

# **DS2 – EIN DATA STEP WEITER**

**MIHAI PAUNESCU**



## **DS2** WAS IST DS2?

- SAS Programmiersprache zur Datenaufbereitung
- Syntax ähnlich zu Data Step
- Kreuzung aus Data Step und ANSI SQL:1999
- DS2 ist ab SAS 9.4 verfügbar als Teil von SAS Base

## **DS2** WARUM IST DER DS2 EIN SCHRITT WEITER?

- Moderne Programmiertechniken
- Mehr Flexibilität
- Höhere Performance

- Neue Datentypen
- Variablen Scoping
- Methods
- Packages

```
data _null_;  
  if _n_ = 1 then do;  
    foo = '**> Starting';  
    put foo;  
  end;  
  
  set ds2_sas.banks  
  end=last;  
  put _all_;  
  
  if last then do;  
    bar = '**> I quit!';  
    put bar;  
  end;  
  
run;
```

```
PROC DS2;  
data _null_;  
  method init();  
    dcl varchar(20) foo;  
    foo = '**> Starting';  
    put foo;  
  end;  
  method run();  
    set ds2_sas.banks;  
    put _all_;  
  end;  
  method term();  
    dcl char(11) bar;  
    bar = '**> I quit!';  
    put bar;  
  end;  
run;  
QUIT;
```

# PROGRAMMIER- TECHNIKEN PACKAGES

- Ein Package ist ein Programm-Modul zum gruppieren von zusammengehörigen Variablen und Methoden.
- User-defined and predefined Packages

```
proc ds2;  
package work.MyPck / overwrite=yes;  
  method add(char(50) first, char(50) second)  
    returns char(101);  
    return catx('+', first, second);  
  end;  
endpackage;  
run; quit;
```

```
proc ds2;  
data;  
  dcl package work.MyPck calc();  
  dcl char(2) W1 W2;  
  dcl char(5) Phrase;  
  method init();  
    W1='20';  
    W2='30';  
    Phrase=calc.add(W1, W2);  
  output;  
  end;  
enddata;  
run; quit;
```

W1	W2	Phrase
20	30	20+30

## DS2 MEHR FLEXIBILITÄT

- Verwendung von FedSQL im Set Statement.

```
proc ds2;  
data sales (overwrite=YES);  
  method run();  
    set {select c.Customer_ID  
            ,Total_Retail_Price  
          from ds2_sas.order_fact f  
            full join  
              ds2_sas.customer_dim c  
            on f.Customer_ID=c.Customer_ID  
          order by 1};  
    ...  
  end;  
enddata; run; quit;
```

- Integration von neuen Funktionen durch Predefined Packages z.B. für Matrizen-Operationen, R-Code, FCMP Funktionen, Hash Operationen.

## DS2 HÖHERE PERFORMANCE

- Parallelisierung durch mehrere Threads.
- Ausführen des Codes in der Datenbank mit SAS In-Database Code Accelerator.

Var1	1x Thread	Thread 1	Thread 2	Thread 3
1	1	1		
2	3	3		
3	6	6		
4	10	10		
5	15		5	
6	21		11	
7	28		18	
8	36		26	
9	45			9
10	55			19
Summe=55		Summe=10+26+19=55		



# HÖHERE PERFORMANCE

## THREADING IN SAS BASE

```
proc ds2;  
  thread sumer / overwrite = yes  
  encrypt=SAS;  
  drop var1;  
  dcl double sum;  
  method run();  
    set input_dataset;  
    sum + Var1;  
  end;  
  
  method term();  
    output;  
  end;  
endthread;  
run; quit;
```

NOTE: There were 100000000 observations read from the data set WORK.INPUT\_DATASET.

NOTE: DATA statement used (Total process time):

real time	9.85 seconds
cpu time	9.84 seconds

```
proc ds2;  
  data ;  
  dcl double total having format 16.0;  
  dcl double sum having format 16.0;  
  dcl thread sumer s_instance;  
  method run();  
    set from s_instance threads=4;  
    total + sum;  
  end;  
enddata;  
run; quit;
```

NOTE: PROCEDURE DS2 used (Total process time):

real time	1.71 seconds
cpu time	6.41 seconds

# HÖHERE PERFORMANCE THREADING IN DER DATENBANK MIT SAS IN-DATABASE CODE ACCELERATOR

Datastep



DS2

Threading



DS2 Threading

mit In-Database

SAS Code

Accelerator



- **What's New in SAS 9.4 DS2 Language Reference**

<http://support.sas.com/documentation/cdl/en/ds2ref/66009/HTML/default/viewer.htm#ds2refwhatsnew94.htm>