Notay, K. V.; Li, C.-Y.; Molkenthin, F.; Simons, F.; Hinkelmann, R.

Model Integration and Coupling in A Hydroinformatics System

Proceedings of ninth International Conference on Hydro-Science and Engineering (ICHE 2010), IIT Madras, Chennai, India.
2 - 5 August 2010

MODEL INTEGRATION AND COUPLING IN A HYDROINFORMATICS SYSTEM

Notay K.V.1, Li C.-Y.2, Molkenthin F.3, Simons F.4, Hinkelmann R.5

Abstract. *As simulation tools are used to study more and more complex phenomena and collaborative engineering gains momentum there is an ever-increasing need for coupling of numerical models. Traditionally this has been achieved by the development of strongly coupled models or by coupling them using technologies such as OpenMI. Such approaches aren't however the ideal solution for every case where coupling of models is required. The current work proposes a new approach in the model-coupling process, which makes use of existing technologies to exchange data through self-sufficient information units – tenor objects – capable of independent existence. Tensor objects contain all the necessary information such as coordinates, units, physical values and the topological and geometric relationships between these values. They are able to adapt themselves to the requirements of the modelling framework in which they exist due to in-built techniques such as mapping, scaling, interpolation etc. Such a framework is quite useful for the process of coupling of interdisciplinary models running on different time and space scales. The models being coupled can produce or use such tensor objects without having to implement routines for processing their own data to suit the requirements of other models. This task is handled within the tensor objects as independent information units. It also makes it possible to couple other types of models such as statistical analysis models, GIS models or visualisation models.*

INTRODUCTION

Over the years a large number of numerical models have been developed for simulating various hydro-engineering problems. There has also been a tendency to try and model increasingly complicated problems in order to get a holistic view of the situation in nature. There two ways of achieving this: either by creating a completely new, tightly integrated model capable of

simulating all the phenomena of interest in a particular study or to couple existing models and to make them interact and exchange information.

1 Research Student, Email: vikram.notay@tu-cottbus.de
2 Research Student, Email: chiyu.li@tu-cottbus.de
3 Scientist, Email: frank.molkenthin@tu-cottbus.de
4 Scientific Assistant, Email: franz.simons@wahyd.tu-berlin.de
5 Professor, Email: reinhard.hinkelmann@wahyd.tu-berlin.de
[1, 2, 3] EuroAquae, Brandenburg University of Technology, Cottbus, Germany
[4, 5] Chair of Water Resources Management and Modeling of Hydrosystems, Technische Universität Berlin, Germany

When coupling numerical models and running them simultaneously in multidisciplinary projects, there is the need to be able to manage information from the various sources and to make it accessible to the different models. Standards have been developed by the Open Modelling Interface and Environment (OpenMI) for the exchange of numerical values between coupled models given the type of the value or value-set and its coordinates in space and time. However, a more general solution which allows information exchange in a neutral format for use in a multidisciplinary environment where a wide range of numerical models such as hydrological models, geophysical models, statistical models etc. are in use does not exist. This paper describes a general approach for model integration and coupling within an interdisciplinary research unit to describe and simulate the complex physical behaviour of natural slopes in mountainous areas as an example.

RESEARCH UNIT "GROSSHANG"

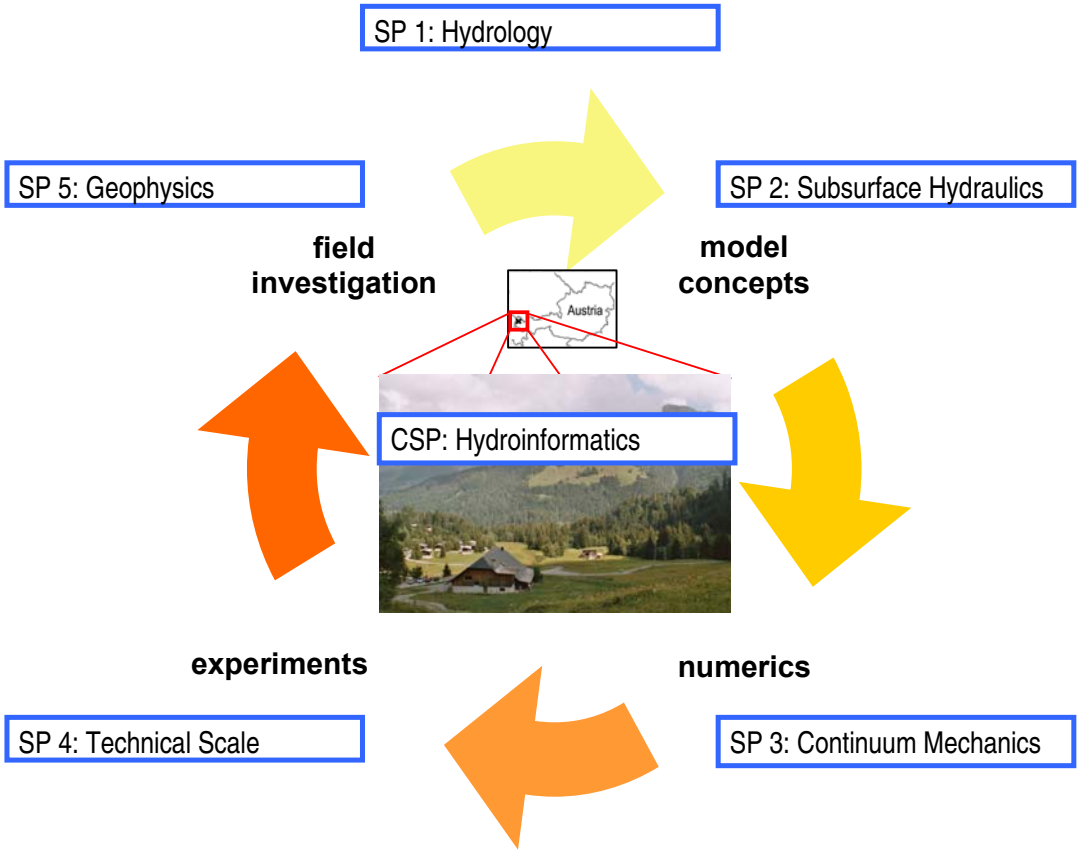

Figure 1: Structure of the research unit "Grosshang"

The interdisciplinary research unit "Grosshang – Natural Slopes" (Hinkelmann & Zehe 2006) deals with "Coupling of Flow and Deformation Processes for Modelling the Movement of Natural Slopes". The target of the research unit is to investigate and simulate the long-term deformation and movement of the mountain slope, more definitely the Heumos Slope in Ebnit, Austria. The project consists of five subprojects (Figure 1) carrying out investigations in various disciplines using field measurements, laboratory experiments and numerical simulation.

The information and data collected from all the sub-projects is stored in a web-based information system, which is done as part of the central subproject.
As part of the project, coupling of the different simulation models applied in the various sub-projects and running on different time and space scales, such as models for simulating surface-runoff, infiltration, subsurface flow, soil-mechanics as well as to integrate field measurements, geophysical measurements etc. is done with the aim of integrating them towards an interdisciplinary, multi-scale simulation approach in order to study the deformation processes in the Heumos slope. Additional laboratory experiments are performed for parameter identification and verification.

MODEL COUPLING CONCEPT

Coupling of models can be done in one of two ways: either to build a tightly coupled model from the beginning according to the needs of the problem at hand or to use existing models and to couple them by exchanging information between them in the form of initial or boundary conditions and so on. The former method might be the best possible solution for one particular problem but may not be easily extensible or applicable to another domain and requires a huge initial investment of time and effort. The latter approach has the advantage that it makes it possible to choose the best models to suit the project requirements but it also requires each individual model to implement techniques for adjusting and manipulating its own data to the requirements of the model requesting it, resulting in duplication of effort.
Coupling of models is also not just a simple case of taking two or more models and starting to exchange numbers between them. As a general case, coupled models differ in various aspects such as the method of solution of the partial differential equations (finite difference, finite element and finite volume models), the grids that these models use, their space and time discretisation and so on. The OpenMI standard makes it possible for models to exchange values amongst themselves when the type, location and time-step of the said value are known. Any model that implements the OpenMI standard has to provide, if possible, data that is requested by another model coupled through this interface. In order to do this it might have to do some sort of manipulation, interpolation, transformation or other such operations on its own data. Such quantities involved in an exchange through the OpenMI interface enclose either scalars or vector values in three-dimensional space along with the information about their units. This is an improvement over the exchange of simple numerical values but still these "quantities" are immutable once they are created and it isn't possible to perform any sort of operations or adjustments on them. In other words, these quantities cannot adapt themselves to the requirements of the model that is going to use them.
The present study proposes a method to perform coupling of research oriented models by making use of "smart" tensor objects. To use the object-oriented terminology, tensor objects are composed of information in the form of state variables and functionality by way of methods. This structure enables them to have both a state and behaviour allowing operations to be performed on them for transforming them according to the model requirements. Because of this nature of the tensor objects, a model making use of such objects doesn't have to worry about the source of the tensor objects it is going to use nor about adjusting the format of its own information to the needs of the model that requests it. It can just use or produce tensor objects and the tensor objects are capable of handling these operations by themselves.

TENSOR OBJECTS IN THE "GROSSHANG" PROJECT: TURTLE

An information model based on Tensor objects for physical state variables as described above is the base of a flexible web-based information system with the name Turtle (Molkenthin 2006). The system acts as the central information base with which different information and simulation models will be able to communicate and exchange information using the model interaction scheme. Turtle receives data from a variety of sources such as the data collected from instruments in the field, results of laboratory experiments or numerical simulations etc. All this data is in a variety of formats and on different time and space scales and is classified by making use of metadata. Such metadata includes the original source of the data, the geographic location, the time-window, units of the data, information about the missing values, the maximum and minimum values of the different physical state variables and so on. The original raw data in the form of files is itself stored its original format in the file base. Turtle also parses these files according to their format and transforms their data into tensor objects, which form the tensor base. It also extracts and stores the metadata from all the data that it receives.

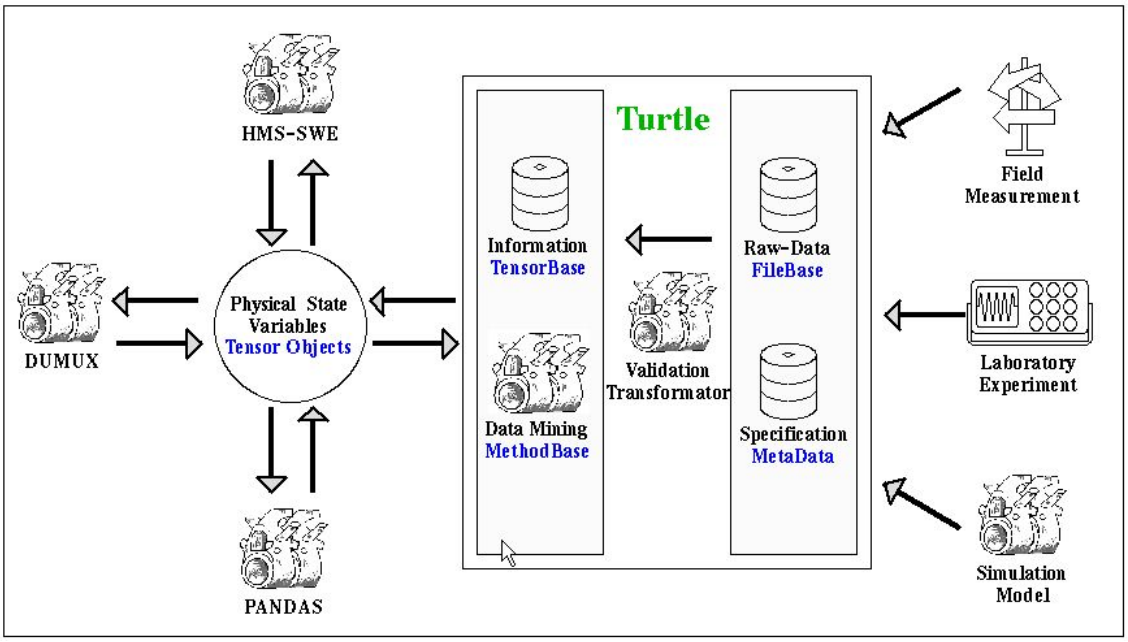

Figure 2: Model integration and coupling in the Grosshang project

The tensor base consists of tensors for a predefined project time-window and spatial region with related time and space scale for all the physical state variables. A tensor in the tensor base might hold for example the readings from an instrument in the field measuring the water level in a creek at regular intervals. This data is easily transformed into different time scales by the tensor itself for example one can request data at time-step resolutions of 3 600s, 21 600s (6h), 86 400s (1 day) and 604 800s (1 week). A tensor can also be asked to provide data at a particular time or at a particular location and it is able to interpolate the result and provide it to the requesting model.

HANDLING INFORMATION: THE MAKEUP OF TENSOR OBJECTS

Tensor objects are information units that are capable of representing any kind of physical

values. The basic building blocks of tensor objects are coordinates, units, coordinate systems, physical values and the topological information between these values.

Coordinates

A coordinate in a tensor object is the representation of a mathematical coordinate. A coordinate has a unit and an index. A set of coordinates will define a coordinate system and all the coordinates in a coordinate system need not have the same unit.

Units

Units are the units used to describe the physical quantities. From the dimensional analysis point of view the units have dimensions and they can be asked for a conversion factor and an offset to SI units to convert them to their SI equivalent (OpenMI). Using this as an intermediary step a value with a unit can be converted to any other equivalent unit.

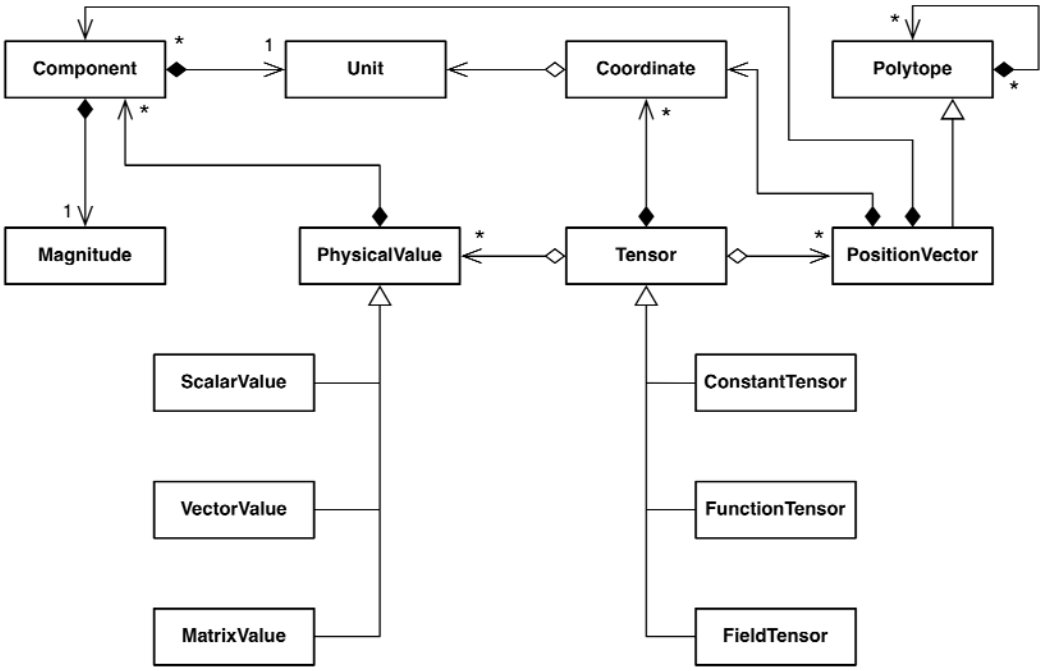

Figure 3: Simplified UML diagram of a tensor object and its constituents

Components

The components making up the physical state variables are the simplest objects that are able to describe some physical value. They are composed merely of a magnitude and a unit. Since components have a unit they are aware of their dimensions, when talking in terms of dimensional analysis, which makes it possible to perform arithmetic operations on the components. Thus one can add or subtract two components that are dimensionally the same. Since the components are aware of their units it is possible to add or subtract values that describe the same physical state variable but have different units e.g. operations on distances in

metres, kilometres or miles. Multiplication and division are possible on components even if they have different dimensions e.g. by multiplying area with velocity, one can arrive at discharge. These simple arithmetic operations make it possible to do more complex mathematical operations at higher levels like calculating the dot or cross product of vectors, integration or differentiation of a function and so on. Since the components only perform operations asked of them the physical interpretation of these operations is up to their end user.

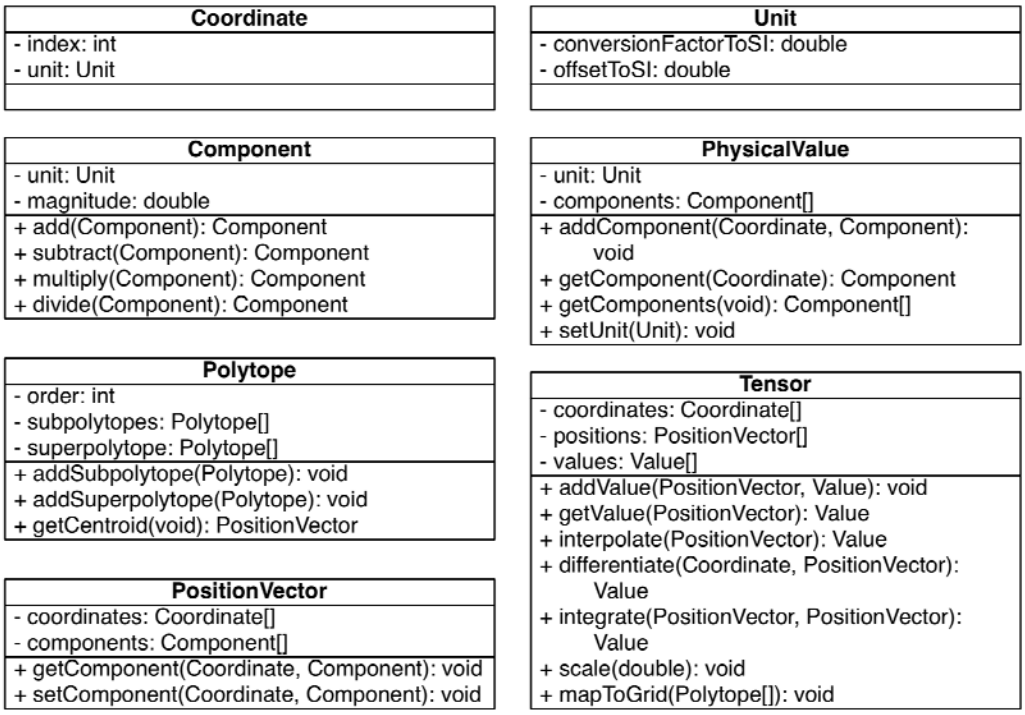

Figure 4: Some of the functionality of tensor objects and their constituents

Physical Values

The physical values are composed of components described above and can be of different kinds. According to their order they can be:

- Scalars 0 th tensor rank
- Vectors 1 st tensor rank
- Matrices 2 nd tensor rank
- … n^{th} tensor rank

The rank of the physical values can be said to be the number of coordinates that are required to reference any of its components. These coordinates may be totally different from the global coordinate system e.g. only one coordinate (time) is required to describe a velocity time-series but each velocity term can have up to three components along the three space coordinates. Thus for a vector, the components are referenced using one coordinate and for a matrix two coordinates are required to pinpoint a component and so on. Although a scalar doesn't have

any components along the coordinate axes, for the sake of concept it can be thought of being made up of a single component that is referenced by an empty set of coordinate axes. In this way we are able to have a generalised definition of any physical value being an object composed of an n^{th} order set of components having units that are dimensionally similar according to dimensional analysis.

As the physical "values" are composed of components they too inherently have a unit and are able to perform operations involving units such as dimensional analysis, conversion of units etc. by delegating these responsibilities to their components. To explain by way of an example, a Scalar temperature of 0°C has one component independent of the coordinate system in which it exists. This component has a magnitude of 0 and units of °C. The value of 0°C can easily be converted into 273.15 K or 32°F by changing this component. Similarly a reading of velocity in 2-dimensional Cartesian coordinate system having v_x = 4m/s and v_y = 3m/s has two components: 4m/s along a *X*-coordinate and 3m/s along a *Y*-coordinate.

Polytopes

As mentioned earlier, tensor objects contain not just physical state variables but also the topological information about these physical state variables. One way of doing this is by making use of polytopes (Weisstein, Wikipedia). Using polytopes it is possible to describe any kind of geometrical object in n-dimensional space. The polytopes are described in terms of their dimensionality in the coordinate system. Thus a point is a 0-polytope, a line-segment and a polyline are 1-polytopes, polygons are 2-polytopes, polyhedra are 3-polytopes and so on. It is possible to describe topology by designing polytopes in such a way that they contain the complete information about their hierarchical structure i.e. an *n*-polytope contains references to all the $(n-1)$-polytopes that it is composed of and also all the $(n+1)$-polytopes that it is a constituent of. It thus forms a tree-like structure with all the topological information being implicitly contained in and easily accessible by walking this tree.

Position Vectors

0-polytopes can be extended so that they also function as position vectors by storing the coordinates of the 0-polytope in the form of components along the coordinates of its coordinates system. In this way it is also possible to describe the geometry of all the polytopes whose 0-polytopes are position vectors and to compute their geometric properties such as the centroid, length or volume.

Tensor objects

Finally we can describe a tensor as a set of values along with their associated positions and topological relationships. Based on the dependence of the values in a tensor objects on the external coordinates, a tensor object can be a:

- Constant Independent of the coordinates
- Function Dependent on one coordinate
- Field Dependent on two coordinates
- … and so on

To explain once again with the help of examples, a tensor object may be used to store the observations made in field about the variation of the water level in a stream. For this purpose it

is possible to set up a 1-dimensional system with a time coordinate. Since these readings constitute a time-series, this tensor would be a function. Each reading will have its position described in the form of a position vector along the solitary time axis. The value itself is a scalar and has therefore just one component i.e. the water level along with the appropriate units. Since this tensor is a function of scalar values it can be described as a scalar-function tensor. Another example would be the permeability distribution that doesn't change with time in a 2-dimensional groundwater model. In this case we need just two coordinates to store the permeability values. Each permeability value will have four components: κ_{xx}, κ_{xy}, κ_{yx} and κ_{yy}. Each value is thus a matrix value and can be referenced by two coordinates. The distribution of values is also in two dimensions and this makes this tensor a matrix-field tensor.

Tensor object adaptability

Tensor objects along with their values contain a lot of information such as the coordinate system, the units of the values, their topological (and possibly the geometrical) relations as well. This makes the tensor objects capable of adapting themselves through operations such as:

- Interpolation of values at any location
- Differentiation
- Integration
- Scaling (upscaling and downscaling)

As an example to demonstrate the functionality achievable through the use of tensor objects consider a tensor with the distribution of elevation in two-dimensional space i.e. a tensor representing a Digital Elevation Model (DEM). Such a tensor is a scalar-field in two dimensions since the elevation values are scalars and are referenced using two space coordinates. This tensor contains the coordinates that could be two-dimensional Cartesian coordinates, latitude-longitude information or something totally different. The locations of the elevations are described by position vectors and may or may not be regularly distributed. This distribution constitutes the topology information of the tensor object. The elevations themselves have a magnitude and some unit of length. This tensor object might be used as the model domain by more than one numerical model and is capable of adapting itself according to the needs of the model requesting the elevation information. It can thus adapt the resolution of the DEM to coarsen it from the metre scale to a kilometre scale or to refine it down to the centimetre scale, performing interpolation as required. If a model uses a different mesh geometry than the grid stored in the tensor then mapping can be performed from this internal grid to the grid of the requesting model e.g. it is possible to map the data from a regular rectangular grid of a DEM to a grid using Voronoi triangles by making use of interpolation techniques. All these transformations of scaling, mapping and interpolation are carried out within the tensor objects themselves by making some object-oriented method calls and the models using the tensors need not care how this happens or how to implement these techniques themselves.

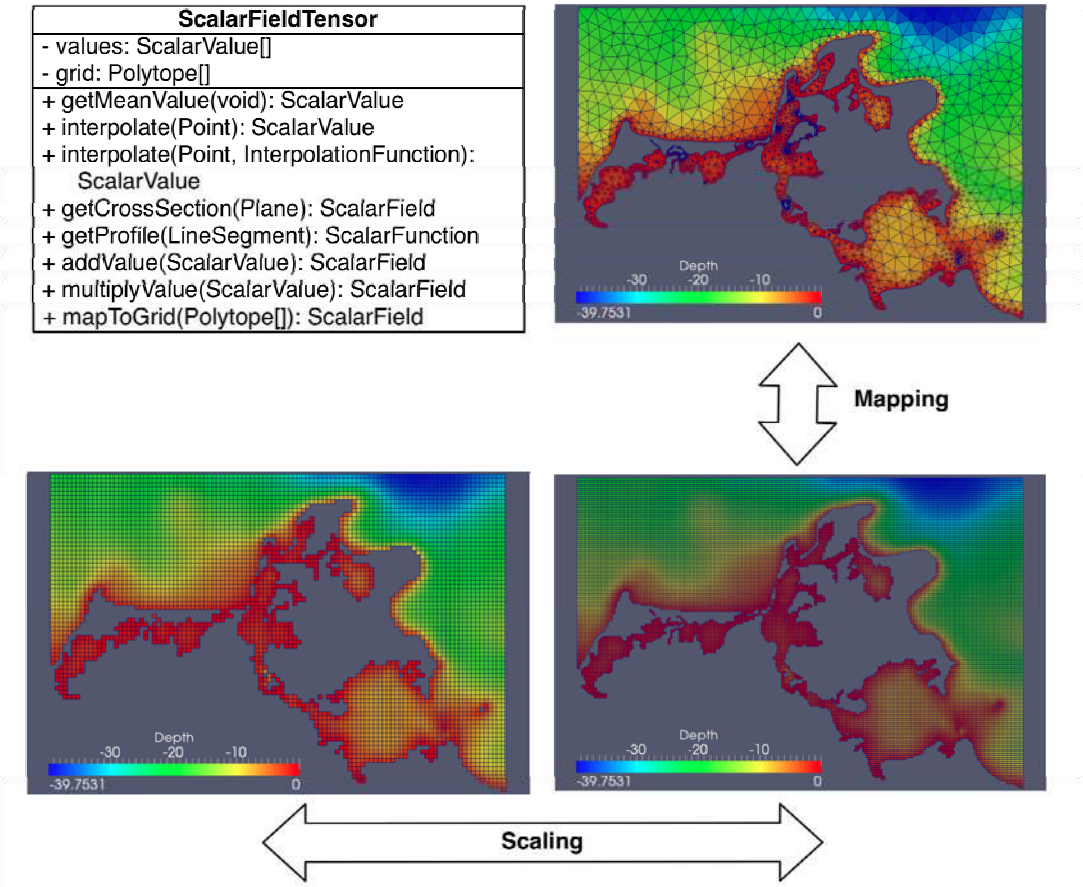

Figure 5: A two-dimensional scalar-field tensor

TENSOR OBJECTS IN MODEL COUPLING

As we have described above, due to the composition of the tensor objects they are self-sufficient and independent from the models that might have created them. In addition to this there is a lot of functionality built into them in the form of object-oriented methods and by this virtue they are versatile and easy to modify to suit the needs of the model that is going to make use of them. Due to these factors tensor objects can be useful tools for the exchange of information during the coupling of models in hydro-environmental engineering projects such as the Grosshang project. (see Figure 2) Each of the models has to specify its requirement from other models (initial conditions, boundary conditions…) in a neutral model interaction scheme. This model interaction scheme is under development in an ongoing PhD work. It is based on the tensor object model approach and implemented as XML scheme. Using existing interfaces (data formats or API interfaces) model wrappers are integrating each model in the tensor based coupling framework as described in this paper.

This method of information exchange is an advancement over the OpenMI interface in that it

allows for the exchange of not just simple numerical values but complex objects that are dynamic and can adapt themselves to the needs of the model that will use them. This makes it easier to couple models since the individual models don't have to implement routines to modify, adjust, interpolate or extrapolate their own numerical data to suit the requirements of the model with which they are coupled because the tensor objects take care of these tasks. Also making use of techniques such as XML-RPC it should be possible to exchange tensor objects remotely making it possible to couple models over the Internet as well.
Finally, the use of tensor objects isn't just restricted to numerical models. They might be used in any other "model" that knows about their structure and functionality. It should therefore be possible to provide initial or boundary conditions to a model from data from field measurements or to use the results of a numerical simulation in statistical analysis, and so on.

CONCLUSION

The described concept is next natural step in the coupling of numerical models from a hydroinformatics point of view. It makes use of existing technologies and extends them to enable the exchange of information among models. The information units used for information exchange between models are quite versatile and can handle many functions relevant to the coupling of models leaving these models free to use the data it requires and provide the data that it needs without worrying too much about its format. This is especially useful in projects, which are multidisciplinary in nature such as the Grosshang project.
The next steps in the current research work are to carry out the coupling of existing models such as the HMS (Busse 2007) developed at TU Berlin with another model applied in the research project and commercial products such as Mike21 through the exchange of tensor objects.

REFERENCES

Busse, Molkenthin, Hinkelmann (2007) A Software Concept of a Numerical Modelling System for an Adaptive Simulation of Coupled Hydrodynamic processes, Forum Bauinformatik, Verlag der Technischen Universität Graz. ISBN 987-3-902465-86-3

DFG-Research Unit FG 581 "Grosshang". http://www.grosshang.de/

Hinkelmann, Zehe (2006): Kopplung von Strömungs- und Transportprozessen für die Modellierung von Großhangbewegungen. Hydrologie und Wasserbewirtschaftung, Heft 1, pp 51-54.

Molkenthin, Hinkelmann, Lindenmaier & Zehe (2006): Web-based information system for multi-scale physical state variables, Proceedings 7th Hydroinformatics Conference, Nice, France, ISBN 81-903170-4-0, pp 2122-29

OpenMI Association. http://www.openmi.org/

Polytope Wikipedia article. http://en.wikipedia.org/wiki/Polytope

The OpenMI document series. Part C – The org.OpenMI.Standard interface specification. http://www.openmi.org/reloaded/about/documents-publications/C_org.openmi.standard_specification.pdf

Weisstein, Eric W. "Poly tope." From Ma thWorld – A Wolfram Web Resource. http://mathworld.wolfram.com/Polytope.html