

Instant Gratification: Calculating Resource Needs Based on Operational Data using an Interactive Tool

Melissa Hill, MPH, Bill Baker, MS, Cd3 Inc.

ABSTRACT

In SAS® Visual Analytics, we leveraged existing operational metrics and a few cleverly placed user-supplied parameters to create a custom, interactive and user-friendly calculator for our clinical operations team, which extrapolates required monitoring days based on current electronic data capture (EDC) system status.

INTRODUCTION

By definition, data visualization is not a new concept. Visualizing data is an art that programmers have been perfecting for years – building reports that display information in ways that make interpretation intuitive and straightforward. Newer data visualization software like SAS® Visual Analytics have made the art of data visualization something much more dynamic and interactive. With interactivity comes efficiency in the form of instantaneous user feedback.

Based on current EDC status counts of outstanding queries and forms to be monitored our project managers calculate required monitoring days for a site visit, using the following formula:

$$\text{Visit Days} = \frac{([\text{open queries} * \text{minutes per query}] + [\text{forms to be monitored} * \text{minutes per form}])}{(60 * \text{hours permitted by the site per day})}$$

In each study, the number of minutes allotted to a form for monitoring or a query for closing may vary. Similarly, the daily limit of hours permitted for a visit, may vary. Using readily available query and form counts, we built an interactive calculator for our operations team, which allows them to provide study specific time estimates and calculate required monitoring days. The tool provides the Clinical Research Associate (CRA) manager with an easy way to forecast monitoring needs and offers the CRA a window into data entry and cleaning activity at the site since the last monitoring visit.

Broadly speaking, we accomplished this in four steps:

1. Extract, prepare, and upload EDC Metrics
2. Create the new report
3. Derive needed values
4. Finalize the Report

EXTRACT, PREPARE, AND UPLOAD EDC METRICS

To develop a single report like this one, all input data needs to be in a single data set. At Cd3, EDC data is extracted, processed, and pushed up to SAS® Visual Analytics at regular intervals using automated functionality supported by our EDC system and our SAS® software environment. As part of this process we store a dataset comprised of one record per electronic case report (eCRF) form, from which we can derive multiple different operational metrics. Below a few sample rows of the dataset are displayed in Table 1.

unique_fom	site	subject	fom_status	monitoring_status	open_queries
006-003 . 111 . Tumor Assessment - Target Lesions-1	006	006-003	Complete	Partially Monitored	1
006-004 . 001 . Cancer History	006	006-004	NotComplete	Partially Monitored	1
006-004 . 001 . Hematology	006	006-004	NotComplete	Partially Monitored	1
006-004 . 001 . Medical and Surgical History/Physical Findings	006	006-004	NotComplete	Partially Monitored	5
006-004 . 001 . Previous Anti-Cancer Therapy	006	006-004	NotComplete	Not Monitored	7
006-004 . 001 . Prior Cancer Surgery	006	006-004	NotComplete	Partially Monitored	1
006-004 . 001 . Unanalysis	006	006-004	NotComplete	Partially Monitored	2

Table 1. Sample Data

CREATE THE NEW REPORT

Once the data is available in Visual Analytics as a LASR data table, it is brought into a blank report, where the remaining values are created. Open a blank report in the Report Designer and add the data source. From the Objects panel, select listing and drop it in the main design space. The end goal is to display required visit days by site. To get started, the value *Site ID* can be added to the data listing object. In the listing, there should be one row per site as shown in Figure 1. Formatting and design can be performed at the end, during development the focus is on accurately deriving and displaying values. As values are derived, we add them to the listing and confirm that the calculations are working properly.

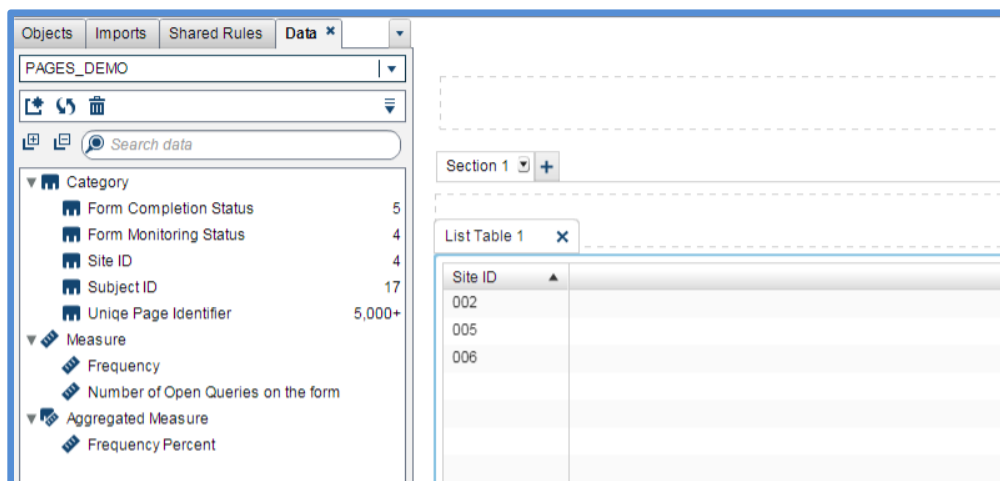


Figure 1. New Report

DERIVE NEEDED VALUES

Some additional calculated variables are required based on the data that we imported. We use calculated items, aggregated values, and parameters to derive and display all of the remaining data points in our report.

CALCULATE THE NUMBER OF FORMS THAT REQUIRE MONITORING

Since not all forms require monitoring at each visit, we need to identify the forms that currently require monitoring and then quantify those forms. To identify them, we create a new variable which is populated with the *Unique Form Identifier* value for forms that need to be monitored, and is missing for all others. We do this by expanding the data actions drop down menu and selecting 'New Calculated Item...' A dialogue box opens and we populate the options as shown in Figure 2.

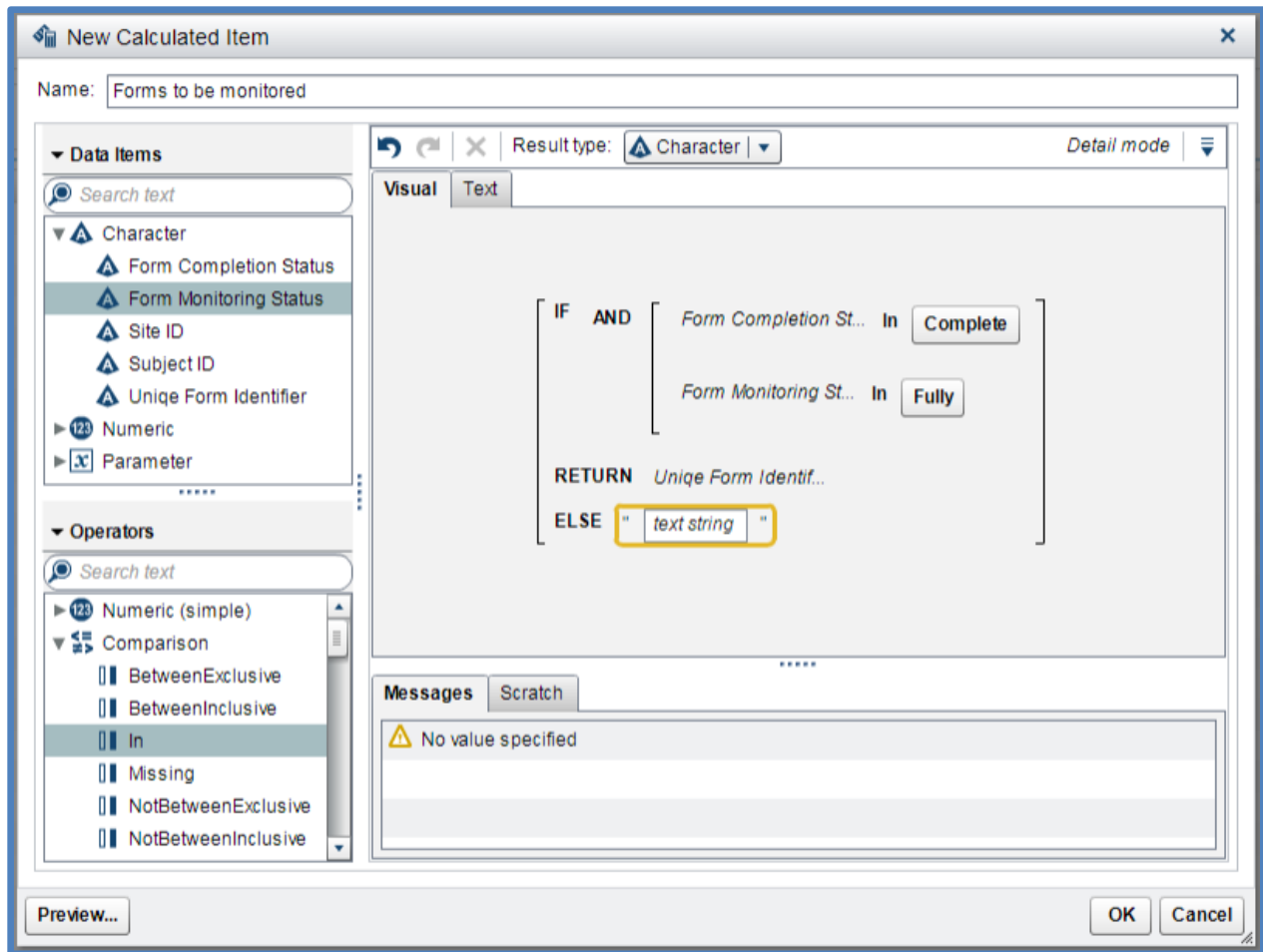


Figure 2. Calculated Item

We use this new value to create an aggregate value for the distinct count of unique forms requiring monitoring. Right clicking on the new variable and selecting 'Create Distinct Count...' creates a new aggregate measure as shown in Figure 3. Using an aggregate measure allows the count to change fluidly within displays depending on the grouping.

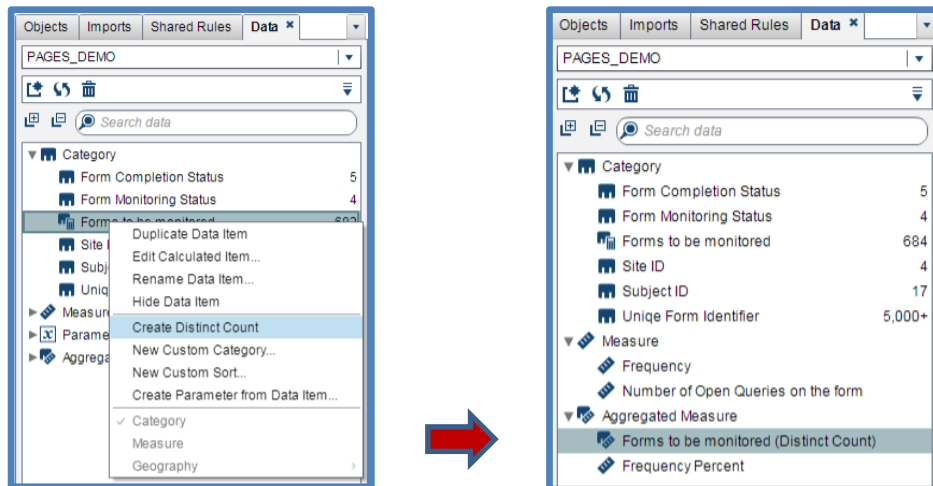


Figure 3. Aggregated Measure for Distinct Count

DISPLAY THE NUMBER OF OPEN QUERIES BY SITE

The *number of open queries* value is numeric in SAS® dataset and is therefore imported as a measure. In Visual Analytics, measure values can be set to display a sum value using the aggregation type 'sum'.

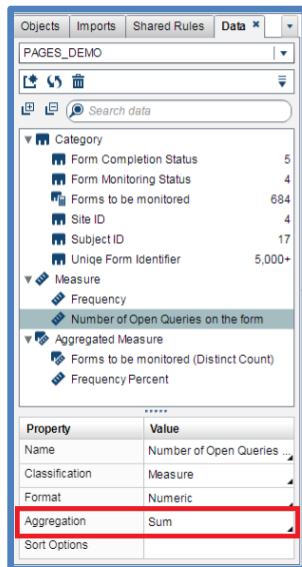


Figure 4. Measure aggregation

At this point in time, we add the number of open queries column and the forms to be monitored values to the listing as shown in Figure 5. Both values are aggregated by the Site ID and we have all the metric data we need to perform our calculations.

Site ID ▲	Number of Open Queries on the form	Forms to be monitored (Distinct Count)
002	107	374
005	26	188
006	88	122

Figure 5. Interim Listing

CREATE USER SUPPLIED PARAMETERS

The values for minutes per form, minutes per query, and hours permitted by the site per day need to be user supplied. SAS® Visual Analytics reports support user supplied data in the form of parameters; we created a new parameter for each of these three items. To create a new parameter, expand the data options drop-down and select 'New Parameter...' as shown in Figure 6.

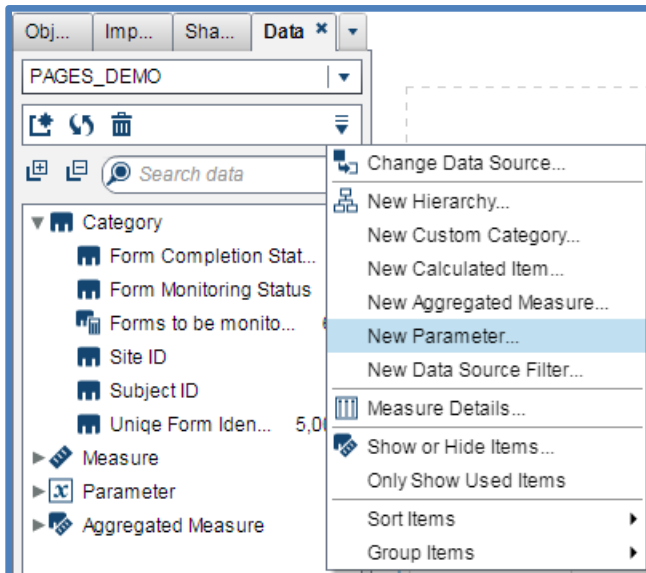


Figure 6. New Parameter

In the dialogue box displayed in Figure 7 we set the options to suit the needs of the project. Below are the specifications for the *minutes_per_form* field. Note that it has been assigned as numeric and we have provided a minimum, maximum, and current value. The current value will appear as the default in our interactive calculator, but can be changed by the user to any value between the minimum and maximum.

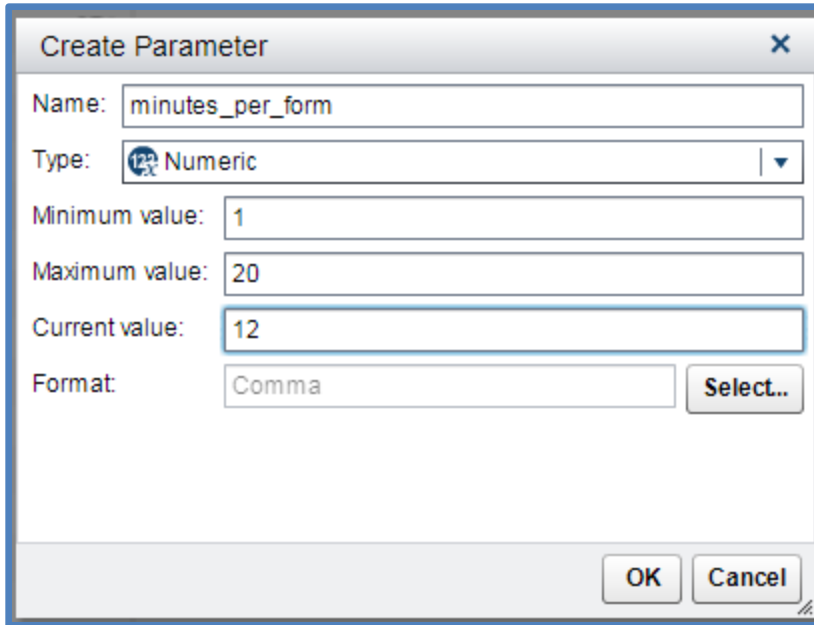


Figure 7. Create Parameter Dialogue Box

We also create two additional parameters *minutes_per_query* and *hours_per_visit*. When the parameters are ready for use, they can be seen in the parameter section of the data window (Figure 8).

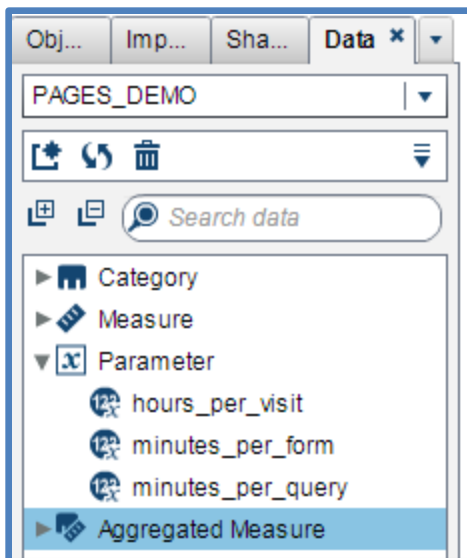


Figure 8. Parameters in the data window

Parameters do not appear in the listing, instead they are displayed in Text Input controls, which are selected from the Objects panel on the left side of the design space (Figure 9).

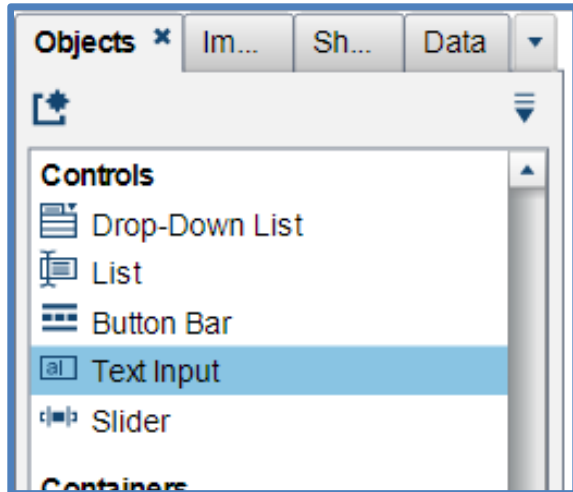


Figure 9. Text Input

We place a Text Input control in the design space and assign one of the newly created parameters to the object using the Roles panel on the right side of the design space. In the properties panel, also on the right side of the design space, we populate the Title with the label we would like to see displayed in our tool for the assigned parameter. We repeat this process for each new parameter until they are displayed and labeled as desired (Figure 10).

The image shows a web form with three text input fields. The first field is labeled 'Minutes per query' and contains the value '5'. The second field is labeled 'Minutes per form' and contains the value '12'. The third field is labeled 'Hours per site visit' and contains the value '6'. Below the text input fields is a table with three columns: 'Site ID', 'Number of Open Queries on the form', and 'Forms to be monitored (Distinct Count)'. The table contains three rows of data.

Site ID	Number of Open Queries on the form	Forms to be monitored (Distinct Count)
002	107	374
005	26	188
006	88	122

Figure 10. Text Input Displays

CALCULATE THE REQUIRED NUMBER OF MONITORING DAYS

Next, we create the new aggregated measure which will calculate the required number of monitoring days as shown in Figure 11. The specifications for that aggregated measure reflect the equation that our operations team uses to estimate the value.

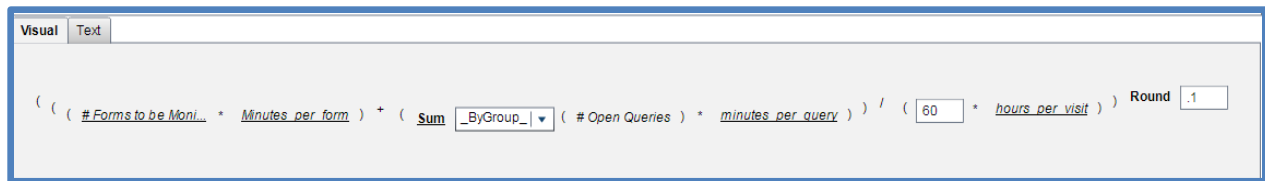


Figure 11. Calculated field for monitoring days

This value is added to the listing, as shown in Figure 12, and manually checked for accuracy before we do any formatting or design enhancements. Note that we needed to use an aggregated measure for our final calculation, because we incorporated an aggregated measure in our derivation.

Site ID ▲	Number of Open Queries on the form	Forms to be monitored (Distinct Count)	Monitoring Days
002	107	374	14.00
005	26	188	6.60
006	88	122	5.30

Figure 12. Calculated field in the report

FINALIZE THE REPORT

Thoughtful design and formatting bring aesthetic appeal and user-friendliness to any tool or visualization. Below is the final version of our interactive tool for CRA time allocation, shown in Figure 13.

Parameters in the orange boxes at the right are user controlled.
They can be changed by placing the cursor in the field and entering a new value.

Minutes per query should contain the average time (minutes) it takes the CRA to resolve a query.
Minutes per form should contain the average time (minutes) it takes the CRA to monitor a form.
Hours permitted by site per day should reflect site permitted daily time limits (hours).

The value *Monitoring Days* in the table below is calculated based on these user supplied parameters and the operational metrics displayed in the table.

Minutes per query

5

Minutes per form

12

Hours permitted by site per day

6

Site ID ▲	Monitoring Days	# Open Queries	# Forms to be Monitored
002	14.00	107	374
005	6.60	26	188
006	5.30	88	122

Figure 13. Final Report

We made a few design enhancements to the basic display. First, the parameters were assigned one background color and the listing a different color. This is a visual cue to the user that the parameters are inherently different than the listing. Accordingly, we also added an instructional text box explaining how the tool should be used.

Variable names are another way to improve report design. We often find that variable names derived during development are not necessarily clear to the end user. Changing variable names can be easily

accomplished by right clicking on the variable in the data panel (left side design space) and selecting 'Rename Data Item...' In this project we changed two variables after development. *Number of Open Queries on the form* was accurate in our source data but in the display, we were showing this value as a sum for the site, so it became '# Open Queries' in our display. Similarly, when we created the aggregated value *Forms to be monitored (Distinct Count)* the phrase (Distinct Count) was automatically appended to the source variable name by SAS® Visual Analytics. This use of terminology is confusing to our users, so we renamed it as simply '# Forms to be Monitored'.

The final design decision was to change the background color of the calculated column for monitoring days in our display. We used a basic display rule to color the entire column one color, but the same method could be used to set a color guide by range. For instance, imagine that policy indicates that site visits should not be longer than three days, then values under three could be assigned one color and values over the three-day limit could be assigned a different color.

Formatting enhancements included changes to font and selection of the wrap text option for the column headers in the listings. Choosing these options together optimizes the likelihood that content is displayed clearly on any monitor, at any zoom, and in pdf format if the display needs to be exported.

CONCLUSION

At Cd3 Inc., we use SAS® Visual Analytics as our reporting platform for both internal and external team members. Internal users see similar tools for data review and monitoring across all clients and projects. External teams, enjoy real-time access to study related data which is processed and supported by Cd3, but accessible outside our firewall.

The interactive report described here, allows data consumers to derive the information they need in a self-serve fashion. The calculator is standard enough that it can be deployed across protocols, installed once during study set up, and then available for the duration. This combination of characteristics minimizes the burden on both operational and programming teams of requesting, developing and re-producing reports on an as-needed basis throughout the course of the trial. Beyond that, resources and effort are minimized, when we can develop such a tool based on data that is already readily available in our system.

CONTACT INFORMATION

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

Melissa Hill
Cd3 Inc.
203-404-7500
mhill@cd3inc.com
[LinkedIn/Hill](#)

Bill Baker
Cd3 Inc.
203-500-6009
bbaker@cd3inc.com
[LinkedIn/Baker](#)

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.

Other brand and product names are trademarks of their respective companies.