Command Reference: SetTimeSeriesValuesFromLookupTable()

Set time series values by using an input time series and a lookup table

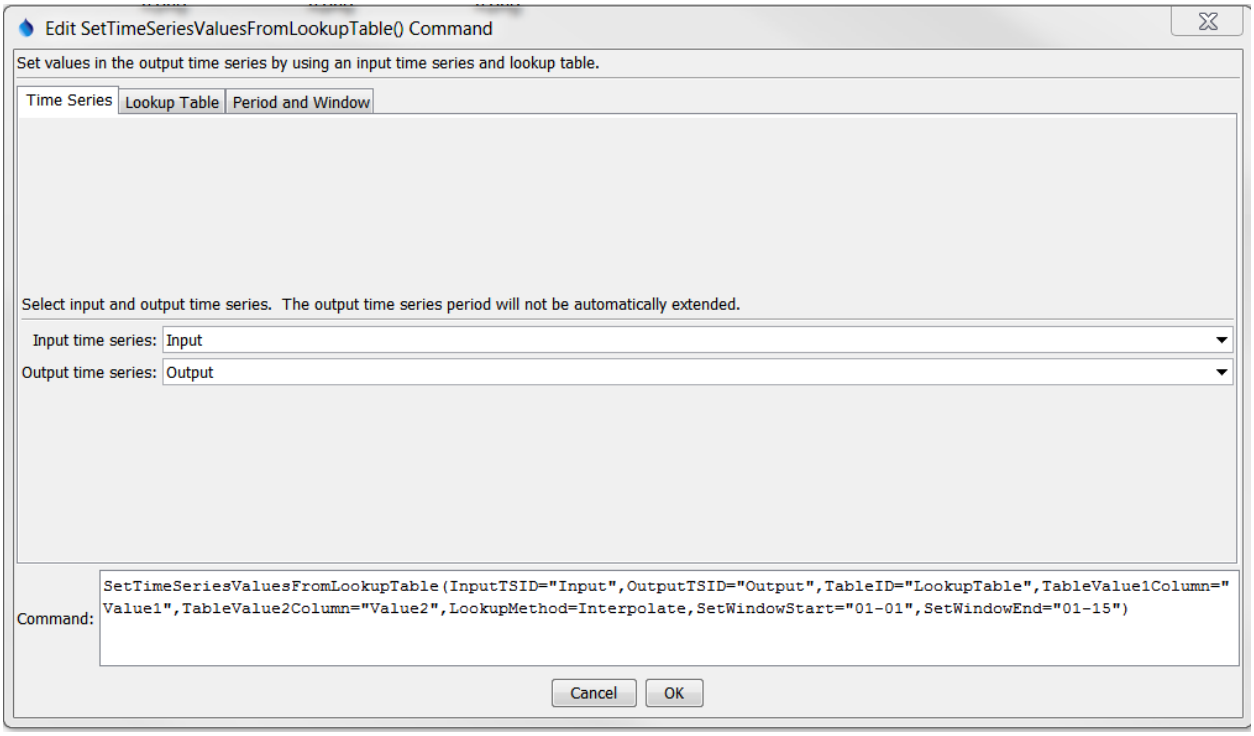
Version 11.08.00, 2016-02-03

The `SetTimeSeriesValuesFromLookupTable()` command uses an input time series and lookup table to set values in the output time series. Examples of using this command include:

- Converting reservoir elevation to storage, surface area, seepage, or other values
- Converting river stage to discharge
- Converting a time series to category values
- Lookup up values from a distribution

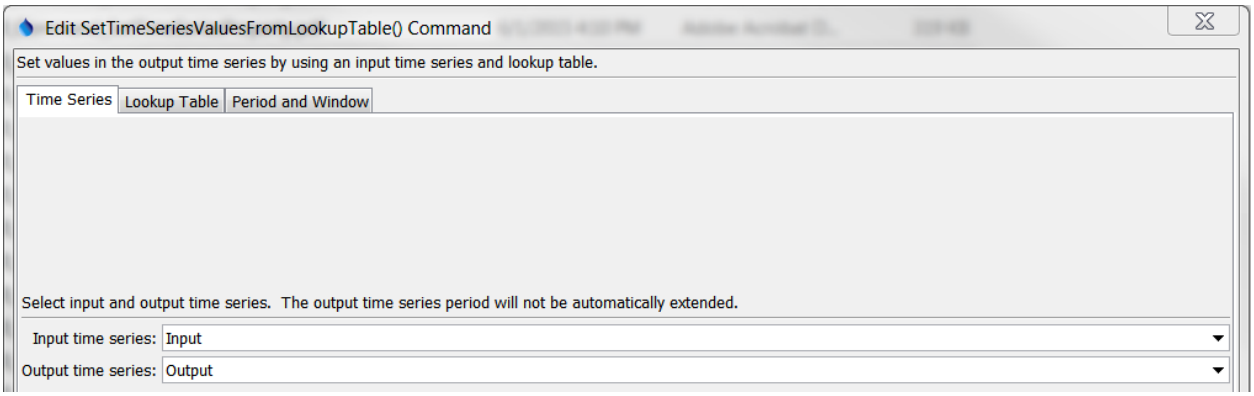
In many cases the lookup table will apply throughout the analysis period. The values in the table should be sorted in ascending order prior to lookup. This command currently does not handle rating table shifts; however, this capability may be added in the future. Missing (null) and `NaN` values in the lookup table are removed before processing so that lookups are performed only on rows with input and output values.

The following dialog is used to edit the command and illustrates the syntax of the command:



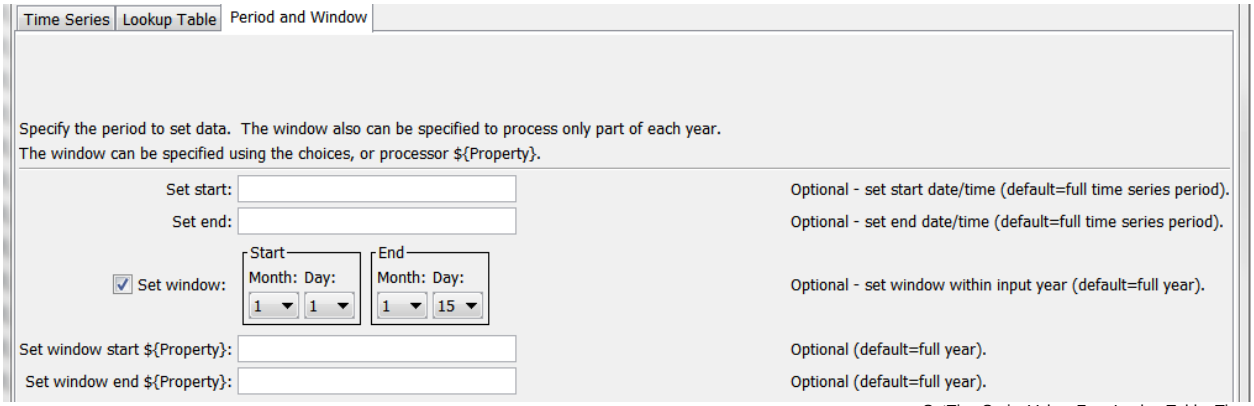
SetTimeSeriesValuesFromLookupTable

SetTimeSeriesValuesFromLookupTable() Command Editor for Time Series Parameters



SetTimeSeriesValuesFromLookupTable_Lookup

SetTimeSeriesValuesFromLookupTable() Command Editor for Lookup Table Parameters



SetTimeSeriesValuesFromLookupTable_Time

SetTimeSeriesValuesFromLookupTable() Command Editor for Period and Window Parameters

The command syntax is as follows:

```
SetTimeSeriesValuesFromLookupTable(Parameter=Value,…)
```

Command Parameters

Parameter	Description	Default
`InputTSID`	The time series identifier or alias for the time series used as input. Can be specified with `${Property}` notation.	None – must be specified.
`OutputTSID`	The time series identifier for the time series being modified. Use the ***Edit*** button to edit the time series identifier parts. Can be specified with `${Property}` notation.	None – must be specified.
`TableID`	The lookup table identifier. Can be specified with `${Property}` notation.	None – must be specified.
`Table TSIDColumn`	Table column name that is used to match the time series identifier for processing. **This parameter currently is not supported but will be enabled in the future.**	If not specified, it is assumed that the entire lookup table applies.
`Table TSIDFormat`	The specification to format the time series identifier to match the `TableTSIDColumn` column. **This parameter**	Time series alias if available, or

Parameter	Description	Default
	currently is not supported but will be enabled in the future.	otherwise the time series identifier.
`Table Value1Column`	Table column name for data values that correspond to the input time series (`InputTSID`). Can be specified with `${Property}` notation.	None – must be specified.
`SortInput`	Whether to sort the lookup table. The order is checked to ensure the data are sorted but forcing the sort when not needed is a performance hit.	Rely on table being sorted.
`Table Value2Column`	Table column name for data values that correspond to the output time series identifier (`OutputTSID`). Can be specified with `${Property}` notation.	None – must be specified.
`Effective DateColumn`	Table column name for the effective date. **This parameter currently is not supported but will be enabled in the future.**	The lookup data apply to the entire period.
`LookupMethod`	Indicate how to select the value to use for output: • `Interpolate` – interpolate between points if input values do not exactly align with table values; if `Transformation=Log`, then interpolation will use the transformed values • `PreviousValue` – pick the previous (smaller) value in the table (exact matches use the lookup table value) • `NextValue` – pick the next (largest) value in the table (exact matches use the lookup table value)	`Interpolate`
`OutOfRange LookupMethod`	Indicate the value to use when estimating values that are outside the range of the rating table: • `Extrapolate` – use the two known values at the end of the table to extrapolate; if `Transformation=Log`, then extrapolation will use the transformed values • `SetMissing` – set output to missing • `UseEndValue` – use the data value on the end	`SetMissing`
`OutOfRange Notification`	Indicate the notification to generate when a value is outside the range of the lookup table: • `Ignore` – do not generate warning or failure message • `Warn` – generate a warning message • `Fail` – generate a failure message	`Ignore`
`Transformation`	Indicates how to transform the data before interpolation, used when `LookupMethod=Interpolate` and `OutOfRangeMethod=Extrapolate`). Specify as `None` to compare raw values or `Log` (for $\log_{10}$) to transform values before interpolation and extrapolation. If the `Log` option is used, zero and negative values are replaced with the value specified by the `LEZeroLogValue` parameter value for analysis (missing data values are ignored in the analysis).	`None` (no transformation).

Parameter	Description	Default
LEZero LogValue	Value to use for data values less than or equal to zero when using a log transformation.	.0010
SetStart	The date/time to start setting values. Can be specified with ${Property} notation.	Set the full period.
SetEnd	The date/time to end setting values. Can be specified with ${Property} notation.	Set the full period.
SetWindowStart	The calendar date/time for the set start within each year. Specify using the format MM, MM-DD, MM-DD hh, or MM-DD hh:mm, consistent with the time series interval precision. A year of 2000 will be used internally to parse the date/time. Use this parameter to limit data processing within the year, for example to output only a single month or a season. A processor ${Property} can be specified using the text field under the window date editor.	Lookup values for the full year.
SetWindowEnd	Specify date/time for the output end within each year. See SetWindowStart for details. A processor ${Property} can be specified using the text field under the window date editor.	Lookup values for the full year.