

Release Notes - EMDK for Xamarin v2.5

[Important News](#)

[Introduction](#)

[Description](#)

[Device Compatibility](#)

[Installation Requirements](#)

[Usage Notes](#)

[Known Issues](#)

[Part Number and Release Date](#)

Important News

The EMDK for Xamarin provides C# developers with a comprehensive set of tools to easily create powerful line of business applications for enterprise mobility devices and is designed for use with Visual Studio or Xamarin Studio. The EMDK for Xamarin includes class libraries, sample applications with source code, as well as all of the associated documentation to help your applications take full advantage of what our devices have to offer. It also includes an exclusive Profile Manager Technology within the Visual Studio and Xamarin Studio, providing a GUI based development tool to use along with Profile Manager API. This allows you to write fewer lines of code resulting in reduced development time, effort and errors.

EMDK for Xamarin support for all KitKat (Android 4.4.3 or higher) devices will be dropped. - The EMDK for Xamarin v2.5 will be the final version supporting KitKat (Android 4.4.3 or higher) devices. All future EMDK versions will support Lollipop or higher devices only.

EMDK OS update package support limited to KitKat devices - The EMDK v2.5 OS update package distributed with the PC/MAC installation will only support installation on KitKat devices. For all Lollipop or higher devices, the EMDK update for a specific device will be distributed with the latest BSP or security incremental patch.

Online Help: The [EMDK for Xamarin Help](#) documentation is now available only online. The EMDK installation will no longer install the documentation on the development computer.

Online Samples: The [EMDK Samples](#) are now available only online. The EMDK installation will no longer install the samples on the development computer. The online samples can be downloaded and used on the development computer.

Applications accessing EMDK for Xamarin APIs are required to use the `Symbol.XamarinEMDK` component.

Introduction

The EMDK for Xamarin provides developers with the tools necessary for creating applications that target the enterprise mobility 'Xamarin' libraries, such as Barcode. This developer kit is

designed for use with Visual Studio or Xamarin Studio with Xamarin.Android. The Xamarin Business or Enterprise license is required.

Note: For EMDK for Xamarin documentation and discussions, please visit the [Launchpad Developer Community Web site at: https://developer.zebra.com/community/android/xamarin](https://developer.zebra.com/community/android/xamarin)

Description

1. Added support for MX v7.0 in Profile Manager:

- o App Manager – Added new feature to add/remove app from battery optimization list.
- o Cert Manager – Fixed: Deleting uninstalled certificate from UI.
- o Bug Report Manager
 - i. Added ability to disable triggering reports via intent.
 - ii. Added ability to disable taking snapshots.
- o Cellular Manager
 - i. Fixed: Gracefully handle invalid OSX version in CellularMgr

CSP: reject only those parameters that depend on OSX.

CellularData parameters.

- o Wi-Fi
 - i. Added support for 802.11ac and 802.11n.
 - ii. Added 802.11ac and 802.11n parameters in advanced parameters.
 - o Bluetooth Manager – Added new feature to support for admin to add new silent pairing rule by providing just the name of the remote Bluetooth device.
2. Enhanced Barcode Manager API to support Unique Device Identification (UDI) barcodes to be used with TC51/TC56 healthcare devices:
- o Added support to enable/disable the UDI Standards such as GS1, HIBCC and ICCBBA. Refer to the parameter “ScannerConfig.UdiParameters” for details.
 - o Added new enum in ScannerConfig:
 - i. ScanMode - Describes the available scanning modes such as Single and UDI.
 - o Added new reader parameter in ScannerConfig.ReaderParameters.ReaderSpecifics.ImagerSpecifics:
 - i. scanMode - Sets the scan mode. This allows to select one type from the ScanMode enum.
 - o Enhancements to ScanDataCollection class:
 - i. Added new property "LabelIdentifier()" to get the decoded type of the barcode. Note: This is especially applicable for UDI decoding. Use "LabelType" defined in each data item to get the individual barcode label types.
 - ii. Added new property "TokenizedData()" to get the list of tokenized data received after processing the raw scanner data.

- o Added new class TokenizedData to provide access to tokenized scan data:
 - i. "Tokens" - Returns all the tokens with it's data.
 - ii. GetTokensByKey(String key) - Returns the tokens for the given token key. If data has multiple values, it returns all the tokens.
- o Update to ScannerResults enum:
 - i. Added new enum value "DATA_PARSING_FAILURE" to notify when data parsing failed.
- 3. Enhanced DataCapture feature in the Profile Manager:
 - o Added support to enable/disable the UDI Standards such as GS1, HIBCC and ICCBBA to be used with TC51/TC56 healthcare devices. Refer to “UDI parameters” section for details.
 - o Scanning Mode - Sets the scanning mode. This allows to select one mode (Single or UDI) at a time.
 - o Use auto when not supported - Use auto when the scanner type is not supported while importing the profile to a different device.
- 4. Enhanced SerialComm API:
 - o Feature type SERIALCOMM has been deprecated and introduced a new feature type SERIALCOMM_EX.
 - o New classes added “SerialCommMgrEX”. You may get a new `SerialCommMgrEX` instance by calling `EmdkManager.GetInstance(EMDKManager.FEATURE_TYPE.SerialcommEx);`
 - o Added support to enumerate available serial ports on a device.
 - i. `SerialCommMgrEX.SupportedPorts` - Returns list of supported serial ports information.
 - o Added support to select a single serial port by passing a PortInfo object.
 - i. `SerialCommMgrEX.GetPort(SerialPortInfo serialPortInfo)` - This method returns the SerialComm object if the serial port info specified is valid. The SerialComm object created will be singleton object for a specific serial port.
- 5. Fixed: SimulScan issue where template parameters are not getting set after a timeout or cancel event.

Device Compatibility

This software release has been approved for use with the following devices.

Device	Android KitKat	Android Lollipop	Android Marshmallow
MC18	*	*	NA
MC40	*	NA	NA
MC92	*	NA	NA
TC55	*	NA	NA

TC70	*	*	NA
TC75	*	*	NA
TC8000	*	*	NA
ET5X	NA	*	*
WT6000	NA	*	NA
TC51	NA	NA	*
TC56	NA	NA	*
MC67	*	NA	NA
TC75x	NA	NA	*
MC32N0	NA	*	NA
TC70X	NA	NA	*

Installation Requirements

The following software must be installed prior to using the EMDK for Xamarin.

- Microsoft® Windows 7 (32-bit and 64-bit) or Microsoft® Windows 8 (32-bit and 64-bit) or Microsoft® Windows 8.1 (32-bit and 64-bit) or or Mac OS X 10.10 or higher.
- [Java Development Kit \(JDK\)](#) v7u45 or higher.
- Visual Studio 2013, Visual Studio 2015, Visual Studio 2017 or Xamarin Studio 6.0, Xamarin Studio 6.1, Xamarin Studio 6.2, Xamarin Studio 6.3.

Usage Notes

None.

Known Issues

None.

Part Number and Release Date

EMDK-X-0205069
August 2nd, 2017

© 2015-2017 Zebra Technologies Corp. All rights reserved.

