Capacity Planning Techniques for Growing SAS Workloads

Fred Forst - SAS Institute, Inc Cary, NC

March 16, 2012
Session Number 10916
Capacity Planning Techniques for Growing SAS Workloads

• SAS Business Intelligence Workload
  • HP ALM Performance Center Load Generator
  • Analysis of results

• Non-BI Workloads
  • Know your workload
  • Data Gathering – What does workload look like?
    ➢ SMF
    ➢ ARM
  • SAS workload modeler macro. ‘What if’ scenarios.
BI Workload

- **SAS Web Report Studio (WRS)**
  - OLAP Cube data
  - Currently 25 users randomly chose amongst 90 reports
  - Each report submits at least one query to the SAS OLAP Server
  - Simulate a growing user base, 25 – 400 users
  - **What happens to report generation elapsed times?**

- **HP ALM Performance Center®**
  - Simulates the human generating random WRS reports
  - Produces summary & detail reports of WRS simulation
Simulating a human

Browser

Logon WRS

Generate Report

Think 1 minute

Logoff WRS

1 sec pause

Repeat 3 hours

Repeat 10 times
Pseudo C-code (script)

```c
    do user=1 to max_users
        logon WRS url
        do rpts=1 to num_reports
            select random report
            generate report
            calculate elapsed time
            think think_time
        end;
    logoff WRS url
    stop if > max_hours
    pause x_seconds
    end;
```
Workflow Profile

- 90 SAS Web Report Studio Reports
- Four categories of reports:
  - Contract status reports (67% of users)
  - HR reports (18%)
  - IT Network reports (10%)
  - HR power user reports (5%)
Hardware and OS Profile

- IBM® zEnterprise z196 model 710 5.2GHz CP
- Four dedicated CPs
- 48Gb Memory
- IBM System Storage DS8800 Subsystem
- Eight Ficon Express8 Channels
- z/OS 1.12
- IBM WebSphere Application Server V7
- JAVA V1.6
SAS Profile

- SAS V9.3
- Four SAS OLAP Servers load balanced
- SAS Metadata server (64 bit)
- SAS Web Report Studio
Architecture of SAS Intelligence Platform

Data Sources
- SAS Data Sets
- SAS OLAP Cubes
- SAS Scalable Performance Data (SPD) Engine Tables
- SAS Scalable Performance Data (SPD) Server
- SAS Framework Data Server
- Third-party Data Stores
- Enterprise Resource Planning (ERP) Systems

SAS Servers
- SAS Metadata Server
- SAS OLAP Server
- SAS Stored Process Server
- SAS Pooled Workspace Server
- SAS Workspace Server
- Spawns SAS processes for distributed clients

Middle Tier
- Web Application Server
  - SAS Web Applications
    - Web Report Studio
      - Information Delivery Portal
      - BI Portlets
      - BI Dashboard
      - Help Viewer for the Web
      - Other SAS Web applications and solutions
    - Other SAS Web applications and solutions
  - SAS Web Infrastructure Platform
    - SAS Content Server
    - Other infrastructure applications & services
    - SAS Remote Services Java Server
  - Spawns SAS processes for distributed clients

Clients
- SAS Add-In for Microsoft Office
- SAS Data Integration Studio
- SAS Enterprise Guide
- SAS Enterprise Miner
- SAS Forecast Studio
- SAS Information Map Studio
- SAS Management Console
- SAS Model Manager
- SAS OLAP Cube Studio
- SAS Workflow Studio
- JMP
- Other SAS analytics and solutions
- Other SAS Web applications and solutions

Web Browser

Communication Protocols:
- HTTP
- JDBC
- Java RMI
Web Report Studio Simulations

Median Report Response Time

Median Report Generation time

Avg CPU Util

Users
CPU usage

• WebSphere – 62% of total (most eligible fo z/AAP)
• OLAP srv – 18%
• Metadata srv – 17%
• Other – 2%
Analysis

• Common Sense approach
  • Elbow in elapsed time curve seems to be around 300 users

<table>
<thead>
<tr>
<th>Users</th>
<th>Elapsed Time (avg report time)</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>13.23sec</td>
</tr>
<tr>
<td>300 (12x)</td>
<td>53.88 (4x) ← A 12x increase in users suffered only a 4x performance degradation</td>
</tr>
<tr>
<td>400 (16x)</td>
<td>378 (28x) ← A 16x increase in users suffered a 28x performance degradation</td>
</tr>
</tbody>
</table>

Conclusion: This hardware/software combination scales very well up to 300 users
Duncan Multiple Range Test

Good ‘ol T-Test
Hypothesis testing:

Avg Report time for 25 users same as for 50 users

\[ H_0: \mu_{25} = \mu_{50} \]
\[ H_1: \mu_{25} \neq \mu_{50} \]

Limitation: Tests the equality of only 2 groups
Duncan Multiple Range Test

\[ H_0: \mu_{25} = \mu_{50} = \mu_{75} \ldots = \mu_{400} \]
\[ H_1: \mu_{25} \neq \mu_{50} \neq \mu_{75} \ldots \neq \mu_{400} \]

Duncan Multiple Range test tests the hypothesis all groups are equal.
Duncan Multiple Range test gives statistical validity to results

<table>
<thead>
<tr>
<th>Duncan Grouping</th>
<th>Mean</th>
<th>users</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>377.997</td>
<td>400</td>
</tr>
<tr>
<td>B</td>
<td>224.995</td>
<td>375</td>
</tr>
<tr>
<td>C</td>
<td>138.599</td>
<td>325</td>
</tr>
<tr>
<td>D</td>
<td>121.200</td>
<td>350</td>
</tr>
<tr>
<td>E</td>
<td>53.879</td>
<td>300</td>
</tr>
<tr>
<td>F</td>
<td>43.920</td>
<td>250</td>
</tr>
<tr>
<td>G</td>
<td>31.319</td>
<td>225</td>
</tr>
<tr>
<td>H</td>
<td>27.643</td>
<td>150</td>
</tr>
<tr>
<td>I</td>
<td>24.959</td>
<td>200</td>
</tr>
<tr>
<td>J</td>
<td>17.760</td>
<td>125</td>
</tr>
<tr>
<td>K</td>
<td>15.413</td>
<td>75</td>
</tr>
<tr>
<td>K</td>
<td>13.319</td>
<td>50</td>
</tr>
<tr>
<td>K</td>
<td>13.230</td>
<td>25</td>
</tr>
</tbody>
</table>

Equal groupings can overlap

75 users and 200 users members of 2 groupings
Non-BI Workload

• Traditional SAS workload
• Don’t have HP Performance center
• Want to simulate growing # of users or growing data
• Step #1 – know your users
  • Batch or interactive?
  • What SAS PROCs do they use and how often?
  • Ad hoc usage or static jobs?
  • What happens to SAS session elapsed times?
  • What happens to SAS PROC elapsed times?
SAS log using option FULLSTIMER

NOTE: DATA statement used (Total process time):
real time 2.97 seconds
user cpu time 1.71 seconds
system cpu time 0.68 seconds
Memory 250k
OS Memory 6072k
Timestamp 1/24/2012 9:12:35 AM

Want these historical metrics in a dataset

1. Parse user SAS logs
2. Use SMF option (z/OS only)
3. Use ARM (any OS)
Data Gathering using SMF

//STEPSAS EXEC SAS,
// OPTIONS='SMFEXIT=SMFEXIT SMF'
//SYSIN DD *

...your SAS code...

//MXG      EXEC SAS9
//SMF      DD DISP=OLD,DSN=<SMF data>
//SOURCLIB DD DISP=SHR,DSN=MXG.SOURCLIB
//LIBRARY  DD DISP=SHR,DSN=MXG.LIBRARY
//SYSIN    DD *

%INC SOURCLIB(TYPESASU);
PROC PRINT DATA=WORK.TYPESASU;
  VAR SASEXCPS SASCORE SASPROC SASTCBTM;
  FORMAT SASTCBTM 8.4;
## Data Gathering using SMF

<table>
<thead>
<tr>
<th>OBS</th>
<th>EXCPS USED</th>
<th>MEMORY USED</th>
<th>PROCEDURE NAME OR SASDATA</th>
<th>TCB TIME</th>
<th>ELAPSED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>471</td>
<td>4766K</td>
<td>DATASETP</td>
<td>0.0726</td>
<td>0.3500</td>
</tr>
<tr>
<td>2</td>
<td>834</td>
<td>3785K</td>
<td>SORT</td>
<td>0.1273</td>
<td>0.6600</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>4601K</td>
<td>DATASETP</td>
<td>0.0012</td>
<td>0.0000</td>
</tr>
<tr>
<td>4</td>
<td>1587</td>
<td>13M</td>
<td>UNIVARIA</td>
<td>0.1513</td>
<td>0.6300</td>
</tr>
</tbody>
</table>
Data Gathering using ARM

%LET _ARMEXEC=1;
OPTIONS ARMLOC="/u/logs/user123.txt" ;
options ARMSUBSYS=(ARM_PROC);

%PERFINIT;
%PERFSTRT(TXNNAME="PROC_GLM_TEST");
   ...User code here...
%PERFSTOP;
Data Gathering using ARM

What is wrong with this picture?
Will users volunteer to code all the ARM stuff?

%LET _ARMEXEC=1;
OPTIONS
ARMLOC="/u/logs/&sysuserid._&datetime.txt";
ARMSUBSYS=(ARM_PROC);

%PERFINIT;
%PERFSTRT(TXNNAME="&get_creative");

...User code here...

%PERFSTOP;

Execute this via the initstmt option

Execute this via the termstmt option
Use the supplied **ARMPROC** macro to read raw data

<table>
<thead>
<tr>
<th>Obs</th>
<th>Timestamp</th>
<th>PROC</th>
<th>cputime</th>
<th>elapsed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20JAN12:16:33:44.0231</td>
<td>DATASTEP</td>
<td>0:00:00.0723</td>
<td>0:00:00.2810</td>
</tr>
<tr>
<td>2</td>
<td>20JAN12:16:33:44.7513</td>
<td>SORT</td>
<td>0:00:00.1259</td>
<td>0:00:00.7215</td>
</tr>
<tr>
<td>3</td>
<td>20JAN12:16:33:44.7549</td>
<td>DATASTEP</td>
<td>0:00:00.0013</td>
<td>0:00:00.0034</td>
</tr>
<tr>
<td>4</td>
<td>20JAN12:16:33:45.4218</td>
<td>UNIVARIA</td>
<td>0:00:00.1503</td>
<td>0:00:00.6417</td>
</tr>
</tbody>
</table>
common SAS config file

sasautos=(‘<sasautos_path>’ sasautos)
INITSTMT="%arm_start;run;"
TERMSTMT="%arm_stop;RUN;"

or

sasautos=(‘<sasautos_path>’ sasautos)
smf
smfexit=smfexit
After gathering data…

• 25 Interactive SAS sessions
• Average session has 40 PROCs/DATA
• Think time between PROCs is 60 seconds
• No pattern: Ad hoc workload
## Distribution of PROCs

<table>
<thead>
<tr>
<th>proc</th>
<th>Frequency</th>
<th>Percent</th>
<th>Cumulative Frequency</th>
<th>Cumulative Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATASTEP</td>
<td>13917</td>
<td>41.30</td>
<td>13917</td>
<td>41.30</td>
</tr>
<tr>
<td>SORT</td>
<td>8200</td>
<td>24.33</td>
<td>22117</td>
<td>65.63</td>
</tr>
<tr>
<td>REPORT</td>
<td>1026</td>
<td>3.04</td>
<td>23143</td>
<td>68.67</td>
</tr>
<tr>
<td>FREQ</td>
<td>1020</td>
<td>3.03</td>
<td>24163</td>
<td>71.70</td>
</tr>
<tr>
<td>MEANS</td>
<td>1019</td>
<td>3.02</td>
<td>25182</td>
<td>74.72</td>
</tr>
<tr>
<td>UNIVARIA</td>
<td>977</td>
<td>2.90</td>
<td>26159</td>
<td>77.62</td>
</tr>
<tr>
<td>SUMMARY</td>
<td>969</td>
<td>2.88</td>
<td>27128</td>
<td>80.50</td>
</tr>
<tr>
<td>CONTENTS</td>
<td>674</td>
<td>2.00</td>
<td>27802</td>
<td>82.50</td>
</tr>
<tr>
<td>CPORT</td>
<td>668</td>
<td>1.98</td>
<td>28470</td>
<td>84.48</td>
</tr>
<tr>
<td>PRINT</td>
<td>667</td>
<td>1.98</td>
<td>29137</td>
<td>86.46</td>
</tr>
<tr>
<td>GLM</td>
<td>663</td>
<td>1.97</td>
<td>29800</td>
<td>88.43</td>
</tr>
<tr>
<td>TRANSPOS</td>
<td>650</td>
<td>1.93</td>
<td>30450</td>
<td>90.36</td>
</tr>
<tr>
<td>APPEND</td>
<td>649</td>
<td>1.93</td>
<td>31099</td>
<td>92.28</td>
</tr>
<tr>
<td>COPY</td>
<td>647</td>
<td>1.92</td>
<td>31746</td>
<td>94.20</td>
</tr>
<tr>
<td>COMPARE</td>
<td>644</td>
<td>1.91</td>
<td>32390</td>
<td>96.11</td>
</tr>
<tr>
<td>CORR</td>
<td>643</td>
<td>1.91</td>
<td>33033</td>
<td>98.02</td>
</tr>
<tr>
<td>PLOT</td>
<td>346</td>
<td>1.03</td>
<td>33379</td>
<td>99.05</td>
</tr>
<tr>
<td>FORMAT</td>
<td>321</td>
<td>0.95</td>
<td>33700</td>
<td>100.00</td>
</tr>
</tbody>
</table>
Load generator SAS macro using SAS/CONNECT

```sas
%macro master(jobs=25,
    obs=.25,
    spacing=5,
    think_time=60,
    procs=40,
    wgt_data=.41,
    wgt_sort=.24,
    wgt_univ=.03,
    wgt_means=.03,
    wgt(glm=).02,
    wgt_freq=.03,
    wgt_compare=.02,
    ... ...
    wgt_format=.01);
```

Create 25 async SAS sessions
Read 25% of the data
Pause 5 secs between sessions
Pause 60 secs between SAS PROCS
Build job with 40 SAS PROCS

Randomly chose a PROC in this Proportion. Must add to 1.0
Master SAS job

```sas
%master(,,,
%do i=1 %to &jobs;
  %sleep(5);
  signon session&i;
  rsubmit session&i;
  %build_job;
  %do j=1 %to &procs;
    ...generate PROCs...
  %end;
  %update_results;
endrsubmit;
  signoff session&i;
%end;
```

1. DATA step—
2. PROC MEANS—
3. PROC SORT—
4. PROC GLM—
5. DATA step—

Each session
Automatically
Uses ARM
40 SAS PROC Session

Think Time=60
DATA Step Elapsed Time
40 SAS PROCs, Think time=60

Median Elapsed Time

0:00:20.0
0:00:19.0
0:00:18.0
0:00:17.0
0:00:16.0
0:00:15.0
0:00:14.0
0:00:13.0
0:00:12.0
0:00:11.0
0:00:10.0
0:00:09.0
0:00:08.0
0:00:07.0
0:00:06.0
0:00:05.0
0:00:04.0
0:00:03.0
0:00:02.0
0:00:01.0

Users

1
25
50
75
100
125
150
175
200
225
250

*** DATA step actual time
red - DATA step predicted(quadratic model: predicted = 4.4 + 0.019*users - 0.0003*users**2 + 0.00000*users**3)
black - DATA step predicted(exponential model: predicted = 3.077*exp(0.00665*users)
### DATA step only – Duncan test

<table>
<thead>
<tr>
<th>Duncan Grouping</th>
<th>Mean</th>
<th>N</th>
<th>jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>21.0045</td>
<td>4251</td>
<td>250</td>
</tr>
<tr>
<td>B</td>
<td>15.8227</td>
<td>3784</td>
<td>225</td>
</tr>
<tr>
<td>C</td>
<td>11.0053</td>
<td>3392</td>
<td>200</td>
</tr>
<tr>
<td>D</td>
<td>8.5921</td>
<td>2896</td>
<td>175</td>
</tr>
<tr>
<td>E</td>
<td>6.9519</td>
<td>2588</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td>5.7296</td>
<td>2110</td>
<td>125</td>
</tr>
<tr>
<td>F</td>
<td>5.2388</td>
<td>1699</td>
<td>100</td>
</tr>
<tr>
<td>F</td>
<td>4.9806</td>
<td>1290</td>
<td>75</td>
</tr>
<tr>
<td>F</td>
<td>4.6975</td>
<td>864</td>
<td>50</td>
</tr>
<tr>
<td>F</td>
<td>4.5963</td>
<td>407</td>
<td>25</td>
</tr>
<tr>
<td>F</td>
<td>4.5896</td>
<td>21</td>
<td>1</td>
</tr>
</tbody>
</table>

No degradation in DATA step performance on the z196 up to 125 concurrent SAS sessions
60 SAS PROC Session
Think Time=0

![Graph showing average elapsed time for different numbers of users. The time increases as the number of users increases.]
60 SAS PROC Session
Think Time=0

![Graph showing the average elapsed time for different numbers of users, with lines indicating 100K, 200K, and 400K observations read.](image-url)
Summary

- HP Performance Center is versatile in it’s ability to model BI web based apps
- DATA step elapsed times can be predicted closely with polynomial regression (slide#29)
- No DATA step degradation on the z196 for up to 125 concurrent SAS sessions (slides 29, 30)
- The z196 scales well (proportionally) across all values of concurrent SAS sessions when amount of data increases (slide#32)
• SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

• IBM is a registered trademark of International Business Machines Corporation (IBM)

• HP is a registered trademark of Hewlett Packard Corporation (HP)

• Other brand and product names are registered trademarks or trademarks of their respective companies.

fred.forst@sas.com