Sierra Wireless
Android RIL
Release Notes

IceCreamSandwich_RELEASE

Revision:	V3.2.18.0

Table of Contents

1 General

1.1 Purpose

This document describes the Sierra Wireless Android IceCreamSandwich RIL Release.

1.2 Summary of Document Revision History

Date	Author	Summary of Changes
Oct 18 2013	Stanley Ng	Initial version for V3.0.0.0
Nov 8 2013	Stanley Ng	Update for V3.0.0.1
Nov 22 2013	Stanley Ng	Update for V3.0.0.2
Dec 6 2013	Stanley Ng	Update for V3.0.0.3
Dec 27 2013	Stanley Ng	Update for V3.0.0.4
Jan 17 2014	Stanley Ng	Update for V3.0.0.6
Jan 30 2014	Stanley Ng	Update for V3.1.0.0
3 Mar 2014	Stanley Ng	Update for V3.1.2.0
14 Mar 2014	Stanley Ng	Update for V3.1.3.0
28 Mar 2014	Stanley Ng	Update for V3.2.0.0
11 Apr 2014	Stanley Ng	Update for V3.2.1.0
2 May 2014	Stanley Ng	Update for V3.2.2.0
16 May 2014	Stanley Ng	Update for V3.2.3.0
3 Jun 2014	Stanley Ng	Update for V3.2.5.0
13 Jun 2014	Stanley Ng	Update for V3.2.6.0
4 July 2014	Stanley Ng	Update for V3.2.8.0
17 July 2014	Stanley Ng	Update for V3.2.9.0
8 Sept 2014	Leo Lee	Update for V3.2.11.0
15 Jan 2015	Stanley Ng	Update for V3.2.18.0

2 Supported Configurations

2.1 Release Configurations Supported

These release notes apply to the following release configurations:

AndroidRIL_V3.2.18.0_ARM_4.0_V1.0_bin
Android RIL version 3.2.18.0
Supports ARM based processors, armv7-a variant and higher
For Android IceCream Sandwich (4.0)
Binary only release
and

AndroidRIL_V3.2.18.0_ARM_4.0_V1.0_src
Android RIL version 3.2.18.0
Supports ARM based processors, armv7-a variant and higher
For Android IceCream Sandwich (4.0)
Source release

2.2 Sierra Modules Supported

The Android RIL supports the following devices and firmware versions:

- MC8305 (G3K) - firmware version 1580 and 3.3
- MC8355 (G3K) - firmware version 1580 and 3.3
- MC7700 - firmware version SWI9200X_03.05.16.01 (or greater)
- MC7710 - firmware version SWI9200X_03.05.16.01 (or greater)
- MC7750 - firmware version SWI9600M_03.05.10.05 (or greater)
- SL909x - firmware version 331403 (and greater)
- SL809x - firmware version P1_0_0_18AP (and greater)

The MC7700 and MC7710 modules can be configured to operate in “QMI Mode” or “Direct-IP” mode. For this RIL to operate properly, the devices must be in “QMI Mode”.

MC7750, MC8305, MC8355 and SL909x only support QMI mode so they will work by default.

SL809x can be configured to operate in “PPP” mode or “Direct-IP” mode. For RIL to operate properly, the devices must be in Direct-IP mode.
If you are unsure if your module has correct configuration, please contact your account rep for support.

3 Supported Android RIL APIs

The table below lists the Android RIL APIs that are supported in the IceCreamSandwich RIL.

Android APIs	Notes
Device Information Requests	
RIL_REQUEST_GET_IMSI	
RIL_REQUEST_DEVICE_IDENTITY	
RIL_REQUEST_BASEBAND_VERSION	
Data Call Requests	
RIL_REQUEST_SETUP_DATA_CALL	
RIL_REQUEST_DEACTIVATE_DATA_CALL	
RIL_REQUEST_DATA_CALL_LIST	
RIL_REQUEST_LAST_DATA_CALL_FAIL_CAUSE	
Network Related Requests	
RIL_REQUEST_SIGNAL_STRENGTH	
RIL_REQUEST_RADIO_POWER	
RIL_REQUEST_OPERATOR	
RIL_REQUEST_SET_BAND_MODE	
RIL_REQUEST_QUERY_AVAILABLE_BAND_MODE	
RIL_REQUEST_SET_NETWORK_SELECTION_AUTOMATIC	
RIL_REQUEST_SET_NETWORK_SELECTION_MANUAL	
RIL_REQUEST_QUERY_NETWORK_SELECTION_MODE	
RIL_REQUEST_QUERY_AVAILABLE_NETWORKS	
RIL_REQUEST_SET_PREFERRED_NETWORK_TYPE	LTE is not yet supported
RIL_REQUEST_GET_PREFERRED_NETWORK_TYPE	LTE is not yet supported
RIL_REQUEST_VOICE_REGISTRATION_STATE	
RIL_REQUEST_DATA_REGISTRATION_STATE	
RIL_REQUEST_SET_LOCATION_UPDATES	
RIL_REQUEST_GET_NEIGHBORING_CELL_IDS	Partial Support for 3G only AT RIL only
SIM Related Requests	

RIL_REQUEST_SIM_IO	
RIL_REQUEST_GET_SIM_STATUS	
RIL_REQUEST_ENTER_SIM_PIN	
RIL_REQUEST_ENTER_SIM_PUK	
RIL_REQUEST_ENTER_SIM_PIN2	
RIL_REQUEST_ENTER_SIM_PUK2	
RIL_REQUEST_CHANGE_SIM_PIN	
RIL_REQUEST_CHANGE_SIM_PIN2	
RIL_REQUEST_QUERY_FACILITY_LOCK	
RIL_REQUEST_SET_FACILITY_LOCK	
RIL_REQUEST_ENTER_NETWORK_DEPERSONALIZATION	
SMS Requests	
RIL_REQUEST_SEND_SMS	
RIL_REQUEST_SMS_ACKNOWLEDGE	
RIL_REQUEST_CDMA_SEND_SMS	QMI RIL only
RIL_REQUEST_CDMA_SMS_ACKNOWLEDGE	QMI RIL only
RIL_REQUEST_WRITE_SMS_TO_SIM	
RIL_REQUEST_DELETE_SMS_ON_SIM	
RIL_REQUEST_GET_SMSC_ADDRESS	
RIL_REQUEST_SET_SMSC_ADDRESS	
CDMA Specific Requests	
RIL_REQUEST_CDMA_SUBSCRIPTION	QMI RIL only
RIL_REQUEST_CDMA_SET_SUBSCRIPTION_SOURCE	Hard-coded to always return success. QMI RIL only
RIL_REQUEST_CDMA_QUERY_ROAMING_PREFERENCE	QMI RIL only
RIL_REQUEST_CDMA_SET_ROAMING_PREFERENCE	QMI RIL only
Voice Call Related Requests	
RIL_REQUEST_GET_CURRENT_CALLS	
RIL_REQUEST_DIAL	
RIL_REQUEST_HANGUP_FOREGROUND_RESUME_BACKGROUND	
RIL_REQUEST_HANGUP	
RIL_REQUEST_ANSWER	
RIL_REQUEST_HANGUP_WAITING_OR_BACKGROUND	
RIL_REQUEST_SWITCH_WAITING_OR_HOLDING_AND_ACTIVE	
RIL_REQUEST_CONFERENCE	
RIL_REQUEST_SEPARATE_CONNECTION	
RIL_REQUEST_EXPLICIT_CALL_TRANSFER	

RIL_REQUEST_GET_CLIR	UMTS only
RIL_REQUEST_SET_CLIR	UMTS only
RIL_REQUEST_QUERY_CALL_FORWARD_STATUS	UMTS only
RIL_REQUEST_SET_CALL_FORWARD	UMTS only
RIL_REQUEST_QUERY_CALL_WAITING	UMTS only
RIL_REQUEST_SET_CALL_WAITING	UMTS only
RIL_REQUEST_UDUB	
RIL_REQUEST_QUERY_CLIP	UMTS only
RIL_REQUEST_LAST_CALL_FAIL_CAUSE	UMTS only
RIL_REQUEST_SET_MUTE	
RIL_REQUEST_GET_MUTE	
RIL_REQUEST_SET_SUPP_SVC_NOTIFICATION	AT RIL only
RIL_REQUEST_DTMF_START	
RIL_REQUEST_DTMF_STOP	
USSD Requests	
RIL_REQUEST_SEND_USSD	
RIL_REQUEST_CANCEL_USSD	
Miscellaneous Requests	
RIL_REQUEST_SCREEN_STATE	
RIL_REQUEST_OEM_HOOK_STRINGS	Used for custom command support QMI RIL only
RIL_REQUEST_RESET_RADIO	See note in section 8.8.4 QMI RIL only
Unsolicited Responses	
RIL_UNSOL_RESPONSE_RADIO_STATE_CHANGED	
RIL_UNSOL_SIGNAL_STRENGTH	
RIL_UNSOL_RESPONSE_NETWORK_STATE_CHANGED	
RIL_UNSOL_NITZ_TIME_RECEIVED	
RIL_UNSOL_DATA_CALL_LIST_CHANGED	
RIL_UNSOL_RESPONSE_NEW_SMS	
RIL_UNSOL_RESPONSE_CDMA_NEW_SMS	QMI RIL only
RIL_UNSOL_CDMA_OTA_PROVISION_STATUS	QMI RIL only
RIL_UNSOL_RESPONSE_NEW_SMS_STATUS_REPORT	Not supported on MC83x5 modems
RIL_UNSOL_RESPONSE_CALL_STATE_CHANGED	
RIL_UNSOL_OEM_HOOK_RAW	Used for custom command support QMI RIL only
RIL_UNSOL_ON_USSD	
RIL_UNSOL_CALL_RING	AT RIL only
RIL_UNSOL_SUPP_SVC_NOTIFICATION	AT RIL only

4 Install Instructions

4.1 Installing RIL on ICS platform

Please refer to the instructions in the following file contained within the distribution:

For ARM based platforms:
\ReleaseARM\readme.txt

5 Revision History

5.1 Version 3.2.18.0

- Improve AT RIL polling mechanism by remove unnecessary AT command
- Enhance SierraImgMgr functionality for SL9090
- Fixed active call may hang up while rejecting call in CDMA network
- Fixed menu key will hide call control widget in Android phone apps
- Added work around to handle duplicate voice mail indication in Sprint network
- Fixed OTAPA failure reported by ITS during VZW certification
- Added 5s wake lock in USB driver to prevent modem immediately go into suspend after a data wakeup

5.2 Version 3.2.11.0

- Fixed “Emergency Call” string didn’t display in Dialer GUI
- Fixed hang up dialing call will disconnect on hold call
- Fixed HFA doesn’t show correct retry count
- Fixed HFA didn’t invoke after modem profile clean up
- Fixed “Roaming Selection” should show “Sprint” instead of “Home” in Sprint
- Add support to ##SCRTN# and ##MSL# menu
- Add provision status to ##Debug->Configuration menu.
- Add roaming alert for voice call for Sprint
- Hide UMTS options in Mobile Network settings for CDMA.
- Complete CI/NI FUMO support

5.3 Version 3.2.9.0

- Added AT command input box in Android setting menu (disabled by default)
- Fixed call waiting may automatically connected issue
- Fixed on hold call may disconnected during rejecting call issue
- Fixed MSISDN not properly refresh after SIM OTA issue.
- Revised the default ECC number defined in RIL

5.4 Version 3.2.8.0

- Improve ECC number handling in AT and QMI RIL
- Disable voice only AT command for data only modem
- Add Cell broadcast SMS support
- Improve emergency callback mode support in VZW
- Integrate customer changes for USB drivers

5.5 Version 3.2.6.0

- Fixed automatic time zone not correct issue
- Implemented QMI wakeup filter in GobiNet driver
- Shortened SMS retry timeout to 60 seconds to prevent AT channel timeout
- Fixed call waiting UI may not appear if user end an active call at the same time
- Fixed QMI RIL may need 2nd try to start a data session
- Fixed voicemail indication not correct in AT&T network

5.6 Version 3.2.5.0

- Removed ECT button in Android Dialer, use keypad to invoke ECT instead
- Added workaround to fix voice mail number cannot read out from 3G SIM issue
- Fixed “congestion” failure cause not correct parsed issue
- Fixed AT RIL will unmute modem during start up issue
- Handle ECC cannot read out from some SIM issue
- Fixed PTCRB 31.8.7 call fail cause not correct parsed issue
- Fixed PTCRB 31.2.1.1.2 call forwarding request error issue
- Fixed PTCRB 31.8.3.1 call barring request error issue

5.7 Version 3.2.3.0

- Fixed AT command channel cannot recover from timeout issue
- Fixed +CLCC command may interrupt by CONNECT response issue
- Fixed GPS may not resume from host suspend issue
- Fixed caller ID display of 2nd incoming call issue in CDMA dialer
- Fixed SIM ECC read issue for 2G/3G/RUIM

5.8 Version 3.2.2.0

- Fixed CDMA TC 5.1 Data Throttling failed issue
- Fixed SPN not correctly show in CDMA network issue
- Fixed CDMA long SMS not correctly shown in UMTS network issue
- Fixed voice call error code not correctly return to Android framework issue
- Enable CREG/CGREG indication during LCD off
- Fixed modem exit emergency mode after dialed out 2nd emergency call issue
- Further improved dialer may show previous caller ID issue
- Disable auto redial feature

5.9 Version 3.2.1.0

- Fixed long SMS not stable on China Telecom network issue
- Fixed auto redial not working on China Telecom network issue
- Fixed CDMA dormant status incorrect issue

- Fixed roaming icon shows incorrectly during searching network issue
- Fixed 911 mode icon not correctly show after 2nd emergency call issue
- Fixed roaming icon shows incorrectly on China Mobile 46002 SIM issue
- Added audio provisioning for SL9090 QMI RIL
- Fixed DM log tools may crash issue
- Fixed AT RIL didn't restart data call issue
- Added new handling for emergency call list
- Fixed dialer can't reject call in CDMA mode issue
- Fixed SL9090 may fall into fail safe image issue

5.10 Version 3.2.0.0

- Completed CDMA features development and VZW pre-testing (check section 6 for known issue)
- Fixed caller ID not correctly show for 2nd call issue
- Added 911 mode support for SL9090 China Tele Com firmware
- Fixed cannot end emergency call without SIM issue
- Fixed call waiting issue for SL9090 China Telecom firmware
- Fixed CDMA SMS issue (check section 6 for known issue)
- Improved dialer keypad not response during dialing issue
- Fixed incorrect call end message in dialer
- Fixed modem volume become 0 after factory reset issue
- Improved radio technology polling mechanism for AT RIL

5.11 Version 3.1.3.0

- Complete VZW CDMA pre-testing with SL9090 (check section 6 for known issue)
- Fixed dialer stay at call history screen after call issue
- Fixed call forwarding setup sometimes return error issue
- Fixed SIM phone book read/write error issue for SL9090 firmware SWI6600U02000002
- Add proper modem reset after audio provisioning
- Add long/normal/string DTMF setting
- Fixed caller ID not display correctly for 2nd call
- Fixed OTASP cannot initiate by dialer issue
- Add SIM PIN lock support for China Telecom firmware

5.12 Version 3.1.2.0

- Complete ##data# menu support for Sprint and VZW firmware
- Add China Telecom RUIM support
- Fixed modem take long time to register to China mobile network issue
- Remove "add contact to SIM" in "People" application
- Fixed T-Mobile SIM read error issue
- Add voice mail indication support for SL8092 in AT&T network
- Add voice mail indication support for SL9090 in AT&T, Sprint & VZW network

- Fixed emergency mode icon not show correctly issue
- Fixed cannot end CDMA voice call issue
- Fixed cannot make 2nd CDMA voice call issue
- Fixed Android dialer show “busy” after voice call ended issue

5.13 Version 3.1.0.0

- Completed PTCRB pre-testing for SL9090 (check section 6 for known issue)
- Add OMA-DM support for Sprint
- Add OTA-SP support for Verizon (not fully tested)
- Support Voice mail indication for Sprint and VZW (not fully tested)
- Add ##debug menu & ##data menu (check section 6 for limitation)
- Fixed emergency call issue for SL9090
- Fixed audio profile provisioning without SIM issue
- Fixed NITZ indication error issue for SL8092
- Fixed call forwarding error issue for SL8092

5.14 Version 3.0.0.6

- Fixed 2 GCF failed test cases for SL8092
- Fixed NITZ not correctly reported issue for SL8092 and SL9090
- Add temporary work around for DHCP issue for SL8092
- Add RIL work around to fix AT+CRSM not working on SL9090 issue

5.15 Version 3.0.0.4

- Added audio profile setup and switching support to SL9090 and SL8092
- Completed GCF pre-testing of SL8092 (check section 6 for known issue)
- Fixed SIM phonebook may crash during “copy contacts to SIM” issue
- Fixed “H” icon not correctly show issue for SL8092 and SL9090
- Added support to mode 2 SMS for SL8092

5.16 Version 3.0.0.3

- Added Audio control support to SL9090 and SL8092, modem mute, volume and noise suppression are supported
- Added Phone, Contacts and Setting source code to release package base on TI OMAP ICS 4.0.3 BSP for HoneyWell device support

5.17 Version 3.0.0.2

- Added support to multi response USSD
- Support SIM phone book
- Support emergency call defined in Android property
- Added International call barring in Mobile Network Setting

- Added Speed dial and auto re-dial (check section 6 for limitation)
- Support PIN2/PUK2 input in Dialer
- Fixed Band Setting and Neighbor cell limitation in previous release
- Added RIL version in Status tab of Setting
- Added Audio control support to SL9090 (check section 6 for limitation)
- Added maximum file size setup in Sierra logging tool
- Added GPS XTRA support. Enable/disable by Android property “persist.sierra.gpsxtra”

5.18 Version 3.0.0.1

Added the following features and fixes:

- Fixed cannot dial emergency call without SIM issue
- Fixed call forwarding settings cause RIL crash issue
- Fixed call waiting status read error issue
- Fixed phone number of SL9090 not show in Android setting menu
- Fixed incorrect flight mode status report issue
- Fixed 1st ring of SL809x cannot wake host issue
- Added DTMF support
- Added ECT support
- Added FDN support
- Added partial support of USSD (check section 6 for limitation)
- Added Band setting and neighbor cell in mobile network settings (check section 6 for limitation)
- Verified GPS support on SL9090 & SL8092

5.19 Version 3.0.0.0

Added the following features and fixes:

- Based on Sierra Linux SDK 3.x
- Add support to wwan device
- Add source release package
- QMI and AT RIL will be release in the same package
- Fixed dialer unstable issues
- Fixed call barring not working issue
- Fixed call waiting setup not working issue
- Fixed conference call not working issue

6 Known Limitations

The following known issues exist in the RIL. We will work to resolve them in a future release.

- Crash in GobiNet driver if in debug mode while modem resets.
  - If debug output has been enabled on the GobiNet driver, and the modem is power-cycled or resets, then this may cause a kernel crash. Note that debug is not enabled by default, so under normal operation there is no problem.

7 Planned Feature Enhancements

7.1 Additional Supported Features that are not part of native Android

The following features are planned to be supported in a future release of the RIL:

TBD

7.2 Planned Feature Enhancements

The following features are planned to be supported in a future release of the RIL:

TBD

8 Usage Instructions

This section provides information and instructions on using the RIL with the supported modems described in section 2.2.

Note that some of the discussed topics are only applicable to one type of modem. This will be noted in the descriptions.

8.1 Custom RIL Commands

Applications use `Phone.invokeOemRilRequestStrings()` to invoke custom RIL commands. This method is defined in Phone.java as:

```
void invokeOemRilRequestStrings(String[] strings, Message response);
```

The custom command is specified in `strings[0]`; arguments for the custom command are specified in `strings[1]`, etc. The currently supported custom commands are defined in RILConstants.java:

```
String OEM_HOOK_STRING_RIL_VERSION = "4";
String OEM_HOOK_STRING_CDMA_ACTIVATION = "5";
String OEM_HOOK_STRING_FACTORY_RESET = "6";
String OEM_HOOK_STRING_PRL_UPDATE = "7";
String OEM_HOOK_STRING_SAR_GET = "8";
String OEM_HOOK_STRING_SAR_SET = "9";
String OEM_HOOK_STRING_SET_SMS_WAKE = "10";
String OEM_HOOK_STRING_GET_ppDATA = "11";
String OEM_HOOK_STRING_SET_ppDATA = "12";
String OEM_HOOK_STRING_VALIDATE_SPC = "13";
String OEM_HOOK_STRING_READ_SIM_PB = "14";
String OEM_HOOK_STRING_OMADM_CANCEL = "15";
String OEM_HOOK_STRING_GET_NOISE_SUPPRESSION = "16";
String OEM_HOOK_STRING_SET_NOISE_SUPPRESSION = "17";
String OEM_HOOK_STRING_GET_MODEM_VOLUME = "18";
String OEM_HOOK_STRING_SET_MODEM_VOLUME = "19";
```

8.1.1 CDMA Activation

For CDMA activation, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_CDMA_ACTIVATION;
```

There are no arguments for this custom command. See section 8.2 for details about how to use CDMA activation.

This command only applies to MC8355 Sprint and Verizon images.

8.1.2 User initiated PRL Update

For user initiated PRL Update, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_PRL_UPDATE;
```

There are no arguments for this custom command. The modem will reset after a successful PRL update. If the update fails, then either the RIL command will return an error right away, or `Phone. CDMA_OTA_PROVISION_STATUS_OTAPA_ABORTED` will be sent. See section 8.2.1 for details on how to register for and receive this notification.

This command should not be used when there is an active data session. If an attempt is made to use this command when there is an active data session, the command will fail, and `Phone. CDMA_OTA_PROVISION_STATUS_OTAPA_ABORTED` will be sent.

This command only applies to MC8355 Sprint images.

8.1.3 Factory Reset

For factory reset, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_FACTORY_RESET;
strings[1] = <SPC>;
```

The service programming code (SPC) to use depends on the carrier. For carriers other than Sprint, it should be six zeros, i.e. “000000”. For Sprint, it is modem specific, and should be obtained from Sprint.

The modem will reset after a successful factory reset. If the factory reset fails, then the RIL command will return an error right away.

This command only applies to MC8355 modems. It is targeted for Sprint images, but this command should work for any carrier images.

8.1.4 Get Current SAR Back-off State

For getting the current SAR state, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_SAR_GET;
```

There are no arguments for this custom command. The response will contain a value between 0 and 8 that indicates the current SAR back-off state.

This command applies to both MC8355 and MC77xx modems. For MC7700 modems, only the 1.x FW versions are supported.

8.1.5 Set Current SAR Back-off State

For setting the current SAR state, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_SAR_SET;
strings[1] = <SAR state>;
```

The <SAR state> should be a number between 0 and 8, represented as a string.

This command applies to both MC8355 and MC77xx modems. For MC7700 modems, only the 1.x FW versions are supported.

8.1.6 Enable Wake on SMS

For enabling/disabling wake on SMS, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_SET_SMS_WAKE;
strings[1] = <enabled>;
strings[2] = <wake mask>;
```

The <enabled> parameter is 1 for enable and 0 for disable. The <wake mask> parameter is a decimal number up to 8 digits in length. All parameters should be represented as strings. See document "Gobi 2000 Wake on Wireless WAN Specification" for details on how to use the Wake on SMS feature.

This command only applies to MC8355.

8.1.7 Get data for ##data and ##debug

For reading data to support ##data and ##debug, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_GET_ppDATA;
```

The status returned by OEM_HOOK_STRINGS will be either success or failure. If the status is success, then the output response is a list of strings in the following order:

USER_NAI	NAI string of MIP profile, ASCII string
REV_TUNNELING_SETTING	Reverse Tunneling of MIP profile, one byte long decimal string
HA_SPI	HA SPI of MIP profile, up to 4 bytes long hexadecimal value string
AAA_SPI	AAA SPI of MIP profile, up to 4 bytes long hexadecimal value string
HOME_IP	Home address(IPv4) of MIP profile, up to 4 bytes long hexadecimal value string
PRIMARY_HA_IP	HA address(IPv4) of MIP profile, up to 4 bytes long hexadecimal value string
SECONDARY_HA_IP	AAA address(IPv4) of MIP profile, up to 4 bytes long hexadecimal value string
MEID	MEID, ASCII string
P_ESN	ESN, ASCII string
STATION_CLASS_MARK	Station Class Mark, one byte long decimal string
SLOTTED_MODE_INDEX	Slot Cycle Index, one byte long decimal string
HOME_NID	Home NID, one byte long decimal string
HOME_SID	Home SID, one byte long decimal string
ACCOLC	ACCOLC, one byte long decimal string
HOME_SYS_REG	Register on home system, one byte long decimal string

FSID	Register on foreign system, one byte long decimal string
FNID	Register on foreign network, one byte long decimal string
1X_CHANNEL	1xrtt channel number, up to 4 bytes long decimal string
1X_BAND_CLASS	1xrtt band class, up to 4 bytes long decimal string
1X_SYSTEM_ID	1xrtt system ID, up to 4 bytes long decimal string
1X_NETWORK_ID	1xrtt network ID, up to 4 bytes long decimal string
1X_BASE_ID	1xrtt base ID, up to 4 bytes long decimal string
1X_LATITUDE	1xrtt base station latitude, up to 4 bytes long hexadecimal value string
1X_LONGTITUDE	1xrtt base station longtitude, up to 4 bytes long hexadecimal value string
1X_RX_PWR	1xrtt RSSI, up to 4 bytes long decimal string
1X_RX_ECIO	1xrtt ECIO, up to 4 bytes long decimal string
1X_RX_ERROR_RATE	1xrtt error rate, up to 4 bytes long decimal string
EVDO_CHANNEL	EVDO channel number, up to 4 bytes long decimal string
EVDO_RX_PWR	EVDO RSSI, up to 4 bytes long decimal string
EVDO_RX_PER	EVDO Packet Error Rate, up to 4 bytes long decimal string
EVDO_RX_SINR	EVDO SINR, up to 4 bytes long decimal string
EVDO_AN_AAA_STATUS	EVDO AN-AAA status, up to 4 bytes long decimal string
EVDO_PRL	PRL version, up to 4 bytes long decimal string
AMSS_VERSION	Firmware version, ASCII string
P_REV	Protocol version, up to 4 bytes long decimal string
PACKET_DATA_PROFILE	Active MIP profile number, one byte long decimal string

This command only applies to MC8355 Sprint images.

8.1.8 Set data for ##data

For setting data to support ##data, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_SET_ppDATA;
```

followed by the values to write in the following order:

SPC	Service Programming Code, expecting up to 6 bytes long decimal value string
USER_NAI	NAI string of MIP profile, expecting ASCII string
REV_TUNNELING_SETTING	Reverse Tunneling of MIP profile, expecting one byte long decimal string
HA_SPI	HA SPI of MIP profile, expecting up to 4 bytes long hexadecimal value string
HA_PASSWORD	HA password of MIP profile, expecting ASCII string
AAA_SPI	AAA SPI of MIP profile, expecting up to 4 bytes long hexadecimal value string
AAA_PASSWORD	AAA password of MIP profile, expecting ASCII string
HOME_IP	Home address(IPv4) of MIP profile, expecting up to 4 bytes long hexadecimal value string
PRIMARY_HA_IP	Primary HA address(IPv4) of MIP profile, expecting up to 4 bytes long hexadecimal value string
SECONDARY_HA_IP	Secondary HA address(IPv4) of MIP profile, expecting up to 4 bytes long hexadecimal value string
ACCOLC	ACCOLC, expecting one byte long decimal string

There is no output response data for this command; just a success or failure status.

This command only applies to MC8355 Sprint images.

8.1.9 Validate SPC

For validating the SPC, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_VALIDATE_SPC;
strings[1] = <spc>;
```

The <spc> parameter is the Service programming code, as a string up to 6 digits long. The return status will be either success or failure. If the status is success, then the output response is a list of strings, containing a single string. This response string will be "1" if the SPC is valid and "0" if the SPC is not valid.

This command only applies to MC8355.

8.1.10 Read_SIM_Phonebook

For reading the SIM phonebook, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_READ_SIM_PB;
```

The return status will be either success or failure. If the status is success, then the output response is a list of strings, containing all of the phonebook entries from the SIM. Each string will contain one phonebook entry. The format of a phonebook entry is as defined for the AT+CPBR command: <index>,<number>,<type>,<text>.

8.1.11 OMA-DM Cancel

For OMA-DM Cancel, `strings` should be:

```
strings[0] = RILConstants.OEM_HOOK_STRING_OMADM_CANCEL;
```

There are no arguments for this custom command. The return status will be either success or failure. There is no data in the command response.

This command can be used to cancel either network-initiated or client-initiated OMA-DM sessions.

This command only applies to MC8355 Sprint images.

8.2 CDMA Activation for MC8355 Modems

The Sierra RIL handles the details of the CDMA activation procedure. Sprint uses OMA-DM and Verizon uses OTASP. The following description is specific to the Android 2.2

(Froyo) release. The assumption is that the target Android release will provide the same or very similar functionality.

8.2.1 Application Layer Operation

The Android Application layer will use the following APIs to trigger a CDMA Activation session and receive status information. From the Android layer, the Activation procedure is the same for Sprint and Verizon.

Initiate Activation:

- `Phone.invokeOemRilRequestStrings()`
- Exact method parameters are still TBD, and will be carrier specific.

Register for Status Notifications

- `Phone.registerForCdmaOtaStatusChange()`
- Although the name mentions OTA, it will be used for all activations.

Status Activation Started:

- `Phone.CDMA_OTA_PROVISION_STATUS_OTAPA_STARTED`
- Sent to the registered method when activation has started

Status Activation Complete:

- `Phone.CDMA_OTA_PROVISION_STATUS_COMMITTED`
- Sent to the registered method when activation has completed

Status Activation Failed:

- `Phone. CDMA_OTA_PROVISION_STATUS_OTAPA_ABORTED`
- Sent to the registered method when activation has failed.

8.2.2 Activation Application Implementation

Currently, activation is done through the PhoneApp using a dialing interface. In order to implement customer specific activation requirements, either changes need to be made to the PhoneApp, or a new Activation App must be created. See comments in OtaStartupReceiver.java (in packages/apps/Phone/src/com/android/phone) for details on how to disable the default behaviour and provide a custom Activation App.

The following will discuss the minimum changes required to the PhoneApp, in order to use the APIs described above. This does not include any customer specific UI changes. Note that this is just example code, and there may be better ways to do this, depending on particular customer requirements.

8.2.2.1 Initiate Activation

In OtaUtils.java, modify otaPerformActivation() to be:

```
    private void otaPerformActivation() {
        if (DBG) log("otaPerformActivation()...");

/* SWISTART */
        PhoneApp app = PhoneApp.getInstance();
        Phone phone = app.phone;
/* SWISTOP */

        if (!mApplication.cdmaOtaProvisionData.inOtaSpcState) {
/* SWISTART */
            /* Exact parameters TBD */
           phone.invokeOemRilRequestStrings(null, null);
/* SWISTOP */
            otaShowListeningScreen();
        }
        return;
    }
```

8.2.2.2 Handle Status Activation Complete

In OtaUtils.java, modify the switch statement in onOtaProvisionStatusChanged() for the Phone.CDMA_OTA_PROVISION_STATUS_COMMITTED case to be:

```
            case Phone.CDMA_OTA_PROVISION_STATUS_COMMITTED:
                otaShowInProgressScreen();
                mApplication.cdmaOtaProvisionData.isOtaCallCommitted = true;
                if (DBG) log("onOtaProvisionStatusChanged(): DONE,
                    isOtaCallCommitted set to true");
/* SWISTART */
                otaShowSuccessFailure();
/* SWISTOP */
                break;
```

8.3 Multi-Carrier Firmware Image Management on MC8355

8.3.1 Image Management Overview

The MC8355 supports a multi-carrier image management solution. This solution allows the host client to switch between different carrier firmware images and configuration files in order to support operation on those carriers.

In order to support this within an Android OS, a custom application must be written. This application must be able to do the following:

Read a list of supported Carrier images

Display the list to end user
Allow end user to select the requested Carrier from the list
Trigger the switching to a new carrier image

The MC8355 will support the following carrier images at this time:

Verizon
Sprint
AT&T
Vodafone
T-Mobile
Telefonica
Generic HSPA

8.3.2 Application Layer Operation

The application layer should call the SierraImgMgr executable described below to run the image switching.

Print a list of supported carrier images:

```
SierraImgMgr --list <image directory>
```

A list of all available carrier images in <image directory> will be printed. The list will include the following comma-separated fields: Id, Carrier, Technology, Region, Version and Status.

For example, if there are 3 images with the Sprint image being currently selected, the output might be:

```
Id,Carrier,Technology,Region,Version,Status
2,AT&T,UMTS,NA,0E123456,Inactive
3,Sprint,CDMA,NA,0E223456,Active
6,Generic HSPA,UMTS,Global,0E323456,Inactive
```

The Id values will always be the same for a given carrier.

Trigger the switch to new carrier image:

```
SierraImgMgr --select <Id> <image directory>
```

The Id value must be one of the values returned by the --list option, and is used to select the active image. The --select option may cause the modem to reset. The command will not return until the modem has finished resetting and the new carrier image is successfully running.

Update images on the modem:

```
SierraImgMgr --update <image directory>
```

This will update all the images that are currently on the modem with newer image files, if they are available. If there is no newer image file available for a particular carrier, then nothing will be done with that carrier image on the modem.

The --update option does not change the active carrier selection.

Print a list of carrier images on the modem:

```
SierraImgMgr --listdevice
```

A list of available carrier images on the modem will be printed. The list will follow the same format that is used by the --list option explained earlier.

Trigger the switch to new carrier image on the modem

```
SierraImgMgr --selectdevice <Id> [<config file path>]
```

The Id value must be one of the values returned by the --listdevice option, and is used to select the active image.

The `config file path` value gives the path to the optional configuration file. If there are multiple versions of the same modem image on the modem, then this config file is used to determine which image version to use for a particular carrier. If there is no config file in this case, then the first version of the modem image that is found will be used.

An example of the config file is given below:

```
#Sprint Image
SP=1579
#Verizon Image
VE=1579
#Vodafone Image
VO=1580
#AT&T Image
AT=1580
#T-Mobile Image
TM=1575
#Generic UMTS Image
GE=1580
#Telefonica Image
TE=1575
#Telecom Italia Image
TI=1575
#Orange Image
OR=1575
```

Any lines starting with '#' are comments. Spaces are not allowed in the assignment of the image version number to the image symbol.

The --selectdevice option may cause the modem to reset. The command will not return until the modem has finished resetting and the new carrier image is successfully running.

Notes:

- the <image directory> argument must be the absolute path to the images directory. It should not contain any spaces.
- the --select, --update, --listdevice and --selectdevice options conflict with the RIL daemon. To ensure correction operation, always stop the RIL daemon before using these options, and restart the daemon afterwards.

Example Usage (from shell command line):

```
stop ril-daemon
SierraImgMgr --list /data/modem/images
SierraImgMgr --select 2 /data/modem/image
start ril-daemon
```

The above can be easily adapted to be called from a Java application.

8.3.3 Image Directory

The image distribution package will contain the images directory. This directory will itself contain a number of sub-directories, usually named with small numbers. It is essential that these sub-directories are never renamed, and that files are not moved from one sub-directory to another sub-directory.

This images directory can be put in any customer specific location. The absolute path to the image library must always be given to the SierraImgMgr executable. This path should include the images directory, as given in the example above.

If a new image distribution package is released, use the following steps to update the images used by the modem:

- To keep any images that have not been updated, merge the new images directory with the existing directory, replacing any existing files.
- To only use the newly released images, delete the existing images directory first, and then put the new directory in the same place.
- Execute SierraImgMgr with the --update option as described above.

8.4 SAR Back-off Tool

A command line tool, SierraSARTool, which runs in the adb shell or an Android console, is provided to get and set the current SAR state value.

This tool is targeted at MC8355 modems, although the set command also works with some FW versions of MC7700 modems. For MC77xx modems, AT commands are available which provide similar functionality.

Query the current SAR value:

```
SierraSARTool    /* without arguments */
```

Set the current value to new SAR state:

```
SierraSARTool <new SAR State>
```

Note that this tool cannot run at the same time as the RIL daemon. Therefore, always stop the RIL daemon before using this command, and restart the daemon afterwards.

Example Usage (from adb shell command line):

```
stop ril-daemon
SierraSARTool          /* to query current value */
SierraSARTool 4        /* to set the new value to 4 */
start ril-daemon
```

8.5 Firmware Image Download on MC77xx

The SierraFwDl77xx executable is provided for downloading firmware to MC77xx modems. It is a command line tool and is available in /system/bin. The -h option can be used to get usage information. The command options and usage of these options is provided below:

- -h --help
    - Display this information and exit
- -? --help
    - Display this information and exit
- -v --verbose
    - Display extra info while running. NOTE: specify 'verbose' first
- -g --get

    - Display the information for the executing device image

- -d --download <CWEimagepath>
    - CWE image to be downloaded is at <CWEimagepath> . NOTE: this must be an absolute path

- -i --info <CWEimagepath>
    - Display the information for a particular CWE image file located at <CWEimagepath>. NOTE: this must be an absolute path

8.6 Firmware Image Download on SL809x

Firmware updates are released as CWE files. The swifwdnld executable is provided to update the modem firmware from the adb shell.
Usage:

- cd /system/data
- ./swifwdnld -p ./swisdk -f /data/<file.cwe> -x

The FW update may take several minutes. The modem will reboot at the end of the update.

8.7 Remote DM Logging

Remote DM logging can be used to connect the DM port on the modem under Android to a Windows PC application that uses the DM port, such as QXDM. There are several options to enable remote DM logging (also called DM port forwarding). The SierraDMLog executable is provided to help with this process. The -h option can be used to get usage information. The command options and usage of these options is provided below:

Local Logging:

- SierraDMLog -l [-d device] [-f filter] [-o logfile]:
    - Logs DM packets to "logfile" if specified, or to a script generated log file stored in the /data directory with name swidmlog if not specified.

Remote Logging:

- SierraDMLog [-d device] [-f filter] [-p netport] [-r rhost]
    - Establishes a TCP connection with a remote machine "rhost" using port number "netport", or a default port number of 2500 if not specified. DM packets are exchanged over the TCP connection.

- Without the -r option, acts as a server application waiting for an incoming connection request. Otherwise, acts as a client and attempts to establish a connection with "rhost".

OPTIONS:

- -l - Local logging if specified, remote logging otherwise
- -d device - (Optional) /dev/ttyUSBx port for DM logging
- -f filter - (Optional) DM filter to send to the device prior to logging
- -p netport - (Optional) Remote logging TCP port (defaults to 2500)
- -r rhost - (Optional) remote host to connect to
- -o logfile - (Optional) fully qualified DM log (output) file name

If the DM port is not specified using -d option, then it will be automatically detected by the command. In this case, the port name will be printed out, since it is needed for some of the other steps, as described in the examples below.

All of the examples below assume the DM port on the Android device is /dev/ttyUSB0.

Using USB cable with Windows host computer:

1) Execute “adb shell SierraDMLog -d /dev/ttyUSB0”
2) Execute “adb forward tcp:2500 dev:dev/ttyUSB0”
3) On Windows host using QPST, configure “IP Server” to use IP 127.0.0.1 and port number 2500 to connect to device.

An alternative for USB connection with Windows host computer:

1) Enable USB tethering in Phone/Tablet Settings->Wireless & Networks -> Tethering and & Portable hotspot -> USB tethering
2) Use “adb shell netcfg” to get ip address of USB0
3) Use USB cable to connect to host, then use “ipconfig” find ip address of host device
4) Use “adb shell SierraDMLog -d /dev/ttyUSB0 -r xx.xx.xx.xx”
5) On host computer, launch QPST to add port
6) On host computer, run QXDM

Using Wifi hotspot provided by the phone/tablet:

1) Enable Wifi in phone/tablet settings -> Wireless & Networks -> Wifi
2) Enable hotspot functionality in Settings ->Wireless & Networks -> Tethering and & Portable hotspot ->Portable Wifi hotspot
3) On host computer, use Wifi to connect to access point “AndroidAP”
4) Use “adb shell SierraDMLog -d /dev/ttyUSB0 –r xx.xx.xx.xx”
5) In host computer, launch QPST to add port
6) In host computer, run QXDM

8.8 RIL Commands

This section provides additional information on using some of the standard RIL commands.

8.8.1 SIM Related Commands

From the RIL documentation, it is unclear whether it is acceptable to return valid data in the RIL command response, when the response status is not RIL_E_SUCCESS. However, for some RIL commands, it is necessary to return data even when an error occurs.

For the RIL commands

- RIL_REQUEST_ENTER_SIM_PIN
- RIL_REQUEST_ENTER_SIM_PIN2
- RIL_REQUEST_ENTER_SIM_PUK
- RIL_REQUEST_ENTER_SIM_PUK2

there is no other documented mechanism available to return the number of retries when the status is RIL_E_PASSWORD_INCORRECT. Thus, for these commands, when the status is either RIL_E_SUCCESS or RIL_E_PASSWORD_INCORRECT, the number of retries is returned in the command response. Note that the format of the command response has not changed, and follows the documented format in the ril.h header file.

A corresponding change in the telephony framework is necessary, in order to extract the retry count from the command response when the status is RIL_E_PASSWORD_INCORRECT, because this is normally only done when the status is RIL_E_SUCCESS.

8.8.2 RIL_REQUEST_SCREEN_STATE Command

This RIL command enables or disables the following unsolicited responses:

- RIL_UNSOL_SIGNAL_STRENGTH
- RIL_UNSOL_RESPONSE_NETWORK_STATE_CHANGED
- RIL_UNSOL_NITZ_TIME_RECEIVED

If these responses are disabled, then the corresponding notifications from the modem are also disabled.

The following unsolicited responses are not controlled by this RIL command.

- RIL_UNSOL_RESPONSE_RADIO_STATE_CHANGED

- RIL_UNSOL_DATA_CALL_LIST_CHANGED
- RIL_UNSOL_RESPONSE_NEW_SMS
- RIL_UNSOL_RESPONSE_CDMA_NEW_SMS

The following unsolicited response only occurs when using some of the custom RIL commands described in section 8.1, and so will not occur during normal RIL operation.

- RIL_UNSOL_CDMA_OTA_PROVISION_STATUS

Note that the full list of unsolicited responses supported by the RIL is given in section 3.

8.8.3 RIL_REQUEST_QUERY_AVAILABLE_NETWORKS Command

This RIL command is used to query available GSM/UMTS/LTE networks

- This command requires radio resources, and so cannot be used when there is an active data session. In this case, an error will be returned immediately.
- This command may take up to 5 minutes to execute. If the network scan has not been completed after 5 minutes, an error will be returned.

8.8.4 RIL_REQUEST_RESET_RADIO Command

This RIL command can be used to reset the modem. Although this command has been deprecated, it is still supported in the telephony framework, and no alternative command has been defined that can be used to reset the modem.

8.8.5 SMS SIM Related Commands

The following are the SMS SIM related commands.

- RIL_REQUEST_WRITE_SMS_TO_SIM
- RIL_REQUEST_DELETE_SMS_ON_SIM

The message/record index for these commands starts at 1, to match the telephony framework which assumes that SMS messages stored on the SIM start at index 1.

8.8.6 Receiving SMS Related Commands and Notifications

To prevent SMS storage on the modem from becoming full, when an incoming SMS is acknowledged by either the SMS_ACKNOWLEDGE or

CDMA_SMS_ACKNOWLEDGE commands, as appropriate for the type of incoming SMS, then the SMS will be deleted from modem storage.

In addition, the first time the new RIL (v1.1.9 or higher) runs on the host device, it will check for any received messages on the modem, and send these to the telephony framework. As a result, the user could receive a number of SMS messages that had already been previously received.

8.8.7 RIL_REQUEST_SET_NETWORK_SELECTION_MANUAL command

If manual registration is attempted with a de-activated SIM, then this command will return ILLEGAL_SIM_OR_ME. To verify that the SIM is de-activated, this command will ensure that

- manual registration fails with a "no network service" error (CME ERROR 30)
- after manual registration fails, the registration state is denied

8.8.8 Voice Related Commands

In order to use the voice related RIL commands:

- the modem must have voice support. This is currently only the SL909x
- the ro.sierra.voice property must be set to 1

8.9 CDMALTEPhone Object

This is a new phone object that will be used to support MC7750 modems on the Verizon network. In order to enable this phone object, the following must be done:

1. Add the property setting

   ```
   telephony.lteOnCdmaDevice=1
   ```

   into an appropriate property file, such as system.prop

2. Ensure that the network mode is 7. If the network mode is not already 7, then this can be done by deleting the database file:

   ```
   /data/data/com.android.providers.settings/databases/settings.db
   ```

   This file will get regenerated when the system boots. Assuming that the property in step 1 is set, the network mode will be initialized to 7.

Initial support has been added in the RIL to auto-detect the MC7750, and set the appropriate property, if the property is not already set. See section 8.13 for details.

8.10 Java App for Capturing Radio and Main Logs

DISCLAIMER:
This App is intended for use with Sierra Wireless modems and should only be used for diagnostic and testing purposes. This App should not be included in any shipping product, due to potential privacy and security issues.

Installation:

- Install as you would any other third party non-Android Marketplace application.

Usage:

- Either the radio log or main log or both can be collected by selecting the appropriate option.
- Output file name prefix can be specified in "Log label" field.
- A timestamp is appended to each output file name, giving the time that the log collection started.
- Output files are located in /data/data/com.sierra.logs/files or on the root of the SD card, if "Save to SD card" is selected.

8.11 RIL Properties

The operation of the RIL can be modified by setting various Android properties.

8.11.1 persist.sierra.sim_ready_delay

During RIL initialization, after the RIL has determined that the SIM is ready, it will send a RADIO_STATE_CHANGED notification to the telephony framework, indicating that the SIM is ready. The framework usually responds by issuing a number of SIM related commands.

This property can be used to delay sending the notification that the SIM is ready, until after the modem has successfully registered. The value of the property is used to specify a timeout value in seconds. If the modem is not registered within the timeout period, then the SIM_READY notification is sent, regardless of the current registration state. Typically this value should be at least 10-15 seconds, but normally not more than 60 seconds.

This setting can be useful for certain modems, such as the MC7750 with 1.x FW and the MC8355, where the SIM related commands sent by the framework during initialization may interface with the registration process. If this happens, then the SIM related command may hang the RIL. The RIL will then have to recover and restart the initialization process.

8.11.2 persist.sierra.block_init_reg

During system start-up, the telephony framework will send a registration command, either automatic or manual depending on what was previously selected. This is separate from the scanning and registration that the modem will do on its own when it is powered up. This initial registration command can sometimes cause problems for test equipment that is not expecting it. No problem has ever been observed on a live network with the initial registration command.

If this property is defined and has a value greater than 0, then any initial registration commands will be ignored by the RIL. The RIL will return success, but will not do anything further. Once an explicit network scan has been performed, using RIL_REQUEST_QUERY_AVAILABLE_NETWORKS, the registration commands will work as normal.

Normally, the user should not notice any difference in behaviour if this property is defined, because the modem will normally retain the last registration settings, even after a power cycle.

8.11.3 ro.sierra.voice

This property must be set to 1 in order to enable the voice related RIL commands. If this property is not set, or is set to some other value, then the voice related RIL commands will return an error.

8.11.4 persist.sierra.sim_poll_delay

In some cases, modem will return SIM_FAILURE during SIM busy. Add this property to enable extra SIM polling when modem returns SIM_FAILURE.

Android RIL will perform 10 times of retry if this property is defined. If this property set to 6, RIL will add extra 1 minute of SIM status polling.

Typical use case is MC7750 (FW3.5.10.6Z) in CDMA mode.

8.11.5 persist.sierra.gpsxtra

Set persist.sierra.gpsxtra to 1to enable the GPS XTRA feature for SL8092 and SL9090.

8.12 Custom RIL Unsolicited Responses

All of the unsolicited responses use the RIL_UNSOL_OEM_HOOK_RAW unsolicited response. The response data is provided as a byte array. The contents of this byte array are described in the following sections.

8.12.1 OMA-DM Session States

The contents of the byte array are:

byte_array[0] = 1;
byte_array[1] = <oma-dm session state>

When an OMA-DM session is in progress, or has been attempted this unsolicited response indicates the current state of the session. This response will be sent for both client and network initiated sessions. Possible values for <oma-dm session state> are:

0x00 - Complete, information was updated
0x01 - Complete, update information is unavailable
0x02 - Failed
0x03 - Retrying
0x04 - Connecting
0x05 - Connected
0x06 - Authenticated
0x07 - Mobile Directory Number (MDN) downloaded
0x08 - Mobile Station Identifier (MSID) downloaded
0x09 - PRL downloaded
0x0A - Mobile IP (MIP) profile downloaded

This unsolicited response only applies to MC8355 Sprint images.

8.13 Detecting LTE modems

Initial support has been added in the RIL to auto-detect LTE modems, such as the MC7700 and MC7750, and set the properties used by the telephony framework, as follows:

- MC7700 detected
  - telephony.lteOnGsmDevice=1
  - telephony.lteOnCdmaDevice=0

- MC7750 detected
    - telephony.lteOnGsmDevice=0
    - telephony.lteOnCdmaDevice=1

If neither of these modems is detected, then both properties are set to 0.

The auto-detection is performed as early in the RIL startup as possible, in order to try to ensure that these properties are set before the corresponding Phone object is created. However, it is possible if RIL startup is delayed, that the Phone object will be created before the RIL has a chance to set these properties. This problem is still under investigation.

NOTE: the properties will not be modified by the RIL as described above, if either of the properties is already set when the RIL starts up. Thus, the above behaviour can be disabled by explicitly setting either of the properties, before the RIL starts up.