Managing & Migrating Multiple Instances of ArcSDE across an Enterprise

Bernard Traphan and Allen Dooley

Abstract

Organizing Raster and Vector data in multiple instances of ArcSDE presents a unique set of challenges and opportunities. The complexity of this task is exacerbated by the need to move this data between production and development servers. This paper will discuss how these challenges are being addressed at the Joint Information Operations Center (JIOC) to provide easy and efficient access to multiple TerraBytes of map data and imagery. ArcSDE 9.0 allows ArcSDE instances to be multi or single spatial. It is possible to break apart databases from a multi-spatial instance into single spatials. Possible does not usually equate to easy though as evidenced by problems we have seen with User SIDs and logins. The purpose of this presentation will be to present the procedures and SQL Server scripts that the JIOC has developed to successfully resolve these issues and be able to quickly move databases between servers and ArcSDE instances.

Background

The Joint Information Operations Center (JIOC) supports the US Combatant Commands (COCOMs)[1] by providing an Enterprise Architecture to store, categorize, and display Information Operation (IO) Data and Intelligence geospatially over map backgrounds. Our data is developed via an ArcGIS desktop, stored in a SQL Server database via ArcSDE, and delivered worldwide on SIPRNET and JWICS via an ArcIMS web server. Most of our underlying map data comes from the National Geospatial-Intelligence Agency (NGA), the United States Department of Defense primary provider and custodian of map products. Much of their rich library of paper maps has been digitized and made available as ADRG.

---

1 http://www.globalsecurity.org/military/agency/dod/unified-com.htm

There are currently about 11,000 ADRG CD-ROMs available, according to NGA's web site.[2] This table shows the number of ADRG products available at various scales.

Prefix	Map Type	Scale	# CDROMs
ARC4	GNC Global Navigation Chart	1:5,000,000	31
ARC3	JNC Jet Navigation Chart	1:2,000,000	124
ARC1	ONC Operational Navigation Chart	1:1,000,000	266
ARC2	TPC Tactical Pilotage Chart	1:500,000	642
ARC11	Russian General Staff Topographic Map	1:500,000	371
ARC5	JOG Joint Operations Graphic	1:250,000	1,937
ARC10	Russian General Staff Topographic Map	1:200,000	348
ARC6	TLM Topographic Line Map	1:100,000	792
ARC7	TLM Topographic Line Map	1:50,000	5,022
ARC12	Russian General Staff Topographic Map	1:50,000	100
ARC9	CG City Graphic	Varies (1:12,500 typical)	731
ARCN	ACO Hydrographic	Varies	356
ARCM	MIM Military Installation Map	Varies (1:25,000 typical)	55
ARCS	Special	Varies	16

Of these eleven thousand available CDROMs, the JIOC loaded approximately two thousand into a single ArcSDE 8.3 instance. These raster datasets were stored into approximately thirty country DB's (ex. Iraq) and incorporated into base map documents and ArcMap layer files that also include elevation rasters (primarily DTED[3]), imagery (primarily POID[4]), and vector datasets (primarily VMAP[5]). The "Information Operations Systems Division" (JIOC/J62) makes these map documents and layers available to the "Operations" (JIOC/J3) and "Plans and Policy" (JIOC/J5) Directorates for use in their analysis and presentation processes. Planners and analysts then add their own layers (ex. Communication Networks, Radar/RFMP coverage, etc.) to these documents as they work, creating derived documents that reference personal geodatabases in addition to the foundation data found in ArcSDE.

When the JIOC migrated to ArcSDE 9.0, based upon the recommendation from ESRI, our data was migrated away from the country DB's and into two databases Rasters (approximately 1 TB) and Vectors (approximately 8 GB). Initially, until we tested having our Vectors DB in a single spatial instance, we stored it on our development box in a multi spatial SDE instance. This paper will discuss the challenges we experienced with this architecture beginning with our attempts to migrate the Vectors DB from our development server to our production server.

---

[2] http://www.nga.ic.gov/products/adrg/adrg_pro.html

[3] DTED stands for Digital Terrain Elevation Data. It is available in multiple scales. DTED0 has 30 arcsecond pixel spacing. DTED1 has 3 arcsecond pixel spacing. DTED2 has 1 arcsecond pixel spacing.

[4] POID stands for Precise Ortho Image Database. POID imagery is provided by NGA in small patches, typically covering a city or urban area.

[5] VMAP stands for Vector Map. The VMAP format is described in a series concerning the Vector Product Format (VPF) which is related to the Raster Product Format (RPF) format.

Case Study

Hardware / Software Environment

Server1&2 (Production SQL Server - Primary and Failover node of a 4 node Active Passive Cluster):
Microsoft SQL Server 2000 with ArcSDE 9.0, Windows Server 2003, 8 GB RAM, two 3.06 GHz Xeon processors on a PowerEdge 1750

Server3 (Development SQL Server):
Microsoft SQL Server 2000 with ArcSDE 9.0, Windows Server 2003, 4 GB RAM, two 2.79 GHz Xeon processors on a PowerEdge 1800

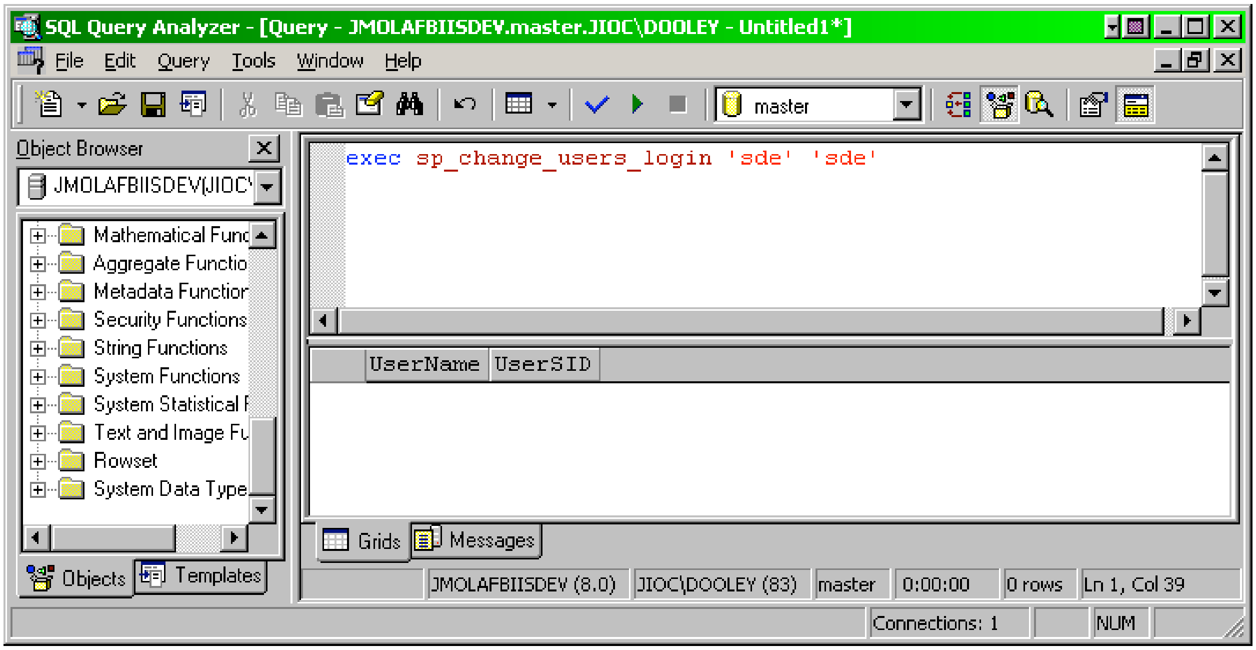
Server4 (Production Web Server):
Microsoft IIS 6.0? with ArcIMS 9.0, Windows Server 2003, ServletExec 4.1.1, 4 GB RAM, two 3.06 GHz Xeon processors on a PowerEdge 1750

Client: ArcGIS 9.0, Windows Workstation 2000, 1 GB RAM, one 2.53 GHz Pentium 4 processor on a Precision 340

The problems moving the Vectors DB from a multi-spatial instance on our development box to a single spatial instance on our production box

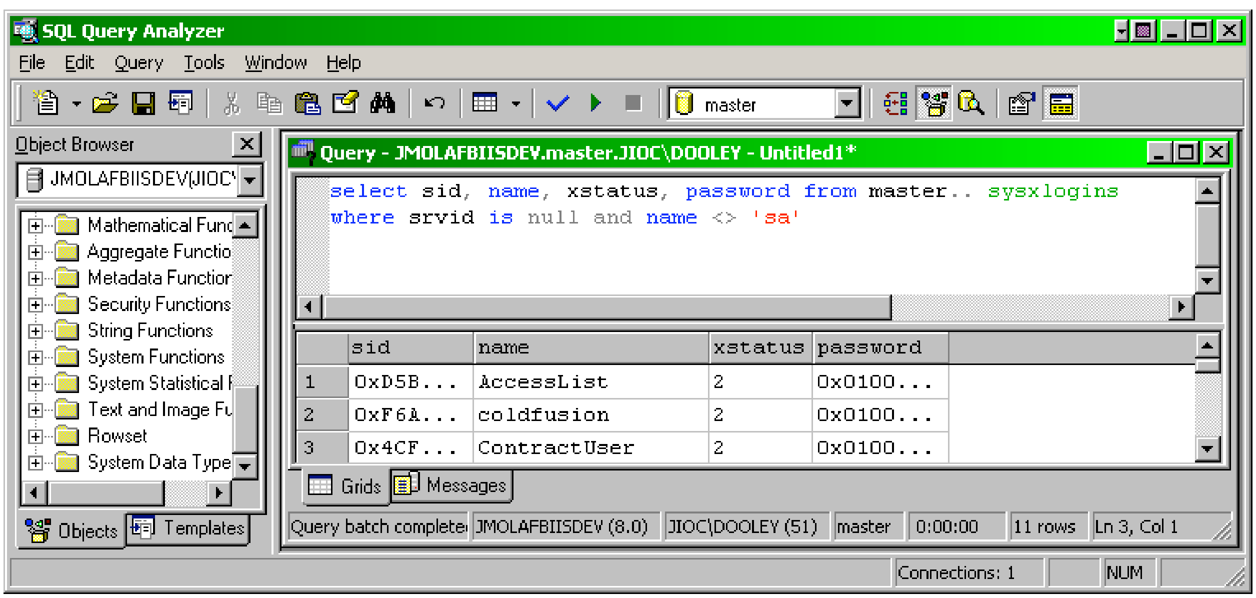
The machine hosting ArcSDE and the client reside inside the organization firewall. All database-related files reside on a Storage Area Network (SAN). The Web Server resides outside of the organization firewall in a public zone. Once the single spatial instance on the production box was tested utilizing a different port within ArcSDE and an ArcIMS site referencing that data, we began the process of moving the existing Vector data out of the multi-spatial instance on our development server over to the single spatial instance on the production server. We utilized the ESRI tool “migrator.exe” ESRI Migrator.exe to convert Vector database into single spatial database. Unfortunately, the database would not come online on the new machine.

SQL database move procedures:
Microsoft has a built in security features that embeds within the SQL database the unique system identification (SID). Our investigation revealed that the problem was related the user SID’s and logins not matching on the development vs. production server. We developed the following SQL scripts and procedures to resolve this problem. To fully understand this process you need to be able to manually do this process. If you are going to move or restore a SQL database, in this example then “vectors_data.mdf” and “vectors_log.ldf “ must be copied to a new MS SQL Server. Once they are moved to the new server then the database must be attached. Once you have the database attached you need to fix the security issues that come along with copying, moving and restoring an SQL database.

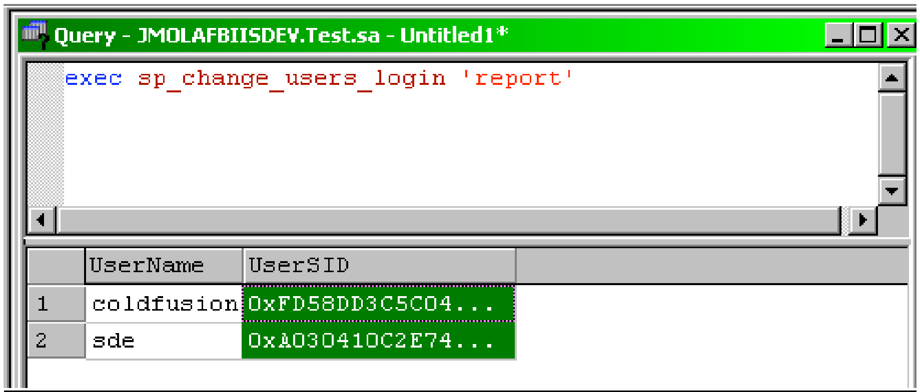


Security Issues:

Here is the command to check login and passwords on the original database server.
Type: (SQL Query Analyzer)
select sid, name, xstatus, password from master.. sysxlogins
where srvid is null and name <> 'sa'

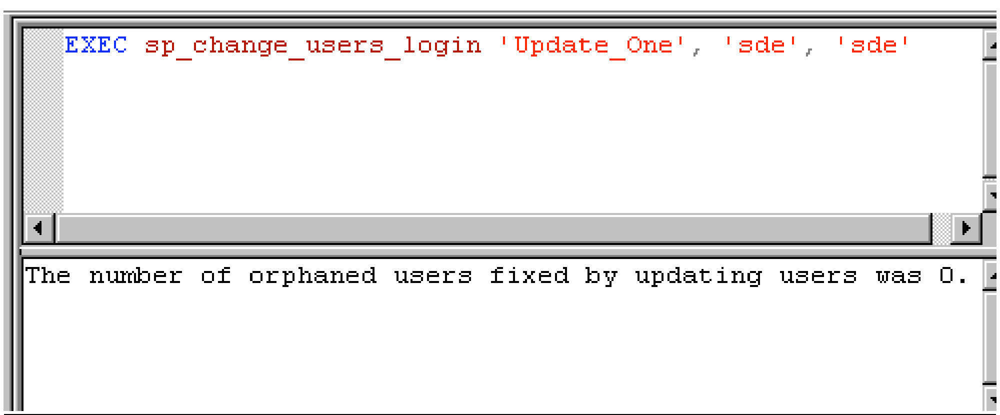


Check Output:
First, you must fix the SID and Login Names. Launch SQL Query Analyzer: login '*sa*' type this command: "*exec sp_change_user_login 'report'*" which will produce an output log showing mismatching or orphan login names associated with this database. The Orphan IDs are unique identifiers related to this database. You need to match all mismatching SIDs with the new ones and you may have to create this login if it doesn't already exist in Master database.



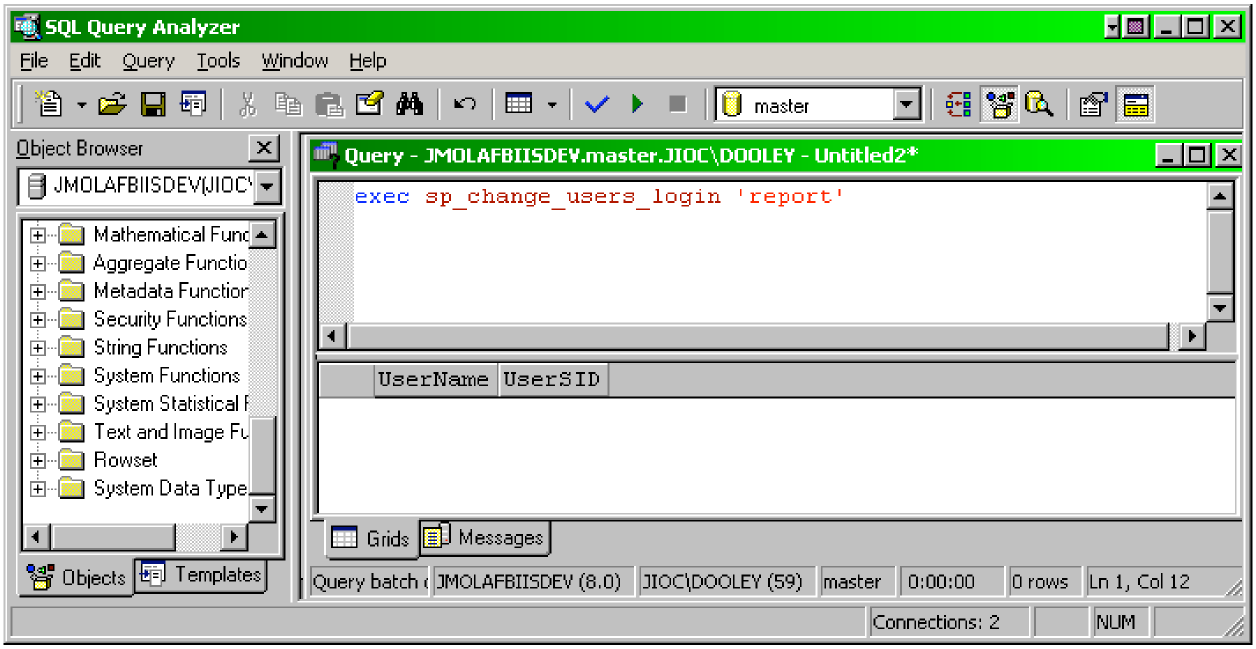
Fix
To resolve the mismatching SIDs you need to run this command to match user SIDs to the current database server.
Example: You need to correct the Orphan logins names to current login names based on the results of the command executed above and its resulting output.

Execute the following:
exec sp_change_user_login 'sde' 'sde'
This will fix the user SID problems.



Verify Fix

If the output screen isn't blank like it is below, then you have mismatching User SIDs. If you have any user name and user's SID you need to fix them. If they don't exist you need to create them. If they do exist you need to change users login to match this database.



The problems in performing backups on our DB's

Our tests of our backups revealed a similar problem. When the backup was restored to our development server it would not come online until the SQL scripts to resolve the problem with the User SID's and logins were run. On our Rasters DB we were experiencing additional problems in regards to backups. The DB had grown to be over 1 TB in size and we no longer had enough space on our SAN to perform a backup on it. Our only option was to back it up to tape, but this process took over 30 hours and was prone to not completing. Due to the increasing amount of SQL Server accounts that had write access to this DB and the amount of ArcIMS services that tied back to it, it could not be backed up unless ArcIMS and ArcSDE were turned off. This presented system availability problems for our user since the system is used worldwide 365/7/24 and as such turning it off every long weekend to do a backup was no longer an option.

The problems encountered connecting to a DB

We also experienced problems in connecting to our DB's too. The connection to our Rasters DB took on average 5 minutes and had over a 50 % chance of failing. The connection to our Vectors DB took on average 2 minutes and failed a third of the time. The Vectors DB was only 8 GB so the slow connection did not seem related to the file size of the DB. When presented with this fact it was suggested by ESRI to eliminate the use of feature datasets to increase access speed. However, if you do this then you can't use topology or geometric networks. Since vectors need to make use of these features, eliminating feature datasets wasn't a good option for our Vectors DB. The frustrating thing about when the connection failed was that the system wouldn't give you any indication that it failed. The only way to know it failed was to go into Task Manager and see that the ArcCatalog was not responding and to close it out.

Breaking up Rasters

By running tests with cross query filter turned off, it was determined that the primary cause of the slow connection was the amount of entities in the DB. It was decided to break up the Rasters DB into smaller DB's of about 200 GB each (see figure 1). This would have a secondary benefit of allowing the DB to be backed up in a more expedient manner. The World0R1 DB contained ESRI's World map, along with the four lowest resolution maps from NGA (i.e. GNC, JNC, ONC, & TPC). Higher resolution maps were placed into four regional DB's (Africa, Americas, EurAsia, & MiddleEast) which contained JOG, TLM, and City Graphics. Finally, a POID DB was created so as to segregate classified data from unclassified data. The new DB's performed extremely well and a connection could be made to them in less than 5 seconds with 100 % reliability. Additionally, since these new DB's were not having active edits done on them, backups could be done on an event driven basis with ArcIMS/ArcSDE turned on, and then the DB's could be put in ReadOnly mode.

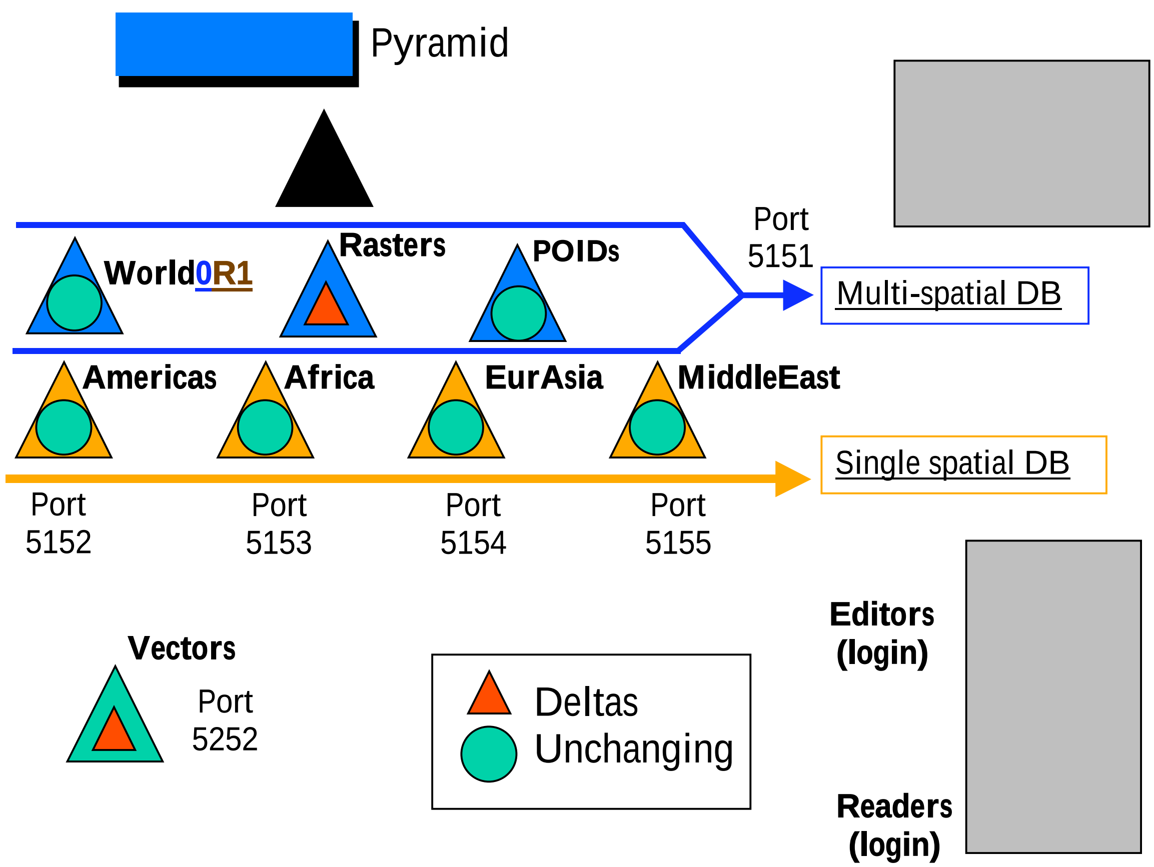


Figure 1 New Database Structure:

Cleaning up User Access

It was determined that the secondary cause of the slow connection to the Rasters DB was due to user contention (too many SQL accounts, the overhead of using SQL created roles rather than the built in SQL roles, and accounts with write access locking up the same data that other people were trying to read). This situation was primarily addressed by employing a Configuration Management triangle (see figure 2). The Rasters/Vectors DB on the production server would be where production data was stored. Users would work on data in their personal geodatabases. Users would post their data in the WorkingGIS DB on the development server. There it would get QC'ed before being posted in the Rasters/Vectors DB. SQL accounts would be reduced by using a standard GisWriter account for data writes and a GisUser account for data reads. SQL created roles (ex. GISroleEditors) were replaced by the built in SQL roles of db_datawriter and db_datareader. After these steps were taken a connection could be made to the Rasters DB in 10 seconds. Additionally, the Raster DB could be backed up in less than an hour with ArcSDE remaining on.

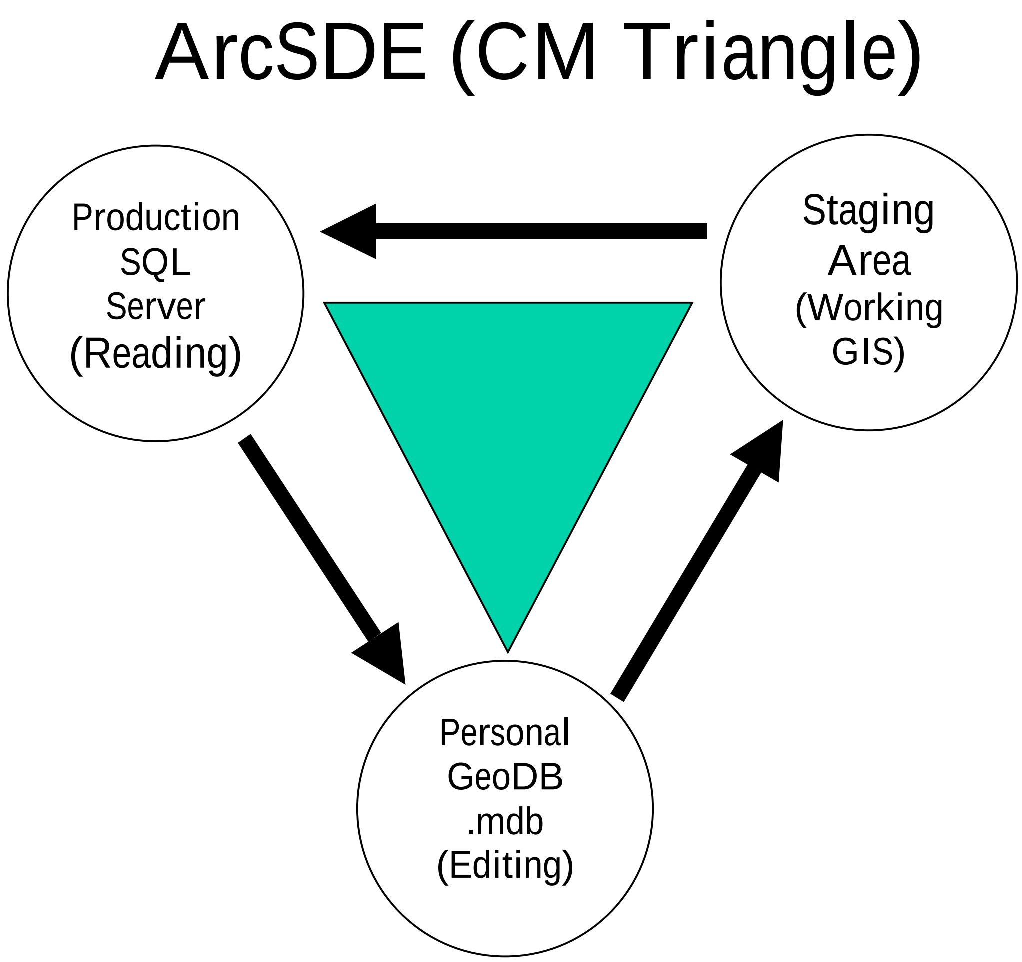


Figure 2 ArcSDE Configuration Triangle:

Performance Impact

Performance benefits were noticed across the board by cleaning up the way data was stored and accessed. ArcCatalog operations (ex. Loading/Viewing Data) performed dramatically faster due to reduced DB contention. DB backups performed faster as well. ArcMap no longer timed out while attempting to make the database connections for each of the layers. ArcIMS as well, stopped timing out while attempting to start new services. The only negative aspect we noticed of our new configuration was that since we had additional DB's we created additional connections to our SQL Server. This is due to the fact that for each unique username/DB name within a map there are additional connections. This situation is magnified within ArcIMS since it makes 4 (and sometimes due to ArcIMS inefficiencies up to 32) connections for each unique name combination. Further exacerbating this situation was that our users were continuing to add more data to the system. Some country maps ended up having over 100 layers. Combining this with the fact that we had 38 ArcIMS services on our production web server and a number more on our development server, all of which pointed back to data on our production SQL Server, we were at a breaking point. Our SQL Server started acting inconsistently. We determined the cause was the total amount of connections. It was at times approaching 500 connections and our studies showed it had degraded performance when the connections got to be over 300. We solved the problem in the short term by prioritizing ArcIMS services and turning off some that weren't needed. Our long term solution is to have a SQL Server dedicated to our Web Server and/or replacing ArcIMS with GIS Server.

Conclusion

The way data is stored and accessed will dramatically affect your performance within ArcGIS by a factor of as much as 100 to 1. This performance impact will be seen across the board from connecting to a DB, to performing a read or write within that DB, to backing up that DB. In summary, at the JIOC we have found that ArcGIS will perform much more efficiently by following these guidelines:

1) Only store data in the production DB that is utilized by your production maps
2) Organize your DB enterprise into manageable DB's to segment data (Unchanging vs. Changing, Unclassified vs. Classified)
3) Keep individual DB's size below 300 GB to allow for timely backup and restore times
4) Keep the number of entities in each DB small enough to allow for connection to that DB within ArcCatalog to be no more than 10 sec
5) Keep the number of feature classes within each feature dataset to be no more than 20
6) Employ a Configuration Management Triangle to separate editing and posting functions and reduce DB contention
7) Employ a standard SQL account for writing (ex. GisWriter) and reading (ex. GisUser) data from your DB to minimize the amount of SQL accounts tied to your DB and minimize the amount of connections each map makes to your SQL Server
8) Make use of built in SQL roles (ex. db_datawriter) instead of SQL created roles
9) Have no more than 50 layers in a map
10) Have no more than 30 services on an ArcIMS Server

Wish List

It would be very nice if the ESRI products would have a progress bar. This way a user could see that the software was still progressing. For example when opening a map with a large number of layers it can take as much as five minutes. However, when minute 4 hits you don't know if the map is about to open in a minute or that it has locked up and needs to be closed within Task Manager. The alternative to the progress bar would be an "Abandon All Hope" message that would be displayed when the program had locked up rather than the user needing to open Task Manager to determine this.

The suggestion from ESRI to use a Feature Class with Subtypes rather than Feature Datasets is not an acceptable alternative since Feature Datasets are the only way you can have Topology or a Geometric Network. It would be nice if one could have both Feature Datasets and fast access.

The cause behind the slow connection to ArcCatalog with an increase in entities needs to be determined. It is unacceptable for a connection to take 5 minutes and have a 50 % chance of failing, especially when within Enterprise Manager it can connect to the same DB and return all the tables or the rows within the table within seconds.

The cause behind the large amount of connections to the SQL Server needs to be determined. Even with connection sharing on we find that a single unique username/DB combination within ArcIMS can generate 32 connections to our SQL Server when it should only generate 4. What is the cause of this 8 to 1 inefficiency? Of additional concern is that if users open the same map within ArcMap then the connections get repeated again. It would be nice if ArcSDE could manage these connections and not allow redundant ones to be generated.

Author Information

Bernard Traphan
Joint Information Operations Center/J62
PO Box 769178
San Antonio, TX 78245
Phone: 210 414 5226
Fax: 888 678 7315
Email: Bernard.Traphan@jioc.osis.gov

Allen Dooley
Contractor, SAIC Information Operations Division
Joint Information Operations Center/J62
2 Hall Blvd, Suite 217
San Antonio, TX 78243-7008
Phone: 210 977 4705
Fax: 210 977 4021
Email: Allen.Dooley@jioc.osis.gov