

Parallel loading into Netezza using SAS

Fulton Lee

Agenda

- What's the problem?
- Parallelism
- Show me the Code
- Questions

What's the problem?

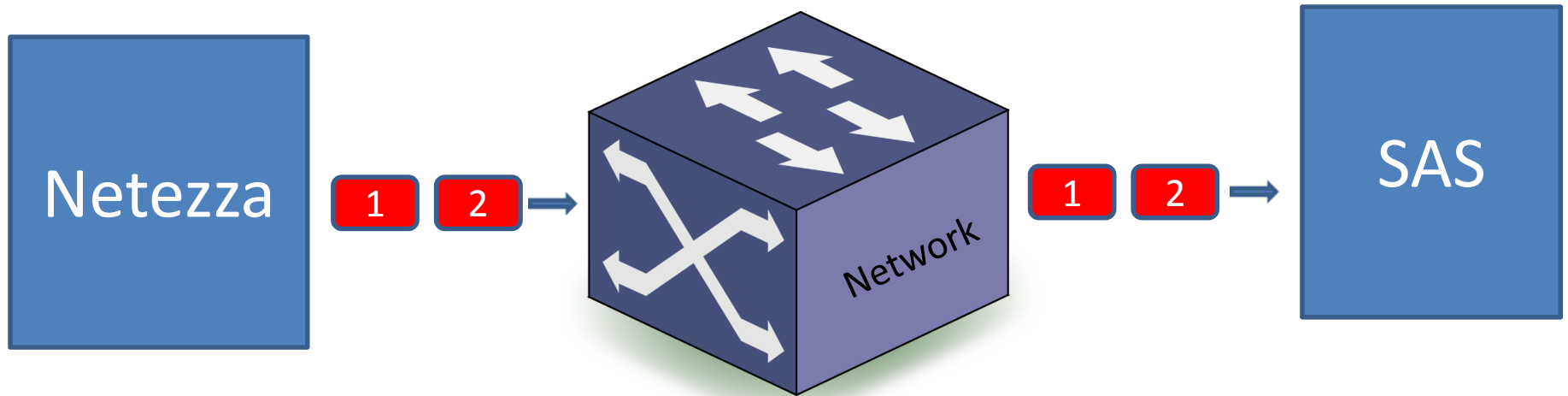
Problem Statement:

We have this fast analytic appliance (Netezza) which can process tons of data quickly. Why is my SAS still so slow transferring the data back?

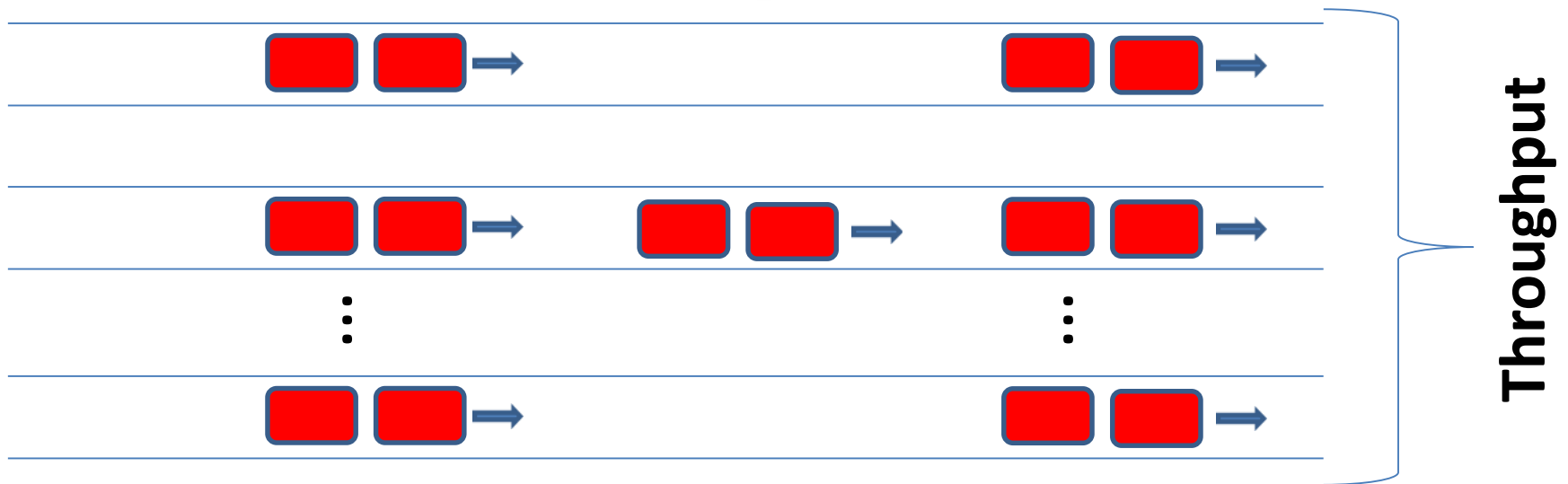
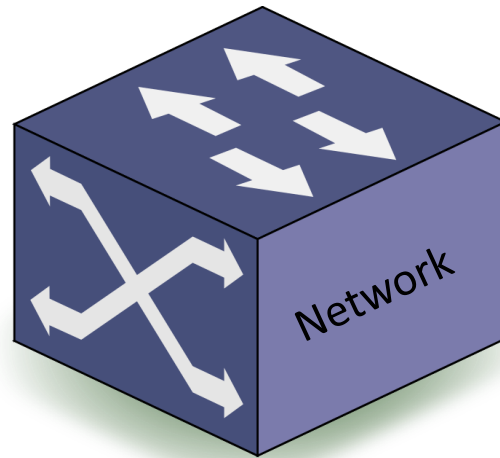
Assumptions:

- Data is distributed for efficiency on Netezza
- A groom has been run on the table being queried and statistics have been run on the tables
- The query is optimized on the database

What's the problem ... continued?



What's the problem ... continued?



What's the problem ... continued?

Technical fact data needs to travel between analytic appliance and SAS. This is done through the network (LAN) and throughput through the LAN at it's maximum can be as follows:

LAN (wired):

100 mega bits/sec network	- < 12.5 MegaBytes/Sec
1000 mega bits/sec network	- < 125 MegaBytes/Sec

WIRELESS:

300 mega bits/sec network	- < 37.5 MegaBytes/Sec
900 mega bits/sec network	- < 112.5 MegaBytes/Sec

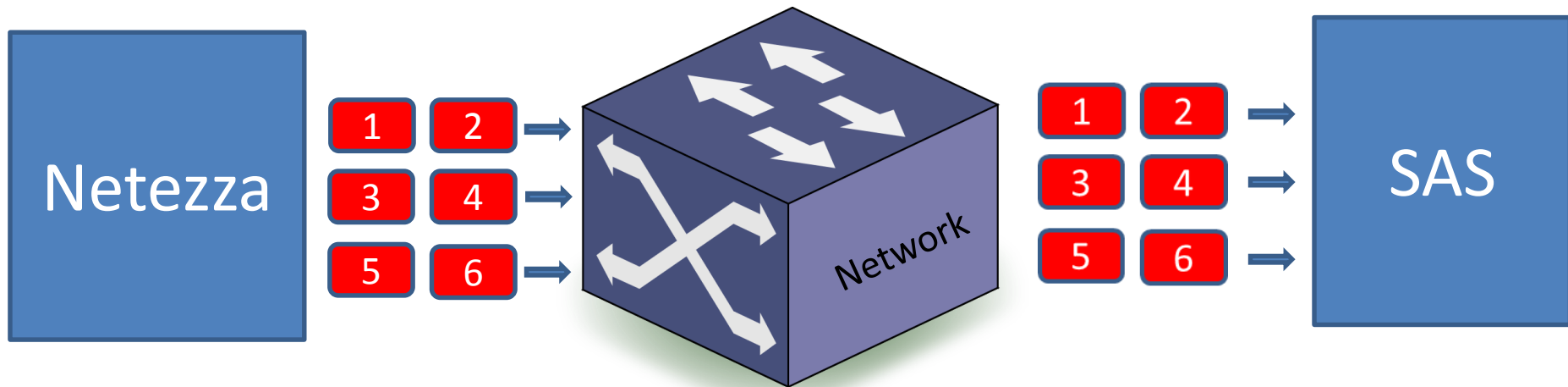
What's the problem ... continued?

- How do we get more throughput between SAS and Netezza?

Parallelism

- To break up a larger process running sequentially into smaller chunks and run them at the same time.

Parallelism ... continued?



Parallelism ... continued?

- Determine the data source to be retrieved.
For our example we are pulling a FACT table which has 540 columns and 3 billion records
- Specifically we want to retrieve a subset of 33 million records
- To create parallel streams we need to determine a way to break-out the 33 million records into equal size portions.

Netezza Query

```
15 proc sql;  
16     create table sample as  
17     select * from rd_rpt.fact where project=200210;
```

NOTE: Compressing data set WORK.PK_200210 decreased size by 76.15 percent.

Compressed is 2651304 pages; un-compressed would require 11114638 pages.

NOTE: Table WORK.PK_200210 created, with **33,343,905** rows and **540** columns.

```
18 quit;
```

NOTE: PROCEDURE SQL used (Total process time):

real time	1:02:37.17
user cpu time	56:51.74
system cpu time	5:41.02
memory	2442.28k
OS Memory	17188.00k
Timestamp	03/01/2016 02:40:12 AM

Netezza Query with bulkunload option

```
15 proc sql;  
16     create table sample(compress=no) as  
17         select *  
18             from rd_rpt.fact (bulkunload=YES bl_use_pipe=NO bl_datafile="/saswork/fact.dat")  
19         where project eq 200210;
```

NOTE: Table WORK.INST_DETAILS created, with **33,343,905** rows and **540** columns.

```
20 quit;
```

NOTE: PROCEDURE SQL used (Total process time):

real time	29:05.27
user cpu time	15:52.43
system cpu time	7:35.94
memory	2533.15k
OS Memory	17700.00k
Timestamp	03/01/2016 11:32:48 AM

Show me the Code

```
options mprint symbolgen fullstimer;
```

```
/* Setup parallel sessions to the number of spawn sessions */
```

```
%macro
```

```
parallel_unload(slice=time_period,out_lib='/saswork',out_tbl=factspxn,tmp_append='/saswork/_temp_');
```

```
libname RD_RPT netezza SERVER="cmcblpntzasrv01.bmocm.com" USER=nz_sasetl  
PASSWORD="xxxxxxx" Database="xxxxxxx" readbuff=15000;  
libname out_lib &out_lib;  
libname tmp &tmp_append;
```

```
/* Number of Spawned Session related to number of Slices */
```

```
proc sql;  
  select count(distinct (time_period)) into: num_spawns from rd_rpt.fact where project=200210;  
quit;  
%put &num_spawns;
```

```
/* Loading all slice criteria */
```

```
proc sql;  
  select distinct (time_period) into :slice1-:slice20 from rd_rpt.fact where project=200210 order by  
time_period;  
quit;
```

```
/* Setting up Session ID */
```

```
%let base_session_id = %sysfunc(floor(%sysvalf(%sysfunc(ranuni(0))*%eval(99999-  
&num_spawns))));  
%put &base_session_id;
```

```
/* loop to create spawn queries */
```

```
%do i = 1 %to &num_spawns;  
  %let split_num=&i;  
  %put &&slice&i;  
  /* Create unique session ID */  
  %let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));  
  %put &session_id;
```

```
/* Create sign-on to sessions */
```

```
signon &session_id sascmd="/opt/sas/SASFoundation/9.3/sas -noterminal -nonews -stimer";
```

```
/* Variables from Parent to Children */
```

```
%syslput i=&i / remote=&session_id;  
%syslput slice=&i-&&slice&i / remote=&session_id;  
%syslput slice=&slice / remote=&session_id;  
%syslput out_tbl=&out_tbl / remote=&session_id;
```

```
filename mylog "/saswork/_temp_/rlog&i..log" mod;
```

```
rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);
```

```
options mprint symbolgen;  
proc printto log="/saswork/_temp_/rlog&i..log" NEW; run;
```

```
libname RD_RPT netezza SERVER="xxxxxx" USER=xxxxxx PASSWORD="{SAS002}xxxxxxx"  
Database="xxxx" readbuff=1500;
```

```
proc sql; /*inobs=100000*/  
  create table tmp.&out_tbl&i (compress=no) as  
  select * from rd_rpt.fact  
  where project=200210 and &slice = "&&slice&i";  
quit;
```

```
endrsubmit;  
%end;
```

```
%do i = 2 %to &num_spawns %by 2;  
  %let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));  
  %put &session_id;  
  %syslput i=&i / remote=&session_id;  
  %let n=%eval(&i-1);  
  %syslput n=&n / remote=&session_id;
```

```
waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));  
waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&n),z5.));  
rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);
```

Show me the Code

```
options mprint symbolgen;
proc printto log="/saswork/_temp_/alog2&i..log" NEW; run;
proc append base=tmp.&out_tbl&i data=tmp.&out_tbl&n force; run;
endrsubmit;
%end;

%do i = 4 %to &num_spawns %by 4;
%let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));
%put &session_id;
%syslput i=&i / remote=&session_id;
%let n=%eval(&i-2);
%syslput n=&n / remote=&session_id;

waitfor_all_%substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));
waitfor_all_%substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&n),z5.));
rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);
options mprint symbolgen;
proc printto log="/saswork/_temp_/alog4&i..log" NEW; run;
proc append base=tmp.&out_tbl&i data=tmp.&out_tbl&n force; run;
endrsubmit;
%end;

%do i = 8 %to &num_spawns %by 8;
%let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));
%put &session_id;
%syslput i=&i / remote=&session_id;
%let n=%eval(&i-4);
%syslput n=&n / remote=&session_id;

waitfor_all_%substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));
waitfor_all_%substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&n),z5.));

rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);
options mprint symbolgen;
proc printto log="/saswork/_temp_/alog8&i..log" NEW; run;
proc append base=tmp.&out_tbl&i data=tmp.&out_tbl&n force; run;
endrsubmit;
%end;
```

```
%let s8=8;
%let s12=12;

waitfor_all_%substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&s8),z5.));
waitfor_all_%substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&s12),z5.));

proc append base=tmp.&out_tbl&s12 data=tmp.&out_tbl&s8 force; run;

/* Wait for all spawned sessions to complete */
waitfor_all_
%do i=1 %to &num_spawns;
%substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.))
%end;;

%do a = 1 %to &num_spawns;
SIGNOFF %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&a),z5.));
%end;
%mend parallel_unload;

%parallel_unload;

proc sql;
select count(*) from tmp.factspwn12;
quit;

proc sql;
select time_period, count(time_period) from tmp.factspwn12 group by time_period;
quit;
```

Show me the Code ... continued

```
options mprint symbolgen fullstimer;
```

```
/* Setup parallel sessions to the number of spawn sessions */
```

```
%macro parallel_unload (slice=time_period, out_lib='/saswork', out_tbl=factsawn,  
tmp_append='/saswork/_temp_');
```

```
libname RD_RPT netezza SERVER="xxxxxxx" USER=xxxxxxx PASSWORD="{SAS002}xxxxx"  
Database="xxxxxxxx" readbuff=15000;
```

```
libname out_lib &out_lib;
```

```
libname tmp &tmp_append;
```

```
/* Number of Spawned Session related to number of Slices */
```

```
proc sql;
```

```
select count(distinct (time_period)) into: num_spawns from rd_rpt.fact  
where project=200210;
```

```
quit;
```

```
%put &num_spawns;
```

Show me the Code ... continued

```
/* Loading all slice criteria */
proc sql;
  select distinct (time_period) into :slice1-:slice20 from rd_rpt.fact where project=200210 order by
time_period;
quit;

/* Setting up Session ID */
%let base_session_id = %sysfunc(floor(%sysevalf(%sysfunc(ranuni(0))*%eval(99999-
&num_spawns)))));
%put &base_session_id;

/* loop to create spawn queries */
%do i = 1 %to &num_spawns;
  %let split_num=&i;
  %put &&slice&i;

  /* Create unique session ID */
  %let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));
  %put &session_id;

  /* Create sign-on to sessions */
  signon &session_id sascmd="/opt/sas/SASFoundation/9.3/sas -noterminal -nonews -stimer";
```


Slicing of large dataset

Slice Number	Row COUNT
Slice12	2,899,470
Slice11	2,899,470
Slice10	2,899,470
Slice9	2,899,470
Slice8	2,899,470
Slice7	2,899,470
Slice6	2,899,470
Slice5	2,899,470
Slice4	2,899,470
Slice3	2,899,470
Slice2	2,899,470
Slice1	1,449,735
Total	33,343,905

Show me the Code ... continued

```
/* Variables from Parent to Children */
%syslput i=&i / remote=&session_id;
%syslput slice&i=&&slice&i / remote=&session_id;
%syslput slice=&slice / remote=&session_id;
%syslput out_tbl=&out_tbl / remote=&session_id;

filename mylog "/saswork/_temp_/rlog&i..log" mod;

rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);

options mprint symbolgen;
proc printto log="/saswork/_temp_/rlog&i..log" NEW; run;

libname RD_RPT netezza SERVER="xxxx" USER=xxxxx
PASSWORD="{SAS002}xxxx" Database="xxxxxx" readbuff=1500;
```

Show me the Code ... continued

```
proc sql; /*inobs=100000*/  
create table tmp.&out_tbl&i (compress=no) as  
  select * from rd_rpt.fact  
  where project=200210 and &slice = "&&slice&i."  
quit;  
  
endrsubmit;  
%end;
```

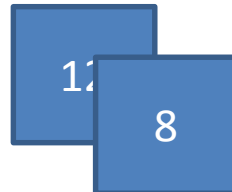
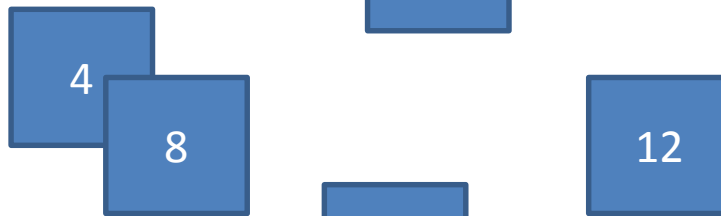
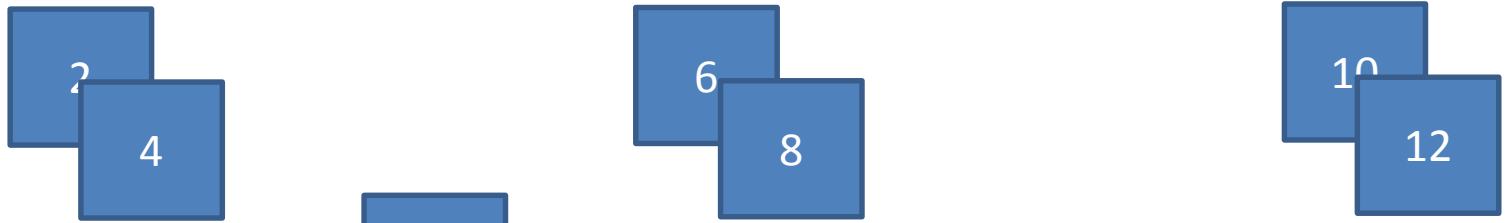
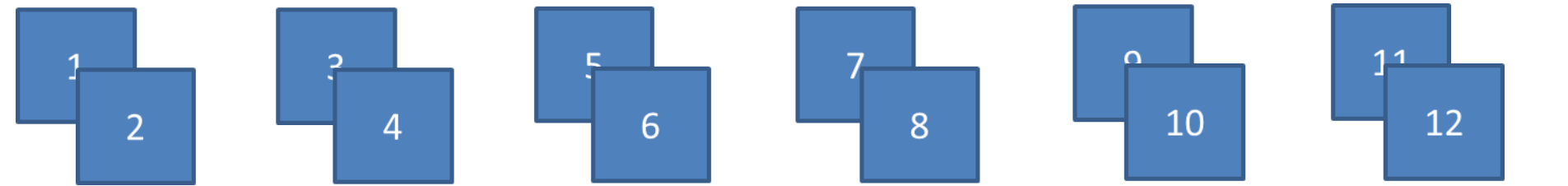
Show me the Code ... continued

```
%do i = 2 %to &num_spawns %by 2;
  %let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));
  %put &session_id;
  %syslput i=&i / remote=&session_id;
  %let n=%eval(&i-1);
  %syslput n=&n / remote=&session_id;

  waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));
  waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&n),z5.));
  rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);

options mprint symbolgen;
proc printto log="/saswork/_temp_/alog2&i..log" NEW; run;
proc append base=tmp.&out_tbl&i data=tmp.&out_tbl&n force; run;
endrsubmit;
%end;
```

Assembling the Pieces



Show me the Code ... continued

```
%do i = 4 %to &num_spawns %by 4;  
  %let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));  
  %put &session_id;  
  %syslput i=&i / remote=&session_id;  
  %let n=%eval(&i-2);  
  %syslput n=&n / remote=&session_id;  
  
  waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));  
  waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&n),z5.));  
  rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);  
  options mprint symbolgen;  
  proc printto log="/saswork/_temp_/alog4&i..log" NEW; run;  
  proc append base=tmp.&out_tbl&i data=tmp.&out_tbl&n force; run;  
  endrsubmit;  
%end;
```

Show me the Code ... continued

```
%do i = 8 %to &num_spawns %by 8;  
  %let session_id = %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));  
  %put &session_id;  
  %syslput i=&i / remote=&session_id;  
  %let n=%eval(&i-4);  
  %syslput n=&n / remote=&session_id;  
  
  waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.));  
  waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&n),z5.));  
  
  rsubmit &session_id wait=no macvar=status_&i inheritlib=(tmp out_lib);  
  options mprint symbolgen;  
  proc printto log="/saswork/_temp_/alog8&i..log" NEW; run;  
  proc append base=tmp.&out_tbl&i data=tmp.&out_tbl&n force; run;  
endrssubmit;  
%end;
```

Show me the Code ... continued

```
%let s8=8;
%let s12=12;

waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&s8),z5.));
waitfor _all_ %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&s12),z5.));

proc append base=tmp.&out_tbl&s12 data=tmp.&out_tbl&s8 force; run;

/* Wait for all spawned sessions to complete */
waitfor _all_
  %do i=1 %to &num_spawns;
  %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&i),z5.))
%end;;

%do a = 1 %to &num_spawns;
  SIGNOFF %substr(&username,1,3)%sysfunc(putn(%eval(&base_session_id+&a),z5.));
%end;
%mend parallel_unload;

%parallel_unload;
```


I'M GIVING HER

**ALL SHE'S GOT
CAPTAIN**

Results - Download

Method	Number of Rows	Number of Columns	Time
Simple Query	33,343,905	540	1:02:37.17
Bulk Unload	33,343,905	540	29:05.27
Parallel Processing	33,343,905 (12 slice)	540	43:06 24:57

Loading from SAS to Netezza

- Determine SAS Dataset and determine how to slice
- Create spawned session based on the number of slices
- Each session append data to the Netezza tables

SAS Code to Netezza

```
%macro example3();
```

```
libname netezza netezza server="xxxx.xxxx.xxxx" database=xxxxx user=xxxxxx  
password="{SAS002}xxxx";
```

```
libname fact '/saswork';
```

```
proc sql ;
```

```
    create table netezza.sample3 as select * from fact.fact;
```

```
quit;
```

```
%mend example3;
```

SAS Code to Netezza

```
%macro example4();
```

```
  libname netezza netezza server="xxxxx.xxxxxx.xxx" database=xxxxxxx user=xxxxxxx  
password="{SAS002}xxxxxxxxx";
```

```
  libname fact '/saswork';
```

```
/* Determine the number of slices */
```

```
%let num_slices=4;
```

```
/* Determine the number of rows */
```

```
proc sql; select count(*) into :num_of_rows from fact.fact; quit;
```

```
%put &num_of_rows;
```

```
/* Number of rows per slice */
```

```
%let nobs = %sysevalf(&num_of_rows/&num_slices,ceil);
```

```
%put &nobs;
```

SAS Code to Netezza

```
/* Start to setup Remote Session */  
%do i = 1 %to &num_slices;  
    %let sessionid = RMT_&i;  
    signon &sessionid sascmd="/opt/sas/SASFoundation/9.3/sas -  
noterminal";  
%end;
```

```
/* Define variables for each remote session */  
%do i = 1 %to &num_slices;  
    %syslput num_slices=&num_slices / remote=RMT_&i;  
    %syslput nobs = &nobs / remote=RMT_&i;  
    %syslput num_of_rows = &num_of_rows / remote=RMT_&i;  
%end;
```

SAS Code to Netezza

```
/* Create Views */
%do i = 1 %to &num_slices;
  %put firstobs=(((&nobs*&i)-&nobs)+1);
    %put obs=%eval(&nobs*&i);
    data fact.slice&i / view=fact.slice&i;
      set fact.fact (firstobs=%eval(((&nobs*&i)-&nobs)+1) obs=%eval(&nobs*&i));
    run;
%end;
```

SAS Code to Netezza

```
%do i = 1 %to &num_slices;
    %syslput i = &i / remote=RMT_&i;

/* load results into table */
rsubmit RMT_&i wait=no macvar=status_&i ;
    options mprint symbolgen fullstimer;
    libname netezza netezza server="xxxxx" database=xxxx user=xxxxxxx password="xxxxx";
    libname fact '/saswork';
    proc printto log="/saswork/log&i..log" NEW; run;

    proc append base=netezza.sample4 (bulkload=yes BL_USE_PIPE=NO
bl_datafile="/saswork/slice&i..dat"
        bl_options="logdir '/saswork' ctrlchars crlnString nullvalue ' ' ")
            data=fact.slice&i force;
endrsbmit;
%end;
```


SAS Code to Netezza

```
waitfor _all_ RMT_1 RMT_2 RMT_3 RMT_4;
```

```
/* Signoff all remote sessions */
```

```
%do i = 1 %to &num_slices;
```

```
    signoff RMT_&i;
```

```
%end;
```

```
%mend example4;
```

```
%example4();
```

Results - Upload

Method	Number of Rows	Number of Columns	Time
Simple Query	14628	420	14.43
Bulkupload	1,477,428	420	21:42.16
Parallel Processing	1,477,428 (4 slice)	420	6:24.13

Questions

Fulton Lee

E-mail: fultonlee@hotmail.com

Linkedin: <https://ca.linkedin.com/in/fulton-lee-aaa7871>

Reference

Parallel Processing Hands-On Workshop

<http://www2.sas.com/proceedings/sugi29/124-29.pdf>

SAS/Connect – Tips and Tricks

<https://support.sas.com/rnd/scalability/tricks/connect.html>