

# ISS 45



*IPM / PPM Technical Reference  
Version 7.7*

## ISS45 7.7 IPM / PPM Technical Reference

<b>Date of Issue</b>	<b>Product Identification Number</b>	<b>Part Number</b>	<b>Brief Description</b>
January 1995	45001/006	80303889	Version 7.0
November 1996	45001/006	80328025	Version 7.3
February 1999	45001/006	Electronic Library 80602986	Version 7.6
August 2000	45001/006	89000054	Version 7.7 (Unchanged)

**Copyright® International Computers Limited 1995-2000  
All rights reserved.**

This publication is protected by federal copyright law into any human or computer language in any form or by any means, electronic, mechanical, magnetic, manual. No part of this publication may be copied or distributed, stored in a retrieval system, or translated or otherwise, or disclosed to third parties without the express written permission of ICL Retail Systems.

ICL Retail Systems makes no representation or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for a particular purpose. ICL Retail Systems further reserves the right to revise this publication and to make changes from time to time in the contents hereof without obligation of ICL Retail Systems to notify any person or organization of such revision or changes.

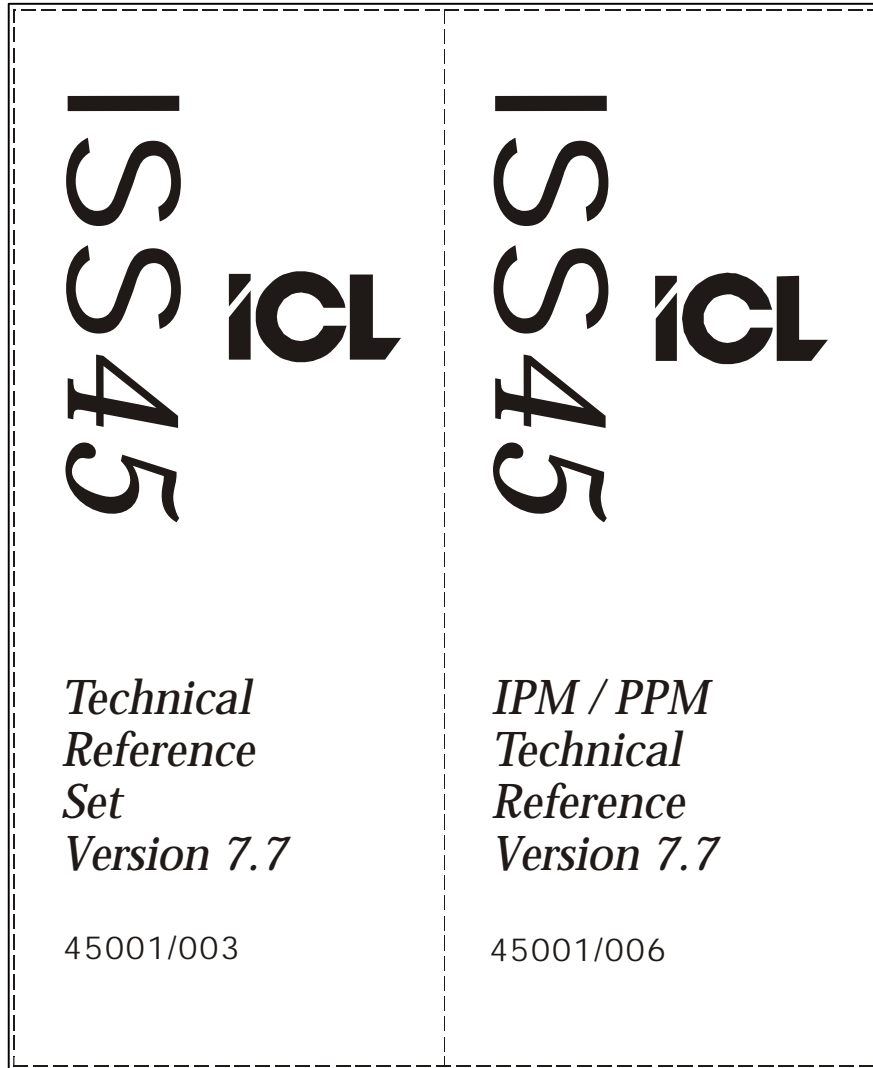
ICL Retail Systems has prepared this manual for use by users, authorized third parties and personnel of ICL Retail Systems as a guide to the proper installation, operation, customization and/or maintenance of ICL Retail Systems equipment and software. The drawings and specifications contained herein are the property of ICL Retail Systems.

Address comments and corrections to:

ICL Retail Systems  
ISS45 Program Director  
2933 Bunker Hill Lane  
Suite 101  
Santa Clara, CA 95054

This documentation is designed for placement in an ICL binder that can be ordered separately. To order the binder, contact your sales representative. Indicate PIN 45007/002 and/or part number 80192817.

This sheet contains spine cards that can be used to identify the binder for the appropriate documentation. Cut one of the cards along the dotted lines and insert it in the binder's spine pocket. Discard the remaining cards or save them for later use.



---

# ***Table of Contents***

<b>Introduction.....</b>	<b>3</b>
<b>Operating an IPM Application (and IPM itself) .....</b>	<b>4</b>
<b>IPM controls .....</b>	<b>5</b>
Menus .....	5
Screens .....	9
Fields.....	10
Push Buttons.....	14
Combo Box .....	19
Errors .....	19
Messages .....	20
Forms .....	21
List Box .....	29
Qlist Box .....	31
View Box.....	34
Query .....	35
Qcombo Box .....	38
<b>Reports .....</b>	<b>40</b>
Print Formats .....	43
POS Formats.....	43
<b>Parameters .....</b>	<b>50</b>
IPM Parameters.....	50
POS Parameters .....	55
POS Format Parameters.....	56

<b>Import / Export .....</b>	<b>57</b>
<b>Tools .....</b>	<b>58</b>
<b>Data Dictionary .....</b>	<b>59</b>

## Introduction

**IPM (ICL Presentation Manager)** is a package for the presentation of, and the simplifying of, development and code management.

The package is based on two main parts:

- **IPM user interface:** For the creation and updating of IPM user files. These files are in the database containing all IPM elements that can be used in an IPM application
- **IPM user library:** Enables the use of an IPM database in a 'C' language application.

By using the IPM library and user interface, you can create IPM applications using advanced MMI (man machine interface) features, without writing any MMI handling code. These include: mouse interface, menus, forms, reports etc.,

IPM lets you customize screens, reports, or any IPM element, without you having to change any source code.

Multi-lingual applications are supported, simply by changing an IPM database file. Through IPM you can define file structures with fields (currently Q-Dex files only), screen interfaces (Forms) with various elements, and print layouts for both conventional and POS printers. Support is included for different layouts of receipt and journal printing. Some IPM features can be operated without any code being written. For example: queries, table updates, lists for Q-Dex files. Other features, however, require compiled code, thus enabling sophisticated applications with unlimited options. To make life easier, IPM includes a 'skeleton' option from the necessary menus, that can build basic C language code, with all possible user entries. This C code can be used 'as is' in an IPM application, and only the data handling must be added. The IPM user interface is itself an IPM application. This fact can help you to become familiar with IPM controls and elements. Generally, IPM itself uses every IPM control.

## Operating an IPM Application (and IPM itself)

IPM and IPM applications use a standard MMI, controlled by IPM libraries.

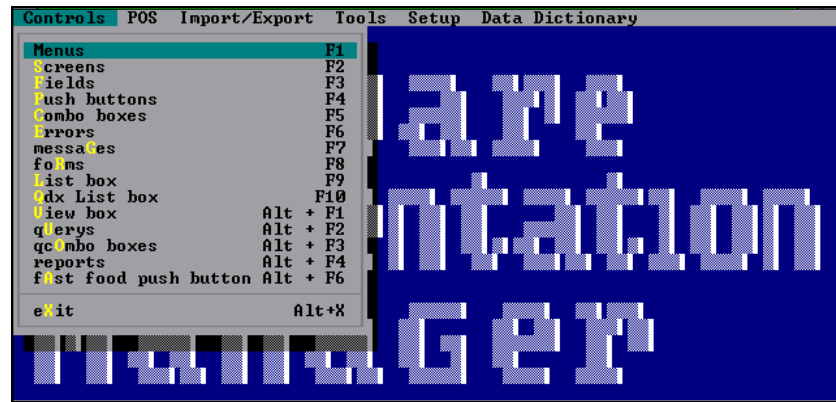
The screen interface relies on several rules, used in most IPM screens:

<b>Feature</b>	<b>Operation</b>
Move between fields and controls	Use <i>Tab/Shift-Tab</i> . In some cases (such as a form with push buttons and/or fields only, the <i>up-arrow</i> and <i>down-arrow</i> keys can also be used).
Radio-Button fields	(For example): Justification of text in push-buttons definition, is activated by the <i>space bar</i> . When activated, the rest of the radio-buttons from the same group are reset.
Combo boxes	Open with the <i>right-arrow</i> key, close with the <i>left-arrow</i> key. Select in the box with <i>Enter</i> . When the box is closed, <i>up arrow</i> and <i>down arrow</i> keys can be used to select. To exit from the field, use <i>Tab/Shift Tab</i> .
List boxes	Use the <i>up-arrow</i> and <i>down-arrow</i> keys to move the selection bar, then use push buttons to execute your choice. If the list box is 'untagged', you can stay 'on' the push button, while the <i>up-arrow/down arrow</i> keys move in the list box. This allows you to use <i>Enter</i> to activate the button. In most cases, double-clicking the selection bar also executes your choice.

## IPM controls

This section details the different controls in IPM.

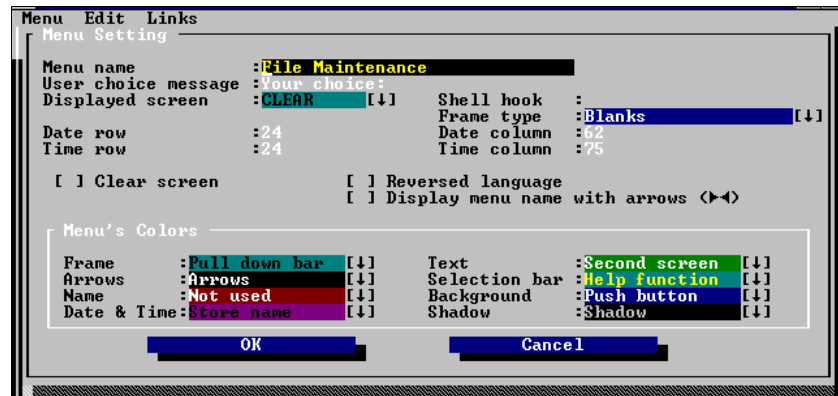
Most of the IPM controls are located in the Controls drop-down menu, which is one of the main menu options. Exceptions are POS formats, (in the POS menu), and Data Dictionary, also on a separate menu.



### Menus

IPM is capable of creating a menu 'tree', with sub-menus as 'branches', and various options for 'leaves'.

A menu in IPM is a 'branch' of the tree. It can be run from previous menus, and chain to sub-menus.



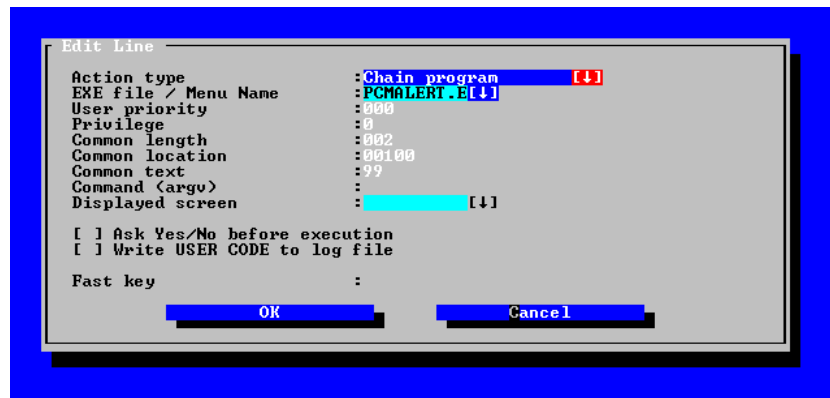
A menu consists of the following basic attributes:

Attribute	Description
Menu Name	Name to be displayed on the top of the menu.
User choice message	Prompt in front of manual entry.
Displayed Screen	Screen (from screens bank) to be displayed under the menu.
Shell Hook	Name of program to be run before entering the menu.
Frame Type	Shape of frame around menu. For example: blanks, single line, double line, and so on.
Date Row	Position (row) on the screen to display Date.
Date column	Position (column) on the screen to display Date.
Time Row	Position (row) on the screen to display Time.
Time Column	Position (column) on the screen to display Time.
Clear Screen	Check-Box: Click to clear the screen before entering menu.
Reversed Language	Check-Box: Reserved for right-to-left interface.
Display arrows	Check-Box: Wrap menu name with arrows (>menu-name<).
Colors	Set of colors for menu customization.

After defining the above attributes, each selection on the menu can be defined to run another menu, to activate an IPM form or report, or to run any other DOS executable file.

### Editing a menu

To edit any line of a menu, click twice on the line to edit. The edit-line screen displays.



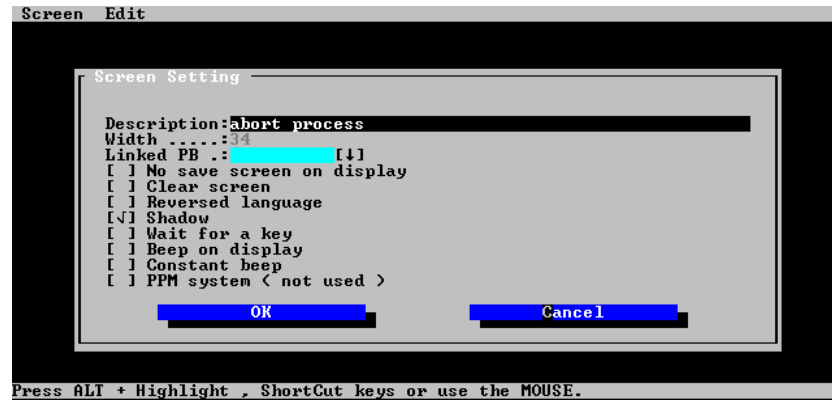
The following attributes can be edited:

Attribute	Description
Action type	Run another menu, program, or IPM form. Action types: <ul style="list-style-type: none"> <li>• Chain program: Execute the selected program.</li> <li>• Run program: Execute the selected program</li> <li>• DOS shell: Execute the selected program and return to IPM when it terminates.</li> <li>• Execute batch: Execute the selected DOS batch file.</li> <li>• Link to menu: Execute the selected IPM sub-menu.</li> <li>• Display screen: Display the selected screen.</li> <li>• Run FORM: Execute the selected form.</li> <li>• Print Query: Print the selected query (hard copy).</li> <li>• Print file: Print the selected DOS file to printer.</li> </ul>
EXE file/Menu name	Name of Menu, Form or program to run.
User priority	User level for security.
Common length	Additional information to pass through IPM common area.
Common location	Additional information to pass through IPM common area.
Common text	Additional information to pass through IPM common area.
Command (argv)	Additional information to pass through DOS command line.
Displayed Screen	Screen to display (from screens bank) when executing option.
Ask Yes/No before execution	Check-Box: Click to ask for confirmation before executing.
Write user code to log file	Check-Box: For System Security Levels - check to list the menu in Set Menu Priority screen.
Fast key	Optional short-cut key.

## Screens

A Screen in IPM is an area to be displayed, up to 80x25 characters, including colors in that area.

The screen is edited as text via an internal or external editor.



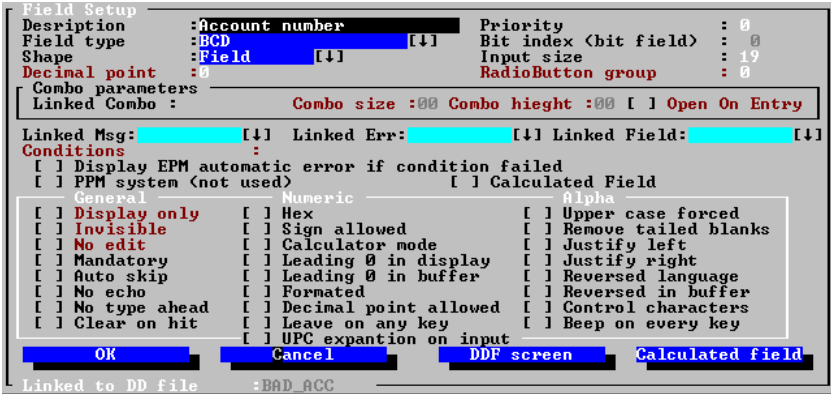
The following options are available for screens:

<b>Attribute</b>	<b>Description</b>
Description	Name for identification.
Width	Determined by IPM according to edited text (display only).
Linked PB	Relates to the POS terminal. For use with touch screens.
No save screen	Check-Box: Set to leave screen after closing.
Clear Screen	Check-Box: Set to clear screen before display.
Reversed language	Check-Box: Reserved for right-to-left interface.
Shadow	Check-Box: Set to wrap shadow around the screen.
Wait for a key	Check-Box: Set to wait for key-hit before closing screen.
Beep on display	Check-Box: Set to beep after display.
Constant beep	Check-Box: (POS software only). Set to get constant beep after display, until 'Clear' button hit
IPM system	Not used: Reserved.

## **Fields**

A field is an element for displaying and editing information in IPM via forms or reports.

Fields can be linked directly to Q-Dex files via the Data Dictionary. This enables access to information without additional code.



A Field definition contains the following:

<b>Attribute</b>	<b>Description</b>
Description	A name for identification.
Priority	Specify that a user with 'x' priority only, can access this field.
Field type	Data form type (ASCII text, byte, word, long, etc.).
Bit index	The bit number for bit fields (usually Check-Boxes).
Shape	Field's shape (Field, Check-box, Combo...). Shapes: <ul style="list-style-type: none"> <li>• Field: Normal field entry (key-in).</li> <li>• ComboBox: Linked to a Combo-Box from the ComboBox bank.</li> <li>• CheckBox: Allow only to check / uncheck a check-box.</li> <li>• RadioButton group: Allow values from the Q-Dex records in the range of the query of the linked Qcombo box, from the Qcombo box bank.</li> </ul>
Input Size	Number of characters or digits.
Decimal point	Number of places after the decimal point (assuming integer numbers, i.e. if parameter set to 2, 100 equals 1.00, and 2345 equals 23.45).
RadioButton group	Group of radio button in order to distinguish if more than one group appears in one form.
Linked Combo size/height	Linked combo-box name, width of combo on the screen, and number of lines on combo-box when opened.
Linked message/error	Message/Error to be displayed when entering the field.
Display EPM automatic error	Check-box: Display IPM error if field is out of range. Range is defined when linking to form, report or query.
IPM system	Check-box: Not used, reserved.
Calculated Field	Not in use.

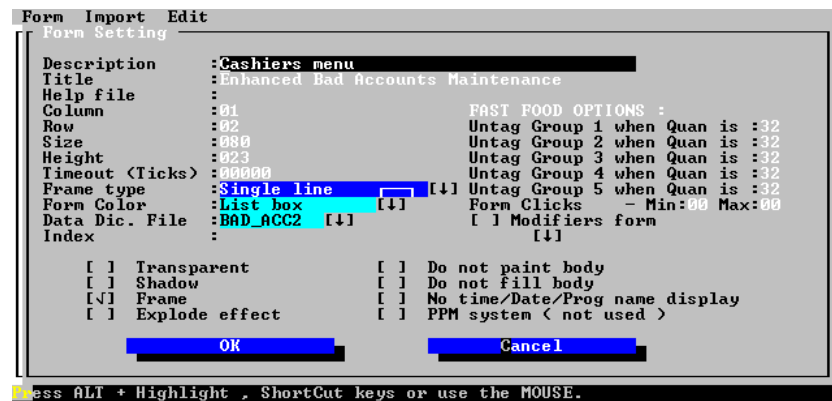
<b>Attribute</b>	<b>Description</b>
Display Only	Check-box: Set to avoid entering the field in forms.
Invisible	Check-box: Use to allow validity check, without actual display.
No edit	Check-box: Set to avoid edit in forms and tables.
Mandatory	Check-box: Set to ensure entry to field in forms.
Auto Skip	Check-box: Exit from field if all digits entered when set.
No echo	Check-box: Display asterisks (*) when typing (for passwords).
No type ahead	Check-box: Ensure clear keyboard buffer before entry.
Clear on hit	Check-box: Clear previous entry if new text typed.
Hex	Check-box: Allow only hexadecimal entry (0-9, A-F) and convert to hex.
Sign allowed	Check-box: Allow '+' and '-' in entry, and set sign to entry on exit.
Calculator mode	Check-box: Allow '+' and '-' for adding and subtracting to current entry.
Leading 0 in display	Check-box: Display zeroes (0) before figures on screen, if numerical entry.
Leading 0 in buffer	Check-box: Add leading zeroes (0) before figures on ASCII formatted buffer, if numerical entry.
Formatted	Check-box: Display field with decimal formatting (i.e. display 1234567 as 1,234,567).
Decimal point allowed	Check-box: Allow typing of decimal point in entry, if numeric.
Leave on any key	Check-box: Exit from entry upon any key-press.
UPC expansion on input	Check-box: Automatically inserts zeroes to expand UPC number.
Upper case forced	Check-box: Convert any lower case letter to upper case.
Remove tailed blanks	Check-box: Convert ASCII formatted strings to ASCII, with trailing blanks removed.

Attribute	Description
Justify left	Check-box: Justify ASCII formatted text to left of buffer.
Justify Right	Check-box: Justify ASCII formatted text to right of buffer.
Reversed language	Check-box: Reserved for right-to-left interface.
Reversed in buffer	Check-box: Reserved for right-to-left interface, for correct sorting of text-strings.
Control Characters	Check-box: Allow ASCII codes lower than 32 (using Alt+Numeric keypad) when set.
Beep on every key	Check-box: Enable key-clicker during typing.

## Push Buttons

Push Buttons are screen elements that perform as function keys on the keyboard. They can also be operated via the mouse or touch-screen interface.

Push buttons appear in interactive forms, to enable user interfaces.



Push button definitions consist of the following:

Attribute	Description
Description	Name for identification.
Button text	Text to be displayed on button. An '&' character in button set can be used to define a fast key (short cut), using the <i>Alt</i> key and the selected character.
Linked message key	Name of message from messages bank to display when button is active.
Size	Width of button in characters.
Height	Height of button in lines.
Button Color	Color of button, when not selected.
Mark Color	Color of button when pressed.
Highlight color	Color of fast-key character, when not selected.
Command Type	You can specify a form, query, program, etc. to run on activating the push button. Specify the relevant name below.
Exe/Screen/Form	Name of executable, if selected, or name of screen or form from the appropriate bank to activate, when button pressed.
Command (argv) string	Additional parameter to add when executing. (DOS command line parameters).
Exit from form...	Check Box: Exit from form when executing, returning the requested EXIT code to the application.
Skip FORM_FIELD_EXIT events...	Check Box: Skip post-field checking of the previous entry, when executing button. This Check Box can be used for EXIT-NO-SAVE button, and validation-check for the previous entry if unnecessary.
Transparent	Check Box: Do not display the button on the form. This check box is useful when allocating more than one control on the same location on the screen.
3D Style	Check Box: Display the button in 3D shape ('above' the surface when displayed, and 'sunk' when pressed).

Attribute	Description
No Shadow (POS only)	Check Box: Display the button without shadow.
IPM System	Check Box: Not used, reserved for IPM.
Beep upon press (POS only)	Check Box: System to beep when button pressed.
Text Output	2 sets of radio-button, for text-alignment inside the button area.

In addition to push-buttons, there is another group of buttons, the Fast-Food Push-Buttons. The difference is that the FF-PB group is oriented for POS operations, such as PLU-sale, displaying sales information on buttons, etc.

```

Push Button
Description      :
Button text     :
( use @ in order to cut text )
Linked message key :
Button color    :List box [↓]
Size           :15      Height :93
Command type   :Regular [↓]
PLU number     :
Form Name      :
Untag when Quantity is : 0      PB Group:None [↓]

[ ] Exit from FORM      Exit code :00000
[✓] Display quantity   [ ] Drags modifiers <no void PB>
[ ] Reset quantity     [ ] Draged modifier <no void LB>
[ ] Count clicks       [ ] Must be followed by Modifier<s>
[✓] 3D style           [ ] Draged Modifier with PRICE
[ ] Dont display text  [✓] Beep upon press

Text location : < > Left   < > Right  <+> Center
               < > Top    < > Bottom  <+> Middle
OK            Cancel      Emulate

```

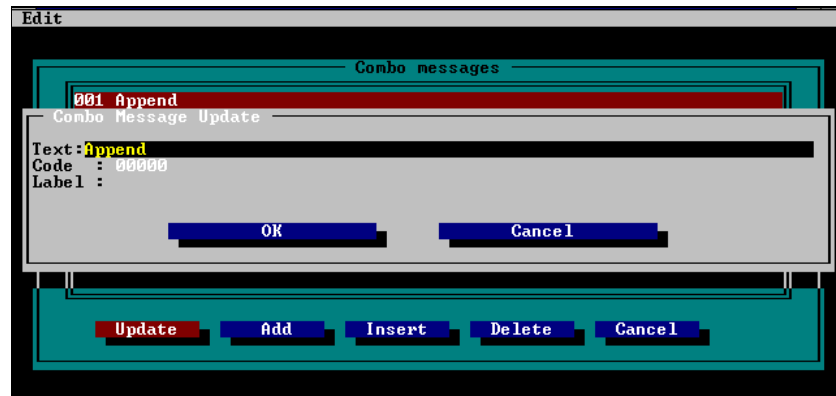
Fast Food Push button definition consists of the following:

<b>Attribute</b>	<b>Description</b>
Description	Name for identification.
Button text	Text to be displayed on button. An '&' character in button set can be used for a fast key, using the <i>Alt</i> key and the selected character.
Linked message key	Name of message from messages bank to display when button is active.
Button Color	Color of button, when not selected.
Size	Width of button in characters.
Height	Height of button in lines.
Command Type	Type of operation to execute when button pressed. Types: <ul style="list-style-type: none"> <li>• Regular PB, handled by application.</li> <li>• Not used, reserved.</li> <li>• Chain program/batch second command processor (spawn): Activate external DOS batch file or executable, and return to form when terminated.</li> <li>• Not used, reserved.</li> <li>• Display screen: Display screen from the screens bank.</li> <li>• Run FORM: Execute a form from the forms bank.</li> <li>• Print query: Print a query of a defined range.</li> <li>• Print file: Print a DOS file to a printer.</li> </ul>
PLU Number	PLU number to chain (to 'sell') when button pressed, or additional information to add when executing button.
Form Name	Form name to run.
Untag when Quantity is:	Set the number of times the button is allowed to be selected. If quantity is set to 1, for example, the user can only select the button once. Used for composite meals.

<b>Attribute</b>	<b>Description</b>
PB Group	The groups in the drop-down list relate to the groups in Fast Food Options, in the Form Setting screen. See page 24.
Exit from FORM	Check Box: Exit from form when executing, returning the requested EXIT code to the application.
Exit Code	Code to return to application when exiting the form.
Display Quantity	Check Box: Display accumulative quantity on button text.
Drags modifiers - No void PB	Check Box: Not Used.
Reset Quantity	Check Box: For multi-item selections.
Dragged Modifier - No void LB	Check Box: Not Used.
Count Clicks	Check Box: For multi-item selections, count total clicks in form when quitting the form.
Must be followed by Modifier(s)	Check Box: Not Used.
3D Style	Check Box: Displays list box in 3D format ('above' the surface when displayed, and 'sunk' when pressed).
Dragged Modifier with Price	Check Box: Displays components of a meal. Components may have a price, or no price if component of a meal.
Don't Display Text	Check Box: Don't display button text. Used for run-time push buttons where application puts in the text.
Beep upon press	Check Box: Beep when press button.
Text location	2 sets of radio-button, for text-alignment inside button area.

## Combo Box

A Combo box is a group of messages, with unique numbering (code) for each message. The Combo box can be linked to a field after it has been defined, using the appropriate settings.

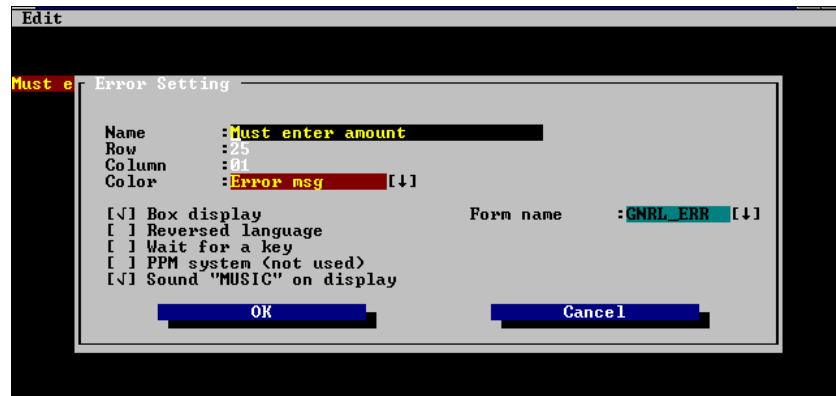


When editing a field linked to a combo box, instead of typing the field's value, a selection from the pre-defined messages is available. Use the *up-arrow* and *down-arrow* keys in the field, or from a list of all of the messages, use the *right-arrow* key (→) or mouse to open it. The *left-arrow* key (←), *F8* key, or mouse selection closes the combo box.

## Errors

Errors in IPM are messages, displayed with audible warnings. Errors can be invoked by various events:

- If an application sets an error (when any illegal operation or function has been determined).
- Upon condition failure in field entry, if set to display an error.
- Upon any IPM system or operation error.

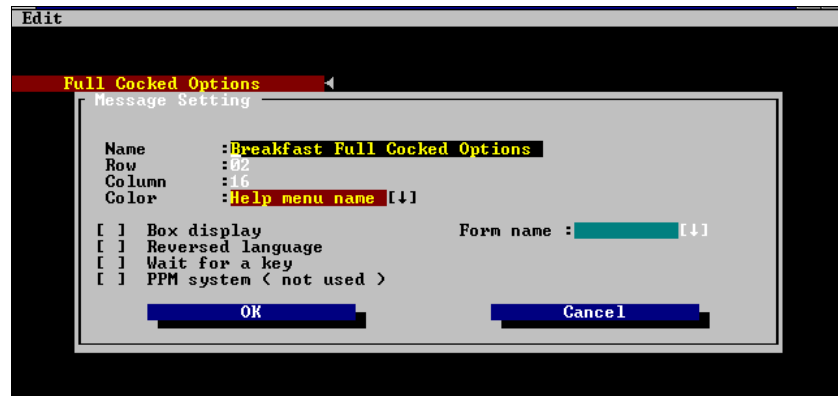


An Error definition consists of the following:

Attribute	Description
Name	Name of error for identification.
Row/Column	Position to display on the screen.
Color	Color of error. (Message and background).
Box display	Check Box: Error is displayed in a form, if set. The form is selected from the Forms bank.
Reversed Language	Check-Box: Reserved for right-to-left interface.
Wait for key	Check-Box: IPM waits for a key/mouse event before clearing error. Not active when box-display is set.
IPM system	Check Box: Not used, reserved for IPM.
Sound 'Music'	Check Box: Instead of a beep sound, make a more distinctive noise. Useful mainly in POS systems, where the environmental noise is loud.
Error text/length	The actual text to be displayed when called. If box-display is set, IPM wraps the error into lines, depending on the selected form's width.

## Messages

A Message in IPM is useful for displaying any information for the user, whether in a certain line on the display, or in a special form.



A Message definition consists of the following:

Attribute	Description
Name	Name of message for identification.
Row/Column	Position to display on the screen.
Color	Color of message. (Message and background).
Box display	Check Box: Message is displayed in a form, if set. The form is selected from the Forms bank.
Reversed Language	Check-Box: Reserved for right-to-left interface.
Wait for key	Check-Box: IPM waits for a key/mouse event before clearing message. Not active when box-display is set.
IPM system	Check Box: Not used, reserved for IPM.
Message text/length	The actual text to be displayed when called. If box-display is set, IPM will wrap the message into lines, depending on the selected form's width.

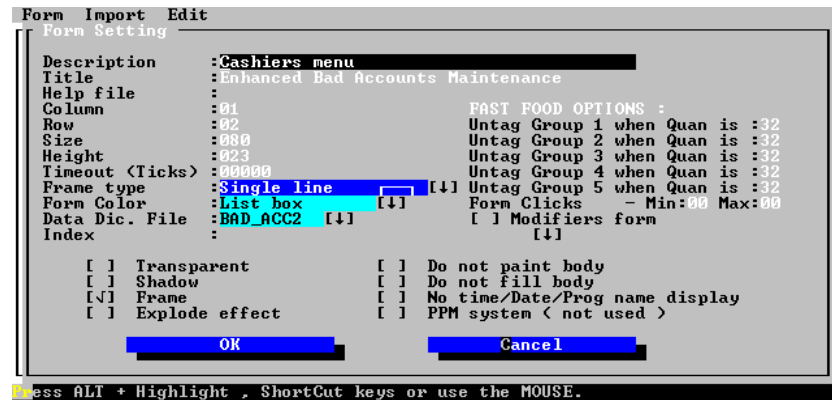
## Forms

An IPM Form is a collection of IPM screen elements, gathered into one label, with some additional information, mainly for the characteristics of this collection.

The form is also the interface for interaction between the user and an application via IPM. In order to do so, IPM has a 'gate' to the

application, in the application's code-segment. Upon every event in the form, IPM calls the application's gate, with the event's operation-code.

A form can be transparent. That is, it is possible to see only the form's elements, but it could also be visible, having a frame and background behind the selected elements.



### Note

While in the form Setting screen, press Ctrl+F to display the form's name at the lower left corner.

A form's definition consists of the following:

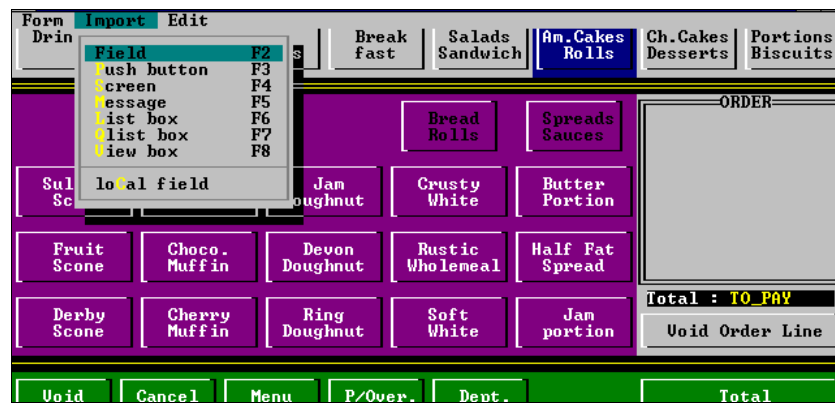
<b>Attribute</b>	<b>Description</b>
Description	Name of form for identification.
Title	Headline on top of form, if not transparent.
Help File	If a help file exists, specify the name here. Users must press <b>CTRL+H</b> to run the specified help file.
Column/Row/Size / Height	Position and dimensions of the form.
Time-out (ticks)	Time interval in 1/18 sec. to activate the time-out routine in the application's gate.
Frame Type	The frame's characteristics (blank, single-line, double-line etc.) if not transparent.
Form Color	Color of form, if not transparent.
Data Dictionary File	Name of file from DD bank, if any, that the form associates with, including index name for relative-expanded files, for field-data-table.
Index	Form works with the selected index file linked to the data dictionary file above.
Transparent	Check-Box: Form is transparent, if set.
Shadow	Check-Box: Display shadow under bottom-right of form's dimensions, if not transparent.
Frame	Check-Box: Display frame around form, if not transparent.
Explode effect	Check-Box: Show 'exploding' window before entry and after exit from form.
Do not paint body	Check-Box: If set, use the color settings on the form's frame only. Invalid when transparent.
Do not fill body	Check-Box: If set, do not blank inside the framed area. Invalid when transparent.
No time/date/prog name display	Check-Box: If set, ignore the date/time/program-name... setting.
IPM system	Check Box: Not used, reserved for IPM.

Fast Food options:

Attribute	Description
Untag Group (1 to 5) when Quantity is	Define the total number of times buttons may be selected, for each group. Once the user reaches the specified number, all buttons in this group become unavailable.
Form Clicks: Minimum and Maximum	Define the minimum and maximum number of clicks allowed in the form.
Modifiers Form	

As shown below, a form can include the following controls:

- Fields
- Push buttons (including fast-food PB)
- Screens
- Messages
- List Boxes
- Qlist Box (Q-Dex List Box)
- View Box
- Local Field



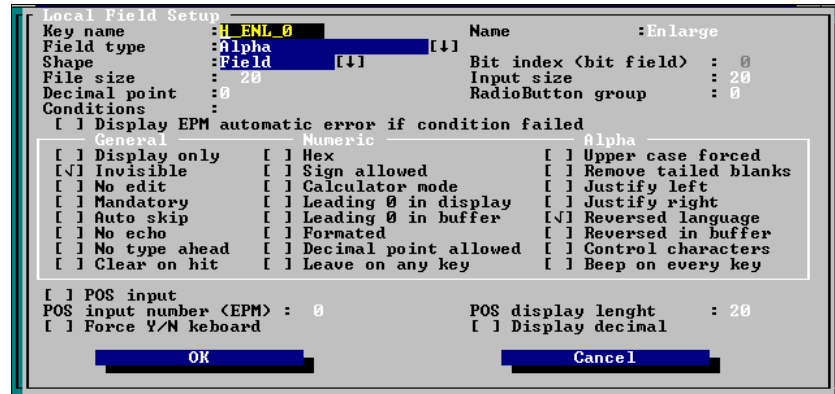
Controls such as queries, combo boxes, Qcombo boxes and errors, cannot be linked directly to a form. The linking of these controls to a form is always via another control (list/Qlist box for queries, field for combo/Qcombo box etc.).



Although the imported controls have been pre-defined, it is possible to change some of these settings. In the above example: a push-buttons local settings in a form that includes colors, size, and a few flag settings.

With these parameters, the controls can be customized to different forms, without duplicating the control.

In order to distinguish these parameters during the initial setting, they are displayed in a purple background.



### Note

The screen above only displays if the parameter 'IPM User Interface' in IPM Parameters is set to 'No POS'.

Local fields in forms, is a method of using entry fields for display or input, that do not exist in the fields bank. This option is useful mainly for POS applications.

Local fields have most of the settings as for regular fields, except link to Combo boxes and some other field types.

On the other hand, local fields have some additional information, unique only for local fields.

A local field consists of the following information:

<b>Attribute</b>	<b>Description</b>
Key Name	For identification in the form's controls.
Name	For identification.
Field type	Data form type (ASCII text, byte word, long etc.).
Shape	Field's shape (Field, Check-box...).
	Shapes:
	<ul style="list-style-type: none"> <li>• Field: Normal field entry (key-in).</li> <li>• Combo Box: Linked to a Combo Box from the Comboboxes bank.</li> <li>• Check Box: Allow only to check / uncheck a Check Box.</li> <li>• Radio Button: Allow only to check one of the files of the same radio group.</li> <li>• Qcombo Box: Allow values from the Q-Dex records in the range of the query of the linked Qcombo Box, from the Qcombo Boxes bank.</li> </ul>
Bit index	The bit number for bit fields (usually Check-Boxes).
File Size	The size in bytes in the data table.
Input Size	The size of the input/display on the screen, in characters.
Decimal point	Number of places after the decimal point (assuming integer numbers, i.e. if parameter is set to 2, 100 equals 1.00, and 2345 equals 23.45).
RadioButton group	Group of radio button to distinguish if more than one group appears in one form.
Conditions	Set of conditions for validity check during input.
Display automatic error	Check-box: Display IPM error if field is out of range.  Range is defined when linking to form, report or query.
Display Only	Check-box: Set to avoid entering the field in forms.

<b>Attribute</b>	<b>Description</b>
Invisible	Check-box: Use to allow validity check, without actual display.
No edit	Check-box: Set to avoid edit in forms and tables.
Mandatory	Check-box: Set to ensure entry to field in forms.
Auto Skip	Check-box: Exit from field if all digits entered when set.
No echo	Check-box: Display asterisks (*) when typing (for passwords).
No type ahead	Check-box: Ensure clear keyboard buffer before entry.
Clear on hit	Check-box: Clear previous entry if new text typed.
Hex	Check-box: Allow only hexadecimal entry (0-9, A-F), and convert to hex.
Sign allowed	Check-box: Allow '+' and '-' in entry, and set sign to entry on exit.
Calculator mode	Check-box: Allow '+' and '-' for adding and subtracting to current entry.
Leading 0 in display	Check-box: Display zeroes (0) before figures on screen, if numerical entry.
Leading 0 in buffer	Check-box: Add leading zeroes (0) before figures on ASCII formatted buffer, if numerical entry.
Formatted	Check-box: Display field with decimal formatting (i.e. display 1234567 as 1,234,567).
Decimal point allowed	Check-box: Allow typing of decimal point in entry, if numeric.
Leave on any key	Check-box: Exit from entry upon any key-press.
Upper case forced	Check-box: Convert any lower case letter to upper case.
Remove tailed blanks	Check-box: Convert ASCII formatted strings to ASCII, with trailing blanks removed.
Justify left	Check-box: Justify ASCII formatted text to left of buffer.

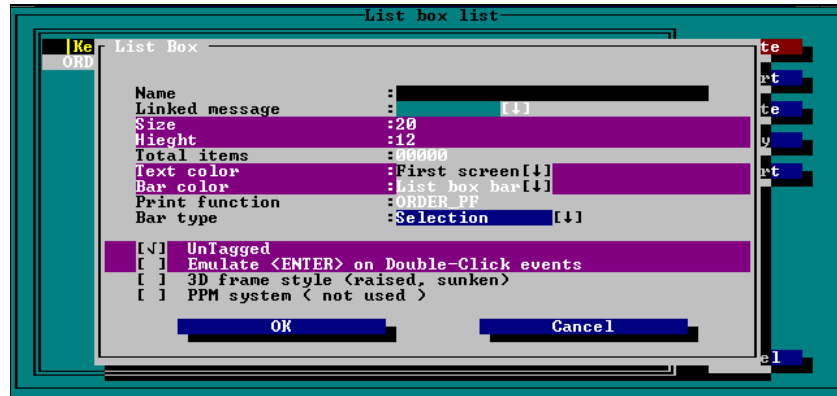
<b>Attribute</b>	<b>Description</b>
Justify Right	Check-box: Justify ASCII formatted text to right of buffer.
Reversed language	Check-box: Reserved for right-to-left interface.
Reversed in buffer	Check-box: Reserved for right-to-left interface, for correct sorting of text-strings.
Control Characters	Check-box: Allow ASCII codes lower than 32 (using <i>Alt</i> +Numeric keypad) when set.
Beep on every key	Check-box: Enable key-clicker during typing.
POS input	Check-box: If set, the field below becomes active.
POS input number...	Identification number for POS application
POS display length	Display length on POS display. Allows scrolling input, if the Input-size is greater than POS display length (eg. 13 digit PLU number entry, in 8 characters space).
Force Y/N keyboard	Check-box: Forces entry of <i>Yes</i> or <i>No</i> (only on POS keyboard), to exit from field.
Display decimal	Check-box: Allows flexible decimal position from POS, when the decimal point is set to 0. Example: If a store works both with KG and LB for weight, a parameter can allow change between 2 and 3 decimal places.

## List Box

A List Box is a framed selection-box, with one line per selection item. The text inside the list box lines is controlled by the application.

In order to link between the list box's contents and the application, IPM sends the line-identifier to the application, and gets back the information, or the application can display the line itself. The link is done through a *Print Function* which is a label that both IPM and the application use in a table of print function labels, against addresses in the application.

In order to format the list box in a flexible way, fields are located on the screen as 'dummy fields', which the application refers to, only find the horizontal location of the fields in the list box line.

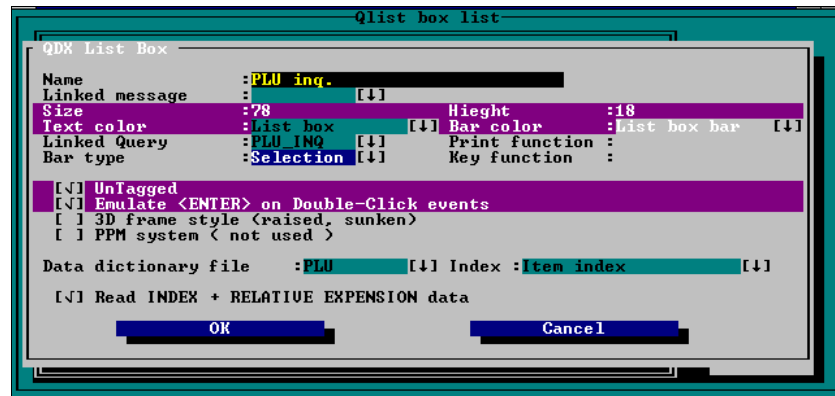


A List Box definition consists of the following:

<b>Attribute</b>	<b>Description</b>
Name	Name of List Box for identification.
Linked Message	Message to display when list box is active.
Size/Height	List Box's dimensions.
Total Items	Limit for maximum number of lines, if set.
Text Color	Color of un-marked text.
Bar Color	Color of current marked text.
Print function	Print function to link to application.
Bar Type	Shape of the selection bar. Bar types: <ul style="list-style-type: none"> <li>• Selection</li> <li>• Left arrow</li> <li>• Right arrow</li> <li>• Double arrow</li> </ul>
untagged	Check-Box: Enables use of push-buttons and double-click features while <i>up-arrow</i> and <i>down-arrow</i> keys are available.
Emulate Enter...	Check-Box: Enables double-click feature.
3D frame...	Check-Box: Displays list box in 3D format ('above' the surface when displayed, and 'sunk' when pressed).
IPM system	Check Box: Not used, reserved for IPM.

### **Qlist Box**

A Qlist-Box (Q-Dex list box), is the same as a normal list box, but linked to a specified Q-Dex file.



This way, the information for the display is supplied from a Q-Dex file, rather than the application itself. In addition, instead of using a print function, a query can be linked to a Qlist box, thus formatting the display without additional code. The advantage is that information from Q-Dex files can be accessed without any code at all.

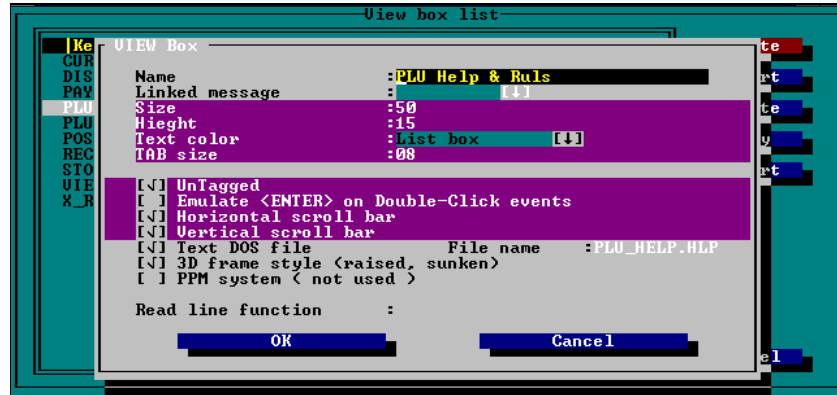
If order to allow further flexibility, a key function can be linked (to the application), thus enabling complex calculations on the key-validity from the accessed index.

The total items for a Qlist box is determined by the Q-Dex file setup.

<b>Attribute</b>	<b>Description</b>
Name	Name of List Box for identification.
Linked Message	Message to display when list box is active.
Size/Height	List Box's dimensions.
Text Color	Color of un-marked text.
Bar Color	Color of current marked text.
Linked Query	Name of query from queries bank to link. Inactive when print function is set.
Print function	Print function to link to application. Inactive when query is linked.
Bar Type	Shape of the selection bar. Bar types: <ul style="list-style-type: none"> <li>• Selection</li> <li>• Left arrow</li> <li>• Right arrow</li> <li>• Double arrow</li> </ul>
Key function	Label to link key function in application.
Untagged	Check-Box: Enables use of push-buttons and double-click features while <i>up-arrow</i> and <i>down-arrow</i> keys are available.
Emulate Enter...	Check-Box: Enables double-click feature.
3D frame...	Check-Box: Displays list box in 3D format ('above' the surface when displayed, and 'sunk' when pressed).
IPM system	Check Box: Not used, reserved for IPM.
Data dictionary file/Index	File from data dictionary files bank to link. If the file type is relative expansion, an index must be linked.
Read INDEX+REL...	Check-Box: In relative expansion files, read full record (index+relative) if set.

## View Box

A View box is an area on the screen used to view a file, in a 'raw' ASCII format. This control can be used to display help files, on-screen reports, or any other text file.



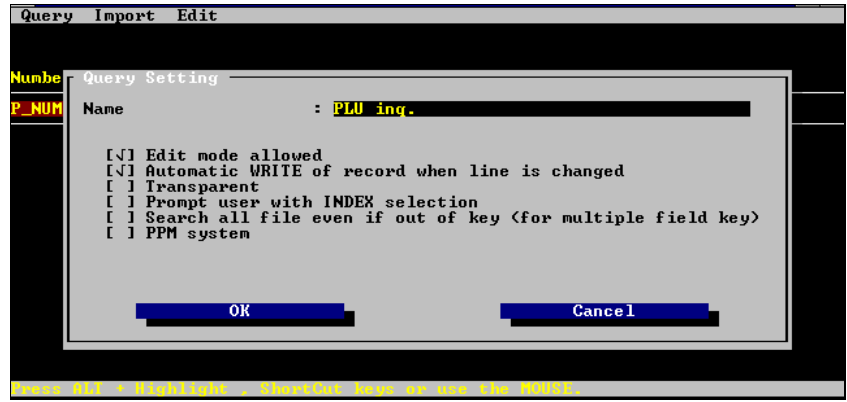
A view box definition consists of the following:

<b>Attribute</b>	<b>Description</b>
Name	Name of view box for identification.
Linked message	Linked message from messages bank to link when active.
Size/Height	View box's dimensions.
Text color	Color of the displayed text in the view box.
TAB size	Number of blanks to next tab stop. Default = 8.
Untagged	Check-Box: Allow <i>up-arrow/down-arrow</i> keys also when the control is <i>not</i> on the view box. This enables use of push buttons and double click features.
Emulate ENTER...	Check-Box: Allow double click feature.
Horizontal scroll bar	Check-Box: Horizontal scroll bar displays if set.
Vertical scroll bar	Check-Box: Vertical scroll bar displays if set.
Text DOS file + File name	Check-Box: Use the specified file for display. If not, a read-function must be supplied.
3D frame...	Check-Box: Displays list box in 3D format ('above' the surface when displayed, and 'sunk' when pressed).
IPM system	Check Box: Not used, reserved for IPM.

## Query

A Query is a set of fields, organized on one horizontal line. A query cannot work as an independent control, but must be linked via another control such as a query or a combo box. The query will then be used to format the displayed text within the host control, thus avoiding the use of print functions and additional code.

For Qlist boxes, a query can also be used for updates in a table mode (i.e. quick price updates).



A query's definition consists of the following:



<b>Attribute</b>	<b>Description</b>
Conditions	Set of conditions for field validity check. This is useful for selection of a subset.
Decimal position	Number of places after decimal position.
Display size	Number of characters on the screen.
Reverse Input	Check-Box: Reserved for right-to-left interface.
Main Field	Check-Box: Use this field for fast-selection by typing, if set.
Hide	Check-Box: If set, use this field only for checking a condition, but not for display.
No Edit	Check-Box: If set, avoid edit on this field, when edit is allowed on query.
Lock field...	Check-Box: If set, ensure that this field is always on screen during horizontal scroll.
Add '   ' after...	Check-Box: Add separator after field, if set.
Prompt with range...	Check-Box: Show input screen for user-defined selection argument for all fields that have this check-box set, before entering the host-element (Qlist-box, Qcombo...).

## **Qcombo Box**

Qcombo, similar to a Combo box, enables a selection box from a pre-defined list of items, as an entry to a field. But in Qcombo, rather than pre-defining the list of items in IPM, this list can be linked directly to a Q-Dex file.

In order to allow formatting of the contents of Q-combo text on screen, it is possible to link the Qcombo to a query, or to a print function in the application. It is also possible to link to a key-function in the application, in order to allow validity-check on the Q-Dex index for the Qcombo. These rules are very similar to Qlist-box rules.

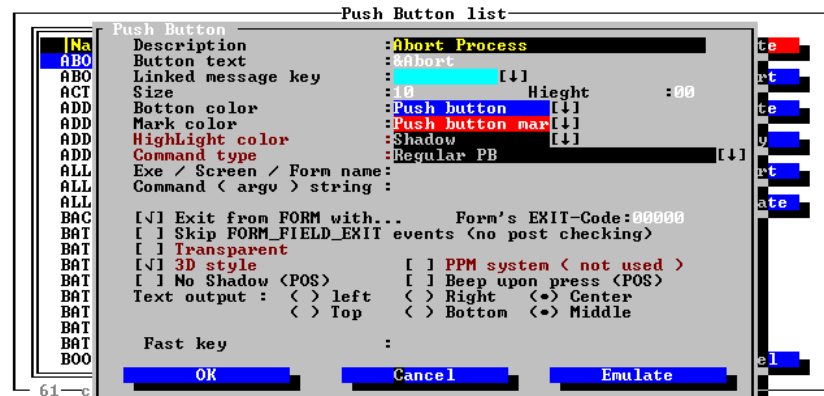


A Qcombo definition consists of the following:

Attribute	Description
Name	Name of List Box for identification.
Size/Height	List Box's dimensions.
Text Color	Color of un-marked text.
Bar Color	Color of current marked text.
Linked Query	Name of query from queries bank to link. Inactive when print function is set.
Print function	Print function to link to application. Inactive when query is linked.
Bar Type	Shape of the selection bar. Bar types: <ul style="list-style-type: none"> <li>• Selection</li> <li>• Left arrow</li> <li>• Right arrow</li> <li>• Double arrow</li> </ul>
Key function	Label to link key function in application.
Data dictionary file/Index	File from data dictionary files bank to link. If the file type is relative expansion, an index must be linked.
Multi User	Check-Box: Set Q-Dex multi-user flag when set.
Relative Read	Check-Box: In relative expansion files, read full record (index+relative) if set.

## Reports

IPM can generate reports, using a pre-defined format, with additional C code for selecting records, and also for formatting complex reports.



All IPM reports consist of the following parts:

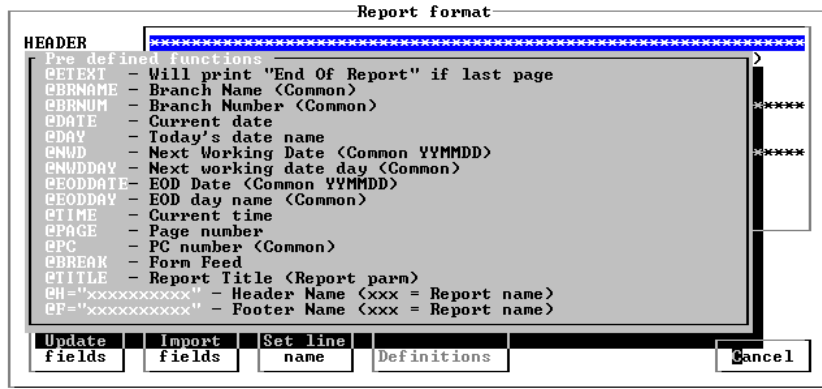
- **Header** (page header): Allows different formatting for page-headers for reports, several headers can be defined for each report, allowing the application to select different headers.
- **Section:** Data-line formatting. Allows more than one line-format, which the application can select from during printing.
- **Footer** (page footer): In the same manner as page headers, allows page-footer formatting.

For each of a report's parts, fields from the fields bank can be imported. The application is then called by IPM through a gate to fill these fields with the appropriate information.

A report definition consists of the following:

<b>Attribute</b>	<b>Description</b>
Description	Name for identification.
Title	Not in use.
Linked Screen	Screen from screens bank to display while printing.
File Name	Name of DOS file, if report destination is a file.
Printer number	Device number if destination is a printer, 0=LPT1.
DD File	Name of file from DD files bank, if any, including index file for relative-expansion files.
Index	Name of index file, if any, including index file for relative-expansion files.
Condensed Print	Check-Box: Printout is condensed (if set to 132 characters on an 80 column printer).
Append file	Check-Box: If set, and the destination is a file, the file is appended rather than emptied each time the report is called.
No header/footer	Check-Box: If set, headers and footers will not be automatically printed for the report. The application is responsible, in this case, for headers/footers.
No control characters (Only LF and FF)	Allow only control characters for line feed and form feed.
No Control Characters at all	Don't allow any control characters. Creates a raw ASCII file.

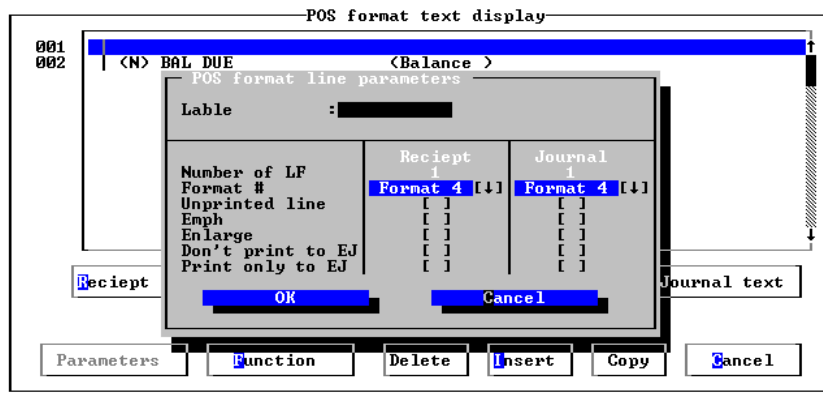
The report formatting screen is initially divided into the report's parts: header, section and footer.



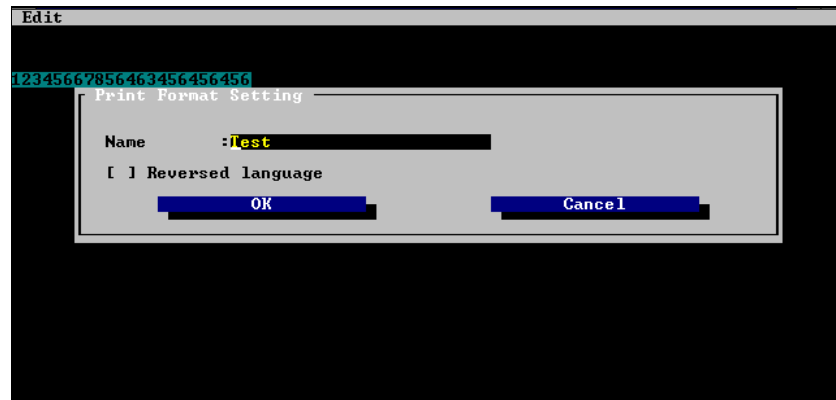
Every line in the report has a unique identifier, displayed on the left of the line. This identifier is initially set by IPM to a sequential numeric value when inserting the line, but can be changed by the user manually afterwards.

The lines can be edited as text. Fields can be imported, allocated, and formatted to fit the report's characteristics (i.e. field lengths).

The *Definitions* button displays a list of the available predefined functions. Use these functions to insert, for example, date and time.



## Print Formats



Print formats is a tool that enables report-formatting in BASIC language. Actually, this IPM element is one line, identified by an IPM key, that can be read into a BASIC program through an IPM library, and used later as a format string for PRINT USING command.

There is no use for this element through C language libraries, and there is no IPM control behind the definitions of these text lines.

## POS Formats

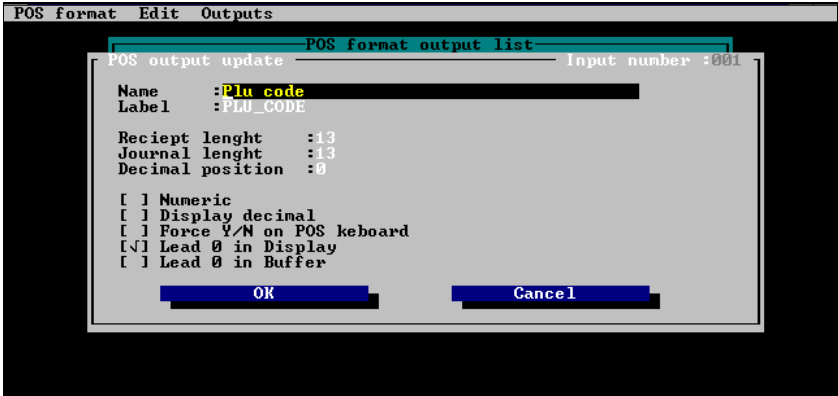
POS format is a tool for formatting POS receipt and audit (journal) printing. A POS format is built of text to print, and fields to be filled by a POS application. All POS format fields are *local fields*.

POS formats are only used by POS applications. They are not supported by a standard IPM library, but by special modules for POS.



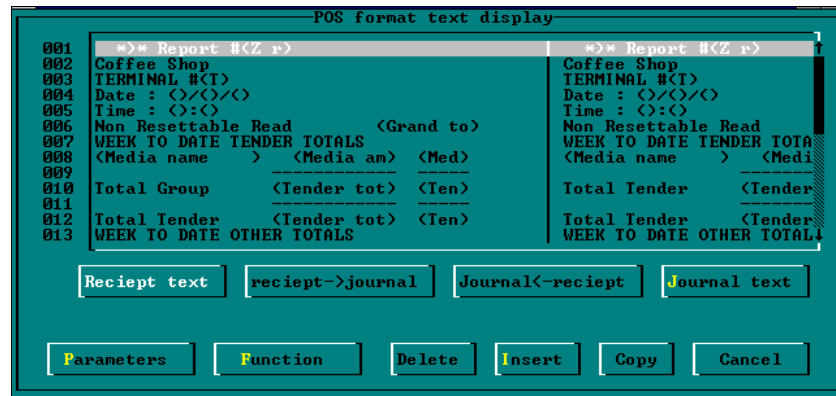
The fields of a POS format can be identified by a unique label, or by a unique number for each field.

The fields can be imported later from the local bank into the formats.



Each local field consists of the following:

Attribute	Description
Name	For identification. This field can be used also for specifying printed text, when requested by application.
Label	For identification by application/IPM. A numeric identifier is also specified when the field is initially inserted.
Receipt/Journal length	Number of characters to print when printing on receipt or journal.
Decimal position	Number of places after decimal position.
Numeric	Check-Box: Specify that field is numeric, if set.
Display decimal	Check-Box: Allow application to define a variable decimal position, if necessary (example for weight fields: allow by internal-POS-parameter for 2 dec. places for Oz, and 3 decimal places for Kg).
Force Y/N....	Check-Box:
Lead 0 in display	Check-Box: Pad numeric fields with leading zeroes.
Lead 0 in buffer	Check-Box:



During editing of POS formats, two modes are available:

- **Text edit mode:** Using Receipt-Text or Journal-Text buttons. This mode also allows you to import fields from the local fields bank.
- **Setup edit mode:** The format of each line can be edited, allowing you to change groupings of lines, line's print-modes, insertion of control-code lines (functions), copy from receipt side to journal side and vice versa, and copy from line to line.

Each line on the format can have a label, thus creating groupings of lines. This label is not unique, and a group of sequential lines with the same label is called a sub-format, or sub-report.

#### Note

If the same label appears in the same format, in non-sequential lines, only the group of lines in the first appearance of this label will be recognized by the POS application.

Usually, an unprinted line with the label 'NU' will be inserted between groups in order to create a visual separator.

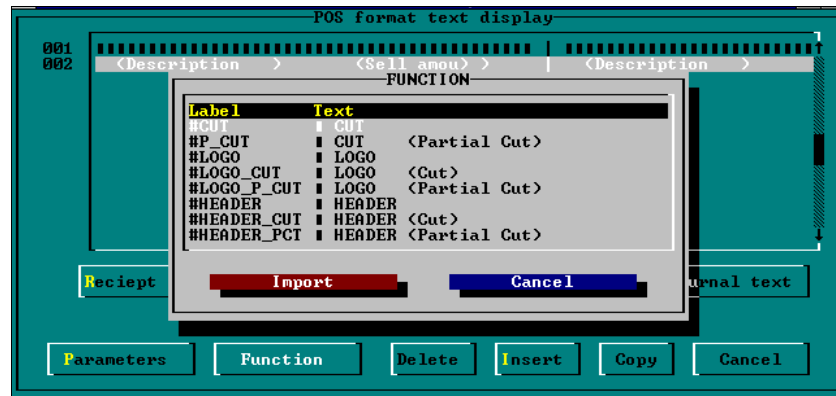
```
Epm parameters
User PPM QDX file number:01      PPM text QDX file number:60
User LOCAL file number :01      <For local inq. user priority etc.>
Yes / No letters :YyyNnn      Decimal position :2
Screen for MENU display :      [↑]
EPM (basic) qdx file number :00  PPM user interface :No POS [↑]
Screen saver screen name:      [↑]
[ ] Use screen saver password   Screen saver password :*****
Activate after minutes :60

[✓] Use DRUFILE database calls   Date format :American [↑]
[ ] Enable system FASTKEYS      Date display seperator : / [↑]
[ ] Language (Reversed)        Time display seperator : : [↑]
[ ] Enable system WARNINGS messages [ ] Use PPM LOG file
[ ] Check for LAN CRASH
Print Server Directory (RS6000) :      ( 0 )
Number of seconds to wait before Checking LAN CRASH:000
Multiplication after LAN CRASH detection :000
Maximum lines to print :062      Default LPT :LPT1 [↑]
COMMON file name:COMMON.PPM      COMMON file size:01024

[OK] [Cancel] [Next Page]
```

The definition of each format line consists of the following:

<b>Attribute</b>	<b>Description</b>
Label	For identification by application and for grouping. This field is usually set to 'NU' when used as a separator between groups.
Number of LF (Receipt and Journal)	Number of lines to feed after print.
Format # (Receipt and Journal)	Number of character set (1-4) of printer.
Unprinted line (Receipt and Journal)	Check-Box: The line is not printed when set. This field is usually set when lines are printed on receipt but not on journal. It is also used when separator is needed between groups. Another common use, is when some information is not to be printed, although treated by application. Example: If number of items is <i>available</i> at the bottom of the receipt, but not printed.
Emph	Check-Box: Emphasized line. The line is printed twice (double-strike) when set.
Enlarge	Check-Box: The line is printed in bold letters when set.
Don't print to EJ	Print data/information only, don't send to Electronic Journal.
Print only to EJ	Send to Electronic Journal only, don't print on ticket.



As described above, pre-defined functions can be imported to the formats.

These functions are recognized by the application according to the line's label.

A function can appear within a sub-format (a group), and it will not affect the printing: function's labels are ignored in terms of grouping.

There are two types of functions:

- **'Atomic'** functions: These functions are hard-coded in the POS formats IPM modules. They are basic actions such as cut-paper, print receipt header or footer, etc. All of these functions start with the sign '#'.
- **'Combined'** functions: These functions are actually print formats in the IPM file that can be called within another print format. All of these functions start with the sign '@'. This way, a print-format called 'TICK\_FRAME' is created, which contains several 'Atomic' functions: print footer, cut paper, print logo, print header.

All of the functions are located in one print-format called 'FUNCTION'. The function list can be then amended from IPM. For the 'Atomic' functions, a code has to be written in the IPM module. For the 'Combined' functions, they can be used as soon as they are added to the print format 'FUNCTION'.

## Parameters

### IPM Parameters

IPM has a set of parameters for defining the characteristics of the system and the Q-Dex file definitions.

```

Epm parameters
User PPM QDX file number: 21      PPM text QDX file number: 60
User LOCAL file number : 71      <For local inq. user priority etc.>
Yes / No letters       : VyyNnn  Decimal position      : 2
Screen for MENU display :          [↑↓]
EPM (basic) qdx file number : 00  PPM user interface      : No_POS [↑↓]
Screen saver screen name :          [↑↓]
[ ] Use screen saver password    Screen saver password    : *~*~*~*~*~*
Activate after minutes : 60

[ ] Use DRUFILE database calls    Date format                : American [↑↓]
[ ] Enable system FASTKEYS        Date display seperator    : / [↑↓]
[ ] Language <Reversed>          Time display seperator    : : [↑↓]
[ ] Enable system WARNINGS messages [ ] Use PPM LOG file
[ ] Check for LAN CRASH
Print Server Directory <RS6000> :          < 0 >
Number of seconds to wait before Checking LAN CRASH: 000
Multiplication after LAN CRASH detection : 000
Maximum lines to print : 062      Default LPT : LPT1 [↑↓]
COMMON file name: COMMON.PRM      COMMON file size: 01024

OK          Cancel          Next Page

```

The parameters are divided into 2 pages.

The first page contains the following parameters:

<b>Attribute</b>	<b>Description</b>
User PPM QDX file number	Number of QDX file for user IPM file. That is, the file the application uses.
PPM text QDX file number	The QDX file that IPM uses. As described in the beginning of this document, IPM itself is an IPM application that uses IPM libraries.
User LOCAL file number	For local inquiries - user priority, etc.
Yes/No letters.	Valid letters for yes/no inquiries. 3 first places are for Yes, 3 last for No.
Decimal position	General parameter for number of places after decimal position. Usually set to 2 (pounds & pence).
Screen for MENU display	Screen from screens bank to display 'under' IPM menus as 'wall paper'.
EPM (basic) QDX file number	Number of QDX file for old EPM system. Used for conversions from old to new MMI.
Screen saver screen name	Select the appropriate screen to display.
IPM user interface	Available options: <ul style="list-style-type: none"> <li>• No POS</li> <li>• New POS</li> <li>• Old POS</li> </ul>
Screen saver screen name	Name of screen from screens bank to be used as screen saver. Next parameters enable password and password entry, and number of idle minutes before screen saver activates.
Use DRVFILE database calls	Check-Box: When set, instead of QDX calls to QDX files, IPM will redirect QDX calls through DRVFILE redirection table.
Date format	Date format in system: European (DD/MM/YY) or American (MM/DD/YY).
Enable system FASTKEYS	Check-Box: Check to enable fast keys defined in push buttons.
Date display separator	Separator between date fields on display.
Language reversed	Check-Box: Reserved for right-to-left interface.

Attribute	Description
Time display separator	Separator between time fields on display.
Enable system WARNING messages	Not used.
Use PPM log file	Check-Box: check to display remarks dialog.
Check for LAN crash	Check-Box: Display warning message if LAN crash.
Print Server Directory (RS6000)	Specify the appropriate directory.
Number of seconds to wait before checking LAN Crash	If LAN check is set, this is the number of seconds between LAN status inquiries.
Multiplication after LAN CRASH detection	If LAN check is set, and LAN crash was detected, this is the number of bad-status answers to ignore before displaying the next LAN error message.
Maximum lines to print	Form (page) length in lines for reports.
COMMON file name	(Cannot be edited) The name of IPM's internal common file.
COMMON file size	(Cannot be edited) The size of IPM's internal common file.

```

EPM parameters
<10> Date row display      :25          Date Column display   : 2
< 8> Time row display     :25          Time column display  :15
Date & Time color       :First screen  [↑]
[ ] Display time with SECONDS
Help file directory      :
Help file common location :0000
COMMON Location:        User Password: 501  User Number : 502
[ ] Do not show "UnAutorised" lines
[ ] Use DOS file for PPM system messages
[ ] Dont display TIME & DATE in eumenu
[ ] Do not ask EXIT TO DOS when existing eumenu
[ ] Read Priority & Privilage from USER LOCAL file
[ ] Write MENU entrys to LOG file
LOG qdx file number : 90
<FIFO file rec len = 128>
OK          Cancel          Prev Page   Next Page

```

The second page contains the following parameters:

<b>Attribute</b>	<b>Description</b>
Date/Time row: column and color	Location on the screen and color for display of date & time.
Display time with SECONDS.	Check-Box: Time is displayed with seconds, when set.
Privilege (1 - 5)	Enter the appropriate password for each priority as required. Authorized users will have to enter these passwords.
Help file directory/location	Default directory to search for help files, and location (offset) in IPM common files.
Do not show 'unauthorized' lines	Check-Box: If set, menu lines with priority setting higher than the current user will not be shown on menu.  If not set, these lines will be displayed, but not available for the user.
Use DOS file for IPM system messages	Check-Box: If set, IPM's default IPM file will be a DOS file (rather than QDX).
Don't display TIME & DATE in eurmenu	Check-Box: If set, date & time are not displayed in menu system.
Do not ask EXIT TO DOS when exiting eurmenu	Check-Box: If set, when escaping from main menu, do not ask for user confirmation.
Read Priority and Privilege from LOCAL USER file	Check-Box: Read menu privilege and priority from local user file.
Write MENU entries to LOG file	Check-Box: Check to enable User access logging.
LOG QDX file number	Usually 90.  Note that FIFO file record length = 128.

EPM parameters - Common location of data							
Program name:	Location	in	IO	common	0401	Length	080 Row 01 Column 01
Shop Name :	Location	in	IO	common	0581	Length	031 Row 25 Column 50
CopyRight :	Location	in	IO	common	0000	Length	000 Row 00 Column 00
Report Data							
Branch Name	Location	in	IO	common	0620	Length	040
Branch Number	Location	in	IO	common	0660	Length	010
EOD Date	Location	in	IO	common	0680	Length	006
Next W Day	Location	in	IO	common	0670	Length	006
PC number	Location	in	IO	common	0686	Length	004

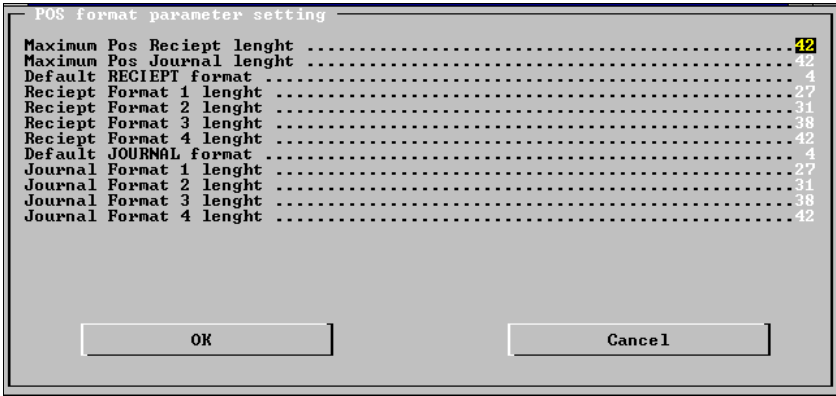
OK Cancel Prev Page Next Page

The third page contains the following parameters:

Attribute	Description
Program Name (On screen)	Location, Length, Row and Column in IO Common.
Shop Name (On screen)	Location, Length, Row and Column in IO Common.
Copyright (On screen)	Location, Length, Row and Column in IO Common.
Report Data: (In Files)	
Branch Name	Location, Length, Row and Column in IO Common.
Branch Number	Location, Length, Row and Column in IO Common.
EOD Date	Location, Length, Row and Column in IO Common.
Next W Day	Location, Length, Row and Column in IO Common.
PC Number	Location, Length, Row and Column in IO Common.

### POS Parameters

POS format parameters are settings for POS-printer characteristics. IPM allows up to 4 different character sets (EPSON standard), both for receipt and journal printing.



The settings (both for receipt and journal) are:

Settings	Description
Maximum print length	The widest print-line, in characters.
Default format	From the four settings below, which is the default when inserting new POS formats/lines.
Format 1/2/3/4 length	The print-line width for the four formats.

## POS Format Parameters

This function is not documented yet.

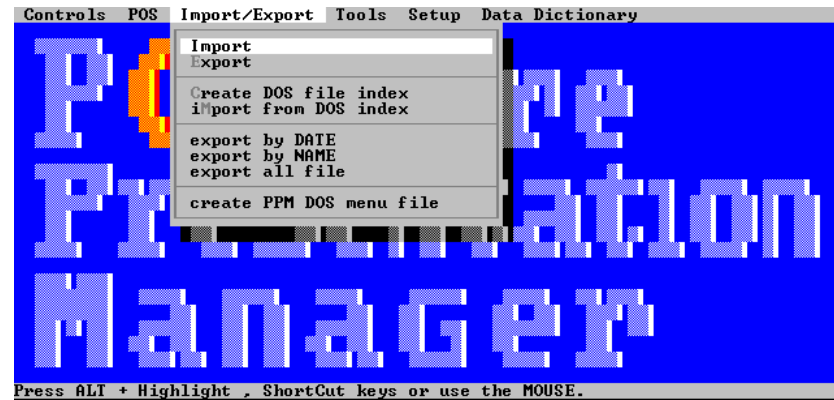
IPM allows you to select from a predefined bank of colors.

The available colors are:

001 Shadow	017 Modifier
002 Data	018 Arrows
003 Function	019 Pull down main
004 Error msg	020 Pull down frame
005 Messages	021 Pull down bar
006 Special	022 Pull down highlight
007 Bottom left	023 List box
008 Help function	024 List box bar
009 Help data	025 List box key
010 Help menu name	026 Push button
011 Not used	027 Push button mark
012 Bottom right	028 System error
013 Store name	029 First screen
014 Frame	030 First screen header
015 Special	031 Second screen
016 Not used	032 Second screen header

## Import / Export

IPM has a utility for importing and exporting information, in various ways.



Options	Description
Import	Import an IPM DOS export file to the current PPM QDX file.
Export	Export selected data to a DOS PPM export file.
Create DOS file index	Export the current IPM file to a DOS file, that can be used later by the application as an IPM file (rather than QDX file).
Import from DOS index	Import an IPM DOS file on top of the current QDX file.
Export by DATE	Export all of the IPM records by range of dates.
Export by NAME	Export all of the IPM records by range of key-names.
Export all file	Export all of the IPM records.
Create IPM DOS menu file	Export IPM's QDX file to a DOS file, in order to run PPM.EXE without a Q-Dex file. Same as Create-DOS-file-index for IPM's Q-Dex file.

## Tools

IPM provides various tools within the IPM menu system. See the list below.

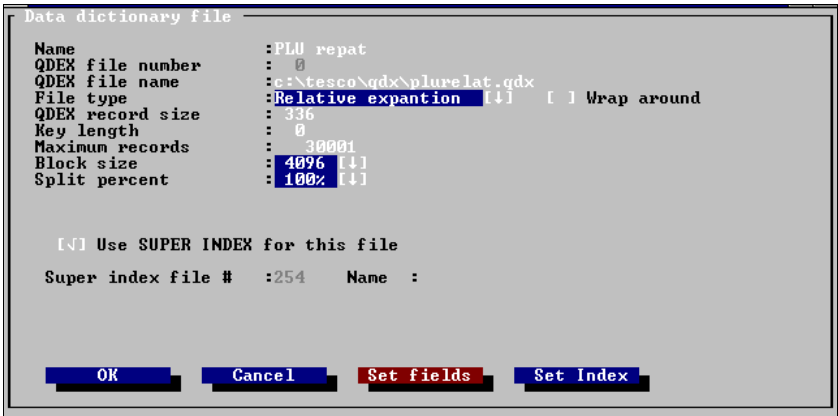


Tool	Description
Statistics	Display information regarding PPM QDX file, including free space and relative usage by each control.
Compress Data	Run compression algorithms on the QDX file. Usually done transparently.
DOS shell	Go to DOS prompt temporarily. Returns to IPM after issuing an EXIT command from DOS.
List.com	Run list (shareware utility for viewing files in ASCII/HEX format).
PE	Run PE editor.

# Data Dictionary

Fields in IPM can be linked directly to a Q-Dex file, using the Data Dictionary.

The Data Dictionary is a definition of a Q-Dex file, with fields from the field's data bank linked to it, with additional information about offset, length and data-type in the file.



### A brief review of Q-Dex files

Q-Dex is a resident driver for handling indexed files and flat-binary files from a few different types.

All Q-Dex files must have a file name, plus a Q-Dex file number, for identification from applications/IPM. All Q-Dex files are fixed-length in records. A maximum number of records must be pre-defined. For index files, a split-percentage is defined to increase efficiency according to ways of data handling for the files.

QDX files can be one of the following types:

- **Stand-Alone** index files: These are index files that contain a unique key for the index, plus additional information for the QDX user. IPM files are from this type, where the key is the `control_type+control_name+internal_number`.

For example: {Screen-op-code}{'EXAMPLE\_SC'}{0} is the key for the first record in IPM of a screen named 'EXAMPLE\_SC'. The rest of the record contains, in this example, information regarding the definition of the screen.

- **Expanded** index files: These are index files, that contain a unique key for the index, plus a pointer to a record of a *relative-expansion-file*, which contains the additional information for the QDX user. The usage for these files is for indexed files that have a lot of additional information, and files that require more than one index, since a relative-expansion file can be linked to more than one index file. A good example is PLU files. The item index record is built from a PLU number and a pointer to the PLU-relative file. The name index record is built of PLU-name+PLU-number as a key (in order to make a unique key for multiple names), and a pointer to the relative file. Then the relative record contains the PLU information: number, name, department, price etc.
- **Relative expansion** files: These files are always related with at least one expanded-index file. The relation is described above.
- **Relative files**: These are flat-binary files, with interface to random-access or sequential-access, according to user request.  
Example: POS Parameters file is a flat file, with 255 records, 1024 bytes each. Each record contains the POS configuration and setup.

The record number represents the POS number (i.e. POS number 1 configuration is located in record number 1).

- **FIFO files:** These are relative files, with a special access mechanism, to read and write records in the next free space. QDX always keeps the next pointers for read/write. FIFO files can be set to be wrap-around files (maintenance files), or not (transaction files).
- **Super-Index files:** These are QDX internal index files, used to allocate the mapping tables on disk rather than in memory. Super index files must be configured with a name, a free Q-Dex file number, and QDX file number of the index. The rest is calculated by QDX. IPM's Data Dictionary can make the link to the super-index automatically. For relative and FIFO files, a super-index can also be linked, without the need for a file name.

Through the IPM Data Dictionary, all types of QDX files can be defined. When definition is complete, fields from the fields bank can be linked to the file and a new field can be inserted. In addition to the normal field definition, some extra information is entered, as described above.



The additional information for fields is shown in the screen above. This information consists of:

<b>Attribute</b>	<b>Description</b>
Name	Field's name for identification.
Field type	Data type on file Types: <ul style="list-style-type: none"><li>• Not defined: Not used</li><li>• Alpha: Alphanumeric.</li><li>• String: ASCII string.</li><li>• BCD: Binary-coded decimal (2 decimal digits per byte).</li><li>• Char: One byte signed numeric integer value.</li><li>• Byte: One byte unsigned numeric integer value.</li><li>• Integer: Two bytes signed numeric integer value.</li><li>• Word: Two bytes unsigned numeric integer value.</li><li>• Long: Four bytes signed numeric integer value.</li><li>• Dword: Four bytes unsigned numeric integer value.</li><li>• Bit of Byte: Boolean variable (only 1 or 0) of one byte.</li><li>• Bit of word: Boolean variable (only 1 or 0) of one word (2 bytes).</li><li>• Bit of Dword: Boolean variable (only 1 or 0) of one Dword (4 bytes).</li><li>• NIX1: 8812 Nixdorf format one word signed numeric integer value.</li><li>• NIX2: 8812 Nixdorf format 4 bytes BCD signed numeric integer value.</li><li>• NIX3: 8812 Nixdorf format 6 bytes BCD signed numeric integer value.</li><li>• NIX4: 8812 Nixdorf format 8 bytes BCD signed numeric integer value.</li></ul>

Attribute	Description
	<ul style="list-style-type: none"> <li>• Date int: Date, represented by 2 bytes numeric integer (binary-coded).</li> <li>• BCD YYMMDD: Date, in BCD yymmdd.</li> <li>• BCD DDMMYY: Date, in BCD ddmmyy.</li> <li>• ASCII YYMMDD: Date, in ASCII yymmdd.</li> <li>• ASCII DDMMYY: Date, in ASCII ddmmyy.</li> <li>• ASCII (numeric): Date, in numeric.</li> <li>• ELDOR ALPHA: 8812 ELDOR format alpha string.</li> <li>• Time BCD: Time, BCD hhmm(ss).</li> <li>• Time ASCII: Time, ASCII hhmm(ss).</li> <li>• Time (LONG format): Time, represented by Dword value, counting seconds in the day, starting midnight.</li> </ul>
Bit index	For bit fields, the bit-number (0-7).
Position in file	Offset in the record to allocate field.
File size	Number of bytes in file needed to keep the field.



© **International Computers Limited 1995-2000**

ICL Retail Systems Inc. endeavors to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission.

The development of ICL Retail Systems products and services is continuous and published information may not be up to date. It is important to check the current position with ICL Retail Systems. This document is not part of a contract or license save insofar as may be expressly agreed.

---

**ICL Retail Systems**  
**2933 Bunker Hill Lane, #101**  
**Santa Clara, CA 95054**

**P/N 89000054**  
**PIN 45001/006**