

## Introduction

Out of the BOX when RFD40/90 is connected through USB, RFD40/90 barcode scanner is not available for Datawedge and can be accessed only through integrated RFID SDK. If users need to use RFD40/90 scanner for Datawedge along with integrated RFID applications, then user need to configure the device for new co-existence mode.

RFD40 barcode scanners are available for either SDK or Data wedge. Users cannot create Data wedge profile to access RFD40 scanner and associate to same application which initializes the same scanner interface through SDK.

Whenever a new RFD40 device is connected to terminal through USB. Scanning framework takes priority and enumerates the Scanning interface. During this period SDK will not be able to acquire the scanning interface until scanning framework finishes the enumeration which takes around 10 seconds. During this period scanner init API

```
 sdkHandler.setSfCoexistenceModelInit(new SDKHandler.SfCoexistenceResultListener() {  
     @Override  
     public void onResult(SfCoexistenceResult sfCoexistenceResult) {  
  
         switch (sfCoexistenceResult){  
             case SF_COEXISTENCE_FAILURE:  
                 //Failure case  
                 break;  
  
             case SF_COEXISTENCE_SUCCESS:  
                 //Success case  
                 break;  
  
             case SF_COEXISTENCE_NOT_SUPPORTED:  
                 //Reader not configured to SSI datawedge mode  
                 break;  
         }  
     }  
});
```

Note: User needs to call **setSfCoexistenceModelInit** as the first API call immediately after creating sdkhandler instance.

Callback listener will provide the result with success or failure. In case of failure user needs to retry after 10 seconds to set the operation mode again.

Co-existence mode has dependency on the following component version to function

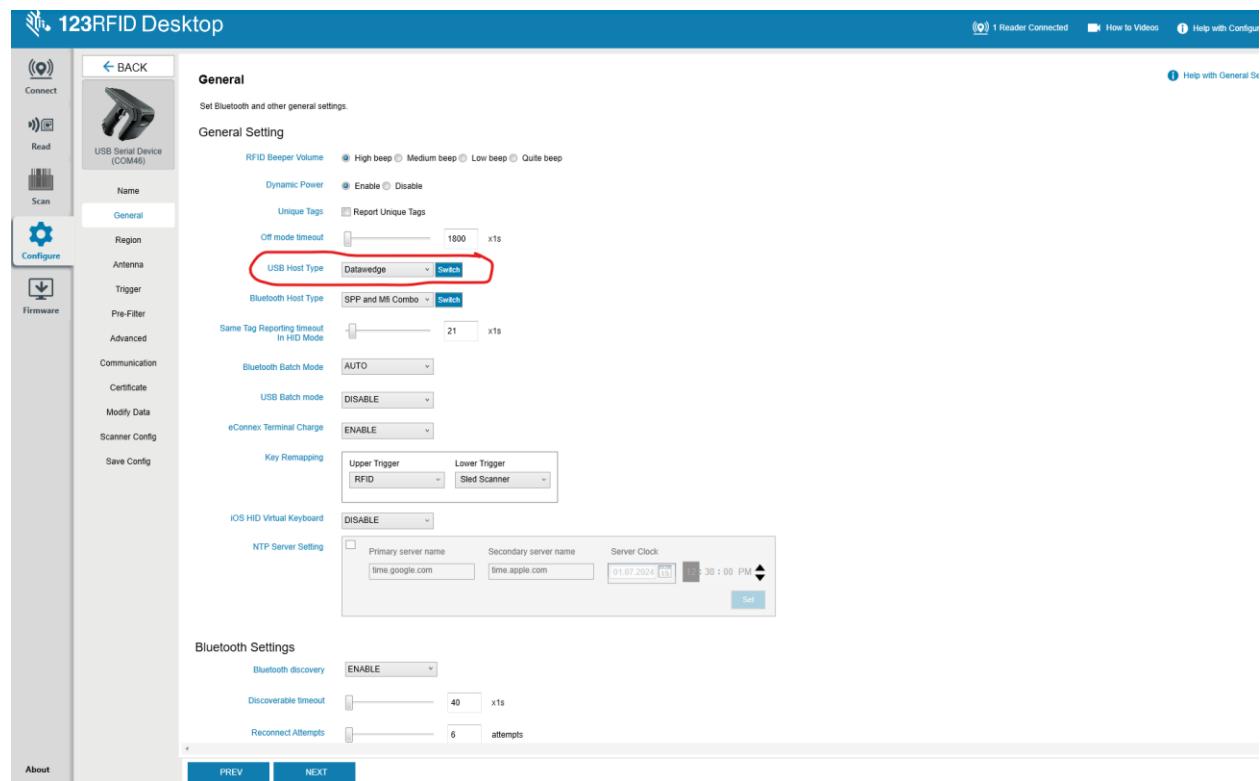
1. Android SDK 92.0.5.236 and above
2. Android Application 92.0.5.236 and above
3. Firmware version
4. Scanning framework 2025 NOV LG release

Note: Please refer to the component version section below to download corresponding components.

Users can also check or confirm if the RFID reader is in the data wedge mode supporting co-existence using the following API.

```
boolean isReaderInUsbDatawedgeMode = sdkHandler.isSSIDwInterfaceCDCScannerAvailable();
```

Please note Device needs to be configured for co-existence mode please refer to the details under firmware configuration.



## Component version

1. Android SDK ([https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/RFID\\_UNIFIED\\_SDK/Q4R2/2.0.5.238/](https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/RFID_UNIFIED_SDK/Q4R2/2.0.5.238/))

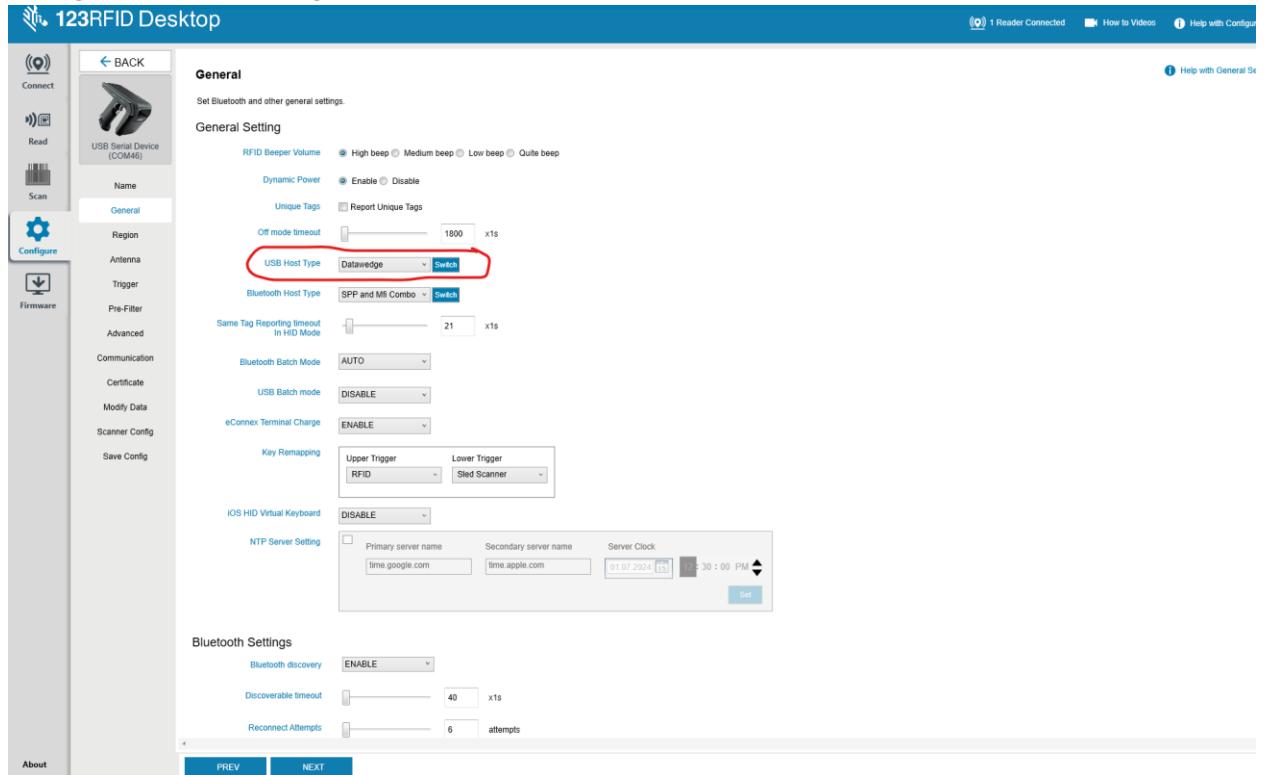
2. Android APP ([https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/RFID\\_UNIFIED\\_SDK/Q4R2/2.0.5.238/](https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/RFID_UNIFIED_SDK/Q4R2/2.0.5.238/))
3. Firmware <https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/Handheld/RFD40/FIRMWARE/DAILY/SAAFKS00-010-K05E0.DAT>
4. Scanning framework  [DataWedgeApp\\_protected\\_aligned\\_signed\\_v5.apk](#)

## Firmware Configuration

Use the FW [SAAFKS00-010-K05E0.DAT](#) to verify data wedge support

Steps to follow:

1. Update the latest shared FW [SAAFKS00-010-K05E0.DAT](#)
2. After successful update perform factory reset
3. By default, out of the box/factory sled is not configured in data wedge mode. User needs to configure the same using below



After above step USB will reset, that is connection to serial terminal will be lost

4. Now user should be able to perform following actions
  - a. Switch to 123RFID sample app and perform barcode scan
  - b. Switch back to DW demo, and scan barcodes
  - c. Steps I & j can be repeated

## Connection failure scenario

In scenario where RFDXX device is configured for USB Datawedge and when you try to connect using incompatible or the old BSP user will observe connection failures.

