Take the Load Off Your Electrical (Engineer)

Jon Atkinson
MEP Application Engineer

Blake Guither
Professional Mechanical Engineer

About the Speakers

Jon Atkinson is currently an MEP Application Engineer with CAD Technology Center. He started in the architectural field in 1996, but moved to engineering in 2010. Jon has been a CAD Manager, BIM Manager, Revit Developer, even a lowly dafter, and holds a BSTM in CAD Management.

About the Speakers

Blake Guither is a Professional Mechanical Engineer with Gausman & Moore responsible with all aspects of HVAC, plumbing, and fire protection design for commercial buildings. He has been in the industry for 10+ years. Blake has done extensive work with energy models and model calibration while also obtaining a Building Energy Modeling Professional (BEMP) certification from ASHRAE.

Synopsis

Learn how to use complex formulas in families and schedules to analyze designs and communicate between mechanical and electrical disciplines. Make your electrical engineer's life easier by providing them the correct circuiting information.

Learning Objectives

- Learn how to get data flowing from one family to another
- Learn how to use formulas to automate tedious calculations
- Learn how to use schedules to maximize productivity
- Learn how to populate text parameters using lookup tables

… but first

In order to begin the electrical coordination process we first need to address the mechanical elements and workflow. By getting information moving and calculating in our mechanical families we can ensure that ours, and our electrical designers', efforts are optimized.

Workflow for optimal coordination

- Spaces, Zones, and Load Calculations
- Space Airflow
- VAV Sizing
- Equipment Sizing

Workflow for optimal coordination

- Electrical Coordination
  - Within shared model
  - With linked model
- Family Creation
  - Connectors and Data
  - Formulas
  - Lookup Tables
- Example formulas and lookup table uses

Spaces, Zones & Load Calculations:

- Define Spaces and Zones to match ASHRAE
- Export gbXML
- Import to Trane Trace and run
- Export updated gbXML
- Import into Revit and get to work

Space Airflow

- Place Air Terminals
- Create a Space schedule
- Embed Air Terminal schedule
- Compare Calculated CFM to Actual CFM
- Make adjustments
- Model ductwork

VAV Box Sizing

If it’s built right there’s not much to do here.

- You should have CFM, GPM, MBH, and KW
- CFM and GPM data transmitted upstream to primary equipment
- KVA transmitted to electrical equipment via circuit

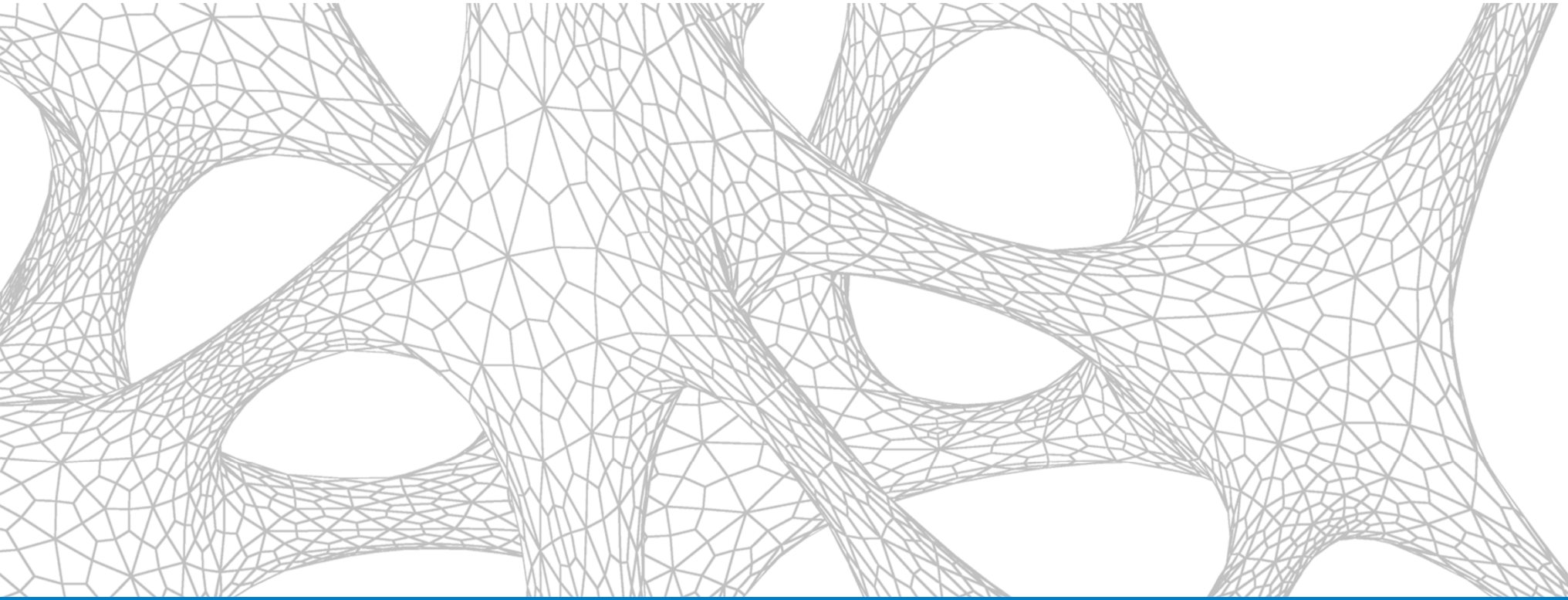
Demo

Space Airflow & VAV Sizing

Equipment Sizing and System

Still need to size primary equipment manually

- Size Air Handling Unit
- Use Duct Sizer on system
- Size Pumps and Boilers
- Use Pipe Sizer on system



Electrical Coordination

Electrical Coordination - Shared

- Families should be reporting correct data...EARLY!
- Verify Voltage and Phase for mechanical equipment
- Circuit equipment
- Verify Max Overcurrent Protection Device
- Make schedules and track changes

Electrical Coordination - Linked

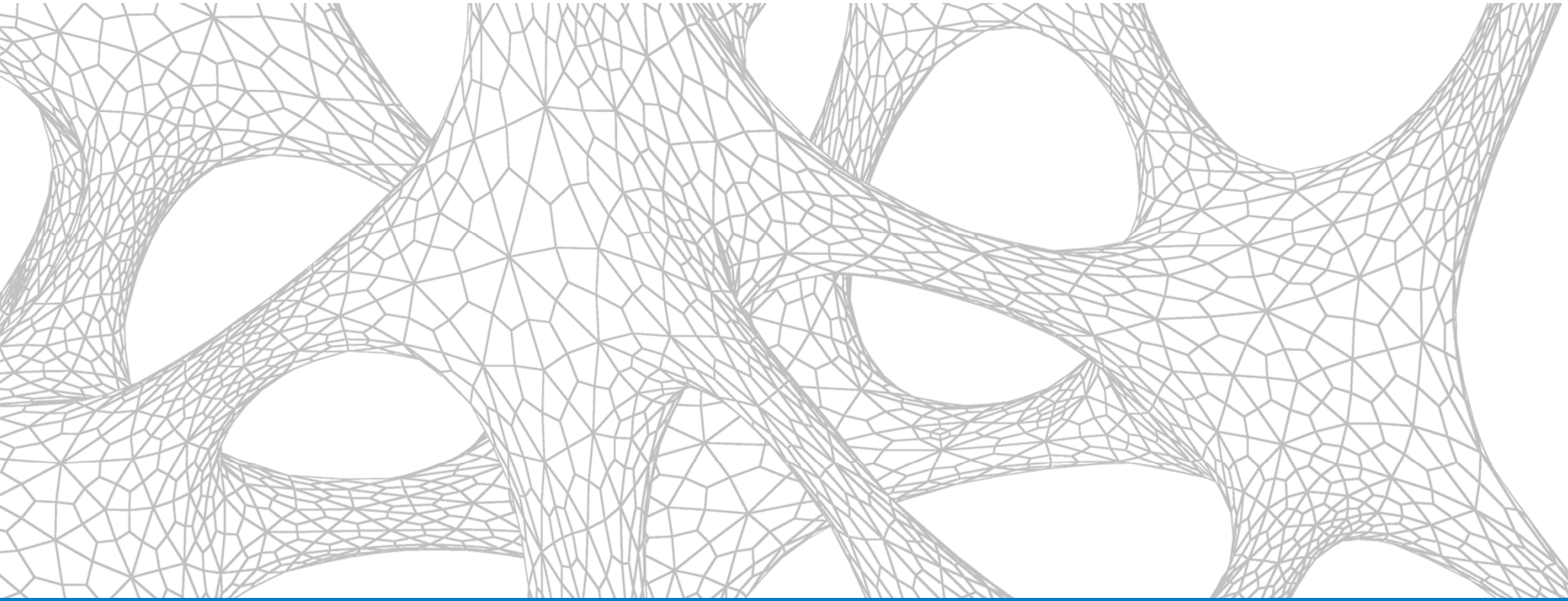
- Place Motor Family and define Mark to match mechanical equipment
- Create coordination schedule
  - Check “Include elements in links”
  - Sort by Mark
  - Match load information from linked elements

Electrical Coordination - Linked

- Verify Voltage and Phase for mechanical equipment
- Circuit equipment
- Verify Max Overcurrent Protection Device
- Track changes when updated links are received

Demo

Motor Coordination



Family Creation

Family Creation

- Geometry is irrelevant as far as data goes
- That’s not to say it isn’t important
- Don’t over-model
- When creating families consider:
  - Where the information comes from?
  - Where it’s going?
  - What it will do when it gets there?
  - How can we use it?

Electrical Connectors

- Associate connector parameters to appropriate parameters
- Two important Electrical settings that can’t be linked to parameters
  - System Type
  - Power Factor State

Mechanical Connectors

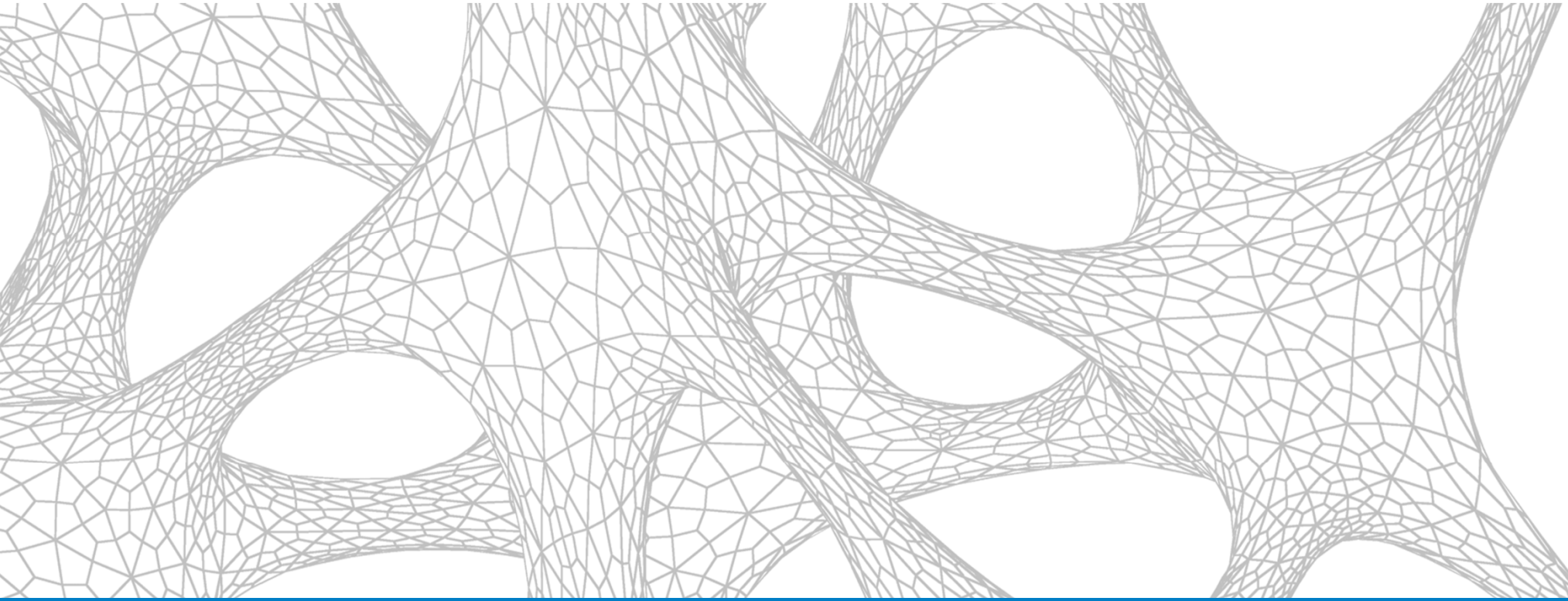
- Associate connector parameters to appropriate parameters
- Several Mechanical settings that can’t be linked to parameters
  - Flow Configuration
  - Flow Direction
  - System Classification
  - Loss Method
  - Shape

Connectors

- Flow Configuration
  - Calculated: Is used when information is being received by this connector from another family. Physical connection.
  - Preset is used to push information to another connector.
  - System: Is used to split a system between two or more pieces of equipment.

Demo

Connector Settings



Formulas!

SIMPLIFY YOUR LIFE WITH FORMULAS AND LOOKUP TABLES

- Use industry standard formulas in your parameters to do tedious calculations
- More accurate than manual computation
  - Do it once, do it right

SIMPLIFY YOUR LIFE WITH FORMULAS AND LOOKUP TABLES

- Use Lookup Tables to return static values
  - Put tables in your families and stop picking up that catalog.

FORMULAS

Formulas can be used for:

- Arithmetic, Algebra, Geometry, Trigonometry
- Conditional statements: IF, AND, OR, NOT, YES / NO
- Round values up, down, or either
- There is no limit to the length or complexity of a formula

FORMULAS

if(Diffuser Slot Qty Input > 0, if(Selection Number Multiplier = 0, 0, if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 1 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 1 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 2 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 2 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 3 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 3 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 4 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 4 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 5 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 5 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 6 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 6 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 7 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 7 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 8 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 8 + Selection Column Count * (Selection Number Multiplier - 1), if((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 9 + Selection Column Count * (Selection Number Multiplier - 1), Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, 9 + Selection Column Count * (Selection Number Multiplier - 1), 88)))))))))), if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 1, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 1, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 1, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 2, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 2, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 2, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 3, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 3, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 3, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 4, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 4, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 4, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 5, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 5, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 5, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 6, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 6, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 6, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 7, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 7, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 7, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 8, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 8, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 8, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 9, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 9, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 9, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 10, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 10, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 10, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 11, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 11, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 11, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 12, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 12, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 12, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 13, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 13, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 13, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 14, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 14, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 14, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 15, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 15, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 15, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 16, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 16, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 16, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 17, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 17, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 17, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 18, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 18, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 41, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 42, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 42, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 42, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 43, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 43, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 43, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 44, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 44, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 44, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 45, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 45, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 45, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 46, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 46, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 46, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 47, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 47, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 47, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 48, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 48, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 48, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 49, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 49, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 49, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 50, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 50, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 50, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 51, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 51, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 51, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 52, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 52, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 52, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 53, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 53, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 53, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 54, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 54, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 54, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 55, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 55, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 55, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 56, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 56, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 56, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 57, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 57, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 57, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 58, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 58, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 58, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 59, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 59, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 59, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 60, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 60, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 60, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 61, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 61, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 61, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 62, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 62, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 62, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 63, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 63, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 63, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 64, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 64, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 64, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 65, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 65, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 65, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 66, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 66, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 66, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 67, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 67, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 67, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 68, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 68, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 68, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 69, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 69, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 69, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 70, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 70, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 70, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 71, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 71, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 71, if(and((size_lookup(Performance Table Selector, "AIRFLOW_PER_LENGTH", 0, 72, Throw Pattern Lookup, Diffuser Slot Width) * 1 CFM + 1 CFM) > Airflow per Length Lookup, (size_lookup(Performance Table Selector, if(Horizontal Throw Pattern, "H_NC", "V_NC"), 0, 72, Throw Pattern Lookup,
Diffuser Slot Width)) < (Design NC + 1)), 72, 99)))

FORMULAS

Tip: write formulas in Notepad++, Notepad, Excel, or Word.

Read the following for great definitions and examples:

- Revit Forum post titled **“Revit Formulas for "everyday" usage “**
- Revit Help – **Conditional Statements in Formulas**

EXAMPLE FORMULAS

Mechanical VAV Box Calculations

- **Heating Coil Capacity (Btu/h)** = Heating Coil Airflow * ([Heating Coil LAT(db)] - [Heating Coil EAT(db)]) * Supply Air Density * Supply Air Sp Ht

EXAMPLE FORMULAS

- **Heating Coil Water Flow (GPM)** = Heating Coil Capacity / ((Heating Coil EWT - Heating Coil LWT) * (Heating Plant Water Density * Heating Plant Water Sp Ht))

EXAMPLE FORMULAS

- **Heating Coil Water Flow (GPM)** = if(or(not(Equipment has Heating Coil), Heating Coil Capacity = 0 Btu/h), 0 GPM, if(((Heating Coil Capacity) / ((Heating Coil EWT - Heating Coil LWT) * (Heating Plant Water Density * Heating Plant Water Sp Ht))) < 0.5 GPM, 0.5 GPM, (roundup(((Heating Coil Capacity) / ((Heating Coil EWT - Heating Coil LWT) * (Heating Plant Water Density * Heating Plant Water Sp Ht))) / 0.1 GPM) * 0.1 GPM)))

EXAMPLE FORMULAS

- **Heating Element Power (W)** = if(not(Equipment has Electric Coil), 0 W, if(Heating Coil Capacity < (size_lookup(Performance Table Name, "HEATING_ELEMENT_MAXIMUM_POWER_KW", 0, Selection Number, Selection Size, Voltage Nominal / 1 V, Phase)) * 1000 W, Heating Coil Capacity, 999999999 W))
- **Message Center (text)** = if(Heating Element Power = 999999999 W, "Calculated KW exceeds heating element max for given voltage and phase. Select higher voltage/phase family type"

EXAMPLE FORMULAS

- **FLA (A)** = Heating Element Power / (Voltage Nominal * if(Phase = 1, 1, sqrt(3)))

- **Apparent Load (VA)** = FLA * Voltage Nominal * if(Phase = 1, 1, sqrt(3))

Motor Family

What are the possibilities:

- NEC Table for Motor Full Load Amps
- Apparent Load calculations from any user input
- Overcurrent Protection Devices – Breakers/Fuses

Motor Family

What are the possibilities:

- NEC Table for Motor Full Load Amps
- Apparent Load calculations from any user input
- Overcurrent Protection Devices – Breakers/Fuses

Demo

Motor Family

EXAMPLE FORMULAS

Electrical Motor Calculations:

- **Motor FLA (A)** = if(Motor Power > 0 hp, size_lookup(Electrical Table Name, "FLA", 0, Voltage Nominal / 1 V, Motor Power NEC Lookup / 1 hp, Phase) * 1 A, 0 A)

- **Motor Power Lookup (hp)** = if(Motor Power < 0.04 hp, 0.03 hp, if(Motor Power < 0.06 hp, 0.05 hp,...))))))

EXAMPLE FORMULAS

- **Apparent Load (VA)** = Motor FLA * Voltage Nominal * if(Phase = 1, 1, sqrt(3))
- **Real Power (W)** = Apparent Load * Power Factor
- **MCA (A)** = if(Motor FLA > 0 A, Motor FLA * Largest Motor Multiplier, 0 A)

EXAMPLE FORMULAS

- **MOCP (A)** = if(**and**(Motor Power > 0 hp, **OCPD is Circuit Breaker**), size_lookup(Electrical Table Name, "BRKSZ", 0, Voltage Nominal / 1 V, Motor Power NEC Lookup / 1 hp, Phase), if(**and**(Motor Power > 0 hp, **OCPD is Fuse**), size_lookup(Electrical Table Name, "FZSZ", 0, Voltage Nominal / 1 V, Motor Power NEC Lookup / 1 hp, Phase), if(MCA < 15 A, 15, if(MCA < 20 A, 20, if(MCA < 25 A, 25, ..., 0))))))* 1 A)

Chart a path for Success

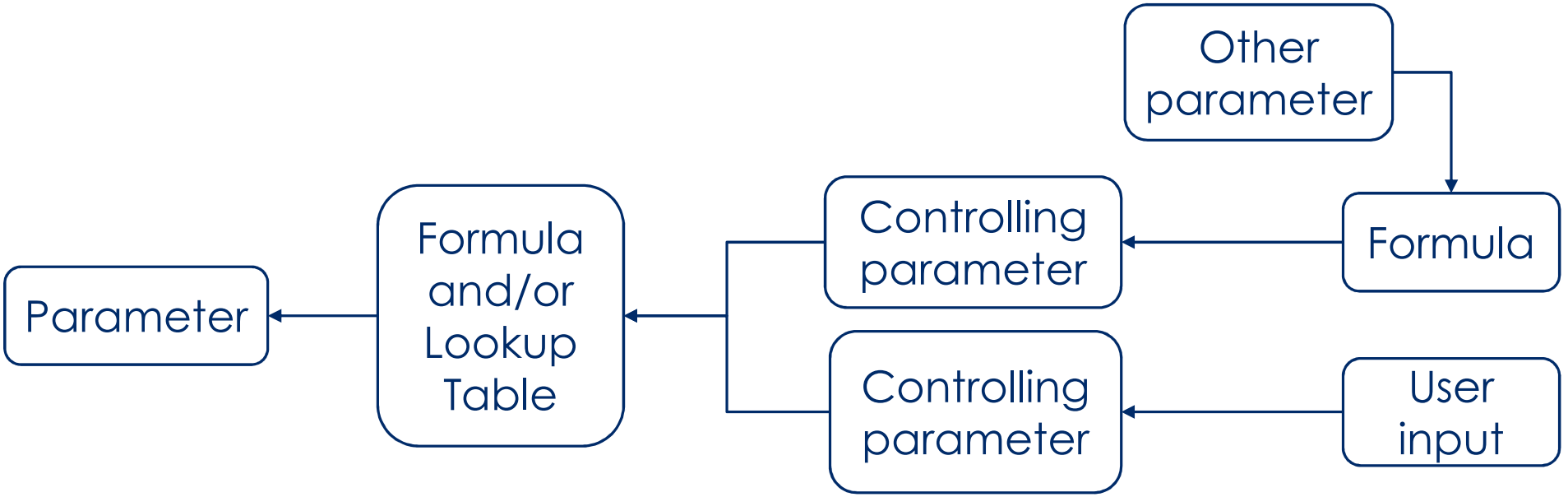
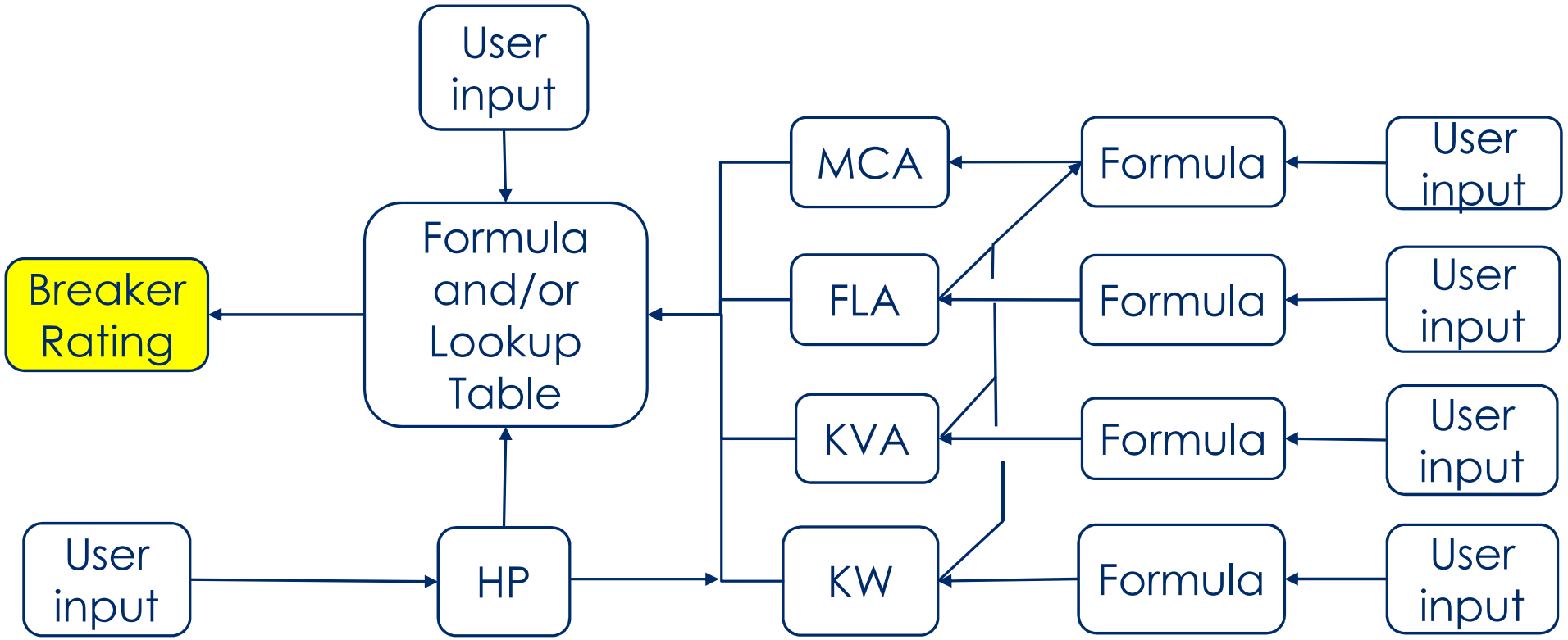
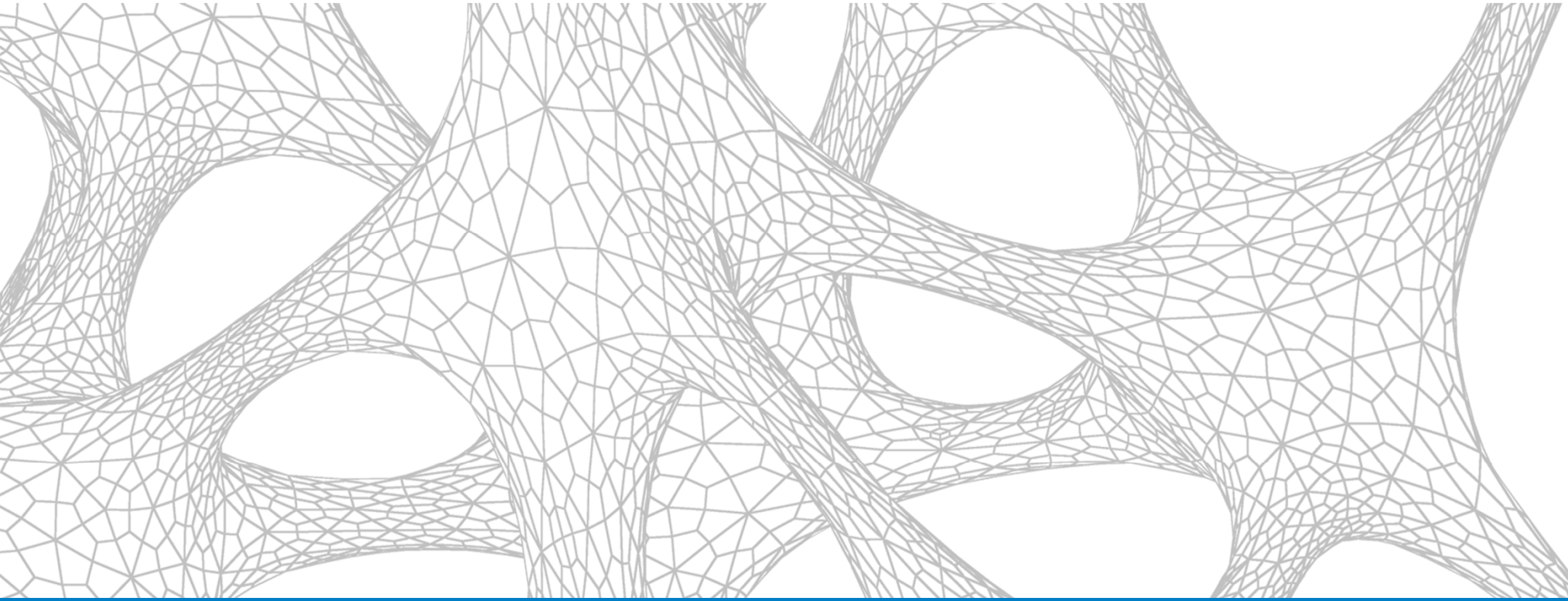


Chart a path for Success





Lookup Tables

LOOKUP TABLES

- Lookup Tables have historically been used to return LENGTH and ANGLE values.
- We can also use them to return NUMBERS
  - This allows us to look up any number and return a unitless value that can be easily converted to another unit
  - We can also remove units from values before we search the table allowing us to use any unit type to return a number that can be converted to any other unit type

LOOKUP TABLES

- We can use them to check or uncheck a YES/NO parameter
- We can also use a YES/NO parameter as a lookup value
- We can even return TEXT values
- We can use more than one Lookup Table in a family.
- Lookup table file type is .CSV, not to be confused with a Type Catalog which is a .TXT file.

TO USE LOOKUP TABLES

Parameter = (size_lookup(Lookup Table Name, “ParameterName", Default, Col B, Col C, (Col D / 1 Unit))) * 1 Unit

- Size_lookup = the function or command
- Lookup Table Name = the name of your lookup table as it appears in the “Manage Lookup Tables” dialog
- ColumnName = the name of the header in your lookup table dropping the ##other##number
- Default = the value returned if no corresponding value is found in the table. Also, note its position before the conditions, opposite what we use in a standard conditional statement.

TO USE LOOKUP TABLES

- The column callouts progress from left to right starting at Column B, and is not limited
- Columns do not need to reference another parameter
- To use a unit-ed parameter you must convert it to unitless by dividing by 1 (the unit will automatically be populated)
- To change the returned value to the appropriate unit, enclose the formula in parentheses and multiply by 1.

NOVEL USES FOR LOOKUP TABLES

- Toggle a Yes/No parameter from a Lookup Table

YESNO =

size_lookup(Lookup Table Name, "YESNO", 0, Col B, Col C) = 1

- This can be used to control a yes/no parameter given certain options
- As long as the statement is TRUE, the box will check

NOVEL USES FOR LOOKUP TABLES

- Return a text value from a Lookup Table

Text parameter =

size_lookup(Lookup Table Name, “”, “default”, Col B, Col C)

- This can be used to return the text value in column A

MULTIPLE MOTOR CALCULATIONS

- Calculated FLA = (Motor FLA * Qty) + (Secondary Motor FLA * Qty) + Accessory FLA

- Calculated MCA = Largest Motor Multiplier *(Motor FLA) + (Motor FLA * (Qty -1)) + (Secondary Motor FLA * Qty) + Accessory FLA

Demo

Multiple Motor Family

Questions?

AUTODESK®

Make anything.