Paper DM02

Bringing Clarity to Complexity

Neville Cope, Amgen Ltd, Cambridge, UK
Matt Whiley, Amgen Ltd, Cambridge, UK

ABSTRACT
Clinical trials frequently feature complex visit schedules and many (often related) endpoints. Ensuring that all the key data points are collected at the right time and are recorded accurately can be a difficult, and daunting task. This paper demonstrates two visual aids, coded in SAS/GRAPH®, which can help in the review of data and speed up database cleaning activities.

INTRODUCTION
This paper shows examples of two data visualisation methods used to help analyse and clean data from an Amgen clinical trial. The study was a Phase 3 trial looking at the safety and efficacy of weekly doses of the drug P118 for the reduction of oral mucositis in head and neck cancer patients receiving adjuvant radiotherapy and chemotherapy.

The study featured a complex visit schedule and a set of several oral mucositis (OM) grading scales, and checking them all for consistency and accuracy was becoming increasingly difficult the more data we received.
Instead of the usual review of data listings and tables of the raw data, several tools were developed to aid in this checking using graphical representations of the data.

TOOL #1: STUDY SCHEDULE

MOTIVATION
The scheduling for the P118 study was very complicated – there were different schedules for:

- Chemotherapy (CT)
- Radiotherapy (RT)
- Investigational Product Dosing (IP)
- Oral Mucositis (OM) grading
- Patient Reported Outcome (PRO) assessment s

These different schedules were all running in parallel. The timing of some events, such as IP administration, were fixed by the protocol. Other events, such as RT and CT administration followed site practice, and were highly variable. In a further complication, events in one schedule could trigger changes to other schedules. All of this meant that just a few small deviations could result in patient schedules that looked markedly different from the “ideal schedule” envisaged at the time of protocol development.

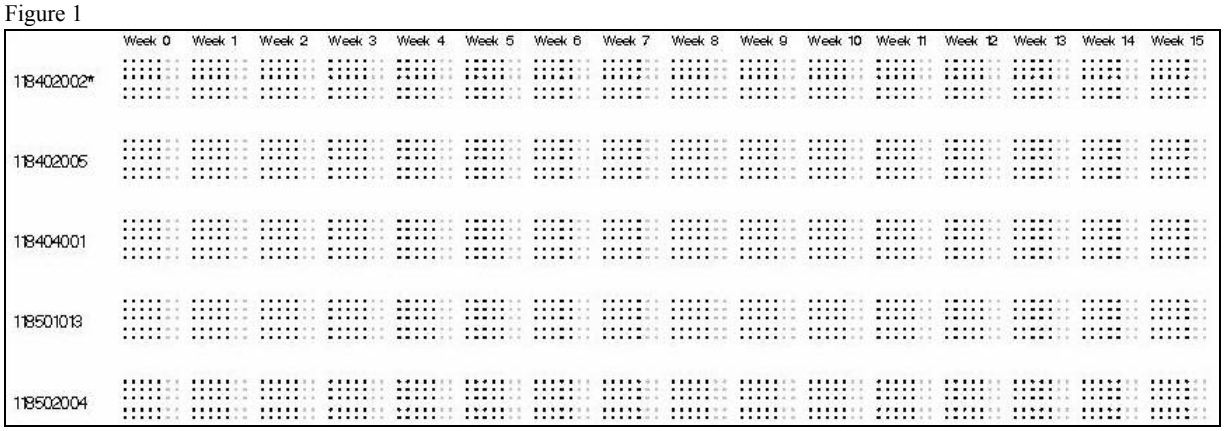
In general this wasn’t an issue, however it was noticed that occasionally subjects completed their course of IP while still undergoing RT and CT. In the trial subjects were scheduled to receive only 7 or 8 doses of IP, but standard practice is to keep treating patients with the RT/CT combination until they have received their full dose of RT.
The study team felt this was a problem, as receiving RT/CT without the protective effect of the P118 drug could prolong OM and increase its severity. However, it was felt that this was probably a very rare occurrence and was therefore unlikely to confound the analysis of this study

IMPLEMENTATION
We wanted a way of looking at a patient’s whole study schedule, to help us to identify cases where IP administration ceased while RT/CT was still ongoing. The usual way of cross-checking several listings was becoming too complex, and looking at all the possibilities programmatically was also difficult.

To start with, we thought it would be logical to show each patient as a horizontal strip, like a timeline with each week and day clearly marked along with when each event occurred. A basic scatter plot wouldn’t be clear enough, so each patient’s week was created as a block, with dots representing the framework. With the study lasting up to 15 weeks, a block needed to be shown for each week.

Within each patient we set up 5 lines, each one representing a particular schedule.
The basic skeleton is shown in Figure 1.

Figure 1



The top line for each patient represents IP, the second CT and the third RT. To help readability these first three schedules were displayed together, with a slight gap below. Following the dosing schedules are lines to represent the PRO and OM schedules. The days of the week are shown as small black dots, with the weekends coloured grey as shown in Figure 2

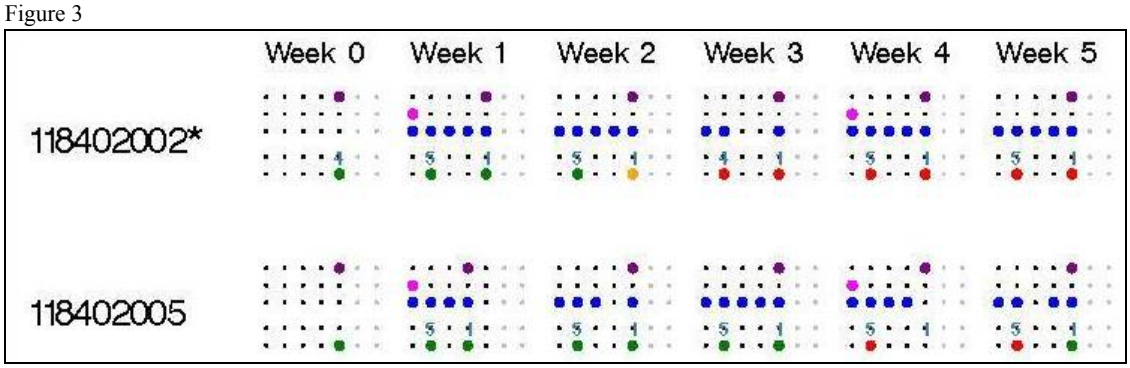
Figure 2

The matrix of dots is created using PROC GPLOT and symbol statements, and the labelling of the week and patient numbers was added using annotate.
Some small macros were written to convert the patient/week/schedule information into an x and y coordinate to make all further annotations quick and easy to implement.

A SPASH OF COLOUR

With the basic study framework in place, each event can be plotted on the framework.
To make it easy to pick out the different events, each row was assigned a colour and the points plotted with larger symbols.

Figure 3



As well as plotting the events, more detailed information can be overlaid onto the framework. In Figure 3, the PRO line displays a number instead of a dot. This number represents the number of questionnaires completed on that day (there were 5 separate questionnaires)

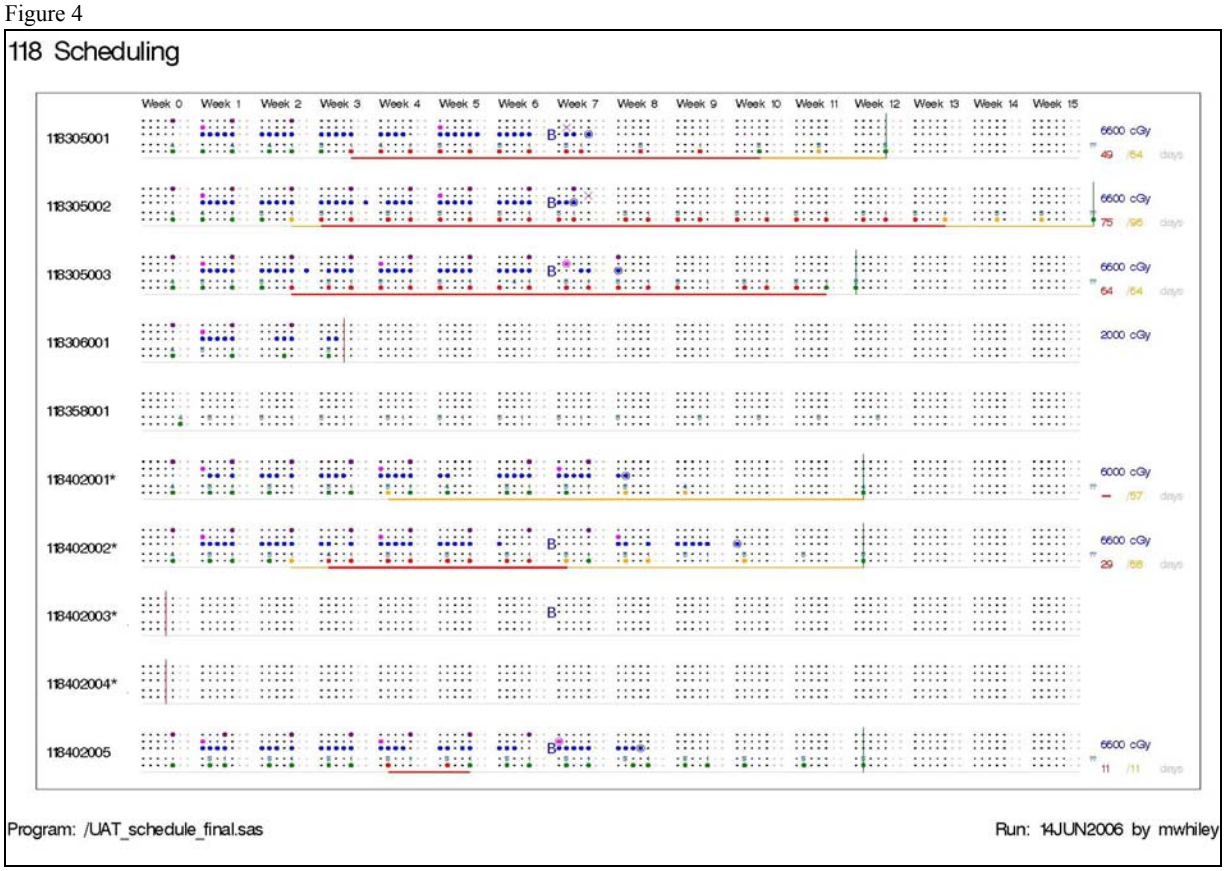
BUILDING ON THE FRAMEWORK

Other information was then added to this framework, which culminated in a compete overview of each patient's key information from the study.

Events such as when a patient discontinued the study or received boost radiotherapy, their total RT dose over the course of the study, and more detailed visual representations of their OM scores were eventually added.

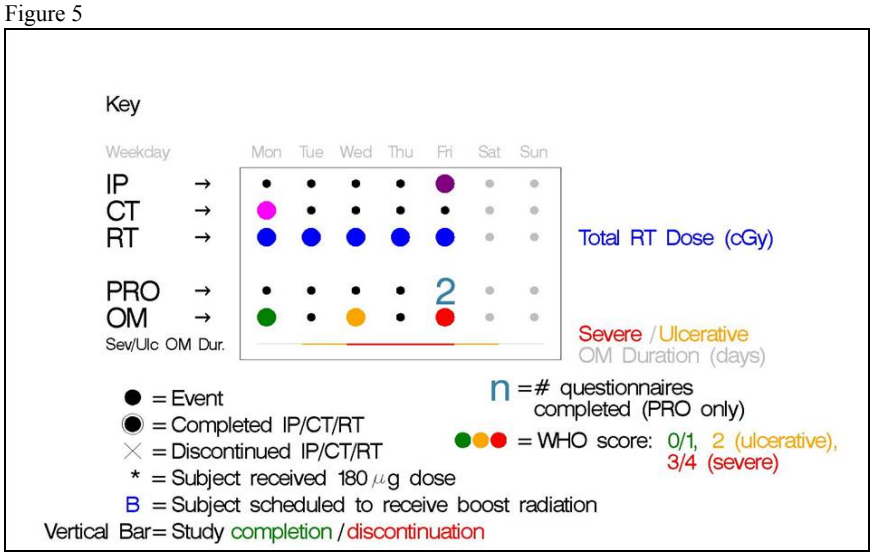
The PROC GPLOT code was then wrapped in ODS RTF statements to create a document that can be reviewed in Microsoft® Word, and Figure 4 shows a page from the finished document.

Figure 4



A key was also created using an annotate dataset, as shown in Figure 5

Figure 5



OUTCOME

The tool proved to be invaluable in helping to spot dosing errors, safety trends and potential problems

One example of this is that while reviewing some early outputs, the team realised that subjects receiving RT/CT without the potential protection of P118 was a bigger problem than anticipated, and they could quickly see the doses and lengths of time involved.

As a result of this finding, a protocol amendment was issued that made IP available to subjects for as long as needed to cover their entire course of RT/CT.

This tool underwent subsequent development as new questions arose, and amongst other things it has been used to:

1) Aid in defining clinically relevant breaks in RT and CT (important secondary endpoints for this study).
2) Explore site compliance with the schedules as specified in the protocol.

TOOL #2: ORAL MUCOSITIS SCALES

MOTIVATION

The P118 study uses 3 main scales for assessing OM, as shown in Figure 6.

Figure 6

	Abbreviated oral mucositis assessment scales		
	WHO	**CTCAE clinical**	**CTCAE functional**
Grade 0	None	None	None
Grade 1	Soreness	Erythema	Ability to eat solids
Grade 2	Ulceration	Patchy ulcerations or pseudomembranes	Requires liquid diet
Grade 3	Ulceration Requires liquid diet	Confluent ulcerations or pseudomembranes	Alimentation not possible
Grade 4	Ulceration Alimentation not possible	Tissue necrosis	Life threatening consequences

Increasing severity

Even though the patient is graded against each of these scales using different criteria, all three scales are essentially capturing different aspects of the same disease, so a certain degree of consistency between them is to be expected. Furthermore, certain combinations of these three scales are impossible as they are contradictory.
For example, if WHO=2 but CTCAE-Clin=0 and CTCAE-Func=0 then that would show a possible error that would need further investigation.
We wanted to put in place some sort of check that would allow us to look at trends in the recording of the OM scores and ensure consistency of the three scales.

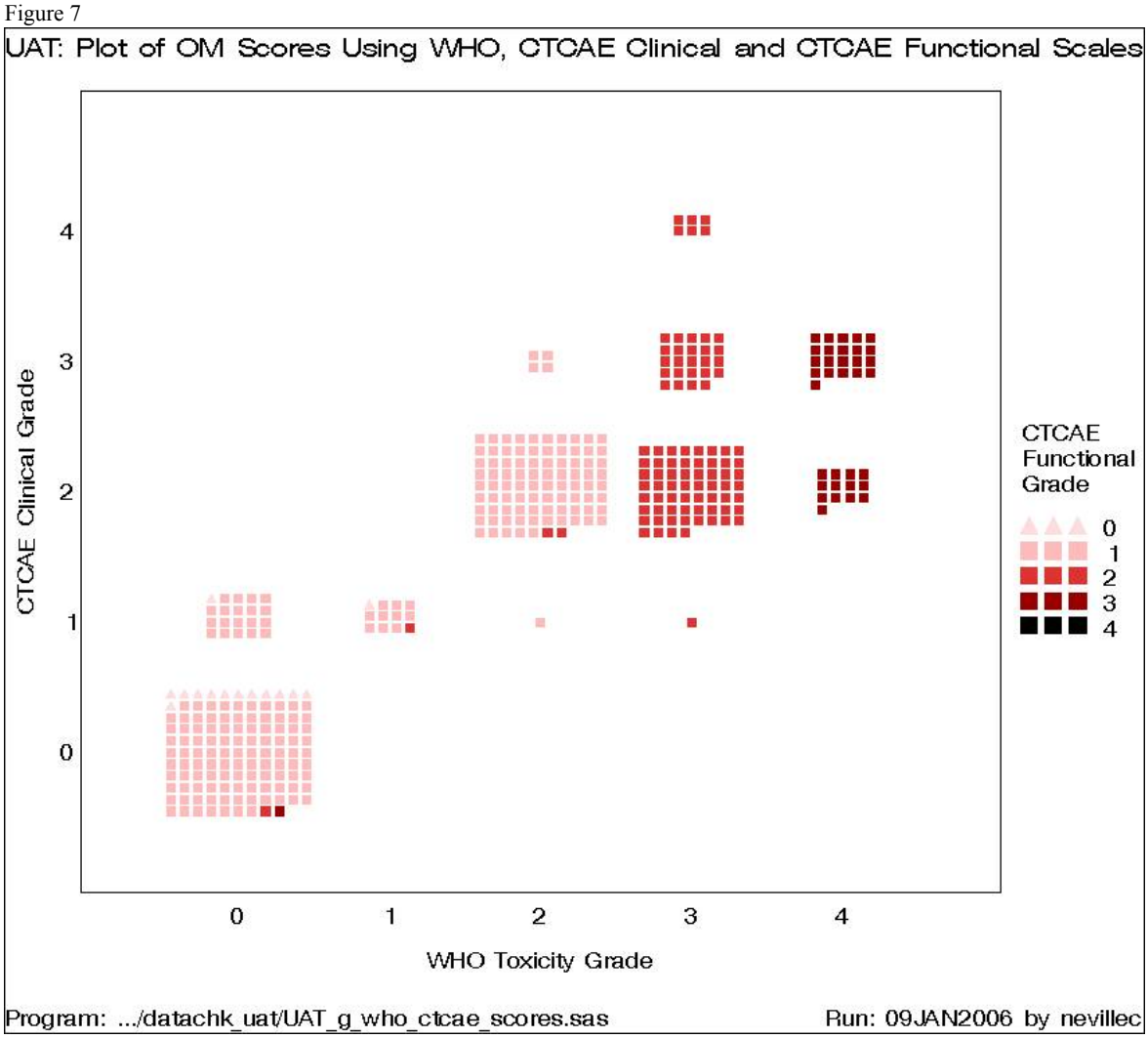
IMPLEMENTATION

Our approach was to create a plot that shows all three scales at once. We also added in some of the features of ODS HTML in SAS/GRAPH, such as drill-down functionality.

The scales were all discrete and had scores of 0 to 4. As a 3D plot would be unwieldy, we decided to assign the WHO score to the x-axis, the CTCAE Clinical scale to the y-axis and represent the CTCAE Functional scale using different symbols and colours, with a darker shade indicating a higher score.
Every assessment could then be plotted onto the graph.

An algorithm was written to examine the data before plotting to see how many points would occupy each intersection of the discrete x and y axis values, and then make small changes to the x and y values of each point to plot them in a block so point's didn't overlap. This is demonstrated in Figure 7.

Figure 7



As you can see from this example, it's immediately obvious that there are two suspicious values in the bottom-left group. If WHO=0 and CTCAE-Clin=0 then the CTCAE-Functional scores of 2 and 3 could represent a data error.

POINT AND CLICK

Seeing these potential problems on a graph was useful, but we needed a quick way of investigating these data points further. In order to do this we had to switch the ODS output from RTF to HTML, but as this was going to be an interactive exploratory tool and not a printed production report this wasn't a problem.

The first thing we added was the equivalent of a "Tool Tip" in Microsoft Office. If the mouse pointer is hovered over a data point, a small box appears displaying the key information of patient and visit number which would allow us to go and look up the data in our database. This is accomplished by adding an extra variable to the dataset that contains the text to be displayed over each data point using the HTML attribute "*alt=*".

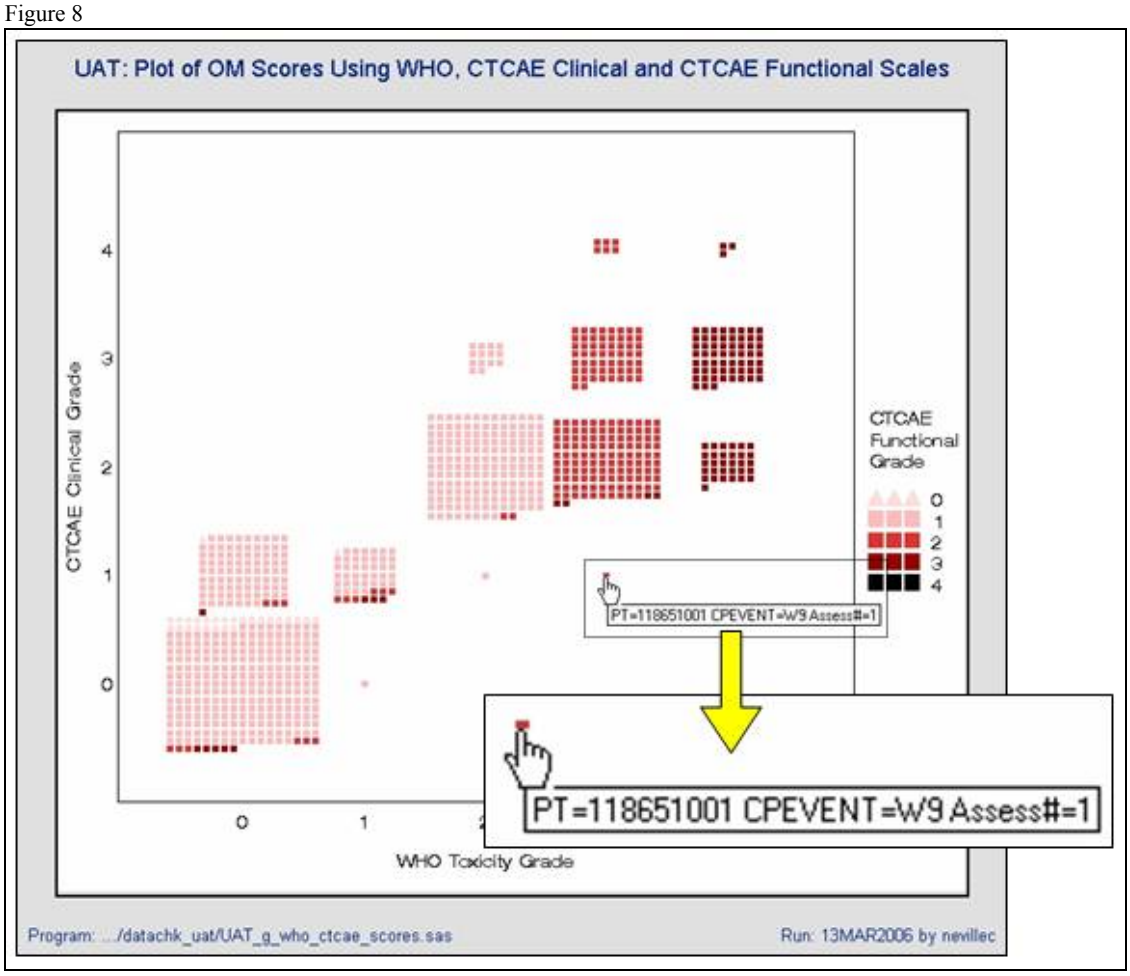
```
data output;
  set raw;
    drill="alt='PT=" ||compress(pt)|| " CPEVENT=" ||compress(cpevent)|| " Assess#="
||compress(qualifyv)||"'";
run;
```

This variable is then referenced in the HTML option of PROC GPLOT.

```
proc gplot data=test5 ;
format func func.;
    plot symp*who=func / legend=legend1 haxis=axis1 vaxis=axis2 html=drill;
  run;
quit;
```

Which, after writing some datastep and ODS code to create a HTML page containing the plot, creates output as shown in Figure 8.

Figure 8



DRILLING DOWN

While this was useful, if a patient has some strange results it is helpful to view all of the data for that patient to see if there is a trend or if it is more likely to be an isolated incident.

We created a proc report of the data for each patient, and outputted this to a separate HTML file. This could then be linked to using the HTML option again, but with the addition of the "*href=*" attribute.

```
data output;
  set raw;
    drill="alt='PT=" ||compress(pt)|| " CPEVENT=" ||compress(cpevent)|| " Assess#="
||compress(qualifyv)|| "' href='patient_&sysdate9..html#PT"||compress(pt)||
"' target='patient'";
run;
```

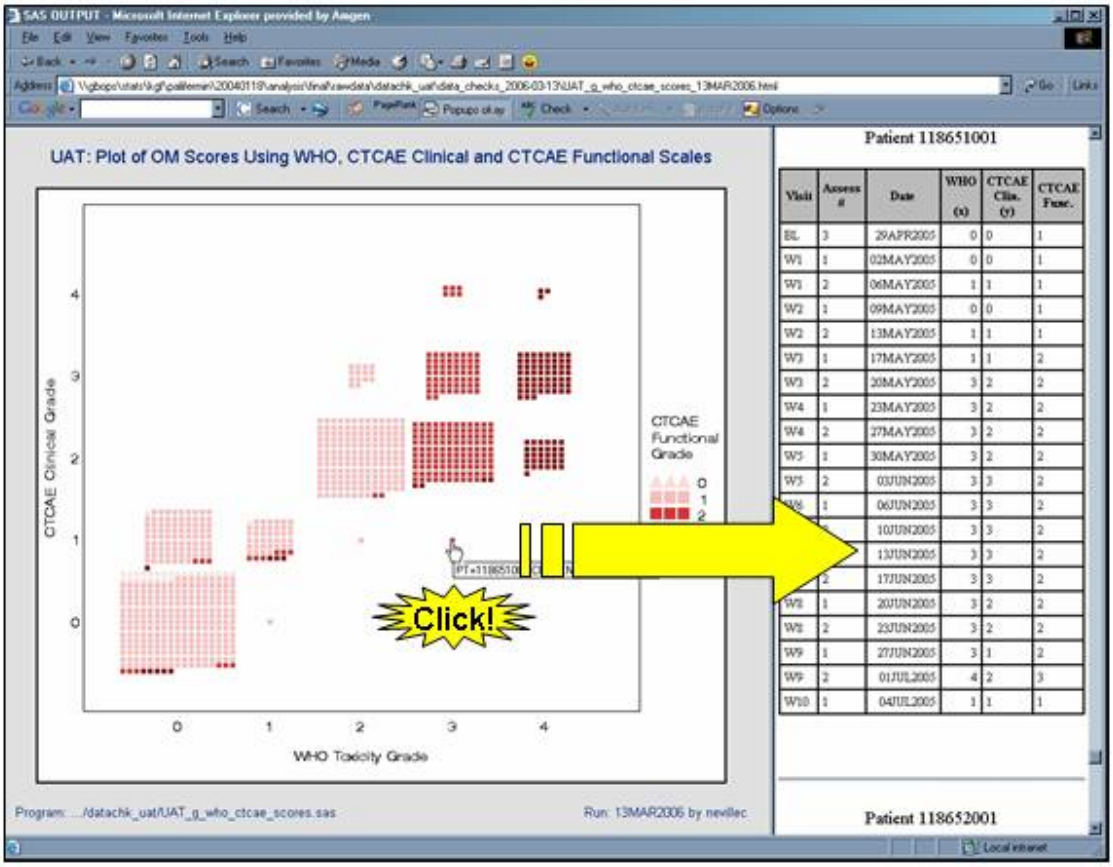
The "*href=*" attribute points to the output file of the proc report, with a patient reference following the # symbol. When the proc report was created, the BYVAL option was used and a link was embedded into the title statement like this:

```
proc report nowd data=test5(where=(pt ne '')) missing style={outputwidth=100%};
title 'Patient #byval1 ~R"<a name=pt#byval1>" ';
footnote;
  by pt;
...
run;
```

The main HTML page contains two panels – the main plot on the left and the proc report output on the right. The right-hand panel was assigned the label "patient" which the "*target=*" HTML attribute refers to.

When this was all put together and run, clicking on a data point immediately displayed all of the data for that patient as shown in Figure 9.

Figure 9



OUTCOME

As more data was entered into the database it became easier to see inconsistencies using the plot. Even though the bulk of the data was looking good, there did seem to be a few more errors than anticipated.

After further investigation, a potential problem was found in the method of collection used.
For the P118 study investigators complete a worksheet as part of the OM assessment process, this worksheet is the source data for assigning an OM score, and this score is QCd by an independent organisation who have access to the source worksheets.
However, after these checks the QCd scores are then transcribed by hand onto the CRF and submitted to Amgen.

It was discovered that most of the errors were occurring at this transcription stage, and so additional checks have been put in place to pick up on these errors so they can be queried with the sites as early as possible.

CONCLUSION

As I'm sure you have noticed, very little SAS code has been presented in this paper. The code for generating these two graphical outputs is quite large (approximately 800 lines for #1 and 250 lines for #2) and quite a lot of it is manipulating data from our databases.

However, the main intention of this paper was not to show code, but to demonstrate that presenting data visually needn't be confined to a couple of PK or survival analysis plots in the back of a final study report. Interrogating data to look for potential problems is normally done using data listings or the odd few summary tables, but for some types of data a visual representation can speed up the review process and highlight potential problems much quicker.

SAS/GRAPH is a very powerful tool, and with the annotate options almost anything is possible. While programming graphics can be time consuming, and often frustrating with all of the options and drivers, with careful planning and a modular programming approach the results can be worth it.

After all, "a picture paints a thousand words "

CONTACT INFORMATION (HEADER 1)

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

Neville Cope, Biostatistics Department
Amgen Limited
240 Cambridge Science Park
Milton Road
Cambridge
CB7 4JH
Work Phone: 01223 436318
Fax: 01223 423049
Email: nevillec@amgen.com