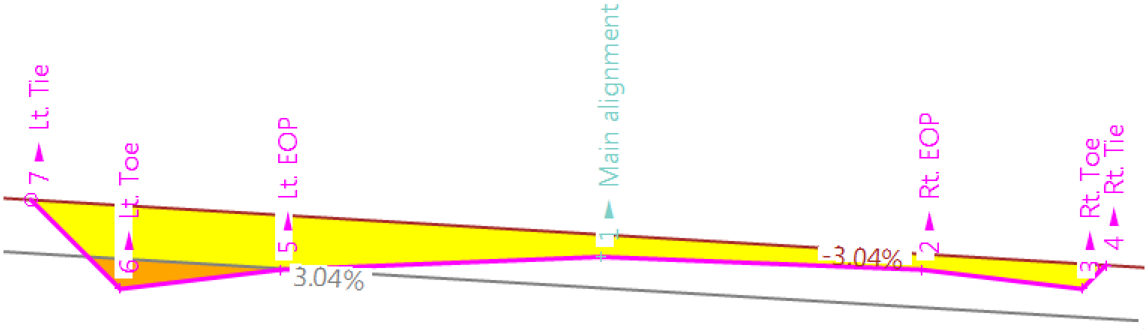
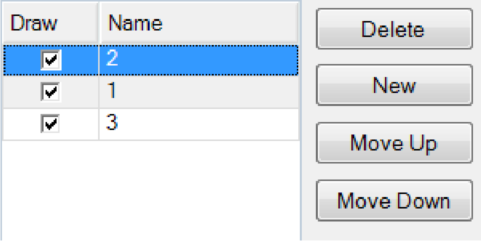
Using Wildcards/Regular Expressions in Labels for Cross-section Sheets

Wildcards (special characters) allow you to modify the results of an alphanumeric character string or other search/match. Regular expressions (regex) are a powerful mechanism which enable you to find character strings based on a specified pattern (in this case, corridor template node names). You can use wildcards and regular expressions in the Sheet Set Editor to specify which node and segment labels and node lines appear on cross-section sheets. This document describes these features. For general information about using regular expressions in the program, search the help for “regular expression”.



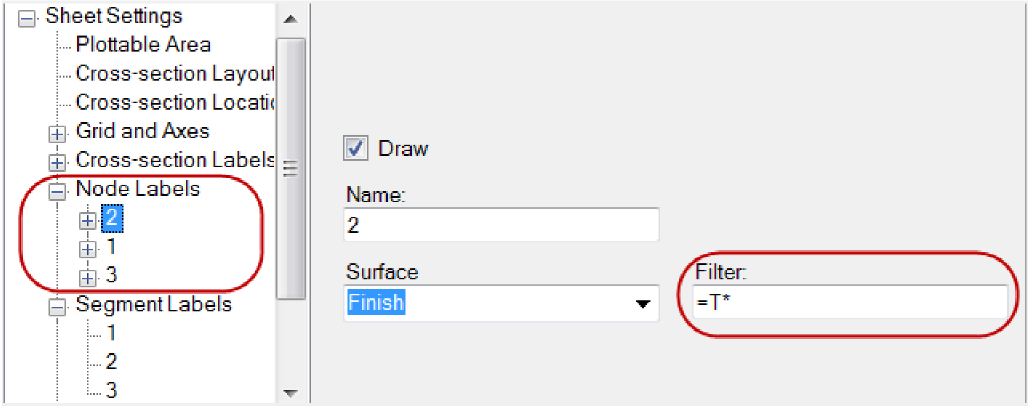
When determining which labels to display, the program uses the filter for each node or segment label in the order they are listed in the Sheet Set Editor. If there are no matches for the filter in the first row, the filter in the second row is applied. Filters in subsequent rows are tested for a match until one or more strings fit the filter syntax. Once a match is found and labels are drawn, subsequent rows are not searched.



Draw	Name
☑	2
☑	1
☑	3

The program matches strings based on the specified filter. Characters following an equals sign (=) are treated as a regular expression. If there is no equals sign, the string is parsed into its combination of strings and wildcards.

To use regular expressions and wildcards to control node and segment labels, enter any of the characters below in the Filter box for node labels or segment labels:

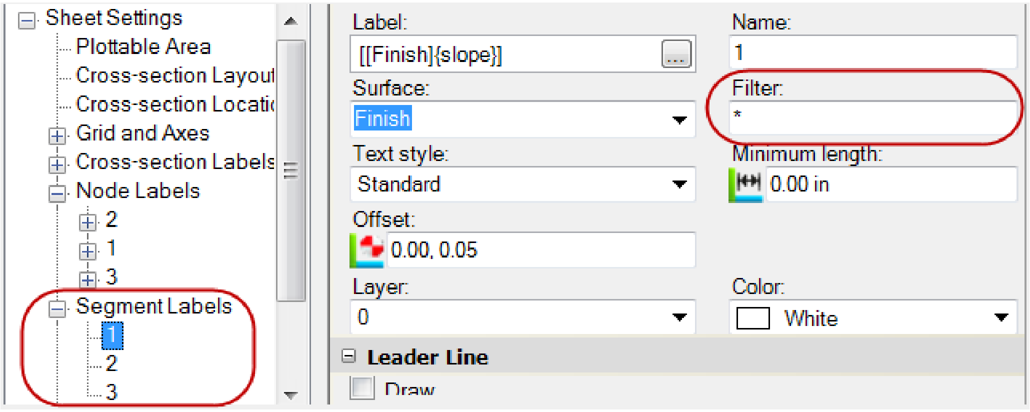


☑ Draw

Name: 2

Surface: Finish

Filter: =T*



Label: [[Finish]{slope}]

Name: 1

Surface: Finish

Filter: *

Text style: Standard

Minimum length: 0.00 in

Offset: 0.00, 0.05

Layer: 0

Color: White

Here are wildcards you can use:

Wildcard	Effect (if used in the Filter box)	Examples (based on the diagram above)
= (regex)	If placed at the beginning of a string, characters afterward are treated as a regular expression.	• '=Lt. Tie' matches regex 'Lt. Tie' • 'Lt. Tie,=Rt. Tie' matches 'Lt. Tie' and regex 'Rt. Tie' • '=^L|^.......$' matches 'Lt. Tie', 'Lt. Toe', and 'Lt. EOP' for the first regex, and the same plus 'Rt. Tie', 'Rt. Toe', and 'Rt. EOP' in the diagram above for the second expression
* (all)	If used alone, all strings are considered a match. If placed anywhere in a string, any match with the rest of the expression that the wildcard precedes and/or follows will be found.	• '*' matches all strings • 'L*' matches 'Lt. Tie', 'Lt. Toe', and 'Lt. EOP' • '* Tie' matches 'Lt. Tie' and 'Rt. Tie'
, (or)	If placed between expressions, any valid either/or matches will be found.	• 'Lt.*,Rt.*' matches 'Lt. Tie', 'Lt. Toe', 'Lt. EOP', 'Rt. Tie', 'Rt. Toe', and 'Rt. EOP' • '*Alignment,*alignment' matches 'Main Alignment'
. (any)	If placed anywhere in an expression, matches any single character in the same position.	• '*E.P' matches 'Lt. EOP' and 'Rt. EOP'

Note: Characters used in regular expressions are case-sensitive.
Note: To see the results of a filter, remember to click the Scan button below the settings explorer.