ABS Tech Note: ARA02 Error Correction & Modem Scripts (11/94)

Article Created: 14 November 1994

TOPIC --

This document discusses error control and data compression when using Apple
Remote Access (ARA). Specifically, it details ARA's use with modems that support
error control using CCITT V.42 as well as data compression using CCITT V.42bis.
Regardless of which error control and/or compression protocols your modem
supports, please refer to the manufacturer's documentation for more information
on their availability and use.

DISCUSSION ---

Why use error control or data compression?

Numerous modem manufacturers are now supplying error control and/or data
compression as a method of assuring accurate and efficient communication between
a pair of modems, even under less than optimal line conditions. Typically, these
options are enabled via Hayes-compatible "AT" commands prior to dialing a number
or readying a modem to answer calls and are employed whenever the overlying
application does not support error correction and/or data compression.

ARA's use of error control and data compression

ARA implements V.42 error correction (MNP 4) from its' Client application to
its' Server application, as the Macintosh Serial Port is judged to be
unreliable. ARA implements V.42bis data compression (LZW) from its' Client
application to its' Server application. Thus it is not necessary to enable these
features on the modem (except for cellular data modems, which are discussed
below). Modem scripts, as distributed by Apple and by cooperating modem vendors,
should specifically disable these features to ensure the best possible
performance. ARA data compression is more effective than what is implemented in
modems, simply due to greater computational power available.

If you plan to write or customize a modem script, it is recommended that you do
not use the error correction or data compression capabilities of your modem
without testing it with varying line quality, file type, and file size to see if
the desired performance improvement has been achieved.

And what if data compression and/or error correction are used with ARA?

Some customers, with exceptionally noisy phone lines, have reported that

enabling the primary V.42 error correction standard (LAP-M) from modem to modem,
has given them improved connection reliability. Note that ARA still does MNP 4
error correction end to end, due to the unreliability of the Macintosh Serial
Port. If you have this need, please consult your modem manual carefully, and
note the use of USERHOOK 2 in the Modem Scripting Guide. This call alerts ARA
that the modems have successfully negotiated a V.42 link, and ARA should relax
its' re-transmit timers. Failure to call USERHOOK 2 could result in unnecessary
timeouts, re-transmissions, and the connection being dropped. There is no method
for turning off ARA V.42.

Some customers have expressed a desire to implement V.42bis data compression (or
perhaps MNP 5) from modem to modem. Generally this hurts throughput, but if you
have this desire, please consult your modem manual carefully, and note the use
of USERHOOK 3 in the Modem Scripting Guide. This call alerts ARA that the modems
have successfully negotiated a data compression link, and ARA should turn off
it's own data compression. Failure to call USERHOOK 3 could result in
performance losses, as ARA will be trying to compress already compressed data.

What's with the 'mlts' resource and varString 4?

During the development of ARA 2.0, we included the infrastructure for requesting
that modems bring up a error correction link. VarString 4 was selected as the
variable that would be set to indicate what type of error correction link was
requested. The 'mlts' resource was selected as the way that the modem script
writer could inform the CCL engine, that the modem had the capability to perform
error correction. VarString 4 is controlled by an algorithm in the CCL engine.
In future versions of ARA, the CCL engine may choose which modem script to use,
based on values in the 'mlts' resource. Currently for the Personal Client/Server
products, varString 4 = 0, unless the MNP 10 box is checked, in which case
varString 4 = 2. Currently for the MultiPort Server product, varString 4 = 0.
Setting 'mlts' byte 1 = 1 does not mandate that varString 4 be passed to the
script with a value of 1. We recommend that modem script writers include 'mlts'
resources and correct modem commands for varString 4 = 1, for future
compatibility.

Using cellular modems with ARA

If an ARA connection is desired using cellular modems, you should enable the
MNP10 capability in the modems. This is performed differently depending on
whether you are configuring for ARA individually (Personal Server or Client) or
for use with the MultiPort Server.

• For the Personal Server or Client, open the 'Remote Access Setup'
  Control Panel document and check the box named 'Use MNP 10 error
  correction in modem'. This will tell ARA to establish a connection
  using MNP10 (provided that modems that support MNP10 are being used).
  You must use a modem script that has been properly configured to support
  both MNP10 and non-MNP10 modems.

• For the MultiPort Server, a check box is not used. Instead, you should
  configure the script to accept calls from both MNP10 and non-MNP10
  modems.

```
Creating modem scripts for use with MNP10
-----------------------------------------
If you need to create or modify a modem script for use with MNP10, ensure that
you perform the following:

• Create an 'mlts' resource using ResEdit with byte 1 = 0x01

• Check if varString 4 is equal to '2'. If so, jump to the label that
  programs the modem to perform MNP-10.

• Issue a 'USERHOOK 4' for a successful MNP10 connection so that anything
  that polls the modem will know that MNP-10 is enabled. ARA data
  correction can not be disabled.

Here are some excerpts from an example CCL detailing the use of MNP10 with ARA.
Note that this is not a complete CCL; only the relevant parts are shown for
clarity.

  !  'mlts' resource info for this modem:
  !    byte 1 == 01 -> modem HAS builtin reliability protocols
  !    byte 2 == 00 -> reserved by Apple
  !    byte 3 == 43 -> max hex chars in varstr 7 (72 dec)
  !    byte 4 == 43 -> max hex chars in varstr 8
  !    byte 5 == 43 -> max hex chars in varstr 9
  ...
  ! If modem mnp10 link requested (var 4 == 2) then jump label 4
  ifstr 4 4 "2"
  ...
  @LABEL 4
  ! Use this label for mnp 10 type stuff...
  !
  matchclr
  matchstr 1 7 "OK\13\10"
  ! \N2 = MNP reliable mode
  ! )M1 = force adjust cellular power level
  ! *H1 = link negotiation occurs at 1200
  ! @M18 = use -18dBm xmit level
  ! -K1 = turn on LAPM to MNP10 upshift (extended services)
  ! :E0 = turn off equalization
  write "AT\\N2)M1*H1@M18-K1:E0\13"
  matchread 300
  ...
  @LABEL 32
  matchclr
  matchstr 29  81 "CONNECT 2400/REL - CELLULAR\13\10"
  matchstr 30  82 "CONNECT 4800/REL - CELLULAR\13\10"
  ...
  note "MNP10 reliable link established." 3
  userhook 4
  note "Communicating at 2400 bps." 2
  CommunicatingAt 2400
  jump 87
  !
```

```
@LABEL 82
note "MNP10 reliable link established." 3
userhook 4
note "Communicating at 4800 bps." 2
CommunicatingAt 4800
jump 87
```

See "Apple Remote Access Modem Scripting Guide" for more information on modem
scripts.

Also See:
• Apple Remote Access Personal Server for Macintosh User's Guide
• Apple Remote Access Client for Macintosh User's Guide
• Apple Remote Access MultiPort Server for Macintosh Administrator's Guide

Support Information Services
Copyright 1994, Apple Computer, Inc.

Keywords: <None>

==

This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 14392