AN1298: Transitioning from the v1.x to the v2.x Bluetooth Mesh SDK

Bluetooth Mesh Software Development Kit (SDK) v2.x contains a number of changes compared to Bluetooth Mesh SDK v1.x. Many of these changes are due to an underlying framework redesign that results in an improved developer experience within the new Simplicity Studio 5. Projects are now built on a component architecture. Simplicity Studio 5 includes project configuration tools that provide an enhanced level of software component discoverability, configurability, and dependency management. These include a Component Editor, a redesigned GATT configurator, a DCD configurator, and the stack configurator.

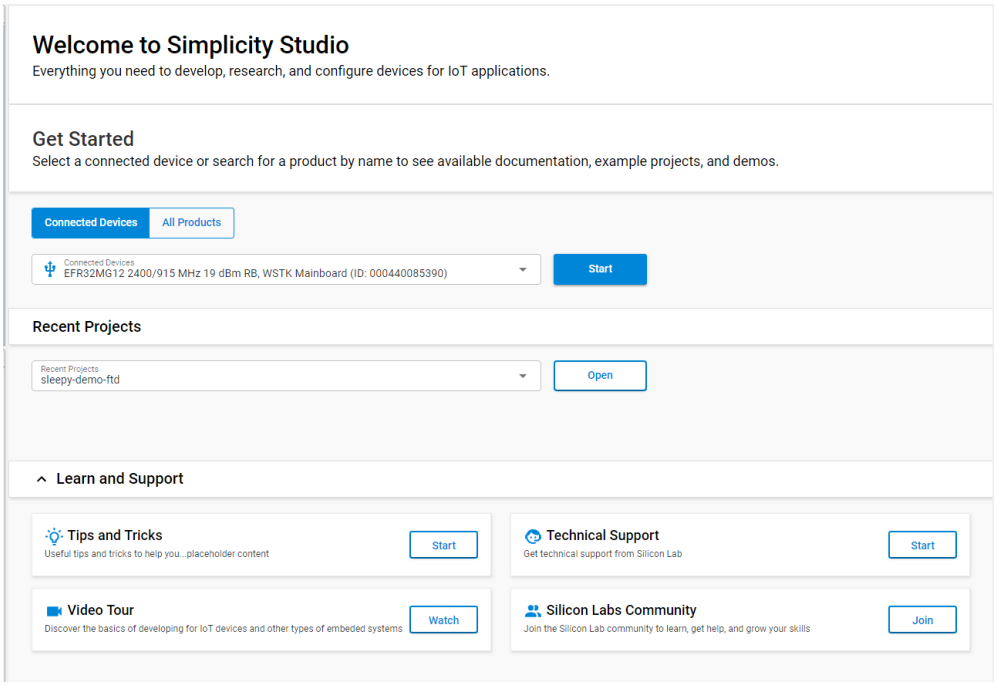
In Bluetooth Mesh SDK v2.x, sample applications have a new software architecture, and the Bluetooth Mesh API is redesigned. Additionally, the stack can now be configured in separate header files, and platform components can be added to the project with the Project Configurator instead of copying and including files manually. While these changes are a result of overall improvements in the SDK and in Simplicity Studio 5, it also means that migrating projects from Bluetooth Mesh SDK v1.x to v2.x is not trivial and has to be done manually. This document explains the steps needed to move your Bluetooth Mesh v1.x project into the v2.x environment.

This document is the Bluetooth Mesh extension of *AN1255: Transitioning from the v2.x to the v3.x Bluetooth® SDK* which contains the general Bluetooth software architecture level information.

KEY POINTS

Reviews difference in:

- Software architecture for Bluetooth Mesh application
- API
- Stack configuration
- New components



1 Introduction

Silicon Labs has introduced both a complete update to its Simplicity Studio tool suite, as well as a new, component-based Gecko Platform architecture. For more information about the changes in Simplicity Studio 5 and the Bluetooth SDK, see *AN1255: Transitioning from the v2.x to the v3.x Bluetooth® SDK*.

Bluetooth Mesh SDK 2.0.0 is released with Gecko Software Development Kit suite version 3.1.0 in Simplicity Studio 5 (SSv5). It is built on top of the component-based Gecko Platform architecture. With this design, Bluetooth Mesh developers benefit the following component-based project configuration features:

- Search and filter to find and discover software components that work with the target device.
- Automatically pull in all component dependencies and initialization code.
- Configurable software components including peripheral inits, drivers, middleware, and stacks.
- All configuration settings in C header files for use outside of Simplicity Studio.
- Configuration validation to alert developers to errors or issues.
- Easily manage all project source using git or other SCM tools.
- Managed migration to future component and SDK versions.
- Simplified transitions from Silicon Labs development kits to custom hardware.

Other features of the SSv5/GSDK v3.x development environment include:

- Project source management options (link to SDK sources or copy all contents to user folder).
- Graphical pin configuration.
- Redesigned Bluetooth Configurator with a fresh UI that is more intuitive for Bluetooth and GATT customization.
- Redesigned Radio Configurator with a fresh UI that is more intuitive for single- and multi-PHY customization.
- Redesigned stack configurator (used to be memory configuration) with a fresh UI that's more intuitive for Bluetooth Mesh application customization.
- Iterative development (configure components, edit sources, compile, debug) using SSv5 configuration tools and third-party IDEs.
- GNU makefiles as a build option.

Other changes are specific to the SDK. Bluetooth Mesh SDK v2.x contains a number of changes compared to Bluetooth Mesh SDK v1.x. The main changes are as follows

1. The project structure of the sample applications has changed. Many autogenerated files and unified configuration files help when adding and configuring software components.
2. The entire project generation is now based on software components. This makes it possible to add functionality to the project with only a click, instead of copying files manually and looking for dependencies.
3. The Bluetooth Mesh API has changed. The API commands and events use a new nomenclature to comply with Silicon Labs standards. Additionally, some new classes and commands are introduced and some of them are removed to make the API more transparent and consistent.
4. The GATT configurator is completely redesigned. The new user interface is more modern, while the generator tool makes it possible to add partial extensions to the GATT database. This means that the GATT database can be easily extended programmatically.
5. The memory configurator is renamed to Bluetooth Mesh stack configurator and separated from the DCD configurator. A Component Configurator tool in SSv5 is available to configure all component parameters, which makes it easy to use predefined values, and to validate custom values.

This document guides you through these changes, and describes the migration steps necessary to move a project from Bluetooth Mesh SDK v1.x to v2.x.

For details about the release, see the release notes provided with the SDK.

2 Project Structure

In Bluetooth Mesh SDK v1.x, a project contains the following folders:

/	Application-specific files
/hardware	Development board configuration files and drivers for external peripherals
/platform	Device configuration files and drivers for the MCU peripherals
/protocol	Bluetooth stack files

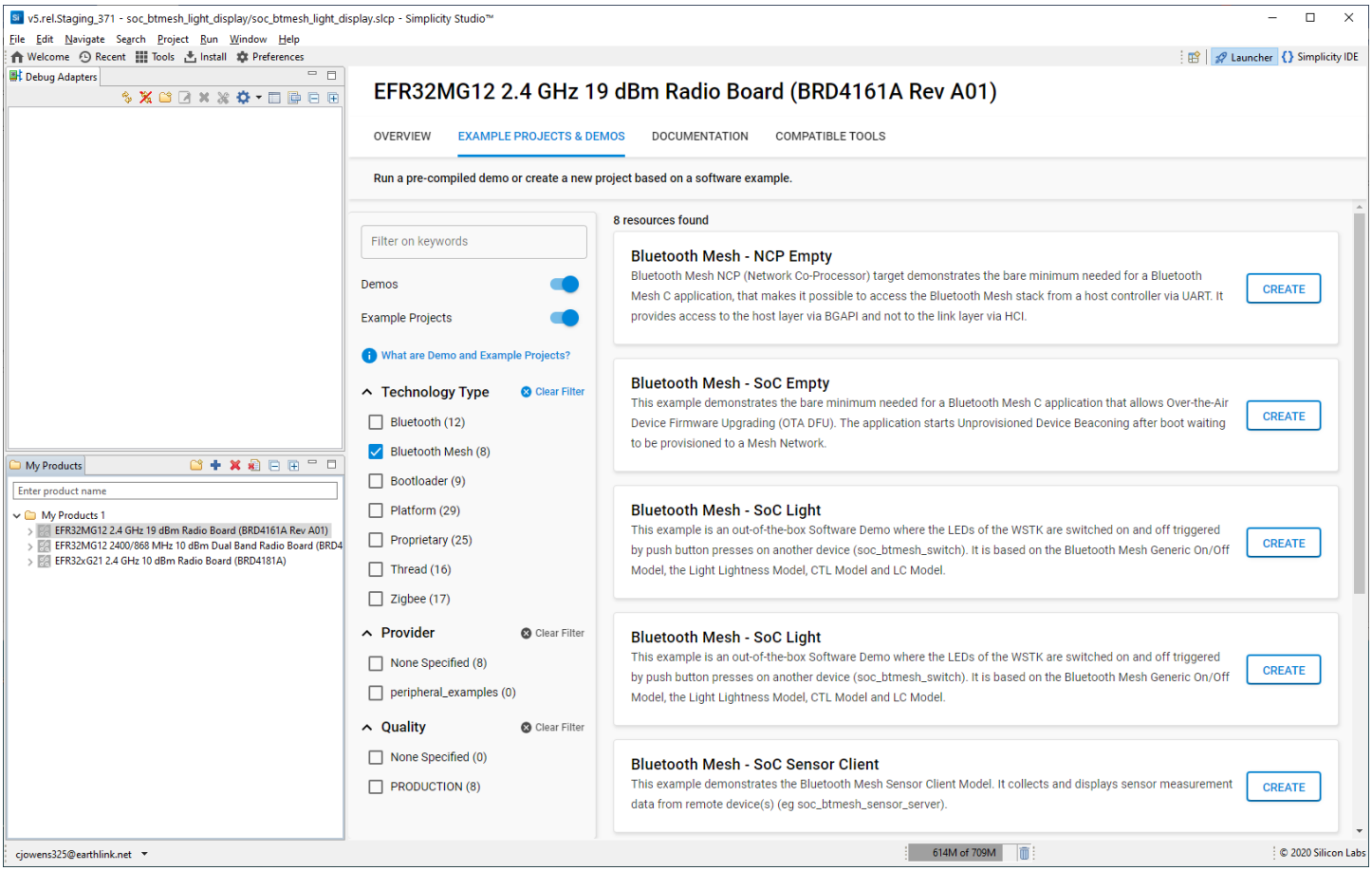
When a new project is generated, a subset of the SDK files (source files, headers, configuration headers) is copied from the SDK folder into the project folders based on hardware type and on the needs of the sample application. Additionally, some files (for example init_mcu.c, hal-config.h, and so on) are generated from templates into the root folder of the project. Again, the output is based on the hardware type and on the needs of the sample application.

In Bluetooth Mesh SDK v2.x auto-generated files and configuration headers are clearly separated from the static SDK files and gathered into separate folders:

/autogen	Automatically-generated files based on the installed software components
/config	Editable configuration files for the software components
/gecko_sdk_3.x.x	
/app	Application-specific files
/hardware	Development board configuration files and drivers for external peripherals
/platform	Device configuration files and drivers for the MCU peripherals
/protocol	Bluetooth stack files
/util	Utilities

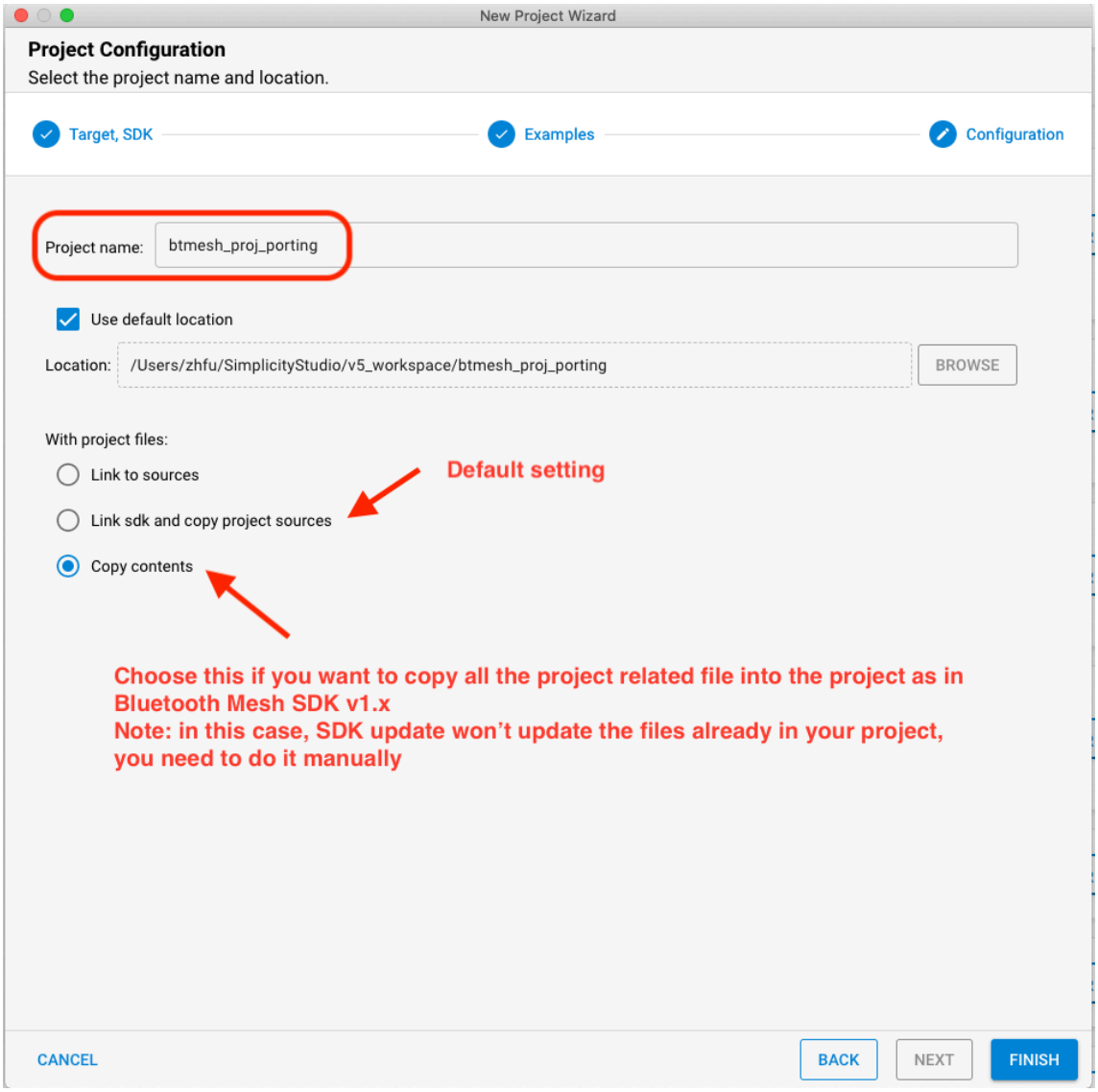
Developers can now easily see which files can be modified by the generator script and it is easy to access all configuration files. This is especially important because in Bluetooth Mesh SDK v2.x many more files are generated by the addition of software components (see Section 3 Software Components).

Due to the new project structure – and due to the new auto-generated files required by the Bluetooth stack – migrating a v1.x project into Bluetooth Mesh SDK v2.x must begin by creating a new project in the new SDK. All Bluetooth Mesh v1.x SDK examples are also in the Bluetooth Mesh v2.x SDK. It is recommended to start from the equivalent or from the Bluetooth Mesh SoC Empty project. Application logic should be pulled into this new project. After you select a compatible part on SSv5's Welcome page, the project can be created from the Part-specific Launcher perspective. The Technology filter makes it easy to find applicable projects.



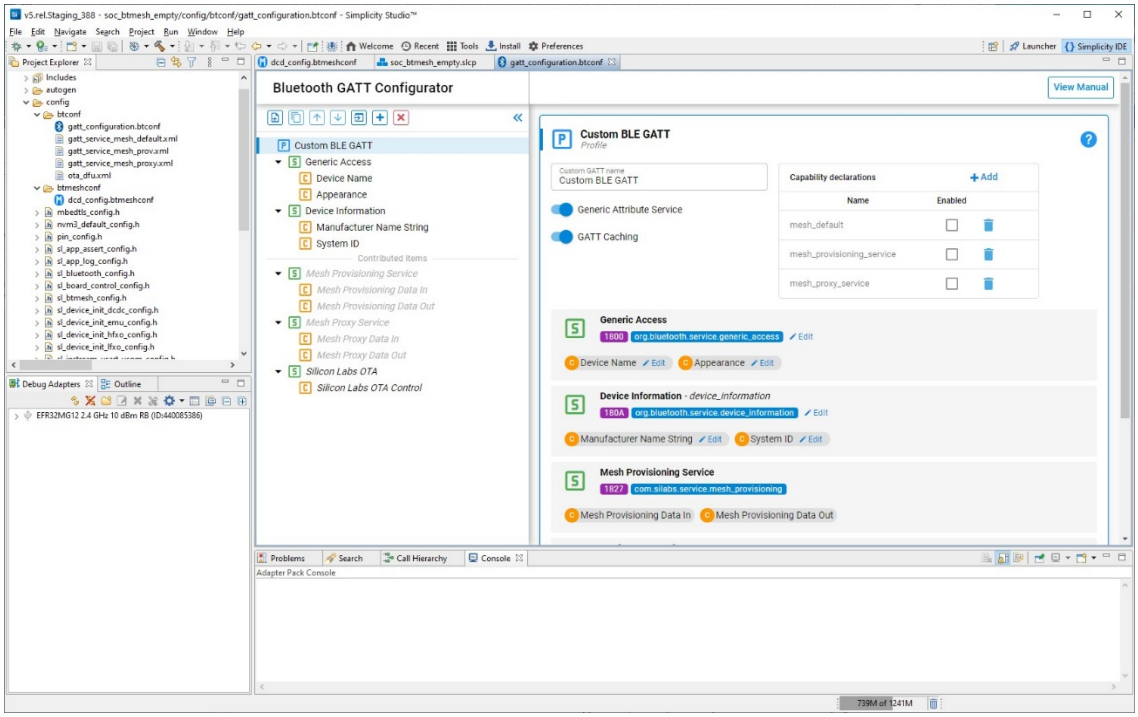
In the project configuration dialog, you can rename the project, change the project location, and define how to handle project files.

Note: In Bluetooth Mesh SDK v1.x all SDK files are copied into the project by default. In Bluetooth Mesh SDK v2.0 SDK files, considered to be static, are linked by default. If you want to version-control your full project, it is recommended to change this setting to “Copy contents” in the Project Configuration wizard, when you create the new project to have all content needed by the project in one folder.

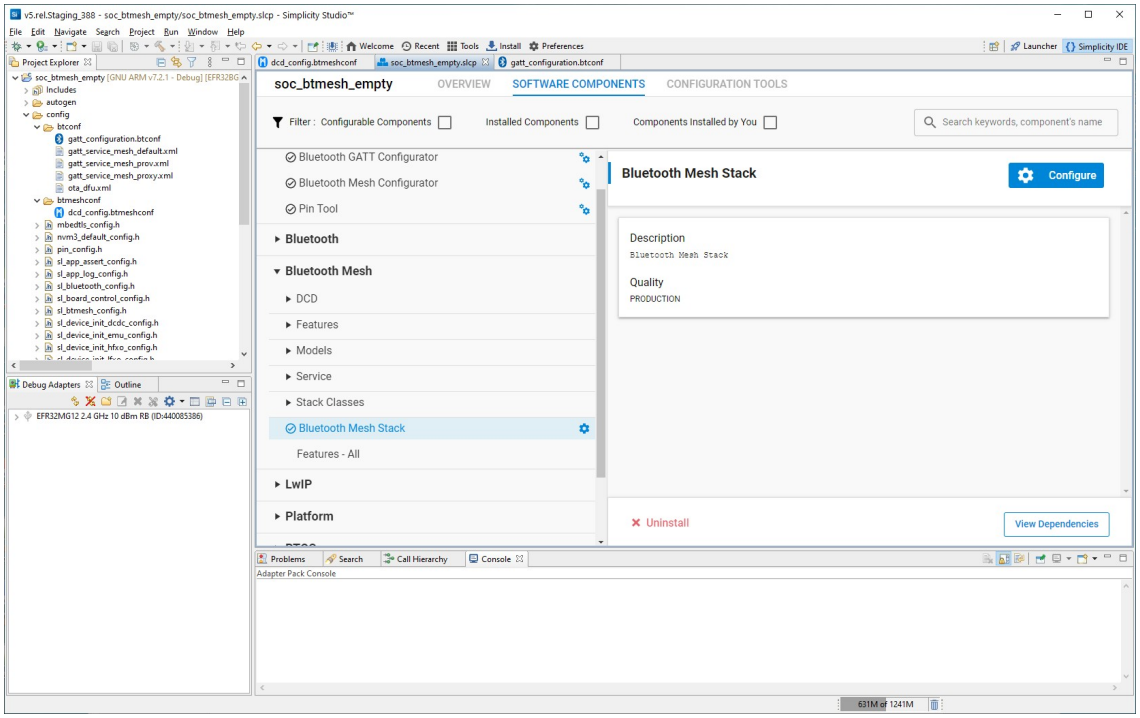


Bluetooth Mesh projects automatically open in the Simplicity IDE's Project Configurator. Three tabs are presented: the GATT Configurator (gatt_configuration.btconf), .the slcp or Project Configurator (<projectname>.slcp), and the Mesh Configurator (dcd_config.btmeshconf). If the example has documentation, the project opens on a readme tab.

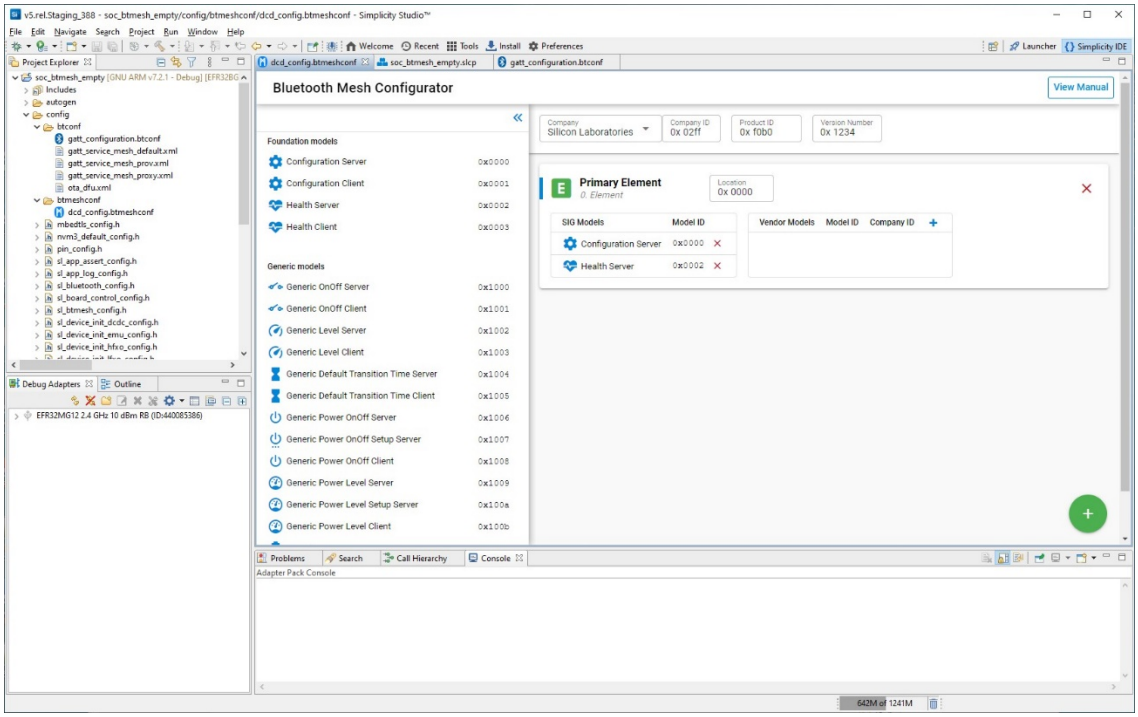
GATT configuration is the same for both Bluetooth and Bluetooth mesh projects. *UG438: GATT Configurator User's Guide for Bluetooth SDK v3.x* describes how to configure the GATT database.



The Project Configurator and its associated Component Editor provide access to components. All the Bluetooth Mesh functionality is provided as components. You can customize projects by installing or uninstalling the components based on the use cases and requirements, and then configuring installed components using the Component Editor. The Bluetooth Mesh Component group shows the components specific to Bluetooth Mesh projects. Components are discussed in more detail in the next section.



The Bluetooth Mesh Configurator provides access to information about a Bluetooth mesh node, the elements it includes, and the supported models. DCD exposes the node information to a configuration client so that it knows the potential functionalities the node supports and based on that can configure the node.



For more details on node configuration using the Project Configurator and Bluetooth Mesh Configurator, see *UG472: Bluetooth® Mesh Node Configurator User's Guide for SDK v2.x*.

3 Components

To add a new software component, for example a new model, in Bluetooth Mesh SDK v1.x the user must:

1. Copy the corresponding SDK files from the SDK folder into the project folder.
2. Copy all the dependencies of the given component into the project folder.
3. Add new include directories to the project settings.

And additionally:

4. Write the initialization code manually in the application.
5. Configure the component manually in the config files.
6. Use the API of the component in the application.

This is quite a cumbersome process, especially when figuring out the dependencies between components.

In Bluetooth Mesh SDK v2.x software components can be added easily by installing them from the Component Library. The installation process will automatically execute the first three steps listed above, and it also modifies the corresponding auto-generated files to integrate the component into the application (*"glue logic"*).

Additionally, the Component Configurator provides the possibility of:

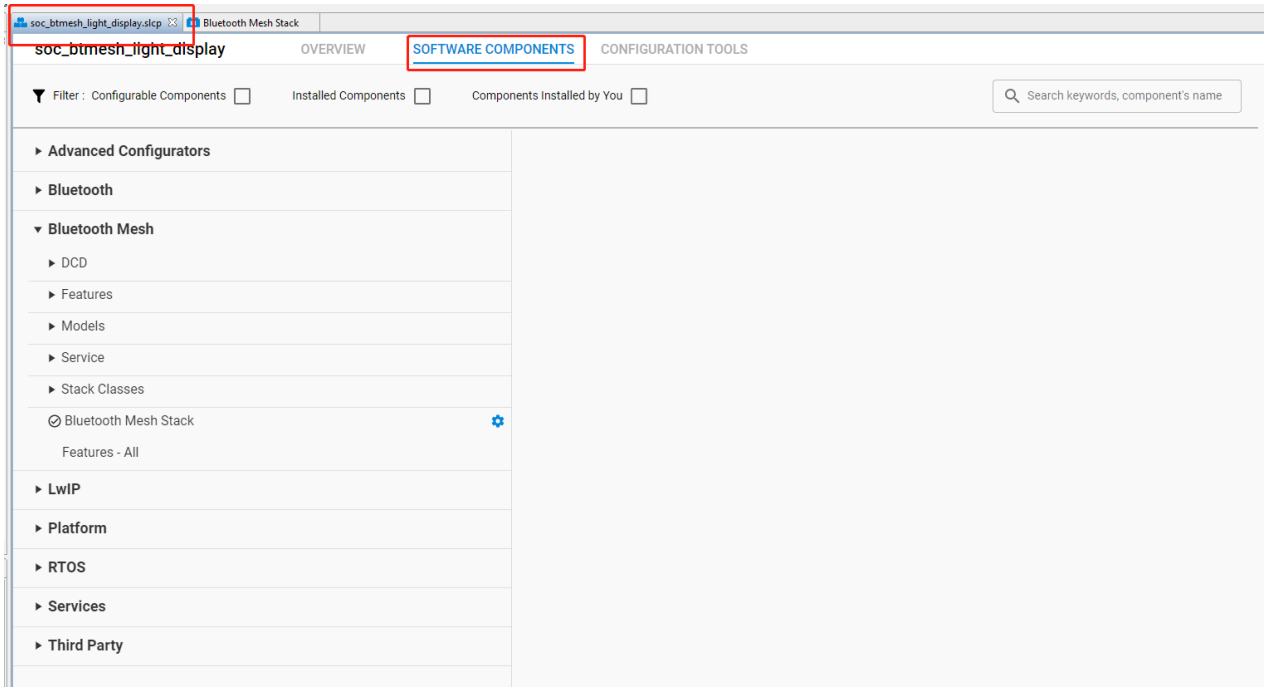
1. Adding an "*init*" type component that initializes the software component
2. Configuring the component with a GUI

Some software components (like OTA DFU) will fully integrate into the application to perform a specific task without the need of any additional code, while other components provide an API to be used in the application. The only task left for the developer is to use the API of those component in the application.

It is important to note that in Bluetooth Mesh SDK v2.x the Bluetooth Mesh stack itself is also just a collection of software components that can be added to and removed from the project.

When migrating a project into Bluetooth Mesh SDK v2.x, start by finding out which functionality can be provided by installing a software component from the Component Library. Although this means you must become familiar with the software components first, it will save time later. Generally speaking, if you would have copied an additional SDK file into your project, you will probably find a software component that solves the integration of that file.

To see the component library, click the <project-name>.slcp tab of your project, and click **Software Components**. A number of filters as well as a keyword search are available to help you explore the various component categories. Note that components for all installed SDKs are presented.



4 Bluetooth Mesh API

The Bluetooth Mesh API changes in Bluetooth Mesh SDK v2.x. The most obvious change is the renaming of all the BGAPI commands and events to align with the unified Silicon Labs coding standard. However, commands have been added and removed, and also new BGAPI classes have been added to make the API more logical and transparent. This section describes all the changes related to the API.

4.1 Functionality Breaks

4.1.1 Provisioning Management

The API for provisioning a device has been changed in order to increase its flexibility to handle different scenarios, such as parallel provisioning or multiple devices, with more ease.

All provisioning events and commands affecting the provisioning of a particular device are linked to a *provisioning session* by UUID of the device as a parameter. This enables provisioning of multiple devices in parallel.

Provisioning flexibility is improved by events that inform the Provisioner of the state of the provisioning session and the capabilities of the device. A simple Provisioner may still choose to allow the stack to select default values but a more complex implementation has fuller control than before.

Previously, the Provisioner would set up provisioning parameters, such as the address allocation for the new device, at the beginning of provisioning without regard to the properties of the device being provisioned. This has been changed to a more event driven flow where the Provisioner can, at certain points, affect how provisioning a device proceeds. Details for the provisioning process are listed below.

Provisioning session control

If the Provisioner needs to control the provisioning process in detail it must call sl_btmesh_prov_set_provisioning_suspend_event() with the status parameter set to 1. This allows the stack to suspend the ongoing provisioning session at stages where the Provisioner could be queried for a decision to continue provisioning, or for a value of a provisioning parameter.

If the Provisioner does not need to control provisioning in detail it can call the function with status parameter set to 0 instead, in which case the stack will use default values and continue provisioning without suspending it.

Starting to provision

The Provisioner first creates a provisioning session by calling sl_btmesh_prov_create_provisioning_session(). The device UUID is used to identify which device the provisioning session relates to at this call and any of the following commands and events.

The Provisioner may, immediately after creating a provisioning session, set the desired provisioning authentication methods for the session by calling sl_btmesh_prov_set_oob_requirements(), or it can defer that until later if it has requested session suspension.

If provisioning over PB-GATT is needed, the Provisioner must establish a LE connection to the device, as before.

The Provisioner can then start provisioning the device by calling either sl_btmesh_prov_provision_adv_device() (for PB-ADV) or sl_btmesh_prov_provision_gatt_device() (for PB-GATT). The stack will then start running the provisioning protocol with the specified device.

Suspension of provisioning session

If the Provisioner requested session control, at the time when Provisioning Capabilities PDU is received from the device being provisioned an sl_btmesh_evt_prov_capabilities() event will be generated, listing the device capabilities and number of elements the device contains. Furthermore, an sl_btmesh_prov_provisioning_suspended() event will be generated to indicate a decision is now needed from the Provisioner

If the Provisioner now wishes to use some particular provisioning authentication method based on device capabilities it can do so by calling sl_btmesh_prov_set_oob_requirements(). Otherwise the stack will make a default choice.

If the Provisioner now wishes to set a non-default Mesh address to the device being provisioned it can do so by calling sl_btmesh_prov_set_device_address(). Otherwise, the stack will allocate an address to the device automatically.

If the Provisioner decides to stop provisioning the device for some reason (for example, the device does not support a desired authentication method), it can do so by calling sl_btmesh_prov_abort_provisioning(). Otherwise, it needs to instruct the stack by calling sl_btmesh_prov_continue_provisioning().

OOB authentication data and public key

If some form of OOB data is involved in the provisioning, the stack will notify the Provisioner when such data is needed, as before.

If the OOB static authentication method or the OOB output authentication method is used when provisioning the device, an sl_btmesh_evt_prov_oob_auth_request() is generated when Provisioner needs to supply authentication data by calling sl_btmesh_prov_send_oob_auth_response() as a response to the event.

If the OOB input authentication method is used when provisioning the device, an sl_btmesh_evt_prov_oob_display_input() event is generated when the Provisioner needs to display authentication data for the user to enter on the device being provisioned.

If an OOB public key is used when provisioning the device, a sl_btmesh_evt_prov_oob_pkey_request() event is generated and the Provisioner must respond with a sl_ntmesh_send_oob_pkey_response() call.

Finalizing a provisioning session

Once provisioning finalizes successfully, an sl_btmesh_evt_prov_device_provisioned() event will be generated. In case of failure, a sl_btmesh_evt_prov_provisioning_failed() event will be generated instead.

4.2 Changes in BGAPI Commands

BGAPI command functions change both their name, to align with Silicon Labs standards, and their structure, to make the error checking and the handling of return values simpler.

In Bluetooth Mesh SDK v2.x BGAPI command function names start with `sl_btmesh_` instead of `gecko_cmd_` used in Bluetooth Mesh SDK v1.x. This means that all function name should be changed according to this rule, when migrating a project from v1.x to v2.x.

No compatibility layer is provided due to the additional changes listed below. However, a “Bluetooth API migration helper” component can be found in the Component Library. If this component is installed, a header file is added to the project that provides verbose compiler errors when an old API call is found in the code. A suggestion for the new API is also present in the error message.

In Bluetooth Mesh SDK v1.x command functions returned a complex structure, providing both error code and return values. In Bluetooth Mesh SDK v2.x only a status code is returned, and the return values are passed back using pointer arguments. If the output of a command contains variable size data, the application needs to give the destination for the data as well as the maximum size of the destination. See an example below.

Command functions in v1.x

```
/* Function */
struct gecko_msg_mesh_config_client_add_appkey_rsp_t *
gecko_cmd_mesh_config_client_add_appkey(uint16 enc_netkey_index,
                                        uint16 server_address,
                                        uint16 appkey_index,
                                        uint16 netkey_index);

/* Response structure */
struct gecko_msg_mesh_config_client_add_appkey_rsp_t
{
       uint16 result;
       uint32 handle;
};
```

Command functions in v2.0

```
sl_status_t sl_btmesh_config_client_add_appkey(uint16_t enc_netkey_index,
                                               uint16_t server_address,
                                               uint16_t appkey_index,
                                               uint16_t netkey_index,
                                               uint32_t *handle);;
```

While for most commands the renaming means only changing `gecko_cmd_` to `sl_btmesh_`, many functions are renamed due to changed functionality, changed API class, or simply to make the API more logical. Furthermore, some API functions are split into multiple ones, and some functions are merged. These name changes are listed in Table 5-1. Changes in the BGAPI Commands.

NOTE: All APIs marked as “deprecated” in the Bluetooth Mesh v1.x have been completely removed from Bluetooth Mesh v2.x. The API change table below does not list them.

Table 5-1. Changes in the BGAPI Commands

API 1.x	API 2.0	Notes
CLASS: mesh_config	CLASS: mesh_config	There is no change to this class except the prefix change.
cmd_mesh_config_client_add_appkey	sl_btmesh_config_client_add_appkey	
cmd_mesh_config_client_add_model_sub	sl_btmesh_config_client_add_model_sub	
cmd_mesh_config_client_add_model_sub_va	sl_btmesh_config_client_add_model_sub_va	
cmd_mesh_config_client_add_netkey	sl_btmesh_config_client_add_netkey	
cmd_mesh_config_client_bind_model	sl_btmesh_config_client_bind_model	
cmd_mesh_config_client_cancel_request	sl_btmesh_config_client_cancel_request	
cmd_mesh_config_client_clear_model_sub	sl_btmesh_config_client_clear_model_sub	
cmd_mesh_config_client_get_beacon	sl_btmesh_config_client_get_beacon	
cmd_mesh_config_client_get_dcd	sl_btmesh_config_client_get_dcd	
cmd_mesh_config_client_get_default_timeout	sl_btmesh_config_client_get_default_timeout	
cmd_mesh_config_client_get_default_ttl	sl_btmesh_config_client_get_default_ttl	
cmd_mesh_config_client_get_friend	sl_btmesh_config_client_get_friend	
cmd_mesh_config_client_get_gatt_proxy	sl_btmesh_config_client_get_gatt_proxy	
cmd_mesh_config_client_get_heartbeat_pub	sl_btmesh_config_client_get_heartbeat_pub	
cmd_mesh_config_client_get_heartbeat_sub	sl_btmesh_config_client_get_heartbeat_sub	
cmd_mesh_config_client_get_identity	sl_btmesh_config_client_get_identity	
cmd_mesh_config_client_get_lpn_polltimeout	sl_btmesh_config_client_get_lpn_polltimeout	
cmd_mesh_config_client_get_model_pub	sl_btmesh_config_client_get_model_pub	
cmd_mesh_config_client_get_network_transmit	sl_btmesh_config_client_get_network_transmit	
cmd_mesh_config_client_get_relay	sl_btmesh_config_client_get_relay	
cmd_mesh_config_client_get_request_status	sl_btmesh_config_client_get_request_status	
cmd_mesh_config_client_list_appkeys	sl_btmesh_config_client_list_appkeys	
cmd_mesh_config_client_list_bindings	sl_btmesh_config_client_list_bindings	
cmd_mesh_config_client_list_netkeys	sl_btmesh_config_client_list_netkeys	
cmd_mesh_config_client_list_subs	sl_btmesh_config_client_list_subs	
cmd_mesh_config_client_remove_appkey	sl_btmesh_config_client_remove_appkey	
cmd_mesh_config_client_remove_model_sub	sl_btmesh_config_client_remove_model_sub	
cmd_mesh_config_client_remove_model_sub_va	sl_btmesh_config_client_remove_model_sub_va	
cmd_mesh_config_client_remove_netkey	sl_btmesh_config_client_remove_netkey	
cmd_mesh_config_client_reset_node	sl_btmesh_config_client_reset_node	
cmd_mesh_config_client_set_beacon	sl_btmesh_config_client_set_beacon	
cmd_mesh_config_client_set_default_timeout	sl_btmesh_config_client_set_default_timeout	
cmd_mesh_config_client_set_default_ttl	sl_btmesh_config_client_set_default_ttl	
cmd_mesh_config_client_set_friend	sl_btmesh_config_client_set_friend	

API 1.x	API 2.0	Notes
cmd_mesh_config_client_set_gatt_proxy	sl_btmesh_config_client_set_gatt_proxy	
cmd_mesh_config_client_set_heartbeat_pub	sl_btmesh_config_client_set_heartbeat_pub	
cmd_mesh_config_client_set_heartbeat_sub	sl_btmesh_config_client_set_heartbeat_sub	
cmd_mesh_config_client_set_identity	sl_btmesh_config_client_set_identity	
cmd_mesh_config_client_set_model_pub	sl_btmesh_config_client_set_model_pub	
cmd_mesh_config_client_set_model_pub_va	sl_btmesh_config_client_set_model_pub_va	
cmd_mesh_config_client_set_model_sub	sl_btmesh_config_client_set_model_sub	
cmd_mesh_config_client_set_model_sub_va	sl_btmesh_config_client_set_model_sub_va	
cmd_mesh_config_client_set_network_transmit	sl_btmesh_config_client_set_network_transmit	
cmd_mesh_config_client_set_relay	sl_btmesh_config_client_set_relay	
cmd_mesh_config_client_unbind_model	sl_btmesh_config_client_unbind_model	
CLASS: mesh_friend	CLASS: mesh_ friend	There is no change to this class except the prefix change.
cmd_mesh_friend_deinit	sl_btmesh_friend_deinit	
cmd_mesh_friend_init	sl_btmesh_friend_init	
CLASS: mesh_generic_client	CLASS: mesh_ generic_client	There is no change to this class except the prefix change.
gecko_cmd_mesh_generic_client_get	sl_btmesh_generic_client_get	
gecko_cmd_mesh_generic_client_set	sl_btmesh_generic_client_set	
gecko_cmd_mesh_generic_client_publish	sl_btmesh_generic_client_publish	
gecko_cmd_mesh_generic_client_get_params	sl_btmesh_generic_client_get_params	
gecko_cmd_mesh_generic_client_init	sl_btmesh_generic_client_init	
gecko_cmd_mesh_generic_client_init_common	sl_btmesh_generic_client_init_common	
gecko_cmd_mesh_generic_client_init_on_off	sl_btmesh_generic_client_init_on_off	
gecko_cmd_mesh_generic_client_init_level	sl_btmesh_generic_client_init_level	
gecko_cmd_mesh_generic_client_init_default_transition_time	sl_btmesh_generic_client_init_default_transition_time	
gecko_cmd_mesh_generic_client_init_power_on_off	sl_btmesh_generic_client_init_power_on_off	
gecko_cmd_mesh_generic_client_init_power_level	sl_btmesh_generic_client_init_power_level	
gecko_cmd_mesh_generic_client_init_battery	sl_btmesh_generic_client_init_battery	
gecko_cmd_mesh_generic_client_init_location	sl_btmesh_generic_client_init_location	
gecko_cmd_mesh_generic_client_init_property	sl_btmesh_generic_client_init_property	
gecko_cmd_mesh_generic_client_init_lightness	sl_btmesh_generic_client_init_lightness	

API 1.x	API 2.0	Notes
gecko_cmd_mesh_generic_client_init_ctl	sl_btmesh_generic_client_init_ctl	
gecko_cmd_mesh_generic_client_init_hsl		Bluetooth Mesh SDK v2.0 doesn't support HSL model yet.
CLASS: mesh_generic_server	CLASS: mesh_ generic_server	There is no change to this class except the prefix change.
gecko_cmd_mesh_generic_server_response	sl_btmesh_generic_server_respond	
gecko_cmd_mesh_generic_server_update	sl_btmesh_generic_server_update	
gecko_cmd_mesh_generic_server_publish	sl_btmesh_generic_server_publish	
gecko_cmd_mesh_generic_server_init	sl_btmesh_generic_server_init	
gecko_cmd_mesh_generic_server_init_common	sl_btmesh_generic_server_init_common	
gecko_cmd_mesh_generic_server_init_on_off	sl_btmesh_generic_server_init_on_off	
gecko_cmd_mesh_generic_server_init_level	sl_btmesh_generic_server_init_level	
gecko_cmd_mesh_generic_server_init_default_transition_time	sl_btmesh_generic_server_init_default_transition_time	
gecko_cmd_mesh_generic_server_init_power_on_off	sl_btmesh_generic_server_init_power_on_off	
gecko_cmd_mesh_generic_server_init_power_level	sl_btmesh_generic_server_init_power_level	
gecko_cmd_mesh_generic_server_init_battery	sl_btmesh_generic_server_init_battery	
gecko_cmd_mesh_generic_server_init_location	sl_btmesh_generic_server_init_location	
gecko_cmd_mesh_generic_server_init_property	sl_btmesh_generic_server_init_property	
gecko_cmd_mesh_generic_server_init_lightness	sl_btmesh_generic_server_init_lightness	
gecko_cmd_mesh_generic_server_init_ctl	sl_btmesh_generic_server_init_ctl	
gecko_cmd_mesh_generic_server_init_hsl		Bluetooth Mesh SDK v2.0 doesn't support HSL model yet.
CLASS: mesh_health_client	CLASS: mesh_health_client	There is no change to this class except the prefix change.
gecko_cmd_mesh_health_client_get	sl_btmesh_health_client_get	
gecko_cmd_mesh_health_client_clear	sl_btmesh_health_client_clear	
gecko_cmd_mesh_health_client_test	sl_btmesh_health_client_test	
gecko_cmd_mesh_health_client_get_period	sl_btmesh_health_client_get_period	
gecko_cmd_mesh_health_client_set_period	sl_btmesh_health_client_set_period	
gecko_cmd_mesh_health_client_get_attention	sl_btmesh_health_client_get_attention	
gecko_cmd_mesh_health_client_set_attention	sl_btmesh_health_client_set_attention	
CLASS: mesh_health_server	CLASS: mesh_health_server	There is no change to this class except the prefix change.

API 1.x	API 2.0	Notes
gecko_cmd_mesh_health_server_set_fault	sl_btmesh_health_server_set_fault	
gecko_cmd_mesh_health_server_clear_fault	sl_btmesh_health_server_clear_fault	
gecko_cmd_mesh_health_server_test_response	sl_btmesh_health_server_send_test_response	
CLASS s: mesh_lc_client	CLASS: mesh_lc_client	There is no change to this class except the prefix change.
gecko_cmd_mesh_lc_client_init	sl_btmesh_lc_client_init	
gecko_cmd_mesh_lc_client_get_mode	sl_btmesh_lc_client_get_mode	
gecko_cmd_mesh_lc_client_set_mode	sl_btmesh_lc_client_set_mode	
gecko_cmd_mesh_lc_client_get_om	sl_btmesh_lc_client_get_om	
gecko_cmd_mesh_lc_client_set_om	sl_btmesh_lc_client_set_om	
gecko_cmd_mesh_lc_client_get_light_onoff	sl_btmesh_lc_client_get_light_onoff	
gecko_cmd_mesh_lc_client_set_light_onoff	sl_btmesh_lc_client_set_light_onoff	
gecko_cmd_mesh_lc_client_get_property	sl_btmesh_lc_client_get_property	
gecko_cmd_mesh_lc_client_set_property	sl_btmesh_lc_client_set_property	
CLASS: mesh_lc_server	CLASS: mesh_lc_server	There is no change to this class except the prefix change.
gecko_cmd_mesh_lc_server_init	sl_btmesh_lc_server_init	
gecko_cmd_mesh_lc_server_deinit	sl_btmesh_lc_server_deinit	
gecko_cmd_mesh_lc_server_update_mode	sl_btmesh_lc_server_update_mode	
gecko_cmd_mesh_lc_server_update_om	sl_btmesh_lc_server_update_om	
gecko_cmd_mesh_lc_server_update_light_onoff	sl_btmesh_lc_server_update_light_onoff	
gecko_cmd_mesh_lc_server_init_all_properties	sl_btmesh_lc_server_init_all_properties	
gecko_cmd_mesh_lc_server_set_publish_mask	sl_btmesh_lc_server_set_publish_mask	
gecko_cmd_mesh_lc_server_set_regulator_interval	sl_btmesh_lc_server_set_regulator_interval	
gecko_cmd_mesh_lc_server_set_event_mask	sl_btmesh_lc_server_set_event_mask	
gecko_cmd_mesh_lc_server_get_lc_state	sl_btmesh_lc_server_get_lc_state	
CLASS: mesh_lc_setup_server	CLASS: mesh_lc_setup_server	There is no change to this class except the prefix change.
gecko_cmd_mesh_lc_setup_server_update_property	sl_btmesh_lc_setup_server_update_property	
CLASS: mesh_lpn	CLASS: mesh_lpn	
gecko_cmd_mesh_lpn_init	sl_btmesh_lpn_init	
gecko_cmd_mesh_lpn_deinit	sl_btmesh_lpn_deinit	
gecko_cmd_mesh_lpn_establish_friendship	sl_btmesh_lpn_establish_friendship	

API 1.x	API 2.0	Notes
gecko_cmd_mesh_lpn_poll	sl_btmesh_lpn_poll	
gecko_cmd_mesh_lpn_terminate_friendship	sl_btmesh_lpn_terminate_friendship	
gecko_cmd_mesh_lpn_config	sl_btmesh_lpn_config	
CLASS: mesh_node	CLASS: mesh_node	
gecko_cmd_mesh_node_init	sl_btmesh_node_init	
gecko_cmd_mesh_node_start_unprov_beaconing	sl_btmesh_node_start_unprov_beaconing	
gecko_cmd_mesh_node_stop_unprov_beaconing	sl_btmesh_node_stop_unprov_beaconing	
gecko_cmd_mesh_node_rssi	sl_btmesh_node_get_rssi	
gecko_cmd_mesh_node_input_oob_request_rsp	sl_btmesh_node_send_input_oob_request_response	
gecko_cmd_mesh_node_get_uuid	sl_btmesh_node_get_uuid	
gecko_cmd_mesh_node_set_provisioning_data	sl_btmesh_node_set_provisioning_data	
gecko_cmd_mesh_node_init_oob	sl_btmesh_node_init_oob	
gecko_cmd_mesh_node_set_ivrecovery_mode	sl_btmesh_node_set_ivrecovery_mode	
gecko_cmd_mesh_node_get_ivrecovery_mode	sl_btmesh_node_get_ivrecovery_mode	
gecko_cmd_mesh_node_set_adv_event_filter	sl_btmesh_node_set_adv_event_filter	
gecko_cmd_mesh_node_get_statistics	sl_btmesh_node_get_statistics	
gecko_cmd_mesh_node_clear_statistics	sl_btmesh_node_clear_statistics	
gecko_cmd_mesh_node_set_net_relay_delay	sl_btmesh_node_set_net_relay_delay	
gecko_cmd_mesh_node_get_net_relay_delay	sl_btmesh_node_get_net_relay_delay	
gecko_cmd_mesh_node_get_ivupdate_state	sl_btmesh_node_get_ivupdate_state	
gecko_cmd_mesh_node_request_ivupdate	sl_btmesh_node_request_ivupdate	
gecko_cmd_mesh_node_get_seq_remaining	sl_btmesh_node_get_seq_remaining	
gecko_cmd_mesh_node_save_replay_protection_list	sl_btmesh_node_save_replay_protection_list	
gecko_cmd_mesh_node_set_uuid	sl_btmesh_node_set_uuid	
gecko_cmd_mesh_node_get_element_address	sl_btmesh_node_get_element_address	
gecko_cmd_mesh_node_static_oob_request_rsp	sl_btmesh_node_send_static_oob_request_response	
gecko_cmd_mesh_node_reset	sl_btmesh_node_reset	
gecko_cmd_mesh_node_set_beacon_reporting	sl_btmesh_node_set_beacon_reporting	
gecko_cmd_mesh_node_set_iv_update_age	sl_btmesh_node_set_iv_update_age	
gecko_cmd_mesh_node_set_model_option	sl_btmesh_node_set_model_option	

API 1.x	API 2.0	Notes
	sl_btmesh_node_get_replay_protection_list_status	
	sl_btmesh_node_get_key_count	
	sl_btmesh_node_get_key	
	sl_btmesh_node_get_networks	
	sl_btmesh_node_get_element_seqnum	
	sl_btmesh_node_get_local_dcd	
	sl_btmesh_node_erase_mesh_nvm	
CLASS: mesh_prov	CLASS: mesh_prov	
gecko_cmd_mesh_prov_init	sl_btmesh_prov_init	
gecko_cmd_mesh_prov_scan_unprov_beacons	sl_btmesh_prov_scan_unprov_beacons	
gecko_cmd_mesh_prov_provision_device	sl_btmesh_prov_provision_adv_device	
gecko_cmd_mesh_prov_create_network	sl_btmesh_prov_create_network	
gecko_cmd_mesh_prov_create_appkey	sl_btmesh_prov_create_appkey	
gecko_cmd_mesh_prov_oob_pkey_rsp	sl_btmesh_prov_send_oob_pkey_response	
gecko_cmd_mesh_prov_oob_auth_rsp	sl_btmesh_prov_send_oob_auth_response	
gecko_cmd_mesh_prov_set_oob_requirements	sl_btmesh_prov_set_oob_requirements	
gecko_cmd_mesh_prov_key_refresh_start	sl_btmesh_prov_start_key_refresh	
gecko_cmd_mesh_prov_get_key_refresh_blacklist	sl_btmesh_prov_get_key_refresh_exclusion	
gecko_cmd_mesh_prov_set_key_refresh_blacklist	sl_btmesh_prov_set_key_refresh_exclusion	
gecko_cmd_mesh_prov_provision_gatt_device	sl_btmesh_prov_provision_gatt_device	
gecko_cmd_mesh_prov_ddb_get	sl_btmesh_prov_get_ddb_entry	
gecko_cmd_mesh_prov_ddb_delete	sl_btmesh_prov_delete_ddb_entry	
gecko_cmd_mesh_prov_ddb_add	sl_btmesh_prov_add_ddb_entry	
gecko_cmd_mesh_prov_ddb_list_devices	sl_btmesh_prov_list_ddb_entries	
gecko_cmd_mesh_prov_provision_device_with_address	sl_btmesh_prov_set_device_address	When provisioning is suspended, use this command to set the address
gecko_cmd_mesh_prov_provision_gatt_device_with_address	sl_btmesh_prov_set_device_address	When provisioning is suspended, use this command to set the address
gecko_cmd_mesh_prov_initialize_network	sl_btmesh_prov_initialize_network	
gecko_cmd_mesh_prov_get_key_refresh_appkey_blacklist	sl_btmesh_prov_get_key_refresh_appkey_exclusion	
gecko_cmd_mesh_prov_set_key_refresh_appkey_blacklist	sl_btmesh_prov_set_key_refresh_appkey_exclusion	
gecko_cmd_mesh_prov_stop_scan_unprov_beacons	sl_btmesh_prov_stop_scan_unprov_beacons	
gecko_cmd_mesh_prov_ddb_update_netkey_index	sl_btmesh_prov_update_device_netkey_index	
gecko_cmd_mesh_prov_key_refresh_suspend	sl_btmesh_prov_suspend_key_refresh	
gecko_cmd_mesh_prov_key_refresh_resume	sl_btmesh_prov_resume_key_refresh	

API 1.x	API 2.0	Notes
gecko_cmd_mesh_prov_get_key_refresh_phase	sl_btmesh_prov_get_key_refresh_phase	
gecko_cmd_mesh_prov_key_refresh_start_from_phase	sl_btmesh_prov_start_key_refresh_from_phase	
gecko_cmd_mesh_prov_flush_key_refresh_state	sl_btmesh_prov_flush_key_refresh_state	
	sl_btmesh_prov_create_provisioning_session	Users have to create the provisioning session before configuring or start the provisioning
	sl_btmesh_prov_set_provisioning_suspend_event	These APIs give more flexibility in provisioning, now it can suspend and users could use callbacks to feed configuration data during provisioning.
	sl_btmesh_prov_continue_provisioning	
	sl_btmesh_prov_abort_provisioning	
	sl_btmesh_prov_test_identity	API for testing if the node identify matches
CLASS: mesh_proxy	CLASS: mesh_proxy	
gecko_cmd_mesh_proxy_connect	sl_btmesh_proxy_connect	
gecko_cmd_mesh_proxy_disconnect	sl_btmesh_proxy_disconnect	
gecko_cmd_mesh_proxy_set_filter_type	sl_btmesh_proxy_set_filter_type	
gecko_cmd_mesh_proxy_allow	sl_btmesh_proxy_allow	
gecko_cmd_mesh_proxy_deny	sl_btmesh_proxy_deny	
CLASS: mesh_scene_client	CLASS: mesh_scene_client	
gecko_cmd_mesh_scene_client_init	sl_btmesh_scene_client_init	
gecko_cmd_mesh_scene_client_get	sl_btmesh_scene_client_get	
gecko_cmd_mesh_scene_client_get_register	sl_btmesh_scene_client_get_register	
gecko_cmd_mesh_scene_client_recall	sl_btmesh_scene_client_recall	
gecko_cmd_mesh_scene_client_store	sl_btmesh_scene_client_store	
gecko_cmd_mesh_scene_client_delete	sl_btmesh_scene_client_delete	
CLASS: mesh_scene_server	CLASS: mesh_scene_server	
gecko_cmd_mesh_scene_server_init	sl_btmesh_scene_server_init	
gecko_cmd_mesh_scene_server_deinit	sl_btmesh_scene_server_deinit	
gecko_cmd_mesh_scene_server_reset_register	sl_btmesh_scene_server_reset_register	
CLASS: mesh_scene_setup_server	CLASS: mesh_scene_setup_server	
gecko_cmd_mesh_scene_setup_server_init	sl_btmesh_scene_setup_server_init	
CLASS: mesh_scheduler_client	CLASS: mesh_scheduler_client	
gecko_cmd_mesh_scheduler_client_init	sl_btmesh_scheduler_client_init	
gecko_cmd_mesh_scheduler_client_deinit	sl_btmesh_scheduler_client_deinit	
gecko_cmd_mesh_scheduler_client_get	sl_btmesh_scheduler_client_get	
gecko_cmd_mesh_scheduler_client_get_action	sl_btmesh_scheduler_client_get_action	

API 1.x	API 2.0	Notes
gecko_cmd_mesh_scheduler_client_set_action	sl_btmesh_scheduler_client_set_action	
CLASS: mesh_scheduler_server	CLASS: mesh_scheduler_server	
gecko_cmd_mesh_scheduler_server_init	sl_btmesh_scheduler_server_init	
gecko_cmd_mesh_scheduler_server_deinit	sl_btmesh_scheduler_server_deinit	
gecko_cmd_mesh_scheduler_server_get	sl_btmesh_scheduler_server_get	
gecko_cmd_mesh_scheduler_server_get_action	sl_btmesh_scheduler_server_get_action	
gecko_cmd_mesh_scheduler_server_set_action	sl_btmesh_scheduler_server_set_action	
CLASS: mesh_sensor_client	CLASS: mesh_sensor_client	
gecko_cmd_mesh_sensor_client_init	sl_btmesh_sensor_client_init	
gecko_cmd_mesh_sensor_client_deinit	sl_btmesh_sensor_client_deinit	
gecko_cmd_mesh_sensor_client_get_descriptor	sl_btmesh_sensor_client_get_descriptor	
gecko_cmd_mesh_sensor_client_get	sl_btmesh_sensor_client_get	
gecko_cmd_mesh_sensor_client_get_column	sl_btmesh_sensor_client_get_column	
gecko_cmd_mesh_sensor_client_get_series	sl_btmesh_sensor_client_get_series	
gecko_cmd_mesh_sensor_client_get_cadence	sl_btmesh_sensor_client_get_cadence	
gecko_cmd_mesh_sensor_client_set_cadence	sl_btmesh_sensor_client_set_cadence	
gecko_cmd_mesh_sensor_client_get_settings	sl_btmesh_sensor_client_get_settings	
gecko_cmd_mesh_sensor_client_get_setting	sl_btmesh_sensor_client_get_setting	
gecko_cmd_mesh_sensor_client_set_setting	sl_btmesh_sensor_client_set_setting	
CLASS: mesh_sensor_server	CLASS: mesh_sensor_server	
gecko_cmd_mesh_sensor_server_init	sl_btmesh_sensor_server_init	
gecko_cmd_mesh_sensor_server_deinit	sl_btmesh_sensor_server_deinit	
gecko_cmd_mesh_sensor_server_send_descriptor_status	sl_btmesh_sensor_server_send_descriptor_status	
gecko_cmd_mesh_sensor_server_send_status	sl_btmesh_sensor_server_send_status	
gecko_cmd_mesh_sensor_server_send_column_status	sl_btmesh_sensor_server_send_column_status	
gecko_cmd_mesh_sensor_server_send_series_status	sl_btmesh_sensor_server_send_series_status	
CLASS: mesh_sensor_setup_server	CLASS: mesh_sensor_setup_server	
gecko_cmd_mesh_sensor_setup_server_send_cadence_status	sl_btmesh_sensor_setup_server_send_cadence_status	

API 1.x	API 2.0	Notes
gecko_cmd_mesh_sensor_setup_server_send_settings_status	sl_btmesh_sensor_setup_server_send_settings_status	
gecko_cmd_mesh_sensor_setup_server_send_setting_status	sl_btmesh_sensor_setup_server_send_setting_status	
CLASS: mesh_test	CLASS: mesh_test	
gecko_cmd_mesh_test_get_nettx	sl_btmesh_test_get_nettx	
gecko_cmd_mesh_test_set_nettx	sl_btmesh_test_set_nettx	
gecko_cmd_mesh_test_get_relay	sl_btmesh_test_get_relay	
gecko_cmd_mesh_test_set_relay	sl_btmesh_test_set_relay	
gecko_cmd_mesh_test_set_adv_scan_params	sl_btmesh_test_set_adv_scan_params	
gecko_cmd_mesh_test_set_ivupdate_test_mode	sl_btmesh_test_set_ivupdate_test_mode	
gecko_cmd_mesh_test_get_ivupdate_test_mode	sl_btmesh_test_get_ivupdate_test_mode	
gecko_cmd_mesh_test_set_segment_send_delay	sl_btmesh_test_set_segment_send_delay	
gecko_cmd_mesh_test_set_ivupdate_state	sl_btmesh_test_set_ivupdate_state	
gecko_cmd_mesh_test_send_beacons	sl_btmesh_test_send_beacons	
gecko_cmd_mesh_test_bind_local_model_app	sl_btmesh_test_bind_local_model_app	
gecko_cmd_mesh_test_unbind_local_model_app	sl_btmesh_test_unbind_local_model_app	
gecko_cmd_mesh_test_add_local_model_sub	sl_btmesh_test_add_local_model_sub	
gecko_cmd_mesh_test_del_local_model_sub	sl_btmesh_test_remove_local_model_sub	
gecko_cmd_mesh_test_add_local_model_sub_va	sl_btmesh_test_add_local_model_sub_va	
gecko_cmd_mesh_test_del_local_model_sub_va	sl_btmesh_test_remove_local_model_sub_va	
gecko_cmd_mesh_test_get_local_model_sub	sl_btmesh_test_get_local_model_sub	
gecko_cmd_mesh_test_set_local_model_pub	sl_btmesh_test_set_local_model_pub	
gecko_cmd_mesh_test_set_local_model_pub_va	sl_btmesh_test_set_local_model_pub_va	
gecko_cmd_mesh_test_get_local_model_pub	sl_btmesh_test_get_local_model_pub	
gecko_cmd_mesh_test_set_local_heartbeat_subscription	sl_btmesh_test_set_local_heartbeat_subscription	
gecko_cmd_mesh_test_get_local_heartbeat_subscription	sl_btmesh_test_get_local_heartbeat_subscription	
gecko_cmd_mesh_test_get_local_heartbeat_publication	sl_btmesh_test_get_local_heartbeat_publication	
gecko_cmd_mesh_test_set_local_heartbeat_publication	sl_btmesh_test_set_local_heartbeat_publication	
gecko_cmd_mesh_test_set_local_config	sl_btmesh_test_set_local_config	
gecko_cmd_mesh_test_get_local_config	sl_btmesh_test_get_local_config	

API 1.x	API 2.0	Notes
gecko_cmd_mesh_test_add_local_key	sl_btmesh_test_add_local_key	
gecko_cmd_mesh_test_del_local_key	sl_btmesh_test_remove_local_key	
gecko_cmd_mesh_test_update_local_key	sl_btmesh_test_update_local_key	
gecko_cmd_mesh_test_set_sar_config	sl_btmesh_test_set_sar_config	
gecko_cmd_mesh_test_get_element_seqnum		
gecko_cmd_mesh_test_set_adv_bearer_state	sl_btmesh_test_set_adv_bearer_state	
gecko_cmd_mesh_test_get_key_count		Moved to mesh_node class
gecko_cmd_mesh_test_get_key		
gecko_cmd_mesh_test_prov_get_device_key	sl_btmesh_test_prov_get_device_key	
gecko_cmd_mesh_test_prov_prepare_key_refresh	sl_btmesh_test_prov_prepare_key_refresh	
gecko_cmd_mesh_test_cancel_segmented_tx	sl_btmesh_test_cancel_segmented_tx	
gecko_cmd_mesh_test_set_iv_index	sl_btmesh_test_set_iv_index	
gecko_cmd_mesh_test_set_element_seqnum	sl_btmesh_test_set_element_seqnum	
gecko_cmd_mesh_test_set_model_option	sl_btmesh_test_set_model_option	
gecko_cmd_mesh_test_get_local_model_app_bindings	sl_btmesh_test_get_local_model_app_bindings	
gecko_cmd_mesh_test_get_replay_protection_list_entry	sl_btmesh_test_get_replay_protection_list_entry	
gecko_cmd_mesh_test_clear_replay_protection_list_entry	sl_btmesh_test_clear_replay_protection_list_entry	
CLASS: mesh_time_client	CLASS: mesh_time_client	
gecko_cmd_mesh_time_client_init	sl_btmesh_time_client_init	
gecko_cmd_mesh_time_client_deinit	sl_btmesh_time_client_deinit	
gecko_cmd_mesh_time_client_get_time	sl_btmesh_time_client_get_time	
gecko_cmd_mesh_time_client_set_time	sl_btmesh_time_client_set_time	
gecko_cmd_mesh_time_client_get_time_zone	sl_btmesh_time_client_get_time_zone	
gecko_cmd_mesh_time_client_set_time_zone	sl_btmesh_time_client_set_time_zone	
gecko_cmd_mesh_time_client_get_tai_utc_delta	sl_btmesh_time_client_get_tai_utc_delta	
gecko_cmd_mesh_time_client_set_tai_utc_delta	sl_btmesh_time_client_set_tai_utc_delta	
gecko_cmd_mesh_time_client_get_time_role	sl_btmesh_time_client_get_time_role	
gecko_cmd_mesh_time_client_set_time_role	sl_btmesh_time_client_set_time_role	
CLASS: mesh_time_server	CLASS: mesh_time_server	
gecko_cmd_mesh_time_server_init	sl_btmesh_time_server_init	
gecko_cmd_mesh_time_server_deinit	sl_btmesh_time_server_deinit	
gecko_cmd_mesh_time_server_get_time	sl_btmesh_time_server_get_time	

API 1.x	API 2.0	Notes
gecko_cmd_mesh_time_server_set_time	sl_btmesh_time_server_set_time	
gecko_cmd_mesh_time_server_get_time_zone_offset_new	sl_btmesh_time_server_get_time_zone_offset_new	
gecko_cmd_mesh_time_server_set_time_zone_offset_new	sl_btmesh_time_server_set_time_zone_offset_new	
gecko_cmd_mesh_time_server_get_tai_utc_delta_new	sl_btmesh_time_server_get_tai_utc_delta_new	
gecko_cmd_mesh_time_server_set_tai_utc_delta_new	sl_btmesh_time_server_set_tai_utc_delta_new	
gecko_cmd_mesh_time_server_get_time_role	sl_btmesh_time_server_get_time_role	
gecko_cmd_mesh_time_server_set_time_role	sl_btmesh_time_server_set_time_role	
gecko_cmd_mesh_time_server_get_datetime	sl_btmesh_time_server_get_datetime	
CLASS: mesh_vendor_model	CLASS: mesh_vendor_model	
gecko_cmd_mesh_vendor_model_send	sl_btmesh_vendor_model_send	
gecko_cmd_mesh_vendor_model_set_publication	sl_btmesh_vendor_model_set_publication	
gecko_cmd_mesh_vendor_model_clear_publication	sl_btmesh_vendor_model_clear_publication	
gecko_cmd_mesh_vendor_model_publish	sl_btmesh_vendor_model_publish	
gecko_cmd_mesh_vendor_model_init	sl_btmesh_vendor_model_init	
gecko_cmd_mesh_vendor_model_deinit	sl_btmesh_vendor_model_deinit	

4.3 Changes in BGAPI events

Just like BGAPI commands, BGAPI events are also renamed. In Bluetooth Mesh SDK v2.x event IDs start with `sl_btmesh_evt_` instead of `gecko_evt_`. Similarly, the event struct types start with `sl_btmesh_evt_` instead of `gecko_msg_evt`, although these types are rarely referenced in the application.

Additionally, some events are renamed because of changing BGAPI class. Table 5-2 contains all the events that have more than just the name change.

Table 5-2. Changes in the BGAPI events

API 1.x	API 2.0	Notes
CLASS: mesh_config	CLASS: mesh_config	There is no change to this class except the prefix change.
gecko_evt_mesh_config_client_appkey_list_id struct gecko_msg_mesh_config_client_appkey_list_evt_t	sl_btmesh_evt_config_client_appkey_list_id sl_btmesh_evt_config_client_appkey_list_t	
gecko_evt_mesh_config_client_request_modified_id gecko_msg_mesh_config_client_request_modified_evt_t	sl_btmesh_evt_config_client_request_modified_id sl_btmesh_evt_config_client_request_modified_t	
gecko_evt_mesh_config_client_netkey_status_id gecko_msg_mesh_config_client_netkey_status_evt_t	sl_btmesh_evt_config_client_netkey_status_id sl_btmesh_evt_config_client_netkey_status_t	
gecko_evt_mesh_config_client_netkey_list_id gecko_msg_mesh_config_client_netkey_list_evt_t	sl_btmesh_evt_config_client_netkey_list_id sl_btmesh_evt_config_client_netkey_list_t	
gecko_evt_mesh_config_client_netkey_list_end_id gecko_msg_mesh_config_client_netkey_list_end_evt_t	sl_btmesh_evt_config_client_netkey_list_end_id sl_btmesh_evt_config_client_netkey_list_end_t	
gecko_evt_mesh_config_client_appkey_status_id gecko_msg_mesh_config_client_appkey_status_evt_t	sl_btmesh_evt_config_client_appkey_status_id sl_btmesh_evt_config_client_appkey_status_t	
gecko_evt_mesh_config_client_appkey_list_id gecko_msg_mesh_config_client_appkey_list_evt_t	sl_btmesh_evt_config_client_appkey_list_id sl_btmesh_evt_config_client_appkey_list_t	
gecko_evt_mesh_config_client_appkey_list_end_id gecko_msg_mesh_config_client_appkey_list_end_evt_t	sl_btmesh_evt_config_client_appkey_list_end_id sl_btmesh_evt_config_client_appkey_list_end_t	
gecko_evt_mesh_config_client_binding_status_id gecko_msg_mesh_config_client_binding_status_evt_t	sl_btmesh_evt_config_client_binding_status_id sl_btmesh_evt_config_client_binding_status_t	
gecko_evt_mesh_config_client_bindings_list_id gecko_msg_mesh_config_client_bindings_list_evt_t	sl_btmesh_evt_config_client_bindings_list_id sl_btmesh_evt_config_client_bindings_list_t	
gecko_evt_mesh_config_client_bindings_list_end_id gecko_msg_mesh_config_client_bindings_list_end_evt_t	sl_btmesh_evt_config_client_bindings_list_end_id sl_btmesh_evt_config_client_bindings_list_end_t	
gecko_evt_mesh_config_client_model_pub_status_id gecko_msg_mesh_config_client_model_pub_status_evt_t	sl_btmesh_evt_config_client_model_pub_status_id sl_btmesh_evt_config_client_model_pub_status_t	
gecko_evt_mesh_config_client_model_sub_status_id gecko_msg_mesh_config_client_model_sub_status_evt_t	sl_btmesh_evt_config_client_model_sub_status_id sl_btmesh_evt_config_client_model_sub_status_t	
gecko_evt_mesh_config_client_subs_list_id gecko_msg_mesh_config_client_subs_list_evt_t	sl_btmesh_evt_config_client_subs_list_id sl_btmesh_evt_config_client_subs_list_t	

API 1.x	API 2.0	Notes
gecko_evt_mesh_config_client_subs_list_end_id gecko_msg_mesh_config_client_subs_list_end_evt_t	sl_btmesh_evt_config_client_subs_list_end_id sl_btmesh_evt_config_client_subs_list_end_t	
gecko_evt_mesh_config_client_heartbeat_pub_status_id gecko_msg_mesh_config_client_heartbeat_pub_status_evt_t	sl_btmesh_evt_config_client_heartbeat_pub_status_id sl_btmesh_evt_config_client_heartbeat_pub_status_t	
gecko_evt_mesh_config_client_heartbeat_sub_status_id gecko_msg_mesh_config_client_heartbeat_sub_status_evt_t	sl_btmesh_evt_config_client_heartbeat_sub_status_id sl_btmesh_evt_config_client_heartbeat_sub_status_t	
gecko_evt_mesh_config_client_beacon_status_id gecko_msg_mesh_config_client_beacon_status_evt_t	sl_btmesh_evt_config_client_beacon_status_id sl_btmesh_evt_config_client_beacon_status_t	
gecko_evt_mesh_config_client_default_ttl_status_id gecko_msg_mesh_config_client_default_ttl_status_evt_t	sl_btmesh_evt_config_client_default_ttl_status_id sl_btmesh_evt_config_client_default_ttl_status_t	
gecko_evt_mesh_config_client_gatt_proxy_status_id gecko_msg_mesh_config_client_gatt_proxy_status_evt_t	sl_btmesh_evt_config_client_gatt_proxy_status_id sl_btmesh_evt_config_client_gatt_proxy_status_t	
gecko_evt_mesh_config_client_relay_status_id gecko_msg_mesh_config_client_relay_status_evt_t	sl_btmesh_evt_config_client_relay_status_id sl_btmesh_evt_config_client_relay_status_t	
gecko_evt_mesh_config_client_network_transmit_status_id gecko_msg_mesh_config_client_network_transmit_status_evt_t	sl_btmesh_evt_config_client_network_transmit_status_id sl_btmesh_evt_config_client_network_transmit_status_t	
gecko_evt_mesh_config_client_identity_status_id gecko_msg_mesh_config_client_identity_status_evt_t	sl_btmesh_evt_config_client_identity_status_id sl_btmesh_evt_config_client_identity_status_t	
gecko_evt_mesh_config_client_friend_status_id gecko_msg_mesh_config_client_friend_status_evt_t	sl_btmesh_evt_config_client_friend_status_id sl_btmesh_evt_config_client_friend_status_t	
gecko_evt_mesh_config_client_lpn_polltimeout_status_id gecko_msg_mesh_config_client_lpn_polltimeout_status_evt_t	sl_btmesh_evt_config_client_lpn_polltimeout_status_id sl_btmesh_evt_config_client_lpn_polltimeout_status_t	
gecko_evt_mesh_config_client_dcd_data_id gecko_msg_mesh_config_client_dcd_data_evt_t	sl_btmesh_evt_config_client_dcd_data_id sl_btmesh_evt_config_client_dcd_data_t	
gecko_evt_mesh_config_client_dcd_data_end_id gecko_msg_mesh_config_client_dcd_data_end_evt_t	sl_btmesh_evt_config_client_dcd_data_end_id sl_btmesh_evt_config_client_dcd_data_end_t	
gecko_evt_mesh_config_client_reset_status_id gecko_msg_mesh_config_client_reset_status_evt_t	sl_btmesh_evt_config_client_reset_status_id sl_btmesh_evt_config_client_reset_status_t	
CLASS: mesh_friend	CLASS: mesh_ friend	There is no change to this class except the prefix change.
gecko_evt_mesh_friend_friendship_established_id gecko_msg_mesh_friend_friendship_established_evt_t	sl_btmesh_evt_friend_friendship_established_id sl_btmesh_evt_friend_friendship_established_t	
gecko_evt_mesh_friend_friendship_terminated_id gecko_msg_mesh_friend_friendship_terminated_evt_t	sl_btmesh_evt_friend_friendship_terminated_id sl_btmesh_evt_friend_friendship_terminated_t	

API 1.x	API 2.0	Notes
CLASS: mesh_generic_client	CLASS: mesh_generic_client	There is no change to this class except the prefix change.
gecko_evt_mesh_generic_client_server_status_id gecko_msg_mesh_generic_client_server_status_evt_t	sl_btmesh_evt_generic_client_server_status_id sl_btmesh_evt_generic_client_server_status_t	
CLASS: mesh_generic_server	CLASS: mesh_generic_server	There is no change to this class except the prefix change.
gecko_evt_mesh_generic_server_client_request_id gecko_msg_mesh_generic_server_client_request_evt_t	sl_btmesh_evt_generic_server_client_request_id sl_btmesh_evt_generic_server_client_request_t	
gecko_evt_mesh_generic_server_state_changed_id gecko_msg_mesh_generic_server_state_changed_evt_t	sl_btmesh_evt_generic_server_state_changed_id sl_btmesh_evt_generic_server_state_changed_t	
gecko_evt_mesh_generic_server_state_recall_id gecko_msg_mesh_generic_server_state_recall_evt_t	sl_btmesh_evt_generic_server_state_recall_id sl_btmesh_evt_generic_server_state_recall_t	
CLASS: mesh_health_client	CLASS: mesh_health_client	There is no change to this class except the prefix change.
gecko_evt_mesh_health_client_server_status_id gecko_msg_mesh_health_client_server_status_evt_t	sl_btmesh_evt_health_client_server_status_id sl_btmesh_evt_health_client_server_status_t	
gecko_evt_mesh_health_client_server_status_period_id gecko_msg_mesh_health_client_server_status_period_evt_t	sl_btmesh_evt_health_client_server_status_period_id sl_btmesh_evt_health_client_server_status_period_t	
gecko_evt_mesh_health_client_server_status_attention_id gecko_msg_mesh_health_client_server_status_attention_evt_t	sl_btmesh_evt_health_client_server_status_attention_id sl_btmesh_evt_health_client_server_status_attention_t	
CLASS: mesh_health_server	CLASS: mesh_health_server	There is no change to this class except the prefix change.
gecko_evt_mesh_health_server_attention_id gecko_msg_mesh_health_server_attention_evt_t	sl_btmesh_evt_health_server_attention_id sl_btmesh_evt_health_server_attention_t	
gecko_evt_mesh_health_server_test_request_id gecko_msg_mesh_health_server_test_request_evt_t	sl_btmesh_evt_health_server_test_request_id sl_btmesh_evt_health_server_test_request_t	
CLASS: mesh_lc_client	CLASS: mesh_lc_client	There is no change to this class except the prefix change.
gecko_evt_mesh_lc_client_mode_status_id gecko_msg_mesh_lc_client_mode_status_evt_t	sl_btmesh_evt_lc_client_mode_status_id sl_btmesh_evt_lc_client_mode_status_t	
gecko_evt_mesh_lc_client_om_status_id gecko_msg_mesh_lc_client_om_status_evt_t	sl_btmesh_evt_lc_client_om_status_id sl_btmesh_evt_lc_client_om_status_t	
gecko_evt_mesh_lc_client_light_onoff_status_id gecko_msg_mesh_lc_client_light_onoff_status_evt_t	sl_btmesh_evt_lc_client_light_onoff_status_id sl_btmesh_evt_lc_client_light_onoff_status_t	

API 1.x	API 2.0	Notes
gecko_evt_mesh_lc_client_property_status_id gecko_msg_mesh_lc_client_property_status_evt_t	sl_btmesh_evt_lc_client_property_status_id sl_btmesh_evt_lc_client_property_status_t	
CLASS: mesh_lc_server	CLASS: mesh_lc_server	There is no change to this class except the prefix change.
gecko_evt_mesh_lc_server_mode_updated_id gecko_msg_mesh_lc_server_mode_updated_evt_t	sl_btmesh_evt_lc_server_mode_updated_id sl_btmesh_evt_lc_server_mode_updated_t	
gecko_evt_mesh_lc_server_om_updated_id gecko_msg_mesh_lc_server_om_updated_evt_t	sl_btmesh_evt_lc_server_om_updated_id sl_btmesh_evt_lc_server_om_updated_t	
gecko_evt_mesh_lc_server_light_onoff_updated_id gecko_msg_mesh_lc_server_light_onoff_updated_evt_t	sl_btmesh_evt_lc_server_light_onoff_updated_id sl_btmesh_evt_lc_server_light_onoff_updated_t	
gecko_evt_mesh_lc_server_occupancy_updated_id gecko_msg_mesh_lc_server_occupancy_updated_evt_t	sl_btmesh_evt_lc_server_occupancy_updated_id sl_btmesh_evt_lc_server_occupancy_updated_t	
gecko_evt_mesh_lc_server_ambient_lux_level_updated_id gecko_msg_mesh_lc_server_ambient_lux_level_updated_evt_t	sl_btmesh_evt_lc_server_ambient_lux_level_updated_id sl_btmesh_evt_lc_server_ambient_lux_level_updated_t	
gecko_evt_mesh_lc_server_linear_output_updated_id gecko_msg_mesh_lc_server_linear_output_updated_evt_t	sl_btmesh_evt_lc_server_linear_output_updated_id sl_btmesh_evt_lc_server_linear_output_updated_t	
gecko_evt_mesh_lc_server_state_updated_id gecko_msg_mesh_lc_server_state_updated_evt_t	sl_btmesh_evt_lc_server_state_updated_id sl_btmesh_evt_lc_server_state_updated_t	
gecko_evt_mesh_lc_server_regulator_debug_info_id gecko_msg_mesh_lc_server_regulator_debug_info_evt_t	sl_btmesh_evt_lc_server_regulator_debug_info_id sl_btmesh_evt_lc_server_regulator_debug_info_t	
CLASS: mesh_lc_setup_server	CLASS: mesh_lc_setup_server	There is no change to this class except the prefix change.
gecko_evt_mesh_lc_setup_server_set_property_id gecko_msg_mesh_lc_setup_server_set_property_evt_t	sl_btmesh_evt_lc_setup_server_set_property_id sl_btmesh_evt_lc_setup_server_set_property_t	
CLASS: mesh_lpn	CLASS: mesh_lpn	There is no change to this class except the prefix change.
gecko_evt_mesh_lpn_friendship_established_id gecko_msg_mesh_lpn_friendship_established_evt_t	sl_btmesh_evt_lpn_friendship_established_id sl_btmesh_evt_lpn_friendship_established_t	
gecko_evt_mesh_lpn_friendship_failed_id gecko_msg_mesh_lpn_friendship_failed_evt_t	sl_btmesh_evt_lpn_friendship_failed_id sl_btmesh_evt_lpn_friendship_failed_t	
gecko_evt_mesh_lpn_friendship_terminated_id gecko_msg_mesh_lpn_friendship_terminated_evt_t	sl_btmesh_evt_lpn_friendship_terminated_id sl_btmesh_evt_lpn_friendship_terminated_t	
CLASS: mesh_node	CLASS: mesh_node	

API 1.x	API 2.0	Notes
gecko_evt_mesh_node_initialized_id gecko_msg_mesh_node_initialized_evt_t	sl_btmesh_evt_node_initialized_id sl_btmesh_evt_node_initialized_t	
gecko_evt_mesh_node_provisioned_id gecko_msg_mesh_node_provisioned_evt_t	sl_btmesh_evt_node_provisioned_id sl_btmesh_evt_node_provisioned_t	
gecko_evt_mesh_node_config_get_id gecko_msg_mesh_node_config_get_evt_t	sl_btmesh_evt_node_config_get_id sl_btmesh_evt_node_config_get_t	
gecko_evt_mesh_node_config_set_id gecko_msg_mesh_node_config_set_evt_t	sl_btmesh_evt_node_config_set_id sl_btmesh_evt_node_config_set_t	
gecko_evt_mesh_node_display_output_oob_id gecko_msg_mesh_node_display_output_oob_evt_t	sl_btmesh_evt_node_display_output_oob_id sl_btmesh_evt_node_display_output_oob_t	
gecko_evt_mesh_node_input_oob_request_id gecko_msg_mesh_node_input_oob_request_evt_t	sl_btmesh_evt_node_input_oob_request_id sl_btmesh_evt_node_input_oob_request_t	
gecko_evt_mesh_node_provisioning_started_id gecko_msg_mesh_node_provisioning_started_evt_t	sl_btmesh_evt_node_provisioning_started_id sl_btmesh_evt_node_provisioning_started_t	
gecko_evt_mesh_node_provisioning_failed_id gecko_msg_mesh_node_provisioning_failed_evt_t	sl_btmesh_evt_node_provisioning_failed_id sl_btmesh_evt_node_provisioning_failed_t	
gecko_evt_mesh_node_key_added_id gecko_msg_mesh_node_key_added_evt_t	sl_btmesh_evt_node_key_added_id sl_btmesh_evt_node_key_added_t	
gecko_evt_mesh_node_model_config_changed_id gecko_msg_mesh_node_model_config_changed_evt_t	sl_btmesh_evt_node_model_config_changed_id sl_btmesh_evt_node_model_config_changed_t	
gecko_evt_mesh_node_reset_id gecko_msg_mesh_node_reset_evt_t	sl_btmesh_evt_node_reset_id	No event anymore
gecko_evt_mesh_node_ivrecovery_needed_id gecko_msg_mesh_node_ivrecovery_needed_evt_t	sl_btmesh_evt_node_ivrecovery_needed_id sl_btmesh_evt_node_ivrecovery_needed_t	
gecko_evt_mesh_node_changed_ivupdate_state_id gecko_msg_mesh_node_changed_ivupdate_state_evt_t	sl_btmesh_evt_node_changed_ivupdate_state_id sl_btmesh_evt_node_changed_ivupdate_state_t	
gecko_evt_mesh_node_static_oob_request_id gecko_msg_mesh_node_static_oob_request_evt_t	sl_btmesh_evt_node_static_oob_request_id	No event anymore
gecko_evt_mesh_node_key_removed_id gecko_msg_mesh_node_key_removed_evt_t	sl_btmesh_evt_node_key_removed_id sl_btmesh_evt_node_key_removed_t	
gecko_evt_mesh_node_key_updated_id gecko_msg_mesh_node_key_updated_evt_t	sl_btmesh_evt_node_key_updated_id sl_btmesh_evt_node_key_updated_t	
gecko_evt_mesh_node_heartbeat_id gecko_msg_mesh_node_heartbeat_evt_t	sl_btmesh_evt_node_heartbeat_id sl_btmesh_evt_node_heartbeat_t	
gecko_evt_mesh_node_heartbeat_start_id gecko_msg_mesh_node_heartbeat_start_evt_t	sl_btmesh_evt_node_heartbeat_start_id sl_btmesh_evt_node_heartbeat_start_t	
gecko_evt_mesh_node_heartbeat_stop_id gecko_msg_mesh_node_heartbeat_stop_evt_t	sl_btmesh_evt_node_heartbeat_stop_id sl_btmesh_evt_node_heartbeat_stop_t	
gecko_evt_mesh_node_beacon_received_id gecko_msg_mesh_node_beacon_received_evt_t	sl_btmesh_evt_node_beacon_received_id sl_btmesh_evt_node_beacon_received_t	
	sl_btmesh_evt_node_local_dcd_data_id sl_btmesh_evt_node_local_dcd_data_t	sl_btmesh_node_get_local_dcd will trigger this event, and it carries the DCD data

API 1.x	API 2.0	Notes
	sl_btmesh_evt_node_local_dcd_data_end_id sl_btmesh_evt_node_local_dcd_data_end_t	sl_btmesh_node_get_local_dcd will trigger this event, and it indicates all DCD data is reported
CLASS: mesh_prov	CLASS: mesh_prov	
gecko_evt_mesh_prov_initialized_id gecko_msg_mesh_prov_initialized_evt_t	sl_btmesh_evt_prov_initialized_id sl_btmesh_evt_prov_initialized_t	
gecko_evt_mesh_prov_provisioning_failed_id gecko_msg_mesh_prov_provisioning_failed_evt_t	sl_btmesh_evt_prov_provisioning_failed_id sl_btmesh_evt_prov_provisioning_failed_t	
gecko_evt_mesh_prov_device_provisioned_id gecko_msg_mesh_prov_device_provisioned_evt_t	sl_btmesh_evt_prov_device_provisioned_id sl_btmesh_evt_prov_device_provisioned_t	
gecko_evt_mesh_prov_unprov_beacon_id gecko_msg_mesh_prov_unprov_beacon_evt_t	sl_btmesh_evt_prov_unprov_beacon_id sl_btmesh_evt_prov_unprov_beacon_t	
gecko_evt_mesh_prov_oob_pkey_request_id gecko_msg_mesh_prov_oob_pkey_request_evt_t	sl_btmesh_evt_prov_oob_pkey_request_id sl_btmesh_evt_prov_oob_pkey_request_t	
gecko_evt_mesh_prov_oob_auth_request_id gecko_msg_mesh_prov_oob_auth_request_evt_t	sl_btmesh_evt_prov_oob_auth_request_id sl_btmesh_evt_prov_oob_auth_request_t	
gecko_evt_mesh_prov_oob_display_input_id gecko_msg_mesh_prov_oob_display_input_evt_t	sl_btmesh_evt_prov_oob_display_input_id sl_btmesh_evt_prov_oob_display_input_t	
gecko_evt_mesh_prov_ddb_list_id gecko_msg_mesh_prov_ddb_list_evt_t	sl_btmesh_evt_prov_ddb_list_id sl_btmesh_evt_prov_ddb_list_t	
gecko_evt_mesh_prov_uri_id gecko_msg_mesh_prov_uri_evt_t	sl_btmesh_evt_prov_uri_id sl_btmesh_evt_prov_uri_t	
gecko_evt_mesh_prov_key_refresh_phase_update_id gecko_msg_mesh_prov_key_refresh_phase_update_evt_t	sl_btmesh_evt_prov_key_refresh_phase_update_id sl_btmesh_evt_prov_key_refresh_phase_update_t	
gecko_evt_mesh_prov_key_refresh_node_update_id gecko_msg_mesh_prov_key_refresh_node_update_evt_t	sl_btmesh_evt_prov_key_refresh_node_update_id sl_btmesh_evt_prov_key_refresh_node_update_t	
gecko_evt_mesh_prov_key_refresh_complete_id gecko_msg_mesh_prov_key_refresh_complete_evt_t	sl_btmesh_evt_prov_key_refresh_complete_id sl_btmesh_evt_prov_key_refresh_complete_t	
	sl_btmesh_evt_prov_provisioning_suspended_id sl_btmesh_evt_prov_provisioning_suspended_t	Provisioning suspended, needs user input the configuration data to continue
	sl_btmesh_evt_prov_capabilities_id sl_btmesh_evt_prov_capabilities_t	
CLASS: mesh_proxy	CLASS: mesh_proxy	
gecko_evt_mesh_proxy_connected_id gecko_msg_mesh_proxy_connected_evt_t	sl_btmesh_evt_proxy_connected_id sl_btmesh_evt_proxy_connected_t	
gecko_evt_mesh_proxy_disconnected_id gecko_msg_mesh_proxy_disconnected_evt_t	sl_btmesh_evt_proxy_disconnected_id sl_btmesh_evt_proxy_disconnected_t	
gecko_evt_mesh_proxy_filter_status_id gecko_msg_mesh_proxy_filter_status_evt_t	sl_btmesh_evt_proxy_filter_status_id sl_btmesh_evt_proxy_filter_status_t	

API 1.x	API 2.0	Notes
CLASS: mesh_scene_client	CLASS: mesh_scene_client	
gecko_evt_mesh_scene_client_status_id gecko_msg_mesh_scene_client_status_evt_t	sl_btmesh_evt_scene_client_status_id sl_btmesh_evt_scene_client_status_t	
gecko_evt_mesh_scene_client_register_status_id gecko_msg_mesh_scene_client_register_status_evt_t	sl_btmesh_evt_scene_client_register_status_id sl_btmesh_evt_scene_client_register_status_t	
CLASS: mesh_scene_server	CLASS: mesh_scene_server	
gecko_evt_mesh_scene_server_get_id gecko_msg_mesh_scene_server_get_evt_t	sl_btmesh_evt_scene_server_get_id sl_btmesh_evt_scene_server_get_t	
gecko_evt_mesh_scene_server_register_get_id gecko_msg_mesh_scene_server_register_get_evt_t	sl_btmesh_evt_scene_server_register_get_id sl_btmesh_evt_scene_server_register_get_t	
gecko_evt_mesh_scene_server_recall_id gecko_msg_mesh_scene_server_recall_evt_t	sl_btmesh_evt_scene_server_recall_id sl_btmesh_evt_scene_server_recall_t	
gecko_evt_mesh_scene_server_publish_id gecko_msg_mesh_scene_server_publish_evt_t	sl_btmesh_evt_scene_server_publish_id sl_btmesh_evt_scene_server_publish_t	
CLASS: mesh_scene_setup_server	CLASS: mesh_scene_setup_server	
gecko_evt_mesh_scene_setup_server_store_id gecko_msg_mesh_scene_setup_server_store_evt_t	sl_btmesh_evt_scene_setup_server_store_id sl_btmesh_evt_scene_setup_server_store_t	
gecko_evt_mesh_scene_setup_server_delete_id gecko_msg_mesh_scene_setup_server_delete_evt_t	sl_btmesh_evt_scene_setup_server_delete_id sl_btmesh_evt_scene_setup_server_delete_t	
gecko_evt_mesh_scene_setup_server_publish_id gecko_msg_mesh_scene_setup_server_publish_evt_t	sl_btmesh_evt_scene_setup_server_publish_id sl_btmesh_evt_scene_setup_server_publish_t	
CLASS: mesh_scheduler_client	CLASS: mesh_scheduler_client	
gecko_evt_mesh_scheduler_client_status_id gecko_msg_mesh_scheduler_client_status_evt_t	sl_btmesh_evt_scheduler_client_status_id sl_btmesh_evt_scheduler_client_status_t	
gecko_evt_mesh_scheduler_client_action_status_id gecko_msg_mesh_scheduler_client_action_status_evt_t	sl_btmesh_evt_scheduler_client_action_status_id sl_btmesh_evt_scheduler_client_action_status_t	
CLASS: mesh_scheduler_server	CLASS: mesh_scheduler_server	
gecko_evt_mesh_scheduler_server_action_changed_id gecko_msg_mesh_scheduler_server_action_changed_evt_t	sl_btmesh_evt_scheduler_server_action_changed_id sl_btmesh_evt_scheduler_server_action_changed_t	
CLASS: mesh_sensor_client	CLASS: mesh_sensor_client	
gecko_evt_mesh_sensor_client_descriptor_status_id gecko_msg_mesh_sensor_client_descriptor_status_evt_t	sl_btmesh_evt_sensor_client_descriptor_status_id sl_btmesh_evt_sensor_client_descriptor_status_t	
gecko_evt_mesh_sensor_client_cadence_status_id gecko_msg_mesh_sensor_client_cadence_status_evt_t	sl_btmesh_evt_sensor_client_cadence_status_id sl_btmesh_evt_sensor_client_cadence_status_t	

API 1.x	API 2.0	Notes
gecko_evt_mesh_sensor_client_settings_status_id gecko_msg_mesh_sensor_client_settings_status_evt_t	sl_btmesh_evt_sensor_client_settings_status_id sl_btmesh_evt_sensor_client_settings_status_t	
gecko_evt_mesh_sensor_client_setting_status_id gecko_msg_mesh_sensor_client_setting_status_evt_t	sl_btmesh_evt_sensor_client_setting_status_id sl_btmesh_evt_sensor_client_setting_status_t	
gecko_evt_mesh_sensor_client_status_id gecko_msg_mesh_sensor_client_status_evt_t	sl_btmesh_evt_sensor_client_status_id sl_btmesh_evt_sensor_client_status_t	
gecko_evt_mesh_sensor_client_column_status_id gecko_msg_mesh_sensor_client_column_status_evt_t	sl_btmesh_evt_sensor_client_column_status_id sl_btmesh_evt_sensor_client_column_status_t	
gecko_evt_mesh_sensor_client_series_status_id gecko_msg_mesh_sensor_client_series_status_evt_t	sl_btmesh_evt_sensor_client_series_status_id sl_btmesh_evt_sensor_client_series_status_t	
gecko_evt_mesh_sensor_client_publish_id gecko_msg_mesh_sensor_client_publish_evt_t	sl_btmesh_evt_sensor_client_publish_id sl_btmesh_evt_sensor_client_publish_t	
CLASS: mesh_sensor_server	CLASS: mesh_sensor_server	
gecko_evt_mesh_sensor_server_get_request_id gecko_msg_mesh_sensor_server_get_request_evt_t	sl_btmesh_evt_sensor_server_get_request_id sl_btmesh_evt_sensor_server_get_request_t	
gecko_evt_mesh_sensor_server_get_column_request_id gecko_msg_mesh_sensor_server_get_column_request_evt_t	sl_btmesh_evt_sensor_server_get_column_request_id sl_btmesh_evt_sensor_server_get_column_request_t	
gecko_evt_mesh_sensor_server_get_series_request_id gecko_msg_mesh_sensor_server_get_series_request_evt_t	sl_btmesh_evt_sensor_server_get_series_request_id sl_btmesh_evt_sensor_server_get_series_request_t	
gecko_evt_mesh_sensor_server_publish_id gecko_msg_mesh_sensor_server_publish_evt_t	sl_btmesh_evt_sensor_server_publish_id sl_btmesh_evt_sensor_server_publish_t	
CLASS: mesh_sensor_setup_server	CLASS: mesh_sensor_setup_server	
gecko_evt_mesh_sensor_setup_server_get_cadence_request_id gecko_msg_mesh_sensor_setup_server_get_cadence_request_evt_t	sl_btmesh_evt_sensor_setup_server_get_cadence_request_id sl_btmesh_evt_sensor_setup_server_get_cadence_request_t	
gecko_evt_mesh_sensor_setup_server_set_cadence_request_id gecko_msg_mesh_sensor_setup_server_set_cadence_request_evt_t	sl_btmesh_evt_sensor_setup_server_set_cadence_request_id sl_btmesh_evt_sensor_setup_server_set_cadence_request_t	
gecko_evt_mesh_sensor_setup_server_get_settings_request_id gecko_msg_mesh_sensor_setup_server_get_settings_request_evt_t	sl_btmesh_evt_sensor_setup_server_get_settings_request_id sl_btmesh_evt_sensor_setup_server_get_settings_request_t	
gecko_evt_mesh_sensor_setup_server_get_setting_request_id gecko_msg_mesh_sensor_setup_server_get_setting_request_evt_t	sl_btmesh_evt_sensor_setup_server_get_setting_request_id sl_btmesh_evt_sensor_setup_server_get_setting_request_t	
gecko_evt_mesh_sensor_setup_server_set_setting_request_id gecko_msg_mesh_sensor_setup_server_set_setting_request_evt_t	sl_btmesh_evt_sensor_setup_server_set_setting_request_id sl_btmesh_evt_sensor_setup_server_set_setting_request_t	

API 1.x	API 2.0	Notes
gecko_evt_mesh_sensor_setup_server_publish_id gecko_msg_mesh_sensor_setup_server_publish_evt_t	sl_btmesh_evt_sensor_setup_server_publish_id sl_btmesh_evt_sensor_setup_server_publish_t	
CLASS: mesh_test	CLASS: mesh_test	
gecko_evt_mesh_test_local_heartbeat_subscription_complete_id gecko_msg_mesh_test_local_heartbeat_subscription_complete_evt_t	sl_btmesh_evt_test_local_heartbeat_subscription_complete_id sl_btmesh_evt_test_local_heartbeat_subscription_complete_t	
CLASS: mesh_time_client	CLASS: mesh_time_client	
gecko_evt_mesh_time_client_time_status_id gecko_msg_mesh_time_client_time_status_evt_t	sl_btmesh_evt_time_client_time_status_id sl_btmesh_evt_time_client_time_status_t	
gecko_evt_mesh_time_client_time_zone_status_id gecko_msg_mesh_time_client_time_zone_status_evt_t	sl_btmesh_evt_time_client_time_zone_status_id sl_btmesh_evt_time_client_time_zone_status_t	
gecko_evt_mesh_time_client_tai_utc_delta_status_id gecko_msg_mesh_time_client_tai_utc_delta_status_evt_t	sl_btmesh_evt_time_client_tai_utc_delta_status_id sl_btmesh_evt_time_client_tai_utc_delta_status_t	
gecko_evt_mesh_time_client_time_role_status_id gecko_msg_mesh_time_client_time_role_status_evt_t	sl_btmesh_evt_time_client_time_role_status_id sl_btmesh_evt_time_client_time_role_status_t	
CLASS: mesh_time_server	CLASS: mesh_time_server	
gecko_evt_mesh_time_server_time_updated_id gecko_msg_mesh_time_server_time_updated_evt_t	sl_btmesh_evt_time_server_time_updated_id sl_btmesh_evt_time_server_time_updated_t	
gecko_evt_mesh_time_server_time_zone_offset_updated_id gecko_msg_mesh_time_server_time_zone_offset_updated_evt_t	sl_btmesh_evt_time_server_time_zone_offset_updated_id sl_btmesh_evt_time_server_time_zone_offset_updated_t	
gecko_evt_mesh_time_server_tai_utc_delta_updated_id gecko_msg_mesh_time_server_tai_utc_delta_updated_evt_t	sl_btmesh_evt_time_server_tai_utc_delta_updated_id sl_btmesh_evt_time_server_tai_utc_delta_updated_t	
gecko_evt_mesh_time_server_time_role_updated_id gecko_msg_mesh_time_server_time_role_updated_evt_t	sl_btmesh_evt_time_server_time_role_updated_id sl_btmesh_evt_time_server_time_role_updated_t	
CLASS: mesh_vendor_model	CLASS: mesh_vendor_model	
gecko_evt_mesh_vendor_model_receive_id gecko_msg_mesh_vendor_model_receive_evt_t	sl_btmesh_evt_vendor_model_receive_id sl_btmesh_evt_vendor_model_receive_t	

4.4 Migration Example

The following code snippets show an example how a Bluetooth Mesh v1.x application is to be updated to work in the Bluetooth Mesh v2.x environment, considering all the changes mentioned above. The sample code simply starts unprovisioned device beacon.

```
#include "native_gecko.h"

void appMain(gecko_configuration_t* pconfig)
{
 initLog();
 gecko_stack_init(&config);
 gecko_bgapi_classes_init();

 while (1)
 {
  struct gecko_cmd_packet* evt;
  evt = gecko_peek_event();

  // Handle stack events
  switch (BGLIB_MSG_ID(evt->header)) {
    case gecko_evt_system_boot_id:
      // Initialize Mesh stack in Node operation
mode, wait for initialized event
      result = gecko_cmd_mesh_node_init()->result;
      if (result != bg_err_success) {
        /* handle errors here */
      }
      break;

    case gecko_evt_mesh_node_initialized_id:
            if (!evt-
>data.evt_mesh_node_initialized.provisioned) {
  gecko_cmd_mesh_node_start_unprov_beaconing(0x3);
}
      break;

    default:
      break;
  }
 }
}
```

```
#include "sl_btmesh_api.h"


void sl_btmesh_on_event(sl_bt_msg_t* evt)
{
  sl_status_t sc;
  // Handle stack events
  switch (SL_BT_MSG_ID(evt->header)) {
    case sl_bt_evt_system_boot_id:
      sl_app_log("Node init\r\n");
      sc = sl_btmesh_node_init();
      sl_app_assert(sc == SL_STATUS_OK,
        "[E: 0x%04x] Failed to init node\n",
        (int)sc);
      break;

    case sl_btmesh_evt_node_initialized_id:
      sl_app_log("Initialized\r\n");
      sc =
sl_btmesh_node_start_unprov_beaconing(0x3);
      sl_app_assert(sc == SL_STATUS_OK,
       "[E: 0x%04x] Failed to start unprovi-
sioned beaconing\n",
       (int)sc);
      break;

    default:
      break;
  }
}
```

5 Migrating NCP projects

Migrating an NCP application is usually easy, since the stack and the application are well-separated. While the stack is running on the NCP target, the application is running on the NCP host. Therefore, a stack update usually does not affect the application except that the API changes must be respected.

An SDK update in the NCP use case means that:

1. The NCP target device must be programmed with the **Bluetooth Mesh - NCP Empty** sample app of the new SDK.
2. UART pins must be configured in the sample app.
3. The GATT database must be imported in the sample app.

Furthermore,

4. The NCP host device must include the new BGAPI header files, so that it can communicate with the target.
5. Deprecated API calls must be updated, if there are any.

Upgrading the NCP target code from Bluetooth Mesh SDK v1.x to v2.x is easy. A new **Bluetooth Mesh - NCP Empty** project must be generated with Bluetooth Mesh SDK v2.x. The UART pins can be easily configured with the Pin Tool, and the GATT database can be easily imported with the GATT Configurator. Should you use deep sleep mode in the NCP target, you must install the Wake Lock component and configure it. For more information about the NCP mode, see *AN1259: Using the Silicon Labs v3.x Bluetooth® Stack in Network Co-Processor Mode.*

The NCP host update involves more changes. After updating the header files, not only the full Bluetooth Mesh API has to be updated but also some BGLIB commands and macros.

An NCP host code using Bluetooth Mesh SDK v1.x must contain the following header files:

- **bg_errorcodes.h**
- **bg_types.h**
- **host_gecko.h**
- **gecko_bglib.h**

and the following source file:

- **gecko_bglib.c**

An NCP host code using Bluetooth Mesh SDK v2.x must contain the following header files:

- **sl_status.h** (in *SDK_DIR/platform/common/inc*)
- **sl_bt_types.h** (in *SDK_DIR/protocol/bluetooth/inc*)
- **sl_bt_api.h** (in *SDK_DIR/protocol/bluetooth/inc*)
- **sl_btmesh_api.h** (in *SDK_DIR/protocol/bluetooth/inc*)
- **sl_bt_ncp_host.h** (in *SDK_DIR/protocol/bluetooth/inc*)

and the following source files:

- **sl_bt_ncp_host_api.c** (in *SDK_DIR/protocol/bluetooth/src*)
- **sl_bt_ncp_host.c** (in *SDK_DIR/protocol/bluetooth/src*)

The new header files use the new nomenclature (commands/events starting with sl_bt_...) even if the underlying BGAPI packet content, which is sent to the target device via UART, may be unchanged in some cases. Therefore NCP host code must be completely updated according the description in section 4 Bluetooth Mesh API, using the new BGAPI.

Beside the changes in BGAPI (Bluetooth commands and events), the host API is also changed similarly to the changes in C API. The following table summarizes the changes in the host API:

Table 5-1. Changes in the Host API

API 1.x	API 2.0	Notes
BGLIB_DEFINE	SL_BT_API_DEFINE	
BGLIB_INITIALIZE	SL_BT_API_INITIALIZE	
BGLIB_INITIALIZE_NONBLOCK	SL_BT_API_INITIALIZE_NONBLOCK	

API 1.x	API 2.0	Notes
struct gecko_cmd_packet	sl_bt_msg_t	
BGLIB_MSG_ID	SL_BT_MSG_ID	
struct gecko_cmd_packet* **gecko_wait_event**()	sl_status_t **sl_bt_wait_event**(sl_bt_msg_t* evt)	In API 2.0, an event object is copied into the memory provided by application.
struct gecko_cmd_packet* **gecko_peek_event**()	sl_status_t **sl_bt_pop_event**(sl_bt_msg_t* evt)	
int **gecko_event_pending**()	bool **sl_bt_event_pending**()	

The NCP host code must be updated according to these changes. For example fetching an event changes from:

```
struct gecko_cmd_packet *p;
p = gecko_wait_event();
switch (BGLIB_MSG_ID(p->header)) {…}
```

to:

```
sl_bt_msg_t evt;
sl_bt_msg_t *p = &evt;
sl_bt_wait_event(&evt);
switch (SL_BT_MSG_ID(p->header)) {…}
```

Regarding the software architecture, the **empty host example** created for PCs (in <SDK_DIR>/app/bluetooth/example_host/empty) is updated to align with the new SoC software architecture. While it is not necessary to update the architecture on the NCP host, it is recommended to use this new architecture on newly created NCP host projects, so that it aligns with SoC code.

Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support & Community
www.silabs.com/community

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com