DATA ENTRY VERIFICATION

IBUC 2010 – Pre Conference workshop
INTRODUCTION

- Data Entry Verification
  - Blaise offers a two pass verification data entry quality control method, also called double data entry.
  - This method of quality control is very useful in catching and correcting random miskeyed strokes.
  - It can also be very useful in interactive data-editing where a user can record a reason for a data-value change.
INTRODUCTION

- A very recent addition to Blaise 4.8.2
  - Available as of build 1562 on (August 17, 2010)

History
- Some years ago there was some discussion with the BCLUB
- Never received high priority. Implemented but never finalised and documented
- Thanks to Mark Pierzchala it is now available
  - Mark will have a poster on this subject tomorrow

- Note that some have used Blaise for verification
  - Programmed through the rules and / or aliens. A lot of work
  - Now it is available as ‘generic’ feature
TWO VERIFICATION MODES

- BLIND verification
  - The verifier does not see what the existing data is
  - If the newly entered value does not match, the verifier will be prompted to select the correct value or to enter another value
  - This is a traditional double data entry verification capability

- OPEN verification
  - The verifier sees what the existing data is
  - If the verifier changes a value that was previously entered, the verifier will be prompted to select a reason for changing the value
HOW TO START?

- Use the Control Centre run parameters dialog
  - Go to DEP, Options tab
  - Set ‘Enable verification’ and then select Blind or Open

- Via the command line of the DEP
  - /V- for Blind verification
  - /V+ for Open verification
LOAD A FORM

- To verify data an existing form needs to be loaded in the DEP
  - It is advised to use a datamodel with a primary key

- During verification the DEP keeps track of 2 instances of the form in memory
  - The changed form = initially the original form
  - The displayed form
BLIND VERIFICATION (1)

- The form is handled as a new form in the DEP. The verifier types the answers.
- A newly entered answer is compared with the original answer. When they are not the same a dialog is displayed.
- The compare is ‘relaxed’ for SET fields (order is not important) and for STRING/OPEN fields (casing, extra spaces and line breaks are not important).
- By default fields that were previously off-route are not compared.
BLIND VERIFICATION (2)

- In the dialog the original answer and the newly entered answer are displayed.
- The verifier can now make a choice between the two available answers via a radio button, or escape the dialog.
- If the newly entered answer is accepted it replaces the original answer.
- The verifier can make a note. The note is optional.
- During an intermediate save it is the changed form that is saved and not the displayed form.
  - Needed because not all data is present in displayed form...
OPEN VERIFICATION (1)

- The form is handled as an existing form in the DEP. All data is visible. The verifier can change the answers.
- An entered answer is compared with the original answer. When they are not the same a dialog is displayed.
- Here also: the compare is ‘relaxed’ for SET, STRING and OPEN.
- By default fields that were previously off-route are not compared.
OPEN VERIFICATION (2)

- In the dialog the original answer and the newly entered answer are displayed.
- The verifier needs to indicate why the answer was changed by selecting one of the options presented in the reason group.
- If the newly entered answer replaces the original answer.
- The verifier can make a note. The note is optional.
**Some Remarks**

- When the DEP is in verification mode the caption of the DEP shows [Verification]
- During verification it is **not** possible to create a new form, to delete an existing form, to create an empty database or to change the primary key
- In Blind mode the verifier is always prompted to save the form also when all fields entered have the same value as the original fields
- All changes are logged in the verification log file (extension .verificationlog), including previously off-route fields
CONFIGURATION - VERIFICATION DATAMODEL

The DEP uses an extra datamodel for the verification

- The default name is verification.bla. A prepared version is shipped with Blaise in Bin folder
- The OPEN verification reason field is based on the TReason type
- You are allowed to define languages. The verification dialog will use the language used during the DEP session
- The length of the string can be adapted to fit your needs
- By default the verification dialog has an input control to make a note. If you do not want the possibility to make a note you need to remove the Note field
- The order of the fields in the model is also the order of the fields in verification log file. You are allowed to change the order
The datamodel can be given a different name and location through the verification ini file (.vif)

You can specify the vif on the command line (
/V{+|-}<vifname> ) or in the CC run parameters dialog

A default verification.vif is shipped with the system (Bin folder)

At this moment it contains 4 settings
SETTINGS IN THE VIF

- **Offroute.** By default no verification is needed for off-route fields. Offroute=1 activates verification for such fields also.

- **Beep.** By default there is a *beep* when the verification dialog pops up. Specify beep=0 to disable the *beep*.

- **PreSelect.** By default *no radio button* is selected in the dialog. Specify preselect=1 for the first one, preselect=2 for the second one etcetera.

- **Meta.** The name of the extra verification datamodel.