The 2007 Blaise NG White Paper and the Blaise 5 Result

Mark M Pierzchala / Mark M Pierzchala, MMP Survey Services, LLC (MMPSS)

1. The White Paper and the Working Group

The Blaise Corporate License Users Group (BCLUB) gave guidance for Blaise Next Generation (Blaise NG) through a document called the White Paper. The BCLUB understood that Blaise Next Generation would include many major changes. While Blaise 4 was (and still is) a very successful system, Blaise NG would have to adapt to a demanding and rapidly changing environment. The BCLUB wanted to define a system that would build on the strengths of Blaise 4, add features, and anticipate future needs.

The BCLUB formed the White Paper Working Group in early 2007. It included 13 people from 8 countries and 10 BCLUB members. These individuals had numerous years of Blaise and survey experience. They came from 9 National Statistics Offices and 3 private firms. Additionally, three other BCLUB institutes contributed strategic requirements. Mark Pierzchala from Mathematica Policy Research, Inc. was named chair.

The Working Group started work in May 2007 and completed in September 2007. The White Paper was discussed at the BCLUB meeting on September 25, 2007 during the 11th International Blaise Users Conference. The White Paper is 45 pages long. The group also produced a 65-page supplement that gives a record of technical discussions. A few years after the White Paper was published, The Blaise Team changed the name of the system from Blaise NG to Blaise 5.

The BCLUB formed a follow-up group in 2012 to revisit a few topics due to changing technological developments. The BCLUB continues to meet and discuss Blaise 5 developments at least once a year. The White Paper remains confidential to the BCLUB and Statistics Netherlands.

All opinions in this paper are of Mark M Pierzchala. He visited Statistics Netherlands in February 2015 and talked with the Blaise Team about this paper's main points. Blaise Team members also reviewed a draft of this paper for technical accuracy and to make sure the plans of the team were correctly stated.

2. The White Paper Contents and Guidance

After an Executive Summary and an Introduction, the White Paper continued with three interesting sections: (1) The Use of Blaise Today, (2) The Strengths of Blaise, and the (3) Scope of the Blaise System. These sections demonstrated the breadth of Blaise 4 and the magnitude of the conversion task. It showed that other systems would have a difficult time duplicating its capability.

2.1 The Use of Blaise Today

This section told of the many uses of Blaise. It listed survey areas such as social, labor, economic, agricultural, health, expenditure, and other topics. It listed kinds of surveys including household surveys with multi-level rostering of people, person surveys, institutional surveys, repeated surveys, longitudinal surveys, programs of surveys, studies that combined several surveys, and mixed mode surveys.

2.2 The Strengths of Blaise

This section described what fundamentally separates Blaise 4 from other systems. It mentioned a very strong programming language with many metadata elements. It explained how the checking mechanism (at once global and selective) enables great performance and flawless navigation. It demonstrated how the split-screen interviewer interface was innovative and effective. It showed how the system architecture enables huge questionnaires, and how its structure allows for clever solutions for difficult problems.

2.3 Scope of the Blaise System: Existing 4.x Tools

This was an elaborated listing of the 26 modules of Blaise 4. It showed that the system possessed modules far beyond an interviewing program. It gave a sense of the size of the development task. Some additional modules have been added since 2007 such as Computer Audio Recorded Interviewing (CARI).

2.4 Technical Direction

This section forms the heart of the paper. It covered 13 technical areas including:

- Platform
- Metadata
- Data
- Layout
- Case Management System
- Mixed Mode
- Development Environment
- Language Enhancements
- Integration and Interoperability
- Documentation
- Migration
- Performance
- Security

Each area gave a summary of guidance points and background explanation. The companion document provided extra details on most points.

2.5 Appendices

Appendix A of the White Paper listed Working Group members and told of their varied backgrounds and areas of technical expertise. Appendix B gave a high-level overview of the .NET environment.

2.6 Guidance, not Instructions

The White Paper gave guidance to Blaise 5 development, not strict instructions. It acknowledged that the Blaise Team would have wide discretion in Blaise 5 development and that for some items, development decisions would differ from the White Paper guidance.

3. Evaluating the White Paper almost 8 Years Later

The White Paper Working Group knew that predicting the future would be difficult. It knew that technical innovation was proceeding very quickly and that platforms would spring up overnight. Nevertheless, it

did its best to anticipate technological change, and overall did a decent job. There are however, a few things that the White Paper missed or underestimated.

3.1 Accessibility

The White Paper said nothing about accessibility and usability for disabled people. This was a major omission. Even in 2007, there was much discussion for the need to make self-administered surveys accessible for the blind, the visually impaired, and those with motor impairments. This omission was noted in 2010 by The Blaise Team and Blaise 5 is being developed with accessibility requirements in mind. The Blaise Team also back-fitted accessibility features into Blaise 4 in 2013.

3.2 The Proliferation of Mobile Devices

Smart phones and modern-day tablets were just emerging in 2007 so the Working Group knew about them. However, it did not estimate (who did?) how quickly these devices would become mainstream. The idea that a respondent would answer a web survey on a cell phone was not considered, or that an interviewer might use his or her own device to conduct an interview. The paper suggested that The Blaise Team investigate how Blaise NG might run natively on a mobile platform; now this capability has become an unavoidable requirement for iOS and Android devices running Blaise 5. Additionally, the White Paper said nothing about *time to release* of new versions.

4. How Blaise 5 Development compares to White Paper Guidance

It is possible to compare the still-developing Blaise 5 against the White Paper guidance. The purpose of this section is to state where those differences are, and the reasons for these differences. Sometimes, the emerging Blaise 5 system has not yet implemented an item but plans to do so in the future. Note that the Blaise Team continues to consult with the BCLUB on priorities.

4.1 Platform

Blaise 5 tracks this section closely. This new Blaise 5 version is the first to operate its runtime modules (e.g., interviewing) not only in Windows, but in non-Microsoft environments such as iOS and Android mobile devices. It can use Linux as a web-survey host for the first time. These multiple runtime environments have proved a challenge, not only because of different operating systems, but because of screen size, pixel density, and different keyboard layouts and available keys.

4.2 Metadata

In Blaise 4, a LAYOUT section in the source code language gave layout instructions that governed page breaks and the use of templates from the Mode Library. The Working Group explicitly stated that the LAYOUT section should stay in the source code. However, the LAYOUT section is no longer valid for Blaise 5. In its place is a Layout Designer in the Control Centre where the developer can manually insert references to templates and page breaks. This is a major departure from the Working Group direction. The Layout Designer is difficult to use for large surveys due to performance issues (which will be fixed) and the number of manual actions that have to be performed. To address this issue, the Blaise Team will design a Layout Language that will replace the LAYOUT section. (See 4.4 on Layout.)

The White Paper suggested that the Blaise Team investigate the possible creation of a Graphical User Interface (GUI) for interactive statement of metadata. The Blaise Team made some investigations but discontinued work for now. Since it was mentioned as a secondary priority by the White Paper, the Blaise Team should not spend any time on this at the moment.

In Blaise 5, it should be possible to associate descriptive text with an assignment statement. This is not yet possible.

4.3 Data

The Working Group wanted to get away from the proprietary Blaise 4 database in favor of an open source solution such as MySQL. The Blaise Team chose SQLite which satisfies the suggestion. However, the Blaise 5 storage within the SQLite database is still proprietary and not of a RDBMS format. A mapping to such a RDBMS format can be achieved in Blaise 5 with the .bdix file. This is much like the .boi file solution in Blaise 4. This choice was made by the Blaise Team in order to minimize data storage size (the typical RDBMS being a model of bloat) and to achieve performance goals (especially field-to-field interviewer performance).

An extended range of field statues in addition to EMPTY, DK, and RF was directed by the Working Group. This is not yet done but the Blaise Team will implement it soon.

BCLUB members wanted more flexibility with a more robust (or forgiving) database definition. For example, if a field range changes from 0..1 to 0..7 the database checksum should not change. That is, minor changes in field definition should not result in incompatible data files that require a data update procedure. The Blaise Team notes that there are unanticipated problems with this White Paper suggestion. These include the situation where you mix databases with different field definitions and it is not clear what should happen in these cases. This can still be a BCLUB discussion item.

Neither the White Paper nor the Companion Document explicitly mentions the phrase client-server. There are several mentions of ‘server’ in the context of web surveys, and several mentions of ‘client’. But you never see client-server as a requirement. (Some BCLUB meetings may have talked about the client-server paradigm.) In Blaise 5, client-server is a required paradigm. Practically speaking, this means that the database must be installed (like Blaise IS in Blaise 4), not just copied into a folder. This also means that the Blaise 5 database is stored in a particular folder location. There are a few awkward effects of this choice. For example, institutes often have external files stored in a sub-folder of the main folder, and this is not yet possible. Some more work needs to be done to give needed folder flexibility.

4.4 Layout

The White Paper section on layout says all the right things. New capabilities are needed above those provided by Blaise 4 because of new devices, screen sizes, screen orientations, and screen resolutions. It also mandated that the Blaise Team find a unifying layout concept that would allow the efficient and flexible generation of screens. In hindsight, this section vastly understated the complexity of achieving the needed layout flexibility. The Blaise Team has spent enormous amount of time achieving the goals of this section. It came up with the Resource Database to replace the Mode Library. It came up with the Layout Designer to (manually) replace the LAYOUT section.

The Working Group specifically requested in this section that the LAYOUT section in the source code remain, and that the generating of screens based on settings and key words should be maintained. The reason the Working Group wanted to maintain the LAYOUT section is because the generation of pages in Blaise 4 is extremely efficient. Some Blaise instruments have tens of thousands of pages. The Working Group did not want to have to manually adjust layout. This suggestion was not taken for Blaise 5. Furthermore, while the LAYOUT section was removed from the source code, the GROUP concept was introduced into the source code (a Working Group suggestion). Maintaining the LAYOUT section in the source code seems entirely consistent with placing GROUP in the source code.

The Blaise Team deems the LAYOUT Section too limited for modern-day challenges, and that a more powerful layout language is needed. Additionally, there can be more efficient use of the Layout Designer such as generating and displaying only one model page instead of many identical pages. At the time of this writing, layout possibilities in Blaise 5 are far more flexible than in Blaise 4. It is also very time consuming and difficult to deal with the Resource Database and the Layout Designer. The Blaise Team has committed to make layout much easier to achieve in Blaise 5. In the end it doesn't matter how this is accomplished as long as developers can efficiently create layouts across modes and devices.

The information-rich split-screen interface, and all of its ease-of-use operability, was also required by the White Paper. This is in progress.

4.5 Case Management System

At the time of this writing, development of case management in Blaise 5 has not yet started. The Blaise Team has been gathering requirements from BCLUB members for this section. The CATI call scheduler in Blaise 5 will be like the one in Blaise 4. Blaise 5 will include a multimode survey management scheme.

4.6 Mixed Mode

Three aspects of mixed mode surveys are mentioned in this section. First is the ability to generate different kinds of screens for different modes. This has been achieved. Second is the use of settings to achieve some multimode capability. This has been achieved.

Third, however, is the ability to handle field attributes and some rules differently between modes, in a more elegant way than in Blaise 4. For example, in an interview, if a respondent does not know the answer to a question, the interviewer should be required to enter a DK through a function key. On the other hand, in a self-interviewing situation such as a web survey, a more common practice is to allow the respondent to pass over the question without answering. The non-answer then is interpreted as a Don't Know or a Refusal answer in the downstream analysis. These differences in mode conventions can be done in Blaise 4 through a lot of programming. A much more elegant solution would be to allow field-level differences between modes. Blaise 5 should be told how to implement DK and RF differently between modes at the instrument level. The Blaise Team is currently implementing this capability.

4.7 Development Environment

Blaise 5 has a fully capable development environment now. It has a few large features that Blaise 4 does not have which are the Layout Designer and real-time parsing. The latter can be turned off if working on an extremely large instrument.

The White Paper talked about a GUI interface to enter metadata, but this was not a priority. This GUI has not been realized. At this point, it does not seem like a GUI module for core metadata would be worth the effort, and any such an effort would distract from other higher priority items.

4.8 Language Enhancements

More than twenty language enhancements were listed in the White Paper. One of these, the request for indefinite arrays, has proved difficult due to page layout issues, but it is still on the to-do list. A few others, having to do with table display and operability (e.g., a vertical table), are really layout features. In fact, it is now possible to do a vertical table in Blaise 5 by defining a template in the Resource Database.

The White Paper mentioned the introduction of GROUP. This would allow an easier way to combine 2 or more fields that visually or otherwise should be combined. A classic example is the idea of a quantity / unit construct, for examples distance travelled (number followed by either miles or kilometers). Groups have been added to Blaise 5 and form the basis for all multiple-item displays including tables.

A few of the proposed language enhancements affect the core of the system and thus would have to done within the Blaise 5 language. These include extended field attributes for multimode surveys and user-defined field attributes. The Blaise Team is currently working on these core improvements. The implementation of these features will considerably ease the development of multimode instruments.

Blaise 5 has a new construct called ROLES. This was explicitly mentioned in the White Paper and is a promising new language feature that helps satisfy some auxiliary requirements. For example, in Blaise 4 it is common to use LANGUAGES to state metadata items such as SAS variables, or to construct item-level help. ROLES now fulfill this need and LANGUAGES are reserved for spoken languages.

The LANGUAGES section in Blaise 5 is much more powerful than in Blaise 4 with the explicit inclusion of all known written languages. Blaise 4 did not handle Asian languages well and struggled with Right-to-Left languages. A demonstration instrument about commuting habits is on the Blaise website. It includes 10 languages including Arabic, Chinese, Dutch, English, French, Greek, Hebrew, Hindi, Japanese, and Spanish. A language control allows you to switch between the languages on the fly.

Some of the White Paper language suggestions are really best implemented as model blocks of code. These may be provided through the use of the SNIPPET capability. This is a new Blaise 5 feature that allows specification of some constructs. You could imagine a SET-most important-second most-important-construct to be modeled as a snippet.

Many of the suggested language improvements would reduce the amount of programming we now have in Blaise 4 to implement some features. These include randomness and rotation, SET-most important-second most important constructs, and others. The desire by the Working Group was to have explicit features in the system to simplify these more demanding requirements. Some of these are still under consideration by the Blaise Team. This especially includes randomization; others are not a high priority at the moment.

A few of the language suggestions such as some string manipulation functions, should be relatively easy for the Blaise Team to implement, but it is a matter of how high a priority this would be.

4.9 Integration and Interoperability

The White Paper mentioned three items: (1) an Application's Programmers Interface (API), (2) a services approach to components, and (3) alien routers. All three of these suggestions have been implemented, including alien routers in early 2015.

4.10 Documentation

This is about an instrument self-documenting feature. In Blaise 4 there is a system called Delta that can generate views on the metadata including a flow chart. A parallel system created by the University of Michigan Survey Research Center called Michigan Questionnaire Documentation System (MQDS) also provides various views on the instrument. Both Delta and MQDS generate their electronic views from the compiled metadata file of the Blaise 4 instrument. Blaise 5 will not have a Delta capability. It does have a metadata API that would allow a third party such as the University of Michigan to recreate their MQDS system for Blaise 5.

Blaise 4 also has the Cameleon metadata language that allows various descriptions of FIELDS, such as a data dictionary or SAS description. Blaise 4 also allows metadata access by Manipula. SAS and Stata scripts are programmed in Blaise 4 Manipula for example. There are no plans to duplicate Cameleon in Blaise 5. Manipula in Blaise 5 will continue the metadata access. This more than replaces Cameleon. The Blaise 5 metadata API will also be available.

4.11 Migration

The White Paper spends quite a bit of space discussing the migration of Blaise 4 instruments and surveys to the Blaise 5 environment. Given the need for a complete restructuring of the Blaise system, no one believed that you would be able to press a button and generate all Blaise 5 changes.

Blaise 5 has a utility that can convert most Blaise 4 source code to Blaise 5 source code. While this capability will greatly ease the conversion of Blaise 4 to Blaise 5, it does not complete the task. There are too many differences between Blaise 4 and Blaise 5 to completely escape manual modifications to the source code and configuration files. This is mainly due to the layout and platform demands of the modern-day computing environment.

Among the Blaise 4 features that will disappear or change in Blaise 5 are the use of environment variables, different kinds of alien routers, tables (become groups) and how layout is accomplished. An upcoming Institutes Guide to Blaise 5 will thoroughly discuss these topics and explain how to efficiently move from Blaise 4 to Blaise 5.

4.12 Performance

The Working Group specifically mentioned performance as a reminder that one of the great attractions of Blaise 4 was its item-to-item fast speed, especially with large interviewing instruments. Performance for Manipula on data files, large numbers of users accessing a web site, and so on, was also mentioned. As Blaise 5 emerges into production, performance will be easier to measure. The Blaise Team is aware of some performance issues and will solve them through optimization analyses and procedures. BCLUB members can assist with their own performance testing.

4.13 Security

The Working Paper had only two requests for this topic. One was the ability to encrypt data on a field-level basis (e.g., for social security numbers) and to be able to work with third-party systems to be able to encrypt data on a database level.

The native Blaise 5 database storage encrypts every field automatically, except that the key fields are exposed. Also, by using the SQLite database as a default, third party solutions should also work.

5. Summary

The development of Blaise 5 has turned out to be very challenging. The Blaise Team has spent a large amount of time on layout concepts and platform in order to meet the challenges of representing a Blaise questionnaire on any device. This has caused them to deviate at times from The White Paper suggestions or to implement the suggestions in a different way.

Blaise 5 development is still underway. Some White Paper suggestions that are not yet implemented will come soon. Following is a discussion of where Blaise 5 should more closely meet the White Paper.

5.1 Layout

The Blaise Team has done a very good job of defining layout concepts that work across devices and modes. They allow the institute to define their own templates in the Resource Database and to implement them in an instrument through the Layout Designer. They have built a solid foundation.

At the current time however, layout is hard to achieve in Blaise 5. This has been noted by several early adopters. The use of the Resource Database requires a large learning curve, as does the Layout Designer. Even for an accomplished user, the manual insertion of layout elements requires too much time.

The addition of a Blaise 5 Layout Language to replace the Blaise 4 LAYOUT section will be a high priority. The Blaise Team should improve the performance of the Layout Designer and a better default Resource Database model should be developed. This will allow institutes to modify an existing solution rather than define one from scratch. Additionally, some advanced techniques to generate layout should be documented (these will be published by the end of 2015 with the Institutes' Guide).

Related to layout are operability aspects of the Blaise 5 runtime. Blaise 5 should be at least as easy to use as Blaise 4 and the item-to-item performance should be the same for interviewers.

5.2 Language Enhancements

Blaise 5 retains the powerful Blaise 4 language with key improvements. Even with some changes (e.g., with GROUPS) the Blaise 4 programmer will have little problem with the Blaise 5 language.

The most important language enhancements yet to be implemented relate to multimode features. This is especially true of mode-specific attributes that would allow instruments by default to more easily handle the different practices between self- and interviewer-administered questionnaires (see 4.6 above).

5.3 Migration

Blaise 5 provides some migration tools, but the differences between Blaise 4 and Blaise 5 require substantial manual adjustments. These fall into the categories of (1) instrument migration, (2) deployment, (3) integration, and (4) performance. These are very practical concerns and impact the daily fielding and operations of an ongoing survey.

Blaise 5 has a very good on-line help. There are 6 tutorials that help with layout and deployment. More documentation is planned and underway. There are now a few good instrument examples, and many smaller datamodels that demonstrate a feature. An Institutes' Guide is planned for end of 2015 that will explain how to address many of the day-to-day procedures that an institute must execute.

6. References

Pierzchala, Mark M., et al. "Blaise Next Generation; A White Paper by the BCLUB Technical Working Group on Blaise NG." Presented to the BCLUB at the 11th International Blaise Users Conference (Survey Research Center, University of Michigan), Annapolis, MD, September 2007. (Available to BCLUB)

Pierzchala, Mark M., et al. "Blaise Next Generation; Companion Document, A Record of Discussion of the BCLUB Technical Working Group." Presented to the BCLUB at the 11th International Blaise Users Conference (Survey Research Center, University of Michigan), Annapolis, MD, September 25, 2007. (Available to BCLUB)

7. Appendix – List of the Members of the Working Group

Mark Pierzchala, US MMP Survey Services, LLC, in 2007 with US Mathematica Policy Research, Inc.

Karen Bagwell, US Census Bureau

Gina-Qian Cheung, US University of Michigan

John Kinder, UK Office for National Statistics

Roger Linssen, the Blaise Team, Statistics Netherlands

Rebecca Lui, Statistics Canada / Statistique Canada

Grayson Mitchell, Statistics New Zealand

Sylvain Moreau, Institute National des Statistiques et des Etudes Economiques (France)

Zsolt Papp, Statistics Hungary

Juanita Pettit, Australian Bureau of Statistics

Mike Rhoads, US Westat

Roger Schou, US National Agricultural Statistics Service

Jeroen van Velzen, User, Statistics Netherlands