

# FlagShip



**Object Oriented  
Database  
Development System**

**Cross-Compatible to Unix,  
Linux and MS-Windows**

 **MULTISOFT**

**Release 7.1**

**Section**

**APP**

## The whole FlagShip 7 manual consist of following sections:

Section	Content	Pages
<b>GEN</b>	General information: License agreement & warranty, installation and de-installation, registration and support	18
<b>LNG</b>	FlagShip language: Specification, database, files, language elements, multiuser, multitasking, FlagShip extensions and differences	176
<b>FSC</b>	Compiler & Tools: Compiling, linking, libraries, make, run-time requirements, debugging, tools and utilities	90
<b>CMD</b>	Commands and statements: Alphabetical reference of FlagShip commands, declarators and statements	486
<b>FUN</b>	Standard functions: Alphabetical reference of FlagShip functions	640
<b>OBJ</b>	Objects and classes: Standard classes for Get, Tbrowse, Error, Application, GUI, as well as other standard classes	368
<b>RDD</b>	Replaceable Database Drivers	38
<b>EXT</b>	C-API: FlagShip connection to the C language, Extend C System, Inline C programs, Open C API, Modifying the intermediate C code	160
<b>FS2</b>	Alphabetical reference of FS2 Toolbox functions	376
<b>QRF</b>	Quick reference: Overview of commands, functions and environment	40
<b>PRE</b>	Preprocessor, includes, directives	30
<b>SYS</b>	System info, porting: System differences to DOS, porting hints, data transfer, terminals and mapping, distributable files	42
<b>REL</b>	Release notes: Operating system dependent information, predefined terminals	8
<b>APP</b>	Appendix: Inkey values, control keys, ASCII-ISO table, error codes, dBase and FoxPro notes, forms	34
<b>IDX</b>	Index of all sections	42
<b>fsman</b>	The on-line manual contains all above sections, search function, and additionally last changes and extensions	variable



*multisoft Datentechnik, Munich, Germany*

Copyright (c) 1992..2009

All rights reserved



***Object Oriented Database Development System,  
Cross-Compatible to UNIX, Linux and MS-Windows***

## **Section APP**

Manual release: 7.1

For the current program release see label on distribution disk and  
your Activation Card, or check on-line by issuing *FlagShip -version*

# Copyright

Copyright © 1992..2009 by multisoft Datentechnik, D-81545 Munich, Germany. All rights reserved worldwide. Manual authors: Jan V. Balek, Ibrahim Tannir, Sven Koester

No part of this publication may be copied or distributed, transmitted, transcribed, stored in a retrieval system, or translated into any human or computer language, in any form or by any means, electronic, mechanical, magnetic, manual, or otherwise; or disclosed to third parties without the express written permission of multisoft Datentechnik. Please see also "License Agreement", section GEN.2

Made in Germany. Printed in Germany.

# Trademarks

**FlagShip™** is trademark of multisoft Datentechnik. Other trademarks: dBASE is trademark of Borland/Ashton-Tate, Clipper of CA/Nantucket, FoxBase of Microsoft/Fox, UNIX of AT&T/USL/SCO, AIX of IBM, MS-DOS and MS-Windows of Microsoft. Other products named herein may be trademarks of their respective manufacturers.

# Headquarter Address

## **Headquarter:**

multisoft Datentechnik  
Harthausen Str. 85  
81545 München  
Germany

**E-mail:** support@flagship.de  
support@multisoft.de  
sales@multisoft.de

**Telephone:** (+49-89) 6490040  
**Fax:** (+49-89) 6412974

**Web/Ftp:** <http://www.fship.com>  
<ftp://mult-soft.de/pub>

Call or e-mail multisoft for your local dealer or distributor

# APP: Appendix

---

INKEY and LASTKEY Return Codes .....	3
Control Keys for READ.....	4
Control Keys for MEMOEDIT .....	5
Control Keys for DBEDIT .....	6
Control Keys for ACHOICE .....	7
PC8 ASCII - ISO 8859-1 Table .....	8
Error Codes .....	10
Differences to dBASE III+.....	18
Supported FoxBase and FoxPro commands and functions .....	20
Selected examples and hints.....	27
Registration Card.....	30
Support Request Form .....	32
Index.....	34



# INKEY and LASTKEY Return Codes

Function Key	Numer.Code	terminfo	Compatibility
F1	28	kf1	Clipper F1
F2	-1	kf2	Clipper F2
F3	-2	kf3	Clipper F3
F4	-3	kf4	Clipper F4
F5	-4	kf5	Clipper F5
F6	-5	kf6	Clipper F6
F7	-6	kf7	Clipper F7
F8	-7	kf8	Clipper F8
F9	-8	kf9	Clipper F9
F10	-9	kf10	Clipper F10
F11	-40	kf11	FlagShip only
F12	-41	kf12	FlagShip only
shift-F1	-10	kf13	Clipper F11
shift-F2	-11	kf14	Clipper F12
shift-F3 to F10	-12 to -19	kf15-22	Clipper F13..20
shift-F11	-42	kf23	FlagShip only
shift-F12	-43	kf24	FlagShip only
ctrl-F1	-20	kf25	Clipper F21
ctrl-F2 to ctrl-F10	-21 to -29	kf26-34	Clipper F22..30
ctrl-F11	-44	kf35	FlagShip only
ctrl-F12	-45	kf36	FlagShip only
alt-F1 (or shift-ctrl-Fx)	-30	kf37	Clipper F31
alt-F2 to alt-F10	-31 to -39	kf38-46	Clipper F32..40
alt-F11	-46	kf47	FlagShip only
alt-F12	-47	kf48	FlagShip only
ctrl-D Cursor right	4	kcuf1	same as Clipper
ctrl-S Cursor left	19	kcub1	same as Clipper
ctrl-E Cursor up	5	kcuu1	same as Clipper
ctrl-X Cursor down	24	kcud1	same as Clipper
ctrl-B ctrl-Cursor right	2	kctab	same as Clipper
ctrl-Z ctrl-Cursor left	26	khts	same as Clipper
ctrl-V Insert (ins)	22	kich1	same as Clipper
ctrl-G Delete (del)	7	kdch1	same as Clipper
ctrl-A Home (home)	1	khome	same as Clipper
ctrl-F End (end)	6	kend	same as Clipper
ctrl-R Page-up (PgUp)	18	kpp	same as Clipper
ctrl-C Page-dwn (PgDn)	3	knp	same as Clipper
ctrl-] ctrl-Home (^home)	29	ked	same as Clipper
ctrl-W ctrl-End (^end)	23	kel	same as Clipper
ctrl-- ctrl-Page-up (^PgUp)	31	kri	same as Clipper
ctrl-^ ctrl-Page-dwn (^PgDn)	30	kind	same as Clipper
ctrl-A...ctrl-^	1...30		see ASCII

# Control Keys for READ

---

Key (note)		lastkey()	terminfo	Action
ctrl-S**	Cursor left	19	kcub1	character left
ctrl-D	Cursor right	4	kcuf1	character right
ctrl-Z	ctrl-Cursor left	26	khts	word left
ctrl-B	ctrl-Cursor right	2	kctab	word right
ctrl-E	Cursor up	5	kcuu1	previous GET
ctrl-X	Cursor down	24	kcud1	next GET
ctrl-M	Return, Enter	13	kent	next GET
ctrl-A	Home (home)	1	khome	first charact. in GET
ctrl-F	End (end)	6	kend	last character in GET
ctrl-]	ctrl-Home (^home)	29	ked	first GET / READ
ctrl-W*	ctrl-End (^end)	23	kel	last GET / READ
ctrl-V	Insert (ins)	22	kich1	insert on/off
ctrl-G	Delete (del)	7	kdch1	delete act.character
ctrl-H	Backspace ( <- )	8	kbs	delete previous char
ctrl-T		20		delete word right
ctrl-Y		25		delete GET from cursor
ctrl-U		21		restore orig. value
ctrl-R	Page-up (PgUp)	18	kpp	end of READ
ctrl-C	Page-dwn (PgDn)	3	knp	end of READ
	Escape (ESC)	27	kext	terminates READ

## Notes:

- \* The marked ctrl-[key] differs from Clipper. The ctrl-keys are in FlagShip always consistently assigned to the appropriate function key, see also the INKEY() table.
- \*\* The key is often redefined by stty, see chapter 9.2



# Control Keys for MEMOEDIT

Key (note)		lastkey()	terminfo	Action	
ctrl-S	**	Cursor left	19	kcub1	character left
ctrl-D		Cursor right	4	kcuf1	character right
ctrl-E		Cursor up	5	kcuu1	previous line
ctrl-X		Cursor down	24	kcud1	next line
ctrl-Z		ctrl-Cursor left	26	khts	word left
ctrl-B	*	ctrl-Cursor right	2 *	kctab	word right
ctrl-A		Home (home)	1	khome	line begin
ctrl-F		End (end)	6	kend	line end
ctrl-]		ctrl-Home (^home)	29	ked	top of page
ctrl-L	*	(instead of ctrl-End)	12*		end of page
ctrl-R		Page-up (PgUp)	18	kpp	previous page
ctrl-C		Page-dwn (PgDn)	3	knp	next page
ctrl--		ctrl-Page-up (^PgUp)	31	kri	begin of text
ctrl-^		ctrl-Page-dwn (^PgDn)	30	kind	end of text
ctrl-T			20		delete word right
ctrl-Y			25		delete line at cursor
ctrl-V		Insert (ins)	22	kich1	insert on/off
ctrl-U	*		21*		reformat text
ctrl-W		ctrl-End (^end) *	23	kel	end of memoedit
		Escape (ESC)	27	kext	terminate memoedit

UDF-status	Description
0	idle, all keys are processed
1	invalid key depressed, text unchanged
2	invalid key depressed, text altered
3	init status

UDF-return	Description
0	continue Memoedit, process actual key
1..31	process key 1 to 31. The key (2) is in FlagShip interpreted as 21 (reformat), for compatibility to Clipper
32	ignore this key
33 *	ignore this key (Clipper: save control key as character value in text)
34	word wrap on/off
35	scroll on/off
100	perform function key (2) word right (ctrl-Cursor right, ^→ )
101	perform function key "text end", same as (12).

Notes: \* and \*\* Differences to Clipper, see note in READ table

# Control Keys for DBEDIT

Key	(note)		lastkey()	terminfo	Action
ctrl-S	**	Cursor left	19	kcub1	column left
ctrl-D	*	Cursor right	4	kcuf1	column right
ctrl-E	*	Cursor up	5	kcuu1	previous row
ctrl-X	*	Cursor down	24	kcud1	next row
ctrl-Z	*	ctrl-Cursor left	26	khts	scroll left
ctrl-B	*	ctrl-Cursor right	2	kctab	scroll right
ctrl-A	*	Home (home)	1	khome	first field
ctrl-F	*	End (end)	6	kend	last field
ctrl-]	*	ctrl-Home (^home)	29	ked	previous field
ctrl-W	*	ctrl-End (^end)	23	kel	next field
ctrl-R	*	Page-up (PgUp)	18	kpp	previous page
ctrl-C	*	Page-dwn (PgDn)	3	knp	next page
ctrl--	*	ctrl-Page-up (^PgUp)	31	kri	first record
ctrl-^	*	ctrl-Page-dwn (^PgDn)	30	kind	last record
ctrl-M	*	Return, Enter	13	kent	end of Dbedit
		Escape (ESC)	27	kext	abort Dbedit

UDF-status	Description
0	idle, all keys are processed
1	cursor up depressed, first record reached (bof)
2	cursor down depressed, last record reached (eof)
3	database is empty
4	undefined key or Esc, Enter

UDF-return	Description
0	exit from DBEDIT
1	continue DBEDIT
2	continue DBEDIT, re-read data
3	append a record

## Notes:

- \* The marked ctrl-[key] differs from Clipper. The ctrl-keys are in FlagShip always consistently assigned to the appropriate function key, see also the INKEY() table.
- \*\* The key is often redefined by stty, see chapter 9.2

# Control Keys for ACHOICE

Key	(note)	lastkey()	terminfo	Action	
ctrl-E	Cursor up	5	kcuu1	one item up	
ctrl-X	Cursor down	24	kcud1	one item down	
ctrl-R	Page-up (PgUp)	18	kpp	previous page	
ctrl-C	Page-dwn (PgDn)	3	knp	next page	
ctrl-A	*	Home (home)	1	khome	first item
ctrl-F	*	End (end)	6	kend	last item
ctrl-]		ctrl-Home (^home)	29	ked	first item in window
ctrl-w		ctrl-End (^end)	23	kel	last item in window
ctrl--		ctrl-Page-up (^PgUp)	31	kri	first choice
ctrl-^		ctrl-Page-dwn (^PgDn)	30	kind	last choice
!..A..Z		first character	asc(..)		go to item ...
ctrl-M	*	Return, Enter	13	kent	choice, returns item no
ctrl-S	*	Cursor left	19	kcub1	abort, returns 0
ctrl-D	**	Cursor right Escape (ESC)	4 27	kcuf1 kext	abort, returns 0    abort, returns 0

UDF-status	Description
0	idle, all keys are processed
1	cursor up pressed, first item reached
2	cursor down pressed, last item reached
3	undefined key pressed
4	display mode only, no choice

UDF- return Description 0 abort selection, return 0 1 terminate selection, return the actual item number 2 continue selection

Notes:

- \* the marked keys are not available to the ACHOICE() when UDF is included. Status 3 is passed to the UDF instead.

# PC8 ASCII - ISO 8859-1 Table

dec	hex	ASCII	ISO	ctrl	dec	hex	ASCII	ISO	dec	hex	ASCII	ISO	dec	hex	ASCII	ISO
0	0x00		NUL	^@	32	0x20		space	64	0x40	@	@	96	0x60	`	`
1	0x01	☺	SOH	^A	33	0x21	!	!	65	0x41	A	A	97	0x61	a	a
2	0x02	☻	STX	^B	34	0x22	"	"	66	0x42	B	B	98	0x62	b	b
3	0x03	♥	ETX	^C	35	0x23	#	#	67	0x43	C	C	99	0x63	c	c
4	0x04	♦	EOT	^D	36	0x24	\$	\$	68	0x44	D	D	100	0x64	d	d
5	0x05	♣	ENQ	^E	37	0x25	%	%	69	0x45	E	E	101	0x65	e	e
6	0x06	♠	ACK	^F	38	0x26	&	&	70	0x46	F	F	102	0x66	f	f
7	0x07	•	BEL	^G	39	0x27	'	'	71	0x47	G	G	103	0x67	g	g
8	0x08	▣	BS	^H	40	0x28	(	(	72	0x48	H	H	104	0x68	h	h
9	0x09	◯	HT	^I	41	0x29	)	)	73	0x49	I	I	105	0x69	i	i
10	0x0A	■	LF	^J	42	0x2A	*	*	74	0x4A	J	J	106	0x6A	j	j
11	0x0B	♂	VT	^K	43	0x2B	+	+	75	0x4B	K	K	107	0x6B	k	k
12	0x0C	♀	FF	^L	44	0x2C	,	,	76	0x4C	L	L	108	0x6C	l	l
13	0x0D	♪	CR	^M	45	0x2D	-	-	77	0x4D	M	M	109	0x6D	m	m
14	0x0E	♫	SO	^N	46	0x2E	.	.	78	0x4E	N	N	110	0x6E	n	n
15	0x0F	☼	SI	^O	47	0x2F	/	/	79	0x4F	O	O	111	0x6F	o	o
16	0x10	▶	DLE	^P	48	0x30	0	0	80	0x50	P	P	112	0x70	p	p
17	0x11	◀	XON	^Q	49	0x31	1	1	81	0x51	Q	Q	113	0x71	q	q
18	0x12	↕	DC2	^R	50	0x32	2	2	82	0x52	R	R	114	0x72	r	r
19	0x13	!!	XOF	^S	51	0x33	3	3	83	0x53	S	S	115	0x73	s	s
20	0x14	¶	DC4	^T	52	0x34	4	4	84	0x54	T	T	116	0x74	t	t
21	0x15	§	NAK	^U	53	0x35	5	5	85	0x55	U	U	117	0x75	u	u
22	0x16	■	SYN	^V	54	0x36	6	6	86	0x56	V	V	118	0x76	v	v
23	0x17	↕	ETB	^W	55	0x37	7	7	87	0x57	W	W	119	0x77	w	w
24	0x18	↑	CAN	^X	56	0x38	8	8	88	0x58	X	X	120	0x78	x	x
25	0x19	↓	EM	^Y	57	0x39	9	9	89	0x59	Y	Y	121	0x79	y	y
26	0x1A	→	SUB	^Z	58	0x3A	:	:	90	0x5A	Z	Z	122	0x7A	z	z
27	0x1B	←	ESC	^[	59	0x3B	;	;	91	0x5B	[	[	123	0x7B	{	{
28	0x1C	┐	FS	^\	60	0x3C	<	<	92	0x5C	\	\	124	0x7C		
29	0x1D	↔	GS	^]	61	0x3D	=	=	93	0x5D	]	]	125	0x7D	}	}
30	0x1E	▲	RS	^^	62	0x3E	>	>	94	0x5E	^	^	126	0x7E	~	~
31	0x1F	▼	US	^-	63	0x3F	?	?	95	0x5F	_	_	127	0x7F	␣	␣

# PC8 ASCII - ISO 8859-1 Table (continued)

dec	hex	ASCII	ISO	dec	hex	ASCII	ISO	dec	hex	ASCII	ISO	dec	hex	ASCII	ISO
128	0x80	Ç	€	160	0xA0	á		192	0xC0	À		224	0xE0	α	à
129	0x81	ü		161	0xA1	í	ì	193	0xC1	Á		225	0xE1	β	á
130	0x82	é		162	0xA2	ó	ô	194	0xC2	Â		226	0xE2	Γ	â
131	0x83	â		163	0xA3	ú	û	195	0xC3	Ã		227	0xE3	π	ã
132	0x84	ä		164	0xA4	ñ	€	196	0xC4	Ä		228	0xE4	Σ	ä
133	0x85	à		165	0xA5	Ñ	¥	197	0xC5	Å		229	0xE5	σ	å
134	0x86	å		166	0xA6	a	!	198	0xC6	Æ		230	0xE6	μ	æ
135	0x87	ç		167	0xA7	o	§	199	0xC7	Ç		231	0xE7	τ	ç
136	0x88	è		168	0xA8	¿	..	200	0xC8	È		232	0xE8	Φ	è
137	0x89	ë		169	0xA9	Γ	©	201	0xC9	É		233	0xE9	Θ	é
138	0x8A	è		170	0xAA	Γ	a	202	0xCA	Ê		234	0xEA	Ω	ê
139	0x8B	ï		171	0xAB	½	<<	203	0xCB	Ë		235	0xEB	δ	ë
140	0x8C	î		172	0xAC	¼	Γ	204	0xCC	Ì		236	0xEC	∞	ì
141	0x8D	ì		173	0xAD	í	-	205	0xCD	Í		237	0xED	φ	í
142	0x8E	Ä		174	0xAE	«	®	206	0xCE	Î		238	0xEE	ε	î
143	0x8F	À		175	0xAF	»	—	207	0xCF	Ï		239	0xEF	∩	ï
144	0x90	É		176	0xB0		°	208	0xD0	Ð		240	0xF0	≡	ð
145	0x91	æ		177	0xB1		±	209	0xD1	Ñ		241	0xF1	±	ñ
146	0x92	Æ		178	0xB2		²	210	0xD2	Ò		242	0xF2	≥	ò
147	0x93	ô		179	0xB3		³	211	0xD3	Ó		243	0xF3	≤	ó
148	0x94	ö		180	0xB4		´	212	0xD4	Ô		244	0xF4	∫	ô
149	0x95	ò		181	0xB5	≡	μ	213	0xD5	Õ		245	0xF5	∫	õ
150	0x96	û		182	0xB6	≡	¶	214	0xD6	Ö		246	0xF6	÷	ö
151	0x97	ù		183	0xB7	≡	·	215	0xD7	×		247	0xF7	≈	÷
152	0x98	ÿ		184	0xB8	≡	¸	216	0xD8	Ø		248	0xF8	°	ø
153	0x99	Ö		185	0xB9	≡	¹	217	0xD9	Ù		249	0xF9	•	ù
154	0x9A	Ü		186	0xBA	≡	º	218	0xDA	Ú		250	0xFA	•	ú
155	0x9B	ç		187	0xBB	≡	»	219	0xDB	Û		251	0xFB	√	û
156	0x9C	£		188	0xBC	≡	¼	220	0xDC	Ü		252	0xFC	n	ü
157	0x9D	¥		189	0xBD	≡	½	221	0xDD	Ý		253	0xFD	²	ý
158	0x9E	₹		190	0xBE	≡	¾	222	0xDE	Þ		254	0xFE	■	þ
159	0x9F	f		191	0xBF	Γ	¿	223	0xDF	ß		255	0xFF		ÿ

# Error Codes

---

Error codes displayed by the Run-Time-Errorsystem (RTE) where additional description appears. See also the FlagShip Error-System and files FSerrmsg.c, FSerrmsg\_ger.c and FErrors.h

Definition	Number	Message / Cause / Fix
GENERAL	0	<b>M:</b> general (unspecified) error <b>C/F:</b> see the displayed error message for details
RECNOOR	1	<b>M:</b> record number out of range <b>C:</b> Access/replace on EOF(), BOF() or with record# < 0 or record# > RecCount() <b>F:</b> check RECNO() or the passed record number, debugger
IOOPEN	101	<b>M:</b> file open error <b>C:</b> General acces/open error. File not available (lower/uppercase?), or insufficient access rights to file or directory, or exceeded quotas or system/user max files number <b>F:</b> see LNG.3.3, LNG.9, FILE(), #include "fspreset.fh", system docu
IOCLOS	102	<b>M:</b> file close error <b>C:</b> Insufficient disk space, exceeded quotas, network Failure <b>F:</b> see system "ls", "df"
IOREAD	103	<b>M:</b> file read error <b>C:</b> File deleted by other process, disk or network failure <b>F:</b> use and check database or file locking
IOWRIT	104	<b>M:</b> file write error <b>C/F:</b> same as IOREAD = 103
IOCREAT	105	<b>M:</b> create file error <b>C:</b> Insufficient access rights to directory <b>F:</b> see LNG.3.3, system "ls -ld ."
IOERASE	106	<b>M:</b> erase file error <b>C/F:</b> same as IOCREAT = 105
IOSTAT	107	<b>M:</b> file status error <b>C:</b> access to directory denied, invalid dir name (lower/upper?), insufficient access rights to directory <b>F:</b> see LNG.3.3, DIRECTORY(), system "ls -ld ."
DTMMACH	201	<b>M:</b> data type mismatch <b>C:</b> operation on or comparison of incompatible operators

		<b>F:</b> check VALTYPE() for both operators, debugger FSC.5
DTILOPR	202	<b>M:</b> illegal operation on data type <b>C/F:</b> similar to DTMMACH = 201
DTILEGL	203	<b>M:</b> illegal data type <b>C/F:</b> similar to DTMMACH = 201
DTILLFL	204	<b>M:</b> illegal field type <b>C/F:</b> similar to DTMMACH = 201
DTILVAL	205	<b>M:</b> illegal field value <b>C/F:</b> similar to DTMMACH = 201
RTNOMEM	301	<b>M:</b> no memory available <b>C:</b> physical and virtual (RAM + swap) memory exhausted <b>F:</b> add RAM or at least swap space, avoid extremely large strings and arrays, use LOCAL variables if possible to automatically free unused space for variables.
RTNOVAR	302	<b>M:</b> no memory variables available <b>C:</b> access to undefined variable <b>F:</b> check for typos and var scope/visibility LNG.2.6.3, VALTYPE(), TYPE(), run with debugger FSC.5
RTNLMVP	303	<b>M:</b> null pointer for memory variable <b>C:</b> internal error or incorrect use of Open-C API <b>F:</b> if in your C program: check there. Otherwise report it to multisoft
RTXPMPT	304	<b>M:</b> too many prompts for menu <b>C:</b> should not happen in VFS, since unlimited. Clipper and previous FS versions supported 32 PROMPTs per MENU TO
RTMXGET	305	<b>M:</b> too many gets <b>C:</b> should not happen in VFS, since unlimited
RTNOFLD	306	<b>M:</b> cannot find that field <b>C:</b> invalid field name <b>F:</b> typo, wrong working area (SELECT), closed database. Check USED(), SELECT() or use debugger FSC.5
RTUNVAR	307	<b>M:</b> undefined variable <b>C/F:</b> similar to RTNOVAR = 302
RTUNFUN	308	<b>M:</b> undefined function <b>C:</b> accessing unknown or not linked-in UDF via macro or index key <b>F:</b> see CMD.EXTERNAL, check by ISFUNCTION(), debugger
RTSYNTAX	309	<b>M:</b> Syntax error in expression <b>C:</b> syntax error which could not be detected earlier by compiler (e.g. in macro)

		<b>F:</b> check the statement for correct syntax
RTEEVAL	310	<b>M:</b> NULL pointer returned from run-time system <b>C/F:</b> similar to RTNLMVP = 303
RTARGCNT	311	<b>M:</b> wrong argument count <b>C:</b> insufficient number of mandatory parameters <b>F:</b> check the description for correct invocation
RTARGETYP	312	<b>M:</b> wrong argument type <b>C:</b> incorrect/unsupported type of passed parameter <b>F:</b> check the description for correct invocation
RTARGVAL	313	<b>M:</b> wrong argument value <b>C:</b> incorrect value of passed parameter <b>F:</b> check the description for correct invocation
RTENDLMEXP	314	endless loop in macro expansion <b>C:</b> the macro calls itself many times directly or indirectly <b>F:</b> check the macro, debugger FSC.5
RTTLMEXP	315	<b>M:</b> too long macro expansion <b>C:</b> exceeded buffer size for macro evaluation <b>F:</b> check the macro
RTMISARG	316	<b>M:</b> missing argument <b>C:</b> insufficient number of mandatory parameters <b>F:</b> check the description for correct invocation
RTILLTYP	317	illegal type <b>C:</b> incorrect parameter or operator type <b>F:</b> check the statement for correct syntax
RTIVSKEL	318	<b>M:</b> illegal skeleton <b>C:</b> incorrect/unsupported NEXT/FOR/WHILE clause for this operation <b>F:</b> check the description for correct invocation
RTSTRTL	319	<b>M:</b> string too long for runtime system <b>C:</b> the string length exceeds its limit (2GB for variables, 250 bytes for file name incl. path, 4096 bytes for macro expansion). Can also happen in Extend-C API when passing strings not terminated by zero byte. <b>F:</b> check the correct string length by LEN(), debugger
RTILLVAL	320	<b>M:</b> illegal value <b>C:</b> incorrect parameter or operator value <b>F:</b> check the statement for correct syntax
RTOSERR	321	<b>M:</b> operating system command returned err <b>C:</b> invalid name by system call via RUN, or failure of



		system function in e.g. COPY/DELETE/RENAME FILE, directory access etc. <b>F:</b> the 'errno' system code is returned via FERROR() or DOSERROR() function
RTNODBF	322	<b>M:</b> no database opened <b>C:</b> database operation w/o open database, wrong work area <b>F:</b> check USED(), SELECT(), NETERR(), debugger FSC.5
RTFILNAM	323	<b>M:</b> wrong file name <b>C:</b> file name incorrect, e.g. empty or with illegal characters <b>F:</b> check the passed file name, debugger FSC.5
RTARGNLS	324	<b>M:</b> argument is null string <b>C/F:</b> similar to RTARGVAL = 313
RTNOLOCK	325	<b>M:</b> DBF file or record not locked <b>C:</b> auto-lock disabled and no own RLOCK()/FLOCK() <b>F:</b> avoid SET AUTOLOCK OFF for automatic locking, check ISDBFLOCK(), ISDBRLOCK(), debugger FSC.5
RTDBFVER	326	<b>M:</b> wrong DBF version <b>C:</b> unsupported non-standard xBase database version, e.g. from FoxPro, dBaseIV etc. <b>F:</b> convert the database to dBaseIII+ format by the associated RDBS or export it to text file and create the database by FlagShip using APPEND...SDF or APPEND..DELIMITED
RTNOMORLCKS	327	<b>M:</b> number of locks for the DBF exceeded <b>C:</b> unsufficient system or user environment setting <b>F:</b> see SYS.3
RTWRNGLIN	328	<b>M:</b> incorrect line in file <b>C:</b> invalid value in the key translation table <b>F:</b> check the displayed line#
RTILLCOL	329	<b>M:</b> illegal color specification <b>C:</b> invalid value in color specification <b>F:</b> see SET COLOR
RTNODBF1	330	<b>M:</b> no database opened <b>C/F:</b> similar to RTNODBF = 322
RTIDXRPT	331	<b>M:</b> index file corrupted <b>C:</b> index key count does not match database records, or index key was not found in the database. Most possibly, the database was REPLACed, APPENDed, ZAPed or PACKed without associated index <b>F:</b> see LNG.4.5 index integrity checking, INDEXCHECK()

RTCONVAR	332	<b>M:</b> wrong type for typed var/par <b>C:</b> assigning invalid value to typed variable, or passing parameter to an UDF not matching typed arguments <b>F:</b> check the assignment type, debugger FSC.5
RTILLPICT	333	<b>M:</b> wrong picture in say or get <b>C:</b> incorrect character in PICTURE's type or function <b>F:</b> check the PICTURE clause, see @..SAY, @..GET
RTILLEOF	334	<b>M:</b> DBF on eof or bof but record needed <b>C/F:</b> similar to RECNOOR = 1
RTILLBRK	335	<b>M:</b> Illegal break <b>C:</b> the BREAK statement was invoked but there is no corresponding BEGIN..END SEQUENCE active <b>F:</b> check by ISBEGSEQ(), debugger FSC.5
RTILLDATE	336	<b>M:</b> wrong dateformat in set() <b>C:</b> incorrect date format <b>F:</b> see SET DATE or SET()
RTILLNAM	337	<b>M:</b> undefined identifier <b>C:</b> undefined variable accessed by macro <b>F:</b> check the macro string and VALTYPE() of variables used there, debugger FSC.5
RTILLCLS	338	<b>M:</b> Illegal object class <b>C:</b> invalid class or object type used <b>F:</b> see the error message for details
RTUSERS	339	<b>M:</b> unable to open file <b>C/F:</b> similar to IOOPEN = 101
RTIDXVER	340	<b>M:</b> wrong IDX version <b>C:</b> index was not created by Visual FlagShip <b>F:</b> reindex via INDEX ON..TO..
RTNOFLD1	341	<b>M:</b> cannot find that field <b>C/F:</b> similar to RTNOFLD = 306
RTILLALIAS	342	<b>M:</b> Illegal Alias <b>C:</b> trying to SELECT work area by invalid ALIAS name <b>F:</b> check available aliases by DBFUSED(), debugger FSC.5
RTILLMET	343	<b>M:</b> Illegal Method <b>C:</b> the used class method is not available in the class <b>F:</b> check for correct class name by ISOBJCLASS() and if the used property is not defined as Access/Assign or Extern instance instead using ISOBJPROPERTY() or debugger
RTILLINS	344	<b>M:</b> Illegal Instance

		<b>C:</b> trying to access or assign an instance not available in the class, or not declared Extern instance nor as Access/Assign method <b>F:</b> check for correct class name by ISOBJCLASS() and if the used property is not defined as method or as protected instance by ISOBJPROPERTY() or debugger
RTDECLARED	345	<b>M:</b> Previously declared identifier <b>C:</b> either trying to declare CONSTANT anew, or illegal use of objects <b>F:</b> see the displayed error message
RTILLPAR	346	<b>M:</b> wrong # of parameters or their types <b>C:</b> incorrect parameter type or insufficient parameters <b>F:</b> check the function call for correct syntax
RTLOCKFAIL	347	<b>M:</b> Lock failed <b>C:</b> insufficient system resources or locking via NFS without NFS locking enabled <b>F:</b> see SYS.3, or "mount"
RTNOAPPOBJ	351	<b>M:</b> Constant oApplic not instantiated <b>C:</b> incorrect modification of initio.prg <b>F:</b> check your changes there
RTNOSTDOBJ	352	<b>M:</b> Constant _oStdIO not instantiated <b>C/F:</b> similar to RTNOAPPOBJ = 351
RTNOMEMOBJ	353	<b>M:</b> Constant _oMemoBrow not instantiated <b>C/F:</b> similar to RTNOAPPOBJ = 351
RTWIDCREA	361	<b>M:</b> Could not create GUI widget <b>C:</b> either trying to run an application in GUI mode without X11/Windows active, or incorrect modification of initio.prg, or insufficient memory <b>F:</b> see the error message for details
RTWIDAVAIL	362	<b>M:</b> GUI widget not available <b>C:</b> the accessed widget/control was already deleted or is out of the visibility scope <b>F:</b> see the error message for details
RTCLASSTYPE	371	<b>M:</b> Illegal class type <b>C:</b> trying to assign an object of incompatible class <b>F:</b> check for correct class name by ISOBJCLASS() or debugger FSC.5
RTCLASSPROP	372	<b>M:</b> Invalid class property <b>C:</b> trying to access or assign an instance or method not available in the class, or not declared Extern <b>F:</b> check for correct class name by ISOBJCLASS() and if the used property is not defined as method or as

		protected instance by ISOBJPROPERTY() or debugger
RTOBJNOPROP	373	<b>M:</b> Missing object property <b>C:</b> the initialization of the object is insufficient for the required operation <b>F:</b> see the class description, displayed message, debugger
INTARGCN	401	<b>M:</b> arguments count negative <b>C:</b> internal error or incorrect use of Open-C API <b>F:</b> if in your C program: check there. Otherwise report it to multisoft
INTARGCB	402	<b>M:</b> too many arguments <b>C:</b> passed more than the supported number of paramers, reported usually for code blocks <b>F:</b> check function description for details
INTARGCL	403	<b>M:</b> too little arumentgs <b>C/F:</b> similar to RTARGCNT = 311
INTARGETYP	404	<b>M:</b> wrong argument type <b>C/F:</b> similar to RTARGETYP = 312
INTARGNEG	405	<b>M:</b> negative argument <b>C/F:</b> similar to RTARGVAL = 313
INTARGNUL	406	<b>M:</b> argument is zero <b>C/F:</b> similar to RTARGVAL = 313
INTARGNLP	407	<b>M:</b> argument is null pointer <b>C:</b> internal error or incorrect use of Open-C API <b>F:</b> if in your C program: check there. Otherwise report it to multisoft
INTARGNLS	408	<b>M:</b> argument points to null string <b>C/F:</b> similar to RTARGVAL = 313
INTARGOR	409	<b>M:</b> argument out of range <b>C/F:</b> similar to RTARGVAL = 313
INTCMDFLG	410	<b>M:</b> flag has impossible value <b>C/F:</b> similar to RTARGVAL = 313
INTMEMOUT	411	<b>M:</b> out of memory <b>C/F:</b> similar to RTNOMEM = 301
INTNDXSEC	412	<b>M:</b> non-existing index data sector access <b>C:</b> index not created by FlagShip, or internal error <b>F:</b> report it to multisoft
INTDBFNO	413	<b>M:</b> database file (DBF) not opened <b>C/F:</b> similar to RTNODBF = 322
INTDBTNE	414	<b>M:</b> database file has no memo file (DBT)

		<b>C:</b> there is a "M" field in the database but the corresponding .dbt file does not exist or is not accessible <b>F:</b> check files by "ls -la"
INTNDXNO	415	<b>M:</b> index file (IDX) not opened <b>C:</b> index operation (e.g. SEEK) w/o open or with disabled controlling index <b>F:</b> check INDEXCOUNT(), INDEXORD(), INDEXDBF(), debugger
INTILLVAL	416	<b>M:</b> illegal internal value <b>C:</b> index not created by FlagShip, or internal error <b>F:</b> report it to multisoft
FTNOTOPN	501	<b>M:</b> unable to open file <b>C/F:</b> similar to IOOPEN = 101
FTFUNARG	502	<b>M:</b> too many function arguments <b>C:</b> passed more than the supported number of paramers, reported for few standard functions which do not accept/do not ignore additional parameters <b>F:</b> check the function description for details
LOCKNDX	701	<b>M:</b> index file already locked <b>C:</b> index not created by FlagShip, or internal error <b>F:</b> report it to multisoft

# Differences to dBASE III+

---

The compatibility of FlagShip to dBASE III plus is similar to their compatibility to Clipper:

dBASE, Fox commands	FlagShip command or function
APPEND, INSERT, EDIT	APPEND BLANK
ASSIST	-
BROWSE	DBEDIT(), BROWSE()
CHANGE	REPLACE
CLEAR FIELDS	-
CREATE LABEL	LABEL EDIT
CREATE REPORT	REPORT EDIT
CREATE QUERRY, SCREEN	-
DISPLAY FILES	DIRECTORY(), FILE()
DISPLAY MEMORY	MEMORY()
DISPLAY STATUS	debugger, SET(), FS_SET()
DISPLAY	display <fields>
STRUCTURE	CREATE STRUCTURE, DBSTRUCT(), DBCREATE()
DISPLAY USERS	USERSACTIVE()
ERROR()	FERROR(), Error handle, Error object
EXPORT TO	-
HELP	Procedure HELP
IMPORT FROM	APPEND FROM
LIST FILES	FILE(), DIRECTORY()
LIST HISTORY, MEMORY	debugger
LIST STATUS, STRUCTURE	debugger, DBSTRUCT()
LOAD	CALL, #Cinline, Open C API
LOGOUT	QUIT
MESSAGE	-
MODIFY COMMAND	Standard text editor
MODIFY QUERRY, SCREEN	-
MODIFY LABEL	LABEL EDIT
MODIFY REPORT	REPORT EDIT
ON ERROR	ERRORBLOCK(), BEGIN SEQUENCE..BREAK/RECOVER
ON ESCAPE, KEY	SET KEY TO
RESUME, RETURN TO	BREAK, BEGIN SEQUENCE...END
RETRY	DO WHILE !lock() ... ENDDO
RETURN TO MASTER	BEGIN SEQUENCE ... BREAK ... END
SET	debugger
SET ...	see SET commands and SET() or FS_SET() funct

Macro evaluation: In FlagShip (same as in Clipper), the macro cannot contain command, or part of commands, or command clauses. A single command value is accepted as macro. See further details in section LNG.2.10

Database and index: standard dBaseIII+ database (.dbf) and memo fields (.dbt) are fully supported and are backward compatible. Indices (.ndx and .mdx) are incompatible, you need to re-index them by INDEX ON..TO..

# Supported FoxBase and FoxPro commands and functions

---

FoxPro (2.5, 2.6) and Foxbase is a superset of dBase containing many xBase incompatible structures. The most of them are supported in FlagShip by using the **-fox** compiler switch (e.g. "FlagShip -fox myapp.prg"). This automatically includes the preprocessor file <FlagShip\_dir>/include/**stdfoxpro.fh**, which extends (or overrides) standard command translation by **std.fh** preprocessor file. Here summary of main features:

= command	supported by DO ...
and / or / not	translated to .AND. / .OR. / .NOT.
m. (prefix)	translated to m->
@...EDIT...	supported by MemoEdit()
@...MENU...	supported by Achoice()
@...MENU TITLE,SHADOW	supported by fox_AtMenu() in foxpro_api.prg
@...SAY	supported including FONT, FUNCTION, PICTURE clauses, but COLOR SCHEME, SIZE, STYLE are ignored
@...GET	supported by standard @..GET, but clauses SCHEME, FONT, STYLE, SIZE, OPEN, WINDOW, FROM are ignored
? COLOR, FONT	supported by fox_Qout() in foxpro_api.prg
?? COLOR, FONT	supported by fox_Qqout() in foxpro_api.prg
ACTIVATE MENU [PAD]	supported by TopBar class
ACTIVATE POPUP	supported by fox_PopupExec() in foxpro_api.prg
ACTIVATE WINDOW	supported by Wselect() or fox_MDlselect(), but ignored for IN WINDOW SCREEN, BOTTOM, TOP, SAME, NOSHOW
AELEMENT(..)	supported by direct array access, 3rd param un_supp
ALEN(..)	supported, 2nd parameter if given, only as 0
APPEND FROM ARRAY	supported except FIELDS clause
APPEND MEMO	supported by REPLACE
BAR()	supported by PopupBar() and PopUp class
BROWSE	supported by fox_Browse() in foxpro_browse.prg
BUILD APP	message, simply compile by FlagShip -fox
BUILD EXE	message, simply compile by FlagShip -fox



CHANGE, EDIT	partially supported via Browse(), message
CLEAR ALL	supported
CLEAR ...	clauses FIELDS, MACROS, MENUS, POPUPS, PROGRAM, WINDOWS are ignored
CLEAR PROMPT	ignored, use @..CLEAR TO...
CLEAR READ	supported by CLEAR GETS
CLOSE ALL	supported by DbCloseAll()
CLOSE PROCEDURE	ignored, since not required
COMPILE	ignored, compile by FlagShip -fox
COPY MEMO TO	supported including ADDITIVE clause
COPY TO ARRAY	supported incl. NOOPTIMIZE, except FIELDS etc. clause
COUNT	supported by DbEval() incl. TO, FOR, WHILE, NEXT, RECORD, REST, ALL, NOOPTIMIZE clauses
CREATE LABEL	supported by LABEL EDIT
CREATE REPORT	supported by REPORT EDIT
CRATE TABLE	message, use DbCreate() instead
DEACTIVATE MENU	supported by TopBar class or fox_DeIMenu()
DEACTIVATE POPUP	supported
DEACTIVATE WINDOW	supported by Wclose() or fox_MDIclose()
DEFINE BAR	supported by PopUp and MenuItem class. but clauses KEY, BEFORE, AFTER, MESSAGE, MARK, COLOR ignored
DEFINE MENU	supported by TopBar and MenuItem class, all clauses except AT LINE are ignored
DEFINE PAD	supported by TopBar and MenuItem class, but clauses KEY, COLOR are ignored
DEFINE POPUP PROMPT	supported by PopUp and MenuItem class, but clauses RELATIVE, SHADOW are ignored
DEFINE POPUP FROM	supported by PopUp and MenuItem class, but only clauses TO, PROMPT FIELD, TITLE, FOOTER considered
DEFINE POPUP ...	supported by PopUp and MenuItem class, but clauses except above listed are ignored
DEFINE WINDOW	supported by Wopen() or fox_MdiOpen() but only clauses AT, SIZE, FROM, TO, TITLE are considered
DELETE	supported by DbDelete(), NOOPTIMIZE clause ignored

DELETE ...	supported by DbEval() including FOR, WHILE, NEXT, RECORD, REST, ALL, NOOPTIMIZE clauses
DIMENSION	supported by DECLARE
DIR	supported, clause ON ignored
DISPLAY FILES	supported incl. LIKE clause, same as DIR
DISPLAY STRUCTURE	supported by DbStruct() and Aeval()
DO ...	supported, .PRG extension and IN clause ignored
EJECT PAGE	supported by fox_EjectPage() in foxpro_api.prg
ENDFOR	supported, same as NEXT
ENDPRINTJOB	supported, see PRINTJOB
ENDSCAN	supported, see SCAN
ERROR()	supported by fox_Error() in foxpro_api.prg
EVALUATE()	supported by &(<var>)
EXTERNAL ...	clauses LABEL, LIBRARY, MENU, PROCEDURE, REPORT, SCREEN ignored
FFLUSH()	supported by DbCommitAll()
FGETS()	supported by FreadTxt()
FILTER()	supported by DbFilter()
FLUSH	supported by DbCommitAll()
FPUTS()	supported by FWrite()
FULLPATH()	supported by TruePath()
GATHER FROM	supported by FieldPutArr() incl. MEMO clause
GATHER MEMVAR	unsupported
GO GOTO IN	supported by dbSelectArea() and dbGoto()
GO GOTO BOTTOM IN	supported by dbSelectArea() and dbGoBottom()
GO GOTO TOP IN	supported by dbSelectArea() and dbGoTop()
INSMODE()	supported by Set(_SET_INSERT)
KEY()	supported by IndexKey()
LINENO()	supported by ProcLine()
LOAD ...	ignored
MEMLINES()	supported by MLCOUNT()
MRSSAGE()	supported by fox_Message() in foxpro_api.prg

MLINE()	supported by MemoLine()
MODIFY MEMO	supported by MemoEdit(), clauses except NOEDIT ignored
MODIFY LABEL	supported by LABEL EDIT, clauses ignored
MODIFY REPORT	supported by REPORT EDIT, clauses ignored
MODIFY STRUCTURE	message, use dbu utility instead
MOVE WINDOW	supported by Wcenter() with FS2, message otherwise
NDX()	supported by IndexNames()
OCCURS()	supported by StrOccurs()
ON BAR	supported by Popup class, ACTIVATE MENU ignored
ON ERROR DO	supported by ErrorBlock()
ON ERROR ...	LABEL and other clauses raises message
ON KEY = DO	supported by OnKey()
ON KEY LABEL	unsupported, message
ON PAD	supported by PopUp and MenuItem class, but clause ACTIVATE MENU is ignored
ON PAGE	supported by fox_OnPage() in foxpro_api.prg
ON SELECTION BAR OF	supported by PopUp and MenuItem class, only clauses DO, WITH, IN, DEACTIVATE MENU are supported
ON SELECTION PAD	supported by PopUp and MenuItem class, only clauses DO, WITH, IN, DEACTIVATE MENU are supported
ON SELECTION POPUP	supported by PopUp and MenuItem class, only clauses DO, WITH are supported
PARAMETERS()	supported by Pcount()
PRINTJOB	supported by fox_Printjob() in foxpro_api.prg
PRINTSTATUS()	supported by IsPrinter()
PRIVATE	supported except ALL clause
PROGRAM()	supported by ProcName() or fox_Sys(16)
PUBLIC	supported except ARRAY clause
READ	supported by READ and fox_Readmenu(), all clases except SAVE and NENU are ignored
RECNO(x)	supported, including optional parameter
RECCOUNT(x)	supported, including optional parameter
RECSIZE(x)	supported, including optional parameter

#REGION	ignored
REGIONAL	supported by LOCAL
RELATION(n,x)	supported by DbRelation(), incl. optional parameter
RELEASE LIBRARY	ignored, not required
RELEASE MODULE	ignored, not required
RELEASE WINDOW	supported by Wclose() or fox_MDIclose()
REPLACE	supported by fox_FieldPut() with WITH, NOOPTIMIZE or by DbEval() for WITH, NOOPTIMIZE, FOR, WHILE, NEXT, RECORD, REST, ALL clauses
RESTORE MACROS	ignored
RESUME	message, use AltId() or debugger
RETRY	message, use loop and NetErr()
RETURN TO	message, use BEGIN SEQUENCE ... BREAK ... END
RLOCK(x)	supported, but only one parameter
SAVE MACROS	ignored
SCAN..ENDSCAN	supported by a loop containing SKIP and filter, but better is to use DbEval() instead
SCATTER	supported by FieldGetArr() incl. MEMO clause
SCATTER MEMVAR	unsupported, use SCARER to array or FieldGetArr()
SCOLS()	supported by MaxCol()
SEEK()	supported by DbSeek()
SELECT	supported by DbSelectArea()
SELECT FROM	unsupported, message, use SQL 3rd party tools
SELECT(0), SELECT()	supported by DbSelectArea()
SELECT(1)	supported by fox_Selexmax() in foxpro_api.prg
SET ...	supported by standard FlagShip SET command, but clauses AUTOSAVE, BLINK, FIELDS, NEAR, LIBRARY, STATUS BAR are ignored
SET DEBUG	ignored, use -d compiler switch
SET FOXYEAR	supported by SET EPOCH TO
SET LOCK	supported by fox_Set(_SET_AUTOLOCK)
SET REPROCESS	unsupported, message
SET SKIP TO	supported fox_setSkip() in foxpro_api.prg

SET STEP	supported, invokes debugger when compiled with -d
SET UDFPARMS	message, use parameter calling convention
SHOW GET	supported by fox_Showget() in foxpro_api.prg
SHOW GETS	message, use @..GET instead
SHOW MENU	supported by TopBar class
SHOW OBJECT	message, use @..GET instead
SHOW POPUP	message, use @..GET or InfoBox() instead
SKIP ... IN ...	supported by DbSkip()
SROWS()	supported by MaxRow()
SUSPEND	supported by AltD() invokes debugger (compile with -d)
SYS()	several SYS() functions are supported by fox_sys() available in foxpro_api.prg, e.g.: SYS(0, 1, 2, 3, 5, 6, 10, 11, 12, 13, 14, 16, 19, 21, 22, 23, 30, 100, 102, 103, 1001, 2001, 2002, 2003, 2004, 2010, 2011)
UNLOCK IN ...	supported
USE	supported by DbUseArea() incl. IN, VIA, ALIAS, NEW, EXCLUSIVE, SHARED, NOUPDATE, READONLY, INDEX, ORDER clauses but AGAIN, ASCENDING, DESCENDING ignored
USE ?	unsupported, specify file name
USE TAG	message, use USE...INDEX...
VARREAD()	supported by ReadVar()

#### *Additional notes:*

- "alias.var" syntax is converted to "alias->var" by -fox compiler switch
- "array(element)" syntax is converted to "array[element]" by -fox switch
- the "=" command (but not assignment) is translated to "DO"
- the "and", "or" and "not" operators are translated to ".and.", ".or." and ".not." when at least one leading and trailing space is included
- .dbf and .dbt databases and memo fields are backward compatible
- .fpt memo files must be converted by FoxPro to dBaseIII+ .dbt format
- .cdx (and Foxbase .idx) indices are not supported by default, you need to re-index them by FlagShip by the common INDEX ON..TO.. command
- REGIONAL variables (and #REGION nn) are not supported, you may replace them by

FlagShip's LOCAL or STATIC variables (in addition to PUBLIC and PRIVATE) with a clear visibility and lifetime scope. Note: as opposite to FoxPro, FlagShip supports unlimited number of variables.

- Macro evaluation: In FlagShip (same as in Clipper), the macro cannot contain command, or part of commands, or command clauses. A single command value is accepted as macro. See further details in LNG.2.10
- Windowing like DEFINE or ACTIVATE WINDOW or MENU or POPUP etc. are emulated by FlagShip's windowing, available fully in the FS2 Tollbox. Without licensed FS2, it is emulated directly in GUI's MDI mode, i.e. the -mdi and -fox compiler switches are required.
- See also <FlagShip\_dir>/include/stdfoxpro.fh for a list of differences, supported FoxPro commands not listed in this manual and it clauses.
- There are several functions/wrappers available in <FlagShip\_dir>/system/ foxpro\_api.prg. These functions are available in the FlagShip library and triggered by the -fox compiler switch. You may freely modify its behavior by re-compiling the source and adding the object in the link list.
- Visual FoxPro is incompatible to itself (i.e. to FoxPro) and therefore only the FoxPro's 2.5, 2.6 syntax is supported by FlagShip.

Hint: compile your FoxPro sources with

```
FlagShip -fox myapp.prg  
or
```

```
FlagShip -fox -mdi -m -Mstart *.prg"
```

and watch possible compiler messages. You may redirect the output to a file (e.g. err.log) by

```
FlagShip ... 2>>err.log
```

or in Linux even better by using

```
script err.log  
FlagShip ...  
FlagShip ...  
exit ##the script, creates err.log file
```

where all the stdin, stdout and stderr are then listed in that log file. Then consult this FlagShip manual for supported commands, it clauses and functions, as well as the stdfoxpro.fh for undocumented but supported FoxPro commands. In the most cases, the displayed errors are solved fast by modifying the corresponding command (at the displayed line number).

# Selected examples and hints

---

Changes of .prg sources for the DOS/UNIX compatibility	QRF, LNG.9.5
FlagShip compiler invocation	FSC.1.2-6
Debugging session	FSC.5
Debugger and break key, user-defined sorting, DOS to UNIX	FUN.FS_SET

Customizing the application for a distribution	SYS.1.2
Determining optimal/max no of users using the same applic	SYS.3
Determining if the current execution is on UNIX or DOS	LNG.9.4-5
Porting hints from/to DOS and another UNIX systems	QRF, SYS.1
System tuning, number of files, resources	SYS.3 [, REL]

## Language

Argument passing from the UNIX shell	LNG.2.3.1
Argument passing to UDFs, usage of optional arguments	FUN.Notation
Usage of optional parameters within a UDF	FUN.PCOUNT
Ambiguous variable vs. dbf field	CMD.FIELD
The usage of aliases with a UDF	CMD.FUNCTION
Speed advantage using typed variables	CMD.LOCAL.AS
Code blocks vs. Macros	LNG.2.3.3
Exceptions and Recovery, e.g. to find a file name	CMD.BEGIN
Scanning an one or a multidimensional array	FUN.ASCAN
Sorting an one or a multidimensional array	FUN.ASORT

## Database, Indices, Files

Creating and modifying a database structure	FUN.DBCREATE
Traversing a database	CMD.DO WHILE
Textual, multiple searching in the index key	CMD.SEEK.EVAL
Report a percentage of the index processed	CMD.INDEX
SHOWINDEX() graphically displays the indexing process	FUN.DBCREATEIN
ADD_NEW_REC() reusing the deleted records instead of PACK	CMD.PACK
Usage of "filtered" index instead of SET FILTER	CMD.INDEX
Usage of SEEK instead of FILTER	CMD.SET.FILTER
Browsing a database using a "filtered" index	FUN.DBEDIT
Soft vs. hard-coded relations	CMD.SET.RELAT
MY_USE() general dbf open UDF with user prompt on error	CMD.USE
MYUSE() opens a .dbf with a user defined error handler	OBJ.ERRORNEW
MYHANDLER() error handler used for denied file access	OBJ.5
Multiuser/multitasking requirements	LNG.4.8
Multiuser/multitasking usage	FUN.RLOCK
Redisplay changed data by other user	CMD.COMMIT

Determine a UNIX or DOS-alike directory  
 Determine/display files in any directory  
 Access denied checking  
 Check the user permission before USEing of database  
 Determining the index/dbf integrity  
 Determining the status of dbf and locking

FUN.CURDIR  
 FUN.DIRECTORY  
 FUN.FCREATE  
 FUN.FOPEN  
 FUN.INDEXCHECK  
 FUN.ISDBxxx

### Input/Otput

Determining the size of the terminal used  
 Nesting of @..GET/READ  
 User modifiable READ  
 SAA look-like menus  
 The usage of vi or other editor alternatively to MEMOEDIT()  
 Usage of a UNIX pipe or device for printing  
 Read lines from a text file  
 General purpose functions for text editing  
 File & path conversion, terminal determining and setting, zero-byte support  
 PRINTSCREEN() prints the current screen contents

FUN.MAXCOL  
 CMD.READ  
 <FlagShip\_dir>/system/getsys.prg  
 CMD.@.PROMPT  
 CMD.!..RUN

CMD.SET.PRINT  
 FUN.FREADTXT  
 FUN.MEMOEDIT

FUN.FS\_SET  
 EXT.4.8

### Numeric, Strings, Screens

Random number generator  
 Sinus, cosinus and other numeric calculation  
 SINUS() demonstrates the usage of UNIX default libraries  
 ROTATESTR() converts string left to right  
 Expanded STRPOKE() or STUFF() function  
 Storing/restoring of screen contents in memo fields  
 DOSSCR2UNIX() converts screen contents from DOS to UNIX  
 Manipulation of the screen variable (contents, colors)  
 Determine coordinates of the SAVESCREEN() variable

EXT.3.3  
 LNG.8  
 EXT.2.5

LNG.8  
 EXT.\_parc  
 FUN.SCREEN2CHR  
 FUN.RESTSCREEN

EXT.\_retscw  
 EXT.VAR\_SCR

### Miscellaneous

SAVERESET() saves and restores all settings  
 Measuring the overall execution time  
 Measuring of the elapsed time, performances  
 Redisplay screen after RUN  
 Executing a UNIX command in background  
 Start-up and program switching script similar to SWITCH.EXE  
 Determining the user name and workstation name  
 Determining the current process id number  
 FUN.FS\_SET("prin  
 Determining the required environment setting  
 Check the number of users and the FlagShip license  
 The usage of TBROWSE

FUN.SET  
 CMD.PROCEDURE  
 FUN.SECONDSCPU  
 CMD.REFRESH  
 CMD.RUN  
 FUN.ERRORLEVEL

FUN.NETNAME  
 t"), FUN.FOPEN

FUN.GETENV  
 FUN.USERSDBF  
 OBJ.3,4



A simple program using TBROWSE and TBCOLUMN  
Inter-process communication, using named pipes

LNG.7  
EXT.5

# Registration Card

---

You will find this Registration Card in text form also in the <FlagShip\_dir>/docu/REGCARD file. Please return it immediately (preferably as e-mail attachment or by fax) to join our free tech support.

-----  
REGISTRATION CARD: FlagShip Developer's License (or Demo)  
-----

This card entitles you to a free initial period support and updates of FlagShip. For DEMO users of FlagShip, this card entitles you for a free installation support and free information about future enhancements and add-ons.

Serial Number \_\_\_\_\_ FS release & license \_7.1.\_\_\_\_\_

[ ] Linux distribut \_\_\_\_\_ OS Release \_\_\_\_\_

cc (cc -v) \_\_\_\_\_

[ ] Windows version \_\_\_\_\_ Release, SP \_\_\_\_\_

-with C compiler: ( ) BCC32 ( ) MS-VC6++ ( ) other \_\_\_\_\_

Operat. System \_\_\_\_\_ OS Release \_\_\_\_\_

Hardware used \_\_\_\_\_ CC release \_\_\_\_\_

LICENSEE:

Company Name \_\_\_\_\_

Last Name \_\_\_\_\_ First Name \_\_\_\_\_

Title \_\_\_\_\_

Department \_\_\_\_\_

Address \_\_\_\_\_

Street, POB \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_

Zip, PostCode \_\_\_\_\_ Country \_\_\_\_\_

Telephone No. \_\_\_\_\_ Fax \_\_\_\_\_

Email (CIS) \_\_\_\_\_ Internet \_\_\_\_\_

Your FlagShip \_\_\_\_\_

Vendor (full  
address pls) \_\_\_\_\_

Purchase Date \_\_\_\_\_

I agree with the FlagShip license conditions and warranty as stated in the manual, section GEN.2. Protection of data privacy: The registration can[ ] (may not[ ]) be passed on to the local distributor.

Date \_\_\_\_\_ Sign \_\_\_\_\_

Sign is not required when  
forwarded by e-mail

\*\*\*\*\*

To better assess and meet YOUR future needs, please answer also the following questions (optional):

Where did you learn  
about FlagShip \_\_\_\_\_

Which Xbase programming  
language are you using \_\_\_\_\_

Do you use also other data-  
base systems, which ones? \_\_\_\_\_

Other programming languages  
you are familiar with \_\_\_\_\_

Your preferred  
programming language \_\_\_\_\_

Do you use FlagShip for  
developing, or porting only \_\_\_\_\_

Other Operat.Systems used \_\_\_\_\_

Your preferred Operat.System \_\_\_\_\_

Do you use X/windows \_\_\_\_\_

What kind of application  
are you developing \_\_\_\_\_

Is it for internal  
use or for distribution \_\_\_\_\_

The size of the  
applic (in KBytes) \_\_\_\_\_

Expected number of  
simultaneous users \_\_\_\_\_

Do you use the application  
also on MS-DOS (or other OS) \_\_\_\_\_

Do you use additional  
tools/libs, which ones? \_\_\_\_\_

You are profi/hobby pro-  
grammer/consultant etc. \_\_\_\_\_

The line of business and  
size of your company (empl) \_\_\_\_\_

Forward this form by e-mail to:

-----	~   )	-----
multisoft Datentechnik	~ )  ))	Telephone : (+49-89) 649-0040
Harthauser Str. 85	))  )))	Fax : (+49-89) 641-2974
81545 München	=====	E-mail : support@flagship.de
Germany	\ /	Internet : www.fship.com
-----	~~~~~	-----

# Support Request Form

Please use this form when contacting the tech support department by e-mail or fax. Note that only registered users can be supported. You will find this form in text format also in the <FlagShip\_dir>/docu/suppreq.txt file.

FlagShip SUPPORT REQUEST (and SUGGESTIONS) My ref.no.:

```

Licensee      (Note 1, name, company) ...:
              address .....:
e-mail address .....:
Purchase Date (Note 1) .....:

```

```
FlagShip ser.No. (Note 2, e.g. abcd...98765):
FlagShip Release (Note 2, e.g. 4.32.1234)...:
License-Users (Note 2, e.g. Demo|unlim|4):
```

```

Oper.System used (Note 3, e.g. SCO 3.2.42)...
cc compiler rel. (Note 4, e.g. mcc 3.2.0)...:
Hardware used (Note 3, e.g. 486DX50,16MB):
Environment used (console,terminal,X,ODTetc):
TERM environment (Note 5, e.g. FSansi) .....

```

```
>Note 1: Data according to the Registration Card. If not registered yet,
> fill in the registration and attach it. The registr. form is ava-
> ilable a) on your computer in <FlagShip_dir>/docu/REGCARD, or b)
> in the manual, or c) via http://www.fship.com/support.html
>Note 2: If possible, execute 'a.out -FSversion [your-params]' and press ^S
> to watch the output or redirect stdout + stderr to file. If the
> executable is not available, forward data from the distribution
> media (e.g. floppies) and the activation card.
>Note 3: forward the full OS release, check e.g. with uname -a, and details
> about your Hardware (computer, model, RAM, terminal type and model
> emulator etc.)
>Note 4: check it with e.g. cc -v or forward the date of /bin/cc
>Note 5: check it with e.g. #env or #echo $TERM $TERMINFO
>
>Our support is FREE for registered users within the initial period of SIX
>months (thereafter, an inexpensive support contract is available). The
>support department will QUICKLY respond, usually within 24 hours.
```

```
> check your request(s):
> ( ) FlagShip installation
> ( ) FS precompiler (phase 1)
> ( ) FS compiler      (phase 2)
> ( ) cc compiler error
> ( ) linker error
> ( ) execution of the applic, input/oupt, terminfo (specify TERM & HW)
> ( ) unix system, tuning etc.
```

- > ( ) FS language, OOP, APIs, RDDs
- > ( ) FS library, std.function
- > ( ) FS compatibility to ( )Clipper\_\_, ( ) VO\_\_, ( )Fox\_\_, ( )dBase\_\_
- > ( ) FS manual, docu
- > ( ) other

> Please describe your problem, question or suggestion:

> Did you check the FlagShip manual? ( )yes, sections (e.g.REL.2,LNG.2.3):

> Include a small, COMPILABLE and TESTED program example (up to approx.  
> 100 lines), if applicable:

> - Compile the example with options:

> - Execute the example with options:

> - To re-produce the problem the following command or key sequence  
> is required:

- > - If required, additional data (.dbf, .dbt, .mem etc.) are included in  
> the file (Note 5) :
- > ( ) put into the dir ftp://fship.com/pub/multisoft/incoming - named:
- > ( ) e-mailed (binary) named:
- > ( ) mailed on floppy (3.5" tar format)
- > ( ) mailed on floppy (3.5" DOS format)
- > ( ) mailed on QIC or DAT tape, tar format

>-----  
>Note 5: For binary files, use tar + gzip (.tgz) or tar + compress (.taz)  
> or Lharc (.lzh) or pkzip (.zip) format, or simle tar for floppies and  
> tape. Specify the file name and forward it (using bin protocol) to:  
>  
> FTP, PUT into dir: ftp://ftp.fship.com/pub/multisoft/incoming  
> e-mail: MIME binary attachment  
>-----

Forward this form by e-mail (or fax) to:

-----	~	)	-----
multisoft Datentechnik	~~ )	))	Telephone : (+49-89) 649-0040
Harthausen Str. 85	))	))	Fax : (+49-89) 641-2974
81545 München	=====	/	E-mail : support@flagship.de
Germany	\	/	Internet : www.fship.com
-----	~~~~~	-----	

# Index

---

<hr/> <b>A</b>		FoxPro	
Achoice( )		- compatibility ..... APP-18, 20	
- control keys ..... APP-7		<hr/> <b>I</b>	
- UDF status ..... APP-7		InKey( )	
<hr/> <b>C</b>		- return codes ..... APP-3	
Compatibility		<hr/> <b>L</b>	
- difference to dBase ..... APP-18		LastKey( )	
- difference to Fox ..... APP-18, 20		- return codes ..... APP-3	
<hr/> <b>D</b>		<hr/> <b>M</b>	
dBase		MemoEdit( )	
- compatibility ..... APP-18		- control keys ..... APP-5	
DbEdit( )		- UDF return ..... APP-5	
- control keys ..... APP-6		- UDF status ..... APP-5	
- UDF return ..... APP-6		<hr/> <b>R</b>	
- UDF status ..... APP-6		READ	
<hr/> <b>E</b>		- control keys ..... APP-4	
Error		RTE	
- code and cause ..... APP-10		- error code and cause ..... APP-10	
<hr/> <b>F</b>		<hr/> <b>S</b>	
FlagShip		Support	
- registration form ..... APP-30		- request form ..... APP-32	
- support request ..... APP-32			
Foxbase			
- compatibility ..... APP-20			





---

**multisoft Datentechnik**  
**Harthäuser Str. 85**  
**D-81545 München**

**<http://www.fship.com>**  
**[sales@multisoft.de](mailto:sales@multisoft.de)**  
**[support@flagship.de](mailto:support@flagship.de)**