Release Notes

Product: Application Framework for EmberZNet 5.7.3
Release Date: June 29, 2016

1 Overview

This release contains the GA release of Application Framework for use with EmberZNet 5.7 for the EFR32 and EM3xx platforms. This release must be used with Silicon Labs Simplicity Studio 3.3, which replaces Ember Desktop and which incorporates AppBuilder. It is installed along with the EmberZNet stack. For more information on software installation and support, see QSG106, Getting Started with EmberZNet."

2 The 5.7 Release

This release includes a number of new features, improvements, and bug fixes. Customers upgrading from previous releases are strongly encouraged to carefully review this section. It is important to become with familiar with the changes in this release before attempting to migrate applications.

Version 5.7.3

2.1 New Features

2.2 Configuration Changes

2.2.1 Graphics Library Functionality

The Graphics Library plugin source files were previously hardcoded into the EFR32 application framework Studio project templates. The hardcoded source files have been removed from the template, so users can now choose whether or not to include Graphics Library functionality in their application. Users upgrading to this release from previous releases should consider if they want to include the Graphics Library files in their application, and enable or disable the Graphics Library plugin accordingly.

2.3 Removed APIs

No APIs were removed in this release.

2.4 Removed Callbacks

No callbacks were removed in this release.

2.5 Removed CLI Commands

No CLI commands were removed in this release.

2.6 Removed Plugins

There were no removed plugins in this release.

2.7 Removed Sample Applications

There were no removed sample applications in this release.

2.8 Changed APIs

No APIs were changed in this release.

2.9 Changed Callbacks

There were no callbacks changed in this release.

2.10 Changed CLI Commands

There were a no CLI commands changed in this release.

2.11 Changed Plugins

The following plugins were changed in this release.

2.11.1 EFR32 Support

The following plugins are now usable on the EFR32 platform.

- Battery Monitor
- Button Interface
- Generic GPIO Interrupt Controller
- GPIO Sensor Interface
- I2C Driver
- Illuminance Si1141
- LED Blinking
- Occupancy PYD-1698
- Relative Humidity Si7021
- SB1 Gesture Sensor
- Tamper Switch Interface
- Temperature Si7021
- Temperature Si7053

2.12 Changed Sample Applications

There was one major update to the existing sample applications in this release.

2.12.1 HaOccupancySensor

The HaOccupancySensor sample application is now usable on the EFR32 platform.

2.13 Deprecated APIs

No APIs were deprecated in this release.

2.14 Deprecated Callbacks

No callbacks were deprecated in this release.

2.15 Deprecated CLI Commands

No CLI commands were deprecated in this release.

2.16 Deprecated Plugins

There was one deprecated plugin in this release.

2.16.1 Color Control Cluster

?? This plugin is being deprecated in this release. It is being replaced by the Color Control Cluster Server plugin (see **??** for more information on the new plugin).

2.17 Deprecated Sample Applications

No sample applications were deprecated in this release.

2.18 New APIs

No APIs were added in this release.

2.19 New Callbacks

There were no new callbacks added in this release.

2.20 New CLI Commands

There were no significant CLI commands added in this release.

2.21 New Plugins

Here are the new plugins added in this release.

2.21.1 Bulb PWM Driver

This plugin provides an API for driving RGBW PWM values for the color, color temperature, and dimmable lighting reference desings.

2.21.2 Bulb Pwm CLI

This plugin provides CLI for the Bulb PWN driver. See the Bulb PWM Driver plugin above for a deeper look into this functionality.

2.21.3 Bulb User Interface

This plugin implements a bulb user interface. When it turns on, it will kick off the join automatically. When it is turned off and on 8 times with no 2 second delay, it will leave the network and try to join. This plugin is used in the new lighting reference designs.

2.21.4 Coexistence Configuration

The Coexistence Configuration plugin is new in this release. Customers can use this plugin to configure their Packet Traffic Arbitration (PTA) signals in the AppBuilder UI. To learn more about PTA, please see the ZigBee Coexistence Application Note.

2.21.5 Color Control Cluster Server

?? The Color Control Cluster Server is a new plugin meant to replace the old Color Control Cluster plugin. Both of these plugins provide application layer functionality for the ZCL Color Control cluster. See **??** for more information on the deprecated plugin.

2.21.6 Configuration Server

This plugin is a generic interface for writing tokens by means of an over the air protocol. Users should note that tokens will be defined by different configuration plugins, but they will be written here.

2.21.7 Electrical Measurement Server

This plugin implements the Electrical Measurement server cluster. It will use a HAL power meter plugin API to retrieve the electrical measurement data to populate the RmsVoltage, RmsCurrent and ActivePower attributes of the Electrical Measurement server cluster. It requires the inclusion of a HAL plugin that implements the power meter API, which will be used to generate new electrical data.

2.21.8 LED PWN Plugins

There are three different plugins new to this release that drive a LED using PWN. Each of the plugins contains a different kind of lighting effect. Users should note that each of these plugins only supports a single endpoint.

- LED Dim PWM
- LED Temp PWM
- LED RGB PWM

2.21.9 Manufacturing Library OTA

This plugin adds functionality for supporting manufacturing library commands over the air.

2.21.10 Metering Server

This plugin implements the Simple Metering server cluster. It will use a HAL power meter plugin API to retrieve the electrical data to populate the CurrentSummation delivered, PowerFactor, Status attributes. It also supports profiling by populate the CurrentPartialProfileIntervalStartTime and CurrentPartialProfileIntervalValueDelivered attributes. It requires the inclusion of a HAL plugin that implements the power meter API, which will be used to generate new electrical data.

2.21.11 Power Meter CS5463

This HAL plugin implements the power meter API using a Cirrus Logic CS5463 Power Measurement device.

2.22 New Sample Applications

Here are the new sample applications delivered in this release.

2.22.1 HaSmartOutlet

This application is intended to accompany the Silicon Labs smart outlet reference design, It acts as a router and has two LEDs for network status and power indication purposes. It also has one button for user control.

The device will begin looking for a network on power up. Once it joins a network, it measures the power, temperature, relative humidity, and illuminance and report the gathered data. One can control the outlet power on/off through bound client.

Button operations and LED indications:

One press (0.5 seconds to 2 seconds) to toggle the power on/off manually. Power LED(red) will turn on for 1 seconds when the power is toggled on and power LED(red) will turn on for 250ms when the power is toggled off.

Two short presses (shorter than 0.5 seconds each) to enable the identify mode if the outlet has joined a network. The network LED will flash 250ms on, 250ms off, 250ms on and then 1.25 seconds off. If the outlet has not joined any network, it will start searching for a new one. In this case The network LED will flash on 250ms every 2 seconds.

Three short presses (shorter than 0.5 seconds each) to indicate the network status. If the outlet has joined a network, the network LED will flash 250ms on, 250ms off. If the outlet has not joined any network, it willl start searching for a new one. In this case The network LED will flash on 250ms every 2 seconds.

Five short presses (shorter than 0.5 sesinds each) to reset theh outlet. The outlet will leave current network and turn off the power.

A long press(more than 3 seconds) to leave current network and start searching for a new one.

This sample application is usable on both the EM35X and EFR32 chip series.

2.22.2 Lighting Reference Designs

There are three new HA lighting reference designs new to this release.

- HaColorControlLight
- HaColorTempLight
- HaDimmableLight

This reference software is for use with the lighting modules IST-A0085 and IST-A0087 for the EFR chip series, and RD-0020 and RD-0035 using board header IST-A22 for the EM35X chip series. The HaColorTempLight and HaDimmableLight applications are HA certifiable, whereas the HaColorControlLight requires extending to be HA certifiable.

Each of the applications contains setup instructions in the Description section on the General tab in AppBuilder.

Version 5.7.2

2.23 New Features

2.24 Configuration Changes

No Configuration Changes were made during this release.

2.25 Removed APIs

No APIs were removed in this release.

2.26 Removed Callbacks

No callbacks were removed in this release.

2.27 Removed CLI Commands

No CLI commands were removed in this release.

2.28 Removed Plugins

There were no removed plugins in this release.

2.29 Removed Sample Applications

There were no removed sample applications in this release.

2.30 Changed APIs

No APIs were changed in this release.

2.31 Changed Callbacks

The following callbacks were changed in this release.

2.31.1 emberAfPluginTemperatureSi7053DataReadyCallback

When the temperature measurement server was released in version 5.4 of the stack, each callback had to be prefaced by the name of the plugin that provided the callback. This was changed in 5.6 to allow a plugin to create a generically named callback, which provides the flexibility to have a family of HAL plugins all generate the same callback based on the provided hardware. As such, the `emberAfPluginTemperatureSi7053DataReadyCallback` has been replaced with the `halTemperatureReadingCompleteCallback`, which is generated by either the Temperature Si7021 plugin, the Temperature Si7053 plugin, or any other HAL plugin that implements the temperature API.

2.32 Changed CLI Commands

There were a no CLI commands changed in this release.

2.33 Changed Plugins

The following plugins were changed in this release.

2.33.1 Connection Manager

The connection manager formerly contained a large amount of code to handle UI elements, including mapping button presses to network activity and blinking the UI LED to indicate device state. This functionality has been migrated to the new Silicon Labs Device UI plugin.

2.33.2 Reporting

Two changes were made to the reporting plugin. On device powerup and reboot, the reporting plugin will now generate a report for all attributes that have been configured as reportable. Also, a fix was included for a bug which would limit the number of attribute reports generated on startup to the size of the callback queue (three by default).

2.33.3 Security Sensor

The security sensor formerly contained a large amount of code to handle UI elements, including mapping button presses to network activity and blinking the UI LED to indicate device state. This functionality has been migrated to the new Silicon Labs Device UI plugin.

2.34 Changed Sample Applications

The following sample applications were changed in this release.

2.34.1 HaCapSenseDimmerSwitch

The capacitive sense dimmer switch sample application was updated to use the Silicon Labs Device UI plugin to handle all user interface elements.

2.34.2 HaContactSensor

The contact sensor sample application was updated to use the Silicon Labs Device UI plugin to handle all user interface elements.

2.35 Deprecated APIs

No APIs were deprecated in this release.

2.36 Deprecated Callbacks

No callbacks were deprecated in this release.

2.37 Deprecated CLI Commands

No CLI commands were deprecated in this release.

2.38 Deprecated Plugins

No plugins were deprecated in this release.

2.39 Deprecated Sample Applications

No sample applications were deprecated in this release.

2.40 New APIs

No APIs were added in this release.

2.41 New Callbacks

There were no new callbacks added in this release.

2.42 New CLI Commands

There were no significant CLI commands added in this release.

2.43 New Plugins

The following plugins were added in this release.

2.43.1 Humidity Si7021

This HAL plugin implements the humidity API using a Silicon Labs Si7021 combined temperature and relative humidity sensor.

2.43.2 Illuminance Measurement Server

This plugin implements the Illuminance Measurement server cluster. It will periodically poll an illuminance sensor and use that value to populate the MeasuredValue, MinMeasuredValue, and MaxMeasuredValue attributes of the Illuminance Measurement cluster. It requires the inclusion of a HAL plugin that implements the illuminance API, which will be used to generate new illuminance values.

2.43.3 Illuminance SI1141

This HAL plugin implements the illuminance API for a Si1141 illuminance sensor.

2.43.4 Occupancy PYD1698

This HAL plugin implements the occupancy API for an Excelitas PYD1698 occupancy sensor.

2.43.5 Occupancy PYD1698 CLI

This plugin provides a command line interface for modifying the operational settings of the PYD1698 plugin at run time.

2.43.6 Occupancy Sensor Server

This plugin implements the Occupancy Sensor server cluster. It will use callbacks from a HAL occupancy sensor plugin to populate the Occupancy and OccupancySensorType attributes of the Occupancy Sensor cluster. It requires the inclusion of a HAL plugin that implements the occupancy API, which will be used to generate new occupancy state values.

2.43.7 Relative Humidity Measurement Server

This plugin implements the Relative Humidity Measurement server cluster. It will periodically poll a humidity sensor and use that value to populate the MeasuredValue, MinMeasuredValue, and MaxMeasuredValue attributes of the Relative Humidity Measurement Cluster. It requires the inclusion of a HAL plugin that implements the humidity API, which will be used to generate new humidity values.

2.43.8 Silicon Labs Device UI

This plugin implements a user interface for a Silicon Labs device. It uses a single button to implement joining and leaving a network and an LED to indicate network activity (such as searching for, joining, and leaving a network).

2.43.9 Temperature Si7021

This HAL plugin implements the temperature API using a Silicon Labs Si7021 combined temperature and relative humidity sensor.

2.44 New Sample Applications

The following sample applications were added in this release.

2.44.1 HaOccupancySensor

This application is intended to accompany the Silicon Labs occupancy sensor reference design. It uses a PYD1698 occupancy sensor, an Si7021 humidity and temperature sensor, and an Si1141 illuminance sensor to populate the Occupancy, Humidity, Temperature, and Illuminance Measurement clusters as a server.

On powerup, the device will begin searching for a network. Once it joins, it will blink a pattern defined by the Silicon Labs Device UI plugin to indicate it has found a network. At that point, the pushbutton is monitored by the Silicon Labs Device UI plugin to generate various network activities: holding the button for one or more seconds will cause the device to leave the network, short pressing twice will cause it to enter identify mode, short pressing three times will display the network state, and short pressing four times will generate a proactive rejoin.

By default, the application will poll the temperature, humidity, and illuminance sensors and use that data to populate the attributes of the respective clusters. If reporting is set up by another node, the sensors will be read frequently enough to ensure that the reports contain fresh data.

Version 5.7.1

2.45 New Features

2.46 Configuration Changes

2.46.1 Network Creator Security Trust Center Functionality

Users now have the ability to include only distributed network security functionality in the Network Creator Security plugin. There is a plugin option, called Trust Center Support, that specifies whether or not the trust center functionality will be included. By default, this is set to true, to support previous implementations. However, this plugin option can be turned off to save code space for applications that form only distributed security networks. As an example, please see the Z3Light sample application. This application will only form a distributed network, so it does not need the centralized security functionality in the plugin, and therefore it can turn the plugin option off to save flash.

2.46.2 Trust Center Rejoin Policy

The Application Framework V2 trust center code has been updated to ignore trust center rejoins by default when there is a non-unique trust center link key. This is to protect networks against a malicious node performing a trust center rejoin and having the trust center send the network key encrypted with a well know link key. Customers that are building trust center applications with HA or Smart Energy Test security should review their trust center rejoin policy, as upgrading to this release will automatically default the policy to ignoring a device performing a trust center rejoin. Customers with ZigBee 3.0 and Smart Energy Full security trust centers will have their policy defaulted to allow trust center rejoins from devices, since these are more secure given the unique trust center link key for each device on the network. Customers wishing to change the default behavior of their trust center application should review the callback `emberAfSecurityInitCallback` and the new CLI command, `option security set-allow-trust-center-rejoin` (referred to in section 2.61.1 of this document).

2.46.3 Simple Metering Attribute Reporting Status

The Smart Energy 1.2a specification added an attribute to all clusters called Attribute Reporting Status, with code 0xFFFE. This attribute has now been moved to all ZCL clusters, with ZCL6. Before this release, the application framework generated the define `ZCL_ATTRIBUTE_REPORTING_STATUS_ATTRIBUTE_ID` for this attribute code. Now, applications can access this attribute for each cluster with a define of the form `ZCL`, *plus the cluster name (for example,* `SIMPLE_METERING_CLUSTER`*), plus* `REPORTING_STATUS_ATTRIBUTE_ID`.

2.47 Removed APIs

`No APIs were removed in this release.`

2.48 Removed Callbacks

`No callbacks were removed in this release.`

2.49 Removed CLI Commands

`No CLI commands were removed in this release.`

2.50 Removed Plugins

`There were no removed plugins in this release.`

2.51 Changed APIs

`No APIs were changed in this release.`

2.52 Changed Callbacks

There were no callbacks changed in this release.

2.53 Changed CLI Commands

There were a no CLI commands changed in this release.

2.54 Changed Plugins

There were no major changes made to plugins in this release.

2.55 Deprecated APIs

No APIs were deprecated in this release.

2.56 Deprecated Callbacks

No callbacks were deprecated in this release.

2.57 Deprecated CLI Commands

No CLI commands were deprecated in this release.

2.58 Deprecated Plugins

No plugins were deprecated in this release.

2.59 New APIs

No APIs were added in this release.

2.60 New Callbacks

There were no new callbacks added in this release.

2.61 New CLI Commands

There was one significant CLI command added in this release.

2.61.1 Set Trust Center Rejoin Policy

The command option security set-allow-trust-center-rejoin will change the default behavior of a trust center application on whether or not to accept trust center rejoins from a device. This is important because it allows users to test their trust center rejoin policy as they review their security code. Upon issuing the command, the trust center will immediately start allowing or ignoring trust center rejoins from devices in the network. Please see 2.46.2 for more information on this security topic.

2.62 New Plugins

There are no new plugins in this release.

Version 5.7.0

2.63 New Features

2.63.1 ZigBee 3.0 Support

This release contains support for ZigBee 3.0 functionality.

There were some changes made to application framework constants to align them with the naming found with in the ZigBee 3.0 document suite. The macro ZCL_PROFILE_WIDE_COMMAND, which indicated the ZCL frame control bits that are used for global ZCL commands, has been renamed to ZCL_GLOBAL_COMMAND. The former macro still exists in the master application framework header to support legacy code.

2.64 Configuration Changes

2.64.1 Transient Key Timeout Configuration

There is now a way to configure how long a trust center stores a transient key in RAM for a joining device to use before it updates its trust center link key per the Base Device Specification. The global variable can also be configured at runtime with AppBuilder using the Security Core Library plugin for the EM35X framework and the NCP Configuration plugin for the EM35X-EZSP framework.

2.64.2 Application Manufacturer Code

AppBuilder allows users to enter a manufacturer code for their application. In previous releases, if the manufacturer code was not set in the AppBuilder user interface, it would default to the Ember manufacturer code (0x1002). Now, the value will default to 0x0000 if it is not set. Users are recommended to use their own manufacturer ID for their application by setting it in the AppBuilder user interface under the ZCL global tab.

2.64.3 Default Response Policy

The default response policy is a value set in AppBuilder that dictates whether the disable default response bit in the ZCL header is set or cleared by default. The ZCL6 specification takes a stronger position on setting this bit for response-style frames (see section 2.4.1.1.4), so the default response policy value in AppBuilder will now default to CONDITIONAL, meaning the bit will be set on response-style frames. Users wishing to comply with the ZCL6 specification should consider the value of their default response policy in AppBuilder.

2.65 Removed APIs

No APIs were removed in this release.

2.66 Removed Callbacks

No callbacks were removed in this release.

2.67 Removed CLI Commands

No CLI commands were removed in this release.

2.68 Removed Plugins

There was one plugin remove in this release.

2.68.1 Network Commissioner

The Network Commissioner plugin seeked to give users an example of a commissioning state machine per the Base Device Specification. This functionality has been moved to sample applications, in order to save code space. Users wishing to view an example of this functionality should study the ZigBee 3.0 sample applications, as they better exemplify a ZigBee 3.0 commissioning state machine.

2.69 Changed APIs

There was a couple of changes to API's in this release.

2.69.1 emberAfGetDifference

The API emberAfGetDifference now compares data sizes up to 8 bytes on platforms that support the uint64_t primitive type. Images that are built with the HAL_HAS_INT64 precompiler symbol will support this type. Before, the API only supported data sizes of 4 bytes or fewer. There should be no action needed from users as compilers should promote values of size 4 bytes and fewer to 8 byte quantities with the same decimal value.

2.69.2 emberAfPluginNetworkSteeringStart

The API emberAfPluginNetworkSteeringStart will now try to join a network using an install code derived key, the centralized security default key, and the distributed security default key. Before, the plugin was unable to join a distributed or security without the caller having to pass in what kind of network they were looking for. Now, the plugin will try to join all of those network types automatically. Because of this change, the API now takes no arguments.

2.69.3 Finding and Binding

Due to the update to the Find and Bind plugins, see 2.72.2, the names of the finding and binding APIs have changed slightly. The API emberAfPluginFindAndBindTarget has been renamed to emberAfPluginFindAndBindTargetStart. The API emberAfPluginFindAndBindInitiator has been renamed to emberAfPluginFindAndBindInitiatorStart. The functionality of the APIs has stayed the same.

2.70 Changed Callbacks

Please see below for the callbacks that were changed in this release.

2.70.1 Network Steering Callbacks

The Network Steering plugin will now try to join networks with each of the three supported ZigBee 3.0 key types: an install code derived key, the default centralized security key, and the default distributed security key. Because of this, the callback emberAfPluginNetworkSteeringCompleteCallback has been simplified to pass the state at which the plugin completed a successful join. Please see 2.69.2 for more information on this change.

2.70.2 Update TC Link Key Callbacks

The Update TC Link Key plugin has changed the name of its callback emberAfPluginUpdateTcLinkKeyCompleteCallback to emberAfPluginUpdateTcLinkKeyStatusCallback. The new name of the callback more accurately reflects what notification the callback is representing. Previously, the callback would only be called once on completion. Now, the callback notification is more generic, and it may be called multiple times in a link key update process. Furthermore, this

callback has changed slightly to help clients communicate better with the plugin through the callback.

The Update TC Link Key plugin uses the callback emberAfPluginUpdateTcLinkKeyStatusCallback to notify its client of an update in the trust center link key update process. The callback implementation now lets the client return a boolean value to notify the plugin when the client is done receiving key exchange status updates. This allows the client to stop receiving updates from the plugin if it receives an EmberKeyStatus value that it deems an error in the process.

2.70.3 Find and Bind Callbacks

The Find and Bind plugin has been reworked to include support for creating multicast bindings. The application can use the callback emberAfPluginFindAndBindInitiatorBindTargetCallback to tell the plugin which type of binding they wish to create for that target. For this to happen, the callback now takes a binding entry and a pointer to a group name. If the application would like for the binding to be created as multicast, it should change the type member of the binding entry struct to EMBER_MULTICAST_BINDING, enter the group ID in identifier member of the binding entry struct, and optionally write the group name to the pointer. Previous implementations wishing to continue to use unicast bindings should not need to change other than possibly using the new binding entry struct for data on the target.

2.71 Changed CLI Commands

There were a couple of CLI commands changed in this release.

2.71.1 Network Steering CLI

The Network Steering plugin has been improved with runtime configurable options to determine specific parts of the network steering process. Because of this, the plugin network-steering start command now takes one 32-bit integer as a command. This 32-bit integer argument allows the user to control pieces of the network steering process when issuing the command from the command line interface. See network-steering.h for these specific options.

2.71.2 Color Control CLI

The Color Control ZCL CLI commands have been updated with a new format to match the rest of the ZCL CLI commands. Previously, the color control cli commands were of the format zcl colorcontrol. Now, the color control cli commands are of the format zcl color-control.

2.72 Changed Plugins

The following changes have been made to a plugin in this release.

2.72.1 Network Steering

The Network Steering plugin will now try to join networks with each of the three supported ZigBee 3.0 key types: an install code derived key, the default centralized security key, and the default distributed security key. Please see 2.69.2 for more information on this change.

2.72.2 Finding and Binding

The Find and Bind plugin has been split into two plugins for the purpose of code space efficiency. There are two new plugins, Find and Bind Initiator and Find and Bind Target, which contain the functionality for a finding and binding initiator and a finding and binding target, respectively. Upon users opening applications that used the Find and Bind plugin, AppBuilder will enable both of these new plugins, which together make up the old plugin. Users

should decide whether or not they need both sides of the finding and binding process in their application and enable or disable either of the plugins if they wish to save code size. See 2.69.3 for slight changes made to the names of the Find and Bind APIs.

2.73 Deprecated APIs

No APIs were deprecated in this release.

2.74 Deprecated Callbacks

No callbacks were deprecated in this release.

2.75 Deprecated CLI Commands

No CLI commands were deprecated in this release.

2.76 Deprecated Plugins

No plugins were deprecated in this release.

2.77 New APIs

No APIs were added in this release.

2.78 New Callbacks

There was one new callback added in this release.

2.78.1 Network Steering Get Node Type

The network steering plugin now provides the ability for a user to set their node type dynamically at join time. For example, if a user wishes to join as a sleepy end device for one type of network, but an RX on when idle end device for another network, they can use this callback to do that. By default, the callback will perform the behavior that was previously hard-coded in the plugin. Therefore, previous implementations will not need to use this callback if they wish to continue functioning the same way.

2.79 New CLI Commands

There were no new CLI commands added in this release.

2.80 New Plugins

There are no new plugins in this release.

3 Known/Fixed Issues

3.1 Fixed Issues

3.1.1 Fixed Issues in EmberZNet 5.7.3

- Issue 158920: SE 1.2 Calendar Client Description, Interface and Dependencies are not accurate.

3.1.2 Fixed Issues in EmberZNet 5.7.2

- Issue 181942: emberAfPluginOtaClientStackStatusCallback can set a random delay of 0 seconds, unintentionally disabling server discovery
- Issue 178053: End Device Support plugin should provide a means to suppress insecure rejoins if still using well-known ZHA Trust Center Link Key

3.1.3 Fixed Issues in EmberZNet 5.7.1

- Issue 163938: IAR reports broken options when opening EFR32 target workspace created by AppBuilder
- Issue 174602: IAR post build script not generating OTA files for ECC-signed images

3.1.4 Fixed Issues in EmberZNet 5.7.0

- Issue 147652: End device attempts to rejoin forever, regardless of Max Rejoin Attempts threshold setting in End Device Support plugin

3.2 Known Issues

- Issue 159611: ZCL attribute tokens creator codes likely to change if you add or remove one or more attributes. For instance, if you add an attribute whose cluster ID or attribute ID is not the largest numerically, then this attribute gets inserted into a list of creator codes and makes all creator codes after it to be different (shifted).
- Issue 159014: AppBuilder for AFv2 applications limits radio TX power levels to +8 dBm
- Issue 136337: Price Client Plugin does not work properly in SE 1.1b. Workaround: Select an older SE 1.1 spec version or disable Price Client plugin.
- Issue 136274: Node State Cache incorrectly indicates no nodes in trace file.
- Issue 110166: Extended Ember Desktop Decoder's events window to be customizable
- Issue 92566: APS Alarm message always decodes to say missing packets
- Issue 66095: ISA3 Firmware Updates may cause Ember Desktop to hang, requiring a restart of both the ISA3 and Ember Desktop
- Issue 65929: Global channel change via Sniffer Settings dialog doesn't impact EM35x sniffers
- Issue 65898: ISD cannot find JRE on some 64-bit Windows 7 machines
- Issue 65867: Active Scan from Sniffer Settings dialog doesn't accurately report results.
- Issue 65687: Console view renders ASCII 0x81 character incorrectly
- Issue 146795: Added Alarm Cluster functionality to IAS Zone Plugin.
- Issue 126087: Sleepy end device would return NO_LOCAL_RESOURCES when a coordinator initiates key establishment with it in Multi-networking.
- Issue 123399: Non-sleepy endpoint does not always keep correct network parameters on reset in multi-networking.
- Issue 121707: Reporting does not send to a group address.
- Issue 119828: ota-client.c does not use the server EUI64 in Partner Link Key Exchange.
- Issue 117894: Image Block Response does not properly handle the one time block request delay.

- Issue 113213: Request Block Delay value on client is misread by server
- Issue 108582: minBlockRequestPeriod is incorrectly compared to imageBlockRequestMinRequestPeriodSeconds
- Issue 108245: Old Custom Cluster XML is unable to be replaced by new XML
- Issue 101644: Add a callback to the check-in interval for poll control.
- Issue 92147: ZLL Scan Response Should Be Sent at Power 0 rather than last-used power level
- Issue 83798: Image Integrity Tag generation in image builder for an OTA file
- Issue 66944: Duplicate Key Confirm Response message can lock up KE plugin state machine
- Issue 66786: "zcl ota server reload" doesn't properly reload image info when using OTA Simple Storage plugin
- Issue 66785: Messaging Client plugin should differentiate between Cancel Msg command and timed out / replaced message
- Issue 66508: Framework should avoid sending unicast loopback messages with APS security since stack doesn't support this