



Managing and Using a SAS/Share Server

Jason Liang

Jason.Liang@cwbank.com



Agenda

- Introduction
- SAS Data Library Connection Diagram
- Managing SAS/Share Server and Libraries
- Understanding SAS/Share Locking
- SAS Programing Consideration
- Performance Consideration

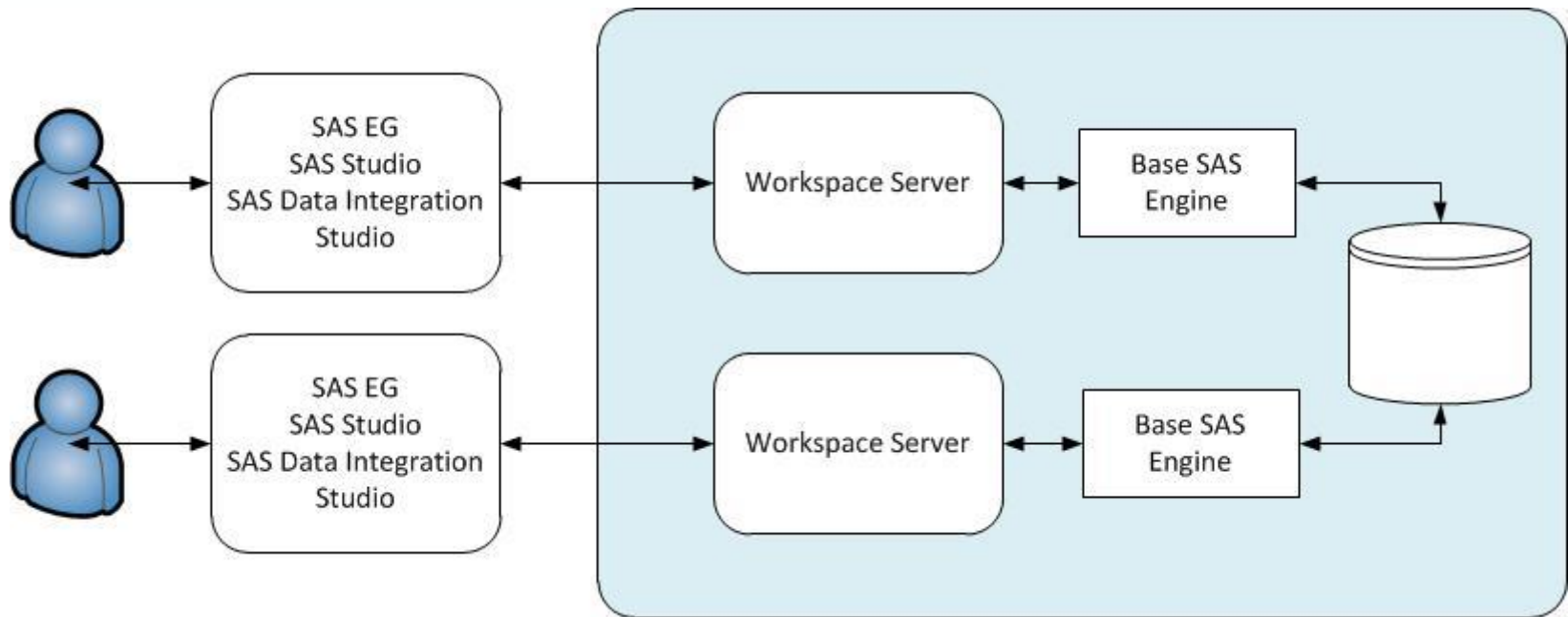


Introduction

- Base SAS library access restriction:
 - Any number of users can read a data set
 - Only a single user can write or update a data set
- As a result, you often get the following error:
 - **ERROR: A lock is not available for MYLIB.IMPORT.DATA.**
- A SAS/Share server:
 - A multi user data server
 - Enables concurrent update access to data sets
 - Allow remote access of SAS data sets through SAS/SHARE Remote Engine and SAS ODBC driver

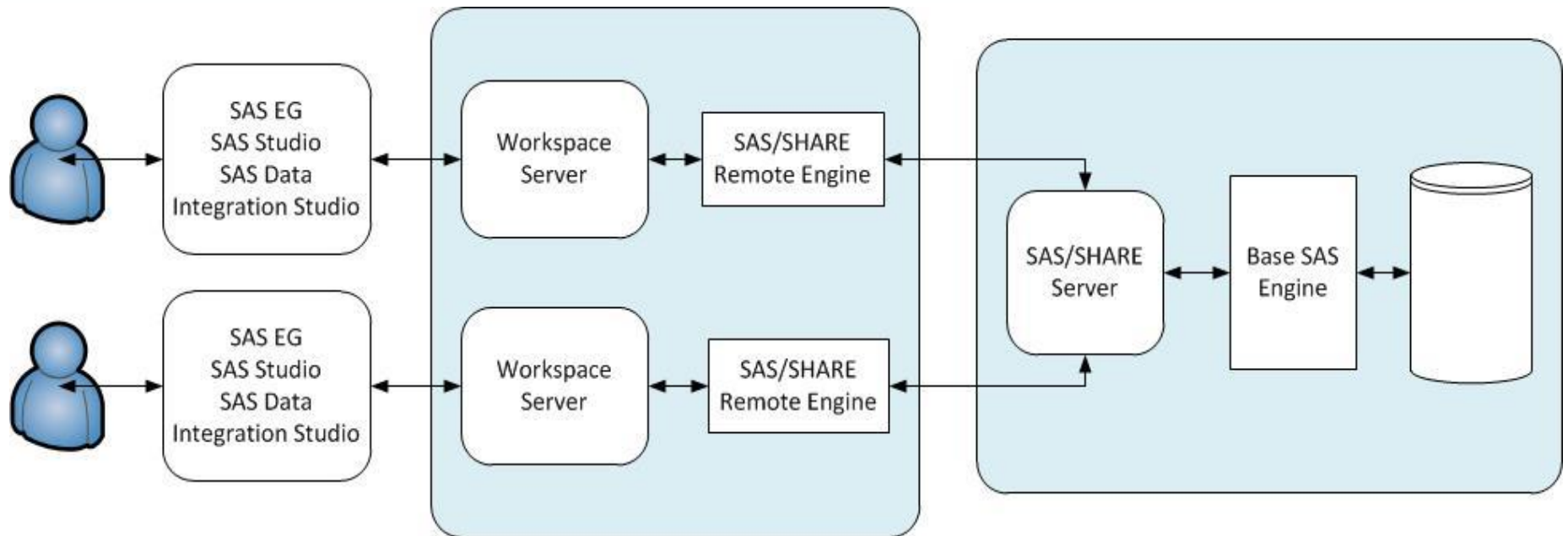


SAS Workspace Library Connection





SAS/SHARE Library Connection





Managing SAS/SHARE Server and Libraries

SAS Procedures for SAS/SHARE:

```
Proc server id=__8551 authenticate=required;  
run;
```

```
Proc operate server=__8551;  
  Stop server __8551;  
Run;
```

Assigning SAS/SHARE (Server Session) Libraries

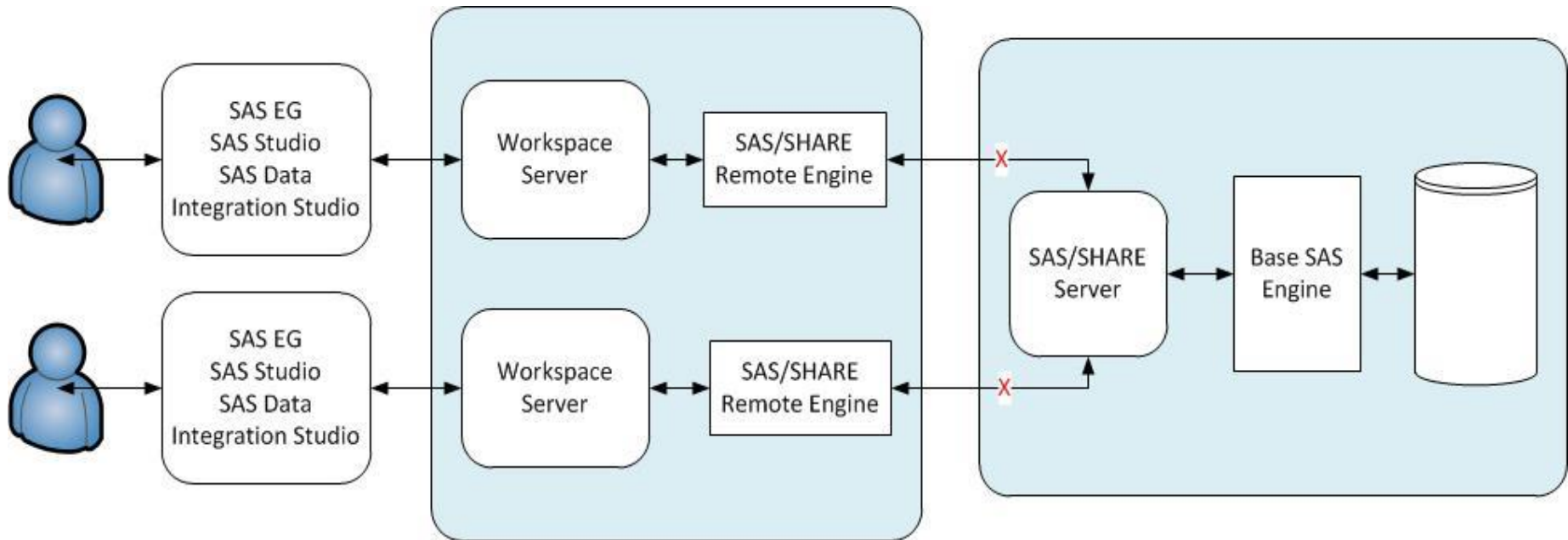
- 1) Pre-assign a library to SAS/SHARE server
- 2) Assign library using SAS/SHARE autoexec SAS program
- 3) Use Proc operate

```
Proc operate server=__8551;  
  Allocate library mylib Base 'G:\SASUsers\Liangj';  
Run;
```



Managing SAS/SHARE Server and Libraries

The libraries assigned to SAS/SHARE server is not accessible in a user's SAS session





Managing SAS/SHARE Server and Libraries

Assigning the Library to Workspace Server (in a user session) Using SAS/SHARE Remote Engine:

- 1) Pre-assign
- 2) Assign library using workspace server autoexec SAS program
- 3) Libname statements in a user session:

```
Libname mylib server=__8551;  
Libname newlib server=__8551 slibref=mylib;
```

- 4) When SAS/SHARE and Workspace server are not on the same host

```
Libname mylib hostname=<host name of SAS/SHARE server> server=__8551;  
Libname newlib hostname= =<host name of SAS/SHARE server> server=__8551 slibref=mylib;
```




Managing SAS/SHARE Server and Libraries

View SAS/SHARE server Libraries

```
Proc operate server=__8551;  
    Display library _all_;  
Run;
```

SAS Log:

25 Display library _all_;

LIBREF	STATUS	NUMBER OF USERS	LIBRARY NAME
SHA_SCOR	ACTIVE	0	G:\Risk Data Mart\Scoring
NL_ARCH	INACTIVE	0	G:\Risk Data
OSCAR	INACTIVE	0	OSCAR
JL_SHARE	INACTIVE	0	G:\SASData\Users\Liangj

=====
26 Run;



Managing SAS/SHARE Server and Libraries

Assigning new library directly from a user session:

```
libname jl 'G:\SASUser\Liangj' server=__8551;
```



Understand SAS/SHARE Locking

Which Data Object Is Locked	Mode in Which Data Object Is Locked		
	Input	Update	Output
Member	Other operations can read the data set but cannot open it for update or output.	No other operations can access the data set.	No other operations can access the data set.
Observation	Other operations can read or update the data set but cannot open it for output.	Other operations can read or update the data set but cannot open it for output.	No other operations can access the data set.



Understand SAS/SHARE Locking

SAS Operation		What the Data Object Is Locked For	The Data Object Locked (Default)
DATA step			
DATA statement			
	without MODIFY statement	Output	Member
	with MODIFY statement	Update	Observation
SET statement			
	without POINT= and KEY= options	Input	Observation
	with POINT= and KEY= options	Input	Member
MERGE statement		Input	Observation
MODIFY statement			
	without POINT= and KEY= options	Update	Observation
	with POINT= and KEY= options	Update	Member
UPDATE statement		Input	Observation
Procedures			
APPEND procedure			
	BASE= option	Update	Member* or observation
	DATA= option	Input	Observation



Understand SAS/SHARE Locking

SAS Operation		Locked for	Default Lock Level
Copy procedure			
	IN= option	Input	Observation
	IN= option with MOVE option	Output	Observation
	OUT= option	Output	Member
PRINT procedure			
	DATA=option	Input	Observation
SORT procedure			
	DATA=option	Input	Observation
	OUT=option	Output	Member



Concurrent Update Example

```
libname nl_arch server=__8551;  
%let leaseArch=nl_arch.lease_arch;  
.....  
proc append base=&leaseArch. data=lease force;  
run;
```

Time Stamp	Action ID	Action Description	Target Table
13MAR2017:16:08:13.870	APPEND (9)(15)	opened for update	Lease_Arch
13MAR2017:16:08:13.886	APPEND (11)(16)	opened for update	Lease_Arch
13MAR2017:16:08:13.886	APPEND (12)(17)	opened for update	Lease_Arch
13MAR2017:16:08:13.995	APPEND (9)(15)	closed	Lease_Arch
13MAR2017:16:08:13.995	APPEND (11)(16)	closed	Lease_Arch
13MAR2017:16:08:13.995	APPEND (10)(18)	opened for update	Lease_Arch
13MAR2017:16:08:14.011	APPEND (12)(17)	closed	



Programing Consideration for Concurrent Update

- When attempting to update a data set, another program may already have it locked.
- When reading a data set, the result may be affected by other program updating the same data set.

Example:

```
libname sharelib server=__8551;  
proc sort data=sharelib.cars;  
    by type origin;  
run;  
  
proc print data=sharelib.cars;  
run;
```



Programming Consideration for Concurrent Update

- Create your own copy of the data set for processing and reporting:

```
proc sort data=sharelib.cars out=work.cars;  
    by type origin;  
run;
```

- Lock of the data set, update only if the lock is successful

```
lock sharelib.cars;  
  
data _null_;  
    x=symget('syslckrc');  
    if (x ne '0') then do;  
        put 'ERROR:No lock is available for sharelib.cars, end sas';  
        abort abend;  
    end;  
run;  
  
proc sort data=sharelib.cars;  
    by type origin;  
run;  
  
proc print data=sharelib.cars;  
run;  
  
lock sharelib.cars clear;
```




SAS/SHARE Performance Consideration

- Separate Readonly data, use option ACCESS=Readonly, don't use SAS/SHARE for them
- For busy sites, create multiple SAS/SHARE servers for load balancing



Questions ?