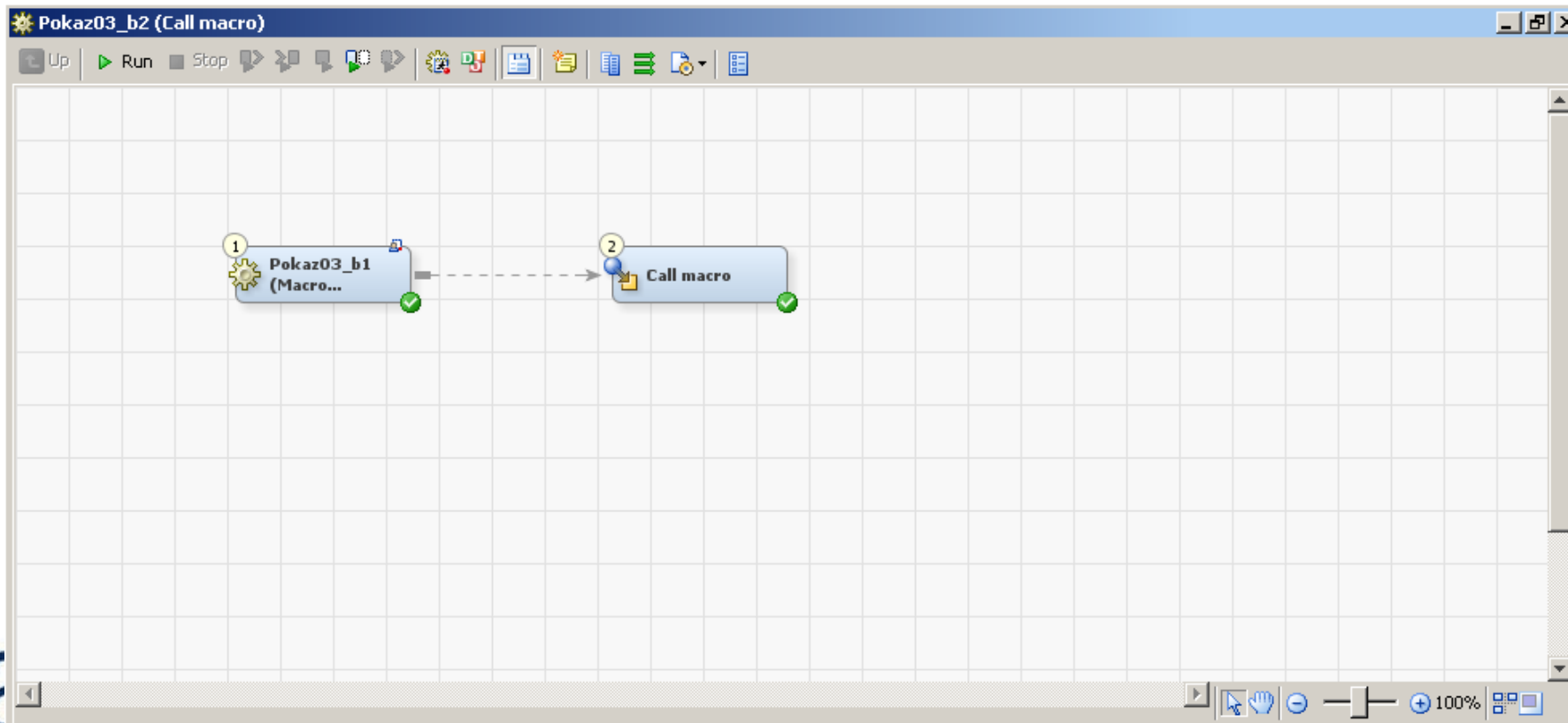
Makroprogramy

- Definicja makra i wywołanie w jednym programie

```
Editor - Untitled2 *
1 %macro podzial(zbior,zmienna);
2 proc sql noprint;
3     create table nazwy as
4     select distinct &zmienna
5     from &zbior;
6 quit;
7 %let ilosc=&sqllobs;
8 proc sql noprint;
9     select &zmienna into :nazwa1-&nazwa&ilosc
10    from nazwy;
11 quit;
12 data %do i=1 %to &ilosc;
13     &&nazwa&i
14     %end;;
15 set &zbior;
16 select (&zmienna);
17 %do i=1 %to &ilosc;
18     when("&&nazwa&i") output &&nazwa&i;
19 %end;
20     otherwise put "Zla nazwa :" &zmienna;
21 end;
22 run;
23 %mend podzial;
24 %podzial(sashelp.prdsale,product);
```

Makroprogramy

- Zadanie #1: definicje makr
- Zadanie #2: wykorzystanie makr



Przetwarzanie równoległe

- Uruchomienie SAS Code Analyzer z odpowiednią opcją
- Analiza tekstu programu pod kątem możliwości uruchomienia N równoległych sesji
- Generacja kodów startujących dodatkowe sesje

DZIĘKUJĘ ZA UWAGĘ

SAS® forum
POLSKA 2009 **warszawa**