
Contents

CHAPTER 1	INTRODUCTION. WHAT IS EMU?	1-1
1.1	THE BASIC EMU PIPELINE.	1-1
1.2	ADDITIONAL EMU PIPELINE COMPONENTS.	1-3
1.3	CURRENT IMPLEMENTATION.	1-4
CHAPTER 2	GETTING STARTED WITH EMU	2-1
CHAPTER 3	CALLING THE EMU INJECTION ROUTINES FROM YOUR APPLICATION.	3-1
3.1	INJECTING A MESSAGE WITH ONE PARAMETER.	3-1
3.2	INJECTING A MESSAGE WITH ZERO PARAMETERS.	3-2
3.3	INJECTING A MESSAGE WITH SEVERAL PARAMETERS.	3-2
3.4	COMPILING AND LINKING YOUR PROGRAM.	3-3
3.5	INJECTING MESSAGES FROM A LIBRARY (EG. A PACKAGE OF FASTBUS ROUTINES)	3-3
3.6	ADVANCED ROUTINES FOR SOPHISTICATED USERS.	3-3
CHAPTER 4	PREPARING YOUR MESSAGE FILE.	4-1
4.1	DESCRIPTION	4-1

Contents

4.1.1	Declaring the name of your program. _____	4-3
4.1.2	Declaring the names of messages associated with your program. _____	4-3
4.1.3	Defining the text of a message _____	4-3
4.1.4	Associating properties with messages _____	4-4
4.1.5	Declaring the names of packages (libraries) _____	4-4
<hr/>		
4.2	SYNTAX CHECKING YOUR MESSAGE FILE.	4-4
<hr/>		
4.3	MAKING YOUR MESSAGE FILE KNOWN TO THE EMU SYSTEM.	4-4
<hr/>		
CHAPTER 5	PREPARING YOUR ROUTE MAP FILE	5-1
<hr/>		
5.1	DESTINATION DECLARATION STATEMENTS.	5-1
<hr/>		
5.2	MESSAGE SELECTION (ROUTING) STATEMENTS.	5-2
5.2.1	Format of a message selection expression. _____	5-2
5.2.2	Message selection keywords _____	5-4
<hr/>		
5.3	READABLE MESSAGE SELECTION EXPRESSIONS.	5-4
<hr/>		
5.4	SYNTAX CHECKING YOUR ROUTE MAP	5-4
<hr/>		
5.5	MAKING YOUR ROUTE MAP KNOWN TO THE EMU SYSTEM.	5-4
<hr/>		
CHAPTER 6	MESSAGE DESTINATIONS	6-1
<hr/>		
6.1	THE PREDEFINED DESTINATION PROGRAMS	6-1
6.1.1	Predefined "Console" destination program _____	6-1
6.1.2	Predefined "Logfile" destination program _____	6-2
<hr/>		
6.2	USER SUPPLIED DESTINATION PROGRAMS	6-2
6.2.1	Mailbox destination type _____	6-3
6.2.2	External destination type _____	6-3

6.2.3	How to write your own destination program _____	6-4
6.2.4	Route map declarations for your destination _____	6-4
6.2.5	Example destination program in Fortran _____	6-4
6.2.6	Linking and running your destination program _____	6-5
<hr/>		
6.3	INTERPRETATION OF "OPERATING_SYSTEM_FILESPEC" UNDER VMS	6-5
<hr/>		
CHAPTER 7	ACTIVATING YOUR MESSAGE FILE AND ROUTE MAP	7-1
<hr/>		
7.1	ACTIVATING MESSAGE FILE DECLARATIONS	7-1
<hr/>		
7.2	ACTIVATING YOUR ROUTE MAP DECLARATIONS	7-1
7.2.1	Dynamic route map update through <i>emu_lib</i> _____	7-1
<hr/>		
CHAPTER 8	EMU SYSTEM BEHAVIOUR - WHAT HAPPENS IF...	8-1
<hr/>		
8.1	...IF USER PROGRAMS DELIVER MESSAGES FASTER THAN EMU CAN PROCESS THEM?	8-1
<hr/>		
8.2	...IF AN OUTPUT DEVICE SEIZES UP?	8-2
<hr/>		
8.3	...IF A DESTINATION PROGRAM CRASHES?	8-2
8.3.1	Setting the <i>retry_interval</i> _____	8-2
<hr/>		
8.4	...IF YOU INJECT A MESSAGE THAT IS NOT DECLARED IN YOUR MESSAGE FILE?	8-3
<hr/>		
8.5	...IF EMU ITSELF CRASHES?	8-3
<hr/>		
8.6	...IF EMU IS NOT INSTALLED (OR IS NOT RUNNING) ON YOUR MACHINE?	8-4
<hr/>		
8.7	...IF EMU COMPLAINS IT IS LOSING MESSAGES?	8-4

Contents

8.8	...IF EMU COMPLAINS OF AN INTERNAL ERROR AND SAYS IT IS LOSING MESSAGES?	8-4
8.9	...IF YOU CALL EMU FROM AST LEVEL?	8-4
8.10	...IF YOU BURST ONE OF THE IMPLEMENTATION LIMITS (EG FOR A MAXIMUM MESSAGE LENGTH)?	8-4
8.11	...IF YOU THINK EMU IS DEAD? (HOW CAN YOU CHECK IT IS ALIVE?)	8-5
8.12	...IF MY OUTPUT LOGFILES ARE ALWAYS EMPTY?	8-5
8.13	...IF THERE IS SOMETHING ELSE YOU DON'T UNDERSTAND?	8-5
CHAPTER 9 IMPLEMENTATION LIMITS		9-1
9.1	TEMPORARY LIMITS AND MISSING FEATURES.	9-1
9.2	PERMANENT LIMITS.	9-1
9.2.1	Size limits	9-2
9.2.2	Philosophical limits.	9-6
9.3	BUGS	9-6
CHAPTER 10 EMU TOOLS		10-1
10.1	TOOLS AVAILABLE TO THE GENERAL USER.	10-1
10.1.1	EMUType	10-1
10.1.2	EMUWrap	10-1
10.1.3	EMUSearch	10-1
10.1.4	EMUMessage_file_compile	10-2
10.1.5	EMURoute_map_compile	10-2

CHAPTER 11	EMU INSTALLATION	11-1
<hr/>		
11.1	RESOURCE REQUIREMENTS FOR INSTALLING EMU.	11-1
<hr/>		
11.2	BACKGROUND TO THE INSTALLATION.	11-1
11.2.1	Questions asked by VMSINSTAL.	11-1
11.2.1.1	Emu_uic • 11-1	
11.2.1.2	Emu_directory • 11-2	
11.2.2	After VMSINSTAL has finished.	11-2
11.2.2.1	SY\$HELP:EMUvvu.RELEASE_NOTES • 11-2	
11.2.2.2	Known Bugs • 11-2	
11.2.2.3	Emu_config • 11-2	
11.2.2.4	Emu_root • 11-3	
11.2.2.5	Lnmodel_table • 11-3	
11.2.2.6	Slimline Emu • 11-3	
11.2.2.7	Reinstallation of Emu. • 11-4	
<hr/>		
11.3	GETTING THE DISTRIBUTION SAVE SET.	11-4
11.3.1	Getting the Distribution Save Set from Magnetic Tape.	11-4
11.3.2	Getting the Distribution Save Set over the network.	11-4
<hr/>		
11.4	INSTALLATION STEPS.	11-4
<hr/>		
11.5	RUNNING EMU FOR THE FIRST TIME ON YOUR MACHINE.	11-6
11.5.1	Starting the Emu detached processes.	11-6
11.5.1.1	Resources required. • 11-6	
11.5.1.2	Starting Emu. • 11-7	
11.5.1.3	What happens when Emu starts to run. • 11-7	
11.5.2	Running the demonstration programs.	11-8
11.5.2.1	Symbol availability. • 11-8	
11.5.2.2	The demonstration files. • 11-8	
11.5.2.2.1	What to expect from run_demo.com • 11-8	
<hr/>		
11.6	BEYOND THE DEMONSTRATION - CONFIGURING EMU FOR YOUR PARTICULAR VAX	11-9
11.6.1	Emu_panic_device	11-9
11.6.2	Emu_panic_bell_string	11-10
11.6.3	Emu_tell_cern	11-10
11.6.4	The output directories Emu_log, Emu_dat, Emu_out, Emu_err	11-10
11.6.5	Emu_decoder_message_file	11-11

Contents

11.6.6	Emu_route_map _____	11-11
11.6.7	Emu_user_startup _____	11-11
11.6.8	Restart_after_decoder_crash _____	11-11
11.6.9	Restart_after_router_crash _____	11-12
11.6.10	Emu_decnet_server_startup _____	11-12
<hr/>		
11.7	RUNNING YOUR OWN PROGRAMS USING EMU - A CHECKLIST.	11-12
<hr/>		
11.8	ANY PROBLEMS?	11-13
<hr/>		
11.9	INSTALLING BABY EMU	11-13
11.9.1	Linking on a machine with Baby Emu _____	11-14
<hr/>		
11.10	SENDING MESSAGES TO VAX-EMU FROM A VALET	11-14
<hr/>		
11.11	OS-9 EMU	11-15
<hr/>		
CHAPTER 12	RESPONSIBILITIES OF THE EMU MANAGER	12-1
<hr/>		
12.1	ACTIVATING A NEW MESSAGE FILE	12-1
<hr/>		
12.2	ACTIVATING OR REPLACING AN ENTIRE ROUTE MAP	12-1
<hr/>		
APPENDIX A	USER CALLABLE ROUTINES IN EMU_LIB	A-1
	EMU	A-2
	EMU_CLOSE_DESTINATION	A-4
	EMU_CLOSE_DOWN	A-6
	EMU_COMMENT	A-7
	EMU_CONNECT_TO_ROUTER	A-8
	EMU_CREATE_DESTINATION	A-10
	EMU_DEFINE	A-12
	EMU_ERASE_ALL_PARAMS	A-14
	EMU_EXPEDITE	A-15
	EMU_EXTRACT_DATE	A-16
	EMU_EXTRACT_INSTANCE_NAME	A-18
	EMU_EXTRACT_MACHINE_NAME	A-20
	EMU_EXTRACT_MESSAGE_NAME	A-22
	EMU_EXTRACT_N_PROPERTIES	A-25
	EMU_EXTRACT_PACKAGE_NAME	A-27

EMU_EXTRACT_PROGRAM_NAME	A-29
EMU_EXTRACT_PROPERTY	A-31
EMU_EXTRACT_ROUTERS_VISITED	A-34
EMU_EXTRACT_TEXT	A-36
EMU_EXTRACT_TIME	A-38
EMU_F_TRNLNM	A-40
EMU_GET_INSTANCE_NAME	A-42
EMU_GET_MACHINE_NAME	A-44
EMU_GET_PROGRAM_NAME	A-46
EMU_INIT	A-48
EMU_INT	A-49
EMU_INT_PACKAGE	A-51
EMU_OPEN_DESTINATION	A-53
EMU_PACKAGE	A-55
EMU_REAL	A-57
EMU_REAL_PACKAGE	A-59
EMU_SET_INSTANCE_NAME	A-61
EMU_SET_MACHINE_NAME	A-62
EMU_SET_PARAM_STR	A-63
EMU_SET_PROGRAM_NAME	A-65
EMU_STR	A-66
EMU_STR_PACKAGE	A-68
EMU_TRNLNM_ITER	A-70
EMU_UNPACK_RECORD	A-72
VMS_TIME_TO_EMU_TIME	A-74

APPENDIX B THE EMU DEMONSTRATION PROGRAMS - WORKING EXAMPLES **B-1**

B.1	FORTRAN SOURCE OF PROGRAMS INJECTING MESSAGES	B-1
B.2	COMPILING, LINKING AND RUNNING APPLICATION PROGRAMS	B-1
B.3	THE MESSAGE FILE AND ROUTE MAP	B-1
B.4	DYNAMIC ROUTE MAP UPDATE	B-2
B.5	YOUR OWN DESTINATION PROGRAMS	B-5

Contents

APPENDIX C	LOGICAL NAMES UNDER VAX-VMS	C-1
-------------------	------------------------------------	------------

APPENDIX D	MESSAGE FILE FORMAT - REFERENCE INFORMATION	D-1
-------------------	--	------------

D.1	MESSAGE FILE SYNTAX	D-1
-----	---------------------	-----

D.2	MESSAGE FILE PARSING RULES (SUMMARY)	D-2
-----	--------------------------------------	-----

APPENDIX E	TOOLS AVAILABLE TO THE EMU MANAGER	E-1
-------------------	---	------------

E.1	DRAIN	E-1
-----	-------	-----

E.2	EMUCREMBX	E-1
-----	-----------	-----

E.3	EMUDELMBX	E-2
-----	-----------	-----

E.4	EMUFORCEEXIT	E-2
-----	--------------	-----

E.5	EMULOAD	E-3
-----	---------	-----

E.6	PEEK_MBX	E-3
-----	----------	-----

E.7	START_EMU, STOP_EMU	E-4
-----	---------------------	-----

E.8	START_DECODER, STOP_DECODER	E-4
E.8.1	Start_router, Stop_router _____	E-4

E.9	SHOW_EMU	E-4
-----	----------	-----

E.10	ONEMESSAGE	E-4
------	------------	-----

E.11	MODEL_VERSIONS	E-4
------	----------------	-----

E.12	PRINT ACTIVE ROUTE MAP	E-5
------	------------------------	-----

APPENDIX F CLEVER THINGS PEOPLE DO **F-1**

F.1	MYTTY	F-1
-----	-------	-----

F.2	SEND MESSAGES TO SYS\$OUTPUT	F-1
-----	------------------------------	-----

F.3	MORE TRICKS	F-1
-----	-------------	-----

APPENDIX G SILLY THINGS PEOPLE DO **G-1**

G.1	LEAVE A DESTINATION RETRYING FOR EVER	G-1
-----	---------------------------------------	-----

G.2	OTHER SILLY THINGS	G-1
-----	--------------------	-----

APPENDIX H REFERENCES **H-1**

GLOSSARY **Glossary-1**

INDEX

EXAMPLES

6-1	Mailbox destination _____	6-3
6-2	Route map with a MAILBOX declaration _____	6-4
8-1	Changing the open_destination re-try interval _____	8-3
8-2	Changing the generic open_destination re-try interval _____	8-3
12-1	Concatenating and loading many small .mess files _____	12-1
12-2	Viewing the current message file name _____	12-1
12-3	Concatenating and Loading many small .route_map files _____	12-2
12-4	Reloading an entire route map _____	12-2
12-5	Establishing the name of the current route map file _____	12-2
B-1	Create_destination.for _____	B-3

Contents

B-2	Example destination program in Fortran _____	B-5
-----	--	-----

FIGURES

1-1	The Emu Pipeline _____	1-2
3-1	Injecting a Message with One Parameter _____	3-1
3-2	Injecting a Message with Several Parameters _____	3-2
4-1	Example Message File _____	4-2
5-1	A Route Map File Showing Message Selection Expressions _____	5-3
6-1	Example console output _____	6-2
6-2	Example logfile record. _____	6-2
9-1	Listing of the implementation_limits program. _____	9-2

TABLES

5-1	Valid Destination Types in a Route Map _____	5-1
A-1	Routines available in EMU_LIB _____	A-77