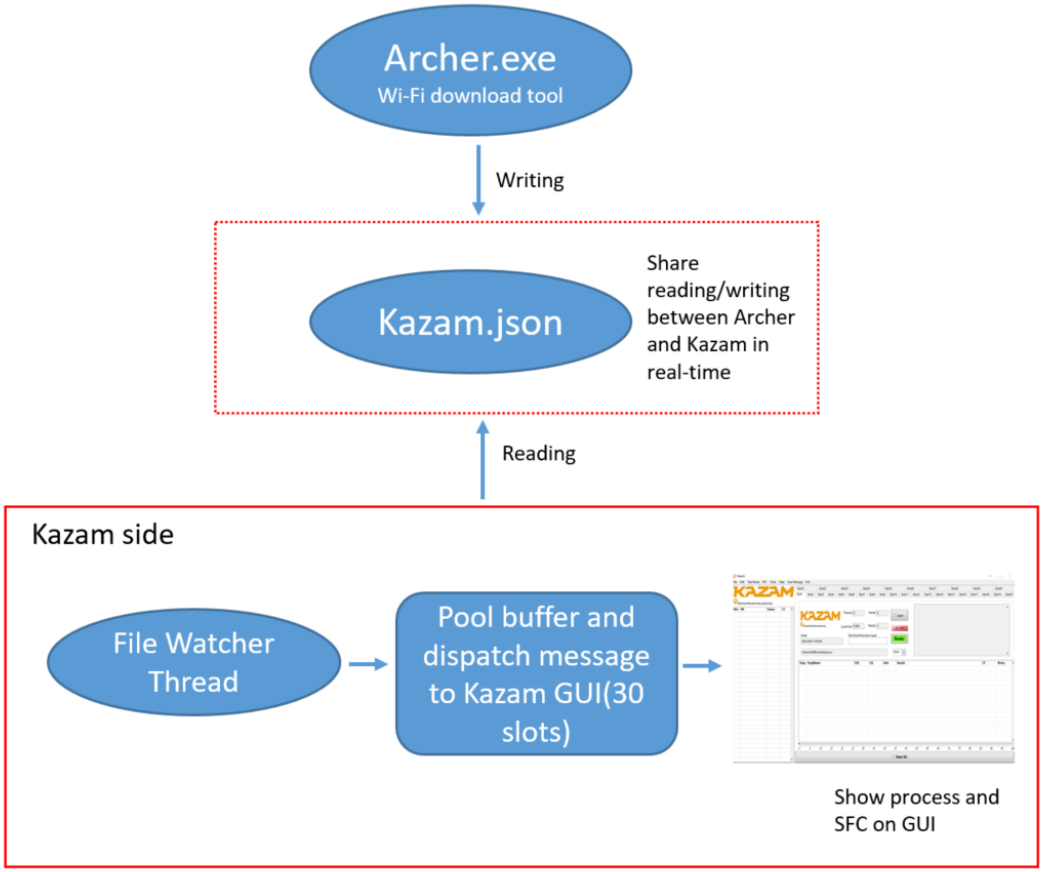
Wi-Fi Download integration on Kazam

1. Purpose

   Wi-Fi download means downloading the user image in production line via OTA, utilizing the AP to broadcast the user image and have up to 30 devices downloading the image in parallel by connecting with AP via Wi-Fi. Wi-Fi download can save the manufacture cost in factory, e.g. Saving the operator and BON fixtures. Under Wi-Fi download only need 1 operator per line and there is no BON fixture. Diag team worked out Archer.exe tool that can run Wi-Fi download, but this tool is not included Shop Floor connection, and there is no test data recorded in AME data hub, and miss test status tracking during xVT and MP. So there is requirement to integrate Wi-Fi download into Kazam and connect shop floor and upload test data to AME data hub.

2. Wi-Fi Download integration structure

   - Archer and Kazam are independent processes running on the PC. The intersection of them is the Kazam.json file. It is sharing to read/write between Archer and Kazam
   - When Archer is running for Wi-Fi download process, it will write the process log into Kazam.json
   - Meanwhile there is file watcher thread existing in Kazam, it is monitoring the changes of Kazam.json. Once detect the file change, it will read Kazam.json and parse the contents, then push them into dispatcher pool buffer, the separate dispatcher thread processes them and dispatch them to 30 slots of Kazam GUI.



3. Installation
   - Install .Net Framework 4.5.2 or above version.
   - Kazam version v4.51 or above
   - Labview runtime lib.
4. Kazam + Wi-Fi download setup

   4.1. Kazam shell_factory.csv setting
   - Display GUI use “GUI_MultiSingle_ForWifi.vi”
   - Add auto start vi in shell factory setting, “WiFiSWDL_Autostart.vi”

CheckValu	8d31fe7cffad42f94e6cf9832e4136be		
	RELEASE_I	UTDP=0.01	
SEQ	Resource_	VI_NAME	CONTROL_STRING
	Sequence_	sys_SequenceCFGReadout.vi	TestPlan=DemoWifiDownload.csv|WaitUntilDone=True|EnableParameter=HVT;REL;
	Sequencer	sys_SequenceLauncher.vi	Slots=30|SeqEngineName=TestSequence.vi|
1	Display	GUI_MultiSingle_ForWifi.vi	FP_Show=True|RefreshTime=5000|Real-time=0|SlotPosition=1
	Datalog	amz_log_Daily.vi	
	Datalog	amz_log_single.vi	
	Datalog	amz_log_timelog.vi	
	Datalog	WiFiSWDL_Autostart.vi	2
EOF			

   4.2. Kazam test sequence modification
   - Implement the SFC control portion(upload the test results etc.) into Kazam test sequence(DemoWifiDownload.csv)
   - Modify the directory of archer.exe in test sequence with the real path in the factory.

Test Name	Input Parameter
Start_WiFiSWDL_App	Function=Start_App|DEBUG=OFF|AppHidden=ON|AppPathName=C:\Work\WiFi Download\bin\Debug\archer.exe| AppParameter=-i ".\log\kazam.json" -l 1 -u 20 -p 3

1

DemoWifiDownload.csv

   4.3. Implement ksfc.exe for Wi-Fi download Archer.exe, the requirement as below, and put it into bin directory of Archer.exe work directory. The return value of ksfc.exe is exit code.

Communication with SFC

Archer tool won’t communicate with SFC, so devices need to get registering status from kazam. Archer will call one tool “**ksfc.exe**”, which is supposed under **bin** directory of archer work directory.
It will call in the following way:

```
$bin\ksfc.exe dsn
//dsn represents device serial number
//return value:
//        -1 or 1: register failed
//        0: register succeed
```