Control –

* Interval2.ini – intervals for each of the systems (60 = 60 seconds)
  + Subsystems do not need to be run as often
  + checkLogs – check to see if messages need to be ACKed, NAKed
  + this is where windows are hidden, by default
  + If the windowMode is shown in this file, remove it for security risks.
  + To remove an item, use ‘-1’ in the file
* sockServer.exe –
  + startupProcess in control.ini
    - startupProcess-“sockserver.exe ..\server\”

System Database – Changes to the database

* lambEDI.mdb in control
  + codemail list is found here
  + nak – not going to be used
  + receiving – equivalent to receiving.log
  + sending – equivalent to sending.log
    - resendAttemptsLeft – tells how many attempts are left
    - resendAttempts –
    - ackCode –
  + transactionCount – tells the message type, system Code, and the next message number
  + log – tells the errors within a system, the program within a system that caused an error, and what the causes are.
    - transactionNumber – is the message number within the system
    - processingTime – time within each process

New Comms – communications methods

* Old methods, when the logs grew, the system grew slower
* Priority of methods
  + Comms is used first (as long as comms works)
  + Sock.ini
    - Connection with a dedicated piece of software within our servers.
    - Encrypted
  + [ftp.ini](ftp://ftp.ini) – uses this site and username and password to grab a message
    - Will not see in the reporting.
    - Z:\harbor\lambEDI\sys\ -
      * Location in case a message does not go
* Automated methods for messages
  + Email is first, point to point (from codemail.lst)

Reporting –

* System knows what kind of data is
* Control/config/dbConfigs – All of our report info
  + dbConfigs/reports – system is pushing in defaults
  + dbConfigs/specs – specs that are stripped into one level for the database (translator step into the process)
  + dbMetadataTopMaps.ini – uses the type of EDI to figure out which one to use
  + dbDDSINV.ini
    - SpecialProcess – building a unique process for that specific type
    - INTERFACE, SEND, FULL, POST –the four stages of the transmission
    - Spec, ini file, and database in case you need to add something to it (most likely not)

Harbor Integration –

* Ddsinv.db
  + newUpdate – 0 (already sent to Harbor), 1 will run a process to create a report transaction, so it is pulled into Harbor
* New System Health
  + Will replace all the other existing reports (no ping report)
  + Prioritized
    - Disk Space is weighted (low disk space brings it to the top)