

Hints on Acquisition / Console Debugging

Last update: 2009-08-12

NOTE: This collection of articles from Agilent MR News focuses on the aspects of debugging / monitoring acquisition and the console, primarily on newer spectrometers such as systems with DirectDrive architecture (Varian NMR/MRI Systems, 400-MR), MERCURYplus, MERCURY-Vx, and UNITY INOVA, mostly restricted to systems with PC / Linux host computer; for information on older systems and information specific to the Ethernet connection between the host computer and the console see also

- [faq/network_debug](#) Debugging Networking and Acquisition Communication

We keep this collection of articles in reverse chronological order; please keep in mind that older articles (further down) may refer to older OS versions and older hardware.

List of Titles in the Sequence of Appearance in the Section Below:

=====

2009-08-12:

- Routers for the Host-to-Console Connection
- Installing VnmrJ 2.2D on Systems with LinkSys Router

2008-11-04:

- Watching the Acquisition Bootup Messages on a PC / RHEL
- Using "cu" to Monitor Acquisition Bootup Messages
- Cabling for Monitoring Acquisition Diagnostics / Bootup

2007-12-10:

- Spectrometer Host Names - Potential Issues and Conflicts

2007-06-19:

- Important Note on Using "setacq" under Linux

2006-05-23:

- Restoring the Linux Console Functionality Under RHEL 4

2005-06-07:

- Locator / VnmrJ - A Warning About Modifications to "/etc/hosts"

2005-04-24:

- Setting up Non-Standard IP Addresses on the Acquisition Network

2004-07-27:

- Using a Router to Connect the Host Computer to the Acquisition CPU

1998-12-03:

- Acquisition Communication on MERCURY-Vx and UNITY INOVA
- What Does "setacq" do on a MERCURY-Vx or UNITY INOVA?
- Replacing the Acquisition CPU on a MERCURY-Vx or UNITY INOVA

=====

2009-08-12:

ROUTERS FOR THE HOST-TO-CONSOLE CONNECTION:

For some time, back in the era of VnmrJ 1.1D, Varian sold systems (UNITY INOVA, MERCURYplus, MERCURY-Vx, both Sun/Solaris- and PC/Linux-based) equipped with a Linksys router in lieu of a second Ethernet network card. Particularly with Sun / Solaris, this avoided the costs for a second Ethernet card (fairly substantial on Sun workstations) and avoided issues such as inadvertent IP packet routing between the acquisition ("private") network and the "public" network connection - see Agilent MR News 2004-07-27 for more information. In addition to the above, the router could theoretically be configured as a firewall, providing additional protection, though we have never really explored that option. Times have changed, however:

- on DirectDrive systems, typical (affordable) routers would be a substantial communication bottleneck between the host and the console and are therefore not a viable option;
- the costs for a second Ethernet card for PCs are MUCH smaller than for SPARC-based Sun workstations - and also small compared to the price of a stand-alone router;
- especially with PCs we encountered a number of support issues (with the above legacy systems) that were hard and time consuming to fix / resolve.
- from a security point-of-view, a dual-Ethernet port Linux configuration is not an issue (e.g., through inadvertent packet routing between the two networks): it is even possible to enable the Linux firewall on the "public" port without affecting the acquisition communication, see Agilent MR News 2009-03-24 and Agilent MR News 2009-04-01 for details.

We have therefore abandoned the idea of using routers for the console connection for any current system.

For legacy systems (UNITY INOVA, MERCURY-Vx and MERCURYplus), SPARC-based Sun host computers are no longer available and are NOT supported by the current (and last) host software, VnmrJ 2.2D (see Agilent MR News 2008-12-02). This eliminates the main reason (expensive network cards for Sun hosts) for using routers in first place, and if you upgrade such a system with a PC host running VnmrJ 2.2D, the PC will by default be equipped with two network cards. Given the potential for hardware and configuration issues and the associated downtime and support costs, we decided to stop supporting routers even for PC hosts running Linux and VnmrJ 2.2D: the addition of a second Ethernet card to the PC is the far more efficient and cost-effective solution overall.

If you have a UNITY INOVA, MERCURYplus, or MERCURY-Vx system with a PC host running VnmrJ 1.1D or VnmrJ 2.1B that is still equipped with a Linksys router, and you want to upgrade to VnmrJ 2.2D, we strongly recommend purchasing a second Ethernet card for the PC (e.g., the Broadcom NetXtreme 10/100/100 Gigabit Ethernet controller PCI express card, available from Dell for around USD 50 when purchased with a PC). The Linksys router is no longer supported by Varian on legacy systems running current software; this does of course not include spectrometers under maintenance agreement and still running the software that was originally installed with the router configuration, such as VnmrJ 1.1D or VnmrJ 2.1B (but of course we recommend upgrading to VnmrJ 2.2D where this is feasible, see Agilent MR News 2008-12-02).

If for whatever reason you still want to upgrade to VnmrJ 2.2D on a system with Linksys router and you don't want to or cannot add a second Ethernet card right now, Bert Heise has found a workaround that makes this combination work, see the article below. Just keep in mind that Varian service and TAC personnel may not be able to help in the case of problems!

[Agilent MR News 2009-08-12]

INSTALLING VnmrJ 2.2D ON SYSTEMS WITH LINKSYS ROUTER (by Bert Heise, Varian):

VnmrJ 2.2D can in principle (though outside of the official Varian support matrix, see the article above) be installed on UNITY INOVA, MERCURYplus and MERCURY-Vx consoles that are equipped with a PC / Linux host and a Linksys router in lieu of the second Ethernet card for the host-to-console connection. However, VnmrJ 2.2D very likely will NOT work out-of-the-box in connection with such a router. In order to make VnmrJ 2.2D work in such a setup, you may need to make a couple adjustments. The primary issue is that the IP address of the Linux PC (if equipped with only one Ethernet card) MUST end with a "1", e.g., "172.16.0.1". Some systems were originally set up with an IP address of, for example, "172.16.0.11", while the IP address "172.16.0.1" was assigned to the router. For hosts running RHEL 5.1, the IP address ending with "1" is hardcoded into line 41 of the file "/etc/bootptab", reading

```
:sa=10.0.0.1
```

(on systems running RHEL 4.x, this line is not present in "/etc/bootptab"). As a result, running "setacq" will fail on such systems - without reporting any error: the MAC address of the console is found, but communication still cannot be established. To make it work, you can either

- change the IP address of the Linux host to "172.16.0.1" (and consequently, change the IP address for the router to a different one, such as "172.16.0.10", OR
- ALTERNATIVELY, edit the file "/vnmr/acq/bootptab.51", by changing the line 41 from

```
:sa=10.0.0.1
```

to

```
:sa=10.0.0.11
```

(assuming your Linux machine's IP address ends with ".11". You must change the last IP address segment ONLY: DO NOT replace it with "172.16.0.11" directly, but leave the first three segments of the IP address as is, otherwise "setacq" may fail ("setacq" will update this line with the correct network ID).
Now execute "/vnmr/bin/setacq", and the console connection should now work OK.
[Agilent MR News 2009-08-12]

2008-11-04:

WATCHING THE ACQUISITION BOOTUP MESSAGES ON A PC / RHEL:

On UNITY INOVA, MERCURY-Vx and MERCURYplus systems with a Sun host computer, it is possible to capture and monitor acquisition bootup and diagnostics messages by connecting the diagnostics serial port on the acquisition CPU via a serial port on the Sun host and using the "tip hardware" command in a shell window to view this information, see Agilent MR News 1996-11-20.

On PC hosts running RedHat Enterprise Linux (RHEL), this recipe is not applicable, as a) the port names and pin assignments are different (requiring a different cable layout), and (more importantly), the UNIX "tip" command is not present under RHEL. So, here's a recipe for doing this on spectrometers with a PC / Linux host, from a software point-of-view (for the cabling requirements see the article below).

First, you need to know the name of the serial port in RHEL. With the configurations delivered by Varian this should be "/dev/ttyS0"; if you want to know for sure you can use "dmesg | grep tty" which will yield output such as

```
serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
```

```
00:08: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
```

This is on the editor's Dell Precision 380N running RHEL 5.2 and should apply to the standard (Varian-supported) RHEL 5.1 as well; on a Dell Precision PC running RHEL 4.0u3 you may see

```
Kernel command line: ro root=LABEL=/ rhgb quiet console=tty0
```

```
ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
```

In the second case, the first line is not referring to the serial port - in both cases, the serial port is "ttyS0".

Then, there are two commands in Linux that can be used in lieu of the UNIX "tip" utility:

- "minicom", and
- "cu" (call up)

For users doing serious diagnostics (mostly service, installers), the former is the preferred option, as with "cu" one cannot abort / control the boot-up process, so it is not possible to initialize a controller on a system with DirectDrive architecture. Also on MERCURY-Vx, MERCURYplus and UNITY INOVA systems you need "minicom" to view the Motorola CPU bootup, and to initialize the acquisition CPU. For those who just want to watch or log the boot-up messages, the simpler "cu" utility may do the job, we therefore still included instructions for using "cu" in the article below.

Using the more powerful "minicom" utility may be slightly more complex, but from the documentation we also think it is more reliable. Using "minicom" is a two-step process:

=> First, you must set up the utility; you should only need to do this once. Opening a shell window, then, as root, type "minicom -s" which will edit the "minicom" defaults which are kept in a file "/etc/minirc.dfl". Ignore the initial warning (there are no pre-defined defaults). Now proceed as follows:

- Use the "down" arrow to select "Serial port setup" and type [Enter]
- In the "pop-up window" select "A"; use the back-space key to erase the existing device name, set the name to "/dev/ttyS0", type [Enter]
- In the same pop-up select "E" to change the "bps/par/Bits" (bits per second / parity / Data & stop bits) settings:
 - On systems with DirectDrive architecture (Varian NMR/MRI Systems, 400-MR) select "I" to set the speed to 115200 bits/second, on UNITY INOVA, MERCURYplus and MERCURY-Vx select "E" to set the speed to 9600 bits/second
 - Select "L" to set the parity checking to "none"

The top line should now read "Current: 115200 8N1" (DirectDrive) or "Current: 9600 8N1" (older systems), respectively. If necessary, select "V" to set the data bits to 8, and/or "W" to set the stop bits to 1, then type [Enter] for the previous menu.

- If necessary, use "F" to toggle the hardware flow control to "No" (also

the software flow control should be "No" - it can be toggled with "G"), the type [Enter] to exit that menu.

- Using the Up/Down arrow keys, select "Modem & dialing", type [Enter] to change the modem and dialing parameters:
 - Select "A" (Init string), then delete all characters to the left (EXCEPT for the dots, of course!), hit [Enter] (these would have been the commands to set up a dial-up modem; we want no commands to be sent when opening the connection). Then use [Enter] again to exit that pop-up.
- Using the Up/Down arrow keys, select "Save setup as dfl", followed by [Enter] (a message "Configuration saved" should show up briefly).
- Using the Up/Down arrow keys, select "Exit from Minicom" and type [Enter] (if necessary, use [Enter] twice) to return to the root prompt.
- Now, still as root, type

```
chmod 666 /dev/ttyS0
```

such that any user can access the serial port through "minicom" (we want to avoid being forced to do things as root, whenever possible).

As mentioned, the above steps only should be required once only.

=> Now, any user can call

```
minicom -o
```

in a terminal window to start "minicom". Note: only ONE instance of "minicom" is permitted to access a given port at any time! With the above command you should see a welcome screen ("Welcome to minicom!"), with the additional text

```
OPTIONS: History Buffer, F-key Macros, Search History Buffer, I18n
```

```
Compiled on ...
```

```
Press CTRL-A Z for help on special keys
```

and a blinking cursor underneath. The last line above means using [Ctrl-a] followed by "z" for help on the command set. You may see a message such as

Parameter xxx is public, but is marked private in global config file where "xxx" is "bits" or "parity" or "stop bits"; you see this only if you enter or change these parameters in the minicom setup as discussed above.

After connecting the host computer serial port to the acquisition CPU (UNITY INOVA, MERCURYplus or MERCURY-Vx) or controller board (DirectDrive), press the reset button to view bootup messages, and you can also preempt the boot-up by hitting the space bar (to initialize the board). To exit "minicom", use [Ctrl-a] followed by "x". If "minicom -o" fails, use "minicom -s" again to check the port / communication settings.

Dave VanderVelde (Caltech) stated that it may be necessary or helpful to add vnmr1 (and/or any other desired accounts) to the (secondary) group "uucp" in the "Users and Groups" RHEL control panel (leaving their primary group set to "nmr"). He then assigned the ownership of "/var/lock" to "uucp", using

```
chown uucp /var/lock
```

None of this was mentioned in our own recipes, so it may not be necessary.

The editor would like to thank Dave VanderVelde (Caltech) for suggesting this topic, and Christine Hofstetter, Alberto Ramirez, and Vladimir Karpovich (all Varian, Palo Alto) for providing the above information!

[Agilent MR News 2008-11-04]

USING "cu" TO MONITOR ACQUISITION BOOTUP MESSAGES:

As mentioned above, the preferred command to monitor and control the acquisition bootup messages (from a DirectDrive controller, or from the acquisition CPU on older systems) is "minicom", see the article above. If, however, you "just want to watch" or long the diagnostics output, also "cu" should be usable - we give you this information mainly for completeness, and because there may be situations where using "cu" is preferable, still. The hardware setup for the serial connection (see again the article above) is the same for both commands. The basic "cu" call is simple:

- on a UNITY INOVA, MERCURYplus or MERCURY-Vx use

```
cu -s 9600 -l /dev/ttyS0
```

where the number following the "-s" (speed) option is the connection speed (Baud rate), the "-l" option is used to specify the device name for the serial port (see the article above). On older Linux versions (RHEL 3?) you may need to use

```
cu -s 9600 -l /dev/ttyS0 > `tty`
```

(note the back-quotes around the "tty" command).

- on systems with DirectDrive architecture (Varian NMR/MRI System, 400-MR), a faster speed setting can be used:

```
cu -s 115200 -l /dev/ttyS0
```

or

```
cu -s 115200 -l /dev/ttyS0 > `tty`
```

Once "cu" is running, it will keep printing the text captured via the serial port. If you want to capture the output of "cu" into a file while watching it you could use

```
cu -s 115200 -l /dev/ttyS0 | tee cu_log
```

where "cu_log" is a log file of your choice. To terminate "cu" you need to type "~.". For more information see "man cu".

In order to control the serial communication setup parameters other than the speed, you need the "stty" command (see "man stty" for complete information). Typically, you need to be root to use "stty" on a serial port:

- "stty -a -F /dev/ttyS0" prints a comprehensive list of all current settings for the specified port ("-F /dev/port_device" option) - you will see output such as

```
speed 9600 baud; rows 0; columns 0; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>;
eol2 = <undef>; swch = <undef>; start = ^Q; stop = ^S; susp = ^Z;
rprnt = ^R; werase = ^W; lnext = ^V; flush = ^O; min = 1; time = 0;
-parenb -parodd cs8 hupcl -cstopb cread clocal -crtcts
-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl
ixon -ixoff -iucrc -ixany -imaxbel -iutf8
opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0
bs0 vt0 ff0 isig icanon iexten echo echoe echok -echonl -noflsh -xcase
-tostop -echoprtr echoctl echoke
```

- the vast majority of these parameters are not relevant for our purpose; you may just need to set very few, specific settings in case the defaults don't work, e.g.:

```
stty 9600 cs8 -parenb -crtcts -echo -F /dev/ttyS0
```

This has the following effect:

- "9600" sets the speed to 9600 Baud,
- "cs8" sets the character size to 8 bits,
- "-parenb" DISables parity checking ("parenb" would enable it),
- "-crtcts" DISables hardware (CRT/CTS) handshaking ("crtcts" would enable it),
- "-echo" disables the local echo (printing of characters that are typed, "echo" would enable the local echo).

As you can see in the output of "stty -a" shown above, in the given example the settings in the "stty" call are all already in place, hence you would not need to call "stty" to change any settings. On systems with DirectDrive architecture (Varian NMR/MRI System, 400-MR) you could use "stty" just to set the speed, with

```
stty 115200 -F /dev/ttyS0
```

which would then simplify a subsequent "cu" call to

```
cu -l /dev/ttyS0
```

Again, the preferred and recommended command is "minicom", NOT "cu" (see the article above); the output of "man cu" explicitly mentions (under "Bugs") that "This program does not work very well." - the main advantage of "cu" appears to be its much simpler interface!

Thanks to Vlad Karpovich, Christine Hofstetter and Bayard Fetler (Varian, Palo Alto) for help in gathering information for this article!

[Agilent MR News 2008-11-04]

CABLING FOR MONITORING ACQUISITION DIAGNOSTICS / BOOTUP:

In the articles above we discussed the software aspects of monitoring acquisition bootup and diagnostics messages on a Linux-based spectrometer host. The cabling setup depends on the console type, but applies to both Sun (Solaris) as well as PC (RHEL) spectrometer hosts:

- on systems with DirectDrive architecture (Varian NMR/MRI system, 400-MR), connect a serial NULL MODEM cable from the PC serial port to any controller diagnostic port. If using a 9F-9F serial null modem (pin 1 is top right on the longer edge), the wiring is as follows:

```
+--> 1      1 <--+
|  2 <--> 3  |
|  3 <--> 2  |
|  4 <--> 6 <--+
|  5 <--> 5
+--> 6 <--> 4
      7 <--> 8
      8 <--> 7
```

The 1 <-> 6 link in both connectors enables DCD (data carrier detect) at either end of the cable.

- on UNITY INOVA, use a straight-through cable to connect the 25-pin D-type RS-232 port on the back of the console to the serial port on the PC.
- on MERCURY-Vx / MERCURYplus, connect a 9-pin serial NULL MODEM cable to the PC serial port. Connect the other 25 pin end to the blue INMAC adapter (25-pin D male to RJ-45). Then connect a straight through RJ-45 (Ethernet) cable (Varian part-# 01-903417-02) to port 1 on the acquisition CPU.

Information on the INMAC adapter pin assignment:

RJ45 end	25pin end
1 (blue)	6/8/20
2 (orange)	5
3 (black)	7
4 (red)	2
5 (green)	3
6 (yellow)	7
7 (brown)	4
8 (not used)	---

Note: You can use the adapter in the MERCURY-Vx or UNITY INOVA console: VT adapter & Ethernet cable plus a null modem 9-pin / 25-pin adapter/cable. Thanks to Alberto Ramirez and Christine Hofstetter (Varian, Palo Alto) for providing the information for this article!

[Agilent MR News 2008-11-04]

2007-12-10:

SPECTROMETER HOST NAMES - POTENTIAL ISSUES AND CONFLICTS:

One of the root causes for some of the issues described in the above article appears to be that after the initial software installation spectrometer hosts occasionally end up with a primary host name "wormhole" (that's a topic still under investigation). As an update to the information posted in Agilent MR News 2006-10-10 and articles referred to therein: "wormhole" is the name of the spectrometer host ON THE ACQUISITION NETWORK and must NOT be used as host name otherwise. Attempts to change a conflicting name to a suitable alternative (irrespective of the means that are used to alter the host name) cause the "disconnect" within the "sudo" setup referred to in the article above and therefore MUST be followed by calling "sudoin" as root:

/vnmr/bin/sudoin

In systems with DirectDrive architecture (Varian NMR/MRI system, 400-MR), the following names are reserved for the acquisition network (note that as of VnmrJ 2.2C the number of supported RF controller and DirectDigital Receiver boards has been increased):

wormhole
master
rf1, rf2, rf3, ... rf16
ddr1, ddr2, ddr3, ... ddr16
pfg1
grad1
lock1

plus a number of names used for optional peripherals, namely

V-Autosampler	V-LcWorkstation
V-Cryobay	V-Pda
V-AS768	V-Slim
V-AS768Robot	V-Trap
V-Protune	

On UNITY INOVA, MERCURY-Vx and MERCURYplus systems the list of reserved names includes

wormhole
inova
inovaauto

(apart from the names for optional peripherals in the second list above). On GEMINI 2000 and the original (not VxWorks-based) MERCURY systems, there are just two reserved host names, "wormhole" and "gemcon".

NONE of these reserved names can be used as principal name for the respective spectrometer hosts; moreover, if any of these names are used as host name for any other system (such as stand-alone processing computers) in a network with the above types of spectrometers, these hosts with conflicting names cannot be contacted directly by the spectrometers.

These restrictions, except for the "passive" ones referred to in the last

paragraph above (i.e., those imposed by other spectrometers), do NOT apply to older, HAL-based spectrometers such as VXR-S, UNITY or UNITYplus systems: these don't use IP (the Internet Protocol) to communicate with the acquisition computer (i.e., their console computers don't appear as TCP/IP network entities).

[Agilent MR News 2007-12-10]

2007-06-19:

IMPORTANT NOTE ON USING "setacq" UNDER LINUX:

A number of OS-related issues with the "setacq" command have come to light since we started supporting Linux with VnmrJ 1.1D, and new issues still keep cropping up. You might ask why the program wasn't "done right" from the beginning - but as with other software there are complications such as

- new console / hardware configuration features are being added every now and then;

- the operating environment and the Dell PC platform keep evolving;
- the Linux installation procedure is evolving, too, PLUS
- the instructions for the OS installation may have oversights and "holes", i.e., there may be installation options that are new or not discussed in detail, causing users and installers (maybe inadvertently) to "explore" new, possibly undesirable options that may cause "setacq" to fail.

In conclusion: a piece of software such as "setacq" is very hard to get "fixed for good", and we will need to keep watching out for possible hidden flaws with that module.

One issue that was detected recently and was documented in bug report "setacq.j1109" is that in certain configurations, Linux may NOT list the main host name and IP address in "/etc/hosts" (as in Linux this file is only a secondary resource, see also bug "setacq.j2201") - this is fatal for "setacq" and may cause acquisition to fail completely. We therefore STRONGLY recommend checking "/etc/hosts" EVERY TIME prior to calling "/vnmr/bin/setacq" on Linux based hosts (at least until we have addressed this issue in a new version of "setacq"). This not only concerns fresh system / software installations and upgrades to a new VnmrJ version, but also the installation of certain VnmrJ patches that require calling "/vnmr/bin/setacq" (note that Sun / Solaris based hosts are NOT affected by this particular issue). We have updated the VnmrJ patch Web site at

<http://www.varianinc.com/products/nmr/software/patches/>
which you also find via our "Software Corner" at

<http://www.varianinc.com/products/nmr/apps/corner.html>
by adding pointers to this issue, and the Readme files for the affected patches have been rewritten to include a cautionary note about this. Also, especially when trying to install earlier versions of VnmrJ under newer RedHat Enterprise Linux (RHEL) releases (which may not be covered in the initial set of our installation manuals), it may be a good idea to consult the on-line bug lists in our "Software Corner" via the above URL for updates and workarounds for the known bugs with the "setacq" command. The current list of bugs that may be relevant to Linux-based systems is as follows:

Bug-ID	Description
setacq.j1104	Acquisition fails with 68040 acquisition CPU
setacq.j1106	Console IP address 10.0.0.1 not detected
setacq.j1107	Fails to connect to console
setacq.j1108	Results in "command not found" error
setacq.j1109	May fail if host name not in "/etc/hosts"
setacq.j2101	Fails to find the correct device name
setacq.j2103	Does not provide an adequate feedback
setacq.j2104	Appears to have lost essential functionality
setacq.j2201	Network restart may delete information in "/etc/hosts"

Note that bugs in newer software MAY also apply to older software, and vice versa (unless the bug is marked as fixed in a newer release).

We are also adding this information to the document "faq/network_debug" in the on-line User Library and in the "FAQ" pages that you can access via our "Software Corner" at

<http://www.varianinc.com/products/nmr/apps/corner.html>

[Agilent MR News 2007-06-19]

2006-05-23:

RESTORING THE LINUX CONSOLE FUNCTIONALITY UNDER RHEL 4:

In RedHat Enterprise Linux 4 (RHEL 4) there is no clear / easily accessible method to invoke a terminal showing the console output - but being able to view console messages is instrumental to a system administrator's ability to follow the progress of an automation run and/or to diagnose eventual problems during processes involving VnmrJ / acquisitions running in "background", i.e., with otherwise no visible output to the user.

For systems running VnmrJ 1.1D or VnmrJ 2.1A under RHEL 4, Bert Heise has found a recipe / workaround that restores the console window functionality - we have posted this in bug report "console.j2101". That solution involves writing a short shell scripts and a few extra steps. Fortunately, with the current software release (VnmrJ 2.1B) for UNITY INOVA and systems with DirectDrive architecture, there is a much easier solution: simply create a file called "/vnmr/tmp/acqlog", using

```
touch /vnmr/tmp/acqlog
```

As soon as this file exists, VnmrJ will route its automation / acquisition background output to that location, and in any shell window you can now use

```
tail -f /vnmr/tmp/acqlog
```

to keep track of such output (to terminate the "tail" command, use "Ctrl-c").

Note, however, that the content of this file has the potential of growing quickly and consuming major amounts of disk space. So, it is a good idea to keep an eye on "/vnmr/tmp/acqlog", and to reset it periodically. This can be done "manually", using

```
cat /dev/null > /vnmr/tmp/acqlog
```

You can also simply remove the file if and when you think don't need to keep track of this information, and re-create it later, when you need it again. Alternatively, you could use "anacron" (see Agilent MR News 2006-04-25) to reset or "rotate" that file periodically (such as to limit that log to a certain number of days or weeks, as appropriate). A detailed recipe for this will be posted in a future issue of Agilent MR News.

[Agilent MR News 2006-05-23]

2005-06-07:

LOCATOR / VnmrJ - A WARNING ABOUT MODIFICATIONS TO "/etc/hosts":

The UNIX administration file "/etc/hosts" contains a definition that is pointing back to the local system itself:

```
127.0.0.1      localhost
```

in Solaris / SunOS, or

```
127.0.0.1      localhost localhost.localdomain
```

in RedHat Linux. Especially on BSD (SunOS 4.x) systems that are NOT networked this line has the name "loghost" added which otherwise is typically added to the line containing the primary local host name (also called "nodename").

The "localhost" entry is useful for testing networking utilities without actually accessing the network - e.g., you could do "rlogin localhost" or "telnet localhost" to check whether and how remote logins into a specific account work, without first having to log into a remote system and from there back to your own system.

It is little known that the "localhost" entry is actually REQUIRED with some parts of the software - removing or changing it means calling for trouble! One example for such software is the Locator database (Postgres / PGSQL): this will fail if the "localhost" definition (and/or "localhost.localdomain" in RedHat Linux) is missing or mis-defined in "/etc/hosts"; the error messages will indicate that the Postgres software starts looking for a PGSQL server on the network, claiming that port 5432 on your gateway is not open etc.

Conclusion: DON'T TOUCH the above entry in "/etc/hosts"!

Also, keep in mind that at least in Solaris the recommendation is that one should NOT use composite Internet domain names ("host.domain.name") as the primary name in "/etc/hosts", while having such entries as SECONDARY name is OK, see also Agilent MR News 1997-12-06. For example, in Solaris,

```
123.45.67.89    myhost.domain.name myhost
```

should preferably be changed to

```
123.45.67.89    myhost myhost.domain.name
```

One advantage of the second convention is that it avoids confusion in shell scripts and other software, when trying to associate simple host names with a given IP address by looking up "/etc/hosts".

[Agilent MR News 2005-06-07]

2005-04-24:

SETTING UP NON-STANDARD IP ADDRESSES ON THE ACQUISITION NETWORK:

The setup for the acquisition network is done using `"/vnmr/bin/setacq"`. This tool uses a set of "internal guidelines / rules" for the setup - particularly those which use two Ethernet interfaces on the host computer (as opposed to communicating with the console through a router, see Agilent MR News 2004-07-27):

- We don't want acquisition traffic to be visible on the "outside network", hence "setacq" installs a flag file `"/etc/notrouter"` to suppress routing.
- If the laboratory network uses any network number but 10.0.0.x, "setacq" selects "10.0.0" as the acquisition network number. "10.*.*" (class A), along with "172.16.*.*" (class B network) and "192.168.0.*" (class C) are "reserved" / "private" network numbers as per RFC 1597 (see also Agilent MR News 2000-06-23) that are NOT routed by standard TCP/IP routing protocols.
- If the laboratory network is "10.*.*", "setacq" selects "172.16.0.*" for the IP addresses on the acquisition network, which is also a "reserved" / "private" network number, see above.

The use of "reserved" IP network numbers in connection with a flag file `"/etc/notrouter"` provides extra safety against acquisition traffic propagating into unwanted areas. For more information see also our FAQ document "Debugging Networking and Acquisition Communication Issues" (`"/faq/network_debug"`) from the on-line user library at

<http://www.varianinc.com/products/nmr/apps/vnmrusers.html>

Recently there was a case of an IT department which demanded that the acquisition network be neither "10.0.0.*" NOR "172.16.0.*". Strictly speaking this is non-sense, as neither of these network address is routed anyway, and therefore all that matters is the IP address on the host computer's main interface: this can only be in one OR the other of these two networks (we don't support spectrometer hosts with more than two Ethernet interfaces) - but it often is hard to argue against such IT rulings. In case this happens to you, here's a recipe that you can try. The trick is to use the third (class C) "reserved" network ID, "192.168.0.*". To do this, first (as vnmr1) make a backup of `"/vnmr/bin/setacq"`:

```
cp -p /vnmr/bin/setacq /vnmr/bin/setacq.bk
```

then edit `"/vnmr/bin/setacq"` as follows:

- look for a line
 `cons_ip="10.0.0"`
and change this to
 `cons_ip="192.168.0"`
- look for a line
 `fix_bootptab="n"`
and change this to
 `fix_bootptab="y"`

then call `"/setacq"` as root. To check whether this has worked as expected, verify that `"/etc/hosts"` uses "192.168.0.*" addresses for "wormhole" and the other acquisition addresses, and check whether `"/etc/bootptab"` also uses "192.168.0.*" on all lines containing `"ip="`.

[Agilent MR News 2005-04-24]

2004-07-27:

USING A ROUTER TO CONNECT THE HOST COMPUTER TO THE ACQUISITION CPU:

With the exception of the HAL-based systems (VXR-S, UNITY, UNITYplus), all our spectrometers use Ethernet as a link between the host workstation and the console. On a stand-alone spectrometer this is usually done via the standard, built-in Ethernet interface on the host computer. However, most spectrometers today are networked, be it

- for sharing data via the network,
- to allow for remote access, or
- to facilitate downloading and installing software and updates.

This means that we need to accommodate two Ethernet links: a local connection to the console (LAN) and one connection to the wider area network (WAN). These two connections need to be logically separated, because

- the acquisition link should be unaffected by the network load on the WAN connection,
- other systems should not be able to "see" the data traffic to and from the console, and
- for reasons of simplicity (of the software setup) and manageability, the

acquisition network for a given class of spectrometers uses a fixed set of IP addresses, and yet multiple spectrometers of the same type need to be able to coexist on the same network.

Traditionally we have achieved this by adding a second Ethernet interface to the host workstation, one for the LAN (console) connection, and one for the WAN connection. These two networks used different IP network addresses, which automatically creates two independent links. We actually added two additional provisions:

- the acquisition link uses a special, reserved IP network number (10.x.y.z or 176.16.x.y, see also Agilent MR News 1998-12-05) that by definition is NOT routed;
- as Solaris automatically starts up "routed" (the Ethernet "route" daemon) when multiple Ethernet interfaces are present, we also create a "flag file" "/etc/notrouter" to prevent this.

All this is working fine and is well established on numerous spectrometers all over the world. Unfortunately, while the prices for computers came down dramatically over the past years, the prices for accessories such as second Ethernet interfaces did not follow that trend to the same degree, such that now the costs for an extra network interface are substantial compared to the price of a low-end computer.

Last year we found a more cost-efficient solution to this, and we started shipping MERCURYplus and UNITY INOVA spectrometers with a single Ethernet interface on the host, plus a Netgear router as interconnect between the host, the acquisition computer and the WAN. In that setup (which requires either VnmrJ 1.1C with the current patch or VnmrJ 1.1D), the host computer was connected in the router's "DMZ" (de-militarized zone), the acquisition system on the LAN connection, and the router's WAN connection was used to connect to the wider area network. While the LAN side of the router remains invisible to systems on the WAN, the host on the DMZ is visible to both sides, just as with a host with two Ethernet interfaces described above.

Over the past months we were able to collect experience with this type of setup, and this experience is essentially all positive, such that we have now decided to make this available (also as a retrofit) as part of our standard offering. At the same time, we have decided to switch to a different router model (see the article below), for several reasons:

- the Netgear router that we used turned out to be incompatible with a Linux PC used as MERCURYplus host computer (see Agilent MR News 2004-05-17),
- with the popularity of the Internet, the number of crackers and other malicious users has gone up as well, and security has become a major concern, especially for systems on open networks (such as in many universities), and we were looking for a router configuration that can be set up as a firewall, offering vastly improved protection, and
- while the Netgear router satisfied the throughput needs between the host and the acquisition computer, its WAN side was designed as a connection to an ADSL router or cable modem with a maximum throughput of a few Mbps only, a rather modest value for sites that would like to use extensive data sharing and/or remote spectrometer control via the WAN connection.

The new router model, a Linksys EtherFast Cable/DSL Router does not have shortcomings in these areas and is now available with the current systems running VnmrJ 1.1C / 1.1D, and as retrofit for systems running that software.

VNMR 6.1 and earlier software does not support such a router configuration - these systems still require two Ethernet interfaces on the host computer. On the other hand, with such systems you can of course still insert a router / firewall box between the spectrometer host(s) and the WAN, without touching the acquisition connection (see Agilent MR News 2002-08-19 and Agilent MR News 2002-08-26).

On the other hand, the Linksys router is a mandatory part of a networked (MERCURYplus) Linux PC host, see the article below.

[Agilent MR News 2004-07-27]

1998-12-03:

ACQUISITION COMMUNICATION ON MERCURY-VX AND UNITY INOVA:

Both the MERCURY-Vx and the UNITY INOVA use standard IP Ethernet protocols to communicate with the acquisition CPU. However, the acquisition CPU is not equipped with a disk or other local storage media: after switching on, it lacks information to run, in particular it does not know its IP address, and the host CPU therefore cannot use IP to establish communication. Instead, after switching on or resetting, the acquisition CPU starts its PROM-based

software (a small version of VxWorks) that reads its hardware Ethernet address (e.g.: 8:0:20:93:bb:fe) and then broadcasts it using the standard network boot protocol (BOOTP).

Spectrometer hosts run a "bootp daemon", /vnmr/acqbin/bootpd, which listens on the Ethernet via ports 67 and 68 (as defined in /etc/services). It uses information from a file /etc/bootptab to assign an IP address (10.0.0.2) to the hardware address in the bootp broadcast. "bootpd" then returns the information from /etc/bootptab to the acquisition CPU. This file also defines which file the console should upload from the Sun to continue the bootup. The file "vxWorks" (the real-time operating system for the acquisition CPU) from the directory /tftpboot/vxBoot is uploaded to the acquisition CPU. On the UNITY INOVA, the file vxWorks.auto from the same directory is uploaded to the Magnet and Sample Regulation (MSR) board (named "inovaauto", IP address 10.0.0.4). The address 10.0.0.3 is used for the "VME port" of the acquisition CPU which acts as a gateway/relay for the communication between the Sun and the MSR board that has no direct Ethernet connection to the host.

The console uses TFTP (the "trivial file transfer protocol") to upload the operating system to its CPUs. The TFTP daemon (tftpd) must be running for this to work. "setacq" activates the following line in /etc/inetd.conf:

```
tftp dgram udp wait root /usr/sbin/in.tftpd in.tftpd -s /tftpboot
before rebooting the system. For security reasons, tftpd only has access to
/tftpboot - that's why we need to store the initial upload files in this
directory rather than in /vnmr.
```

Once VxWorks is uploaded to the console, the standard IP protocols can be used: the acquisition CPU now almost behaves like a UNIX system (with some restrictions, particularly when the "small kernel" is loaded).

We normally use the (class A) network address 10, but if that is used for the local network, the class B network address 176.16 is used instead. With a netmask of 255.255.255.0, the acquisition network address therefore is either 10.0.0 or 176.16.0. According to RFC 1597, the network addresses 10, 176.16, and 192.168.0 (class C) are reserved for "private networks", i.e., networks that remain invisible to public TCP/IP routing.

[Agilent MR News 1998-12-03]

WHAT DOES SETACQ DO ON A MERCURY-VX OR UNITY INOVA?

The shell script "setacq" has the task to set up all files and facilities at a UNIX level for acquisition to function. This is a rather complex task and involves many steps - normally there is no need for the user to know about any of this - but in the case of problems it may be useful to have some more information:

- it kills any running "bootpd" process. This forces the console to resend the broadcast package (otherwise the console would boot and be silent). The Ethernet hardware address is extracted from the broadcast package and stored in /etc/bootptab. This requires that the acquisition CPU has been rebooted and that there is no other device on the acquisition network!
- setacq determines the IP network address for the acquisition: if "10" is used for the local network, 176.16 is used instead.
- if a file /etc/hostname.* does not exist yet for the acquisition Ethernet port, it is created (the host name on the acquisition port is "wormhole"), and the Ethernet interface is activated when the Sun is booted next time.
- the following lines (with network address 10.0.0) are added to /etc/hosts:

10.0.0.1	wormhole
10.0.0.2	inova
10.0.0.4	inovaauto

Remember that it is not possible to use these three names for Internet communication on your local network!
- the acquisition CPU (inova) and the MSR board (inovaauto) are added to /etc/hosts.equiv, such that rsh can work for these two CPUs. This is needed only in case the large kernel is run.
- tftpd is activated in /etc/inetd.conf (see the previous article)
- if necessary, the lines referring to "hosts", "ethers", and "bootparams" in /etc/nsswitch.conf are modified to ensure that the local files are looked up ([NOTFOUND=return] is changed into [NOTFOUND=continue] on the relevant lines)
- if necessary, /tftpboot is created, and the subdirectory vxBoot.small from /vnmr/acq is copied into to /tftpboot/vxBoot (by default, we use the small kernel, rather than /vnmr/acq/vxBoot.big).
- ports 67 (bootps) and 68 (bootpc) are activated in /etc/services
- the file /vnmr/rc.vnmr is copied into /etc/init.d; two symbolic links

/etc/rc3.d/S19rc.vnmr and /etc/rc0.d/K19rc.vnmr are added: these are called at bootup and shutdown time.

- /etc/acqpresent is created: this is a flag file that causes the acquisition processes to be started on the Sun at bootup time.
- if a default route exists and /etc/defaultrouter is absent or empty, this file is created.
- to ensure that the Internet routing daemon (in.routed) is not started at bootup time, a file /etc/notrouter is created.
- if necessary, setacq asks the user to reboot Solaris, otherwise it just starts the BOOTP daemon (bootpd).

[Agilent MR News 1998-12-03]

REPLACING THE ACQUISITION CPU ON A MERCURY-VX OR UNITY INOVA:

After replacing the acquisition CPU, the console Ethernet hardware address has changed (every Ethernet hardware interface comes with a unique, hard-coded 6-byte address). The bootpd process catches the BOOTP broadcasts sent out by the acquisition CPU, but it cannot find an IP address for it in /etc/bootptab. therefore, the acquisition cannot be booted, and it will of course not respond to "ping inova".

The solution: you must call "setacq" after replacing the acquisition CPU. It is not necessary to reboot the Sun unless "setacq" tells you to do so.

You may need to initialize the NVRAM in the acquisition CPU, as described in Appendix D, "UNITY INOVA Acquisition CPU Initialization" in the "Software Installation Manual".

[Agilent MR News 1998-12-03]

=====