# Data and provenance management for climate effect studies

Adaption of climate data with distribution based scaling for hydrological simulations.

Lena Strömbäck, Kean Foster, Jörgen Rosberg
Swedish Meteorological and Hydrological Institute (SMHI)
Norrköping, Sweden
e-mail: {lena.stromback, kean.foster, jorgen.rosberg}@smhi.se

*Abstract* — **Climate effect studies are currently of high interest to predict the impact of a changing climate. The results of such studies are used by decision makers as a basis for planning how to mitigate and adapt to the effects of expected climate changes. However, these studies require heavy computations on large sets of data in several steps. This combination of heavy computation and results being basis for important decisions makes it extremely important to have an efficient and well documented process for computations, to provide accurate results in an efficient way. In this paper we describe the problem and present our DBS (Distribution Based Scaling) tailoring tool that has been implemented to support the process. We discuss the problem and our solution in relation to scientific workflow systems and provenance engines in general.**

***Keywords-workflow systems; climate studies; hydrology; data management; quality insurance; DBS tailoring tool; provenance***

## I. INTRODUCTION

Today there is a huge demand for knowledge on climate change and how this has an impact on our environment. Information on possible outcomes of climate change comes from the numerical global circulation models (GCMs). The GCMs model the climate for the entire globe, from the past into the future. Different assumptions on how the greenhouse gas emissions will evolve can thus be tested within this framework. However, the scale of the information obtained from those models is often too coarse for any impact study on a regional scale. To downscale the information from the GCMs either statistical methods or regional climate models (RCM) are used. However, to be able to use this data for hydrological predictions we need realistic input data to the hydrological model about future occurrences of rain and temperature. This requires that the regional information is even further downscaled and bias corrected to ensure that the provided data represents a realistic distribution of precipitation and temperature in time and space. Therefore historical simulations of temperature and rain are compared with historical observations to calibrate an adjustment schema which is applied to future predictions. This process is called Distribution Based Scaling, DBS.

From a computer science perspective this process involves a number of interesting challenges. First of all climate studies involves time series of daily values with a high geographic resolution. This means that we need to consider gigabytes of data that need to be efficiently stored and processed. Secondly, the input data from RCMs can have different representation, meaning that there is a need to manage and translate these huge datasets between different data formats. In addition, we need methods for quality insurance i.e. to detect and correct faulty data or errors in the processing. Finally, there is a need to document the process for further scientific development of the procedures, data and models. An important aspect of the problem is recording provenance, i.e. to record the history of the correction process making a new result comparable to older runs.

The problem in many ways resembles problems in other scientific disciplines where derived results are dependent on data from many different data sources, versions of data, and versions derived from analyses and simulations. For all these scientific areas it is extremely important to keep track of all steps in the process. One of the most common approaches for recording provenance is scientific workflows. In this field a number of workflow based tools have been implemented ([1] and [2] gives an overview.)

In this paper we will discuss the issues and requirements around DBS applications, present our current system for the process and discuss how it relates to scientific workflows and scientific workflow tools in general. The paper starts with a more thorough introduction to the DBS process. After this we give a more thorough discussion of the computational challenges and how they relate to problems addressed in general by scientific workflows. Finally, we present the main features of our implemented system and discuss how this system relates to scientific workflow engines in general.

## II. DISTRIBUTION BASED SCALING

As explained in the introduction, even though output data from climate scenarios give accurate predictions on the expected long term changes in climate, they do not provide the accurate detailed information needed as input for hydrological simulations. For instance, even though the total amount of precipitation is the same for a longer period, rainfall in the climate models tend to be much less varied on a daily basis than can be expected if we compare with current observations. Therefore this data need to be adjusted according to an observed reference period to provide more accurate input.

In the case of precipitation, the DBS approach uses two steps: (1) spurious drizzle generated by the climate model is removed to obtain the correct percentage of wet days and (2)

 ISBN: 978-1-61208-247-9 

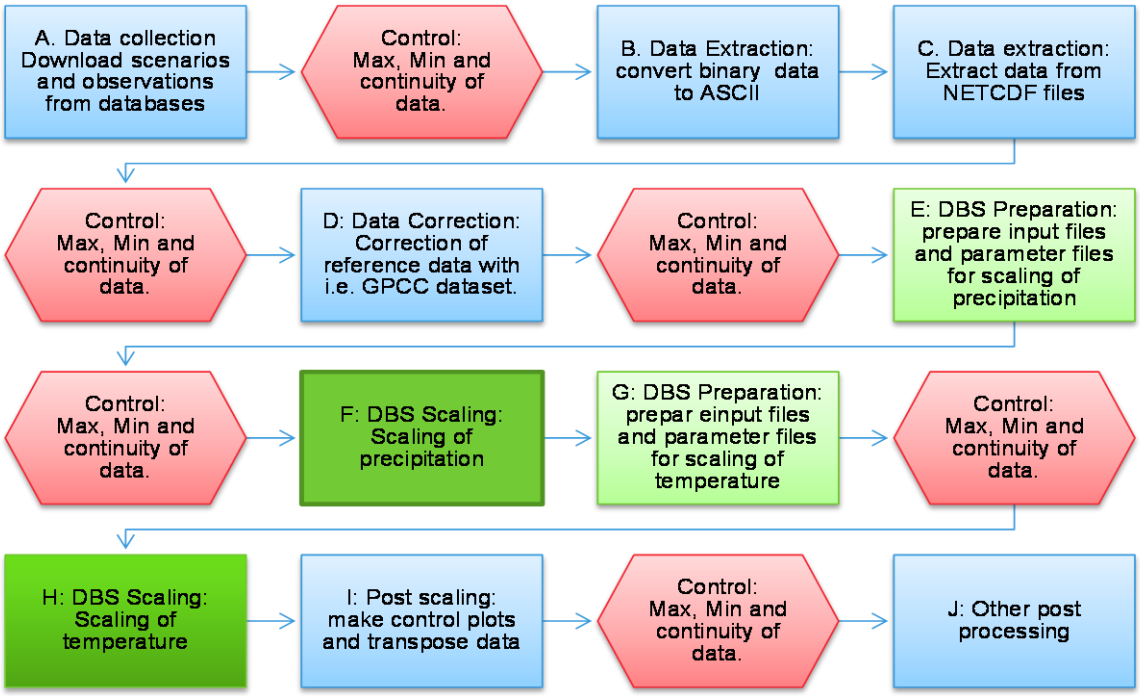


Figure 1. A schematic figure of the general scaling process. The dark green boxes (F, H) correspond to the actual DBS scaling, while the lighter green boxes (E, G) correspond to preparation of statistical information (step 1, 2, 4 and 5 in the process).. Blue (A, B, C, D, I and J) represent other necesseray preparation and conversion of the raw input data to the process as well ass post processing needed to prepare the data for different hydrological engines. Finally, the read diamonds represent data quality controls that are necessary to detect errors in the data transformation process.

the remaining precipitation is transformed to match a sample for current observed frequency distribution. To obtain the percentage of wet days correctly, a threshold is identified for each sub-basin and season. The sub-basin represents the geographical resolution and for Europe we typically work with around 40 000 sub-basins. Days with precipitation amount larger than the threshold value were considered as wet days and all other days as dry days [3], [4], [5]. After this the temperature from the simulation is adjusted based on the new precipitation and an observed reference period for the temperature.

This means that a typical DBS process can be broken down into six general steps:

1. Calculate statistical parameters for the wet day threshold and distribution of the observed precipitation.
2. Calculate statistical parameters for the wet day threshold and distribution of the precipitation provided by the climate model.
3. Scaling of the precipitation from the climate model.
4. Calculate statistical parameters related to observed temperature.
5. Calculate statistical parameters related to the temperature provided by the climate model.
6. Scaling of the temperature from the climate model.

In practice the process includes several steps of data conversion as climate simulations and reference data may occur in different formats. In addition, to ensure a correct result from the process it is also important to perform quality control between the steps in order to detect errors as early as possibly in the processing. A schematic picture describing the general process is given in figure 1.

## III. PROBLEMS AND CHALLENGES

In this section we discuss the main challenges we face for an efficient data management for the DBS scaling process.

### A. *Large datasets*

For each step in the processing chain we are faced to manage and transform very large volumes of data. As an example, our hydrological model [6] represents Europe by approximately 40 000 sub-basins. The model uses time series representing daily values for temperature and precipitation as input for simulating hydrological conditions. For climate impact studies a typical time series represents 100 years of daily values. This results in gigabytes of data that need to be efficiently processed. This large volumes of data put high requirements on data storage as well as efficient processing.

### B. *Long processing chains and processing time*

As seen in the schematic picture in figure 1, the processing chain consists of several steps. In reality, most of the included boxes can be further broken down to several individual steps. In many of the steps the processing time is long resulting of computation times of days for the whole chain.

 ISBN: 978-1-61208-247-9 

### C. Data formats and data conversion

The data formatting and conversions can involve several different tasks; for example, to select relevant data in time and space from the simulated climate model, to convert data between different geographic representation (e.g. grid vs. sub-basin), to convert data between different data representations (e.g. NetCDF vs. ASCII). As the data quantities are large each of these processes has to be efficient. Moreover, in many cases the order of conversions affects the efficiency of the process.

### D. Quality assurance

As we are working with research, data and processes are under constant development. This means that quality assurance is extremely important. Errors may occur from faults in the observed or climate data, but also by mistakes in the selection or data conversion process. In many cases strange values of data origins from new geographical conditions and it is important to analyze to further improve the scaling method. Thus, detecting and determining the source of error or strange value is critical. As the total computation time for the chain is long it is important to do this as soon as possible to avoid delays in producing results. Therefore, it is crucial to check data quality after each of these steps.

### E. Reproducibility

The final result of the DBS process is used for hydrological simulations and in many cases published or exported to a customer. For comparison with other similar results, future reruns with updates of the model or discussions about the validity of results, it is extremely important to store a record of the whole processing chain, i.e. the provenance of the final result.

### F. Cooperation

Due to the long processing chains there is often a need for cooperation of researchers to produce one result. In many cases we need to keep records of old data and details of a run to train new researchers in performing the scaling process. Therefore recording of details around the process is very important.

These challenges in many ways address the same problems as scientific workflows or provenance management systems [1], [2]. However, there are also some main differences. Although our process chains include variation, it is in general more static than processes represented in scientific workflows systems. Moreover, the large volumes of data and long execution times for our process must be taken into account when designing a tool. We will further elaborate on these differences after presenting the main features of our implemented system.

## IV. THE DBS TAILORING SYSTEM

The DBS tailoring system is used for facilitating bias correction/downscaling using the DBS approach described above. The DBS tailoring system is used both to prepare the different data and control files needed to perform a DBS bias correction/downscaling and run the DBS motor. As described in section II a typical DBS job can be broken down into six general steps. Around these main steps we need supporting modules for data conversion and quality control. The DBS tailoring system helps to create the required files for the different steps and controls the process. In this section we give an overview of the system, for a more detailed description see [7].

### A. General Architecture

The general architecture for the system is, as described in figure 2, a general process engine and a set of different modules for the different processing steps or tasks. In the general engine, the user can define the processing chain and put together the different tasks as desired. The processing chain is described by XML files that define how to perform the different tasks in the chain. These XML files can be directly defined by the user but there is also a graphical user interface for support. This means that the different modules are loosely coupled and that we achieve a high flexibility in how to combine the processes to achieve the desired functionality.

In addition, the design of the system gives a high flexibility in supporting new modules. The general process engine is implemented in Java, and processes can be implemented in any programming language that can be called from Java. Currently the process modules are implemented in Java, for data conversion, or Fortran for the core DBS scaling. However, the general architecture makes it easy to integrate new process modules in the system independently on whether these are fetched from existing libraries or whether they are developed by our research team.

### B. XML process scripts

One of the most important features of the system is the process chain description used for controlling the execution. The process script are used to document a run, but is also the key for reproducibility, as it contains all details about the process, such as versions of data and processes, that is needed to rerun a process. The example in figure 4 shows the principles behind the process description. In the example we can find the six steps involved in the DBS scaling, parameter preparation and scaling for the precipitation and then corresponding for the temperature. Between these main tasks

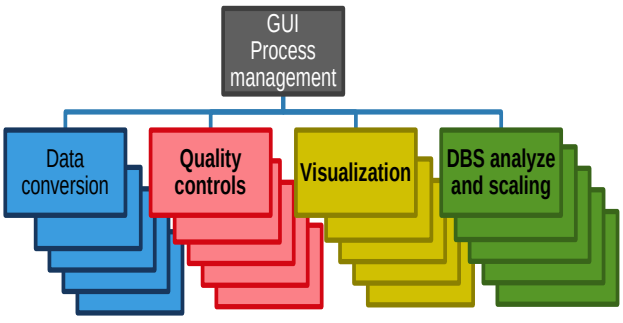


Figure 2. A schematic description f the general architecture for the DBS tailoring system. The system consist of a graphical user interface and process managementt system and a large number of modules for data conversion, quality control, visualization and the DBS scaling. The user can use the GUI to design the desired process chain.

 ISBN: 978-1-61208-247-9 

```
- <Tasklist>
+ <Task Tasktype="LOAD" Taskgroupid="1" Taskorder="1">
  ...
+ <Task Tasktype="EXPORT" Taskgroupid="1" Taskorder="4">
  ...
- <Task Tasktype="DBS" Taskgroupid="1" Taskorder="9">
  <Critical_task>true</Critical_task>  >
  <DBStype>REFP</DBStype>  </Task>
  ...
- <Task " Tasktype="DBS" Taskgroupid="1" Taskorder="12">
  <Critical_task>true</Critical_task>
  <DBStype>SIMP</DBStype> </Task>
  ...
- <Task Tasktype="DBS" Taskgroupid="1" Taskorder="15">
  <Critical_task>true</Critical_task>
  <DBStype>SCALET</DBStype> </Task>
  ...
- <Task Tasktype="DBS" Taskgroupid="1" Taskorder="18">
  <Critical_task>true</Critical_task>
  <DBStype>REFT</DBStype> </Task>
  ....
- <Task Tasktype="DBS" Taskgroupid="1" Taskorder="21">
  <Critical_task>true</Critical_task>
  <DBStype>SIMT</DBStype>  </Task>
  ...
- <Task Tasktype="DBS" Taskgroupid="1" Taskorder="24">
  <Critical_task>true</Critical_task>
  <DBStype>SCALET</DBStype> </Task>
  ...
- <Task Tasktype="PLOT" Taskgroupid="1" Taskorder="28">
  ...
- <Task Tasktype="PLOT" Taskgroupid="1" Taskorder="31">
- <Task Tasktype="FLIPPER" Taskgroupid="1"
      Taskorder="32">
- <Task Tasktype="FLIPPER" Taskgroupid="1"
      Taskorder="33">
  </Tasklist>
```

Figure 4. An example of the XML script describing the DBS Tailoring process. The example shows a typical process schema for DBS scaling. The example have been shortened to save space and improve readability. In principle detailes about the rpocesses have been removed. In addition we have removed many LOAD and export taks, these placement of these is marked with dots.

there are a number of supporting processes. Here LOAD and EXPORT is used to define and extract data needed for the process, PLOT derives maps for validating the results and finally FLIPPER provides matrix transposition of data, as needed to prepare it for the hydrological simulation.

### C. User Interface

The DBS tailoring system provides a user interface to manage the XML script files. From the user interface the user can define the sequence of modules by selecting the desired module types. For each module type the interface provides a menu that prompts the user with required values and data. Figure 3 shows an example from the he final DBS step, the scaling of the temperature.

To further aid the user old XML process files can be loaded into the system and used as templates for defining new jobs. This is beneficial as the processes are similar. In principle there are a few types of template processes depending on the data types of input and output data and for such cases, the end user only need to alter paths to actual data files and periods for calculations.

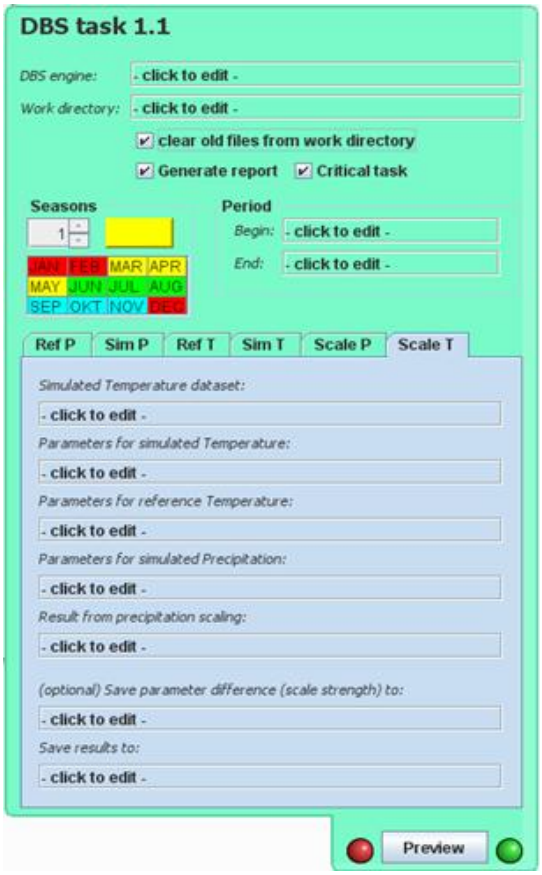


Figure 3. The user interface for defining last of the DBS tasks scaling the temperature dataset. The user need to enter where the DBS engine and the work folder are located. As calculations is season dependent he also need to define the seasons. The user also must enter the period for this task, i.e. dates between which calculations need to be done. Finally, the user enters he files the scaling is dependent on, note that, in this case, several files are needed more as the temperature scaling is dependent information from the precipitation files.

The interface also provides an easy environment for running the processes and monitors the results.

### D. Data conversion and efficiency

The current implementation support geographical and time series data represented as shape files, netCDF files or plain ASCII files. ASCII files can be of three types; Discrete – non continuous data such as parameter files; Time series – 'time' orientated time series i.e. they have time on the x-axis; and Id series – 'id' orientated time series i.e. they have id on the x-axis. The implemented functionality for managing these files is LOAD and EXTRACT.

The load task allows loading file information for the different file types into the system. This step creates an index file called a POSMAP file which contains information regarding the data contained in the file i.e. number of data points, data ranges etc. These POSMAP files allows the system to identify where in the ASCII files the desired data is stored which allows for faster reading of the data.

 ISBN: 978-1-61208-247-9 

The export task allows you to export data from a file. The task can be divided into two main types of export i.e. using geographic information and using unique ids to select the relevant data. The former type is mostly used to extract data from netCDF files for the scaling process. This task uses the geographic data in the shape file to select the nearest corresponding point in space from the netCDF file and assigns the id number from the shape file to the data at that point.

The latter export type is more versatile and can be used for a number of different tasks, for instance, to export statistics for the data to a shape file for analysis purposes, or extract a subset from an existing dataset.

This machinery gives a flexible management of data conversion. It is efficient since the load functionality with POSMAP files avoids duplicating data Also choices in conversion orders allows for optimization choices. As an example, it is often preferable to keep the grid representation of geographic information as long as possible as it has a lower geographic resolution than the sub-basin representation of data.

### *E. Quality insurance*

Within the framework we typically use two kinds of quality checks, automatic statistical checks and generation of maps that can be manually inspected by an expert. The second type is demonstrated by the plot functionality in the example in figure 4. This functionality allows for plotting for instance maximal and minimal values for precipitation and temperature for each sub-basin. This allows an expert to easily inspect that the results are valid.

Another kind of control, not demonstrated by the example is statistical computations for finding flaws in the data. Examples of these are; too high or too low values; or sudden jumps in the measured or simulated values which indicate that something has gone wrong in the process.

This is important as errors can occur, for instance, by having to few observed values for one single sub-basin and period for a point. Such data can make the statistical predictions uncertain which indicate that the process has to be rerun with other input data or different parameter settings.

## V. RELATED WORK

As the implemented system covers many areas there are a lot of interesting related work, for instance, within data management and efficient processing. However, in this paper we will focus on how it relates to provenance and scientific workflows. Scientific workflows and workflow based systems [1], [2], [8], [9], [10] have emerged as an alternative to ad-hoc approaches for documenting computational experiments and designing complex processes. They provide a simple programming model whereby a sequence of tasks (or modules) is composed by connecting the outputs of one task to the inputs of another. Workflows can thus be viewed as graphs, where nodes represent modules and edges capture the flow of data between the processes.

The actual features and representation of a scientific workflow differ between the systems, due to varied needs from the application areas and users they are designed for.

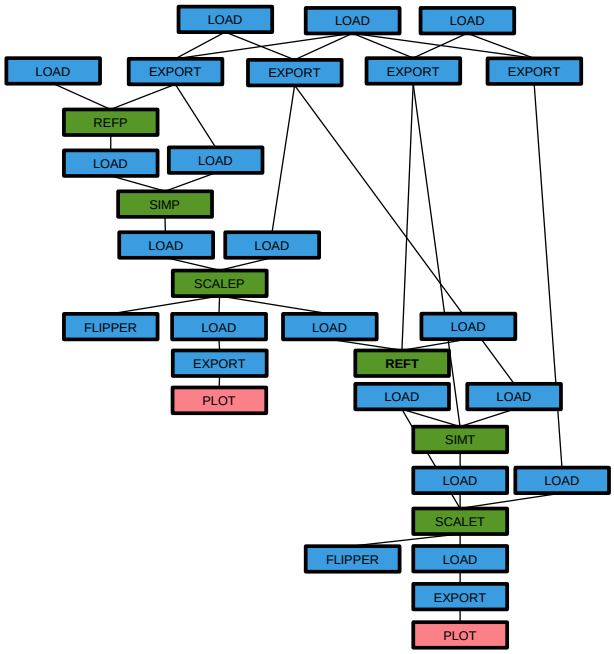


Figure 5. The sample DBS tailoring process represented as a scientific workflow. The figure gives an overview that shows the general flow of data between the processes. As in figute 1, the actual DBS scaling correspond to the green boxes, whike control plots are red. The blue boxes represent different kinds of data conversions. The blue LOAD typically creates an index for more efficient processing of the file, needed before each step in the process.

Therefor, systems tend to work in their own internal format and it is becoming common to provide conversion to other formats, e.g., the Open Provenance Model [11] and mediation approach [11]

There is a very strong relationship to our problem and scientific workflows. As discussed already in section III the basic requirements, i.e. documenting the scientific process and provenance of data is the same for our application and scientific workflows. In addition, the structure of our process chain is similar to the graph structure used for scientific workflows. Figure 5 shows schematically how the process chain in figure 2 could be represented as a scientific workflow. Although our current implementation uses an XML format and user interface that is tailored for our needs, it would be possible to translate this representation to the Open Provenance Model and thus, it can be imported to several scientific workflows systems. This would in give access to the features implemented in many of the available workflow systems. Here, we will discuss some of the features provided by VisTrails [10] which is one of the most advanced available systems.

VisTrails supports exploratory computation tasks. It has a graphical user interface that is used for the composition and execution of workflows. Data and workflow provenance is uniformly captured to ensure reproducibility of results by others. Workflows can be composed by program libraries (Python) or by external web services. VisTrails has been used in the fields of biology and earth science.

One of the most important features of VisTrails is its ability to document the provenance of the development

 ISBN: 978-1-61208-247-9 

process. This means that the system records whether one workflow is developed based on other used solutions. Even though workflows in general support cooperation between researchers, this feature further enhances cooperation as the relation between different versions is visualized by the tool. This is something not provided by our implemented system, but very useful for recording provenance and exploring the difference between workflows. Therefore it is interesting to explore how this or something similar can be included in our system in the future.

In addition to this VisTrails contains a number of interesting features, such as; the possibility to explore the parameter space of a workflow; comparing two workflows and applying the changes between them to another similar workflow; and various search and presentation facilities. All these features are very powerful when working with large collections of workflows and of interest also for our application.

## VI. FUTURE WORK

The current implementation of the DBS tailoring system is in use and supports the process of preparing data used for hydrological impact studies. In a near future we will run several of these studies, on different geographic areas and based on different climate models. During this phase we will use the system for documenting the process and follow up the quality. This is a perfect opportunity to test the ability of the system and learn where it can further improve.

As the current system gives a good documentation of each process chain, but lacks information on relations between different chains this is a particular point of interest for us. Here the functionalities for scientific workflow systems in general and in particular VisTrails will be very interesting to explore further to see whether they can be adapted to our settings. In principle, two solutions are possible, extending our implementation with these features or exporting our process description to make use of an existing tool, such as VisTrails.

One of the most interesting issues to explore is how the inherent properties of our application, i.e. large data sets, long processing times and relatively static processing chains compared to many other applications where scientific workflows are used affects how these feature is realized. For instance, how can the recorded information be used for avoiding duplication of data and rerun of expensive computation processes in an optimal way.

## VII. CONCLUSION

Climate effect studies require heavy computation and the results are being the basis for important decisions in society which makes it extremely important to have an efficient and well documented process for computations. This paper gave an overview of the problem and our DBS tailoring tool that has been implemented to support the process and compare it with scientific workflow tools in general. The comparison shows that our tool and scientific workflow tools has many common properties, such as documentation of the process, even though scientific workflows systems in general have a number of additional functionality. In the future we will investigate how we can reuse some of these features in our setting to further improve our computation process.

## ACKNOWLEDGMENT

The development of the DBS tailoring system has been done as a part of several projects involving hydrological effect studies. The work in this paper has been funded by the Swedish Research Council (Vetenskapsrådet), The Swedish Environmental Protection Agency, The Swedish Research Council Formas and the European Commission. We are also grateful to Jonas Olsson for valuable comments on the work.

## REFERENCES

[1] J. Freire, D. Koop, E. Santos, and C. Silva, Provenance for Computational Tasks: A Survey. *Computing in Science and Engineering*, 2008

[2] S. Davidson, and J. Freire J: "Provenance and scientific workflows: challenges and opportunities", SIGMOD, 1345-1350, 2008

[3] D.S. Wilks, 1995 "*Statistical Methods in the Atmospheric Sciences: An Introduction."* Academic Press, INC, Burlington,MA, p. 86, 1995

[4] M. R. Haylock, G. C. Cawley, C. Harpham, R.L. Wilby, C. M. Goodess, "Downscaling heavy precipitation over the United Kingdom: a comparison of dynamical and statistical methods and their future scenarios", *Int. J. Climatol*. 26, 1397–1415, 2006

[5] W. Yang,, J. Andréasson, L.P. Graham, J. Olsson, J. Rosberg, F. Wetterhall "Improved use of RCM simulations in hydrological climate change impact studies", *Hydrol. Res*., 41, 211-229, 2010

[6] G. Lindström, C. Pers, J. Rosberg, J. Strömqvist, B. Arheimer, "Development and test of the HYPE (Hydrological Predictions for the Environment) model - A water quality model for different spatial scales", *Hydrol. Research 41* (3-4): 295-319, 2010

[7] K. Foster "*DBS Tailoring System An operators manual*". CLEO project report. SMHI, 2012

[8] D. Hull, K. Wolstencroft, R. Stevens, C. Goble, M. Pocock, P. Li, *et al.*, "Taverna: a tool for building and running workflows of services*." Nucleic Acids Research*, 2006.

[9] T. Oinn, M. Greenwood, M. Addis, N. Alpdemir, J. Ferris, K. Glover, et al. "Taverna: lessons in creating a workflow environment for the life sciences," *Concurrency and Computation: Practice and Experience*, 2006.

[10] L. Bavoil, S. P. Callahan, P. J. Crossno, J. Freire, C. E. Scheidegger, C. T. Silva, *et al.*, "Vistrails: Enabling interactive multiple-view visualizations," In Proceedings of IEEE Visualization, 2005. Information Sciences Institute, "Pegasus:home,"

[11] L. Moreau, J. Freire, J. Futrelle, R. E. McGrath, J. Myers, and P. Paulson, "The open provenance model," 2008. [Online]. Available: http://eprints.ecs.soton.ac.uk/14979/1/opm.pdf (Last visited 2012-12-06)

[12] T. Ellkvist, D. Koop, J. Freire, C. Silva, and L. Strömbäck, "Using mediation to achieve provenance interoperability," in IEEE Workshop on Scientific Workflows, 2009. [Online]. Available: http://vgc.poly.edu/~juliana/pub/provinterop-swf2009.pdf (Last visited 2012-12-06)

 ISBN: 978-1-61208-247-9 