GS1 Barcodes

A **GS1 barcode** is a standardized type of barcode or 2D code that can contain multiple pieces of data, and identifies what each piece means.

For example, it could contain an item code, a batch number and an expiry date, all in one barcode.

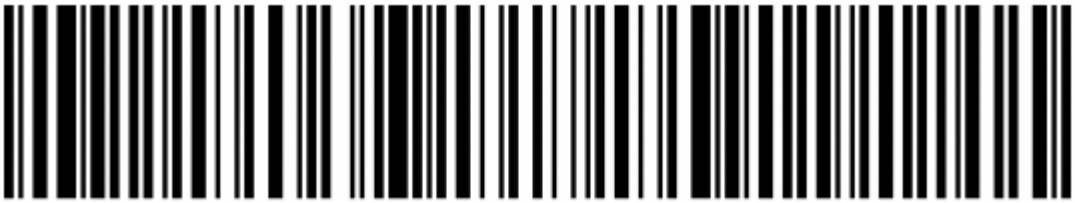
COBI.wms supports these types of codes and will use the different pieces of data found in the code to fill in the correct fields in the app.

Possible formats

GS1 barcodes can be encoded with the physical formats **Code 128**, **DataMatrix**, or **QR Code**.

These are then referred to as **GS1-128**, **GS1 DataMatrix**, and **GS1 QR Code**.

All three of these formats are supported by COBI.wms, as long as your hardware supports them.



GS1-128



GS1 DataMatrix



GS1 QR Code

All three codes above contain the same data.

As you can see, DataMatrix and QR Code can be much smaller, since they are two dimensional.

The difference between DataMatrix and QR Code is that DataMatrix was designed for industrial use. It is more resilient against damage. A dirty or torn label might still readable by the scanner when

DataMatrix is used. (QR Code can also be damage-resistant, but not as much as DataMatrix.)

These characteristics make **GS1 DataMatrix** the optimal choice for modern warehouse applications, but if your scanner hardware is not capable of scanning 2D codes, then you must use **GS1-128**.

Note that using a modern format like DataMatrix might also make your barcodes incompatible with other, more primitive scanning solutions. So in case COBI.wms is not the only software you use for scanning your codes, or in case you want to make it possible for your business partners to scan your codes (who might not be using COBI.wms), then you might want to use GS1-128, or not use GS1 at all and instead print separate barcodes for separate pieces of data (like one for item code and another one for serial number) for maximum compatibility with old scanning solutions.

Supported data fields

The data fields in a GS1 code are identified with standardized numeric codes called **Application Identifiers (AI)**.

Currently, COBI.wms supports the following AI codes:

AI	Meaning	Function in COBI.wms
00	SSCC	Add a package with the scanned SSCC in delivery packing slips
01	GTIN	Identify item (item code or barcode)
02	Contained GTIN	Identify item (item code or barcode)
10	Batch number	Auto-insert batch number, or lot number field of serial number
11	Production date	Auto-insert production date field for batch number
15	Best before date	Auto-insert best-before/expiry date field for batch number
17	Expiry date	Auto-insert best-before/expiry date field for batch number
21	Serial number	Auto-insert serial number
30	Number of items	Auto-insert quantity
37	Contained quantity	Auto-insert quantity
90	Item Code	Auto-insert Item
91	Bin location Code	Auto-insert Bin location
240	Product number	Identify item (item code or barcode)
310	Net weight (kg)	Auto-insert quantity
330	Gross weight (kg)	Auto-insert quantity

Common mistakes

There are a number of common mistakes that can be made when generating new GS1 barcodes.

Parentheses in data

When the content of a GS1 code is displayed as human-readable text, the AI codes are usually enclosed in parentheses, like this:

```
(01)1234568901234(10)1234(17)221200
```

If you use a text like this, including the parentheses, and use the Code 128, DataMatrix, or QR Code physical formats to encode that text, the barcode generation might work, but the result will **not** be a valid GS1 code. Your hardware might be able to scan the result, and COBI.wms might receive some data, but it won't be handled correctly because the contents will be invalid.

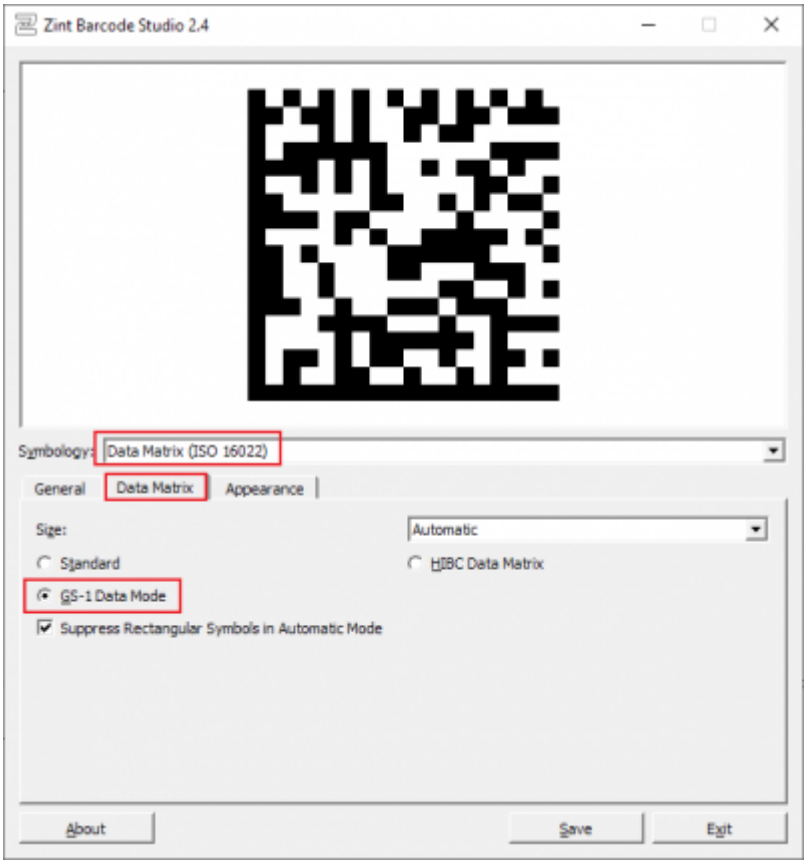
Instead, you must configure your barcode generating software to operate in **GS1 mode** and enter the AI codes in an application-dependent way. For example, in the program **Zint Barcode Studio**, the AI codes must be enclosed in square brackets, like this:

```
[01]1234568901234[10]1234[17]221200
```

The barcode generation software you use might allow normal parentheses and handle them correctly, but it's important to make sure that your software doesn't simply put the parentheses into the resulting barcode as plain text.

(The actual contents of a valid GS1 code don't contain any kind of parentheses at all. The AI codes and data contents appear bare, without any characters in between, except the FNC1 special character that may be used to mark the end of a field.)

Not setting GS1 mode



As already mentioned in the previous section, it's important to tell your barcode generation software that you want to generate a valid GS1 code; it's not enough for the contents to be valid GS1 contents.

This is because all the physical formats used for GS1 codes (Code 128, DataMatrix, and QR Code) contain special symbols at the beginning which identify what type of data they contain. If they don't contain the special symbol identifying the barcode as a GS1 code, then the hardware and/or the software (COBI.wms) might not correctly handle the code.

How to set your barcode generation software to GS1 mode depends on the software being used. For example, in Zint Barcode Studio, it's selected in the format-specific configuration options after selecting the format from the Symbology drop-down.

NOTE: If you cannot find a way to configure your barcode generator to use a GS1 mode, but it allows you to manually insert the special, invisible barcode character called **FNC1** (Function Character 1), then putting an FNC1 at the beginning of the data should also work, since this is how GS1 codes are generally identified.

Incorrect or missing FNC1

Some GS1 data fields have a variable length. For example, while the GTIN (AI 01) must be exactly 14 digit long, the serial number (AI 21) can contain between 1 and 30 alphanumeric characters.

When such a variable-length field is not the last field, and its length is less than maximum, then the end must be marked with a special invisible character called FNC1 (Function Character 1).

If the barcode generation software you use allows you to specially mark the AI codes, like square brackets in Zint Barcode Studio (see above), then FNC1 should be inserted automatically to all the places where it's necessary. However, if the software requires you to specify the raw contents of the barcode manually, it's important to put the FNC1 in all the correct places.

For example, let's say we're using barcode generation software that doesn't know anything about GS1, but allows us to manually insert FNC1 characters in the data, shown here as `[FNC1]`. Then the following barcode contents should be turned into a valid GS1 code:

`[FNC1]01123456890123410123[FNC1]17221200`

Explanation:

- The data starts with FNC1 to identify the barcode as a GS1 barcode. Then multiple combinations of `AI code` (red) + `data` follow.
- The first data field is a GTIN (AI 01). This field must contain exactly 14 digits, so it doesn't require an FNC1 at the end.
- The next data field is the batch number (AI 10) which may be between 1 and 30 characters long. Our batch number is only 4 characters long, so it must be ended with FNC1.
- The last field is the expiry date (AI 17) which must be exactly 6 digits, but it wouldn't need an FNC1 at the end anyway because it's the last field.

From:
https://docs.cobisoft.de/wiki/ - **COBISOFT Documentation**

Permanent link:
https://docs.cobisoft.de/wiki/cobi.wms/gs1_barcodes

Last update: **2025/05/22 11:53**