Maniplus, part 1 & 2: *"Accept or Cancel"*

Pre-Conference Session, IBUC 2018

Statistics Netherlands

Overview

- This presentation is about the new Blaise 5.4 feature 'Manipula Dialogs'

- It will cover…
  - Dialog basics
  - The designer
  - Using lookups
  - Using EDIT
  - Conversion from Blaise 4 to Blaise 5

  … and we will make & show samples

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Introduction (1)

- Blaise 5 Manipula Dialogs is a big change compared to Blaise 4
  - From defining dialogs in source code without any visual aid in Blaise 4 …
  - …to defining dialogs through a 'mixed approach' using source code and a designer in Blaise 5
- Manipula Dialog use the same architecture that is used for data entry:
  - pages with layout based on templates,
  - events in which you can invoke actions, expressions, …

Introduction (2)

- Defining a dialog is a two step process
  - First you define in the source code what fields and data sources are available in the dialog
  - And next you can work on the design and behaviour in the dialog designer
- Knowledge of Blaise 5 layout comes in handy when implementing a dialog in Manipula

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Dialog basics: the syntax in Manipula

A dialog requires a definition in the source

- In a setup that starts with the reserved word PROCESS you are allowed to define one or more **dialog sections**
- In the **dialog section** you are allowed to specify
  - a **fieldrefs** section
  - a **datasourcerefs** section
- In the **fieldrefs** section you specify what fields from your setup are needed in the dialog
- In the **datasourcerefs** section you define what data sources are available for use in the dialog
- We will focus first on the **fieldrefs** section

The field reference section (1)

```
PROCESS FirstDialog

AUXFIELDS
  Name   "Name of participant": STRING[40]
  Age    "Age of participant": 0..120
  Gender "Gender of participant": (Male, Female, Other)
  Buttons: (OK, Cancel)

DIALOG MyFirstDialog "My first dialog"
FIELDREFS
  ASK Name, Age, Gender, Buttons (OK)

MANIPULATE
  MyFirstDialog
```

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

The field reference section (2)

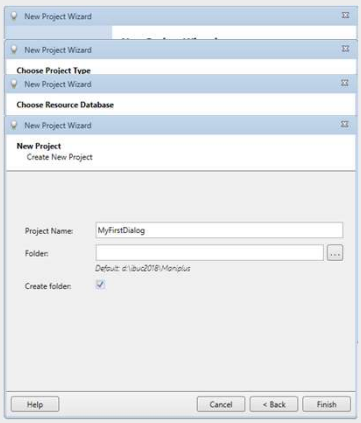
- The AUXFIELDS *Name*, *Age* and *Gender* can be given a value in the dialog
- The AUXFIELDS *Buttons* plays a special role in the dialog
  - Because of the value between ( ) it is the **result field** of the dialog
  - Only **one** enumeration field is allowed as **result field** of the dialog; more than one result value is allowed
    Example: *MyButtons (OK, Edit, Select)*
  - In the dialog there will be a push button for each enumeration value of the result field
  - When the dialog is closed by pressing the OK button, the auxfield *Buttons* receives the value *OK &* the value of the auxfields *Name*, *Age* and *Gender* will be available in the setup

A first example: step 1

How to implement this dialog?

Step 1. Create a new project

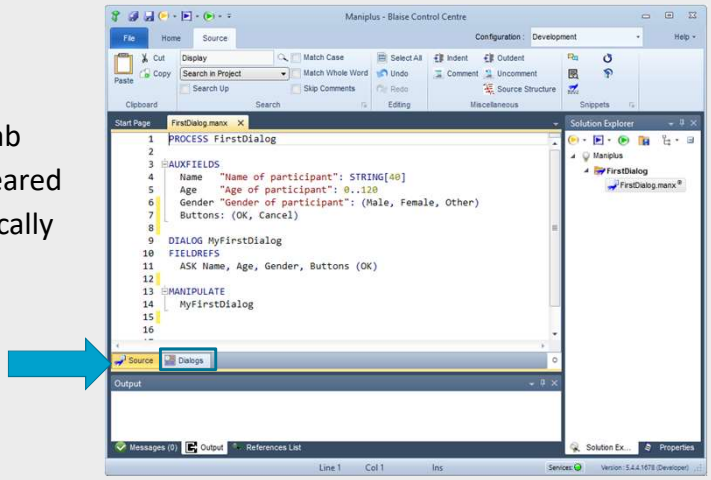
- Choose 'New Project from scratch'
- Choose 'Select 'Manipula Project'
- Select 'Use a Resource Database for Manipula dialogs', click Next
- Give your project a Name
- Click Finish



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

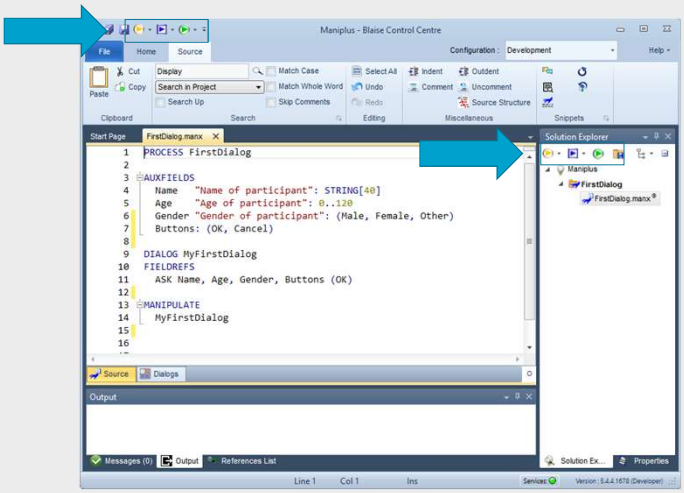
A first example: step 2

- Step 2: Edit the setup in the Editor in the Blaise Control Centre
- Note the Dialogs tab that appeared automatically



A first example: step 3

- Step 3: Prepare the setup and run it

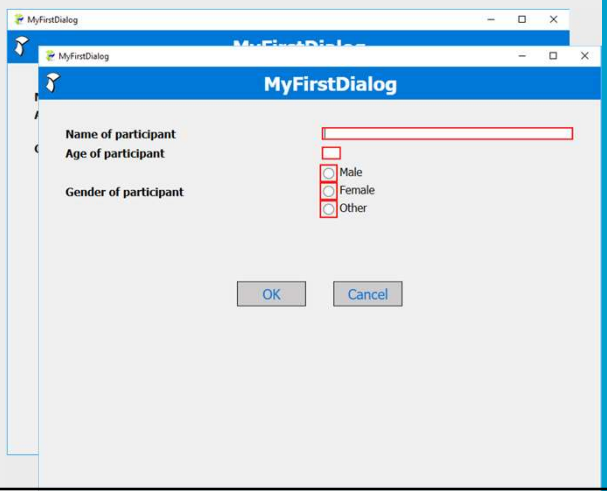


Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

A first example: the result

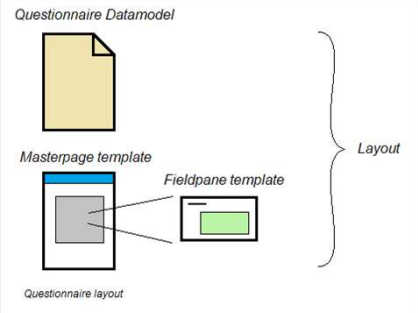
- A dialog with 5 input controls and 2 buttons
- 4 field panes
- Height: 600
- Width: 800
- Resizable

- And when you click the OK button you see…



Layout explained (1)

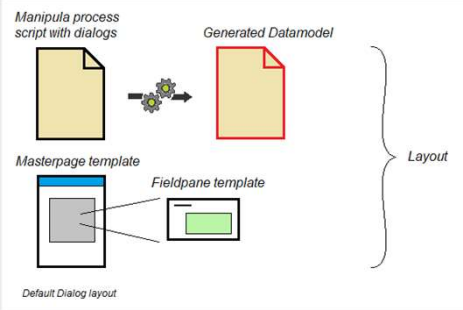
- The layout of a questionnaire is based on templates
- Each page of a questionnaire is based on a *MasterPage* template
- For each field on the page a *fieldpane* template is used



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Layout explained (2)

- Layout of dialogs is, just like the layout of a questionnaire, based on templates
- This requires a datamodel. The datamodel is generated based on the dialog definitions in the setup
- For each dialog a parallel block is generated



A first example: the result explained

How the dialog looks and behaves depends on

- The templates in the **resource database** that have been applied
  - Field panes that contain input controls
- The default value of some dialog properties
  - The *Dialog Width*, *Dialog Height* and *Allow Resize*
- The attributes of the used auxfields
  - The red boxes when clicking OK are caused by the auxfield definitions **not** having the EMPTY attribute

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Layout explained (3)

- The templates needed for producing dialogs are defined in the resource database, resource set 'Dialogs'
- We introduced some additional templates for Dialogs
  - For instance: the *InputWithActionButton* field pane template
    - This template provides a button behind the input line for which you can specify an action
    - It can for instance be used to activate a file selector
  - Other useful templates can be added by us or by yourself
- We defined parameters for the templates
  - Parameters enable you to change property values without changing the template definition in the resource database

Layout explained (4)

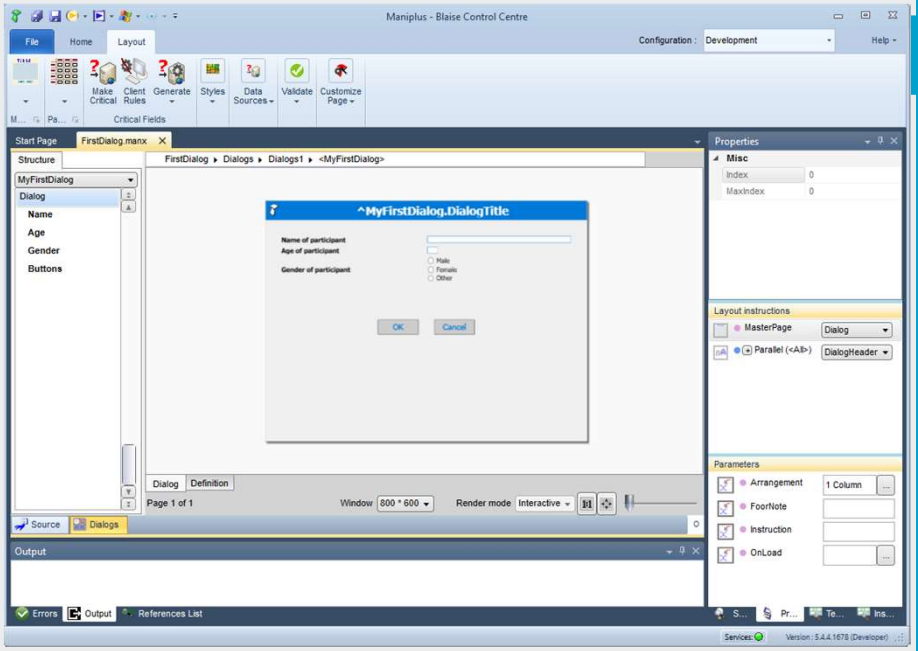
- We made choices with respect to the parameters
  - Not too many parameters but also not too few parameters
  - We need feedback on whether the parameters provided in the default resource database are enough to handle the common programming tasks for dialogs

- Note that during this presentation we are using version 5.4.5 and we are using an improved resource database (intended for 5.5)
  - We will make it available for download

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

A first example: some changes

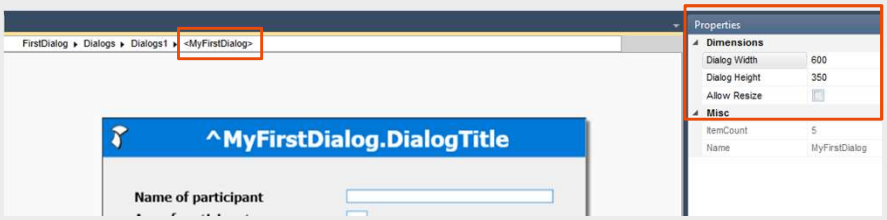
- Added a title in the source code
- Added EMPTY attribute
  - When clicking OK the dialog is now closed with NO red borders around the controls that are empty
- Adapted the *Dialog Width*, *Dialog Height* and *Allow Resize* property in the **Dialog designer**



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Change dialog properties

- Click on the name of dialog in the crumb bar
- Then change properties on the right



The dialog designer (1)

- For those who already worked with layout in Blaise 5:
  The **dialog designer** resembles the **layout designer** for a datamodel
- But there are fundamental differences:
  - A dialog corresponds with one page and you cannot force a dialog to become more pages
  - You can customize a dialog in ways not possible in the layout designer
    - Change the grid (add columns / rows)
    - Add new controls (including grids)
    - Move fieldpane instances
    - …
  - When the parameters are not sufficient, customization in the layout designer can only be done by customizing templates (using the Resource Editor)

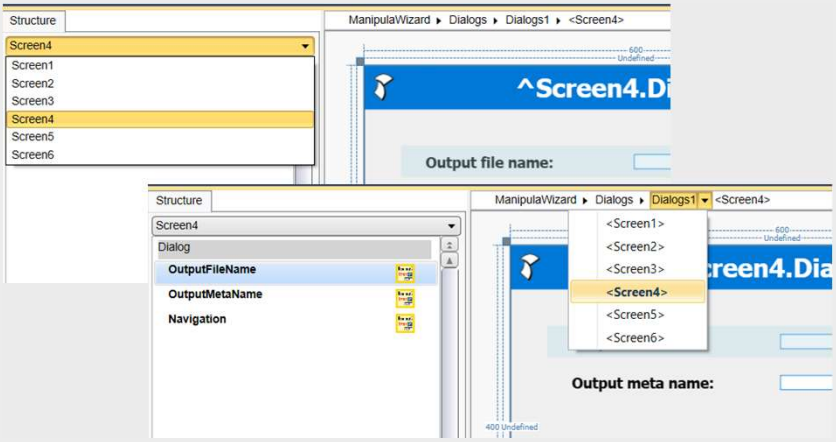
Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

The dialog designer (2)

- To open the dialog designer click the **Dialogs** tab
- The initial view that you see in the layout designer shows
  - the list of fields (on the left)
  - the current design of the layout (in the middle)
  - the Properties window for the current selected field (on the right)
- In the Properties window you can change
  - What template has to be used
  - One or more parameters of a template
- Sometimes that is enough…

When you have multiple dialogs

You can select a dialog in the drop down list in the structure window or in the navigation crumb bar

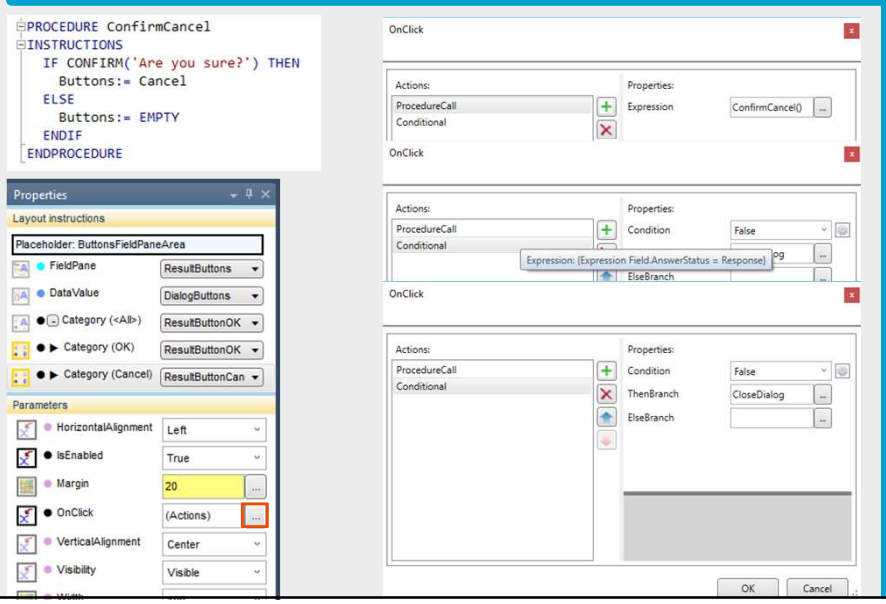


Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

A first example: more changes (1)

- Let's make a change to the dialog:
  - Show a message box when Cancel is pressed to confirm the Cancel
- This can be done by calling a procedure in the OnClick event of the Cancel button
- The recipe is as follows:
  - Add a procedure to the setup to confirm Cancel and store the result in the Buttons field
  - Specify the OnClick event: call the procedure, use the result in a ***conditional*** to close the dialog

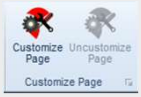
Confirm Cancel button click



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

To customize or not to customize?

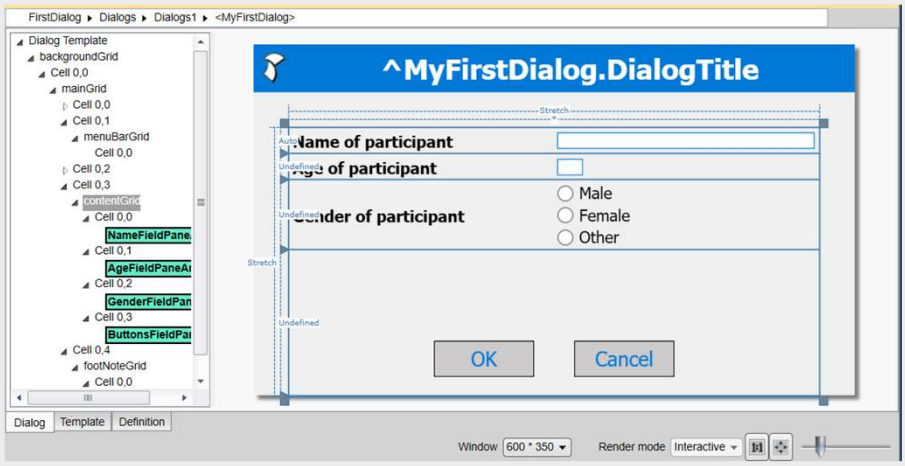
- When you want to customize your dialog beyond what is allowed by the initial view in the layout designer you click the **Customize Page** button



- When clicked you can make changes to the layout of the dialog and you have a lot of control over the behaviour of the dialog
- You now see an extra tree view left of the current design: **The Dialog Template tree view**

- You will probably always decide to click the **Customize Page** button…

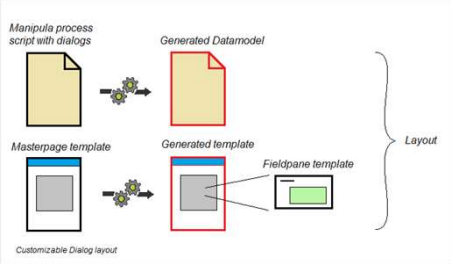
The dialog template tree view



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Layout explained (5)

- When the 'Customize Page' button is pressed, a Dialog Page template is generated based on the (active) MasterPage Template.
- This template contains ***FieldPaneArea*** placeholders for each Field on the dialog.
- In the expressions you now have access to the fields on the dialog



Layout explained (6)

- When the Dialog is a Customized Page, any subsequent changes made to the original **master page** template in the resource database are not reflected in the design of the customized page
- This does not apply to templates at a lower level such as *FieldPane* templates and button templates because these are replaced in the usual way at a later stage

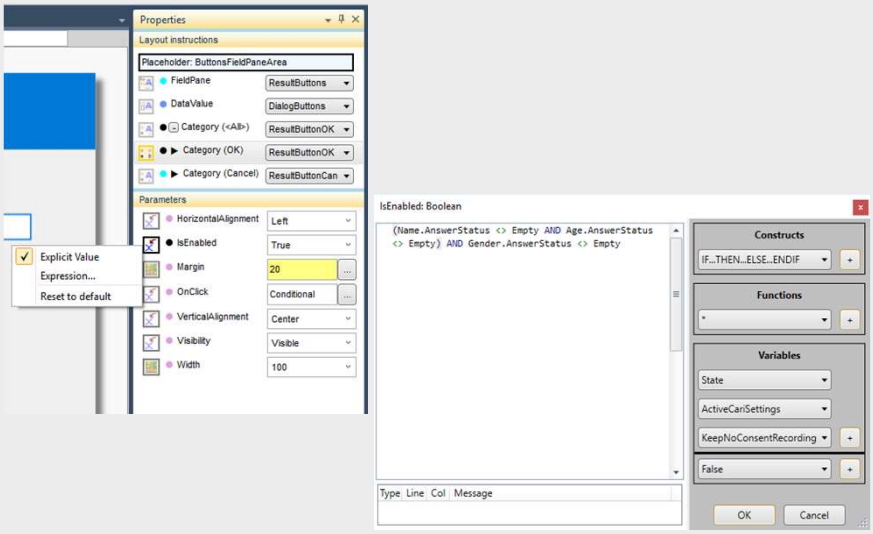
Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

A first example: more changes (2)

- Let's make another change to the dialog:
  - Enable the OK button when all fields have a value

- This can be done when the Dialog is a Customized Page
- The recipe is as follows:
  - Click the ***ButtonFieldpaneArea*** place holder
  - Select the OK button in the Properties windows
  - Click on the icon before ***IsEnabled*** and select Expression…
  - Enter the expression in the Expression Editor and click OK

Specify IsEnabled for OK button



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

More on the FIELDREFS section (1)

- You can specify how a field can be used
  - **ASK**. The data can be changed when the field can accept a value in the setup
  - **SHOW**. The field is for display only purposes
  - **KEEP**. Not for display but the value can be used in expressions
- Any elementary field known in the setup can be used
  - So you are not allowed to use a block instance / array instance
  - Use the file-identifier-dot-name notation to use a field from a file. Example: MyInputFile.Name

More on the FIELDREFS section (2)

- You can specify a text for a field

```
DIALOG MyFirstDialog
FIELDREFS
  ASK Name "Last name at birth:"
  ASK Age "The age on ^ReferenceDate:"
  ASK Gender "The gender:"
  ASK Buttons (OK)
  KEEP Cncl
```

- You can use a fill in the text
  - You do not need to mention the fill variable in the field reference section

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

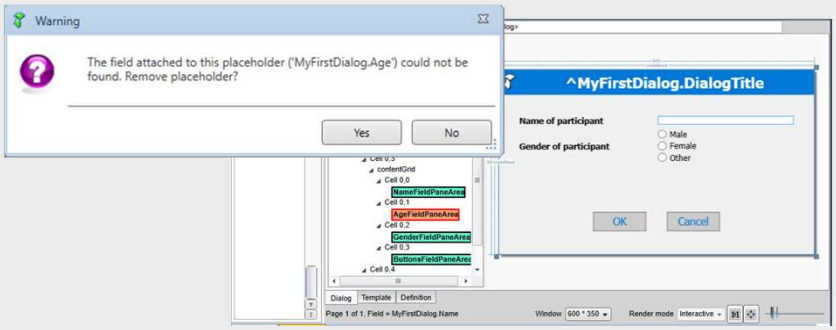
Adding / removing a field (1)

- When the dialog is not a Customized Page the Dialog is updated automatically
- When the dialog is a Customized page
  - New fields only show up in the list of fields (on the left) and you will need to drag the field to the Dialog template tree view
    - Demo

  - Deleted fields still show in the template tree view in red

Adding / removing a field (2)

- After removing field *Age* from the fieldrefs section in the source:
  - It still shows in the tree view
  - Needs to be deleted. When you click it you will be prompted to delete it



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Using a data source in a dialog

- Using a data source is similar to using a lookup in Blaise 4
- To use a data source in a dialog you need to specify a data source reference
- A data source reference is either a direct reference to a file identifier or a reference to a data source
- A data source is specified in a **datasource** section
- In a **datasource** section you specify:
  - The file identifier
  - The return field (optional)
  - What keys to allow (optional)
  - What fields to show (optional)

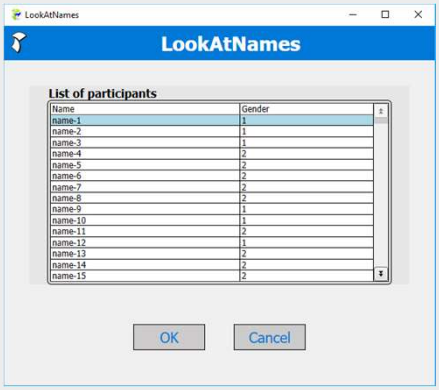
Datasource example (1)

```
PROCESS FirstLookup

USES Meta 'Participant'

INPUTFILE Data:Meta ('Participants',BLAISE)

AUXFIELDS
  Ret: (OK,Cancel)

DIALOG LookAtNames
DATASOURCEREFS DataV "List of participants"
FIELDREFS ASK Ret (OK)

DATASOURCE DataV:Data
  FIELDFILTER = Name, Gender

MANIPULATE
  LookAtNames
```

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Datasource example (2)

- The datamodel has a secondary key and that will be used by default to display the data
  - Specify SEARCHKEY=PRIMARY to search in the order of the primary key
- After clicking the OK button the selected record is available in the setup



The DataView template

- This template from the resource database is used to display the data source.
- It has several Data Controls:
  - Data Grid
  - Search TextBox, Search Button
  - Key Selector
- Parameters of the Template allow you to specify Properties and Events of these Controls. For instance:
  - Visibility of the SearchButton
  - OnRecordSelected event of the DataGrid
  - OnConfirmSelection event of the DataGrid to handle double click

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Behind the scenes (1)

- The name of the generated .blax for setup *setup.manx* is *setup.manx.blax*
- The name of the layout file is *setup.manx.blax.layout*
- When you run a setup in the IDE a package is made with name *setup.bpkg*
  - You can add files to the package using a Blaise Package Specification File (.bcps)
- When Manipula.exe executes a .bpkg file, the package is installed in the *<deploypath>/standalone/<setupname>* folder
  - Note that the **working folder** is the folder where the **package** is located (source / develop folder)

Behind the scenes (2)

- When a dialog is started: the values of the fields mentioned in the fieldrefs section are copied to the corresponding fields in the parallel
- A dialog has a Result Field that receives a value when the dialog is closed. The value corresponds with the button that was used to close the dialog
- When the Result Field has an 'OK'-Value: values are copied from the parallel to the corresponding fields in the setup
- Otherwise: fields mentioned in the fieldrefs section are given the value before calling the dialog (except the Result Field)

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

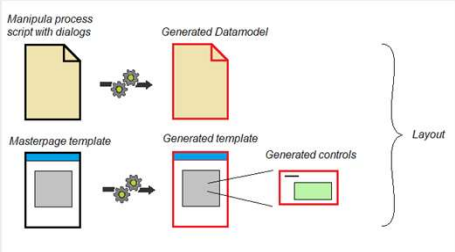
Dialog conversion Blaise 4 –> 5 (1)

- To convert a Blaise 4 Maniplus setup that contains dialogs to a Blaise 5 Manipula setup with dialogs you need:
  - The sources (.man & .inc files)
  - The prepared setup (.msu) including the meta files (.bmi) that are being used
- The source is adapted based on what is read from the .msu file

- The conversion tool generates the whole dialog page with all its controls without using field pane templates
  - No resource database used during conversion

Dialog conversion Blaise 4 –> 5 (2)

- Compared to a dialog designed in Blaise 5:
  - There are no placeholders in the generated template
  - Instead, the necessary controls like text controls and input controls, are directly inserted into a content grid.
- Because of this: the generated layout is not easy to maintain or customize



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Start a data entry session (1)

Blaise 5 allows you to start a data entry session from within a Manipula script. There a 3 ways of doing it:

- Res:= EDIT('<command-line>')
  - Command line must contain name of installed survey, GUID of installed survey or package name
- F.EDIT('<command-line>')
- F.EDITFORM ('<command-line>')

– F.EDIT and F.EDITFORM require:
- An already installed survey
  - Can be done using the INSTALLPACKAGE function Res:= INSTALLPACKAGE ('LaborForce2018.bpkg')
- A BDIX that points to the database of an installed survey

Start a data entry session (2)

– The required BDIX can be made using the wizard
– Note that the database for the survey resides in the folder *<deploypath>/standalone/<surveyname>*

– The F.EDIT command line needs to specify the value of the primary key:
- r:= data.EDIT('-KeyValue:'+STR(Data.ID))

– F.EDITFORM works on the current record in the setup so you do not need to specify the key value on the command line
- r:= data.EDITFORM('')
- After the Edit the current record is updated

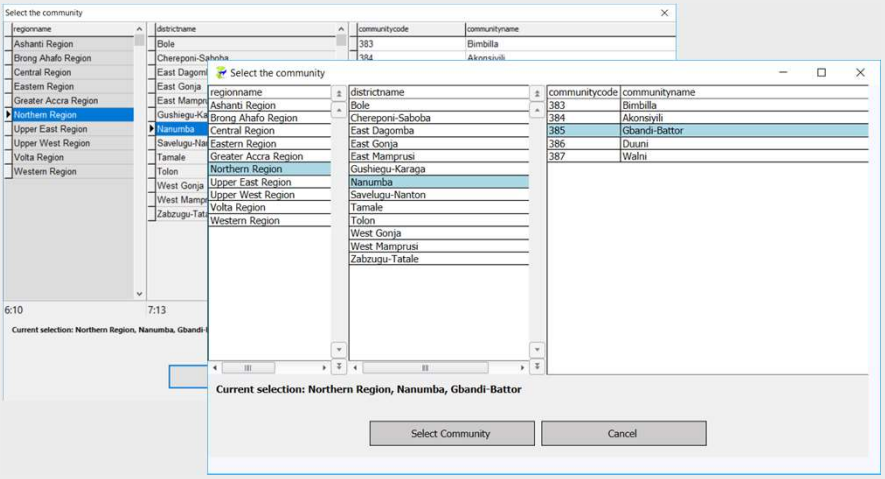
Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Example with dialog and edit

– Demo of a small sample that combines dialogs and edit

Conversion example (1)

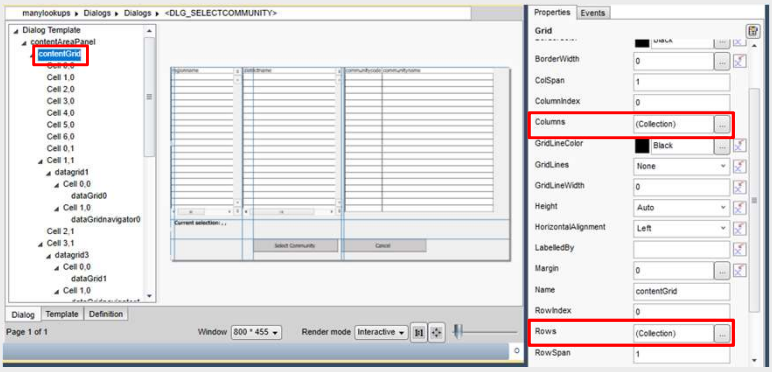
– Blaise 4 ‘3 lookups’ example



Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Conversion example (2)

- Note that the width and height are not completely accurate
- This can easily be corrected by increasing the width / height of the last column / row in the designer



Conversion explained

- When .msu not available or it can not be loaded you get a message in the log for each dialogbox in the source
- The conversion tries to handle most of what has been defined in Blaise 4. There are some exceptions, for instance
  - Very complex Boolean expressions for enabled/visible/…
  - More than one result field (is not allowed in Blaise 5)
- There is still room for improvement based on feedback from the users
  - So let us know when it does not work for you or when you have suggestions for improvement

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.

Samples and snippets

- Blaise 5 has some basic Manipula Dialog samples
- The Manipula Wizard is a more elaborated sample

- You can also use two snippets in the source tab ribbon
  - Dialog
  - Lookup example

Copyright © 2018 by Blaise, Statistics Netherlands. All rights reserved.