Working Paper No. 7
ENGLISH ONLY

STATISTICAL COMMISSION and ECONOMIC COMMISSION FOR EUROPE

STATISTICAL OFFICE OF THE EUROPEAN COMMUNITIES (EUROSTAT)

CONFERENCE OF EUROPEAN STATISTICIANS

Joint UNECE-EUROSTAT Work Session on Registers and Administrative Records for Social and Demographic Statistics
(Geneva, 9-11 December 2002)

Invited paper

LONGITUDINAL ANALYSIS USING DATA WAREHOUSE TECHNIQUES

Submitted by Marton Vucsan and Jan Kardaun, Statistics Netherlands

I. INTRODUCTION

1. In the past decades, Statistics Netherlands has used sequential files and databases to do its analysis. To that end, many a program has been written and even a number of home-grown standard tools have evolved. However, the view on data processing has been almost exclusively sequential, in spite of database use.

2. The limitations of this mode of processing are many; most important is the problem with linking different files, using one or several common variables. Usually linking is done with two files at a time. Little context is present since most of the time a limited number of variables is present. Because of this, linkage errors are common. Although progress has been made with providing a common linking identifier on a person level for a great number of related files, it is still difficult to see all the variables as one large dataset from which subsets of arbitrary sets of variables can be drawn.

3. To complicate the linkage, there is also a temporal pitfall. The majority of the observations are time dependent. In order to obtain correct linkage, everything has to be observed and stored with a time-stamp. Good examples are surveys that relate to the actual situation, or that are linked using the current address where the respondent is living. A full linkage would imply lasting simultaneous linking across person, address, time, etc. After the linkage, the combined data have to be made consistent ('complete micro integration'). When the prerequisite of correction of errors with preservation of the original values is added, one sees a system of overwhelming complexity when carried out with the traditional (sequential) means of automated processing. Furthermore, if we are to undertake successful longitudinal analysis, time has to be present as a variable and not as a snapshot date. New and very different tools are called for.

4. In this paper, we will elaborate on these new tools and why they are in essence just a different way of using old principles. We will show, with the copy we keep of the Dutch municipal registers and the daily updates, that a data warehouse will suit our needs. Furthermore, we will show that with an event-based data warehouse some interesting basic longitudinal analysis can be done.

5. With a system that enables instantaneous analysis, a better understanding of the data can be achieved, in a way that is a 'natural' human way of learning: exploration. When it is possible to explore the relevant data in more than a hundred different ways a day, it will be possible to gain a better intuition of the phenomena hidden inside these data.

II. NEW TOOLS FOR THE POPULATION REGISTER (AND OTHERS…)

II.1 The population register

6. Statistics Netherlands is legally entitled to receive a yearly snapshot of the municipal population registers and is connected to the network through which the daily mutations of these registers flow. Therefore, we keep two sets of files with two sets of processes. One amounts to the integrated yearly snapshot from which all the yearly statistical output comes that is based on totals of persons living at an address at January first of each year. The other could in theory amount to a file with longitudinal data in it and is used for fertility, mortality, migration and marriage.

7. As mentioned before, linking files is already difficult enough, and is complicated by the time element, i.e. the validity period of an observation. In the files Statistics Netherlands keeps as a more or less obsolete copy (obsolete meaning a few months late) from the municipal registers, this phenomenon is painfully present. The two sets of files representing the yearly snapshot and the continuous stream of events are very difficult to synchronize, even though they come from the same source. The main reasons are errors and corrections, and inaccurate time-stamping or non-synchronous processing.

8. Because of the fact that Dutch statistics are increasingly based upon registers, and the registers are used as a foundation for the surveys underway, it pays to do some serious bookkeeping. What this means is that the better the quality of the registers, the better the quality of the overall statistics.

II.2 New storage approach

9. Statistics Netherlands (like other NSIs) processes a considerable amount of data, with emphasis on efficient throughput rather than on fast response even at high peak loads. The main objective is efficient use of computer resources.

10. In the past decade, the commercial sector developed a need for software that would enable salespeople and business analysts to obtain meaningful figures from their administrations and point of sales machinery. The basic problem was that analysts and salespeople could not tell in advance what kind of figures they were interested in, but they wanted fast answers. The new software had to provide answers for all possible questions with little data manipulation or programming (salespeople are not very good at programming). What came up was a construction that would later be called a dimensional data model. In this model, which is a form of the relational database model, the observation or event is the central point of interest.

11. Because of the fact that the salespeople could not tell in advance what figures they wanted, the system had to deal with individual data (‘micro data’). Only micro data would enable all kinds of arbitrary aggregates to be produced. The naive form of such a system would be a large flat file with all the attributes present. With all the attributes, we mean things like age, membership of a 5-year age class, address, nationality, etc. With this file, restricting on attributes and accumulating the fields with measured values in them can make all possible aggregates. Although theoretically possible, the practical drawback would be the large amount of data you would have to read.

12. For a file containing the Dutch population, that would amount to following statistics: there are 16 million people in the Netherlands. To fully describe the person in terms of the municipal registration attributes or variables, it would suffice to reserve 2000 bytes per person. The file size would then be 32 Gigabyte. With an average machine like those present in most NSIs, this would take an hour for every result that is to be produced. For salespeople this would truly be unacceptable. It becomes even worse when they want to produce a cross-reading of one or more variables because then it is possible that a number of complete reads would be necessary.

13. However, it is not very likely that 16 million people would have nothing in common. For instance, there are only 6 million addresses where they live. Therefore, there are ways to reduce the size of the naïve-form dataset. Many variables have only a limited number of values, even for a dataset with 16x106 Dutch residents. There are only 120 age values (in years) possible, or 1200 on months, and so on. Therefore, if we can code attributes and put them in separate tables, we could reduce the width of the main file. It would make a huge difference. Suppose we create the following sub files; address, age and classifications, relation or marriage, country of origin, etc. The record width of the main file would shrink to 150 bytes; the record would consist of a large number of keys and a few measured values. The price is that we have to link (join) with the sub-files containing the labels or descriptions.

14. One additional step is needed: instead of coding all the variables separately, we will combine a few variables that usually belong to one entity (e.g. street name, house number, house number suffix, postal code, city) to a sub-table (i.e. address). The idea is that there are relatively rare changes in these sub-tables, as compared to changes in the main table. Above all, we separated our file into the events we observed and the entities that describe the events.

15. To process this construction, we would greatly benefit from using a database. This has no theoretical impact; it is just a practical measure. A database is a very well-suited tool for maintaining the relations between files. Of course, by doing this we will be able to use the SQL language to our advantage. The SUM, COUNT and GROUP BY functions and clauses will be particularly handy. If we want to

tabulate for a selection of the people older than 50 years, we would restrict the age table with the condition [>50] and find all records of our main table that have one of these keys (from the age table). This will require a complete read of the main table and some reads from the age table. It will be finished in 5 minutes. (Reading 10 Mbytes/sec.) With this solution, we achieved a 10-fold increase in performance. This is the basic form of a data warehouse. The large table from which we took all the addresses, ages, etc., so that nothing was left other than the keys and the measures, is what data warehouse specialists call a fact table. The tables with address, age, etc. are the dimensions; a fact table with its dimensions is what is called a data mart. The main difference with our traditional way of doing things is the way we group the codes together in dimensions to create meaningful entities.

16. So far, there is no magic involved; we just split a record into its (meaningful) components and put them in a database. However, there is still no mention of time. The file considered is, in essence, a snapshot of the population at a given moment - let us say 1 January. In order to do longitudinal analysis we have to put time into the file and record events. As was mentioned in the paragraph on the population register the other input stream is a series of messages that constitute actual mutations made to the original population register. With these messages, we can build a file that carries all information that was received for all the events that have something to do with migration. Of course, this file would be huge. If we assume that for every person, there are approximately 12 events during his or her lifetime (birth, marriage, 8 changes of address, death….) then, clearly, a more sophisticated form of data storage is required.

17. If again we were to dissociate this file into its dimensions and one fact table, we would obtain a data mart where, for all persons in the Netherlands, their whereabouts are being recorded over time. The reduction would be the same as for the snapshot file: a factor of 12. Still, the file would be too large for rapid response sequential processing. However, once the data is structured into data marts with fact tables and dimensions, there is another trick that comes into play; We can use what is called an OLAP tool to improve the performance and bring the response times into the realm of seconds. The abbreviation OLAP stands for On Line Analytical Processing. In the next paragraph, the workings of these (commercial) OLAP tools will be explained.

18. We have been concentrating on the size of the fact table to achieve performance and to make it easier to query the whole dataset. There is, however, more to gain than performance, and that is ease of maintenance. Splitting the longitudinal dataset into its dimensions and facts makes it possible to create lasting links with other datasets through so-called conformed dimensions. This can be explained as follows: when we split the dataset into different dimensions, we decided to put all the events in the form of a set of keys into fact records. This means that a dimension is rather stable. The events are recorded as combinations of references to the various dimensions records. Most of the attributes in a dimension will change very seldom. Still, if we are to obtain correct linkage through time with e.g. the person dimension, we will have to see to it that the changes will be made with respect to the time they occurred. For this, the "slowly changing type 2 dimensions" were conceived. If an attribute, or variable, changes at a given moment in time, we create a new record for the person with all attributes equal, except the one attribute or variable which was changed. New facts will use the new dimension record. So, if a person has a change of nationality, a new (person-) dimension record will be created and used from thereon. The advantage is that, if we query the facts (or events) in the past, the same person will still will have the same nationality. Another advantage is that, in addition to the traditional code labels, we now also have composite code labels adding up to meaningful entities like person and address, etc.

19. The dimensions can now be used for other datasets as well. This means that, with the same person and address dimensions, we can have both a migration fact table as well as a fact tables that describes the amount of monthly salary or marriages and divorces. The implications of this are that all data of a department of social statistics could be used as one big online dataset.

II.3 OLAP tools explained

20. The basic mechanism behind OLAP tools is the fact that an aggregate produced from one of the dimensions with the fact table can be used for all computations using an equal or higher aggregation level

in the hierarchy within the dimension. The making of an aggregate means that the fact table is summed over a lower level of a dimension. For instance, we create an aggregate of 5 years classes from the migration data mart. What happens is that all the fact records that belong to the same 5-year class are cumulated into one fact record. For the condensed fact table and the condensed dimension where the lowest level is missing, separate tables are created. Now in every query where age is on or above the 5-year class (10 or 15 or just young/middle/old), we can use the newly created aggregate. If we just want to know the number of people moving from one city to another, irrespective of age, we use the aggregate we created. This will save us an estimated factor 5 or more in response time. We don't use many of the dimensions in our query. This means that if we had aggregates summed over these dimensions, there would be fewer records to process.

21. The above describes exactly what an OLAP engine does; it will try to create as many aggregates as there is space or time to make them. Most engines have a piece of artificial semi-intelligence that will try to optimise the choice of aggregates. Normally very little human intervention is necessary. Performance increase can be quite dramatic using this method. Modern OLAP engines have a response that is within seconds for all queries that are common and do not drill down to the micro level. These are precisely the type of queries undertaken in the statistical offices.

22. The OLAP engine takes care of automatic navigation of aggregates and decides when to use the micro data. Most engines keep the aggregates in a proprietary file structure to further speed things up. Normally these engines carry all kinds of luxury such as access security, audit trails, etc.

III. LONGITUDINAL USAGE OF THE WAREHOUSE

III.1 Longitudinal data mart for the population register

23. For the population register, a number of longitudinal data marts were created. The most important one is the migration mart. Let us take a closer look at how such a data mart looks in its simplified form:

The fact table (simplified) looks like this:

Date of event key	
Event type key	(migration, death, birth, immigration and emigration)
From location key	(location or address in the role of "From")
To location key	(location in the role of "To")
Person key	(to the record that describes person at this moment)
Age Person key	(age at the moment the event took place)
Person Mother key	(person in the role of mother)
Person Father key	(person in the role of father)
Elapsed years since previous event	
Number one	(just to make it easier for SQL; field = 1)

The dimensions (simplified) look like this:

DATE Dimension	
Date key	(just an integer key for linking to facts)
Year	year numeric
Month	name of month
Daynr	day number
WeekdayName	Weekday in words
Date as YYYMMDD	exact date text from source data
IsValid	Is this a valid date (Y/N)

EVENT TYPE Dimension	
Event key	(just an integer key for linking to facts)
Event type	categorizes the event
Event descriptor	describes the event: birth, migration, death.

AGE Dimension	
Age key	(just an integer key for linking to facts)
Age 10 Year class	
Age 5 Year class	
Age 1 Year class	
Age in months	we could have days but this is good enough
PERSON Dimension	(slowly changing)
Person Key	(just an integer key for linking to facts)
Register identification	(number of the official file of birth)
Date of birth	(YYYYMMDD) just text
Gender	(Male/Female/something else)
Country of birth	
Etc. lots of variables that are carried along for a lifetime with only incidental change	
LOCATION Dimension	(slowly changing)
Address key	(just an integer key for linking to facts)
Province	name of province
Municipality	name of
Street	name of
Number	
MAP coordinate	the actual location on the globe

24. In the above, all births, deaths and migrations (within and outside the country) are stored as they occur. The aim is to create running totals of the population with respect to all kinds of variables, such as location or age or gender, etc. Births and deaths are in the same data mart because they have the same effect on the population as immigration and emigration, or even normal migration between areas. They all add up to the sum of the number of people present at a given time in a given area. (even if you never move during your lifetime, in the end you move, albeit from the planet).

25. Although the datamart is populated with real population register mutations from 1995 onwards, the history for all living persons in 1995 has been reconstructed using additional data from the population snapshot. This snapshot carries additional birth and marriage records from the past. Together with the birth date and place, a (fragmented) history of all living persons could be created. The advantage of having a complete history of all living persons is that longitudinal analysis will be much easier.

26. With the above described data mart we can do quite interesting longitudinal analyses. We start with some basic examples: using the elapsed time since the previous event, we can see what the average number of years is that people live in the same house,. We can see this with respect to age (all age classes), location, gender, etc. By creating an OLAP cube we can see all this in a matter of seconds. In the next paragraph we will explore this type of analysis further.

III.2 Simple longitudinal analysis

27. Event history analysis based on duration models generally has the following form:
"The dependent variable measures the duration of time that units spend in a state before experiencing some event. Generally, a researcher knows when the observations enter the process, i.e., when the history begins, and when, and whether or not, the process ends (with the occurrence or non-occurrence of some event). Analysts are typically interested in the relationship between the length of the observed duration and independent variables, or covariates, of theoretical interest. A statistical model can then be constructed to link the dependent variable to the covariates. Inferences can be made regarding the influence of the covariates on the length of the duration and the occurrence (or non-occurrence) of some event. " (J. Box-Steffensmeier)

28. Normally, we would pick one or two variables that define a population, and analyse, from a given moment on, the events that happen in this population. For instance, we could take all married people and analyse the moment of childbirth. First, we have to define “All married people”. Looking back and forth over time there is no such thing as “all married people”. We have several choices: all married people at a certain moment in time (t0); all people that are or have ever been married at t0; all people born in a certain year (or period p) and that are married at t0; all people born in p and ever married before t0, whether they are dead or alive at t0. We can also select all people born after a certain year, but do not perform analysis by their calendar age, rather by the time expired after the moment of marriage.

29. We could study, for example, in this population the sequence of marriage and childbirth. The essence here is that the population where the event of childbirth took place is always related to the other part of the population that has not (yet) experienced the event at a given moment in time, because our observation was too short to see the event happen. In this way, we can now compute the chance of childbirth occurring in every married year, and model this in various ways,. In a way, we are looking into the future, not from the present, but from the past, computing probabilities.

30. The above model is not very easy to implement in an OLAP environment (at least, that is our current thinking). The main problem is that OLAP only works well if all the data can be considered to come from one "rectangular super-table", from which tabulations can be sped-up using the techniques outlined above. For example, we can very well observe the population that gave birth and compute the time elapsed since their marriage, we cannot equally easily compute the population that could have given birth, but did not. Of course, in theory, we could create a number of queries and obtain these figures from the data warehouse for further processing, but this would not satisfy our wish for instantaneous analysis. In longitudinal analysis, population elements (e.g. persons) and events have to be rearranged in complex ways, depending on the research question. Moreover, there would be some work involved to solve the problems of censoring.

31. For some phenomena it is justified to simplify the model by looking at events only, and not at the population “at risk” for some events. So, if we define the observed population as the population that experienced an event during the time interval under observation, we would only need the elapsed time since the previous event to do some interesting analysis using OLAP cubes. When we record the duration of the previous state, or the time elapsed since the previous event, together with the event, we can easily and quickly perform most of the analyses. We still have to deal with problems of left and right censoring. These arise because some of the events might be outside our observation possibilities (e.g. childbirth while temporarily expatriated).

32. Studying events only, and not the population from which the events can originate, is a good approximation if the population is stable (during the observation period) in its characteristics, if not in its composition of elements. Moreover, it is to be hoped that the systematic non-observation of events (e.g. non-response) is no worse than observation errors of the population.

33. Sometimes, we don't have a choice, e.g. if the events are countable, but the underlying population evades observation (like capital crimes and 'criminals'). Then events-only analysis can still be useful. The obvious risk with this type of analysis is that it is quite possible to misinterpret the results. This model can be profitably used for measuring events that themselves influence and define a new sub-population. For example, people that married (widowed), moved, obtained (lost) a job, etc..

34. If we take migration as an example, and confine ourselves to event-only studies, we can calculate the mean time people spent at their previous address, how many people (i.e. movers) move 0, 1, 2, etc. times (during the observation period), how many people return to their place (or region) of birth (or youth) after retirement.

35. We can now explore two different types of analysis:
(i) we can explore, within the same interval the differences in different age groups or geographical differences etc. all within the same time frame; and

(ii) we can explore, applying the same set of variables, the change in behaviour over time by looking over the same interval duration to different moments in history.

Type 1: exploring within the same time interval

36. Let us examine the case of people moving to a given city. First, we want to know if this city attracts people in certain age classes; we could use our OLAP cube to tabulate in 5 or 1-year classes the movers. We could look at their nationality, or income (not available immediately), how many persons are in the 'moving party', including dependent children. Ideally, we have to compare these characteristics with those of the Dutch population, but if this is not available, we can compare the characteristics of movers to one city with those of all movers in the Netherlands.

Type 2: exploring over time, using same interval duration

37. As a last example we could be interested in the behaviour over time of this moving population and do the same exercise in different years over a period of, say, half a year (in order to obtain enough cases). If we do this every 5 years, we can see the time spent at an address as a function of cultural and economic parameters. We can also see the influx of residents changing over the years in absolute terms, and we can see the changing composition of the town or district.

IV. CONCLUSIONS

38. We tried to make two points. One: by use of data warehouse and OLAP techniques, it is possible to present answers to many ad hoc requests for tables from a large data set (a population register) within seconds. Two: the population register contains many interesting and important facts that require a more complex analysis (e.g. longitudinal), but meanwhile, rapid exploration with event-only data, using these same data warehouse techniques, is an important help. By using OLAP techniques, we will be able to better calculate which variables are of importance and what models will be adequate to describe the observed phenomena.

LITERATURE

The Data Warehouse Toolkit, Ralph Kimball, Wiley; ISBN 0-471-15337-0

The Data Warehouse Lifecycle Toolkit, Ralph Kimball, Wiley; ISBN 0-471-25547-5

Building the Data Warehouse, W.H. Inmon, Wiley; ISBN 0-471-14161-5

Dynamics in Marriage and Cohabitation, D. Manting ISBN 90-5170-295-7

H. Goldstein. The design and Analysis of Longitudinal Studies - Their Role in the Measurement of Change. London: Acad. Press, 1979. [xvi, 199 pp.] ISBN 0-12-289580-0

J.A.P. Hagenaars. Categorical Longitudinal Data: Log-linear Panel, Trend, & Cohort Analysis. Sage Publications, 1990.

– – – – –