WP. 10
ENGLISH ONLY

UNITED NATIONS
ECONOMIC COMMISSION FOR EUROPE

CONFERENCE OF EUROPEAN STATISTICIANS

Work Session on Statistical Data Editing
(Oslo, Norway, 24-26 September 2012)

Topic (ii): Global solutions to editing

On the general flow of editing

Prepared by J. Pannekoek, Statistics Netherlands and L.-C. Zhang, Statistics Norway

Abstract

The GSBPM (2009) provides the common reference for statistical production processes, but it is not directly operational in the sense that it does not prescribe how the processes are organized into a *work flow* so as to achieve a given purpose. General descriptions of the editing (or E&I) work flow is given in EDIMBUS (2007) and de Waal et al. (2011) Figure 1.1. In this paper we update and expand on these previous works, taking into consideration some recent developments in the theory and practice of editing. The work flow consists of a breakdown of the overall editing process into sub-processes. Each sub-process involves up to 3 generic types of editing activities with its own purpose and ordering, and is composed of one or several statistical functions. We shall describe the delineation of the sub-processes and some different ways by which the activities may be organized within the sub-processes in terms of the statistical functions, so that the resulting overall process meets quality criteria such as efficiency, accuracy and timeliness. The description of the general editing flow will be illustrated and commented using the editing practice of the Structural Business Statistics (SBS) in Netherlands and Norway.

I. Introduction

1. In the GSBPM statistical data editing (or E&I) is part of the process 5 *Process*. but it is also mentioned that the editing process can be subdivided into a number of sub-processes, and that such sub-processes can be part of other main processes with e.g. references to "edit", "impute", "scoring", "validate", *etc.* in process 4 *Collect*, 5 *Process* and 6 *Analyse*. For example, Process 5.3 *Review, validate and edit* and 5.4 *Impute* clearly deal with editing of micro data. But micro-data editing near-the-source can happen already in Process 4, and "macro editing" and "output controls" are mentioned in Process 6.2 *Validate output*. The GSBPM recognizes that the processes and sub-processes need not be carried out in the given order and the actual sequence of execution need not be linear and iterations between sub-processes can occur. For the design and implementation of an E&I process it is important to describe how the different activities involved are organized in a workflow and one also needs more detailed descriptions of the activities than the GSBPM is intended to provide.

2. In terms of purpose, the different *activities* involved in editing fall in three broad categories: (a) *verifying* consistency of the data with formally specified edit rules (expressed in terms of quantitative, or numerical, measures) and the plausibility of the data in a less formal sense, for instance the absence of outlying and otherwise suspect values and estimates; (b) *selecting* the values and/or units that may need to be adjusted or imputed, and (c) actually changing or assigning data values, that is *amending* the data in a way that is considered appropriate. The different types of activities are linked and ordered as follows: *verifying* leads to quality indicators or measures (including scoring) that point out specific problems in the data, *selecting* takes the quality indicators and the data as input and results in a selection of records or fields within records for further treatment. This treatment will often consist of *amending* the data values in order to resolve the problems detected earlier, and the results may be subject to another (or the next) verifying activity.

3. The generic editing activities data verification, selection and amendment are carried out through actions that have a functional nature because they can be expressed formally as a mapping from one set of elements to another, although the formalization may not always be made explicit either because it is cumbersome or unnecessary. The activities can therefore be decomposed into the so-called *statistical functions* (Renssen and Camstra, 2011), and an E&I process can be considered as a sequence of process steps where in each step one or several properly configured statistical functions are executed. In the design of an E&I process we must choose these functions, configure them and decide on the sequence in which they are carried out. Referring to the GSIM (2012), we are dealing here with the information objects and sub-objects in the group Production. In the choice and configuration of the statistical functions we are e.g. concerned with the information object Methodology and sub-object Method in the set Process Component. As an example, imputation is at the level of information object Methodology, whereas the nearest neighbour imputation is at the level of information sub-object Method. To put the statistical functions into a sequence we must also involve information objects in the set Process Control and set Rule, such as the objects Input, Output and Control construct.

4. In this paper we present a general flow of editing (Figure 1), which consists of a breakdown of the E&I process into sub-processes (represented as squares and diamonds). Each sub-process may involve a different mix of editing activities, and is composed of one or several statistical functions. In doing so we will be concerned with how an E&I process is built from the constituent statistical functions and the order in which these functions are executed, which carries us to a level of details that goes beyond what the GSBPM is meant to provide. In Section II, we will briefly review the statistical functions on the level of Methodology, and present an overview for each of the 3 generic activities. Attention will be given to the distinction between functions that are generic in the sense that they can be applied in different E&I processes and functions that are subject-specific and therefore need to be re-specified for each particular E&I process. This helps to facilitate a plug-and-play approach to the process design and enhances the reuse of methodological components and software tools. The general flow of editing is described in Section III. We shall discuss and motivate the delineation of the sub-processes (Section III.A) and some different ways by which the activities may be organized within the sub-processes (Section III.B). We consider the delineation to be fixed, in which sense the flow diagram is generic. But the activities inside each sub-process may be organized differently, giving rise to the various operation scenarios. In practice, the choice of a particular flow is likely to be influenced by historical, technological, theoretical or even philosophical considerations. But it is important to *empirically* test and improve the flow, so that the E&I process can better meet the various quality criteria such as efficiency, accuracy and timeliness.

II. A summary of E&I functions

5. In this Section we summarize a number of statistical functions of which an E&I process is composed, respectively, for the three generic activities of verification, selection and amendment. Some of these functions apply to the input data on a record-by-record basis (i.e. *unit-mode*), while the others apply to all or a subset of them together (i.e. *batch-mode*). Many functions will have business rules as part of the input. The most important of such rules are the *edit rules* (or *edits* for short), where the *hard* edits specify the logical conditions that a consistent record must satisfy, and the *soft* edits specify the conditions that are statistically plausible. For instance, that an observed value should be non-negative may be a hard edit, but that its difference to its expectation should be within a certain range is usually a soft edit based on statistical assumptions. Or, that two observed values should add to a certain total may be a hard edit, but that the two should stand in certain proportion to each other may be a soft edit. Another common type of business rules are the simple if-then rules for selection of fields to change, correction of systematic errors and imputation of missing values.

A. Verification functions

6. Apart from being unit- or batch-mode, a verification function is either numerical or graphical.

7. ***Verification of edit-rules (unit-mode, numerical)*** Prior knowledge on the values of the variables and combinations of them can be formulated as a set of edits, which specify or constrain the admissible values. For numeric business data, many of these edits take the form of linear equality or inequalities. Some simple examples of such edit rules are, *turnover* ≥ 0, *profit* + *total costs* – *turnover* = 0 and *total costs* = *employee costs* + *costs of purchases* + *other costs*. The edits may be *connected* to each other by certain common variables, which is true for many of the edits used in business statistics and has consequences for error localisation and adjustment for consistency (see below).

8. The edit-rules can be verified for each record individually, i.e. in unit-mode. On the record-level this results in a binary *failure* status for each edit. The failure statuses can be the input to an error localization function that selects among the variables involved in the failed edits for amendment or further verification. It also provides information on the quality of the unedited (raw) data and the effectiveness of the edit rules. If we have N records and E edits, all the failure statuses can be summarized in an $N \times E$ *failed-edit-matrix,* corresponding to all the record-by-edit combinations.

9. ***Verification by score functions (Unit- or batch-mode, numerical)*** Selection of records for further treatment can be based on a *score-function* approach (Hidiroglou and Berthelot, 1984, Lawrence and McKenzie, 2000). Each record is assigned a score to measure the potential effect that editing may have on the estimated totals or other parameters of interest. The approach is also termed "significance editing", expressing the purpose of limiting the editing to those units where it can be expected to have a significant effect on the results.

10. The score of a record is usually a combination of *local* scores for each of a number of important variables. Each local score measures the significance for the variable of concern. Often it can be decomposed into a "risk" component that measures the size or likelihood of a potential error, and an "influence" component that measures the contribution or impact of that value on the estimated target parameter. A record- or unit-level score is a function of the local scores, e.g. a weighted sum of the local scores or the maximum among them. Before the local scores are combined to form a record score, they should be standardized so that they become comparable in size.

11. A common measure of risk for a single observed value is its deviation from its "anticipation". The latter can be a historical value for the same unit, provided such a value exists, in which case the score is unit-mode. Or it can be the current mean or median for a group to which the unit belongs, or a prediction based on a model that is fitted to the current data, in which case the score is batch-mode. A third possibility is to treat the values after some or all amendment as the "anticipated" values, so that the risk measures the change actually made to each observed value, in which case the score is unit- or batch-mode depending on the amendment involved. To standardize a local score one can divide it by a measure of dispersion such as the standard deviation or the inter-quartile range. If the dispersion is measured for the historic data and fixed for the current processing, the score can still be unit-mode. But if it is measured based on the current data themselves, then the score must be batch-mode.

12. It is usually not possible to calculate a *regular* score, as described above, for all the records due to missing values. A unit for which a regular score cannot be calculated may be classified according to the observed *irregularity pattern.* A *generalized* score can be defined as a semi-continuous variable, where the irregular categories are identified by negative integer values, and the non-negative range is reserved for the regular score value. Moreover, that a unit is considered highly influential or *critical,* can be incorporated as a separate category and assigned a corresponding (negative) value. In this way, formally speaking, it is possible to use a single score to identify: (1) the highly influential or *critical* units that requires clerical review in any case, (2) the irregular but *uncritical* unit that can be treated as unit-missing, and (3) the rest units that may be subject to further selection schemes. The approach is being implemented in the Norwegian SBS. We notice that, although it is not necessary to formally implement such a generalized score function in practice, the idea is important for the understanding of the general editing flow to be discussed later.

13. ***Macro verification (batch-mode, numerical or graphical)*** Macro-verification functions are batch-mode by definition. For instance, the aforementioned HB-score based on current deviance

measures, or the usual regression diagnostic measures such as the studentized residual and the DIFFITS, all of which are numerical functions defined for a set of units. Graphical macro-verification functions are also common in practice. For instance, a scatter plot of the current values against the auxiliary or historic values may provide the basis for detection of outliers and anomalies.

14. It is perhaps worth noting that macro-verification functions may arise from treating aggregates as "micro-data" on the corresponding level of aggregation. For instance, the total within each NACE-group may be considered a variable associated with the "units", which are the NACE-groups at that level of aggregation, and a risk measure can e.g. be given by the difference between the current- and previous-year totals. Then, even if the corresponding score for the NACE-groups is formally constructed as a unit-mode score function described above, it is batch-mode in execution.

B. Selection functions

15. Apart from unit- or batch-mode, we distinguish between selection of units and variables.

16. ***Selection of units using scores*** Scores can be used for the selection of units to be interactively edited in two ways: (i) by comparing them to a (predetermined) threshold value and (ii) by inspection of the distribution of scores, for instance by ordering of possibly several scores, usually partial in nature, and select the highest ranking unit or units (and terminate by some stopping rule). Type (i) selection is unit-mode, and the threshold value is obtained in a simulation study based on historic data (Latouche and Berthelot, 1992). In contrast, type (ii) selection is batch-mode even if the score itself is unit-mode. A stopping rule can be based on the absolute changes in the *current* parameter estimates that are induced by editing. Since these changes should be diminishing as units with lower scores are edited, the editing can be stopped when the changes are below an acceptable level.

17. It is important to make clear that batch-mode selection of units based on unit-mode scores can commence as soon as the data collection starts. At any given moment, one could perform type (ii) selection to get the next 'batch' of units to work on, which is optimal under a score-based approach. Having finished the batch, one could re-select among all the available units, including the ones that have arrived in the meantime. The scores being unit-mode, they do not change because of the new arrivals, so that there will be no 'reverse' of the previous selection.

18. ***Selection of fields for amendment (error localisation, unit-mode***) When a record does not satisfy the edit rules, some fields need to be changed in order to make the record *edits-consistent*. The selection of these fields is referred to as the *error localization* problem. The most commonly applied generic methods are based on the approach proposed by Fellegi and Holt (1976). The FH-approach is based on the principle that a record should be made to satisfy all edits by changing the fewest possible number of fields. The FH-selections can be found by solving an optimization problem where the sum of the fields to be changed is minimized, such that new values for these fields exist to yield an edits-consistent record. A very useful generalisation of this principle is to assign reliability weights to each variable and to minimize the sum-of-weights of the variables to be changed. Knowledge about the preference of selection among the variables may be incorporated in this way. Very large weights effectively exclude the corresponding variables from being selected. Moreover, using differential weights can help to avoid multiple equivalent solutions that often occur in the un-weighted approach.

19, Simple explicit rules are also used to solve the error localisation problem. For instance in a balance edit concerning the total costs and its details, it may be prescribed that if the details do not sum up to the total, some of the details are erroneous. However when these details are also part of other edits, changing a detail variable will have consequences for other variables that should be taken into account. To avoid such complications, it is often preferable to translate such simple rules into appropriate weights for the FH-approach.

20. ***Macro-selection (batch-mode, selection of units or variables)*** Macro-selection of either units or variables based on numerical macro-verification functions is similar to selection using scores. For instance, detection of outliers or anomalies can be based on type (ii) selection. In output controls, various

aggregates over time are aligned, and the corresponding absolute and/or relative differences obtained. Either type (i) or type (ii) selection can be used to pick out the variables for closer scrutiny. Since the underlying verification is batch-mode, also type (i) selection here is batch-mode in execution. A particular approach called *drilling (or drilling-down)* amounts to repeated type (ii) selections, from a suspect aggregate to its suspect sub-aggregates, and so on eventually to the suspect units. Macro-selection based on graphical macro-verification is similar to type (ii) selection: only the extraordinary units or variables are picked out by eyes, instead of by numerical risks or deviances. This can be very useful for detecting *abnormal patterns* in the data which are unknown in advance.

C. Amendment functions

21. Apart the execution mode, we contrast between amendment of systematic and random errors.

22. ***Amendment of systematic errors (typically unit-mode)*** From a pragmatic point of view, a systematic error is an error for which a plausible cause can be detected and knowledge of the underlying error mechanism enables a satisfactory treatment in an unambiguous deterministic way. De Waal and Scholtus (2011) distinguish between generic systematic errors and subject-related systematic errors. Generic systematic errors occur for a variety of variables in a variety of surveys and registers. A well-known generic systematic error is the so-called unity measure error which is the error of, for example, reporting financial amounts in Euros instead of the requested thousands of Euros. See Al-Hamad, Lewis and Silva (2008) for the detection of unity measure errors. Scholtus (2009, 2011) has developed algorithms to reliably detect and correct the following types of errors with a recognizable cause: typing errors, such as interchanging digits or adding or omitting a digit, sign errors, such as forgotten or incorrectly added minus signs, and rounding errors.

23. Subject-related systematic errors on the other hand occur for specific variables, often in specific surveys or registers. An example of a subject-related systematic error is wrong categorization of components of Revenues or Costs by the respondents. Restaurants, for instance, often incorrectly classify their revenues as revenues from trade (because they sell food) rather than revenues from services as it should be. Aelen and Smit (2009) found this to be the case in 10% of the questionnaires. Direct if-then rules can easily be used to correct such errors.

24. Amendment of systematic errors may be executed in batch-mode as the result of macro-verification and selection, which is relatively seldom in sample surveys but not uncommon in register-based statistical production. As typical examples we could mention meta-data related misclassification errors, or definition-related relevance errors.

25. ***Deductive imputation of missing or discarded values (unit-mode)*** Imputation is the estimation and derivation of values that are missing due to no-response or discarded for being erroneous. In *statistical* imputation, the randomness of the incomplete data is modelled either explicitly or implicitly. In contrast, *deductive imputation* proceeds deterministically by logical reasoning or from subject-related knowledge. When feasible, the true values are recovered by deductive imputation if the underlying rules are correct and the observed values used in these rules are true. It is possible to make the distinction between generic deductive imputation using methods that do not refer to specific variables and subject-related deductive imputation that does use substantive knowledge on possible values of specific variables and is applied using simple “if-then” rules.

26. By far the most common “if-then” rule is to impute a missing value by zero. Many of the detailed items in the SBS questionnaires are frequently left blank by the respondent. Although it is a practice that has been warned against by Kovar and Whitridge (1995), imputing zero for missing based on subject-matter knowledge, can be reasonable if the chance of an error in doing so is overwhelmed by that of correct reasoning. For example, if the cost for temporary employees is zero and the number of temporary employees is missing, then the missing value can be deductively imputed by zero. In practice there are many such pairs of variables where one of them must be zero if the other one is zero.

27. Generic deductive imputation of numerical data makes use of hard edit-rules, in particular in the form of equalities (balance edits) and non-negativity. In some cases these edit rules alone are sufficient to determine some of the missing values unequivocally. For systems of linear edits it is generally not obvious if some of the missing values can be determined uniquely by the edit rules, but simple algorithms exists that can solve this more general problem (De Waal et al., 2011).

28. ***Model-based imputation (batch- or unit-mode, random error)*** The term model-based is used here in a broad sense, covering not only parametric statistical models but also non-parametric approaches such as nearest neighbour imputation. Model-based imputation is batch-mode in nature, and treats the missing values as if they have arisen randomly, regardless of the true underlying cause.

29. Still, unit-mode model-based imputation functions can be motivated. For instance, in SBS, a missing component of turnover may be imputed by multiplying the previous (t-1) value of this component, provided it is available, with the ratio of the current total turnover to the previous total turnover. For units that are not observed at (t-1), a unit-mode imputation of the missing turnover component can be based on the (t-1) stratum mean proportion of this component towards the total turnover. In either case, the imputation is motivated by an underlying ratio model, but the parameter is not necessarily estimated in an optimal or efficient manner.

30. ***Adjustment of imputed values for consistency (unit-mode, random error)*** Partially imputed values often violate the edit rules. Although some imputation methods have been developed that can take edit rules into account (De Waal et al., 2011, Ch. 9), for many problems such models become too complex. The inconsistency problem can be solved by the introduction of an adjustment step in which adjustments are made to the imputed values, such that the record satisfies all the edits and the adjustments are as small as possible. This is an optimization problem (minimize adjustments) under constraints (satisfy edits). Various formulations have been discussed in Pannekoek and Zhang (2011). Notice that, in a general setting, the adjustable variables do not have to be among the missing values. Similar to the weights in an FH-algorithm, "adjustment weights" can be specified such that variables with large weights are adjusted less than variables with smaller weights, whether or not they are directly observed or missing initially and imputed. Such an adjustment function can be the last automatic amendment of micro-data, which ensures edits-consistent records with no missing values.

III. General flow of editing

A. Sub-processes and their delineation

31. Figure 1 shows the *general* flow of the E&I process. The square boxes may or must contain data-amending activities whereas the diamond ones *never* contain amending activities. Both are referred to as the E&I *sub-processes*, and we consider the delineation between them to be fixed, as well as the transitions (or flows) between any two sub-processes (i.e. provided they are connected by an arrow), in which sense the description may be considered generic. Depending on the process design, however, the sub-processes may consist of different statistical functions and activities, giving rise to variations in practice. Three different scenarios of implementation will be discussed in Section III.B. Here in Section III.A we first remark on the delineation among the sub-processes.

32. The diagram begins and ends with *Input micro data* and *Edited micro data*, respectively. The implications regarding the scope of the E&I process are embedded here. There are many necessary and important activities for editing, such as record matching and linking, unit and variable classification and coding, which are mentioned in the GSBPM under process 5.1 Integrate data and 5.2 Classify and code. It is possible that all such activities are already completed by Input micro data. But it is also possible that some of these activities need to be performed by the E&I process between Input micro data and Edited micro data. For instance, when editing data from multiple sources, data integration may not be a straightforward task, and may require considerable verification, selection and amendment activities. The same can happen to data classification and coding. Likewise, Edited micro data does not imply finalization of statistical micro data. Activities such as unit-imputation and weighting are usually to be completed *after* the data have been "edited". But it is also possible that such "estimation" activities may

have to be initiated during the E&I process. For instance, preliminary sample-weighting and the associated macro editing and output controls may have to be performed before "editing" can be considered finished. Thus, between Input micro data and Edited micro data, the E&I process may have to involve many other sub-processes, activities and statistical functions, but in the flow diagram here we only focus on the activities of data verification, selection and amendment.

33. The sub-process *Clerical interactive editing* is directly connected with all the other E&I sub-processes: it may be preceded by either of the two selection sub-processes (i.e. micro- and macro-selections), and it may lead to any of the sub-processes for automated processing afterwards. The sub-processes being connected in this way, it is possible to skip over some or iterate among them in execution. Although flexibility (or adaptiveness) is ideal in principle, the necessary logistics and management demands may lead one to prefer a more rigid work flow in practice. Moreover, the real-time process-flow control may very well be organized as a separate process and performed by different staff than the ones carrying out Clerical interactive editing.

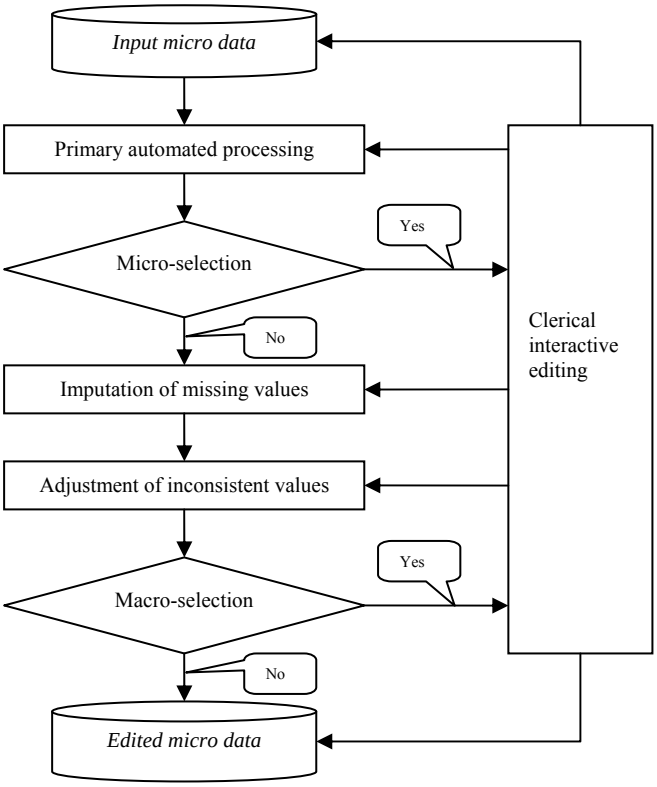

Figure 1: General flow of editing process

34. *Micro-selection* is carried out in unit-mode and can commence immediately as the data start to arrive, whereas *Macro-selection* requires batch-mode processing by definition. The distinction can be useful for improving the production efficiency in a situation where the data collection extends over a certain period of time. Still, it is important to make clear that the ordering is not necessarily in their initiation but rather their finalization. Take for example register data with persons as the units. All the data may arrive simultaneously on a single delivery, and there are apparently no critical units if the variables are categorical, such as the level of education, employment status, *etc.*, because all the units 'count' the same in aggregation and no one carries more weight or is more important than another. In such a situation, macro-verification and selection are often carried out first. Nevertheless, Macro-selection cannot be completed until all the eventual micro-data verification and selection are finished, and Macro-selection will always be the last stop in the iterations between two selection sub-processes.

35. *Primary automated processing* is a sub-process that is to be carried out before both selection sub-processes. Formally speaking, it may contain two types of bundling of activities: (a) pure data verification, (b) data verification-selection-amendment. In practice, a design will typically contain both

types of activity bundling, but with a varying balance between the two. The contrast will be illustrated between Scenario A and B in Section III.B, with examples from the Structural Business Statistics in both the Netherlands and Norway.

36. Between the two selection sub-processes are the sub-processes *Imputation (of missing values)* and *Adjustment (of inconsistent values)*. All the batch-mode amendment functions must be placed here because they cannot be considered finished until Micro-selection is. But the choice to put automatic amendment after Micro-selection is ultimately subjected to the process design, even when the involved functions are all unit-mode and can in principle be conducted in advance. The issue will be discussed under Scenario A and B in Section III.B. Next, Imputation or Adjustment is not necessarily chosen according to whether some data are missing or nor *per se*. Adjustment typically changes data as little as possible. For errors that are irregular or very large in size, minimal adjustment may not be a good idea. Localizing the errors first and then replacing them with plausible values by imputation may get us closer to the truth. Thus, even if some data are not "missing" to start with, they might still be amended mainly by imputation first and "minimally" by adjustment afterwards.

B. Alternative scenarios for the process flow

37. In this subsection we sketch three different scenarios of the E&I process that are constructed as a combination of the different E&I functions reviewed in Section II. The different scenarios constitute practical variations of the generic flow described in Section III.A.

38. ***Scenario A*** (Limited primary amendment, micro-selection followed by clerical editing, automatic verification-selection-amendment of uncritical units, macro-selection and possible interactive editing)

39. A scenario that emerges from the literature on selective editing (see e.g. Farwell, 2005), and that has been applied in the Dutch SBS until recently, is based on the belief that all important amendment should be manual and that the role of automatic editing is only to ensure internal consistency of the records, so as to avoid inconsistencies at aggregated levels. Such a scenario puts the emphasis on selection and clerical editing, automatic editing is confined to less influential errors. Since some important systematic errors, most notably the thousand-error will likely be treated automatically also under this scenario, we could have the following design of sub-processes:

- A1. Primary automated processing with limited automatic amendment of systematic errors. For instance, only the following functions are applied: unit measure (1000) errors and some subject-specific systematic errors. Unit-mode score functions are evaluated *after* the amendments.
- A2. Micro-selection of *critical* units for Clerical interactive editing based on score values and a unit-mode pre-specified criterion. For the remaining *uncritical* units, verification of the edit rules and (FH-) error localization of amendable variables.
- A3. Clerical interactive editing of the critical units.
- A4. Automatic amendment of the uncritical units, including deductive and model-based imputation of missing or discarded values, followed by adjustment for edits-consistency.
- A5. Macro-verification and selection and possible Clerical interactive editing.

40. Note that Steps A1 thru A3 are all unit-mode and are performed during data collection since timeliness is an issue. All functions in Step A4 can be performed automatically and the execution can wait until data collection and manual editing have been finished. For most functions in Step A4 this is not a necessity since they are unit-mode. For model-based imputation, however, batch-mode methods are preferred where the model parameters are estimated using the manually edited and the edits-consistent records from the current data rather than based on previous data or other reference data.

41. ***Scenario B*** (All unit-mode automatic verification-selection-amendment in primary automated processing, micro-selection followed by manual editing, possible batch-mode imputation and adjustment of uncritical units, macro-selection and possible interactive editing)

42. In Scenario B the emphasis on automatic editing is increased, to the extent that *all* the unit-mode automatic editing activities are placed in Primary automated processing. Because the input data are

amended or 'cleaned' to a greater degree in this way, Micro-selection may result in fewer selected units for clerical editing, and thereby reducing the amount of manual editing and improving the production efficiency. The overall design of sub-processes can be given as follows:

- B1. Primary automated processing with *all* unit-mode automatic editing, including all possible amendments by simple rule-based methods, methods for various generic systematic errors, error localization followed by deductive imputation based on edit constraints, and unit-mode imputation. Unit-mode score functions are evaluated *after* the amendments.
- B2. Micro-selection based on scores for Clerical interactive editing.
- B3. Clerical interactive editing of the units selected in Step B2.
- B4. Possible batch-mode Imputation and Adjustment.
- B5. Macro-verification and selection and possible Clerical interactive editing.

43. Now that the automatic amendment functions before Micro-selection are necessarily unit-mode, there is a particular design issue concerning model-based imputations (and subsequent adjustments). From a theoretical point of view, it is preferable to place them at Step B4, which has a drawback that Micro-selection by scores needs to be done on data with more missing values. The alternative is to perform unit-mode model-based imputation in Step B1, although the imputation is not optimal or efficient.

44. Both the recent redesign of the Dutch SBS and the on-going one in Norway move towards Scenario B. All the unit-mode amendment functions including unit-mode imputation and, in the Dutch case, FH-error localization and deductive imputation are executed before the scores for Micro-selection are calculated. To obtain results for the localization that the subject-matter experts found acceptable, the weights for the FH-algorithm had to be tuned carefully. Macro-selection is strengthened e.g. through type (ii) batch-mode selection based on unit-mode scores, as explained in Section II.B, together with macro-level indicators such as the weighted sum of all the relevant unit scores. Of course, other choices are possible. The point is that by monitoring the macro-level indicators over time, one is able to assess whether the editing process is 'converging' or not.

45. ***Scenario C*** (Limited primary automated processing and possibly no micro-selection; iteration between Macro-selection, Clerical interactive editing, and Imputation and Adjustment)

46. In this scenario the input micro data are received within a very short period of time or all at once, and the need to start editing during the data collection process is not urgent. Depending on the state of Input micro data, the sub-process Primary automated processing may consist only of data verification, unless processes Data integration and Classification and coding require verification-selection-amendment activities --- otherwise they are considered completed for the input data. Since the selection for clerical editing can be based on all the data from the beginning, Micro-selection may be skipped altogether. Scores can still be used but not necessarily in unit-mode, expected values can be calculated based on the current data, and type (ii) selection can be performed in drilling, graphical detection, *etc.* From then on, the editing process consists basically of iterations between Clerical interactive editing, automatic Imputation and/or Adjustment and Macro-selection. For instance, after having manually edited some problematic cases, the editors may recognize that the other cases may be dealt automatically by suitable imputation and adjustment functions. They can then invoke the sub-processes Imputation and Adjustment to do that. Apart from amending values, this can also include a localization (i.e. verification-selection) function which just sets the recognizable erroneous values to missing and let the automatic imputation and adjustment procedures to achieve edits-consistency. The overall design of sub-processes under Scenario C can thus be given as follows:

- C1. Primary automated processing, possibly with only data verification.
- C2. No Micro-selection, or possibly very limited Micro-selection.
- C3. Iteration between:
    - C3.1. Macro-selection
    - C3.2. Clerical interactive editing
    - C3.3. Imputation and/or Adjustment.

47. This approach is applied in the Dutch statistics on child care institutions. The questionnaire contains the common SBS variables and a number of specific variables related to the types of care that is provided. There is no step C2 of Micro-selection. There are many applications of the approach in Norway, since Scenario C is typical for 'straight-forward' statistical production based on administrative data. Data integration and Classification are often the more demanding activities there.

V. References

Aelen, F. and R. Smit (2009), *Towards an Efficient Data Editing Strategy for Economic Statistics at Statistics Netherlands*. European Establishment Statistics Workshop, Stockholm

Al-Hamad A., D. Lewis and P.L.N. Silva, (2008), *Assessing the Performance of the Thousand Pounds Automatic Editing Procedure at the ONS and the Need for an Alternative Approach*. Working paper No. 22, UN/ECE Work Session on Statistical Data Editing, Vienna.

De Waal, T., J. Pannekoek and S. Scholtus (2011), *Handbook of Statistical Data Editing and Imputation*. John Wiley and Sons, New York.

De Waal, T. and S. Scholtus, Methods for Automatic Statistical Data Editing (2011), Report, Statistics Netherlands, Den Haag.

EDIMBUS (2007), *Recommended Practices for Editing and Imputation in Cross-Sectional Business Surveys.* Manual prepared by ISTAT, SN and SFSO.

Farwell, K. (2005). *Significance Editing for a Variety of Survey Situations.* Paper presented at the 55th session of the International Statistical Institute, Sydney.

Fellegi, I.P. and D. Holt (1976), *A systematic Approach to Automatic Edit and Imputation*. Journal of the American Statistical Association 71, pp. 17-35.

Granquist, L. and J. Kovar (1997), Editing of Survey Data: How Much is Enough? In: *Survey Measurement and Process Quality*, L.E. Lyberg, P. Biemer, M. Collins, E.D. De Leeuw, C. Dippo, N. Schwartz, and D. Trewin, eds. John Wiley & Sons, New York, pp. 415-435.

GSBPM (2009) Generic Statistical Business Process Model. *Version 4.0*, UNECE.

GSIM (2012) Generic Statistical Information Model: Communication. *Version 0.4*, UNECE.

Hidiroglou, M.A. and Berthelot (1986), *Statistical Editing and Imputation for Periodic Business Surveys*. Survey Methodology 12, pp. 177-199.

Kovar, J. and P. Whitridge (1995, Imputation of Business Survey Data. In*: Business Survey Methods*, B.G. Cox, D.A. Binder, B.N. Chinappa, A. Christianson, M.J. Colledge, and P.S. Kott, eds. John Wiley & Sons, New York, pp. 403-423.

Latouche, M. and J.M. Berthelot (1992), *Use of a Score Function to prioritize and Limit Recontacts in in Editing Business Surveys*. Journal of official Statistics 8, pp. 389-400.

Lawrence, D. and R. McKenzie (2000), The General Application of Significance Editing. *Journal of Official Statistics 16*, pp. 243-253.

Pannekoek, J. and L.-C. Zhang (2011), *Partial (Donor) Imputation with Adjustments.* Working Paper No. 40, UN/ECE Work Session on Statistical Data Editing, Ljubljana.

Renssen, R. and A. Camstra (2011), *Standard Process Steps in Statistics*. Report, Statistics Netherlands, Heerlen.

Scholtus, S. (2008), *Algorithms for Detecting and Resolving Obvious Inconsistencies in Business Survey Data*. UN/ECE Work Session on Statistical Data Editing, Vienna.

Scholtus, S. (2011), Algorithms for Correcting Sign Errors and Rounding Errors in Business Survey Data. *Journal of Official Statistics 27*, pp. 467-490.