auterion

STARTING SOON

“Controlling MAVLink drones with MAVSDK”

auterion

Controlling MAVLink drones with MAVSDK

Jonas Vautherin, Julian Oes

2019-06-20

Speakers

Jonas Vautherin
MAVSDK Maintainer
Software Engineer, Auterion

Julian Oes
PX4 Maintainer
Senior Software Engineer, Auterion

MAVSDK

MAVSDK is a set of libraries providing a high-level API to MAVLink

MAVSDK

Can be used to build a ground station, like QGC…

MAVSDK

…or an onboard module, like with MAVROS…

MAVSDK

...or both!

Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

Not really...

Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

→ Not really...

Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

→ Not really...

Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

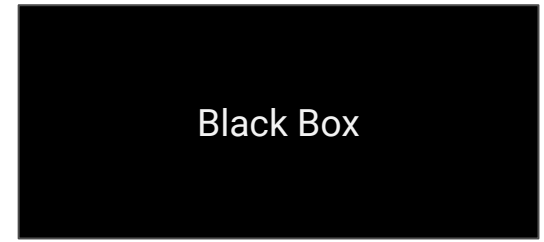
→ Not really...

Python

Swift

Java

JavaScript



Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

→ Not really...

Python

Swift

Java

JavaScript

Black Box

Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

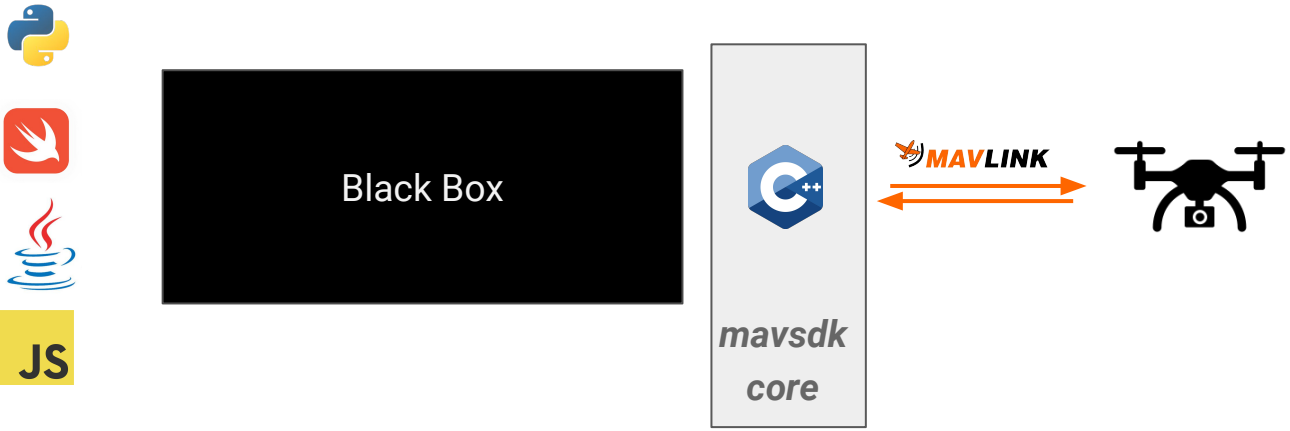
→ Not really...

Python

Swift

Java

JavaScript



Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

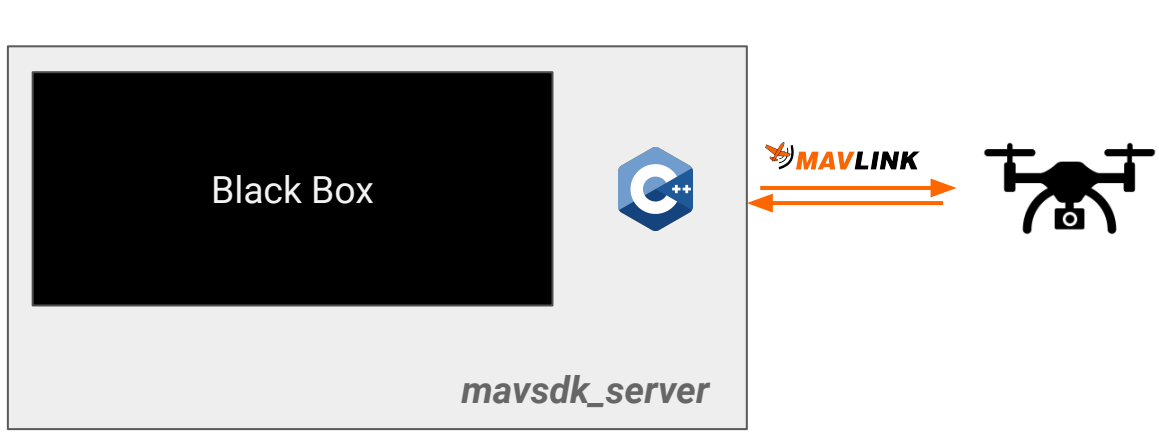
→ Not really...

Python

Swift

Java

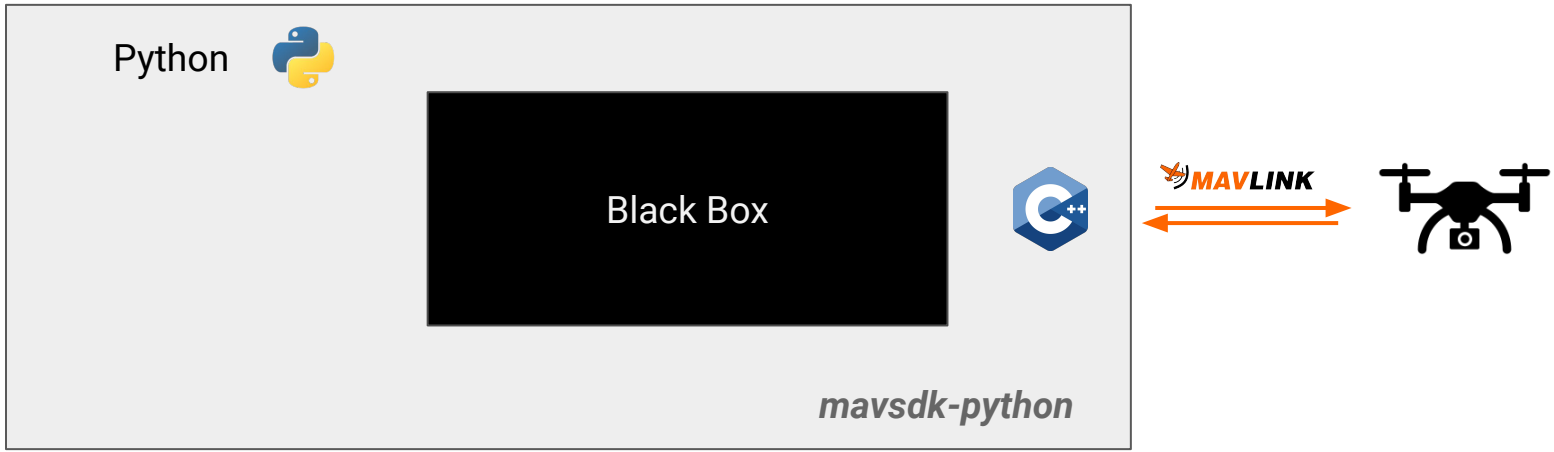
JavaScript



Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

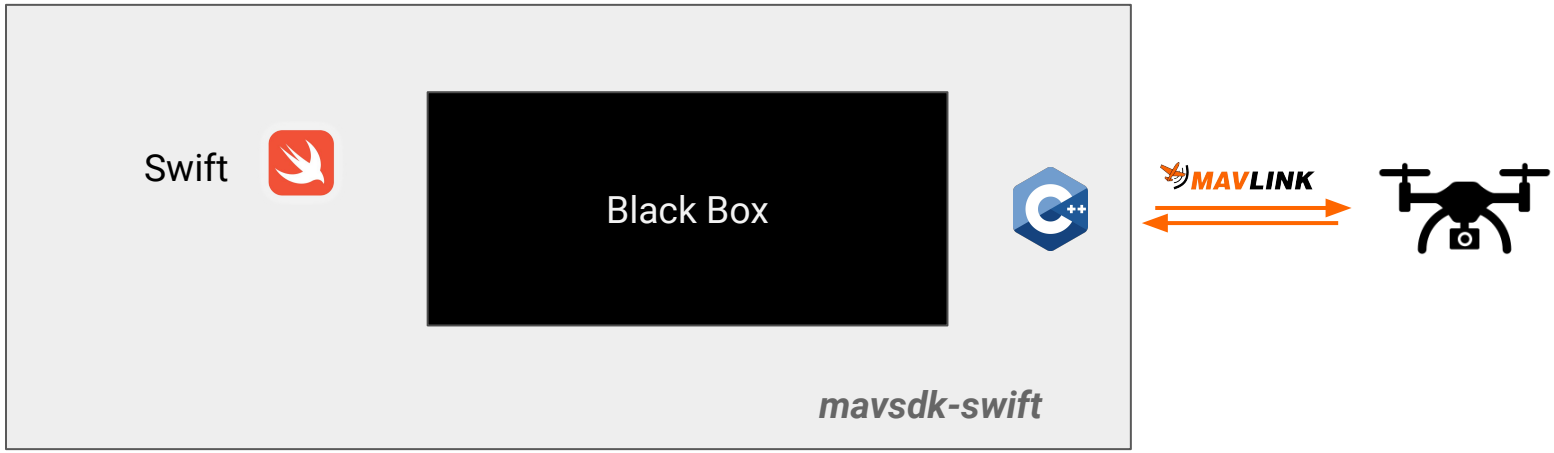
→ Not really...



Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

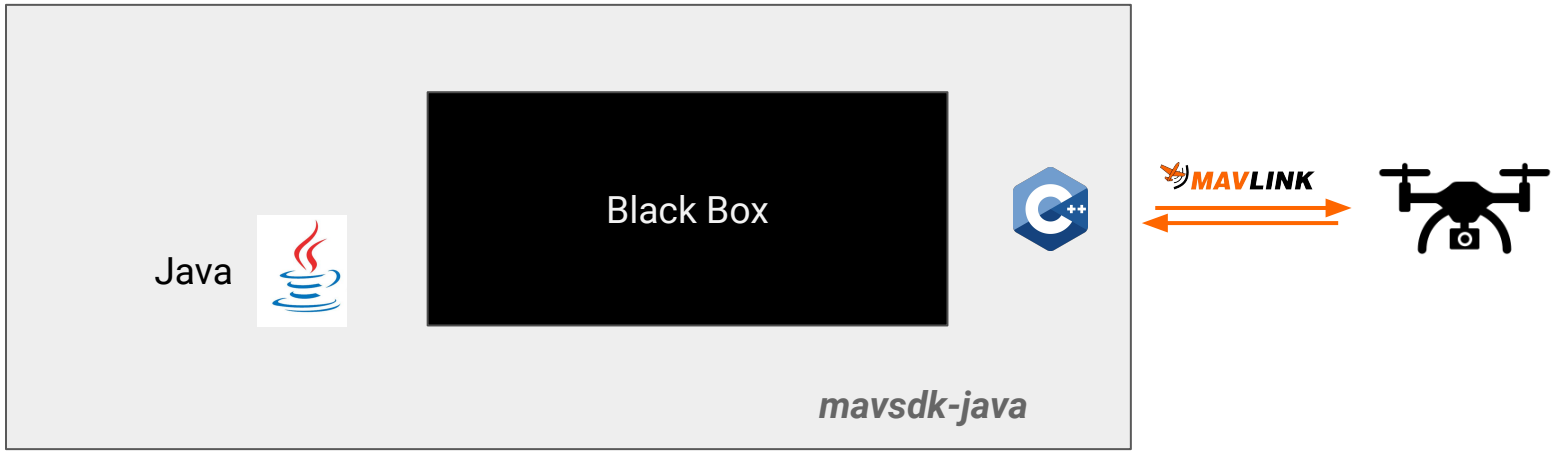
→ Not really...



Why MAVSDK?

- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

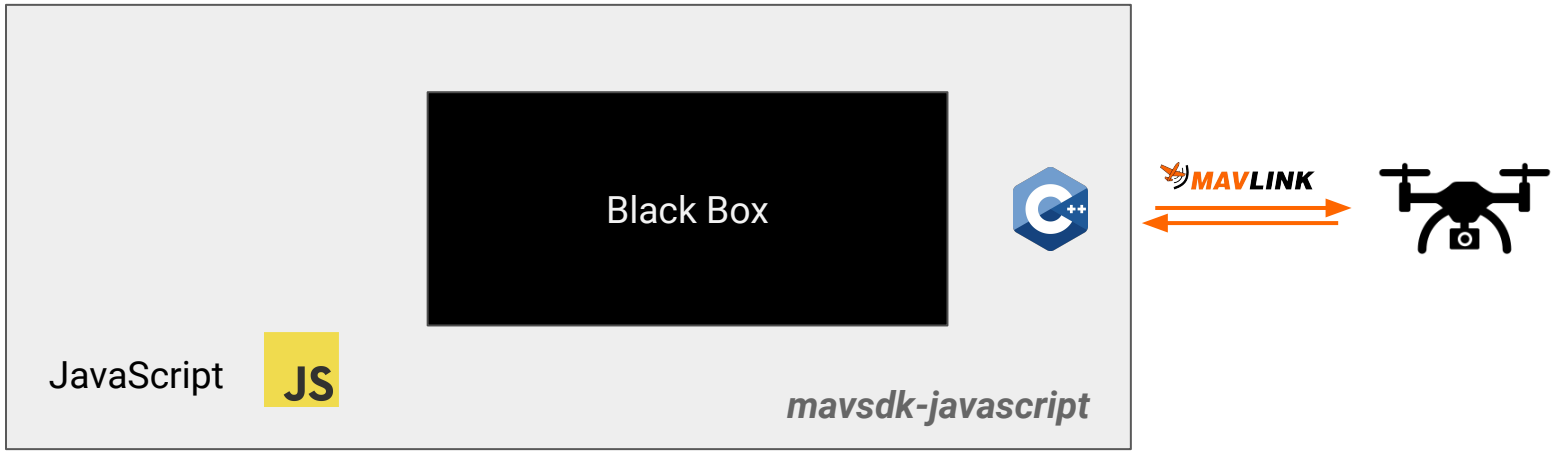
→ Not really...



Why MAVSDK?

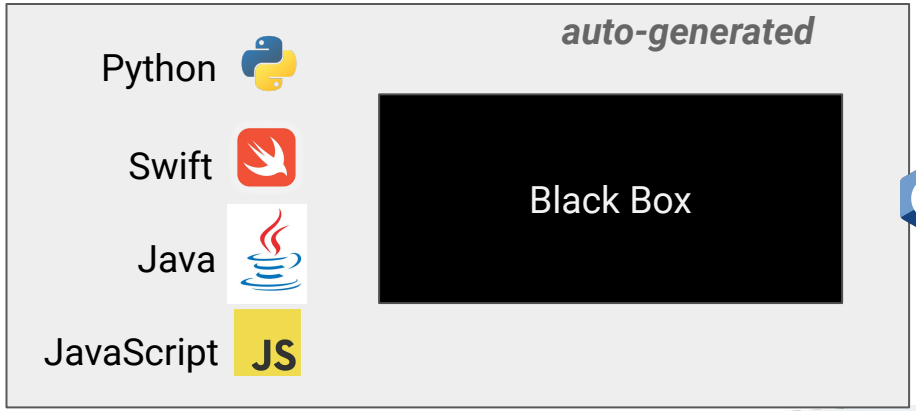
- Is it meant to replace QGC?
- Is it meant to replace MAVROS?

→ Not really...



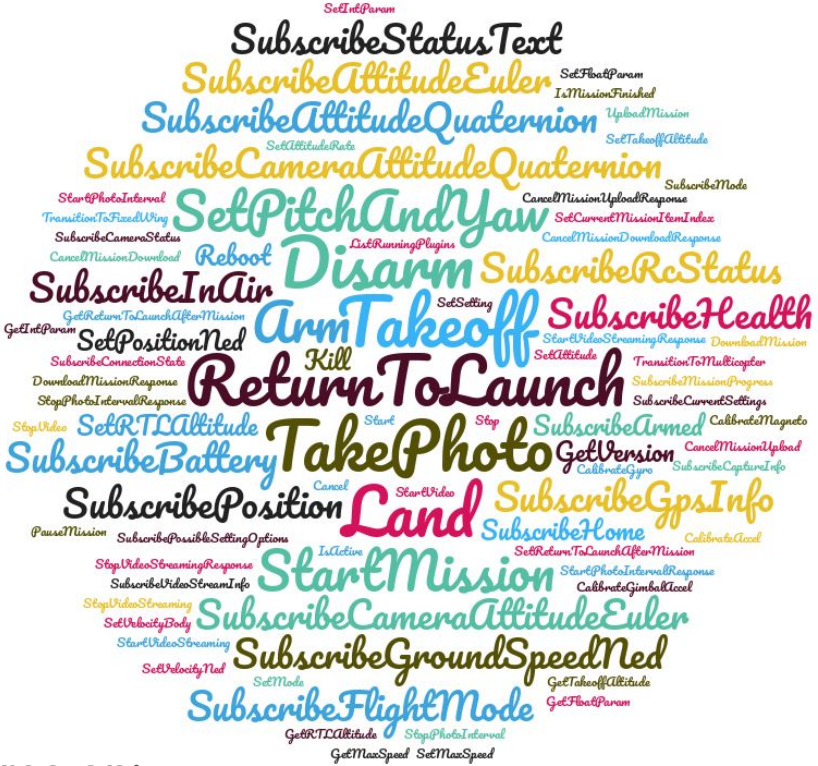
What does that bring?

- One single MAVLink implementation for all platforms/languages
- Everything else (from the C++ API to the language bindings) is auto-generated
- Consistent set of features across MAVSDK
- Stable API



More on the stable API

- Protobuf IDL
- CI tests to ensure stability



Find the API definition / available features on:

https://github.com/mavlink/MAVSDK-Proto

And that’s not all!

- Plugin architecture
- Packaged and distributed
    - C++ ⇒ ***.deb** and ***.rpm** packages
    - Python ⇒ PyPI (*pip install mavsdk*)
    - Swift ⇒ SwiftPM
    - iOS ⇒ Carthage, Cocoapods
    - Java ⇒ Maven/Gradle

In short: if you need to talk MAVLink…

- Easy to install
- Easy to use
- Stable API
- Cross-platform and multi language
- Extensible

…then you should consider MAVSDK!

Current status

Python

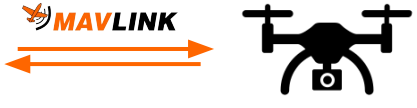
Swift

Java (MVP)

JavaScript (prototype)

Black Box

(not auto-generated)



Roadmap

- Auto-generate ***mavsdk_server***
- Improve C++ packaging and distribution
- Package Java (including Android)
- Get some more experience with MAVSDK onboard
- Auto-generate documentation!

- We always review/support contributions!!!

When can I start using MAVSDK?

- Please do!
    - The C++ core is 3 years old, and used in production
    - MAVSDK-Swift is used in production
    - Other bindings use the C++ core
- Version 0.x because the API is not stable yet (that will come with the *mavsdk_server* auto-generation). It does NOT mean MAVSDK is in beta!
- Documentation may be lacking, but we are more than happy to help on slack/discuss/github

Let’s run some code

Prerequisite: A working SITL (jmavsim, gazebo, …)

If you don’t have it installed, a quick way to run a headless SITL through docker is:

```
$ docker run --rm -it jonasvautherin/px4-gazebo-headless:v1.8.0
```

(source: https://github.com/JonasVautherin/px4-gazebo-headless)

MAVSDK-Python

Quickstart (assuming Python 3.6+)

```
$ pip3 install mavsdk
$ pip 3 install aioconsole
$ apython

> from mavsdk import connect
> from mavsdk import start_mavlink
> start_mavlink()
> drone = connect(host='localhost')
> await drone.action.arm()
> await drone.action.takeoff()
```

MAVSDK

Install latest version

Download package: https://github.com/mavlink/MAVSDK/releases

or build from source:

```
$ git clone -b v0.18.0 https://github.com/mavlink/MAVSDK --recursive

$ cmake -Bbuild -H. && cmake --build build --target install
```

MAVSDK

Install the package (here on Ubuntu 18.04):

```
$ dpkg -i mavsdk_0.18.0_ubuntu18.04_amd64.deb
```

Simple C++ example:

https://gist.github.com/JonasVautherin/027724e93aeec4fc92140b14d1c01d28

Resources

Python: https://github.com/mavlink/MAVSDK-Python

Swift: https://github.com/mavlink/MAVSDK-Swift
RxSwift: https://github.com/ReactiveX/RxSwift (Getting Started)

C++: https://github.com/mavlink/MAVSDK

Docs: https://sdk.dronecode.org/en/

Slack: https://slack.px4.io (#mavsdk)

Forum: http://discuss.px4.io/c/sdk

Call For Contributions

- Report bugs
- Documentation
- Examples (C++, Swift, Python)
- Testing
- CI (Jenkins, Travis, AppVeyor, docker)
- SDK features

(Some) contributors - Thanks!

auterion

Q&A

Resources

Slack: slack.px4.io (#mavsdk)
Documentation: dronecode.org/sdk
Forum: discuss.px4.io/c/sdk
GitHub: https://github.com/mavlink/MAVSDK

Thank you!