

SAS® Viewer giving way to Universal Viewer

Steve Wright, Quintiles, RTP, NC

ABSTRACT

The SAS® Viewer has been a useful, free, but increasingly limited tool. It is free. It has bugs and limitations. Starting with SAS9, this tool often will be unable to access the default data tables. The Universal Viewer is a fresh take on a tool to fit this niche. Everyone will find that a number of the things this tools does are just right. It does not lock datasets. When filtering data, it will not return the converse when using missing values in the logic clause. To make your first impression to be the best it can be, you will want to be aware of a surprise or two.

INTRODUCTION

The Universal Viewer (UV) is a totally new application. It handles and presents data differently. It is a library or directory oriented tool whereas its older sibling, SAS Viewer (SV), focused only on individual datasets. The filtering process is data aware and largely menu driven.

YES! I don't have to spend hours trying to figure out who has the file opened and locked which I must update in the next 5 minutes! But do remember, you need to get updated data in UV when it has been recreated. The old data is still there in the UV workspace; it is not automatically updated.

The old application took observation limits lightly, searching all records when the filter was updated or a sort selected, but only displaying the maximum rows allowed. The new tool is dead serious. If you specify to only use x records, that is all you get. Each sort or filter applies only to the range of rows that you are currently accessing.

In this paper, I will elaborate on the similarities and differences between these two tools. UV has areas of real merit, things that are missing, and places where some beautification is needed. This paper is based on the first real release of UV, available earlier this year. The prior 'technology preview' had a number of bugs that were fixed in this release. The paper in the published proceedings was based on the earlier version. All my comments are based on a Windows **data** platform. Note that all example datasets, except the datetime example, are copies from the SASHELP files shipped with SAS.

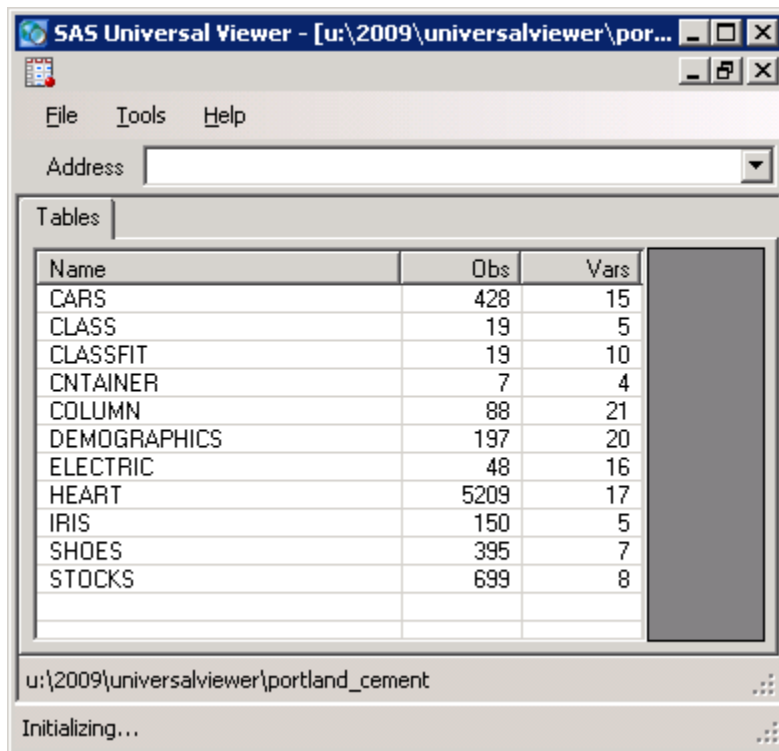
THE FINEST ASPECT OF UNIVERSAL VIEWER

One issue that has caused more lost production time than anything else with the old SV is locked datasets. If someone had a file open in SV and SAS tried writing to that file, you get an error in SAS. At this point, you have to figure out if you have the file open, or someone else on your team. You hope that person is not at lunch or in a meeting. When you figure out whom the person is SV had to be shut down. I have seen times when a team waited until the next day, after everyone had logged out, to be able to access a file. The crux of the issue is that unlike some other Windows applications, SV does not in any way tag the file so that you can determine who has the file locked.

Now with UV, data files are not locked. This means that data creators and data gazers will no longer lock horns over a file locked by someone 'just' viewing it. This does not sound too exciting except in light of the issues SV creates.

FORMAL INTRODUCTION OF THE UNIVERSAL VIEWER

When you open the first dataset with UV, you probably will be surprised to find that it brought all its pals with it. Not only did it pull in the Demography file, but it also opens everything in the same directory! This is nice if you want to look around. It can be problematic if you have a lot of data files in this directory, and you just want to take a quick peek at a single file.



When you bring a file into UV, all data files in the directory get scanned, and the metadata gets stored. However, the file is not opened until requested. Now if the length, width, or content of the file changes between "bringing it in" when the whole directory of files is opened, and when you actually open it, the file might be different than the metadata describes it, because the meta data is only read once.

If the number of obs has changed, it just opens the file with the actual number of records, up to the previously recorded number of rows. The metadata still has the old number of records listed. But if any variable characteristic or any other dataset characteristic has been changed it gives you an error. That message is as follows: "Item cannot be found in the collection corresponding to the requested name or ordinal." At this point, close UV or the window containing your directory of files, and reopen that group of files.

MORE ABOUT METADATA

With a single click on any dataset, the 'contents' of that file pops up. Very standard information is presented, and you can sort on the metadata columns! The sort only requires you to click the column head. If you click a second time, the sort is inverted. The sort is focused only on a single column. It always refers back to the original order ('varnum') for each sort. For instance, sorting on Type, then Length, does not accomplish anything different than sorting on just Length. All variables of length x are sorted in the original 'varnum' order with no regard to Type.

| Name | Obs | Vars | # | Variable | Type | Length | Format | Informat | Label |
|--------------|------|------|---|------------|-----------|--------|--------|----------|---------------|
| CARS | 428 | | 1 | Status | Character | 5 | | | |
| CLASS | 19 | | 2 | DeathCause | Character | 26 | | | Cause of D... |
| CLASSFIT | 19 | | 3 | AgeCHDdiag | Numeric | 8 | | | Age CHD ... |
| CNTAINER | 7 | | 4 | Sex | Character | 6 | | | |
| COLUMN | 88 | | 5 | AgeAtStart | Numeric | 8 | | | Age at Start |
| DEMOGRAPHICS | 197 | | 6 | Height | Numeric | 8 | | | |
| ELECTRIC | 48 | | 7 | Weight | Numeric | 8 | | | |
| HEART | 5209 | | | | | | | | |
| IRIS | 150 | | | | | | | | |
| SHOES | 395 | | | | | | | | |

One annoyance you might notice is if you click in the right window, the dataset name on the left is no longer highlighted, or anyway identified by UV. If you click back on any heading on the left side, the file in question is again highlighted. Compare the left panel in the graphic above and below. Several other changes have also been made.

| Name | Obs | Vars | # | Variable | Type | Length | Format | Informat | Label |
|--------------|------|------|---|------------|---------|--------|--------|----------|-------------------|
| CLASSFIT | 19 | 10 | 1 | Status | Char... | 5 | | | |
| CNTAINER | 7 | 4 | 2 | DeathCause | Char... | 26 | | | Cause of Death |
| COLUMN | 88 | 21 | 3 | AgeCHDdiag | Num... | 8 | | | Age CHD Diagnosed |
| DEMOGRAPHICS | 197 | 20 | 4 | Sex | Char... | 6 | | | |
| ELECTRIC | 48 | 16 | 5 | AgeAtStart | Num... | 8 | | | Age at Start |
| HEART | 5209 | 17 | | | | | | | |
| IRIS | 150 | 5 | | | | | | | |
| SHOES | 395 | 7 | | | | | | | |

You can change the column width by dragging the separators, but there seems to be no way to change the relative size of the left and right metadata windows.

HANGING OUT WITH DATA

Opening a data file is just a double click away. You may open several files and move between them using the tabs along the top. When you first open a dataset in UV, all columns are the same width, and do not change width with the width of the overall window. Note that in the metadata view, the right panel adjusts the column width with the size of the overall window. This was probably easy for the developers to do, and desirable, since the 'contents' information contains only the 7 columns.

| | Make | Model | Type | Origin | Weight | DriveTrain | MSRP |
|-------|---------|---------------------|-------|--------|--------|------------|----------|
| 390 | Nissan | Quest S | Sedan | Asia | 4012 | Front | \$24,780 |
| 391 | Toyota | 4Runner SR5 V6 | SUV | Asia | 4035 | Front | \$27,710 |
| 392 | Toyota | Sienna CE | Sedan | Asia | 4120 | Front | \$23,495 |
| 393 | Toyota | Sienna XLE Limit... | Sedan | Asia | 4165 | Front | \$28,800 |
| 394 | Nissan | Quest SE | Sedan | Asia | 4175 | Front | \$32,780 |
| ▶ 395 | Pontiac | Montana EWB | Sedan | USA | 4431 | All | \$31,370 |
| 396 | Toyota | Tundra Access ... | Truck | Asia | 4435 | All | \$25,935 |
| 397 | Porsche | Cayenne S | SUV | Europe | 4950 | All | \$56,665 |
| 398 | Toyota | Sequoia SR5 | SUV | Asia | 5270 | All | \$35,695 |
| 399 | Nissan | Titan King Cab XE | Truck | Asia | 5287 | All | \$26,650 |
| 400 | Toyota | Land Cruiser | SUV | Asia | 5390 | All | \$54,765 |

Rows 301-400 of 428 | Filter: none | Sort: Weight | 13,95

u:\2009\universalviewer\portland_cement

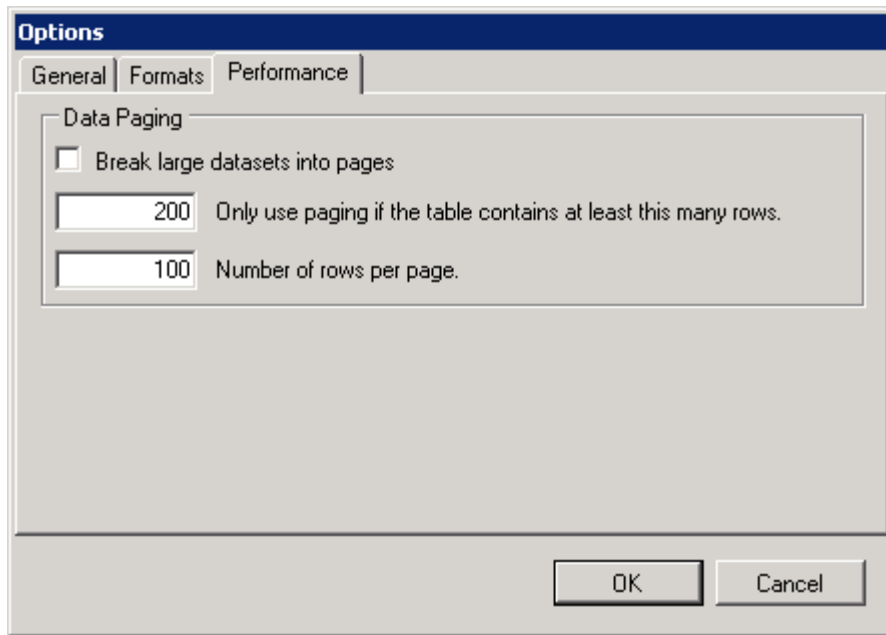
Initializing...

In the above screen shot, notice several things. First note, from the information just below the data, that this dataset is divided into pages of 100 rows each. If the window is long enough, the row numbers are displayed as here (Rows 301-400 of 428). If you shrink the height of this window a little, the row information disappears (see later CLASSFIT example). In this 'small' size, the scroll bars partially or completely disappear. Hopefully that will make it into the list of beautifications for the next release.

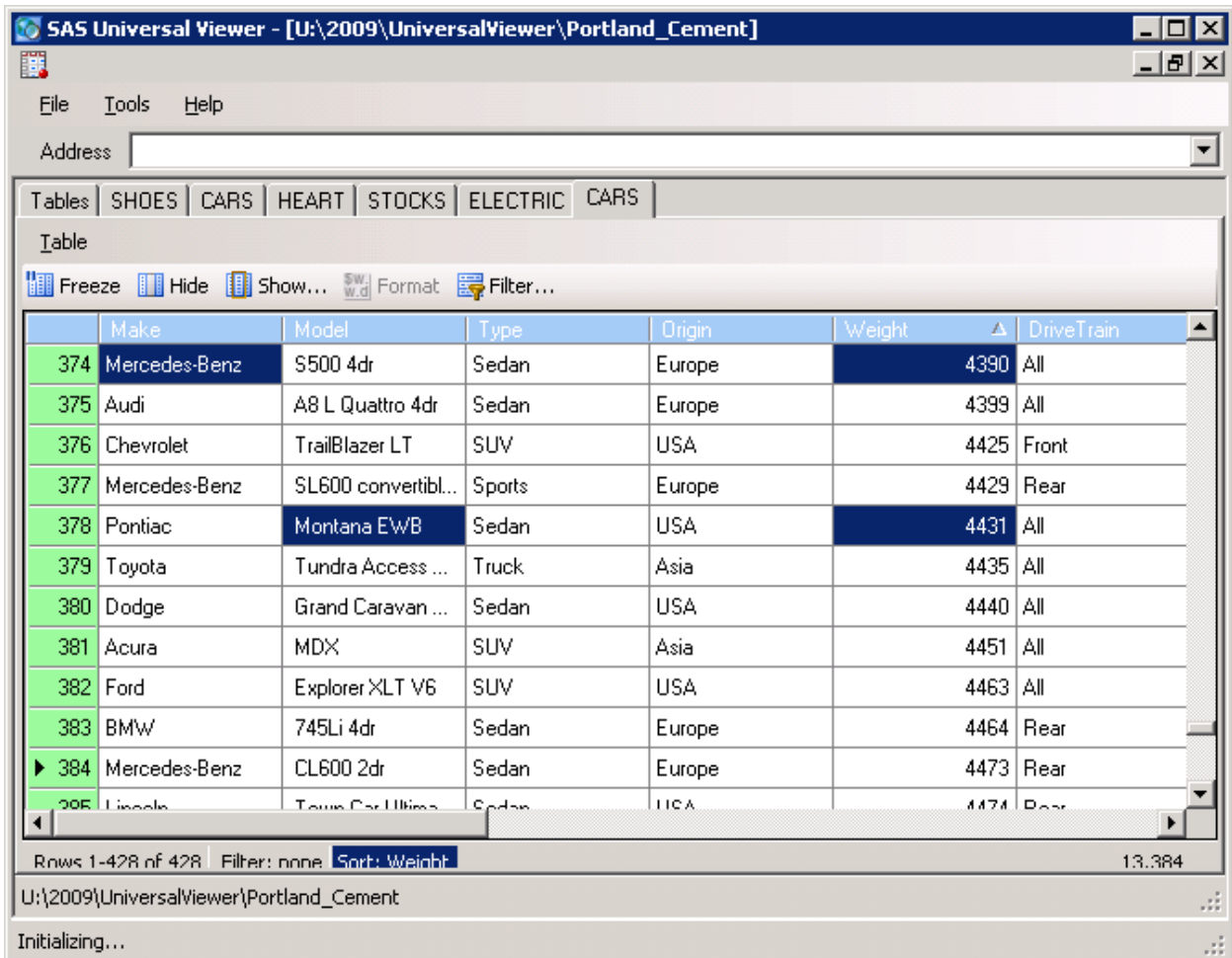
When the data are divided into pages, you have single arrows and arrows pointing to a line on the right side. Use these to step through the data a page at a time, or jump to the first or last page of data. Notice that a sort on this page of data is indicated with the arrow or triangle in the 'Weight' column. The triangle, unlike the highlighted 'sort' box at the bottom, also indicates that this is an ascending sort.

You can adjust the pagination by doing the following: go to Tools > Options, then the Performance tab. Remove the check in the box for "Break large datasets into pages". You can also adjust other aspects of the paging, as is obvious in that dialog box. This checkbox did not function in the test release of UV. You had to increase the number of rows.

If you change the performance settings without closing the existing data windows you can reopen any of the files with the new performance settings and view previously opened files both ways. This can be seen in the data window screen shot below (the second graphic); there are two open copies of the CARS dataset.



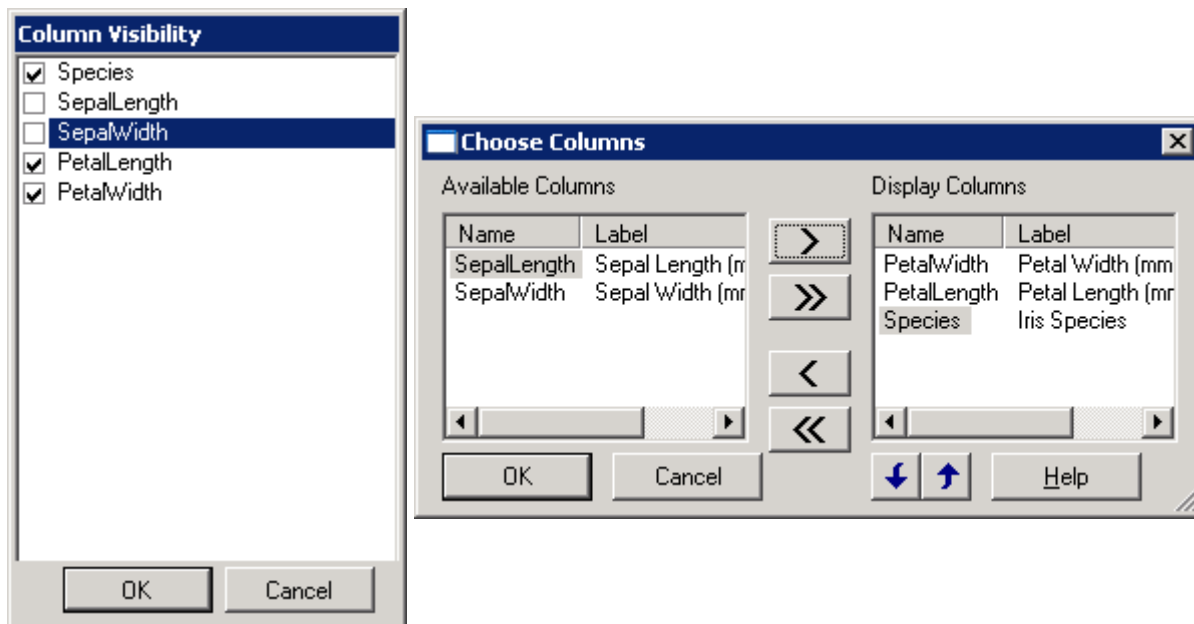
You see that all 428 observations are available below. The light blue 'paging' arrows on the right side are gone. Notice now, in the 11 rows displayed, that there is a much narrower range of weights (around the Pontiac Montana) than when you were just viewing rows 301-400 of the originally sorted data. You now have 3 Mercedes, whereas before you had none. This is because you have now accessed all the data, not just 100 lines of it. In the old SV, a sort or filter always accessed ALL data, and then displays the maximum number of rows specified by the settings.



UV ONLY considers the data in the page it has open. You can actually watch memory usage go up in "Task Manger" as you open new datasets. If the performance options are set such that there is no pagination then you have all the data. I am not sure which I like better. If I am just trying to get a feel for the data in a large dataset turning paging on makes it is a lot quicker to access the first page of data. But if I am looking for data patterns or especially outliers, I am going to want to sort or select on ALL the data.

I do want to clarify a point about SV. For sorting and filtering, SV accesses all record, but if an observation limit is set, it will read only those first x records when you first open a dataset. This is fast, and can lead to a big surprise if you sort or filter a large dataset, because it suddenly can be very slow in accessing the entire data.

Many other things are similar between these two applications. You can resize columns, hide or rearrange columns, and new to UV, you can freeze columns. This can be very helpful especially on wide data tables (see comments on wide files in last paragraph of Miscellanea). The column selector shown here is very easy to use, but you cannot rearrange the column order with it like with SV. You have to drag column headers in the data view. Compare the "Column Visibility" widget for UV on the left with the SV "Choose Columns" tool on the right. Note that SV displays the label here. SV has an option to display labels in the column header of the data view. UV does neither, but it does make the transition between data and metadata more graciously.



Sorting the data is just like sorting the metadata. Click on a single column, and you get an immediate sort. Click a second time, and the sort is reversed. Click on another column and you get a new 'clean' sort. The basis for each sort is the original data order of the source data file. So there is no advantage in sorting on columns sequentially, in hopes of getting a compound sort.

Remember that UV by default may bring in only a page of data, not the whole dataset. Therefore your sort reflects just the data in that page (if the option is on and the number of rows is larger than the limits).

FINDING WHERE YOU WANT TO BE

The Filter has matured from a very simple line command to a GUI with a chooser for both field names and Boolean expressions. It is nice to be able to simply choose your field. The tool also is data aware, so it enters the quotes around character fields. Note also the enclosing of the column names in a new UV specific 'function'. These are not like what a SAS programmer might expect, but you will soon become comfortable with them.

FilterEditor

Standard | Advanced

Height > 55 AND

Height <= 65 AND

Name like Ja*

Clear

```
Convert([Height], 'System.Decimal') > 55
AND
Convert([Height], 'System.Decimal') <= 65
AND
Convert([Name], 'System.String') like 'Ja*
```

OK Cancel

Note the results below from the above selection.

SAS Universal Viewer - [u:\2009\universalviewer\portland_cement]

File Tools Help

Address

Tables CLASSFIT

Table

Freeze Hide Show... Format Filter...

| | Name | Sex | Age | Height | Weight | predict | lower | upper |
|---|-------|-----|-----|--------|--------|-----------|---------|------------------|
| 1 | James | M | 12 | 57.3 | 83 | 80.387... | 55.4... | 105.299345471199 |
| 2 | Jane | F | 12 | 59.8 | 84.5 | 90.135... | 65.6... | 114.592205503751 |
| 3 | Janet | F | 15 | 62.5 | 112.5 | 100.66... | 76.3... | 124.96374544715 |

u:\2009\universalviewer\portland_cement

Initializing...

In the current, spring 2009 release, using the Standard tab in the Filter Editor, you cannot use the IN comparator, though it is a choice in the product. The tool starts by constructing something like

```
Convert([Name], 'System.String') in "
```

then inserts the text you type between the two single quotes. The IN comparator expects everything to be in parentheses, but not with single quotes around the entire expression. This is a recently reported defect.

Using the Advanced tab in the FilterEditor you can use a combination of GUI selections and actual text entered in the editor box to achieve a filter like this:

```
Convert([Name],'System.String') IN ('Jane','Judy')
```

similarly you can specify this

```
Convert([Name],'System.String') > 'Ja'
```

to get names falling alphabetically greater than 'Ja'. This last selection is not currently possible with the standard tab because '>' is not an option for character data. In our industry this could be used to select a date range based on an Isodate field – for instance `Convert([mydate],'System.String') > '2008-07'` for dates on or after August 1, 2008. Also note that the IN operator can be used with a numeric column in the advanced tab.

To make full use of the filter, you need to familiarize yourself with some new concepts. You must follow the syntax used in UV. You cannot simply write something like this `basedate < tday`. Submit that query and you will get an error message like this: "Cannot perform ... on System.Decimal and SAS.UV.Utility.SASNumeric"

To perform the above query requires this syntax.

```
Convert([basedate],'System.Decimal') < Convert([tday],'System.Decimal')
```

You have to actually supply the convert function on the right side of the inequality. Fortunately you can copy the left side directly from the FilterEditor window (Standard or Advanced tab) or you can use some of the more complicated functions found by scrolling down in the Operator list in the Advanced tab.

The screenshot shows two windows. On the left is the 'FilterEditor' window with the 'Advanced' tab selected. The filter expression is `Convert([basedate],'System.Decimal') < Convert([tday],'System.Decimal')`. On the right is the 'SAS Universal Viewer' window showing a table with 7 rows and 4 columns: 'basedate', 'tday', and 'un_tday'. The 'un_tday' column contains the unformatted date values corresponding to the 'tday' column.

| | basedate | tday | un_tday |
|---|----------|---------|---------|
| 1 | 18040 | 24MAY09 | 18041 |
| 2 | 18040 | 30MAY09 | 18047 |
| 3 | 18040 | 06JUN09 | 18054 |
| 4 | 18040 | 14JUN09 | 18062 |
| 5 | 18040 | 23JUN09 | 18071 |
| 6 | 18040 | 03JUL09 | 18081 |
| 7 | 18040 | 14JUL09 | 18092 |

Note that the variable 'un_tday' above is the unformatted equivalent of 'tday'. Similarly I have formatted and unformatted date time variables in the next example.

The advanced filter has four Types of numeric conversions that it permits. The datetime conversion is very helpful. You can then use the third part of the chooser to insert a date time token that UV understands. The date and time it inserts is just text with operators, so you can always edit that string.

The screenshot shows the SAS Universal Viewer interface. On the left is the FilterEditor dialog with the following settings:

- Standard / Advanced tabs: Advanced is selected.
- Column: dt
- Type: System.DateTime
- Operator: <
- Date: Saturday, May 23, 2009
- Filter string: Convert([dt], 'System.DateTime') < #2009/05/25 09:04:37#

On the right is the SAS Universal Viewer window showing a table with 10 rows. The table has columns 'dt' and 'unf_dt'.

| | dt | unf_dt |
|----|------------------|----------------|
| 1 | 20MAY09:01:21:53 | 1558401713.103 |
| 2 | 20MAY09:15:00:12 | 1558450812.103 |
| 3 | 21MAY09:04:38:31 | 1558499911.103 |
| 4 | 21MAY09:18:16:50 | 1558549010.103 |
| 5 | 22MAY09:07:55:09 | 1558598109.103 |
| 6 | 22MAY09:21:33:28 | 1558647208.103 |
| 7 | 23MAY09:11:11:47 | 1558696307.103 |
| 8 | 24MAY09:00:50:06 | 1558745406.103 |
| 9 | 24MAY09:14:28:25 | 1558794505.103 |
| 10 | 25MAY09:04:06:44 | 1558843604.103 |

The old SV application had a confusing way of dealing with missing numeric data. If you selected (excluding the quotes) "abc<." you get all records. The selection "abc>." displays no records. However, "abc=." displays what you probably wanted all along; it displays the converse, only the NON-missing values! In the current revision of UV it does not accept '.' as a valid reference. There may be a way to write this with the UV functions, but I have tried several things unsuccessfully. Writing something like $-0.0001 < x < 0.0001$, in UV syntax, is getting close. Hopefully something will be improved in the next release.

It is instructive to notice that in a column of integers a missing value sorts in between -1 and 0 in a normal ascending sort, but is between zero and (positive) 1 if sorted descending. When you think about the real number line mathematically, you will realize that missing values really do not fit anywhere, so a missing value is always a special case. In normal SAS data processing missing is 'smaller' than an infinitely large negative number. Then there are the special SAS numeric missing values of .A, .B,Z!

The observation limit in the old SV application only impacted presentation. Any sorting or filtering used all the records in the file. This was very useful. If you wanted to find max or min values or rare events you could set the obs limit to 100, and usually get all the information you desired via a sort or selection. With UV, all processing is restricted to the current set of rows based on settings in the Performance tab. If the records you are interested in are outside the bounds of the current 'page', you must look elsewhere. You must step through the pages of data to find the desired values. Alternately, you can just take the row limit off, but then you may need to deal with performance issues if your dataset is large. For users familiar with SV this obs range limit can be a significant issue if they are not aware of this restriction. "WHERE IS MY DATA!" is a common response. SV was able to scan all records even when the display limit was set. UV works differently.

MISCELLANEA

SV has had issues with newer functionality, even those from SAS8 (e.g. TimeAMP format). UV will present these newer SAS formats whereas its less 'in shape' older brother just ignored the façade of these formats and will presents just 'the facts' - just the bare numbers. With some of the more advanced features of SAS9 data, like long format names, SV just fails to open the file at all. Thus in these cases, UV is the only game in town. Please note that moving a dataset from one platform to another can render a very normal dataset created in SAS9 to be not viewable with SV.

If the field width is smaller than needed to display the data, UV presents three trailing dots – 'continued' (see previous examples). This is true for any width of column, and for metadata as well as data. Lack of such presentation was a real nuisance with SV when you were looking at a field that got cut off, and you did not realize immediately that the column was really wider than shown. UV will also deliver the entire contents of any field into the paste buffer. I quickly created a column that was 30,000 characters wide, and ran a loop to fill it. When I copied and pasted that **cell** into a word processor it had almost twice as many words as this paper. This copy and paste method might be a useful way of reading some of those really long AETEXT fields.

UV has an embedded version of the SAS Enhanced Editor, including syntax highlighting. This is an **embedded** application, therefore its menus are included **inside** the application. They are not in the top level. If you count the horizontal bands of information and menus, the File menu for the editor is in the 6th band from the top. UV also can open other types of files, some more gracefully than others. It does not like having a *.lst file dragged and dropped on it, but it will open it via the application File menu. Let me mention one final area of beautification – it needs a means of navigation between windows that are maximized.

Here is a trick for those using double monitors, and one that might make you want to use them if you don't already do so. Working with double monitors is very easy to do with a laptop and a monitor, especially if your laptop docks into a device that allows it to sit open like a book in a reading stand, but in the upright position. You will need to open your desktop properties and "Extend my desktop". OK, hardware aside, if you have two monitors, try resizing your UV window to fill the width of both monitors combined. This can affordably give you nearly 30 inches of horizontal desktop. Few people have 22 inch monitor (that is about 18 inches horizontally), much less 30s. With dual monitors you can really see a whole lot of data. While mentioning the underlying stuff supporting UV, I would also comment that I was sad to see that I could not install UV on my SP3 version of XP Home edition.

CONCLUSION

It has been clear for a long time that we needed a big update to SAS Viewer. The Institute has done a good job launching an effort to rethink and rebuild a tool for this niche. Not everyone is going to be happy with the new, free Universal Viewer, but it is the supported tool of the future. I have outlined many of the features and methods of use for this application and hopefully guided you past some things that can a point of discouragement or frustration.

The fact that it does not lock data files will catapult it forward in many community work environments where locked files have been an issue. It has made switching between files more intuitive. Importantly it supports new data features found in newer releases of SAS. The filtering functionality is significantly changed and mostly improved. When experienced users of SAS Viewer are doing sorting or subsetting they will need to recall that Universal Viewer only looks at the current 'page' of data. As for most new things you will gain a real appreciation for this new tool with exposure and use.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Steve Wright

Quintiles, Inc.

RTP, NC 27709

Email: steve.wright@quintiles.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. © indicates USA registration.