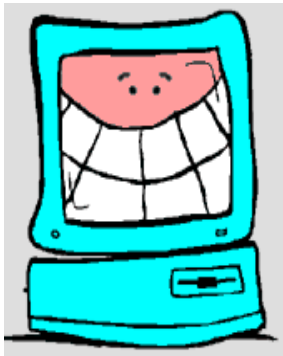
DATABASE ADMINISTRATION

ECS Pre-Release B Testbed Training

Objectives

- DBA Tasks and Functions
- DBA's Environment
- Startup and shutdown SQL servers
- Create new database devices
- Create and Modify User Databases
- Database Segments
- Transaction Logs
- Backup and recovery
- Configuring, tuning, and monitoring

DBA Tasks and Functions

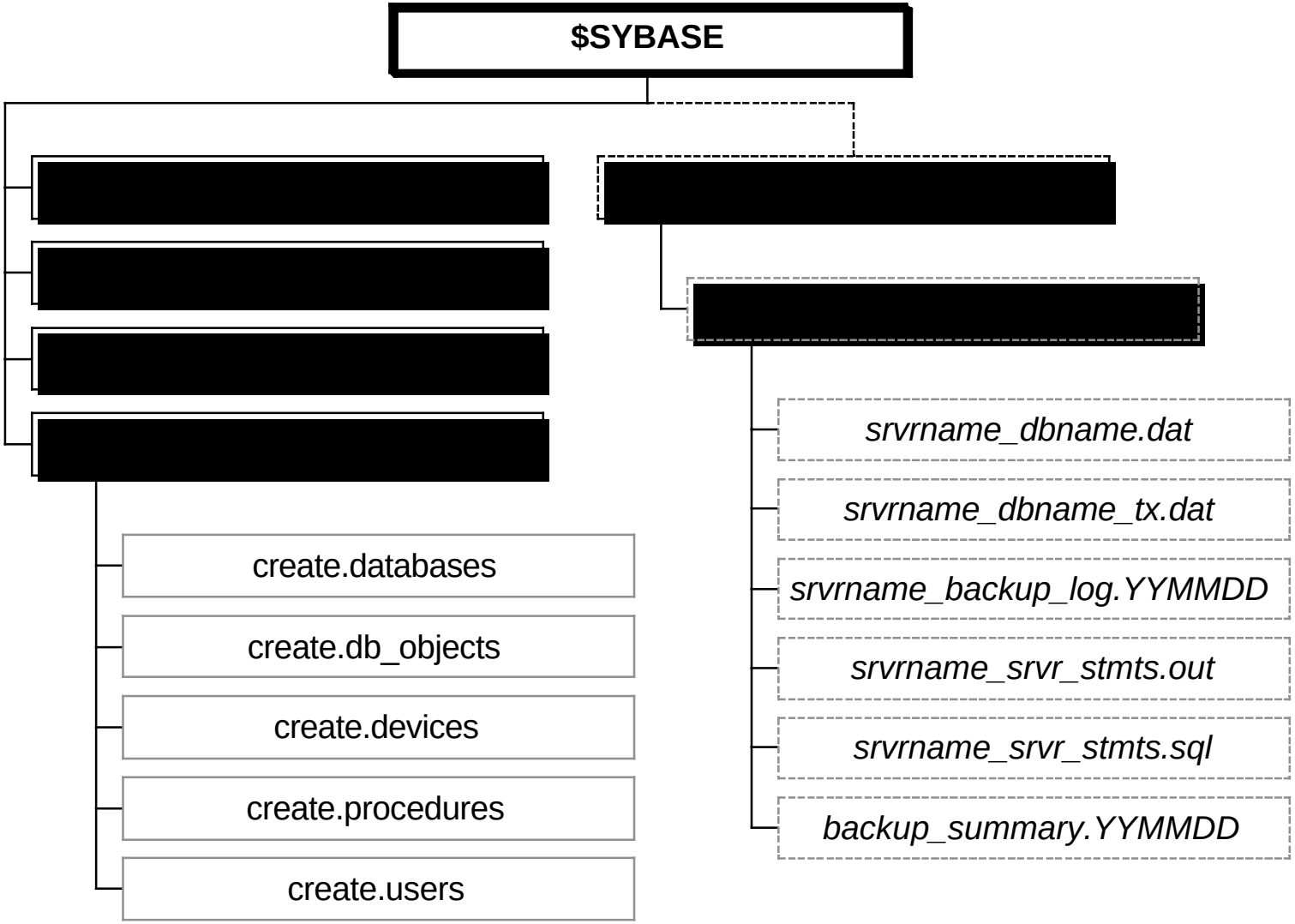


- Perform database backup, transaction log maintenance, and database recovery.
- Monitor and tune the physical allocation of database resources.
- Maintain user accounts.
- Create user registration and account access control permissions in the security database.
- Work with data specialists on DB design, data sets, and metadata management.

Databases To Be Administered

- autosys_srvr
- autosys_srvr_backup
- pdps_db_ops_srvr
- pdps_db_ops_srvr_backup
- master
- model
- sybsystemprocs
- sybsecurity

Directory Structure



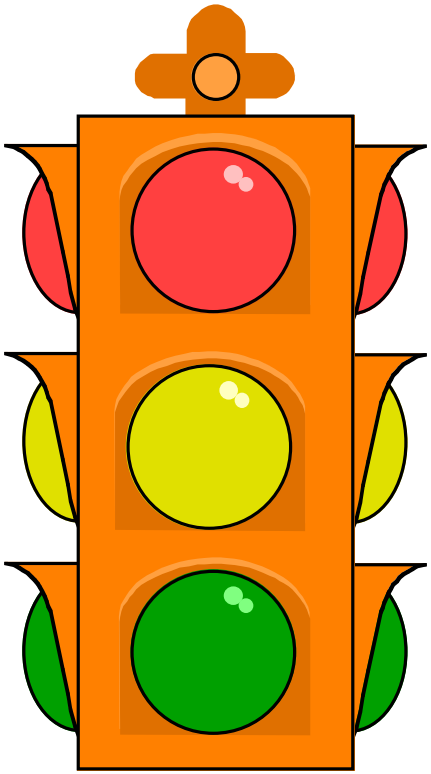
File Naming Conventions

- File name indicates function and/or content.
- Easily understood abbrevs. only.
- Underscore (_) to separate words.
- One optional suffix with period separator.
- Full path is part of the name.
- All SQL files end with *.sql*
- All output files end with *.out*
- All backup data files end with *.dat*

The Key to Success

S C R I P T S
S C R I P T S
S C R I P T S
S C R I P T S
S C R I P T S
S C R I P T S
S C R I P T S
S C R I P T S
S C R I P T S

Start & Stop SQL Server

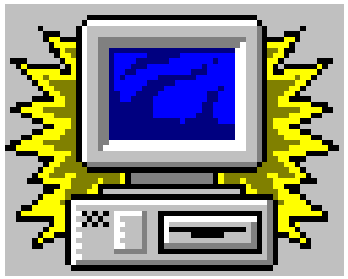


Stop the server before maintenance

Start the server after installation, system outage or maintenance

Database Device

- **Stores objects that make up databases**
- **May be:**
  - **a distinct physical device**
  - **a disk partition**
  - **a file**



- **Must be initialized first**

Initializing a Database Device Command Script Template

```
/*****************************************/
/* name: [template.sql]                  */
/* purpose:                              */
/* written:                              */
/* revised:                              */
/* reason:                               */
/*****************************************/
disk init name = [device name],
physname = "/dev/[device name]",
vdevno = [#],
size = [size]
```

Initializing a Database Device Completed Sample Script

```
/*****************************************/
/* name: datadev1.sql                    */
/* purpose: tracking training program    */
/* written: 4/25/97 mresnick             */
/* revised:                              */
/* reason:                               */
/*****************************************/
disk init name = datadev1,
physname = “/dev/rdsk/c0t0d4s3“,
vdevno = 5,
size = 5120
```

User Databases

- Special databases
- Requested by a system user to track specific data.
- REQUESTER must know the following things:
  - Name of the database
  - Approximate size of the database
- DBA must know the following things:
  - Which database device to use
  - Location of database and transaction logs

User Database Request Form

User Database Request Form

REQUESTER INFORMATION:
Name: _____
Office Phone Number: _____
E-Mail Address: _____ Office Location: _____

DATABASE(S) TO BE CREATED:

_____
_____
_____

JUSTIFICATION: _____
_____
_____
_____

Date of Request: _____ Date Required: _____

Supervisor Approval: _____ Date: _____
Ops Supervisor Approval: _____ Date: _____

Create New Database Command Script Template

```
/*****************************************/
/* name: [template.sql]                  */
/* purpose:                              */
/* written:                              */
/* revised:                              */
/* reason:                               */
/*****************************************/
create database [DBName]
on [DBDeviceName] = [DBSize_in_MB]
log on [LogDBDeviceName] = [LogSize_in_MB]
```

Create New Database
Completed Sample Script

```
/*****************************************/
/* name: mydb.sql                        */
/* purpose: track specific data items    */
/* written: 4/26/97 mresnick             */
/* revised:                              */
/* reason:                               */
/*****************************************/
create database mydb
on datadev1 = 10
log on datadev1 = 2
```

Database Segments

- **Collection of database devices or fragments available to a particular database**
- **Can have tables and indexes assigned to it**
- **Can span more than one physical device**
- **Created when the database is created *or* when DBA deems necessary *or* as part of the database recovery procedure**

Why Database Segments?

- **Reduces read/write access time**
- **Increases SQL Server performance**
- **Added administrative control over placement, size, and space usage of specific database objects**

Create Database Segments Command Script Template

```
/*****************************************/
/* name: [template.sql]                  */
/* purpose:                              */
/* written:                              */
/* revised:                              */
/* reason:                               */
/*****************************************/

sp_addsegment [seg_name], [DBname], [DBDevice]
go
```

What is a Transaction Log?

- **Automatically records every transaction issued by each user of the DB.**
- **Keeps track of all changes to the database.**
- **Each database has its own transaction log.**
- **Cannot be turned off.**
- **Write-ahead file. Changes are reversed if transaction fails to complete.**

Database Data Backup

- Databases data are backed up with the regular system backups
- Can be requested at any time
- Need to know the following:
  - Name of DB to be backed up
  - Name of the server on which the DB resides
  - Name of the backup volume
  - Name of the dump file on the backup volume

Transaction Log Backup

- **Transaction Log - an expanding file that records all database transactions.**
- **Used for complete recovery of DB if media fails.**
- **Maintained on a different device than DB**
- **Backed up with regular system backups, but non-scheduled backups can be performed with permission.**

RUN_backup.sh Script

```
# !/bin/sh
# name: RUN_backup.sh
# purpose: executes backups for plnn1sun_srvr and
#            autosys1_srvr on the plnn1sun only
# written:  3/27/97
# revised:
# reason:

YMD=`date -u +%y%m%d`
DUMPDIR=/usr/ecs/Rel_A/COTS/sysbase/backups/backups_for_$YMD

# create a directory for nighly backups
mkdir $DUMPDIR

# run dbcc script for each server
/usr/ecs/Rel_A/COTS/sybase/scripts/backups/START_dbcc

# run backup script for each server
/usr/ecs/Rel_A/COTS/sybase/backups/START_backup
```

dbcc memusage Sample Output

	Meg.	2K Blks	Bytes
Configured Memory:	400.0000	204800	419430400
Code size:	3.4259	1755	3592296
Kernel Structures:	5.9769	3061	6267212
Server Structures:	13.9494	7143	14627040
Cache Memory:	357.0625	182816	374407168
Proc Buffers:	0.6974	358	731272
Proc Headers:	18.8848	9669	19802112

START_backup Script

```
# !/bin/sh
# name: START_backup
# purpose: executes backups for plnn1sun_srvr and
#            autosys1_srvr on the plnn1sun only
# written:  3/27/97
# revised:
# reason:

YMD=`date -u +%y%m%d`
BACKUPDIR=/usr/ecs/Rel_A/COTS/sybase/backups
DUMPDIR=/usr/ecs/Rel_A/COTS/sysbase/backups/backups_for_$YMD
SCRIPTDIR="/usr/ecs/Rel_A/COTS/sytbase/scripts/backups"
PASSWORD="`cat $SCRIPTDIR/sp`"

# run backup script for each server
isql -Usa -P$PASSWORD -Splnn1sun_srvr $i$SCRIPTDIR/backup/plnn1sun.sh > $BACKUPDIR
isql -Usa -P$PASSWORD -Sautosys1_srvr $i$SCRIPTDIR/backup/autosys1.sh > $BACKUPDIR
```

Database Recovery/ Database Device Restoration

- Device Failure verified by SA
- SA requests a restoration from the DBA
- Transaction log for each DB on the failed device is backed up
- DBA examines space usage of each DB on failed device.
- DB(s) on the failed device are deleted; then device is deleted
- DBA initializes new DB device
- DBA recreates each user DB on the new device
- Each DB is restored from DB backups and transaction logs
- DBA notifies SA when restoration is complete.

Configure SQL Server

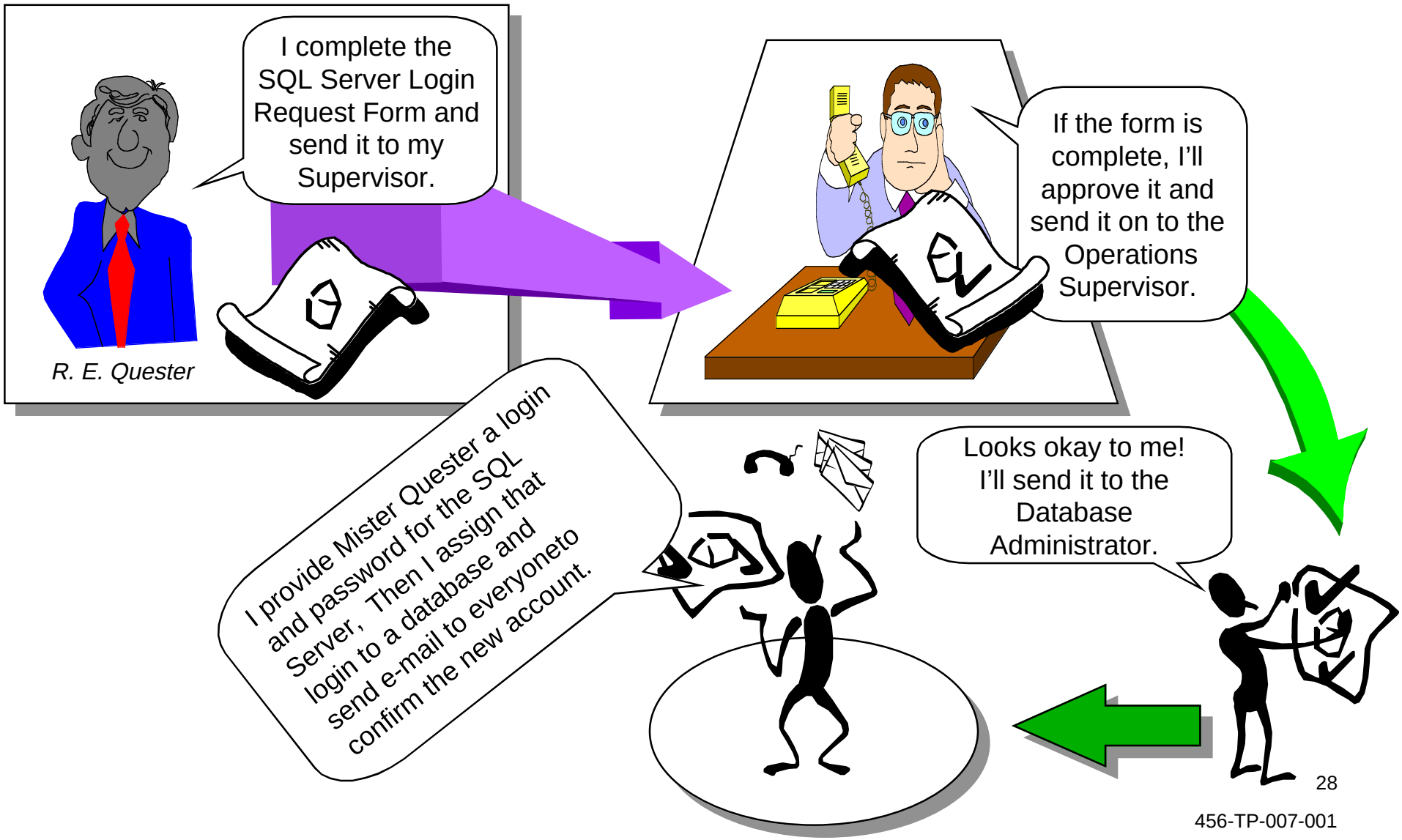
- Customization
- Fine Tuning
- Optimize memory allocation or performance
- Configuration Variables:
  - allow/deny updates
  - audit queue size
  - password expiration interval
  - remote access
- Some values take effect immediately, others require a server reboot.
  - When in doubt, reboot!

sp_config Sample Output

```
>go
 name                          minimum     maximum     config_value run_value
 ----------------------------- ----------- ----------- ------------ -----------
 recovery interval                       1       32767            0           5
 allow updates                           0           1            0           0
 user connections                        5  2147483647            0          25
 memory                               3850  2147483647            0        5120
 open databases                          5  2147483647            0          12
 locks                                5000  2147483647            0        5000
 open objects                          100  2147483647            0         500
 procedure cache                         1          99            0          20
 fill factor                             0         100            0           0
 time slice                             50        1000            0         100
 database size                           2       10000            0           2
 tape retention                          0         365            0           0
 recovery flags                          0           1            0           0
 nested triggers                         0           1            1           1
 devices                                 4         256            0          10
 remote access                           0           1            1           1
 remote logins                           0  2147483647            0          20
 remote sites                            0  2147483647            0          10
 remote connections                      0  2147483647            0          20
 pre-read packets                        0  2147483647            0           3
 upgrade version                         0  2147483647         1002        1002
 default sortorder id                    0         255           50          50
 default language                        0  2147483647            0           0
 language in cache                       3         100            3           3
 max online engines                      1          32            1           1
 min online engines                      1          32            1           1
 engine adjust interval                  1          32            0           0
 cpu flush                               1  2147483647          200         200
 i/o flush                               1  2147483647         1000        1000
 default character set id                0         255            1           1
 stack size                          20480  2147483647            0       28672
 password expiration interval            0       32767            0           0
 audit queue size                        1       65535          100         100
 additional netmem                       0  2147483647            0           0
 default network packet size           512      524288            0         512
 maximum network packet size           512      524288            0         512
 extent i/o buffers                      0  2147483647            0           0
 identity burning set factor             1     9999999         5000        5000
 allow sendmsg                           0           1            0           0
 sendmsg starting port number            0       65535            0           0

(40 rows affected, return status = 0)
```

SQL Server Login Approval Process



SQL Server Login Account Request Form

SQL Server Login Account Request

REQUESTER INFORMATION:

Name: _____

UNIX ID: _____ Group: _____

Office Phone Number: _____

E-Mail Address: _____ Office Location: _____

Database(s) to be accessed: _____

_____

_____

Permissions require for database objects:_____

_____

Justification: _____

_____

_____

Date of Request: _____ Date Required: _____

Supervisor Approval: _____ Date: _____

Ops Supervisor Approval: _____ Date: _____

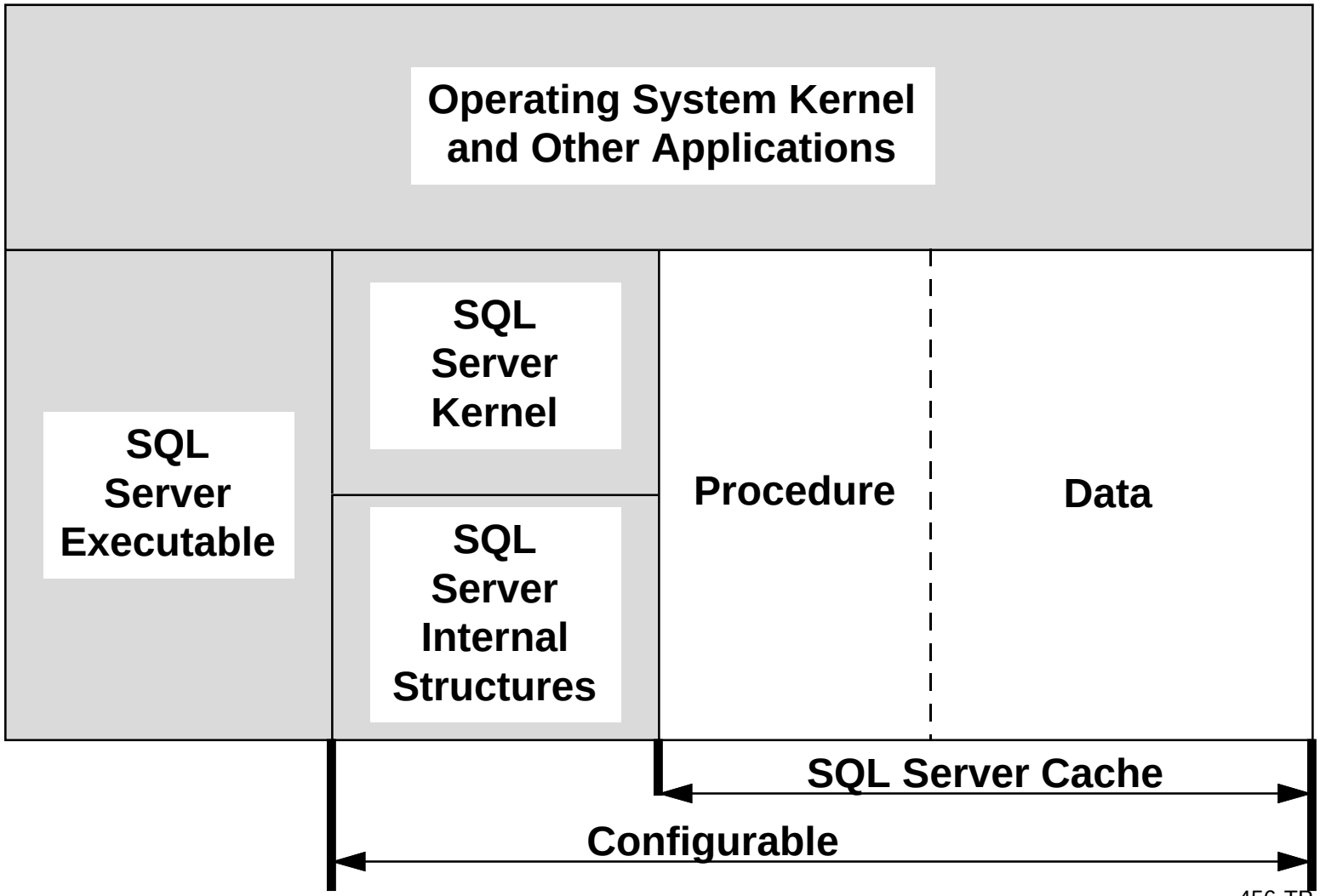
Database Access Privileges

Assign a user to a group that has specific access privileges.

Assign a user object permissions.

Assign a user command permissions.

Database Tuning and Performance Monitoring

- Use *sp_config* to determine current configuration parameters and set future run values.
- Use *dbcc memusage* to determine current memory usage.
- Use *sp_spaceused* to determine how much space has been used on the device.

Physical Memory Utilization Scheme