Table of Contents

G-SDK download link & Manual

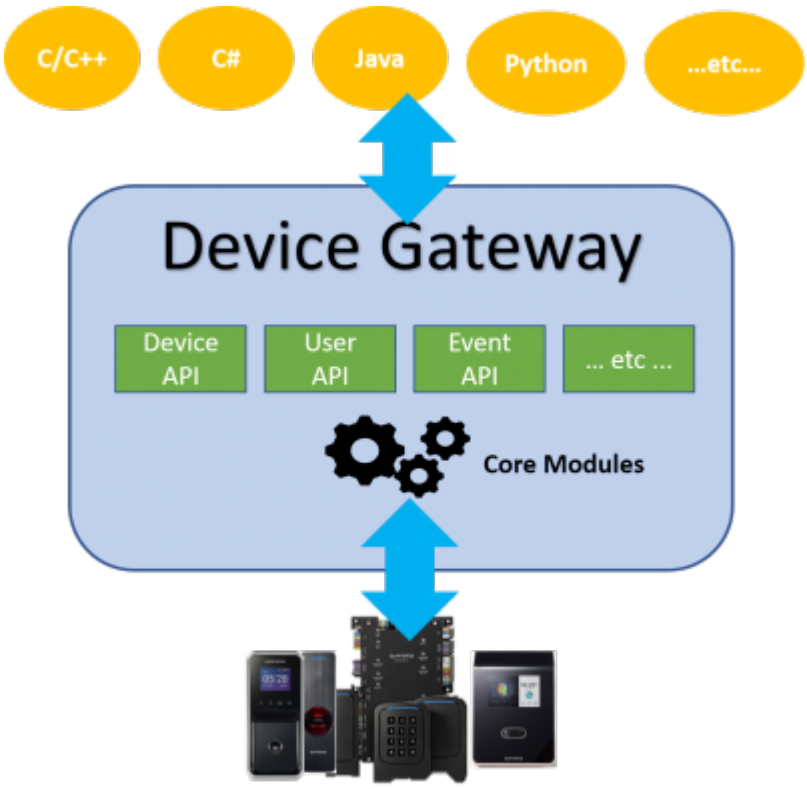
Download : https://github.com/biostar-dev/g-sdk/
Manual : https://biostar-dev.github.io/g-sdk/

G-SDK Overview

G-SDK is one of integration options for communicating with BioStar 2 devices. It is a lightweight, scalable, and cross-platform solution which will expedite your development.
Based on gRPC, it supports many programming languages such as Java, C#, Python, Node.js, Go, etc.

Multi-language support

- Language-neutral IDL
- Native client libraries



[Figure 1. Device Gateway Structure]

Device Gateway that communicates with BioStar 2 device can be run on a server or cloud. **Core modules** use BioStar 2 protocols and written by GO language.

What is different from BioStar 2 Device SDK?

Existing BioStar 2 Device SDK provides C++ library with C#. G-SDK is released to provide flexible development for those who would like to meet the requirements below.

Let's take a look for an easy comparison between BioStar 2 Device SDK and G-SDK.

	BioStar 2 Device SDK	G-SDK
Deployment	• Shared library	• Client libraries • Device Gateway
Supported OS	• Windows • X86 Linux	• Windows • X86/Arm Linux • Mac OS
Supported Language	• C++ • C# sample	• C++, Java, Python, Go, Ruby, C#, Node.js, Android Java, Objective-C, PHP, Web • C#, Java, Python, GO, node samples

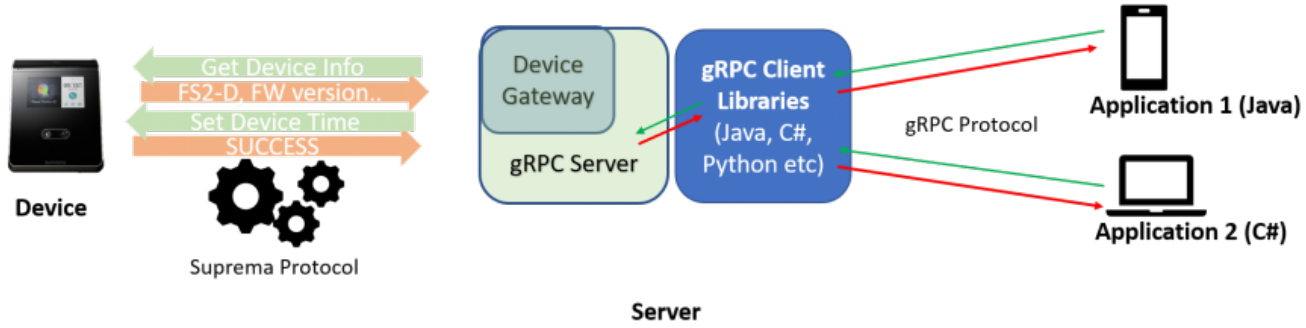
[Figure 2. BioStar 2 Device SDK vs G-SDK]

One of the biggest advantages of G-SDK compare to Device SDK is that it supports various languages. For the last years, Device SDK users have had difficulty using development language other than C++ or C# which is in the sample code.
Since BioStar 2 Device SDK can communicate with C++ library, you have to use C++ language or need to marshall all the APIs and structures in the library into other languages. Currently BioStar 2 Device SDK provides only C# marshalling sample.

[Figure 3. BioStar 2 Device SDK structure]

This is a limitation of BioStar 2 Device SDK due to the structure.

`How G-SDK can support so many languages?`
It is because G-SDK is based on gRPC. gRPC is an open RPC framework that supports many languages developed by Google.
Basically, the device gateway is the one that communicates with BioStar 2 device. Your client applications can connect to it using gRPC client libraries.



[Figure 4. G-SDK structure]

What are the next features?

1. Master Gateway

G-SDK can support up to 1000 device connections with a Device gateway.
Currently, Master gateway that will support up to 100,000 device connections is under development.
Master Gateway helps you handle multiple Device gateways and it can be deployed on a cloud server or provide mobile direct accessibility too.

	Device Gateway	Master Gateway
Role	• Manage devices	• Manage devices through gateways • Manage gateways • Provide additional services such as authentication, rate limit, user synchronization, etc.
Deployment	• Single binary	• Single binary • Docker
Target Applications	• Desktop • Single site	• Mobile • Cloud • Multi sites
Max. Devices	• 1,000	• 100,000
Design Goals	• Performance • Stability	• Scalability • Extensibility

2. Advanced AC API

G-SDK doesn't support advanced AC such as Zone, Wiegand or RS485 but it supports only simple configurations & features.
In the future, G-SDK will support all the APIs that are supported in the BioStar 2 Device SDK : Zone, Wiegand, RS485, etc.

3. More language support

G-SDK provides C#, Java, Python, Go and Node. Furthermore, G-SDK is planning to support Android, iOS, Ruby.