

In-house Support for Blaise Application Developers and Users

Janica Cajhen & Pavle Kozjek (Statistical Office of the Republic of Slovenia, SORS)

1. Introduction

The Blaise system is used at SORS since 1993. In this period the usage was extended from supporting single CADI and CATI surveys to a general data collection and editing tool, completely covering CATI and CAPI mode of data collection, as well as data entry and editing from paper forms. In some surveys, Blaise is also used in a process of editing of data from secondary sources or as a tool for different auxiliary processes. Manipula is successfully replacing old mainframe tools used in statistical processes.

On the other hand, some more recent Blaise methods and techniques are not yet implemented in production. In 2005, SORS began pilots on using BCP and Blaise IS. The pilots are under way and we expect to discuss and implement the results during this year.

To summarize, instead of trying to rapidly implement some new technical solutions we focused first on **setting a basic common standards on two main groups of survey applications**: CADI (including secondary sources) and CATI – CAPI. Common standards in this context mean that one survey can be added to a system, covered by generic user interface that supports all the steps in a data entry and editing process. The main ideas of this approach were presented in the SORS IBUC 2003 paper (Blaise for Post-collection Data Editing - Building general data editing system based on Blaise).

One of the key conditions to be fulfilled in order to implement the ideas of common standards was to give the necessary knowledge to developers and users of Blaise applications. Once the systems are implemented, the knowledge about using and maintaining them is inevitably. When building our own systems we can not expect the help and support from the outside.

The following article describes how the support to developers and users of Blaise applications at SORS is organized.

2. Development from the good practices

The idea on setting common processing standards on groups (or clusters) of surveys comes from the good experience with our in-house developed Gentry system for high speed data entry. In this case the target group of surveys was defined very clearly - all surveys reporting on paper forms, and using high speed data entry.

With CADI data editing it was already harder to find common standards for the group of surveys, since the activities and processes understood as data editing are already more complex and “colourful”. Common solution should cover various kinds of import into editing system, different modes of editing, various reference files and databases, different levels of users to work with the data, different ways of final data export etc. In a few words: generic GUI for data editing needs much more buttons.

The first step in preparing standard solutions for CADI surveys was to harmonize the needs of statisticians and data editing staff about the generic data editing application and belonging user interface. As soon as it was developed and implemented, the knowledge about adding new surveys into the system was spread to the larger group of application developers. At the same time, the group of application users was trained about using the new solution..

With CAPI – CATI surveys the things get more difficult and complex, since the processes are more different more from survey to survey. The system to cover and control the most important common processes is under development.

3. Help and support for developers of Blaise applications

Since Blaise was adopted at SORS as standard data editing tool, all developers were expected to be able to use it when preparing data editing applications. Most of developers from *Department for data entry and editing applications* had no previous experience with Blaise, and the **courses** were the first thing they needed. They were also not familiar with LAN environment. As the first step, the initial three-day course at a very basic level was organized. Content of the course was development of applications for the in-house standardized CADI mode of data editing. A short developer's manual was prepared, and the course was based on a practical work on examples.

With the help of the first course, the new data editing system was implemented in practice at the end of 2002. The second internal course (beginning of 2004) was mostly about different possibilities using Manipula and the third course (November 2005) was about standardized application development on LAN.

It was noticed that Blaise language was accepted very fast and successfully amongst former mainframe developers. Much bigger issue was migration to a new platform. For that reason a special **user interface for developers** was designed. Its main purpose was to enforce standard development and to bridge the gap between different environments. It supported:

- direct access to development environment, based on parameters (code of survey, developer etc.)
- copying all templates and standard setups (modelib, depmenu etc) to survey development folder, to begin the new application
- import and export data
- implementation - sending prepared application to production environment

The use of this user interface was not mandatory. It seems that with growing knowledge of developers this interface is becoming obsolete, and other modes of support are replacing it. But certainly it was a good start for beginners.

On-line manual for developers is at the moment probably main and most important support for developers. It is located on a server, accessible through the user interface, organized systematically (by contents) and regularly updated with the new contents. The manual gives a brief overview of all the steps necessary to prepare the survey application, it includes instructions, flowcharts, names and locations of standard datamodels, setups and templates etc. For the most important developing issues the translated terms and links to original Blaise Developers Guide are added.

Templates and standard solutions (data models, settings, Manipula setups) were designed to streamline the work of developers and to enforce in-house conventions

about developing applications. They are located in a read-only library on LAN server, with folders separated regarding to the contents.

The personal **help desk** for developers is not formally organised. But in practice it works well, and it is one of the fundamentals of the support. Help for less experienced developers is always obtained by the colleagues from department: personally, by phone or by e-mail. Using the same approach to development makes things much less complicated and time consuming.

There are also ideas about establishing Blaise developer's **forum** on the SORS intranet site. At the moment it seems to be not necessary, since most needs are covered by above mentioned ways of support to developers.

4. Help and support for users of Blaise applications

When talking about users at SORS, we need to distinguish between two different groups:

- “Specialized” data entry and editing staff that takes care only about entry and editing, but for a large number of surveys
- Statisticians that take care about complete survey process (data entry-editing is just a part of it)

In general, user access to Blaise applications is enabled through **generic user interface**. This user interface (prepared in VB) gives as much as possible intuitive access and control to all processes of data entry and editing.

At SORS, users are involved in the process of development of applications and interfaces by checking and testing the prototypes until they are ready for the “production”. Prototyping brings an important advantage: when the applications are implemented, some users already know all the functionality of the solutions, and they can immediately spread and disseminate the knowledge to their colleagues. Beside clear and intuitive interfaces, this is the second reason why the **courses** for the in-house Blaise users can be relatively short and to the point. Of course, the training for CAPI and CATI interviewers does not belong to this category and is not discussed in this paper.

Like for the group of developers, a **general manual** for users of Blaise applications was prepared. The first part explains user interface and all the steps supported in the process, and the second part gives a brief information on operational work - data entry and editing in Blaise. Explanation can be rather short, since most of the surveys in the system use standardized Blaise environment. When necessary (complex surveys, exceptions etc) the **content-based manual** can be prepared for the users. Both types of manuals are located on a LAN server (content-based in a survey working folder) and available through buttons on the generic user interface. With development of applications the manuals are updated and maintained.

When courses and manuals are not enough, the first level of personal **help desk** for users are usually their colleagues from the department, more experienced users of Blaise applications. If it doesn't work, a developer of application is contacted, and the problem is redirected to the development department.

Above mentioned methods of user support are used by both groups. There are also some specific needs of the second group – statisticians. They often ask developers for some tailor-made additions to the basic data entry-editing application, usually fast analytical Manipula setups, or special kind of data export. If it is not a part of

interface, a short instructions for the user should be added to the solution. But in general we try to include all the entry and editing processes into a small number of generic user interfaces.

5. Conclusions

By generalizing Blaise as main data end editing tool, the need for standardization of solutions at SORS became crucial. By classifying survey processes in a few main clusters with regard to their entry and editing characteristics we were able to set some standards in the development, as well as in the usage of applications. Consecutively, it was possible to organize some basic modes of support to developers and users in a relatively clear, effective and controlled way. We believe this approach will be working with the new, more advanced Blaise applications as well.

6. References

De Bolster, G. Generating Blaise with Blaise, Proceedings of the 9th International Blaise Users' Conference, Ottawa, 2004

Hansen, S. E. Programming Guidelines for Good Data Documentation, Proceedings of the 9th International Blaise Users' Conference, Ottawa, 2004

Kozjek, P., Blaise Generator for High Speed Data Entry Applications, Proceedings of the 6th International Blaise Users' Conference, Cork, 2000

Kozjek, P., Blaise for Post-collection Data Editing - Building general data editing system based on Blaise, Proceedings of the 8th International Blaise Users' Conference, Copenhagen, 2003

Pierzschala, M., and Manners, T., Revolutionary Paradigms of Blaise, Proceedings of the 7th Blaise User's Conference, Washington D.C., 2001

Sundgren, B., An Information Systems Architecture for National and International Statistical Organizations, CES/AC.71/1999/4

Ziesing, K. Generic System Interface Instrument, Proceedings of the 9th International Blaise Users' Conference, Ottawa, 2004