

```

HHC01027I Hercules version 4.00, built on Dec 31 2012 17:30:45
HHC01031I Running on z10 (Linux-2.6.18-194.17.4.el5.el SMP Wed Oct 20 13:03
HHC01018I 0:0440 COMM: client 127.0.0.1 devtype 1052: connected
)2132
005) 10.12.25 04/05/13 PILATUS

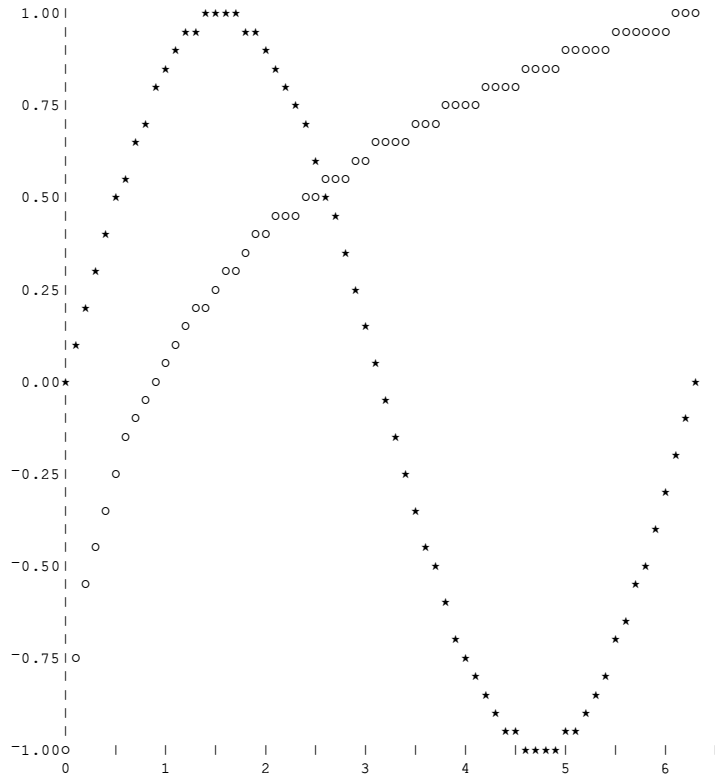
A P L \ 3 6 0

```

```

)LOAD 1 PLOTFORMAT
SAVED 7.07.27 07/21/68
80 PLOT 83 64PA, (-1+O÷r/(O÷*1+6×A)÷2),10A÷((2×01)×-1+164)÷63

```



MVT for APL Version 2.00

OS/360-MVT 21.8F

customized for use with

APL\360 Version 1 Modification Level 1

User's Manual

March 2013

```

*****
*
* This User's Manual applies to MVT for APL Version 2.00
*
* © See Appendix B for copyright information.
*
*****
*
* The MVT for APL Version 2.00 distribution (mvt4apl-2.00) features:
*
* • An OS/360 MVT 21.8F system designed for use with APL\360.
* • An automated procedure to build APL\360 and the MVT supervisor
*   calls it relies on from source.
* • An automated MVT operator (APLPILOT).
* • IBM 1052-7 and IBM 2741 terminal support through an adapted
*   version of the Hercules System/370, ESA/390, and z/Architecture
*   Emulator.
* • An OPFNS workspace supporting APL\360 operations.
* • A comprehensive public workspace library.
* • Scripts and related software combining all components.
* • The MVT for APL Version 2.00 User's Manual.
*
*****
*
* Jürgen Winkelmann, winkelmann@id.ethz.ch, March 29, 2013
*
*****

```

Table of Contents

APL\360 Resurrection.....	5
Installation and Use of APL\360	6
Prepare the Terminal Emulation	6
Windows Systems	7
Linux Systems	7
Set the Time Zone	8
Obtain the APL\360 Source Code	8
Start the APL\360 System	8
Initialize the Recording Terminal	9
Open Additional IBM 1052-7 Terminal Emulation Windows.....	10
Terminate the APL\360 System	11
Reference Information	11
IBM 1052-7 Keyboard Map for Use with the APL Printing Element	11
IBM 1052-7 Terminal Emulation Notes	11
Telnet Connectivity	12
IBM 2741 Keyboard Map for Use with the rxvt4apl Terminal Window	13
IBM 2741 Terminal Emulation Notes	13
Telnet Connectivity	14
Recording Terminal Notes	15
Unlocking and Locking the Keyboard of the Recording Terminal	15
Automated Recording Terminal Operations	16
Date and Time in APL\360	17
Sample APL\360 Session	18
Operating System Information	23
System Consoles.....	24
IPL Methods	24
TSO Access	25
Important Datasets	27
To Do	31
Appendix A: Acknowledgments.....	32
Appendix B: About the MVT for APL distribution.....	32

About the APL\360 Software	32
About the Public Library Workspaces.....	33
About the Hardware.....	33
About the Operating System	33
About the MVT for APL distribution	33

APL\360 Resurrection

In October 2012 IBM licensed the source code of the APL\360-OS Version 1, Modification Level 1 program product (Program Number 5734-XM6) to the Computer History Museum (Mountain View, CA). The license authorizes the Museum to make that source available to the general public which it did by providing it as a download from its website at

<http://www.computerhistory.org/atchm/computer-history-museum-software-license-agreement>

Once the license agreement is accepted the APL\360 source can be downloaded. It is a single IEBUPDTE format file that restores to 50 source members and 38 macros making up to a total of 37404 statements in basic assembler language.

The source comes without any documentation. On bitsavers and other internet resources many manuals on APL\360 can be found. But most of them rely to the 1968 Type III Contributed Program or to the DOS versions of the Program Product (5736-XM1 or 5736-XM6), which were designed to run under DOS or standalone, but not under OS/360. As functionality is very close the primers, user guides and reference manuals suffice perfectly when it comes to working with APL\360 but the installation and system generation manuals aren't very useful.

To make things worse there is also none of the original installation materials available any more, neither the sample jobs used to assemble, link and execute APL\360 nor the distribution tape, which is needed to initialize the library DASDs and load the *OPFNS* and the famous *)LIB 1* workspaces that were delivered with the Program Product.

The purpose of the "MVT for APL" distribution (mvt4apl) is to close the gap and provide the information and tools necessary to get APL\360-OS Version 1, Modification Level 1 up and running on OS/360-MVT 21.8F in accordance with the given license conditions. It comprises:

- A ready to run OS/360-MVT 21.8F system configured to accommodate APL\360 but not containing any copyrighted code.
- An automated procedure to build APL\360 and the MVT supervisor calls it relies on from source.
- An automated operator (APLPILOT) for the OS/360-MVT system.
- A minimal automated operator for the recording terminal, which is intended for server type Linux systems to enable deployment of AaaS (APL as a Service).
- The MVT for APL Version 2.00 User's Manual.
- A partial reimplementaion of the OPFNS workspace with sufficient functionality to operate APL\360 on emulated hardware.
- A comprehensive public workspace library which has been imported from APL\MTS, a port of the Type III version of APL\360 to the MTS operating system. Amongst many others the public library includes the *)LIB 1* workspaces, carefully adapted to be as close as possible to those delivered originally using information found in the APL\360 User's Manual.
- The Hercules System/370, ESA/390, and z/Architecture Emulator with IBM 1052-7 and IBM 2741 terminal support adapted to the communication and character set requirements of APL\360.

No OS/360-MVT or Hercules skills are required to install and operate the system. To use APL\360, however, basic knowledge of the APL language is advantageous. Although a

minimal APL\360 session is provided as a sample this documentation is not meant to be an APL\360 tutorial. Having original APL\360 documentation at hand is required to get acquainted with the APL\360 system and language. Bitsavers have a vast amount of documentation available that can be downloaded from <http://bitsavers.org/pdf/ibm/apl>. To start using APL\360 download at least:

- [GH20-0906-1 APL 360 OS DOS Users Manual Jan73.pdf](#)
- [S210-0007-0 APL 360 Reference Card.pdf](#)
- [GH20-0850-1 APL 360 OS DOS General Information Manual Dec70.pdf](#)

Installation and Use of APL\360

This chapter presents automated procedures to install and operate an APL\360 system on Windows or Linux. The APL\360 system will come up with two IBM 1052-7 terminal emulation windows using the PuTTY telnet client; additional ones (up to a total of eight) can be opened as needed.

It should be noted that the present APL\360 implementation also features an IBM 2741 terminal emulation. Both, the IBM 1052-7 and the IBM 2741 emulations, work with a variety of telnet clients as long as certain protocol, font and character set encoding requirements are matched. Modifying the terminal configuration is outside the scope of this chapter. See the “Telnet Connectivity” sections on pages 12 and 14 for details if you want to change the terminal configuration.

Four easy steps need to be performed to get APL\360 up and running:

1. Prepare the terminal emulation.
2. Set the time zone.
3. Obtain the APL\360 source code from the Computer History Museum.
4. Start the APL\360 system.

No Hercules or OS/360-MVT skills are required to follow the instructions given in this chapter. To be as straightforward as possible technical information is kept to a minimum. For technical details refer to the “Reference Information” chapters starting at page 11.

Before you begin unzip the `mvt4apl-2.00.zip` archive to an arbitrary folder. This folder will be referred to as the `mvt4apl` folder from now on.

Prepare the Terminal Emulation

The automated APL\360 installation and operation procedures are designed for use with emulated IBM 1052-7 terminals using a US-English keyboard layout. The terminal emulation uses the PuTTY telnet client to connect to the APL\360 system and the `apl360i` font (which is the APLPLUS semi-italic font, encoded specifically to the needs of APL\360) to display the APL character set.

The PuTTY telnet client comes ready to run with MVT for APL. There are, however, a few configuration steps required to install the `apl360i` font and the telnet parameter settings on your system and to make a US-English keyboard mapping available if your system isn't setup with a US-English keyboard natively.

Windows Systems

- Open folder `mvt4apl\Fonts` and install the

`apl360i-semi-italic_SYMBOL.ttf`

true type font using the procedure recommended for your Windows version. On Windows XP or older this means drag and drop the file into the `C:\Windows\Fonts` folder, on newer systems it's right clicking the file and selecting `Install` from the context menu.

- Open folder `mvt4apl\PuTTY-0.62_for_Hercules\sessions` and click (or double click, depending on your settings) the file

`APL360_Windows.reg`

to merge the PuTTY session parameters into the registry. Allow any permission requests Windows might present until it confirms successful registry insertion.

- If your Windows system doesn't use a US-English keyboard natively add the US-English input scheme with a US-English keyboard to your list of input schemes and activate the display of the language bar in the taskbar using the "Region and Language" menu of the control panel. This allows setting the keyboard layout of the IBM 1052-7 terminal emulation windows to US-English without affecting your native keyboard settings.

Linux Systems

- Open folder `mvt4apl/Fonts` and install the

`apl360i-semi-italic_ISO-8859-1.ttf`

true type font using the procedure recommended for your Linux system. Procedures may vary greatly depending on the Linux distribution and version. Besides performing a system wide installation by copying the font into a designated true type font folder and updating some caches (see "Reference Information" on page 11 for examples) on many Linux distributions a local installation (font is usable by the current user only) can be done by simply clicking or double clicking the font to view it and pressing some "Install" button on the font-viewer panel.

- Create folder `.putty/sessions` in your home directory if it doesn't exist already. Then install the PuTTY session parameters by entering the following command in a shell window from the home directory:

```
mv mvt4apl/PuTTY-0.62_for_Hercules/sessions/APL360_Linux .putty/sessions/APL360
```

Adapt the pathname of the `mvt4apl` folder in the above command if necessary.

- If your Linux system doesn't use a US-English keyboard natively add a US-English keyboard to the list of keyboards available and activate the "keyboard applet" on your desktop. How keyboards are added, how the "keyboard applet" is named and how it is activated varies greatly amongst Linux distributions and versions, so refer to your system's documentation for that. Once done, verify that you can change the keyboard layout on a window by window basis, which is required to set the keyboard layout of

the IBM 1052-7 terminal emulation windows to US-English without affecting your native keyboard settings.

Set the Time Zone

The MVT for APL system comes with a time zone setting of UTC+2 (MEST, Middle European Summer Time). It is recommendable to set the time zone in the Hercules configuration files to your current time zone such that APL\360 creates meaningful time stamps on saved workspaces and returns the correct time when the time of day function ($\pi 20$) is called. This is done by editing the two files

`mvt4apl/conf/mvt_extcons.cnf` and `mvt4apl/conf/mvt_intcons.cnf` (Linux systems)
`mvt4apl\conf\mvt_extcons.cnf` and `mvt4apl\conf\mvt_intcons.cnf` (Windows systems)

Open these files with your editor of choice and change the time zone offset in line 16 to the setting your native Windows or Linux system uses currently, for example: If your system currently uses PDT (Pacific Daylight Time), change it to -7.

Obtain the APL\360 Source Code

Because the license conditions don't allow redistribution of any APL\360 licensed code or derivative works MVT for APL comes without any APL\360 licensed materials installed. The automated operations procedure of the APL\360 system checks upon startup whether it is installed already and kicks off an automated installer if not. The installer builds the executable APL\360 system from source. Thus it is required to obtain the APL\360 source code before starting the system for the first time:

Point your browser to

<http://www.computerhistory.org/atcm/computer-history-museum-software-license-agreement>

and accept the license agreement. This will start the download (depending on your browser settings you might have to confirm the download). Open the downloaded ZIP archive and extract the source file

`APL360_source_code.txt`

to folder `mvt4apl\source` (Windows systems) or `mvt4apl/source` (Linux systems).

That's it! You're now ready to start APL\360:

Start the APL\360 System

On a Windows system open the `mvt4apl` folder and click (or double click, depending on your settings) the `apl360.bat` file.

Note: A message stating that `MSVCR100.dll` is missing might pop up after clicking the `apl360.bat` file. This means that a Microsoft runtime library is missing. In this case download and install

<http://www.microsoft.com/download/en/details.aspx?id=5555> for 32-bit systems
<http://www.microsoft.com/download/en/details.aspx?id=14632> for 64-bit systems

On a Linux system open a shell window, change directory to the `mvt4apl` folder and enter the command

```
./apl360
```


at the shell prompt.

If this is the first use of the system APL\360 needs to be installed from source before it can be started. This is indicated by the prompt

```
starting APL\360 installation...
Continue [Y/n]?:
```

Press return to confirm the installation, which will take a few minutes to complete. During this process OS/360-MVT will be IPLed four times. After the fourth IPL, APL\360 will come up. So, get yourself a coffee, sit back, relax and wait for the APL\360 terminal windows to open.

Note: For technical reasons the automated installation process waits at some stages for several seconds, so be patient if the installation seems to have stalled: It will continue shortly.
For enhanced robustness of the automated process several crucial MVT commands, particularly monitoring settings and replies to system requests are issued multiple times, resulting in occasional IEE701I messages. These can safely be ignored.

Subsequent starts of the system will of course bring up APL\360 directly, without performing the installation again.

APL\360 is now up and running. It will, however, not accept any user sign on before the recording terminal has been initialized.

If you are not using a US-English keyboard natively **switch** the IBM 1052 terminal emulation windows to **a US-English keyboard layout now**.

If this is your **first use** of APL\360 it is recommendable to have a look at the “IBM 1052-7 Keyboard Map for Use with the APL Printing Element” on page 11 and read “Recording Terminal Notes” on page 15 **now**.

Continue with “Initialize the Recording Terminal” below to enable user sign on.

Initialize the Recording Terminal

Before users can sign on to APL\360, the recording terminal must be initialized. The recording terminal is the terminal being signed on as user 3 1 4 1 5 9, which is the operator (OPR) user. So, enter in one of the IBM 1052-7 terminal emulation windows:

```
) 314159
```

APL\360 will respond with

```
OPR) HH.MM.SS MM/DD/YY OPR
```

Finally load the *OPFNS* workspace by pressing ctrl-a, followed by entering:

```
)LOAD OPFNS
```

APL\360 will respond with

```
SAVED 14.32.54 02/17/13
```

APL\360 is now ready to use and user sign on will be enabled 90 seconds after the *OPR* sign on message was displayed. In the era of large production APL\360 systems the purpose of

the 90 seconds delay was allowing the operator to set a message of the day or to perform other initial actions defined for the system.

You probably noticed that the value of *YY* in the above sign on message of the OPR user lies 28 years in the past. This comes from the fact that Hercules is configured to use an epoch starting at 1960 with the system date offset to be 28 years behind the current date, accommodating for the date range supported by OS/360-MVT while still having day of week, day of month and month correctly set. APL\360 reads the current date once at startup and from that point on keeps it as a character constant in memory. Changing that constant allows setting any arbitrary date without affecting APL\360 operations. The *SETDATE* function has been added to the *OPFNS* workspace for that purpose. So, if you like APL\360 to use the correct year in date stamps and in the date function (∓ 25) press ctrl-a and enter

```
SETDATE 'MM/DD/YY'
```

where *MM/DD/YY* is the current date. There will be no response from APL\360 to this command.

Enter ctrl-a followed by *DESCRIBE* at the recording terminal for more information on the functionality provided by the *OPFNS* workspace and read “Date and Time in APL\360” on page 17 for background information on how date and time are presented in APL\360.

If this is your **first use** of APL\360 it is recommendable to conduct the “Sample APL\360 Session” on page 18 **now**.

Open Additional IBM 1052-7 Terminal Emulation Windows

The MVT for APL system is configured to support up to eight IBM 1052-7 terminal emulation windows being connected concurrently, two of them being opened automatically at startup. Additional windows can be opened as follows:

On a Windows system open the mvt4apl folder and click (or double click, depending on your settings) the apl360_client.bat file.

On a Linux system open a shell window, change directory to the mvt4apl folder and enter the command

```
./apl360_client
```

at the shell prompt.

If you are not using a US-English keyboard natively **switch** the newly opened IBM 1052 terminal emulation window to **a US-English keyboard layout now**.

Terminate the APL\360 System

To terminate APL\360 press ctrl-a at the recording terminal and enter:

SHUTDOWN

APL\360 will respond with

SYSTEM SHUTDOWN INITIATED, PLEASE SIGN OFF NOW

Now press ctrl-a and enter:

)OFF

APL\360 will respond with

```
OPR  HH.MM.SS MM/DD/YY OPR
OPR  HH.MM.SS MM/DD/YY OPR
CONNECTED HHHH.MM.SS  TO DATE HHHH.MM.SS
CPU TIME  HHHH.MM.SS  TO DATE HHHH.MM.SS
```

and terminate shortly. The automated operator (APLPILOT) will detect the APL\360 termination and bring OS/360-MVT to an orderly halt which in turn will be detected by Hercules' automated operator (HAO) and cause it to terminate Hercules.

Reference Information

IBM 1052-7 Keyboard Map for Use with the APL Printing Element



The above image shows the original keyboard layout used on IBM 1052 and IBM 2741 terminals when equipped with the APL printing element. The keyboard layout of an IBM 1052-7 terminal emulation window is exactly the same, thus providing a fully authentic APL\360 experience.

IBM 1052-7 Terminal Emulation Notes

The 1052-7 emulation is implemented as a modification of Hercules' console.c driver which emulates *local* 1052, 3215 and 3270 terminals. Thus APL\360 sees it as a *local* 1052 model 7 not as a full function *remote* 1052. 1052-7s typically were used as integrated console printer on smaller System/360 models and, in an APL\360 environment, probably were used primarily as the recording terminal and not so much for real APL work.

These console printers often didn't even have attention or backspace keys (as those were separately orderable features) which lead the APL\360 developers to use Shift-Z for

backspace and Shift-X for linefeed. On the one hand this yields ugly results on paper (printing a `<` APL symbol instead of backspacing and a `>` APL symbol instead of performing a line feed) while of course still executing the requested line editing functions. On the other hand it preempts the `<` and `>` symbols from being used within APL, which fortunately isn't a problem as no function is assigned to these symbols at the APL\360 language level.

To overcome the ugly behavior of backspace and line feed the 1052 emulation has been implemented such that it behaves as a local 1052-7 towards APL\360 only, while it behaves as a full function remote 1052 towards the user's telnet session. So, Shift-Z doesn't print a `<` but backspaces instead and Shift-X doesn't print a `>` but performs a linefeed while printing a `␣` (down caret) instead. Additionally the keyboard's backspace key is assigned the same function as Shift-Z.

So from the user side all keyboard functions work **exactly** as described in the APL User's Manual for 1052 terminals, with the only exception that Shift-X has to be used for linefeed and that `<` and `>` cannot be used as APL symbols (Shift-Y of course may still be used for backspace, but one typically will tend to use the real backspace key for that). Notable is in particular the seamless visual fidelity simulation and the printing of input using the red "ribbon", as is described in the manual.

Removing the ambiguity between the `<` and `>` symbols and the backspace and linefeed functions would require a 2 byte change to the original APL\360 code which isn't done in this implementation for reasons of authenticity. It eventually might be done in a later "modernized" version.

Telnet Connectivity

The IBM 1052-7 emulation should work with any telnet client that negotiates successfully with the Hercules console telnet server and is able to use the apl360 or apl360i font. Tested are HyperTerminal (Windows only) and PuTTY (Windows and Linux).

The MVT for APL system comes preconfigured with PuTTY but any other compatible one can be used by adapting `mvt4apl/apl360_client` (Linux, manual open) `mvt4apl\apl360_client.bat` (Windows, manual), `mvt4apl/scripts/client_linux` (Linux, automatic open after APL\360 startup), and `mvt4apl\scripts\client_windows` (Windows, automatic open after APL\360 startup) to call it with appropriate connection parameters.

In addition to the connection parameter settings used for the preconfigured PuTTY sessions, a sample HyperTerminal parameter file is also provided (`mvt4apl\HyperTerminal Configuration\APL.ht`), allowing you to get up and running with HyperTerminal if you prefer HyperTerminal over PuTTY. If you intend to use any other telnet client it is recommendable to carefully compare the sample session parameters provided for PuTTY or HyperTerminal and adopt them to your preferred client.

Being a Hercules console device the telnet client connects to the regular Hercules console port. As to not affect the functionality of regular Hercules IBM 3270, IBM 3215 or IBM 1052 consoles the APL\360 specific functionality is activated for an IBM 1052 device if it is member of a console group named APL only. The MVT for APL system comes preconfigured with this setting.

When running the MVT for APL system you will typically connect your telnet client to localhost, port 3600, using a terminal type of 1052@APL to select the appropriate console terminal type and group.

IBM 2741 Keyboard Map for Use with the rxvt4apl Terminal Window



openAPL X11 Keyboard Overlay
(PC 101 Key)

The above image shows the keyboard layout used with an emulated IBM 2741 terminal running in an rxvt4apl window.

Note that this layout is *not* identical to the keyboard of the original IBM 2741 terminal when used with an APL printing element. It comprises most of the glyphs used in APL\360 and some additional ones that will return a *CHARACTER ERROR* when used.

IBM 2741 Terminal Emulation Notes

The APL version of the IBM 2741 terminal emulation has been created by Max H. Parke as an adaption of his original Hercules driver for correspondence or papertape encoded 2741s to the needs of APL\360. The device is emulated as being attached through an asynchronous line to an IBM 2703 transmission control unit.

The IBM 2741 emulation is accessed using the Linux native telnet client from an rxvt4apl (this is an rxvt version adapted to the needs of openAPL) terminal window. It is protocol wise fully functioning while functionality wise relying on the telnet client to present keystrokes and display characters using openAPL encoding and to handle any APL\360 specific features (namely visual fidelity). The combination of the openAPL font, the rxvt4apl terminal window, and the native Linux telnet client provide enough functionality to use the emulated IBM 2741 terminal with APL\360. However, the APL\360 character set isn't fully supported and the look and feel differs greatly from the original IBM 2741 typewriter terminal.

Nonetheless the IBM 2741 implementation is structurally cleaner than what has been done for the IBM 1052-7 (full implementation in the Hercules console driver, without requiring more than standard telnet functionality from the client) but the drawback clearly is the requirements to the telnet client it imposes.

For convenience openAPL (openapl-0.14.tar.gz) and rxvt4apl (rxvt4apl-2.4.5.tar.gz) have been included in folder commadpt-apl360 of this distribution. As both, openAPL and rxvt4apl are available on Linux/Unix systems using X11 based desktops only using the IBM 2741 emulation on Windows currently isn't feasible.

- It is not necessary to install the whole openAPL system, the fonts and encodings need to be extracted only. To install the fonts on a Linux system perform these steps:
 - o Untar the openAPL distribution.
 - o Copy the .pcf files from folder openAPL/terminal/XFree86/fonts to your systems folder for bitmap fonts. Locations are for example /usr/share/X11/fonts/misc (RHEL5) or /usr/share/fonts/X11/misc (Ubuntu 12) and gzip them (to end in .pcf.gz). Then issue from this folder mkfontdir to rebuild the directory information (Note: the misc fonts folder might be at a different location depending on your Linux distribution and system setup).
 - o The font apl8x13 will typically work out of the box, while the aplox.xxx fonts may raise problems getting recognized by xfs due to the dot in their names. Should this be the case just alias them to something else by manually editing the fonts.alias file.
 - o A reboot is recommended to be sure the new fonts got picked up by xfs.
 - o Copy files apl2741.xmap and modeswitch.xmap from openAPL/terminal/XFree86/keymap to a convenient location (home directory, for example), they will be needed at execution time.
- Install rxvt4apl using the usual configure, make, make install approach. Review Max Parke's readme-rxvt4apl.txt and readme.txt files for further instructions.

Once the fonts, encodings, and rxvt4apl are installed the following command sequence can be used to open an rxvt4apl window for use with the IBM 2741 emulation (replace <fontname> with the name of one of the openAPL fonts installed above):

```
xmodmap apl2741.xmap
xmodmap modeswitch.xmap
rxvt4apl --meta8 -fn <fontname> -ls &
```

The two xmodmap commands need to be issued once per Linux desktop session only. To open multiple rxvt4apl windows just repeat the rxvt4apl command as often as needed.

Telnet Connectivity

To start an APL360 session in an rxvt4apl window issue the native Linux telnet command in that window:

```
telnet <hostname> <port>
```

where <hostname> is to be replaced with the hostname or IP address of system running the Hercules emulator and <port> with the port at which the desired 2741 device is listening. When running the MVT for APL system you will typically use localhost as the hostname and one of 57411 to 57414 or 57421 to 57424 as port (these ports take one session each).

It should be noted that the rxvt4apl/telnet/2741 combination doesn't default to an 8-bit data path with all Linux distributions and locale settings. When using a 7-bit data path many

APL glyphs cannot be accessed from the keyboard using the mode switch key defined in `modeswitch.xmap`, which makes working with APL\360 extremely difficult. Should your system happen to default to a 7-bit data path use the following sequence of commands to start the telnet session instead of the standard `telnet` command shown above:

```
telnet
toggle binary
open <hostname> <port>
```

Recording Terminal Notes

The terminal being signed on as user 314159 (the operator user *OPR*) is also called the recording terminal. It is crucial to APL\360 that the recording terminal is signed on and has its keyboard locked most of the time. In particular, APL\360 doesn't allow users to sign on when the recording terminal isn't signed on and any user sign on requested while the recording terminal's keyboard is unlocked will not complete until the keyboard has been locked.

As is suggested by its name the recording terminal is the hardcopy log of an APL\360 system: Its primary purpose is to log user sign on and sign off times and ports. In addition, the recording terminal is authorized to issue privileged commands and use privileged π -Beam APL\360 functions. So, in total, the recording terminal is a combination of being the master console and the hardcopy log of an APL\360 system.

To distinguish these two modes of operation APL\360 uses the convention that the recording terminal is the hardcopy log when its keyboard is locked while it is the master console when its keyboard is unlocked. This does in particular mean that the recording terminal cannot print log messages while its keyboard is unlocked and for this reason no user sign on can run to completion while the recording terminal's keyboard is unlocked: APL\360 simply waits until it is able to log the sign on which it can do when the recording terminal is in hardcopy log mode only.

So, remember: A "hanging" user sign on almost always is caused by the recording terminal not being signed on or not having its keyboard locked.

Unlocking and Locking the Keyboard of the Recording Terminal

The exact operating scheme defined by APL\360 for the recording terminal is as follows:

- Once the *OPR* user has signed on APL\360 locks the keyboard of the recording terminal and visualizes this by *not* moving the cursor to column seven as would be done after sign on of a regular user.
- In locked state APL\360 is not expecting input from the terminal and operates it as the hardcopy log printer.
- If APL\360 receives an attention signal from the recording terminal it is switched to input mode by moving the cursor to column 7 and unlocking the keyboard.
- In unlocked state any APL\360 statement or system command (particularly *OPFNS* functions and privileged system commands) can be entered and the hardcopy log printer function is temporarily suspended.
- Once input has been *accepted* the keyboard is locked and the hardcopy log printer function is resumed.
- When the input has been *processed* the cursor will *not* be moved to column seven and the keyboard will remain locked until the next attention signal is received.

That scheme works well on real terminals with a keyboard lock feature because it is simply impossible to enter anything while the keyboard is locked. Today's telnet based emulation, however, accepts input from the keyboard at any time and generates an attention signal automatically when the user presses return. Consequently APL\360 sends out the six blanks to move the cursor to column seven only after it already has received the input, resulting in the output being indented instead of the input. Besides being visually disturbing, unpredictable side effects (hangs to crashes) can happen in timing sensitive situations when the recording terminal is operated in this "send attention and data at once" mode.

To get around this behavior a special "send attention only" function has been assigned to the ctrl-a keyboard sequence of the IBM 1052-7 emulation (the IBM 2741 emulation uses the ctrl-c sequence for the same function). So, the recommended operating scheme of the recording terminal when using the IBM 1052-7 emulation is to **always press ctrl-a before entering** system commands or APL\360 statements.

Automated Recording Terminal Operations

For single user APL\360 systems, which presumably is the most common use case of the MVT for APL system, operating the recording terminal manually as described above is no problem at all. Thus there is only very little incentive to automate it.

On the other hand recording terminal automation is for sure beneficial when operating APL\360 as a multiuser service as it was designed to be used back in the '60s and '70s. That would mean running it on a server type system with telnet access enabled for remote clients.

The MVT for APL system comes with a very basic sample implementation of such a recording terminal automation which currently is available for Linux based APL\360 systems only as this probably would be the configuration of choice for running AaaS (APL as a Service). Prerequisite for using that implementation is the ncat utility being installed on your Linux system (ncat comes with the nmap package on most Linux distributions).

Usage of the recording terminal automation is easy: Instead of initializing the recording terminal after APL\360 has started manually, perform the following steps:

- Logon to a console tty on your Linux system (alternatively a shell window on your Linux desktop will work also, but it is assumed that on a server system there will not necessarily be a desktop logged on all the time).
- Include mvt4apl/aplopr in your shell's path.
- Enter aplopr hostname port
where hostname is the name or IP address of the system APL\360 is running on (hostname typically is localhost and port is a port at which an IBM 2741 terminal defined in the Hercules configuration is listening (57424 would be a recommendable port candidate when APL\360 is running on an MVT for APL system)).

The automated operator will now do the following:

- Sign on the *OPR* account.
- Load the *OPFNS* workspace.
- Set the date.
- Wait for the following events to occur:

- User *PILATUS* (2132) signs on: In this case it will privilege the terminal which was used for the sign on. (The user name and number can be changed by editing the *action_opr* script).
- TOD passes midnight: In this case the time and date of the APL\360 system are adapted to match the new day.

While the automated operator is running APL\360 statements and system commands can be entered at the recording terminal. As opposed to manual operation, however, an attention interrupt should *_not_* be raised before making a recording terminal entry. Note that entries are shown in lower case letters as opposed to manual operations on an IBM 1052-7 where they are shown in uppercase.

A shutdown of the APL\360 system is initiated the same way as with manual operations: Just enter shutdown, followed by)off. To prevent the automated operator from re-signing on the *OPR* user ctrl-c should be pressed as soon as a message comes up advising to do so. If no ctrl-c comes in within two seconds the *OPR* will get signed on again, inhibiting shutdown completion.

Read the comment headers of the scripts in *mvt4apl/aplopr* for details on the automated APL\360 operator.

Date and Time in APL\360

APL\360 handles date and time in a somewhat special way when compared with computer systems being in use today:

- The current date is fetched once at APL\360 startup only. This date is used throughout that whole execution of the system, i.e. it is never updated, regardless how long APL\360 is up.
- The TOD clock is *_not_* reset to zero at midnight. It just continues to count.

For example: If it is now 10am and APL\360 was started three days ago then it will report the date of three days ago as the current date and 82:00:00 (10 plus three times 24) as the current time.

From today's perspective that is really weird. But from the perspective of the APL\360 developers it had the advantage that generating usage and performance histograms was much easier as when permanently having to convert hours and days forward and backward. At least, they definitely looked at it more as being a feature than as being a bug: Over the many years of existence of APL\360 one can see in every version of the various manuals time stamps of saved workspaces in the above weird format and they never bothered to "fix" this.

Looking at the APL\360 source reveals that it would need less than 100 lines of code to change that behavior to what one would expect to see today. Being an authentic resurrection a great deal of effort has been put into implementing APL\360 without changing a single byte of code. Thus a correction of the weird date and time behavior clearly is out of scope of the given implementation.

It should be noted, however, that the *CORRDATE* function provided with the *OPFNS* workspace allows for correction of date and time on the fly when called once per day after

midnight, followed by a sign off and sign back on of the *OPR* user. This method is used by the automated recording terminal operator mentioned in the previous chapter.

Sample APL\360 Session

After having initialized the recording terminal the APL\360 adventure begins. Conduct the following sample session to create your first user accounts and verify some basic functionality.

The system comes “empty”, the *OPR* user is the only one defined as it is needed to get started. Thus the first step is to define regular users. Signal attention and enter `)ADD NUMBER NAME QUOTA` at the recording terminal to define any user you like. For the sample session user 2132 named *PILATUS* and user 1798 named *RIGI* are defined with a quota of 51 saved workspaces each:

```
HHC01018I 0:0440 COMM: client 127.0.0.1 devtype 1052: connected
)314159
OPR) 10.11.34 04/09/85 OPR
      )LOAD OPFNS
SAVED 14.32.54 02/17/13
      SETDATE '04/09/13'
      )ADD 2132 PILATUS 50
      )ADD 1798 RIGI 50
```

Note that both `)ADD` commands don't yield a reply. Now you're ready to sign on the newly created users. Enter `)2132` at the second terminal:

```
HHC01018I 0:0441 COMM: client 127.0.0.1 devtype 1052: connected
)2132
006) 10.28.26 04/09/13 PILATUS

A P L \ 3 6 0
```

Now open a third terminal following the procedure explained in “Open Additional IBM 1052-7 Terminal Emulation Windows” on page 10 and enter `)1798` at that terminal:

```
HHC01018I 0:0442 COMM: client 127.0.0.1 devtype 1052: connected
)1798
007) 10.35.45 04/09/13 RIGI

A P L \ 3 6 0
```

Each user sign on will be reported on the recording terminal which should look like this now:

```
HHC01018I 0:0440 COMM: client 127.0.0.1 devtype 1052: connected
)314159
OPR) 10.11.34 04/09/85 OPR
      )LOAD OPFNS
SAVED 14.32.54 02/17/13
      SETDATE '04/09/13'
      )ADD 2132 PILATUS 50
      )ADD 1798 RIGI 50
006) 10.28.26 04/09/13 PILATUS 2132
007) 10.35.45 04/09/13 RIGI 1798
```

This demonstrates that the recording terminal now is in “hardcopy log” mode.

Continue the sample session as follows:

- Start with the *RIGI* user:
 - display an array
 - create and test a trivial times tables function *M* and a *DESCRIBE* variable
 - save the workspace under the name *TIMESTABLES*
 - clear the workspace and check the library contents
 - sign off

The *RIGI* terminal should look like this now:

```
HHC01018I 0:0442 COMM: client 127.0.0.1 devtype 1052: connected
)1798
007) 10.35.45 04/09/13 RIGI

  A P L \ 3 6 0

      3 15p13x15
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
16  17  18  19  20  21  22  23  24  25  26  27  28  29  30
31  32  33  34  35  36  37  38  39  40  41  42  43  44  45
      ∇E←M B
[1]   E←B×2 5p110
[2]   ∇
      M 7
  7   14  21  28  35
42  49  56  63  70
      DESCRIBE←'WELCOME TO TIMESTABLES, ENTER M NUMBER'
      )SAVE TIMESTABLES
12.09.37 04/09/13
      )CLEAR
CLEAR WS
      )LIB
TIMESTABLES
      )OFF
007 12.10.26 04/09/13 RIG
CONNECTED      1.34.41  TO DATE      1.34.41
CPU TIME      0.00.00  TO DATE      0.00.00
```

- Continue with the *PILATUS* user:
 - load the *TIMESTABLES* workspace saved by *RIGI*
 - display its contents and the *DESCRIBE* variable
 - compute a times table
 - clear the workspace

```
HHC01018I 0:0441 COMM: client 127.0.0.1 devtype 1052: connected
)2132
006) 10.28.26 04/09/13 PILATUS
```

```
A P L \ 3 6 0
```

```
)LOAD 1798 TIMESTABLES
SAVED 12.09.37 04/09/13
)VARS
DESCRIBE
)FNS
M
DESCRIBE
WELCOME TO TIMESTABLES, ENTER M NUMBER
M 77
  77 154 231 308 385
462 539 616 693 770
)CLEAR
CLEAR WS
```

- Explore the public library, still using the *PILATUS* user:
 - load the *NEWS* workspace from library 1
 - display the *DESCRIBE* variable
 - display the *LIBRARY* variable and anything else mentioned in *DESCRIBE* you are interested in.
 - load the *PLOTFORMAT* workspace from library 1
 - create the plot depicted on the title page of this manual

Note: The *PLOT* function call used to create the plot isn't exactly trivial to type in if you are not used to the APL notation. Because copy and paste from the manual (PDF) to the APL360 session isn't possible due to encoding differences file `mvtaapl/doc/plotformat_sample.txt` is provided. It contains the *PLOT* call encoded for the `apl360i` font and can be pasted into the PuTTY session (the PuTTY paste function is assigned to the right mouse button on Windows and the middle mouse button on Linux): Open the file in an 8-bit clean viewer, then copy the whole line and paste it into the APL360 window.

 - sign off

```
)LOAD 1 NEWS
SAVED 21.52.22 03/29/13
DESCRIBE
```

```
WORKSPACE 1 NEWS PROVIDES INFORMATION ABOUT THIS APL360
SYSTEM WHICH WAS RESURRECTED IN AN EFFORT TO PRESERVE AS CLOSE
AS POSSIBLE THE ORIGINAL DISTRIBUTION. APL360 IS A MULTIUSER
SYSTEM RUNNING ON IBM MAINFRAME HARDWARE OF THE 60'S AND 70'S.
```

```
ENTER 'ABOUT', 'CREDITS', 'LIBRARY', 'TERMINALS', 'INDEX',
'APLNOW M D Y', OR 'PRINT I' FOR SPECIFIC INFORMATION OR 'ALL'
```

TO DISPLAY ALL INFORMATION AVAILABLE.

ABOUT PRINTS VERSION AND COPYRIGHT INFORMATION.

CREDITS PRINTS A LIST OF ACKNOWLEDGEMENTS.

LIBRARY PRINTS A TABLE OF CURRENT PUBLIC WORKSPACES.

TERMINALS SUMMARIZES THE TERMINALS SUPPORTED CURRENTLY AND THEIR CAPABILITIES.

INDEX PRINTS INDICES, DATES, AND THE FIRST FEW WORDS OF EACH NOTE IN APLNOW.

APLNOW TAKES AS ITS SINGLE ARGUMENT A THREE-ELEMENT VECTOR REPRESENTING A DATE, AS MONTH, DAY, YEAR. APLNOW PRINTS NOTES ON THE RESURRECTION HISTORY AND THE PRESENT SYSTEM. ONLY THOSE NOTES ENTERED INTO APLNOW ON OR AFTER THE DATE GIVEN AS AN ARGUMENT ARE PRINTED. TO LIST THE COMPLETE FILE, ONE MAY USE APLNOW 1 1 1.

PRINT TAKES AS ITS SINGLE ARGUMENT THE INDEX (AS INDICATED BY THE INDEX FUNCTION) OF A NOTE FROM APLNOW, AND PRINTS THE NOTE.

LIBRARY

1 ADVANCEDEX	2 CATALOG	3 BRFS	5 ALGFORM	6 KINGDOM	100 BEER
1 APLCOURSE	2 STP1	3 CLASS	5 EDIT		100 CHARSET
1 NEWS	2 STP2	3 COMPLEX	5 COGO		100 SINPLT
1 PLOTFORMAT	2 STP3	3 EIGENVALUES	5 COMPCENT		100 TIMESTABLES
1 TYPEDRILL	2 STP4	3 INDEX	5 CW		100 UTILITIES
1 WSFNS	2 STP5	3 UTIL	5 MATRIXALG		
	2 STP6		5 SNOBOL		999 NEWSAPLMTS
	2 STP7		5 SRADEMO		999 TESTS
	2 STP8				

LIBRARIES 1 TO 6 CONTAIN WORKSPACES THAT HAVE BEEN TAKEN FROM APL\MTS, WHICH IS A PORT OF A VERY EARLY APL\360 IMPLEMENTATION TO THE MTS OPERATING SYSTEM. WHERE NECESSARY THESE WORKSPACES HAVE BEEN ADAPTED TO MATCH THE LANGUAGE LEVEL OF APL\360.

THE WORKSPACES IN LIBRARY 1 ARE PART OF THE APL\360 DISTRIBUTION, WHICH ISN'T AVAILABLE ANYMORE TODAY. THEY HAVE BEEN ADDITIONALLY ADAPTED TO BEHAVE AS CLOSE AS POSSIBLE AS DESCRIBED IN PART 4 AND APPENDIX B OF THE APL\360 USER'S MANUAL. IN PARTICULAR, WORKSPACES ADVANCEDEX, PLOTFORMAT, AND WSFNS CAN BE ASSUMED TO BE FULLY IDENTICAL TO THE ONES PROVIDED WITH THE ORIGINAL APL\360 DISTRIBUTION.

LIBRARY 999 CONTAINS WORKSPACES THAT ARE RELEVANT TO APL\MTS ONLY, AND AS SUCH ARE MORE OF HISTORICAL THAN OF PRACTICAL INTEREST:

- '999 TESTS' HAS A FUNCTION TO DISPLAY THE FULL CHARACTERSET WHICH PROBABLY WAS USED TO SUPPORT PORTING APL\360 TO MTS AND TO CREATE THE transliteration scheme that supported output to non APL capable printers (WHICH HAPPENED TO BE THE ENABLER FOR BACKPORTING THE WORKSPACES TO THE PRESENT APL\360 SYSTEM).
- '999 NEWSAPLMTS' IS THE ORIGINAL WORKSPACE '1 NEWS' OF APL\MTS. IT CONTAINS MUCH INFORMATION ABOUT THE STATUS OF THE IMPLEMENTATION AND ITS OPERATION FROM 1969 UNTIL 1980.

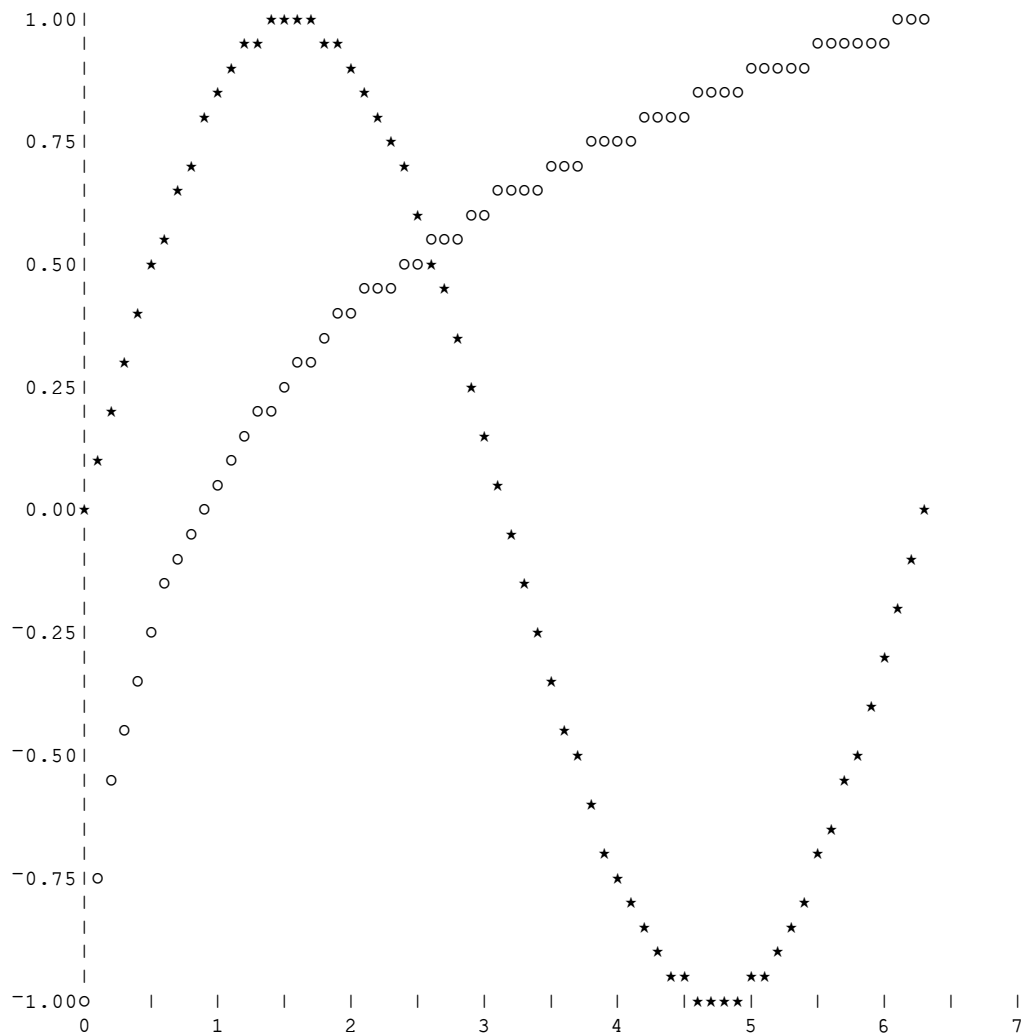
LIBRARY 100 CONTAINS WORKSPACES THAT WERE USED PRIMARILY FOR BASIC FUNCTIONALITY AND TERMINAL BEHAVIOR VERIFICATION DURING THE IMPLEMENTATION OF THE PRESENT APL\360 SYSTEM.

MOST WORKSPACES ARE DOCUMENTED INTERNALLY. THEIR FUNCTION CAN BE LEARNED BY LOADING THE WORKSPACE AND TYPING 'DESCRIBE' OR IN SOME CASES 'HOW'. THOSE NOT DOCUMENTED CONTAIN VARIED FUNCTIONS WHOSE DEFINITIONS MAY BE DISPLAYED TO DISCLOSE THEIR PURPOSE AND PROPERTIES.

```

)LOAD 1 PLOTFORMAT
SAVED 7.07.27 07/21/68
80 PLOT Q3 64pA, (-1+O÷r/(O←*1+6×A)÷2), 10A←((2×01)×-1+164)÷63

```



```

)OFF
006 14.14.37 04/09/13 PIL
CONNECTED 3.39.55 TO DATE 3.46.03
CPU TIME 0.00.00 TO DATE 0.00.00

```

This concludes the sample session. The recording terminal will report the *RIGI* and *PILATUS* sign off times. To terminate APL\360 press ctrl-a at the recording terminal, enter *SHUTDOWN*, press ctrl-a once again, then enter *)OFF* as shown below:

```
HHC01018I 0:0440 COMM: client 127.0.0.1 devtype 1052: connected
)314159
OPR) 10.11.34 04/09/85 OPR
      )LOAD OPFNS
SAVED 14.32.54 02/17/13
      SETDATE '04/09/13'
      )ADD 2132 PILATUS 50
      )ADD 1798 RIGI 50
006) 10.28.26 04/09/13 PILATUS 2132
007) 10.35.45 04/09/13 RIGI 1798
007 12.10.26 04/09/13 RIG
006 14.14.37 04/09/13 PIL
      SHUTDOWN
SYSTEM SHUTDOWN INITIATED, PLEASE SIGN OFF NOW
      )OFF
OPR 15.42.51 04/09/13 OPR
OPR 15.42.51 04/09/13 OPR
CONNECTED 5.31.17 TO DATE 957.27.26
CPU TIME 0.00.01 TO DATE 0.01.34
```

Operating System Information

This chapter provides information specific to the OS/360-MVT 21.8F operating system implementation being the base of the MVT for APL system. None of this information is needed to use and operate APL\360. Maintenance Tasks like dumping APL\360 workspaces to or restore them from tape, modifying the system configuration, etc. will, however, require interaction with the operating system.

Basic MVT or similar (for example MVS) and Hercules knowledge will be needed to use the information provided in this chapter, which is by no means intended to be an MVT tutorial. Bitsavers have a very comprehensive collection of MVT documentation which should be consulted for general operating system questions. It can be found at <http://bitsavers.org/pdf/ibm/360/os>.

Important Note: Each run of the APL\360 system is good for one IPL of MVT only. After APL\360 termination it is mandatory to shut down the MVT system.

When operating in automatic mode this is ensured by the fact that after an APL\360 shutdown the autopilot automatically brings the system down and terminates Hercules.

When operating in manual mode, however, you must make sure to always comply with that rule. **Failing to re-IPL MVT after an APL\360 shutdown may cause data integrity issues, particularly in the APL\360 library structure, and unexpected hang or race conditions.**

System Consoles

The MVT for APL system comes with two Hercules configurations:

- `mvt4apl/conf/mvt_intcons.cnf` has an integrated IBM 3215 console defined at 009. MVT console output gets displayed at the Hercules console, MVT console commands are issued from the Hercules console by prefixing them with a slash (/). Thus, to IPL this configuration, only the Hercules console is required. This configuration is used for automated operations.
- `mvt4apl/conf/mvt_extcons.cnf` has three external consoles defined: IBM 3270s at 010 and 011, and an IBM 3215 at 009. To IPL MVT using this configuration the desired consoles need to be connected to console group CONS at port 3600 using a tn3270 client for the IBM 3270s and a regular telnet client for the IBM 3215. Unused console ports must be detached before IPL to avoid MVT running out of WTO buffers. This configuration is used for manual operations.

Both Hercules configurations have an HTTP server listening at port 8360 allowing Hercules console communications even when running in daemon mode.

IPL Methods

Any IPL method valid for an OS/360-MVT system running on real iron is of course working with the MVT for APL system too. But for the specific use cases three methods are recommended:

Fully Automated APL\360 Operations

This is the method described in “Start the APL\360 System” on page 8. It is meant to enable a user to just work with APL\360 without even needing to know that there is Hercules and OS/360-MVT “under the hood”. There is no console input required neither at the Hercules nor at the MVT level. Console interaction is nonetheless possible using the HTTP server listening at port 8360.

Automated OS/360-MVT Operations without APL\360

This method provides a fully automated IPL of the MVT system in the same way as the “Fully Automated APL\360 Operations” does but it does not bring up APL\360. It is meant to be used for APL\360 maintenance and to run APL\360 utility subcommands that are not usable while APL\360 is running.

To use this method simply run the `mvt4apl\os360-mvt.bat` (Windows) or the `mvt4apl/os360-mvt` (Linux) script. This brings the system up to the point where TSO logons are enabled and HASP is ready to process batch jobs.

Console interaction is possible using the HTTP server listening at port 8360. To shut down the system enter

```
/f c3po,shutdown
```

in the “Command:” field of the Hercules System Log window.

Manual Operations

This method gives full control over IPL, operations, and shut down:

- Run the `mvt4apl\start_herc.bat` (Windows) or the `mvt4apl/start_herc` (Linux) script to start Hercules, which will present the Hercules console window.

- Create the desired MVT console configuration as described in “System Consoles” on page 24 for the mvt4apl/conf/mvt_extcons.cnf Hercules configuration.
- Enter `ipl 350` at the Hercules command prompt and press return.
- When message

IEA101A SPECIFY SYSTEM PARAMETERS FOR RELEASE 21.8F MVT

is displayed on the master console press enter.

- If you have defined IBM 3270 consoles switch them to roll mode: Enter `K S` on each one and accept the proposed set of parameters by pressing enter a second time.
- Now reply to message

*00 IEE114A DATE=yy.ddd,CLOCK=hh.mm.ss – REPLY WITH SET PARAMETERS OR U

as follows:

- `r 00,u` to complete IPL and start APL\360. This is equivalent to “Fully Automated APL\360 Operations” and consequently, when shutting down APL\360, the APLPILOT will perform an orderly MVT shutdown.
- `r 00,auto=ny` to complete IPL without starting APL\360. This is equivalent to “Automated OS/360-MVT Operations without APL\360”, starting TCAM, TSO and HASP but not APL\360. To shut down enter `f c3po`, shutdown at an MVT console.
- `r 00,auto=none` to just complete IPL. Neither the APLPILOT nor any tasks will be started. You may manually enter
 - `S HASP` to start HASP (followed by `r nn, ,noreq` to warm start and `r nn,u` to get the STCWTR OS writer up).
 - `S TCAM` to initialize TCAM (alternatively procedure names TCAM3335 or TCAM2741 can be used, see “System, TCAM and HASP” on page 28 for details).
 - `S TSO` to start TSO (requires TCAM to be started first).

To shut down a system brought up using `auto=none` it is recommendable to manually stop all services that were started in reverse order and enter `z eod`, once everything except the master scheduler has been brought down.

Regardless how the system was brought up and shut down, there will be no automatic Hercules “power off” in manual mode. After having shut down the MVT system enter at the Hercules console `stopall`, then `sysclear`, and finally `quit` to terminate Hercules.

TSO Access

The most convenient way to work with the MVT operating system is through TSO terminal sessions using IBM 3270 terminals emulated by a tn3270 client. Most of the many free and commercial tn3270 clients being available will work with TSO on OS/360-MVT 21.8F. The author recommends Tom Brennan’s Vista 3270 (Windows only, commercial) or the c3270/x3270/wc3270 family of clients (free, Windows and Linux).

The following TSO users are defined on the MVT for APL system:

- **APMAINT:** The APL\360 maintenance user. All APL\360 related work should be done using the APMAINT user.
- **HERC01, HERC02, HERC03 and HERC04:** General purpose users.

- IBMUSER: IBM default user.

These users are defined without passwords, all have the OPER, ACCT and JCL attributes assigned.

Logon

Connect a tn3270 client to port 3600 of the system running MVT for APL (which usually will be localhost). The following logo will be displayed:

```

Hercules Version : 4.00
Host name       : HPM9180
Host OS        : Windows-6.1.7601 7 Enterprise Edition 64-bit
Host Architecture : Intel(R) x64
Processors     : LP=4, Cores=4, CPUs=1
LPAR Name      : HERCULES
Device number   : 0:00C0
Subchannel     : 0015

      /_/_/ !/_/_/ /_/_/ /_/_/
     /_/_/ !/_/_/ !/_/_/ /_/_/
    /_/_/ !/_/_/ !/_/_/ /_/_/
   /_/_/ !/_/_/ !/_/_/ /_/_/
  /_/_/ !/_/_/ !/_/_/ /_/_/
 /_/_/ !/_/_/ !/_/_/ /_/_/
/_/_/ !/_/_/ !/_/_/ /_/_/

                                A P L \ 3 6 0

                                Version 1
                                Modification Level 1
                                5734-XM6 (c) Copyright IBM Corp
                                1969, 1970, 1972

OS/360-MVT 21.8F

Hercules
The S/370, ESA/390 and z/Architecture
Emulator

      /_/_/ !/_/_/ /_/_/ /_/_/
     /_/_/ !/_/_/ !/_/_/ /_/_/
    /_/_/ !/_/_/ !/_/_/ /_/_/
   /_/_/ !/_/_/ !/_/_/ /_/_/
  /_/_/ !/_/_/ !/_/_/ /_/_/
 /_/_/ !/_/_/ !/_/_/ /_/_/
/_/_/ !/_/_/ !/_/_/ /_/_/

```

Press the 3270 Clear key, then overwrite message

IKJ54012A ENTER LOGON -

with

logon user

where user is one of the users listed above. Press the 3270 Erase EOF key to delete the rest of the IKJ54012A message, then press enter. After a successful logon the screen will look like this:

logon herc01

```

HERC01 LOGON IN PROGRESS AT 21:50:35 ON APRIL 9, 1985
NO BROADCAST MESSAGES
READY

```

The colors will of course vary according to the mapping defined in your tn3270 client for the 3270 protection and highlighting attributes.

Logoff

Enter logoff at the TSO READY prompt to logoff the session. After a successful logoff the screen will look like this:

```
IKJ54012A ENTER LOGON -  
READY  
logoff  
HERC01 LOGGED OFF TSO AT 22:20:21 ON APRIL 9, 1985+
```

Message IKJ54012A will always be displayed at the top row of the screen, regardless at which row the cursor was positioned at logoff. It is mandatory to press the 3270 Clear key before attempting to logon again.

Useful Tools

Given the very limited native functionality of the OS/360-MVT TSO component its usability greatly depends on the third party tools available. Kevin Leonard did a great job in retrofitting some well-known tools from the MVS CBT tape to MVT. The following tools are installed on the MVT for APL system to enhance TSO usability:

- COMPARE: Comprehensive utility to compare sequential datasets and PDS members. Enter help compare for details.
- REVIEW: 3270 full screen browser for sequential datasets and PDS members. Only basic browsing functionality is implemented, the edit functionality of today's versions is not available. Given that MVT's TSO doesn't natively provide an easy method to browse files REVIEW is an important enhancement. There is no TSO HELP member for REVIEW, but basic usage help can be obtained by pressing PF1 once a dataset or member has been opened.
- PDS: The "Swiss Army Knife" for PDS handling. Although, as compared to today's versions, basic functionality is implemented only, its concept of calling external functions (including REVIEW and COMPARE) from a command line makes it a veritable "shell" for working with PDS members. Enter help pds at the TSO READY prompt or simply help from the PDS command line for details.

Important Datasets

The MVT for APL base OS/360-MVT system has been generated using Jay Maynards cookbook style instructions as found on <http://www.conmicro.com/hercos360>. The resulting system has been enhanced by installing programs and usermods provided by Kevin Leonard on <http://www.i76.org/mvt>.

The system and TCAM generation decks as well as any modifications and additions making up the final MVT for APL system have been stored on the system itself with the exception of Kevin's program and usermods, which can be found in distribution folder software/os360mvt.

System, TCAM and HASP

The system and TCAM generation decks as well as a small HASP modification can be found in SYS1.SYSGEN.CNTL as follows:

```
SYS1.SYSGEN.CNTL ----- 9 MEMBERS ON MVTRES
Command ==>                               Scroll ==> CS
  NAME      TTR    VV.MM  CREATED      CHANGED      INIT  SIZE  MOD  ID
. HASPMOD    000004  01.00  13-03-09  13-03-09  16:45:00   42   42   0  JUERGEN
. STAGE1     000006  01.00  13-03-06  13-03-06  20:38:00  254  254   0  JUERGEN
. STAGE2     000204  01.00  13-03-06  13-03-06  20:38:00 8505 8505   0  JUERGEN
. TC2741S1   004505  01.00  13-03-08  13-03-08  09:30:00   36   36   0  JUERGEN
. TC2741S2   004601  01.00  13-03-08  13-03-08  09:30:00  251  251   0  JUERGEN
. TC3270S1   004705  01.00  13-03-08  13-03-08  09:30:00   28   28   0  JUERGEN
. TC3270S2   004801  01.00  13-03-08  13-03-08  09:30:00  158  158   0  JUERGEN
. TC3335S1   004903  01.00  13-03-08  13-03-08  09:30:00   32   32   0  JUERGEN
. TC3335S2   004905  01.00  13-03-08  13-03-08  09:30:00  204  204   0  JUERGEN
**END**      004A04      1985-03-10 MVTRES          9510  IBMOS/360
```

- HASPMOD modifies the HASP printer initialization, allowing OS to “own” its hardcopy log and STC/TSO printers. Additionally it starts an OS writer to print off all STC/TSO output.
- STAGE1 is the stage 1 system generation deck, the “system configuration”.
- STAGE2 is the stage 2 system generation deck, the output of the STAGE1 job.
- TCxxxxS1 is a TCAM stage 1 deck, representing a specific TCAM configuration.
- TCxxxxS2 is a TCAM stage 2 deck, the output of the corresponding TCxxxxS1 job edited to use compliant assembler language and comprehensive parameters, and fitted with matching JCL.

Three different TCAM configurations have been generated, matching the xxxx in the above jobnames:

- xxxx = 3270: This is the default configuration, generated as load module IEDQTCAM and executed by starting the TCAM cataloged procedure. It enables eight local IBM 3270 model 2 devices at addresses 0C0-0C7.
- xxxx = 3335: This configuration is generated as load module TCAM3335 and executed by starting the TCAM3335 cataloged procedure. In addition to the default procedure’s eight local IBM 3270 terminals it enables eight dial-up 3335 (TWX) terminals at addresses 420-423 and 4A0-4A3, attached through an IBM 3705 transmission control unit.
- xxxx = 2741: This configuration is generated as load module TCAM2741 and executed by starting the TCAM2741 cataloged procedure. In addition to the eight local IBM 3270 terminals and eight dial-up 3335 (TWX) terminals supported by the TCAM3335 configuration it enables eight IBM 2741 dial-up terminals at addresses 400-403 and 480-483, attached through an IBM 3705 transmission control unit.

It should be noted that the TCAM2741 procedure allocates the same IBM 2741 terminal addresses as does APL\360. So, the TCAM2741 procedure must not be started while APL\360 is running.

MVT for APL: Specific Additions

The source for the autopilot (APLPILOT) and the balloon (a memory placeholder to enforce the APL\360 region being loaded at a compatible location) can be found in SYS1.SETUP.ASM as follows:

```
SYS1.SETUP.ASM ----- 4 MEMBERS ON MVTRES
Command ==>                               Scroll ==> CS
  NAME      TTR    VV.MM  CREATED      CHANGED      INIT  SIZE  MOD  ID
. APLPILOT  000007  01.00  13-03-19  13-03-19  21:40:00  131  131   0  JUERGEN
. BALLOON   000103  01.00  13-01-07  13-01-07  09:22:00   45   45   0  JUERGEN
. R2D2      000106  01.00  13-03-19  13-03-19  21:40:00  318  318   0  JUERGEN
. WAIT4APL  000305  01.00  13-03-19  13-03-19  21:40:00   69   69   0  JUERGEN
**END**     000402      1985-03-17 MVTRES                563 IBMOS/360
```

The jobs to generate the above tools can be found in SYS1.SETUP.ASM as follows:

```
SYS1.SETUP.CNTL ----- 2 MEMBERS ON MVTRES
Command ==>                               Scroll ==> CS
  NAME      TTR    VV.MM  CREATED      CHANGED      INIT  SIZE  MOD  ID
. APLPILOT  000007  01.00  13-03-19  13-03-19  21:40:00   51   51   0  JUERGEN
. BALLOON   00000A  01.00  13-01-07  13-01-07  10:47:00   23   23   0  JUERGEN
**END**     00000B      1985-03-17 MVTRES                74 IBMOS/360
```

See the comment headers of the respective members for details.

APL\360 Datasets

The following datasets are used by APL\360:

APLOS.ASM:	The assembler source members are placed in this PDS during installation.
APLOS.MACLIB:	The assembler macros are placed in this PDS during installation.
APLOS.LINKLIB:	The assembled source members are linked into this library during installation.
APLOS.LOAD:	The APL\360 runtime and utility load modules are linked into this loadlib during installation.
APL.SWAP:	Inactive workspaces are swapped out to this dataset when more than three users (including the recording terminal) are signed on.
APL.LIB.ZERO:	APL\360 library extend.
APL.LIB.ONE:	APL\360 library extend.
APL.LIB.TWO:	APL\360 library extend.
APL.LIB.THREE:	APL\360 library extend.
APL.LIB.FOUR:	APL\360 library extend.
APMAINT.ASM:	Local assembler source. MVT for APL comes with an APLSCONF member describing the APL\360 configuration in this PDS.
APMAINT.LINKLIB:	Local module library. APLSCONF will be assembled and linked into this library during installation.
APMAINT.CNTL:	This PDS contains all jobs needed to install APL\360 and to operate the APL\360 utility program. See below for a member list.

The jobs to install APL\360 and to operate the APL\360 utility program can be found in APMAINT.CNTL as follows:

```
APMAINT.CNTL ----- 17 MEMBERS ON HERC01
Command ==>
NAME      TTR    VV.MM  CREATED      CHANGED      INIT  SIZE  MOD  ID
. @1APLSRC 000007 01.00 13-03-22 13-03-22 14:09:00 65   65   0 JUERGEN
. @2MKLINK 000101 01.00 12-11-11 12-11-11 17:09:00 87   87   0 JUERGEN
. @3MKLOAD 000104 01.00 12-11-11 12-11-11 17:09:00 25   25   0 JUERGEN
. @4CONFIG 000106 01.00 13-03-22 13-03-22 14:09:00 15   15   0 JUERGEN
. @5APLGEN 000201 01.00 12-11-11 12-11-11 17:09:00 95   95   0 JUERGEN
. @6LNKSVC 000204 01.00 12-11-09 12-11-09 21:45:00 33   33   0 JUERGEN
. @7MKLIB 000301 01.00 13-03-22 13-03-22 14:09:00 54   54   0 JUERGEN
. @8CREATE 000303 01.00 13-03-22 13-03-22 15:10:00 33   33   0 JUERGEN
. ACCTG 000305 01.00 13-03-22 13-03-22 12:16:00 29   29   0 JUERGEN
. CREATE 000307 01.00 13-03-22 13-03-22 12:16:00 31   31   0 JUERGEN
. DUMP 000401 01.00 13-03-22 13-03-22 12:16:00 30   30   0 JUERGEN
. INC DUMP 000403 01.00 13-03-22 13-03-22 12:16:00 30   30   0 JUERGEN
. RESTORE 000405 01.00 13-03-22 13-03-22 12:16:00 30   30   0 JUERGEN
. RETRIEVE 000407 01.00 13-03-22 13-03-22 12:16:00 32   32   0 JUERGEN
. SELDUMP 000501 01.00 13-03-22 13-03-22 12:16:00 32   32   0 JUERGEN
. SELREST 000503 01.00 13-03-22 13-03-22 12:16:00 30   30   0 JUERGEN
. TVERIFY 000505 01.00 13-03-22 13-03-22 12:16:00 30   30   0 JUERGEN
**END** 000506      1984-11-11 HERC01      681 IBMOS/360
```

- The jobs named @1APLSRC to @8CREATE perform the full APL\360 installation from source when run in ascending numeric sequence. To run @1APLSRC the complete source deck as downloaded from the Computer History Museum must be placed in the card reader at 00C, to run @8CREATE a valid APL\360 utility full dump tape must be mounted on 280 upon request.

Except for @1APLSRC these jobs are restartable to allow partial APL\360 generation: For example to create a new configuration just edit APLSCONF in APMAINT.ASM, then run DUMP (it's always good to have a backup!), @4CONFIG, @5APLGEN, @7MKLIB, and @8CREATE in that sequence and mount the tape created by DUMP when @8CREATE requests it.

All @nxxxxxx jobs have TYPRUN=HOLD coded on their job cards to prevent system destruction when submitting them inadvertently.

- All other jobs execute the APL\360 utility program's subcommand matching their member name. Except for the ACCTG, SELDUMP and TVERIFY jobs APL\360 must not be running when executing these jobs. It is recommended to stop APL\360 and re-IPL OS/360-MVT without bringing APL\360 up before executing any of these jobs (except ACCTG, SELDUMP and TVERIFY).

There isn't much documentation on the APL\360 Version 1 Modification Level 1 versions of the utility subcommands available any more. The APL\360 General Information Manual (GH20-0850-1) has a complete list of the subcommands without specific usage information. The documentation of the older Type III contributed APL\360 program (360D-03.3.007) has a more detailed description of the then available subcommands. Both manuals can be found on bitsavers.

To Do

- Provide an authentic look and feel (“visual fidelity”) on 2741 terminals (low priority).
- Consolidate 1052 and 2741 behavior to provide a more seamless user experience (low priority).
- Optimize 1052 coding to use tables instead of nested if--else constructs (very low priority).

Appendix A: Acknowledgments

- Len Shustek, Chairman of the Board of Trustees, Computer History Museum, Mountain View, went through 10 years of persistent negotiations with IBM to make the impossible happen: Obtain a license to make the APL\360 source code available to the public for non-commercial use.
- Brian and Barry Silverman shared many insights they gained during their 1998 APL\360 resurrection project, especially on how to handcraft a library structure on DASD. This greatly helped overcome the chicken-egg problem of how to create an initial library setup without having the original distribution tape available.
- Catherine Lathwell spread the word when the APL\360 source became available and provided the connection to Len Shustek.
- Max H. Parke provided the IBM 2741 terminal support: Having created the Hercules device driver for the 2741 and other asynchronous terminals attached to an IBM 2703 TCU, Max is most knowledgeable when it comes to implementing emulations of complex communications equipment and topologies. With the availability of APL\360 running on OS/360-MVT under Hercules he added the functionality and features required by APL\360 to the 2741 device support to provide a true generic implementation of the most commonly used APL terminal back in the APL\360 era.
- Tony Harminc made information about APL internal structures (workspaces, libraries, control blocks) available that was very helpful in creating the initial library setup.
- Jay Maynard's cookbook style MVT system generation instructions enabled getting up to speed with OS/360-MVT quickly.
- Kevin Leonard provides a lot of tools (namely PDS, COMPARE and REVIEW) and usability enhancements for OS/360-MVT on his website that helped transforming the MVT system initially meant as a runtime only environment into a veritable development system.
- Phil Roberts extracted the public library workspaces from APL\MTS. This was a major effort including installation of MTS and APL\MTS, exploring how to execute APL\MTS and how to print output, unlocking hidden functions, putting together some 4,000 APL statements to extract and finally print the full contents of the workspaces.

Appendix B: About the MVT for APL distribution

About the APL\360 Software

APL\360 Version 1 Modification Level 1 is

Program Number 5734-XM6 © Copyright IBM Corp 1969,1970,1972

The APL\360 source code has been licensed to the general public by the

Computer History Museum
1401 N. Shoreline Blvd
Mountain View, CA 94043

See <http://www.computerhistory.org/atcm/computer-history-museum-software-license-agreement> for license details.

APL\360 is written in Basic Assembler Language for IBM System/360 mainframes running under an OS/360 MFT or MVT operating system. It comprises 37404 lines of code.

The license doesn't allow redistribution of the source code or of any derived works (executables, etc.). For that reason the MVT for APL distribution doesn't contain any APL\360 licensed source or executables.

To use APL\360 the source must be downloaded using the above link (thereby accepting the license terms) prior to the first execution of the MVT for APL system. Upon first execution a fully automated procedure assembles the source and generates the executable APL\360 system.

About the Public Library Workspaces

Workspaces in libraries 1, 2, 3, 5, 6, and 999 have been created by adapting workspaces from APL\MTS to the APL\360 language level. APL\MTS is '*APL IBM's APL\360' on volumes 5.0T1 and 5.0T2 of the MTS distribution archive which is

Copyright © 2011 by the Regents of the University of Michigan
acting as agent for the MTS Consortium

and licensed under the Creative Commons Attribution 3.0 Unported License (CC BY 3.0).
See <http://creativecommons.org/licenses/by/3.0> for license details.

About the Hardware

The present APL\360 implementation runs on an IBM System/370 model 158 mainframe emulated by the Hercules System/370, ESA/390, and z/Architecture emulator which is

Copyrighted (c) by Roger Bowler and others.

Hercules is licensed and distributed under the terms of the Q Public License Version 1.0.
See <http://www.hercules-390.eu/herclib.html> for license details. According to the terms of the license an unmodified copy of the license text has been placed in folder doc and all patches applied to Hercules to adapt it to the communication and character set needs of APL\360 have been placed in folder software/Hercules of the MVT for APL distribution.

About the Operating System

APL\360 Program Number 5734-XM6 was designed to run under an OS/360 operating system. For the present implementation the final version 21.8F of IBM's OS/360-MVT operating system was chosen, which is in the public domain.

About the MVT for APL distribution

The MVT for APL distribution is the glue that combines the many components making up an APL\360 system while fully adhering to the restrictive license conditions of the source code.

The OS/360-MVT setup, the automated procedure to build the APL\360 executables and the MVT supervisor calls from source, the automated MVT operator (APLPILOT), the IBM 1052-7 terminal support, the handcrafted initial library structure on DASD, the APL\MTS public workspace import and adaption, the minimal OPFNS workspace, the automated recording terminal operator, and the present User's Manual have been created by Juergen Winkelmann, who also put together the MVT for APL distribution.

The MVT for APL distribution itself and the components named in the above paragraph are put into the public domain without claiming any copyright by the author as far as no third party copyrights are affected.

