
Anaplan
Driving a new age of
connected planning

Model Building Tips and Tricks

Toronto User Group
July 12, 2018

Anaplan

Ryan Kohn

Senior Business Partner
Customer Success

Anaplan

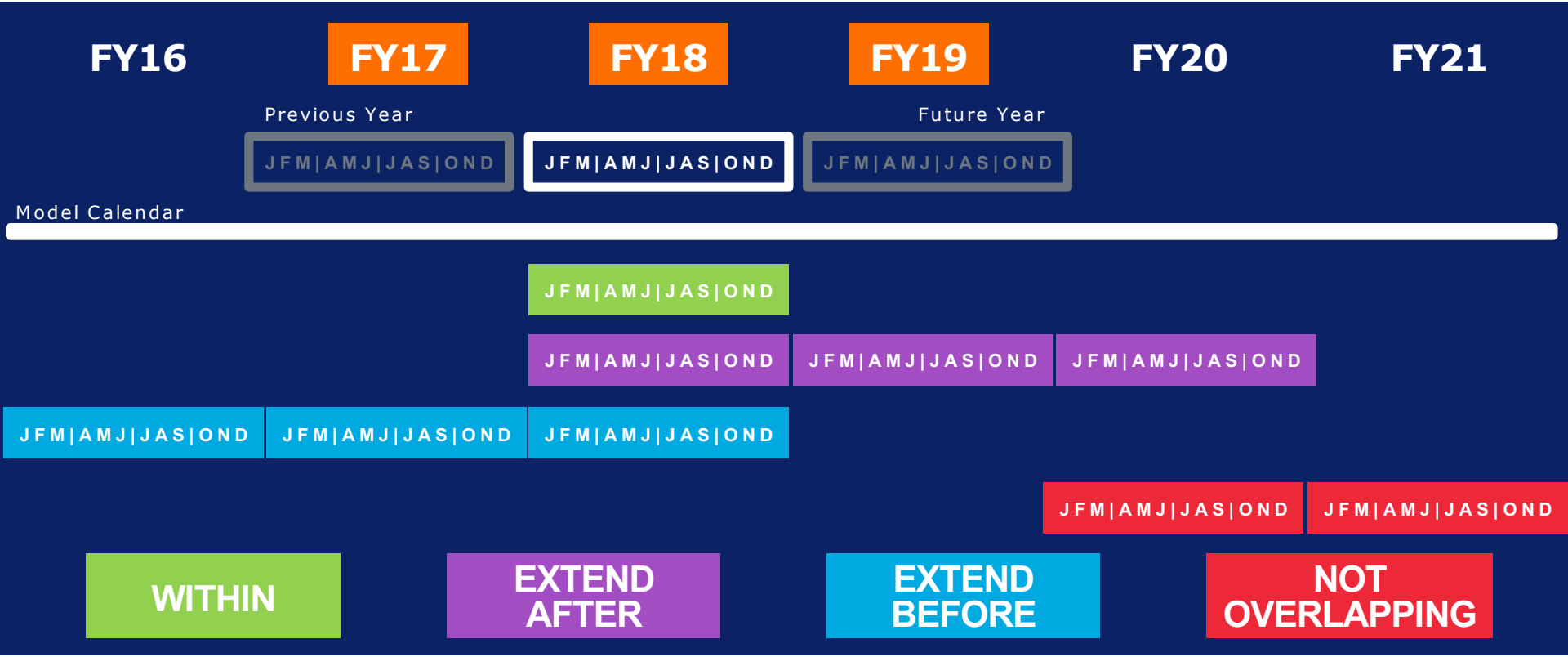
Agenda

Time Ranges

User Input Validation

Time Ranges

Introduction to Time Ranges



Why Use Time Ranges?

- Reduce Model Sparsity
- Eliminates need to set up alternate time lists
- Increases model flexibility
- Improved model efficiency

Start off using your model calendar as the base and then use time ranges when they will help provide the benefits listed above

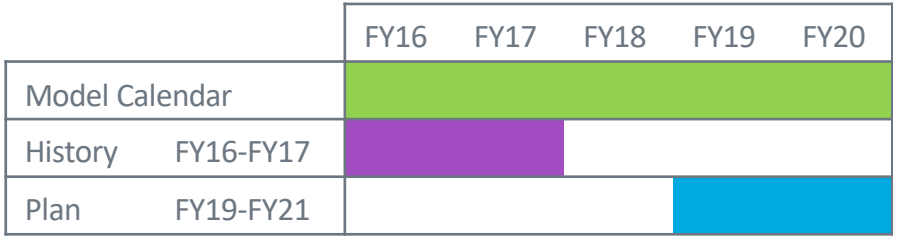
Naming Convention

- FYxx-FYxx. For example, FY16-FY18, or FY18 for a single year
- FY16-FY19 Q (to signify Quarter totals)
- FY16-FY19 QHY (Quarter and Half Year totals)
- FY16-FY19 HY (Half Year totals only)

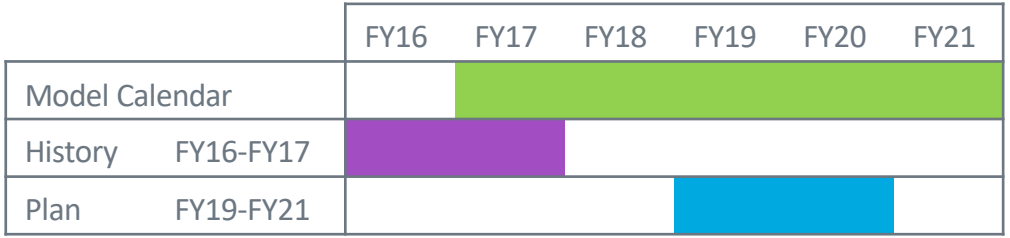
DEMO

Time Ranges rolling to a new Year

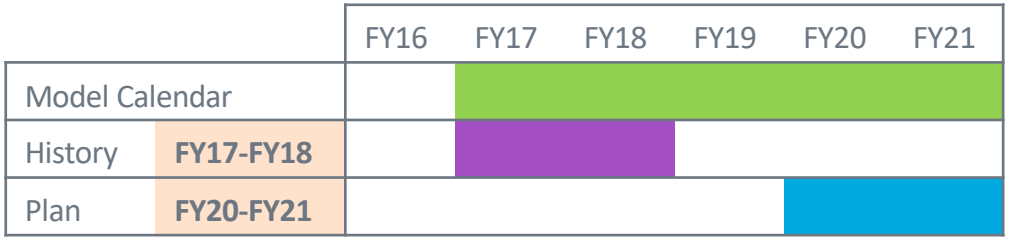
Starting Model Calendar
Two time ranges



Roll Model calendar forward
Time ranges stay the same



Determine how to handle Pr Yr data
Update / Rename time ranges



Additional Considerations

- ALM
  - Time ranges are considered structural data
  - All changes should be made in a development model
- Sum & Lookup
  - Consider the time periods used for line items before writing these formulas
  - Make sure the inputs and results have overlapping time periods
- Time Series Functions (ie. Post, Lag, MovingSum, Cumulate)
  - Make sure the inputs and results have the same range

Looking for more information

Time Ranges:
https://help.anaplan.com/anapedia/Content/Modeling/Dimensions/Time%20Ranges.htm

https://community.anaplan.com/t5/Training-Classes/Introduction-to-Time-Ranges/ta-p/30542

https://community.anaplan.com/t5/Shared-Best-Practices/Time-Range-Application/ta-p/30479#toc-hId--2076447299

User Input Validation

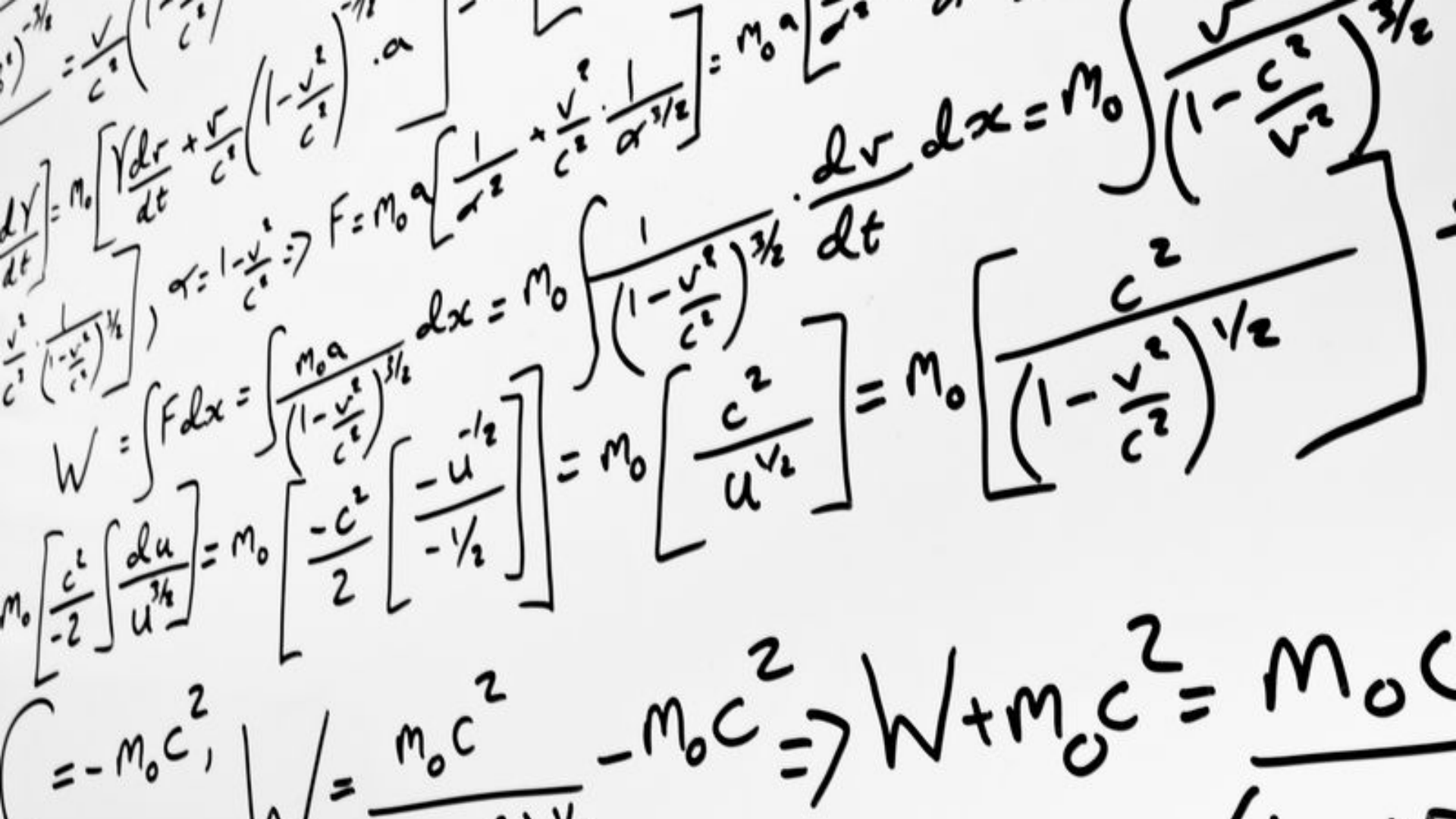
Why validate user input?

- Lower risk of incorrect results downstream
- Reduce time spent fixing data later
- Improve end user experience by providing immediate feedback

Input Validation in Anaplan

Native in Anaplan	Some assembly required
• Data formatting (e.g. numbers, dates, preset list of values)	• Required vs optional fields • Ranges of values • Business rules

Conditional formatting for validation



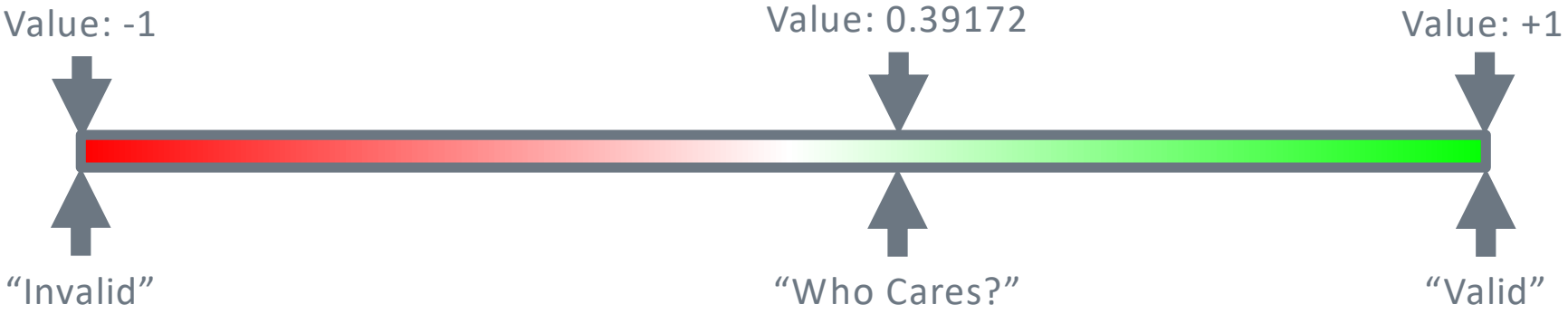
Continuous vs Discrete Variables

- A **continuous variable** can take on any value between two specific values
- A **discrete variable** can only have specific values

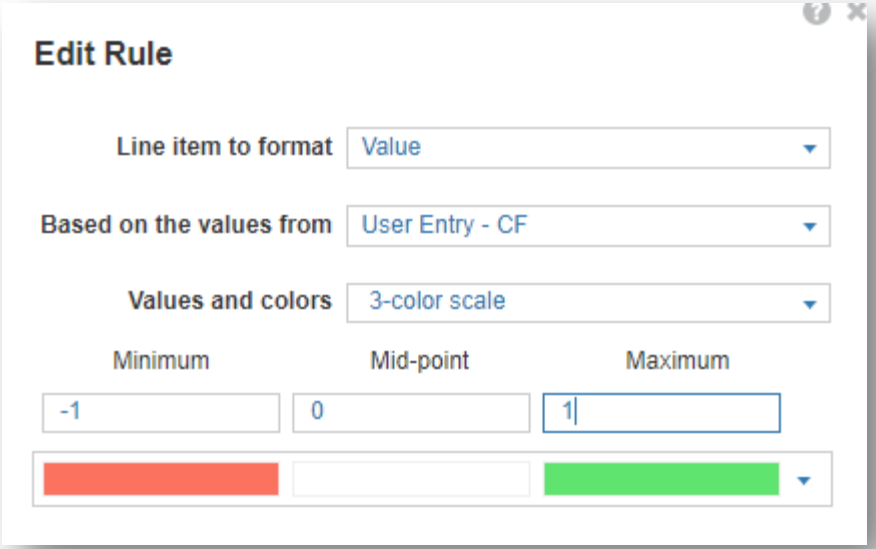
Continuous vs Discrete Variables

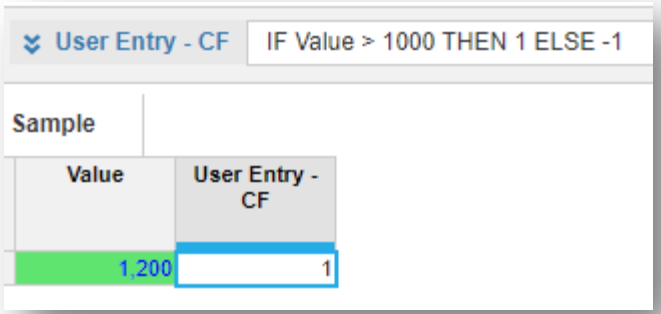
- Reminder: conditional formatting line items are continuous (i.e. numeric)
- However, **results of validation** are discrete (i.e. Valid, Invalid, or Partially Valid)
- We need to “fake” discreteness in order to configure conditional formatting for validation

Continuous to Discrete Values

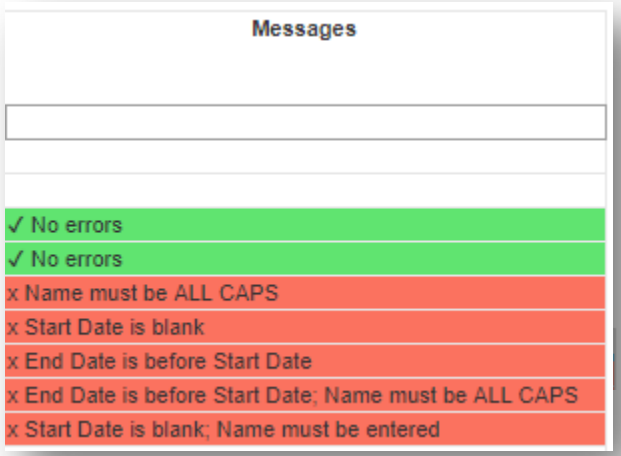


In Anaplan

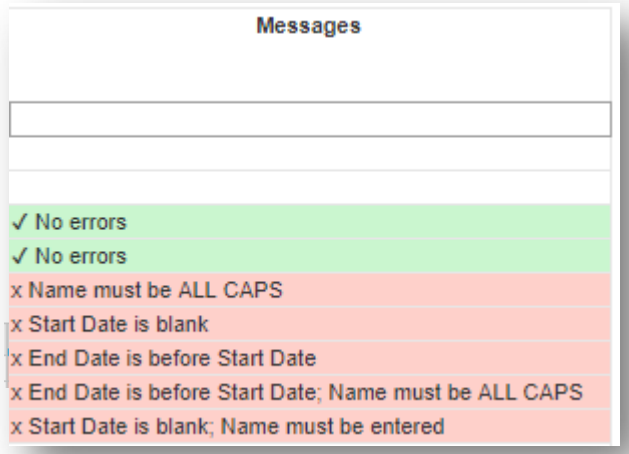

Edit Rule
Line item to format
Value
Based on the values from
User Entry - CF
Values and colors
3-color scale
Minimum
Mid-point
Maximum
-1
0
1

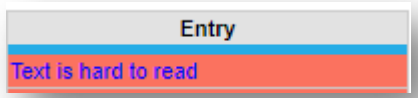

User Entry - CF
IF Value > 1000 THEN 1 ELSE -1
Sample
Value
User Entry - CF
1,200
1

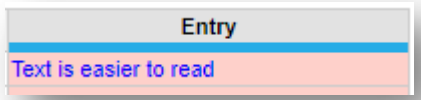
Softer Colors



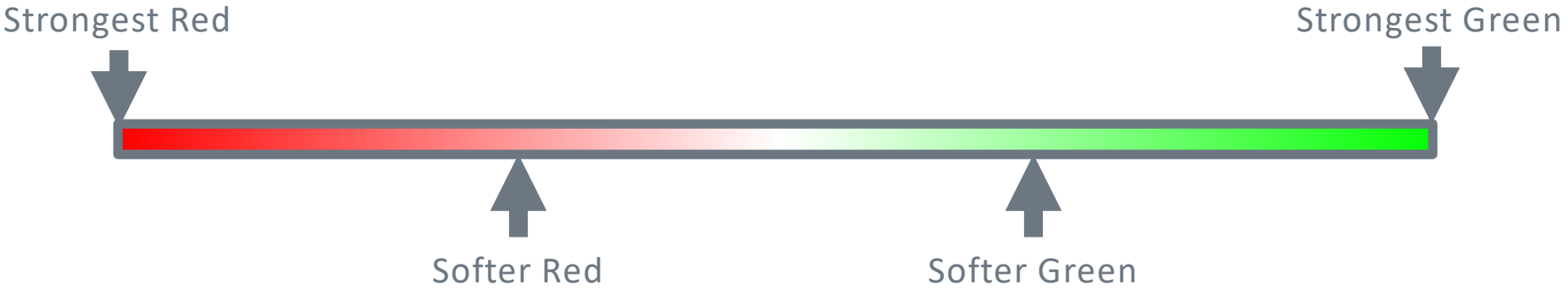
VS



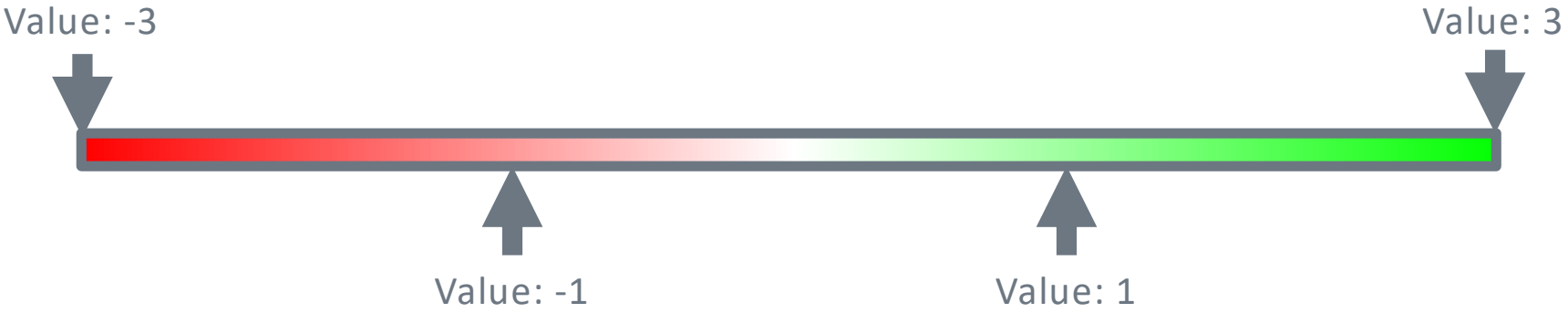




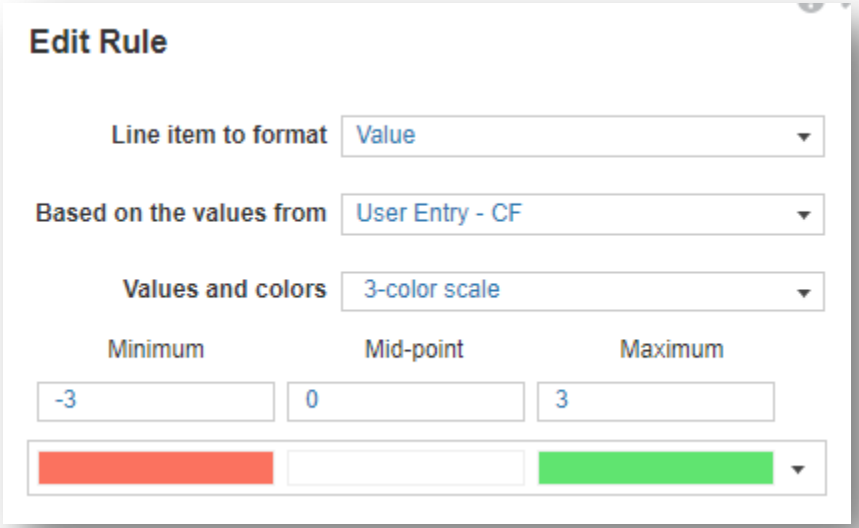
Softer Colors

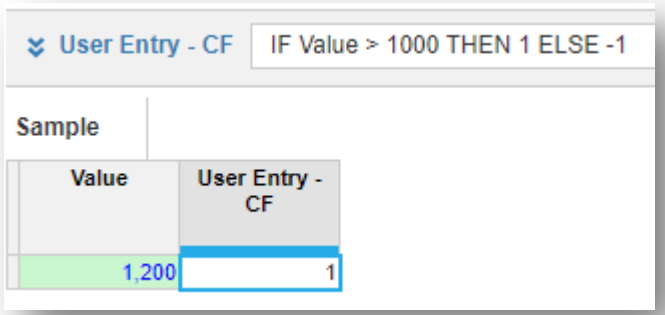


Softer Colors



In Anaplan


Edit Rule
Line item to format
Value
Based on the values from
User Entry - CF
Values and colors
3-color scale
Minimum
Mid-point
Maximum
-3
0
3

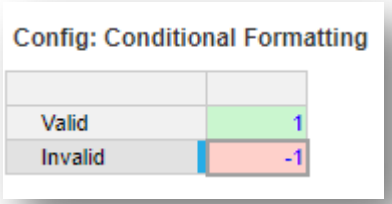

User Entry - CF
IF Value > 1000 THEN 1 ELSE -1
Sample
Value
User Entry - CF
1,200
1

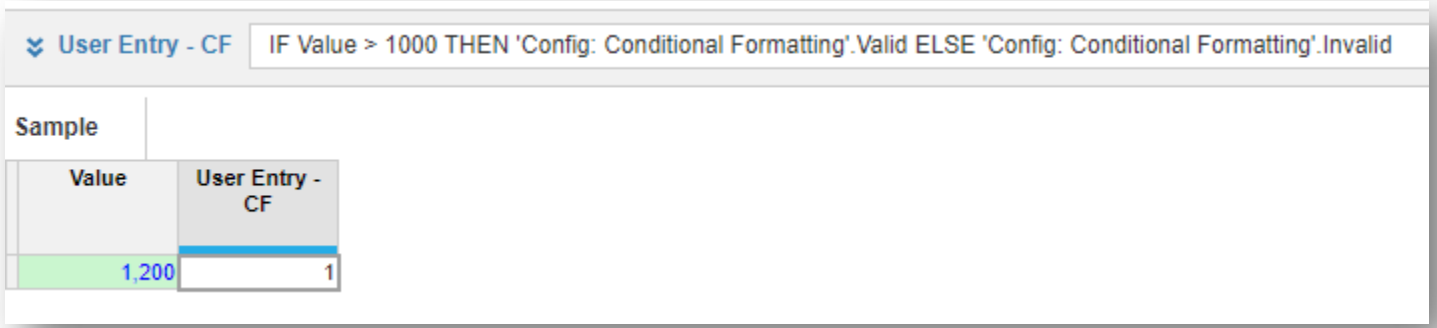
DEMO

Advanced: Centralizing the Color Scheme

- **Common Configuration** module to define the intensity of the colors used
- **Tip:** Maintain a Model Builder standard for conditional formatting
    - E.g. Always use “-3”, “0”, and “3” for setting up conditional formatting, and use a common configuration module to define the discrete values
- **Consistency is key!** Especially as you build out your COE and have multiple model builders across the organization. Building these standards into your models allows end users to have a similar experience across models.

In Anaplan


Config: Conditional Formatting
Valid
1
Invalid
-1


User Entry - CF
IF Value > 1000 THEN 'Config: Conditional Formatting'.Valid ELSE 'Config: Conditional Formatting'.Invalid
Sample
Value
User Entry - CF
1,200
1

Anaplan