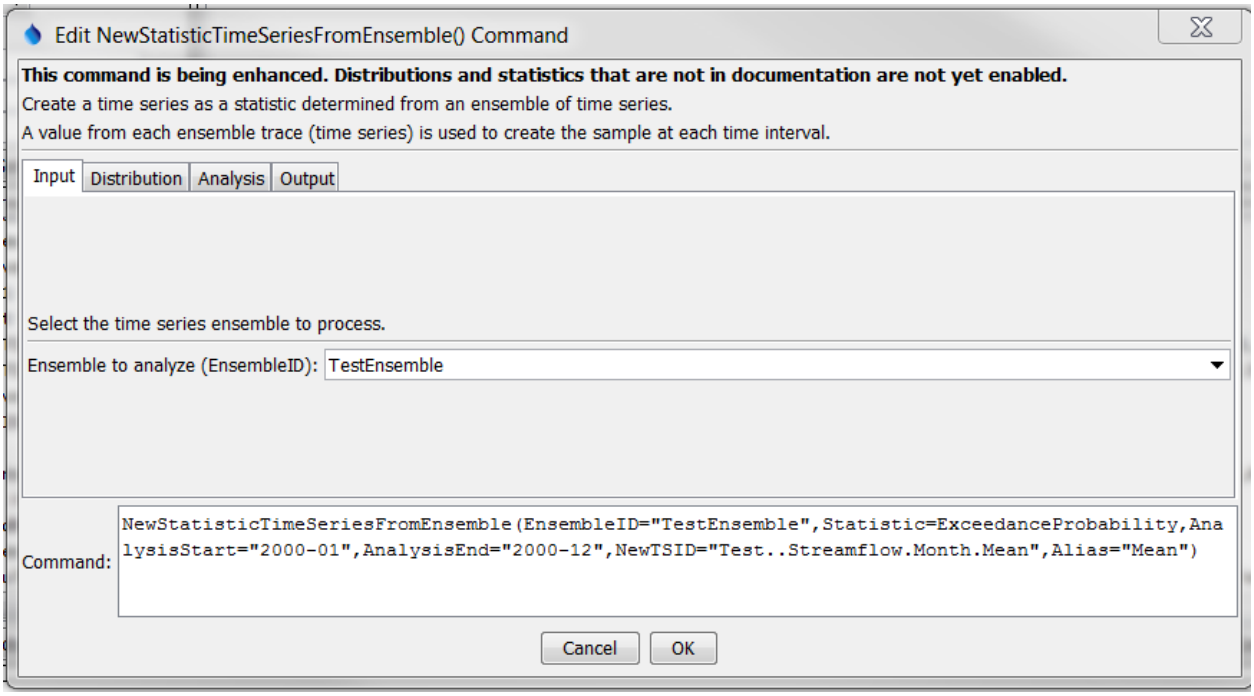
Command Reference: NewStatisticTimeSeriesFromEnsemble()

Create a time series containing a statistic determined from a time series ensemble

Version 11.08.01, 2016-02-14

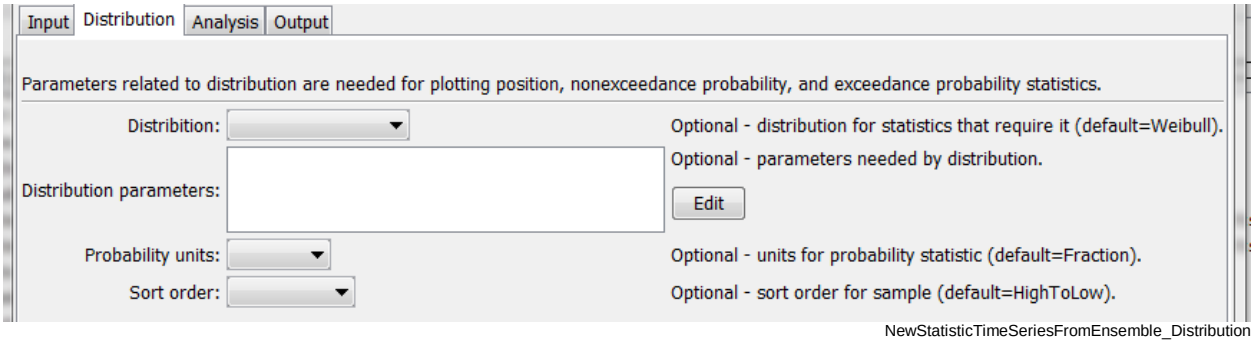
Yellow highlights in this document indicate work in progress. The `NewStatisticTimeSeriesFromEnsemble()` command uses data from time series in an ensemble to calculate a statistic for each interval in the ensemble, and assigns the statistic value to the corresponding interval in the result. For example, for a statistic of `Mean` applied to a daily time series, all January 1, 1970 values will be used for the sample and the mean value will be assigned to January 1, 1970 in the output time series. Leap year values will be included if they are included in the period of the ensemble.

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



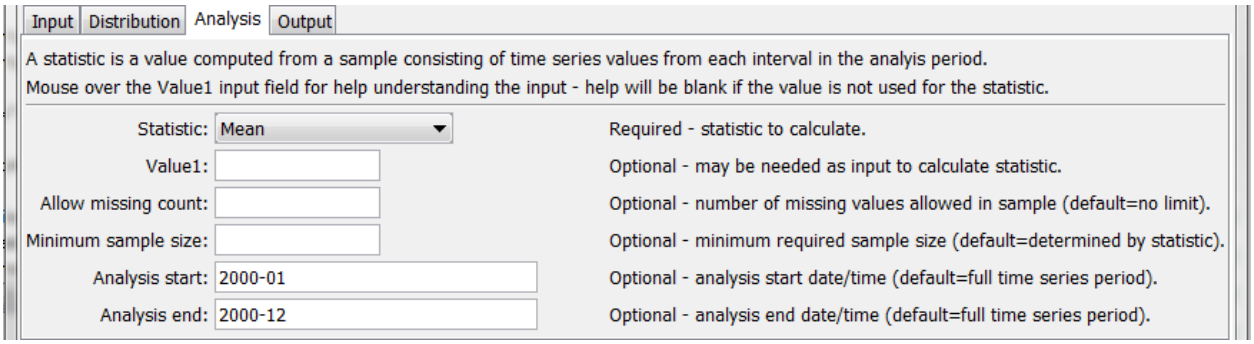
NewStatisticTimeSeriesFromEnsemble

NewStatisticTimeSeriesFromEnsemble() Command Editor showing Input Parameters



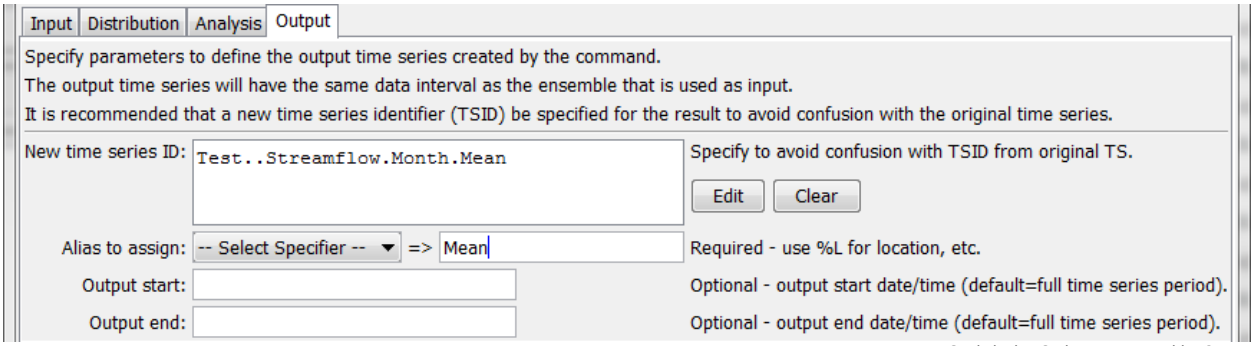
NewStatisticTimeSeriesFromEnsemble_Distribution

NewStatisticTimeSeriesFromEnsemble() Command Editor showing Distribution Parameters



NewStatisticTimeSeriesFromEnsemble_Analysis

NewStatisticTimeSeriesFromEnsemble() Command Editor showing Analysis Parameters



NewStatisticTimeSeriesFromEnsemble_Output

NewStatisticTimeSeriesFromEnsemble() Command Editor showing Output Parameters

The command syntax is as follows:

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

The following older command syntax is updated to the above syntax when a command file is read:

```
TS Alias = NewStatisticTimeSeriesFromEnsemble(Parameter=Value,…)
```

Command Parameters

Parameter	Description	Default
`EnsembleID`	The identifier for the ensemble to analyze. Can be specified using `${Property}` notation.	None – must be specified.
`Distribution`	Indicates the distribution, needed for certain statistics (see **Statistics Summary** table below for indication or statistics that need distribution information). See the **Distribution Summary** table below for information about distributions.	
`Distribution Parameters`	Additional parameters needed to specify a distribution. See the **Distribution Summary** table below.	
`Probability Units`	Units to use for calculated probability statistics: • `Fraction` • `Percent` or `%`	`Fraction` (0 – 1).

SortOrder	Order to sort the sample, used with exceedance probability, plotting position and rank: • LowToHigh – rank 1 in plotting position is smallest value • HighToLow – rank 1 in plotting position is largest value	HighToLow for Exceedance Probability, Nonexceedance Probability, and PlottingPosition.
Statistic	The statistic to compute. See the **Available Statistics** table below.	None – must be specified.
Value1	Input data required by the statistic. Currently the dialog does not check the value for correctness – it is checked when the statistic is computed.	See **Statistic Details** table below.
Allow Missing Count	The number of missing values allowed in the sample of values in order to produce a result. This capability should be used with care because it may result in data that are not representative of actual conditions.	Missing values are ignored in the sample used to compute the statistic.
MinimumSample Size	The minimum number of values in the sample that are required to compute the statistic.	Use the sample with no restrictions, although some statistics may have requirements.
AnalysisStart	The date/time for the analysis start, using a precision that matches the original time series. Can be specified using ${Property} notation.	Analyze the full period.
AnalysisEnd	The date/time for the analysis start, using a precision that matches the original time series. Can be specified using ${Property} notation.	Analyze the full period.
NewTSID	The time series identifier to be assigned to the new time series, which is useful to avoid confusion with the original time series. **This parameter may be required in the future.** Can be specified using ${Property} notation.	None – use the same identifier as the original time series.
Alias	The alias to assign to the time series, as a literal string or using the special formatting characters listed by the command editor. The alias is a short identifier used by other commands to locate time series for processing, as an alternative to the time series identifier (TSID).	None – must be specified.
OutputStart	The date/time for the output start, using a precision that matches the original time series. An output period longer than the analysis period will result in missing values in output. Can be specified using ${Property} notation.	Output the full period.
OutputEnd	The date/time for the output start, using a precision that matches the original time series. An output period longer than the analysis period will result in missing values in output. Can be specified using ${Property} notation.	Output the full period.

Available Statistics

Statistic	Description	Limitations
`ExceedanceProbability`	The data value corresponding to an N% chance of value being exceeded. Specify the probability as a fraction using `Value1`.	Small sample size will impact – see statistic details.
`ExceedanceProbability10`	The data value corresponding to a 10% chance of value being exceeded.	Small sample size will impact – see statistic details.
`ExceedanceProbability30`	The data value corresponding to a 30% chance of value being exceeded.	Small sample size will impact – see statistic details.
`ExceedanceProbability50`	The data value corresponding to a 50% chance of value being exceeded.	Small sample size will impact – see statistic details.
`ExceedanceProbability70`	The data value corresponding to a 70% chance of value being exceeded.	Small sample size will impact – see statistic details.
`ExceedanceProbability90`	The data value corresponding to a 90% chance of value being exceeded.	Small sample size will impact – see statistic details.
`GECount`	Count of values greater than or equal to `Value1`.	None.
`GTCount`	Count of values greater than `Value1`.	None.
`GeometricMean`	Geometric mean of all values in the sample.	All values must be >= 0.
`LECount`	Count of values less than or equal to `Value1`.	None.
`LTCount`	Count of values less than `Value1`.	None.
`Max`	Maximum of all values in the sample.	None.
`Mean`	Arithmetic mean of all values in the sample.	None.
`Median`	Median of all values in the sample.	None.
`Min`	Minimum of all values in the sample.	None.
`MissingCount`	The count of values that are missing.	This statistic will be computed regardless of `AllowMissingCount` and `MinimumSampleSize`.
`MissingPercent`	The percent of values that are missing.	See above.
`NonmissingCount`	The count of values that are not missing.	See above.
`NonmissingPercent`	The percent of values that are not missing.	See above.
`Rank`	Rank order, based on `SortOrder` command parameter. Duplicate values are each assigned a rank that is the average of the ranks for the duplicate values. This is necessary because selecting one of the ranks would be arbitrary. A new command parameter may be added to allow control of this behavior.	None.
`Skew`	Skew coefficient, as follows: $Cs = \frac{N \sum_{i=1}^{N}(Y_i - Y_{mean})^3}{(n-1)(n-2)s^3}$	None.

Statistic	Description	Limitations
	where *s* = standard deviation.	
StdDev	Sample standard deviation.	None.
Total	Total of values in the sample.	None.

Statistic Details

Statistic	Description
Exceedance Probability*	The statistic for each time step in the analysis period is computed as follows: 1. The data values are extracted for each trace with missing values being ignored. The sample size is *n*. 2. The data values are sorted into ascending order. 3. Exceedance probabilities are computed for the number of sample values according to distribution (Weibull by default) plotting positions as follows (for $i=1,\ldots,n$): a. If $n = 1$, the exceedance probability $P_i=1.0$. This is an extreme case due to small sample size. b. Otherwise, $P_i=(n-(i-1))/(n+1)$. Therefore, when $i=1$, $P_i=n/(n+1)$ and when $i=n$, $P_i=1/(n+1)$. The probabilities will be listed from high to low value (the opposite order of the sorted data values). 4. The data value corresponding to the requested probability is calculated by iterating over the probabilities until the calculated probability for a value is less than the requested probability: a. If the first probability satisfies the condition, the computed value is set to the minimum value in the sample (no extrapolating past the end). b. Otherwise, the value is interpolated from the previous and current sample values. If no calculated probability is less than the requested probability, the computed value is set to the maximum value in the sample (no extrapolating past the end). To create an exceedance probability plot, use several commands with different exceedance probability levels (listed low to high). Graphing the time series in a bar graph with BarOverlap=True will draw the bars on top of each other to give the desired appearance. The edges of the colors will represent the specific exceedance probabilities and the colored areas will represent ranges of exceedance probabilities.

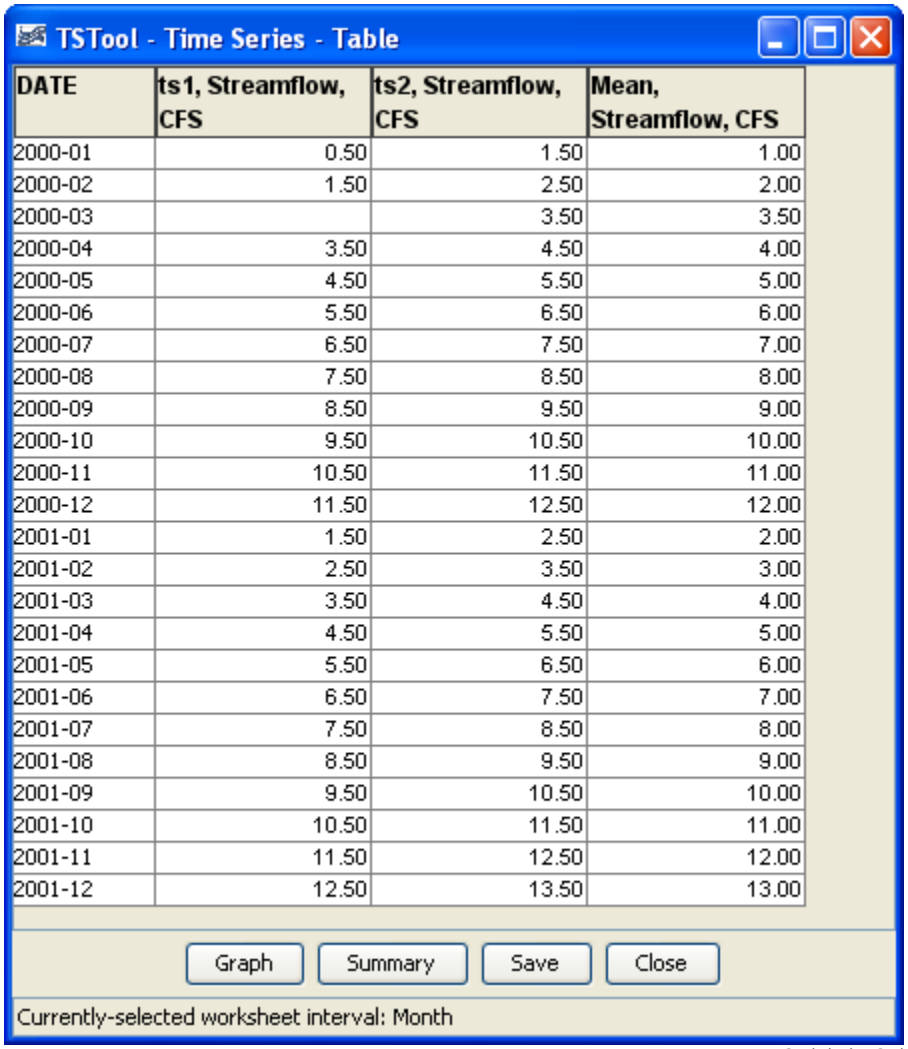
Examples

The following example command file illustrates how to compute the mean statistic for one monthly data:

```
# Test computing a statistic time series for Month data where Statistic=Mean
StartLog(LogFile="Results/Test_NewStatisticTimeSeriesFromEnsemble_Month_Mean.TSTool.
log")
# Define 2 years of data that when averaged equal even numbers
# The 2nd time series is shifted by 1 from the first.
# Include missing values in the first time series but not the second.
NewPatternTimeSeries(Alias="ts1",NewTSID="ts1..Streamflow.Month",
    Description="test data 1",SetStart="2000-01",SetEnd="2001-12",Units="CFS",
    PatternValues=".5,1.5,,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,
    1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,12.5")
```

```
NewPatternTimeSeries(Alias="ts2",NewTSID="ts2..Streamflow.Month",
    Description="test data 2",SetStart="2000-01",SetEnd="2001-12",Units="CFS",
    PatternValues="1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,12.5,
    2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,12.5,13.5")
# Create an ensemble to hold the above time series
NewEnsemble(TSList=AllTS,NewEnsembleID="TestEnsemble",NewEnsembleName="Test
Ensemble")
# Compute the statistic
NewStatisticTimeSeriesFromEnsemble(Alias="Mean",EnsembleID="TestEnsemble",
    NewTSID="Test..Streamflow.Month.Mean",Statistic=Mean)
```

The following figure illustrates the results:



DATE	ts1, Streamflow, CFS	ts2, Streamflow, CFS	Mean, Streamflow, CFS
2000-01	0.50	1.50	1.00
2000-02	1.50	2.50	2.00
2000-03		3.50	3.50
2000-04	3.50	4.50	4.00
2000-05	4.50	5.50	5.00
2000-06	5.50	6.50	6.00
2000-07	6.50	7.50	7.00
2000-08	7.50	8.50	8.00
2000-09	8.50	9.50	9.00
2000-10	9.50	10.50	10.00
2000-11	10.50	11.50	11.00
2000-12	11.50	12.50	12.00
2001-01	1.50	2.50	2.00
2001-02	2.50	3.50	3.00
2001-03	3.50	4.50	4.00
2001-04	4.50	5.50	5.00
2001-05	5.50	6.50	6.00
2001-06	6.50	7.50	7.00
2001-07	7.50	8.50	8.00
2001-08	8.50	9.50	9.00
2001-09	9.50	10.50	10.00
2001-10	10.50	11.50	11.00
2001-11	11.50	12.50	12.00
2001-12	12.50	13.50	13.00

NewStatisticTimeSeriesFromEnsemble_Table

NewStatisticTimeSeriesFromEnsemble() Command Results