Command Reference: TS X = changeInterval()

DRAFT – THIS COMMAND IS UNDER DEVELOPMENT

Create a New Time Series by Changing a Time Series Data Interval

Version 06.10.00, 2005-02-18, Color, Acrobat Distiller

A `changeInterval()` command can be inserted to change the data interval of an existing time series, resulting in a new time series. The majority of the original header data (e.g., description, units) are maintained in the new time series. Time series data values have a time scale, which is accumulated (e.g., volume), mean (e.g., flow), or instantaneous (e.g., temperature). Changing the interval can also result in a change in the time scale (e.g., converting instantaneous values to a mean value). Currently, the time scale is NOT determined from the data type and interval and must be specified as ACCM, MEAN, or INST. For regular time series, the data intervals must be align so that each larger interval aligns with the end-points of the corresponding smaller intervals (e.g., the ends of 6-hour intervals align with the daily interval). IS THIS BEING CHECKED IN THE CODE? The following conversions are currently supported, with a description of the conversion process: CODE NEEDS TO THROW EXCEPTIONS IF A CONVERSION IS NOT SUPPORTED. The period of the time series will include any intervals where there were one or more values in the original data??? In TSTool an overall output period can be set (setOutputPeriod() – we may need to deal with this later.

Small Interval ACCM (Accumulation) to Large Interval ACCM (Accumulation)

Changing the interval for small interval accumulated data to large interval accumulated data involves summing the small interval data values for the period that overlaps the large interval.

Accumulated data have a timestamp corresponding to the interval-end for the accumulation. Conversions involving time intervals that have zero values (e.g., Hour 0, Minute 0) result in a perceived shift in time because the zero occurs on the boundary between larger intervals. The following examples illustrate the accumulation for common cases. In cases where an accumulation jumps over two or more interval categories (e.g., minute to day), the accumulation occurs as if the two intermediate accumulations occurred in succession. In the following examples, the general representation is shown first, followed by an example where appropriate.

NHour to Day (6Hour to Day example, *i* equals the hour multiplier):

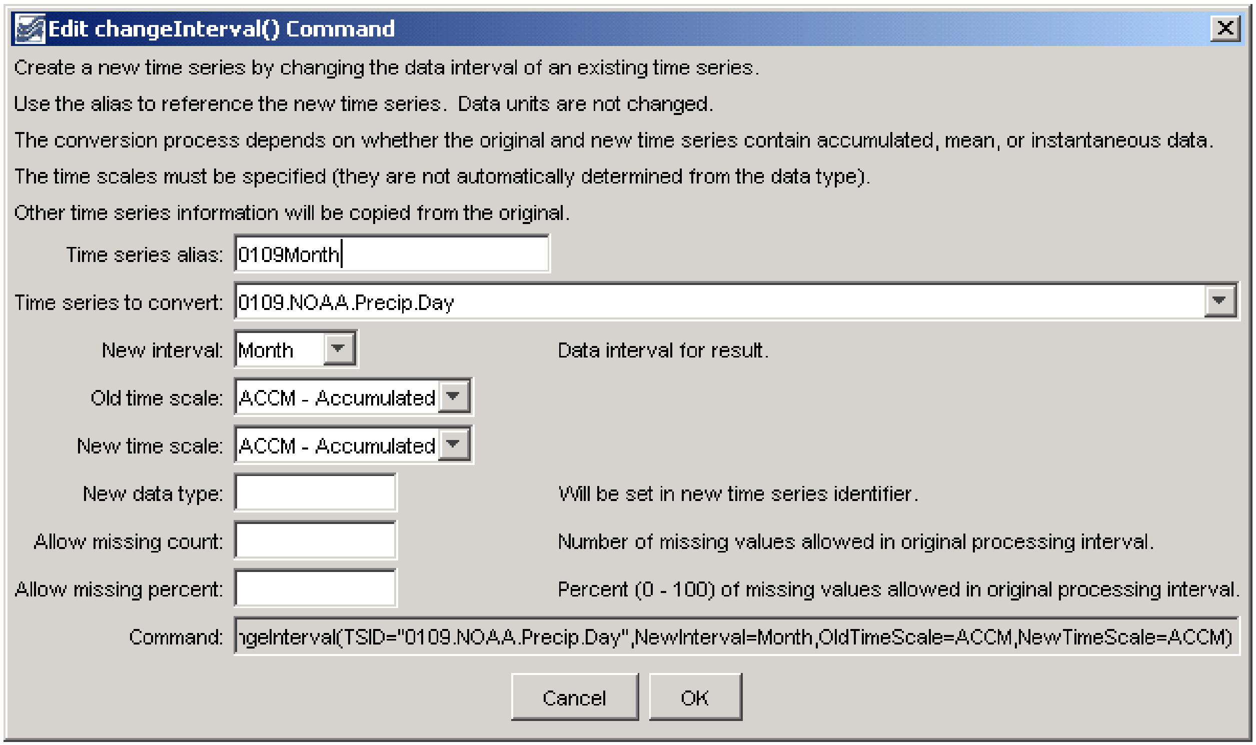
Day 1, Hour *0*	Day 1, Hour *i*	Day 1, Hour *2i*	Day 1, Hour *3i*	Day 2, Hour *0*
Day 1, Hour 0	Day 1, Hour 6	Day 1, Hour 12	Day 1, Hour 18	Day 2, Hour 0
	Day 1 accumulation			

NDay to Month (example for a month with 30 days):

Month 1, Day 1	…		…	Month 1, Day 30
Month 1 accumulation				

Need to insert examples of ALL supported conversions so users have NO QUESTIONS about what is done.

The following dialog is used to edit the command and illustrates the syntax for the command.



Edit changeInterval() Command

Create a new time series by changing the data interval of an existing time series.
Use the alias to reference the new time series. Data units are not changed.
The conversion process depends on whether the original and new time series contain accumulated, mean, or instantaneous data.
The time scales must be specified (they are not automatically determined from the data type).
Other time series information will be copied from the original.

Time series alias:	0109Month	
Time series to convert:	0109.NOAA.Precip.Day	
New interval:	Month	Data interval for result.
Old time scale:	ACCM - Accumulated	
New time scale:	ACCM - Accumulated	
New data type:		Will be set in new time series identifier.
Allow missing count:		Number of missing values allowed in original processing interval.
Allow missing percent:		Percent (0 - 100) of missing values allowed in original processing interval.
Command:	ngeInterval(TSID="0109.NOAA.Precip.Day",NewInterval=Month,OldTimeScale=ACCM,NewTimeScale=ACCM)	

changeInterval

changeInterval() Command Editor

The command syntax is as follows:

```
TS X = changeInterval(param=value,…)
```

Command Parameters

Parameter	Description	Default
`Alias`	The alias to assign to the new time series. The time series identifier for the new time series will be the same as the original time series, with the new interval and optionally a new data type (see the `NewDataType` parameter).	None – must be specified.
`TSID`	The time series identifier or alias for the original (old) time series.	None – must be specified.
`NewInterval`	The data interval for the new time series, from the provided choices. For example: `6Hour`, `Day`, `Month`, `Year`.	None – must be specified.
`OldTimeScale`	The time scale for the original time series, one of: `ACCM` – accumulated data `INST` – instantaneous data `MEAN` – mean data In the future, this parameter may be made optional if the time scale can be determined from the data type.	None – must be specified.
`NewTimeScale`	The time scale for the new time series (see `OldTimeScale` for possible values). In the future, this parameter may be made optional if the time scale can be determined from the data type.	None – must be specified.
`NewDataType`	The data type for the new time series. This will be set in the identifier of the new time series.	Use the data type from the original time series.
`AllowMissingCount`	The number of missing values allowed in the source interval(s) in order to produce a result. For example, if converting daily data to monthly, a value of `5` would allow <= `5` missing daily values and still compute the result. This capability should be used with care because it may result in data that are not representative of actual conditions.	`0` – do not allow any missing data in the source data when computing a result.

A sample commands file is as follows (convert a daily precipitation time series to monthly total):

```
0109.NOAA.Precip.Day~HydroBase
TS 0109Month =
changeInterval(TSID="0109.NOAA.Precip.Day",NewInterval=Month,OldTimeScale=ACCM,NewTimeScale=ACCM)
```

This page is intentionally blank.