Chapter 21
Global Data Base Assembly

Betina V. Dimaranan and Robert A. McDougall

This final chapter of the documentation has three major sections. The first section provides a brief summary of the entire data base construction process. Section 21.2 describes the actual procedures involved the last stage of the process, the final data assembly module. Section 21.3 summarizes the standards adhered to in the data base construction process.

21.1 Data Base Construction Summary

The data base construction procedure is devoted largely to the construction of the main global data base file. The procedure also generates the global sets file, parameters file, energy volumes data file, data summary (GTAPView) file, and tax rates summary file. The whole process is divided into sub-processes or modules with each module designed to handle a well-defined component of the data base, e.g., I-O disaggregation, trade data, etc. (see more about modules in section 21.3).

The construction of the main data base file involves the preparation of global sets and mapping files used in the construction process; the preparation of domestic data bases and international data sets; updating the regional data bases to match the macroeconomic, trade and protection data for the base year; and final data assembly. Figure 21.1 provides a simplified illustration of the data base construction process.

The domestic data bases or input-output tables undergo some preliminary evaluation upon receipt from data contributors. A representative I-O table is constructed from the I-O tables which have full GTAP sectoral disaggregation (chapter 14). The contributed input-output tables which have satisfied the guidelines for contributors then go through checking and cleaning procedures (see chapter 11.A). The I-O tables which do not have full agricultural and/or non-agricultural disaggregation then go through the I-O disaggregation procedure (chapter 13). Input-output tables are constructed for the composite regions, the GTAP regions for which there are no contributed tables (chapter 14). The domestic data bases are then prepared for the updating and adjusting procedure, i.e. the FIT process (chapter 19). Some international data sets are required in the processes pertaining to the I-O tables (thus the arrow from the international data sets box to the domestic data bases box in figure 21.1). These include the macroeconomic data (GDP and GDP per capita) and the agricultural I-O data used in the disaggregation process.

The international data sets obtained from external data contributors are also processed. The procedures involved are laid out in the previous chapters. The international data sets are the macroeconomic data (chapter 18.A), agricultural I-O data (chapter 12.A and 12.B), trade data (chapter 15), protection data (chapter 16), and energy data (chapter 17). Raw or semi-processed data at the disaggregate country level are obtained from data contributors. Further processing, including extending the data to all standard countries, filling in missing values, and aggregation to the GTAP regional and sectoral classifications, is performed.

Since the reference periods of the I-O tables vary, they are updated and reconciled to a common base year of the GTAP Data Base (2001 for GTAP 6) using the macroeconomic data set and the other international data sets trade, protection, energy using the FIT procedure (chapter 19). The regional data bases are then assembled to construct an interim global data file.

Prior to the final assembly module, factor shares data for use in adjusting the payments to land, labor, and capital in the interim global data file are also prepared. For primary factor shares in agriculture, external estimates of factor earnings share are assembled. For primary factor shares in the natural resource based sectors, a proportion of the earnings of labor and capital is reallocated to natural resources to achieve target supply elasticities (chapter 18.C). In splitting the labor payments into skilled and unskilled components for each region and sector, the labor payments are disaggregated using labor earnings shares from the labor data set (chapter 18.D).

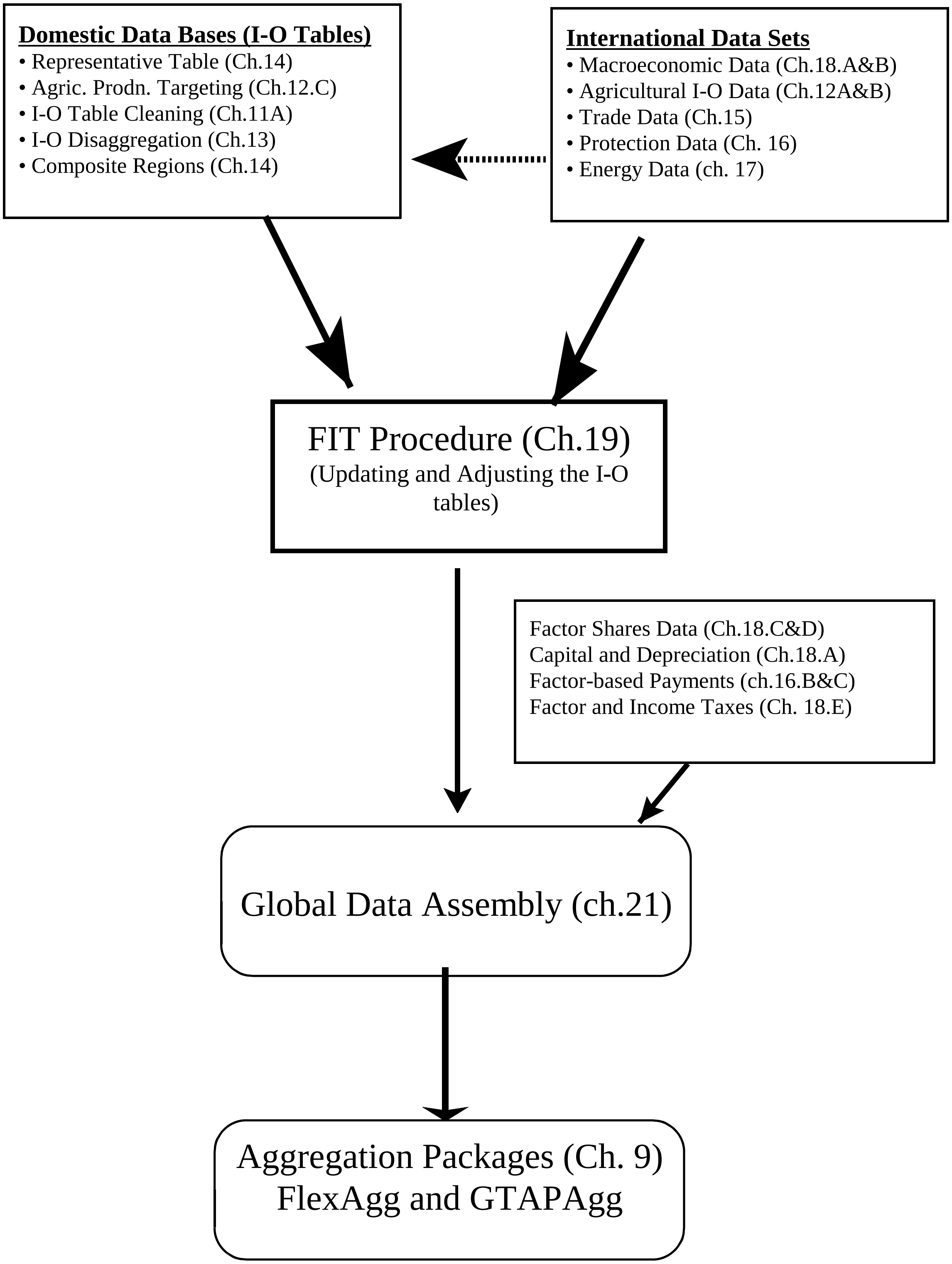
21.2 Final Assembly Module

The final data assembly module is where the various international data sets and domestic data bases are put together to form the main global data base. It is also where the global sets, parameters, energy volumes, and tax rates files undergo some minor, final processing.

The assembly procedure for the main global data base involves:

- disaggregation of the labor payment data into skilled and unskilled labor payments using payment shares generated in the estimation procedure documented in chapter 18.D;
- revision of factor employment data for primary agriculture and natural resource-based sectors using primary factor shares documented in chapter 18.C;
- merging the interim global data file (from FIT procedure, see chapter 19) with the bilateral trade data (chapter 17), protection data (chapter 16), and capital stock and depreciation data (chapter 18.B)
- re-balancing the data base to correct imbalances between domestic product supply and usage and between income and disposition
- further checks on data base for sign violation, imbalances, zero-divide problems.

Figure 21.1 GTAP Global Data Base Construction Procedure


Domestic Data Bases (I-O Tables)
• Representative Table (Ch.14)
• Agric. Prodn. Targeting (Ch.12.C)
• I-O Table Cleaning (Ch.11A)
• I-O Disaggregation (Ch.13)
• Composite Regions (Ch.14)
International Data Sets
• Macroeconomic Data (Ch.18.A&B)
• Agricultural I-O Data (Ch.12A&B)
• Trade Data (Ch.15)
• Protection Data (Ch. 16)
• Energy Data (ch. 17)
FIT Procedure (Ch.19)
(Updating and Adjusting the I-O tables)
Factor Shares Data (Ch.18.C&D)
Capital and Depreciation (Ch.18.A)
Factor-based Payments (ch.16.B&C)
Factor and Income Taxes (Ch. 18.E)
Global Data Assembly (ch.21)
Aggregation Packages (Ch. 9)
FlexAgg and GTAPAgg

The central task in the data assembly procedure is merging the interim global data file with the trade, protection and capital stock data. Various data adjustments are done here. These include:

— international trade and transport margins data and capital stock and depreciation estimates are incorporated into the data base
— the implied revenues/payments associated with the trade-related protection measures are calculated
— the value of trade at market prices are calculated from the protection revenue data (for imports and exports) and trade data at world prices
— the factor payments data are adjusted to incorporate land- and capital-based payments
— the value of firms' purchases of intermediate inputs are adjusted, as necessary, to eliminate cases of very large subsidies on extremely small base value flows
— regional savings is calculated and included as a data array

The global sets, parameters, and energy volumes data file, which are generated in previous modules undergo some minor processing in the final assembly module. The global data summary or GTAPView file and tax rates summary file (see Chapter 10) are also generated in the final data assembly module.

21.3 Data Base Construction Standards

One of the more important innovations related to the GTAP 6 Data Base is the way in which it is being produced. Relative to how earlier versions of the data base were produced, the data base construction procedure has seen immense improvement such that it is now completely automated, replicable, and flexible. This section provides a brief discussion of the standards that have been introduced in the data base construction process to better handle the large and complex task.

21.3.1 Flexibility

There are two main aspects to the flexibility that we want to achieve in the data base construction process. Firstly, we want to make it easy to revise the data base when we get new source data, and secondly, we want to be able to easily revise the regional classification. To make it easy to revise the data base when we get new source data, we keep data and programs in separate files. We count as data not only the economic flow data but also the sets and mapping information. To make it easy to revise the regional classification, we rely partly on the practice described above, but also on another practice, that of encouraging contributors of multi-country data sets to provide the data on a country basis rather than on a GTAP region basis. This lets us revise the GTAP regional classification without going back to data contributors for new reclassified source data.

Regional flexibility is achieved when new regions can be added by simply including the new region(s) in a list or regions used in the data construction process and supplying the additional country's input-output table. Collecting international data bases at the country level helps achieve this objective. The international data sets are then mapped or transformed to apply to a standard set of countries. A mapping between the standard set of countries and the set of GTAP regions is then used to aggregate the data. The mapping file is revised when a new region is introduced in the GTAP data base. Regional flexibility is also facilitated by ensuring that there is no hard-coding of regions in the program codes. The set of region and mappings are read from files that are generic to the construction process.

21.3.2 Build Management

We automate the construction process using the program *`make`*. The *`make`* program identifies the commands needed to create or update the data base, runs those commands, and displays an error message and stops if it encounters an error in running them.

To tell *`make`* how to create the data base, we maintain a *Makefile* showing which target and intermediate files depend on which intermediate and original files, and the commands needed to create the target files from the files on which they depend. The Makefile also contains brief comments, providing basic documentation of the build procedure.

To keep the Makefile manageable and to support a modular structure, we use *`make`* recursively. The master Makefile does not contain any of the commands directly used to create the data base; instead it shows how the different modules depends on each other – how each modules outputs are another modules inputs – and contains commands to run *`make`* on each module. These recursive `make` commands read information from module-specific Makefiles.

In practice, we do not bring the entire construction process under build management. More specifically, we do not bring under build management some procedures involved in the initial conversion or formatting of original data files. We do, however, automate these steps as far as practicable. We confine these un-managed procedures to initial file conversion and formatting. Their function is to convert the data to GEMPACK-readable form, or some other form which we can handle automatically, as directly as possible. All substantive data processing, as opposed to formatting, is done under build management.

21.3.3 Modular Structure

We implement the data base construction process as a collection of modules arranged in a directory tree. For example, there is a trade module where merchandise and services trade data is prepared (chapter 15). There is a module wherein the I-O tables for the composite regions are created (chapter 14). There is also a FIT module wherein the I-O table for each region is reconciled with the trade, protection, and energy data (chapter 19). Each module may be run separately or collectively, under the master Makefile program.

The data base construction root directory contains the master `make` description files, various top-level module directories, and a module-generic directory. The root directory contains no data or program files (other than the make description file); these are all held in subdirectories.

Modules may contain sub-modules; those that do are laid out similarly to the construction root directory, with all data and program files relegated to module-specific subdirectories and a module-generic subdirectory.

Within each bottom-level module we create a standard module directory structure. This structure collects files according to their function and treatment within the module. The module root directory contains as many as are needed of the following basic files and directories (directories are marked by an appended slash character /):

— Makefile : make description file
— lcl/ : data input files - files which are local or specific to the module
— in/ : data input files - include generic sets and mapping information files
— src/ : program source - includes TABLO source code and command files
— wrk/ : working files - files that only have an intermediate role within the module
— dmp/ : dumpable reports - includes log files
— out/ : data output - final output files or files required for use in other modules

21.3.4 Version Control

Version control systems have now been implemented in the data construction process. A publicly available software, Concurrent Versions Systems (CVS), is used with the program files. Under this system, each program file is stored in a repository. Modifications to each file, corresponding to a version of the file, are also stored. Each version is tagged and can be retrieved. The use of CVS enables more efficient use of computer disk space since only the latest version of each program file is actually stored as opposed to completely storing all versions of a file.

The Data Version System (DVS), developed by Robert McDougall at the Center, is used for retrieving input data files (including header array files) since CVS does not handle HAR files very well.

The version control systems are very important for quality control in the complex data base construction procedure. They enable the developer to start from a clean build each time, *i.e.* to start with an empty directory structure and populate the directory, in automated fashion, with the latest program code files and also with the latest version of the input files. The data developer can also readily recreate a previous version of an output data file or rebuild an earlier version of the data base by selecting from the version archive system the appropriate set of files used to create that version.

21.3.5 Common Tool Set

The data base construction process uses a common tool set to spare developers the need to search for and reimplement tools already on hand, learn multiple versions of the same tool, or maintain multiple version of the same tool; and to make it easy to use the same tool set as originally used when replicating old builds.

For data processing, the tool of choice is the GEMPACK software suite, which is also used to implement the standard GTAP model. This is because most of the existing programs in the data base construction package use GEMPACK. The more we stick with GEMPACK, the less time we waste in converting file formats, for instance between GEMPACK and GAMS.

For build management we use `make`. For formatting and text processing, we use `sed`, `awk`, and `perl` since these are all available as free software in implementations of excellent quality. The programs `awk` and `perl` overlap in function.

For batch jobs, where these are not conveniently handled with `DOS` batch files within `command.com`, we use `bash` and shell scripts written for `bash`. However, there should be little need for this since for the most part, we can run the necessary command sequences direct from the `make` file.

References

Dimaranan, Betina V. and Robert A. McDougall. 2000. “GTAP 5: A Large-Scale Data Base Construction Project,” paper presented at the Third Annual Conference of Global Economic Analysis, Melbourne, Australia.

McDougall, Robert A. 2000. GTAP Data Base: Developers’ Reference. Center for Global Trade Analysis, Purdue University, West Lafayette, Indiana.

Oram, Andy and Steve Talbott. 1991. *Managing Projects with Make*. 2nd Edition. O'Reilly & Associates.