

# selectRankedMember - GenerationEnsemble

## Description:

This transformation can be used to select individual member(s) of an ensemble using a statistical feature, such as the driest member(s), or the wettest member(s).

The statistical feature is computed for the entire ensemble in another transformation and expressed as a single value statistical result at T0 for each ensemble member. The ensemble timeseries with the statistical result is specified as the ranking variable within the selectVariable. In addition, one needs to specify the order of ranking (default is ascending) and the number of members to select (starting from the top).

The member selection (e.g. member 3) is used in the inputOutput section to select member 3 from the input ensemble and assign that member to the output ensemble. The input ensemble id needs to be the same ensemble id as the ensemble id of the ranking variable. The output ensemble id should be a different ensemble id.

***Be aware:*** This transformation generates a subset of members that meet the ranking criteria for this ensemble forecast. However, as long as previous selected members from a previous forecast are not yet expired, the FEWS-datastore will include all members of this (output) ensembleid in its processing, also the (empty) members from a previous ranked ensemble. The FEWS-datastore is thus not capable to only include the selection of the current/active forecast in your processing, it always includes all members available the database.

How does this work:

1. a subset of the ensemble forecast of 2019-02-02 is created resulting in member 2 and 4 being part of the output ensemble OUT. These member will expire on 2019-02-04.  
On 2019-02-02 (after the selection) the FEWS-datastore view on ensemble OUT sees two members: 2 and 4.
2. a subset of the ensemble forecast of 2019-02-03 is created resulting in member 3 and 5 being part of the output ensemble OUT. These member will expire on 2019-02-05. On 2019-02-03 (after the selection) the FEWS-datastore view on ensemble OUT sees four members: 2, 3, 4 and 5, where members 2 and 4 are present but empty (since not current/part of the last selection), while member 3 and 5 are populated.

On 2019-02-04, the timeseries created on 2019-02-02 with member 2 and 4 is expired. Hence the FEWS-datastore view on ensemble OUT sees only two members: 3 and 5 (which are populated as they are current/the last selection).

*How can you prevent that you keep carrying (empty) ensemble members that are not in your last selection*

The best methods is to ensure that your OUT ensemble timeseries are expired before you process a new input ensemble again to create a new subset.

An alternative method to get rid of the ensemble id is to copy the output ensemble to a (set of) deterministic series. Since a copy in the transformation module will always include all members in a copy, it is advised to create a file export-import with the general adapter.

An example how this could be achieved is attached: [SampleOrderedByQ](#) creates the ensemble subset based on the ranking. [Copy\\_SampleOrderedbyQ](#) turns the members into a deterministic series using a general adapter.

## fews:SelectRankedMembersTransformationComplexType



### fews:className

Optional element to define the name of the class that should be executed to perform this transformation. Normally this element can be left out of the xml configuration, and the default class name will be used. This default class name is defined as the default value for this element in the xsd schema for this transformation. Only use this element if you need to use a custom class that is present in the program code.



### fews:period

This element can be used to configure the run period of the transformation.

### fews:selectRankedMembers

Since 2018.02 Selects ensemble members by ranking of the (statistical) results of the ensemble members

### fews:selectVariable

Ensemble time series used for selecting of ensemble member Id's

## fews:SelectMembersVariableComplexType

### fews:rankingVariable

Ensemble time series. Each member holds just one value. This value is a (statistical) result of the analysis of the ensemble member. RankingVariable members will be sorted ascending or descending

### fews:sortAscending

Ordering method to rank the members of 'rankingVariable'. Ascending by default

### fews:numberOfMembers

Number of members to select, starting from top of list



### fews:inputOutput

1..∞

Selected member Id's are used to select the same members from the inputVariable. The selected members are written to the outputVariable

## fews:InputOutputVariableComplexType

### fews:inputVariable

### fews:outputVariable

### fews:className