

```
=====
Visteon Vendor Integration with QSSI - Full Guide
=====
```

This document explains how to integrate Visteon-provided components (AIDL HAL, JAR manager, and native service) into QSSI. All code is structured and works in an AOSP environment with minimal modification.

```
=====
Folder Structure
=====
aosp/
    device/<oem>/<device>/device.mk
    packages/apps/QssiWifiClient/
        Android.bp
        src/com/qssi/client/QssiWifiClient.java
    -> vendor/visteon - symlink to aidls and jar file
    vendor/visteon/
        wifihostapd/
            aidl/
                IWifiHostapd.aidl
                Android.bp
            cpp/
                hostapd_service.cpp
                Android.bp
                hostapd.rc
            vist_wifi_manager/
                VistWifiManager.java
            vist_wifi_manager.jar
        Android.bp
```

```
=====
1. AIDL Service Integration (Vendor Binder)
=====
```

```
IWifiHostapd.aidl: -----
-
package vendor.visteon.wifihostapd;
interface IWifiHostapd { void startHostapd(String config); }
```

```
Android.bp: -----
aidl_interface {
    name: "vendor.visteon.wifihostapd",
    srcs: ["IWifiHostapd.aidl"],
    vendor_available: true, stability: "vintf",
    backend: {
        java: { enabled: true },
        ndk: { enabled: true },
        cpp: { enabled: true }, },
    versions: ["1"], }
```

```
hostapd_service.cpp: -----
-
#include <android/binder_manager.h>
#include <android/binder_process.h>
#include <android/log.h> #include "aidl/vendor/visteon/wifihostapd/BnWifiHostapd.h"

#define LOG_TAG "VisteonHostapdService" #define LOGI(...) \
__android_log_print(ANDROID_LOG_INFO, LOG_TAG, __VA_ARGS__)

using namespace aidl::vendor::visteon::wifihostapd;

class WifiHostapdImpl : public BnWifiHostapd {
public:
    ndk::ScopedAStatus startHostapd(const std::string& config) override {
        LOGI("Hostapd started with config: %s", config.c_str()); return
        ndk::ScopedAStatus::ok(); } }

    int main() { ABinderProcess_setThreadPoolMaxThreadCount(1); std::shared_ptr<WifiHostapdImpl>
        service = ndk::SharedRefBase::make<WifiHostapdImpl>(); binder_status_t status =
        AServiceManager_addService(service->asBinder().get(), "visteon.wifihostapd"); if (status != STATUS_OK)
        { LOGI("Failed to register Visteon wifihostapd service."); return 1; }
        LOGI("Visteon wifihostapd service registered successfully.");
        ABinderProcess_joinThreadPool(); }
```

```
Android.bp (cpp): -----
cc_binary { name:
```

```
"hostapd_service",      srcs:  
["hostapd_service.cpp"],  
shared_libs: [  
"libbinder_ndk",  
    "vendor.visteon.wifihostapd-V1-ndk_platform",  
,     init_rc: ["hostapd.rc"],     vendor: true, }  
  
hostapd.rc: -----service hostapd_service  
/vendor/bin/hostapd_service      class late_start  
user system      group system      oneshot  
  
=====  
2. Vendor JAR Usage (VistWifiManager)  
=====  
  
VistWifiManager.java: -----  
-----package  
com.visteon.wifi;  
  
public class VistWifiManager {  
    public static void init(android.content.Context ctx) {  
        System.out.println("VistWifiManager initialized");  
    }  
    public void startSoftAp() {  
        System.out.println("SoftAP started from VistWifiManager");  
    }  
}  
  
Android.bp: -----  
java_prebuilt_library {      name:  
"vist_wifi_manager",      jars:  
["vist_wifi_manager.jar"],  
installable: true,      visibility:  
["//visibility:public"], }  
  
=====  
Symlink usage for linking aidls and .jar  
  
#!/bin/bash  
  
ln -s  
/path/to/vendor/visteon/hostapd/IWifiHos  
tapd.aidl vendor/visteon/  
  
ln -s  
/path/to/vendor/visteon/wifi_service/Vis  
teonManager.jar vendor/visteon/  
  
=====  
=====  
3. QSSI Java Client Using the Manager  
=====  
  
Android.bp: -----java_library {      name:  
"qssi_wifi_client",      srcs:  
["src/com/qssi/client/QssiWifiClient.java"],  
static_libs: ["vist_wifi_manager"],      sdk_version:  
"system_current", }  
  
QssiWifiClient.java: -----  
package com.qssi.client; import  
com.visteon.wifi.VistWifiManager;  
  
public class QssiWifiClient {      public static void  
main(String[] args) {          System.out.println("QSSI  
Client starting...");  
        VistWifiManager.init(null);          VistWifiManager manager  
= new VistWifiManager();          manager.startSoftAp();  
System.out.println("QSSI Client completed WiFi operation.");  
    }  
}  
  
=====  
4. C++ Client in QSSI Accessing AIDL HAL  
=====  
  
Add shared_libs : ["vendor.visteon.wifi"]  
  
Static_libs: ["libbinder_ndk"]  
  
in .bp file
```

2025-05-26 09:13:27 UTC
Igunasek@visteon.com

```
#include <vendor/visteon/wifihostapd/IWifiHostapd.h>

#include <android/binder_manager.h>

int main() {
    auto binder = ndk::SpAIBinder(AServiceManager_getService("visteon.wifihostapd"));
    auto service = aidl::vendor::visteon::wifihostapd::IWifiHostapd::fromBinder(binder);
    if (service) {           service->startHostapd("config.json");       }
}

=====
5. device.mk Setup in QSSI
=====

device.mk: -----
and also in
boardconfig.mk ----

PRODUCT_SOONG_NAMESPACES += vendor/Visteon/
```

Qualcomm

2025-05-26 09:13:27 UTC
Igunasek@visteon.com