BLAISE AND THE AGRICULTURAL CENSUS
Batch processing on a CDC mini computer

Hans de Jong
Netherlands Central Bureau of Statistics, Voorburg

1. Abstract

When a CBS data processing system, like the one for the Agricultural Census, has to be built or rebuilt, the use of software packages like Blaise is always considered and encouraged. In this case the substantial size of the data files presents some severe difficulties if all error checking has to be done on personal computers. Nevertheless it turned out that a batch version of Blaise, running on a minicomputer network, could be applied successfully in the data validation process, starting with the 1992 census. The same Blaise questionnaire was used both for the integral check on the CDC minicomputer network and for the interactive data editing on PC's in a LAN. This paper gives an overview of this Blaise application and discusses some related problems and solutions.

2. The Agricultural Census

This yearly Census, organised by the Ministry of Agriculture, Nature Management and Fishery and the Netherlands Central Bureau of Statistics examines about 125,000 agricultural holdings. The questionnaire is made up of approximately 300 questions about labour force, areas of crop, and size of livestock. The census results in the publication of a number of basic figures, which are used for crop estimation, and several more detailed statistics, giving insight in the size and structure of the agricultural holdings. The edited data files are used for other surveys and further analysis.

There is a considerable demand for early provisional figures, within a few weeks after the questioning period. To be able to respond to this demand it is necessary to enter and edit the huge bulk of data in a very short time.

3. Never trust external data

From 1992 on the data entry process is no longer executed by the CBS itself, but accomplished by the Ministry of Agriculture, using Oracle. Both institutions agreed upon the specification of the questionnaire and the error checking to be used in the data entry process. The agricultural holders are invited to fill in the questionnaires at home and present them at local meetings, where they are entered into the Oracle system. In this way it is possible to detect and correct errors immediately. Further investigation by the Ministry can lead to changing some of the data. The edited data is offered to the CBS.

Even though the data is checked by the Ministry, the CBS wants to check it again, on the grounds that the Ministry cannot guarantee the data to be valid, because:

- the data is kept and sent to the CBS from provincial computer sites;
- it is not improbable that the validating software used by the Ministry is incomplete or even bugged;
- most of the checks by the Ministry are just warnings, and they might be suppressed at large by some of the interviewers;

To get some idea of the quality of the data the CBS also wants to do some additonal checks; if the quality of the data is high, the checking by the CBS will not yield a lot of errors, and so not much edit work is generated. If the quality is less than expected, the errors can be corrected or, which is more likely, the Ministry can be asked to adjust its error checking programs or procedures.

4. Synchronising two data editing processes

The data entry process by the Ministry takes place mainly from April until the middle of June, but data can be entered and modified till the end of August. If the CBS waited for the Ministry data to be complete before starting the extra data editing, the provisional figures would come out much too late, as they are required in July. Starting earlier, however,

raises some problems too: the data of a particular holding could be modified by both the CBS and the Ministry at the same time. Because the CBS is only able to bring about some statistical corrections, it is very interested in receiving the corrected data from the Ministry. One could ask the Ministry to pass on just modifications instead of all available data, but the Ministerial data systems are not capable of doing so, and earlier experiences with external registrations have proved that this approach does not work. A better solution is to ask the twelve provincial sites to forward to the CBS all available data three or four times during the questioning period. By comparing the successive data files the CBS can determine if the data of a particular holding have changed. New and modified holdings can now be selected and joined with the already existing and partially corrected data files. All existing data of a holding will be substituted by the modified data, thereby loosing some of the CBS corrections, and necessitating the checking and correcting of the same errors again.

The Ministry told us it was not possible for the data of a holding to be deleted. This, however, often does happen. A temporary solution for this problem has been found.

5. To Blaise Or Not To Blaise

After some prototyping it turned out that, due to the size and complexity of the questionnaire and the validation requirements, importing and integral checking of the data in Blaise could not be done on a PC at a rate above 25 forms per minute. This would lead to a typical computing time of 6 to 7 hours for the data of one province, which is valued as too long (what time would be acceptable?). Another problem would be the necessary disk space of 15 to 30 Megabytes per province. A solution was requested in which the integral check could be run on the minicomputer network, and only the faulty data would have to be edited on a PC.

A possible way out of this problem, but not a favourable one, would have been to build a program for the integral error checking using a third generation language like Cobol or Pascal, and using Blaise for the data

editing. This would mean to write two identical error checking programs in two languages. Twice as much work, and in later years twice as much maintenance effort. Furthermore, it would be likely that these programs would differ in small details, giving rise to unexpected problems.

A solution where all of the processing, including the interactive data editing process, would take place in the minicomputer environment was out of the question because it would seize too much of the available resources. The same was true for using Oracle in client/server mode, or trying new technology like a LAN batch server.

As it was known that the Blaise team was developing a batch version of Blaise for running the integral checking process on the CDC minicomputer network, this option, together with data editing in Blaise PC, was seen as the best alternative. Use of this product was agreed upon, provided that it would be operational at the appointed time and perform well enough, compared to the PC version. Passing the deadline would imply that a Cobol program had to be written after all. Care was taken that the other data processing systems, to be built for the Agricultural Census, were able to co-operate with both alternatives.

The Blaise team succeeded in producing a usable and well performing version just in time, and was able to eliminate a few bugs in short time, so fortunately there was no need to do the extra work of building a Cobol program.

6. Developing a Blaise Batch application

To be able to cope with the many possible file formats on a minicomputer, a Blaise Batch program is embedded in a Cobol program, generated by Manipula. The Manipula setup can be adapted by the systems developer. Inside Blaise, just a plain file format is used. For the parsing and compiling of the questionnaire some special actions have to be undertaken. Constructing the batch program is done in the following way: first a regular Blaise specification for a PC CADI-machine is developed and tested. After clearing all the small errors, the developer

calls the Blaise parser with a special parameter, telling it to generate a number of C source files and header files. These files are transported to the minicomputer network, where they are compiled and stored in an object library. Manipula is used to generate a Cobol program with which it is possible to call the C subprogram. All this is completed with a tailor-made JCL file. Cobol is used because it has - on CDC computers - an extremely good file input and output performance.

The Blaise team has the intention to include the batch option in the Blaise menu system and to have these actions be executed automatically.

7. The performance of Blaise Batch

The performance of Blaise Batch, compared to Blaise PC, was an important factor in the decision to use the package. The performance was expected to be better, but how much better it would be was not known. It can be fascinating to compare the performance of a package like Blaise on different hardware configurations. Yet we have to be careful drawing universal conclusions, because there are many interfering circumstances, some of which are CBS-specific. The brand and type of PC, operating this PC without a local harddisk in a LAN instead of using it as a stand-alone, the performance of the LAN and the fileserver, and the number of other LAN users will have an influence on the performance of Blaise on PC, while the sharing of a minicomputer with other users and other batch jobs will have a major effect on the performance of Blaise Batch.

Moreover, the two programs do not carry out exactly the same job. In this application the PC version converts a main file and subfile from ASCII to Blaise and performs the integral check. The mini version has to convert a main file with a variable length to a fixed-length format, do the checking, and write the dirty and suspect forms to a main file and subfile in ASCII.

After a number of production runs it turned out that the Olivetti M300 can process about 24 forms per minute, while the minicomputer network reaches an average rate of 103 forms per minute, ranging from a

minimum of 36 to a maximum of 343 forms per minute. Of course, processing a different questionnaire will give different, and in many cases better, results on both configurations.

8. Restrictions of Blaise

A frequency table of the errors and warnings was one of the requirements for the validation process. As Blaise does not produce this kind of control information it was necessary to create a special error block in the Blaise questionnaire and do some extra computing.

Unfortunately Blaise Batch does not produce the file format needed by Blaise PC, so instead of just transporting the Blaise files, one now also has to convert from Blaise Batch to ASCII and from ASCII to Blaise PC, and to repeat the integral checking. This can be a nuisance if the number of faulty records is relatively large.

9. Conclusions

It is possible to apply Blaise for validating large data files. Using Blaise Batch on a minicomputer network for the integral check and selection of dirty and suspect forms, and Blaise PC for the interactive data editing is favourable, compared to using the PC version only, because:

- the run time of the integral checking process is shorter;
- the integral checking can be done at any time, especially at night;
- several checking batch jobs can be started at the same time;
- the PC can be used for other activities;
- disk space on the PC or LAN can be preserved;

As the same Blaise dictionnaire is used in Blaise PC and Blaise Batch, development and maintenance time can be saved. At this time a tailor-made system still has to be constructed around the Blaise Batch application.

Of course there are a few drawbacks:

- in this application it is not easy to edit a clean form, as this never reaches Blaise PC;
- managing an application on two different hardware configurations, and controlling the communications between these configurations can form an extra burden for the users.
- the designer has to be acquainted with the internals of Blaise to be able to write the main program in Manipula.

Nevertheless it should be obvious that Blaise Batch will be used in many new or renewed statistical data processing systems at the Netherlands Central Bureau of Statistics.