

Best Practices for Storing the Product Library Workspace in an Enterprise Geodatabase for Oracle



Copyright © 2011 Esri
All rights reserved.
Printed in the United States of America.

The information contained in this document is the exclusive property of Esri. This work is protected under United States copyright law and other international copyright treaties and conventions. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage or retrieval system, except as expressly permitted in writing by Esri. All requests should be sent to Attention: Contracts and Legal Services Manager, Esri, 380 New York Street, Redlands, CA 92373-8100 USA.

The information contained in this document is subject to change without notice.

Esri, the Esri globe logo, ArcSDE, ArcMap, ArcCatalog, ArcGIS, SDE, Spatial Database Engine, esri.com, arcgis.com, and @esri.com are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions. Other companies and products mentioned herein may be trademarks or registered trademarks of their respective trademark owners.

Best Practices for Storing the Product Library Workspace in an Enterprise Geodatabase for Oracle

An Esri White Paper

Contents	Page
Introduction.....	1
Overview of SDE DBTUNE.....	1
Disk Configuration.....	2
Reduce Disk I/O Contention.....	2
Step 1: Create Data Files.....	2
Step 2: Create the PRODLIB User	3
Step 3: Modify DBTUNE	4
Step 4: Create the Product Library Database Connection	4
Step 5: Configure Oracle Parameters.....	4
Step 6: Configure ArcSDE Parameters.....	5
Step 7: Create the CKB_USERS Role.....	5
Step 8: Create the Product Library Workspace.....	5
Step 9: Verify the Storage.....	7

Contents	Page
Step 10: Register as Versioned	7
Step 11: Validate Permissions and Roles.....	7
Grant Permissions Using ArcCatalog.....	8
Step 12: Configure Log File Tables.....	9
Create Log File Tablespaces.....	9
Change DBTUNE Log File Parameters.....	9
Create Log File Tables.....	9
Step 13: Create the SDE Product Library User	10
Step 14: Database Connections for Product Library Users	10
Step 15: Product Library Permissions.....	10
Step 16: Add New Users to the Product Library	10
Assigning Permissions to Users.....	12
Conclusion	13

Best Practices for Storing the Product Library Workspace in an Enterprise Geodatabase for Oracle

Introduction

The product library is part of the Esri® Production Mapping extension. It is a geodatabase that allows multiuser environments to centralize information and behavior for cartographic and digital data production. Production business rules, documents, and spatial information are stored inside the product library, allowing an organization to enforce and standardize production. Data model information, data validation rules, geographic extents, symbology rules, and map documents can all be managed inside the product library as examples of production business rules. In other words, the product library is essentially a geographic document management system. When stored in an enterprise geodatabase, the workspace supports versioning. This white paper is intended for database administrators to help them establish the product library workspace in an enterprise geodatabase for Oracle. The enterprise geodatabase uses ArcSDE® technology as the gateway between geographic information system (GIS) clients and Oracle.

Overview of SDE DBTUNE

DBTUNE storage parameters let you control how ArcSDE technology creates objects within an Oracle® database. They allow you to determine things such as how to allocate space to a table or index and which tablespace a table or index is to be created in, as well as other Oracle-specific storage attributes. They also let you specify one of the available storage formats for the geometry of a spatial column.

The DBTUNE storage parameters are stored in the DBTUNE table. The DBTUNE table, along with all other metadata tables, is created during the setup phase that follows the installation of ArcSDE. ArcSDE installation creates a dbtune file under the etc directory from which the DBTUNE table is populated. If no dbtune file is present during setup, the DBTUNE table will be populated with default values.

If a large number of database connections are accessing the same files in the same location on disk, database performance will be slower, because the connections are competing with one another for the same resources. To reduce this competition, you can store database files in different locations on the disk.

Thus, DBTUNE can be modified to store the Reviewer feature dataset and tables in separate data files in different locations on the disk. This will lead to reduced disk contention and improved database input/output (I/O).

Standard GIS storage recommendations favor keeping index and log files separate from vector and tabular business tables. For performance reasons, it is better to position the business, feature, and spatial index tables separately and position tablespace data files based on their usage pattern. For a multiversed, highly active editing geodatabase,

database files of VERSIONS tablespace may be separated and dispersed across available disks to avoid I/O contention.

Disk Configuration

Large production enterprise geodatabase systems should employ a hardware striping solution. The best strategies for disk and data organization involve spreading your data across multiple disks so that more spindles actively search for it. This can increase disk read time and decrease disk contention. However, too many disks can slow down a query. There are two main ways of achieving striping: tablespaces and redundant array of independent disks (RAID). You can also combine the two by creating tablespaces within disk arrays. You can employ data segregation strategies; keeping tables from indexes or certain types of tables from other tables will improve performance and alleviate administrative burdens.

Suggested Oracle optimal configuration is as follows:

- DISK 0—Oracle/App Software DISK 0
- DISK 1—SYSTEM, Control File 1
- DISK 2—RBS, TEMP, Control File 2
- DISK 3—REDO 1, 2, 3, Export Files
- DISK 4—Feature Data Tables
- DISK 5—Spatial Index Data Tables
- DISK 6—Attribute Data/Business Tables
- DISK 7—Oracle Indexes

Reduce Disk I/O Contention

As a rule, you should create the largest possible database files, based on the maximum amount of data you estimate the database will contain, to accommodate future growth. By creating large files, you can avoid file fragmentation and get better database performance. In many cases, you can let data files grow automatically; just be sure to limit autoextend by specifying a maximum growth size that leaves some hard disk space available. By putting different tablespaces on different disks, you can also help eliminate physical fragmentation of your files as they grow.

Below is a suggested design to reduce disk I/O contention:

File Type	Database Activity	Move File to Disk With
Redo log	Frequent edits	Relatively low I/O
Redo log	Few or no edits	Moderate I/O
Undo log files	Frequent edits	Low I/O but separate from redo log files
System data	Frequent edits	Moderate I/O
Temporary tablespace	Few edits	High I/O

Step 1: Create Data Files

Create new tablespaces to store the product library feature classes and tables:

TABLESPACE	SDE_PARAMETER
PRODLIB_BDATA	Business table
PRODLIB_BINDEX	Business table index
PRODLIB_FDATA	Feature table
PRODLIB_FINDEX	Feature table index
PRODLIB_SDATA	Spatial Index table
PRODLIB_SINDEX	Spatial Index table index
PRODLIB_ADATA	Adds table (versioned)
PRODLIB_AINDEX	Adds table index
PRODLIB_DDATA	Deletes table (versioned)

J10023

```
PRODLIB_DINDEX    Deletes table index

CREATE SMALLFILE TABLESPACE "PRODLIB_BDATA"
DATA FILE 'D:\oracle\ORADATA\PRODLIB\prodlib_bdata01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_BINDE"
DATA FILE 'D:\oracle\ORADATA\PRODLIB\prodlib_bindex01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_FDATA"
DATA FILE 'D:\oracle\ORADATA\PRODLIB\prodlib_fdata01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_FINDEX"
DATA FILE 'D:\oracle\ORADATA\PRODLIB\prodlib_findex01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_SDATA"
DATA FILE 'D:\oracle\ORADATA\PRODLIB\prodlib_sdata01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_SINDEX"
DATA FILE 'D:\oracle\ORADATA\PRODLIB\prodlib_sindex01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_ADATA"
DATA FILE 'D:\oracle\ORADATA\PRODLIB\prodlib_adata01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_AINDEX"
DATAFILE 'D:\oracle\ORADATA\PRODLIB\prodlib_aindex01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_DDATA"
DATAFILE 'D:\oracle\ORADATA\PRODLIB\prodlib_ddata01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE "PRODLIB_DINDEX"
DATAFILE 'D:\oracle\ORADATA\PRODLIB\prodlib_dindex01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
400M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K SEGMENT SPACE MANAGEMENT AUTO;
```

Step 2: Create the PRODLIB User

1. Create a new database user to store the product library feature classes and tables and grant the appropriate permissions:

```
CREATE USER PRODLIB PROFILE DEFAULT IDENTIFIED BY prodlib
DEFAULT TABLESPACE "PRODLIB_BDATA"
TEMPORARY TABLESPACE "TEMP"
QUOTA UNLIMITED ON "PRODLIB_BDATA"
QUOTA UNLIMITED ON "PRODLIB_BINDE"
QUOTA UNLIMITED ON "PRODLIB_FDATA"
QUOTA UNLIMITED ON "PRODLIB_FINDEX"
QUOTA UNLIMITED ON "PRODLIB_SDATA"
QUOTA UNLIMITED ON "PRODLIB_SINDEX"
QUOTA UNLIMITED ON "PRODLIB_ADATA"
QUOTA UNLIMITED ON "PRODLIB_AINDEX"
QUOTA UNLIMITED ON "PRODLIB_DDATA"
QUOTA UNLIMITED ON "PRODLIB_DINDEX"
ACCOUNT UNLOCK;
```

2. Grant privileges:

```
GRANT CREATE SESSION TO PRODLIB;
GRANT CREATE SEQUENCE TO PRODLIB;
GRANT CREATE TRIGGER TO PRODLIB;
GRANT CREATE VIEW TO PRODLIB;
GRANT CREATE TABLE TO PRODLIB;
```

Step 3: Modify DBTUNE

1. Export the dbtune file before making any modification:

```
sdedbtune -o export -f dbtune_exp.sde -u sde -p sde -i 5151
```

2. Copy *dbtune_exp.sde* to *dbtune_prodlib.sde*.

3. Modify the ##DEFAULTS configuration keywords:

```
dbtune_prodlib.sde
##DEFAULTS
A_INDEX_RASTER "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_AINDEX NOLOGGING"
A_INDEX_XML "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_AINDEX NOLOGGING"
D_INDEX_STATE_ROWID "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_DINDEX STORAGE (INITIAL 409600)
NOLOGGING"
D_INDEX_DELETED_AT "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_DINDEX STORAGE (INITIAL 409600)
NOLOGGING"
B_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_BDATA STORAGE (INITIAL 409600) "
B_INDEX_XML "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_BINDEX NOLOGGING"
B_INDEX_USER "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_BINDEX STORAGE (INITIAL 409600)
NOLOGGING"
B_INDEX_TO_DATE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_BINDEX NOLOGGING"
B_INDEX_SHAPE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_BINDEX NOLOGGING"
B_INDEX_ROWID "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_BINDEX STORAGE (INITIAL 409600)
NOLOGGING"
B_INDEX_RASTER "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_BINDEX NOLOGGING"
GEOMETRY_STORAGE "ST_GEOMETRY"
F_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_FDATA "
F_INDEX_LEN "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_FINDEX NOLOGGING"
F_INDEX_FID "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_FINDEX NOLOGGING"
F_INDEX_AREA "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_FINDEX NOLOGGING"
D_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_DDATA STORAGE (INITIAL 409600) "
ST_GEOM_LOB_STORAGE "STORE AS (TABLESPACE PRODLIB_FDATA ENABLE STORAGE IN ROW CHUNK 8K RETENTION
CACHE INDEX (TABLESPACE PRODLIB_FINDEX) )"
S_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_SDATA STORAGE (INITIAL 409600) "
S_INDEX_SP_FID "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_SINDEX NOLOGGING"
S_INDEX_ALL "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_SINDEX STORAGE (INITIAL 409600)
NOLOGGING"
A_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_ADATA STORAGE (INITIAL 409600) "
A_INDEX_USER "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_AINDEX STORAGE (INITIAL 409600)
NOLOGGING"
A_INDEX_ROWID "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_AINDEX STORAGE (INITIAL 409600)
NOLOGGING"
A_INDEX_SHAPE "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_AINDEX STORAGE (INITIAL 409600)
NOLOGGING"
A_INDEX_STATEID "PCTFREE 0 INITRANS 4 TABLESPACE PRODLIB_AINDEX STORAGE (INITIAL 409600)
NOLOGGING"
END
```

4. Import the modified *dbtune_prodlib.sde* file:

```
sdedbtune -o import -f dbtune_prodlib.sde -u sde -p sde -i 5151
```

Step 4: Create the Product Library Database Connection

Create a database connection in ArcCatalog™ with the PRODLIB user; this will be the product library workspace location.

Step 5: Configure Oracle Parameters

It is recommended that you ensure that the following parameter values are used when creating an Oracle database:

Parameter Name	Value
Configure with Database Enterprise Manager	Enabled
Automatic Memory Management	Enabled

Parameter Name	Value
Character Sets	Unicode (AL32UTF8)
OPEN_CURSORS	2000–10000
SESSION_CACHED_CURSORS	50–150
PRE_PAGE_SGA	False

Oracle parameters for product library

Step 6: Configure ArcSDE Parameters

You need to configure the MAXBLOBSIZE and TCPKEEPALIVE parameters for the ArcSDE geodatabase used as the product library. The MAXBLOBSIZE value is -1 by default. However, if you are using Oracle or another enterprise DBMS, make sure that this value is set to -1 and the TCPKEEPALIVE value is set to 1. This command should be used from the command prompt of a machine where ArcSDE is installed.

```
sdeconfig -o alter -v MAXBLOBSIZE=-1-i <service> -u sde -p <sde_password>
```

For more information, see the ArcSDE Administration Command Reference.

Step 7: Create the CKB_USERS Role

A role needs to be assigned to the users who are going to be working with the product library so they can view or add components, information, and data. The role CKB_USERS must be created for the users to be recognized by the product library. This can be done using the following statement:

```
CREATE ROLE CKB_USERS NOT IDENTIFIED
```

Step 8: Create the Product Library Workspace

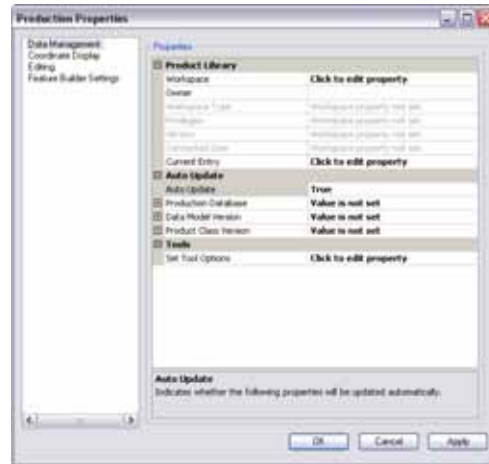
After the geodatabase has been created, various tables and feature classes that are part of the product library need to be added to it. This process can be completed in ArcMap™.

The steps in this section are for defining and upgrading the geodatabase as a product library in ArcMap.

1. Start ArcMap.
2. On the main menu, click **Customize > Production > Production Properties**.

The *Production Properties* dialog box appears.

3. If necessary, click **Data Management**.



4. Click the cell next to **Workspace** in the **Product Library** section and click the ellipsis (...) that appears.

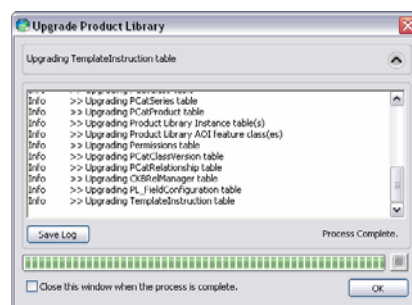
The *Please browse to the location of the Product Library Workspace* dialog box appears.

5. Navigate to the product library database.
6. Click **Open**.

The *Upgrade Workspace* dialog box appears.

7. Click **OK** to upgrade the geodatabase you want to use as the product library.

The *Upgrade Product Library* dialog box appears with the progress of the upgrade.



8. Click **OK** when the process completes.

The *Production Properties* dialog box appears.

9. Click **OK**.

J10023

Step 9: Verify the Storage

Run the SQL queries below to verify that the product library workspace was created under the correct tablespaces:

```
sqlplus prodlib/prodlib@prodlib
--TABLES--
SELECT TABLE_NAME, TABLESPACE_NAME, STATUS FROM USER_TABLES ORDER BY 1;
--INDEXES--
SELECT INDEX_NAME, TABLE_NAME, TABLESPACE_NAME, STATUS FROM USER_INDEXES ORDER BY 2,1;
--LOBS--
SELECT * FROM USER_LOBS WHERE SEGMENT_NAME LIKE 'SYS_LOB%' ORDER BY TABLE_NAME;
```

If any tables or indexes are stored in the wrong tablespace, ALTER TABLE and ALTER INDEX can be used to change the tablespace. See the SQL syntax.

```
ALTER TABLE <table_name> MOVE TABLESPACE <tablespace_name>;
ALTER INDEX <index_name> REBUILD TABLESPACE <tablespace_name>;
```

If moving large objects (LOB), please read Oracle MetaLink Doc ID: 130814.1, How to move LOB Data to Another Tablespace.

Step 10: Register as Versioned

You need to register product library components for versioning to allow them to be editable.

1. Start ArcCatalog.
2. Expand **Database Connections**.
3. Double-click the product library administrator connection geodatabase to connect to it.
4. Right-click each feature class in your product library and choose **Register as Versioned**.

Do not check the **Register the selected objects with the option to move edits to base** check box.

5. Click **OK**.
6. Right-click each table in your product library and choose **Register as Versioned**.

Do not check the **Register the selected objects with the option to move edits to base** check box.

7. Click **OK**.

Step 11: Validate Permissions and Roles

All the tables in the product library need to have read/write privileges assigned to them except the PCAT_PERMISSION table. The PCAT_PERMISSION table only needs read privileges assigned to it. The permissions need to be assigned to the CKB_USERS role. You can re-create the CKB_USERS role and grant the right permissions to the role by using the following script:

```
set echo off;
set verify off;
set heading off;
set feedback off;
set newpage none;
```

```
set termout off;
set lines 200;
set trims on;
ttitle off;
btitle off;
clear;
SET SERVEROUTPUT ON;
spool ROLE_CKB_USERS.sql;

select 'DROP ROLE "CKB_USERS";' from dual;
select 'CREATE ROLE "CKB_USERS" NOT IDENTIFIED;' from dual;

select 'grant select,insert,update,delete on ' ||owner|| '.' || table_name || ' to CKB_USERS;'
from sys.dba_tables where lower(owner) = 'prodlib' order by table_name;

select 'REVOKE INSERT,UPDATE,DELETE ON PRODLIB.PCAT_PERMISSION FROM CKB_USERS;' from dual;

spool off;
set echo off;
set verify off;
set heading off;
set feedback off;
set newpage none;
set termout off;
set lines 200;
set trims on;
ttitle off;
btitle off;
clear;

SET SERVEROUTPUT ON;
/
@ROLE_CKB_USERS.sql;
/
```

Grant Permissions Using ArcCatalog

Both the administrator and other user accounts in the underlying database management system should have appropriate privileges and roles assigned to them. When you set up your connection to your spatial database, ensure that you are connecting as the appropriate user.

1. Start ArcCatalog.
2. Expand **Database Connections**.
3. Double-click the product library administrator connection geodatabase to connect to it.
4. Select all tables except PCAT_PERMISSION, right-click, then click **Privileges**.
5. Type CKB_USERS into the text box on the **Privileges** dialog box.
6. Check the check boxes next to **SELECT**, **UPDATE**, **INSERT**, and **DELETE**.
7. Click **OK**.
8. Select the PCAT_PERMISSION table, right-click, then click **Privileges**.
9. Type CKB_USERS into the text box on the **Privileges** dialog box.
10. Check the check box next to **SELECT**.
11. Click **OK**.

J10023

Step 12: Configure Log File Tables

Enterprise geodatabases use log file tables to maintain lists of selected records. Records are written to log file tables for later use by the application whenever a selection of a specific size is made, a reconciliation or post on a versioned database is performed, or a disconnected editing checkout is done in a client application. The log file tables store the ObjectIDs of the selected features so they can be redisplayed. This allows faster analysis and processing of information.

In ArcGIS®, by default, log file tables are used if the selection set contains 100 or more records. This selection threshold of 100 features is set in the registry. It can be changed; however, Esri does not recommend doing so. There is no proven performance reason for changing it, and doing so could cause performance problems. Thus, log file tables store feature selections in ArcMap that are greater than 100 for each connected ArcSDE editor/viewer user. Hence, it is recommended that you store the log file tables in a separate tablespace; this can be achieved with the DBTUNE table.

Log file options are set using specific parameters in the SERVER_CONFIG and DBTUNE tables. Parameters in these tables are altered using the sdeconfig and sdedbtune commands, respectively.

Create Log File Tablespaces

```
CREATE SMALLFILE TABLESPACE SDELOGFILE
DATA FILE 'D:\oracle\ORADATA\PRODLIB\SDE\sdelogfile01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M MAXSIZE
800M
LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 512K SEGMENT SPACE MANAGEMENT AUTO;

CREATE SMALLFILE TABLESPACE SDELOGFILEIDX
DATA FILE 'D:\oracle\ORADATA\PRODLIB\SDE\sdelogfileidx01.dbf' SIZE 10M AUTOEXTEND ON NEXT 1M
MAXSIZE 400M LOGGING EXTENT MANAGEMENT LOCAL UNIFORM SIZE 512K SEGMENT SPACE MANAGEMENT AUTO;
```

Change DBTUNE Log File Parameters

1. Export the DBTUNE table:

```
sdedbtune -o export -f dbtune_logfile.sde -u sde -p sde -i 5151
```

2. Modify the dbtune_logfile.sde ##LOGFILE_DEFAULTS configuration keyword:

```
##LOGFILE_DEFAULTS
LD_INDEX_DATA_ID "PCTFREE 0 INITRANS 4 TABLESPACE SDELOGFILEIDX NOLOGGING "
LF_INDEXES "PCTFREE 0 INITRANS 4 TABLESPACE SDELOGFILEIDX NOLOGGING "
LF_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE SDELOGFILE"
SESSION_INDEX "PCTFREE 0 INITRANS 4 TABLESPACE SDELOGFILEIDX NOLOGGING "
SESSION_TEMP_TABLE 0
SESSION_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE SDELOGFILE"
LD_STORAGE "PCTFREE 0 INITRANS 4 TABLESPACE SDELOGFILE "
LD_INDEX_ROWID "PCTFREE 0 INITRANS 4 TABLE PACE SDELOGFILEIDX NOLOGGING "
END
```

3. Import the modified dbtune_logfile.sde:

```
sdedbtune -o export -f dbtune_logfile.sde -u sde -p sde -i 5151
```

Create Log File Tables

1. Grant QUOTA permissions to the users on SDELOGFILE and SDELOGFILEIDX:

```
ALTER USER SDE QUOTA UNLIMITED ON "SDELOGFILE";
ALTER USER SDE QUOTA UNLIMITED ON "SDELOGFILEIDX";

ALTER USER PRODLIB QUOTA UNLIMITED ON "SDELOGFILE";
ALTER USER PRODLIB QUOTA UNLIMITED ON "SDELOGFILEIDX";

ALTER USER PRODLIBUSER QUOTA UNLIMITED ON "SDELOGFILE";
ALTER USER PRODLIBUSER QUOTA UNLIMITED ON "SDELOGFILEIDX";
```

2. Grant CREATE TABLE permission for the ArcSDE editor/viewer user.

3. In ArcMap, select more than 100 features; this automatically creates the log file tables.
4. Remove CREATE TABLE permissions as appropriate.

Learn more about ArcSDE log file tables at help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/What_are_ArcSDE_log_file_tables/002n00000013000000/.

Step 13: Create the SDE Product Library User

The example below shows how to create an ArcSDE user to access the product library:

1. Create the PRODLIBUSER user:

```
CREATE USER PRODLIBUSER PROFILE "DEFAULT"  
IDENTIFIED BY mapeditor  
DEFAULT TABLESPACE "USERS"  
TEMPORARY TABLESPACE "TEMP" ACCOUNT UNLOCK;
```

2. Grant privileges:

```
GRANT "CONNECT" TO "PRODLIBUSER";  
GRANT CREATE TABLE TO "PRODLIBUSER";  
GRANT "CKB_USERS" TO "PRODLIBUSER";
```

3. Grant QUOTA on log file tablespaces:

```
ALTER USER PRODLIBUSER QUOTA UNLIMITED ON "SDELOGFILE";  
ALTER USER PRODLIBUSER QUOTA UNLIMITED ON "SDELOGFILEIDX";
```

Step 14: Database Connections for Product Library Users

Database connections need to be created for the other product library users if the product library is stored in Oracle. Create a database connection in ArcCatalog with the PRODLIBUSER user; this will be the product library sde connection. Note: If you are using database authentication, type the user name of the product library user for which you want to create a database connection.

Step 15: Product Library Permissions

There are two different levels of interaction with the product library in a Spatial Database Engine™ (SDE™) implementation of the product library: administrators and users. These levels of access are controlled through ArcCatalog database connections. The administrators manage the overall product library including the structure, components, and user permissions. This level of permissions through ArcCatalog database connections is related to the database role CKB_USERS.

Users can have varying degrees of access to parts of the product library based on whether they have edit, read/write, or read-only permissions based on their Windows® login. Using the administrator's database connection, user accounts are created for anyone who is going to have access to the product library. To create a new user, you must first add the person as a product library user, then assign permissions.

Learn more about product library permissions at help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Product_library_permissions/010300000043000000/.

Step 16: Add New Users to the Product Library

Using the administrator's database connection, user accounts are created for anyone who is going to have access to the product library. Create a new user according to instructions in step 15.

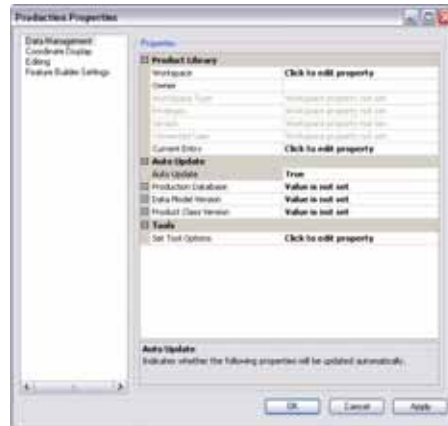
Initially, the user must be added to the geodatabase by the administrator. Each user is added using the first name, last name, and Windows user name.

J10023

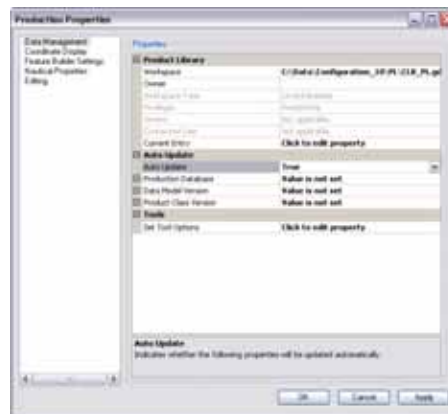
1. Start ArcMap.
2. On the main menu, click **Customize > Production > Production Properties**.

The *Production Properties* dialog box appears.

3. If necessary, click **Data Management**.

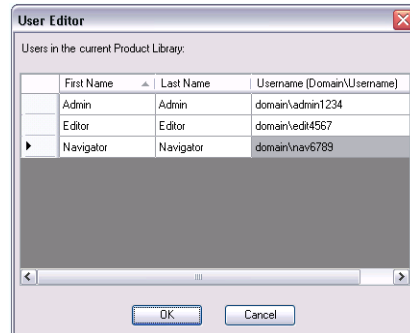


4. Click the cell next to **Set Tool Options**.



5. Click the drop-down arrow, then click the ellipsis (...) next to **Product Library Users**.

The *User Editor* dialog box appears.



Tip: If you are using an ArcSDE geodatabase as your product library, you can also add new users to the product library by right-clicking a series and clicking **Permissions**.

6. Right-click anywhere in the **Users in the current Product Library** list and choose **New User**.

A new row appears in the list.

Tip: If you are using an ArcSDE geodatabase as your product library, you can also add users when you are assigning permissions to existing users.

7. Type the user's first name in the **First Name** cell.
8. Type the user's last name in the **Last Name** cell.
9. Type the user's Windows login name in the **Users** cell.
10. Repeat steps 6–9 for each user you need to add to the product library.
11. Click **OK**.

The **Data Management** pane appears.

12. Click **OK**.

Assigning Permissions to Users

Once the user is added, the permissions can be granted at the series level of the product library. Permissions are passed down to all products within a given series. Permissions are also passed up from the series to the class and the solution. For example, if a user is given permission to one or more series below a particular class or solution, the user has access to those entries. By default, the permissions for a user are set to Not Available, but there are four different levels:


- Not Available—The series and all components beneath it are hidden from the user.
- Read Only—Properties can be viewed for all levels of the product library.
- Check In/Check Out—Files can be checked in and out.
- Edit—Product library levels can be added, modified, and removed, and files can be checked in and out.

J10023

Caution: This only applies if you are using an ArcSDE geodatabase as your product library.

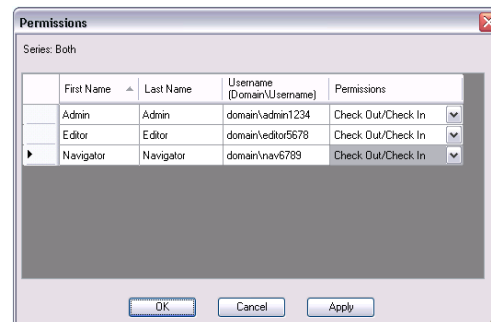
1. Start ArcMap.
2. On the menu bar, click **Customize > Production > Product Library**.

Tip: If **Product Library** is not enabled, you may need to enable the Production Mapping extension by clicking **Customize > Extensions** and checking the check box for **Production Mapping**.

Tip: You can also open the window by clicking the **Product Library Window** button  on the *Production Cartography* toolbar.

3. Navigate to the series level of the product class for which you want to assign permissions.
4. Right-click the series name and click **Permissions**.

The *Permissions* dialog box appears.



5. Click the **Permissions** drop-down arrow for the user to whom you want to grant permissions and choose an option.
6. Repeat step 5 for all users to whom you want to give permissions.
7. Click **OK**.

Conclusion

You can reduce disk contention and improve database I/O by storing the product library workspace in different locations on disk. However, this practice alone does not guarantee optimal database performance, and additional tuning tasks may be needed.

Learn more about the recommended tuning tasks at help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Minimize_disk_I_O_contention_in_Oracle/002n00000025000000/.

For more information on the product library, visit the Esri Production Mapping page at esri.com/epm.

To learn about setting up the product library in an ArcSDE environment, visit help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/Setting_up_the_product_library_in_an_ArcSDE_environment/010300000299000000/.

Access blogs, forums, downloads, and more via the Esri Production Mapping Resource Center at resources.arcgis.com/content/esri-production-mapping/10.0/about.

You can access other resources at ArcGIS Desktop 10 Help at help.arcgis.com/en/arcgisdesktop/10.0/help/index.html and Esri Support at support.esri.com.



About Esri

Since 1969, Esri has been helping organizations map and model our world. Esri's GIS software tools and methodologies enable these organizations to effectively analyze and manage their geographic information and make better decisions. They are supported by our experienced and knowledgeable staff and extensive network of business partners and international distributors.

A full-service GIS company, Esri supports the implementation of GIS technology on desktops, servers, online services, and mobile devices. These GIS solutions are flexible, customizable, and easy to use.

Our Focus

Esri software is used by hundreds of thousands of organizations that apply GIS to solve problems and make our world a better place to live. We pay close attention to our users to ensure they have the best tools possible to accomplish their missions. A comprehensive suite of training options offered worldwide helps our users fully leverage their GIS applications.

Esri is a socially conscious business, actively supporting organizations involved in education, conservation, sustainable development, and humanitarian affairs.

Contact Esri

1 800 GIS XPRT (1 800 447 9778)

T 909 793 2853

F 909 793 5953

info@esri.com

esri.com

Offices worldwide

esri.com/locations



380 New York Street
Redlands, California 92373-8100 USA