

## Defining an LU 6.2 connection

MQSeries for Digital OpenVMS uses the DECnet SNA APPC/LU 6.2 Programming Interface. This interface requires access through DECnet to a suitably configured SNA Gateway, for example, the SNA Gateway-ST, or SNA Gateway-CT.

### SNA configuration

To enable MQSeries to work with DECnet APPC/LU 6.2 you **must** complete your Gateway SNA configuration first. The Digital SNA configuration **must** be in agreement with the Host SNA configuration.

#### Notes:

1. When configuring your host system, be aware that the DECnet SNA Gateway supports PU 2.0 and **not** node type 2.1. This means that the LUs on the Digital SNA node must be dependent LUs. They reside on the Digital SNA node and so must be defined and configured there. However, because they are dependent LUs, they have to be activated by VTAM, by means of an ACTLU command, and so they also need to be defined to VTAM as dependent LUs.
2. Ensure that the SNA libraries are installed as shared images upon each system IPL by running the command @SYS\$STARTUP:SNALU62\$STARTUP.COM in the system startup procedure.

To configure your SNA Gateway, set up the SNAGATEWAY\_<node>\_SNA.COM file,

where <node> is replaced with the node name of your DECnet SNA gateway.

Do this by responding to the configuration prompts in the Gateway installation procedure, or by directly editing the file.

The SNA Gateway installation procedure creates the file in the directory SYS\$COMMON:[SNA\$CSV].

The configuration information in this file is downloaded to the Gateway when you run the NCP LOAD NODE command.

#### Notes:

1. Online changes to the current Gateway configuration can be made using the utility SNANCP.
2. SNA resources can be monitored using the SNAP utility.

A sample SNA Gateway Configuration file follows:

```
$!+-----+
$! Start of file: SYS$COMMON:[SNA$CSV]SNAGATEWAY_SNAGWY_SNA.COM
$! DECnet SNA Gateway-ST SNA configuration file
$! Created: 23-FEB-1996 19:10:43.68 by SNACST$CONFIGURE V1.2
$! Host node: CREAMP User$ CHO
$!+-----+
$ v = f$verify(1)
$ RUN SYS$SYSTEM:SNANCP
```

```
SET LINE SYN-0 - // Line definition
  DUPLEX FULL -
  PROTOCOL SDLC POINT -
  SIGNALLING NORMAL -
  CLOCK EXTERNAL -
  MODEM TYPE NORMAL -
  RECEIVE BUFFERS 34 -
  LOGGING INFORMATIONAL -
  BUFFER SIZE 265
SET CIRCUIT SDLC-0 - // Circuit definition
  LINE SYN-0 -
  DUPLEX FULL -
  RESPONSE MODE NORMAL -
  STATION ADDRESS C1 -
  LOGGING INFORMATIONAL -
  STATION ID 0714002A // XID
SET PU SNA-0 CIRCUIT SDLC-0 -
  LU LIST 1-32 -
  SEGMENT SIZE 265 - // must equal MAXDATA on Host PU definition
  LOGGING WARNING
SET CIRCUIT SDLC-0 STATE ON
SET LINE SYN-0 STATE ON
SET SERVER SNA-ACCESS -
  LOGGING WARNING -
  NOTE "Gateway Access Server" -
  STATE ON
SET ACCESS NAME VTAMSDR PU SNA-0 LU 2 APPL MVSLU LOGON LU62SS
SET ACCESS NAME VTAMRCVR PU SNA-0 LU 3 APPL MVSLU LOGON LU62SS
$ EXIT $STATUS + (0 * 'f$verify(v)')
$!+-----+
$! End of file: SYS$COMMON:[SNA$CSV]SNAGATEWAY_SNAGWY_SNA.COM
$!+-----+
```

### Defining access names

You should set up a separate Access name for each MQSeries channel. This ensures that the VMS system and the remote system agree on the LU used for the channel.

**Note:** If you use a single access name, with a range of LUs specified, the Gateway selects the LUs in a circular order. Therefore the LU selected by the Gateway may not correspond with the LU used by the Host channel, because the Host associates a specific LU with a channel.

The access name is used only to communicate between the DECnet SNA APPC program and the Gateway. It has no network meaning.

#### Notes:

1. The LUs are single session. You **must** define a separate LU for each channel if they are to run simultaneously.
2. You are advised to use names that associate the access name to the corresponding channel, but you can choose any name.
3. The APPL in the ACCESS name definition must match the remote (in this case MVSLU) APPL in VTAM.
4. The LU number must correspond to the LOCADDR in the LU definition statement in VTAM. Here is an example VTAM line and LU definitions:

```

IYA8L007 LINE ADDRESS=(007,FULL),
           ISTATUS=ACTIVE
IYA8P307 PU ADDR=C2,
           ISTATUS=ACTIVE,
           IRETRY=NO,
           MAXDATA=521,
           MAXOUT=7,
           PASSLIM=7,
           PUTYPE=2
IYA83071 LU LOCADDR=2,PACING=1,DLOGMOD=LU62CP1
IYA83072 LU LOCADDR=3

```

5. The LOGON must specify the logmode entry on the VTAM host that specifies parameters acceptable to the SNA Gateway. Here is an example of a single session logmode entry:

```

LU62SS  MODEENT LOGMODE=LU62SS,
        TYPE=0,                ONLY TYPE RECOGNIZED
        FMPPROF=X'13',         SNA
        TSPROF=X'07',         SNA
        PRIPROT=X'B0',        PRIMARY PROTOCOL
        SECPROT=X'B0',        SECONDARY PROTOCOL
        COMPROT=X'50B1',      COMMON PROTOCOL
        SSNDPAC=X'00',
        SRCVPAC=X'00',
        RUSIZES=X'8989',      RUSIZES  IN-4096 OUT-4096
        PSNDPAC=X'00',
        PSERVIC=X'06020000000000000002C00',

```

The *DECnet SNA Gateway Guide to IBM Parameters* details the parameters expected by the Gateway.

### Specifying SNA configuration parameters to MQSeries

MQSeries obtains knowledge of the SNA resources by passing the Gateway Node name and the Access name to the channel program.

#### Passing parameters to sender and requester channel pairs

For sender and requester channel pairs specify the Gateway Node and Access Name in the CONNAME string in the channel definition.

The CONNAME also includes the TPNAME that is used by the SNA Allocate verb to invoke the remote program.

The format of the CONNAME is: CONNAME('GatewayNode.AccessName(TpName)').

For example: CONNAME('SNAGWY.VTAMSDR(MQSERIES)'), where SNAGWY is the Gateway node, VTAMSDR is the access name, and MQSeries is the TPNAME.

**Note:** Do not use the TPNAME field in the channel definition.

#### Running senders and requesters

Senders, requesters, and fully qualified servers can be explicitly run by performing a START CHANNEL command in runmqsc.

Senders and requesters on Digital OpenVMS initiate a session by issuing an INIT-SELF to request a

BIND from the host. In issuing the Allocate verb, the MQSeries channel program takes the LU name and the Mode Name from the Access Name.

MQSeries then allocates a conversation using the specified TPNAME.

#### Passing parameters to servers and receivers

For servers and receivers, specify the Gateway Node, Access Name, and TPNAME as command line parameters to the **runmqslsr** command.

#### Running servers and receivers

Servers and receivers are started by running the runmqslsr command.

```
$ RUNMQLSR -m QMname -n TPname -g GatewayNode (AccessName)
```

**Note:** Each server and receiver channel requires its own listener process.

You can include these commands in the MQSeries startup file, SYS\$STARTUP:MQS\_STARTUP.

Receivers and servers issue the ACTIVATE\_SESSION request to the Gateway in passive mode. In passive mode the channel program waits for a BIND from the remote system, which puts the LU into the active-listening state, waiting for a bind from the host.

You can check the LU status using SNANCP to make sure that you are in active-listening state on the correct LU. If a BIND from the host arrives specifying the LU that is in active-listening state, the session will be established. After establishing the session, the host attempts to allocate a conversation.

The TPNAME used by the host sender/requester channel **must** be the same name as that specified on the command line in order to establish the conversation.

**Note:** RUNMQLSR recycles when a remote channel disconnects. There is a finite period of time before the listener is ready to accept further binds from the host.

#### Ending the SNA Listener process

To find the batch job number for the SNA listener process, type: \$ show queue / all

To end the SNA Listener process type:

```
$ delete /entry=<jobnumber>
```

where <jobnumber> is the job number of the listener batch job.

#### Sample MQSeries configuration

```

*
*      channel configuration for saturn.queue.manager for LU6.2
*
def ql('HOST_SENDER_TQ') usage(xmitq)

def chl('HOST.TO.VMS') chltype(rcvr) trptype(lu62) +
  seqwrap(999999999)

```

```
def chl('VMS.TO.HOST') chltype(sdr) trptype(lu62) +
  conname('SNAGWY.VTAMSDR(MQSERIES)') +
  xmitq('HOST_SENDR_TQ') seqwrap(999999999)
```

In this example two channels, a sender and a receiver, have been set up.

On the remote system you need to configure the corresponding channels. Channels that talk to each other must have the same name.

- The OpenVMS sender, VMS.TO.HOST, talks to a receiver called VMS.TO.HOST on the host system.
- The OpenVMS receiver, HOST.TO.VMS talks to a sender HOST.TO.VMS on the host system.

The commands to start each channel are:

```
// Start sender channel to host system
$ runmqchl -m "saturn.queue.manager" -c "VMS.TO.HOST"
// Set up listener to listen for incoming SNA requests.
$ runmqlsr -m "saturn.queue.manager" -n "TPNAME" -g SNAGWY (VTAMRCVR)
```

**Note:** The TPNAME must match the outbound TPNAME on the MVS sender channel side. This is specified in the MVS side information, for example:

```
SIDELETE
  DESTNAME (ID1)
SIADD
  DESTNAME (ID1)
  MODENAME (LU62SS)
  TPNAME (MQSERIES)
  PARTNER_LU (IYA83072)
```

## Problem solving

### Error PUNOTAVA - PU has not been activated

This error indicates a lack of connectivity between the two machines. Make sure your line and circuit are set to state ON. Use SNATRACE at the circuit level to verify that the Digital OpenVMS machine is polling. If no response is received for the poll, check that the PU on the host is enabled. If the line will not go to the ON STATE check your physical line. If the trace shows the host responding to the poll, but the PU still does not become active, check your setting of the STATION ID.

### Failure to allocate conversation

This error is returned by a sender or requester to indicate that allocate failed. Run trace to verify that the session can be established. Verify that the Digital OpenVMS machine sends the INIT-SELF (010681). If there is no response to the INIT-SELF make sure that the host MQSeries channel is started. If the BIND from the host is rejected by the Digital OpenVMS machine analyze the Digital bind response. Use the *DECnet SNA Gateway Guide to IBM Parameters* to see what is set incorrectly in the mode. If a session is established and the conversation allocate request is rejected verify that the TPNAMEs are configured the same on both systems.

For receivers and servers verify that a BIND is sent by the host. If not, enable the Host MQSeries channel. If the BIND is rejected check the reason for rejection. Make sure that the Digital

OpenVMS listener LU is the LU with which the host is trying to establish a session.

### MQSeries connection failure

After establishing a conversation the two MQSeries channels engage in a protocol to establish an MQSeries channel connection. If this fails, the reason for failure should be indicated in the error logs on the two systems. Check both logs and correct the indicated problem. For example the connection fails if one system has a SEQWRAP value of 999999999 and the other 999999. In the SNATRACE you will see that the allocate succeeded and that MQ is trying to establish a channel connection. At this point the MQSeries logs are the best aid in resolving problems.

© IBM Corporation 2000. All Rights Reserved