BeSD-TI

Behavioural and Social Drivers of Vaccination – Tools for Impact

User's Guide

Draft Version 1.0
Revised December 2024

This page is intentionally blank.

Table of Contents

List of Tables

Acknowledgments

The Behavioural and Social Drivers of Vaccination – Tools for Inference (BeSD-TI) software was developed in 2024 for the World Health Organization by Biostat Global Consulting.

This User's Guide was written by Caitlin Clary; some material was repurposed from the User's Guides for VCQI[1] and MISS VCQI.[2]

[1] Vaccination Coverage Quality Indicators: https://www.biostatglobal.com/VCQI_resources.html

[2] Missed Opportunities for Vaccination Coverage Quality Indicators: https://github.com/BiostatGlobalConsulting/missvcqiR

Chapter 0. Preliminary Material

Document Revision History

Date	Version	Notes
2024-12-13	Version 1.0	Original draft

Terminology note: user-defined input parameters

This User's Guide uses the term *global value* to refer to values in the global environment that users define. These values are often established in BeSD-TI control programs using the `besd_global` R function. The term *global value* is not typically used in R documentation, but in our context it refers to a value that has been assigned in the global environment (using the `besd_global` function or the <- operator) and which remains defined until its value is changed or until the conclusion of the current BeSD-TI analysis session.

Acronyms and Abbreviations

BeSD	Behavioural and Social Drivers of Vaccination
BeSD-TI	Behavioral and Social Drivers of Vaccination – Tools for Inference
CI	Confidence Interval
CM	Cluster Metadata dataset
COVID-19	Coronavirus Disease 2019
DEFF	Design Effect
DESC	Descriptive indicators for summarizing multiple choice questions
FVL	Forms and Variable List document
WHO	World Health Organization

License Agreement

(Pending approval by WHO)

Redistribution and use of the Behavioural and Social Drivers of Vaccination – Tools for Inference software (in source and binary forms) and documents, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.[3]

[3] https://opensource.org/license/BSD-3-Clause

Chapter 1. Introduction

Behavioural and social drivers of vaccination (BeSD) are "beliefs and experiences specific to vaccination that are potentially modifiable to increase vaccine uptake."[4] The BeSD framework developed by the World Health Organization (WHO) groups these drivers into four domains:[5]

- **Thinking and feeling** about vaccines and vaccine-preventable diseases
- **Social processes** and norms that affect vaccine uptake
- **Motivation** (or hesitancy) to seek vaccination
- **Practical issues** involved in seeking and receiving vaccination, such as accessibility and cost

The Behavioural and Social Drivers of Vaccination – Tools for Inference (BeSD-TI[6]) software is a set of programs written in R to make standard output and tables from analysis of BeSD surveys. The BeSD-TI programs are freely available, courtesy of the World Health Organization (WHO). BeSD-TI and its supporting manuals and materials are available on GitHub.[7]

To install and run BeSD-TI, users need to:

- Have installed R version 4.2.1 or later[8]
- Have installed RTools; users should install the version corresponding with their version of R[9]

The R package can be installed directly from GitHub using the following commands:

```
if (!requireNamespace("pak")){install.packages("pak")}
pak::pkg_install("BiostatGlobalConsulting/BeSDTI")
```

We also recommend using the freely-available RStudio integrated development environment.[10]

This guide assumes that the user has basic knowledge of how to use R. The details in this guide will help users understand the detailed definitions and run-time options of BeSD-TI's indicators and understand the flexible features of the software which may be changed by editing the code in Blocks B, D, and F of the control program.

---

[4] World Health Organization. (2022). Understanding the behavioural and social drivers of vaccine uptake. WHO position paper – May 2022. Weekly Epidemiological Record vol. 97(20). https://www.who.int/publications/i/item/who-wer9720-209-224.

[5] World Health Organization. (2022). Behavioural and social drivers of vaccination: tools and practical guidance for achieving high uptake. https://iris.who.int/handle/10665/354459.

[6] Pronounced *bestie*.

[7] https://github.com/BiostatGlobalConsulting/BeSDTI

[8] https://cran.r-project.org/

[9] Note that RTools is a standalone program to install; it is *not* an R package. If your version of R begins with 4.4, then download RTools 4.4; if your version of R begins with 4.3, then download RTools 4.3, and so on.

https://cran.r-project.org/bin/windows/Rtools/

[10] https://posit.co/download/rstudio-desktop/

Chapter 2. Overview

This section of the document gives an overview of a) files that comprise BeSD-TI, b) datasets and parameter files that need to be assembled in order to run BeSD-TI, and c) the output files that are produced by BeSD-TI.

2.1 Running BeSD-TI

The usual practice is to copy a BeSD-TI control program (.R file) from the examples provided, edit the file (to provide the appropriate file locations and analysis parameters), save it and run it in R. Open the resulting spreadsheet and check the log sheet for errors or warnings. If BeSD-TI ran successfully, examine the results to see if they make sense. If yes, you might copy tabulated results or automatically generated figures into a report. Save the control program and output for future reference. To run a second analysis, copy the control program to a new, empty folder; edit the new program to send its output to that new folder where the control program is saved; save the control program, and run it.

Please note that you will need to spend some time customizing the control program for each new project. You need to specify parameters to describe the survey and the analyses you wish to run. The main purpose of this *User's Guide* is to help you understand the rich set of parameter options available in Blocks B, D, and F of every control program. When you begin a new project the first challenge is to make your dataset compatible with BeSD-TI. The *BeSD-TI Forms and Variable Lists (FVL)* document will help you accomplish that. The second challenge is to edit a control program template and specify parameters that are relevant for YOUR SURVEY. The control program templates are filled with parameters to describe fictional surveys in the fictional country of Harmonia. You must replace the parameters that describe the fictional survey with parameters to describe your survey; the chapters that follow demonstrate how to accomplish this.

BeSD-TI performs a series of checks to be sure the user has defined the necessary inputs and that the input datasets and necessary variables are all present. When something goes wrong, it tries to provide informative error messages both to the R console and in a BeSD-TI log file. If BeSD-TI detects an important error, the log is copied into the output spreadsheet before the program halts. If an unanticipated error occurs, the incomplete log will be a .csv file saved in the BeSD-TI output folder. If you open the spreadsheet and find only placeholder text in the Log worksheet, then close the Excel file and run the command `besd_cleanup()` in the R console. In most cases this will cause the log to be closed, processed, and copied to the output spreadsheet file. Re-open the spreadsheet and look at the log tab. Otherwise follow the instructions found in the placeholder Log tab in the spreadsheet.
If you experience problems running BeSD-TI, contact Caitlin.Clary@biostatglobal.com.

2.2 BeSD-TI Indicator Types

- DESC: Descriptive indicators that summarize responses to multiple-choice questions; these indicators may be calculated for any survey.

WHO BeSD guidance defines five *priority indicators* for vaccination of children—particular questions from the BeSD childhood vaccination questionnaire that measure the following constructs: confidence in vaccine benefits, norms among close family and friends, intention to get child vaccinated, knowledge of

where to get child vaccinated, and affordability of vaccination. To summarize these priority questions, BeSD-TI has a dedicated indicator:

- besd_ch_core: Summarize responses from the priority indicators for a childhood vaccination BeSD survey and produce a bar chart in standard format.

WHO guidance also defines priority indicators for a COVID-19 vaccination survey, measuring the following constructs: confidence in COVID-19 vaccine benefits, norms among close family and friends, intention to get vaccinated, knowledge of where to get a COVID-19 vaccine, and affordability of COVID-19 vaccination. Once again, BeSD-TI has a dedicated indicator to summarize these priority questions:

- besd_covid_core: Summarize responses from the priority indicators for a COVID-19 vaccination BeSD survey and produce a bar chart in standard format.

2.3 Files that Comprise BeSD-TI

R Programs

BeSD-TI is a set of R programs that work together to analyze the survey data. No special license key is required to run the BeSD-TI programs – if you have the BeSD-TI package (see Chapter 1 for instructions on installing the package) and you have a copy of R (version 4.2.1 or later), then you can run BeSD-TI.

Control Program

A primary purpose of this *User's Guide* is to help you understand the BeSD-TI control program and learn to confidently adapt it to meet your needs. A control program is a set of R commands saved in an R script. In most cases a BeSD-TI user will not need to look inside of any programs except the control program because the control program calls all the other programs that it needs. Every control program alternates between clearly marked blocks of code that the user *should edit* and blocks that they *should not edit*. Portions of the control program that you might edit include those that point to folders and datasets, lines that describe the survey, and those that list which indicators you want to calculate. To run BeSD-TI, the user copies a control program and edits the appropriate sections before running the control program in R. Sample control programs are provided in the BeSD-TI GitHub and several are annotated in this *User's Guide*. See Chapter 7 for detailed examples; however, the entire document is meant to help users understand how to use the control program.

2.4 Files Used by BeSD-TI

Datasets

You need to assemble a small set of files to run BeSD-TI – precisely which files you need depends on whether you are analyzing data from a childhood vaccination survey or a COVID-19 vaccination survey. Details appear in later sections of this document. BeSD-TI assumes that the survey data were collected using the survey questions described in the accompanying document named *Behavioural and Social Drivers of Vaccination Surveys – Forms & Variable Lists (FVL) Structured for Compatibility with BeSD-TI* (described hereafter as 'the FVL document'). BeSD-TI assumes that variables are named and coded as described there. If the data were collected using other survey instruments, it will be necessary to recode the data to look as though it comes from questions in the FVL document. BeSD-TI can also analyze data collected from other surveys, however, the variables must be renamed and recoded to be consistent

with those described in the FVL document. The FVL provides information on how childhood vaccination BeSD survey datasets, COVID-19 BeSD survey datasets, and cluster metadata (CM) datasets should be structured. The CM dataset is optional, but is often useful for providing information about survey clusters and their characteristics.

WHO has developed questionnaires in ODK[11] for collecting BeSD data. These forms, available from GitHub,[12] are designed so that the datasets produced will be compatible with BeSD-TI, i.e. the variable names and the coding of response options will match the specifications in the FVL by default. Analysts working with datasets from the WHO ODK forms may still need to modify the survey dataset prior to using BeSD-TI, for instance to add survey weights or to merge other variables of interest.

When summarizing variables, BeSD-TI uses *variable label* and *value label* attributes of input datasets, if those attributes exist, when constructing tables and figures. Variable labels are descriptive names for variables and value labels are descriptions attached to different values that a variable can take. As an example, consider the variable recording a child's gender in the Childhood Vaccination BeSD questionnaire, named *CHI_chigender* and taking the values 1 and 2 signifying Female and Male respectively. If variable and value labels were not used, the data for this question might look like the below:

CHI_chigender
2
2
1
2
1
1
1

And a summary of that unlabeled variable would look something like:

	1 (%)	2 (%)
CHI_chigender	57.1	42.9

By assigning a variable label and value labels, analysts attach attributes to the dataset:

Child's Gender
Male
Male
Female
Male
Female
Female
Female

[11] https://getodk.org/

[12] https://github.com/WorldHealthOrganization/BeSD

The presence of those attributes allows BeSD-TI to produce output that is much easier to interpret:

	Female (%)	Male (%)
Child's Gender	57.1	42.9

The **haven** package[13] in R provides functions to assign variable and value labels in R, as well as functions to read datasets from other statistical software, such as Stata or SPSS, where variable and value labels may be present.

To add variable and value labels using haven, use syntax like the following when making datasets BeSD-TI compatible:

```
dat <- dat %>%
  mutate(
    CHI_chigender = haven::labelled(
      CHI_chigender,
      label = "Child's Gender",
      labels = c("Female" = 1, "Male" = 2)
    ))
```

Alternatively, analysts might use a different R package (such as **labelled**[14] or **sjlabelled**[15]) to attach variable and value labels, or use software like Stata or SPSS to attach these labels. While variable and value labels are not required for BeSD-TI to run, attaching these attributes is strongly recommended to improve the interpretability of BeSD-TI output.

Parameter Files

The user has the option to provide a file listing strata or subgroups for which results should be summarized. This is referred to as an output layout file.

Summary

The following table summarizes the files needed to conduct different types of BeSD analyses using BeSD-TI. Input datasets and parameter files may be saved in any of the following formats: .rds, .dta, or .csv.

Table 2-1. File inputs for BeSD-TI analyses

	BeSD of Childhood Vaccination	**BeSD of COVID-19 Vaccination**
BeSD Child Survey Data	Required	N/A
BeSD COVID-19 Survey Data	N/A	Required
Cluster Metadata	Optional	Optional
Output Layout	Optional	Optional

[13] https://haven.tidyverse.org/

[14] https://larmarange.github.io/labelled/

[15] https://strengejacke.github.io/sjlabelled/index.html

2.5 Files Produced by BeSD-TI

BeSD-TI can produce several types of files:

Analysis Datasets

For each indicator, there is usually an intermediate analysis dataset (flat file) produced that includes only the variables required for that indicator. The analysis file usually includes elements from datasets, along with *derived variables* that BeSD-TI calculates from the survey data and uses to calculate the indicators.

Output Databases

Most indicators produce one or more flat files called *databases* documenting indicator outcomes. These databases include one row of output per row of the output layout file. Databases are saved as R datasets and are suitable for importing by other programs. They could be used for later calculations or to tabulate or graph results in a way that is not supported by BeSD-TI. The database files have the word *_database* in their filenames. Later sections of this document list the databases saved by each indicator.

Tabulated Output

BeSD-TI saves tabulated output in an Excel file, generating one or more tabs (worksheets) per indicator. The output includes one row per row of the output layout file. It is formatted and ready to be copied and pasted into project reports. The user controls which strata appear in the tables and in what order (see Annex A). Typically, a BeSD-TI control program analyzes only one sort of survey data and produces only one Excel output file. If a survey asked questions about BeSD of childhood vaccination *and* COVID-19 vaccination, those data would be analyzed using at least two separate control programs and the outputs for each portion of the survey would be saved in a different Excel workbook.

Graphic Output

Some BeSD-TI indicators generate figures. The control program includes options so the user can stipulate whether the program should make any figures at all, and if so, which types. As a rule, the same strata that appear in the tabulated output also appear in the figures. Each figure is saved as a portable network graphics file (extension .png). The control program has options to save the data underlying the figures as .rds datasets, to facilitate recreating and modifying the figures using R.

Report Template

BeSD-TI can compile a Microsoft Word document that can serve as the basis for a survey report. The report template contains contextual information about BeSD and BeSD-TI, placeholders for further details about the survey to be inserted, and tables and figures produced during the BeSD-TI run. The control program includes options to stipulate whether the report template should be produced and what the title of the report should be.

2.6 Program Progress Log

Every BeSD-TI session generates a log file with messages to document the user's inputs and inform the user which programs were used and whether their progress was successful or if they issued errors or warnings. While BeSD-TI is running, the log entries are stored in a .csv file. The many BeSD-TI functions append new comments onto the dataset throughout the run. When BeSD-TI exits, the log entries are copied into the output Excel workbook in a sheet named "Log".

BeSD-TI users should look at the Log tab in the spreadsheet before focusing on other output. Errors are shaded red and warnings are yellow and all errors and warnings appear at the top of the Log tab. Errors

typically must be addressed and BeSD-TI must be re-run. Warnings do not require you to re-run BeSD-TI but they are messages important enough to be brought to your attention before you interpret BeSD-TI output.

As far as most BeSD-TI users are concerned, the only portion of the log that is of interest is whether there are errors, and if so, how to correct them. The many hundreds of other lines in the log are useful to BeSD-TI developers for debugging problems. You will not need to interpret them, but you may be asked to e-mail your log to the BeSD-TI developers if you have difficulty with a BeSD-TI analysis.
The error messages are meant to be worded in a clear enough manner to help you correct the problem. If the messages are not clear, please send feedback to Caitlin.Clary@biostatglobal.com.

2.7 Structure of BeSD-TI Control Programs

Users should copy and edit the control programs that are provided with BeSD-TI. It is good practice to use a different control program for each analysis and save the control program and resulting output for later reference.

BeSD-TI control programs consists of seven sections or blocks of code. There are four sections that the user *should not edit* and three that they *should edit*. Chapter 7 lists example control programs, line by line, and describes what they do.

Table 2-2. BeSD-TI control programs consist of seven blocks of code

Block of R Code	User Edits Code in this Block?
A. Initialize BeSD-TI run – clean out old data, programs, and global values	✗ No
B. User specifies input and output folders and a name for this analysis	✓ Yes
C. Open the log file & document which version of BeSD-TI programs are running	✗ No
D. User specifies datasets and metadata about survey, schedule and analysis	✓ Yes
E. BeSD-TI checks inputs; pre-processes analysis dataset	✗ No
F. User specifies which indicators to calculate, and any required inputs	✓ Yes
G. BeSD-TI closes log, deletes temporary files, informs the user of any errors	✗ No

2.8 Specifying BeSD-TI Input Parameters

In the remainder of this guide there is a lot of specific guidance that describes BeSD-TI input parameters that you specify in the control program. Parameters are specified by creating objects in the R environment.

The syntax to specify a parameter looks like this:

```
PARAMETER_NAME <- parameter_value
```

Or

```
besd_global(PARAMETER_NAME, parameter_value)
```

R programmers will be familiar with the usual <- assignment process to create objects in the R environment. BeSD-TI sometimes uses this familiar method to establish parameter values but it usually uses the function named `besd_global` which accomplishes two purposes: 1) it creates an object called PARAMETER_NAME to hold the value `parameter_value`, and 2) it writes a message in the BeSD-TI log listing the name of the object and its updated value. Please note that BeSD-TI user-specified parameters are always listed in upper case letters and the user may not change their names. The names are fixed and hard-coded. In most cases the allowable values are also fixed and described in this User's Guide. If you specify a value that is not allowed, BeSD-TI will issue a clear error message. Although it is possible to edit code in blocks B, D, and F, the user should not change the syntax of `besd_global` calls or the names of parameters. They should only edit the parameter values. In many cases, to turn on a feature, the user will specify a value of 1. For instance, we will see later that to indicate to BeSD-TI that plots should be generated in this run, you specify:

```
besd_global(MAKE_PLOTS, 1)
```

and to tell BeSD-TI not to make plots , you specify:

```
besd_global(MAKE_PLOTS, 0)
```

Only those two values are allowed for MAKE_PLOTS: 1 or 0, which correspond conceptually to YES and NO. When we say that the user may edit Block D, we mean that the user may turn various parameters on or off or select different values for some categorical parameters. But the user should not try to change the names of any BeSD-TI parameters.

2.9 Multilingual Output

BeSD-TI contains options to produce output[16] in English, Spanish, French, or Portuguese. The output language is specified in Block D of the control program.

Throughout the template control program, you will see uses of the `language_string` function, which is used to look up particular phrases in a reference database of phrases in the core languages that BeSD-TI supports. Many of these phrases have been machine translated, so users may encounter errors. If you identify an error, please send an email to the BeSD-TI developers at Caitlin.Clary@biostatglobal.com so the translations can be improved in future releases.

Users may wish to modify the multilingual phrases database for a number of reasons – to fix an error, to modify the wording of a particular phrase, or to add translations into another language. To create and use a customized version of the multilingual phrases database, users should:

1. Create a copy of the multilingual phrases database. This can be done by either:
    a. Downloading the database from the package repository on GitHub[17]

[16] Note that there is not currently an option to translate text in the report template, though this is an enhancement which is being considered for future versions of BeSD-TI.

[17] Download the Excel file called "Multi-Lingual Phrases - En Fr Es Pt - with BeSD-TI.xlsx" in the inst/extdata/_extensions subfolder:
https://github.com/BiostatGlobalConsulting/BeSDTI/tree/main/inst/extdata/_extensions

b. With the BeSD-TI package loaded, running the following command in R to create a copy of the multilingual phrases file in a folder of your choosing:

```
besd_copy_multilingual(folder = "C:/Users/Me/BeSD-TI Work")
```

2. Modify your copy of the phrases database. **Do not** modify the contents of Column A (string_name) for any existing phrases—the keys in that column are used in the BeSD-TI code to identify unique phrases. Users can change the translations in the existing language columns or, if desired, add additional phrases and translations in new rows.

   To add translations into a new language, create a new column in the Excel file, add the name of the language in the first cell in that column, and begin adding translations for all rows. For example, the screenshot below illustrates the initial steps of adding translations into Esperanto:

string_name	English	Spanish	French	Portuguese	Esperanto
OS_1	(%)	(%)	(%)	(%)	(%)
OS_2	(N)	(N)	(N)	(N)	(N)
OS_3	(Weighted N)	(N ponderado)	(N pondéré)	(N ponderado)	(Pesita N)
OS_4	95% CI (%)	IC del 95(%)	IC 95 % (%)	IC de 95% (%)	95% KI (%)

   Save your modified version of the phrases database and note the location where the file is saved, since you'll specify the path to this file in the BeSD-TI control program.

3. In Block D of the control program, change the line that reads

   ```
   besd_global(LANGUAGE_FILE, "default")
   ```

   to instead contain the file path to your customized version of the multilingual phrases Excel file:

   ```
   besd_global(
      LANGUAGE_FILE,
      "C:/Users/Me/BeSD-TI Work/BeSD-TI Multilingual Custom.xlsx")
   ```

   If you have added translations into a new language, you can specify that language in the OUTPUT_LANGUAGE global in Block D. Make sure that what you specify for this global *exactly matches* the name of the language you entered in the Excel spreadsheet. For example, in the screenshot above the language Esperanto has been added to the spreadsheet; to use that language in BeSD-TI, the user should specify:

   ```
   besd_global(OUTPUT_LANGUAGE, "Esperanto")
   ```

Note that BeSD-TI uses variable labels and value labels from the input dataset to populate various pieces of text in figures and tables. To avoid language inconsistencies in output, users should ensure that variable and value labels for any variables being analyzed are written in the desired output language.

Chapter 3. Analyzing Childhood Vaccination BeSD Surveys

To analyze survey data about the BeSD of childhood vaccination, you will use a dedicated control program *copied* from an example control program and *modified* to fit your survey and dataset. Start with a childhood vaccination control program template, which you can find in the Control_Programs folder of the BeSD-TI GitHub repository.[18] BeSD-TI currently calculates these indicators from childhood vaccination BeSD surveys:

- One indicator summarizing *core* BeSD questions regarding childhood vaccination (besd_ch_core)
- Two indicators summarizing multiple choice questions from a BeSD survey
    - DESC_02 for summarizing questions where the respondent could select *one* response
    - DESC_03 for summarizing questions where the respondent could select all responses that applied

3.1 Dataset File Names (Block D)

The first lines of code in Block D tell BeSD-TI the names of input datasets:

```
# ************************************************
# Code Block D                (User may change) ----

# Specify dataset names and important metadata. Datasets should be saved in
# DATA_FOLDER. File names should include file extensions; accepted file
# types: .rds, .dta, .csv

# Name of dataset that holds childhood vaccination BeSD data
besd_global(CH_DATASET, "BeSD_Child_Data_Faux_2024-09-19.rds")

# Name of dataset that holds cluster metadata (optional)
besd_global(CM_DATASET, "BeSD_CM_Data_Faux_2024-09-19.rds")
```

Only the survey dataset (called CH_DATASET for this childhood vaccination analysis) is required for this analysis. If data were collected in a cluster survey, it is good practice to include information about each cluster in a cluster metadata (CM) dataset and include that dataset as well.

Variables from datasets specified here may be used as stratifiers for BeSD-TI output.

There is no need to specify a full file path when specifying these dataset names – just specify the file name. All the input datasets must be present in the DATA_FOLDER that was named in Block B. BeSD-TI can use .dta, .rds, and .csv input files; the file extension must be included when defining the dataset names in Block D so that the appropriate package can be used to read those files. Please note that dataset names are case sensitive.

To purposefully omit a file name, simply delete the file name in the template and replace it with NULL. For example, to omit the CM dataset, use a line like this:

```
besd_global(CM_DATASET, NULL)
```

[18] https://github.com/BiostatGlobalConsulting/BeSDTI

(You could also comment out this line. Alternatively, you could omit the line entirely, but that might make it difficult to remember the keywords you need to type if you add a CM dataset to the analysis later.)

3.2 Analysis Metadata and Options

Defining the Rows in Output Tables

Most of the indicators generate tables that present results for various user-specified sub-groups of survey respondents. Those groups may be a flexible combination of geographic domains and/or demographic groups. See Annex A for details on how to use the OUTPUT_VARLIST and OUTPUT_LAYOUT globals to control which strata and groups appear in tables.

Options for Individual Indicators (goes in Block F)

The user specifies a title, subtitle, and as many footnotes as they like for the Excel worksheet that holds the indicator output. These are specified using global values in the control program. For example, the following code specifies globals for a DESC_02 indicator: a title, an empty subtitle, and one additional footnote to be added as footnote #4 because DESC_02 assigns footnotes #1 to #3 automatically.

```
besd_global(DESC_02_TO_TITLE,
            "Has your child had none, some, or all of the vaccines in
             the schedule?")

besd_global(DESC_02_TO_SUBTITLE, NA)

besd_global(DESC_02_TO_FOOTNOTE_4,
           "Responses to this question were not validated by examining
           the child's vaccination record.")
```

Footnotes are numbered *sequentially*, and you may specify as many footnotes for a single measure as you wish. It is important not to skip any numbers. Some indicators include automatic footnotes, based on user inputs; information about automatic footnotes is laid out in Chapter 5. If an indicator does *not* include automatic footnotes, then begin with footnote 1 and increase by 1 up to the number you wish to list. If an indicator *does* include automatic footnotes, then begin adding custom footnotes with the integer that follows the number of automatic footnotes (as in the example above – there are three automatic footnotes for DESC_02, so custom footnotes begin at number 4).

If you skip a number when specifying footnotes (e.g., 1, 2, 4, and 5) then BeSD-TI will only list the footnotes from before the break (i.e., 1 and 2).

Particular indicators will require particular user inputs to specify precisely what to analyze and how. For more details. see the individual descriptions of the indicators in later sections of this document.

Chapter 4. Analyzing COVID-19 Vaccination BeSD Surveys

To analyze survey data about the BeSD of COVID-19 vaccination, you will use a dedicated control program *copied* from an example control program and *modified* to fit your survey and dataset. Start with a COVID-19 vaccination control program template, which you can find in the Control_Programs folder of the BeSD-TI GitHub repository.[19] BeSD-TI currently calculates these indicators from COVID-19 vaccination BeSD surveys:

- One indicator summarizing *core* BeSD questions regarding COVID-19 vaccination (besd_covid_core)
- Two indicators summarizing multiple choice questions from a BeSD survey
    - DESC_02 for summarizing questions where the respondent could select *one* response
    - DESC_03 for summarizing questions where the respondent could select all responses that applied

3.1 Dataset File Names (Block D)

The first lines of code in Block D tell BeSD-TI the names of input datasets:

```
# ************************************************
# Code Block D                (User may change) ----

# Specify dataset names and important metadata. Datasets should be saved in
# DATA_FOLDER. File names should include file extensions; accepted file
# types: .rds, .dta, .csv

# Name of dataset that holds COVID-19 vaccination BeSD data
besd_global(COV_DATASET, "BeSD_COVID_Data_Faux_2024-11-14.rds")

# Name of dataset that holds cluster metadata (optional)
besd_global(CM_DATASET, "BeSD_CM_COVID_Data_Faux_2024-09-19.rds")
```

Only the survey dataset (called COV_DATASET for this COVID-19 vaccination analysis) is required for this analysis. If data were collected in a cluster survey, it is good practice to include information about each cluster in a cluster metadata (CM) dataset and include that dataset as well.

Variables from datasets specified here may be used as stratifiers for BeSD-TI output.

There is no need to specify a full file path when specifying these dataset names – just specify the file name. All the input datasets must be present in the DATA_FOLDER that was named in Block B. BeSD-TI can use .dta, .rds, and .csv input files; the file extension must be included when defining the dataset names in Block D so that the appropriate package can be used to read those files. Please note that dataset names are case sensitive.

To purposefully omit a file name, simply delete the file name in the template and replace it with NULL. For example, to omit the CM dataset, use a line like this:

```
besd_global(CM_DATASET, NULL)
```

[19] https://github.com/BiostatGlobalConsulting/BeSDTI

(You could also comment out this line. Alternatively, you could omit the line entirely, but that might make it difficult to remember the keywords you need to type if you add a CM dataset to the analysis later.)

```
# Surveys about the BeSD of COVID-19 vaccination are sometimes
# administered to health workers and sometimes administered to a
# general population of adults. Specify the respondent type for this
# survey. Valid values are 'Health Workers' or 'Adults'
besd_global(COV_SURVEY_RESPONDENTS, "Health Workers")
```

Block D also specifies the type of respondents for the COVID-19 survey being analyzed; "Adults" to indicate that the survey was administered to a general population of adults or "Health Workers" to signify that the survey respondents were health workers.

3.2 Analysis Metadata and Options

Defining the Rows in Output Tables

Most of the indicators generate tables that present results for various user-specified sub-groups of survey respondents. Those groups may be a flexible combination of geographic domains and/or demographic groups. See Annex A for details on how to use the OUTPUT_VARLIST and OUTPUT_LAYOUT globals to control which strata and groups appear in tables.

Options for Individual Indicators (goes in Block F)

The user specifies a title, subtitle, and as many footnotes as they like for the Excel worksheet that holds the indicator output. These are specified using global values in the control program. For example, the following code specifies globals for a DESC_02 indicator: a title, an empty subtitle, and one additional footnote to be added as footnote #4 because DESC_02 assigns footnotes #1 to #3 automatically.

```
besd_global(DESC_02_TO_TITLE,
            "How concerned are you about getting COVID-19?")

besd_global(DESC_02_TO_SUBTITLE, NA)

besd_global(DESC_02_TO_FOOTNOTE_4,
           "This question was asked in a survey administered between
            April and June 2021")
```

Footnotes are numbered *sequentially*, and you may specify as many footnotes for a single measure as you wish. It is important not to skip any numbers. Some indicators include automatic footnotes, based on user inputs; information about automatic footnotes is laid out in Chapter 5. If an indicator does *not* include automatic footnotes, then begin with footnote 1 and increase by 1 up to the number you wish to list. If an indicator *does* include automatic footnotes, then begin adding custom footnotes with the integer that follows the number of automatic footnotes (as in the example above – there are three automatic footnotes for DESC_02, so custom footnotes begin at number 4).

If you skip a number when specifying footnotes (e.g., 1, 2, 4, and 5) then BeSD-TI will only list the footnotes from before the break (i.e., 1 and 2).

Particular indicators will require particular user inputs to specify precisely what to analyze and how. For more details. see the individual descriptions of the indicators in later sections of this document.

Chapter 5. BeSD-TI Indicator Descriptions and Options

The following pages list the individual indicators that are available in BeSD-TI. Each contains an overview, a list and description of required global value inputs (if any), and a short list of outputs that the software generates. The BeSD-TI files that you download include examples of control programs to run each of these indicators.

5.1 Weighted and unweighted analyses

Users have the choice to calculate indicators using survey weights or to perform unweighted analyses. If the BeSD data were collected as part of a survey from a probability sample that includes survey weights, then incorporate those weights into the analysis. If the data were collected from a convenience sample or a sample that is not meant to be representative of a broader population, then do an unweighted analysis.

BeSD-TI does not currently provide estimates of sampling error for unweighted analyses. The estimate is a description of a proportion observed in the sample and is reported without an estimate of uncertainty.

5.2 Analysis Counter

Block F of the control program sets a global value named ANALYSIS_COUNTER. It is required, and usually set to 1. In most control programs it will only be set once and never changed.

In the remainder of this chapter, you will note that the analysis counter appears in the names of many BeSD-TI output files and worksheets.

In advanced analyses, the user can run an indicator repeatedly (for sensitivity analyses or to produce output with different properties) by running an initial analysis and then changing some of the analysis parameters, changing the analysis counter, and re-running the indicator. In the first run, the output files and tabs would list the value 1 for ANALYSIS_COUNTER and in the second run they would be named with the value 2 and would therefore not overwrite the first set of output. This can be accomplished in a single control program.

For example, analysts might wish to produce several versions of a core indicator bar plot. With ANALYSIS_COUNTER set to 1 they might specify parameters for the core indicator program to produce a plot with no stratification:

```
# Intervening code from Block F omitted
besd_global(BESD_CORE_PLOT_STRATIFIER, NA)

besd_ch_core(cleanup = TRUE)
```

Then they might set ANALYSIS_COUNTER to 2 and specify that the core plot should be stratified by urban/rural areas:

```
besd_global(ANALYSIS_COUNTER, 2)

# Intervening code from Block F omitted
```

```
besd_global(BESD_CORE_PLOT_STRATIFIER, "Area")

besd_ch_core(cleanup = TRUE)
```

And then set ANALYSIS_COUNTER to 3 and stratify the plot by the child's gender:

```
besd_global(ANALYSIS_COUNTER, 3)

# Intervening code from Block F omitted
besd_global(BESD_CORE_PLOT_STRATIFIER, "CHI_chigender")

besd_ch_core(cleanup = TRUE)
```

This analysis would produce three versions of the core indicator bar plot, all saved with a different value of ANALYSIS_COUNTER in the file name.

This same sensitivity analysis could be accomplished using different CONTROL programs that send output to different folders. In that case, there is no need to change the value of ANALYSIS_COUNTER.

5.3 besd_ch_core: Core Indicators for BeSD of Childhood Vaccination

Description: Summarize five core indicator questions from a childhood vaccination BeSD survey

Weighted: Yes or No – the user decides

Denominator: Depends on user selections:

Table 5-1. Denominator definitions for besd_ch_core

Weighted	Denominator	Denominator Description
No	Responded	Number of respondents who answered the question
No	All	Number of all respondents
Yes	All	Sum of weights for all respondents

Numerator: Depends on user selections:

Table 5-2. Numerator definitions for besd_ch_core

Weighted	Numerator Description
No	Number of respondents who selected a particular choice
Yes	Sum of weights for respondents who selected that choice

User inputs:

```
besd_global(BESD_CORE_WEIGHTED, "YES")
besd_global(BESD_CORE_DENOMINATOR, "ALL")
```

Define how core outcomes should be reported in tables
1 = show dichotomized responses in tables
2 = show all response options to core indicators in tables
3 = show all response options *with subtotals* for dichotomized categories

```
besd_global(BESD_CORE_TABLE_STRUCTURE, 1)
```

Optional inputs:

Should bar plots show separate bars for respondents in different categories, e.g. male vs. female? If so, specify the variable to stratify by:

```
besd_global(BESD_CORE_PLOT_STRATIFIER, "CHI_chigender")
```

Customize colors for the core indicator bar plot if desired. Provide one color (a valid R color or hex code) if BESD_CORE_PLOT_STRATIFIER is not specified, or a color for each level of the variable specified in BESD_CORE_PLOT_STRATIFIER.

```
besd_global(BESD_CORE_PLOT_COLORS,
            c("#3c701d", "aliceblue"))
```

Add a caption to the core bar plot.

```
besd_global(BESD_CORE_PLOT_CAPTION, "Custom caption")
```

Include headers in the core indicator bar plot specifying which construct each indicator is measuring? Set to 1 to show headers (recommended).

```
besd_global(BESD_CORE_PLOT_SHOW_HEADERS, 1)
```

Suppress bars in the core indicator bar plot when sample size is under a particular threshold? If so, specify the threshold in this global.

```
besd_global(BESD_CORE_PLOT_SUPPRESS_LOW_N, 25)
```

Control Program Command:

```
besd_ch_core(cleanup = TRUE)
```

Output: This indicator produces a database for each core indicator:

- DESC_02_<analysis counter>_afford_database.rds
- DESC_02_<analysis counter>_confb_database.rds
- DESC_02_<analysis counter>_intent_database.rds
- DESC_02_<analysis counter>_normf_database.rds
- DESC_02_<analysis counter>_where_database.rds

Five tabs in the Excel worksheet are produced by this indicator, one for each core indicator.

One plot is produced, summarizing all five core indicators in a standardized format. The plot is saved as: Plots_CORE/BeSD_Core_Child_<analysis counter>.png

5.4 besd_covid_core: Core Indicators for BeSD of COVID-19 Vaccination

Description: Summarize five core indicator questions from a COVID-19 vaccination BeSD survey

Weighted: Yes or No – the user decides

Denominator: Depends on user selections:

Table 5-3. Denominator definitions for besd_covid_core

Weighted	Denominator	Denominator Description
No	Responded	Number of respondents who answered the question
No	All	Number of all respondents
Yes	All	Sum of weights for all respondents

Numerator: Depends on user selections:

Table 5-4. Numerator definitions for besd_covid_core

Weighted	Numerator Description
No	Number of respondents who selected a particular choice
Yes	Sum of weights for respondents who selected that choice

User inputs:

```
besd_global(BESD_CORE_WEIGHTED, "YES")
besd_global(BESD_CORE_DENOMINATOR, "ALL")
```

Define how core outcomes should be reported in tables
1 = show dichotomized responses in tables
2 = show all response options to core indicators in tables
3 = show all response options *with subtotals* for dichotomized categories

```
besd_global(BESD_CORE_TABLE_STRUCTURE, 1)
```

Optional inputs:

Should bar plots show separate bars for respondents in different categories, e.g. male vs. female? If so, specify the variable to stratify by:

```
besd_global(BESD_CORE_PLOT_STRATIFIER, "Area")
```

Customize colors for the core indicator bar plot if desired. Provide one color (a valid R color or hex code) if BESD_CORE_PLOT_STRATIFIER is not specified, or a color for each level of the variable specified in BESD_CORE_PLOT_STRATIFIER.

```
besd_global(BESD_CORE_PLOT_COLORS,
            c("red", "#251966"))
```

Add a caption to the core bar plot.

```
besd_global(BESD_CORE_PLOT_CAPTION, "Custom caption")
```

Include headers in the core indicator bar plot specifying which construct each indicator is measuring? Set to 1 to show headers (recommended).

```
besd_global(BESD_CORE_PLOT_SHOW_HEADERS, 1)
```

Suppress bars in the core indicator bar plot when sample size is under a particular threshold? If so, specify the threshold in this global.

```
besd_global(BESD_CORE_PLOT_SUPPRESS_LOW_N, 25)
```

Control Program Command:

```
besd_covid_core(cleanup = TRUE)
```

Output: This indicator produces a database for each core indicator:

- DESC_02_<analysis counter>_afford_database.rds
- DESC_02_<analysis counter>_confb_database.rds
- DESC_02_<analysis counter>_intent_database.rds
- DESC_02_<analysis counter>_normf_database.rds
- DESC_02_<analysis counter>_where_database.rds

Five tabs in the Excel worksheet are produced by this indicator, one for each core indicator.

One plot is produced, summarizing all five core indicators in a standardized format. The plot is saved as: Plots_CORE/BeSD_Core_COVID_<analysis counter>.png

5.5 DESC: Descriptive Indicators Summarizing Multiple Choice Questions

DESC_02: Response to multiple-choice question (e.g., education, occupation, etc.)

Description: Most BeSD questions are multiple-choice questions which should be summarized in the survey report.

DESC_02 summarizes responses to questions where the respondent must *select only one response*, and the responses are saved in a single outcome variable. The variable is usually saved as an integer with a value label to describe the response option.

To summarize responses to questions where the respondent can select more than one response, see DESC_03.

Weighted: Yes or No – the user decides

Denominator: Depends on user selections:

Table 5-5. Denominator definitions for DESC_02

DESC_02_WEIGHTED	DESC_02_DENOMINATOR	Denominator Description
No	Responded	Number of respondents who answered the question
No	All	Number of all respondents
Yes	All	Sum of weights for all respondents

Numerator: Depends on user selections:

Table 5-6. Numerator definitions for DESC_02

DESC_02_WEIGHTED	Numerator Description
No	Number of respondents who selected a particular choice
Yes	Sum of weights for respondents who selected that choice

User inputs: For each requested table, the user specifies

Required:

```
besd_global(DESC_02_DATASET, "dataset_name.rds")
```

Often the DESC_02 dataset is a version of the survey dataset augmented with BeSD-TI identifier variables. In a child vaccination BeSD analysis this dataset is named CH_with_ids.rds. In a COVID-19 vaccination BeSD analysis this dataset is named COV_with_ids.rds.

Specify the name of the variable to summarize:

```
besd_global(DESC_02_VARIABLES, "var_to_summarize")
```

Or to summarize multiple variables using the same analysis settings, specify multiple variable names in a concatenate, c():

```
besd_global(DESC_02_VARIABLES,
            c("var1_to_summarize", "var2_to_summarize"))[20]

besd_global(DESC_02_WEIGHTED, "YES" or "NO")[21]
besd_global(DESC_02_DENOMINATOR, "ALL" or "RESPONDED")[22]
```

If the user asks for weighted output then the denominator must be ALL. The estimated proportion for each response will be weighted and will be accompanied by a confidence interval. If the user asks for unweighted output, the user can stipulate that the denominator should be ALL respondents, or only those who RESPONDED to the question (response is not missing).

Optional:

Several inputs are optional. If you wish to override the label for one of the response options, you may do so using the "RELABEL" options. If you want to report subtotals of response options, you may do so using the "SUBTOTAL" options. To work correctly, the items indicated in <angle brackets> in Table 5-7 should be replaced with integers.

[20] If more than one variable is listed in DESC_02_VARIABLES, a table will be made for each variable and all the currently defined DESC_02 global variables will be applied to all of those tables; if you wish to make tables using different options (e.g., some weighted and some not) then run DESC_02 once with the WEIGHTED option and then turn that option off and run DESC_02 again.

[21] Many BeSD-TI globals are set to 1 to indicate YES and 0 to indicate NO. This global is an exception; use the words YES or NO.

[22] If WEIGHTED is YES then DENOMINATOR must be ALL.

Table 5-7. Optional inputs for DESC_02

Optional Globals	Description	Values	Notes
DESC_02_TO_TITLE	Table title	Text	*Default value: variable label if available, variable name if not*
DESC_02_TO_SUBTITLE	Table subtitle	Text	
DESC_02_N_LABEL	New label for N column	Text	*Default value: N*
DESC_02_NTWD_LABEL	New label for Weighted N column	Text	*Default value: Weighted N*
DESC_02_N_RELABEL_LEVELS	Number of replacement labels	Integer	
DESC_02_RELABEL_LEVEL_<1 up to the N_RELABEL_LEVELS>	Response value that new label will apply to	Integer	
DESC_02_RELABEL_LABEL_<1 up to the N_RELABEL_LEVELS>	New value label	Text	
DESC_02_N_SUBTOTALS	Number of response groups user would like to create for variable	Integer	
DESC_02_SUBTOTAL_LEVELS_<1 up to the N_SUBTOTALS>	Response values that will be grouped together	List of Integers	
DESC_02_SUBTOTAL_LABEL_<1 up to the N_SUBTOTALS>	New value label for grouped responses	Text	
DESC_02_SUBTOTAL_LIST_<1 up to the N_SUBTOTALS>	Placement of DESC_02_SUBTOTAL_LABEL_<1 up to the N_SUBTOTALS> within table.	Contains the word "before" or "after" and the appropriate variable value (e.g., `after 4` would put the subtotal column after the column for the value #4.	*Default value: At end of table* If global DESC_02_SHOW_SUBTOTALS_ONLY is set, this global is ignored.
DESC_02_SHOW_SUBTOTALS_ONLY	When set to yes will only output the subtotals requested in the table.	Yes or blank	Subtotals will show in order in which they are requested.
DESC_02_LIST_N_BEFORE_PCT	Lists N and percent for each variable value.	Yes or blank	If not selected only percent will appear in table.
DESC_02_LIST_NWTD_BEFORE_PCT	Lists Weighted N and percent for each variable value.	Yes or blank	If not selected only percent will appear in table.
DESC_02_TO_FOOTNOTE_<4 up to the total number of desired footnotes>	Additional footnote(s) for table	Text	Start with FOOTNOTE_4 as FOOTNOTE_1 and _2 and _3 are hard coded in program.

The RELABEL options are implemented to give the user control over table column labels. By default, BeSD-TI uses the variable's value labels as column headers. In many cases those will be appropriate, but one common use for the RELABEL options is to assign text

labels to the column header for missing values. In other cases the options are used to specify a label that differs from the value label. The user can replace the label for any number of levels. Levels that are not specified with RELABEL options will use the value label. If the user wishes to specify a label for a single level, set N_RELABEL_LEVELS to 1 and identify which level. A common option will be to specify the missing level denoted in R with *NA* (see example below). If the user wishes to specify revised labels for other levels, then supply those as well.

In the example below, the value label for the 3 has a value label that says "Other, please specify". The user wishes to shorten this label to "3: Other". The user also wants to label the column of missing values with the word "Missing".

```
besd_global(DESC_02_N_RELABEL_LEVELS, 2)

besd_global(DESC_02_RELABEL_LEVEL_1, 3)
besd_global(DESC_02_RELABEL_LABEL_1, "3: Other ")

besd_global(DESC_02_RELABEL_LEVEL_2, NA)
besd_global(DESC_02_RELABEL_LABEL_2, "Missing ")
```

When the DENOMINATOR is set to RESPONDED, BeSD-TI will not list missing as a level in the output table – it assumes that you are not interested in tabulating missing responses. But the RELABEL options may still be used to re-label non-missing responses, like 3 in the example above.

When the DENOMINATOR is set to ALL, it will be a good idea to specify a LABEL for missing values (NA). Otherwise, the output table may include a column header that says "NA (%) ", which may confuse some readers who are unfamiliar with the convention of using NA to represent missing values.

A second goal is accomplished with optional inputs to specify subtotals. Tell BeSD-TI how many subtotals there will be, and then for each, specify the list of response categories that should be grouped together. Stipulate what label to put at the top of the column and optionally specify where the subtotal should appear from left-to-right in the table. The default is for subtotal columns to appear at the far right of the tabulated output.

```
besd_global(DESC_02_N_SUBTOTALS, 1)
besd_global(DESC_02_SUBTOTAL_LEVELS_1, c(1, 2))
besd_global(DESC_02_SUBTOTAL_LABEL_1, "Subtotal: 1 or 2 ")
```

Sometimes the user may wish to show only subtotal columns. This is possible with the DESC_02_SHOW_SUBTOTALS_ONLY option.

The default order for each output category is to list the column for PCT before the column for N (or NWTD) but the user can ask BeSD-TI to reverse the order with options LIST_N_BEFORE_PCT and LIST_NWTD_BEFORE_PCT.

While most yes/no globals in a BeSD-TI control program are specified using the value 1 for yes and 0 for no (e.g., the global named MAKE_PLOTS must be set to 0 or 1), DESC_02 and _03 have several options that expect to see the words "Yes" or "No" or "All". (Any combination of letter case is fine, but the words must be exactly one of the valid options.)

Control Program Command: DESC_02

Outputs: This indicator makes one database per variable summarized. The file will be named DESC_02_<*analysis counter*>_<DESC 02 counter>.rds. The DESC 02 counter starts at 1 and increases by 1 every additional time DESC_02 is called. The database lists the % for each option along with total N for every stratum at every level. If the user requests weighted results, it reports weighted N and reports a 95% CI for each response level. This indicator makes one Excel worksheet per variable summarized. The worksheet will be named DESC_02_<*DESC_02 counter*>_<*name of variable being summarized*> <*analysis counter*>. If the variable name is longer than 16 characters, the tab name will be shortened to only include the first 16 characters of the variable name due to Excel character limits.

This indicator does not generate any plots.

This indicator generates three footnotes automatically, so user-specified footnotes should begin with DESC_02_TO_FOOTNOTE_4.

Interpretation: Depends on the user's selections:

Table 5-8. Interpretation of DESC_02

DESC_02_WEIGHTED	DESC_02_DENOMINATOR	Interpretation
Unweighted	All	"Among the N respondents, X% selected this response option."
Unweighted	Responded	"Among the N respondents who answered the question, X% selected this response option."
Weighted	All	"X% of eligible respondents in the population are estimated to be in the category of person who would select this response option."

More Notes:

It is possible to summarize the responses to numerous questions. After setting the input global macros and calling DESC_02, simply change the DESC_02 global macros and call DESC_02 again. The output for each call is summarized in a new tab in the Excel worksheet. See the example control programs that you downloaded with BeSD-TI for examples.

Note that it is *not necessary* to change the ANALYSIS_COUNTER when calling DESC_02 multiple times.

DESC_03: Response to multiple-choice question (e.g., education, occupation, etc.)

Description: Most surveys will include some multiple-choice questions to be summarized in the survey report. This indicator summarizes responses to questions where the respondent may *select more than one response option.* For analysis with DESC_03, responses to the question should be stored across multiple variables, indicating which answers each respondent chose. Below is an example of a question with responses stored in six variables, each representing one response option:

Question: What makes it hard to get vaccination services for your child?

Respondent	Nothing, it's not hard	Getting to the clinic is hard	The clinic opening times are inconvenient	The clinic sometimes turns people away	Waiting time in the clinic takes too long	Something else
1	0	0	1	0	1	1
2	1	0	0	0	0	0
3	0	1	1	0	1	1

To summarize responses to questions where the respondent can select only one response, see DESC_02.

Weighted: Yes or No – the user decides

Denominator: Depends on user selection:

Table 5-9. Denominator definitions for DESC_03

DESC_03_WEIGHTED	DESC_03_DENOMINATOR	Denominator Description
No	Responded	Number of respondents who answered the question
No	All	Number of all respondents
Yes	All	Sum of weights for all respondents

Numerator: Depends on user selection:

Table 5-10. Numerator definitions for DESC_03

DESC_03_WEIGHTED	Numerator Description
No	Number of respondents who selected a particular choice
Yes	Sum of weights for respondent who selected that choice

User inputs: For each requested table, the user specifies

Required:

```
besd_global(DESC_03_DATASET, "dataset_name.rds")
```

Often the DESC_03 dataset is a version of the survey dataset augmented with BeSD-TI ID variables. In a child vaccination BeSD analysis this dataset is named CH_with_ids.rds. In a COVID-19 vaccination BeSD analysis this dataset is named CV_with_ids.rds.

```
besd_global(DESC_03_SHORT_TITLE, "Title for Excel Tab")

besd_global(DESC_03_VARIABLES,
          c("response_variable_1", "response_variable_2"))
```

When specifying DESC_03_VARIABLES, list all the variables where response options for this question are stored. Variable names should be in quotes, inside a concatenate command, c().

```
besd_global(DESC_03_SELECTED VALUE, 1)
```

DESC_03_SELECTED_VALUE defines the value that indicates that the respondent selected the response. The same coding should be used for all the variables listed in DESC_03_VARIABLES. Often this value is 1.

```
besd_global(DESC_03_WEIGHTED, "YES" or "NO")[23]
besd_global(DESC_03_DENOMINATOR, "ALL" or "RESPONDED")[24]
```

Example:

```
besd_global(DESC_03_DATASET, "CH_with_ids.rds")
besd_global(DESC_03_SHORT_TITLE, "Access_Services")
besd_global(DESC_03_VARIABLES, c("CHI_lowacnoth",
        "CHI_lowachard", "CHI_lowacopen", "CHI_lowacaway",
        "CHI_lowactime", "CHI_lowacelse"))
besd_global(DESC_03_SELECTED VALUE, 1)
besd_global(DESC_02_WEIGHTED, "NO")
besd_global(DESC_02_DENOMINATOR, "RESPONDED")
```

If the user asks for weighted output, then the denominator must be ALL. The estimated proportion for each response will be weighted and will be accompanied by a confidence interval. If the user asks for unweighted output, they can stipulate that the denominator should be ALL respondents, or only those who RESPONDED to the question (response is not missing).

Optional:

The user will need to complete the Global Names below with the correct integer value in place of < > text if they wish to utilize the RELABEL_LEVELS or SUBTOTAL functions.

[23] Many BeSD-TI globals are set to 1 to indicate YES and 0 to indicate NO. This global is an exception; use the words YES or NO.

[24] If WEIGHTED is YES then DENOMINATOR must be ALL.

Table 5-11. Optional inputs for DESC_03

Optional Globals	Description	Values	Notes
DESC_03_TO_TITLE	Table title	Text	
DESC_03_TO_SUBTITLE	Table subtitle	Text	
DESC_03_N_LABEL	New label for N column at end of table.	Text	*Default value: N*
DESC_03_NTWD_LABEL	New label for Weighted N column at end of table.	Text	*Default value: Weighted N*
DESC_03_N_RELABEL_LEVELS	Number of replacement labels	Integer	
DESC_03_RELABEL_LEVEL_<1 up to the N_RELABEL_LEVELS>	Variable that new label will apply to	Variable Name	
DESC_03_RELABEL_LABEL_<1 up to the N_RELABEL_LEVELS>	New value label	Text	
DESC_03_N_SUBTOTALS	Number of response groups user would like to create for variable	Integer	
DESC_03_SUBTOTAL_LEVELS_<1 up to the N_SUBTOTALS>	Response values that will be grouped together	List of Variables	
DESC_03_SUBTOTAL_LABEL_<1 up to the N_SUBTOTALS>	New value label for grouped responses	Text	
DESC_03_SUBTOTAL_LIST_<1 up to the N_SUBTOTALS>	Placement of DESC_03_SUBTOTAL_LABEL_<1 up to the N_SUBTOTALS> within table.	Contain the word "before" or "after" and the appropriate variable name; so after	*Default value: At end of table* *If global* DESC_03_SHOW_SUBTOTALS_ONLY is set, this global is ignored.
DESC_03_SHOW_SUBTOTALS_ONLY	When set to yes will only output the subtotals requested in the table.	Yes or blank	Subtotals will show in order in which they are requested.
DESC_03_LIST_N_BEFORE_PCT	Lists N and percent for each variable value.	Yes or blank	If not selected only percent will appear in table.
DESC_03_LIST_NWTD_BEFORE_PCT	Lists Weighted N and percent for each variable value.	Yes or blank	If not selected only percent will appear in table.
DESC_03_TO_FOOTNOTE_<4 up to the total number of desired footnotes>	Additional footnote(s) for table	Text	Start with FOOTNOTE_4 as FOOTNOTE_1 and _2 and _3 are hard coded in program.

Recall that DESC_02 defaults to using the variable label to populate cell A1 of the Excel worksheet, but for DESC_03 (where multiple variables are analyzed and therefore multiple variable labels could apply) the user is asked to *specify* the title for the worksheet.

```
besd_global(DESC_03_TO_TITLE, "What makes it hard to get
vaccination services for your child? ")
```

The “RELABEL” options accomplish the same goals described above in DESC_02: allow the user to over-ride variable labels and specify what label they would like to see in the output table. If the user wishes to specify a label for one level, set N_RELABEL_LEVELS to 1 and identify which level. A common option will be to override the label for “Other, Please specify” and make it simply “Other”. E.g.:

```
besd_global(DESC_03_N_RELABEL_LEVELS, 1)
besd_global(DESC_03_RELABEL_LEVEL_1, "CHI_lowacelse")
besd_global(DESC_03_RELABEL_LABEL_1, "Other")
```

Another goal for optional inputs is to allow the user to identify several response options that should be aggregated into a subtotal. The resulting table will list each individual response and the subtotal. To specify a subtotal, tell BeSD-TI how many subtotals there will be, and then for each, specify the list of response categories that should be grouped together. Specify the label to put at the top of the column and where to list the subtotal column. It is also possible to list only the subtotal columns.

In this example we will consider reasons for dissatisfaction with vaccination services and create one subtotal to summarize the percentage of people who mentioned dissatisfaction with staff by answering ‘yes’ to CHI_qualipoor, CHI_qualirespect, or CHI_qualitime.

```
besd_global(DESC_03_VARIABLES,
            c("CHI_qualinoth", "CHI_qualiavail",
              "CHI_qualiopen", "CHI_qualiwait",
              "CHI_qualiclean", "CHI_qualipoor",
              "CHI_qualirespect", "CHI_qualitime",
              "CHI_qualielse"))

besd_global(DESC_03_N_SUBTOTALS, 1)
besd_global(DESC_03_SUBTOTAL_LEVELS_1, c("CHI_qualipoor",
            "CHI_qualirespect", "CHI_qualitime"))
besd_global(DESC_03_SUBTOTAL_LABEL_1,
            "Subtotal: Problems with staff")
```

To create multiple subtotal categories, use syntax like the following:

```
besd_global(DESC_03_VARIABLES,
            c("varA1", "varA2", "varA3", "varB1", "varB2",
              "varC1", "varC2", "varC3"))

# We will create three subtotal categories:
besd_global(DESC_03_N_SUBTOTALS, 3)

# Settings for subtotal category 1:
besd_global(DESC_03_SUBTOTAL_LEVELS_1,
```

```
            c("varA1", "varA2", "varA3"))
besd_global(DESC_03_SUBTOTAL_LABEL_1,
            "Subtotal: Category A")

# Settings for subtotal category 2:
besd_global(DESC_03_SUBTOTAL_LEVELS_2,
            c("varB1", "varB2"))
besd_global(DESC_03_SUBTOTAL_LABEL_2,
            "Subtotal: Category B")

# Settings for subtotal category 3:
besd_global(DESC_03_SUBTOTAL_LEVELS_3,
            c("varC1", "varC2", "varC3"))
besd_global(DESC_03_SUBTOTAL_LABEL_3,
            "Subtotal: Category C")
```

The default order for each output category is to list the column for PCT before the column for N (or NWTD) but the user can ask BeSD-TI to reverse the order with options LIST_N_BEFORE_PCT and LIST_NWTD_BEFORE_PCT.

> While most yes/no globals in a BeSD-TI control program are specified using the value 1 for yes and 0 for no (e.g., the global named MAKE_PLOTS must be set to 0 or 1), DESC_02 and _03 have several options that expect to see the words “Yes” or “No” or “All”. (Any combination of letter case is fine, but the words must be exactly one of the valid options.)

Control Program Command: DESC_03

Outputs: This indicator makes one database per variable summarized. The file will be named DESC_03_<*analysis counter*>_<*DESC 03 counter*>.rds. The DESC 03 counter starts at 1 and increases by 1 every additional time DESC_03 is called. The database lists the % for each option along with total N for every stratum at every level. It reports weighted N if the user requests weighted results and reports 95% CI for each option if the calculation is weighted.

This indicator makes one Excel worksheet per variable summarized. The worksheet will be named DESC_03_<*DESC_03 counter*>_<*DESC_03_SHORT_TITLE*> <*ANALYSIS_COUNTER*>. If the <*DESC_03_SHORT_TITLE* > string is longer than 16 characters, the tab name will be shortened to only include the first 16 characters of the < *DESC_03_SHORT_TITLE*> due to Excel character limits.

This indicator does not generate any plots.

This indicator generates three footnotes automatically, so user-specified footnotes should begin with DESC_03_TO_FOOTNOTE_4.

Interpretation: Depends on the user's selections:

Table 5-12. Interpretations for DESC_03

DESC_03_WEIGHTED	DESC_03_DENOMINATOR	Interpretation
Unweighted	All	"Among the N respondents, X% selected this response option."
Unweighted	Responded	"Among the N respondents who answered the question, X% selected this response option."
Weighted	All	"X% of eligible respondents in the population are estimated to be in the category of person who would select this response option."

Notes:

This indicator, unlike DESC_02, does summarize responses even if no one selected them. It is possible to summarize the responses to numerous sets of questions. After setting the input global macros and calling DESC_03, simply change the DESC_03 global macros and call DESC_03 again. The output for each call is summarized in a new tab in the Excel worksheet.

For each response option, DESC_03 uses the variable label as the title of the column to summarize those responses. So if the variables being summarized are XY14 and its label is "Apples" and XY15 and its label is "Oranges", then those will form the labels. The variable labels may be over-ridden by the user, using the DESC_03_RELABEL options.

Note that it is *not necessary* to change the ANALYSIS_COUNTER when calling DESC_03 multiple times.

Chapter 6. Examples of Control Programs

As described at the end of Chapter 2, BeSD-TI control programs usually consist of seven blocks of code. Three blocks are edited and customized by the user and the other four blocks are usually not edited but are necessary for the program to run correctly.

This chapter shows examples of each of the seven blocks for a BeSD of childhood vaccination analysis.

6.1 Block A: Start with clear memory

The first block of a BeSD-TI control program loads the BeSD-TI package and clears out old global values. The section below shows the top of a childhood vaccination control program.

```
# BeSD-TI Control Program - Childhood Vaccination - Version 1.00 - 2024-10-16
# Biostat Global Consulting

# Behavioral and Social Drivers of Vaccination: Tools for Inference (BeSD-TI)
# control program to analyze data from a childhood vaccination BeSD survey

# Change log

# Date        Version Number    Name              What Changed
# 2024-10-16  1.00              Caitlin Clary     Original version

# ************************************************
# Code Block A                 (Do not change) ----

# Load the BeSD-TI package
library(BeSDTI, attach.required = TRUE)

# Start with clear memory
cleanup_BeSDTI_globals()
```

6.2 Block B: Specify input/output folders and analysis name

Block B is the first of three sections that the user edits. It consists of three lines of code: the first defines the path to the folder where BeSD-TI will find the survey datasets, the second defines the path to the folder where output files will be saved, and the third gives a name to the analysis. The analysis name will appear in the name of the output spreadsheet. This page shows Block B from a childhood vaccination control program.

```
# ************************************************
# Code Block B                (User may change) ----

DATA_FOLDER <- "Q:/WHO - BeSD support/CBC Working Folder/BeSD Test Data"

OUTPUT_FOLDER <- "Q:/WHO - BeSD support/CBC Working Folder/BeSD Child Vx Test Output"

ANALYSIS_NAME <- "Child_Vx_Test"
```

Enter the full paths to the DATA_FOLDER and OUTPUT_FOLDER here. Note that these two folders cannot be the same.

List a short text string here for the ANALYSIS_NAME. Your tabulated output will be stored in the OUTPUT_FOLDER in a file named <ANALYSIS_NAME>_TO.xlsx

Note: Every time you do a new BeSD-TI analysis:

1. Make a new empty folder.
2. Copy a control program template to that new folder.
3. Edit the control program, setting OUTPUT_FOLDER to be the path to the new folder.
4. Make other changes to Blocks B, D, or F to customize your analysis.
5. Save the control program and run BeSD-TI.

6.3 Block C: Set working directory to output folder and open log

Block C has R change the working directory to be the output folder specified in Block B. Then it deletes any old copies of the Excel output file so this new run will be putting output into a new file. Next it opens the BeSD-TI log, putting some initial messages in there to document the user-inputs that have been specified up to this point.

```
# ************************************************
# Code Block C                 (Do not change) ----

setwd(DATA_FOLDER)

# Start with a clean, empty Excel file for tabulated output (TO)
unlink(paste0(OUTPUT_FOLDER, "/", ANALYSIS_NAME, "_TO.xlsx"), force = TRUE)

# Give the current program a name, for logging purposes
VCP <- "BeSD-TI_ChildVx_Control_Program"

# Open the log and put a comment in it
besd_log_comment(VCP, 3, "Comment", "Run begins...log opened...")

# Document the global macros that were defined before the log opened
besd_log_global(DATA_FOLDER)
besd_log_global(OUTPUT_FOLDER)
besd_log_global(ANALYSIS_NAME)

# Write an entry in the log file documenting the package version
besd_log_comment(VCP, 3, "Package",
                 paste0("BeSD-TI package version ", utils::packageVersion("BeSDTI")))
```

6.4 Block D: Specify dataset names and important metadata

Block D holds the second set of lines that a user typically edits. The user specifies the names of the R/Stata/CSV datasets that hold the survey data. Variable names and coding conventions for those datasets are in BeSD-TI's Forms and Variable List (FVL) document. The user also specifies some parameters or metadata to describe the vaccination schedule, the coverage survey, and some parameters to control what BeSD-TI generates and how it looks.

```
# ***********************************************
# Code Block D                (User may change) ----

# Specify dataset names and important metadata. Datasets should be saved in
# DATA_FOLDER. File names should include file extensions; accepted file types:
# .rds, .dta, .csv

# Name of dataset that holds childhood vaccination BeSD data
besd_global(CH_DATASET, "BeSD_Child_Data_Faux_2024-09-19.rds")

# Name of dataset that holds cluster metadata (optional)
besd_global(CM_DATASET, "BeSD_CM_Data_Faux_2024-09-19.rds")

# Output layout parameters

# The output layout parameters determine the geographic and/or demographic
# strata for which results are displayed in tabular output and plots. Users may
# specify a single stratifier (like urban/rural) or a set of several stratifiers
# (like urban/rural and sex and household wealth). Any variable used as a
# stratifier must be listed in OUTPUT_VARLIST. If you wish to summarize output
# for the entire survey without any stratification, OUTPUT_VARLIST may be left
# undefined.

# If OUTPUT_VARLIST is populated and OUTPUT_LAYOUT is not defined, BeSD-TI will
# generate a default layout for tables and figures. That layout file will be
# saved in the OUTPUT_FOLDER.

# The user may create their own OUTPUT_LAYOUT file defining the conditions,
# preferred order, and row labels for the output strata and point to that layout
# file in the control program, e.g.
# besd_global(OUTPUT_LAYOUT, "Q:/My_BeSD-TI_Output/my_output_layout.rds").
# See the BeSD-TI User's Guide for more details on creating a layout file.

besd_global(OUTPUT_VARLIST, c("Districtname", "Area"))

besd_global(OUTPUT_LAYOUT, paste0(DATA_FOLDER, "/custom_layout.rds"))
```

OUTPUT_VARLIST is always required while OUTPUT_LAYOUT is optional. See Annex A for examples where the user requested output for different geographic/demographic strata.

```
# User specifies survey::svydesign syntax to describe the sample. The data
# argument in survey::svydesign should *not* be specified here.
besd_global(SVYDESIGN_SYNTAX,
            list(ids = ~Clusternum, strata = ~Districtname, weights = ~psweight_ch,
                 fpc = NULL, nest = FALSE))

# User specifies the method for calculating confidence intervals
# Valid choices are "Logit", "Wilson", "Jeffreys" or "Clopper"; our default recommendation is "Wilson"
besd_global(CI_METHOD, "Wilson")

# Specify whether the code should export to excel, or not (usually 1)
besd_global(EXPORT_TO_EXCEL, 1)

# User specifies the number of digits after the decimal place in coverage outcomes
besd_global(NUM_DECIMAL_DIGITS, 1)

# Specify whether a report template should be made using the output of this
# BeSD-TI run. Set to 1 for yes.
besd_global(MAKE_TEMPLATE_REPORT, 1)

# Specify whether the code should make plots, or not (usually 1)
# MAKE_PLOTS must be 1 for any plots to be made
besd_global(MAKE_PLOTS, 1)

# Make bar plots? Set to 1 for yes
besd_global(MAKE_BAR_PLOTS, 1)

# Save the data underlying bar plots? Set to 1 for yes. If this option is turned
# on, inchworm and barplot programs will save a dataset in the corresponding
# plots folder that makes it possible to understand the quantitative details of
# each plot component and can be used to recreate the plot.
besd_global(SAVE_BAR_PLOT_DATA, 1)
```

Set this parameter to 1 to generate tabular output in an Excel file. Set it to 0 if you only wish to make figures or database output.

BeSD-TI usually provides one digit after the decimal point in tabular and graphic output, but sometimes we like to change this to zero (0) to produce compact tables.

If this is set to 1, BeSD-TI will compile a template report Word document using output from this BeSD-TI run.

If this is set to 0, BeSD-TI will not make any plots or figures or charts.

```
# Specify whether the code should delete or save BeSD-TI output databases.
# WARNING!! If this global is set to 1, BeSD-TI will delete ALL files that end
# in _database.rds in the OUTPUT_FOLDER at the end of the run. If you want to
# save the databases, change the value to 0.
besd_global(DELETE_DATABASES_AT_END, 1)

# Specify whether the code should delete intermediate datasets
# at the end of the analysis (Usually 1)
# If you wish to keep them for additional analysis or debugging,
# set the option to 0.
besd_global(DELETE_TEMP_DATASETS, 1)
```

BeSD-TI generates datasets (flat files) of analysis results that it calls *databases*. When this parameter is set to 1, BeSD-TI deletes them because we assume that most users want to look at graphical or tabular output rather than unformatted flat files. If you wish to keep the database files, set this parameter to 0.

This parameter is usually set to 1 so BeSD-TI will delete its temporary datasets when it is finished running. A user might set it to 0 to keep those datasets for the purpose of debugging a program or following along to understand some of BeSD-TI's intermediate work products.

```
# Specify the language for table and figure text. In the default BeSD-TI
# language file, the options are ENGLISH, SPANISH, FRENCH, or PORTUGUESE. Users
# can modify this file to add their own language(s) - for instructions on adding
# a language, see the BeSD-TI User's Guide.
besd_global(OUTPUT_LANGUAGE, "English")

# Specify the file to use when populating table and figure text. Omit this
# global or set it to "default" to use the standard BeSD-TI language file, or
# specify a file path if using a user-modified file.
besd_global(LANGUAGE_FILE, "default")
```

6.5 Block E: Pre-process survey data

Block E should not be changed by the user. This block differs depending on whether a childhood vaccination or COVID-19 vaccination analysis is being conducted.

```
# ***********************************************
# Code Block E                  (Do not change) ----

# ..........................................................
# Check the user's metadata for completeness and correctness
# ..........................................................

# Establish an empty object before starting to record temp dataset list
TEMP_DATASETS <- NULL

check_besd_metadata(analysis = "child")

# ..........................................................
# Establish unique IDs
# ..........................................................

# The name of the dataset coming out of the ID step is CH_with_ids
establish_unique_CH_ids()

# ..........................................................
# Generate derived variables for child BeSD analysis
# ..........................................................
gen_ch_dv()
```

6.6 Block F: Calculate BeSD-TI indicators

Block F is the third and final section that the user edits.

Broadly speaking, there are four steps to run an indicator:

1. Specify required (and optional) inputs via the besd_global function.
2. Specify the title, subtitle, and footnotes for the Excel worksheet that will hold tabular results.
3. Call the function that calculates the indicator and generates output.
4. If you will calculate this indicator again later in the same control program, clear out the input global values so old values are not mistakenly used again.

These steps are quite similar across indicators, but the details of the code in Block F differs substantially depending on the type of survey being analyzed. A few examples are copied below; see the control programs that accompany this guide for full examples of how to run the BeSD-TI indicators.

```
# ************************************************
# Code Block F                (User may change) ----

# Calculate BeSD-TI indicators requested by the user

# First, set the analysis counter. This is a counter that is used to name
# datasets. It is usually set to 1 but the user might change it if requesting
# repeat analyses with differing parameters - see the User's Guide

besd_global(ANALYSIS_COUNTER, 1)

# ------------------------------------------------------------------------
# If compiling a template report, set options before calculating indicators
# ------------------------------------------------------------------------

if (MAKE_TEMPLATE_REPORT == 1){

  # Define a title for the report template
  besd_global(REPORT_TITLE, "BeSD Report: Harmonia 2024")

}
```

```
# ------------------------------------------------------------------------
# Summarize BeSD core indicators for childhood vaccination
# ------------------------------------------------------------------------

besd_global(BESD_CORE_WEIGHTED, "NO")
besd_global(BESD_CORE_DENOMINATOR, "RESPONDED")

# Define how core outcomes should be reported in tables
# 1 = show dichotomized responses in tables
# 2 = show all response options to core indicators in tables
# 3 = show all response options *with subtotals* for dichotomized categories
besd_global(BESD_CORE_TABLE_STRUCTURE, 1)

# Should bar plots show separate bars for respondents in different categories,
# e.g. male vs. female? If so, define the variable for stratification here.
besd_global(BESD_CORE_PLOT_STRATIFIER, "CHI_chigender")

# What color(s) should bars in the bar plot be? Leave this global undefined (NA
# or NULL) to use default colors, provide one bar color if
# BESD_CORE_PLOT_STRATIFIER is undefined, or provide colors for each level of
# the variable specified in BESD_CORE_PLOT_STRATIFIER.
besd_global(BESD_CORE_PLOT_COLORS, NA)

# If desired, specify a caption to add to the core indicator bar plot
# besd_global(BESD_CORE_PLOT_CAPTION, "")

# Add headers to the core indicator bar plot specifying which construct each
# core indicator is measuring? Set to 1 to show headers (recommended)
besd_global(BESD_CORE_PLOT_SHOW_HEADERS, 1)

# Suppress results in core plot when N is under a specified threshold?
besd_global(BESD_CORE_PLOT_SUPPRESS_LOW_N, 25)

besd_ch_core(cleanup = TRUE)
```

```
# ------------------------------------------------------------------------
# Summarize responses to some multiple-choice questions using DESC_02 & DESC_03
# ------------------------------------------------------------------------

# Standard BeSD question:
# Has your child had none, some, or all of the vaccines in the schedule?
besd_global(DESC_02_DATASET, "CH_with_ids.rds")
besd_global(DESC_02_VARIABLES, "CHI_vaccs")
besd_global(DESC_02_WEIGHTED, "NO")
besd_global(DESC_02_DENOMINATOR, "RESPONDED")

besd_global(DESC_02_TO_TITLE, language_string(language_use = language_use, str = "OS_B11"))

besd_global(DESC_02_TO_SUBTITLE, NA) # No subtitle

# Remember that DESC_02 automatically assigns three footnotes, so if you
# want to include another, start with the number 4.
# We are not using it here, but clear it out in case it was used earlier.
besd_global(DESC_02_TO_FOOTNOTE_4, NA)

BESD_DESC_02(cleanup = TRUE, database_id = "vaccs")
```

6.7 Block G: Exit gracefully

Block G concludes and cleans up after the BeSD-TI run. It makes the template report if the user asked for one, then it calls a program that cleans up after BeSD-TI, moving the log file into Excel and deleting temporary files if the user set DELETE_TEMP_DATASETS to 1. The BeSD-TI log is moved into a worksheet of the Excel output file; errors are shaded red and warnings are shaded yellow; and the log is sorted so errors and warnings appear at the top of the log worksheet.

```
# ************************************************
# Code Block G                  (Do not change) ----
#
# Exit gracefully

# Compile report

if (MAKE_TEMPLATE_REPORT == 1){
  besd_report(analysis = "child")
}

# 1. Close the datasets that hold the results of hypothesis tests and put them
#    into the output spreadsheet
# 2. Close the log file and put it into the output spreadsheet
# 3. Clean up extra files and global values
# 4. Send a message to the screen if there are warnings or errors in the log

besd_cleanup()
```

Chapter 7. Troubleshooting BeSD-TI

As mentioned in section 2.1, BeSD-TI traps obvious errors and omissions concerning input parameter values and files and tries to provide helpful error messages. If you see error messages in the R console along with the BeSD-TI billboard (see below) then you can either scroll back through the console to read the error messages or open BeSD-TI's tabulated output spreadsheet. That spreadsheet can be found in the OUTPUT_FOLDER defined in Block B; the name of the spreadsheet depends on the ANALYSIS_NAME defined in Block B, and has the format <ANALYSIS_NAME>_TO.xlsx.

```
======================================================================
     BEHAVIOURAL AND SOCIAL DRIVERS OF VACCINATION: TOOLS FOR IMPACT
======================================================================
                      ____      _____ ____      __________
                     / __ )____ / ___// __ \    /_  __/  _/
                    / __  / _ \\__ \/ / / /_____/ /  / /
                   / /_/ /  __/__/ / /_/ /_____/ / _/ /
                  /_____/\___/____/_____/     /_/ /___/
======================================================================
                        (C) WORLD HEALTH ORGANIZATION
======================================================================
```

If R encounters a hard error and halts without showing the BeSD-TI billboard, you will need to run the command `besd_cleanup()` in the R command prompt to cue BeSD-TI to put the log and its informative messages into the output spreadsheet file. Try to follow the instructions in those error messages, if any. If you are puzzled, send a note to Caitlin.Clary@biostatglobal.com.

The remainder of this chapter describes several challenges that can occur with BeSD-TI and recommends steps to address them.

1. R halts with an error message like:

   ```
   Error in gzfile(file, mode) : cannot open the connection
   In addition: Warning message:
   In gzfile(file, mode) :
    cannot open compressed file <file path>, probable reason
   'Invalid argument'
   ```

   This problem is usually caused by a momentary conflict between R and your cloud-based syncing or backup service, like Dropbox. Here at Biostat Global Consulting, when we run BeSD-TI, we usually pause Dropbox syncing first, which prevents this error. When BeSD-TI is finished we resume Dropbox syncing.

2. R shows a warning message like:

   ```
   Warning message:
   In file.create(to[okay]) :
    cannot create file <path to the Excel file>, reason
   'Permission denied'
   ```

This warning is usually caused by having an Excel file that BeSD-TI writes to opened while trying to run BeSD-TI again. Please make sure that output Excel files are closed before running BeSD-TI again.

This list of possible problems may grow over time, so check this chapter in future versions of the *User's Guide,* too. If you encounter challenges, send an email to Caitlin.Clary@biostatglobal.com describing the issue and the BeSD-TI developers will assist.

Annex A. Controlling How Strata Are Listed in BeSD-TI Output

Users can control how results are stratified, as well as the order and names of strata, by using the OUTPUT_VARLIST and OUTPUT_LAYOUT settings in Block D of the control program.

The OUTPUT_VARLIST option in the control program is **required**. The variables listed will determine how results are stratified and displayed in output tables. If no stratification is desired, use a variable that has the same value for all observations in the dataset, e.g. the name of the country in which the survey was conducted:

```
besd_global(OUTPUT_VARLIST, "Country_Name")
```

Multiple stratification variables can be provided, e.g. to see results (a) overall, (b) by district, and (c) by urbanicity, the following might be specified:

```
besd_global(OUTPUT_VARLIST, c("Country_Name", "Districtname", "Area"))
```

If the stratifier variables have clear and succinct variable labels and value labels then it may not be necessary to also define an OUTPUT_LAYOUT dataset. When the user does not define such a dataset, BeSD-TI builds one and saves it in the OUTPUT_FOLDER. The file will be named OUTPUT_LAYOUT_automatic.rds. This dataset controls the layout of strata in the tables. If the user wants to modify how the strata are listed – including nesting some strata within others – they may edit this dataset and rename it and then tell BeSD-TI to use their modified layout dataset by specifying the OUTPUT_LAYOUT global.

Structure of an OUTPUT_LAYOUT dataset

The LAYOUT dataset holds four variables.

1. **order** is a numeric variable that takes integer values starting with 1 and increasing by 1. It indicates the order in which the rows should appear in BeSD-TI tables.
2. **label** is a string variable that holds the label, if any, that should appear in this row in the table
3. **rowtype** is a string variable that takes three possible values:
    a. LABEL_ONLY means the row contains a label (i.e., Sex)
    b. DATA_ROW means the row contains a condition (i.e., sex == 1)
    c. BLANK_ROW means the user wants tables to include an extra blank row
4. **condition** is a string variable that holds R syntax to identify the demographic sub-group. For urban respondents, the condition might read “Area == 2” and for rural respondents it might read “Area == 1”.

So a simple LAYOUT dataset might look like this:

order	label	condition	rowtype
1	Area Type (Urban/Rural)		LABEL_ONLY
2	Rural	Area == 1	DATA_ROW
3	Urban	Area == 2	DATA_ROW

And if the user did not want the initial label to appear, they could edit the dataset to look like this:

order	label	condition	rowtype
1	Rural	Area == 1	DATA_ROW
2	Urban	Area == 2	DATA_ROW

And if the user wanted the urban row to appear first and the rural row to appear second, the dataset might look like this:

order	label	condition	rowtype
1	Urban	Area == 2	DATA_ROW
2	Rural	Area == 1	DATA_ROW

If the user specified three demographic stratifiers, like this:

```
besd_global(OUTPUT_VARLIST, c("Area", "CHI_chigender", "CHI_travel"))
besd_global(OUTPUT_LAYOUT, NA)
```

If the variables in the OUTPUT_VARLIST have variable labels and value labels (as they might if the input datasets are Stata .dta files, or if variable/value labels have been applied using the haven package in R), then BeSD-TI would write a file named OUTPUT_LAYOUT_automatic.rds that incorporates those labels and looks like this:

order	label	condition	rowtype
1	Area Type (Urban/Rural)		LABEL_ONLY
2	Rural	Area == 1	DATA_ROW
3	Urban	Area == 2	DATA_ROW
4	Child's gender		LABEL_ONLY
5	Female	CHI_chigender == 1	DATA_ROW
6	Male	CHI_chigender == 2	DATA_ROW
7	Permission needed for mother to take child to clinic?		LABEL_ONLY
8	No	CHI_travel == 0	DATA_ROW
9	Yes	CHI_travel == 1	DATA_ROW

If the Area, CHI_chigender, and CHI_travel variables did *not* have variable and value labels, then the OUTPUT_LAYOUT_automatic.rds file would look like the following:

order	label	condition	rowtype
1	Area		LABEL_ONLY
2	1	Area == 1	DATA_ROW
3	2	Area == 2	DATA_ROW
4	CHI_chigender		LABEL_ONLY
5	1	CHI_chigender == 1	DATA_ROW
6	2	CHI_chigender == 2	DATA_ROW
7	CHI_travel		LABEL_ONLY
8	0	CHI_travel == 0	DATA_ROW
9	1	CHI_travel == 1	DATA_ROW

The appearance of the OUTPUT_LAYOUT files automatically created by BeSD-TI will differ depending on whether the stratification variables have variable and/or value labels in the input datasets or whether they do not. Including variable and value labels is recommended to improve the appearance and interpretability of output.

The user might edit the layout dataset to remove and revise some labels, like this:

order	label	condition	rowtype
1	Rural areas	Area == 1	DATA_ROW
2	Urban areas	Area == 2	DATA_ROW
3			BLANK_ROW
4	Boys	CHI_chigender == 2	DATA_ROW
5	Girls	CHI_chigender == 1	DATA_ROW
6	Would the caregiver need permission to take the child to the clinic for vaccination?		LABEL_ONLY
7	No	CHI_travel == 0	DATA_ROW
8	Yes	CHI_travel == 1	DATA_ROW

The user could rename the dataset “layout_edited.rds” and re-run BeSD-TI, specifying:

```
besd_global(OUTPUT_VARLIST, c("Area", "CHI_chigender", "CHI_travel"))
besd_global(OUTPUT_LAYOUT,
            paste0(OUTPUT_FOLDER, "/layout_edited.rds"))
```

And the resulting Excel table would look like this:

How important do you think vaccines are for your child's health?

	Not at all or a little important (%)	Moderately or very important (%)	N
Rural areas	12.1	87.9	417
Urban areas	10.7	89.3	388
Boys	11.5	88.5	402
Girls	11.6	88.4	403
Would the caregiver need permission to take the child to the clinic for vaccination?			
No	12.9	87.1	94
Yes	11.0	89.0	711

The ability to edit the layout file gives the user substantial flexibility in specifying which stratifiers will appear in the tables and in what order. The conditions in the example above are simple, but the user could specify strata using more complex conditions, e.g.:

order	label	condition	rowtype
1	North	Region %in% c("Northwest", "Northeast")	DATA_ROW
2	South	Region %in% c("Southwest", "Southeast")	DATA_ROW
3	Boys in rural areas	CHI_chigender == 2 & Area == 1	DATA_ROW
4	Girls in rural areas	CHI_chigender == 1 & Area == 1	DATA_ROW
5	Mother or father	CHI_chirel == 1 | CHI_chirel == 2	DATA_ROW
5	Other relative	CHI_chirel %in% c(3:5)	DATA_ROW