I thought it might be easier to bullet point the items Dave, Elizabeth and I have worked on in preparing ECM Library for full use.

* When David and I first started the installation for the ECM Repository and applications, we were initially under the impression the windows server and SQL Server would be version 2016. We at ECM were preparing for next year’s release to incorporate Windows and SQL Server 2019. It was a little bit of a surprise to find that the 2019 versions were already in place. So David and ECM Support had to scramble to bring ECM up to level to deal with 2019. This is not a complaint, this is just stating that we had to expedite development and testing to an unforeseen schedule. We accomplished this in record time, but it did require extended and unexpected effort on the part of David and ECM Support. We at ECM are also appreciative of the fact that it is done and in place significantly ahead of next year’s schedule.
* Decision to keep the previously populated repository
  + At the onset of the project, we discussed several options with regards to the data previously loaded into ECM Library. We knew that due to some of the missing updates, any decision would be a challenge. The choice was made to migrate the existing repository’s contents into the most current ECM repository.
* Data Migration
  + This proved to be a challenging approach. As we began the transfer of the old to the new, issues began to arise that required unique and individual solutions caused predominately by corrupted files, zipped files greater than 100GB, and poor and missing load structure in the existing repository.
  + We spent substantial time and effort to correct these issues with unique and individual solutions as the new ECM repository would not allow this data to be entered without correction.
* Data cleansing
  + The cleansing of the data both before and after migration was a challenge during the transfer between repositories.
  + And today, we are still finding unanticipated problems with the migrated. However, at this point in time the issues are invariably difficult to specifically identity and determine the error/errors being generated. These errors have been in the past few weeks, reflective errors that have been transmitted from the server across to the application. The most difficult to find, to date, and manifest itself as a cross memory violation issue on the server that hosts the ECM repository. It was found after many hours to be caused by file names that lived in the directory to be processed (the load directory) whose names exceeded 700 characters. When the name was transmitted from the holding workstation to the server, the file name breached the boundary of a fixed memory allocation based in the Microsoft Operating Systems’ mandated file name limitation of 260 bytes.
  + Similarly, we soon hit a similar problem when encountering existing directory names. The maximum length of a directory as defined by Microsoft is 254 byes. We encountered concatenated directory considerably greater than 1,000 bytes. Additionally, this issue, being a reflective error, generated itself in a different manor showing a different error and cause. Time and effort – that was how it too was solved.
* Size of previously uploaded Files
  + We found that some files were large, extremely large. In addition, these files were zipped containing upwards of a million files. Many, if not most of the files within these Zipped files were of no use as they were, in fact, files that were defined not to be loaded into the repository such as binaries, executables, ~tmp files, and a myriad of others that if loaded, would cause (and did cause) significant load errors as they placed disallowed files into the repository and had no existing indexing counterpart.
* Length of Directory Paths
  + David and I had to find a way to locate illegal directory names before they were attempted to process as they caused OS level errors and they cannot be captured and processed at the application level. Time and effort – we found a way.
* Length of File names
  + Precisely the same as above – T&M which translated to a lot of trial and error, republishing and then the master difficulty…
  + Debugging all of these problems remotely and NOT having the ability to attach to a remotely running process.
  + Dave and I literally had to resort to the debugging methods that were used in the 1980’s on the IBM-360s. Painfully slow and excruciating, but highly effective if you know how to do it.
* Proposed Microsoft fix to naming limitations
  + This is such an extreme problem that Microsoft is planning to implement a corrective method to the currently existing and very limited naming conventions contained within their existing operating systems that started in Windows 1.0 and exist in all Microsoft platforms to date.
* Unable to run the Archiver on the server containing the Repository
  + This was a bit of a challenge as it became necessary to rewrite parts of the archiver such that memory usage was drastically reduced and memorized
  + Because of memory optimization to the nth degree, we had to modify several processing class libs in an attempt to get more than 100 times the processing speed with less memory.   
    BE ADVISED, DAVE AND I DID THIS. No small task.
* Number of files contained within one single directory
  + We had not anticipated one single parent directory containing considerably more than 1,000,000 files and upwards of 2 terra bytes of data. It became necessary to re-architect the inventory algorithm. Dave and I did this
  + Rewrite all existing directory listeners to handle millions of files per directory. Dave and I did this

# Newly Discovered

* All file contents not being indexed
  + We are just now starting to work on this as Elizabeth found it two days ago. Certain documents are not being indexed and others of the same type are… this points directly to most likely being a data issue or potentially an iFilter issue.   
      
    Dave and I will start trouble shooting this tomorrow as I spent the weekend writing some new utilities that will help us zero in on the issue. The fact that this has happened in one or two documents indicates to us that it is happening on many more.
* iFilters
  + Microsoft has a release of iFilters that Dave and I are going to install.
  + WE WILL
    - Remove existing iFilters
    - Install new iFilters
    - Correct all identified missing document metadata (I wrote the utilities to find and replace if found to be missing.)
    - Rebuild all full-text indexes
  + ALL OF THESE processes take considerable run time due to the substantial quantity of data that must be analyzed and processed… one fix at a time.
* Next Year’s Alpha Release
  + With all of this unanticipated work in finding and correcting data and upgrading to 2019 versions of software, we are a jump ahead of next year’s work. Upon completion of this work effort and corrective measures, we feel that a good portion of next year’s release and update will already be in place and save considerable work and time at update time.

# PENDING WORK

The issues that David, Elizabeth and I have completed so far are all completed except for troubleshooting and correcting the newly found issue concerning all documents not being indexed. We, David and I, are starting on that issue today. With this completed, we will do one final check and validation of the data and if it does check out and index properly, ECM will be functional on your hardware and the 2019 software versions.

# Finally

In closing, let’s just say this has been the most interesting and most challenging upgrade I have ever participated in… especially with Covid taking out my two primary developers for ECM Library. It is more than fair to say that 99% of the problems Dave and I have solved have been either existing data issues, operating system level issues, updating all applications to run under 2019 software, and/or finding ways or methods to work with the existing file structure.

Last problem to solve and fix is that some documents are not being indexed when documents of an exact nature are… we are on it.

# NORMAL WORKFLOW

The normal work flow is:

* Problem detected
* Logs sent to ECM Support
* Support logs in remotely verifies and troubleshoots issue
* Support makes correction if coding issue or identifies bad data if data issues
* If coding issue, Support publishes the update to the ECM Portal
* Client opens the ECM Application and the corrections are automatically applied

# Modified Workflow

This is not a complaint whatsoever, it is an explanation. Because of the nature and sensitivity of DOD work, ECM support people are not allowed to remote into government workstations or servers. We believe that is a correct decision. Therefore, changes and fixes that normally take place in a day, can take up to a week or longer. It is a requirement for our support to work through a proxy DBA or Server administrator and troubleshoot issues while viewing a screen and talking the individual through necessary steps to locate and identify issues.

* Problem detected
* Call is made to ECM Support and a meeting time is arranged
* Logs sent to ECM Support
* Support logs into a secure meeting with the DOD Admin
* Admin works with Support to remotely verify and troubleshoot issue
* Support makes correction if coding issue or identifies bad data if data issues
* Support makes correction if coding issue or if needed, writes corrective SQL Code to remove or fix bad data
* Support compiles the updated code
  + Zips the compiled application
  + Uploads the compiled code to a government portal
  + The government admin downloads the code
  + Unzips and installs the corrections
* Admin contacts support and sets up a meeting
* Admin and support verify the change(s) are effective

This is the process we have developed and put in place so that we can maintain the highest degree of security without compromise. Having said that, the difference in time and resource required is significant. More often than not within areas that do not have or need strict security in place, the time required for the support is much, much shorter and the time and effort required on the part of the government Administrator goes from several days being required to a couple of hours.

Again, we fully and completely understand and support the need for extreme security in these cases but wanted to provide a full explanation of the unavoidable impact on the Proxy’s and support staff’s time – it is significant.