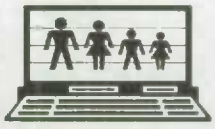
SPSD/M 

X-tab User's Guide

This guide contains a complete description of the SPSM user-specified cross-tabulation facility, as well as a set of examples. It is intended to be used as both a self-contained tutorial and as the definitive reference guide. A summary of the X-tab facility is located in the SPSD/M User's Guide.

October 31, 1997

Statistics Canada Statistique Canada

Canada

Table of Contents

Introduction

This guide describes the X-tab facility of the SPSM, documenting what it is, what it can do for an analyst, how it interacts with other portions of the SPSM, and how one uses it to produce the desired tables.

The X-tab facility is one of the analyst's three major channels for obtaining analytic results from an SPSM run. The other two are:

- the set of built-in standard tables that are an integral part of the SPSM, described in the *SPSD/M User' Guide*; and
- the SPSM's ability to export selected data to a file for subsequent processing by PC-SAS, described elsewhere in the same guide.

One additional channel, the export of information to an ASCII file, also exists, but is used primarily for examining specific households or individuals; this text output facility is also described in the *SPSD/M User' Guide*. The analyst will use the X-tab facility when:

- the desired information is not available from the standard set of tables; and,
- the X-tab facility provides a more effective route than exporting the data for secondary analysis using PC-SAS or the text output facility.

Small enough and fast enough to be integrated right into the SPSM, the X-tab facility offers considerable power in an easy to use format. For example, the X-tab facility recognizes the importance of well-labeled results, offering substantial automatic labeling, but permitting the analyst to supply customized labels. It can tabulate a wide variety of variables directly from the database or from SPSM models, as well as expressions involving such variables, and to roll the results up to any of five levels of unit of analysis. Other features include the capacity to generate several tables in a single table specification, to generate multi-dimensional tables, and to include several tabulated variables in a given table.

This guide is written first as a reference manual and secondarily as a training guide. Experienced users will be able to find things quickly using the table of contents, while first time users can read the chapters in order for an introduction to all aspects of the X-tab facility. The numerous examples are intended to serve to both classes of readers. Users who are familiar with the principles of cross-tabulation, and who prefer to learn by example, may prefer to turn directly to Section *The Table Set Specification*, and fill in the gaps later by scanning appropriate sections of this guide.

Terminology

This section introduces the terminology used in this guide. The presentation uses a "typical" table that illustrates several of the more important automatic and user-controlled components. Subsequent sections of this chapter then define the individual components more fully. The focus of this terminology section is to provide an introduction prior to the subsequent detailed definitions. In essence, it gives a first cut look at cross-tabulation in the SPSM.

The presentation is keyed to an annotated version of a typical SPSM crosstabulation, shown in Figure 1, and to the table specification that produced it. Since the goal of the presentation is primarily one of terminology, this illustration makes no attempt to demonstrate all the capabilities of the X-tab facility. Further, the reader should be aware that, both here and elsewhere throughout this documentation, the illustrative tables are based on a particular subsample of the SPSD. Attempts to replicate the examples using the SPSM may produce tables that do not precisely match those shown.

Figure 1:

Table 1U: Unit Count (000) for Census Families by Province and Census family type

	Census family type						
Province	With Kids, 1 Adult	With Kids, 2+ Adult	With Elderly, 1 Adult	With Elderly, 2+ Adult	Other, 1 Adult	Other, 2+ Adult	All
Newfoundland	2.4	80.3	18.1	24.3	33.2	17.8	176.2
P.E.I.	2.9	12.5	8.2	4.1	4.8	6.9	39.4
Nova Scotia	7.2	119.9	36.7	44.2	46.5	63.6	318.2
New Brunswick	11.8	87.6	30.3	36.1	37.7	53.3	256.9
Quebec	138.1	854.0	206.5	253.7	566.8	580.6	2599.6
Ontario	262.2	941.8	463.4	448.2	613.0	883.8	3612.5
Manitoba	7.9	130.6	48.2	36.9	82.1	93.8	399.5
Saskatchewan	40.7	135.8	20.5	46.0	75.1	94.5	412.6
Alberta	59.8	311.8	86.5	60.0	306.8	192.0	1016.9
B.C.	59.1	385.1	129.3	94.9	318.3	234.9	1221.6
All	592.3	3059.5	1047.8	1048.3	2084.3	2221.2	10053.4

It was produced by the specification:

```
XTSPEC
CF: {units}
    * hdprov+
    * cftype+;
```

or on a single line:

```
XTSPEC
CF: {units} * hdprov+ * cftype+;
```

Tabulated Variable(s)

The essence of a cross-tabulation is that it should tabulate, i.e. count or add up, something, and present the results in a table. That "something" is termed a "tabulated variable". Generally speaking, a tabulated variable can be any analysis variable available in the SPSD/M. (Analysis variables contain data that can be "added up".)

Tabulated variables may come from:

- the SPSD (database) variables,
- SPSM (model) variables,
- User's defined expressions or variables, or
- may be expressions defined "on- the-fly" inside the table specification.

The X-tab facility permits multiple tabulated variables in a single table. It permits tabulations of either "unit" type variables (e.g. census families) or dollar-denominated variables (e.g. Family Allowance benefits), as well as those defined by the analyst.

In Figure 1, the tabulated variable is the number of "units". We recognize this from the title that the X-tab facility has automatically generated for the table, "Table 1U: Unit Count (000) for Census Families by Province and Census family type". In composing this title automatically, the X-tab facility has incorporated its knowledge of the "unit of analysis" described immediately below.

Unit of Analysis

Analysts know that for some kinds of tables, the inherent unit (level) of analysis is critical. If we want to count the number of families in Canada, it will make a difference whether we are talking about economic families, census families, or nuclear families. Similarly, precision in defining the unit of analysis is important to tabulate average income per unit. The X-tab facility permits the analyst to present tables at five distinct units of analysis: households, economic families, census families, nuclear families and individuals.

In Figure 1, the analyst has chosen the census family as the appropriate unit of analysis. As described above, the X-tab facility indicates this choice in the title that it constructs for the table; i.e. the title indicates explicitly that the table provides a "Unit Count (000) for Census Families".

Classificatory Variables

The goal of the X-tab facility is to permit the analyst to display certain tabulated variables in terms of classes or categories defined for other variables, and in terms of combinations of these categories. From the perspective of both the mechanics of producing such a table, and the analyst's ability to understand it, the number of such categories should generally be small. In the SPSM, such variables are termed "classificatory" since they classify any record into exactly one class or category. The SPSM ensures this discreteness property by requiring that

all classificatory variables be integers. The variables will either be defined as integers in the database, or created as integers by their definition using the split function in the user variable facility UVAR or user defined variables in "glass box".

In Figure 1, the analyst has tabulate the numbers of census families in two classificatory variables. The column categories correspond to various family types (e.g., older units, two-adult families with children, etc.), with the category definitions implemented such that each census family falls into exactly one category from the set. The row categories correspond to the province of residence, with any given census family residing in exactly one province at the point in time that the Survey of Consumer Finances collected the information. Both of these classificatory variables are available from the SPSD.

Most of the classificatory variables are variables include in the **Individual and Family Characteristics** section, and flag variables in the other sections of the *SPSD/M Variable Guide*.

Labeling and Format

Tables are not of much use to their readers if they are not easily comprehensible. Consequently, the X-tab facility ensures that the content is clear and that it appears in an appropriate format. This cross-tabulation documentation is above and beyond the extensive page labeling that the SPSM provides for all of its outputs.

- The heading of a table indicates what is being tabulated and what classifications are being applied.
- Row and column labels specify the relevant combinations of the table's classificatory variables.
- For tables of dimension three or higher, additional "segment control" information is also included in the heading information of the individual table segments.

In Figure 1 the row categories for the tabulation, province of residence, appear in the left-most column, clearly labeled by text strings identifying the provinces of residence. The X-tab facility retrieved these strings from the "variable description" portion of the database and applied them automatically. Similarly, the X-tab facility retrieved and applied the column headings that denote the type of family.

Less obviously, the X-tab facility has chosen appropriate formatting conventions. For example, it has scaled the cell entries as thousands of families, indicating this choice by the "(000)" portion of the table title. Similarly, it has chosen to present a single digit after the decimal place. The analyst would have been able to override either of these choices had other formats been more appropriate.

The Table Set Specification

The means of specifying what tables are desired is termed the "table set specification" or "cross-tabulation specification" and is implemented via the `XTSPEC` parameter in the

control parameter file. The parameter permits the analyst to specify the set of desired cross-tabulations.

Appended to the bottom of Figure 1 is the table set specification that produced the table occupying the bulk of the figure. The SPSM does not itself include the specification in the tables that it produces. This particular specification requests only the single table shown; in general, however, a single XTSPEC parameter can initiate the production of a wide variety of distinct user-defined tables.

The XTSPEC used here consists of the text string:

```
XTSPEC
CF: {units}
    * hdprov+
    * cftype+;
```

- The request can be set on two lines but it is suggested to use a multi-line request to ease the reading and modification of the request.
- The CF: symbol in the request indicates that the table uses the census family as the underlying unit of analysis.
- The identity of the tabulated variable is conveyed by the "units" identifier appearing within the curly braces; i.e. census family units are to be tabulated.
- The desired row and column variables are defined by the hdprov and cftype identifiers familiar to the reader from their inclusion in the list of SPSD variables (see the *SPSD/M Variable Guide).*
- The + characters that appear after the names of these classificatory variables indicate that an "ALL" category is to be created for each of them.
- A semi-column is required at the end of a table request.

This brief example provides only the most general introduction to the capacities of the X-tab facility. The next sections take up the required detail, addressing the data to be included in the tables, the nature of the table set specification and the individual table requests it contains.

Record Selection and Record Weighting

Record Selection

Any meaningful discussion of table generation process must begin with a precise specification of the data that are to be processed in the construction of the tables. To begin with, the user specifies, in the control parameter file, the identity of the database to be used for the run, and thus for the generation of the tables produced during the run. For example, the analyst might specify the full SPSD or the 5% subsample of it.

In addition, also via the control parameter file, the analyst may specify that only certain records are to be processed. More specifically, the analyst can use the SELFLAG, SELUNIT, and SELSPEC parameters to restrict the data processed by the SPSM (See the *SPSD/M User's Guide* for a full description of these parameters).

This selection capacity makes for considerably greater computational efficiency (and thus shorter run times) and for more focussed tables when only certain records are relevant for an analysis (e.g., when only nuclear families with children are of interest). When the analyst uses the selection parameters to exclude certain records, he or she must accept the responsibility for interpreting the resulting tables only in the context of the data selected for the run.

The analyst should also note that the record selection criterion is related to the choice of the unit of analysis. The analyst controls whether the selection specification is interpreted as counting all individuals in a unit that meets the selection criterion, or just those individuals within a larger unit that meet the selection criterion. For a more detailed discussion the analyst should refer to the sections of the *SPSD/M User's Guide* pertaining to the selection parameters and the notion of "roll-up" across units of analysis.

Record Weighting

Similarly, because the SPSD is a weighted data set, the generation of tables must address how the weights of the individual records affect the outcomes. The Survey of Consumer Finances, on whose sampling frame the SPSD/M is based, is a stratified random sample. This means that not every household in Canada had an equal chance of participating in the Survey. Consequently, attached to each record in the SPSD is a "record weight", essentially the inverse of the probability that the household would be chosen for the Survey, or, equivalently, the number of Canadian households that the record represents.

In the implementation of the SPSD/M the record weights are stored in a different file than the information about the characteristics of the households and individuals. Thus, the same file of unit characteristics can be used in conjunction with multiple weight files (e.g. weight files corresponding to different demographic growth assumptions). The control parameter INPWGT contains the name of the particular weight file to be used for a run. Additional detail on these record weights, their interpretation and their derivation, is available in the

SPSD/M User's Guide and the *SPSD/M Database Creation Guide*.

In most X-tab facility tabulations, as well as in the SPSM menu of standard encoded tables, the table generation process weights the tabulated variable by the relevant record weight. The intention here is that the table is thus "blown-up" to the overall Canadian totals (for the kinds of records meeting the selection criterion) that are of primary interest to the analyst. It is only such weighted totals that are comparable with exogenous data sources, e.g. with an estimate of the total paid employment income in Canada.

There are however, two exceptions to this general rule of displaying tables whose entries are weighted by the relevant record weights.

- A particular run of the SPSM might not process all of the SPSD records satisfying the selection criterion. This might occur because a) the analyst was intentionally using a sub-sample of the SPSD, e.g. the 5% sample because only rough approximations were desired, or b) because the analyst interrupted the run before the sample was completely processed.

In both of these instances the SPSM carries out a set of appropriate "normalizations" on the records that it has processed in order to "blow them up" to a Canada total (subject to the record selection criterion). Thus, for example, if the analyst interrupted a run one-third of the way through the 5% SPSD sample, then the tables would be weighted using weights sixty times those actually attached to the records. Here, the factor of 60 is derived as the inverse of (0.05)*(1/3).

- The second exception, described in greater detail below when the concept of a tabulation variable is addressed, occurs for certain variables specifically intended not to be weighted. These "special" tabulation variables are used when the analyst wants to count the records themselves rather than a more typical tabulation variable such as families, earnings or taxes.

  The special variables (spsdrecs, scfrecs and fxvrecs) are relevant for applications such as ascertaining the number of SPSD records on which various averages are based, e.g. as one means of judging the reliability of the measure. Thus, when the X-tab facility creates tabulations of these special variables, it does not apply the record weights. However, the interpretation of tabulations involving these special variables will still be dependent on the analyst's choice of the level of roll-up for the tables.

The Table Set Specification

This section explains the SPSM's use of table set specifications. It discusses the specification's status as a control file parameter (XTSPEC), the extent to which the specification contains one or more individual table requests, the syntax of the table set specification, and other control parameters that affect the functioning of the table set specification.

Control File Parameter

The table set specification parameter is one of many parameters in an SPSM control file. (For a full description of SPSM control files, which are characterized by their .cpr file extensions, see the *SPSD/M User's Guide*.) The table set specification is given to the SPSM via the XTSPEC parameter. This XTSPEC parameter takes the form of a text string, frequently a rather long one. Because the string may be long, the SPSM permits it to contain carriage returns. Thus, the parameter may occupy multiple lines in the control parameter file.

Normally, the user will create or alter the XTSPEC parameter by using a text editor (such as BRIEF) when creating or editing the appropriate control parameter file. Although the analyst can enter an XTSPEC parameter "on-the-fly" at the beginning of an SPSM run, this would typically be done only by experienced analysts and for entering specifications that are not particularly long or complicated. The analyst might also wish to use the SPSM's "on-the-fly" facility to correct syntax errors in an existing XTSPEC parameter.

The Container of Table Requests

Two points are relevant here. First, XTSPEC is the unique means by which the analyst can request that the SPSM produce customized tabulations during a run. The SPSM offers a variety of standard tables and data export facilities, but only the XTSPEC parameter permits the analyst to perform customized cross-tabulations within the SPSM.

Second, the table set specification is primarily a "container" for the analyst's individual table requests. Such table requests, the topic of the next section, are "packages of information" that request the SPSM to produce the individual tables desired. Indeed the only information contained in the `XTSPEC` parameter is a series of such table requests, i.e. `XTSPEC` is empty except for such table requests. The number of requests can be zero, i.e. `XTSPEC` itself can be empty if the analyst does not wish to make use of the X-tab facility; however, our discussion here will proceed as if `XTSPEC` contains at least one table request.

Syntax -- Content

In an SPSM control parameter file, an empty table set specification consists of the keyword XTSPEC followed by a carriage return (after which the next parameter in the control file begins). A non-empty table set specification consists of the keyword XTSPEC, followed by

one or more spaces or tab characters, followed by a semi-colum. Semi-colon characters (";") serve to separate the individual table requests. The final carriage return serves to terminate the XTSPEC parameter definition.

The definition and internal structure of the component table requests is clearly important and will be developed in more detail in the next section. Appendix B to this guide provides a formal definition. At this more general level of description, treating the general structure of XTSPEC, it is sufficient to consider table requests simply as "packages", each of which provides all of the information necessary for the SPSM to produce "one" table.

Syntax -- Appearance

One advantage of the ability to split an XTSPEC across multiple lines lies in the capacity to enter longer tabulation specifications. Another is to improve the readability of individual table requests by putting their components on separate lines, surrounded by appropriate "whitespace". This capacity is developed more fully in the discussion of the table requests.

Invalid Strings

Because the table set specification, coupled with its component table requests, is potentially fairly complex, even experienced users may enter one incorrectly. If the error is strictly one of content, (i.e. the analyst has entered a legitimate specification, even if it isn't very meaningful or isn't what was intended,) the X-tab facility will, of course, simply generate the requested tables. The analyst will be forced to look at the resulting table(s) to discover that there was a specification error.

However, if the analyst's error is syntactical, such that the X-tab facility cannot "understand" the specification, things are different. The SPSM will discover this syntactical type of error while reading the control parameter file when it tries to interpret the meaning of the XTSPEC string. The presence of this kind of error will be immediately apparent to the analyst because the SPSM will generate an appropriate error message, declining to continue with the run until an acceptable parameter has been entered.

Generally speaking, the error messages have been designed to be very informative as to the nature of the problem detected. Many of them are written so that, when invoked, they make specific reference to the content of the analyst's specific parameter; in this manner the analyst knows immediately what part of the parameter is causing the problem. Appendix A in this guide provides a listing of the major error messages that might occur. The individual entries also indicate what the analyst must do in the way of corrective action. Also relevant for the interpretation and correction of XTSPEC errors is the message within "" which describes the formal syntax that the SPSM expects.

If the error is a simple one and the nature of the correction is obvious, the analyst can correct the XTSPEC parameter "on-the-fly" using the parameter editing facilities built into the SPSM. On a multi-line request the use of CTRL-X permits to move to the next line. Fuller

details on the parameter editor may be found in the *SPSD/M User's Guide*. For more complicated errors, the analyst may choose to abandon the run, and employ a more powerful text editor to modify the XTSPEC parameter specification.

Related Parameters

Although the XTSPEC parameter is the major, and most complex, of the control file parameters controlling the X-tab facility and governing its output, it is not the only one. Three other control parameters affect the way in which XTSPEC works.

XTFLAG

The XTFLAG parameter is a flag or switch that tells the SPSM whether or not to produce the tables specified by XTSPEC. If XTFLAG is "off", i.e. has a value of zero, then the SPSM will not produce the tables defined in the XTSPEC parameter.

Consider the following example of the use of XTFLAG. An analyst has already developed an extensive set of analytic tables, having entered the associated XTSPEC in the control parameter file. Now the analyst wishes to make several runs of the model to investigate a variety of policy options, seeking to find a policy parameter configuration that will meet some policy objective, e.g. not requiring any additional federal expenditure. However, the detailed distribution tables are not useful to the analyst until the desired policy parameter configuration has been discovered. Consequently, to speed up the processing of the model, the analyst leaves XTSPEC unchanged, but "turns off its execution" during the sensitivity testing by setting XTFLAG to zero for those runs. Once the desired configuration is known, the analyst makes an additional run, with XTFLAG again set "on", and receives the desired tables.

XTDBLFLAG

SPSM variables are stored and manipulated using single precision arithmetic, which provides about 6.5 digits of accuracy, which is more than sufficient for tax/transfer calculations. By default, the X-tab facility also uses single precision arithmetic when cumulating tables. A new parameter, XTDBLFLAG, will instead cause the X-tab facility to use double precision arithmetic when cumulating tables. This causes SPSM to use more memory and run slower, but results in increased accuracy in certain kinds of tabulations.

XTLINES

The XTLINES parameter allows the user to control the appearance of the X-tab facility output with respect to the inclusion of multiple table segments on a single page Numerically the parameter has the interpretation of the number of lines on a logical page. Generally speaking, a small value will force a new page break (and the printing of a page heading) for each table segment, while a large value will induce the X-tab facility to print several table segments before issuing a page break and printing a new page heading. When the value of XTLINES permits the X-tab facility to place multiple table segments on a single logical page, the software prints three blank lines between adjacent segments to separate them.

The XTLINES parameter must be present in the control parameter file. The analyst should ensure that XTLINES takes on a value in the range 0 to 32767. A typical value for it is 66, this value corresponding to the 66 lines that will fit on an eleven-inch page at the standard six lines per inch. The analyst might elect to use a very large value for XTLINES in order to suppress page breaks that could make more difficult the exporting of the output file to a spreadsheet package. If the analyst desires to export the tables to a spreadsheet package, however, he or she should consider using the import utility, which has been designed specifically for this task. It is documented in the *SPSD/M Tools User's Guide*.

XTCOLS

The XTCOLS parameter allows the user to exercise some control over the appearance of X-tab facility output by altering the tradeoff between having multiple lines in column headings and having greater column width.

The XTCOLS parameter must be present in the control parameter file. The analyst should ensure that XTCOLS takes on a value in the range 80 to 32767. The general interpretation of the parameter's value is that of a desired maximum number of print columns for the output file. A typical value of XTCOLS is 132 corresponding to the maximum number of print positions available on many printers.

Even though the X-tab facility will never (horizontally) break a wide table into multiple segments, it will attempt, if possible, to shrink the widths of individual columns so that the whole table falls within the XTCOLS value. Given the constraints imposed by the numbers that must be printed in the table and the aesthetics involved in folding the text of column labels, the X-tab facility will not always be able to keep the table within a width of XTCOLS; thus, the X-tab facility treats the XTCOLS value more as an objective than an absolute constraint.

The specific algorithm used by the X-tab facility to choose column widths depends on sufficiently many factors that no simple description is feasible. Generally speaking, the user should enter a large value when the width of the output device is not a problem and it is desired to avoid folding column labels. In contrast, when the width of the output device is at issue and folded column labels are tolerable, then a smaller value is appropriate.

At the general level of the XTSPEC parameter as a whole, our description is now complete, characterizing XTSPEC as a container for the individual table requests that do all the real work. The boundaries between these component table requests are marked by the presence of the semicolon separating character. In the next section we turn our attention to the component requests, describing their content and internal structure in a level of detail appropriate to an analyst communicating his or her own tabulation requirements.

The Table Request

This section describes the SPSM's use of table requests. Recall the XTSPEC parameter for the introductory example:

```
XTSPEC
CF: {units}
   * hdprov+
   * cftype+;
```

The table request can also be written on a single line:

```
CF: {units} * hdprov+ * cftype+;
```

All of the critical parts of a table request were present: a unit of analysis (census families), a tabulated variable (units), and classificatory variables (hdprov and cftype). In this section we undertake a more complete treatment of table requests, including a number of features (use of expressions, normalizations, labelling, etc.) not previously introduced.

Broad Structure

The broad structure of a table request is as follows: it begins with an (optional) unit of analysis specification, followed by one or more "levels", exactly one level of which must be a specification of the table's tabulation variable(s). The term tabulated "variable" is used here for simplicity; however, the SPSM explicitly recognizes either an explicit variable or an expression. Asterisks serve to separate the request's levels, much as the semi-colons serve to separate the requests themselves.

Levels of Tabulation and Their Order

With fairly minor (and somewhat obvious) exceptions noted later, the levels, left to right, in a table request are deemed to be in "descending" order. Levels coming earlier in the table request are "higher", and those that come later are "lower". For a level to be "lower", in this context means that the table cycles through its categories more frequently. Thus, the "lowest" level is the right-most or last level in the request. That lowest level controls the column categories for the table, those categories cycling once within each row. The "next-to-lowest" level comes next to last; it controls the row categories, which cycle once within each table segment. The next level up controls the segment categories, which cycle only across segments, and so on. **The X-tab facility permits up to 6 levels combining into less than 260 classification variables** in a table request, more than sufficient when one considers the reduced data density that inevitably results from imposing progressively finer categorizations on the data.

Units of Analysis

The SPSM and the X-tab facility support five kinds of units of analysis. These five types form a hierarchy, with a higher unit necessarily containing one or more instances of all lower

levels of units of analysis. From the highest level to the lowest, the five supported units of analysis are as follows:

HH (Household)

The household (e.g. single family dwelling, townhouse, apartment unit, etc.) is the basic unit of the Survey of Consumer Finances, the unit on which the SPSD is essentially based. The definition of household is keyed to dwelling location, and does not consider the interrelationships of its members beyond the fact that they live in the same unit.

EF (Economic Family)

An economic family, which may be identical to a household, consists of a person, or of multiple persons related by blood, marriage (including common-law marriage) or adoption, living together as an economic unit within a single household. Although a single household may contain multiple economic families, every household necessarily includes exactly one primary economic family.

CF (Census Family)

A census family in the SPSD/M, which may be identical to an economic family, consists of a person, the person's spouse if present (including a common-law spouse), and any of their never-married children living in the economic family. Given its policy focus, the SPSD/M treats an unattached individual as if it were a true census "family"; thus, an analyst can make a table request for census families without needing to make a separate parallel request to tabulate individuals. Although an economic family may contain multiple census families, every economic family necessarily includes exactly one primary census family.

NF (Nuclear Family)

A nuclear family in the SPSD/M is very similar to the census family, except that nuclear families are defined to exclude any never-married children aged 18 or older. Consistent with emerging practice in taxation that treats 18 year old individuals as adults, any such children in the household are deemed to constitute their own nuclear families. Although a census family may include multiple nuclear families, each census family necessarily includes exactly one primary nuclear family; the head of the census family is necessarily the head of that primary nuclear family.

IN (Individual or Person)

The concept of the individual or person is obvious. The SPSD/M contains substantial information for each person aged 15 years or older. The existence of younger children is recognized in the SPSD/M by means of including the most basic information about them

(e.g. age, sex and school status) in their own individual record. Obviously, however, such records will be limited in the other information they contain. For example, the variable for self-employment income will be present, but will invariably have the value zero. Each census family or nuclear family includes one person designated as the head of the family. In the SPSD that is based on the 1985 (1984 incomes) SCF, when there are two parents in the relevant census or nuclear family, the male is arbitrarily designated as the head of the family.

The first item in a table request, normally beginning in the first column position on a line, is the unit of analysis. This specification consists of the two-character code for the desired unit (`HH`, `EF`, `CF`, `NF`, or `IN`) followed by a colon that separates the unit specification from the remainder of the table request. The user must enter the two-character code in capitals. As noted above, the unit specification is optional. If the user omits it (also omitting the colon), then the X-tab facility uses individuals,"`IN:`", as the default specification.

Tabulation Variables

In order for a table request to be meaningful, it must indicate just what is to be tabulated, i.e. summed or counted, in the table. Thus, (exactly) one of the levels in the table request must identify the tabulation variable or variables.

The Tabulation Variable Level of a Table Request

In the X-tab facility, a table request initiates the tabulation of one or more tabulation variables in terms of categories defined by other, classificatory, variables. Consequently, a table request must include exactly one level designating these tabulation variable(s). The analyst distinguishes this unique level by enclosing it in "curly braces", i.e. the { and } characters. As described below, the tabulation variable level is "special", being different in both form and content from the other levels that designate classificatory variables.

Generally speaking, a tabulation level's contents, i.e. what lies inside the braces, consists of "appropriate" variables or expressions. "Appropriate" in this application refers to database and model analysis variables, or to expressions that derive from these database and model variables and that also evaluate to analysis values. The possible sources of such values are enumerated below. Each of the tabulation variables/expressions in the tabulation level is further subject to explicit "qualifiers"; these qualifiers are described below.

Multiple Tabulation Variables in One Request

The X-tab facility permits the analyst to include multiple tabulated variables in a single table request. Syntactically, the multiple inclusion is straightforward. The analyst puts the desired variables or expressions within the tabulation level and separates the component variables or expressions by commas. Thus, for example, to tabulate both the child tax credit and GIS benefits in the same table, the analyst would use a tabulation expression such as

```
{ imctc, imigis };
```

or

```
{ imctc,
  imigis };
```

When the analyst is using multiple tabulation variables or expressions in the tabulation level, the list of relevant variables and expressions acts much like the categories of a classificatory level. For example, if a multi-variable tabulation level is the lowest level in the request, its component variables and expressions serve to define the columns for the table. If it is the next to lowest level, the multiple tabulation variables and expressions define the table's row categories, and so on.

Sources of Tabulation Variables

Only certain kinds of variables and expressions are permissible as tabulation "variables" in the X-tab facility. Fortunately, this set includes most of those of interest to the analyst. The four recognized sources for tabulation "variables" are as follows:

SPSD (database) analysis variables:

The analyst can specify, as variables to be tabulated, analysis variables available from the SPSD. The mechanism for this is the obvious one; the analyst simply uses the variable's name in the table request. Thus, for example, using the analysis variable idiemp as the tabulated variable would create a table that contained cells corresponding to (paid) employment income. Some of these variables will be defined at the level of the individual person (the id variables), while others will be defined at the level of the household (the hd and fx variables). Other "derived" variables will exist in the database at intervening levels, e.g. cf for census family variables, or nf variables for nuclear families.

Database variables are documented in the *SPSD/M Variable Guide*. Clearly, not all of them will be suitable as tabulation variables in a table request. For example, the hdwgthh variable (that provides the household weight) would be an illogical choice as a tabulation variable (since it would then be weighted by itself during the tabulation process). The important issue of "appropriate" variables, and their treatment in rolling up a table to the several levels of unit of analysis are treated in considerable detail somewhat later in this guide. Generally, however, the X-tab facility will allow the analyst to tabulate any analysis database variable that the analyst is likely to consider reasonable as part of a policy analysis.

SPSM (model) analysis variables:

It is the essence of the SPSM, in both its "black box" and "glass box" versions, that it derives variables of interest to the analyst. Typically, but not exclusively, these derived variables are the taxes and benefits relevant for the units. Just as with the analysis variables available directly from the database, the analyst can tabulate appropriate model analysis variables simply by using their names in a table request. For example, using imctc as the tabulation variable would generate a table that accumulates amounts of child tax credit payable.

Model variables available from the "black box" version of the SPSM are also documented in

the *SPSD/M Variable Guide*. As with the database variables, not all of them will be suitable as tabulation variables in a table request. Generally, however, the X-tab facility will allow the analyst to tabulate any analysis model variable that the analyst is likely to consider reasonable as part of a policy analysis.

Table request analysis expressions:

The X-tab facility's further ability to tabulate "arbitrary" expressions provided by the analyst (expressed in terms of database, model, and/or ex variables) is a great time saver and simplifier. Its existence sometimes means that the analyst is not forced to (a) go back into the model, (b) modify the model's source code to create the desired variable, then (c) re-compile and re-link the model, and (d) re-run the model requesting the desired table, all in order to tabulate something that is easily specified in terms of existing variables. The qualifier "sometimes" reflects the dependence of these expressions on the level of roll-up; i.e. the table entries are calculated as expressions of aggregates rather than aggregates of expressions.

For some expressions, e.g. purely additive ones, the distinction is not important, but for other forms, the interpretation of the table will depend critically on the distinction.

The X-tab facility provides considerable flexibility in the definition of these "on-the-fly" expressions. Essentially, a tabulation expression can be any "appropriate" C expression. "Appropriate" in this context means (analysis) expressions involving those unary and binary operators that are either arithmetic (`+`, `-`, `*`, `/`), or logical (`<`, `<=`, `==`, `>=`, `>`, `!=`, `!`, `&&`, and `||`). Logical operators, although allowed syntactically in a tabulation expression, are rarely used since variables in a tabulation expression refer to aggregated quantities.

The analyst may further employ parentheses for grouping as required. A restriction is that the expression may use, in its arguments, only analysis variables; this restriction is, however, not particularly onerous because virtually all of the variables the analyst might wish to use as arguments in an expression are "already" defined as analysis variables.

The X-tab facility would create any given cell entry for an expression by first accumulating, across cases falling into the cell and at the table- defined level of unit of analysis, a value for each of the symbolic arguments in the expression. To arrive at the numeric value that will appear in the cell, the X-tab facility then evaluates the expression, plugging into it the various accumulated values.

Thus, for example, a tabulated expression of the form {imctc/immtot} would produce the ratio of total child tax credit benefits (for units falling into the cell) to the total income for those units. It would not produce the sum of the imctc/immtot fractions across those units, this requires the creation of a user variable. Note that this X-tab facility interpretation of the expression is precisely the one that the analyst will usually want.

Special Tabulation Variables

In addition to the "usual" tabulation variables described above, the X-tab facility permits the

analyst to specify certain tabulation variables (or expressions involving them) that are "special" in the sense of receiving a non-standard treatment.

Units and *persons*:

Even though the variables *units* does not appear as a database or model variable, analysts are sufficiently likely to want to tabulate the (weighted) numbers of the various units of analysis that units has been included as a variable permissible in any tabulation variable or expression. When used, it designates the unit of analysis specified at the beginning of the table request. Thus, if the analyst tabulates units when the level of unit of analysis is the census family, the result will be interpreted as a count of the number of (weighted) census families.

Similarly, the analyst will often wish to tabulate the numbers of persons (e.g. for counting the numbers of persons living in units below the Statistics Canada low income cutoffs) or to use the number of persons in expressions to be tabulated (e.g. for assessing per capita income in a census family). For this purpose the variable *persons* is made available. When used, it designates the number of persons in the unit of analysis specified at the beginning of the table request. Thus, for an appropriate selection specification, tabulating the expression `{immdisp/persons}` will result in cell entries properly interpreted as "per capita disposable income".

Variables not subject to weighting:

Three of the SPSD variables, spsdrecs, scfrecs and fxrecs, are exceptions to the rule that tabulated variables are weighted during the tabulation process. All three of these variables have been included in the database explicitly in order to let the analyst ascertain the numbers of cases underlying a given table entry, especially where data density is of concern in the interpretation of the tabulation results.

When interpreting tables that use any of these variables, the analyst must recognize that the cell entries are sensitive to the level of roll-up for the table. Thus, for example, there are many more records relevant when the tabulation unit is individuals than when it is economic families.

spsdrecs: Used in the absence of an "`M`" qualifier, the spsdrecs tabulation variable permits the analyst to tabulate the number of SPSD records on which a cell or total is based. Such a count is based on the number of households occurring in the SPSD.

scfrecs: Used in the absence of an "`M`" qualifier, the *scfrecs* tabulation variable permits the analyst to tabulate the number of original SCF records on which a cell or total is based. Such a count is based on the number of household records in the Survey of Consumer Finances that forms the basis for the SPSD. The smaller number of SCF records, as compared to SPSD records, reflects the cloning, during the SPSD development, of some SCF records to provide a greater richness of information for some units having higher incomes or UI benefits.

fxvrecs: Used in the absence of an "`M`" qualifier, the *fxvrecs* tabulation variable permits the

analyst to tabulate the number of original FAMEX records on which a cell or total is based. Such a count is based on the spending unit records in the FAMEX database that supplies the expenditure variables for the SPSD records. Clearly, because the FAMEX database is so much smaller than the SCF, a single FAMEX record may have had to be used to supply expenditure patterns for multiple SCF/SPSD records.

The "M" qualifier referred to in the three preceding paragraphs requests that the X-tab facility perform a normalization along one of the classificatory dimensions in the table. Thus, if the "M" qualifier is present, the table would show not the numbers of cases in the cells, but their percentage distribution. The several qualifiers and their roles in customizing tabulations are taken up in the next section.

Tabulation variable qualifiers

The X-tab facility provides the analyst with the capacity to optionally "qualify" a tabulation variable in several ways that may enhance the readability or utility of the resulting table. Using these qualifiers, the analyst may, for example, provide more informative labeling, force a scaling of the cell entries, or initiate normalization. These kinds of enhancements involved are relevant only for tabulation variables; one would not expect, for example, to scale or normalize a classificatory variable.

Optional Nature: The use of qualifiers is completely optional. The X-tab facility provides defaults for all of the features that the qualifiers control, with some of these defaults conditional on the characteristics of the requested table (e.g. the nature of the variable or expression being tabulated). In most cases the X-tab facility produces very reasonable looking tables even if no qualifiers are used. However, the analyst may wish to use qualifiers to initiate certain characteristics (labeling, scaling, marginals, etc.) that make the table more useful.

Appearance: An individual qualifier looks a bit like an assignment statement. It consists of an upper case letter (`L, S, P, or M`) indicating the particular qualifier being invoked, immediately followed (i.e. with no intervening whitespace) by an equals character (`=`), immediately followed by the string or numeric value to be assigned to the qualifier. Thus, for example, the qualifier `S=6` indicates that the tabulated variable is to appear in the table in units of one million.

Syntax/placement: The analyst indicates the presence of qualifiers for a tabulated variable by placing a colon after the tabulated variable or expression. The qualifier(s) then follow the colon and precede (1) the comma that separates the tabulation variable from subsequent additional tabulation variables or (2) the curly brace that marks the end of the tabulation level portion of the table request.

If multiple qualifiers are present, they are separated from each other and from other material by appropriate "whitespace", (spaces, carriage returns and tabs). Thus, for example, a tabulation level of `{ units : S=6 P=1 }` indicates that the X-tab facility is to count the weighted units of analysis, and to present them in terms of millions, with one digit to the

right of the decimal point. A table entry of `6.1` would thus indicate the value 6,100,000.

Each of the tabulation variables or expressions in the tabulation level of a table request may have its own set of qualifiers.

The four possible qualifiers (`L`, `S`, `P` and `M`) are described individually below.

1. `L`: Label -- The L qualifier permits the analyst to specify, for the output table, a textual label for the tabulated variable. This label takes the form of a text string delimited by double quotes. The text string, used for labeling the tables, may be up to 40 characters long. If the X-tab facility is using the string in a column heading, and the string is too wide for the column, the X-tab facility will break up the string intelligently, choosing cutting points between words if that is possible. When the string is used as a row heading, it is never broken; instead, the whole table body is shifted right to accommodate the heading.

   The `L` qualifier, when present, instructs the X-tab facility to override the default label that would have obtained had there been no qualifier. By default, the X-tab facility will use the database or model label, if one exists. If no such default label exists, then the X-tab facility uses the name of the tabulated variable, or the text string that defines the tabulated expression. The `S` and `M` qualifiers also affect, as described below, the text string that will be used as a label.

   As an example, a tabulation level of the form

   ```
   {immdisp/persons: L="Per Capita Disp. Inc."};
   ```

   would ensure that the tabulated expression would be labeled with the text string "`Per Capita Disp. Inc.`" rather than the more cryptic default string "`immdisp/persons`".

2. `S` Scale factor -- The `S` qualifier permits the analyst to specify a scale factor for the presentation of the tabulation variable or expression. This scale factor takes the form of an integer exponent; e.g. `S=3` indicates that the analyst wishes to see the results presented in thousands. The analyst should avoid scale factors falling outside the range [-6, 9].

   The "default" is 6, since most tabulation will involve summing dollar amounts. However, if the analyst is tabulating *units* or *persons* and there is no use of the divide (/) operator, then the default becomes 3. The use of the divide operator or the tabulation of any of the three special variables (*spsdrecs*, *scfrecs* and *fxvrecs*) will set the default to 0 (no scaling). In addition, the use of the `M` qualifier will override the default scale factor, setting it to -2 to reflect the typical presentation of marginals in terms of percentages. An explicit specification of an `S` qualifier will, of course, override these defaults.

   The scale factor, in addition to affecting the numbers that appear in the cells of the resulting table, naturally also affects the labels that apply to the tabulation variable or expression. That is, the labeling for the table must indicate when a scale factor has been applied and, if it has, which specific factor was used. This impact on labels occurs whether the scale factor was specified explicitly via an S qualifier or selected as the default value by the X-tab facility.

Generally, if a non-zero factor is being applied, the text of the label will receive a suffix to indicate the specific numeric-scaling factor being used, e.g. appending (000) for a scale factor of 3. However, for certain commonly used values, other suffixes are used for readability. These specific substitutions are "(%)" when `S=-2`, `"(M)"` when `S=6` and `"(B)"` when `S=9`.

To take a concrete example, a tabulation level of `{idiint:S=9}` would tabulate interest income and present the results in terms of billions of dollars. Presumably this specification would be used only in a table with a relatively small number of cells so that most entries would exhibit scaled values greater than 1.0.

3. `P` Places -- The P qualifier permits the analyst to control the presence of a decimal point in table entries and the number of digits appearing to the right of any decimal point. `P` must be equal to 0 (zero) or some positive integer. A zero value for `P` indicates that no decimal point will appear for the tabulated variable; of course, in the absence of a decimal point there can be no trailing digits. A positive value of `P` indicates the number of digits to follow the decimal point. The analyst should restrict the value of `P` to the domain [0, 8].

   As is the case with the `S` qualifier, the `P` qualifier has "intelligent defaults". The "default" is 1, but the X-tab facility will override this value depending on the nature of the tabulation variable or expression. Specifically, if the analyst uses the divide operator, and does not employ *units* or *persons* in the tabulation expression, then the X-tab facility assumes that the analyst is creating a ratio and sets `P` to `4`, selecting this as a reasonable value for most fractions. However, (1) if the analyst uses the divide operator and tabulates *units* or *persons*, or (2) if the analyst tabulates *spsdrecs*, *scfrecs* or *fxvrecs*, the default `P` is set to `0` (zero). As well, if the analyst uses the `M` qualifier, the X-tab facility sets the default `P` value to `1`. An explicit specification of the `P` qualifier will, of course, override any of these defaults.

4. `M` Margin -- The `M` qualifier permits the analyst to normalize the tabulated variable or expression with respect to (exactly) one of the classificatory variables appearing in the table request. Such normalization is appropriate when the analyst is concerned about percentage distributions or the relative sizes of variables; automated generation of the marginals within the SPSM is considerably more convenient than alternative mechanisms such as hand calculations.

   The normalization must be specified in terms of exactly one of the classificatory variables defining a level in the table request. The analyst's specification of this classificatory variable defines for the X-tab facility the dimension along which the normalization is to be carried out. The X-tab facility does not permit multiple normalizations on a single tabulation variable. Thus, for example, the analyst could generate either row or column percentages in a given table, but not both. However, to carry this example to its conclusion, one could use a tabulation request such as --

```
CF:{units,
    units:M=cftype,
    units:M=hdprov}
```

```
* cftype+
* hdprov+;
```

to generate three parallel table segments that collectively present the numbers of census family units and the relevant row and column percentages.

Similarly, at the present time there does not exist any straight forward way to generate "segment percentages" for a two dimensional table, e.g. to display the relative contributions of combinations of family size and size of place of residence to the population of poor families. (The analyst could of course generate a new composite classificatory variable that has one category for each combination, but that mechanism is not particularly simple for the user.)

A typical use of the qualifier is the generation of row or column percentages in a table. For example, if the analyst is tabulating *mdctc* and employs an `M` qualifier of `M=cftype`, then the resulting table will show the percentages of child tax credit received by the several family types.

The mechanics of the normalization are relatively simple, and consist primarily of calculating a ratio. The numerator of the ratio is the value of the cell entry as it would have appeared had there been no normalization. The denominator is the value of the entry that would occur for the "all categories of the normalization classificatory variable lumped together" entry as it would have appeared had there been no normalization (were it to be computed). Thus, when the "ALL" entry for the normalization classificatory variable is actually requested in a table, (via the + suffix described below) its normalized value will always be equal to 1.0.

As described above, the use of the `M` qualifier affects the defaults for the other table features. Because analysts most often wish to see normalizations in percentage terms, the use of `M` sets the default (scale) factor `S` to `-2`. In addition, the use of the `M` qualifier establishes a default value of 1 for the `P` (number of decimal places) factor. Recall that an analyst unhappy with these choices can easily override them by supplying explicit `S` and `P` qualifiers to the relevant tabulated variable or expression.

The `M` qualifier further affects the label used for the tabulation variable because readers of the resulting table must understand both that a normalization has occurred and which classificatory variable was used for the normalization. The X-tab facility meets these needs by placing the name of the normalization variable in square brackets at the end of the text string that labels the tabulation variable (after any suffix relating to the scale factor).

Cell Content and Labeling for Tabulation Variables and Expressions

With the discussion of qualifiers complete, it is now feasible to address the fundamental question of the relationship between the possibly qualified tabulation variable and the contents of a cell in the resulting table. Four distinct cases are relevant:

a) Simple Variables with No M Qualifier:

This will be the single most common case. The analyst wishes to tabulate a single variable, rather than an expression that may involve other variables or constants. The cell entry will simply be the sum (typically weighted by the record weights) of the value of that analysis variable over all of the records meeting the criteria for inclusion in the cell. For example, if *immicons* is the tabulation variable, then the cell entry will correspond to total consumable income.

When the analyst has not used an `L` qualifier, the labeling will typically consist of the label for the variable, if one exists. If it does not, then the default label is the name of the variable. For scale factors other than zero, the label will include a suffix that reflects the factor applied to the entry.

b) Simple Variables with an M Qualifier:

When the analyst tabulates a single variable (rather than an expression), but uses an `M` qualifier, the cell entry is no longer the simple sum of the variable, but its percentage distribution over the dimension of the classificatory variable given in the qualifier. Thus, for example, if the analyst uses the *immicons* variable, but qualifies it with `M=hdprov`, then the cell entries will be the percentages that the individual provinces contribute to the consumable income.

When the analyst has not used an `L` qualifier, the labeling will typically consist of the label for the variable, if one exists. If it does not, then the default label is the name of the variable. If the scale factor is not equal to zero, then the label will include a suffix that reflects the factor applied to the entry. In all cases the label will conclude with a suffix consisting of the name, enclosed in square brackets, of the classificatory variable used for the `M` qualifier.

c) Expressions with No M Qualifier:

When the analyst tabulates an expression (as opposed to a database or model variable), but does not use an `M` qualifier, the X-tab facility accumulates the relevant sums for all of the variables in the expression, and then generates the cell entry by computing the result of the expression with the sums as arguments. It does not sum up the results of the expression as computed for each of the individual weighted records. Note that this is exactly what the analyst is likely to want. For example, if average income is desired, then the analyst wants to divide total income by total units, exactly what the X-tab facility does. In contrast, the analyst is very probably not interested in seeing the sum of the individual ratios for the relevant units.

When the analyst has not used an `L` qualifier, the labeling will consist of the text string used to specify the expression. The use of an `L` qualifier will, of course, override that text string. If the scale factor relevant for the entry is non-zero, the label will also include a suffix that reflects the factor applied to the entry.

d) Expressions with an M Qualifier:

The most complicated situation occurs when the analyst tabulates an expression, but qualifies it using the M qualifier. In this case, the X-tab facility in essence calculates the cell entry using the "expression of sums" approach, as in c), but then, before putting it in the table, applies the "normalization over the qualifier variable" approach, as in b).

The X-tab facility's ability to combine the flexibility of expressions with a normalization can prove quite powerful. For example, suppose that the user's expression tabulates the proportion of economic families whose income falls below the LICO. If the user invokes the M qualifier across family type, then the resulting table displays the relative incidences of such low income status across family type, normalized to the overall incidence.

When the analyst has not used an L qualifier, the labeling will consist of the text string used to specify the expression to be tabulated. The use of an explicit L qualifier overrides this default. In addition, if the scale factor is other than zero, then the label will include a suffix that reflects the factor applied to the entry. In all cases the label will conclude with a suffix consisting of the name, enclosed in square brackets, of the classificatory variable used in the M qualifier.

Classificatory Variables

Just as the tabulation variable level controls the content of what is being tabulated, the classificatory variable level(s) tell the X-tab facility how that content is to be broken out, e.g. the shape of the table in terms of rows, columns and segments. This section describes the nature of such classificatory levels, their sources (as regards the classificatory variables that comprise them) and the manner in which they affect the tables produced.

Nature of Classificatory Variables

Classificatory variables describe to the X-tab facility any given case's status among a set of mutually exclusive and exhaustive categories, e.g. province of residence or sex of individual. Thus, these variables are sometimes termed "categorical". Such variables make sure that any given record falls into exactly one nominal category. Since the number of categories must be relatively small to be useful in a table, the values of such variables tend to be "small non-negative integers." This "nominal, integer" status for classificatory variables stands in sharp contrast to the analysis status required of tabulation variables. Consequently, the X-tab facility imposes certain requirements on the variables that it is willing to accept as classificatory for purposes of table construction. The allowable sources for such classificatory variables are described a bit later in this section.

Contribution to Table Dimensionality

Generally speaking, it is the classificatory variables that determine the shape of the output table, with each classificatory variable adding one dimension to the table. (Recall that the use of multiple tabulated variables or expressions contributes yet another dimension to the table.) Thus, the categories of the classificatory variables will typically correspond to the

rows or columns or segments of a table.

Nature of Classificatory Levels

Classificatory levels in a table request are optional. Although a table request normally includes one or more such levels, no classificatory levels need be present. However, each classificatory level that does exist must consist of a single classificatory variable.

This SPSM requirement of one exactly one classificatory variable per level stands in contrast to some statistical packages such as SAS, TPL, or SPSS where the analyst can request parallel tables in a single table request (e.g. separate tables showing income as a function of province, family type, poverty status, and labour force attachment). In the SPSM, the analyst would generate such parallel tables via parallel table requests that had the same tabulated variable (income), but different classificatory variables (province, etc.).

Sources of Classificatory Variables

Just as the SPSM imposes certain restrictions on variables to be used as tabulation variables and accepts them only from certain sources, it will accept as classificatory variables only certain variables from certain sources. (Similar restrictions apply in the tabulation facilities of other packages such as SAS or SPSS.) In the SPSM, these sources of classificatory variables are as follows:

Database classificatory variables:

Certain variables in the SPSD are inherently classificatory, e.g. province (variable *hdprov*) or the coded variable for the size of place of residence (variable *hdurb*). The analyst can use any of these variables simply by entering the variable's name as a level in the table request. A list of available database classificatory variables is available as part of ""; fuller descriptions of the variables themselves, including their specific categories, appear in the *SPSD/M Variable Guide*.

Model classificatory variables:

Other classificatory variables are defined in the SPSM; e.g. the underlying model creates variables such as nominal tax filer status (variable *imfiler*) and nominal classification of families by categories relevant to the administration of GIS (variable *imgistyp*). The analyst can use any of these variables simply by entering the variable's name as a level in the table request. A list of the available model classificatory variables is available as part of ""; fuller descriptions of the variables themselves, including their specific categories, appear in the *SPSD/M Variable Guide*.

User defined classification variables:

The SPSD/M does not limit the analyst to choosing classificatory variables from those available in the database or model. In addition to those sources, the analyst can create new,

customized classificatory variables that will be acceptable to the X-tab facility. The analyst creates such new classificatory variables user defined UVAR variables in the control parameter (`.cpr`) file. In general, a user defined variable is defined in terms of the values of some "previously defined" integer or analysis variable whether it be a database variable or a modeled variable.

A description of the UVAR capacity is available in the *SPSD/M User's Guide*. Here it is sufficient to note that this capacity gives the analyst the option to specify how the continuum of an existing variable is to be subdivided (via a finite series of explicitly given numeric cutting point values) to create a set of mutually exclusive, exhaustive and non-overlapping domains that correspond to the classificatory variable's categories.

Creating a Summary "All" Category

At times the analyst will wish to request a table that includes totals for one or more of the classificatory variables. For example, even though a table might use province of residence as a classificatory variable, the analyst might also wish to see the corresponding results for the whole of Canada. The SPSM's X-tab facility permits the analyst to make this request by entering a `+` character immediately after the classificatory variable along which the "collapsing" is desired. Thus, a classificatory level of `hdprov+` would not only generate a break out of the tabulated variable across provinces, but also initiate an "All" (all Canada) category. In the event that the classificatory variable has only two categories, the X-tab facility uses "Both" in place of "All".

The facility for requesting the "All" category is both optional and general. The user can request it for any subset of the classificatory variables in a table request (including the extreme cases of "none of them" or "all of them"), and can use it whether or not an `M` qualifier is used for the tabulated variable.

Labeling of Classificatory Variables

The classificatory variables contribute two kinds of labels to X-tab facility tables. First, in common with other SPSD/M variables, classificatory variables have text strings that indicate their general nature, e.g. "Province" for the variable *hdprov*. These variable descriptions appear in the tables to identify the classificatory variables appearing as control variables for the table. For database and model classificatory variables, these labels are defined in the database or model, respectively. If no variable description is defined for a classificatory variable, then the X-tab facility default the label to the name of the classificatory variable.

The locations for these variable labels are straightforward. For example, the label for the row variable appears within the upper left corner of the table or table segment, while the column variable's variable description is printed just above the table or table segment. When a table stretches across multiple segments, then the label for a classificatory variable that is constant within a segment appears as part of the information that labels the segment as a whole.

Second, classificatory variables have labels that correspond to the individual values, value

ranges, or classes of the variable. These "class labels" normally make up a fairly large proportion of the labeling information in a table or table segment. For model and database variables, the class labels are pre-defined for the user, and will usually be textual, e.g. "Manitoba" as a class label for one particular value of the variable *hdprov*.

Level of Analysis

Each classificatory variable is originally defined at one of the five SPSD/M family levels of analysis: individual, nuclear family, census family, economic family, and household. For example, *hdprov* (province of residence) is defined at the household level, *cfnch* (number of children in census family) is defined at the census family level, and *idcfrh* (relation of individual to `head' of census family) is defined at the individual level. As indicated in Section 5.3, the first item of a table request (`IN:, NF:, CF:, EF:, HH:`) indicates a family level of analysis. This section describes what happens when the level of analysis of a table request differs from the level of analysis of one of the classificatory variables in the request.

If the level of analysis of a classificatory variable is `higher' than the level of analysis of a table request, the variable value is well defined and is that of the `higher' unit. For example, the table request

```
CF: hdprov * {units}
```

specifies counts of census families by province of residence. Even though *hdprov* is defined at the household level, no ambiguity arises, since the province of residence of a census family is the province of residence of the containing household. Consider, however, the following table request:

```
HH: cfnch * {units}
```

Counts of households have been requested, broken out by the number of children in census families. If a household contains only one census family, the meaning of *cfnch* is clear, but what if a household contains two census families, with differing numbers of children? In such a case SPSM designates one of the census families to be the `reference' census family in the household, and takes the value of cfnch to be that of the reference census family. The reference census family for the household is the first census family in the household.

To define in general what a `reference' family unit means, it is necessary to understand the ordered hierarchical nature of an SPSD household. An SPSD household contains an ordered set of economic families, each of which contains an ordered set of census families, each of which contains an ordered set of nuclear families, each of which consists of an ordered set of individuals. Individuals are ordered in nuclear families starting with the eldest person, followed by that person's spouse if present, followed by children in increasing order of age. Nuclear families are arranged in census families starting with the married couple or lone parent nuclear family if present, followed by `old' children in increasing order of age. Census families are arranged in economic families starting with the `primary' census family. Economic families are similarly ordered within households.

The `reference' individual or family unit is thus defined to be the first such unit at the higher level of analysis. For example, the reference nuclear family in an economic family is the first

nuclear family in the economic family, according to the ordering given in the previous paragraph.

If selection has been activated through the SELFLAG and SELSPEC control parameters, the definition of reference person or family requires further clarification: the 'reference' individual or family unit then becomes the first selected unit at the higher level of analysis. In the following artificial example, if the selection control parameters

```
SELFLAG      1
SELUNIT      0
SELSPEC      idiemp > 5000
```

are given, then only individuals with over $5,000 in employment income will be selected. The table request

```
CF: idsex * {units};
```

will then count census families, categorized by the sex of the first person in each census family unit to have over $5,000 of employment income. If there is no such person in a census family, the census family will not be counted in the tabulation.

Table Titles

Because tables are of very limited utility if their contents cannot be easily understood, the X-tab facility constructs an informative title for each table request. These table titles occur independently of the page headings that the SPSM places on all of its output (i.e. in addition to the information about SPSD/M version, run date, identification for the base and variant version used for the run, the selection criterion, if any, used for the run, the sub-sample used and the particular aging algorithm employed). The table or segment titles are best understood as consisting of one or two text strings that are conditional upon the contents of the table. From left to right, the contents of the lines of a title are as follows:

a) The first line of a table or segment title begins with the phrase "Table" followed by an integer giving the cardinal position of the table request in the XTSPEC parameter, followed by the letter "U" (to indicate that it is a user-defined table), followed by a colon. Thus, a title with the first portion of "Table 3U" indicates that the succeeding table is the third user-defined table processed in the run. If the table is one that continues across several segments because it has three or more dimensions, then the string "(cont.)" will appear just before the colon for all segments other than the first segment.

b) The first line of the title next indicates the "central" aspect of the table, i.e. what is being tabulated.

   If the analyst is using a tabulation level that includes only a single variable, then that variable's label appears (or the name of the variable if there is no label).

   If the analyst is using a single expression, then the text of the expression appears. Of course, if the analyst has used the L qualifier in the tabulation level, then the explicitly given label overrides these automatic defaults. Also relevant for these two cases, the

"what" identification will include an appropriate suffix, e.g. "(000)," if the scale factor is other than zero. If there is an `M` qualifier, then a suffix will indicate the classificatory variable over which the normalization is being carried out.

Finally, if the analyst uses a tabulation level that includes multiple variables or expressions, then the title includes the phrase "Selected Quantities." This mechanism serves to prevent titles that might otherwise become unwieldy in their attempts to be complete. Of course, the critical identifying information (labels or names, or scale factors or identity of the "marginal" variable) then appears elsewhere in the table or segment headings where it will better inform the reader.

c) The first line of the title continues with "for" followed by the unit of analysis for the table. For example, if the table request began with `"CF:"`, then this portion of the title would consist of the text string "for Census Families." This portion of the title is present whether or not the unit of analysis was defaulted in the table request.

d) The first line of the title concludes with the identification of any classificatory variables controlling the column, row, segment, etc. categories of the table. The classificatory variables are presented in descending order by level in the table request, with the highest level preceded by the string "by" and subsequent levels by the text string "and". The identification itself takes the form of the labels for the classificatory variables (with the names of the variables appearing instead should no labels be defined). Thus the title for a table having province as the row variable and family type as the column variable might include the text string "by Province and Family Type" as the final portion of the first line.

e) The first line of a table title, consisting of the text described in a) through d) above appears for all segments of the table. If the table occupies more than one segment, then a second line will indicate those things that are "held constant" within the segment; if the table consists of multiple segments, then there necessarily exists at least one such constancy. Thus, for example, if a table uses sex of the individual as a classificatory variable, and a particular table segment contains only males, then the text string "Sex = Male" will appear in the second line of the segment title. This text string is composed of two pieces (the variable description and the class label) separated by an equals sign (" = "). If there are multiple such "control" variables for the segment, then the classificatory variables all occur on the second line of the title, appearing in descending order by level, and the individual text strings comprising the line are separated by commas.

Exceptions and Miscellaneous

Now that the notions of tabulation and classificatory variables and labeling have been developed in some detail, we can indicate how the left-to-right "increasing frequency of cycling through values" variables may be overridden when there is only a single tabulation variable. In the working description presented above, the rightmost level was treated as varying most frequently (column categories), the next level to the left as varying next most frequently (row categories), etc. Indeed, this is precisely what will happen if the analyst employs a tabulation level that include more than one variable or expression.

However, in the interests of simplicity and elegance of table appearance, the X-tab facility will override this general rule in the case where a tabulation request uses only a single tabulation variable or expression. In that special case, the X-tab facility creates the table (and its title) as if the analyst had entered the tabulation level of the request as the highest (first or leftmost) level. The order of the remaining, classificatory variable, levels is left unchanged. This override maximizes the number of classificatory variables appearing in a table or table segment. It also prevents the occurrence of multi-segmented tables for which each of the segments would consist of a single row or single column.

Worked Examples

With development of the components of X-tab facility tabulation requests complete, attention shifts to tabulation requests taken as a whole, and to the kinds of tables that result. This section presents a graduated set of examples. Numbered X1 through X9, the examples consist of the table requests and the tables that result. The first seven of the examples use the 5% sample of the SPSD in 1988 with a selection specification that does not exclude any records. The remaining examples use the full (100%) base, but apply a selection criterion that selects only persons employed full year in a full time job.

Aggregating a Single Variable

The first example illustrates the simplest possible request. The user wishes to find out what the SPSD/M 5% sample uses as the population of Canada in 1988. The XTSPEC parameter is given as:

```
XTSPEC
{persons};
```

i.e., the analyst designates only a single tabulation variable, uses no classificatory variables, and accepts the X-tab facility's default for the unit of analysis (individuals). The resulting table appears as:

```
Table X1: Person Count (000) for Individuals

+--------------------+--------+
|Person Count (000)  | 25212.8|
+--------------------+--------+
```

The table contains only a single number, well labeled, with the X-tab facility providing the default scaling factor of 1,000.

A Multiple Table Request with One Classificatory Variable

For the second and third example tables, the analyst wants to see how much private pension income (variable *idipens*) exists in the SPSD, broken out by the labour force status of the recipient individual (variable *idlfst*). Here the XTSPEC parameter includes two distinct tabulation requests:

```
XTSPEC
IN:idlfst
   *{idipens};
IN:idlfst
  *{persons};
```

Tables X2 and X3 display the resulting tables.

Table X2: Pension income (M) for Individuals by Labour force status

Labour force status	Pension income (M)
N/A	0.0
Employed	1270.1
Unemployed	55.1
Not in LF	5169.8

Table X3: Person Count (000) for Individuals by Labour force status

Labour force status	Person Count (000)
N/A	5536.8
Employed	10996.6
Unemployed	1465.6
Not in LF	7213.8

As one would expect, the bulk of private pension income, some $5 billion, is associated with persons not in the labour force, presumably because they are retired. Some private pension also accrues to those who are still employed, or who are unemployed, but the average amounts for these other recipients are much lower than for those classed as "Not in LF".

Requesting Multiple Tabulation Variables in a Single Table

Although analysts will often wish to include multiple tabulation requests in a single XTSPEC, with different tabulation variables appearing in different tables, they may also wish to include multiple tabulation variables in a single table. In this example, the analyst wishes to examine, by province, the aggregate amounts of income according to four different definitions of income. The aggregates of these incomes across Canada are also desired. In this example, the tabulation request appears as follows --

```
XTSPEC CF:hdprov+
         {immtot,
          imitax,
          immdisp,
          immicons};
```

Here the analyst has chosen to specify a census family unit of analysis; note, however, that since the selection specification includes all records and since the tabulated variables are income measures, the unit of analysis will be irrelevant to the resulting table. That resulting table appears as follows --

```
Table X4: Selected Quantities for Census Families by Province

+--------------+----------+----------+----------+----------+
|Province      |  Total   | Taxable  |Disposable|Consumable|
|              |income (M)|income (M)|income (M)|income (M)|
+--------------+----------+----------+----------+----------+
|Newfoundland  |    4146.9|    2067.0|    3544.5|    3150.7|
|P.E.I.        |     940.7|     466.5|     803.8|     719.6|
|Nova Scotia   |    8440.7|    4653.8|    7025.8|    6347.9|
|New Brunswick |    6668.6|    3491.6|    5541.4|    4951.5|
|Quebec        |   79654.1|   44294.7|   62291.6|   56486.1|
|Ontario       |  117893.7|   70393.8|   96575.5|   87004.8|
|Manitoba      |   11645.7|    6561.4|    9586.1|    8605.6|
|Saskatchewan  |   11387.1|    6063.1|    9591.2|    8740.6|
|Alberta       |   31204.8|   19142.6|   25509.8|   24027.2|
|B.C.          |   38351.3|   22733.6|   31243.9|   28608.8|
+--------------+----------+----------+----------+----------+
|All           |  310333.5|  179868.0|  251713.7|  228642.8|
+--------------+----------+----------+----------+----------+
```

The total income (Column 1) is, of course, considerably larger than the portion that is taxable for income tax purposes (Column 2) because of the deductions and exemptions that are available. Disposable income (Column 3) then falls between those first two definitions because income taxes have been subtracted. Finally, consumable income (Column 4) is somewhat lower than disposable income because various commodity taxes have been further subtracted.

Introducing Tabulated Expressions and Qualifiers

In this example the analyst also wants to look at multiple variables in the same table, specifically disposable income, census families and average disposable income per census family. However, as distinct from the previous examples, this last variable is not available either from the database or the model. Consequently, the analyst enters an expression, rather than a variable, for the third item to be tabulated. This example is also the first to use qualifiers for the dependent variables. Here the analyst chooses to report the disposable income in billions of dollars `(S=9)` and to provide a label ("Avg Disp Inc") for the average disposable income per unit. The analyst also puts the tabulation level before the classificatory level so that the tabulated variables will serve as row headings while the province categories appear as column headings. The relevant `XTSPEC` parameter then appears as --

```
XTSPEC
CF:{immdisp:S=9,
    units,
    immdisp/units:L="Avg Disp Inc"}
   *hdprov+;
```

The analyst's use of *units* as a tabulation variable, and as part of the tabulation expression, refers to the CF or census family selection as the relevant unit of analysis for the table. The resulting table then appears as follows --

```
Table X5: Selected Quantities for Census Families by Province
```

Quantity	NFLD	PEI	NS	NB	QUE	ONT	MAN	SASK	ALTA	BC	ALL
Disp. Income (B)	3.5	0.8	7.0	5.5	62.3	96.6	9.6	9.6	25.5	31.2	251.7
Unit Count (000)	176.2	39.4	318.2	256.9	2599.6	3612.5	399.5	412.6	1016.9	1221.6	10053.4
Avg Disp Inc	20114	20420	22082	21568	23962	26734	23996	23247	25085	25577	25038

As expected, the average census family disposable income is rather lower in the Maritime provinces, with Ontario and Alberta having the highest averages.

Two Classificatory Levels

Up to this point the examples have not illustrated true cross-tabulation because they have included at most one classificatory variable. In this example the analyst wishes to look at occupational categories (variable *idocc*) controlling for the sex of the individual (variable *idsex*). In the tabulation request shown here, two other things are new. First, the request

uses the household (HH) as the unit of analysis, and second, the analyst wishes to count (unweighted) SCF records in the database rather than display aggregate persons or dollars.

The request for this table appears as follows:

```
XTSPEC
HH: idocc+
  * idsex
  * {scfrecs};
```

Recall that when there is only a single tabulation variable in the tabulation level, the X-tab facility processes the request as if the tabulation level were the first level in the request. The resulting table appears as follows:

Table X6: SCF Records for Households by Occupation and Sex

	Sex	
Occupation	Male	Female
Never worked	3	71
Managerial	143	33
Professional	104	47
Teaching	23	15
Clerical	53	58
Sales	103	15
Services	124	83
Agricultural	117	4
Mining, Processing	104	3
Fabrication, Assembly	105	7
Construction	113	0
Transport, Handling	106	5
Last worked > 5 years	214	181
All	1312	520

Parts of the table are exactly as one might expect on the basis of historical analyses in

occupational choices. Women are relatively concentrated in classifications involving Teaching, Clerical, Services, Never worked, and Last worked > 5 years (ago). Men are over-represented in occupations such as Managerial, Mining, Fabrication, Construction, and Transport.

The X-tab facility's roll-up facility explains the relative sizes of the two columns, i.e. why there are over twice as many records for men as for women. Because the analyst requested a tabulation for households, even though occupation is normally considered to be an individual characteristic, the SPSM "rolled up" the reported characteristics for occupation to give a single occupation for each household, specifically, the occupation of the male (for couples) in the primary economic family in the household. Thus men, who tend to have higher incomes, are disproportionately in the record counts that make up the table's entries.

A second feature of note is how, even with only two classificatory variables, the numbers of records entering into certain cells of the table is very low or even zero. The analyst must regularly be on guard to ensure that table entries are based on sufficiently many records to permit meaningful conclusions. The SPSM's ability to tabulate numbers of records is a very useful tool toward this goal.

A User-defined Classificatory Variable and a Multi-line Request

In this example the analyst wants to see how housing expenditures vary as a proportion of income for various combinations of income level and tenure. Once again, the desired tabulation variable is not available directly from the database or model, so the analyst constructs it. The construction here involves adding together a number of housing-related expenses from the FAMEX-sourced portion of the database and dividing the sum by a measure of the families' total income. Another new thing is the use of a user-defined classificatory variable, `cl1`, to provide the income categories. For this purpose the analyst has included, elsewhere in the UVAR request the following sentences:

```
classinc = split(immtot, 10000,20000,30000,40000,50000,60000,70000);
levels(classinc) = "MIN-10","10-20","20-30","30-40","40-50","50-60","60-
70","70-MAX";
label(classinc) = "Total income Group";
```

The split request creates 8 categories of income based on immtot defined in the levels() definition. The levels() assign classinc as a classification variable with a maximum length of 60 caracters. The user defined labels for the classes correspond to the following limits: minimum to 10000, 10001 to 20000, 20001 to 30000, 30001 to 40000, 40001 to 50000, 50001 to 60000, 60001 to 70000, 70001 to maximum. If no labels are assigned the user should use nlevels() to assign the correct number of classifications to the variable classinc and the output will identify the classes as 0 to 7. The label "Total income Group" is used to give a title to the variable in the table output.

Note that the variable *immtot* affects the table in two distinct fashions: 1) the classification of variables by income group and 2) the definition of the tabulation expression. The tabulation

request to generate the table then appears as

```
XTSPEC
HH:{(fxio7+fxio8+fxio9+fxio10+fxio11+fxio12)/immtot:
            L="Housing Exp Prop" S=% P=3 }
   * classinc+
   * hdtenur+;
```

In the request the analyst wishes also to supply a label for the tabulated variable, as well as to control both the scaling factor and the number of decimal places in the table. The analyst can always use as many lines required to make the request easy to read. The resulting table then appears as

```
Table X7: Housing Exp Prop (%) for Households by Total Income Group and
Tenure

                                        Tenure
+--------------------+----------+----------+----------+----------+
|Total income Group  |  Rented  |Owned with|Owned w/o |   All    |
|                    |          |  Mtge.   |  Mtge.   |          |
+--------------------+----------+----------+----------+----------+
|Min-10              |    37.025|     8.951|    19.065|    30.457|
|10-20               |    25.711|     6.938|    10.656|    18.352|
|20-30               |    20.248|     5.995|     6.523|    11.904|
|30-40               |    17.661|     5.199|     4.803|     9.186|
|40-50               |    13.273|     4.337|     4.284|     6.072|
|50-60               |    14.459|     4.128|     2.722|     4.500|
|60-70               |     9.631|     4.070|     2.631|     6.541|
|70-Max              |     9.387|     3.653|     2.041|     4.219|
+--------------------+----------+----------+----------+----------+
|All                 |    17.829|     4.498|     4.583|     8.494|
+--------------------+----------+----------+----------+----------+
```

The analyst's choice of scaling factor means that the table's entries correspond to percentages of total income. As expected, the proportion of total income spent on this particular combination of housing expenditures falls steadily as one moves across the income categories. Also as expected, homeowners spend a lower fraction of income on these categories of housing expenditure than do renters. The table also suggests a crossover in the proportion between owners with and without a mortgage as one moves up the income distribution.

Introducing Normalization and a Third Dimension

In the final pair of examples the analyst wishes to compare the relative earnings of men and women, normalizing them to an average annual earnings amount for persons working fifty or more weeks per year `(idlyww>=50)` and full time rather than part time `(idlyfp==1)`, and so begins by establishing the selection specification parameter for the analysis:

```
SELFLAG      1
SELUNIT      0
SELSPEC      idlyww>=50 && idlyfp==1
```

The analyst wonders if marital status (variable *idmarst*) and age (variable *idage*) seem to contribute to the patterns observed, and so includes them as classificatory variables. As well

the analyst is very concerned not to draw misleading conclusions from the analysis and thus elects to use the full (100% sample) SPSD rather than the 5% sample. Along these same lines, the analyst decides, somewhat arbitrarily, that in deriving the normalized values, it may be inadvisable to use any average annual earnings not based on at least one hundred cases in the database.

Because the analyst wants to use a relatively small number of age categories, the SPSM facility UVAR should be used. The relevant lines are --

```
agegrp = split(idage, 35,45,55,65);
levels(agegrp) = "MIN-35","36-45","46-55","56-65","66-MAX";
label(agegrp) = "Age Group ";
```

Thus the analyst wishes to generate two tables. The first of these will show the desired normalized annual earnings per person, and the second will show the numbers of database records on which the normalized values are based (so that potentially untrustworthy values can be suppressed, or at least given less weight). The XTSPEC parameter for this pair of tables appears as:

```
XTSPEC
IN: {(idiemp+idisefm+idisenf)/persons:
              L="Per Capita Normalized" M=idsex S=% P=2}
      * idsex+
      * agegrp+
      * idmarst+;
IN: {scfrecs: L="Instances (records)" P=0 S=0}
   * idsex+
   * agegrp+
   * idmarst+;
```

In the first tabulation request the analyst adds together the income from paid employment and from farm and non-farm self-employment and divides that sum by persons to arrive at average earnings per person. Qualifiers provide a label "Per Capita Normalized" for the resulting quantity, impose the normalization along the sex dimension `(M=idsex)`, and fix the scaling factor `(S=%)` and number of decimal places `(P=2)`. The ordering of the three classificatory dimensions ensures that sex will vary only across segments, while individual table segments will display age as the row categories and marital status as the column categories. This is the table the analyst cares most about.

The second table request is very similar, except that the analyst is counting the SCF records for individuals. If, for either men or women, a table segment contains a number of records less than one hundred, then both of the normalized values for the corresponding locations of segments from the first request will be regarded as potentially suspect. The several table segments that the X-tab facility generates as a result of this pair of requests appear on the following page.

```
Table X8: Per Capita Normalized (%) [idsex] for Individuals by Sex and Age
Group and Marital status
          Sex = Male
                                  Marital status
+----------+---------+---------+---------+---------+---------+
|Age Group | Married | Single  |Widow(er)|Divorced |   All   |
|          | or CLU  | (never  |         |         |         |
```

		married)			
Min-35	115.07	106.14	119.44	115.26	113.28
36-45	115.32	108.27	122.58	120.79	115.96
46-55	114.34	104.52	139.26	113.91	115.34
56-65	110.25	101.32	137.48	122.35	112.86
66-Max	101.78	103.64	85.02	97.71	105.97
All	114.50	105.57	132.32	117.32	114.99

Table X8 (cont.): Per Capita Normalized (%) [idsex] for Individuals by Sex and Age Group and Marital status

Sex = Female

Marital status

Age Group	Married or CLU	Single (never married)	Widow(er)	Divorced	All
Min-35	75.33	92.14	76.13	88.86	80.76
36-45	66.11	91.23	94.10	83.95	69.92
46-55	62.57	96.07	83.05	87.28	67.64
56-65	63.84	99.23	81.20	85.79	70.09
66-Max	82.78	97.86	106.39	104.93	77.29
All	68.62	93.35	84.62	86.50	73.75

Table X9: Instances (records) for Individuals by Sex and Age Group and Marital status

Sex = Male

Marital status

Age Group	Married or CLU	Single (never married)	Widow(er)	Divorced	All
Min-35	5056	2158	16	201	7431
36-45	4166	251	14	222	4653
46-55	2867	135	41	148	3191
56-65	1769	96	56	63	1984
66-Max	249	22	12	8	291
All	14107	2662	139	642	17550

Table X9 (cont.): Instances (records) for Individuals by Sex and Age Group and Marital status

Sex = Female

Marital status

Age Group	Married or CLU	Single (never married)	Widow(er)	Divorced	All
Min-35	2878	1629	20	285	4812

36-45	1868	237	55	286	2446
46-55	1135	192	117	175	1619
56-65	505	167	140	98	910
66-Max	26	58	26	6	116
All	6412	2283	358	850	9903

If the analyst takes the one hundred cases requirement seriously, then nearly all of the entries in the third column, and in the fourth and fifth rows are suspect (all of those cells except for the first column and fourth row). Should the analyst elect to suppress these cells (the X-tab facility does not itself do this), then the tables from the the first request would look something like:

Table X8a: Per Capita Normalized (%) [idsex] for Individuals by Sex and Age Group and Marital status
Sex = Male

Age Group	Marital status				
	Married or CLU	Single (never married)	Widow(er)	Divorced	All
Min-35	115.07	106.14		115.26	113.28
36-45	115.32	108.27		120.79	115.96
46-55	114.34	104.52		113.91	115.34
56-65	110.25				112.86
66-Max					105.97
All	114.50	105.57	132.32	117.32	114.99

Table X8a (cont.): Per Capita Normalized (%) [idsex] for Individuals by Sex and Age Group and Marital status
Sex = Female

Age Group	Marital status				
	Married or CLU	Single (never married)	Widow(er)	Divorced	All
Min-35	75.33	92.14		88.86	80.76
36-45	66.11	91.23		83.95	69.92
46-55	62.57	96.07		87.28	67.64
56-65	63.84				70.09
66-Max					77.29
All	68.62	93.35	84.62	86.50	73.75

Generally speaking, men appear to have annual earnings higher than the average for all persons. For example, married men have annual earnings roughly 14% higher than the average for all married persons, and, given the relative numbers of fully employed men and women, women have average annual earnings that are only about 68% of that same "all married persons" value. Marital status appears to have some explanatory power as to the degree of the disparity between equality of incomes (given the analyst's failure to control for

other variables such as occupation and the age mix of the labour force), but age doesn't seem to make a great deal of difference.

Cross-tabulation Hints

The preceding sections have focussed primarily on the more technical aspects of how the X-tab facilityworks. Building on this foundation, we here briefly take up several topics addressing how the analyst uses the X-tab facility to perform typical analyses; i.e. the focus is considerably less technical and much more procedural.

External Limitations Affecting the X-tab Facility

As described above, the design goal for the X-tab facility was the creation of a facility that would address typical tabulation requirements within the SPSD/M. Thus, the X-tab facility enables analysts to generate, quickly and conveniently, a variety of customized tables without the necessity to export the relevant data, coupled with a requirement to import that exported information into another package and then process it there. However, given the requirement that it operate inside the SPSD/M without imposing significant penalties on the time needed for a run, the X-tab facility cannot be as comprehensive as a stand-alone package devoted exclusively to statistical analysis. (For example, the X-tab facility does not have the capacity to nest one classificatory variable within another along a single table dimension.)

Some of the resulting limitations occur primarily as a result of the machines on which the SPSM is designed to operate and on their operating systems. For example, the architecture of the machines means that the SPSM (including the X-tab facility and any "glass box" algorithms contributed by the analyst) and all of the intermediate values for tables and variables required by the analyst must collectively fit into a fixed amount of memory. Although the SPSM and the X-tab facility have been designed to conserve this memory as far as possible, it is possible that the analyst may reach the limits by requesting too many variables and tables. Other "mechanical" limitations, e.g. a maximum of 16K cells in a tabulation request, are technically present, but will not be particularly constraining in typical analyses.

Because the limitations will be different for different machines, e.g. depending on the version of the operating system being used or the presence of RAM-resident software, it is impossible to provide specific limitation values or even simple algorithms to compute them. Generally speaking, the analyst should be aware that these kinds of limitations exist and further that the memory requirements inherent in cross-tabulation requests may trigger them. With this awareness, the analyst will be better able to recognize them when they occur and, when necessary, to discover alternative methods of achieving the same analytic goals.

Choosing the Right-sized SPSM Run

One of the SPSM's primary distinguishing characteristics is that it runs on a microcomputer. For analysts without access to the mainframe computers and packages previously required for meaningful microsimulation analysis, this means that such analyses now fall into the realm of the possible; they no longer have to be contracted out to others. For analysts previously relying on mainframe computation, the SPSM often slashes both computational

costs and turnaround time. Less obviously, the availability of the SPSM and its X-tab facility may also change the manner in which the analyst approaches an analysis.

The big advantage will be that analysts no longer face high marginal costs for simulation analyses. They will be able to explore directions suggested by previous runs. They will be able to dig into the causes of curious or unexpected results. They will be able to explore alternative perspectives on various phenomena. Above all, they will be able to do so within the same time and dollar resource limits previously associated with less sophisticated analyses.

Analysts used to performing their own simulations on a mainframe will probably be most affected. A mainframe environment contains a number of incentives for making big, comprehensive runs. For example, there may be turnaround problems associated with submitting runs, obtaining hardware such as tape drives or printing results; the analyst responds by including in runs more results that "might" be needed and would hold up work if they were unavailable. Again, there might well be time and cost economies associated with generating tables in a minimum number of runs to cut the number of passes over the relevant database. With the SPSM and X-tab facility available, the reasons behind those former workstyles may no longer apply. As well, the microcomputer-related limitations of the SPSM may mean that some previous "kitchen sink" style runs are no longer feasible. We expect that these analysts will gradually become accustomed to a more interactive style of analysis. They will find it effective to make more runs with fewer outputs per run. They should increasingly be able to explore new avenues of analysis based on the results of previous runs.

Exporting Data to Other Packages

As useful and powerful as the X-tab facility may be, it is, as described above, not intended to compete with fully-fledged statistical analysis packages. Thus, the analyst should not feel compelled to execute all tabulations via the X-tab facility. Instead, a more pragmatic approach is called for. Things that can be done most effectively within the SPSM and X-tab facility should be done there. However, if something is unreasonably difficult to do within the SPSM/X-tab facility, then the analyst should give serious thought to doing it elsewhere. Earlier sections of this chapter have already discussed the SPSM's ability to export data in both ASCII and SAS formats, taking full advantage of the SPSM's powerful roll-up capacity in the generation of the exported data. It is also possible to create new tables from SPSM tables by using the `import` utility in conjunction with a spreadsheet package such as Lotus 1-2-3 or Symphony. See the *SPSD/M Tools User's Guide* for more information on this topic.

Using Parameter Include Files

Recall that the SPSM provides a parameter editing facility so that an analyst can read in a parameter file, including control parameter files, and then modify selected parameter values. In addition, the analyst can use the SPSM's `read` facility to override whole sets of parameters, drawing the replacement values from "pieces" stored on disk files.

Because XTSPEC (as well as related parameters such as XTFLAG, XTCOLS and XTLINES) is a parameter in the control parameter file, the read facility applies to it. Thus, an analyst who regularly needs sets of certain customized tables that are common across SPSM runs can store the XTSPEC for generating these tables in a disk file. Subsequently, when the table set is next needed, the analyst can quickly and conveniently call it into use via the SPSM's read facility (in conjunction with whatever other control parameters are relevant).

Generating Efficient and Elegant Tabulation Requests

The SPSM provides "intelligent defaults" for a number of items that affect the readability of tables (titles, variable and class labels, segmentation, etc.). However, the analyst's description of the desired tables also has a major impact on the appearance, as well as the content, of those tables. The following items identify some of the more important such choices.

a) Take advantage of the `L` qualifier for tabulation variables, especially when tabulating an expression. Take advantage of the `S` and `P` qualifiers to apply appropriate scaling and to control the number of digits after the decimal point in table entries.

b) Consider the limitations of the output device, (e.g. printer) into account when choosing the identity and order of the classificatory variables in a table. For example, avoid generating a table wider than your printer can print.

c) Consider the shape that the requested tables will take, and the suitability of that shape for their intended uses. For example, three table segments of six columns and twelve rows each, all on one page, might be far more useful than twelve table segments of three columns and six rows, each segment appearing on a separate page. Use the XTCOLS and XTLINES parameters and a thoughtful ordering of levels to produce a more pleasing table.

d) Remember that the X-tab facility permits the analyst to include multiple tabulation variables in a single table or table segment, and to position them along either the row or column dimension. Sometimes this can be much more effective than having the multiple tabulation variables spread across multiple tables or table segments.

e) Normalizing as an Analytic Technique: Policy applications of simulation results often require looking at comparative results, e.g. relative rates of poverty. The SPSM's ability to perform various normalizations (via the `M` qualifier for tabulated variables and expressions) can be a big help to the analyst. The X-tab facility's normalizing capacity sees regular use in generating desired row and column percentages.

Summary

This section offers a capsule summary of the key points relating to the X-tab facility.

a) The X-tab facility provides a convenient mechanism for the analyst to generate customized cross-tabulations without paying the costs of exporting intermediate results for processing by another software package.

b) XTSPEC, an element of the control parameter file, is the major device by which the analyst specifies the desired tables. XTSPEC is primarily a collection of tabulation requests, with the individual requests separated by semicolons.

c) A tabulation request consists of multiple levels; with the exception of tables containing only one tabulated variable (or expression), each of these levels corresponds to a dimension in the resulting table. Levels in a tabulation request are separated by asterisks. From right to left the successive levels correspond first to columns, then to rows, next to segments, etc.

d) Exactly one level tells what is being tabulated, be it variables and/or expressions. This level is called the tabulation level, and it is distinguished by being enclosed in braces (the `{` and `}` characters).

e) A tabulation level may contain multiple tabulated variables or expressions. When multiple items are present, they are separated by commas.

f) The tabulation level may also involve various qualifiers that control labels, scaling, digits in the printed result and normalization. The scope of a qualifier is limited to the tabulation variable that it follows. The presence of qualifiers for a given tabulation variable/expression is indicated by a colon, with multiple qualifiers separated by spaces.

g) Levels other than the tabulation level are termed classificatory levels. A table request may have from zero to six classificatory levels. Each classificatory level consists of a single classificatory variable.

h) In constructing tabulation requests, the analyst may employ both database and model variables as well as being able to tabulate analysis expressions.

i) The SPSM and X-tab facility's power in handling multiple levels of unit of analysis, and in permitting the analyst to mix these levels within analyses and within single tables, carries with it certain dangers and responsibilities. Although the design of the X-tab facility ensures that most cases are handled naturally, the analyst must be knowledgeable about, and sensitive to the implications of, roll-up issues affecting both tabulation and classificatory variables.

j) The SPSM provides, independently of the X-tab facility, various capacities to select subsets of the overall database. It is the responsibility of the analyst, when interpreting tables, to be aware of the selection criteria relevant to those tables.

Appendix A Major X-tab Facility Error Messages

This appendix presents the major SPSM error messages associated with the X-tab facility. This summary is placed here to make the X-tab facility documentation more self-contained for the SPSM analyst.

Where possible, the X-tab facility includes in its error messages its identification of the information that is causing the error. In the error messages and interpretations appearing below, strings in italics represent critical text string extracted from the user's request. As well, the notation "XTSPEC # n" should be interpreted as the "n'th" table request within XTSPEC; thus, for example, "XTSPEC # 2" refers to the second table request within XTSPEC.

`error(100): out of memory`

The X-tab facility has run out of memory (RAM) in which to collate the requested outputs. (This is a run time message, in contrast to most of the other X-tab facility messages, which occur when the SPSM is interpreting the control parameter file.) The analyst must break the request into smaller components that can be executed in multiple SPSM runs.

`error(957): error in expression` *`expr`*

The X-tab facility is unable to process the expression *expr*, probably due to a syntactical error. The analyst must rework the expression so that it is understandable.

`fatal error(959): too many classification variables in XTSPEC #` *`n`*

The X-tab facility has a limited amount of internal storage for handling classificatory variables, and the indicated table request has pushed past this limit. The analyst must reduce the number of classificatory variables being used, possibly by breaking the table requests up across multiple runs.

`fatal error(960): too many expressions in XTSPEC #` *`n`*

The X-tab facility has a limited amount of internal storage for processing the user-defined expressions, and the indicated table request has pushed past this limit. The analyst must reduce the number of these expressions being used, possibly by breaking the table requests up across multiple runs.

`fatal error(961): too many analysis variables in XTSPEC #` *`n`*

The X-tab facility has a limited amount of internal storage for storing the information associated with analysis (tabulation) variables, and the indicated table request has pushed past this limit. The analyst must reduce the number of analysis variables being used, possibly by breaking up the table requests across multiple runs, and/or analyzing fewer tabulated variables within a table request.

`fatal error(962):` *`var`* `not a valid analysis variable in XTSPEC #` *`n`*

The X-tab facility interprets the string (shown here as var) as an intended tabulation variable or expression, but it does not meet the requirements of something that can be tabulated. Possibly an integer variable was used as a tabulation variable or in a tabulation expression, even though the SPSM requires that these variables and expressions involve only analysis variables. The analyst should review the documentation as to what the X-tab facility can

tabulate.

`fatal error(963): var not a valid class variable in XTSPEC # n`

The X-tab facility permits only certain kinds of variables to be used as classificatory variables in table requests. The analyst should ensure that the intended variable is defined as an integer variable in the database or model, or that it is a "CL" variable whose labels have been properly defined (e.g. are equal in number to the categories permitted for the variable).

`fatal error(964): no analysis expression in XTSPEC # n`

The X-tab facility requires that each table request contain exactly one level denoting the tabulation variables or expressions. This level is identified by being delimited with braces ({ and }). The X-tab facility has found no such level in the indicated table request. The analyst must eliminate the table request or edit it to include an appropriate tabulation level. If the analyst desired simply a count in each cell, use the units variable.

`fatal error(965): bad margin dimension in XTSPEC # n`

The X-tab facility is processing a table request that includes an M (margin) qualifier to request a normalization; however, the classificatory variable appearing in the qualifier is not one of the classification variables for the table, rendering the normalization impossible. The analyst must ensure that the normalization variable is one of the classificatory variables.

`fatal error(966): too many variable crossings in XTSPEC # n`

The X-tab facility has discovered that the indicated table request is nested too deeply. Only six classificatory levels are permitted. The analyst must eliminate at least one of the classificatory variable levels from the table request.

`error(967): invalid expression label in XTSPEC # n`

The X-tab facility has discovered, in processing an L qualifier for a tabulation expression in the indicated table request, that the qualifier does not obey the syntax for a valid label. The analyst should edit the L qualifier, ensuring that it takes the form of the character pair L=, followed immediately by a "short" string delimited by double quotes, followed by any of (1) the brace that terminates the tabulation level, (2) a comma that indicates the end of the qualifiers for the tabulation expression, or (3) one or more whitespace characters preparatory to the next qualifier.

`error(971): invalid constant const encountered`

The X-tab facility, in evaluating a tabulation expression, has encountered a string, *const* that it "believes" is intended as a constant, but which it cannot interpret. The analyst must convert the string to something that the X-tab facility can understand. This error may also indicate a syntax error in the expression.

`error(972): expecting operator but encountered string`

The X-tab facility, in evaluating a tabulation expression, encountered the text string *string* when the syntax of an expression would require a mathematical/logical operator. The analyst must edit the expression so that it obeys the syntax requirement.

`fatal error(975): class variable value error in XTSPEC # n`

The X-tab facility, in attempting to accumulate a table for the indicated table request, encountered an inappropriate/unexpected value for one of the classificatory variables in the

request. If this occurred for a database or "black box" model variable, then the analyst should notify the SPSM development team so that they can correct the error. If the cause is one of the analyst's variables from a "glass box" model or from one of the "CL" variables, then it is the analyst's responsibility to make sure that the classificatory variable takes on only legitimate values and that labels defined for that classificatory variable exist in a one to one correspondence with the valid values for the variable.

`fatal error(976): unknown maximum for variable var in XTSPEC # n`

The X-tab facility, in trying to use the variable var in the indicated table request, probably in the context of using an analyst defined integer variable as a classificatory variable, has noted that there does not exist an appropriate set of labels for the values of the variable (the cardinality of which set defines the number of categories deemed to be legitimate for the variable). The analyst must ensure that an appropriate set of "value labels" is defined prior to the SPSM run. This error should occur only in "glass box" applications, and will typically result from an incorrect or incomplete definition of a new integer variable.

Appendix B Parsing Details

SPSM analysts with a computer science background will recognize the XTSPEC parameter as a string that must be parsed and interpreted by the SPSM before the SPSM can generate the desired tables. This appendix documents the basis for that parsing, i.e. the formal syntax of XTSPEC. In the following production rules, braces group repeated constructs and are followed by one of the symbols `*', `?', or `+'. `*' means 0 or more, '+' means 1 or more, and '?' means 0 or 1 of the preceding construct. Quoted symbols indicate literal values.

```
Xtspec              ::=   table_request { `;' table_request }*

table_request       ::=   { level_of_analysis }? level_list

level_of_analysis   ::=   IN: | NF: | CF: | EF: | HH:

level_list          ::=   level_spec { `*' level_spec }*

level_spec          ::=   class_level | tabulation_level

class_level         ::=   class_var { `+' }?

Tabulation_level    ::=   `{' tab_expr_list `}'

tab_expr_list       ::=   tab_expr { `,' tab_expr }*

tab_expr            ::=   expr { `:' qualifier_list }?

Qualifier_list      ::=   { qualifier }+

qualifier           ::=   L="label"
                    |     S=integer
                    |     P=integer
                    |     M=class_var

expr                ::=   `(' expr `)'
                    |     unary_op expr
                    |     expr bin_op expr
                    |     analysis_var
                    |     number

unary_op            :=    `-'

binary_op           :=    '+' | '-' | '*' | '/'
```