
IQ for Unix and Windows Installation & Reference Guide

IQ for Unix and Windows Installation & Reference Guide

This document is intended for users of IQ for Unix and IQ for Windows with dL4 and UniBasic files.

Information in this document is subject to change without notice and does not represent a commitment on the part of Dynamic Concepts, Inc. (DCI). Every attempt was made to present this document in a complete and accurate form. DCI shall not be responsible for any damages (including, but not limited to consequential) caused by the use of or reliance upon the product(s) described herein.

The software described in this document is furnished under a license agreement or nondisclosure agreement. The purchaser may use and/or copy the software only in accordance with the terms of the agreement. No part of this manual may be reproduced in any way, shape or form, for any purpose, without the express written consent of DCI.

© Copyright 1997 Dynamic Concepts, Inc. (DCI). All rights reserved.
One Columbia, Suite 100
Aliso Viejo, CA. 92656
www.dynamic.com

dL4 is a trademark of Dynamic Concepts, Inc.

UniBasic is a trademark of Dynamic Concepts, Inc.

Dynamic Windows[®] is a trademark of Dynamic Concepts Inc.

BITS is a trademark of Dynamic Concepts, Inc.

IRIS is a trademark of Point 4 Data Corporation.

c-tree is a trademark of FairCom.

IQ is a trademark of IQ Software Corporation.

Unix is a registered trademark of Unix Systems Laboratories.

SYBASE is a registered trademark of Sybase, Inc.

CodeBase is a trademark of Sequiter Software Inc.

Microsoft, MS, MS-DOS, Microsoft Access, and FoxPro are registered trademarks, and ODBC, Windows and Windows NT are trademarks of Microsoft Corporation in the USA and other countries.

CONVENTIONS	1
INTRODUCTION	3
INSTALLATION & CONFIGURATION OF IQ FOR UNIX	5
<i>System Requirements</i>	5
<i>Directories</i>	6
<i>Load IQ Distribution</i>	6
<i>Execute the Install Script</i>	8
<i>Errors During Installation</i>	9
<i>Configuring IQ for Your Terminals</i>	10
<i>Using MAKETERM</i>	12
<i>Define the TERM Variable</i>	12
<i>Start MAKETERM</i>	12
<i>Define the Terminal</i>	13
<i>Editing the Terminal Control Strings</i>	15
<i>Defining Box-Drawing (Graphics) Characters (optional)</i>	21
<i>Saving Your Changes</i>	23
<i>Activating IQ for Unix</i>	23
<i>Licensing Options</i>	23
<i>SSNs and DCI Security Daemon</i>	24
<i>Setting the Unix Environment for IQ</i>	25
<i>Setting IQDD</i>	25
<i>Setting IQCAP</i>	25
<i>Setting IQDATA</i>	26
<i>Setting IQEDITOR</i>	26
<i>Setting IQOUTPUT</i>	26
<i>Testing the Installation</i>	26
<i>If You Are Upgrading From Version 2</i>	29
<i>Running IQ2 and IQ for Unix On The Same System</i>	30
NEW FEATURES IN IQ FOR UNIX	31
<i>New Output Types</i>	31
<i>Performance Enhancements</i>	32
<i>New Temporary Field Functions</i>	32
<i>Other New Features and Functions</i>	32
<i>New Configuration Options</i>	34
<i>New Formatting Abilities</i>	35
<i>User Interface Improvements</i>	35
<i>Report Writer Enhancements (Intelligent Query Only)</i>	36

<i>Data Dictionary Improvements</i>	36
INSTALLATION & CONFIGURATION OF IQ FOR WINDOWS 95/NT	41
<i>System Requirements</i>	41
<i>ODBC Requirements</i>	42
<i>New Installation</i>	43
<i>From Media:</i>	43
<i>From Internet:</i>	43
<i>Backing up IQ for Windows</i>	44
<i>Activating IQ for Windows 95/NT</i>	44
<i>Upgrading From Older Versions</i>	45
NEW FEATURES IN IQ FOR WINDOWS 95/NT	47
<i>New Server Package</i>	47
<i>Subset Reporting</i>	47
<i>ODBC Support</i>	47
<i>Date-Time and Interval Support</i>	48
<i>Better Speed</i>	48
<i>Report Preview</i>	48
<i>Default Report Areas</i>	48
<i>Default Design Changes</i>	49
<i>Label Design</i>	49
<i>Matrix Changes</i>	49
<i>Chart Changes</i>	49
<i>Picklist Support</i>	50
<i>Help Changes</i>	50
<i>Procedure Export & Import</i>	50
<i>Toolbar Hints</i>	50
<i>Easier Column Selection</i>	50
<i>New Error Message Handling</i>	51
<i>New Configuration Options</i>	51
DEFINING UniBasic & dL4 DATABASES FOR IQ	53
<i>Introduction</i>	53
<i>Data Dictionary</i>	53
<i>Running DDMAINT</i>	54
<i>Defining Files and File Information</i>	55
<i>UniBasic and dL4 File Information</i>	55
<i>File Name</i>	56
<i>File Types</i>	57
<i>Record Length</i>	57

<i>Header Records</i>	58
<i>Is This File Indexed?</i>	58
<i>Primary Read Sequence</i>	59
<i>Security Level</i>	59
<i>Location of Deleted Record Mark</i>	60
<i>Size of Deleted Record Mark</i>	60
<i>Deleted Record Mark</i>	61
<i>Data Field Separator Marks</i>	61
<i>Multiple Record Types</i>	61
<i>Defining File Key Information</i>	62
<i>Index File Name</i>	62
<i>Key Number</i>	63
<i>Key Part Number</i>	63
<i>External Field Name</i>	64
<i>Type of Key Part</i>	64
<i>Location of Key Part</i>	65
<i>Storage Size of Key Part</i>	65
<i>Date Format</i>	66
<i>Date Layout</i>	66
<i>Defining Field Information</i>	67
<i>Field Name</i>	67
<i>Sequence Number</i>	68
<i>External Name</i>	68
<i>Field Type</i>	69
<i>Location of Field</i>	72
<i>Record Number of Field</i>	73
<i>Storage Size of Field</i>	73
<i>Sub-Field Offset</i>	73
<i>Sub-Field Size</i>	74
<i>Date Format</i>	74
<i>Date Layout</i>	75
<i>Output Size of Field</i>	76
<i>Decimal Digits</i>	76
<i>Array Elements (1st and 2nd Dimension)</i>	77
<i>Array Group Size</i>	77
<i>Security Level</i>	78
<i>Help Message Number</i>	78
<i>Output Format</i>	79
<i>External Info</i>	79
<i>Database Category Notes</i>	80

CONVENTIONS

Values you must choose and enter are shown in italic type.

directory/filename

IQ executables are all printed in upper case.

DDMAINT

File names are printed in lower case letters in a special font.

ddmaster.dat

Commands to be entered at the operating system prompt are shown like this:

cd /usr/lib/iq3

INTRODUCTION

This publication contains information on the installation, configuration and data dictionary maintenance of IQ for Unix and IQ for Windows. Also included is a section on the new features.

The publication is divided into three parts:

- ❑ **Installation, Configuration and Activation**
A guide to installing, configuring and activating the Unix and Windows environment and configuration settings for use with DCI products.
- ❑ **Data Dictionary Maintenance**
A supplemental section to the *IQ Data Dictionary Reference* that discusses set-up of dL4 and UniBasic supported databases
- ❑ **New Features**
A section describing the new features of IQ for Unix and IQ for Windows.

INSTALLATION & CONFIGURATION OF IQ FOR UNIX

The major steps to install IQ for Unix are:

1. Create a directory and load IQ files onto your disk
2. Execute the install script
3. Configure IQ for your terminal
4. Activate IQ (Passport daemon version 3.1 or higher and a new SSN may be required)
5. Set the Unix environment for IQ
6. Edit the configuration settings as necessary for your system

The installation instructions are required for both new installations and upgrades. If you are upgrading, please follow the instructions under the title "If You Are Upgrading From version 2" after completing all the steps through "Execute the Install Script".

System Requirements

IQ installation requires:

- 4MB of free disk space
- A tape or diskette drive
- A character-based terminal
- DCI security Passport daemon version 3.1 or higher (see Activating IQ)
- Any of a wide variety of standard printers (optional)

Directories

IQ for Unix installation creates or uses the following directories:

Location	Directory Contents
/usr/bin	Shell scripts linked to all IQ programs placed in /usr/lib/iq3
/usr/lib/iq3	IQ programs, configuration and data files
/usr/lib/iq3	Demo dictionary and demo support files
/usr/lib/iq3/ <i>language</i>	Foreign language files for IQ

Load IQ Distribution

1. Back up your hard drive or, at minimum, your current IQ directories, if any.
2. Log in as root.
3. Create a directory to store the installation media's contents. For example:

```
# mkdir /tmp/iq
```

4. Change to the directory you just created. For example:

```
# cd /tmp/iq
```

5. Place the IQ media in the appropriate drive.
6. Use the command shown on the media label to load IQ files to your disk. For example, if the format type on the label is cpio then you would type:

```
# cpio -iavcd </dev/{device_name}
```

where *{device_name}* is the name of the tape or diskette drive on the Unix system. In Unix, device names are similar to:

Device Type	Device Name
5.25" Diskette	rfd096
3.5" Diskette	rfd1135ds18
1/4" Tape	rct0

The device names shown above are typical for an SCO Unix system. The device names on your system may differ. For example, the device names used on an IBM RS6000 AIX are:

Device Type	Device Name
5.25" Diskette	*not used*
3.5" Diskette	/dev/fd0h
1/4" Tape	rmt0

If the `cpio` command is successful, a list of filenames is displayed as they are loaded into the installation directory.

If the command shown is other than `cpio`, such as `'installpkg'`, follow the standard installation instructions of your Unix system for installing software products.

7. Print and read the file `README`, loaded into the installation directory. This is an ASCII text file and contains information pertinent to the current release of IQ.
8. Store your original media in a cool, fireproof location.

Execute the Install Script

1. You must be in root account.
2. You must be in the Bourne shell to install IQ. Make sure you are in that shell. You can usually start a Bourne shell by typing **/bin/sh**.
3. Run the installation script by typing `./iqinstall`. This script moves the files to their proper locations in `/usr/bin` and `/usr/lib/iq3`. `iqinstall` attempts to determine if there is an existing IQ version 2 on your system. If it determines that IQ version 2 has been previously installed, it notifies you and asks if you wish to make IQ for Unix the default.

IQ2 appears to be installed in `/usr/bin`. Only one version of IQ may be installed in `/usr/bin`. Please select one of the following choices:

Retain existing files in `/usr/bin` as the default IQ

Install new IQ3 files in `/usr/bin` as the default IQ

Existing files will be relocated to `/usr/lib/iq2`. Please consult your Installation Guide for "Running IQ2 and IQ3 on the same system"

Enter number of your choice

If you answer 1, `iqinstall` will leave your existing files in `/usr/bin` untouched and will install all of its files under `/usr/lib/iq3`. If you answer 2, `iqinstall` will move the existing IQ version 2 files to `/usr/lib/iq2` and will replace any files under `/usr/bin` with the shell scripts from your IQ for Unix media.

If you answer 2, `iqinstall` will move the existing IQ version 2 files to `/usr/lib/iq2` and will replace any files under `/usr/bin` with the shell scripts from your IQ for Unix media.

NOTE: To run both versions 2 and IQ for Unix, please refer to the section titled "Running IQ2 and IQ for Unix On The Same System". Also note that IQ2 is no longer a DCI supported product.

4. `iqinstall` creates the following directory if not already created:

/usr/lib/iq3

5. `iqinstall` uncompresses the files and checks their crc values to verify that no damage has occurred to the files.
6. `iqinstall` will display a list of available languages and ask you which language you want IQ to use. All other language files are stored in their proper directories under /usr/lib/iq3. To eliminate a language, remove the associated directory.

Languages available at the time of this publication's printing are English, German, French, Spanish, Danish and Dutch. If other languages are available, they will be included in the menu option that `iqinstall` displays.

Enter the language number of your choice.

7. Files are moved to their proper destinations on your system and the install directory is deleted.
8. Once the script is ended, check to make sure all files contain their proper permissions for use.
9. Exit from the root account and login as a normal user.

Errors During Installation

If an error occurs while running `iqinstall`, the files `errfile` and `instllog` will remain in the directory from which you are installing. (If no errors occur, these files are deleted by the installation process.) These files may be displayed on your screen at the time that the installation process fails and contain information on what failed with the installation. In all cases of error, an error message is displayed on your screen and the installation is aborted.

The following are possible explanations to errors that may occur during the installation:

You are not logged in as root and attempt to run `iqinstall`.

You are attempting to run `iqinstall` from the `/bin`, `/usr/bin`, or `/usr/lib/iq3` directories.

You are attempting to run `iqinstall` from a directory that does not contain the contents of the IQ distribution files.

`iqinstall` cannot determine the user-id of the account from which you are installing.

The installation fails due to the crc values of a file not matching the value for the file stored in `iqchksum`, or the `iqchksum` file is determined to be corrupted.

The installation fails to find the `iqchksum` file required to verify correct crc values.

The installation fails to locate the crc program file.

Configuring IQ for Your Terminals

NOTE: You do not need to follow the instructions here if you have already installed a previous version of IQ (IQ version 2 or IQ for Unix). Instead, copy your `iqcap` file from your old IQ directory to your new directory and then continue with the instructions in the section for Activating IQ.

To use any IQ program or utility (other than the `MAKETERM` Utility), IQ must be configured to respond to the character sequences generated by the keys on your terminal.

When IQ is executed, it first reads the `TERM` environment variable setting for your Unix shell. It looks for the file, `iqcap`, for information pertinent to running on your terminal type based on the `TERM` variable.

You must run `MAKETERM` for each different type of terminal using IQ. To define mnemonics for a terminal, you must run `MAKETERM` from that terminal. Each time you run `MAKETERM` you will create (or change) the entry for that terminal only in the `iqcap` file being used. A predefined `iqcap` file is provided with IQ. You may wish to alter its contents for your specific terminal types. The `iqcap` file included with IQ contains terminal definitions for the following terminal types:

wyse50	wyse60	ansi	xterm	vt100	
vt220	hp2392	tvi920	qvt102	AT386	
hp	7901+	att610bct	ibm5081	ibm3151	
ibm3161	ibm3162	ibm3101	sun-cmd	altos4	altos5

IQ can use any box-drawing capabilities your terminal has. If your terminal can use special characters or alternate sets for them, IQ will do so to improve the appearance of the screens only if you define them to IQ with MAKETERM. If your terminal does not support these characters, IQ will use the +, -, and | characters to draw its boxes.

You can also use MAKETERM to set the character sequences used to control terminal operation. This includes clearing the terminal, positioning the cursor and so on. Most of these entries will be found in the default terminal definition and should not be changed. This information is also stored in `iqcap`.

Keyboard definition is a two-layer process:

1. First you will use the MAKETERM utility to assign physical keystrokes on your terminals to IQ mnemonics. This ensures that terminals of different types invoke the same mnemonic when a particular key is pressed. It is the mnemonic to which IQ reacts, not the physical keystroke.

In this step, you can also define keystrokes to substitute for any keys not found on a particular terminal's keyboard. For example, you might select CONTROL-E (^E) to serve in place of a missing END key. Keyboard definition information is stored in `iqcap` (or any other file pointed to by the variable IQCAP).

2. Second, and optionally, you can create an `iqkey.dat` file to map each mnemonic to different logical keys. Logical keys are referred to by generic names such as the Help Key, the Backup key, and so on. This process is discussed in your *IQ System Manager's Guide*.

Using MAKETERM

MAKETERM looks in the following places for terminal information, and uses the first set of information it finds for the terminal you are using:

The file specified by the `iqcap` environment variable

`iqcap` in the current directory

`iqcap` in a directory that is part of your path

The file specified by the `TERMcap` variable or the entry specified by that variable

`/etc/termcap`

The file specified by the `TERMinfo` variable

`/usr/lib/terminfo`

If MAKETERM cannot locate a terminal definition, its definition screens will be blank and you must enter each definition yourself.

NOTE: If you have a TERMINFO system, make sure there is no `/etc/termcap` file which might give MAKETERM incorrect terminal information.

Define the TERM Variable

Make sure the `TERM` environment variable is set to the terminal type you are using. (This variable is usually correct and normally does not need to be changed.)

MAKETERM uses this variable for the name of the terminal's entry in the `iqcap` file, and IQ uses it to look up the terminal characteristics in that file.

Start MAKETERM

Before starting MAKETERM, set the `TERM` environment variable to the proper terminal type. Also, make sure you are in the directory where you loaded IQ (`/usr/lib/iq3`). Start MAKETERM like this:

```
$ maketerm
```

You must have read/write permission for the `iqcap` file. (IQ users need only read permission for this file.)

Define the Terminal

MAKETERM presents you with its main screen:

- The upper left portion of the screen shows where the default terminal information is being taken
- The next line shows the location of the `iqcap` file
- In the upper right portion of the screen is the type of terminal being defined

Make sure each of these headings is correct before continuing.

The first MAKETERM screen is the keystroke definition screen. This screen allows you to assign keystrokes to the mnemonics used by IQ. Each mnemonic is identified by a number. These numbers are used to define or change the mnemonics. To change any of the entries shown on the screen, follow these steps:

- A. Type the number shown next to the mnemonic you want to define.
- B. Press ENTER

MAKETERM prompts you to press the key to be used for the entry.

- C. Press the key you wish to use for that function. (Press only the key--do not press ENTER).

MAKETERM captures the character sequence generated by the keystroke and assigns it to the key.

When IQ executes, it performs a mapping process to determine handling of keystrokes. The character string generated by each keystroke is compared to the entries defined here. If it matches one of the entries, IQ recognizes the key as a key with special meaning and executes the proper IQ function. If not, IQ processes the character sequence as normal characters.

If a key cannot be processed as a normal character and is not recognized as a key with special meaning, IQ beeps and does nothing.

The following table lists each mnemonic and its default use. Entries surrounded by asterisks must be correctly defined for IQ to operate properly. All other entries are optional, but they provide additional capabilities and should be defined if possible.

Entry Number	Mnemonic	Default Function
1	k1	Help
2	k2	Keyboard/Menu toggle key
3	k3	Select/Display fields key
4	k4	Field Commands key
5	k5	Delete Line key
(6)	kh	Home key--Moves the cursor on the Report Writer (INTELLIGENT QUERY ONLY)
7	ku	Up Arrow
8	kl	Left Arrow
9	kr	Right Arrow
10	kd	Down Arrow
11	kE	End key--Used to finish selection of items from a list
(12)	Ku	Page Up key--Used for paging in lists
(13)	kb	Backspace key
(14)	kI	Insert key—Used to toggle insert mode.
(15)	kD	Delete key—Used to delete a character

(16)	Kl	Beginning of Line key--Used to move the cursor to the beginning of the line.
(17)	Kr	End of Line key--Used to move the cursor to the end of the line
(18)	Kb	Back Tab key--Used to move the cursor back 8 spaces
(19)	Kd	Page Down key--Used for paging in lists
(20)	Ek	Backup key--Used to backup one menu level or selection
(63)	k6	None assigned--usually defined as function key 6, which can then be assigned to specific functions by iqkey.dat
(64)	k7	None assigned--usually defined as function key 7
(65)	k8	None assigned--usually defined as function key 8
(66)	k9	None assigned--usually defined as function key 9
(67)	k10	None assigned--usually defined as function key 10
(68)	Ep	Insert Literal (Report Writer)

Editing the Terminal Control Strings

The terminal control strings entry screen is used to define the character sequences IQ uses to control terminal operations: clearing the terminal, positioning the cursor, and so on. This screen is option 98 from the keystroke definition screen.

Most of these entries will have been taken from the default terminal definition and should not need to be changed, nor do you need to verify them at this time.

However, check the cm string before proceeding. Instructions for setting this string

are described in the sub-section "Setting The Cursor Motion String". Also, if there are required entries--those whose number is bracketed with asterisks in the table below--that are blank on your screen, you must complete them before proceeding.

If IQ seems to have any problems drawing, clearing, or displaying information on your screen, return here and verify each of those entries--they must be defined correctly for the terminal to function properly. Consult your terminal manual to determine the correct settings for your terminal.

Entry Number	Name	Function
(21)	am	Automatic Margin Control See the note in the section "Defining Box-Drawing Characters".
(22)	xs	Standout Glitch Required? (TRUE/FALSE). Set this to <i>True</i> if the reverse video does not turn off when moving the cursor. Otherwise, enter <i>False</i> . Note that this is a Boolean function, not a string entry. Normally <i>True</i> is required only for some Hewlett Packard and WYSE terminals.
(23)	sg	Number of screen positions for reverse attribute (0 or 1)
(24)	mc	Number of characters sent by each function key (1 to 9)
(25)	al	Add blank line
26	ce	Clear to end of line
27	cl	Clear screen
28	cm	Cursor motion (cursor positioning)
(29)	ch	Move to column 1
(30)	le	Move one space left
(31)	nd	Move one space right
(32)	dl	Delete line

(33)	is	Terminal initialization
(34)	ke	Turn off keypad transmit mode
(35)	ks	Turn on keypad transmit mode
36	se	End reverse video mode (standout)
37	so	Start reverse video mode (standout)
(38)	ve	Make cursor visible
(39)	vi	Make cursor invisible
(40)	as	Start alternate character set --See "Defining Box-Drawing (Graphics) Characters" for more information
(41)	ae	End alternate character set --See "Defining Box-Drawing (Graphics) Characters" for more information
(42)	ti	Terminal initialization. Determines how %i and %d substitutions are calculated for cursor motion--in relation to window or screen. See your terminal documentation for further details
(43)	te	Terminal reset on exit. See above

Setting The Cursor Motion (cm) String. If your system uses the terminfo database to define terminals, the cm string (cursor motion, used to position the cursor on the screen) might be incorrect. If you run IQ and the first screen (the logo screen) appears jumbled, this is most likely the problem. In this case, you will need to change the cm entry on MAKETERM's string entry screen.

Use the following table to find the cm string for your terminal:

Terminal Type	cm String
Adds Regent 100	^K%+\040%B ^P%
Adds Viewpoint	\EY%+\040%+\040
ADM 3A,31,42 (Lear Siegler)	\E[%I%d;%dH
Altos 2, Altos 3	\E[%I%d;%dH
ANSI Standard CRT	\E[%I%d;%dH
AT&T 4410, 5410, 4425	5425 Teletype
AT&T 4415, 5420	\E[%I%d;%dx
AT&T 4420 (Teletype)	\EY%+\040%+\040
DEC vt52	\EY%+\040%+\040
DEC vt100	\E[%i%d;%dH
Hazeltine 1420	\E^Q%r%.%+\040
Hazeltine 1500	~^Q%r%.%.
Hewlett Packard 2621, 2626, 2645	\E&a%r%dc%dY
IBM 3101, 3161, 3163	\EY%+\040%+\040
Televideo 912, 920, 925, 950	\E=%+\040%+\040
Wyse 30, 50, 60	\Ea%i%dR%dC
Wyse 75, 85	\E[%i%d;%dH
Wyse 100	\E=%+\040%+\040

If your terminal is not listed and you are familiar with defining termcap information, consult your terminal manual for the appropriate cm string. If you are not familiar with terminal definitions, try the following:

- Remove any %p1 or %p2 strings from the entry
- Change any %2d or %3d strings to %d
- Remove any %c strings
- Change any %p1%"%+ strings to %+|040

If none of these work, try using one of the entries shown for the ANSI standard CRT or the DEC VT52. If the terminal still does not work, please contact your technical support group.

Setting the Number of Characters String. The mc string controls how many characters IQ reads at one time from the keyboard buffer. If this string is set incorrectly, IQ may not read all the characters of a keystroke and interpret the keystroke incorrectly. Similarly, if the characters from multiple keystrokes are in the keyboard buffer, IQ may read one-and-a-fraction keystrokes, also causing it to misbehave. The default for this string is 3. Like most other MAKETERM strings, this one is normally set correctly without user intervention. These instructions are provided in case you have difficulty with IQ misinterpreting keystrokes.

The mc string may be set to a value from 1 to 9. Follow these guidelines to determine the best value for your terminal:

- A. If all keys defined in IQCAP (that is, visible on the MAKETERM main screen) generate the same number of characters use that number.
- B. If the arrow keys generate one number of characters and many or most other keys another number, and these two numbers when multiplied by each other result in 9 or less, use the result. For instance, if arrow keys generate 2 characters and function keys generate 3, use 6 as your entry.

- C. If the arrow keys generate one number of characters and many or most other keys another number, and these two numbers when multiplied by each other result in a number greater than 9, use a multiple of the arrow key number of characters that is greater than the number of characters for other keys. For instance, if arrow keys generate 2 characters and function keys generate 5, use 6 as your entry.

These guidelines emphasize the arrow keys because those are the keys that are most likely to be pressed several times in rapid succession.

The most frequent symptom of an incorrectly set mc string is IQ incorrectly reacting as if the Backup key has been pressed, either consistently or occasionally.

If this occurs:

- As an initial step in attempting to solve the problem, try increasing the value for the K1 configuration record (keyboard minimum wait) in `iqconfig.dat`.
- If this symptom cannot otherwise be cured, set the K2 configuration record (keyboard escape key flag) in `iqconfig.dat` to a value of 1 and define the mnemonic `Ek` to a key other than the Escape key. This will take care of the situation. After doing this, however, you cannot use Escape as the Backup key--you can only use the key assigned to the `Ek` mnemonic. Note that if you do this for one terminal, you must do it for all terminals. See the *IQ System Manager's Guide* for more information about the K2 record.

If you are able to set the mc string to exactly the number of characters as are generated by your arrow keys, you may wish to set the K1 configuration record (keyboard minimum wait) in `iqconfig.dat` to a value of 1. This will make IQ's response to the keyboard faster. In the previous examples described above, you would set the K1 record to 1 for example 1, and to 3 for examples 2 and 3. (Networks may require a higher setting.) See the *IQ System Manager's Guide* for more information about the K1 record.

Defining Box-Drawing (Graphics) Characters (optional)

The graphics characters entry screen is used to define the characters IQ uses to draw boxes on your terminal. Special box-drawing characters are not required to run IQ, but provide a more visually appealing product if available. If box-drawing characters are not defined, IQ will draw boxes with +, - and | characters. After you have installed IQ, you may wish to return here and define or redefine the box-drawing characters. However, it is not a necessary step during installation.

IQ uses 2 different types of boxes:

- Normal Boxes—Used for regular windows, non-emphasized windows, the help window, etc. They are drawn with the single-line box-drawing characters.
- Emphasized Boxes--Used to indicate the active window or menu on the screen. They are drawn with the double-line box-drawing characters.

If your terminal doesn't have both kinds of box characters, you can improvise. You might use graphics characters for the emphasized box and let the regular boxes default to '+-|', or vice-versa. You can also use the same graphics characters for both kinds of boxes--it's not essential that the two kinds of boxes look different.

To make changes to these entries, select option 99 from the keystroke definition screen. MAKETERM then displays the graphics entry screen. To define the box-drawing characters, type the number of the entry you wish to define. You are then asked for the decimal value of that character. Enter the value for the desired character. These are the box drawing character codes:

Entry Number	Name	Function
44	UI	Normal upper left box corner
45	UL	Emphasized upper left box corner
46	Ur	Normal upper right box corner
47	UR	Emphasized upper right box corner
48	T1	Normal left tee box character

49	TL	Emphasized left tee box character
50	Tr	Normal right tee box character
51	TR	Emphasized right tee box character
52	L1	Normal lower left box corner
53	LL	Emphasized lower left box corner
54	Lr	Normal lower right box corner
55	LR	Emphasized lower right box corner
56	Ho	Normal horizontal bar
57	HO	Emphasized horizontal bar
58	Ve	Normal vertical bar
59	vE	Emphasized vertical bar
60	SO	Solid fill character
61	Hb	Upper half-box character
62	HB	Lower half-box character

If your terminal uses an alternate character set to draw graphics characters, you must define the terminal strings as and ae on the terminal control strings entry screen. IQ will then switch to the alternate character set before printing the box-drawing characters defined here. If your terminal does not require an alternate character set to draw boxes, leave the as and ae entries undefined.

NOTE: IQ normally leaves the position at the bottom right corner of the screen blank because some terminals do an automatic scroll or line feed if a character is placed there. If you wish to write to this position to complete the box, then set the `am` (automatic margin) entry on the terminal control strings entry screen to `False`. This allows IQ to write to the position to complete the boxes. If the terminal then does not work properly (that is, it scrolls information up the screen), you should set this entry back to `True`.

Saving Your Changes

End `MAKETERM` by pressing `q` at the menu. If you have made any changes to any of the entries, you will be prompted to save them. Answer `y`. If you have left some required entries undefined, you will receive an error message and `MAKETERM` will return you to the menu to finish the definition process. Otherwise, your changes will be saved in the `iqcap` file (creating a new one if necessary).

Activating IQ for Unix

Prior to running IQ for Unix, you must obtain the correct license for the various options available to IQ. This section describes those options and combinations of options available with activating IQ for Unix.

Licensing Options

IQ for Unix is licensed in 4 different forms:

- **IQ Runtime**

This type of license allows users to use the standard IQ product. No capability to define or maintain data dictionaries (`DDMAINT`) or export information to other file types (`IQACCESS`) is provided.

- **IQ Runtime and DDMAINT**

This type of license allows users to use the standard IQ product and define and maintain data dictionaries. No capability for the export of information to other files types (IQACCESS) is provided.

- **IQ Runtime and IQACCESS**

This type of license allows users to use the standard IQ product and export information to other file types. No capability to define or maintain data dictionaries (DDMAINT) is provided.

- **IQ Runtime, DDMAINT and IQACCESS**

This type of license allows users all the capabilities described above.

SSNs and DCI Security Daemon

IQ is a companion product purchased and licensed separately. DCI product licensing and security consists of two parts: 1) your unique 8-digit license number, and 2) your Software Security Number (SSN). It is the combination of a license number and SSN that allows the operation of IQ on your computer. In order to run IQ, you must obtain a proper SSN.

IQ for Unix requires DCI's security daemon `/etc/passport` version 3.1 or higher. Verify your system is running the correct version of `/etc/passport`. The revision number of `/etc/passport` can be found in `/etc/DCI/passport.log` or its equivalent log file. If your system does not have the correct version of `/etc/passport`, you must obtain and install it. Please refer to the DCI Passport Reference Guide for installing `/etc/passport` version 3.1 or higher and changing your SSN.

NOTE: IQ for Unix will not run properly with IQ version 2 SSN's. Make sure your SSN is for IQ for Unix.

Setting the Unix Environment for IQ

This section discusses environment variables that directly impact your IQ for Unix operation. DCI recommends that you add the commands discussed in this section to either:

- The `.profile` or `.login` file on your system
- or-
- A script you create specifically to run IQ for Unix

Your IQ System Manager's Guide contains a complete list of the system variables used by IQ.

NOTE: In addition to the variables listed here, `TERM` must be set correctly for your terminal type.

Setting IQDD

Purpose: To allow users to run IQ from directories other than the one in which it is installed. When IQ executes, it first searches the default directory for the data dictionary. If IQ does not find the dictionary there, it searches the directory specified by `IQDD`. It also uses this variable to search for stored procedures.

Examples: `IQDD= directory ; export IQDD`
 `IQDD=/usr/lib/iq3 ; export IQDD`

Setting IQCAP

Purpose: To allow IQ and IQ utilities to locate the terminal characteristics file.

Examples: `IQCAP=directory/filename; export IQCAP`
 `IQCAP=/usr/lib/iq3/iqcap; export IQCAP`

See the "Configuring IQ for Your Terminal" section for more information.

Setting IQDATA

Purpose: To allow IQ to locate the database files defined in your data dictionary. When IQ executes, it searches the directories listed in your IQDATA variable. If IQDATA is not defined, it searches the current directory for files defined in the data dictionary.

Examples: IQDATA=directory ; export IQDATA

 IQDATA=/usr/files/data:/u/files/data ; export IQDATA

Setting IQEDITOR

Purpose: To invoke an external editor instead of IQ's built-in line editor when editing a procedure.

Examples: IQEDITOR=/usr/bin/vi; export IQEDITOR

 IQEDITOR=vi; export EQEDITOR

Setting IQOUTPUT

Purpose: To force all files containing IQOUTPUT to be written to a specific directory.

Examples: IQOUTPUT=directory; export IQOUTPUT

 IQOUTPUT=/usr/iq3/output; export IQOUTPUT

Testing the Installation

Before you start IQ, be sure you have completed all the steps previously listed, including:

- Setting up the Unix environment
- Configuring IQ for your terminals

- ❑ Activating IQ
- ❑ Upgrade steps

If these steps have been completed successfully, start IQ at your system prompt:

```
$ iq
```

IQ should begin execution and the logo screen should appear. If it does not, or you receive an error, check to make sure that you have followed the installation instructions, including setting the environment variables as appropriate. If the logo screen appears jumbled, check and correct the `iqcap` entry for the `cm` string, which controls cursor motion and positioning by using `MAKETERM`'s string entry screen. See the "Configuring IQ for Your Terminals" section for more information. Error messages are explained in Appendix A of the IQ User's Guide.

If the logo screen displays correctly, follow these steps.

Press ENTER

IQ displays a list of database categories from the data dictionary. Each category represents a grouping of fields from the database.

Press ENTER to select the highlighted database category

IQ displays the main menu.

Select Output Types by pressing ENTER

IQ displays the output types menu.

Select Columnar by pressing ENTER

As soon as you select Columnar, IQ prompts for the type of columnar output.

Select Detail Only by typing D

IQ displays a list of the fields available in the current database category. It is this list that you will use to test your keyboard setup. To do so, follow these steps:

Press the Down Arrow Key and hold it down for a few seconds

The highlighted bar should scroll through the list. The list should remain displayed and no other action should occur.

Press ENTER

Hold the Down Arrow key down for a few seconds and immediately afterwards press the End key and the Backup key (usually ESCAPE) in rapid succession.

The highlighted bar should scroll through the list. One item should remain highlighted while this occurs. You should end up at a prompt telling you that your current procedure is not saved and asking for verification that you wish to exit.

Type a *Y*

IQ displays the database category selection box.

Press the Backup key to exit IQ (or turn to your IQ User's Guide and begin the IQ tutorials from the demo data dictionary and data files installed in the `/usr/lib/iq3/iq3demo` directory). The tutorial is documented in the IQ User's Guide.

If you have problems with IQ incorrectly interpreting keystrokes, the most likely cause is an incorrectly set `mc` (number of characters) string. See the "Configuring IQ for Your Terminals" section for instructions for the correct setup of this string. In particular, if IQ seems to consistently, or occasionally, act as if you pressed the Backup key when you pressed other keys, an incorrect `mc` entry may be the cause.

Testing should be performed with your system or network under a moderate to heavy load, if possible. You should repeat this testing on every type of terminal for which you have configured IQ. You should also test as a user (not root) to verify that permissions are correct.

If the testing is successful, you may want to improve IQ's response time to the keyboard by lowering the value set in the `K1` configuration record in the `iqconfig.dat`. This value determines the interval between reads from the keyboard buffer. The default value is 3. A higher value is sometimes needed in network situations to prevent IQ from reading partial character sequences from the buffer. On the other hand, a lower value may work and will make IQ response to keyboard commands snappier. See the IQ System Manager's Guide for more information about this record.

If IQ only processes 10 (or fewer) records from your database, repeat the steps titled 'Activating IQ.'

If You Are Upgrading From Version 2

1. Install IQ for Unix version 3
Follow all the steps given earlier for the installation.
2. Run ddrefmt utility program.
The ddrefmt utility program is documented in the System Manager's Guide. Please note that ddrefmt must be able to open your data files.
3. Run ddload utility program. The ddload utility program is documented in the System Manager's Guide.
4. Update Configuration Information
 - A. Copy your iqcap in your old directory to your new directory.
 - B. Print a copy of iqconfig.dat from your old IQ directory and a copy from your new IQ directory. Compare them and update the newer copy with the same settings you have been using. Use IQ's configuration program (IQCONFIG) and/or edit iqconfig.dat to make the changes.
 - C. Print iqprfile.def from your old directory. Edit iqprtcfg.dat in your new directory to match this, first deleting any printers you do not use in iqconfig.dat.
 - D. Add the IQDATA directory to your environment, specifying those directories which contain your database files.

NOTE: File formats for IQ for Unix configuration files are contained in the *IQ System Manager's Guide*. File formats for older versions are contained in Appendix E of the *IQ User's Guide* for those versions.

Running IQ2 and IQ for Unix On The Same System

If you are upgrading your system from IQ version 2, you have the option during installation to retain the older version files in /usr/bin or install the newer version and overwrite the existing files in /usr/bin.

If you choose to retain the older version, you can still run IQ for Unix by changing to the /usr/lib/iq3 directory and typing ./iq. Make sure you have installed the proper /etc/passport daemon, entered in the correct SSN and verified your PATH and IQDD environment variables do not conflict with IQ version 2 programs and files.

If you choose to install the newer version, your existing IQ files in /usr/bin are moved to a newly created directory, /usr/lib/iq2. Should you have a need to run the older version, you can change directories to /usr/lib/iq2 and run the older version by typing ./iq. Make sure your PATH and IQDD environment variables do not conflict with IQ for Unix programs and files. If you have a need to revert back to the older version as your default IQ (/usr/bin), you can move the older version files from /usr/lib/iq2 back to /usr/bin. The IQ for Unix files that get installed in /usr/bin are shell scripts that execute the actual files stored in /usr/lib/iq3.

In both cases, the DCI security Passport daemon version 3.1 or higher and your new SSN allow both IQ version 2 and IQ for Unix to run.

NEW FEATURES IN IQ FOR UNIX

This section describes the new features in IQ for Unix. Please read your *IQ System Manager's Guide*, *IQ User's Guide* and *IQ Data Dictionary Reference* for more detailed information

New Output Types

Label (Multiple-Up) Output. Built-in support for producing labels is now provided. Labels can be printed in as many columns across the page as necessary. They are defined using a visual layout screen based on the version 2 Report Writer screen. This feature can be used to print any detail-only multiple -up report. Support for Multiple label formats (or forms) is provided, along with the ability to select the default printer for labels at the time you define them. (INTELLIGENT QUERY ONLY)

Matrix (Cross-Tabulation) Output. Support for matrix reports. By specifying a column field, a row field, a field to base a calculation upon and the type of calculation, IQ automatically produces a completely formatted matrix report. (INTELLIGENT QUERY ONLY)

Columnar Output. This output replaces some and enhances other displays and print output types. It also computes totals and averages. It provides four types of output: detail, detail with subtotal, subtotals only and grand totals only.

New "Transfer" Options. IQ Access provides transfer ability to dBASE, Excel, Word for Windows, Word Perfect, several Lotus 1-2-3 formats and more. Files can be created in both a detailed and summary form. (IQ ACCESS ONLY)

Performance Enhancements

Improved Sorting. IQ now makes improved use of keys in primary files when performing a sort, both ascending and descending. Prior versions did not take full advantage of existing key fields for sorting purposes. In certain cases, this can result in significant performance increases.

Enhanced Search on Key. Before version 3, IQ performed sequential searches when selecting against key fields in primary files. This enhancement allows IQ to “search on key” to find desired records rapidly. The result may be a dramatic performance improvements when searching against key fields. Version 3 also features improved algorithms for searching on multi-part alpha keys.

New Data Dictionary Format. The IQ data dictionary has been converted to an ISAM file. This results in performance improvements for both IQ and DDMAINT.

New Temporary Field Functions

Substring and Concatenate Functions. New functions to perform substring and concatenation functions have been added. Substrings can be extracted from either alphanumeric or date fields.

PROMPT at Execution. The PROMPT command been added to the menu system.

New INITIALIZE Command. An INITIALIZE command has been added to give increased control over the manipulation of temporary fields. It gives you the ability to create a temporary field with an initial value. It is very useful when trying to count or perform control break logic.

Other New Features and Functions

Device Independence. All types of output can now be sent to printer, display, or file. IQ prompts for the output destination at execution. It can be pre-selected via a new utility menu selection.

Format Independence. Output is now formatted by default for the format (printer, display, or file) appropriate for the type of output and the output destination. Format can be overridden by a new selection from the utilities menu.

Output-Independent Author and Title. All output is displayed using the standard previewer previously provided only with the Report Writer. This eliminates the multi-line wrapping for output that exceeds 80 columns and provides a consistent previewer for all types of output. The Back Tab key is now supported in the previewer as well as the Report Writer.

Command Line Database Category Option. The database category to be used can be passed on the command line. This allows you to bypass the database category screen and proceed directly to the main menu. Note that the current database category is now always displayed on the main menu.

Shell Capability. IQ now supports the capability to shell out to the operating system. You can perform any desired commands and return to IQ just where you left it. The shell key and the shell prompt are configurable.

File and Field Level Security. Full IQ-controlled security is available to limit access to data at both the field and record level. Password security is also now available. See the IQ System Manager's Guide for information.

Case-Sensitive IF/SEARCH. Comparisons against alphanumeric fields can now be made in a case-sensitive manner. Version 2 performed its comparison after converting both values to uppercase. This enhancement allows users to control how the comparison is made.

User-Defined Editor. Support for a user-defined editor is now provided. The editor is specified via a system variable called IQEDITOR. When this variable is defined, IQ invokes the desired editor instead of IQ's line editor.

Comment Capability. The ability to include comments in procedures is now available. This allows users to create documentation for their more complex procedures.

Easier Counting Ability. A stand-alone SEARCH command can now be used to count records. (Prior versions required that some type of output be defined before you could execute a procedure.) You can now create a procedure with just a SEARCH command to find the number of matching records.

Enhanced Link Key Conversions. This new feature allows IQ to perform linkages using numeric fields stored in different formats. IQ converts numeric fields from one type to another during the linkage process. Support is provided for most numeric field types.

Hex Values for DDMAINT. DDMAINT has been changed so that all constant values are entered using ASCII hexadecimal representation.

New Configuration Options

Configurable Keystrokes and Screen Labels. All of the keystrokes used to operate IQ can now be configured. In addition, the matching screen labels can now be tailored to the end-user's needs.

New iqconfig.dat Layout. The primary configuration file has been changed to make edits easier. The complete file layout is included in your IQ System Manager's Guide.

New iqconfig.dat Items. In addition to items in support of other new features, many items were added to iqconfig.dat, including those to control whether or not IQ prompts for case-sensitivity when you set up a search, and prompts evoked when creating file output using IQ ACCESS.

Simplified and Enhanced Printer Definition. Printer definition has been simplified by the consolidation of all printer definitions (other than the default printer) into a single file, iqprtcfg.dat. Definitions of several of the most popular printers are provided. Start-of-job, end-of-job, form feed, normal, and compressed mode control codes can now be configured.

New Command Names. Several command names have changed in version 3. A list of these changes can be found at the beginning of the "IQ Command Reference" chapter of your IQ User's Guide.

New Formatting Abilities

New Formatting Options. Formatting options for zero suppress, leading/trailing minus and plus signs, and DB/CR (debit/credit) strings are now supported for formats in the data dictionary and (INTELLIGENT QUERY ONLY) in the Report Writer.

Expanded Date Formatting. The ability to format dates has been greatly increased. In addition to providing three additional formats for four-digit years, the format of any date field can be specified in either the dictionary or (INTELLIGENT QUERY ONLY) in the Report Writer.

Internationalization. Support for 8-bit character sets has been added. In addition, there are operations to support local currency, date and time conventions. A user-defined collating sequence is also provided.

User Interface Improvements

Select/De-Select of items in Windows. All selection boxes now support the ability to de-select previously selected items by moving the highlighted bar to them and pressing ENTER.

Single-Letter Hot Keys for Menus. All menu options can be selected via a single keystroke.

Circular Menus. All menus wrap to the top item when pressing Down Arrow at the end of the menu, and to the bottom when pressing an Up Arrow at the beginning.

More Indicator. Multiple-window selection lists now show a “More” indicator to inform the user there are multiple windows.

Display Database Category. The current database category is always shown on the main menu.

Report Writer Enhancements (Intelligent Query Only)

New Subheading and Page Footing Areas. New report areas have been added for both subheadings (similar to subtotals) and page footings.

Report Writer Template. Columnar output can now be used as a starting point (or template) for the Report Writer. If an existing columnar output is present when you enter the Report Writer, it is used to start the report layout. This allows you to prototype with columnar output and then use the Report Writer to finish the report.

Improved Suppress Feature. The suppress field attribute in the Report Writer is changed to suppress the display of a field until the value changes. This is an improvement over the former method of “suppress until break”. The ability simultaneously to trim and suppress a field has also been added to the menus.

Improved Trim Availability. The trim attribute is now available on alphanumeric fields in all report areas.

Word-Wrap. The capability to wrap text fields on word boundaries within a fixed rectangular block has been added. This allows IQ to support extremely long text fields.

Font Attributes. Support for fixed-width bold, underline, and italic print styles is new.

Date Field Formatting. Formatting of date fields is now possible.

Page Up and Page Down. Support has been added for the paging keys during report design.

Data Dictionary Improvements

New Data Dictionary Layout. The data dictionary layout has been changed to allow for version 3 features. The new data dictionary format adds new data elements to existing records, increases the size of some existing data elements, and adds 3 new record types (C, H and S) to the dictionary to support the new security features. The data dictionary file type has been changed from sequential to ISAM.

New Data Dictionary use of Key and Data Buffer Areas. The buffer areas for both the key and data areas have been separated. This makes it easier to define one-to-many relationships and Multi-Record Type files. In previous versions, a calculation had to be made when defining the above to offset the data area from the key in the buffer.

Data Dictionary Field Changes. These fields have been changed or added:

File Information Fields	Old Size	New Size	Comments
File type	2	3	
Type of group ID	2	3	
Security level		3	New Fields- initialized to 0
Sub-field offset of group ID		5	

Record Information Fields	Old Size	New Size	Comments
Record Numbers	2	3	
Record sequence		3	New fields - initialized to 0
Type of group ID field		3	
Sub-field Offset of Record ID		4	
Record ID code field			Converted to hexadecimal representation with a maximum size of 32 hexadecimal digits

Key Information Fields	Old Size	New Size	Comments
External field name		45	New fields - initialized to blanks
Date Format		1	
Date Layout fields		6	
Sub-field offset key		4	New field initialized to 0
Record number		3	New field, for future use- initialized to blanks
Key Number	2	3	
Size of key part	2	5	
Type of key part	2	3	

Data Field Information Fields	Old Size	New Size	Comments
Security level		2	New field- initialized to 0
Field name	20	45	
Field type	2	6	
Record number	2	3	
Key Number	2	3	
Size of key part	2	5	
Type of key part	4	5	

Database Category Information Fields	Old Size	New Size	Comments
Sequence number		3	New field - initialized to 999
Security level		2	New fields- initialized to 0
Help message number		4	
File number			No longer included in the database category record

Linkage Information Fields	Old Size	New Size	Comments
Link part		2	New field - initialized to 0
Link-To key number		3	
Link-To key part		2	
Date format		1	
Date layout		6	
From field record number		3	
From field storage size	2	5	
From field type	2	3	
Type of chain field	2	3	
Reference and match field information	2		No longer used (but included in dictionaries converted from IQ v.2)

Procedure Information Fields	Old Size	New Size	Comments
Date and time of last reference field		14	New field

INSTALLATION & CONFIGURATION OF IQ FOR WINDOWS 95/NT

There are two major steps in installing IQ for Windows 95/NT:

1. Loading IQ files onto your system--the actual program installation. This is detailed in this section for users upgrading from older versions of IQ.
2. Product activation, if appropriate.

The file “files.doc”, in your IQ installation directory lists and identifies all distributed IQ files.

System Requirements

IQ for Windows requires:

Microsoft Windows 95 or Windows NT, and up to fifteen (15) megabytes of free disk space.

Additionally:

IQ for Windows will run on any IBM PC and compatibles, 386-based processor or better, and requires a minimum of four megabytes of RAM. As with any Windows application, increasing the amount of memory available to Windows will increase application speed.

A mouse or other pointing device is recommended. A printer is optional.

ODBC Requirements

IQ for Windows supports all ODBC version 2 database drivers with ODBC API core and level 1 functions and grammar. Note these requirements for using IQ with ODBC databases:

You must obtain and install a compatible 16-bit ODBC database driver from your database vendor or a third party. ODBC drivers are not distributed with the product. For a list of ODBC drivers and vendors, contact Microsoft at 800-426-9400 (voice) or 800-727-3351 (FAXback service).

You must create an ODBC data source before you can use IQ to access your data. See your ODBC driver documentation for information about this process. In most cases, you will use the ODBC database administrator utility to do so. You can check the file `odbc.ini`, in your Windows directory, for a list of data sources that have been set up on your system.

When setting up your data source, note that:

If your database driver supports setting the name of a default database, you must do so.

You should also set any logon parameters that are not included in the database driver's logon dialog.

You must have your IQ directory as part of your path. Many ODBC drivers switch to the database directory when they connect to the database. If this occurs and your IQ directory is not part of your path, IQ will be unable to find files it requires.

The E9 entry in the configuration file `iqconfig.dat` must be set to a value of 1 (its default) or not be present to allow use of ODBC databases, `iqconfig.dat` includes an E9 entry set to a value of 0 (disable ODBC support).

IQ itself requires no specific system variables to be set for use with ODBC support. See your ODBC driver documentation for information about any system variables used by your driver.

IQ reads ODBC databases similarly to the way it reads SQL databases. Unless noted otherwise, all information in IQ documentation pertaining to SQL databases also applies to ODBC databases. This includes both relational and non-relational

databases. Notes pertaining to “non-SQL” databases do not apply to ODBC databases.

New Installation

From Media:

The installation program will copy the required files to your hard disk and configure your system.

To begin: (note: Windows NT users must log on to the administrator account)

1. Run Windows and insert IQWIN Disk #1 into any floppy drive on the system.
2. From inside Windows, pull down the File menu from the Program Manager Menu (or from Windows 95, press Start) and select RUN.
3. Windows will prompt you for the name of an application to run; type the command:

```
a:setup
```
4. and press enter. If you are loading setup from drive B, type b:setup instead of a:setup.
5. Follow the instructions on your screen.

From Internet:

1. Download the proper version of IQ for Windows and run the executable file from the Run section located in the Start menu of the task bar.
2. The installation program will lead you through the installation process. Follow the instructions on your screen.
3. The installation program creates a program group for IQ FOR WINDOWS on Windows NT when installation is complete.

Backing up IQ for Windows

If your daily backup routine looks for specific files, add `ddmaster.dat` (the IQ data dictionary), `ddmaster.idx` (the IQ data dictionary index) and `*.p??` (stored procedure files) to the list.

If you create dictionaries with different names, add them to the backup list as well. Although the file and database category definitions in these files are usually static, pointers to the files containing user-created procedures are added to the dictionary and must be present to run the procedures.

If you are not familiar with IQ, DCI recommends that you complete the IQ for Windows tutorial. After you complete the tutorials, you are ready to begin using IQ for Windows to access your own data.

Activating IQ for Windows 95/NT

In order to use the full version of IQ for Windows, you must first provide proper SSN information. We suggest you start IQ now so that you can verify that it is correctly installed, and enter your license information. Please refer to DCI Passport Reference Guide for installation and activation of IQ for Windows.

NOTES: IQ utility and data dictionary programs are all character based and must be run from the DOS prompt or using Windows' File Run menu selection. Alternately, you can set up PIF files for these and add program items (icons) for them to your IQ for Windows program group.

If you ever change your IQ program item so that the working directory is other than an IQ directory (character or Windows), be sure to add the IQ directory to your path.

If you start IQ with the `-d` command line option and no dictionary name, IQ lets you use a dialog box to select a dictionary. In that case, Windows changes the working directory to be that of the file you select, and the IQ directory must be on your path for IQ to run successfully.

IQ dictionaries can be set up to allow users to select data files from a dialog box. If you do this, Windows changes the working directory, and the IQ directory must be on your path for IQ to run successfully.

Upgrading From Older Versions

- ❑ **Installation:** Follow the steps in this chapter if you are upgrading from an older version of IQ for Windows. It describes how you can share configuration and dictionary files between these products.
- ❑ **Start Windows:** The installation program will copy the required files to your hard disk and configure your system.
- ❑ **To Begin:** First, before installing new version of IQ for Windows, copy the following files into a temporary directory:

ddmaster.dat	ddmaster.idx	iqconfig.dat	iqprtcfg.dat
iqjoins.dat	iqexcs.dat	iqhelp.fld	
iqlimits.dat (or other security file as named by your system manager)			

If any of these files do not exist in your old directory, don't worry. Several of them are optional files.

- ❑ **Also copy:**

Stored procedure files (*.p??)

Any additional IQ data dictionary files and the index for them. (These files have the extension `.dat` and `.idx`, respectively. However, note that not all files with these extensions are data dictionary files.)

Follow the procedures from New Installation in this chapter.

NEW FEATURES IN IQ FOR WINDOWS 95/NT

The following describes the changes made for IQ for Windows 95/NT:

New Server Package

IQ SmartServer, as documented in other IQ manuals, is not part of DCI's IQ for Windows 95/NT.

Subset Reporting

IQ now will create or update an IQView to correspond to any ASCII data file you create with IQ. Subset reporting features include:

- ❑ Add or update an IQView in any data dictionary including the one currently in use.
- ❑ Instant access with IQ to the data files that you create.

ODBC Support

IQ for Windows now supports 16-bit ODBC database drivers. See chapter 7 of your System Manager's Guide for detailed information about requirements and features. ODBC support is included with every copy of IQ. It can be enabled by use of the E9 entry in iqconfig.dat. (This is enabled by default)

ODBC databases are handled as if they were SQL databases. In IQ documentation, all notes for SQL users apply to ODBC users as well.

Date-Time and Interval Support

IQ now supports date-time and interval columns in many SQL database types. IQ formatting entries in the IQCONFIG utility and in iqconfig.dat have been extended to allow you to specify default formats for these columns. The INITIALIZE, SUBSTRING and CONCATENATE commands have been extended to support these column types.

A new command, CONVERT (Data/Convert from the menu bar), allows you to convert intervals to numbers and vice versa. You can also use it to convert a numeric column from one time unit to another. For instance, you could convert an interval to a number of days, and later convert that to a number of hours.

Better Speed

IQ now caches query results and does not re-query the database unless needed.

IQ now uses a new sort package for improved sort performance.

Report Preview

You can now keep an output window open while you work in report design. You can limit the number of rows output during preview by selecting Data/Limit from the report design window's menu bar. This feature works together to give you a live data preview capability that doesn't require the complete execution (or even leaving report design).

Default Report Areas

We have added user-configurable default report area sizes. IQ now adds blank areas to new reports. The RB, RF, RG and RH entries in iqconfig.dat control how many lines each new area has.

Default Design Changes

Column selection has been reworked to give you a floating window from which you can select multiple columns. You can select to have automatic column headings added for each column you add to the body area of your report.

You can also set sort and search criteria while you are in report design.

Object resizing now works more smoothly as you change and reformat objects.

Label Design

IQ now has a complete label design facility. You can design character or graphical mailing labels using the same features and interface as in report design, including live data previewing. IQ is shipped with many pre-defined label forms and you can customize the label layout as necessary.

Matrix Changes

You can now transfer matrix output direct to spreadsheet files, including Lotus 1-2-3 and Excel.

Chart Changes

Charts no longer use scientific notation. We've also added title and author display to the chart output window and provided a black and white output option for printing charts.

We've optimized chart processing so that IQ does not need to retrieve data when you have made only cosmetic or minor changes to a chart that has already been executed.

Picklist Support

You cannot select “compare to” values from a picklist of database values when you are setting up a search. You can also create temporary columns using Prompts that let the user select from (a) a picklist of database values or (b) a picklist of values generated by an IQ procedure.

Help Changes

The IQ help system has been changed to include the entire IQ User’s Guide. DBA kits also include the entire IQ System manager’s Guide as a help file.

Procedure Export & Import

IQ’s procedure export and import functions have been added to the File menu. This lets you share procedures with users on other systems without exiting IQ or using a separate utility.

Toolbar Hints

We’ve added toolbar hints as well as a preferences item to turn them on and off.

Easier Column Selection

We’ve added horizontal scroll bars to column selection lists to help you see the names of columns with very long names.

New Error Message Handling

We've changed error message handling so that a complete informative error message is displayed instead of just error codes. (Help messages are located in the file iqmsg.txt. They are no longer included in the User's Guide)

New Configuration Options

The following list includes all options added after the initial release of IQ for Windows version 4.

Label	Description
DT	Default Column Type for Date-Time Columns
E9	Enable ODBC Database Access
F7	Numeric Replacement Character in Table Names
F8	Alpha Replacement Character in Table Names
F9	Array Element Designator
GB	WordPerfect Multinational Character Set (IQ Access)
R0	Label Form Name (Graphical labels)
R1	Label Form name (Character labels)
R2	Number Of Labels Across
R3	Label Width
R4	Label Height
R5	Label Printer Name
R6	First label Print Column (Character labels)
R6	Left Page Margin (Graphical Labels)
R7	Top Page Margin (Graphical labels)
R8	Printable Area Width (Graphical labels)
R9	Printable Area Height (Graphical labels)
RA	Number of Labels Down (Graphical labels)

RB	Default Number of Report Body Lines
RF	Default Number of Report Footer Lines
RG	Default Number of Report Grand Total Lines
RH	Default Number of Report Header Lines
T5	Time Characters
T6	Time Separator Character
T7	Default Interval Format
T8	Default Date-Time Format
T9	Date-Time Offset
TT	Default Column Type for Time Columns
XB	Optimized Sort Usage
XP	Enable Batch and Background Processing
XQ	Force Reads Before Calculations
XT	Output of Date and Numeric Values When No Record matches

DEFINING UniBasic & dL4 DATABASES FOR IQ

Introduction

This section is a supplement to the IQ Data Dictionary Maintenance Reference. It contains specific information to be entered while using DDMAINT to set up UniBasic and dL4 databases for use with IQ. Do not use it to replace the more-detailed DDMAINT instructions in the IQ Data Dictionary Reference. Please read the file README, in your IQ directory, for updates to your IQ documentation and information specific to the current release.

IQ supports the UniBasic version database files and dL4 versions 2 and higher Portable and Full-ISAM database files.

Please read the IQ Data Dictionary Reference prior to running DDMAINT.

NOTE: dL4 Full-ISAM database files have some restrictions at the time of this writing. Please check the README file on your distribution for any updated information.

Data Dictionary

A data dictionary specific to your database is needed by IQ to access your data. File and file information in the data dictionary contains two levels of information:

1. **File Information:** Contains all the files you want IQ to access. These definitions are invisible to the user.
2. **Database Category Information:** These are pointers to one or more file definitions. When a database category points to more than one file, it includes information about how the files are related. Database categories are visible and provides the user a means of selection.

In creating a data dictionary, you must define your file or files prior to defining a database category. You must have knowledge of your file structures to properly define them using DDMAINT.

Running DDMAINT

The program name for creating, modifying or examining a data dictionary is DDMAINT. The data dictionary is stored in an ISAM (Indexed Sequential Access Method) file. The default name is ddmaster.dat. You can use another name by specifying it after the -d command line option. For example, to create or modify a data dictionary named payroll.dat, start DDMAINT like this:

```
$ ddmaint -dpayroll.dat (IQ for Unix)
```

```
c:>ddmaint -dpayroll.dat (IQ for Windows)
```

You may create multiple data dictionaries by using different names for each one. For example, you may have one data dictionary for payroll and another for sales orders. See the IQ System Manager's Guide for more information about command line options.

DDMAINT searches the current directory for the dictionary specified (or for ddmaster.dat, if you do not use the -d option). If DDMAINT does not find the dictionary there, it searches the alternate directory specified by the IQDD environment variable, if it is set. If this search is unsuccessful, it creates a new data dictionary in the IQDD directory, if set, or in the current directory if IQDD is not set. See the IQ System Manager's Guide for more information about this variable.

If you build a data dictionary using a name other than the default ddmaster.dat, you must start iq with the -d command line option to use that dictionary. See the IQ System Manager's Guide for more information about command line options.

Defining Files and File Information

The first step in setting up the data dictionary for your files is to define them to IQ. The data dictionary you create describes your database to IQ. By creating a data dictionary, you tell IQ:

- How to retrieve your files
- What database files to use
- Where the files are located
- What types of files are in the database
- The fields you want IQ to read
- How the files are related

A file definition may consist of four parts:

File Information: Information about the file itself. (Required for all files.)

Record Type Information: Information about multiple record types (if any).

File Key Information: Information about keys to the file (if any). This is required for indexed files if you want IQ to use the keys to access them. Keys must also be defined for any file you want to use as a to-file in a multiple-file database category with indexed linkages.

Data Field Information: Information about the data fields in the file. This is required for any fields you want IQ users to be able to access.

UniBasic and dL4 File Information

The following sections describe the entries for each file type supported in IQ for Unix and Windows and descriptions for entering information for each file type. File types can be one of four major types:

UniBasic Files:	Indexed, Contiguous and Formatted (aka Item)
dL4 Portable Files:	Indexed, Contiguous and Formatted (aka Item)
dL4 Full-ISAM Files:	Full-ISAM.
Text Files:	ASCII 7-bit Text (follow the instructions in the IQ Data Dictionary Reference when defining these file types).

NOTE: For information on the use of file types other than those mentioned above, please contact your DCI distributor.

File Name

Enter the name of the file you are defining. You may include absolute or relative path. Additionally, IQ supports several ways to have a single file definition apply to more than one file using filename replacement characters and wildcard characters. See the IQ Data Dictionary Reference for more information.

NOTE: Filenames with replacement and wildcard characters are not supported on IQ for Unix as of this writing.

IQ uses an environment variable IQDATA as a search path for locating files. You need to set this variable prior to running IQ. IQ prefixes its value to any file name that does not contain another variable as part of the name. IQ searches directories in the order in which they appear in the IQDATA definition. It stops searching as soon as it locates a directory containing a file that matches the file name. If the filename contains a wildcard character, IQ retrieves only from the first directory in which it locates a file.

NOTE: DDMAINT appends a unique number to the file name to keep each file definition separate. When creating a definition, you will see this number as the last part of the file name. When creating multiple definitions of a file, keep track of each file number and what definition it represents.

File Types

File Type	IQ File Type
UniBasic Files	Enter 261 for Indexed, Contiguous, Formatted, and Item.
dL4 Portable Files	Enter 262 for Indexed, Contiguous, and Formatted files.
dL4 Full-ISAM	Unix users - Enter 44 Windows users – Use ODBC

Record Length

File Type	IQ File Type
UniBasic Files	Any non-zero value. Defaulted at runtime based on file's actual record length.
dL4 Portable Files	
dL4 Full-ISAM	Unix users - Enter actual record length. Not applicable for IQ for Windows.

Header Records

File Type	Description
UniBasic Files dL4 Portable Files	Enter any non-zero value. Defaulted at runtime based on file's actual record length.
dL4 Full-ISAM	Unix users - Enter actual number of header bytes. Not applicable for IQ for Windows.

Is This File Indexed?

File Type	Description
UniBasic Files dL4 Portable Files	If you are defining an indexed file, answer Y (yes). Otherwise, answer N (no).
dL4 Full-ISAM	Enter N (no). dL4 Full-ISAM files do not support use of the index at this time of writing for Unix. It is supported on IQ for Windows via ODBC.

Primary Read Sequence

File Type	Description
UniBasic Files	<p>Enter 1 to have IQ process the file in the sequence of the first key defined. In general, you should enter a 1 for an indexed file.</p> <p>Enter 0 to have IQ first read the lowest key value and its associated data record. All subsequent data records will be read sequentially starting with the next data record.</p> <p>If you did not define the file as indexed, this prompt is skipped.</p>
dL4 Portable Files	
dL4 Full-ISAM	<p>Unix users - This prompt is skipped</p> <p>Not applicable for IQ for Windows</p>

Security Level

File Type	Description
All File Types	<p>If IQ's file security is enabled, only users whose security level is higher or equal to the entry you make for Security Level will be able to access this file. See the <i>IQ System Manager's Guide</i> for more information. This is not the only entry that controls access.</p>

Location of Deleted Record Mark

File Type	Description
UniBasic Files	Enter any non-zero value. Defaulted at runtime based on the actual value in file header
dL4 Portable Files	0
dL4 Full-ISAM	Unix users – 0 Not applicable for IQ for Windows.

Size of Deleted Record Mark

File Type	Description
UniBasic Files	1
dL4 Portable Files	0
dL4 Full-ISAM	Unix users – 0 Not applicable for IQ for Windows.

Deleted Record Mark

File Type	Description
UniBasic Files	FF
dL4 Portable Files	<CR>
dL4 Full-ISAM	Unix users - <CR> Not applicable for IQ for Windows.

Data Field Separator Marks

Field Type	Description
All File Types	Press Enter (default).

Multiple Record Types

File Type	Description
UniBasic Files	Enter Y (yes) if your file contains records with different physical layouts. Otherwise, enter N (no). See Chapter 4 of your <i>IQ Data Dictionary Reference</i> for more information.
dL4 Portable Files	
dL4 Full-ISAM	Enter N (no). MRT type files are not supported for Full-ISAM files.

Defining File Key Information

If you answered *Y* (yes) to the question, **Is This File Indexed**, you must enter key information. If your file contains multiple keys, you do not need to define all of them. If you wish to access more than one key in a file, you will need to define a separate file entry for each key you wish to access. IQ can only process an index file definition with one key defined at a time. You may copy a file definition for one indexed file then modify the file's key information.

Each key may be composed of one or more parts. The parts do not need to be contiguous in each key. Defining a key field does not make the field accessible to IQ users. Any key field that you want users to access must also be defined as a data field.

NOTE: Attempting to link to a key containing both 7-bit and 8-bit data may cause unexpected results.

Index File Name

File Type	Description
UniBasic Files	Enter the key file name (upper case) as it appears on your system.
dL4 Portable Files	Enter the key file name (file name ending with .idx) as it appears on your system.
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Key Number

File Type	Description
UniBasic Files dL4 Portable Files	Enter the index number. The first index number of the file is 1. NOTE: Only one index may be defined for a single file definition. To access a secondary index, you must define a new file in your data dictionary.
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Key Part Number

File Type	Description
UniBasic Files dL4 Portable Files	If the key you are defining contains a single data field, enter 1 as the key part number. If the key you are defining has several parts, enter 1 for the first part and increment by 1 for each remaining part.
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

External Field Name

File Type	Description
UniBasic Files dL4 Portable Files	Leave this entry blank.
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Type of Key Part

File Type	Description
UniBasic Files dL4 Portable Files	Enter 0 for alphanumeric. Enter 19 (binary field) for CALL 15 packed field. Enter 24 for an alpha subfield that is contained within another field.
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Location of Key Part

File Type	Description
UniBasic Files dL4 Portable Files	Enter 0 for key type 0 or 19. Enter byte offset for key part (type 24).
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Storage Size of Key Part

File Type	Description
UniBasic Files dL4 Portable Files	Enter size in actual bytes of the key part you are defining. IQ will allocate key buffer area to the largest key size, either user specified or the largest key size in the file.
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Date Format

File Type	Description
UniBasic Files dL4 Portable Files	Enter the date format type as described in the <i>IQ Data Dictionary Reference</i> .
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Date Layout

File Type	Description
UniBasic Files dL4 Portable Files	Define the storage format of the field as MnDnYn, where letters M, D, and Y represent the month, day and year, respectively, and n represents the number of characters used by that part of the data. For more information, please refer to the <i>IQ Data Dictionary Reference</i> .
dL4 Full-ISAM	Not supported at this time for Unix. Not applicable for IQ for Windows.

Adding additional Key Parts: Once you have completed information for a key part, DDMAINT asks **More? Yes/No**. Select **Yes** to continue defining key parts. Select **No** when you have defined all desired key parts for this index entry.

Defining Field Information

This step defines the actual fields a user can select when using IQ. You do not need to define all the fields in the file to IQ. Some fields you may wish to leave out because they may either be meaningless to the user or contain confidential information. In those cases, do not define the fields or use IQ's hierarchical security system (see the *IQ System Manager's Guide* for more information).

For each field you include for use with IQ, you must define the field name, type, location, storage and output sizes. If you want your users to access key fields, you must define them as data fields.

Field Name

File Type	Description
All File Types	<p>Enter a name for each data field you wish to define. The name must be unique from all other fields in this file. The field name must begin with a letter and may not include these characters:</p> <p style="text-align: center;">.,()+';'"*?</p> <p>It may not contain a space and cannot be an IQ keyword. See the <i>IQ Data Dictionary Reference</i> for a list of IQ keywords.</p>

Sequence Number

File Type	Description
All File Types	When presenting fields to the user, IQ lists them in order by sequence number. You can enter a sequence for each field, the same number for all fields (alphabetical), or use several sequence numbers and assign each to a different group of fields. Defaults to 999 (alphabetical listing).

External Name

File Type	Description
All File Types	Leave this entry blank.

Field Type

File Type	Data Type	Description
UniBasic Files (Unix users only)	String Data	<p>If the field you are defining is in the data record:</p> <p>Enter 00 for Alphanumeric Data.</p> <p>Enter 01 for ASCII Numeric, Explicit Decimal Point.</p> <p>Enter 15 for ASCII Numeric, No Explicit Decimal Point.</p> <p>Enter 16 for Alphanumeric Date Field.</p> <p>If the field you are defining is in the Key:</p> <p>Enter 00 for Alphanumeric Data.</p> <p>Enter 24 for defining a field within another field.</p> <p>For other field types, please refer to the <i>IQ Data Dictionary Reference</i>.</p>
	Numeric Data	<p>Enter 3 for 16-bit signed integer.</p> <p>Enter 3 for 32-bit signed integer.</p> <p>Enter 171 for 3-word Base 10000 floating.</p> <p>Enter 171 for 4-word Base 10000 floating.</p> <p>Enter 171 for 2-word Base 10000 floating.</p> <p>Enter 171 for 6-word Base 10000 floating.</p> <p>Enter 172 for 16-bit signed BCD integer.</p> <p>Enter 172 for 32-bit signed BCD integer.</p> <p>Enter 173 for 2-word IRIS BCD floating.</p> <p>Enter 173 for 3-word IRIS BCD floating.</p> <p>Enter 173 for 4-word IRIS BCD floating.</p> <p>Enter 173 for 5-word IRIS BCD floating.</p> <p>Enter 174 for 2-word IEEE BCD floating.</p>

File Type	Data Type	Description
UniBasic Files (Unix users only)	Numeric Data	<p>Enter 174 for 3-word IEEE BCD floating.</p> <p>Enter 174 for 4-word IEEE BCD floating.</p> <p>Enter 174 for 5-word IEEE BCD floating.</p> <p>Enter 175 for 2-word IEEE floating scaled X 100.</p> <p>Enter 175 for 3-word IEEE floating scaled X 100.</p> <p>Enter 175 for 4-word IEEE floating scaled X 100.</p> <p>NOTE: For files, all floating point number types can be entered as any of the above mentioned types using the same number of words. For example, any 3-word number type can be entered as any 3-word number type. IQ determines the actual type by reading this information stored in the file's header when the file is opened.</p>
	Packed Data (Unix users only)	<p>Enter 19 (hex field) for UniBasic "CALL 15" packed fields. This is considered the source field. You must again define the field as type 10 (externally derived field) using the field name given to the type 19 field as the "External Info" entry.</p> <p>To get the correct values of a CALL 15 packed field, you must define it twice, once as a type 19 (consider this the packed information), and again as a type 10 (consider this the unpacked information).</p> <p>The location of unpacked field (field type 10) can be any positive integer number as it uses the location of the source packed field (field type 19). The storage size for the unpacked field should be twice the size of the packed field. Please refer to the <i>IQ Data Dictionary Reference</i> for additional information.</p>

File Type	Data Type	Description
dL4 Portable Files	String Data	If the field you are defining is in the data record: Enter 00 for Alphanumeric Data. Enter 01 for ASCII Numeric, Explicit Decimal Point. Enter 15 for ASCII Numeric, No Explicit Decimal Point. Enter 16 for Alphanumeric Date Field. If the field you are defining is in the Key: Enter 00 for Alphanumeric Data. Enter 24 for defining a field within another field. For other field types, please refer to the <i>IQ Data Dictionary Reference</i> .
	Numeric Data	Enter 30 for 16-bit signed integer. Enter 30 for 32-bit signed integer. Enter 174 for 2-word IEEE BCD floating. Enter 174 for 3-word IEEE BCD floating. Enter 174 for 4-word IEEE BCD floating. Enter 174 for 5-word IEEE BCD floating. NOTE: Unlike numeric field types on IQ for Unix, dL4 numeric fields require the correct field type, except for formatted files. Formatted files may be entered as any numeric type with the same word-length. This is important if you have converted a UniBasic file to a dL4 portable file. See Numeric Data types under UniBasic Files.
	dL4 Date Data (not supported on Unix)	Date variable fields are not supported at the time of this writing for Unix. Please check the README file on your distribution for more information. Use field type 176 for dL4 date fields on IQ for Windows. Use sizes of 2, 4, or 6 for 1%, 2%, and 3% date fields, respectively. In addition, use Y0M0D9 for "Date Layout", 10 for "Output Size of Field", and use MM/DD/YYYY for "Output Format".

File Type	Data Type	Description
dL4 Full-ISAM Files (Unix users only)	String Data	<p>If the field you are defining is in the data record:</p> <p>Enter 00 for Alphanumeric Data.</p> <p>Enter 01 for ASCII Numeric, Explicit Decimal Point.</p> <p>Enter 15 for ASCII Numeric, No Explicit Decimal Point.</p> <p>Enter 16 for Alphanumeric Date Field.</p> <p>Key fields are not supported at this time. Please check the README file on your distribution for updated information.</p>
	Binary Data	Enter 19 to display the field in hexadecimal notation. You may interpret as other field types where appropriate.
	Date Data	Enter 16 for Alphanumeric Date Field. Stored as YYYYMMDD. dL4 date fields are not supported.

Location of Field

File Type	Description
All Files	<p>For fields defined in the data portion, enter the Record Offset Location plus 1 (Byte offset in BASIC is origin 0).</p> <p>For fields defined in the Key, enter a field location of 0. Use of key fields may allow faster searches and sorts on the field.</p> <p>WARNING: Do not change the F1 record in <code>iqconfig.dat</code> unless your keys are always duplicated in your data record.</p>

Record Number of Field

File Type	Description
All Files	<p>If you are defining a MRT (multiple record type) file, enter the record number that this field belongs to. The record number is assigned by DDMAINT when you define the record types in the file. If you have a field which is present in all record types, enter 0 for the record number. This field will be included whenever fields from any record type are output.</p> <p>For non-MRT files, this field will be skipped.</p>

Storage Size of Field

File Type	Description
All Files	<p>Enter the storage size of the field in bytes.</p> <p>For arrays, use the storage size for a single element.</p>

Sub-Field Offset

File Type	Description
All Files	<p>For substring fields (field type 24) enter the offset, in characters, for the first character within the field.</p> <p>Other fields, leave this entry 0.</p>

Sub-Field Size

File Type	Description
All Files	<p>For substring fields (field type 24) enter the number of characters to retrieve.</p> <p>Other fields, leave this entry 0.</p>

Date Format

File Type	Description
All Files	<p>Date fields can be defined to IQ in two ways:</p> <ol style="list-style-type: none"> 1. As any numeric field type. To define any numeric field as a date, enter <i>Y</i> for the Date Format. 2. As a date field (type 16) <p>For the special date field (16), enter 1 (ASCII numeric Date) for the Date Format.</p>

Date Layout

File Type	Description
All Files	<p>IQ can process dates stored with all 3 parts (year, month and day), in two parts (year and day of year) and Julian Dates. For Julian dates, see your <i>IQ System Manager's Guide</i> for more information.</p> <p>Enter M2D2Y2 for dates stored as MMDDYY. Enter Y2M2Y2 for dates stored as YYMMDD. Enter Y4M2D2 for dates stored as YYYYMMDD. (All dL4 Full-ISAM date fields are stored in this format.)</p> <p>NOTE: If you are defining a date field, the meaning and location of each date part must be defined. If you do not enter a Date Format, this field is skipped.</p> <p>NOTE: Use of dL4 date variable types in Portable Files are not supported at the time of this writing on IQ for Unix. Please check your README file for more information.</p>

Output Size of Field

File Type	Description
All Files	<p>Enter the default size, in characters, that IQ should reserve for this field when formatting output.</p> <p>NOTE: The size may be smaller (or larger) than the storage size of the field. For type 01 fields, it must include any positive/negative sign, or decimal separator stored as part of the field. For other field types, it does not need to include the sign character, but must include a byte for the decimal separator if decimal digits are present.</p> <p>NOTE: For hex fields (type 19), allow two characters for each character to be displayed in hexadecimal notation.</p> <p>WARNING: If the output size is smaller than a value found in the field, IQ truncates the value.</p>

Decimal Digits

File Type	Description
All Files	Enter the number of decimal (fractional) digits to appear to the right of the decimal point.

Array Elements (1st and 2nd Dimension)

File Type	Description
All Files	<p>If you are defining an array, enter number of elements in the array.</p> <p>IQ can process either a single element of an array, or every element of the array.</p> <p>One-Dimensional Arrays. Enter the number of occurrences of the field for the 1st dimension. Enter 0 for the second dimension.</p> <p>Two-Dimensional Arrays. For the 1st dimension, enter the number of times that the first dimension occurs. For the 2nd dimension, enter the number of occurrences in the 2nd array.</p> <p>See the <i>IQ Data Dictionary Reference</i> for more information.</p>

Array Group Size

File Type	Description
All Files	Leave blank.

Security Level

File Type	Description
All Files	<p>If you are using IQ's hierarchical security, enter a security level from 0 to 99. Only users with an access level equal to or greater than this level will be able to use IQ to access the field.</p> <p>All other users, leave this entry 0.</p>

Help Message Number

File Type	Description
All Files	<p>Enter the help message number you have defined for the field you are defining.</p> <p>All other fields, leave this entry blank.</p> <p>See Chapter 5 of the <i>IQ Data Dictionary Reference</i> for more information on IQ's customized help messages.</p>

Output Format

File Type	Description
All Files	<p>This entry controls the IQ output format. If you leave this entry blank, IQ formats the fields as follows:</p> <p>Alphanumeric fields appear as they are stored in the file.</p> <p>Numeric fields are output with leading zeros suppressed and trailing sign if negative.</p> <p>Date fields appear in the format specified in <code>iqconfig.dat</code>.</p> <p>Alphanumeric formatting. The output formatting for alphanumeric fields can be specified using a combination of screening characters and insertion characters. See the <i>IQ Data Dictionary Reference</i> for more information.</p>

External Info

File Type	Description
UniBasic Files	For CALL 15 packed field type 10, enter field name of defined type 19 (binary) field. See Field Types in this section for more information on CALL 15 packed fields.
dL4 Portable Files	Leave this entry blank.
dL4 Full-ISAM Files	Leave this entry blank.

Database Category Notes

Please refer to the *IQ Data Dictionary Reference* for details on how to create database categories and set up linkages between files. The following link types are supported:

Type 1	Index Link and Index Chain Link
Type 2 or 3	Relative Record and Relative Record Chain Link
Type 4	Byte Offset Link
Type 5	Binary Search Link
Type 6	First Record Link
Type 10	Coordinated Link

NOTE: When defining a type 1 index link, if a numeric field is linked to a key field which is zero filled, then use 0 for "Sub-Field Offset". If the to-key space is filled, then use any positive number for "Sub-Field Offset". The to-key is assumed to be right justified with zeroes or spaces. You will first need to define the "from field" to be a field type 24, a sub-field, so that you may enter proper values for "sub-field offset". After completing the definition, you will need to go back and change the "from field" type to its proper field type value.

NOTE: When defining a type 5 binary link, both the link-from-field and the link-to-field have to be an Alphanumeric field. IQ will strip off the most significant bit (bit 8) in both fields. IQ field types 0,1,15,24 and 16 always have their msb stripped.

NOTE: When linking to a Multi-Record Type (MRT) file, IQ links to the first defined record number. Typically, this is record number 1 (one).

NOTE: Link type 10 operates based on ordinal, not cardinal, record numbers. That is, first record links to first record, second to second, etc., regardless of the record numbering used in BASIC.

NOTE: When defining a type 2 or 3 Relative Record Link, IQ takes the First Real Data Record (FRDR) into consideration.