8-1992

Database Structure for the Indonesian Food Crop Monitoring System

Gary Stampley
Iowa State University

Database Structure for the Indonesian Food Crop Monitoring System

Abstract
The development of the Indonesian Food Crop Monitoring System (FCMS) database required the compilation of general information on the principal food crops, as well as supportive data on climate, land use, prices, trade, and various macroeconomic conditions. The purpose for the collection and management of these data was twofold: (1) to support several specific food-crop policy analyses to be conducted by the FCMS project team, and (2) to provide a simple capability for monitoring and describing for the food crop sector over time.

Disciplines
Agricultural and Resource Economics | Agricultural Economics | Econometrics

The Database Structure for the Indonesian Food Crop Monitoring System

Gary Stampley

Technical Report 92-TR 27
August 1992

Center for Agricultural and Rural Development
Iowa State University
Ames, Iowa 50011

Gary Stampley is an associate scientist at CARD.

This project was conducted at the Indonesian National Planning Agency with funding from the Asian Development Bank. The opinions expressed are solely those of the authors and may not reflect the official views of either the Indonesian Planning Agency or the Asian Development Bank.

CONTENTS

FIGURES

TABLES

THE DATABASE STRUCTURE FOR THE FOOD CROP MONITORING SYSTEM OF INDONESIA

The development of the Indonesian Food Crop Monitoring System (FCMS) database required the compilation of general information on the principal food crops, as well as supportive data on climate, land use, prices, trade, and various macroeconomic conditions. The purpose for the collection and management of these data was twofold: (1) to support several specific food-crop policy analyses to be conducted by the FCMS project team, and (2) to provide a simple capability for monitoring and describing for the food crop sector over time.

Each database variate is maintained as a time series of either annual or monthly data points; each series represents a set of statistical measurement values for either an Indonesian province, an island group (e.g. Java, Sumatera, Sulawesi, Kalimantan), a region on or off Java, or a national level of aggregation. These area aggregations are consistent with the data presentation formats of the government of Indonesia (GOI) which currently support the administration of national development programs within the food crop sector.

In all, these data establish an unprecedented collection of computerized, well-documented, and validated sets of food crop data from official GOI sources. The FCMS database has been validated by reviewing the official publications of the Indonesian government and/or by requesting that the agencies and institutions responsible for the data implement review and validation procedures. This validation process for all data values was necessary to maintain the integrity and comparability of each series and to guarantee its acceptance as official data. A more pragmatic reason was to ensure a reasonable capability of updating each variate over time.

In conjunction with the established validation procedures, every attempt was made to develop each variate as a "strong data series." Once a starting year and/or month has been identified, a "strong data series" is complete through the most currently available data. Therefore, the time series for each variate is dependent upon three factors:

1. The period of time that the variate was expected to react to policy changes;
2. The economic relationships between specific sets of variates as these relate to analytical requirements; and
3. The statistical procedures currently followed by GOI institutions in the collection, validation, and presentation of primary data.

For every series, the beginning year, as well as subsequent years and months, would of course depend upon historical availability. Therefore, many of the data series are not available for exactly the same term, but the majority have a substantial number of overlapping years. Moreover, values for the most recent years of data should be available for each series. However, no series begins prior to 1968, and most start in 1971 or later and are complete through 1989.

The primary sources of information for the database were from secondary data (i.e., survey data) collected and processed by the Central Bureau of Statistics (CBS) and/or the Ministry of Agriculture (MOA). Approximately 90 percent of the database is derived from the official publications or the unpublished data of these two institutions. The remaining 10 percent comes from various publications of other Indonesian government institutions.

The rest of this report is divided into sections which describe in detail the design and structure of the FCMS database. As it emphasizes the basics, it may be considered to be a primer for using the database. The first section describes in general the data and multiple spreadsheet structure of the database; then the coding and indexing scheme used for accessing specific data is described. Next, various suggestions and techniques for manipulating the database within Lotus 1-2-3 are given, and the steps necessary to link the database with several other analytical software packages are described. Finally, the appendix provides a detailed description of a computer program for a prototype interface between the database and the Statistical Analysis System (SAS).

Lotus 1-2-3 Spreadsheet Database

The FCMS database is structured as a set of Lotus 1-2-3 (release 2.2) spreadsheets with a standard Lotus database format. It is assumed that the reader has had some experience using the Lotus 1-2-3 software, and has a basic understanding of the Lotus database structure. If the reader is unfamiliar with the Lotus database structure, it must be emphasized that it operates somewhat differently than most data applications in Lotus. The Lotus database structure allows consistent data organization and management via structured records with named input fields, and provides a means for the direct access of specific data records through various built-in data query commands. It is strongly recommended that the reader become familiar with these concepts prior to working with the FCMS database. Background on the Lotus database structure and the data query commands is provided in the data management section of most basic Lotus 1-2-3 reference manuals.

There are several obvious advantages in using the Lotus 1-2-3 database structure for the management of the FCMS data. First, there are a large number of experienced Lotus 1-2-3 users within the ministries and agencies of the Indonesian government. Second, the other analytical software to be used in the project required different input formats for the data, and these are available through the "Translate" subprogram of Lotus. Third, the Lotus software is relatively inexpensive when compared to the cost of other microcomputer database software. Fourth, it is easily adapted for use on microcomputers with minimal hardware configurations.

Multiple Spreadsheet Organization

After deciding to use Lotus as the data management software for the FCMS database, there were several reasons for developing the database as multiple spreadsheets. First, it was not possible to store the monthly and annual data series with the same general Lotus database format. Second, because of limitations on how this software manages available computer memory in loading a large spreadsheet, the complete FCMS annual or monthly data series could not be stored as single spreadsheets, even with a computer hardware configuration having a large amount of expanded memory (approximately two megabytes of total conventional and expanded memory can be used by Lotus 1-2-3). Third, large spreadsheets are problematic due to differences in the hardware configurations of computers (i.e., the availability of conventional and expanded memory, disk drive densities, etc.). Fourth, some computer memory must be held in reserve for processing data. For these reasons the utility of large spreadsheets is limited, and smaller spreadsheets containing related data series were devised.

Given these processing constraints, there were two organizational considerations for structuring the database as a set of smaller spreadsheets. First, it was necessary to consider how the database might best be maintained and possibly modified by updating existing data or adding new data series. Second, the proposed analytical uses of these data placed constraints on the general organization of the information for use by 1-2-3 and other software. By organizing the data into small subsets of related variates, it was much easier to locate specific database records in order to meet the proposed analytical requirements of the FCMS team.

Organization of the Spreadsheets

There are currently 24 separate spreadsheets used in the management of over 2,000 separate data series within the FCMS database (see Table 1). Since no spreadsheet is currently over 250 KB in size, each may be easily stored in the standard 640 KB conventional memory of most microcomputers.

Table 1. FCMS database spreadsheets

Spreadsheet Name	Type of Data	Region	Period
CSJY.WK1	Cassava	On Java	Annual
CSOJY.WK1	Cassava	Off Java	Annual
GNJY.WK1	Groundnuts	On Java	Annual
GNOJY.WK1	Groundnuts	Off Java	Annual
MZJY.WK1	Corn	On Java	Annual
MZOJY.WK1	Corn	Off Java	Annual
RCJY.WK1	Total Rice	On Java	Annual
RCOJY.WK1	Total Rice	Off Java	Annual
RCAJY.WK1	Wetland Rice	On Java	Annual
RCAOJY.WK1	Wetland Rice	Off Java	Annual
RCAAJY.WK1	Dryland Rice	On Java	Annual
RCAAOJY.WK1	Dryland Rice	Off Java	Annual
SPJY.WK1	Sweet Potatoes	On Java	Annual
SPOJY.WK1	Sweet Potatoes	Off Java	Annual
SYJY.WK1	Soybeans	On Java	Annual
SYOJY.WK1	Soybeans	Off Java	Annual
JOJINDXY.WK1	Price Indices	On/Off Java	Annual
JOJINDXM.WK1	Price Indices	On/Off Java	Monthly
JOJKLM.WK1	Climate	On/Off Java	Monthly
IRRIGASI.WK1	Irrigation	On/Off Java	Annual
MR-CS01.WK1	Cassava Intensification	On/Off Java	Annual
MR-MZ01.WK1	Corn Intensification	On/Off Java	Annual
MR-RC01A.WK1	Wetland Rice Intensification	On/Off Java	Annual
NATIONAL.WK1	General	National	Annual

The content of each spreadsheet is based upon the specific type of data maintained. The criteria for placing data records in the same spreadsheet were based on three decisions: whether the data series were monthly or annual, what the level of regional representation was, and whether or not the data were for a specific crop.

There are in general two basic types of database information, commodity- and noncommodity-specific data. The commodity data are specific to the major food crops of rice (*padi*) and secondary food crops (*palawija*). Data series for rice are represented for wetland (*sawah*), dryland (*ladang*), and total rice production. The *palawija* crops for which data are included are corn (*jagung*), cassava (*ubi kayu*), sweet potatoes (*ubi jalar*), groundnuts (peanuts, *kacang tanah*), and soybeans (*kedelai*).

The major data series for each food crop are annual, provincial data for: (1) areas harvested and planted; (2) yields; and (3) production output; the data series for (4) production costs are at the level of island group for off Java and province for on Java. The FCMS database dictionary provides a complete listing of the specific variate series. (This dictionary is available as a separate document.) A total of 16 spreadsheets organized as commodity-specific data for on or off Java (eight commodities by two on- or off-Java area designations) were developed for storing these data.

Four spreadsheets contain the noncommodity-specific segment of the database. JOJINDXY.WK1 and JOJINDXM.WK1 maintain annual and monthly data, respectively, for general price indices of nine essential goods in rural areas on or off Java. JOJKLM.WK1 contains monthly data from 14 specific provincial meteorological stations. IRRIGASI.WK1 has provincial-level, annual data on irrigated land availability. Provincial data on the use of intensified land areas in the production of cassava, corn, and *padi sawah* are maintained in spreadsheets MR-CS01, MR-MZ01, and MR-RC01A, respectively. These annual data series represent two types of BIMAS land intensification data, "targeted" and "realized."

The final component of the current FCMS database, NATIONAL.WK1, is a spreadsheet containing annual commodity and noncommodity data at the national level. These macroeconomic data were included in an effort to reflect the agricultural infrastructure and general economic environment of Indonesia.

Organization of Data within a Spreadsheet

In conjunction with the multiple spreadsheet organization of the database, there are two separate Lotus database formats that were defined to maintain the annual and monthly data, respectively. The general structure of these may be viewed as a "relational table" in which the data are organized and managed in a consistent format.

The columns of these spreadsheet tables are given specific data field names. Three types of fields were defined for each variate series (i.e, database record) in the database: *index keys*, *time-series data points*, and *supplemental information*. The cells in each of the rows contain values for the variate data fields. Therefore, the data values contained in each of the information fields define a complete database record. The values for index keys and supplemental information are Lotus character labels, while the time-series data point values are numeric.

Annual Spreadsheets

Field names for the major index key fields of the annual data series are CCODE, CRCODE, SCCODE and FECODE (see Table 2). These index keys are used to locate specific variate time series contained in the fields Y1968...Y1992. The CCODE field contains regional or geographical area code values; the CRCODE field contains code values for crops or general variates; the SCCODE field contains the institutional source code values for the data series; and the FECODE field contains code values for the type of data contained in the series (such as cost structure, rainfall, or specific crop). *These index key values may be viewed as defining the descriptive information for a specific variate series. Therefore, each index key field must have a value for every variate present in the database*. The UNIT1 and UNIT2 fields are supplementary fields containing variate scale factors and measurement unit information, respectively. The fields Y1968...Y1992 contain the annual time-series data points for the years 1968 to 1992. (The data series for monthly data has a slightly different format; see discussion below.) Each value for the time-series data, Y1968...Y1992 has a fixed format of two significant digits to the right of the decimal point. The CODE_F field is a composite character string (including blanks) of the CCODE, FECODE, SCCODE and CRCODE index values, in this order, for each variate series. The value for CODE_F is unique for each variate, and it may be used as a single indexing value for a specific series. The supplementary field, NOTE, contains a code value indicating whether the current series has been validated.

Table 2. Lotus database fields defined for annual data

Field Name	Type of Field	Cell / Width
CCODE	Index Key: regional/geographical	A1 / 4
CRCODE	Index Key: crops; general	B1 / 6
SCCODE	Index Key: institutional source of data	C1 / 3
FECODE	Index Key: type (rainfall, crop, cost structure)	D1 / 8
UNIT1	Supplemental: scale factors	E1 / 9
UNIT2	Supplemental: unit of measurement	F1 / 15
Y1968	Data	G1 / 13
Y1969	Data	H1 / 13
Y1970	Data	I1 / 13
Y1971	Data	J1 / 13
Y1972	Data	K1 / 13
Y1973	Data	L1 / 13
Y1974	Data	M1 / 13
Y1975	Data	N1 / 13
Y1976	Data	O1 / 13
Y1977	Data	P1 / 13
Y1978	Data	Q1 / 13
Y1979	Data	R1 / 13
Y1980	Data	S1 / 13
Y1981	Data	T1 / 13
Y1982	Data	U1 / 13
Y1983	Data	V1 / 13
Y1984	Data	W1 / 13
Y1985	Data	X1 / 13
Y1986	Data	Y1 / 13
Y1987	Data	Z1 / 13
Y1988	Data	AA1 / 13
Y1989	Data	AB1 / 13
Y1990	Data	AC1 / 13
Y1991	Data	AD1 / 13
Y1992	Data	AE1 / 13
CODE_F	Index: composite of index keys; unique for variate	AF1 / 21
NOTE	Supplemental: validation (yes/no)	AG1 / 1

The general layout of the annual spreadsheets, providing the relative positions of the fields, is shown in Table 3. Starting with cell A1 (CCODE) and ending at cell AG1 (NOTE), all of the annual spreadsheets have the same field names. Of course, the number of rows for each annual spreadsheet depends upon the number of variate series maintained.

Table 3. Lotus 1-2-3 database format for FCMS annual time-series data

	A	B	C	D	E	F	G	...	AE	AF	AG
1	CCODE	CRCODE	SCCODE	FECODE	UNIT1	UNIT2	Y1968	...	Y1992	CODE_F	NOTE
.	.	.	.	.	.	.	.	.	.	.	.
21	0121	MZ01	011	SO01	1	Rp/ha	0.00	.	0.00	0121SO01 011MZ01	N
22	0121	MZ01	011	SO02	1	Rp/ha	0.00	.	0.00	0121SO02 011MZ01	N
23	0121	MZ01	011	SO03	1	Rp/ha	0.00	.	0.00	0121SO03 011MZ01	N
24	0121	MZ01	011	SO04	1	Rp/ha	0.00	...	0.00	0121SO04 011MZ01	N

Monthly Spreadsheets

The monthly data are maintained in spreadsheets that have data fields similar to those for the annual data (see Table 4). The key index field names, CCODE, CRCODE, SCCODE, and FECODE have exactly the same interpretation and field widths as those for the annual data. The additional index field, YEAR, is used to access the monthly data for a specific year. The data values for the months, January through December, are contained in the fields O01JAN, O02FEB, O03MAR, O04APR, O05MAY, O06JUN, O07JUL, O08AUG, O09SEP, O10OCT, O11NOV, and O12DEC. The CODE_F index field contains a composite of the index key values for CCODE, FECODE, SCCODE, CRCODE, and the last two digits of the YEAR index field value. The use of the NOTE field is exactly the same as for the annual data.

The field names for the monthly data spreadsheets begin in cell A1 (CCODE) and continue through cell U1 (NOTE). Table 5 shows the relative positions of the names of the database fields specified for the management of monthly data (the February through November field names are not shown).

Table 4. Lotus database fields defined for monthly data

Field Name	Type of Field	Cell / Width
CCODE	Index Key: region/geographical	A1 / 4
CRCODE	Index Key: crops; general	B1 / 6
SCCODE	Index Key: institutional source of data	C1 / 3
FECODE	Index Key: type (rainfall, crop, cost structure)	D1 / 8
YEAR	Index Key	E1 / 4
UNIT1	Supplemental: scale factors	F1 / 9
UNIT2	Supplemental: units of measurement	G1 / 15
O01JAN	Data	H1 / 13
O02FEB	Data	I1 / 13
O03MAR	Data	J1 / 13
O04APR	Data	K1 / 13
O05MAY	Data	L1 / 13
O06JUN	Data	M1 / 13
O07JUL	Data	N1 / 13
O08AUG	Data	O1 / 13
O09SEP	Data	P1 / 13
O10OCT	Data	Q1 / 13
O11NOV	Data	R1 / 13
O12DEC	Data	S1 / 13
CODE_F	Index: composite of index keys; unique for variate	T1 / 23
NOTE	Supplemental: validation (yes/no)	U1 / 1

Table 5. Lotus 1-2-3 database format for FCMS monthly time-series data

	A	B	C	D	E	F	G	H		S	T	U
1	CCODE	CRCODE	SCCODE	FECODE	YEAR	UNIT1	UNIT2	O01JAN	......	O12DEC	CODE_F	NOTE
2	0111	GEN	012	RAIN01	1978	1	ml	117.00	......	257.00	0111RAIN01 GEN 01178	N
3	0111	GEN	012	RAIN01	1979	1	ml	90.00	......	105.00	0111RAIN01 GEN 01179	N
4	0111	GEN	012	RAIN01	1980	1	ml	44.00	......	321.00	0111RAIN01 GEN 01180	N
5	0111	GEN	012	RAIN01	1981	1	ml	44.00	......	141.00	0111RAIN01 GEN 01181	N

Index Coding Scheme for the Data

The index key codes (values) of each database record provide descriptive information on what the time-series values represent, and are utilized for identifying and locating specific data series within the database. These index key codes and the codes for the UNIT2 supplemental field are important because of their meaning, not their specific values. Moreover, the index codes used for accessing the FCMS data are consistent across spreadsheets with respect to interpretation. The numeric scale factor, UNIT1, is used to multiply the values maintained in the data fields in order to reflect the UNIT2 field unit of measurement.

As described in Tables 2 and 4, the annual and monthly FCMS database spreadsheets four and five index key fields, respectively. The index fields with the same names serve the same purpose for both data formats, and the index fields with the same code values have exactly the same interpretation across spreadsheets. For example, to access data for the province of Aceh in any spreadsheet, one would look for the value 0111 in the CCODE field of a database record; all database records specific to Aceh have this value for CCODE.

Specific codes for each of the index keys follow a development scheme. For example, the first two digits of CCODE are always 01 to represent Indonesia. The rest of the CCODE value represents specific areas or regions within Indonesia. The first two characters for CRCODE field values for the food crops are CS, cassava; GN, groundnuts; MZ, corn; RC, rice; SP, sweet potatoes; and SY, soybeans. For all data that are not commodity-specific, the code in the CRCODE field is either GEN, for general information that would affect all crops in a given area, such as rainfall, or NAT, for national data, such as the value of exports. For the FECODE field, the first few characters are usually abbreviations for the general type of data, such as IRR, irrigation; INT, intensified land; PROD, production; SO, cost structure; INDX, index data; and RAIN, rainfall. The last two characters of the FECODE field names are always a sequence of digits that assist in the identification of related data.

This type of systematic coding greatly reduces the coding effort and maintains the intrinsic relationships between specific sets of variates. Documentation is provided on the complete set of codes used for the index key and supplemental fields of each database variate as part of the FCMS Database Dictionary. It is strongly recommended that the reader closely review this dictionary in order to become familiar with the FCMS data and its coding scheme before attempting to work with these data.

Database Manipulation

Specific Lotus keystrokes are provided in brackets {} in the following discussion. It is assumed that the reader is capable of retrieving {/FR} the database spreadsheets for use within Lotus. If so, access to and manipulation of the FCMS data is straightforward. The design of the database allows the use of the Lotus data query commands {/DQ}. These query commands are used to search the database for records that meet a user-specified criterion, and may be used to find, extract, or delete selected records. (Database records should not be deleted unless the user is constructing a customized version of the database spreadsheet.) Therefore, much of this section will concentrate on the use of these data query operations in working with the database, as well as on how these commands might be used in creating a user-customized spreadsheet.

The use of the Lotus data query commands requires that INPUT, CRITERIA, and OUTPUT (for record extraction only) work areas be established. First, the CRITERIA and OUTPUT work areas should be specified within the general work area of the spreadsheet (i.e., not over the database).

When working with the annual data, specify field names for the CRITERIA area by copying {/C} the index key field names, CCODE, CRCODE, SCCODE, FECODE in cells A1...D1 to cells AI1...AL1. The OUTPUT area is then specified by copying the database field names from cells A1...AG1 to cell locations AN1...BT1. The CRITERIA and OUTPUT field names should have the same column widths as the corresponding database fields. There are actually many alternative locations for the CRITERIA and OUTPUT areas, but these are the recommended locations. After establishing the work areas, provide criteria for searching the database by entering valid index key values directly under the appropriate index key field names of the CRITERIA area. Complex search criteria may be established by stacking and combining code values of the index keys.

The search activity may now be initiated by issuing the data query command {/DQ}. At this time indicate to Lotus the work areas defined above, and reset these areas if necessary with the reset command {R}. The INPUT area {I} will normally be the entire range of cells of the database information, including field names. For example, the INPUT area for the annual spreadsheets is specified by first locating the cursor in cell A1 (CCODE). At this point, use the cursor to highlight the cells A1...AG1, and all of the rows (data records) in the spreadsheet. Information to be used as input criteria {C} is specified by highlighting the CRITERIA field name and the index key values that were previously specified to be used in the search. The output area {O} is indicated by highlighting the cells AN1 through BS1; the field name NOTE is not included because Lotus 1-2-3 allows a maximum of 32 field names to be specified for the OUTPUT area. After these three work areas have been identified, the user indicates a find {F}, an extract {E}, or a delete {D} operation based upon the specified search criteria. The data query operation is completed by entering the quit command {Q}.

The data query operations described above for the annual spreadsheets may also be used to locate data records from the monthly spreadsheets. However, the YEAR index key field should be included as part of the CRITERIA work area, and the OUTPUT area field names should correspond to the field names of the monthly database spreadsheets. These differences require changes in the recommended cell locations for the CRITERIA and OUTPUT work areas. The CRITERIA should be specified in cells V1...Z1, as CCODE, CRCODE, SCCODE, FECODE, and YEAR. The database field names should be copied from cells A1...U1 to the OUTPUT area cells, AA1...AU1. From this point the data query operations may be performed in the same way as for the annual data.

Data Entry and Updating

Data entry and updating operations for specific database records are accomplished by using data query techniques to find records. Database records meeting the search criteria will be located and highlighted in the database INPUT area, and the user can scroll within the record to a cell (usually a year/month column) in which a data value is to be entered. An alternative process follows: Scroll the spreadsheet either up or down; locate the record (row) in which data is to be entered and/or updated. Scroll the spreadsheet to the right; stop at the appropriate column and enter the data value. After completing the data entry, the updated version of the database spreadsheet should be saved {/FS}. It is recommended that this updated spreadsheet be saved with a file name that is different from the name of the original FCMS database spreadsheet. After reviewing the new spreadsheet for accuracy, rename it to replace the old version of the spreadsheet. Every attempt should be made to add or update database records in batches in order to minimize this renaming activity.

Adding New Variate Series

Whether to add a new variate series to an existing database spreadsheet or to create an entirely new spreadsheet as part of the database is an important decision. If the new data fits well with the information currently residing in a spreadsheet, and its addition will not make the spreadsheet too large (files should not exceed 250 KB), then adding the information to an existing spreadsheet will facilitate its use. Otherwise, a new spreadsheet should be created.

Adding new data series to an existing spreadsheet is very simple. First, retrieve the spreadsheet {/FR} for which new data is to be added; then scroll the spreadsheet to the last row of existing data. In the row directly below the last one start to enter a new data record in the appropriate database fields. Remember to provide meaningful code values for all of the index key fields. Repeat this process for all new variate series to be added. Be sure to format all new data cells {/RFF2}, and save the revised spreadsheet with a new name with the file save command {/FS}. Review this revised spreadsheet for accuracy before renaming it with the name of the original database spreadsheet.

A new database spreadsheet may be easily created by retrieving one of the existing database spreadsheet shells, ANNSHELL.WK1 for annual data or MONSHELL.WK1 for monthly data. Beginning in row two of this shell, each new data series should be entered in consecutive rows. It is recommended that a new database spreadsheet be created only if a substantial amount of new data is to be entered, or if the new data does not fit well with any data of any existing spreadsheet. *Be sure when saving this new spreadsheet that the file name is different from the spreadsheet shell file name.*

Creating a User-Customized Spreadsheet

It may be useful for the FCMS database user to collect specific database records from several of the database spreadsheets and to reformat the information for the purpose of a specific analysis, such as description, graphical monitoring, or simple regressions. To accomplish this a user would usually need to extract and combine database records, possibly from several spreadsheets, in order to create a user-customized spreadsheet. It is recommended that specific manipulation of the FCMS database be carried out in customized spreadsheets or copies of the original database files, not in the original database files themselves.

Use the process already detailed to search and extract from the spreadsheets the desired database records. After each record extraction, save the OUTPUT work area as a separate spreadsheet. The OUTPUT area(s) may be saved with the file extract {/FXV} command. Issue this command; then by moving the cursor highlight the range of cells to be saved (i.e., the OUTPUT work area). It may be necessary to perform several record and file extractions to retrieve all of the desired database records.

Be sure the OUTPUT areas are saved as files with unique names after each file extract. These files may be combined to complete the customized spreadsheet. To combine files, use the following process:

Step 1. Retrieve {/FR} one of the newly created files.

Step 2. Locate the last database record in the spreadsheet.

Step 3. Place the cursor in column A directly below the last data record row.

Step 4. Issue the file combine command {/FCCE} and indicate another of the newly created extracted files to be combined.

Repeat steps 2 through 4 until all of the extracted files have been combined. Then delete the spreadsheet rows {/WDR} (except for row one) that contain only the field names, CCODE, CRCODE, ... etc., and save this file as the customized spreadsheet.

The customized spreadsheet may be used with a variety of data manipulation processes in order to prepare the information for further analysis. These operations may include the following: deleting unwanted years of data with the column delete command {/WDC}; deleting complete data records with the row delete command {/WDR}; transposing rows and columns of data {/RT} for use in a regression and/or matrix operation; or moving data {/M} in preparation for using a file extract {/FXV}.

Custom Spreadsheet Example 1

Following is an example for constructing a custom spreadsheet to accomplish a specific analysis objective. Suppose this objective was to gain an understanding of the responsiveness of per hectare cassava yields in East Java to the use of chemical fertilizer by cassava producers over time. A simple, exploratory relationship would be the ordinary least-squares regression (OLS) of the per hectare cassava yields on per hectare amounts of chemical fertilizer, controlling for the amount of manure used, since it is a close chemical fertilizer substitute. (Note: This is a data manipulation example, not necessarily a well-specified econometric exercise.) Three annual variate series are needed in order to perform this regression.

The FCMS annual cassava data for Java is located in the file CSJY.WK1. The data dictionary shows East Java data records defined for the average cassava yields per hectare, the average quantity of chemical fertilizer applied per hectare, and the average value of manure applied per hectare; for the example we will assume constant real prices of manure (see Table 6).

Table 6. Lines from the data dictionary that describe the CSJY.WK1 data records used in Example 1

CCODE	CRCODE	SCCODE	FECODE	CSJY.WK1--Variate Description
0135	CS01	011	SO02	East Java cassava: yields (kilograms/hectare)
0135	CS01	011	SO12	East Java cassava: chemical fertilizer quantity (kilogram/hectare)
0135	CS01	011	SO13	East Java cassava: manure value (Rupiah/ hectare)

First, we need to extract these three database records from CSJY.WK1. Therefore, retrieve the CSJY.WK1 file using {/FR}. Copy {/C} the index key field names, cells A1...D1, to cells AI1...AL1. Copy all of the database field names contained in cells A1...AG1 to cells AN1...BT1. Now, the cell AI1 should have the label value CCODE, and the cell AG1 should have the value FECODE. Enter in each cell in the column AI2...AI4 the value '0135, which is the code value for East Java. In the cells AL2, AL3, and AL4 enter the values 'SO02, 'SO12, and 'SO13. Move the cursor to cell A1 {Home Key}; enter the data query command {/DQ}; and if necessary reset the data query fields {R}. Then by highlighting with the cursor, indicate the following data query areas: the database INPUT area {I}, cell range A1...AG137; the CRITERIA area {C}, cell range AI1...AL5; and the OUTPUT area {O}, cell range AN1...BS1 (remember the 32-field-name limit). After identifying these areas, enter the data extract command {E} followed by data query quit command {Q}.

The three selected database records should now be in the OUTPUT area. Move the cursor to the cell AN1; issue the file extract command {/FXV}; give this spreadsheet extract the file name, CSRGR.WK1; highlight the cell range AN1...BS5, and press the enter key {Enter}. This is the custom spreadsheet that will be used for the regression analysis.

Press the keyboard home key {HOME} and retrieve CSRGR.WK1 with the file retrieve command {/FR}. (Note: Annual data exists for the years 1971 through 1989 for all three variates; we will use this period to form our regression.) Move the cursor to cell J2 and enter the cell range transpose command {/RT}. Indicate the cell range to copy, J2...AB4. Then by moving the cursor to cell A8, indicate where you want the transpose range (data columns) to start. Issue the data regression command {/DR} and if necessary the reset command {R}. Specify the X-range {X} as cells

B8...C26, the Y-range {Y} as cells A8...A26, and the OUTPUT area {O} as cells E8...H16. Next issue the intercept command {IC} and the go command {G}; results of the regression will be placed in the output area.

Custom Spreadsheet Example 2

This example demonstrates the use of the database as a monitoring tool through graphical presentation of information in Lotus 1-2-3. The trend in the real value of returns to labor (i.e., wage payments or labor income) within a region for a specific crop over time is an important concept in monitoring the economic returns to labor in specific crop production. This relationship may be represented graphically with a line plot that shows real per-hectare returns to labor over time. Another graphical representation would be to plot the total returns to labor for a specific crop within a region, calculate a trend line for the total returns, and then overlay the two lines. In order to accomplish this it is necessary to have the following information: crop-specific regional returns to labor per hectare, a general regional price index, and the crop area harvested within the region.

First, the selection of the crop and region to be analyzed must be made. Since the Indonesian province which historically has harvested the largest area of cassava is East Java, we will monitor the real value of the returns to labor there. The FCMS database has the total annual area of cassava harvested in East Java, and the average nominal wages paid per hectare for grubbing, ploughing, sowing/planting, weeding, and harvesting activities for cassava in East Java in the file CSJY.WK1 (see Table 7).

The file JOJINDXY.WK1 has an annual rural price index for nine essential commodities that will be used to deflate the nominal labor returns (see Table 6).

First, we want to extract these seven database records from CSJY.WK1. Therefore, retrieve the CSJY.WK1 file using {/FR}. Copy {/C} the index key field names, cells A1...D1 to cells AI1...AL1. Copy all of the database field names contained in cells A1...AG1 to cells AN1...BT1. Now, the cell AI1 should have the label value CCODE, and the cell AG1 should have the value FECODE. Enter in each cell in the column AI2...AI8 the value '0135, which is the regional code value for East Java. In the cells AL2...AL8 enter the values 'PROD02, 'SO22, 'SO23, 'SO24, 'SO25, 'SO26, and 'SO27. Move the cursor to cell A1 {Home Key}; enter the data query command

Table 7. Lines from the data dictionary that describe the CSJY.WK1 data records used in Example 2

CCODE	CRCODE	SCCODE	FECODE	CSJY.WK1--Variate Description
0135	CS01	011	PROD02	East Java cassava: total area harvested (hectares)
0135	CS01	011	SO22	East Java cassava: grubbing workers wages (Rupiah/hectare)
0135	CS01	011	SO23	East Java cassava: ploughing workers wages (Rupiah/hectare)
0135	CS01	011	SO24	East Java cassava: sowing/planting workers wages (Rupiah/hectare)
0135	CS01	011	SO25	East Java cassava: weeding workers wages (Rupiah/hectare)
0135	CS01	011	SO26	East Java cassava: harvesting workers wages (Rupiah/hectare)
0135	CS01	011	SO27	East Java cassava: other workers wages (Rupiah/hectare)

Table 8. Line from the data dictionary that describes the JOJINDXY.WK1 data record used in Example 2

CCODE	CRCODE	SCCODE	FECODE	JOJINDXY.WK1--Variate Description
010B	GEN	011	INDX10	On Java annual rural price index for nine essential commodities

{/DQ}, and if necessary reset the data query fields {R}. Then by highlighting with the cursor, indicate the following data query areas: the data base INPUT area {I}, cell range A1...AG137; the CRITERIA area {C}, cell range AI1...AL8; and the OUTPUT area {O}, cell range AN1...BS1 (remember the 32-field-name limit). After identifying these areas, enter the data extract command, {E}, followed by the data query quit command {Q}.

The seven selected database records should now be in the OUTPUT area. Move the cursor to the cell AN1; issue the file extract command {/FXV}; save this spreadsheet extract with the file name CSMON1.WK1; highlight the cell range AN1...BS8, and press the enter key {Enter}.

Retrieve {/FR} the database spreadsheet JOJINDXY.WK1, and as previously described establish the INPUT, CRITERIA, and OUTPUT data query work areas. In cell AI2 enter the code value '010B, and in cell AL2 enter the code value 'INDX10. Move the cursor to cell A1 {Home Key};

enter the data query command {/DQ}; and if necessary reset the data query fields {R}. Then by highlighting with the cursor indicate the following data query areas: the database INPUT area {I}, cell range A1...AG137; the CRITERIA area {C}, cell range AI1...AL2; and the OUTPUT area {O}, cell range AN1...BS1. After identifying these areas, enter the data extract command {E} followed by the data query quit command {Q}.

Only one database record should now be in the OUTPUT area. Move the cursor to the cell AN1; issue the file extract command {/FXV}; save this spreadsheet extract with the file name CSMON2.WK1; highlight the cell range AN1...BS2, and press the enter key {Enter}.

In order to have all the necessary database records in the same spreadsheet, we want to combine the two files, CSMON1.WK1 and CSMON2.WK2. First, retrieve {/FR} the file CSMON2.WK1, and then move the cursor to cell A3 (below the data record). Issue the file combine command for entire files {/FCCE}, and indicate CSMON1.WK1. Row three will now contain the field names from CSMON1.WK1, and this row needs to be deleted. Place the cursor in cell A3 (it should be there already), and enter the row delete command {/WDR}. The data from CSMON1.WK1 will now have been moved up one row.

After examining the eight data records for periods of overlapping availability, you will find data available for all variates during the period 1974 through 1989. First, compute the total nominal labor payments per hectare per year, by entering the formula @SUM(M4..M9) in cell M10 (the Y1974 column), and then copy this formula {/C} to cells N10..AB10. The copy will automatically change the cell references to the appropriate column for subsequent years. Second, compute the real value of the wage payment based on the 1971 price index for nine essential commodities in rural Java, by entering in cell M11 the formula (M10/M2)*100; copy this formula {/C} to cells N11...AB11. Third, find the total real value of wage payments (billions of rupiah) for cassava in East Java, by multiplying the real wage payments in rupiah per hectare times the total cassava area harvested. Enter the formula (M11*M3)/1000000000 in cell M12, and copy it {/C} to cells N12...AB12.

The cell range M10...AB12 contains values that will potentially be used in the subsequent analysis. However, after reviewing the computed nominal wage labor data series, the total per-hectare nominal wage payments in 1979 and 1984, cells R10 and W10, demonstrate unexpected movement in both magnitude and direction. These two points will be considered as "extreme outliers" and will be deleted from the analysis. Use the range erase command {/RE} to erase the values in the cell ranges R10...R12 and W10...W12. Now, establish a time trend variate by entering in cells M13...AB13 the values 1...14, being sure to skip cells R13 and W13.

A regression analysis will be used to compute an average growth rate in real wage payments for the period 1974-1989. Specify this regression as the real value of total labor returns on the time trend. With the range transpose command {/RT}, copy two rows of cell values M12...AB13, to two columns of cells, M16...N31 (i.e., after specifying the rows to transpose move the cursor to cell M16). After transposing the row data vectors, use the row delete command {/WDR} to remove the empty data cells in the cell range M16...M31 (i.e., rows that would have contained 1979 and 1984 data points). Now issue the data regression command {/DR} and if necessary the reset command {R}. Specify the X-range {X} as cells N16...N29, the Y-range {Y} as cells M16...M29, and the OUTPUT range {O} as cells P17...S25. Set the INTERCEPT to compute with the command {IC}, and then issue the go command {G}. The results of the regression will be placed in the regression OUTPUT area.

In the OUTPUT area, cells P17...S25, the coefficient values for the intercept and time trend are located in S18 and R24, respectively. To calculate the predicted annual values of total real wage payments, enter in cell M14 the formula ($S$18+$R$24*M13), and then copy {/C} this formula to cells N14...AB14. Use the range erase {/RE} to remove any values in the cells R14 and W14 (i.e., the zeroes for 1979 and 1984). Calculate the average growth rate of real labor payments in East Java cassava production by entering the formula @RATE($AB$14,$M$14,16) in cell T19. Next enter the label 'Inflation Rate '74-'89 in cell P29, and calculate the average growth rate of the price index (i.e., inflation rate) by entering the formula @RATE($AB$2,$M$2,16) in cell R29.

To compare the average growth rate in nominal wage payments to the inflation rate, begin by regressing nominal wages on the time trend. Use the range transpose command {/RT} to copy the cells M10...AB10 to the cells M32...M45. Remove any blank cells in the cell range M32...M45 with the row delete command {/WDR}. Copy the cells N16...N29 to cells N32...N45. Now issue the data regression command {/DR} and the reset command {R}. Specify the X-range {X} as cells N32...N45, the Y-range {Y} as cells M32...M45, and the OUTPUT range {O} as cells P33...S41. Set the intercept to compute with the command {IC}, and then issue the go command {G}. The results of the regression will be placed in the regression output area, P33...S41.

In the OUTPUT area, cells P33...S41, the coefficient values for the intercept and time trend are located in S34 and R40, respectively. To calculate the predicted annual values of total real wage payments, enter in cell M15 the formula ($S$34+$R$40*M13), and then copy {/C} this formula to cells N15..AB15. Use the range erase {/RE} to remove any values in the cells R15 and W15 (i.e., the zeros for 1979 and 1984). Calculate the average growth rate of nominal wage payments per hectare in East Java cassava production by entering the formula @RATE($AB$14,$M$14,16) in cell T35.

In order to specify a line graph of real per hectare wage payments over time, issue the Lotus graph command {/G} and if necessary reset all the graphic information fields {R}. Enter the graph type command {TL} to specify a line graph. Next enter the data range command for the X axis {X}; specify the cells M1...AB1 (i.e., the database year field names for 1974 through 1989); and specify the A data range {A} as cells M11...AB11 (i.e., the average annual real per hectare wage payments).

To include graph titles, axis scale information, or legends, enter the graphics options submenu {O}. Use the titles subcommand {T} to specify the following titles: the first title {F} as "Real Returns to Labor"; the second title {S} as "East Java; Cassava"; the X-axis label {X} as "Year"; and the Y-axis label {Y} as "Rupiah/Ha. B.Y. '71." Use the scale subcommand {S} to specify the Y scale detail as Manual {M}, Lower bounds {L} as 3100, Upper bounds {U} as 7800; these should approximate the largest and smallest values of the real per hectare wage payments. Specify Format {F} as format fixed 1 {F1}, and Indicator {Y}. Specify the Skip values for the X scale option {S} as 5. Use the escape key {Esc} to return to the options submenu and enter the quit command {Q}. From the graphics main menu enter the view command {V} to see the graph. After viewing, name this graph with the name create command {NC}, CSEJRWH, and exit the graphics menu {E} (Figure 1).

To complete this descriptive monitoring example, we will specify an overlay of the line graphs of actual and predicted total annual real wage payments. Issue the graph command {/G} and reset the information fields {R}. Enter the graph type command {TL} to specify a line graph. Next enter the axis data range commands: for the X axis {X} specify the cells M1...AB1 (i.e., the database year field names for 1974 through 1989); for the A data range {A} specify the cells M12...AB12 (i.e., the actual total annual real wage payments); and for the B data range {B} specify the cells M14...AB14 i.e., the predicted total annual real wage payments).

To include graph titles, scale information, legends, etc., enter the graphics options submenu {O}. Use the titles subcommand {T} to specify the following titles: the first title {F}, as "Real Returns to Labor"; the second title {S} as "East Java: Cassava"; the X-axis label {X} as "Year"; and the Y-axis label {Y} as "Rupiah (Bil)." Use the scale subcommand {S} to specify the Y scale detail as format

```
Type X A B C D E F Reset View Save Options Name Group Exit
Use Create Delete Reset Table

                Graph Settings
Type:  Line          Titles: First  Real Returns to Labor
                             Second East Java:  Cassava
X:  M1..AB1                    X axis Year
A:  M11..AB11                   Y axis Rupiah/Ha. B.Y  '71
B:
C:                                Y scale:  X scale:
D:                          Scaling  Manual     Automatic
E:                          Lower    3100
F:                          Upper    7800
                            Format   (F1)      (G)
Grid: None     Color: Yes    Indicator Yes        No

   Legend:          Format: Data Labels:       Skip: 5
A                 Both
B                 Both
C                 Both
D                 Both
E                 Both
F                 Both
```

Figure 1. Graph fields: Real per hectare cassava wages in East Java

{F} with format fixed 2 {F2}, and indicator, {Y}. Specify the skip values for the X scale option {S} as 5. Use the escape key {Esc} to return to the options submenu. To provide descriptive information on the plotted lines use the legend command {L}. For the A legend {A} enter the label "Actual," and for the B legend {B} enter the label "Predicted." Use the escape key {Esc} to return to the options submenu. To exit the options submenu enter the quit command {Q}. From the graphics main menu enter the view command {V} to see the graph. After viewing this graph, name it with the name create command {NC} CSEJTRW (see Figure 2). Exit the graphics menu with the command {E}. Finish the example by using the file save command {/FS} to save this spreadsheet with the name CSMON.WK1.

Linkages to Other Analytical Software

The linkages to other microcomputer software packages is viewed by the FCMS project team as essential in making the database available for other analytical purposes. Although most software packages cannot directly read the database spreadsheet files (see Table 1), these spreadsheets may be translated by Lotus into data storage formats that are accepted by many. The platform for file translation of spreadsheets is the "Translate" subprogram of Lotus 1-2-3. Two data exchange formats, the DIF (Data-Interchange Format) and the DBF format used by DBase III software, were used by the FCMS project team in linking the Lotus database spreadsheets with other software.

```
Type X A B C D E F Reset View Save Options Name Group Exit
Use Create Delete Reset Table

                Graph Settings
Type:  Line          Titles: First  Real Returns to Labor
                        Second East Java:  Cassava
X:  M1..AB1                X axis Year
A:  M12..AB12               Y axis Rupiah  (Bil.)
B:
C:                            Y scale:  X scale:
D:                       Scaling   Automatic Automatic
E:                       Lower
F:                       Upper
                        Format    (F2)     (G)
Grid: None    Color: Yes    Indicator Yes       No

   Legend:          Format: Data Labels:        Skip: 5
A   Actual           Both
B   Predicted        Both
C                   Both
D                   Both
E                   Both
F                   Both
```

Figure 2. Graph fields: Real total cassava wages in East Java

The General Translation Process

Translation of the database spreadsheets to either of the above formats is straightforward. To use the Lotus "Translate" subprogram, enter {T} from the main menu. The program will prompt with the type of files available for translation. Respond to Lotus by indicating the file format, Lotus 1-2-3 release 2.2. The next screen will prompt for information on what format to use to translate the file. *Select the file format your software will require for data input.* For example, the SHAZAM econometric software requires the translation format to be DIF, while the Statistical Analysis System

(SAS) and the Statistical Package for the Social Sciences (SPSS PC+) for the personal computer will accept either DIF or DBF files. After selecting the file formats to translate "FROM" and "TO," the "Translate" subprogram prompts for the "SOURCE" and "TARGET" file names.

If the source file is a Lotus release 2.2 spreadsheet, the file must have been saved with the file save command {/FS} before translation. Moreover, most analytical software requires well-organized input data, and at times it will be beneficial to develop a custom spreadsheet in Lotus, rather than translating a complete FCMS database file. Users of a computer software package should understand its specific capabilities for restructuring and editing data before deciding upon the best approach.

To translate to a DBF format, the "Translate" subprogram requires the Lotus 1-2-3 spreadsheets to have a database format (i.e., structured data with field names). Therefore, the FCMS spreadsheet files have been well designed for translation purposes. If only subsets (i.e., specific records and fields) of the FCMS database records are required for a particular analysis, consider creating a custom spreadsheet prior to the translation process, and translating this newly constructed smaller file.

APPENDIX
USING THE DATABASE WITH SAS

For demonstrating the use of the FCMS database with another software package, a prototype for a user interface between the Lotus 1-2-3 database files and the Statistical Analysis System (PC/SAS) was developed. Specifically designed for many types of descriptive and econometric analyses, SAS is a very complete analytical software tool. The capacity of the SAS software to restructure and analyze time-series data is quite extensive. It integrates a powerful data manipulation language with a structured format for data management and pre-programmed "Procedures" for data analysis and presentation.

The interface is a SAS computer program, written in the SAS macro language, designed as a menu-driven system to access data from the annual crop production and cost structure files. These files are CSJY.WK1, CSOJY.WK1, GNJY.WK1, GNOJY.WK1, MZJY.WK1, MZOJY.WK1, RCJY.WK1, RCOJY.WK1, RCAJY.WK1, RCAOJY.WK1, RCAAJY.WK1, RCAAOJY.WK1, SPJY.WK1, SPOJY.WK1, SYJY.WK1 and SYOJY.WK1. It uses as input DBF files created from the translated versions of these database spreadsheets. The interface's primary purpose is to select specific subsets of database records from these crop files, and to restructure these data as SAS data sets with a format suitable for many SAS analytical procedures. Within the PC/SAS environment these data are then available to the SAS procedures for further processing.

SAS is also available for many different types of computer environments, and the SAS language is nearly 100 percent compatible across many of these environments. There are minor differences in computer operating system input/output specifications and in certain language features and syntax. Therefore, with minimal effort it is possible to make the FCMS data available in other computer environments that would allow centralized access and the development of analytical applications that require greater computing capacities than a microcomputer can offer.

This SAS interface was demonstrated at several seminars at various ministries of the Indonesian government. The reactions of the participants to the application interface was not enthusiastic and may be summarized as follows:

1. SAS software was believed to be too difficult for people to learn fully.
2. SAS is expensive.

3. There was not much need for such a strong analytical tool as SAS.

These reactions limited the amount of FCMS project time allocated for developmental work on this interface. However, the consideration of user reactions like this are very important in efforts to develop useful types of computer software for information management and data analysis.