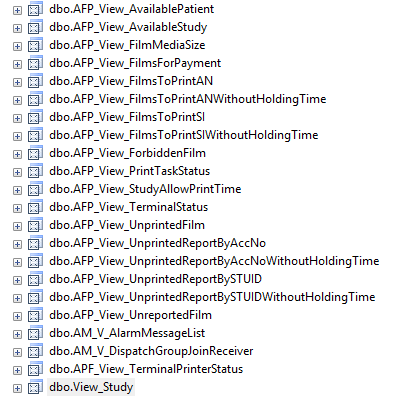
## Tune Suggestion

### System related:

1. The physical disks have some stress during testing, suggest enhance the DICOM and report archived and transfer process to avoid too much reads and writes.
2. Enhance the LRU and data deleted work rule. Make sure them can execute successfully with big data. Organize the tasks execute with sequential to avoid the recourse contend issues.
3. There are many application pools exist in the system. Some applications are not used in the system except the system which upgrade from history version. Please organize the services.
4. Limited the memory usage for IIS web service to save the memory resource.
5. The integration applications will use 3-4 web services in our system. They will use many memory resources. Normally, the integration will query data from the view from 3rd system. The view performance is slowly, suggest use table instead of view or materialization view.

### Database related:

1. Deleted the unused views from the database to save the CPU and memory resources.



1. The terminals query the patient information from the views in the WGGC database. The performance of view is slowly because it cannot use the index. The view is created with very complex condition and cost many memory resource. Please try to query data with table or limited the size with conditions.
2. Deleted the unused tables, functions, store procedures in the database to save the memory resource. Some of them are not used in PS system.
3. Some SQL used the tempdb tables in the database. It will cost 0.3-2MB/s IO recourse during testing. QA team suggests use the CTE instead of the temp tables to realize the functions. If the system uses more than 4MB/S IO reads/writes, it will cause the issues of main workflow.
4. Reduce the trigger usage in the database to enhance the performance for DB. The trigger performance is slow and cause the locks issue for our system.
5. Split the hot tables in our system. There are too many operations for our main work flow tables such as filminfo, reportinfo, examinfo, print task, print mode etc. The design will cause some locks issues.
6. The statistic function in web cannot works well. If user does this operation, it will block the main workflow in our system. Suggest create a table which include all information of patient to do the statistic operation on this table. Notice that the LRU service should back up the data to this table before they delete data. Add the redundancy tables for statistic functions and history query operations.
7. Use the sequenced column as the cluster key or primary key to avoid the latch issues. Many table cluster key is created by GUID, the GUID value is not a sequenced. IT will cause the index fragment issues. It will make the cluster query very slow and some tables` index fragment rate will more than 90%. It is big issues for our database.
8. There are still some key lookup issues in our database. The index design should be enhance.
9. Design the code rule for database such as case sensitive, avoid use \* to query all data, avoid use count (\*) instead of count (cluster), avoid use in and avoid use cursor etc.
10. The database full backup should include to LRU services. The operations should execute before the data clean operations in LRU.
11. Some SQL use too much reads or writes during this testing:

SQL1:

*update Study set AcquireStatus = '1'*

This sql will cause the full tables scans, please add some filter conditions for the sql statement like:

*update Study set AcquireStatus = '1' where AcuirStatus = ‘0’*

Add the index for the filter conditions.

SQL2:

*IF EXISTS(SELECT StudyInstanceUID FROM [Printer].[dbo].[DeliveryJob] D WHERE D.StudyInstanceUID = @SIUID)*

The column StudyInstanceUID is not indexed, the SQL will cause the full table scans. Please add the index on the column for table. Other issues for table *DeliveryJob* is that the column SessionInstanceUID is indexed in two indexes. Please delete one from the tables.

SQL3:

*SELECT JobInstanceUID, ForwardStatus = CASE WHEN ForwardStatus IS NULL THEN -1 ELSE ForwardStatus END,*

*ForwardFlag = CASE WHEN ForwardFlag IS NULL THEN -1 ELSE ForwardFlag END,ForwardDestination,ModalityType, CallingAE ,CallingIP,PatientName,PatientID,*

*PatientType = CASE WHEN PatientType IS NULL THEN '' ELSE PatientType END*

*FROM Printer.dbo.DeliveryJob J INNER JOIN printer.dbo.Session S*

*ON J.SessionInstanceUID = S.SessionInstanceUID*

*WHERE J.DeleteStatus = 0 AND (*

*(ForwardStatus = -1 OR ForwardStatus IS NULL ) OR*

*(ForwardStatus = 0 AND ForwardFlag = 1)*

*)*

The SQL will cause the key lookup issues because some columns are not including in the index. Suggest using “include” operations to create the index like :

*CREATE NONCLUSTERED INDEX IX\_Name ON table (index-column)*

*INCLUDE (column1, column2 which key lookup columns);*

*GO*

SQL4:

*INSERT INTO [Study]([StudyInstanceUID],[StudyDate],[StudyTime],[StudyID],[StudyDescription],[AccessionNo],[ReferPhysician],[Printed],[Readed],[PatientGUID],[SeriesCount],[ImageCount],[Reserved],[Hide],[OperateServerAE],[OperateStatus],[Problemed],[StudyDir],[AcqDateTime],[QCStatus],[Send],[AcquireStatus]) values(@1,@2,@3,@4,@5,@6,@7,@8,@9,@10,@11,@12,@13,@14,@15,@16,@17,@18,getdate(),@19,@20,@21)*

QA cannot find why insert sql will use many reads/write. Maybe the sql statement is included in a procedure or trigger. Other SQL caused the issues.