

How to use GNAUpdaterSDK Demo

Contents

How to use GNAUpdaterSDK Demo.....	1
Prerequisites	3
How to run an application.....	3
Versions	3
Device detection	4
Device information	4
Cease activity / Initialize	5
Install Drivers	5
Other functions	6
Test Server Connection	6
Subscribe events	6
Clear subscriptions	7
Offline mode	7
Send No-op.....	7
Cancel Async Call-back.....	7
Get Current Device Name	7
Get Current Device FW Version.....	7
Get Interval Current Device FW Versions	7
Get Current Device Last FW Version.....	7
Refresh Config Async	8
Get SDK Version Async.....	8
Device update	8
Offline mode.....	8

Prerequisites

In order to run the Demo application, the following components should be installed on target machine:

Microsoft .NET 4.5 Full Framework
Microsoft Visual C++ 2005 Redistributable Package (x86)
Microsoft Visual C++ 2012 Redistributable Package (x86)
Microsoft Visual C++ 2013 Redistributable Package (x86)
Microsoft Visual C++ 2015 Redistributable Package (x86)
Microsoft Visual C++ 2017 Redistributable Package (x86)

How to run an application

To run the application, navigate to the system folder with unpacked distributed package and run the "GNAUpdaterSDK_Demo.exe".

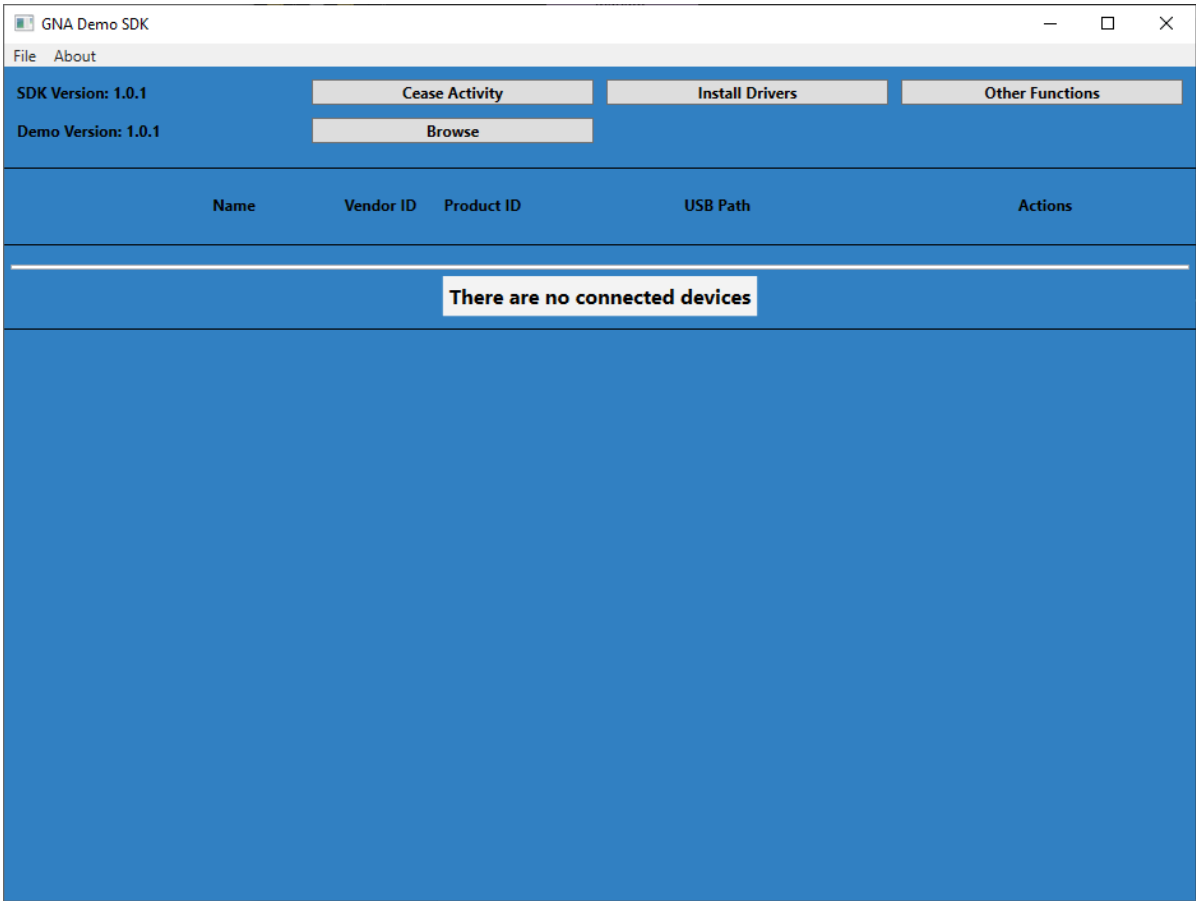


Figure 1. GNA Updater SDK Demo user interface (main window)

Versions

Versions of an Updater SDK being used and the Demo application itself can be found at the top left corner of the main form.

Device detection

When a device is attached to computer, it will appear in the connected devices list automatically:



Figure 2. Device detection

Device information

When device is attached, some of its properties like Name, Vendor ID, Product ID and USB Path are shown directly in the device list. Additional information can be viewed by clicking the "Info" button at the right edge of the device view. New window with device information will be shown in this case:

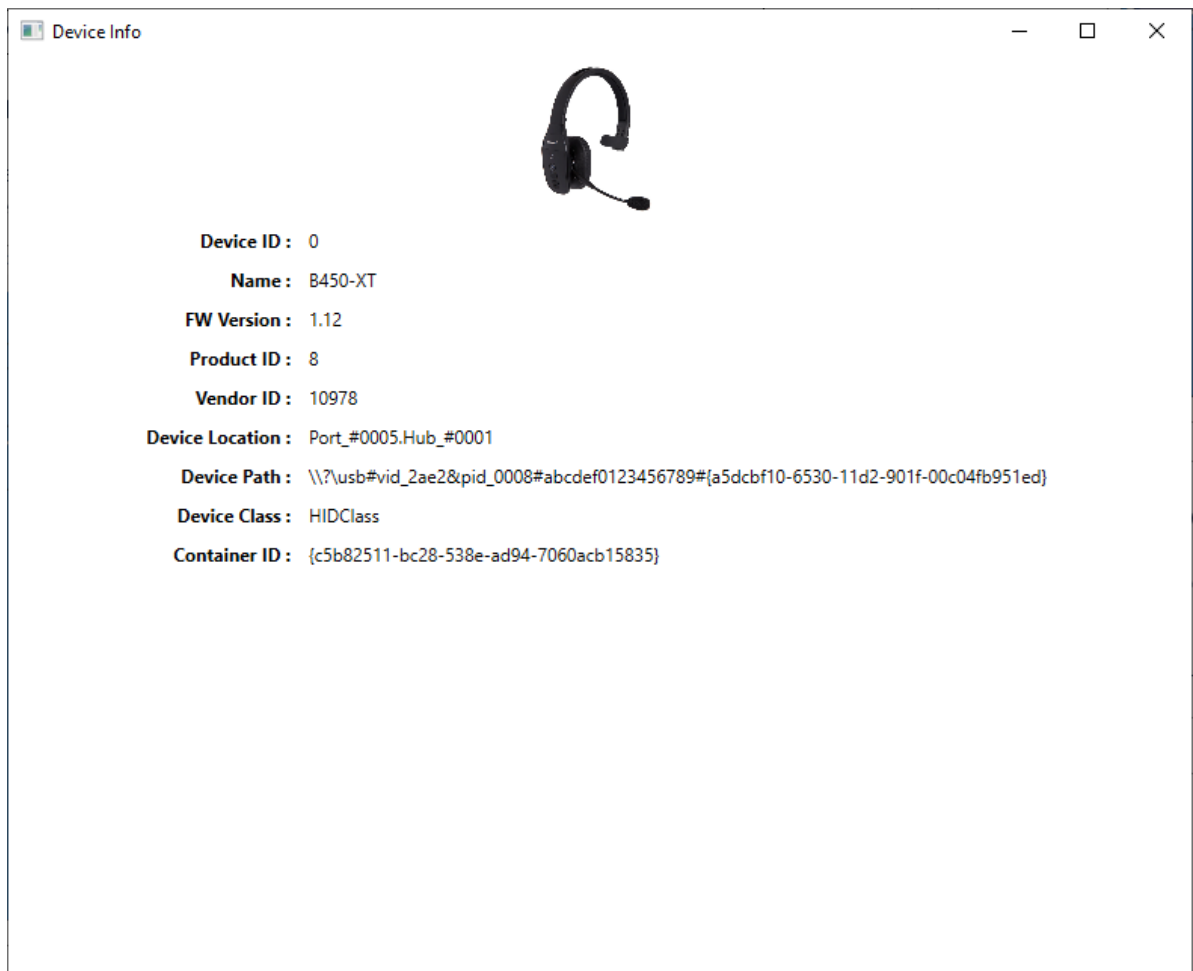


Figure 3. Device information window

Cease activity / Initialize

"Cease activity" / "Initialize" button allows user to control current state of the SDK. SDK is initialized upon Demo application start automatically and is in the "Initialized" state, if user is to perform a "Cease Activity" command, SDK will transfer into the "Idle" state. In "Idle" state, operations with devices are not available, no server connection is maintained nor attempted in this state. More information on SDK API states can be found in GNA Updater SDK API Documentation.

Install Drivers

Some legacy devices might require driver installation. Demo provides an ability to install necessary drivers for current platform via SDK API. This can be performed by "Install Drivers" button at the top of the main form. You might be prompted by Windows UAC to allow the drivers installation and after that, depending on whether installation succeeded, a result window will pop up showing either success message, or an error code. Error code descriptions can be found in GNA Updater SDK API Documentation.

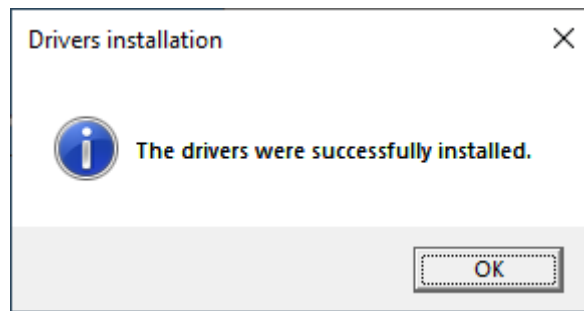


Figure 4. Drivers installation

Other functions

“Other functions” button allows to open a new window that allows to test different Updater SDK functionality:

Other Functions			
Test Server Connection		Get Current Device Name	
Result: OK		Name: B450-XT	
Subscribe Events		Get Current Device FW Version	
Result: OK		FW Version: 1.12	
Clear Subscriptions		Get Interval Current Device FW Versions	
Result: OK		FW Versions: 1.12	
Offline Mode		Get Current Device Last FW Version	
Result: Disabled		Last FWVersion: 1.12	
Send No-op		Refresh Config Async	
Result: Hello World!		Result: OK	
Cancel Async Call-back		Get SDK Version Async	
Result: OK		Result: 1.0.1	

Figure 5. Other functions window

Test Server Connection

This button invokes the Updater SDK API that checks the availability of the server and attempts to perform a connection to it. The result will show either “OK” (Success) status or “Error” (Failure) status.²

Subscribe events

This button subscribes different handlers in demo to the Updater SDK events, including the error events.

Note: There will be no error pop up messages unless "Subscribe events" is performed beforehand.

Clear subscriptions

This button unsubscribes any handlers in demo from the Updater SDK events, including the error events.

Note: There will be no error pop up messages unless "Subscribe events" is performed again after clearing the subscriptions.

Offline mode

This button allows to check whether Updater SDK is in the offline mode or not.

Send No-op

This button allows to test the event subscriptions. It sends invokes a no-op callback at the Demo application side that should display the test string ("Hello, world!"). If there are no subscriptions, no-op handler won't trigger and the test string will not appear.

Cancel Async Call-back

This button allows to cancel any callbacks for async operations running inside specific context. For display purposes all async operations in Demo application run in the same context. If there was a callback subscribed to an async operation, it will not trigger if "Cancel Async Call-back" is performed before the async operation finishes.

Get Current Device Name

This button uses the Updater SDK API to get the device name.

Get Current Device FW Version

This button uses the Updater SDK API to get the currently installed device firmware version.

Get Interval Current Device FW Versions

This button uses the Updater SDK API to get the interval of firmware versions available for installation.

Get Current Device Last FW Version

This button uses the Updater SDK API to get the latest available device firmware version on the server.

Refresh Config Async

This button uses the Updater SDK API to get the latest database config from the server. That data is written down to the "offline/models" folder if it differs from the current data.

Get SDK Version Async

This button gets the current Updater SDK version asynchronously, using a callback to display the result.

Device update

Two options are available to update the firmware for the devices:

- Online update, where firmware is downloaded from the server
- Offline update, where user selects a DFU/BIN file to install from

Online update only allows to install the latest available firmware compatible with currently installed one. It can be performed by clicking the "Update" button on the right side of the device panel. (Make sure that no custom firmware file is selected, offline update would be performed instead).

To perform an update from file, Click the "Browse button" at the top of the main window, then select appropriate DFU or BIN firmware file, then click the Update button to perform an upgrade to selected firmware.

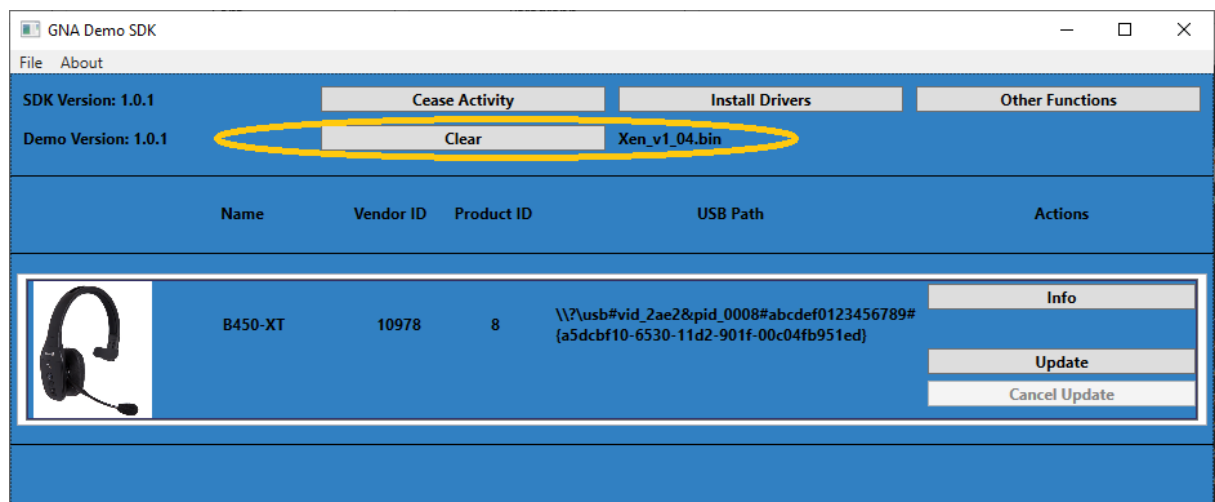


Figure 6. Custom firmware file is selected

If the selected firmware is not compatible with an upgrading device, upgrade error will occur, and device firmware will not be changed.

Offline mode

In this mode, application never attempts any internet connections and only update from file is available for any device.

In order to enable the offline mode, config file needs to be created with "OfflineModeEnabled" property set to true. Config file should be named "GNAUpdaterSdk.cfg" and be located in the same directory as the main application executable.

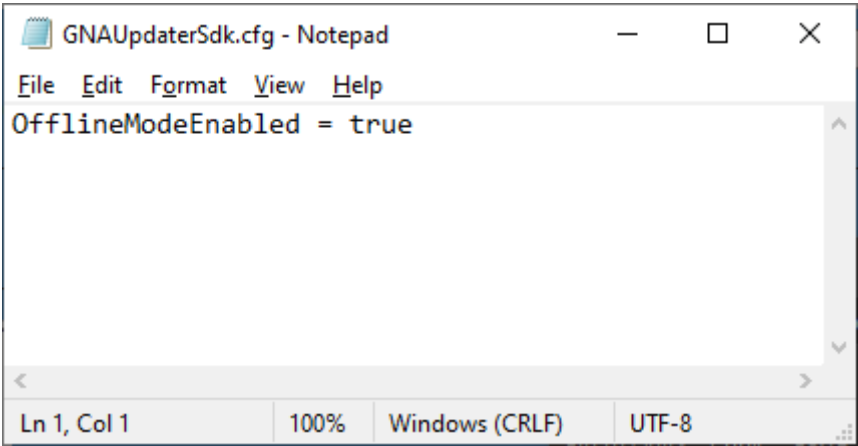


Figure 7. Config file

After this file is created and application is re-launched, the application would be working in offline mode. You can check if the application is currently in Offline mode, by clicking the "Offline mode" button at the "Other functions" screen.

Offline Mode	Get Current Device Last FW Version
Result: Enabled	Last FWVersion: N/A

Figure 8. Offline Mode enabled

In Offline mode, all data is stored in "offline" folder, so if it gets corrupted or removed, the application would work in Opaque mode (more information on modes in GNA Updater SDK API Documentation). This data is updated only when application is connected to the internet, so in order to update or restore this data, user have to turn off the Offline mode, launch the application and wait while data is automatically received from server.