

Set_Pipeline_Parameter_From_Expression /PARAMETER_NAME=t_Mass /EXPRESSION=METRIC::PROCESSED::HEIGHT * 4.6 /AS_INTEGER=TRUE ; The pipeline parameter contains the value of the expression.

Getting the syntax correct can be challenging because there can be a conflict between the & as a concatenation parameter in a pipeline parameter and the & as a boolean “and” operator.

For many pipeline commands Visual3D recognizes the conflict and handles the situations in a context specific way.

Visual3D cannot make that distinction and “bad” things happen.

Consider the following which uses string comparisons and a pipeline parameter in the expression:

Set_Pipeline_Parameter_From_Expression /PARAMETER_NAME= MVAL /EXPRESSION=(&::TAG_NAMES& == Utility) * 4.6 /AS_INTEGER=FALSE ; The goal was to set MVAL to 4.6 when the pipeline parameter TAG_NAMES was equal to the string “Utility”;

One solution is to separate the pipeline parameter from the expression.

Set_Pipeline_Parameter /PARAMETER_NAME=TEMP /PARAMETER_VALUE=(&::TAG_NAMES& == Utility)*4.6 ;

Set_Pipeline_Parameter_From_Expression /PARAMETER_NAME=t_Mass /EXPRESSION=::TEMP /AS_INTEGER=FALSE ;

Another solution, which is even trickier in terms of syntax is:

Set_Pipeline_Parameter_From_Expression /PARAMETER_NAME=SCOTT /EXPRESSION=(“&::TAG_NAMES&”==“ Utility”) ! /AS_INTEGER=TRUE ; This forces Visual3D to recognize the strings as strings and the & is treated correctly.

Get global metrics

Global metrics can be set to a pipeline parameter by setting “GLOBAL” in front of the metric name. More information about expressions can be found [here](#).

Set_Pipeline_Parameter_From_Expression /PARAMETER_NAME=KATIE /EXPRESSION=GLOBAL::METRIC::PROCESSED::MASS /AS_INTEGER=FALSE ;

From:
<https://has-motion.com/wiki/> - Wiki Documentation Site

Permanent link:
https://has-motion.com/wiki/doku.php?id=visual3d:documentation:pipeline:pipeline_commands:set_pipeline_parameter_from_expression&rev=1718805770

Last update: 2024/06/19 14:02

