

Conceptual Approach to Groundwater Modeling – A New Generation of Waterloo Hydrogeologic Software

Serguei Chmakov¹, Petr Sychev², Wayne Hesch¹, Collin Tu¹, Marconi Lima³,

¹Schlumberger Water Services, schmakov@slb.com, whesch@slb.com, xtu@slb.com, Waterloo, ON, Canada,

²Schlumberger Water Services, sychov@dubna.ru, Dubna, Russia,

³Schlumberger Water Services, jlima4@slb.com, Recife, Brazil

Abstract

A new generation of Visual MODFLOW software represents an extendable framework for building a conceptual model independent of any particular numeric model. The ultimate goal is to have a conceptual model that could, without any changes, be translated into the simulator of the modeler's choice such as MODFLOW, FEFLOW, ECLIPSE, analytical models, etc. The model building process is exposed as a natural workflow progressing from input or importing raw data (points, lines, polygons, surfaces) to structural and property modeling in a grid-independent manner. Boundary condition modeling is considered as linking different sub-models to the main model either directly or via OpenMI interface. Using a conceptual model allows for generating a variety of numerical discretizations from the same source, thus improving the quality, credibility, and efficiency of the modeling. The user interface provides a set of flexible 3D- and 2D-visualizers/editors that can be combined with spreadsheet-based data editing to provide the best user experience.

Introduction

The groundwater modeler has to deal with different types of uncertainties, in particular with parameter uncertainties (Hill, 2007, Doherty 2007) and conceptualization uncertainties (Poeter, 2006). In order to handle conceptualization uncertainty, the modeler needs to create a reasonable set of different alternative conceptual models. This in turn, produces a demand for the software giving the modeler a tool for developing such conceptualizations. Currently the cost of developing such alternative models is usually so prohibitive that the majority of the projects can only afford to explore the effects of parameter uncertainties.

The goal of this paper is to present a conceptual approach to groundwater modeling that addresses these issues. This approach allows the modeler to maintain a number of conceptual models within a single project and provides an easy way to generate a number of different numerical models from the same conceptual source.

Workflow Overview

The groundwater model development is inherently very complex and comprises of a number of tasks that requires the hydrogeologist to use a vast variety of tools. One of the main

challenges for the graphical user interfaces and visualization software is to organize the tools and provide an intuitive workflow for the model development from raw data to the numerical model. Sometimes, even though the appropriate tools are available, the modeler is getting lost trying to navigate to the right tool at the right time.

Another challenge – raw data are usually handled outside of the model building workflow. Workflows for creating groundwater meaningful objects from the raw data are usually left beyond the graphical user interfaces for simulation software, which makes it difficult to trace the final model to the original data.

For additional information, contact:

Schlumberger Water Services
460 Phillip Street, Suite 101
Waterloo, ON N2L 5J2 Canada
Tel: +1-519-746-1798
Fax: +1-519-885-5262
Email: sws-services@slb.com
www.water.slb.com

To address these issues, the Conceptual Model Builder (CMB) approach focuses on arranging the building of the model into a natural workflow from Data Processing => Conceptual Model => Numerical Model => Simulation => Analysis of Results (Figure 1). The simulation related part of workflow is taken care of by Visual MODFLOW. When the simulation is done, the results (heads, concentrations, etc.) can be analyzed in Visual MODFLOW or brought back into the CMB for further analysis.

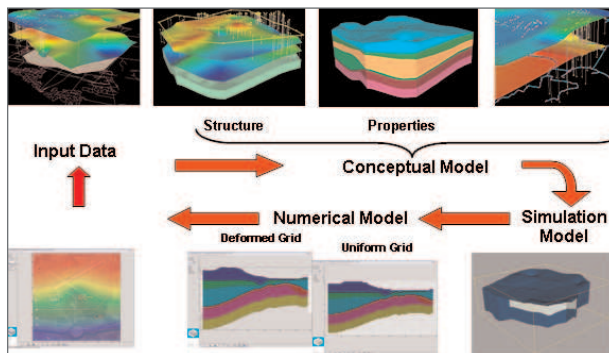


Figure 1. General workflow outline

Data Processing

The Conceptual Model Builder provides a workspace for organizing and management of the raw data. The object complexity ranges from rather simple ones such as points, polylines, polygons and surfaces to as complex as wells (vertical, deviated, or horizontal), or vertical cross-sections.

These objects all share a common characteristic: they are not required to carry any hydrogeological semantics. The objects at this level are considered to be just pure geometry with attached attributes having little or no semantics with respect the modeling goal. This is the way how the CMB allows the modeler to load the raw data in the context of the modeling project and organize them for future use depending on the modeling objectives. At all times the raw data stored in the objects are left intact and kept grid-independent.

All objects can be imported from a variety of data sources such as Digital Elevation Models (DEM), shapefiles, spreadsheets, databases, or created manually. The list of supported formats is open and can be extended as necessary. To facilitate import of geographical information, the CMB supports a number of geographic (NAD27, NAD83, WGS72, WGS84) and projected (UTM NAD27, UTM NAD83, UTM WGS84 North, UTM WGS72

North, SPCS27, SPCS83) coordinate systems and transformations between them. User-defined non-earth coordinate systems are also supported.

Although these objects may not be used directly in the numerical model, they serve as its building blocks. To facilitate this, each object exposes a set of operations to change its own state or generate other objects. Possible examples of operations includes creating surfaces from points using various interpolation methods, spatial transformation like shifting and rotating, converting points to polylines, etc. Operations may include another object's as operands, for instance it is possible to drape a polygon on a surface. To facilitate traceability of the model, each object carries with it a revision history: the creation date and the author of the object, what other data were used, and what operations were applied to it.

The CMB makes use of plug-in based architecture and allows adding more data objects with required functionality as necessary. Adding a new data object does not require recompiling the whole application – they are deployed in the form of .Net assemblies. Each such assembly is accompanied by a manifest that allows the CMB to discover and load them into the project data workspace dynamically. The set of consistent programmatic interfaces facilitates using the objects as the operands for the operations.

Viewing and Editing

Another feature of the Conceptual Model Builder is a set of 2D and 3D viewers and editors. Rather than limit the user to a set of fixed views of the model being developed, the CMB introduces a concept of universal viewers and editors. Any object created in the project workspace can be visualized provided it exposes a set of pre-defined programmatic interfaces. The modeler can display an number of 2D and 3D views and have different objects simultaneously visualized in those views.

The editing process includes geometry editing and attributes editing. Geometry editing can be performed concurrently in a graphical or tabular view thus allowing for increased level of control on the object geometry changes. If the object being edited is simultaneously visualized in other graphical views, all the changes are reflected live in those views during the editing session. To increase usability all editing includes multi-level

undo capabilities.

Attribute editing is based on a selector mechanism that allows the modeler to delineate zones in data objects where attributes of the object's geometrical elements can be specified in a uniform way. Figure 2 shows a simple example, where lines imported from a shapefile were delineated in a number of zones and used for assigning a river boundary condition. Each zone is assigned a particular method for defining attributes.

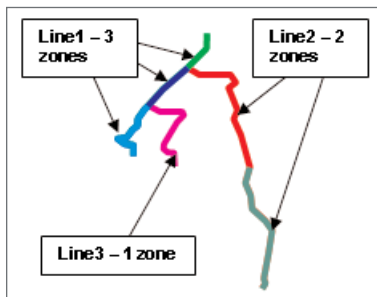


Figure 2. Feature Zonation

There are a number of methods that can be used to define the attributes: constant value, linear interpolation between nodes, values from a surface data object (imported from DEM), values from a set of river gauge stations, etc. Once again, the modeler can make use of data objects loaded into the data workspace – for instance assign river stage from a DEM imported during the course of data management activity. This mechanism is similar to the one used in Visual MODFLOW (Chmakov, 2003) to handle the MODFLOW Stream Routing Package (STR) (Prudic, 1989).

Although the example presented in Figure 2 is one dimensional, the zone based approach for assigning and editing the attributes does not depend on the object dimension, and is similarly used for attributes of 2D and 3D objects. This mechanism is used for defining both property and boundary condition attributes.

Conceptual Model

The Conceptual Model in CMB is represented by a workspace that is separate from that of the raw data. Similar to the data workspace, the content of the conceptual workspace is comprised of objects. The difference between the objects in the conceptual model and the data workspace is that conceptual model workspace objects must have particular semantics and

adhere to business rules specific to the conceptual model objects.

The Conceptual Model (CM) workspace contains a fixed structure of folders for organizing the CM objects. In contrast to the data workspace, there is no option to have arbitrary objects in the conceptual model – the CM structure is fixed and the modeler typically builds CM objects using data objects as building blocks. The modeler can create as many conceptual models as (s)he wants, using objects from data workspace.

Every conceptual model is comprised of three sub-models: Structural Model, Property Model and Boundary Condition Model, each one with its own (fixed) structure. Accordingly, the CM creation workflow includes creating these models in succession.

The Structural Model (SM) consists of a bounding polygon and a set of horizons; the horizons represent the geological structure of the site. Typical SM creation workflow assumes creating horizons from surfaces existing in the data workspace (surfaces can be imported or created from 2D-XY scatter points, cross-section interpretations, or well tops). At this step, the CMB enforces business rules for different types of horizons (base, erosional, conformable, and discontinuous (Figure 3)) by properly modifying the original surfaces used for horizon creation. The volumes in between the horizons constitute the structural zones used later on for property modeling.

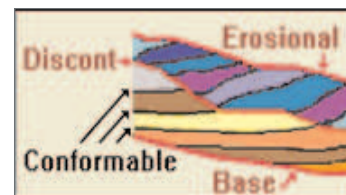


Figure 3. Horizon Types

At the next step the property zones are created using structural zones as the compartmental basis; the modeler has an option to further subdivide the compartment structure provided by the structural model using data objects such as polygons and surfaces in order to create more property zones. Based on these property zones the modeler assigns property attributes required by the modeling objectives.

The most complex part of conceptual modeling is defining the Boundary Conditions (BC). In essence, these are independent models ranging from relatively simple ones that are incorporated in the main simulator (like River BC in MODFLOW) to rather complex models like MIKE 11 (Havnø et al, 1995) providing very detailed model for channel flow.

In both cases, however, the external models share the same geometry (river network) but require a different set of model attributes. In the first case (MODFLOW River BC), it is sufficient to specify just a few attributes; in the second case (MIKE 11) there is a complex workflow defining the model which is linked together through OpenMI interface. On the CMB side, however, specifying MIKE 11 as the BC model of choice is rather simple – the modeler should just specify a path to the .omi manifest of the respective MIKE 11 project (Graham, Chmakov et al., 2006). A simplified linking, where boundary conditions for the numeric model are taken from another model – numerical or analytical – can also be handled this way.

Numerical Model

On the conceptual level a numerical model is represented by its simulation domain. There can be more than one simulation domain for every conceptual model, thus providing a method to generate a regional-local relationship between the numerical models and facilitate scenarios with local grid refinement.

Each simulation domain can have one or more numerical grids attached to it, which defines the horizontal discretization. The grid (mesh), though not linked to the conceptual model, is defined in this workspace only at the time of translating the conceptual model to the numerical model. It is important to emphasize, that in the contrast to the conventional approach, the numerical grid/mesh is not used as the definition domain for the model properties – it only serves as one of the inputs to the process of translating a conceptual model to the numerical one.

If the project objectives change, a new numerical model can be easily generated (Figure 4), or existing ones updated, from the conceptual objects. This way the modeler can explore different modeling scenarios by changing discretizations, boundary conditions or, ultimately, even switch to another main simulator (MODFLOW, FEFLOW, ECLIPSE, analytical models, etc.).

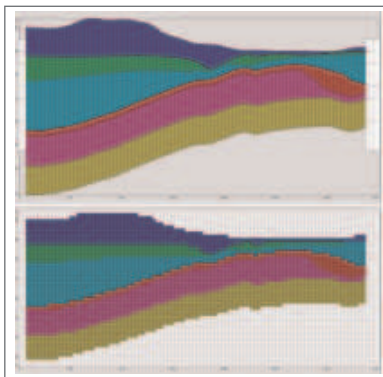


Figure 4. Different grid types translated from the same conceptual model

Data Analysis

Including numerical grid/mesh into conceptual model allows one to establish a link between the conceptual model and the resulting numerical model(s) and simplifies the process of bringing the simulation result back into the Conceptual Model Builder.

Summary and Future Developments

A fully conceptual, simulator-independent approach to ground-water model building is presented. The current implementation, however, provides support only for the MODFLOW engine, and future plans include extending it to other flow and transport engines.

Acknowledgements

The workflow based approach was strongly motivated by the seismic to simulation workflows available in Petrel (<http://www.slb.com/content/services/software/geo/petrel/index.asp?>), developed by Schlumberger. For more information on the OpenMI project, please refer to the extensive OpenMI website at www.openmi.org.

References

- Hill, M., C., Tiedeman, C., R., 2007. Effective Groundwater Model Calibration: With Analysis of Data, Sensitivities, Predictions and Uncertainties, John Wiley and Sons.
- Doherty, J., 2007. Model predictive error: how it arises and how it can be accommodated, Model Care, 493-498
- Poeter, E., 2006. All Models Are Wrong: How Do We Know Which Are Useful?, MODFLOW and More 2006, Proceedings v.1, 11.
- Chmakov, S., Delaney, P., 2003. Grid-Independent, Graph-Based Approach for Supporting the Streamflow-Routing Package in Visual MODFLOW, MODFLOW and More 2003, Proceedings, 144-148.
- Prudic, D.E., 1989. Documentation of a computer program to simulate stream-aquifer relations using a modular, finite-difference, ground-water flow model, U.S. Geological Survey Open-File Report 88-729, 113 p.
- Havnø, K., Madsen, M.N. and Dørgé, J., 1995. MIKE 11 – A generalized river modeling package, Computer Models of Watershed Hydrology, Singh, V.P., Ed., Water Resources Publications, Colorado, USA, (1995), p809-846.
- Graham, D.N., Gijsbers, P.J.A, Gregersen, J.B., 2003. OpenMI: Harmonizing Water Management Through an Open Modelling Interface, MODFLOW and More 2003, Proceedings, 121-125.



Graham, D. N., Chmakov S., Sapozhnikov, A., Gregersen, J. B., 2006. Coupling the MIKE 11 Channel Flow Model to MODFLOW, MODFLOW and More 2006, Proceedings v.2, 727-731.