

**INFORME INTERNO**

**MANUAL TECNICO INTRODUCTORIO PARA EL USO  
DEL MANEJADOR DE BASES DE DATOS  
KnowledgeMan (KMAN)**

**Luis A. Ugalde A.**

**La preparación y publicación de este trabajo han sido  
patrocinadas con fondos del Proyecto Cultivo de Arboles de  
Uso Múltiple CATIE-ROCAP No. 596-0117**

**CENTRO AGRONOMOICO TROPICAL DE INVESTIGACION Y ENSEÑANZA  
Programa Producción de Desarrollo Agropecuario Sostenido  
Area de Producción Forestal y Agroforestal  
Turrialba, Costa Rica, Agosto 1989**

## CONTENIDO

	Pag.
1. INTRODUCCION .....	1
2. INSTALACION DE KMAN .....	2
3. DESCRIPCION DE KMAN .....	2
4. MODULO DE BASES DE DATOS .....	4
4.1 INICIAR CON KMAN .....	4
4.2 DEFINIR UNA TABLA .....	4
4.3 CREAR REGISTROS .....	6
4.4 SALIR DE KMAN .....	7
4.5 PONER EN USO UNA TABLA .....	7
4.6 EDITAR O CORREGIR UNA TABLA .....	8
4.7 VER LA ESTRUCTURA DE UNA TABLA .....	9
4.8 SELECCIONAR INFORMACION DE UNA TABLA .....	9
4.9 OBTENER REGISTROS .....	11
4.10 ORDENAR LA INFORMACION UNA TABLA .....	12
4.11 EL USO DE MACROS .....	12
4.12 REDEFINIR UNA TABLA .....	13
4.13 ELIMINAR REGISTROS DE UNA TABLA .....	13
4.14 DESTRUIR UNA TABLA .....	14
4.15 REORDENAR CAMPOS DE UNA TABLA .....	14
4.16 VARIABLES AMBIENTALES .....	15
4.17 SOLICITAR AYUDA .....	16
5. MODULO DE TEXTO (TEXT) .....	16
5.1 CREAR UN ARCHIVO EN TEXTO .....	16
5.2 CORRER UN PROGRAMA EN TEXTO .....	18
4.3 EXTENSION DE LOS ARCHIVOS CREADOS .....	20

## **MANUAL TECNICO INTRODUCTORIO PARA EL USO DEL MANEJADOR DE BASES DE DATOS KnowledgeMan (KMAN)**

### **1. INTRODUCCION**

El objetivo de este manual técnico introductorio es de servir como una guía para el uso del manejador de bases de datos KnowledgeMan, conocido como **KMAN** versión 2.5 para microcomputadora IBM o compatibles.

La necesidad de contar con un manual técnico para el uso de **KMAN** en español, es de gran importancia para diferentes usuarios en América Central debido a que este programa esta siendo utilizado como parte del **sistema MIRA** (Manejo de Información sobre Recursos Arbóreos), el cual incluye información silvicultural y socio-económica generada desde 1980 por los proyectos **LEÑA** y **MADELEÑA** del CATIE en colaboración con las instituciones forestales de los países de América Central. Para la utilización del sistema **MIRA**, y el desarrollo de otras bases de datos, se esta dando capacitación en el uso de **KMAN** en CATIE y en cada uno de los países donde opera el proyecto **MADELEÑA**.

El programa KnowledgeMan es una marca registrada de Micro Data Base Systems, Inc. (mdbs, Inc.). P.O. Box 248 Lafayette, IN 47902. El costo total del programa es de aproximadamente \$800 incluyendo el costo de envío.

Esta es la primera versión en español del manual y describe el uso de los comandos básicos de dos de los seis módulos que forman **KMAN** considerados de mayor interés para el uso del sistema **MIRA**. Se espera que este manual introductorio sea actualizado y complementado con otros módulos cada cierto tiempo, de acuerdo con las necesidades e interés de los usuarios en la región. En algunos casos se ha incluido vocabulario en inglés, con el propósito de familiarizar al usuario con los manuales técnicos de **KMAN** en ese idioma. Para los usuarios que deseen profundizar en el uso de aplicaciones más avanzadas se sugiere consultar los manuales técnicos de **KMAN**, en inglés:

- KnowledgeMan Installation Guide
- KnowledgeMan User's Guide
- KnowledgeMan Technical Reference
- KnowledgeMan Quick Reference Guide
- Discovering KnowledgeMan

## 2. INSTALACION DE KMAN

Para la utilización de **KMAN** se requiere de una microcomputadora con disco duro. El programa consta de nueve disquetes (floppy disk) de cinco y cuarto pulgadas. La instalación se realiza colocando el disquete número uno en la unidad **A** y escribiendo el comando **INSTALL**. A continuación el usuario debe responder una serie de preguntas de acuerdo con las características de su microcomputadora. Se debe poner especial cuidado al responder a la pregunta sobre las cuatro posibilidades relacionadas con el tipo de tarjeta de gráficos, para que el módulo de gráficos funcione correctamente. Las cuatro posibilidades son:

1. IBM AT, XT PC, o compatible con IBM
2. Compatible con tarjeta Hércules
3. Tarjeta EGA de resolución media
4. Tarjeta EGA de alta resolución

## 3. DESCRIPCION DE KMAN

**KMAN** es un programa integrado que permite una gran variedad de aplicaciones en el manejo y procesamiento de información. **KMAN** utiliza como base el lenguaje **SQL** (Structure Query Language). **KMAN** Es uno de los manejadores de bases de datos para microcomputadora más poderosos que existen en el mercado, especialmente en la utilización del **SQL** para bases de datos relacionales, en el desarrollo de aplicaciones de programación, y en la integración de varios módulos para manejar y presentar la información. **KMAN** esta compuesto de diez componentes: seis módulos y cuatro interfases que le permiten al usuario interactuar entre los componentes dentro de **KMAN**, utilizando los mismos comandos.

Existen dos versiones de **KMAN**, una para usuarios individuales (single user), y la versión multiusuarios que puede ser utilizada por sistemas operativos multiusuarios y por redes como la LAN (Local Area Network), en una red de microcomputadoras. También existen versiones para minicomputadoras o computadores grandes (mainframe). Adicionalmente existe la versión de **KMAN** conocida como "Run-time version", un tipo de semicompilador, la cual permite distribuir las aplicaciones de programación elaboradas por un usuario a otros usuarios, estos últimos pueden hacer uso de las aplicaciones, aún sin tener el programa **KMAN**.

### Especificaciones generales y capacidad de **KMAN**:

- Número de tablas (archivos) en uso al mismo tiempo .....				45
- Número de registros por tabla .....	2	137	483	647
- Número de caracteres por registro ..			65	535
- Número de campos por registro .....				255
- Número de caracteres por campo .....			65	535
- Tipo de gráficos .....				15
- Códigos de seguridad .....				65535
- Número de índices por tabla .....				ilimitado

### Los módulos de **KMAN** son:

- Manejo de Información (**KMAN**), se define como el módulo de la base de datos. Permite definir tablas, crear registros, editar registros y manipular la información. Se invoca con el comando **KMAN -G** para el modo de uso con comandos y con **KMAN** para el modo a través de menú de ventanas con comandos.

- Procesador de Texto (**K-TEXT**), es el procesador de palabras que permite escribir programas de **KMAN**. Se invoca con el comando **TEXT**.

- Hoja de Cálculo (**CALC**), es una hoja electrónica, semejante a Lotus 123, que facilita hacer cálculos y la presentación de información en columnas y filas. Se invoca con el comando **CALC**.

- Gráficos (**K-GRAPH**), para realizar gráficos a partir de diferentes fuentes de información. Se invoca con el comando **PLOT**.

- Informes con Formato (**K-REPORT**), para presentar en forma interactiva la información generada de las búsquedas, en diferentes tipos de informes con diferentes formatos. Se invoca con el comando **DESIGN**.

- Diseño de Formatos (**K-PAINT**), para diseñar en pantalla ventanas que faciliten la entrada o búsqueda de información en forma interactiva. Se invoca con el comando **PAINT**.

### Las interfases son:

- Interfase con el lenguaje C (**K-C**), permite correr programas desarrollados en el lenguaje C, utilizando funciones de biblioteca o subrutinas en lenguaje C.

Interface Guía con Menú (**GUIDE**), facilita el aprendizaje de **KMAN** a través de menú con ventanas en inglés.

Se recomienda para usuarios que inician su uso por primera vez y que desean usar el tutor de **KMAN**.

- Interfase de Comunicación (**K-COMM**), sirve para comunicación externa entre microcomputadoras.

- Interfase en Inglés como Lenguaje Natural (**K-CHAT**), permite la flexibilidad de hacer búsquedas usando el inglés.

#### 4. MODULO DE BASE DE DATOS (KMAN)

##### 4.1 INICIAR CON KMAN

Descripción de comandos:

**KMAN -G** (y pulse la tecla de entrada, sinónimo de las teclas: **<-**, **Enter**, **Intro**, o **Return** para el teclado en inglés). Recuerde dejar un espacio en blanco entre la palabra **KMAN** y **-G**. Para ejecutar cualquier comando el usuario debe pulsar la tecla **Enter**. El usuario entra directamente al módulo de bases de datos y aparece el cursor después de **KMan>** (Figura 1), indicando al usuario que puede comenzar a usar los comandos de **KMAN**.

```
MDBS KnowledgeMan Ver 2.5 (ne)
The Universal Knowledge Management System
(C) Copyright 1983, 1988, mdbs, Inc.
Lafayette, IN 47902

KMan>_
```

Figura 1. Ingreso al programa KMAN

##### 4.2 DEFINIR UNA TABLA

**DEFINE (F3)** Para crear o definir una nueva tabla, y luego el nombre de la tabla con un máximo de ocho caracteres. La tecla **F3** se puede pulsar en vez de escribir el comando **DEFINE**. Los comandos se pueden escribir en minúscula o mayúscula, indiferentemente.

Ejemplo:

**DEFINE** DPARC (DPARC es el nombre de la tabla), y al pulsar **Enter** aparece:

**FILE?** (pulse **Enter**), si no desea dar otro nombre al archivo que contiene la tabla. Normalmente se prefiere dejar el mismo nombre.

**FIELD?** A continuación entre el nombre del primer campo o variable (field), con un **máximo de ocho caracteres** e indique el tipo de campo con las siguientes opciones:

**NUM** = numérico, en caso de que la variable sea un número con o sin decimales,  
**INT** = para números enteros (sin decimales),  
**STR** = para alfanumérico (para letras únicamente, o para una combinación de números y letras). Solo en el caso de **STR** se debe, además, indicar el número máximo de caracteres que va a tener el campo,  
**DATE** = para fechas,  
**LOG** = lógico, para un valor falso (false) o verdadero (true)

Se puede especificar, aunque **es opcional**, el **formato** (picture) de cada campo. Esto ayuda a controlar y verificar la entrada de información en cada campo, no permitiendo el ingreso de información que no cumpla con el formato especificado en el campo. El **formato debe estar en minúscula y entre comillas**. Las opciones más comunes son:

- a** letras únicamente
- c** alfanumérico, letras, números, o letras y números
- d** números, permite los signos + (más), - (menos), o decimales
- n** solamente números, ningún símbolo u otro caracter
- r** cualquier caracter ASCII

El símbolo **%** se puede usar para abreviar el formato del campo, por ejemplo: "%8r" es igual a "rrrrrrrr"

**ENDDEF** Para terminar de definir la tabla, o pulse **Enter**.

Inicio de la práctica para el usuario:

A continuación se muestra un ejemplo de la creación de una tabla con nombre DPARC (descripción de parcelas de experimentos forestales) con seis campos: NSPARC (número serial de parcela), COPAIS (código de país, usando dos letras), COEXP (código de experimento), COTRAT (código de tratamiento de la parcela), NSREP (número serial de repetición de la parcela dentro de un experimento), y COESPEC (código de especie). Se presenta el caso con y sin formato para cada campo o variable.



**KMAN -G (pulse Enter)**

Ejemplo con formato:

```
KMan>DEFINE DPARC
FILE? (Enter)
FIELD? NSPARC INT USING "nnnn"
FIELD? COPAIS STR 2 USING "aa"
FIELD? COEXP STR 5 USING "ccccc"
FIELD? COTRAT STR 8 USING "cccccccc"
FIELD? NSREP INT USING "nn"
FIELD? COESPEC STR 6 USING "aaaaaa"
ENDDF (o pulse Enter)
```

Ejemplo sin formato:

```
KMan>DEFINE DPARC
FILE? (Enter)
FIELD? NSPARC INT
FIELD? COPAIS STR 2
FIELD? COEXP STR 5
FIELD? COTRAT STR 8
FIELD? NSREP INT
FIELD? COESPEC STR 6
ENDDF (o pulseEnter)
```

### 4.3 CREAR REGISTROS

Para ingresar información (crear registros) en una tabla ya definida se usa **CREATE (F4)**:

**CREATE RECORD FOR** (y el nombre de la tabla). Para iniciar a crear o agregar nuevos registros en la tabla:

ejemplo: **CREATE RECORD FOR dparc (Enter)**

Después de este comando aparece un registro con todos los campos vacíos para iniciar la entrada de la información en cada campo (Figura 2). Pulse **Enter** para colocarse en el primer campo y comenzar a entrar información, continúe en la misma forma pulsando **Enter** para entrar información en los restantes campos del primer registro y en los demás registros de la tabla. Al terminar de ingresar la información del último registro, pulse la tecla **Esc** para salirse de la tabla donde estaba ingresando información.

Record Number: 1	<==Start creating this record	Esc=quit
NSPARC:	_____	
COPAIS:	_____	
COEXP:	_____	
COTRAT:	_____	
NSREP:	_____	
COESPEC:	_____	

Figura 2. Para iniciar a ingresar información

El usuario debe, como práctica, ingresar los siguientes doce registros en la tabla **dparc**. Esto le permitirá seguir utilizando los comandos que a continuación se describen en este manual:

NSPARC	COPAIS	COEXP	COTRAT	NSREP	COESPEC
1	CR	025L	CRECI	1	LEUCLE
2	CR	025L	REBRO	3	LEUCLE
3	PA	102L	NPK25	1	GLIRSE
4	PA	102L	NPK50	2	GLIRSE
5	NI	058L	FERTI	1	EUCACA
6	NI	034L	FERTI	1	LEUCLE
7	HN	112L	CRECI	3	GMELAR
8	HN	112L	CRECI	4	GMELAR
9	SV	001L	BIOMASA	1	TECTGR
10	SV	002L	BIOMASA	2	TECTGR
11	GT	016L	CRECI	3	BOMBQU
12	GT	122L	CRECI	4	BOMBQU

#### 4.4 SALIR DE KMAN

Para salir completamente del programa **KMAN**. El usuario debe usar el comando **BYE** cada vez que desea dejar de usar **KMAN**.

ejemplo: **Kman>BYE**

#### 4.5 PONER EN USO UNA TABLA

Para utilizar la información ya ingresada en una tabla después de haber salido de **KMAN**, primero se entra a **KMAN** de nuevo, con el comando:

**KMAN -G (Enter)**

Seguidamente se debe poner en uso la tabla, para esto el usuario debe escribir:

**USE (F1)** (y el nombre de la tabla). Para poner en uso una tabla. Cada vez que se sale y se vuelve a entrar a **KMAN** es necesario poner en uso las tablas que se van a utilizar.

ejemplo: **USE dparc (Enter)**

#### 4.6 EDITAR O CORREGIR UNA TABLA

Para ver y **editar** (corregir la información), registro por registro, con todos los campos de una tabla, primero se debe poner en uso la tabla que se desee utilizar o editar, con el comando **USE** y el nombre de la tabla, como se mencionó anteriormente (sección 4.5),. Posteriormente use el comando:

**BROWSE (F8)** (y el nombre de la tabla)

ejemplo: **BROWSE dparc (Enter)**

Después de pulsar **Enter**, aparece en la pantalla (Figura 3) el primer registro de la tabla con la información, el cursor aparece en la línea de comandos (parte superior de la pantalla). Si el usuario pulsa **Enter** el cursor pasa al primer campo y el usuario puede hacer cambios si desea, y seguir pulsando **Enter** para pasar por los otros campos de ese registro. Si el usuario desea ver solamente la información en cada registro, cuando el cursor está en la línea de comandos, debe mantener presionada la tecla **Ctrl** y pulsar la tecla de la flecha hacia la derecha **->** al mismo tiempo, cada vez que desee ver un nuevo registro. Para terminar de ver registros y salir de la tabla, pulse la tecla **Esc** cuando el cursor está en la línea superior de comandos.

```
Record Number:1    ^->=Start creating this record Esc=quit

NSPARC:    1
COPAIS:    CR
COEXP:     0251
COTRAT:    CRECI
NSREP:     1
COESPEC:   LEUCLE
```

Figura 3. Información del primer registro

**SHOW (F2)** Para ver o verificar las tablas que estén en uso:

ejemplo:**SHOW (Enter)**

#### 4.7 VER LA ESTRUCTURA DE UNA TABLA

**SHOW** (y el nombre de la tabla). Para ver la estructura o definición de la tabla con todos los campos (Figura 4). En la parte superior aparece información descriptiva sobre la tabla. La estructura de la tabla está descrita por cuatro columnas: el nombre, el tipo, el tamaño y el formato de los campos.

ejemplo: **SHOW DPARC (Enter)**

Field	Type	Size	Picture
-----	----	----	-----
#MARC	LOGIC	1	(default)
NSPARC	INT	2	%4n
COPAIS	STR	2	%2r
COEXP	STR	5	%5c
COTRAT	STR	8	%8c
NSREP	INT	2	nn
COESPEC	STR	6	%6a

Figura 4. Estructura de la tabla DPARC

#### 4.8 SELECCIONAR INFORMACION DE UNA TABLA

**SELECT (F6)** y a continuación el nombre de los campos. Se pueden usar uno o varios campos, y poner condiciones para ordenar o seleccionar solo parte de la información que satisfaga las condiciones deseadas.

Obsérvese en los ejemplos 3, 4 y 5 que se muestran a continuación, que si el campo definido en la tabla es alfanumérico (**STR**), la condición debe estar entre comillas, y si el campo fue clasificado como numérico (**NUM**) o entero (**INT**) no se necesitan las comillas.

Pulsando la tecla con la flecha hacia arriba **^** se recobran todos los comandos utilizados anteriormente, en el orden en que fueron usados.

Al final de cada búsqueda por omisión (default) se obtiene la estadística básica (descriptiva) para los campos que sean de tipo numérico (**NUM**) o entero (**INT**). La estadística básica incluye:

**Sum** (sumatoria de todos los valores de la columna)  
**Ave** (el promedio)  
**Var** (la varianza)  
**Sdv** (la desviación estándar)  
**Min** (el valor mínimo)  
**Max** (el valor máximo)  
**Number of Observations:** (el número de observaciones o registros obtenidos en la búsqueda)

La estadística se puede obviar, si se desea, como se verá más adelante, con el uso de otros comandos (variables ambientales).

Ejemplos de búsquedas (recuperación de información) utilizando la tabla DPARC. La búsqueda se escribe en una misma línea. Recuerde pulsar **Enter** hasta que termine de escribir la búsqueda.

1. Selección de todos los campos de la tabla DPARC:

```
SELECT FROM DPARC
```

2. Selección de tres campos de la tabla DPARC:

```
SELECT NSPARC COPAIS COESPEC FROM DPARC
```

3. Selección de cuatro campos y condicionando para únicamente cuando el país es Costa Rica:

```
SELECT NSPARC COPAIS COEXP NSREP FROM DPARC FOR COPAIS="CR"
```

4. Selección de tres campos con dos condiciones, código de país igual Panamá y las parcelas con número serial mayor a uno:

```
SELECT NSPARC COEXP COESPEC FROM DPARC FOR COPAIS="PA" AND NSPARC>1
```

5. Selección de información, agrupando por campos y ordenando en orden ascendente (AZ) o descendente (ZA):

```
SELECT COEXP NSREP COESPEC FROM DPARC FOR COPAIS="NI" GROUP BY COEXP COESPEC ORDER BY AZ COEXP
```

6. Selección de información en forma única por registro, usando **ALL UNIQUE** para todos los campos o **UNIQUE** para un campo en particular. Esto hace que en el momento de la búsqueda no se repitan los registros, si los campos en la búsqueda tienen la misma información, dos ejemplos:

```
SELECT ALL UNIQUE COEXP COESPEC FROM DPARC
```

```
SELECT COESPEC COPAIS UNIQUE COEXP FROM DPARC
```

7. El comando **STAT** puede utilizarse en vez del comando **SELECT**, esto permite obtener unicamente la estadística básica para cada campo, y no lista la información de cada registro, por ejemplo:

**STAT NSPARC NSREP FROM DPARC (Enter)**

Para hacer correcciones en una tabla se puede usar **BROWSE** y dar condiciones, en forma similar a los ejemplos con **SELECT**. Esto permite recuperar rapidamente uno o un grupo de registros que se desee corregir, por ejemplo:

**BROWSE FOR COPAIS="CR" AND COEXP="025L" (Enter)**

Esta búsqueda permite recuperar todas las parcelas que existen en la tabla DPARC, que estén en Costa Rica y que pertenezcan al experimento 025L.

Los posibles **operadores** para poner condiciones en una búsqueda son:

Igual:	<b>EQ</b>	=	Mayor o igual:	<b>GE</b>	>=
No igual:	<b>NE</b>	<>	Menor que:	<b>LT</b>	<
Mayor que:	<b>GT</b>	>	Menor o igual:	<b>LE</b>	<=
En:	<b>IN</b>				

Se puede notar que para cada operador hay dos posibilidades, por medio de letras o utilizando símbolos. Ambas posibilidades se pueden utilizar indistintamente y se pueden utilizar varias condiciones en una misma búsqueda.

#### 4.9 OBTENER REGISTROS

Si el usuario desea ver u obtener un registro específico de una tabla, puede utilizar el comando **OBTAIN (F5)** y especificar el registro que desee recuperar (no permite editar), con este comando también se pueden incluir condiciones:

ejemplo: **OBTAIN FIRST FROM DPARC (Enter)**  
**OBTAIN 10 FROM DPARC (Enter)**

Las opciones para especificar registros son:

**FIRST** (el primero)  
**LAST** (el último)  
**NEXT** (el siguiente)  
**PRIOR** (el anterior)

#### 4.10 ORDENAR LA INFORMACION DE UNA TABLA

La información en una tabla puede ser reordenada en diferente orden de acuerdo a uno o varios campos, según lo desee el usuario, mediante el comando **SORT**. A diferencia de ordenar la información que se obtiene en una búsqueda usando el comando **SELECT**, el comando **SORT** deja ordenada la información en la tabla en forma permanente, hasta que el usuario vuelva a hacer un nuevo **SORT**. Por ejemplo para ordenar la tabla **dparc** por orden descendente de **COPAIS** y **NSPARC**:

```
SORT dparc BY ZA COPAIS NSPARC (Enter)
```

De igual manera se puede especificar más de una condición cuando se usa el comando **SORT**, como en los ejemplos con **SELECT**.

#### 4.11 EL USO DE MACROS

**KMAN** permite construir "Macros" (instrucciones). Un macro permite guardar en forma temporal o permanente los comandos de **KMAN**, o las instrucciones de una búsqueda bajo un nombre que el usuario especifica. Estos pueden utilizarse las veces que se desee en una sesión, o pueden quedar permanentes como archivos, si el usuario lo desea. Por ejemplo, la búsqueda número dos, puede quedar en un macro con el nombre **CAS01**:

```
MACRO CAS01 SELECT NSPARC COPAIS COESPEC FROM DPAR (Enter)
```

Para ejecutar el macro el usuario debe escribir únicamente el nombre del macro y dar entrada, por ejemplo:

```
CAS01 (Enter)
```

Después de construir el macro, si el usuario desea asegurarlo para que quede permanente, debe usar el comando **SAVE** y el nombre del macro con la extensión **.ICF**, por ejemplo:

```
SAVE CAS01.ICF
```

Cada vez que se ingresa a **KMAN**, el usuario debe poner en uso los macros permanentes que desea usar utilizando el comando **LOAD** y el nombre del macro, por ejemplo:

```
LOAD CAS01 (Enter)
```

#### 4.12 REDEFINIR UNA TABLA

A continuación se presentan ejemplos para redefinir la estructura de una tabla:

Para agregar o eliminar un campo (variable) de una tabla se debe redefinir la tabla usando el comando **REDEFINE**. Por ejemplo si se desea agregar un campo que sea número de árboles por parcela (NOARB), y eliminar el campo de código de tratamiento (COTRAT), en la misma tabla DPARC que fue definida antes. Nótese en la figura 5, el uso del signo + (más) para agregar y - (menos) para eliminar. En el caso de eliminar un campo no es necesario indicar el tipo de campo, pero para añadir un campo, sí es necesario incluir el tipo de campo y recordar (sesión 4.2) que en el caso que el campo sea tipo alfanumérico debe añadirse **STR** e incluir el número máximo de caracteres.

```

REDEFINE dparc      (Enter)
FILE?              (Enter)
FIELD?+NOARB INT  (Enter)
FIELD?-COTRAT     (Enter)
FIELD?             (Enter)

```

Figura 5. El proceso de redefinir la tabla DPARC

#### 4.13 ELIMINAR REGISTROS DE UNA TABLA

Para eliminar un registro o un grupo de registros de una tabla, primero se escogen (marcan) los registros especificando el rango de los registros que se deseen eliminar de la tabla con el comando **MARK**, y luego se eliminan con el comando **COMPRESS**. Por ejemplo si se desea eliminar el registro número tres de la tabla DPARC que fue creada con 12 registros:

```
MARK IN DPARC WITH TRUE RANGE 3,3 (Enter)
```

Después que se ejecute este comando, escriba:

```
COMPRESS DPARC (Enter)
```

Al finalizar la ejecución de este comando, los registros quedan eliminados. Para comprobar esto, haga de nuevo una selección de todos los registros:

**SELECT FROM dparc (Enter)**

El usuario podrá observar que ahora la tabla contiene únicamente once registros.

#### 4.14 DESTRUIR UNA TABLA

Una tabla se puede destruir (eliminar) en forma permanente, si el usuario **esta seguro** que no desea mantener la información, usando el comando **DESTROY** y el nombre de la tabla.

ejemplo: **DESTROY DPARC (Enter)**

No se recomienda que el usuario use este comando en esta práctica, ya que después de ejecutado este comando, la información no se puede recuperar. Es preferible tener la tabla DPARC, para seguir practicando los demás comandos.

#### 4.15 REORDENAR CAMPOS DE UNA TABLA

En el caso que el usuario olvide incluir un campo en la tabla, puede redefinir la tabla y agregar el campo o los campos al final de la tabla (sección 4.12). Posteriormente si desea, puede cambiar el orden de los campos en la tabla mediante el comando **CONVERT**, seguido de los **nombres** de los campos en el orden deseado. En esta forma los campos se reordenan en una tabla con distinto nombre. Para esto, primero se necesita usar la variable ambiental **CF=4** (descritas en la sección 4.16), la cual hace que al usar el comando **CONVERT**, la salida sea enviada a una nueva tabla. **CONVERT** funciona en forma similar al comando **SELECT**. Por ejemplo, el usuario puede reordenar o eliminar algunos de los campos de la tabla DPARC y enviarlos a una nueva tabla **DPARC2**. La tabla **DPARC2** no necesita haber sido definida con anterioridad.

**E.CF=4 (Enter)**

**CONVERT NSPARC COEXP COPAIS ..... TO DPARC2 (Enter)**

En esta forma la tabla **DPARC2** contiene los campos en el nuevo orden. Si el usuario desea puede eliminar o destruir la tabla original **DPARC** (ver sección 4.14) y dejar solo la tabla **DPARC2**.

#### 4.16 VARIABLES AMBIENTALES

Son variables que permiten manejar o cambiar el ambiente en que opera **KMAN**. Estas deben usarse (activarse) antes de hacer la búsqueda o la selección de información. Después de cada búsqueda se pueden volver a cambiar. Existen un total de 99 variables ambientales. Todas las variables ambientales inician con **E.** y el usuario debe especificar la condición en que desea dejar la variable ambiental. A continuación se describen las de uso más frecuente:

**E.OPRN=FALSE** o **TRUE**. Para enviar la información que se desea a la impresora, por omisión (default) es **FALSE**, es decir que la información no se esta enviando a la impresora. El usuario puede escoger **FALSE** o **TRUE**.

**TRUE** significa verdadero, que sí desea imprimir  
**FALSE** significa falso, que no desea imprimir

**E.PAUS=FALSE**. Para ver la información en el monitor de la microcomputadora en forma pausada, es decir pantalla por pantalla, por omisión es falso (**FALSE**).

**E.STAT=TRUE**. Permite obtener la estadística básica para los campos numéricos (**NUM**) o Enteros (**INT**), por omisión es verdadero (**TRUE**).

**E.LSTR=15**. Para el largo máximo de caracteres que se desea usar para los campos alfanuméricos de tipo **STR**. Por omisión es 15, y se puede escoger hasta un máximo de 255 caracteres.

**E.DECI=2** Para el número de decimales que se desea en los campos que son numéricos (**NUM**). Por omisión es 2 y se puede cambiar hasta un máximo de 5.

**E.SUPH=FALSE**, para evitar que salgan los nombres de los campos al inicio de cada página. Por omisión es falso (**FALSE**).

**E.SPGN=FALSE**, para evitar la numeración de las páginas, por omisión es falso.

**E.PWID=120**, controla el ancho de la página al momento de imprimir la información de una búsqueda. Por omisión es 120, y se puede cambiar hasta un máximo de 255 caracteres.

Por ejemplo, supongamos que queremos imprimir la búsqueda número uno, primero hay que **verificar** que la impresora este conectada a la microcomputadora, y después escribir:

**KMan>E.OPRN=TRUE** (Enter), y a continuación la búsqueda:

**SELECT FROM DPARC** (Enter)

La información que se observa en el monitor, esta siendo impresa. Si no desea seguir imprimiendo, debe cambiar la variable de nuevo:

**KMan>E.OPRN=FALSE (Enter)**

El usuario puede saber en qué condición está la variable ambiental usando el signo de interrogación ? y luego el nombre de la variable ambiental, por ejemplo:

**?E.OPRN (Enter)**

#### **4.17 SOLICITAR AYUDA**

Si el usuario tiene duda o desea conocer la sintaxis de los comandos puede solicitar ayuda en forma interactiva, en dos formas, una es usando el comando:

**KMan>HELP (Enter)**

De esta forma **KMAN** presenta un sistema de menú y submenú en inglés, donde el usuario puede mover el cursor y seleccionar el tema que desee. La otra posibilidad es usar el comando **HELP** y escribir el nombre de un comando, una variable o una función para la cual desea información, por ejemplo:

**KMan>HELP SELECT (Enter)**

### **5.MODULO DE TEXTO**

Este módulo es el procesador de palabras que permite, por ejemplo escribir búsquedas o programas que puedan ser corridos o invocados directamente desde el módulo de bases de datos. Se invoca con el comando **TEXT**. Este módulo facilita que el usuario mantenga las búsquedas en archivos. Un archivo puede contener varias búsquedas, incluyendo las variables ambientales. Esto hace que el usuario, cada vez que desee hacer una búsqueda, ejecute el programa o archivo que contenga esa búsqueda.

#### **5.1 CREAR UN ARCHIVO EN TEXTO**

Supongamos que el usuario desea tener en un archivo la búsqueda número cuatro que se creó como práctica (sección 4.8), además, desea que la salida de la búsqueda se imprima. Primero hay que ingresar al módulo de **TEXT** con el comando:



Después de terminar de escribir la búsqueda, pulse la tecla **Esc** para regresar a la línea de comandos y escriba el comando **STOP** para grabar (guardar) el archivo y regresar al módulo de base de datos.

## 5.2 CORRER UN PROGRAMA EN TEXTO

Desde el módulo de bases de datos se puede invocar (correr) el archivo con nombre **PRUEBA1**, el cual contiene el programa con la búsqueda número cuatro (Figura 7), usando el comando **PERFORM** y el nombre del archivo. Asegúrese que la impresora este conectada y lista para imprimir:

```
KMan>PERFORM PRUEBA1 (Enter)
```

El usuario puede ingresar de nuevo al módulo de texto y hacer correcciones o agregar búsquedas en el mismo archivo. Supongamos que el usuario desea agregar la búsqueda número cinco de la sección 4.8, y un comentario, explicando para qué sirve cada búsqueda. La figura 8 muestra cómo hacer esto:

```
KMan>TEXT (Enter)
```

```
File Name? PRUEBA1 (Enter)
```

```
Line:1 ----- Col:1 ----- pruebal.ipf ----- Insert
! ESTA BUSQUEDA RECUPERA INFORMACION PARA LAS PARCELAS
! EN PANAMA Y MAYORES A UNO

E.OPRN=TRUE
SELECT NSPARC COEXP COESPEC FROM DPARC FOR COPAIS="PA" \
AND NSPARC>1

! ESTA BUSQUEDA RECUPERA INFORMACION PARA NICARAGUA Y LA
! AGRUPA POR EXPERIMENTOS

SELECT COEXP NSREP COESPEC FROM DPARC FOR COPAIS="NI" \
GROUP BY COEXP COESPEC ORDER BY AZ COEXP
```

Figura 8. Ejemplo de un archivo de TEXTO

Pulse **Esc** para regresar a la línea de comandos, y **STOP** para grabar los cambios y regresar al módulo de bases de datos.

Obsérvese que se pueden dejar líneas en blanco entre las búsquedas, o entre los comentarios y las búsquedas, pero no dentro de una búsqueda.

El símbolo de admiración (!) al inicio de una línea permite hacer comentarios sin que el programa sufra alteraciones. Si desea dejar (bloquear) varias líneas para comentarios o porque no desea ejecutar parte del archivo, se puede usar los caracteres /\* al inicio del párrafo y \*/ al final del párrafo.

Por ejemplo supongamos que se quiere modificar el archivo **PRUEBA1** (figura 8), para ejecutar solo la segunda búsqueda y además, el usuario **no desea imprimir** la información. En este caso, se debe bloquear la primera búsqueda. El archivo debe quedar como se muestra en la Figura 9.

```
KMan>TEXT (Enter)
File Name? PRUEBA1 (Enter)
```

```
Line:1 Col:1 pruebal.ipf Insert
! ESTA BUSQUEDA RECUPERA INFORMACION PARA LAS PARCELAS
! EN PANAMA Y MAYORES A UNO

E.OPRN=FALSE
/*
SELECT NSPARC COEXP COESPEC FROM DPARC FOR COPAIS="PA" \
AND NSPARC>1
*/

! ESTA BUSQUEDA RECUPERA INFORMACION PARA NICARAGUA Y LA
! AGRUPA POR EXPERIMENTOS

SELECT COEXP NSREP COESPEC FROM DPARC FOR COPAIS="NI" \
GROUP BY COEXP COESPEC ORDER BY AZ COEXP
```

Figura 9. Ejemplo de un archivo de TEXTO con bloqueo

Después de hacer los cambios, pulse la tecla **Esc** para regresar a la línea de comandos y escriba el comando **STOP** para grabar (guardar) el archivo y regresar al módulo de base de datos.

Para invocar de nuevo el archivo, escriba:

```
KMan>PERFORM PRUEBA1 (Enter)
```

### 5.3 EXTENSION DE LOS ARCHIVOS CREADOS

Al crear diferentes tipos de archivos en **KMAN**, el programa les asigna una extensión en particular:

**.ITB** para las tablas  
**.ICF** para los macros y para las hojas de cálculo  
**.IPF** para los archivos de texto.

De manera que, como práctica, si el usuario escribe el comando:

**KMan>DIR (Enter)**

Podrá observar los archivos creados durante la práctica:

<b>DPARC.ITB</b>	(la tabla de descripción de parcelas)
<b>CAS01.ICF</b>	(el macro permanente que contiene la búsqueda número 2)
<b>PRUEBA1.IPF</b>	(el archivo de texto que contiene dos búsquedas)

## **PERSONAL TECNICO DEL CATIE/PROYECTO MADELEÑA\***

### **JEFATURA**

Ronnie de Camino  
Hernán Rodríguez  
Carlos Navarro

Coordinador Regional  
Administración  
Asistente Técnico

### **SILVICULTURA**

Miguel A. Musálem  
David Hughell  
Valentín Jiménez  
Héctor A. Martínez  
Rodolfo Salazar  
Luis Ugalde

Silvicultor Principal  
Modelación  
Silvicultura  
Silvicultura  
Biometrista  
Manejo de Información

### **SOCIOECONOMIA**

Thomas McKenzie  
Dean Current  
Héctor Chavarría  
Manuel Gómez  
Leyla González  
Stanley Heckadon  
Carlos Reiche  
Carlos Rivas

Economista Principal  
Socioeconomía/Manejo de Información  
Extensionista Asistente  
Economía  
Sociología  
Sociología  
Economía  
Extensión

### **EDICION**

Emilio Hidalgo de Caviedes  
Xinia Robles

Editor  
Documentalista

### **PAISES**

#### **GUATEMALA**

Carlos Figueroa  
Eberto de León

Coordinador - Silvicultura  
Economía

#### **HONDURAS**

Rolando Ordóñez  
Juan F. Pastora

Coordinador - Silvicultura  
Economía

#### **EL SALVADOR**

Hugo Zambrana  
Modesto Juárez

Coordinador - Silvicultura  
Economía

#### **COSTA RICA**

Walter Picado  
Carlos Luis Díaz

Coordinador - Silvicultura  
Economía

#### **PANAMA**

Blás Morán  
Rafael Tirado  
Sebastián Sutherland

Coordinador - Silvicultura  
Economía  
Silvicultor

---

\* Madeleña es un proyecto de investigación, capacitación y diseminación del cultivo de árboles de uso múltiple en América Central y Panamá. Es financiado por AID/ROCAP, y ejecutado por INRENARE de Panamá, DGF de Costa Rica, COHDEFOR de Honduras, CENREN de El Salvador, DIGEBOS de Guatemala con la coordinación regional del CATIE.