

enioka scan  
Say No! to vendor lock-in  
for your barcode scanners

Open-source through reverse engineering

February 2024

Antoine Gonzalez

HAUTE COUTURE  
enioka

# Context

## The wonderful world of barcodes

- Barcodes are everywhere: they are probably the best way to attach digital data to physical objects
- Many “types” of barcodes:



- Some workflows require dedicated devices for decoding them as fast as possible

# Context

## The not-so-wonderful world of barcode scanners

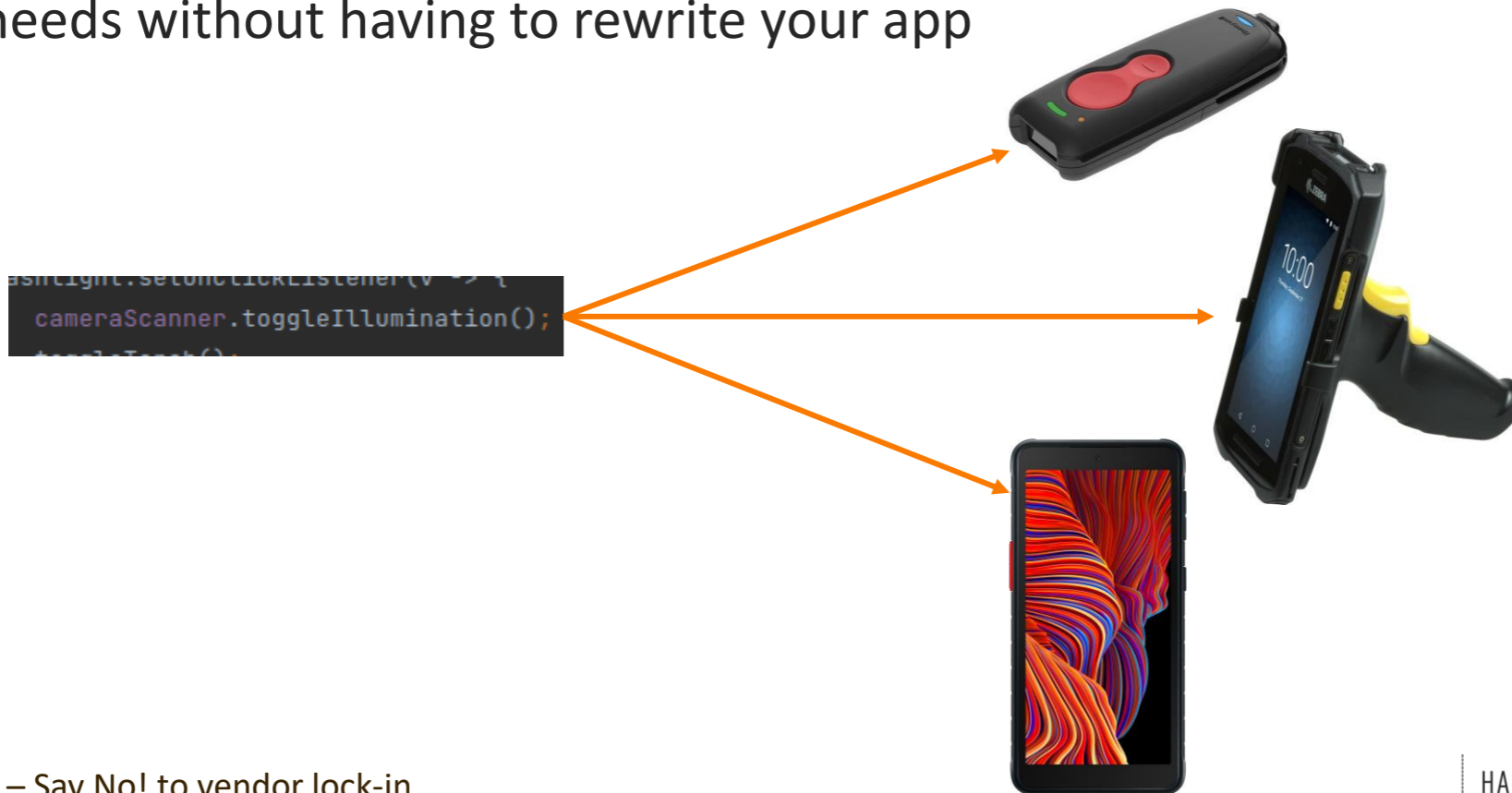
- Wide variety of barcode scanning devices, each with their own use case and complexity
- Wide variety of manufacturers, each with their own APIs, SDKs, licenses, documentation completeness (if available), and most of the time, everything is proprietary
- Usually, picking a manufacturer means being stuck with this choice



# enioka scan

## What it is

- An Android library exposing a common API to interact with barcode scanners from different manufacturers
- Freedom to change manufacturers at will and pick the devices that best fit your needs without having to rewrite your app



# enioka scan

## How it works

- Each manufacturer (or device) exposes their own API, which we implement either with the help of official doc or through reverse-engineering
- The library provides an abstraction layer, which translates common high-level commands to the appropriate device-specific protocol

```
flashlight.setOnClickListeners(v -> {  
    cameraScanner.toggleIllumination();  
});
```

```
private void configureRuntimeItem(String key, String value) {  
    Bundle b = new Bundle();  
    b.putString(key, value);  
    ZebraDwCommand cmd = ZebraDwCommand.create().putExtra(ZebraDwIntents.DW_API_PARAM_EXTRA, b)  
        .setCallback(intent -> Log.d(LOG_TAG, msg: "DW action is done - configured [" + key + "] to value [" + value + "]"));  
  
    // Go!  
    this.broadcastDwAction(cmd);  
}
```

```
void setTorchInternal(boolean value) {  
    if (anyAttributeMissing( caller: "setTorchInternal")) {  
        return;  
    }  
  
    try {  
        if (value) {  
            captureRequestBuilder.set(CaptureRequest.FLASH_MODE, CaptureRequest.FLASH_MODE_TORCH);  
        } else {  
            captureRequestBuilder.set(CaptureRequest.FLASH_MODE, CaptureRequest.FLASH_MODE_OFF);  
        }  
        if (captureSession != null) {  
            captureSession.stopRepeating();  
            captureSession.setRepeatingRequest(captureRequestBuilder.build(), listener: null, backgroundHandler);  
        }  
        isTorchOn = value;  
    } catch (CameraAccessException e) {  
        isTorchOn = false;  
        failed = true;  
        throw new RuntimeException(e);  
    }  
}
```



# enioka scan

## Easy to expand

- API described by a single interface to implement (SDK), letting the library know how to translate each command to communicate with a device
- Commands divided in feature groups, easy to implement or ignore depending on device capabilities
- Nothing changes for the end-user: SDKs are plug-and-play

# enioka scan

## Compatibility so far

- Bluetooth Classic and Low Energy
- Integrated “All-in-one” devices
- Android camera (both with legacy Camera and new Camera2 APIs)

Supported devices	Tested devices	Device type
Any device with a camera	Smartphones, integrated devices...	Camera
Athesi E5L	Athesi E5L	Integrated
Athesi SPA43	Athesi SPA43	Integrated
Bluebird integrated scanners	Bluebird EF500	Integrated
GeneralScan bluetooth ring scanners	GeneralScan GS R5000BT	BT Classic
Honeywell bluetooth scanners	Honeywell Voyager 1602g	BT Classic
Zebra bluetooth scanners using the SSI protocol	Zebra RS5100	BT Classic
Zebra bluetooth scanners using the SSI protocol	Zebra RS5100	BT Low Energy
Honeywell integrated scanners	Honeywell EDA52	Integrated
Proglove devices interfacing with their application	Proglove Glove Mark II	BT Low Energy
Honeywell AIDC / Intermec integrated devices	Honeywell EDA50	Integrated
Koamtac bluetooth KDC devices	Koamtac KDC180	BT Low Energy
M3Mobile bluetooth ring scanners	M3 Ring Scanner	BT Classic
Zebra bluetooth scanners	Zebra RS6000, RS5100	BT Classic
Zebra EMDK integrated scanners	Zebra TC25	Integrated
Any device compatible with Zebra Datawedge	Zebra TC25, Zebra TC27	Integrated

# Roadmap

## What comes next

- An external documentation containing guides and code examples
- Complete separation of core library with existing SDKs into different artefacts, to let you install only what you need
- A standalone app and service published on the Play Store
- Better Bluetooth support (alternate pairing modes, like NFC or pairing barcodes)
- AndroidX support



# enioka scan needs you

## Contribute !

- There are lots of manufacturers out there, each selling multiple devices, with their own specifics and proprietary roadblocks. We cannot test all of them on our own. But maybe you have access to some of them.
- How you can help:
  - By testing the library's compatibility with your device, to let us know if more work is needed
  - By adding a compatibility SDK for a yet-unsupported device
  - By improving existing SDKs and features

[https://github.com/enioka-Haute-Couture/enioka\\_scan](https://github.com/enioka-Haute-Couture/enioka_scan)



**Thank you for listening**  
**Any question?**