TELLING THE PATIENT STORY WITH GRAPHICS IN A RARE DISEASE TRIAL

D. Streng / M. Litzsinger

Disclaimer Slide

- *The presenters are employees of BioCryst Pharmaceuticals Inc. and any opinions expressed herein are their own.*

Overview

- Challenges in Rare Disease
- Best Practices
- “Traditional reporting” versus “Combined graphics”
- Output examples

Mock data in this presentation does not depict actual studies or patients.

Rare Disease Differs from Common Indications…

RARE

The Orphan Drug Act[1] defines a rare disease as affecting <200k people in the US. Typically, a few patients across many sites.

UNEXPECTED

Patients are seriously ill, without treatment alternatives, and may be <18 years old.

REPORT EACH PATIENT in near-real time

- Make intuitive, visual presentations.
- Supplement with objective summary displays and sortable listings.

Rare Disease Best Practices

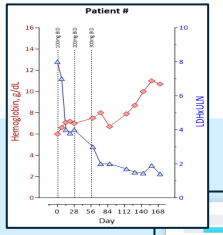
#1 Innovation

In real-time as each patient doses

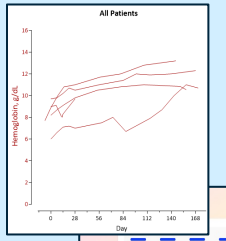
- Build ongoing, flexible reporting for Safety and Endpoints.
- Re-use for periodic reporting, interims, and finally for the Clinical Study Report.

#3 Opportunities

A picture is worth a thousand words



- Graph and List individual patients.



- Graph and Tabulate trends.

#2 Go Fast

Standardize programming using the 80-20 rule: 80% the same across studies, 20% standalone exceptions

- Make simple, lean standards for 80% of datasets and outputs; Limit exceptions.

Investigate Safety (S) and Endpoints (E)

Be curious about *unexpected* and *expected* events

- **E:** Change from baseline in <Measurement 1>
- **E:** Change from baseline in <Measurement 2>
- **S:** Incidence of <Intervention 1>
- **S:** Incidence of graded treatment-emergent AEs; Incidence of graded laboratory abnormalities

UNEXPECTED

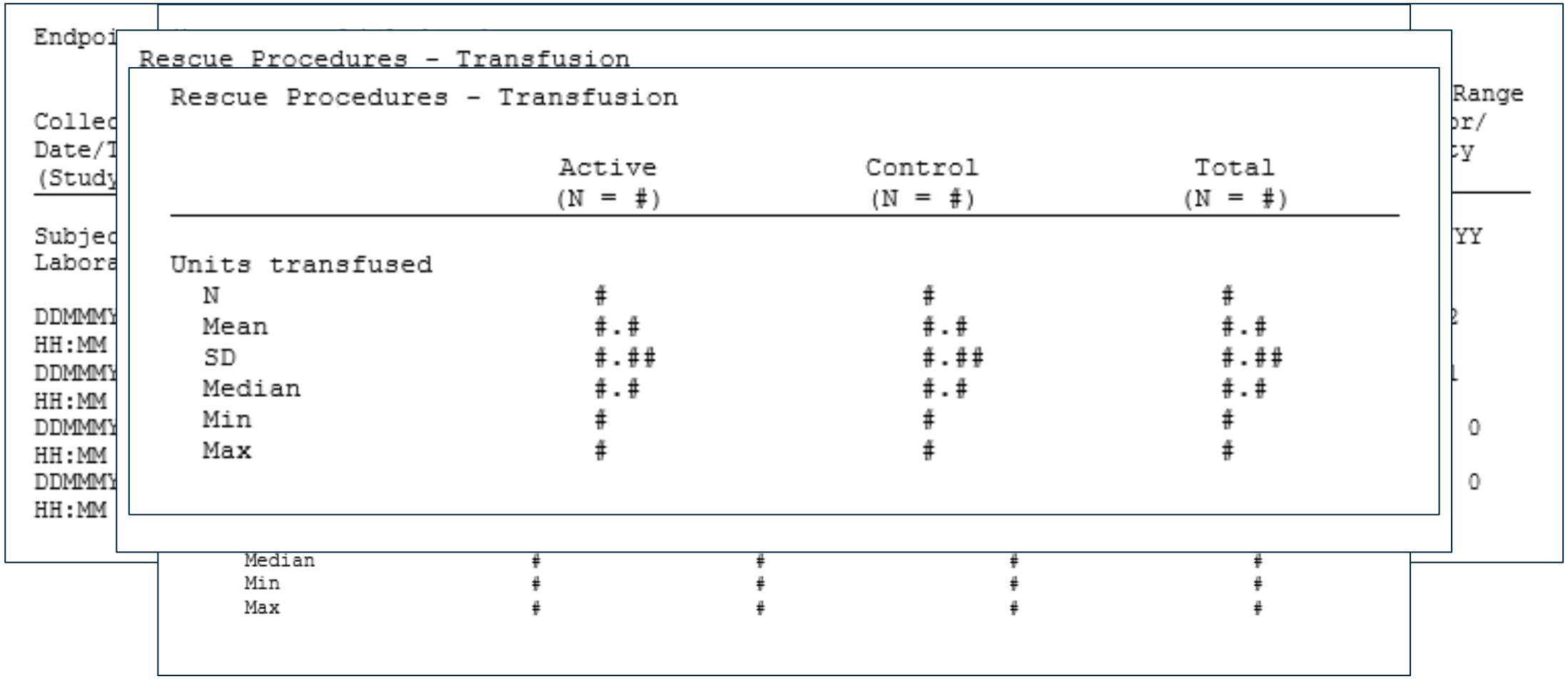
- Interventions (Rescue procedures)
- Events (SAEs, Labs with Toxicity Grade >= 3)

EXPECTED

- Primary Outcomes, Secondary, Outcomes

Traditional Reporting Emphasizes CSR Outputs

Expected examples (using mock data):



Rescue Procedures - Transfusion

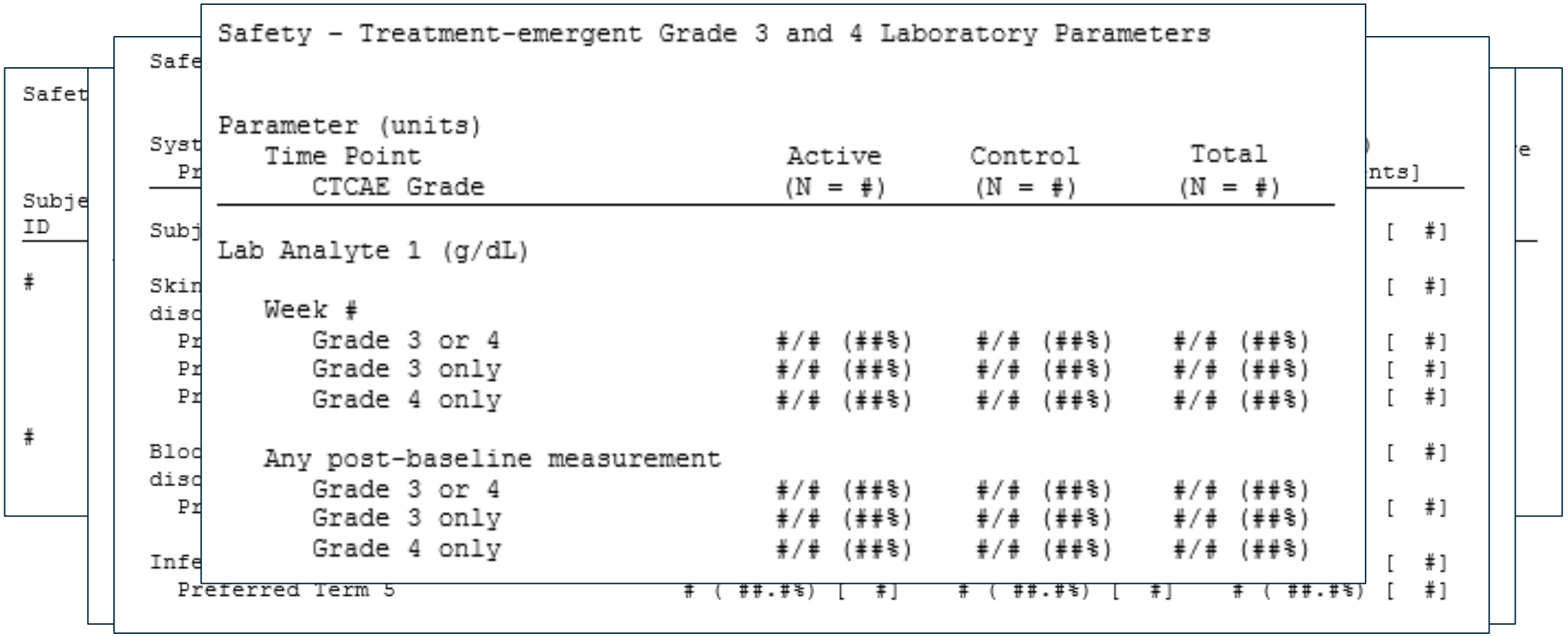
Rescue Procedures - Transfusion

	Active (N = #)	Control (N = #)	Total (N = #)
Units transfused			
N	#	#	#
Mean	#.#	#.#	#.#
SD	#.##	#.##	#.##
Median	#.#	#.#	#.#
Min	#	#	#
Max	#	#	#

Median # # # #
Min # # # #
Max # # # #

Traditional Reporting Emphasizes CSR Outputs (continued)

Unexpected examples (using mock data):

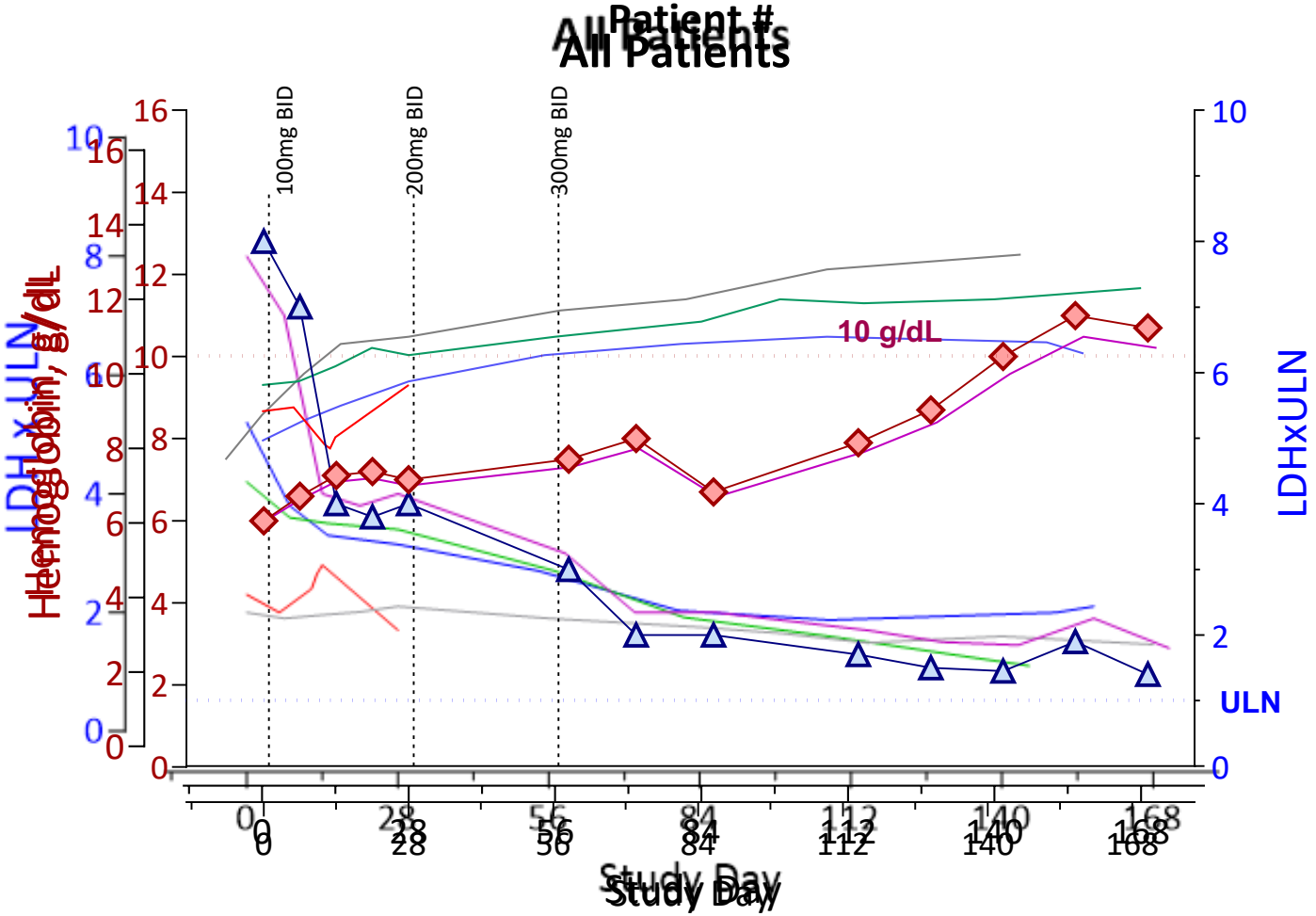


Safety – Treatment-emergent Grade 3 and 4 Laboratory Parameters

Parameter (units) Time Point CTCAE Grade	Active (N = #)	Control (N = #)	Total (N = #)	
Lab Analyte 1 (g/dL)				
Week #				
Grade 3 or 4	#/# (##%)	#/# (##%)	#/# (##%)	[#]
Grade 3 only	#/# (##%)	#/# (##%)	#/# (##%)	[#]
Grade 4 only	#/# (##%)	#/# (##%)	#/# (##%)	[#]
Any post-baseline measurement				[#]
Grade 3 or 4	#/# (##%)	#/# (##%)	#/# (##%)	[#]
Grade 3 only	#/# (##%)	#/# (##%)	#/# (##%)	
Grade 4 only	#/# (##%)	#/# (##%)	#/# (##%)	[#]
Preferred Term 5	# (##.#%) [#]	# (##.#%) [#]	# (##.#%)	[#]

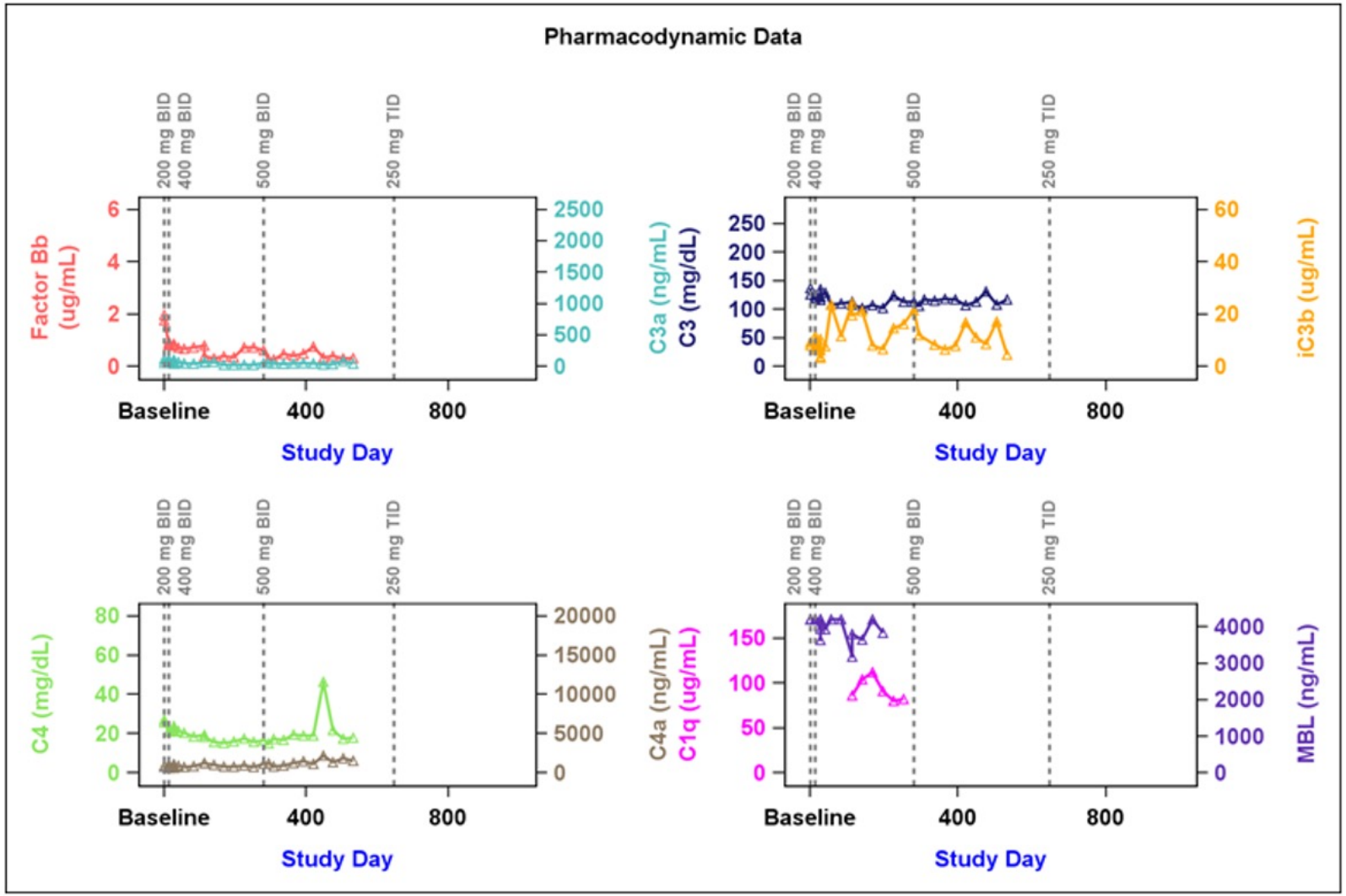
Ongoing Figures Tell the Story

Figure example (using mock data): **Endpoint 1 + Endpoint 2 + Dose Level**



Combined Displays – Lattice

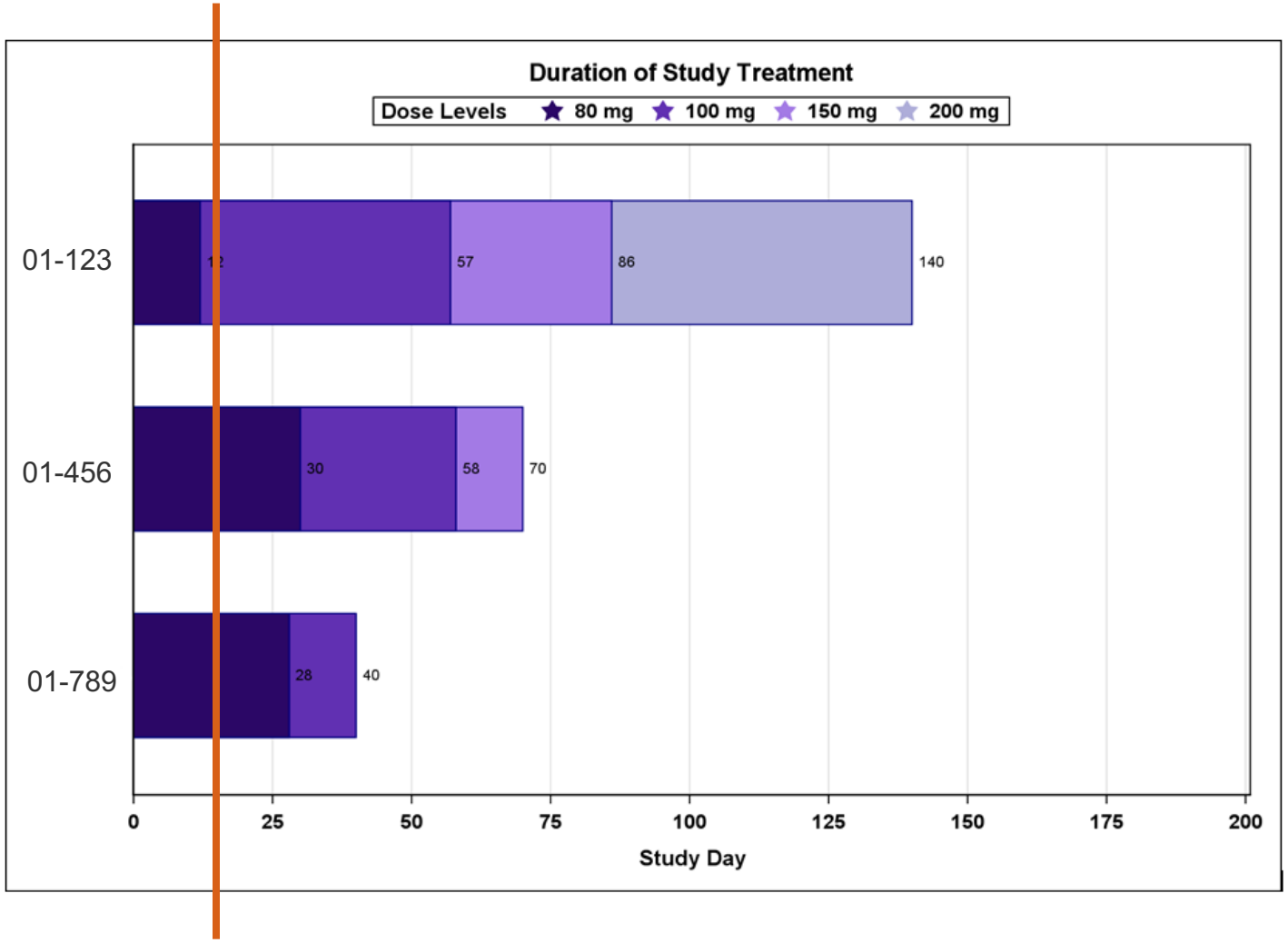
- A single page of analytes of interest could yield key insights into the study drug.
- SAS GTL, data-driven lattice graphs.[99]
- Once an initial template is built, you can add more panes (or substitute) with minimal effort.
- Data-Driven: macrotize the plotting code and use CALL EXECUTE to loop through all treated subjects. This creates one file containing lattice plots for all subjects (1 page/subject).



Combined Displays – Swimlanes

Example (using mock data):

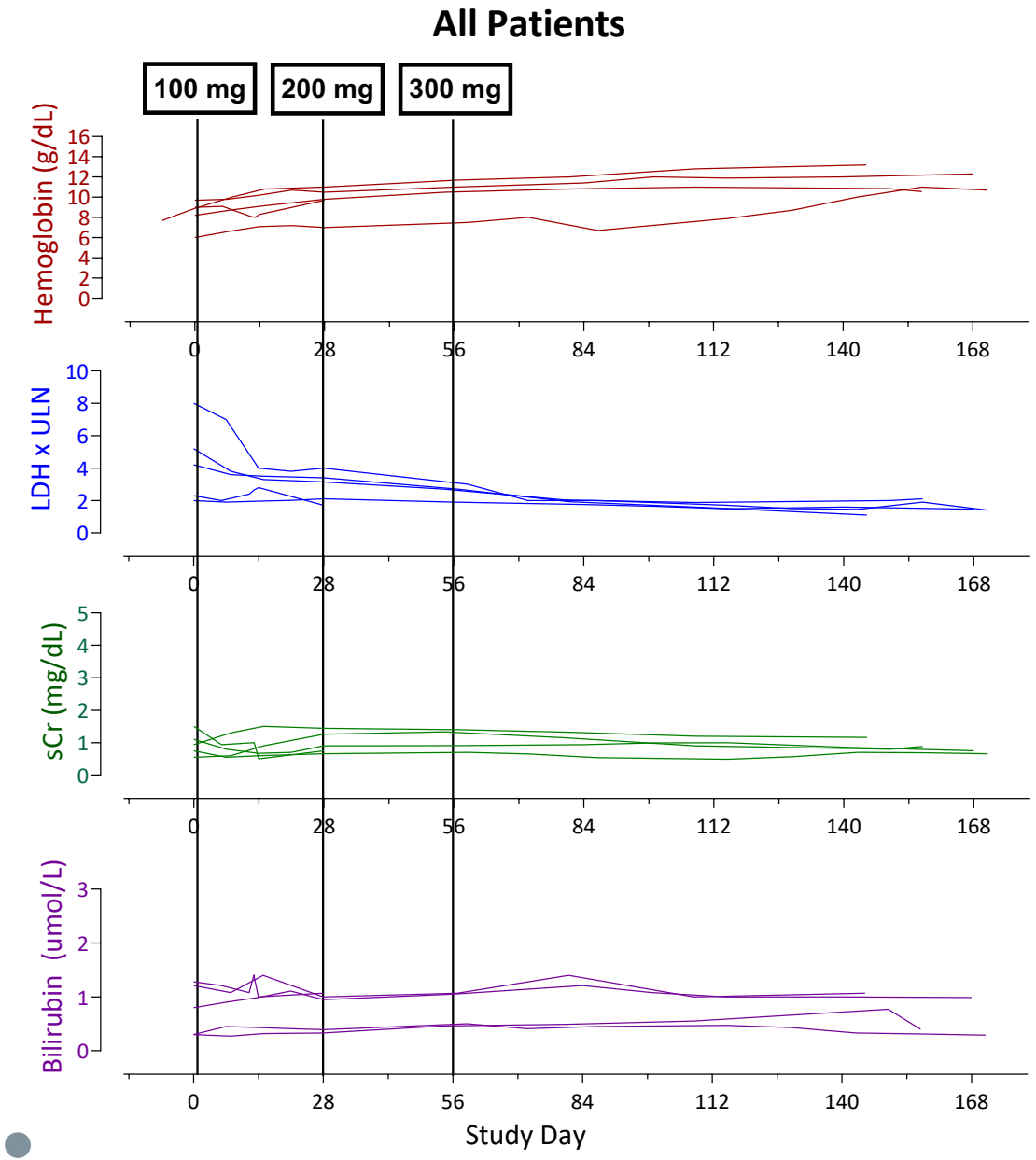
- Swimlanes clearly present treatment duration and dose progression by subject.
- This type of display is helpful if subjects are on different dose levels at the same timepoint.



Combined Displays – Stacked

Example (using mock data):

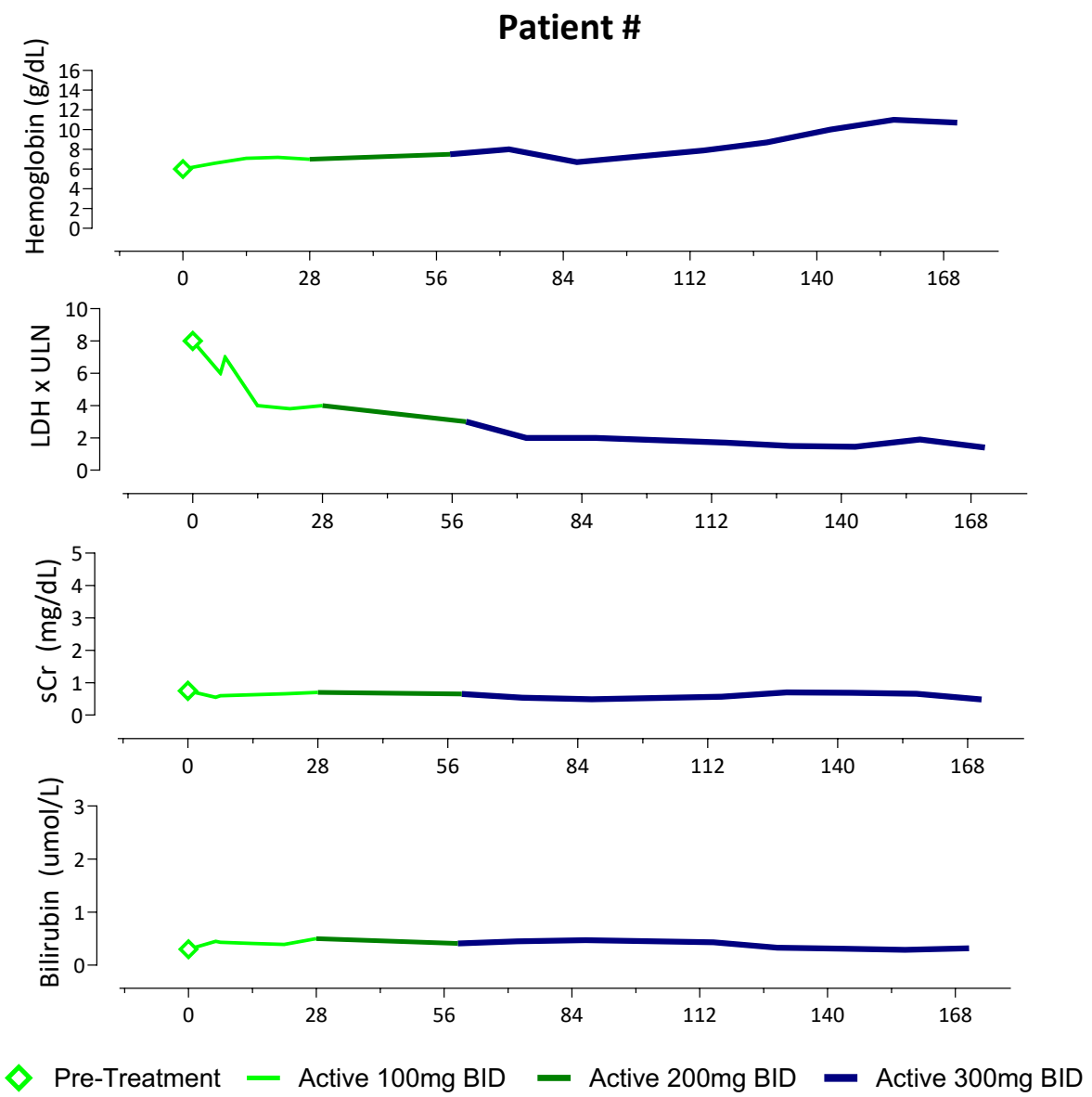
- As an alternative to other combined displays like lattice plots, we can show several analytes in a stack where all subjects are plotted and dose labels are annotated.
- Each plot is aligned by time point, and trends and outliers are easily identified.
- These plots were generated using Prism 10.[3]
- SAS JMP Clinical[2] (not shown) generates similar “stacked” plots for patient profile reporting.



Combined Displays – Stacked (continued)

Example (using mock data):

- In this by-subject stacked layout, each plot is aligned by time point and dose level is indicated by color.
- These plots were generated using Prism 10.[3]



Dynamic Listings

Filterable and searchable listings support the combined displays
Examples (using mock data):

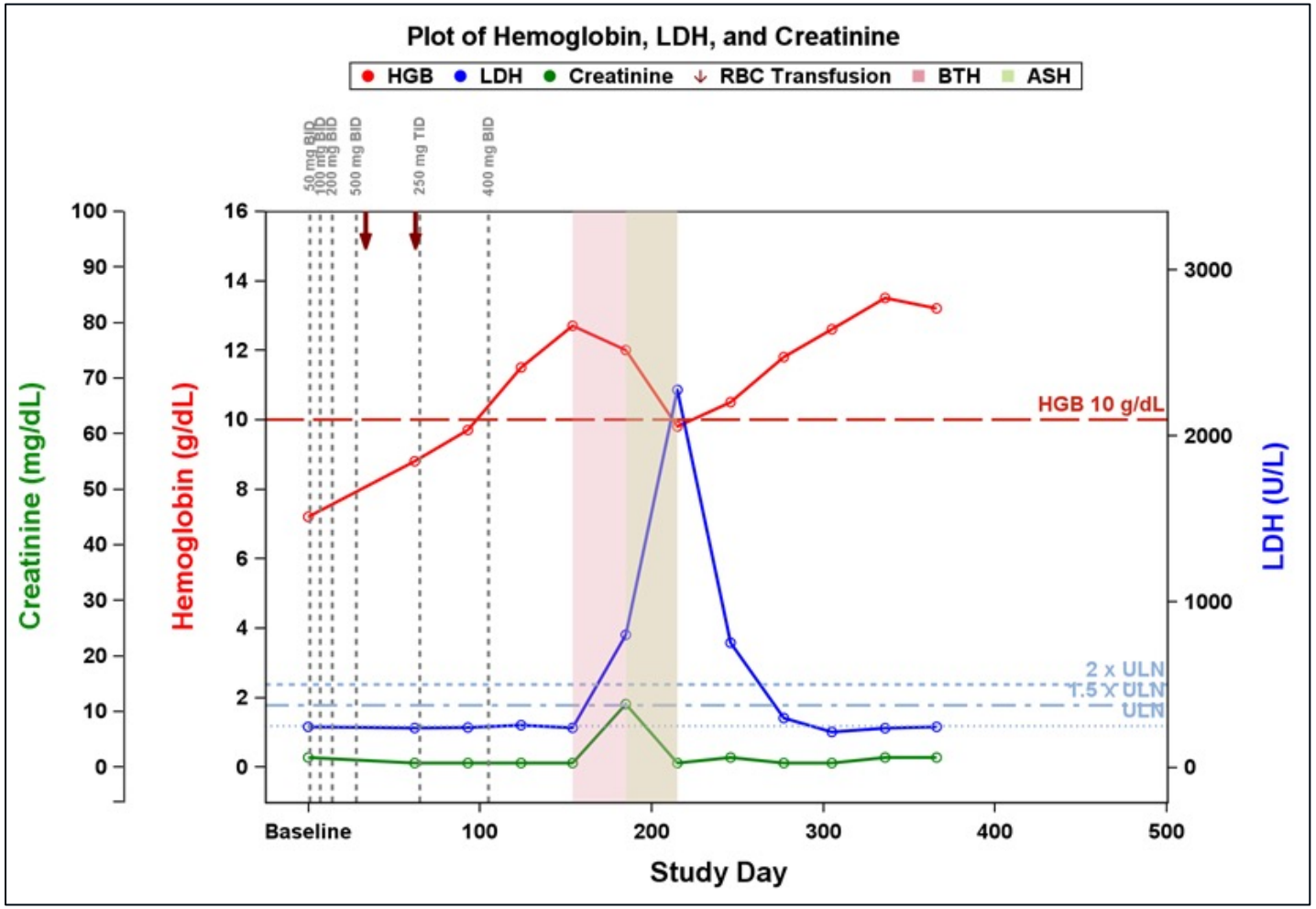
Adverse Event Data
Laboratory Data

Subject Identifier for the Study	Body System or Organ Class	Dictionary-Derived Term	Reported Term for the Adverse Event	Analysis Start Date	Analysis Start Relative Day	Analysis End Date	Analysis End Relative Day	Analysis Duration (N)	Analysis Duration Units	Outcome of Adverse Event	Action Taken with Study Treatment	Analysis Toxicity Grade (N)	Causality	Serious Event	Actual Treatment (N)
1	General disorders	Preferred Term 1	Verbatim text	DDMMMYYYY	-4	DDMMMYYYY	12	16	DAYS	Text	Text	1	Text	N	
2	General disorders	Preferred Term 4	Verbatim text	DDMMMYYYY	31	DDMMMYYYY	54	24	DAYS	Text	Text	1	Text	N	300
2	General disorders	Preferred Term 4	Verbatim text	DDMMMYYYY	59	DDMMMYYYY	84	26	DAYS	Text	Text	2	Text	N	300
2	General disorders	Preferred Term 3	Verbatim text	DDMMMYYYY	41	DDMMMYYYY	104	64	DAYS	Text	Text	1	Text	N	300
2	General disorders	Preferred Term 2	Verbatim text	DDMMMYYYY	120	DDMMMYYYY	122	3	DAYS	Text	Text	1	Text	N	100
2	General disorders	Preferred Term 2	Verbatim text	DDMMMYYYY	138	DDMMMYYYY	139	2	DAYS	Text	Text	2	Text	Y	300

Compelling Elements Combined Tell the Story

Multiple Endpoints and Events of Interest, by Subject

Example (using mock data):



In Conclusion

- When supporting decision-making and identification of trends, traditional CSR TLFs may not provide sufficient insight.
- Tell the story by creating combined displays to profile patients and show critical events + key endpoints.
- Speak with your team to find what types of figures are most useful, create sets of standard displays for efficiency.
- Filterable, dynamic Listings with dose level can provide detail.

References

1. https://www.fda.gov/patients/rare-diseases-fda
2. https://www.jmp.com/en_us/industries/data-analysis-software-for-the-pharmaceutical-industry.html
3. Prism 10 by Graphpad. https://www.graphpad.com/features

99. About SAS Graph Template Language

- (a) Fifth edition user guide dated 26 May 2022 (734 pages). Helps you understand important concepts and offers many complete code examples that illustrate frequently used features.
- (b) SAS 9.4 Graph Template Language: Reference, Fifth Edition (1612 pages). The language dictionary for GTL.
  - The Graph Template Language (GTL) is the heart of ODS Graphics. Graphs created by SAS analytical procedures and by the SAS statistical graphics procedures are generated using GTL.
  - Users who need to go beyond the graphs created by these SAS procedures can use GTL directly to design their graphs using the TEMPLATE and SGRENDER procedures.
  - There are several types of Multicell Graphs[(a)] (like the lattice plots shows in this presentation) which can be data-driven.

Thank You!

- Diana Streng dstreng@biocryst.com
- Mike Litzsinger mlitzsinger@biocryst.com

Backup Slides

https://www.mariowiki.com/Power_Star

A Note on Presenting Means

Typical tables and figures present subject counts and **means** (lab analytes, Questionnaires scores, etc…) by treatment group and timepoint.

With small populations, one or two outliers could skew the means.

Consider presenting the **median**, which will exclude very large or very small values and may better reflect typical subject experience.

About the by-Subject Plots

To generate by-subject lattice plots, I use the general framework below:

- Proc Format to define the axis tick marks
- Use a Style template to distinguish groups (eg, colors applied consistently per subject)
- SAS GTL Template > lattice
  - SERIESPLOT within each lattice cell (using IFN function to select parameter of interest from the full lab dataset)
  - Dynamic variables for subject-specific information (like SEX, AGE, Treatment, Dose Level information…)
  - PROC SGRENDER with ANNOTATE option for marking events of interest :
    - Transfusions – use ARROW function to add arrow markers across the top of the plot
    - Hemolysis events – use POLYGON function to create a shaded, transparent block
  - DRAWTEXT is used to apply rotated labels (to make things fit in space provided)
  - Conditional code is allowed within GTL (eg, if you need to add a reference line specific to only female subjects)
  - REFERENCELINE or DROPLINE for adding reference information like dose levels or normal ranges

About the by-Subject Plots (continued)

In the PROC TEMPLATE, I use dynamic variables to send in static information like subject SEX, page number, start dates for different dose levels, and discontinuation dates.

```
proc template;
   define statgraph LABS_lattice3;
   dynamic _pg _sex _sd50 _sd100 _sd200 _sd400 _sd500 _discdy;
     begingraph;
```

I create a subject list dataset, where the subject- specific macro variables are set, and then assign each subject a temporary number (&i.)

Keep in mind that dynamic variables DO NOT RESOLVE to the actual value in the dataset variable. So, within PROC SGRENDER, I set these dynamic variables to their values using nested macro variables – a combination of a temporary subject number (&i.) and the desired value (&sd50, the start day of dose level 50mg).

I use a macro %DO loop for each subject and page (since each page may have a different template) inside which I call PROC SGRENDER.

```
%do i = 1 %to &subjcount.;
  %do j = 1 %to 3; /*for each page, 3 pages per subject*/
    %if &j.=3 %then %do;
       proc sgrender data = Templab(where=(pg=&j.)) template = LABS_lattice3;
      dynamic _sex="&&sex&i." _sd50=&&sd50&i _sd100=&&sd100&i _sd200=&&sd200&i _sd400=&&sd400&i _sd500=&&sd500&i
              _discdy=&&discdy&i;
      where subjidc="&&subjidc&i.";
      by trtgrp subjidc pg;
      run;
    %end;
  %end; /*over each page, j*/
%end; /*over each subject, i*/
```