PhUSE 2017

Paper DV06

Interactive Visualization of Clinical Laboratory Parameters with SAS Stored Processes and jQuery AJAX

Volker Harm, Bayer AG, Berlin, Germany

ABSTRACT

Starting with an appropriate clinical laboratory parameter data set graphs will be created presenting individual and summary data. To select subjects, treatments, or groups dynamic HTML select tags will be derived from the input data. HTML option tags will be used to determine the appearance of the graph such as axis styles and reference lines. These interface elements will be put together to an interactive dashboard for investigating the laboratory parameters of a clinical study by retrieving and presenting the data dynamically with SAS Stored Processes and jQuery AJAX.

INTRODUCTION

In our department Research and Clinical Sciences Statistics for a certain type of phase I standards could be established that allow to automatically provide laboratory data from new studies in a predefined data structure that allow the creation of standard visualization tools. As a prototype we developed a tool that presents laboratory data in individual and summary form. We will use SAS ODS Graphics to create single graphs, SAS Stored Processes to present them on our Intranet and then we will try to implement a versatile navigation from graph to graph and from to tabular data. Once we have these building blocks the responsiveness of the application will be enhanced by using AJAX techniques.

GRAPHICAL PRESENTATION OF CLINICAL LABORATORY PARAMETERS

INDIVIDUAL DATA

The following graph is created for the individual data of any laboratory parameter using the following code:

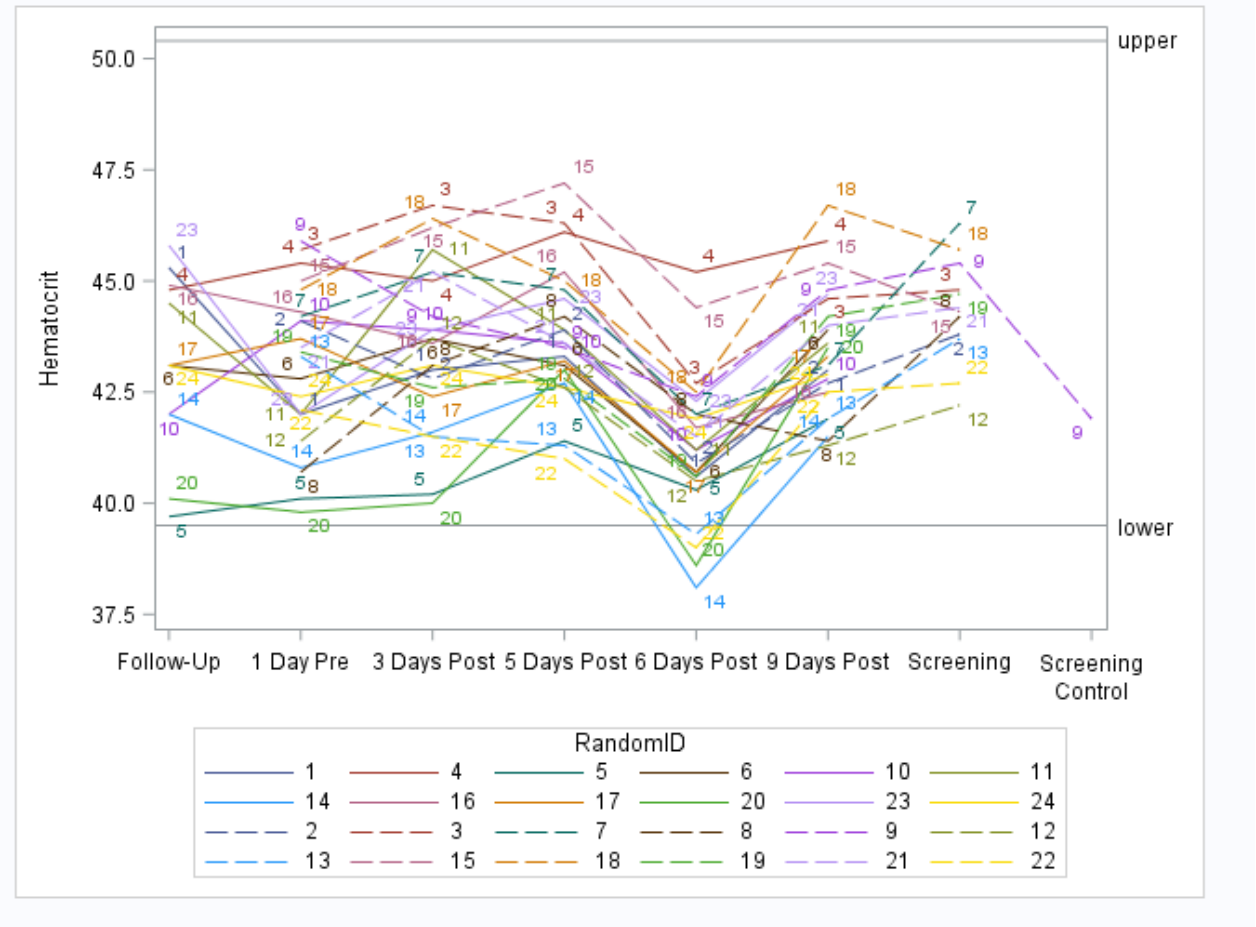
```
%macro plotLabValues (cLBParameterShort =, mOrder =);

    data GraphData;
        set org.LabValues (where = (cLBParameterShort = "&cLBParameterShort." and
mOrder = &mOrder.));

        if _n_ = 1 then do;
            call symputx ('cLBParameter', cLBParameter);
            call symputx ('cTreatment', cTreatment);
            call symputx ('nLBLLN', nLBLLN);
            call symputx ('nLBULN', nLBULN);
        end;
    run;

    proc sgplot data = GraphData;
        title "&cTreatment.";
        xaxis discreteorder = data display = (nolabel);
        yaxis label = "&cLBParameter.";
        refline &nLBLLN. /label = ('lower') labelloc = outside noclip;
        refline &nLBULN. /label = ('upper') labelloc = outside noclip;
        series x = cLBRT y = nLBValue /datalabel = RandomID group = RandomID;
    run;

%mend;
```



SUMMARY DATA

The following graph is created for the summary data of any laboratory parameter using the following code:

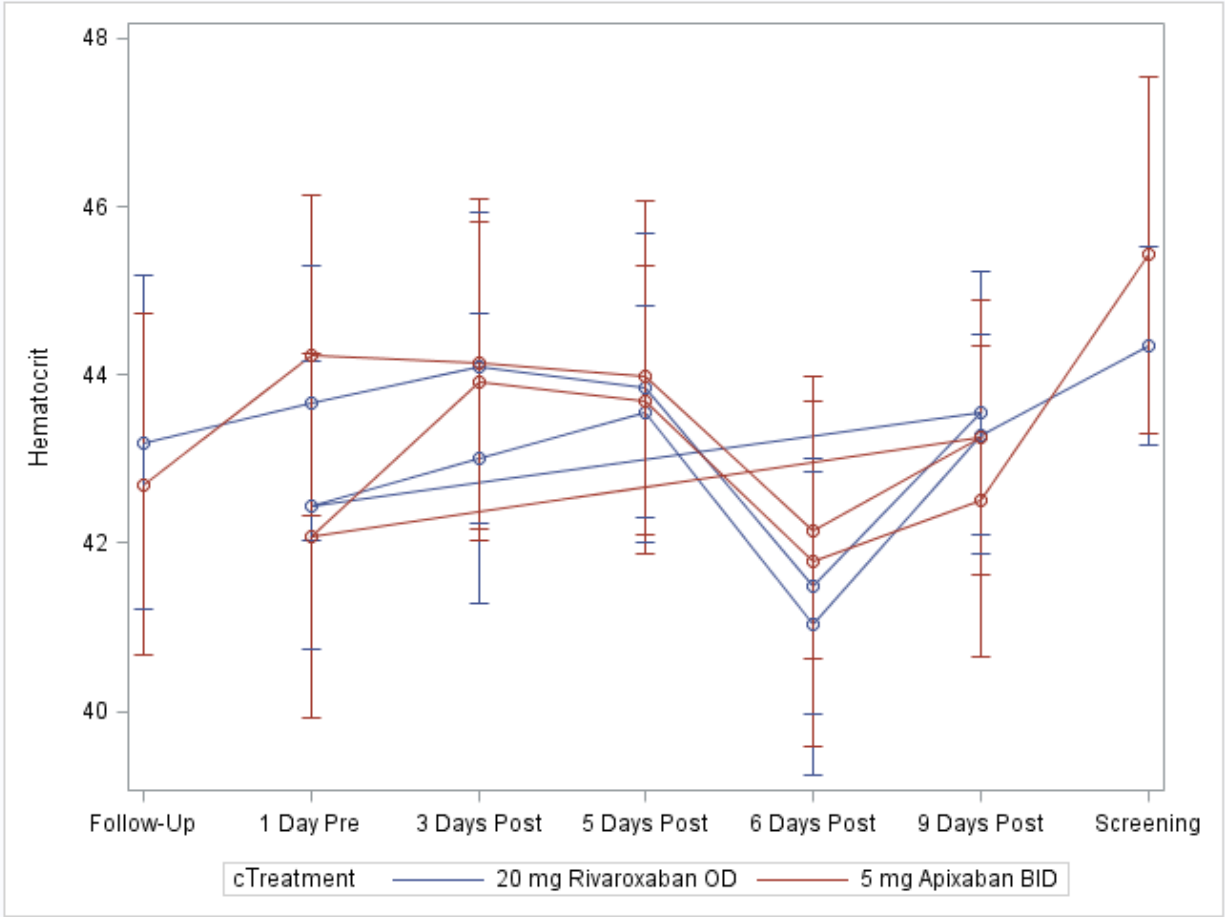
```
%macro plotLabSampleStats (cLBParameter =);

    data GraphData;
        set der.LabSampleStats (where = (cLBParameter = "&cLBParameter."));
        lower = nLBValue_Mean - nLBValue_STD;
        upper = nLBValue_Mean + nLBValue_STD;

        if _n_ = 1 then do;
            call symputx ('nLBLLN', nLBLLN);
            call symputx ('nLBULN', nLBULN);
        end;
    run;

    proc sgplot data = GraphData;
        xaxis discreteorder = data display = (nolabel);
        yaxis label = "&cLBParameter.";
        series x = cLBRT y = nLBValue_Mean /group = cTreatment;
        scatter x = cLBRT y = nLBValue_Mean /group = cTreatment yerrorlower = lower
yerrorupper = upper;
    run;

%mend;
```



USER INTERFACE ELEMENTS

SELECT TAGS

Select Parameter:

HCT

CRP
CA
CL
CHOL
CK
CKMB
CREAT
CREATCLR
EOS
EOSLE
MCHC
MCH
MCV
RBC
GGT
GLUC
HDL
HCT
HGB
LDL

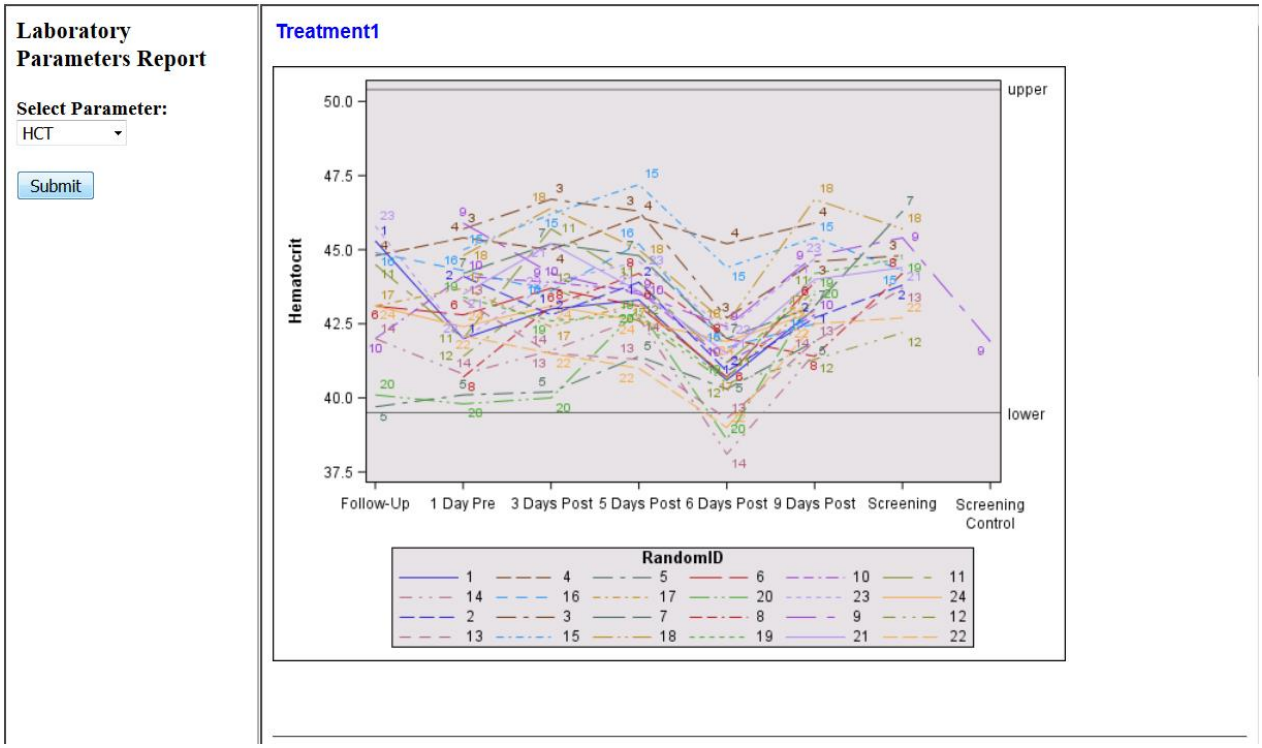
```
data _null_;
    set Parameters end=alldone;
    file _webout;

    if _n_ = 1 then do;
        put '<b>Select Parameter: </b>';
        put '<select name="parameter">';
    end;

    if cLBParameter ne " " then do;
        put '<OPTION VALUE="' parameter '">' cLBParameterShort;
    end;

    if alldone then do;
        put '</select>';
        put '<br><br>';
    end;
run;
```

DASHBOARD



SAS STORED PROCESS ARCHITECTURE

```
*ProcessBody;
%macro main;
    %global reqtype _odsstyle;

%macro first_request;
    /* This is the first request. Create an HTML page with Frames. */
    data _null_;
        file _webout;
        thissrv = symget('_url');
        thispgm = urlencode(symget('_program'));
        put '<html>';
        put '<table>';
```

```
        put '<tr>';
        put '<td width="20%">';
        put '<iframe name="frame1" scrolling="no" width="100%" height="600" src="'
thissrv +(-1)  '?_program=' thispgm +(-1) '&reqtype=create_selection&_debug=0">';
        put '</iframe>';
        put '</td>';
        put '<td width="80%">';
        put '<iframe name="frame2" width="100%" height="600" >';
        put '</iframe>';
        put '</td></tr>';
        put '</table>';
        put '</html>';
    run;

%mend first_request;

%macro create_selection;
    /* This is the Second Request. Create the Selection Memu. */
    /* Get age groups */
    proc summary data=sashelp.class;
        class age;
        output out=summary;
    run;

    data _null_;
        set summary end=alldone;
        file _webout;

        if _n_ = 1 then do;
            thissrv = symget('_url');
            thispgm = symget('_program');
            put '<html>';
            put '<h3>Laboratory Parameters Report</h3>';
            put '<FORM ACTION="'  thissrv +(-1) '" method=get  target="frame2">';
            put '<input type="hidden" name="_program" value="'
                thispgm +(-1) '">';
            put '<input type="hidden" name=reqtype value="report">';
    run;

    proc sort data = org.Labvalues out = Parameters nodupkey;
        by cLBParameter;
    run;

    data _null_;
        set Parameters end=alldone;
        file _webout;

        if _n_ = 1 then do;
            put '<b>Select Parameter: </b>';
            put '<select name="parameter">';
        end;

        if cLBParameter ne " " then do;
            put '<OPTION VALUE="' parameter '">' cLBParameterShort;
        end;

        if alldone then do;
            put '</select>';
            put '<br><br>';
        end;
    run;

    data _null_;
```

```
        file _webout;
        put '<input type="submit" value="Submit">';
        put '</form>';
        put '</html>';
    run;

%mend create_selection;

%macro create_report;   /* Produce the report */
    %let _odsstyle=BarrettsBlue;

    %stpbegin;
    Title "Students who are age &age";

    %plotLabValues (cLBParameterShort = HCT, mOrder = 1);
    %stpend;
%mend create_report;

%if "&reqtype" = "create_selection" %then %do;

    /* Produce the Selection Menu   */
    %create_selection;
%end;
%else %if "&reqtype" = "report" %then %do;

    /* Produce the Report   */
    %create_report;
%end;
%else %do;

    /* Produce the HTML page that contains Frames */
    %first_request;
%end;
%mend main;

%main;
```

POSSIBLE ENHANCEMENTS WITH JQUERY AJAX

```
<div id="UI" style="float:left;">
<form url="&_url" id="myForm">
&streamDelim;%include srvrpgs(&formUI);
<script type="text/javascript">
    var frm = $('#myForm');
    frm.submit(function () {
    $("#region1").html('<div style="margin-left:50px">'
                    + '<h1>Processing . . . Please Wait . . . </h1>'
                    + '<img src="images/progress.gif""></div>');
    var theURL = "&_url?"+frm.serialize()+'&nocache='+Math.random();
        $.ajax({
            url: theURL,
            success: function (data) {
               $("#region1").html(data);
               toggleUI();
            }
        });
        return false;
    });
</script></form>
</div>
```

CONCLUSION

In this paper we showed the techniques needed to build interactive visualizations using SAS Stored Processes and enhancing the interactivity using jQuery AJAX..

REFERENCES

Henderson, Don. 2013. *SAS Server Pages: Generating Dynamic Content*. Cary, NC: SAS Institute Inc.

CONTACT INFORMATION

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

Volker Harm
Bayer AG
Muellerstrasse 178
Berlin 13353
Work Phone: 004946811208
Email: volker.harm@bayer.com