

- 14 TEST ENVIRONMENT14-2**
- 14.1 THE ACPT ENVIRONMENT 14-2
- 14.2 COMMUNICATION VIA FILE TRANSFER TO THE ACPT ENVIRONMENT 14-2
 - 14.2.1 *FTP* 14-2
 - 14.2.2 *NetviewFTP*..... 14-3
- 14.2 COMMUNICATION VIA MQ TO THE ACPT ENVIRONMENT 14-3
 - 14.2.1 *Protocol LU6.2*..... 14-3
 - 14.2.2 *Protocol TCP/IP* 14-3
 - 14.2.3 *Local receiving queue at VPC*..... 14-3
 - 14.2.4 *Remote queues at VPC*..... 14-4
 - 14.2.5 *Values in the MQMD header* 14-4
 - 14.2.6 *Values in the MQIH header*..... 14-5
 - 14.2.7 *Diagram - overview* 14-6

14 TEST ENVIRONMENT

14.1 The Acpt environment

The Acpt environment is used for the education and training of users, and for the testing of computer-to-computer functions. It is therefore in the A environment that you test modified and new computer-to-computer functions from your home ground.

Those persons at an AO who work with the development and testing of computer-to-computer functions or with other connections to the VPC system must be authorised as users in the A environment.

The Acpt environment comprises all functions in production and, for some time before a production start-up, even new or modified functions to be put into operation.

In the Acpt environment, the VPC system is run in the same way as in production, which means, amongst other things, that the registration of instructions and so on are processed and reported according to the processing schedule.

As a terminal user in the Acpt environment, you have access to all AO transactions that are included in the VPC system. This means, for example, the ability to:

- get help instructions for transactions and term descriptions via transactions that are part of the User Documentation system
- create test data and review the results of tests. The registration of instructions, for example, can be made and then retrieved via the computer-to-computer functions. If you test registrations computer-to-computer, enquiry transactions in the VPC system can be used to follow up the registrations, both individual entries and different summary levels
- order files and reports and retrieve files/reports via transactions
- allow tests of your own new or adapted routines to include that registrations be made in the VPC system and processed during a complete settlement timetable.

The description of routines is now available on VPC's homepage, www.vpc.se.

14.2 Communication via file transfer to the Acpt environment

There are two different ways of transferring files to VPC: FTP (IP) and NetviewFTP (SNA).

14.2.1 FTP

VPC places the following demands on clients to enable them to connect up to VPC's FTP server:

- that the FTP client connects from an IP address authorised by VPC
- that the FTP client must be able to perform in the passive mode

VPC's FTP server

IP address: 194.132.134.81 (Acpt)

Port no: 5021

Protocol: IP

The client receives a user ID from VPC and a password for logging on to the FTP server. After logging on, the client is connected up to its own home library, where the file is placed according to the following format:

File name.A

File name = no restrictions on appearance

A = environment code for Acpt

The code page for file transfer via VPC's FTP server must be ISO88591 in accordance with ANSI standard. This applies both inbound to VPC and outbound from VPC.

For further information, please contact VPC's network group.

14.2.2 NetviewFTP

VPC places the following demands on clients to enable them to connect up to VPC's NFTP server:

- that the LU name used by the client must be recognised in the VPC system.

VPC's NFTP server

LU name: AVPCFTAR (receiving)

LU name: AVPCFTAS (sending)

Protocol: LU6.2

For details regarding the appearance of the file name, etc. please refer to chapter 3.

For further information, please contact VPC's network group.

14.2 Communication via MQ to the Acpt environment

14.2.1 Protocol LU6.2

QMGR name: VPQA

VPC sets up the following channels in the acceptance environment for each AO connected to MQ

Channel name: KImgr.VPQA.1 RECEIVER
 VPQA.KImgr.1 SENDER
 KImgr = Name of QMGR at AO.

LU name: AS1VPQA

TP name: VPQA

14.2.2 Protocol TCP/IP

QMGR name: VPQX

VPC sets up the following channels in the acceptance environment for each AO connected to MQ

Channel name: KImgr.VPQX.1 RECEIVER
 VPQX.KImgr.1 SENDER
 KImgr = Name of QMGR at AO.

IP address: VPQX = 194.132.134.81

14.2.3 Local receiving queue at VPC

VPC sets up the following channels in the acceptance environment for each AO connected to MQ. The AO distributes its transactions as it wishes on these queues.

Queue name: aaaM001A for real time
 aaaM002A for computer-to-computer
 aaa = AO identity

The "real time" queue is intended for transactions of an online type, where one person is waiting for a reply.

The "computer-to-computer" queue is intended for transactions of batch type, such as the transmission of a large number of transactions.

Both queues are in fact processed in real time, though the difference lies in the speed at which these transactions are handled. This enables us to distribute the resources in our system more effectively to those transactions of higher priority. The transactions in the real time queue are given the same status as 3270 transactions. Transactions to the computer-to-computer queue are processed in a separate queue at a slower pace. The reason for this is to achieve a shorter queueing time for real time transactions.

14.2.4 Remote queues at VPC

For those services that generate data to an AO, VPC sets up remote queues based on the incoming requests from AOs. This does not, however, apply to enquiry-reply from an AO. VPC can set up only one remote queue at most per service that the AO subscribes to. It is up to the AO to choose whether more than one service is to use the same remote queue or not.

14.2.5 Values in the MQMD header

Field name	Where customer creates the trans and VPC replies	Where VPC creates the trans
StrucId	MQMD-STRUC-ID	MQMD-STRUC-ID
Version	MQMD-VERSION-1	MQMD-VERSION-1
Report	According to choice. Steers reportmsg, processing of dead-q and how msgid and correlid are to be handled when VPC replies. MQRO-NEW-MSG-ID MQRO-PASS-MSG-ID MQRO-COPY-MSG-ID-TO-CORREL-ID MQRO-PASS-CORREL-ID MQRO-DEAD-LETTER-Q MQRO-DISCARD-MSG	MQRO-NONE
MsgType	MQMT-REQUEST VPC always gives an answer.	MQMT-DATAGRAM VPC does not require an answer.
Expiry	MQEI-UNLIMITED. If a limited duration is desired for the message, which is expressed in 1/10's second.	MQEI-UNLIMITED
Feedback	MQFB-NONE	MQFB-NONE
Encoding	MQENC-NATIVE	MQENC-NATIVE
CodedCharSetId	MQCCSI-Q-MGR	MQCCSI-Q-MGR
Format	MQFMT-STRING (VPC itself creates if necessary an IIH-header on the incoming trans) If the customer specifies an IIH header, this field must be filled in with MQFMT-IMS.	MQFMT-STRING or MQFMT-IMS
Priority	MQPRI-PRIORITY-AS-Q-DEF	MQPRI-PRIORITY-AS-Q-DEF
Persistence	MQPER-PERSISTENT MQPER-NOT-PERSISTENT if logging of msg is not desired. Default on the receiving queue at VPC is always PERSISTENT.	MQPER-PERSISTENT
MsgId	According to choice. Preferably MQMI-NONE so that a unique id is generated by MQ.	MQMI-NONE
CorrelId	According to choice. VPC replies according to the Report field	MQCI-NONE
BackoutCount	0	0
ReplyToQ	Name of local queue to which the reply from VPC is to be sent.	Blank
ReplyToQmgr	Own qmgrnamn	Blank

Field name	Where customer creates the trans and VPC replies	Where VPC creates the trans
UserIdentifier	Valid userid at VPC. The same userid is returned in the reply. Another userid can be obtained in the reply if the field <i>ID-TERMINAL-USER</i> in the transaction header in the incoming transaction is set at a non-blank value. Then that userid will be returned instead in the reply.	Valid userid at the customer, at customer's own choice.
AccountingToken	Not used by VPC.	Generated by MQ
AppIdentityData	Reserved for VPC.	Blank
PutAppType	Not used by VPC.	Generated by MQ
PutAppName	Not used by VPC.	Generated by MQ
PutDate	Should be set by MQ.	Set by MQ.
PutTime	Should be set by MQ.	Set by MQ.
AppOriginData	Not used by VPC.	Generated by MQ

14.2.6 Values in the MQIIH header

It is not necessary for the customer to create this header on messages to VPC. VPC will generate the header as required if it is missing. If the customer uses a header on messages to VPC, it should look like the examples in the table below. VPC will consequently let the header remain in place in its reply to the customer.

In those cases where the customer requests an MQIIH header on the messages created at VPC, VPC will create a header like those shown below.

Field name	Value
Strucid	MQIIH-STRUC-ID
Version	MQIIH-VERSION-1
Struclength	MQIIH-LENGTH-1
Encoding	MQENC-NATIVE
Codedcharsetid	MQCCSI-Q-MGR
Format	MQFMT-IMS-VAR-STRING
Flags	MQIIH-NONE
Ltermoverride	space
MFSmapname	space
Replytoformat	MQFMT-IMS-VAR-STRING
Authenticator	MQIAUT-NONE
Traninstanceid	MQITII-NONE
Transtate	MQITS-NOT-IN-CONVERSATION
Commitmode	MQICM-COMMIT-THEN-SEND
Securityscope	MQISS-CHECK
Reserved	space

14.2.7 Diagram - overview

