Git version: 1.4 Easier XML setup

Benchmarks full run on Core version 2.26

# All Scripts started at 2/11/2016 06:34:03

1. Running generate UAR runs in **18 minutes 55 seconds**
   1. Removing old messages 1 minute 9 seconds
   2. Importing models 5 minutes 17 seconds
   3. Analysing models 59 seconds
   4. Generating R, S, and Y messages 10 minutes 40 seconds
   5. Generating UAR file 43 seconds
2. Running loaducdata runs in **2 minutes 3 seconds**
3. Running DBupdate for all tables runs in
   1. Getting latest components 45 seconds
   2. Getting latest models 24 seconds
   3. Getting latest include procs 7 seconds
   4. Getting latest global procs 19 seconds
   5. Importing models 5 minutes 2 seconds
   6. Analysing models 2 minutes 13 seconds
   7. Generating R, S, and Y messages 16 minutes 30 seconds
   8. Importing components 30 minutes 2 seconds
   9. Compiling services
   10. Compiling forms (happens in parallel with services compiling)
4. Total elapsed time **2 hours 30 minutes**

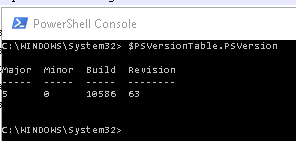
(This includes generate uar and loaducdata completeing in parallel)

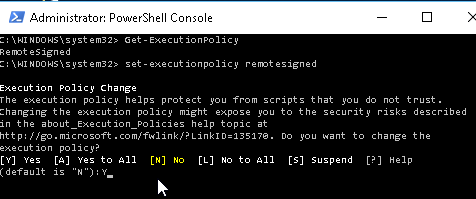
Be aware This is the worst case - picking just the Table(s) you have changed is much faster.

# All Scripts ended at 2/11/2016 08:51:47

Computer Install/Setup

1. Install recent Power shell (windows includes this but please check)
   1. To open power shell console
      1. Go to start menu and type power shell
      2. If it is not there then install it
   2. To see your power shell version number
      1. Open a power shell console and do this



* + 1. My Windows 10 box is version 5
  1. Now Get power shell to allow scripts to run on your box
     1. Open a **Admin** power shell console and do this
     2. 
     3. Yours will probably be set to restricted initially (You want remote signed)
     4. You just need to do this just one time at setup
  2. Ok, now you can run the scripts

1. Install SQL server 2014 together with all the tools
   1. Make sure to check the power shell support (not loaded down by the default install)
   2. If you have older SQL server on your box let do the upgrade to get the new stuff
      1. SQL server will look on your system when doing this and you might need to install service packs first
      2. Keep relaunching until your SQL server is updated
2. Install TFS Power Tools 2012 or later
   1. The tool we want is named tf.exe
      1. This tool allows TFS get and checkout and check-in support
      2. Once it is installed you will need to update its location in the script setup file
3. Install 7-ZIP tool
   1. You can just point to the 7za.exe Command has already put in place
   2. I copied this exe local to my box and pointed it location out in dbupdate.xml
4. Place Power shell scripts on your local drive
   1. TFS location of scripts is $/CSCE/IDF/Projects/Powershell\_DBUpdate
   2. Location of this document is $/CSCE/IDF/Projects/Powershell\_DBUpdate/doc
   3. Location of the C# supporting tool loaducdata.exe is $/CSCE/IDF/Projects/Powershell\_DBUpdate/cs
      1. Compile this tool in Release mode
      2. Move the EXE file local if you wish
5. Place batch file shortcuts on your desktop
6. Point your TFS messages USYS workspace on your local drive (cannot be a networked drive)
7. Set up your local directory structure
8. Update your dbupdate.xml file with your version and your local directory structure settings
9. Move custom dbupdate.xml to your local directory structure (if necessary)
   1. Copy dbupdate.xml from the power shell scripts location to the local Core ASN file location
      1. Dbupdate.xml will load from the standard location to find the version you want to run. It will then look in your local directory for the setting file
      2. If it is found there this will override the standard settings
   2. Example: When we come up under a newer Uniface version
   3. Example: you want a non-standard log location or to use a different Zip tool

Global XML Settings

|  |  |  |
| --- | --- | --- |
| **Tag** | **Meaning** | **Example** |
| UnifaceIDFPath | Uniface location of idf.exe | T:\UNIFACE\U9605\X505\common\BIN\idf.exe |
| TFSToolPath | Location of TFS power tools (need to install) | C:\Program Files (x86)\Microsoft Visual Studio 12.0\Common7\IDE\tf.exe |
| TFSIncludePath | Location of our include procs | H:\unicomp\CSCE\CS06\CS08.2.27\IncludeProcs |
| IncludeArgs | Source location for include procs | $/CSCE/CS06/CS08.2.27/USYS/IncludeProcs |
| ImportIncludes | Import includes switches | XML:H:\Unicomp\CSCE\CS06\CS08.2.27\IncludeProcs\\*.ipx |
| TFSModelPath | Location of our models | H:\unicomp\CSCE\CS06\CS08.2.27\Models |
| ModelArgs | Source location of models | $/CSCE/CS06/CS08.2.27/USYS/Models |
| ImportModels | Import model switches | XML:H:\Unicomp\CSCE\CS06\CS08.2.27\Models\\*.xml |
| TFSComponentPath | Location of components | H:\unicomp\CSCE\CS06\CS08.2.27\Components |
| ComponentArgs | Source location of models | $/CSCE/CS06/CS08.2.27/USYS/Components |
| ImportComponent | Import component switches | XML:H:\Unicomp\CSCE\CS06\CS08.2.27\Components |
| ASNCorePath | Local location of Core ASN file (I have this local) | D:\DBUpdate\Devo\_v2\ |
| INICorePath | Location of INI file (can be main) | /ini=P:\CS08\_2X\CS08\_2\_27\USERS\MAIN\idf96.ini |
| TempFileLocation | Local temp file dir | D:\DBUpdate\Temp\ |

RefreshCore Settings

|  |  |  |
| --- | --- | --- |
| **Tag** | **Meaning** | **Example** |
| LogPath | Location of the IDF log files | D:\DBUpdate\Devo\_v2\Logs\ |
| ModelPrompt | Prompt which shows at the top of the main script. To pick the tables to compile | DBUpdate script - Table(s) ordr,schl or (cr = all[slow]) |

LoadUCData Settings

|  |  |  |
| --- | --- | --- |
| **Tag** | **Meaning** | **Example** |
| ASNFileName | ASN filename | idf.asn |
| Tool | C# executable to create loaducdata file | D:\DBUpdate\Devo\_v2\LoadUCData.exe |
| LoadUCDataFile | Production location of loaducdata file | P:\CS08\_2X\CS08\_2\_27\UTILS\LoadUCData.sql |

GenerateUARFile Settings

|  |  |  |
| --- | --- | --- |
| **Tag** | **Meaning** | **Example** |
| TFSWorkspace | Local TFS workspace | d:\messages |
| MessageArgs | Source location of file | $/CSCE/CS06/CS08.2.27/USYS/messagesgenerated.uar |
| ASNMessagePath | Local ASN location | D:\DBUpdate\MessagesGenerated\_v2\ |
| INIGenerated | Local ini file | /ini=D:\DBUpdate\MessagesGenerated\_v2\idf96.ini |
| ResourcesGenerated | Local place to generated messages | D:\DBUpdate\MessagesGenerated\_v2\resources\msg |
| ZipLocation | Location of Zip tool | D:\DBUpdate\MessagesGenerated\_v2\7za.exe |
| LogPath | Location of the IDF log files | D:\DBUpdate\MessagesGenerated\_v2\Logs\ |
| SQLServer | SqlServer settings for drop table | (See belowTable) |

SQLServer Settings

|  |  |  |
| --- | --- | --- |
| **Tag** | **Meaning** | **Example** |
| Server |  | AL-SQL2K8R2-S01 |
| Database |  | Messages2xGenerated |
| User |  | ital |
| Password |  | ital |
| DropTableList | Tables to be dropped from the DB before regenerating the models and messages | uobj,ouobj,usource,ousource |

LoadUCData script Walkthrough

GenerateUARFile script Walkthrough

RefreshCore script Walkthrough

DBUpdate script Walkthrough

Future ideas for improvement

1. Better scanning of files to compile only what is necessary
2. Parallel compiling (Like all EDT\* forms at the same time as all RPT\* forms)
3. Scan logs when done to see model Analyze warning messages and form/service compile errors
4. Scan logs for certain Warnings that we do not want to see
5. Perhaps it can be faster to update a local resources directory when compiling and at the end do a copy / overwrite to the networked P drive resources folder? (may need to experiment)
   1. Add robocopy commands to see if I can speed up compiling
   2. Only refresh the newer files on the P Drive
6. Speed up the scripts as long as the tool stays accurate and consistent