ABSTRACT
This paper describes a prototype SAS/GRAPH* driver for the HP LaserJet that was developed at HP's Corporate Computing Center. The paper also discusses several important issues that were uncovered during that development, including data communications between IBM and the LaserJet, vector-to-raster conversion, and selecting a protocol converter. These issues are presented in a manner designed to assist the reader in creating his own LaserJet driver. Finally, a few commercially available solutions are presented as well.

1. Introduction
Shortly after Hewlett-Packard Co. (HP) introduced the LaserJet* personal laser printer in 1984, many IBM mainframe sites decided the HP LaserJet, in combination with a low-cost protocol converter, would make an ideal local-area printer for their 3270 networks. It was faster, quieter, and less expensive than their current IBM 3287 and 3268 printers, and produced higher quality output. As a result, the HP LaserJet has found a happy home in many IBM mainframe shops around the world.

Besides being a high-quality text printer, the HP LaserJet is also capable of producing high-resolution monochrome graphics. Many PC programs already exploit this capability, using HP's Printer Control Language (PCL*) to send raster information to the printer, which the HP LaserJet loads into its internal memory and prints a page at a time.

It only makes sense, then, that the HP LaserJet could be an ideal device for SAS/GRAPH output. The same attributes that make the HP LaserJet such a good 3270 network printer and graphic device for PCs, combined with the spooling capabilities of the mainframe (thereby allowing unattended operation), make it a very good choice for SAS/GRAPH users who need monochrome graphics.

However, SAS Institute does not offer a SAS/GRAPH driver for the HP LaserJet in its mainframe product. HP's Corporate Computing Center in Palo Alto, CA has created a prototype driver for the LaserJet. This paper will describe some of the issues that went into the hardware selection and design of the driver software itself. It will also discuss some of the commercially available solutions for SAS/GRAPH graphics on the HP LaserJet.

2. Description of the Problem
Creating graphics on the LaserJet with SAS/GRAPH on an IBM mainframe isn't nearly as straightforward as with a PC. There are several problems which must be overcome first.

Connecting a piece of non-IBM equipment to an IBM mainframe and having them converse is the most obvious problem which must be overcome, but fortunately, is one of the easiest to solve.

Electrically speaking, the serial and parallel connections of the HP LaserJet cannot be connected directly to the IBM mainframe. Serial connections are sometimes available from a Front-End Processor (like an IBM 3705 or 3275), but these are not addressable by the spooler, so they wouldn't be suitable for this application. Some sort of electrical conversion must be made from the coaxial cables of a 3270 network to the serial or parallel interfaces of the HP LaserJet.

Besides the electrical differences, the IBM mainframe uses a different communications protocol than the HP LaserJet. IBM 3270 networks use a bisynchronous protocol, which requires carefully timed two-way "conversations" between the devices and the control unit. HP LaserJet printers use an asynchronous protocol. In order to make the two converse, one must convert the bisync protocol of the 3270 network into async for the LaserJet, and vice versa.

Additionally, IBM mainframes and HP LaserJets speak different languages. Conversions must be made between the EBCDIC binary codes used by the mainframes and the ASCII binary codes understood by the LaserJet, especially for text printing. To confound the problem, there are at least three "standard" EBCDIC-to-ASCII conversion standards in common use today.

Fortunately, all of these problems are easily solved by using a small, inexpensive protocol converter. A protocol converter will take care of the electrical incompatibilities between a 3270 network and serial (or parallel), convert the communications protocols, and convert EBCDIC data into ASCII. Protocol converter issues will be discussed later in this paper.

SAS/GRAPH, like most graphics software, creates its pictures by drawing them one line at a time on a graphics device. While this approach works well on vector-oriented devices like plotters and graphic terminals, it will not work for raster-oriented devices like the HP LaserJet.

For SAS/GRAPH, this means the vectors must be converted to rasters, requiring a vector-to-raster converter. These converters may be implemented in hardware, as in an outboard rasterizer, or in
software. Outboard rasterizers are devices which accept vector commands, such as HP-GL* or PLOTIO*, and output raster data. SAS/GRAF currently supports several outboard rasterizers.

Software rasterizers are programs which accept vector commands or subroutine calls and, instead of plotting them on a graphics device, plot them into a large area of memory, called a "bitmap." The dimensions of this bitmap correspond to the raster area of the intended graphics device. When a vector command is plotted into this bitmap, individual bits are turned an where the line would lie. When all the vectors are plotted, the bits in the bitmap will form an image of the picture in memory. To output the bitmap, the software needs to scan the bitmap one line at a time, writing to the output device or the output spool.

Depending on the resolution and graphic size, software rasterizers can be fairly expensive programs to run. They can consume a great deal of CPU and memory. On the other hand, outboard rasterizers are often very expensive pieces of hardware, and not very flexible at what they do.

Once all the vectors are rasterized (by whatever method), the rasters must be output to the LaserJet. However, even at 9600 baud, the data can take a long time to transmit. Assuming a graphics area of 8.5" x 11" and a resolution of 300 dpi, the raster information will amount to more than a megabyte of data. Transmitting that much data at 9600 baud will take more than 18 minutes. Increasing the speed to 19.2Kb will reduce that time somewhat; using a parallel protocol will reduce it a lot more.

Once the rasters are transmitted, they must be stored in the LaserJet's internal memory. Being a page printer, the HP LaserJet must store an entire page worth of raster information before printing it. The same calculations that were used to determine transmission time apply here; i.e. an 8.5" x 11" 300 dpi graphic will require a megabyte of memory on board the LaserJet. Not all LaserJet printers are capable of supporting that much memory.

3. Selecting a Protocol Converter
As mentioned before, a protocol converter will allow an HP LaserJet to be connected into an IBM 3270 network. However, not all protocol converters are alike. There are several issues which must be considered when selecting a protocol converter for use with an HP LaserJet.

A. Parallel vs. Serial. The HP LaserJet is capable of using either serial or parallel interfaces. If the parallel interface is chosen, data can move much more quickly from the protocol converter to the LaserJet than with the serial interface, especially when transferring raster data. In fact, a parallel connection can move raster data to the LaserJet almost as fast as the LaserJet can accept it. Tests have shown from 3 to 6 times improvement in speed when using a parallel instead of serial interface.

B. Actual Data Throughput. Just because a protocol converter is rated at 9600 baud doesn't mean it will actually output data at 960 bytes a second. Data throughput of a protocol converter is very dependent on the speed of the microprocessor inside.

C. "Cluster-Controller"-type Protocol Converters. Protocol converters which emulate cluster controllers, like the IBM 7171, may be a viable option for your installation. These protocol converters attach directly to the IBM channel and emulate a 3274 controller. Async ASCII devices like terminals and LaserJet printers may be attached to any of the serial ports on the controller.

Cluster controllers offer economies of scale that single-line protocol converters are unable to match. Their powerful processors also assure fast data throughput.

D. Ease of Setup and Configuration. Some protocol converters can be configured simply by attaching a terminal and following some simple menu-driven prompts. Others can be configured using front-panel buttons. Still others require removing the cover of the device and manipulating DIP switches inside the box. Select a protocol converter that you are confident can be configured and used by your users in remote locations.

E. Transparency Mode. One of the most important items to consider when selecting a protocol converter is "transparency mode." This feature allows a protocol converter to transmit certain data without converting it. Without this feature, none of the special features of the LaserJet, including graphics, can be accessed.

3270 network controllers cannot pass 8-bit binary data. With single-line protocol converters, any non-EBCDIC data must be passed to the protocol converter in "hex-pairs" (two hex characters per
byte of binary data), where the protocol converter will decode it and transmit the binary information to the LaserJet. Obviously, this doubles the amount of information that must be transmitted to the protocol converter.

Cluster controllers are directly attached to the IBM channel and are not themselves part of a "real" 3270 network. Thus they are capable of accepting and forwarding 8-bit binary data. A simple escape sequence tells the cluster controller that the following text is to be passed directly to the LaserJet with no modification. This method preserves the original binary information and cuts down on transmission times.

There are two ways to interface a locally written SAS/GRAPH driver to SAS: via the LINKABLE driver or the METAFILE driver. Both of these methods are supported by SAS Institute and documented in Institute documentation.

Creating raster data from vectors really isn't difficult. Simply define a large, two-dimensional array that can be accessed a bit at a time, and create subroutines that will "draw" the vectors into the bitmap. It helps if you tell SAS that the device has no intelligent features whatsoever, and is only capable of simple MOVES and DRAWS. This reduces the amount of intelligence the programmer must put into the driver software, thereby making the entire job much simpler.

To draw vectors into the bitmap, you need the beginning and ending coordinates of the line to be drawn. From them, you can calculate the slope of the line.

You may begin drawing the vector in the bitmap by turning on the bit corresponding to the initial point on the vector. From the initial point, calculate the remaining points on the vector using the linear equation described by the slope and beginning point. A very interesting algorithm, "Quadrantal DDA," is described in the book "Applied Concepts in Microcomputer Graphics," by Bruce A. Artwick (Prentice-Hall, 1984), which determines the points that will most accurately describe the line. This algorithm divides the graphic lines into four quadrants based on their slope. Depending on the quadrant, different methods are used in determining the raster points that describe the line. This helps to reduce error in the rasterization of the line and reduce the "jagged" appearance of the graphic.

After the bitmap is completed and all vectors are drawn, the driver may output the bitmap to the device or the output spool. At this point, the driver should include all the necessary characters to make sure the protocol converter is put into transparent mode and the LaserJet is put into graphics mode.

Before outputting the rasters, though, the driver should examine the bitmap for any efficiency enhancements that can be made. White space surrounding the drawing need not be transmitted, solid black areas can be compressed into single commands, and lines that are straight up-and-down or across the page can be transmitted as "rules" or simple line-drawing commands to the LaserJet.

All HP LaserJets include a Printer Control Language called PCL. These commands (graphics mode, cursor position, line draw, etc.) may be imbedded in the raster data transmitted to the printer. By following these suggestions, the driver can easily cut transmission times in half.

After the printer is in graphic mode and any simple PCL commands have been sent, the raster data may be transmitted. To do this, the driver must scan the bitmap in memory, one raster line at a time, gather all the bits into bytes, and transmit them to the printer.

Remember that only EBCDIC character data can be sent through 3270 networks, so if single-line protocol converters are used any 8-bit data will have to be converted into hex pairs. Channel-attached protocol converters (a la IBM 7171) will not have this problem.

5. Memories...
LaserJets come in all sizes and shapes, with varying amounts of memory. For best results, your LaserJet should have at least 1.5 megabytes of memory. The original LaserJets had only 65 Kb of memory, which was only sufficient for 75 dpi graphics on a full page, or approximately the resolution of most dot-matrix printers. The LaserJet Plus had 512 Kb of memory, sufficient for 150 dpi full-page graphics. HP later introduced memory upgrade boards for the LaserJet Plus, which gave up to 2 megabytes of additional memory, easily enough for 300 dpi graphics. The LaserJet Series II, while coming standard with only 512 Kb, can be easily and inexpensively upgraded to support additional memory. The LaserJet 2000 comes standard with enough memory for graphics.

6. Other Solutions
For the less adventuresome soul, there are several other SAS/GRAPH-LaserJet solutions available on the commercial market.
Maersk Data (25 Vreeland Road, Florham Park, NJ 07932) offers a package called MD-LASER, which converts GDDM ADMPRINT files into HP LaserJet raster data, downloads the raster data via an IDEX protocol converter, and produces an image on the LaserJet. This, of course, requires the site to have licensed and installed GDOM as well as SAS/GRAPH, and specifically requires IDEX protocol converters. This is not a cheap solution, but does provide a turnkey system for graphics on the LaserJet.

Teknigraphics (4635 SW Freeway #510, Houston, TX 77027) sells a series of PC software products which allow IBM personal computers and compatibles equipped with a 3270 emulation board to produce on-screen graphics as well as several hard-copy options, including the LaserJet. This is a relatively inexpensive option, but doesn’t allow the LaserJet to be shared with other SAS/GRAPH users. Also, the series of products based on Tektronix terminal emulation only allows a resolution of 75 or 150 dpi on the LaserJet (their brand-new GDDM-based products allow 300 dpi).

Insight Development (1024 Country Club Drive, Menlo Park, CA 94556) has developed the LaserPlotter software product, which can take an HP-GL file, such as might be captured using one of the Teknigraphics products, and convert it into a PCL raster file suitable for printing on the LaserJet. Its advantage over the Tektronix-based Teknigraphics solution is that it can produce raster data at full 300 dpi resolution.

SAS Institute (Box 8000, Cary, NC 27512) is finally marketing SAS/GRAPH for the PC. Using their GRLINK software, GREPLAY libraries in the PC format can be created on the mainframe and downloaded to the PC, where the PC SAS/GRAPH LaserJet driver can output the graphics to the LaserJet.

There are many LaserJet upgrade products on the market today, including JetScript from Laser Connection (7852 Schillinger Park West, Mobile, AL 36608) and PC Publishers Kit from Imagen (2650 San Tomas Expwy, Santa Clara, CA 95051) which consist of a high-speed video interface board in the optional I/O slot of the LaserJet Series II, and another board in the PC which actually does the post-processing and rasterization of PostScript and/or DDL data. Obviously, both of these solutions require a file to be downloaded to the PC and then post-processed offline. This fact makes it difficult to share a LaserJet among several SAS/GRAPH users.

Both manufacturers also manufacture boards which transform older LaserJets into QMS or Imagen printers. When this is done, however, the printer loses its "HP Personality," including the ability to understand PCL. This may make the printer incompatible with many PC-based applications.

7. Missing Products
In the authors opinion, there are some great opportunities waiting out there for the right company to pick them up. These "missing products" include:

An HP-GL converter capability onboard the LaserJet itself. Adding vector capability to the LaserJet would greatly increase its value to graphics users, who may not need the full utility of a Page Description Language like PostScript.

For those who do need a full PDL like PostScript, there should be a PostScript interpreter/post-processor that fits in the optional I/O slot of the LaserJet, rather than putting that function on a board inside a PC.

If it proves impossible to develop a PostScript interpreter for inside the LaserJet, I would settle for one that lived outside the LaserJet. An outboard PostScript interpreter would also satisfy this issue.

And finally (this is a good one), somebody should manufacture a protocol converter that fits in the optional I/O slot of the LaserJet, rather than putting that function on a board inside a PC.

8. Conclusions.
The intent of this paper has been to provide some information and findings regarding the myriad of issues that surround connecting the HP LaserJet to IBM equipment, and then to make it work with SAS/GRAPH.

In doing so, this paper has discussed many of the technical issues like data communications, different types of protocol converters, all the different choices for vector-to-raster conversion, efficiency considerations, and finally some of the commercially available solutions.

While HP’s Corporate Computing Center cannot offer copies of its prototype LaserJet driver (due to legal and warranty implications) it is hoped that the information provided in this paper will assist others in drafting their own LaserJet drivers (and if somebody does it, I want to know about it!).
9. Acknowledgements.
The author would like to acknowledge the efforts and contributions of two members of HP's Corporate Computing Center staff: Louis Crowder, who did most of the programming effort on the prototype driver, and Stuart Stevens, who provided much technical information and assistance with networking and protocol converter issues.

10. Author, Author....
The author may be contacted via electronic mail at:

DCASE%MATTEC%sc.intel.com@RELAY.CS.NET

Or if that doesn’t make any sense, you can always try the good old USPS:

Dennis Case
Intel Corporation
2250 Mission College Blvd SC9-06
Santa Clara, CA 95051

11. Trademarks.
SAS and SAS/GRAPH are registered trademarks of SAS Institute, Cary, NC, USA.


MD-LASER is a registered trademark of Maersk Data, Florham Park, NJ, USA.

JetScript is a registered trademark of QMS and Laser Connection, Mobile, AL, USA.

PC Publishers Kit is a registered trademark of Imagen Corp., Santa Clara, CA, USA.

LaserPlotter is a registered trademark of Insight Development, Menlo Park, OR, USA.

PLOT10 is a registered trademark of Tektronix Inc., Beaverton, OR, USA.