21 Secondary Validation

What	*nameofinstance*.xml
Description	Configuration for the Secondary Validation module
schema location	https://fewsdocs.deltares.nl/schemas/version1.0/secondaryValidation.xsd

SecondaryValidation

The SecondaryValidation module can be used to perform certain checks on time series data and generate log messages when the specified criteria are met.

- Checks for counting reliable, doubtful, unreliable and missing values
- SeriesComparisonCheck
- FlagsComparisonCheck
- SpatialHomogeneityCheck
- Mann-KendallCheck
- FlagPersistencyCheck

General overview of this page

Configuration

An XML file for configuring an instance of the SecondaryValidation module called for example CheckImportedData would be the following:

CheckImportedData 1.00 default.xml

CheckImportedData	File name for the CheckImportedData configuration.
1.00	Version number
default	Flag to indicate the version is the default configuration (otherwise omitted).

A SecondaryValidation configuration file is typically located in the ModuleConfigFiles folder and can be used to configure one or more checks. The configured checks will be processed one by one in the specified order. The checks can generate log messages, which can trigger actions in the master controller, like e.g. sending warning e-mails. A special type of check is available for automatically modifying flags to 'doubtful' or 'unreliable' per time step when a condition on multiple time series becomes true.

Checks for counting reliable, doubtful, unreliable and missing values

These checks are intended for generating log events when a specific constraint is violated. The time series configured in these checks will be processed one by one. If a time series does not pass the check, then the configured log message is logged with the specified event code and level. The log event code can be used to trigger a certain action in the master controller, e.g. sending warning emails.

The following different types of checks are available:

- **minNumberOfValuesCheck**: Logs a message when there are not enough values within a configured period. This check is only useful for nonequidistant timeseries because also missing values are counted. In case of equidistant timeseries simply the number of timesteps in the relative viewperiod are returned.
- **minNonMissingValuesCheck**: Logs a message when there are not enough non-missing values within a configured period. A non-missing value is a value that is reliable, doubtful or unreliable.
- **minReliableOrDoubtfulValuesCheck**: Logs a message when there are not enough values that are reliable or doubtful within a configured period.
- **minReliableValuesCheck**: Logs a message when there are not enough reliable values within a configured period.

Check for setting flags per time step using an expression

The **seriesComparisonCheck** check is available for testing constraints between multiple time series per time step.
This check verifies constraints between multiple time series sets or multiple parameters and automatically modifies the flags per time step when the required input data was available (reliable or doubtful) and the specified expression is evaluated and is true.

Check for setting flags per time step using other timeseries

The **flagsComparisonCheck** check is available for comparing and setting flags for multiple time series per time step.
This check determines for each timestep the most unreliable input flag within the input flags, and if it is more unreliable than the output flag it updates the output flag.

Check for setting flags per time step using comparisons with neighbouring locations

The **spatialHomogeneityCheck** check is available for comparing and setting flags for multiple time series per time step.
This check determines for the selected output timeseries at each timestep the mean error with neighbouring locations. If the mean error exceeds the specified absolute threshold or exceeds the specified relative factor times the standard deviation, then the output flag is updated to the selected output flag whenever that new flag is more unreliable than the existing flag.

Check for detecting trends

The **mannKendallCheck** check is available for detecting trends. This can be useful for monitor sensors on drift.

Variable Definitions

The configuration contains variable definitions for one or more time series that can be used as input for checks. Each variable definition contains a variableId and a timeSeriesSet. The variableId can be used to reference the time series in a check. Alternatively, depending on which check it is, either variable definitions or variables can be embedded in the checks.

Multiple checks for a time series

When multiple Secondary Validation checks that could change data for the same time series are necessary within 1 workflow, all checks should be present within 1 and the same module config file.

This is because at the beginning of the Secondary Validation module all previous secondary validations are being undone for consistent behaviour and for significant database performance reasons.

Behavioural changes in validating simulated time series created in previous workflow

Since 2019.02 #92895 there have been fixes in the writing of output time series in the Secondary Validation module. These fixes improved consistency in writing behaviour making sure repeated runs will result in the same outcome.

This, however, resulted in significant different outcome for some very specific use cases involving simulated time series. In case simulated time series were being validated as output time series in a workflow that did not create them, it results in the original data being hidden since the new write action creates a newer module run instance for the data. Changing simulated data in a different workflow than it was created should not have been allowed, but because of the previous inconsistencies in writing behaviour it was possible to use this data as output for the secondary validation module. In order to still support validation of simulated data in a different workflow than it was created, a "logs only" mode has been introduced. This mode can be considered as "read only" and will be described below.

Logs only (read only) mode

This mode has been introduced to enable validation of simulated time series that have been created in a previous workflow.

When simulated time series created in a previous workflow are being validated outside of the "logs only" mode it results in the original data being hidden since the new write action creates a newer module run instance for the data resulting in hiding of the original data.

It lets the Secondary Validation module know that there will be no data changes and therefore previous Secondary Validation does not need to be undone which normally is required for consistent and repeatable behaviour.

The Secondary Validation module will at the start check whether there in any check configured that could possibly change the data and therefore requires a write action.

As long as if it is certain there will not be any write actions then the "logs only" mode will be enabled.

- **minNumberOfValuesCheck**: never required write actions
- **minNonMissingValuesCheck**: never required write actions
- **minReliableOrDoubtfulValuesCheck**: never required write actions
- **minReliableValuesCheck**: never required write actions
- **seriesComparisonCheck:** requires write actions when there is no <outputMode> defined or <outputMode> is flags_and_logs (default). Does not require write action when <outputMode>logs_only</outputMode> is configured
- **flagsComparisonCheck:** requires write actions when there is no <outputMode> defined or <outputMode> is flags_and_logs (default). Does not require write action when <outputMode>logs_only</outputMode> is configured
- **spatialHomogeneityCheck:** requires write actions when in ANY of the thresholds there is no <outputMode> defined or <outputMode> is flags_and_logs (default). Does not require write action when in ALL of the thresholds <outputMode>logs_only</outputMode> is configured
- **mannKendallCheck:** requires write actions when in ANY of the thresholds there is no <outputMode> defined or <outputMode> is flags_and_logs (default). Does not require write action when in ALL of the thresholds <outputMode>logs_only</outputMode> is configured
- **flagPersistencyCheck:** ALWAYS requires write actions, when this check is present in the Secondary Validation module, it will NEVER go into logs_only mode