**Electronic Logbook SW Design**

In SIVB and IMA SSDL, there are equipments, simulation participants, test work stations. They are connected with each other as below. Electronic Logbook is to get their configurations and save to a file.

…

Equipment n (HW, SW)

Equipment2 (HW, SW)

Equipment1 (HW, SW)

Test Work Station 2 (IO Gateway, A664, A429, A825, Analog, Ethernet) (Electronic Logbook Slave)

Test Work Station1 (Participants, 3rd party SW) (Electronic Logbook)

Test Work Station n (Participants, 3rd party SW) (Electronic Logbook Slave)

…

1. Top-Level Design

Electronic Logbook needs to collect 4 types of configurations:

1) Equipment Configuration: this will be sent out by equipment through ACR (Aircraft Configuration Report) message. Logbook subscribes to the ACR of each equipment and reads the configuration. If an equipment can’t send ACR, its configuration need to be input manually.

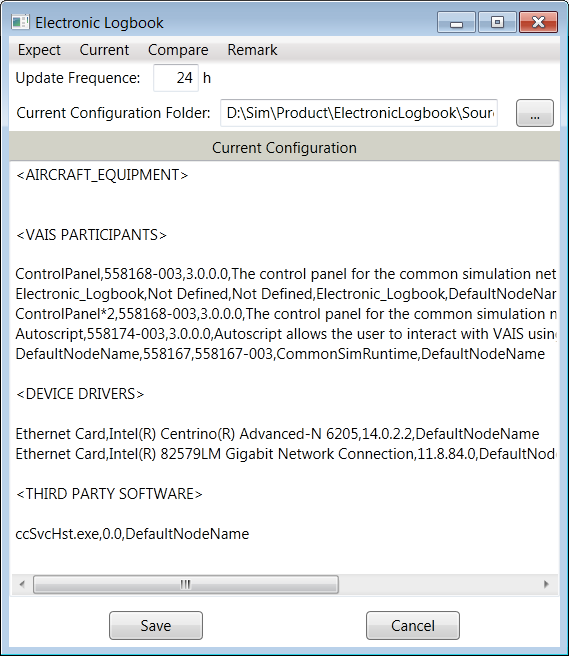
2) Participants Information: the information of all the participants is stored in CSI. So CSI APIs is used to get all participants information.

3) I/O Cards information: it is difficult for a remote computer to read the I/O card and driver information of another computer. So an “ElectronicLogbookSlave.exe” is put on each test work station. The “ElectronicLogbook.exe” can invoke each slave and then the slave begins to collect the I/O card and driver information automatically. The information will be sent to “ElectronicLogbook.exe” through CSI then.

4) 3rd Party SW on each Test Work Station: this information is collected as same as I/O cards information.

2. Details Design

2.1 GUI Design



2.2 Equipment Configuration

1. Let “ElectronicLogbook” be a CSI participant and connect it to CSI. This is done in function “App.RegisterParticipant()”.

2. ACR message is in ASN.1 format. So it is needed to build the ACR grammar tree first. Then the “ASN.1 Decoder” can use this grammar tree to parse an ACR message. This is done in class “ASN1\_Decoder\_ConfigReport”.

3. ACR message is wrapped in A664 and sent through a virtual link. So an A664 message definition is needed and the definition is used to get such message from ADN. This is done in class “A664ACRMessagePeriodicInput”.

4. When the equipment configuration is needed, “ElectronicLogbook” tries to get an ACR message from that equipment, this is done in “GetEquipmentConfig()”.

5. When an A664 ACR message is received, the grammar tree is used to parse it. This is done in “GetDecodedMessage()”.

2.3 Participants Information

Each participant runs on a test station. In order to connect to CSI, the test station needs to start the CSI runtime. Then all participants on that test station will register themselves to the runtime. “ElectronicLogbook” can get all the registered participants information through all the runtimes. This is done in “GetParticipantConfig()”.

2.4 I/O cards information and 3rd party software information on each test station

1. Use CSI API to iterator the VAIS runtime. For each runtime, it means there is a test station.
2. Each runtime subscribes to message “Runtime Name” + “.Startup”. Any participant can send such message to a runtime. With setting the parameter “ExecutablePath” in this message to the path of “ElectronicLogbookSlave.exe”, the “ElectronicLogbookSlave.exe” on the test station can be started.
3. The “ElectronicLogbookSlave.exe” uses the class GUID to query the card and driver information of Ethernet, A429, A664, A825, and Analog cards on the test station; and put them into a string.
4. The “ElectronicLogbookSlave.exe” takes a snapshot of the running processes on the test station. Then it puts all the processes names and versions into a string.
5. The “ElectronicLogbookSlave.exe” sends the strings back to “ElectronicLogbook.exe” through CSI.
6. The “ElectronicLogbook.exe” parses the strings and does some filtering according to configuration. Then it puts the result into “current configuration file”.

2.5 Compare “Expected Configuration” with “Current Configuration”

The “Beyond Compare” is used to compare the “Expected Configuration File” and “Current Configuration File”. And the result is shown in “Beyond Compare” also.

2.6 Configuration Files

1. ExpectConfig.txt: this file contains the expected configuration of all the equipments and test stations. User can change it manually.
2. Remarks.txt: this file contains users comments, history records and etc. User edits it from the GUI and saves it to this file.
3. ElectronicLogbook\_Config.txt: in this file, user can set 4 types settings:
   1. Last Setting: the folder used to put the “Current Configuration” files last time; the auto save frequency setting last time; and the “Current Configuration” file saved last time.
   2. 3rd party Software Check List: user can list all the interested 3rd party Software to check on each test station. All other 3rd party Software not on this list won’t be shown in “Current Configuration”.
   3. Equipment Definition: User can list all the equipments in the SSDL/SIVB, and the ACR message virtual link for each equipment.
4. Current Configuration files: when user saves the current configuration to a file, a new “Current Configuration” file is generated with name as “Station\_Configuration\_YearMonthDay\_HourMinuteSecond.txt”.