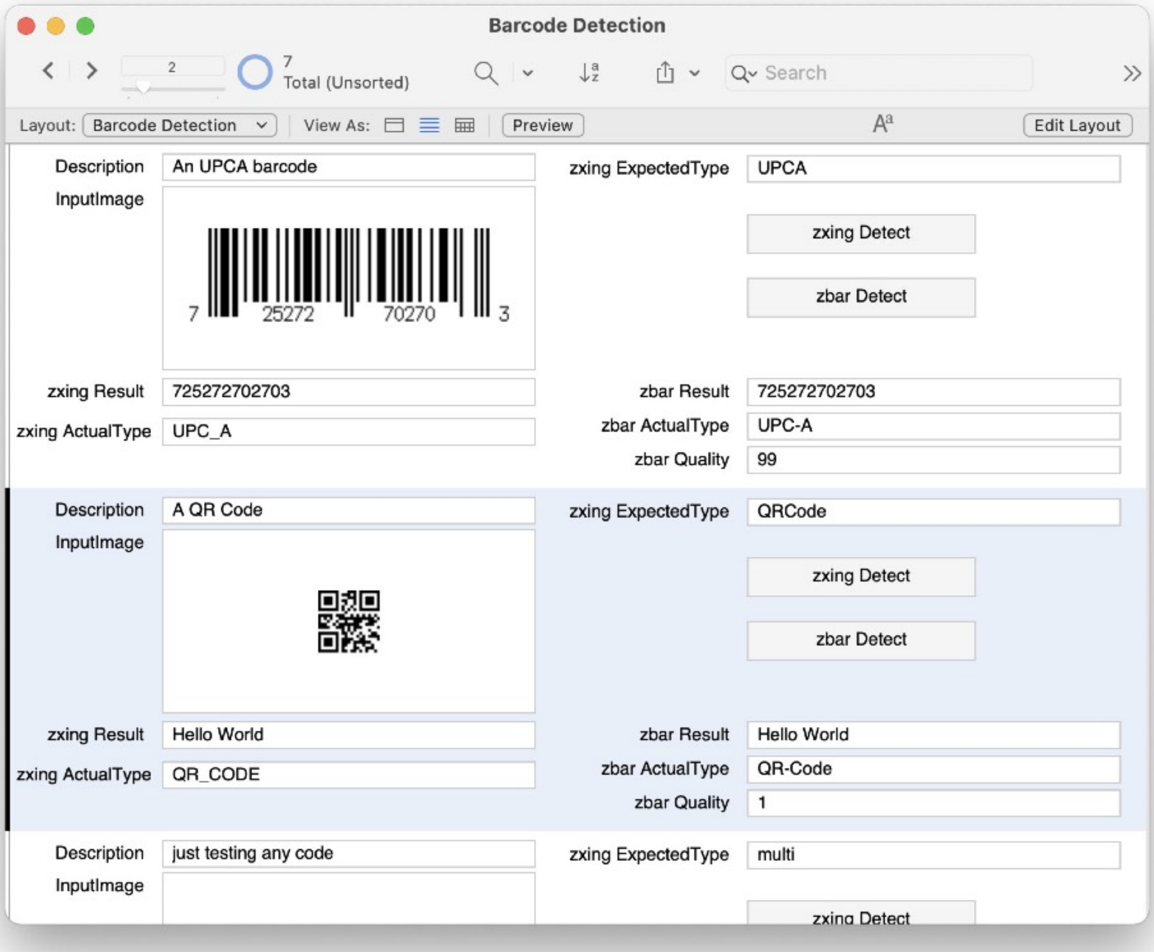
Using zbar library with FileMaker

You may know that our MBS FileMaker Plugin has functions for barcode generation and recognition. The Barcode.Detect function is based on the open source zxing library can recognize various types, but sadly not all QRCodes recently.

To solve this we add support for the zbar open source library with our new Barcode.Scan function. Since the library is GPL, you have to include the library file with the application and load it at runtime with Barcode.LoadLibrary function. You may get a zbar library from the project website or from our Libs folder in download section. Our example project shows how to load the library at startup of application. On Linux you may just be able to install libzbar0 package and get libzbar installed.



Description: An UPCA barcode
InputImage
zxing ExpectedType: UPCA
zxing Result: 725272702703
zxing ActualType: UPC_A
zbar Result: 725272702703
zbar ActualType: UPC-A
zbar Quality: 99

Description: A QR Code
InputImage
zxing ExpectedType: QRCode
zxing Result: Hello World
zxing ActualType: QR_CODE
zbar Result: Hello World
zbar ActualType: QR-Code
zbar Quality: 1

Description: just testing any code
InputImage
zxing ExpectedType: multi

When you like to scan a barcode, get a picture for the barcode. Make sure contrast is good and your barcodes are fully visible with a bit of border space around. Since scanning is done in black and white, you can preprocess the image to increase visibility with auto leveling, e.g. GMImage.Level function. You call Scan() function and pass the picture and you receive a JSON array with barcodes found. You may get multiple result with different quality values, so you may pick the highest one. Sometimes a few lines or some text is recognized as a barcode. To increase reliability you may limit the list of barcodes types you like to have and thus ignore all others.

Supported types includes EAN-13/UPC-A, UPC-E, EAN-8, Code 128, Code 39, Interleaved 2 of 5 and QR Code.

Our tests show that zbar can recognize a few barcodes, which didn't work with Barcode.Detect. But since we also found EAN and UPC barcodes recognized in zxing, but not in zbar. Well, you may eventually use both to see what recognizes what.

Please ty with 11.3pr3 or later and please don't hesitate to contact us with your questions.

For macOS and iOS see also Vision.DetectBarcode and Vision.SupportedSymbologies functions to use Apple's Vision framework.