GPL4

Copyright © 2010 – 2015 JetLab S.r.l. All rights reserved.

4i is a JetLab S.r.l. trademark.

Microsoft, Windows, Windows NT, Windows XP, Windows Vista, Windows 7 and the Windows logo are trademarks or registered trademarks of Microsoft Corporation U.S. or other countries or both.

Java and all Java based trademarks and logo are trademarks or registered trademarks of Sun

Microsystems, Inc. in the U.S. or other countries, or both.

UNIX is a registered trademark in the United States and other countries licensed exclusively

through X/Open Company Limited.

IBM Server, IBM i5 iSeries 400, OS/400, IBM i, i5/OS for Power Systems, AS/400 are a trademark of IBM Corporation, U.S.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Contents

Contents 3
General Description 4
Requirements 4
Execution 4
Components Analysis 6
PDF Model 6
Sources 7
Compiling 10

Copyright © 2010 – 2015 JetLab S.r.l. All rights reserved

General Description

This White Paper shows how to create an RPG program which, through ***GPL4i***, can generate a PDF document containing an image and a barcode. The PDF will be generated on the IBM I.

Requirements

These are the requirements that need to be met in order to complete the activities proposed in this document

- ***GPL4i*** is already installed and configured on the IBM i system.
  `http://www.jetlab.com/downloads/gpl4i/GPL4i-V2R0M0-WP001_Installation-ENG.pdf`
- The ***GPL4i*** User Tool library is already installed on the IBM i system.
  `http://www.jetlab.com/downloads/gpl4i/GPL4i-V2R0M0-WP002_Examples-ENG.pdf`

This document is oriented towards IBM i programmers with knowledge of RPG language and the ILE environment.

Execution

These are the steps to activate the RPG program that generates the PDF and verify the results.

__ **Step 1**

Connect as user R4SP020000 and activate the runtime and the default device.

```
STRSBS JTJT040000
STRR4DEV JTJT040000
```

__ **Step 2**

Add the GPL4IUT library in the library list

```
ADDLIBLE GPL4IUT
```

__ **Step 3**

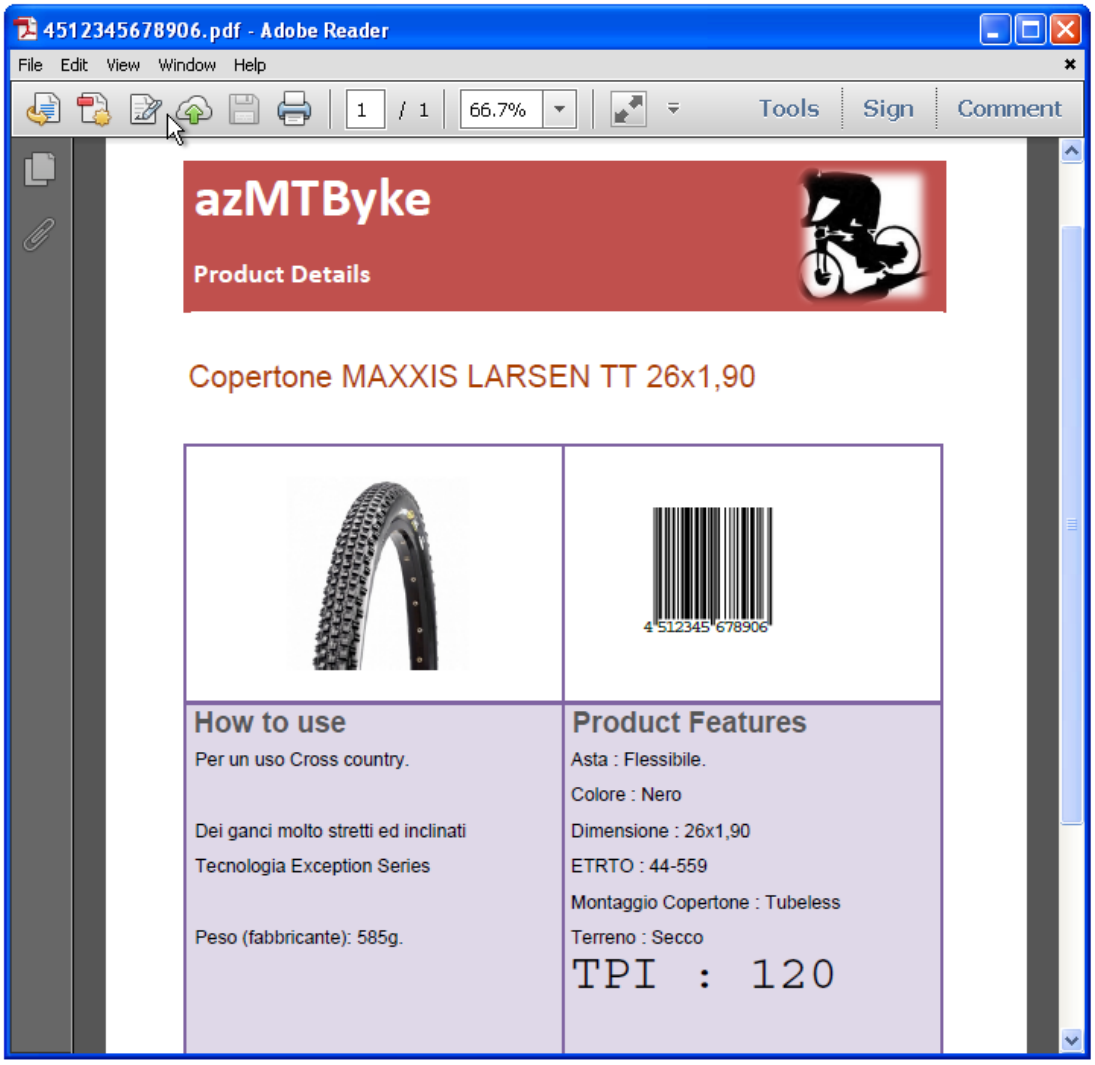
Run the PRDDTL program

```
CALL PGM(PRDDTL) PARM('JTJT040000')
```

__ **Step 4**

Verify that the document
`/gpl4iut/temp/4512345678906.pdf`
was generated. Open the document with a PDF Reader.

This is the result



Copyright © 2010 – 2015 JetLab S.r.l. All rights reserved

Components Analysis

To obtain the PDF document use the following resources.

PDF Model

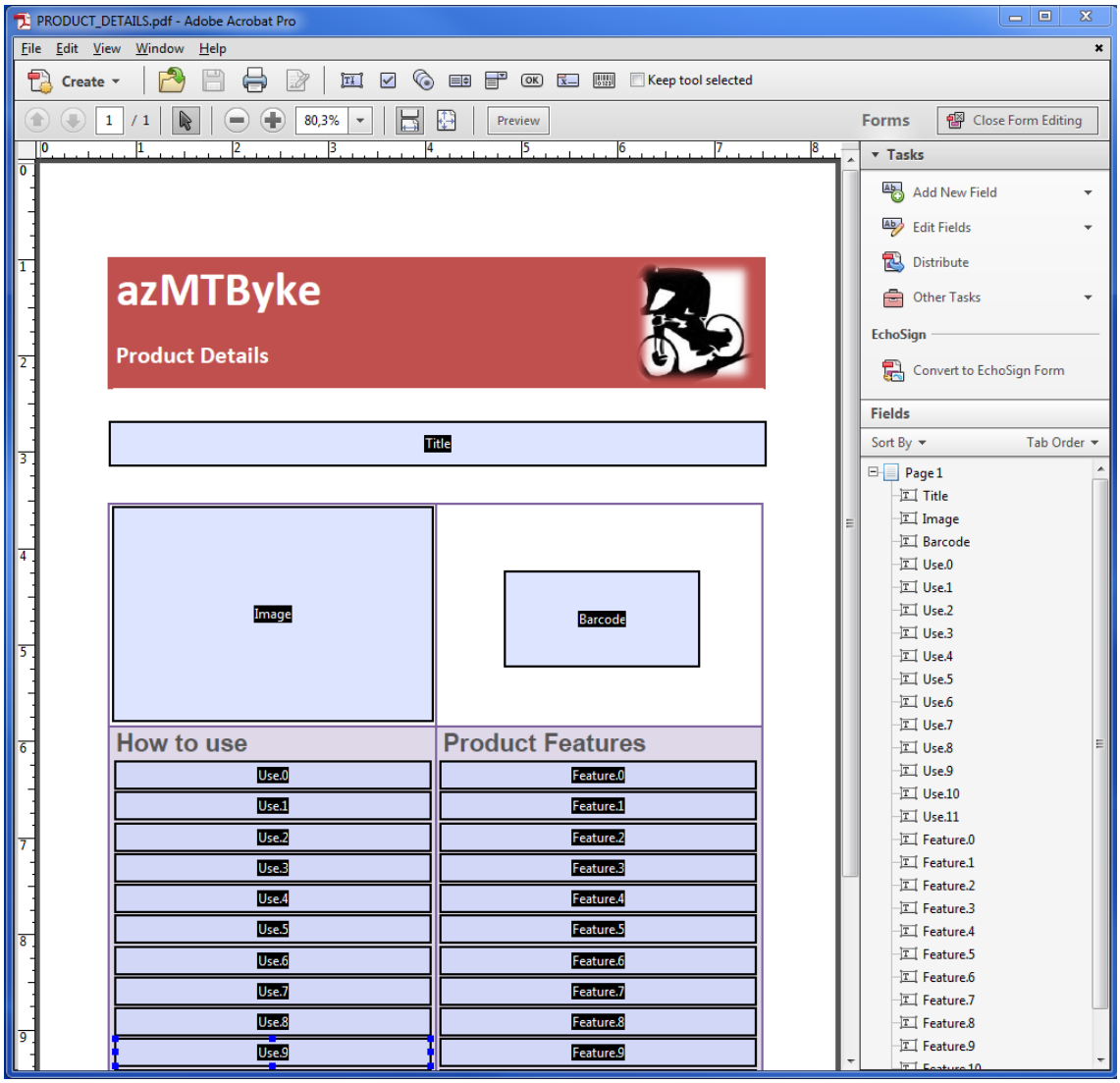
The form utilized is contained in the file

`/gpl4iut/PRODUCT_DETAILS.zip`

and is called

`PRODUCT_DETAILS.pdf`

As shown in the following image, it is a PDF which contains the following text fields:



The RPG source

`GPL4IUT/QRPGLESRC(PRDDTL)`

will be the reference for these names in order to compile the form with the program data.

Sources

The RPG source

`GPL4IUT/QRPGLESRC(PRDDTL)`

carries out the following operations to generate the document:

- Includes definitions of GPL4i API

```
D/copy *LIBL/QRPGLECPY,GPL4I
```

- Decides which model to generate

```
c                   callp     GPLSetDataModel(
c                              'PRODUCT_DETAILS'+x'00'
c                             :'/gpl4iut/PRODUCT_DETAILS.zip'+x'00'
c                             )
```

- Opens a new document

```
c                   callp     GPLOpenDoc
```

- Assigns user data to the document (Product_code = 4512345678906)

```
c                   eval      val = code
c                   callp     GPLAddDMInfo(
c                              'Product_code'+x'00'
c                             :%addr(val)
c                             :%len(%trim(val))
c                             :handle
c                             )


c                   callp     GPLSetFolder('Products'+x'00')
```

- Assigns a name to the .pdf file to be generated and activates the attribute replace=true to overwrite the document if it already exists.

```
c                   eval      val = 'true'
c                   callp     GPLAddSrvInfo(
c                              'save'+x'00'
c                             :'replace'+x'00'
c                             :%addr(val)
c                             :%len(%trim(val))
c                             )
c                   callp     GPLSetPdfName(
c                              '/GPL4IUT/temp/'
c                             +%trim(code)+'.pdf'+x'00'
c                             )
```

- Decides which form it should compile from the available forms in the model

```
c                   callp     GPLNewForm(
c                              'PRODUCT_DETAILS.pdf'+x'00'
c                             )
```

- Initializes the fields of the form (eg. the Title field)

Copyright © 2010 – 2015 JetLab S.r.l. All rights reserved

```
c           eval      val='Copertone MAXXIS LARSEN TT 26x1,90'
c           callp     GPLSetTextFld(
c                     'Title'+x'00'
c                     :%addr(Val)
c                     :%len(%trim(Val))
c                     )
```

- Initializes the EAN type Barcode field

```
c                   eval      val = code
c                   callp     GPLSetBarcode(
c                              'Barcode'+x'00'
c                              :'EAN'+x'00'
c                              :%addr(val)
c                              :%len(%trim(val))
c                              )
```

- Initializes the image field

The real dimensions of the image are also reduced by 60%

```
c                   eval      val = 'percente'+x'00'
c                   callp     GPLSetFldAttr(
c                              'scale'+x'00'
c                              :%addr(val)
c                              )
c                   eval      scale100 = 60.0
c                   callp     GPLSetFldAttr(
c                              'scale100'+x'00'
c                              :%addr(scale100)
c                              )
c                   eval      val
c                             ='http://www.jetlab.com'
c                             +'/gpl4i/gpl4iut/copertone.png'
c                   callp     GPLSetImage(
c                              'Image'+x'00'
c                              :%addr(Val)
c                              :%len(%trim(Val))
c                              )
```

- Ends form compilation

```
c                   callp     GPLSetBreak
```

- Closes the document

```
c                   callp     GPLCloseDoc
```

- Activates PDF creation by giving it over to the Device JTJT040000

In case of errors the glpErr variable will contain information on the error encountered.

```
D gplErr          ds                  likeds(GPLErr_T) based(gplErr_ptr)
D gplErr_ptr      s               *
```

```
...
c                   if        GPLRun(device) = 0
c                   eval      gplErr_ptr = GPLGetCurErr
c     gplErr.MSG    DSPLY
c                   endif
```

Copyright © 2010 – 2015 JetLab S.r.l. All rights reserved

Compiling

The PRDDTL source is compiled in Bid with the following Service Programs

R4SP020000/GPL4I

```
ADDLIBLE JTJT040000
ADDLIBLE R4SP020000
ADDLIBLE GPL4IUT
CRTRPGMOD MODULE(GPL4IUT/PRDDTL) DBGVIEW(*ALL)
CRTPGM PGM(GPL4IUT/PRDDTL) BNDSRVPGM((GPL4I))
```