The Michigan Questionnaire Documentation System (MQDS) Version 5 Update

Karl Dinkelmann, Peter Sparks, Jason Ostergren, & Gina-Qian Cheung, University of Michigan

1. Introduction

The Michigan Questionnaire Documentation System (MQDS) is an application that extracts data and metadata from Blaise instruments and generates various types of output. These outputs include codebooks, questionnaire documentation, and human readable metadata documentation. Additionally, MQDS can serialize the extracted information as a Data Documentation Initiatives (DDI) XML instance. DDI is an internationally recognized XML standard for documenting and disseminating social survey data and metadata. The introduction of the next generation of Blaise, version 5 and the recently released DDI version 3.2 standards offer a unique opportunity to rewrite MQDS to adapt to these changing technologies. MQDS version 5 is being rewritten to work with Blaise 5 data models and datasets using the Blaise 5 API (Application Programming Interface). MQDS 5 will also incorporate work done by Ostergren & Ash, 2013 allowing one to visualize the Blaise logic and dynamically manipulate it within a visual GUI interface.

We are taking a modular approach to constructing the components necessary for the next version. With MQDS 5, one goal will be to clean and unify the code. However, the core features of extracting Blaise metadata, generating statistics and frequencies from data, representing human readable output, and generating a DDI instance remain intact.

2. Background

For over a decade, Survey Research Operations (SRO), a unit within the University of Michigan's Institute for Social Research's Survey Research Center, has developed and enhanced MQDS to allow users to export Blaise metadata and data to document and disseminate Blaise questionnaires to their users. The development of the first version began in 2003 (Sparks and Liu 2004, Guyer and Cheung 2007). Sparks and Liu (2004) describe early development of MQDS version 1 and what was referred to as "BlaiseDoc" in the early stages of developments. Guyer and Cheung (2007) further explained development activities that lead to MQDS versions 2 and 2.5, rewriting the program in .NET, and described utilities. The creation of version 3 began in 2009 and was outlined in Dinkelmann et al (2009). Version 3 was database driven and written to use the DDI 3.0 standard. In 2011, MQDS 4 was enhanced to more efficiently process larger data models and the DDI 3.1 standard. All versions of MQDS 4 and earlier used either Blaise version 4.6, 4.7, or 4.8.

2.1 MQDS 3 & 4

The introduction of MQDS 3 added the feature of writing the Blaise metadata and data to a database. In theory, this would have been the ideal design. However, it was tightly coupled to the DDI 3.0 specification. Therefore, as new specifications were released, an in depth understanding of the changes was required in addition to the knowledge required to make the DDI 3.0 database schema work with the updated 3.1 and 3.2 DDI specifications. This led to limitations in the database design used in MQDS 3 and 4. In creating MQDS 5, we will no longer continue active development of MQDS 4. However, support will remain for MQDS 4 as needed moving forward. Depending on the adoption rate of Blaise 5, it is possible that MQDS 5 could support Blaise 4 in the future.

2.2 DDI 3.2 Standard

The Data Documentation Initiatives (DDI) XML version 3.2 adds features that will assist in creating DDI compliant metadata with MQDS. One of the primary features added to DDI 3.2 is the introduction of the *FragmentInstance* that allows one to generate DDI instances that are valid and only contain the actual

elements that are needed. In previous versions of DDI, required elements that were not in the Blaise metadata either needed to be captured or empty XML tags were written to ensure the XML instance validated to the DDI Schemas. Other features introduced in DDI 3.2 include, but are not limited to: modeling of the data flow throughout an instrument with input and output parameters as well as bindings, question grids, and scale questions. DDI Version 4 is already in the development phase although a release date has not been set.

2.3 Blaise 5

The next generation of the Blaise software created by Statistics Netherlands is Blaise 5. It brings forth many major changes in what survey developers are able to accomplish. It also has a completely new implementation of the Application Programming Interface (API). This is important, as all tools that currently exist around Blaise 4.8 will need to be updated to take into account the new structure of the API. Therefore, in order to access the metadata within Blaise 5 MQDS updates are needed using the new API.

3. MQDS 5

The guiding principles in creating MQDS 5 are as follows:

- Continued support in Blaise 5 for derived metadata information like fill resolution and gotos
- Standalone executables allowing for faster testing and easier reuse
- Internal use of XML tailored to Blaise vs. attempting to do processing within a DDI framework; DDI will be an output product instead
- Ability to support multiple DDI versions

As Blaise 4 matured, many new methods were exposed to allow one to generate Blaise metadata. In fact, Blaise Manipula became powerful enough to export metadata. de Bolster (2013) demonstrated how to export DDI instances using Maniplus. However, with Blaise 5 the data manipulation, re-coding, exporting and importing features of Manipula remain under construction. This also means that the tools within Blaise 5 for extracting data and metadata are minimal to date. One can export the meta information from the control center to an xml file within Blaise 5. However, to use the exported information one would need to create an XSL transformation stylesheet to create useful documentation or to any other XML specification representation (such as DDI). We have not explored using this file but it would be interesting to see how one could potentially use this file. Below is an example of exporting the "Flight Survey" from the *Meta View* (figure 1) and the resulting XML file that is created (figure 2).

Figure 1: Generating the Meta View XML file from Blaise 5

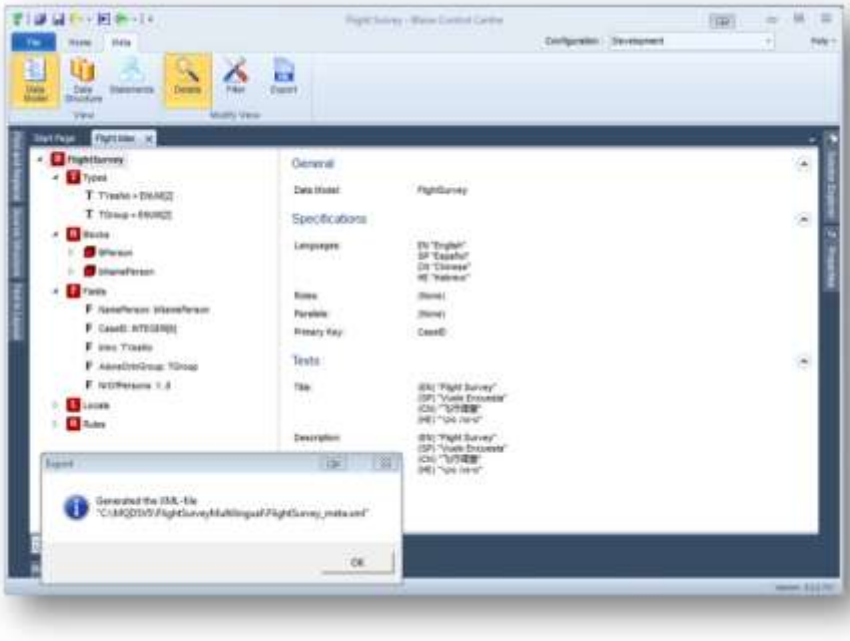
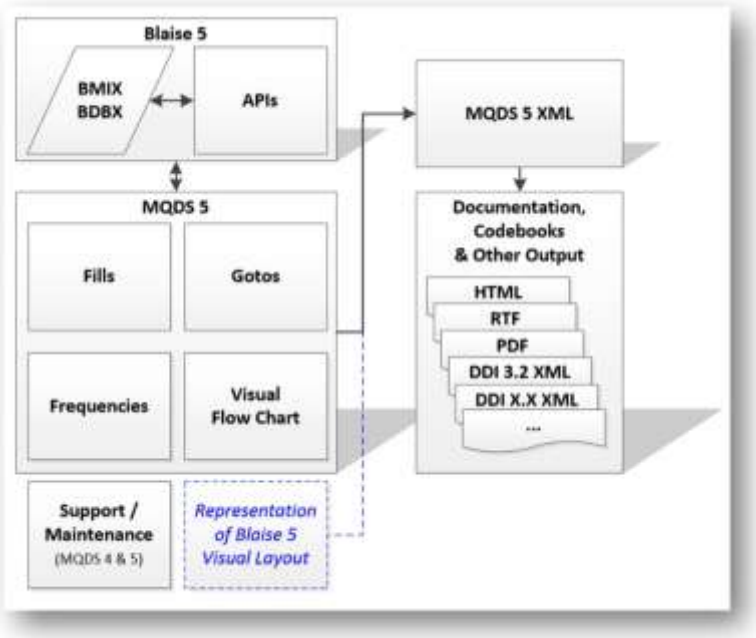

Configuration
Development
Data Model
Data Structure
Statements
Details
Filter
Export
View
Modify View
Start Page
Flight Man
FlightSurvey
Types
Blocks
Fields
Locale
Rules
General
Data Model
FlightSurvey
Specifications
Languages
Roles
Parallels
Primary Key
CaseID
Texts
Title
Description
Export
OK

Figure 2: Example of the "Fight Survey" Meta XML File

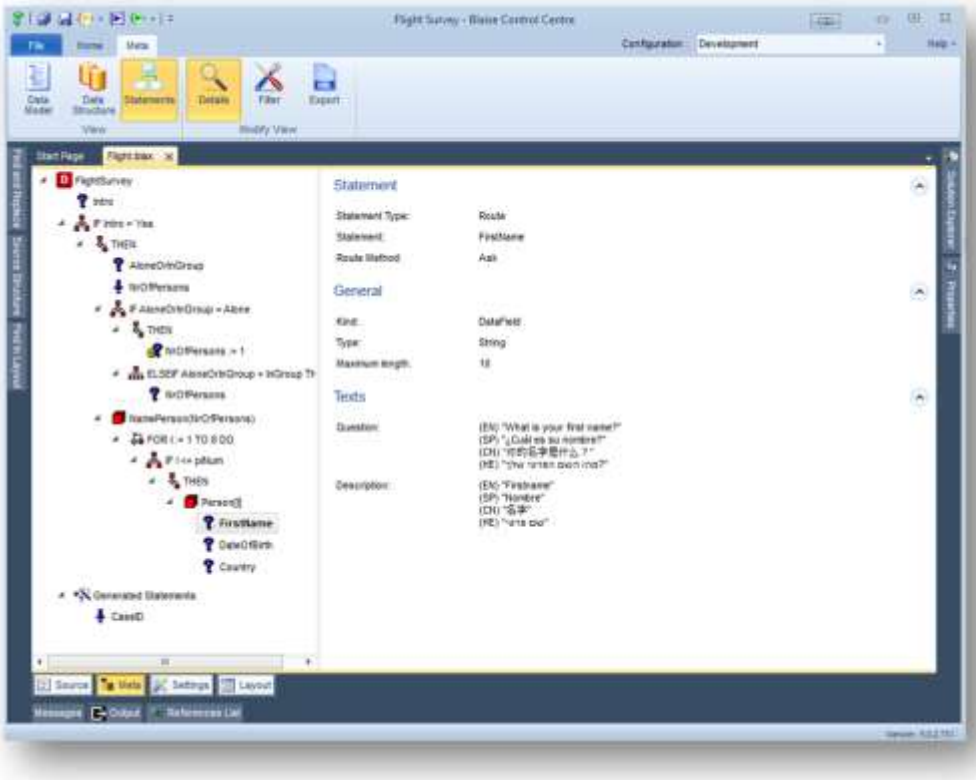
Using the DDI 3.X XML specification as the basis for the XML metadata documentation is not desired for similar reasons. The MQDS 5 XML is structured for fast retrieval of meta information. For these reasons, MQDS 5 will use its own XML schema. This also aids in the extraction of metadata and data in a format that assist in completing some of the main features of MQDS. Two of these features are its ability to resolve question fills and generate goto statement representation. These two features will comprise the core features included within MQDS 5. An initial hurdle in creating MQDS 5 was that we had few Blaise 5 instruments to use for testing. While as the number of production Blaise 5 instruments available for testing is still limited, a few large Blaise instruments exist that SRC is currently in the process of converting to use with Blaise 5. Additional components and features currently part of MQDS 5 can be seen below in Figure 3.

Figure 3: MQDS Version 5



Representing the visual layout of a Blaise 5 instrument is a feature that would be nice to support in the future. However, this feature will not be incorporated for some time due to reasons discussed in section 4. One of the newest and most exciting features is the inclusion of the Visual Flow Chart program developed by Ostergren & Ash, 2013. First demonstrated at the Washington, D.C. Blaise Users Conference, this feature has been updated to use Blaise 5. Blaise 5 itself allows users to drill into an instrument within the *Meta View* inside the control center. However, the *Meta View* offers a static treeview-style visual representation of the data model. Figure 4 shows how this looks in the *Meta View* within Blaise 5 when selecting the *Statement* view.

Figure 4: *Meta Statement View* in Blaise 5

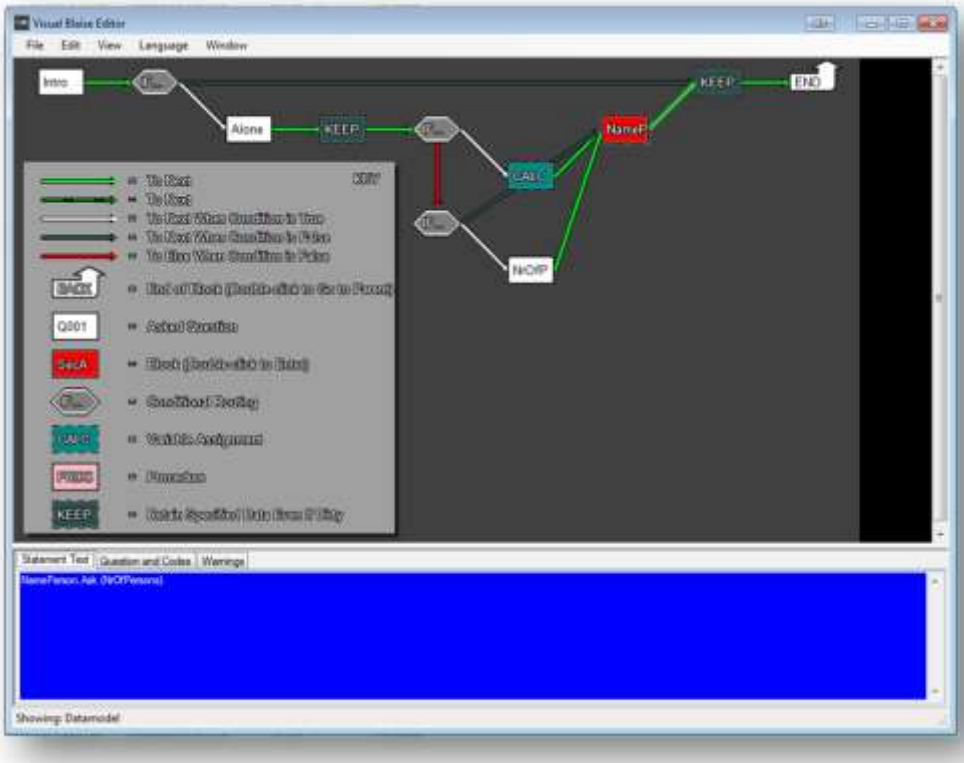


Ostergren and Ash (2013) describe their Visual Flow Chart program (known internally as "Visual Blaise") as :

> "… an application that graphs the logic of a Blaise instrument one block at a time. It allows elements to be dragged and dropped, cut and pasted, and allows for the addition of new elements. Finally, it is capable of exporting a text file with the logic for each block written as a RULES section in correct Blaise syntax."

A graphical representation of the same "Flight Survey" code shown in figure 4 is displayed in figure 5.

Figure 5: Visual Flow Cart in MQDS 5



The main purpose of this flow chart is to provide an alternative visual representation which is easier to understand in many cases. One other difference between *Meta View* inside Blaise 5 and the Visual Flow Chart program is that the user can drill in and out of blocks and dynamically change the block structure, questions placement, and instrument logic. These changes can be exported to a text file which can be used to update the rules in the code.

4. Discussion of Potential Issues & Future Exploration

Perhaps one of the core differences between Blaise 5 and Blaise 4 with regards to the metadata is that it divides the survey instruments' core metadata from the layout. Blaise 5 does this by placing the layout information in a separate .layout file that is used to map between the metadata and the Resource Database. This adds to the complexity of documenting the look and feel of a given survey instrument. Blaise 5 features have far-reaching capabilities allowing survey developers to modify virtually all the aspects of the presentation of surveys for the end user. With all previous versions of MQDS the default presentation of the output was to mirror the Blaise 4.8 split screen layout presentation. However, the introduction of Blaise features allowing one to create their own visual presentation from the ground up will present some obstacles for one to programmatically document the actual presentation as seen by the end user. Layers of complexity are added with the addition of text roles, grouping, parameters, applicability conditions, and many other layout concerns that can all be impacted differently depending on the target device, platform, and mode. Therefore, accurately and adequately documenting Blaise 5's end users visual presentation will not be simple. To do this programmatically, further exploration and expansion of the API may be needed.

To what extent do these features need to be documented and/or disseminated? This is not something that is necessarily needed by all end users. However, this may be something that those who research effects of the visual presentation of instrumentation will want and require that they be documented.

It is possible that MQDS 5 would support Blaise 4 data models, although this is not currently on the development schedule. One of the main advantages of MQDS using its own XML standard is that it allows one to create any DDI standard. This makes it possible to only add additional modules instead of needing to rewrite the entire program when wanting to export to multiple versions of the DDI standard, regardless if it is a newly supported version or an older version.

5. Summary

MQDS 5 will continue to support the extraction of Blaise 5 data and metadata and will support new methods of viewing and modifying the Blaise 5 logic through the inclusion of the visual layout editor. As Blaise 5 evolves additional features will be identified and considered. The initial phases of development have begun on MQDS 5 and a pre-beta version release is expected by April 2015, with the first production release expected in summer 2015. Questions regarding MQDS 5 can be submitted to SRC.MQDS-5@umich.edu.

6. References

Blaise 4.8 Online Assistant (Version 4.8.4.1899). (2015).

Blaise 5 Help (Version 5.0.2.731). (2015).

DDI Lifecycle 3.2 http://www.ddialliance.org/Specification/DDI-Lifecycle/3.2/, Retrieved March 10, 2015.

de Bolster, G. (2013). Generating Blaise from DDI. *The 15th International Blaise Users Conference (IBUC)*, Washington, DC.

Dinkelmann, K. (2006). DDI Version 3 and Instrument Documentation. *The proceedings of the 32nd International Association for Social Science Information Services & Technology (IASSIST).* Ann Arbor, Michigan.

Dinkelmann, K., Kirgis, N., and Cheung, G. (2009). Michigan Questionnaire Documentation System, Version 3 (MQDS-V3). *The proceedings of the 12th International Blaise Users Conference (IBUC)*. Riga, Latvia.

Guyer, H. & Cheung, G. (2007). Michigan Questionnaire Documentation System (MQDS): A User's Perspective. *The proceedings of the 11th International Blaise Users Conference (IBUC)*. Annapolis, Maryland.

Ostergren, J. & Ash, R. (2013). A New Tool for Visualizing Blaise Logic. *The 15th International Blaise Users Conference (IBUC*), Washington, DC.

Sparks, P. & Liu, Y. (2004). Blaise Documentation System. *The proceedings of the 9th International Blaise Users Conference (IBUC)*. Gatineau, Québec.

Thomas, W., Gregory, A., Gager, J., Johnson, J., and Wackerow, A. (2015) Data Documentation Initiative (DDI): Technical Specification Part 1: Technical Document Version 3.2. © DDI Alliance 2015.