Eunseon Jang
Johannes Boog
Wenkui He
Thomas Kalbacher

OpenGeoSys Tutorial

Computational
Hydrology III:
OGS#IPhreeqc
Coupled Reactive
Transport Modeling

EXTRAS ONLINE

Springer

SpringerBriefs in Earth System Sciences

Series editors

Eunseon Jang · Johannes Boog
Wenkui He · Thomas Kalbacher

OpenGeoSys Tutorial

Computational Hydrology III: OGS#IPhreeqc
Coupled Reactive Transport Modeling

Eunseon Jang
Department of Environmental Informatics
Helmholtz Centre for Environmental
    Research—UFZ
Leipzig, Sachsen
Germany

Johannes Boog
Department Centre for Environmental
    Biotechnology
Helmholtz Centre for Environmental
    Research—UFZ
Leipzig, Sachsen
Germany

Wenkui He
Dr. Knoell Consult GmbH
Mannheim, Baden-Württemberg
Germany

Thomas Kalbacher
Department of Environmental Informatics
Helmholtz Centre for Environmental
    Research—UFZ
Leipzig, Sachsen
Germany

Preface

This tutorial presents the application of the open-source software *OpenGeoSys* (*OGS*) (Kolditz et al. 2012) with a geochemical solver *PHREEQC* (Palandri and Kharaka 2004) for hydrological simulations concerning reactive transport modeling. This tutorial is a result of the close cooperation within the *OGS* community (www.opengeosys.org), and these voluntary contributions are highly acknowledged.

The book contains general information regarding the reactive transport modeling and a step-by-step setup of models with *OGS* and *PHREEQC*, and related components such as *GINA_OGS*, *ParaView*, and *Data Explorer*. Benchmark examples are presented in detail.

This book is intended primarily for graduate students and applied scientists who deal with reactive transport modeling. It is also a valuable source of information for professional geo-scientists wishing to advance their knowledge in numerical modeling of hydrological processes including reactive transport modeling.

This tutorial is the third in a series that will represent further applications of computational modeling in hydrological sciences.

- Computational Hydrology I: Groundwater flow modeling, Sachse et al. (2015), DOI 10.1007/978-3-319-13335-5, http://www.springer.com/de/book/97833191 33348
- Computational Hydrology II: Groundwater quality modeling, Sachse et al. (2017), DOI 10.1007/978-3-319-52809-0, http://www.springer.com/gp/book/9783319528083
- Computational Hydrology III: OGS#IPhreeqc coupled reactive transport modeling, Jang et al. (2017, this volume)

These contributions are related to a similar publication series in the field of environmental and energy sciences:

- Geoenergy Modeling I: Geothermal Processes in Fractured Porous Media, Böttcher et al. (2016), DOI 10.1007/978-3-319-31335-1, http://www.springer.com/de/book/9783319313337

- Geoenergy Modeling II: Shallow Geothermal Systems, Shao et al. (2016), DOI 10.1007/978-3-319-45057-5, http://www.springer.com/de/book/9783319 450551
- Geoenergy Modeling III: Enhanced Geothermal Systems, Watanabe et al. (2016), DOI 10.1007/978-3-319-46581-4, http://www.springer.com/de/book/ 9783319465791
- Geoenergy Modeling IV: Computational Geotechnics: Storage of Energy Carriers, Nagel et al. (2017), DOI 10.1007/978-3-319-56962-8, http://www. springer.com/gp/book/9783319569604
- Geoenergy Modeling V: Models of Thermochemical Heat Storage, Lehmann et al. (2017*)
- OGS Data Explorer, Rink et al. (2018*),

   (*publication time is approximated).

Leipzig, Germany   Eunseon Jang
August 2017   Johannes Boog
                                                                                                           Wenkui He
                                                                                                  Thomas Kalbacher

References

O. Kolditz, S. Bauer, L. Bilke, N. Böttcher, J.O. Delfs, T. Fischer, U.J. Görke, T. Kalbacher, G. Kosakowski, C.I. McDermott, C.H. Park, F. Radu, K. Rink, H. Shao, H.B. Shao, F. Sun, Y.Y. Sun, A.K. Singh, J. Taron, M. Walther, W. Wang, N. Watanabe, Y. Wu, M. Xie, W. Xu, B. Zehner, OpenGeoSys: an open-source initiative for numerical simulation of thermo-hydro-mechanical/chemical (thm/c) processes in porous media. Environ. Earth Sci. **67**, 589–599 (2012). doi:10.1007/s12665-012-1546-x. https://doi.org/10.1007/s12665-012-1546-x
J.L. Palandri, Y.K. Kharaka, A compilation of rate parameters of watermineral interaction kinetics for application to geochemical modeling. Technical report, 2004

Acknowledgements

We deeply acknowledge the continuous scientific and financial support to the *OpenGeoSys* development activities by the following institutions:

Contents

Chapter 1
Introduction

Thomas Kalbacher

1.1 Reactive Transport Modeling (RTM)

Reactive transport modeling is an important tool for the analysis of complex interactions between physically and hydro-bio-geo-chemically coupled processes in the subsurface. It is an effective method of assessing the relative importance and role of basic processes, which are otherwise treated as isolated, as it provides the opportunity to describe the interaction of competing processes on different spatial and temporal scales. Furthermore, RTM provides an important building block for the connection of scale-encompassing material characterization and is an essential tool for system analysis from the pore to the field scale.

With the help of RTM, it is possible to simulate the elementary matter fluxes in soil and deeper subsurface to study the effects of chemical micro-environments, mechanical-chemical-coupled processes, influences of mineral–liquid reactions, and/or phase transitions. In general, such models usually deal with a combination of flow, mass, and heat transport in single-phase and multiphase systems with reaction path models based on them. However, it should be noted that these models can always provide only a limited description of the multicomponent reaction network and its time dependency. The related uncertainties, even if differently large or small, must never be excluded and should always be discussed detailed.

Multicomponent reactive transport models that treat such a combination of bio-hydro-geo-chemical processes and transport have already been developed in the 1980s (Aagaard and Helgeson, 1982; Kirkner and Reeves, 1988; Lasaga, 1984; Lichtner, 1985; Pruess, 1985; Reed, 1982; Yeh and Tripathi, 1989). A very good overview of all relevant developments of recent decades and current challenges in the area of RTM can be found in Steefel et al. (2015).

From our point of view, the greatest challenge for the development of realistic reactive transport (RT) models in the future is perhaps the integration of all spatial and temporal scales, which are characteristic of the natural process in the subsurface. The spatial scales can range from the molecular scale through pores and fracture scales to

the field-, catchment-, and reservoir scale. In other words, from less than a nanometer to hundreds of kilometers.

In many real-world applications of RT-models, and especially if several chemical processes are effective, it can be necessary to separate relevant processes and structures from less relevant ones. Therefore, we have to deal in some way with the question of physical and chemical heterogeneities additionally. Typically this requires feeding the deterministic simulations with stochastically generated high-resolution data sets that contain both physical and chemical heterogeneities on the basis of geophysical and other field data and assumptions, e.g., Bayer et al. (2015); Fiori et al. (2011); Jang et al. (2017); Li et al. (2007); Savoy et al. (in press). A comprehensive analysis of such systems may require ensemble-runs on high-performance computing (HPC) systems, like a Linux cluster as the use of parallelized RTM codes can reduce the overall simulation times of larger problems tremendously.

1.2   Reactive Nitrogen in the Environment

This book contains several examples dealing with the problem of nitrogen surplus in subsurface systems and in particular with denitrification processes. Data-driven RT-models can support risk assessment and decision-making in order to reduce the losses of nitrogen from agriculture to the environment and to ensure a sufficient yield as well as water protection. Therefore, in this chapter, the role of reactive nitrate in the environment is to be addressed in general. We deliberately omit chemical equations since these are explicitly given later in the examples.

Depending on the compound and concentration, nitrogen can be either a vital nutrient or a pollutant. Nitrogen and its compounds are also very different in the environment. While, for example, atmospheric nitrogen is nearly inert, nitrogen oxides and reduced compounds such as ammonia are reactive.

In the last century, the mankind has massively altered the natural nitrogen cycle, and the amount of reactive nitrogen in the environment has increased dramatically. However, the global distribution is very mixed because, in some parts of the world, the nutrient deficit and the progressive nutrient depletion of soils is a major problem. For the local population, this means that the agricultural crop yields are decreasing (Umweltbundesamt, 2014).

In contrast, many industrialized countries release millions of tons of reactive nitrogen into the environment, which leads to a series of problems such as the loss of aquatic and terrestrial biodiversity, the increased release of greenhouse gases, and the complication of the already difficult drinking water production from groundwater resources. In the recent past, many countries have achieved a significant reduction in nitrogen emissions from industrialized industry, the energy, transport, and wastewater sector. In agriculture, the existing directives, regulations, and measures, if any, do not yet have a lasting and comprehensive effect (Umweltbundesamt, 2014).

In many countries, the most important source of drinking water is groundwater. Reactive nitrogen from agricultural sources or insufficiently treated wastewater can

enter the aquifer systems and cause high levels of nitrate concentrations that deteriorate groundwater quality. Due to the high concentration of nitrate, water suppliers may have to give up groundwater resources and have to access new and deeper groundwater resources. This avoidance strategy leads to additional costs. Even more worrying is the long residence time of such contaminations because groundwater moves slow and reacts very slowly to changes. Several years through decades can pass until these contaminants are detected or reemerge at rivers.

Nitrate is also degraded in the environment, whereby elementary nitrogen and traces of nitrous oxide are formed. This so-called denitrification can take place in surface waters, in soil and in the aquifer as well as in sewage treatment plants. The nitrate decomposition by microorganisms usually occurs under oxygen-free conditions. Organic carbon compounds (heterotrophic reduction) or inorganic compounds such as iron sulfides (autolithotrophic reduction) are suitable as the necessary reaction partners.

In soil and surface waters, the reaction with organic carbon plays the major role. By the decomposition of biomass, organic carbon compounds are regularly produced, but there is rarely a complete degradation, due to other limited reaction conditions (e.g., oxygen content).

In fully saturated aquifer, the contents of organic carbon are generally low, and the degradation capacity is thereby limited. It is often much more effective if denitrification occurs via oxidation of iron sulfides, but it has to be considered that the minerals are gradually used up. The degradation capacity is therefore increasingly exhausted, since a subsequent delivery of these minerals is not possible (Bergmann et al., 2014; Hansen et al., 2011).

1.3 Coupling of Code and Software

In the field of numerical modeling, the term"coupling" is usually understood as the linking of different processes or software modules, as well as a measure that describes the strength of this linkage or the resulting dependency. Only such process couplings enable us to perform thermo-hydro-mechanical-chemical (THMC) process modeling in porous and fractured media using simulation programs such as OGS. The coupling mechanisms of THMC processes using OGS are described by He et al. (2015); Kalbacher et al. (2012); Kolditz et al. (2012). In this tutorial, we focus on software coupling of OGS and PHREEQC (Charlton and Parkhurst, 2011; Parkhurst and Appelo, 2013), which is described more in detail in Chap. 2.

The two main advantages of such a code combinations are obvious. First, new possibilities arise in model applications. Second, additional synergies can arise through the intersection of the expertise of entire users and developers groups, which then may generate new concepts, structures, methods, and procedures.

Frequently, software couplings are realized by formatted files. Data is transported between applications using simple files. All applications involved must understand

the appropriate format. As file formats, tab-separated or comma-separated, XML-based or software-specific file formats are often used.

Another and faster possibility is the use of coupling procedures which access common or shared data structures. In the case of coupling via such procedures, the client system calls a function on the server with which data structures are generated, and data are transferred and integrated. At the same time, this makes it possible to directly start a logic on the server side. In most of such cases, the source code of one or even both tools must be modified. Different techniques and concepts exist to make this possible in different programming languages. In the case of OGS and the PHREEQC, both source codes were written in C++ and thanks to the IPhreeqc interface (Parkhurst and Appelo, 2013), both software projects could be compiled together easily.

One of the major challenges for such coupled software systems is their maintenance because most of the source code projects develop independently on both sides. Usually, such developments are not coordinated across groups, and this makes it difficult couple always the latest releases of both tools.

The individual components of the coupled tools can interact in various ways. In the case of a one-sided coupling, only one code considers the data flow from one system to another. In a case of both-sided coupling, both codes consider parameter changes from the other. In other words, each individual system acts as both a client and a server. In the case of OGS-PHREEQC, the two tools were both-sided coupled, and both modify the concentration of the chemical species alternately.

The types of couplings can be also differentiated and arranged by their strengths. The weaker the coupling, the more independent is the module. A weak coupling of several software components indicates a small degree of dependency. Code changes to individual components can, therefore, be made more easily, because a change has only a local effect. If the tight coupling is given, the code change does not remain local and may require additional adjustments in much deeper code and data structures. The disadvantage of weak couplings is in some cases a lower performance.

References

D.J. Kirkner, H. Reeves, Multicomponent mass transport with homogeneous and heterogeneous chemical reactions: Effect of the chemistry on the choice of numerical algorithm: 1. theory. Water Resour. Res. **24**(10), 1719–1729 (1988). ISSN 1944-7973, http://doi.org/10.1029/WR024i010p01719

P.C. Lichtner, Continuum model for simultaneous chemical reactions and mass transport in hydrothermal systems. Geochimica et Cosmochimica Acta **49**(3), 779–800 (1985). ISSN 0016-7037, http://doi.org/10.1016/0016-7037(85)90172-3

M.H. Reed, Calculation of multicomponent chemical equilibria and reaction processes in systems involving minerals, gases and an aqueous phase. Geochimica et Cosmochimica Acta **46**(4), 513–528 (1982). ISSN 0016-7037, http://doi.org/10.1016/0016-7037(82)90155-7

P. Aagaard, H.C. Helgeson, Thermodynamic and kinetic constraints on reaction rates among minerals and aqueous solutions; i, theoretical considerations. Am. J. Sci. **282**(3), 237–285 (1982), http://doi.org/10.2475/ajs.282.3.237, http://www.ajsonline.org/content/282/3/237.short

A.C. Lasaga, Chemical kinetics of water-rock interactions. J. Geophys. Res.: Solid Earth **89**(B6), 4009–4025 (1984). ISSN 2156-2202, http://doi.org/10.1029/JB089iB06p04009

K. Pruess, A practical method for modeling fluid and heat flow in fractured porous media. Soc. Pet. Eng. J. **25** (1985). ISSN 01, https://doi.org/10.2118/10509-PA

G.T. Yeh, V.S. Tripathi, A critical evaluation of recent developments in hydrogeochemical transport models of reactive multichemical components. Water Resour. Res. **25**(1), 93–108 (1989). ISSN 1944-7973, http://doi.org/10.1029/WR025i001p00093

C.I. Steefel, C.A.J. Appelo, B. Arora, D. Jacques, T. Kalbacher, O. Kolditz, V. Lagneau, P.C. Lichtner, K.U. Mayer, J.C.L. Meeussen, S. Molins, D. Moulton, H. Shao, J. Simunek, N. Spycher, S.B. Yabusaki, G.T. Yeh, Reactive transport codes for subsurface environmental simulation. Comput. Geosci. **19**(3), 445–478 (2015). ISSN 1573-1499, https://doi.org/10.1007/s10596-014-9443-x

A. Fiori, I. Jankovic, G. Dagan, The impact of local diffusion upon mass arrival of a passive solute in transport through three-dimensional highly heterogeneous aquifers. Adv. Water Resour. **34**(12), 1563–1573 (2011). ISSN 0309-1708, https://doi.org/10.1016/j.advwatres.2011.08.010

L. Li, C.A. Peters, M.A. Celia, Effects of mineral spatial distribution on reaction rates in porous media. Water Resour. Res. **43**(1) (2007). ISSN 1944-7973, http://doi.org/10.1029/2005WR004848

P. Bayer, A. Comunian, D. Höyng, G. Mariethoz, High resolution multi-facies realizations of sedimentary reservoir and aquifer analogs. Sci. data **2** (2015), http://doi.org/10.1038/sdata.2015.33

E. Jang, W. He, H. Savoy, P. Dietrich, O. Kolditz, Y. Rubin, C. Schüth, T. Kalbacher, Identifying the influential aquifer heterogeneity factor on nitrate reduction processes by numerical simulation. Adv. Water Resour. **99**, 38–52 (2017), https://doi.org/10.1016/j.advwatres.2016.11.007

H. Savoy, T. Kalbacher, P. Dietrich, Y. Rubin, Geological heterogeneity: Goal-oriented simplification of structure and characterization needs. Adv. Water Resour., in press

Umweltbundesamt. Reaktiver stickstoff in deutschland - ursachen, wirkungen, maSSnahmen (eds geupel, m., frommer j.). umweltbundesamt. Technical report, Umweltbundesamt, Dessau-RoSSlau, Germany, 2014

A. Bergmann, F.-A. Weber, C. Hansen, S. Wilde, L. van Straaten, W. van Berk, S. HäuSSler, P. Dietrich, U. Franko, J. Kiefer, M. Rödelsperger, Konsequenzen nachlassenden nitratabbaus in grundwasserleitern. Energie-, Wasser-Praxis **62**(2), 36–42 (2014)

C. Hansen, A. Bergmann, F.-A. Weber, L. van Straaten, S. Wilde, W. van Berk, S. HäuSSler, M. Rödelsperger, P. Dietrich, U. Franko, Konsequenzen nachlassenden nitratabbaus in grundwasserleitern. Energie-, Wasser-Praxis **59**(10), 44–49 (2011)

W. He, C. Beyer, J.H. Fleckenstein, E. Jang, O. Kolditz, N. Dmitri, T. Kalbacher, A parallelization scheme to simulate reactive transport in the subsurface environment with ogs#iphreeqc 5.5.7-3.1.2. Geosci. Model Dev. **8**(10), 3333–3348 (2015), https://doi.org/10.5194/gmd-8-3333-2015

T. Kalbacher, J.O. Delfs, H. Shao, W. Wang, M. Walther, L. Samaniego, C. Schneider, R. Kumar, A. Musolff, F. Centler, F. Sun, A. Hildebrandt, R. Liedl, D. Borchardt, P. Krebs, The iwas-toolbox: Software coupling for an integrated water resources management. Environ. Earth Sci. **65**(5), 1367–1380 (2012). ISSN 1866-6299, https://doi.org/10.1007/s12665-011-1270-y

O. Kolditz, S. Bauer, L. Bilke, N. Böttcher, J.O. Delfs, T. Fischer, U.J. Görke, T. Kalbacher, G. Kosakowski, C.I. McDermott, C.H. Park, F. Radu, K. Rink, H. Shao, H.B. Shao, F. Sun, Y.Y. Sun, A.K. Singh, J. Taron, M. Walther, W. Wang, N. Watanabe, Y. Wu, M. Xie, W. Xu, B. Zehner, OpenGeoSys: an open-source initiative for numerical simulation of thermo-hydro-mechanical/chemical (thm/c) processes in porous media. Environ. Earth Sci. **67**, 589–599 (2012), https://doi.org/10.1007/s12665-012-1546-x

S.R. Charlton, D.L. Parkhurst, Modules based on the geochemical model phreeqc for use in scripting and programming languages. Comput. Geosci **37**(10), 1653–1663 (2011). ISSN 0098-3004, https://doi.org/10.1016/j.cageo.2011.02.005

D.L. Parkhurst, C.A.J. Appelo. Description of input and examples for PHREEQC version 3—A computer program for speciation, batch-reaction, one-dimensional transport, and inverse geo-

chemical calculations: U.S. Geological Survey Techniques and Methods (2013), https://pubs.usgs.gov/tm/06/a43/

Chapter 2
Methods

Eunseon Jang and Wenkui He

To simulate numerical reactive transport problems, *OpenGeoSys(OGS)*, a scientific open-source software, is coupled to the *IPhreeqc* module of the geochemical solver *PHREEQC(PQC)*. This new coupling scheme (hereinafter referred to as "*OGS#IPhreeqc*") is capable of simulating various chemical reactions alongside different processes such as water flow and solute mass transport. After a brief description of both codes, *OGS* (Sect. 2.1) and *PQC* (Sect. 2.2), the coupling interface *OGS#IPhreeqc* will be introduced (Sect. 2.3), verified on the basis of an benchmark example (Chap. 5), and applied to two different reactive transport problems (Chaps. 6 and 7).

2.1 OpenGeoSys (OGS)

OGS is a scientific open-source initiative based on the Galerkin finite element method (FEM), and is able to simulate multidimensional THMC (thermo-hydro-mechanical-chemical) coupled processes in porous and fractured media (Kolditz et al., 2012). This code has been applied in environmental science, such as in the fields of contaminant hydrology, water resources management, waste deposition, geothermal systems, and energy storage. *OGS* also has been participating several international benchmarking initiatives, such as DECOVALEX (with applications mainly in radioactive waste repositories), CO2BENCH(CO_2 storage and sequestration), HM-Intercomp (coupled hydrosystems), and SeSBENCH (reactive transport processes). A large number of benchmarks have been developed for the source code and algorithm verification over the time.

  OGS is based on an object-oriented (C++) FEM concept to provide a flexible numerical framework (Fig. 2.1). It gives various possibilities to simulate a broad range of different processes which include flow process such as groundwater flow, overland flow, density-driven flow, unsaturated flow, and two-phase as well as multiphase flow processes. Picard and Newton–Raphson schemes can be applied to solve nonlinear problems such as Richards flow and density-dependent flow processes.
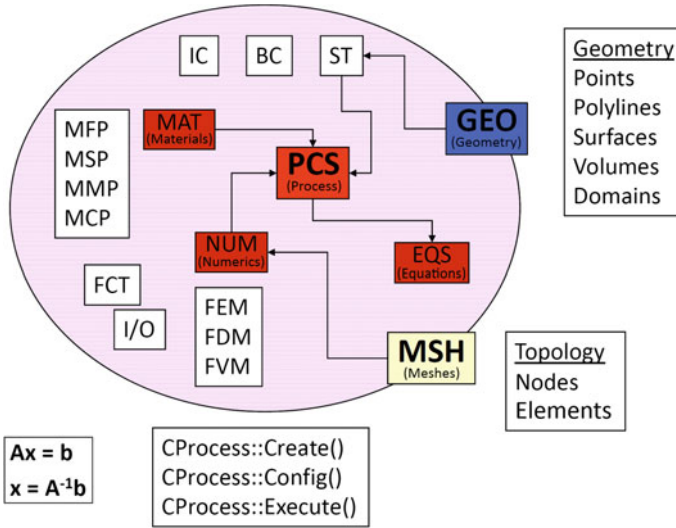
Fig. 2.1 Object-oriented structure of *OpenGeoSys* Version 5 (Kolditz et al., 2012)

Mass transport in the fluid phase is calculated based on the advection–dispersion equation (ADE). For the simulation of flow and transport processes, both implicit and explicit time discretization schemes can be applied. To couple different processes, such as flow, mass, and heat transport, either the monolithic or staggered approach can be used (Wang et al., 2011). *OGS* also can handle a random walk particle tracking (RWPT) methods for Euler–Lagrange simulations (Park et al., 2008). To increase the computational efficiency, the *OGS* code has been parallelized (Wang et al., 2009). This helps to deal with computationally intensive tests in the modeling of complex problems such as the present 3D model of the Nankou area in China (Sun et al., 2011) and nitrate reduction processes under coupled physical and chemical aquifer heterogeneity (Jang et al., 2017).

There are two possible ways to simulate reactive transport modeling (RTM) with *OGS*. One is to use its internal KinReact modules which is able to simulate kinetically controlled biogeochemical reactions (Ballarini et al., 2014). The other way is to couple the external geochemical solvers such as *OGS-PHREEQC* (Xie et al., 2006; He et al., 2015), *OGS-GEMs* (Kosakowski and Watanabe, 2014), *OGS-BRNS* (Centler et al., 2010), and *OGS-ChemApp* (Beyer et al., 2012; Li et al., 2014).

More detailed information regarding *OGS* developments, code resources, and benchmarks can be found at http://www.opengeosys.org/.

2.2 PHREEQC and IPhreeqc Module

PHREEQC is a geochemical solver to simulate a variety of aqueous geochemical calculations in natural waters or laboratory experiments (Parkhurst and Appelo, 1999). It is freely available and one of the most widely used open-source geochemical solvers in the fields of hydrochemistry and chemical modeling (Lucia and Kühn, 2013). *PHREEQC* offers a wide range of equilibrium reactions and kinetic formulations for modeling nonequilibrium mineral dissolution and precipitation, microbial reactions, decomposition of organic compounds, and other kinetic reaction (Charlton and Parkhurst, 2011). *PHREEQC* can simulate batch reaction and one-dimensional (1D) transport calculations with reversible and irreversible reactions. Inverse modeling for interpretation of different processes is also provided (Parkhurst and Appelo, 2013). Several databases provide a variety of possibilities for aqueous speciation calculations, such as Debye–Huckel formations (*phreeqc.dat, wateq4f.dat, llnl.dat, minteq.dat, minteq.v4.dat* and*iso.dat*); the Pitzer specific-ion interaction model (*pitzer.dat*) and the specific-ion interaction theory (SIT) model (*sit.dat*). All databases include temperature dependence of activity coefficient constants and van's Hoff or analytical expressions for equilibrium constants (Parkhurst and Appelo, 1999).

IPhreeqc, where"I" stands for interface, is a C++ *PHREEQC* module designed for coupling *PHREEQC*'s capabilities to other software programs (e.g., multi-dimensional transport simulators) (Charlton and Parkhurst, 2011). There are several works to couple *IPhreeqc* with other codes such as *COMSOL Multiphysics* (Wissmeier and Barry, 2011), *OpenGeoSys* (He et al., 2015; Kolditz et al., 2012), *UTCHEM* (Korrani et al., 2015), and *Matlab* (Muniruzzaman and Rolle, 2016). The interested reader is referred to Charlton and Parkhurst (2011) for the detailed information on *IPhreeqc* and its data manipulation methods.

2.3 OGS#IPhreeqc Coupling Scheme

In this tutorial, we use the coupling scheme developed by He et al. (2015). The nonchemical processes such as flow process and mass transport part are simulated by *OGS*, and *PHREEQC* is then applied to handle the local chemical system in each time step (Fig. 2.2). The coupling scheme is realized at source code level. It means that *IPhreeqc* functions can be accessed directly in the coupling interface. This is the major difference between *OGS#IPhreeqc* and the existing coupling between *OGS* and *PHREEQC*, in which *PHREEQC* is executed externally with a system call (Xie et al., 2006). The Sequential Non-Iterative Approach (SNIA) for Operator Splitting (OS) is applied for the coupling between *OGS* and *IPhreeqc*. *PHREEQC* input data can be prepared as either a file or a character string in the client program (e.g., *OGS*). Both file and character string based approaches are available for data exchange between *OGS* and *IPhreeqc*.
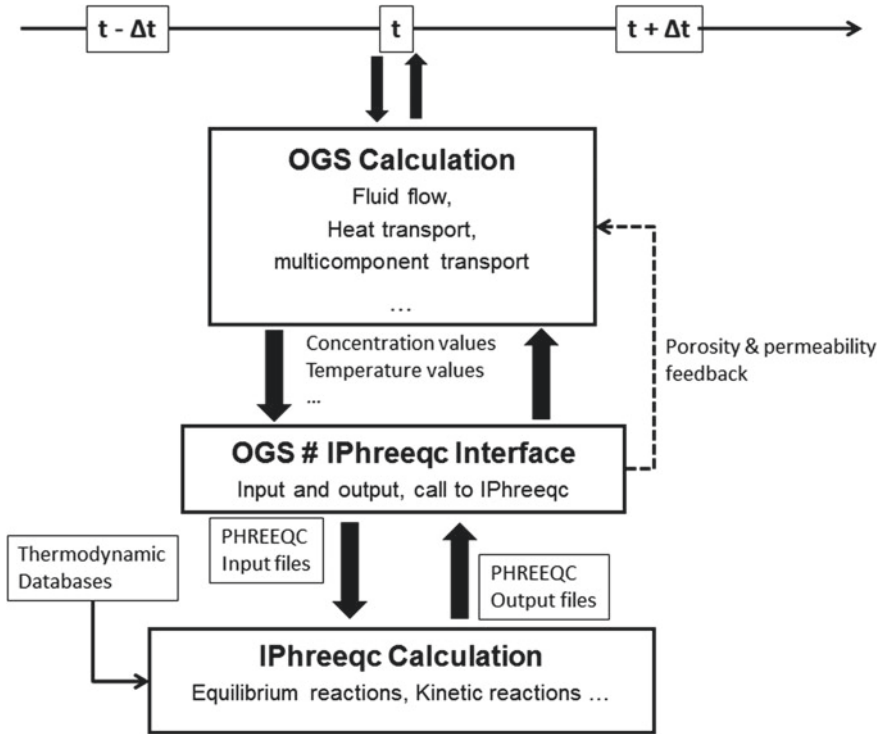
Fig. 2.2 General concept of the coupling between *OpenGeoSys* and *IPhreeqc* (He et al., 2015)

The coupling between *OGS#IPhreeqc*'s main advantages is listed below.

1. Open-source softwares: Both codes are available free of charge for scientific use.
2. Easy to update: The coupling interface itself is version independent and can stay unchanged after any updates. For example, a new release version from *IPhreeqc* is given, it can be integrated efficiently by updating the source code the *IPhreeqc* side. When new *IPhreeqc* files are added (or old files are removed) in the new version, only a reconfiguration of the build system is required. This allows user to benefit continuously from code developments of both sides (He et al., 2015).
3. Capability of simulating a variety of geochemical reactions with dynamic scale/different processes: Coupled with the geochemical code by utilizing the *IPhreeqc* module, thus enabling to perform the variety of geochemical reactions included in the *PHREEQC*'s reaction package. Moreover, multi-dimensional scale can be provided by *OGS*.
4. Computational performance: A significant reduction of the computation time is achieved by using a parallelization scheme based on the message passing interface (MPI). MPI grouping techniques are applied for the parallelization to enable a flexible distribution of different amounts of computer resources for the calculation of two kinds of tasks, i.e., geochemical reactions and the domain decom-
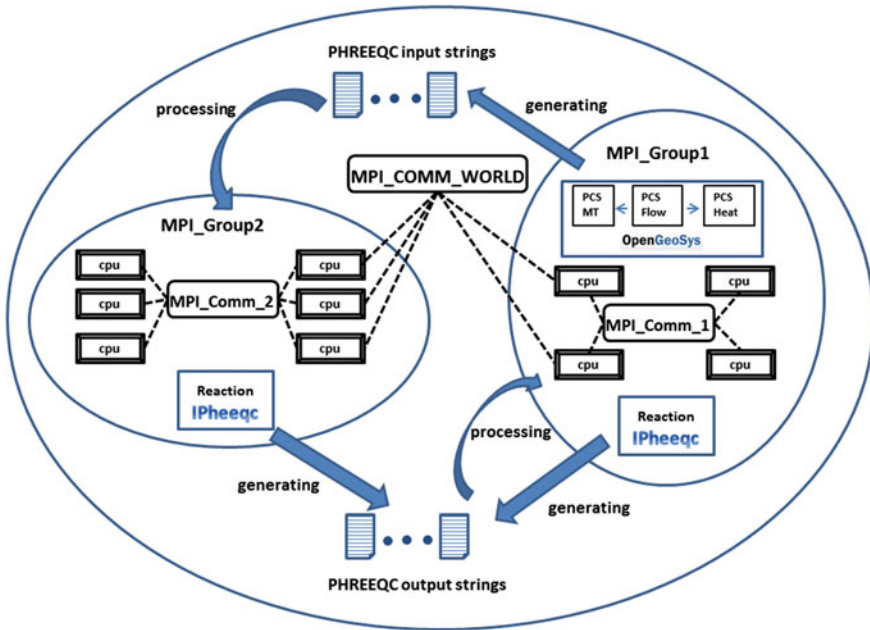
Fig. 2.3 Parallelization scheme for *OGS#IPhreeqc* (He et al., 2015)

position (DDC) related processes. As geochemical reaction is solved locally on each finite element node, its parallelization procedure is independent of the DDC approach implemented in *OGS* (Wang et al., 2009), which is applied to partition the computational tasks of the global assembly and the linear solver for the calculation of processes such as flow and transport. Figure 2.3 represents the general idea of the parallelization scheme. Two different MPI groups, i.e., MPI_Group1 and MPI_Group2, and related intercommunicators are created. The compute cores which belong to MPI_Group1 will be assigned to calculated DDC related processes (groundwater flow, mass, and heat transport) as well as geochemical reactions; whereas those of MPI_Group2 will only take a small part in the calculation of geochemical simulation. By using this method, optimized allocation of the number of compute cores for both types of processes can be realized.

More detailed information about the *OGS#IPhreeqc* interface and benchmarks can be found in (He et al., 2015).

References

O. Kolditz, S. Bauer, L. Bilke, N. Böttcher, J.O. Delfs, T. Fischer, U.J. Görke, T. Kalbacher, G. Kosakowski, C.I. McDermott, C.H. Park, F. Radu, K. Rink, H. Shao, H.B. Shao, F. Sun, Y.Y. Sun, A.K. Singh, J. Taron, M. Walther, W. Wang, N. Watanabe, Y. Wu, M. Xie, W. Xu, B. Zehner, OpenGeoSys: an open-source initiative for numerical simulation of thermo-hydro-mechanical/chemical (thm/c) processes in porous media. Environ. Earth Sci. **67**, 589–599 (2012), http://doi.org/10.1007/s12665-012-1546-x

W. Wang, T. Schnicke, Parallel finite element method and time stepping control for non-isothermal poro-elastic problems. Comput. Mater. Contin. **21**(3), 217 (2011), https://doi.org/10.3970/cmc.2011.021.217, http://www.techscience.com

C.H. Park, C. Beyer, S. Bauer, O. Kolditz, A study of preferential flow in heterogeneous media using random walk particle tracking. Geosci. J. **12**(3), 285–297 (2008), http://doi.org/10.1007/s12303-008-0029-2

W. Wang, G. Kosakowski, O. Kolditz, A parallel finite element scheme for thermo-hydro-mechanical (thm) coupled problems in porous media. Comput. Geosci **35**(8), 1631–1641 (2009). ISSN 0098-3004, http://doi.org/10.1016/j.cageo.2008.07.007

F. Sun, H. Shao, T. Kalbacher, W. Wang, Z. Yang, Z. Huang, O. Kolditz, Groundwater drawdown at nankou site of beijing plain: model development and calibration. Environ. Earth Sci. **64**(5), 1323–1333 (2011). ISSN 1866-6299, https://doi.org/10.1007/s12665-011-0957-4

E. Jang, W. He, H. Savoy, P. Dietrich, O. Kolditz, Y. Rubin, C. Schüth, T. Kalbacher, Identifying the influential aquifer heterogeneity factor on nitrate reduction processes by numerical simulation. Adv. Water Resour. **99**, 38–52 (2017), https://doi.org/10.1016/j.advwatres.2016.11.007

E. Ballarini, C. Beyer, R.D. Bauer, C. Griebler, S. Bauer, Model based evaluation of a contaminant plume development under aerobic and anaerobic conditions in 2d bench-scale tank experi. Biodegrad **25**(3), 351–371 (2014). ISSN 1572-9729, https://doi.org/10.1007/s10532-013-9665-y

M. Xie, S. Bauer, O. Kolditz, T. Nowak, H. Shao, Numerical simulation of reactive processes in an experiment with partially saturated bentonite. J. Contam Hydrol **83**(1—-2), 122–147 (2006). ISSN 0169-7722, https://doi.org/10.1016/j.jconhyd.2005.11.003, http://www.sciencedirect.com/science/article/pii/S0169772205002093

W. He, C. Beyer, J.H. Fleckenstein, E. Jang, O. Kolditz, N. Dmitri, T. Kalbacher, A parallelization scheme to simulate reactive transport in the subsurface environment with ogs#iphreeqc 5.5.7-3.1.2. Geosci. Model Dev. **8**(10), 3333–3348 (2015), https://doi.org/10.5194/gmd-8-3333-2015

G. Kosakowski, N. Watanabe, Opengeosys-gem: A numerical tool for calculating geochemical and porosity changes in saturated and partially saturated media. Phys. Chem. Earth, Parts A/B/C **70**—-**71**, 138–149 (2014). ISSN 1474-7065. Mechanisms and Modelling of Waste-Cement and Cement-Host Rock Interactions, http://dx.doi.org/10.1016/j.pce.2013.11.008

F. Centler, H. Shao, C. De Biase, C. H. Park, P. Régnier, O. Kolditz, M. Thullner. Geosysbrns—a flexible multidimensional reactive transport model for simulating biogeochemical subsurface processes. Comput. Geosci. **36**(3), 397 – 405 (2010). ISSN 0098-3004, http://dx.doi.org/10.1016/j.cageo.2009.06.009

C. Beyer, D. Li, M. De Lucia, M. Kühn, S. Bauer, Modelling co2-induced fluid-rock interactions in the altensalzwedel gas reservoir. part ii: coupled reactive transport simulation. Environ. Earth Sci. **67**(2), 573–588 (2012), https://doi.org/10.1007/s12665-012-1684-1

D. Li, S. Bauer, K. Benisch, B. Graupner, and C. Beyer. Opengeosys-chemapp: a coupled simulator for reactive transport in multiphase systems and application to co2 storage formation in northern germany. Acta Geotech. **9**(1), 67–79 (2014). ISSN 1861-1133, http://doi.org/10.1007/s11440-013-0234-7

D.L. Parkhurst, C.A.J. Appelo. User's guide to phreeqc (version 2) : a computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. Technical report, U.S. DEPARTMENT OF THE INTERIOR U.S. GEOLOGICAL SURVEY (1999), http://pubs.er.usgs.gov/publication/wri994259

M. De Lucia, M. Kühn, Coupling r and phreeqc: Efficient programming of geochemical models. Energy Procedia **40**, 464–471 (2013). ISSN 1876-6102, http://doi.org/10.1016/j.egypro.2013.08.053

S.R. Charlton, D.L. Parkhurst, Modules based on the geochemical model phreeqc for use in scripting and programming languages. Comput. Geosci. **37**(10), 1653–1663 (2011). ISSN 0098-3004, https://doi.org/10.1016/j.cageo.2011.02.005

D.L. Parkhurst, C.A.J. Appelo. Description of input and examples for PHREEQC version 3—A computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations: U.S. Geological Survey Techniques and Methods (2013), https://pubs.usgs.gov/tm/06/a43/

L. Wissmeier, D.A. Barry, Simulation tool for variably saturated flow with comprehensive geochemical reactions in two- and three-dimensional domains. Environ. Model. Softw. **26**(2), 210–218 (2011). ISSN 1364-8152, https://doi.org/10.1016/j.envsoft.2010.07.005, https://www.sciencedirect.com/science/article/pii/S1364815210002094

A.K.N. Korrani, K. Sepehrnoori, M. Delshad, Coupling iphreeqc with utchem to model reactive flow and transport. Comput. Geosci **82**, 152–169 (2015).ISSN 0098-3004, http://dx.doi.org/10.1016/j.cageo.2015.06.004, https://www.sciencedirect.com/science/article/pii/S0098300415001351

M. Muniruzzaman, M. Rolle, Modeling multicomponent ionic transport in groundwater with iphreeqc coupling: Electrostatic interactions and geochemical reactions in homogeneous and heterogeneous domains. Adv. Water Res. **98**, 1–15 (2016). ISSN 0309-1708, https://doi.org/10.1016/j.advwatres.2016.10.013

Chapter 3
Software Requirements and Installation

Eunseon Jang and Johannes Boog

Before staring to simulate with *OGS#IPhreeqc*, an overview about the single software component and software installation will be introduced. Figure 3.1 summarizes the main important constituents. The software *GINA*, *GMSH*, and *OGS Data Explorer* can help to create model geometries and finite elements meshes. *ParaView* is applied for visualization and further post-processing steps.

The first step to start the numerical reactive transport modeling is to download and install all necessary softwares. We use only open access softwares which are easily available for any user from different web sources. Please download the softwares individually from the following sources considering your user system (32-Bit or 64-Bit).

1. *GINA*: To get *GINA*, which is only free for research and teaching, please contact Dr. Herbert Kunz (BGR, Hannover, Germany, e-mail: herbert.kunz@bgr.de) or the *OGS* Team (e-mail: info@opengeosys.org).
2. *OGS*: The latest version of *OGS* is available here: https://docs.opengeosys.org/docs/quickstart/basics/quickstart
3. *ParaView*: *ParaView* can be downloaded from the website for free here: http://www.paraview.org/download/
4. Text editor: Any kinds of text editors can be applied. For example, *Notepad++* can be found here: https://notepad-plus-plus.org/download/v7.4.2.html

For the application, a step by step description of the model setup will be provided in the following steps. The model setup covers several steps, and two different RTM models will be created within the tutorial.

1. Application: Nitrate reduction processes (Chap. 6)

   a. Model geometry (*GINA*)
   b. Subsurface meshes (*GINA*)
   c. *OGS* input files (Text editor)
   d. Kinetic geochemical reaction definition (Text editor)
   e. Simulation (*OGS#IPhreeqc*)
   f. Results and post-processing (*ParaView*)

Fig. 3.1 Workflow for
OGS#IPhreeqc
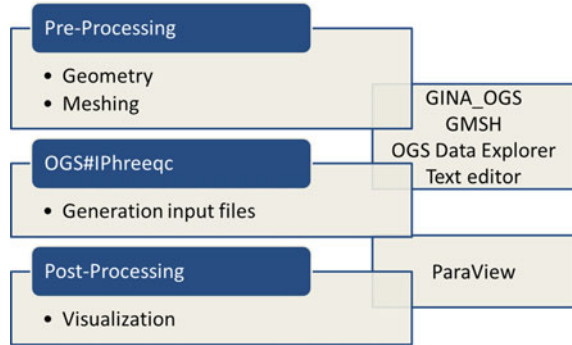
2. Application: Treatment wetlands (Chap. 7)

   a. Model geometry (Text editor)
   b. Subsurface mesh (*GMSH, OGS Data Explorer*)
   c. Simulation conditions (IC, BC, ST etc.) (Text editor)
   d. Kinetic biodegradation reaction definition (Text editor)
   e. Simulation (*OGS#IPhreeqc*)
   f. Results and post-processing (*ParaView*)

3.1 Preprocessing I: GINA

The meshing of complex geometrical domains can be done using a pre- and post-processor *GINA* developed by the German Federal Institute for Geosciences and Natural Resources (BGR). *GINA* is an interactive graphical user interface including geometry and mesh generation. It provides a preparation of the finite element (FE)—calculation and visualization of the calculated data.

Geometrical objects (e.g., points, polylines, surfaces, and volumes) can be defined by using coordinates input and with a mouse click. These geometrical objects can be used to define initial and boundary conditions and view results. *GINA* can generate structured meshes in 1D (line), 2D (quad or triangle) and 3D (prism and hexahedral) elements. For unstructured meshes in 2D and 3D, an interface to the open-source software, such as *GMSH* (described in Sect. 3.2) and TetGen (Si, 2015), is implemented in *GINA*. With the help of these interfaces, a high-quality mesh of 2D and 3D complex structures can be realized.

Table 3.1   Files needed to install *GINA*

Main program	GINA_OGS.exe (version 3.0.9 or higer)
System files	comdlg32.ocx
	msflxgrd.ocx
	richtx32.ocx
	comctl32.ocx
	mscomctl.ocx
	msvbvm60.dll

GINA needs no installation. However, you have to put system files in one directory. If you use Windows 7 (64 bit), please put the system files (listed in Table 3.1) in the following folder: C:\Windows\SysWOW64\. Then, open a command prompt as administrator (you can select this option from right-click menu) and run following command: regsvr32 C:\Windows\SysWOW64\comctl32.ocx.

3.2   Preprocessing II: GMSH and OGS Data Explorer

You can create domain geometries and finite element meshes by using *GMSH* and the *OpenGeoSys Data Explorer*. *GMSH* is used to create the finite element mesh and the *OGS Data Explorer* is employed to convert this mesh into a readable format for *OGS*. *GMSH* is an open-source software tool to develop structured and unstructured finite element meshes up to 3D for various numerical simulations (Geuzaine and Remacle 2009). It also includes capabilities of processing numerical simulation and visualization, however, in this tutorial it will be used as mesh generator only. It can be downloaded from http://gmsh.info by following the installation instructions on the web page. Meshing with *GMSH* is divided in two steps: at 1st you will have to set up the domain geometry, and 2nd, generate the mesh. Setting up the geometry is straight forward and can be done in *GMSH* via point-and-click, or, by writing text files in an external editor; two additional clicks will create the mesh itself (Fig. 3.2). Please consider the *GMSH* help file for further information (http://gmsh.info/doc/texinfo/).

By default, *GMSH* mesh files cannot be read by *OGS* without additional pre-processing. You have to convert it into a format that is suitable for computations in *OGS* by using the *OGS Data Explorer*. The *OGS Data Explorer* is the graphical user interface for *OGS* and includes functionalities such as data transformation capabilities for a variety of formats of third party software, manipulation of input data, visualization of input, and simulation output data. This tutorial will just make use of the data transformation capabilities to preprocess *GMSH* mesh files for later use in *OGS*. A list of support the file formats of the *OGS Data Explorer* is listed in Table 3.2.
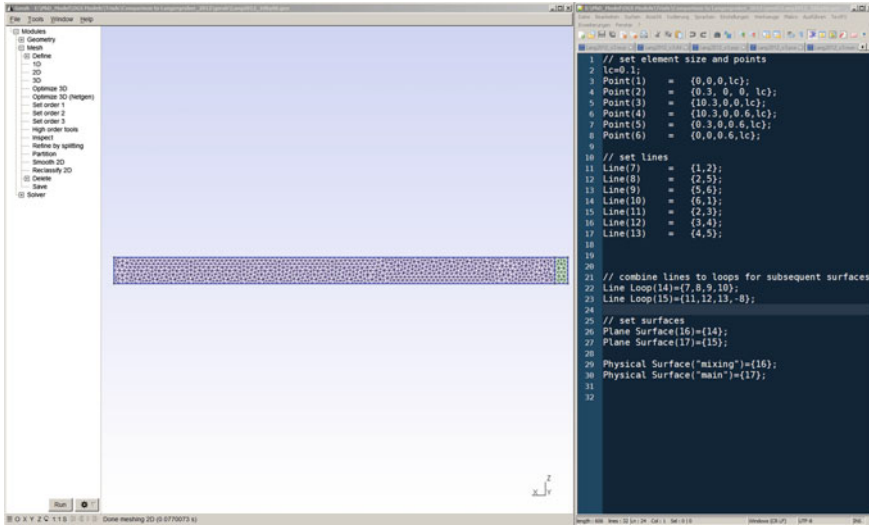
Fig. 3.2 Meshing using *GMSH* and a text editor

Table 3.2 Supported file formats of *OGS Data Explorer*

Data type	Formats/ Programs
Raster data	GeoTIFF, Esri ACII Raster, NetCDF, JPEG, etc.
Features	Esri Shapes, Petrel borehole data, GMS borehole data
Meshes	FEFLOW, GMS, GMSH, TetGen, VTK, etc.
Time series data	CSV, WaterML
Graphics	VTK, OpenSG, Unity, VRML

3.3 Processing: OGS#IPhreeqc

3.3.1 Download the Source Code

You can download *OGS* source code from the official *OGS*-5 repository here: http://github.com/ufz/ogs5. If you use a Git command line to clone the remote repository on GitHub to your PC, the source code can be obtained by typing in the command line prompt. For interested readers, please visit *OGS* Developer Guide http://docs.opengeosys.org/docs/devguide5/getting-started/get-the-source-code. Another approach is to use *TortoiseSVN* on the Windows platform. *TortoiseSVN* can help to
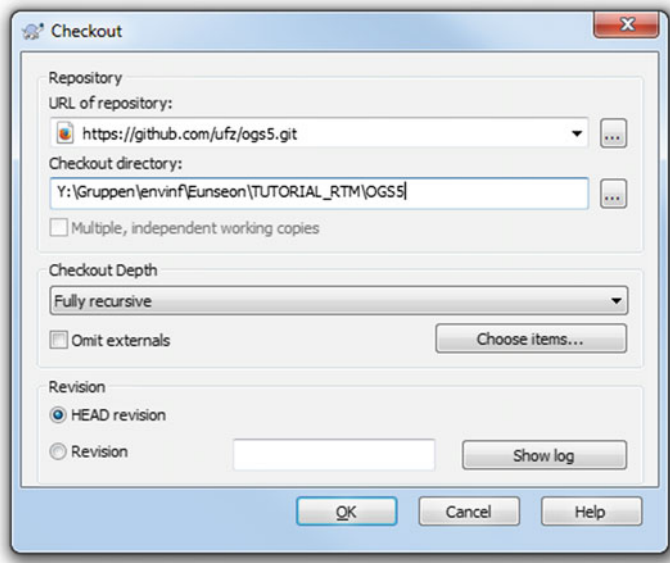
Fig. 3.3 SVN dialog with the location of the repository

manage different versions of the source code, and it is also a free software. After an installation of the *TortoiseSVN*, the following interface will appear (Fig. 3.3). You have to provide a Github link (URL of repository) and your folder (Checkout directory) and then, click OK.

3.3.2 Configure the Building Project

Before the compilation of the source code, *CMake* has to be applied. It helps to generate the configuration and makes files which are specific to the building environment. Here, we are using *CMake* version 3.5.2 as an example. First, a folder where the *OGS* source code is located (source code directory) and a folder where files will be generated (build directory) have to be assigned (Fig. 3.4).

The *CMake* tool runs inside this build directory with a reference to the source code directory of the project and user-chosen options. After clicking on the Configure button, you have to choose your desired configuration option by toggling the corresponding check boxes. Here, for example, we choose the "Visual Studio 2010X86" option. Once the configuration has been done, the build options will be demonstrated in the *CMake* Graphical user interface (GUI). It defaults to FEM. To build the OpenGeoSys code with IPhreeqc, one only needs to choose the IPQC option (Fig. 3.5). Finally, click the Generate button and then, *CMake* will prepare all makefiles in the build folder.
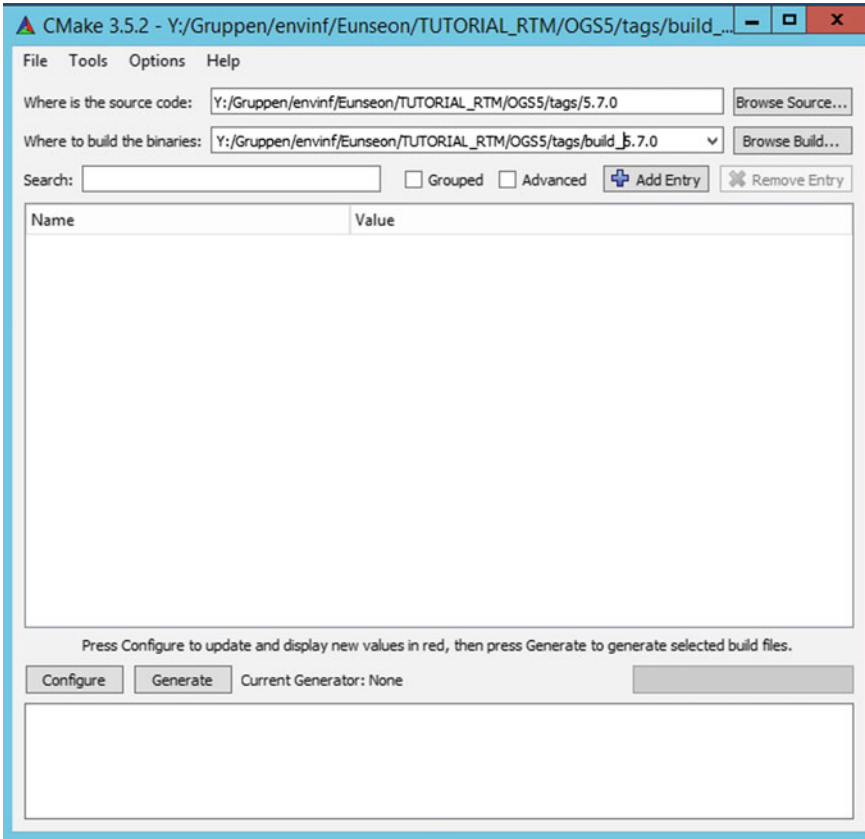
Fig. 3.4 *CMake* interface of configuring the build information

3.3.3   Compile the Code

When the configuration is accomplished by *CMake*, "OGS.sln" file is generated in the build. Open the file either by double-clicking it in the file browser or opening in Visual Studio via File → Open → Project. On the project explorer, right-click on ogs and choose Set as startup project. Then, click BUILD from the file menu, and then click on the first option Build solution (Fig. 3.6). It takes a couple of minutes to run the full building process for the first time. When it is finished, the executable (ogs.exe) will be found under the bin or Debug/Release folder, depending on which building mode was chosen.
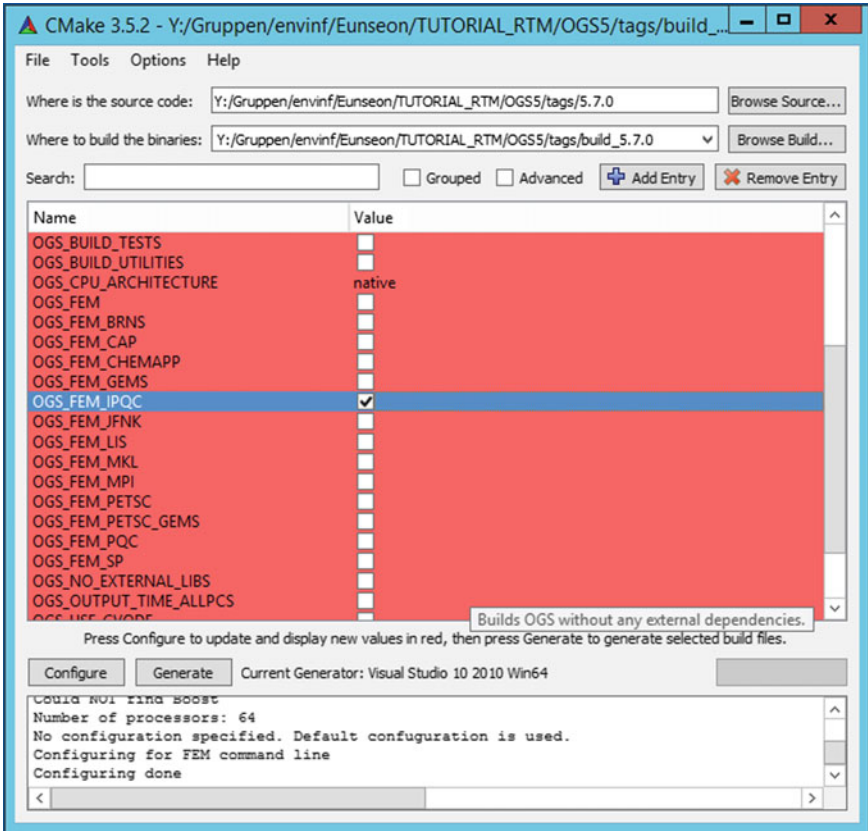
Fig. 3.5 *CMake* interface showing different building options

3.4 Post-processing: ParaView

ParaView is an open-source, multi-platform data analysis, and visualization application tool. *ParaView* provides a graphical user interface (Fig. 3.7) to analyze data using qualitative and quantitative techniques. *ParaView* builds this functionality on parallel and distributed VTK (Visualization Tool Kit) file for visualization and data processing. For more general information on *ParaView*, refer to Ayachit (2015) or Wiki pages http://paraview.org/Wiki/ParaView. The complete source code and installation file can be downloaded from the *ParaView* website: http://www.paraview.org/. They provide binaries for the major platforms; *Linux, Mac OS X*, and *Windows*.

Using *OGS*, simulation results can be written as VTK files (*.vtk) which can be visualized with *ParaView*, *OGS Data Explorer*, or any other VTK viewer. Additionally, *OGS* is capable to output Tecplot-files (*.tec) as well.
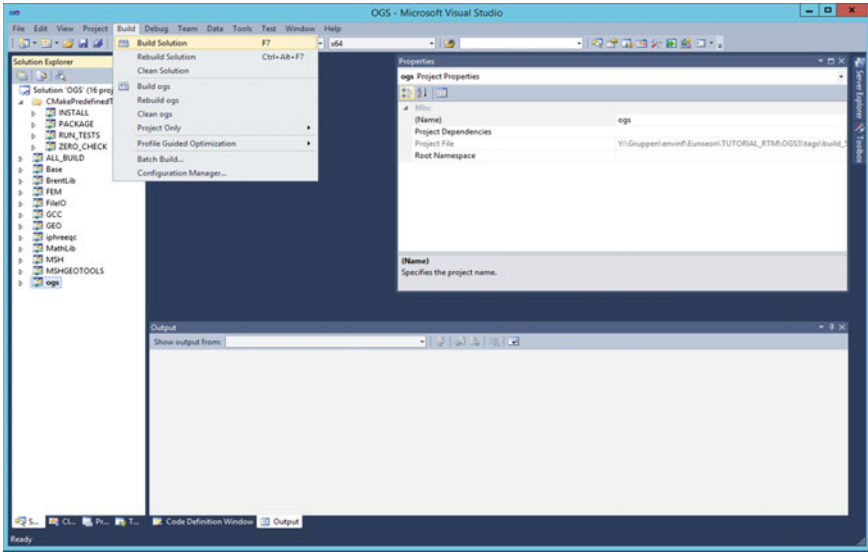
Fig. 3.6 Visual Studio interface after opening the *OGS* solution file
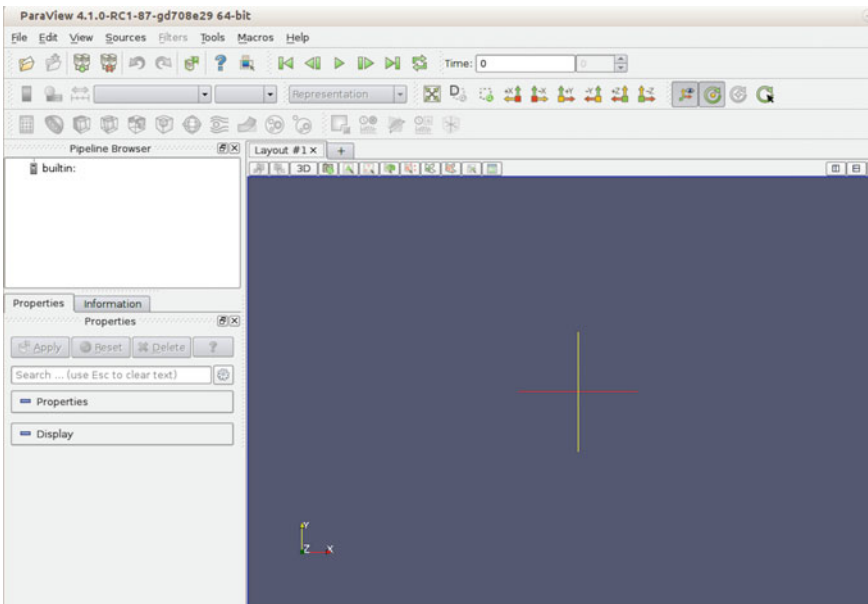
Fig. 3.7 *ParaView* application windows (from Ayachit 2015)

References

H. Si, Tetgen, a delaunay-based quality tetrahedral mesh generator. ACM Trans. Math. Softw. **41**(2), 11:1–11:36 (2015). https://doi.org/10.1145/2629697. ISSN 0098-3500

C. Geuzaine, J.-F. Remacle, Gmsh: a 3-d finite element mesh generator with built-in pre- and post-processing facilities. Int. J. Numer. Methods Eng. **79**(11), 1309–1331 (2009). https://doi.org/10.1002/nme.2579. ISSN 1097-0207

U. Ayachit, The paraview guide: a parallel visualization application. Technical report, 2015

Chapter 4
File Description

Eunseon Jang and Thomas Kalbacher

In this chapter, the structure and content of the input files for setting up reactive transport models are described. First, *OGS* input file description will be presented (Sect. 4.1), and then, a short description of the *PHREEQC* input file will be followed (Sect. 4.2).

4.1 OGS Input Files

Each of input files is responsible for defining a certain aspect of the model (Table 4.1). Note that all *OGS* input files share the same name but with different file ending.

The following is a brief description of each file:

1. PCS—Process definition (*.pcs*): This file is used to specify all simulated processes. The possible processes types are listed at the Chap. A.1 (Appendix A).
2. GLI—Geometry (*.gli*): The geometry file contains a description of the basic geometric objects. The file structure follows a list of ordered points, polygons/polylines, and surfaces. The point, polyline, and surface can be used for different assignment of physical parameters, such as for initial and boundary conditions, source term, and output configuration.
3. MSH—Finite element mesh (*.msh*): The mesh file contains a geometric and topological description of a finite element mesh as well as a spatial and element-related material property distribution. Lines, triangles quadrilaterals, tetrahedral, prisms, and hexahedral elements can be supported. Note that the consecutive numbering has to be completed without any gaps.
4. NUM—Numerical properties (*.num*): The numeric file enables to define and control the numerical solvers for each simulated process.
5. TIM—Time discretization(*.tim*): This file contains a description of time stepping schemes for each simulated process.
6. IC—Initial condition (*.ic*): In the initial condition file, the values of a primary variable (such as the concentration of the species, temperature, hydraulic head,

Table 4.1 *OGS* input files and descriptions

Input files	Explanation
*.pcs	Process definition
*.gli	Geometry
*.msh	Finite element mesh
*.num	Numerical properties
*.tim	Time discretization
*.ic	Initial condition
*.bc	Boundary condition
*.st	Source/sink term
*.mcp	Component properties
*.mfp	Fluid properties
*.mmp	Medium properties
*.msp	Solid properties
*.out	Output configuration

and so on) can be specified at a certain location in the model domain at the starting time of the simulation.

7. BC—Boundary condition(*.bc*): The boundary condition file allows defining boundary conditions by assigning them to geometric objects.
8. ST—Source/Sink term (*.st*): The source/sink term defines the distribution and definition of source and sink terms, as well as Neumann boundary conditions by assigning them to geometric objects.
9. MFP—Fluid properties (*.mfp*): The fluid property file contains a description of one or more fluids.
10. MMP—Medium properties (*.mmp*): This file contains a description of different porous and fractured media. The properties and parameters are assigned to finite elements. The spatial distribution of these materials is given in the mesh file together with the topological description.
11. MSP—Solid properties (*.msp*): This file contains additional solid properties information such as solid density.
12. OUT—Output configuration (*.out*): The output file defines how and when simulation results are written. Usually, the simulation results are written as VTK files (*.vtk*) which can be later visualized with *Paraview, ViSit, OGS-DataExplorer* or any other VTK viewer. Additionally, tecplot files (*.tec*) are also supported.

OGS homepage provides more detailed information and additional literature under https://svn.ufz.de/ogs/wiki/public/doc-auto and http://www.opengeosys.org/project/publications. Also, an *OGS* keyword description is presented in the Appendix A at the end of this book.

Table 4.2 Phreeqc input files and descriptions

Input files	Description
`*.pqc`	*PHREEQC* input definition
`phreeqc.dat`	*PHREEQC* database

4.2   PHREEQC Input Files

To simulate reactive transport modeling coupled with the geochemical code *PHREEQC*, an additional *PHREEQC* input file (**.pqc*) and *PHREEQC* database (*phreeqc.dat*) are required (Table 4.2). The input file specifies the related geochemical reactions (such as equilibrium and kinetic reactions) via keywords and associated data-blocks. The database file contains necessary information (eg., molar mass, stoichiometry, etc.) about chemical elements and substances used in the model computations. A full description of the keywords can be found in the manual of the program by Parkhurst and Appelo (1999).

The `*.pqc` file is based on *PHREEQC* input file. There are some changes before running.

1. `#comp`: At the right-hand side of each component, this command has to be added. The important thing is that all components have to be defined in the component properties (**.mcp*), as well.
2. `#ende`: This keyword has to be added after each *PHREEQC* module (e.g., `SOLUTION`, `EQUILIBRIUM_PHAES`, `RATES`, `PRINT` and so on).
3. There is no need to include the *PHREEQC* transport module since the transport processes will be handled by *OGS*.

By using `SOLUTION`, we can define the chemical species which will be considered for the simulation. Note that all mobile and immobile chemical species (also including pH and pe) have to be defined under `SOLUTION` or with a special ending `#comp`. In the *PHREEQC*, the concentration of each component is defined with `SOLUTION`. In the *OGS#IPhreeqc*, however, the concentration of component has to be defined in the initial and boundary condition files (**.ic* and **.bc*). That means the concentrations assigned in the **.pqc* file does not affect the system. The solution can be equilibrated with mineral assembles together in `EQUILIBRIUM_PHASES`, or the initial concentration of a component can be adapted when defining the `SOLUTION`. Do not forget to add `#ende` after each *PHREEQC* module (Listing. 4.1).

Listing 4.1 Define the SOLUTION and EQUILIBRIUM_PHASES

```
SOLUTION 1
units mmol/kgw        ; default units mmol/kg water
temp 25               ; temperature in degrees Celcius, default = 25C
pH 7 charge #comp 1   ; default pH = 7
pe 4.0  #comp 2       ; pe = -log(electron activity), default = 4
N(5) 3.5E-3  #comp 3     ; nitrogen in the form of nitrate, N(5)
Ca   1.2E-3  #comp 4     ; calcium concentration
[...]
#ende

EQUILIBRIUM_PHASES 1
Gibbsite  0 10  #comp 7 ; Here, SI=0.0 and amount 10 mol
Goethite 0 0    #comp 8
[...]
#ende
```

For kinetic calculations, the keyword KINETICS and RATES are used. In the data block RATES, the mathematical expression of the kinetic reactions is defined via a BASIC interpreter while the parameters controlling the reaction rates set in the data block KINETIC. Furthermore, KINETICS can define the stoichometrics to relate the consumption/production of specific substances or elements. For example, Listing 4.2 shows the kinetic rate of organic carbon, and the overall rate reaction is described as follows:

$$r = k_{02}\frac{m_{02}}{2.94 \cdot 10^{-4} + m_{O2}} + k_{NO3}\frac{m_{NO3}}{1.55 \cdot 10^{-4} + m_{NO3}} + k_{SO4}\frac{m_{SO4}}{1 \cdot 10^{-4} + m_{SO4}} \quad (4.1)$$

where k is the reaction rate constant value ($mol L^{-1}s^{-1}$) with $k_{O2} = 7.5E\text{-}12$, $k_{NO3} = 3.25E\text{-}12$ and $k_{SO4} = 1.5E\text{-}12$.

Listing 4.2 Define the KINETIC reactions

```
KINETICS 1
Organic_C   # comp 9
-tol 1e-8  ; tolerance
-m0 5e-3   ; Initial moles of reactant
-m  5e-3   ; Current moles of reactant
#ende

RATES 1
Organic_C ; Name of rate expression
-start
10   if (m <= 0) then goto 200
20   mO2 = mol("O2")
30   mNO3 = tot("N(5)")
40   mSO4 = tot("S(6)")
50   rate_1 = 7.5E-12*mO2/(2.94E-4 + mO2) + 3.25E-12*mNO3/(1.55E-4 +
     mNO3)
60   rate = rate_1 + 1.5E-12*mSO4/(1.E-4 + mSO4)
70   moles = rate * m * (m/m0) * time
80   if (moles > m) then moles = m
200 save moles
-end ;Note that the hyphen is required to avoid a conflict with the
     keyword END
#ende
END
#STOP
```

When −`selected_output` is set to `true`, `SELECTED_OUTPUT` data block in a subsequent simulation can be applied. By using `SELECTED_OUTPUT` data block, you can print selected entities from the composition of solution, exchange assemblage, pure-phase assemblage, solid solution assemblage, and surface assemblage after the completion of each type of calculation. You can control printing of information to the selected output file using `USER_PUNCH`. Information defined in `USER_PUNCH` will be written to the selected output file. In the Listing 4.3, for example, `SELECTED_OUTPUT` block is set to `true` and the concentration of Na, K and Cl will be written in the `phout_set.dat` file.

For the further information, the interested reader can visit the *PHREEQC* web page: https://wwwbrr.cr.usgs.gov/projects/GWC_coupled/phreeqc/phreeqc3-html/phreeqc3.htm.

Listing 4.3 Define output

```
PRINT
-reset true
-selected_output true
#ende

SELECTED_OUTPUT
-file phout_sel.dat    ; file name where selected results are written.
-selected_output  true
-user_punch true
#ende

USER_PUNCH
-head Na_mmol K_mmol Cl_mmol
10 PUNCH TOT("Na")*1000, TOT("K")*1000, TOT("Cl")*1000
#ende
#STOP
```

Reference

D.L. Parkhurst, C.A.J. Appelo, User's guide to phreeqc (version 2): a computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. Technical report, U.S. Department of the Interior U.S. Geological Survey, 1999, http://pubs.er.usgs.gov/publication/wri994259

Chapter 5
Code Verification: Engesgaard Benchmark

Wenkui He

The *OGS#IPhreeqc* coupling interface was tested and verified using several benchmarks which include degradation of chlorinated hydrocarbons (van Breukelen et al., 2005), mineral precipitation/dissolution (Engesgaard and Kipp 1992) and uranium leaching These benchmarks were described in details in He et al., (2015). In this chapter, the Engesgaard Benchmark (Engesgaard and Kipp 1992) is presented, which simulates the geochemical reactions involved when a solution containing magnesium chloride is flushed through a calcite column. Both kinetic and equilibrium reactions are taken into account in the definition of the geochemical system. The simulation results obtained by using *OGS#IPhreeqc*, *PHREEQC* and *OGS-Chemapp* (Beyer et al., 2012) are compared.

5.1 Benchmark Description

A solution containing magnesium chloride ($MgCl_2$) is injected into a calcite column with a length of 0.5 m (see Fig. 5.1). Calcite dissolves as the solution moves downstream in the column; whereas temporary precipitation of dolomite occurs at the calcite dissolution front. This phenomenon can be described by Eq. 5.1.

$$2CaCO_3 + MgCl_2 \iff CaMg(CO_3)_2 \downarrow + CaCl_2 \tag{5.1}$$
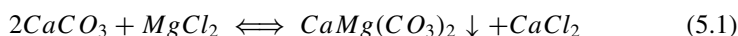
The precipitation of calcite is modeled as equilibrium reaction, whereas the dissolution/precipitation of dolomite is modeled as kinetic reaction using a rate law based on Lasaga et al., (1994) (see Eq. 5.2).

$$r_{dolomite} = KA(1 - SI(\text{``dolomite''})^{\theta})^{\eta} \tag{5.2}$$

where the $r_{dolomite}$ is the reaction rate, K is the overall rate constant, A is the reactive surface area of dolomite, SI("dolomite") is the saturation index of dolomite, θ and η are empirical exponents.

Fig. 5.1 Schematic
representation of the
Engesgaard benchmark

MgCl$_2$ solution
Injection rate: 0.26 m³/d
Concentration: 1 mol/m³

Calcite

0.5 m
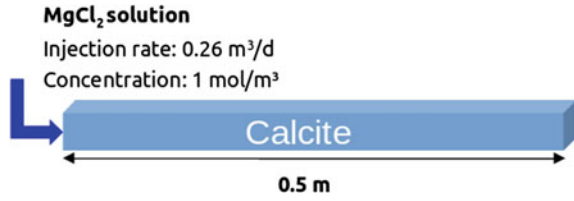
For this benchmark, the overall rate consists of a base mechanism rate constant $K^{neutral}$ and a concentration-dependent contribution from acid mechanism, as shown in Eq. 5.3.

$$K = K^{neutral} + K^{acid}\gamma_{H^+}^{\beta} \tag{5.3}$$

where K^{acid} is the acid mechanism rate constant, γ_{H^+} is the activity coefficient of H^+, and β is a empirical exponent.

The rate constants $K^{neutral}$ and K^{acid} at a certain temperature other than $25°C$ can be calculated based on Arrhenius equation (see Eq. 5.4).

$$K^i = K_{25}^i \exp\left(\frac{-E_a^i}{R}\left[\frac{1}{T} - \frac{1}{298.15}\right]\right) \tag{5.4}$$

5.2 Model Setup

A 1D model domain with a length of 0.5 m is defined and evenly discretized into 100 line elements. The material properties of the porous medium are summarized in Table 5.1, whereas the initial and boundary conditions (upstream boundary) are listed in Table 5.2. The total simulation time of 21333.32 s is equally discretized into 40 time steps.

The geochemical reactions in the benchmark mainly include the carbonate equilibrium reactions and kinetically controlled precipitation/dissipation of dolomite. The thermodynamic data of the default *PHREEQC* database *phreeqc.dat* (Parkhurst

Table 5.1 Material properties of the 1D calcite column (from He et al., (2015))

Parameter	Value	Unit
Effective porosity	0.32	-
Bulk density	1.80×10^3	$kg \cdot m^{-3}$
Longitudinal dispersivity	6.70×10^{-2}	m
Flow rate	3.00×10^{-6}	$m \cdot s^{-1}$
Temperature	298.15	K

Table 5.2 Initial and boundary conditions for mass transport of the Engesgaard benchmark (from He et al., (2015))

Species	Initial conditions	Boundary conditions	Unit
Ca^{2+}	1.23×10^{-1}	1.00×10^{-7}	$mol \cdot m^{-3}$
Mg^{2+}	1.00×10^{-9}	1.00	$mol \cdot m^{-3}$
C(4)	1.23×10^{-1}	1.00×10^{-7}	$mol \cdot m^{-3}$
Cl^-	1.00×10^{-9}	2.00	$mol \cdot m^{-3}$
pH	9.91	7	–
pe	4	4	–
Calcite	5.7412×10^{-2}	–	$mol\ m^{-3}$
Dolomite	0.0	–	$mol\ m^{-3}$

Table 5.3 Definition of the equilibrium reaction system

Reactions	Log K values
$H_2O \leftrightarrow OH^- + H^+$	-12.780
$CO_3^{2-} + H^+ \leftrightarrow HCO_3^-$	10.329
$CO_3^{2-} + 2H^+ \leftrightarrow H_2CO_3$	10.329
$Ca^{2+} + CO_3^{2-} \leftrightarrow CaCO_3$	3.224
$Ca^{2+} + H^+ + CO_3^{2-} \leftrightarrow CaHCO_3^+$	11.435
$Mg^{2+} + CO_3^{2-} \leftrightarrow MgCO_3$	2.980
$Mg^{2+} + H^+ + CO_3^{2-} \leftrightarrow MgHCO_3^+$	11.399
$CaCO_3(s) \leftrightarrow Ca^{2+} + CO_3^{2-}$	-8.480

Table 5.4 Parameters for dolomite kinetics (from Palandri and Kharaka (2004))

Parameter	Value	Unit
A	0.001	$m^2 \cdot kg^{-1}$
θ	1.0	-
η	1.0	-
E_a (neutral)	52200	$J \cdot mol^{-1}$
$\log(K_{25})$ (neutral)	-7.53	$mol \cdot m^2 \cdot s^{-1}$
E_a (acid)	36100	$J \cdot mol^{-1}$
$log(K_{25})$ (acid)	-3.19	$mol \cdot m^2 \cdot s^{-1}$
species (acid)	H^+	-
β	0.5	-

and Appelo 1999) are applied for the carbonate equilibrium reactions (see Table 5.3). The rate parameters from Palandri and Kharaka (2004) are used for the kinetic rate law (see Table 5.4).
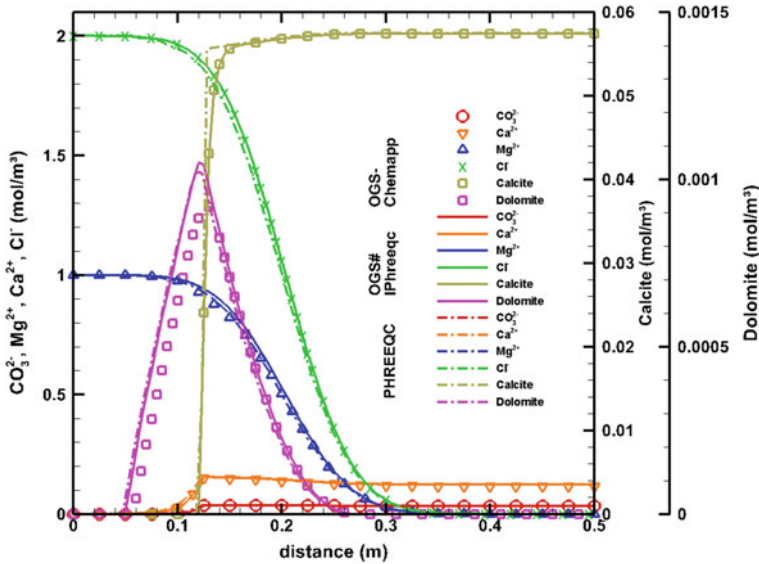
Fig. 5.2 The concentration distribution of aqueous species and minerals simulated by *OGS-Chemapp*, *OGS#IPhreeqc*, and *PHREEQC* (He et al., 2015)

5.3 Simulation Results

This benchmark is simulated by using *OGS#IPhreeqc*, *OGS-Chemapp* (combined with the KinReact module of *OGS* to enable kinetic reactions) and the batch version of *PHREEQC*. The *PHREEQC* script for the simulation is available in the supplementary material of He et al., (2015). Figure 5.2 illustrates the simulated concentration profiles of different aqueous species and minerals at the end of simulation. Generally, good agreements are obtained using the three codes.

References

B.M. van Breukelen, D. Hunkeler, F. Volkering, Quantification of sequential chlorinated ethene degradation by use of a reactive transport model incorporating isotope fractionation. Environ. Sci. Technol. **39**(11), 4189–4197 (2005), https://doi.org/10.1021/es048973c. PMID: 15984799

P. Engesgaard, K.L. Kipp, A geochemical transport model for redox-controlled movement of mineral fronts in groundwater flow systems: a case of nitrate removal by oxidation of pyrite. Water Resour. Res. **28**(10), 2829–2843 (1992), https://doi.org/10.1029/92WR01264

W. He, C. Beyer, J.H. Fleckenstein, E. Jang, O. Kolditz, N. Dmitri, T. Kalbacher, A parallelization scheme to simulate reactive transport in the subsurface environment with ogs#iphreeqc 5.5.7-3.1.2. Geosci. Model. Dev. **8**(10), 3333–3348 (2015), https://doi.org/10.5194/gmd-8-3333-2015

C. Beyer, D. Li, M. De Lucia, M. Kühn, S. Bauer, Modelling CO2-induced fluid-rock interactions in the altensalzwedel gas reservoir. part ii: coupled reactive transport simulation. Environ. Earth Sci. **67**(2), 573–588 (2012), https://doi.org/10.1007/s12665-012-1684-1

A.C. Lasaga, J.M. Soler, J. Ganor, T.E. Burch, K.L. Nagy, Chemical weathering rate laws and global geochemical cycles. Geochimica et Cosmochimica Acta. **58**(10), 2361–2386 (1994). ISSN 0016-7037, https://doi.org/10.1016/0016-7037(94)90016-7

D.L. Parkhurst, C.A.J. Appelo, User's guide to PHREEQC (version 2): a computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. Technical report, U.S. Department of the Interior U.S. Geological Survey (1999), http://pubs.er.usgs.gov/publication/wri994259

J.L. Palandri, Y.K. Kharaka, A compilation of rate parameters of water-mineral interaction kinetics for application to geochemical modeling. Technical report (2004)

Chapter 6
Application: Nitrate Reduction Processes

Eunseon Jang

6.1 Introduction

The coupled reactive transport code ("*OGS#IPhreeqc*") is applied to a pyrite-driven denitrification of nitrate-contaminated groundwater scenario. This nitrate reactive transport model is a simplification of the model involving both autotrophic and heterotrophic denitrification processes based on results from a field study in the Hessian Ried, Germany (Kludt et al., 2016). For the interested reader, more detail information can be found in Jang et al., (2017).

After a brief description of the nitrate reduction processes in the groundwater system (Sect. 6.2), a step-by-step guideline for setting up a 2D RT model will be introduced.

6.2 Background

Nitrate (NO_3^-) contamination in groundwater is a widespread water quality problem and can cause health concerns and environmental impact such as eutrophication in lakes, reservoirs, and rivers (Fan and Steinberg 1996; Smolders et al., 2010). Excessive nitrate can be migrated naturally under anaerobic condition by reduction processes, so-called denitrification, which reduces nitrate to nitrous oxide (N_2O) and dinitrogen (N_2) (Korom 1992; Rivett et al., 2008). The nitrate reduction reaction can be written as a half-equation that describes the role of electron (e^-) transfer as shown in Eq. 6.1.

$$2NO_3^- + 10e^- + 12H^+ \rightarrow 2N_2 + 6H_2O \tag{6.1}$$
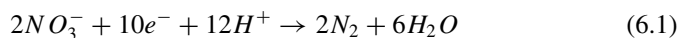
Two important denitrification pathways have been suggested; heterotrophic denitrification by oxidation of organic carbon sources and autotrophic denitrification by oxidation of chemoautotrophic energy sources (e.g., reduced iron, reduced sul-

fur and methane) (Rivett et al., 2008). Heterotrophic denitrification occurs mainly in the shallow unsaturated or saturated zones, where organic source is present due to decaying crop material and leaching from the soil profile (Bailey et al., 2012). Many studies have been reported that the nitrate reduction by heterotrophic denitrification is thermodynamically favored than reduction coupled to chemoautotrophic species(autotrophic denitrification) (Korom 1992). However, the latter pathway can be utilized by denitrifying organisms in the presence of pyrite (FeS_2), as well (Torrentó et al., 2010; Zhang et al., 2013). Under anaerobic condition, when electron acceptors (such as nitrate) are present, pyrite oxidation can be described corresponding to redox reactions.

6.3 Model Components and Governing Equations

6.3.1 Model Scenario

A solution with oxygen, as a primary electron acceptor and nitrate, is entering from the top boundary at y = 0.5 m, representing an oxygenated water source exposed to the model domain (Fig. 6.1). Since the aquifer's major reductant pyrite is assumed to be abundant throughout the model domain, nitrate is reduced to nitrogen by denitrification reaction.

In this simulation, the geochemical reactions between the oxidized recharge water and aquifer's reductant are kinetically controlled (See Sect. 6.3.3). Neither sorption nor volatilization is accounted.

The model domain used for the numerical investigation is a 2D cross-sectional model with 5 m length and 0.5 m depth. The model domain is discretized with 200 elements. The groundwater flow is simulated by *OGS*, and a fully saturated steady-state condition is assumed. The flow direction is from right to left. Two constant hydraulic head boundaries are assigned to the right and left model boundaries, imposing a regional hydraulic gradient of 0.001. A mean hydraulic conductivity of 1.55E-4 $m \cdot s^{-1}$ is assumed. Flow field parameters including bulk density, porosity,
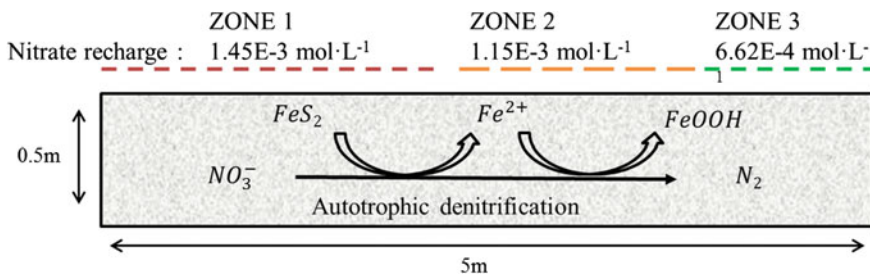
Fig. 6.1 Two-dimensional domain used in the simulation (modified from Jang et al., (2017))

Table 6.1 Summary of aquifer hydrology, geometry, and transport parameters used for simulation (modified from Jang et al., (2017))

Parameter	Value	Unit
Hydraulic conductivity	1.55E-4	$m \cdot s^{-1}$
Porosity	0.34	-
Column bulk density	1.72E+3	$kg \cdot m^{-3}$
Time step	30	hours
Reaction time	3000	hours
Number of elements	200	-

and hydraulic conductivity are listed in Table 6.1. The simulation runs with a time step of 30 hours (1.25 days) to the total simulation time of 3000 hours (125 days).

For each time step, flow processes are calculated first, and mass transport using Advection–Dispersion Equation (ADE) is solved sequentially for each mobile component by *OGS* (see Sect. 6.3.2). After *OGS* calculation is complete, the values of the each variables on all finite element nodes will be updated, and *PHREEQC* is applied to handle local chemical systems.

6.3.2 Groundwater Flow and Solute Transport Equations

Fully saturated groundwater flow based on the water balance and Darcy's law can be described by Eq. 6.2:

$$S_s \cdot \frac{\partial h}{\partial t} = \nabla \cdot (K \nabla h) + q \qquad (6.2)$$

where S_s is the specific storage capacity m^{-1} of the porous medium, ∇h is the hydraulic gradient, t is the time sec, K is the hydraulic conductivity $m \cdot s^{-1}$, and q is a general source/sink term $m \cdot s^{-1}$. The velocity of water moving through a flow field described by Eq. 6.3 is shown here:

$$V = -\frac{K \nabla h}{n} \qquad (6.3)$$

where V is the velocity vector $(m \cdot s^{-1})$ and n is the porosity of the medium (-). The advection–dispersion equation can be expressed as, e.g., in Bear and Bachmat (2012) as

$$\frac{\partial C_i}{\partial t} = -\nabla \cdot (V C_i) + \nabla \cdot (D_i \nabla C_i)(i = 1, 2, 3, ..., m) \qquad (6.4)$$

where C_i $(mol \cdot L^{-1})$ is the molar concentration of the i-th species of the m multi-species system. D_i is the diffusion–dispersion coefficient of component i $(m^2 \cdot s^{-1})$. The mass transport in the liquid phase can be influenced by convection, diffusion, decay and biodegradation, sorption and chemical reactions. If we specify the reaction terms in the ADE (Eq. 6.4), the coupled set of advection–dispersion–reaction equations can be written as

$$\frac{\partial C_i}{\partial t} = -\nabla \cdot (\frac{V}{R_i} C_i) + \nabla \cdot (\frac{D_i}{R_i} \nabla C_i) + \frac{\beta_i}{R_i} (i = 1, 2, 3, ..., m) \tag{6.5}$$

$$\frac{\partial S_j}{\partial t} = \beta_j \, (j = 1, 2, 3, ..., n) \tag{6.6}$$

where R_i is the linear retardation factor of the i-th mobile component ($R = 1 + \rho K_d/n$), ρ is the bulk density $(mg \cdot L^{-1})$ and K_d is the linear sorption constant ($L \cdot mg^{-1}$)); and β_i and β_j are the reactions involving mobile and immobile components, respectively. S_j is the solid phases concentration $(mol \cdot kg^{-1})$.

Since the sequential non-iterative approach (SNIA) for operator splitting is applied in the *OGS#IPhreeqc* coupling scheme, Eq. 6.5 is decoupled into a transport step (Eq. 6.7) and reaction steps (Eqs. 6.8 and 6.9) so that transport and reactions are solved sequentially.

$$\frac{\partial C_i}{\partial t} = -\nabla \cdot (\frac{V}{R_i} C_i) + \nabla \cdot (\frac{D_i}{R_i} \nabla C_i) \tag{6.7}$$

$$\frac{\partial \bar{C}_i}{\partial t} = \frac{\beta_i}{R_i} \tag{6.8}$$

and

$$\frac{\partial S_j}{\partial t} = \beta_j \tag{6.9}$$

The advection–dispersion terms for all mobile species (Eq. 6.7) are solved at first before the resulting concentrations ($\bar{C}_i$) are used to calculate a set of coupled reaction terms (both mobile and immobile components, Eqs. 6.8 and 6.9).

6.3.3 Geochemical System

The geochemical reaction system is defined to consider water–mineral interaction affecting redox reactions (i.e., nitrate reduction processes). Related reactions are summarized in Table 6.2. The default *PHREEQC* database "*phreeqc.dat*" (Parkhurst and Appelo 1999), which contains the thermodynamic data for aqueous species and mineral phases, is used for the simulation. The reactions between the oxidized recharge

Table 6.2 Definition of the geochemical system (modified from Jang et al., (2017))

Pyrite oxidation and related oxidation of Fe(II)	
$FeS_2 + 3.5O_2 + H_2O \rightarrow Fe2 + +2SO_4^{2-} + 2H^+$	Pyrite oxidation by oxygen
$FeS_2 + 14Fe^{3+} + 8H_2O \rightarrow 15Fe^{2+} + 2SO_4^{2-} + 16H^+$	Pyrite oxidation by Fe^{3+}
$FeS_2 + 2.8NO_3^- + 0.8H_2O \rightarrow Fe^{2+} + 2SO_4^{2-} + 1.4N_2 +$ $0.4H_2O$	Autotrophic denitrification
$Fe^{2+} + 0.25O_2 + H^+ \rightarrow Fe^{3+} + 0.5H_2O$ and	Fe^{2+} oxidation by oxygen
$Fe^{3+} + 2H_2O \rightarrow FeOOH + 2H^+$	
$5Fe^{2+} + NO_3^- + 7H_2O \rightarrow 5FeOOH + 0.5N_2 + 9H^+$	Fe^{2+} oxidation by nitrate
Carbonate equilibrium for buffer reaction	
$CaCO_3 \leftrightarrow Ca^{2+} + CO_3^{2-}$	Calcite dissolution
$CO_3^{2-} + 2H^+ \leftrightarrow H_2CO_3$	Carbonate equilibrium I
$CO_3^{2-} + H^+ \leftrightarrow HCO_3^-$	Carbonate equilibrium II
$H_2O \leftrightarrow OH^- + H^+$	Dissociation of water

water and the aquifer's main reductant pyrite are kinetically controlled. Pyrite oxidation using oxygen, nitrate, and sulfate as electron acceptors are assumed to follow a previously developed and applied rate expression (Williamson and Rimstidt 1994):

$$r_{O2} = 10^{-8.19} \cdot \frac{m_{O2}^{0.5}}{m_{H+}^{0.11}} \quad (6.10)$$

$$r_{O2_{Fe3}} = 10^{-5.78} \cdot \frac{m_{Fe3+}^{0.92}}{(1 + m_{Fe2+})^{0.43}} \quad (6.11)$$

$$r_{Fe3} = 10^{-8.84} \cdot \frac{m_{Fe3+}^{0.28}}{m_{H+}^{0.3}(1 + m_{Fe2+})^{0.52}} \quad (6.12)$$

$$r_{NO3} = f_{nit} \cdot \frac{m_{NO3}^{0.5}}{m_{H+}^{0.11}} \quad (6.13)$$

$$r_{pyrite} = A_{pyrite} \frac{m}{m_{0}}_{pyr}^{0.67} \cdot (1 - SI(\text{"Pyrite"})) \cdot (r_{O2} + r_{O2_{Fe3}} + r_{Fe3} + r_{NO3}) \quad (6.14)$$

where r is the reaction rate, m refers to the concentration of either O_2, Fe^{2+}, Fe^{3+} or H^+, $(\frac{m}{m_0})_{pyrite}$ indicates the change in the surface area of pyrite due to its dissolution, A_{pyrite} is a pyrite surface area and $SI(\text{"Pyrite"})$ is the Saturation Index of pyrite

Table 6.3 Required input files for the nitrate RT modeling

File	Description
Denitrification.gli	Geometry
Denitrification.msh	Finite element mesh
Denitrification.pcs	Process control
Denitrification.num	Numeric file
Denitrification.tim	Time discretization
Denitrification.ic	Initial condition
Denitrification.bc	Boundary condition
Denitrification.mcp	Component properties
Denitrification.mfp	Fluid properties
Denitrification.mmp	Medium properties
Denitrification.msp	Solid properties
Denitrification.pqc	*PHREEQC* input definition
Phreeqc.dat	*PHREEQC* database
Denitrification.out	Output configuration

mineral, where SI is equal to the logarithmic value of the ratio between the ion activity product (IAP) and the solubility product (K_{sp}) for the mineral phases considered. In the rate equilibrium of Williamson and Rimstidt (1994), $f_{nit} m_{NO3}^{0.5} m_{H+}^{-0.11}$ was added for modeling oxidation by nitrate where $f_{nit} = 1$ similar done by Eckert and Appelo (2002). The precipitation and dissolution of calcite and goethite is simulated through the use of mass action equations and the equation describing equilibrium with $Log K_{sp} = -8.48$ and $Log K_{sp} = -1$, respectively.

6.4 Model Setup

6.4.1 Preprocessing

6.4.1.1 Create Geometry

With *GINA* version 3.2.1 or higher, we can create the geometry and mesh. First, open *GINA* to see the following window (Fig. 6.2).
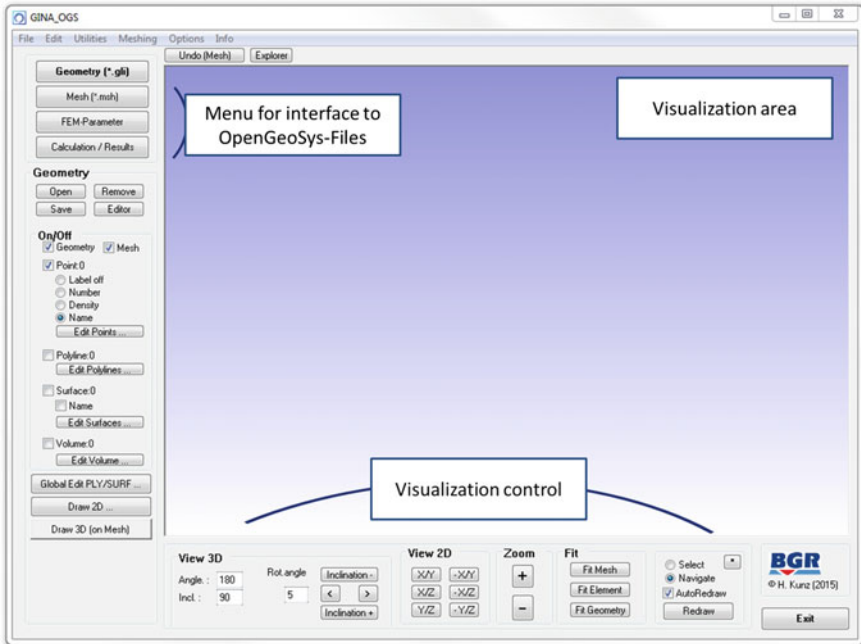
Fig. 6.2 Start in *GINA*

To create the geometry, select the button Geometry(*.gli) and turn on the Geometry in left navigation. Click the Edit Points to enter point coordinates. A table will appear where the point number, x-, y-, and z- coordinates, and point name can be assigned. Please enter seven points as shown in Fig. 6.3-① and Listing 6.1. Note that a default unit is the meter (m).

Then, we have to create polylines based on the points. Click Edit Polylines in the left navigation menu. Lines are an ordered list of the points and also can be closed to represent polygons by specifying the first and last points of the line to be identical. An important thing is point numbers in the right field have to be listed one below the other. Figure 6.3-② exemplary displays for the polyline named BOUNDARY. All other points and polylines that have to be entered are summarized in Table 6.4.

A surface is created based on the polyline. Click Edit Surfaces in the left navigation menu and enter the name of a surface. Then, click a list of polylines and choose the relevant polyline. In this model, only one surface is needed and therefore, we create a surface called "AQUIFER" and select "BOUNDARY" for the relevant polyline (Fig. 6.3-③). Finally, Save the created geometry ("*denitrification.gli*") by using File →Save at →Geometry. Now, a geometry file is ready for *OGS* simulations (Listing 6.1).
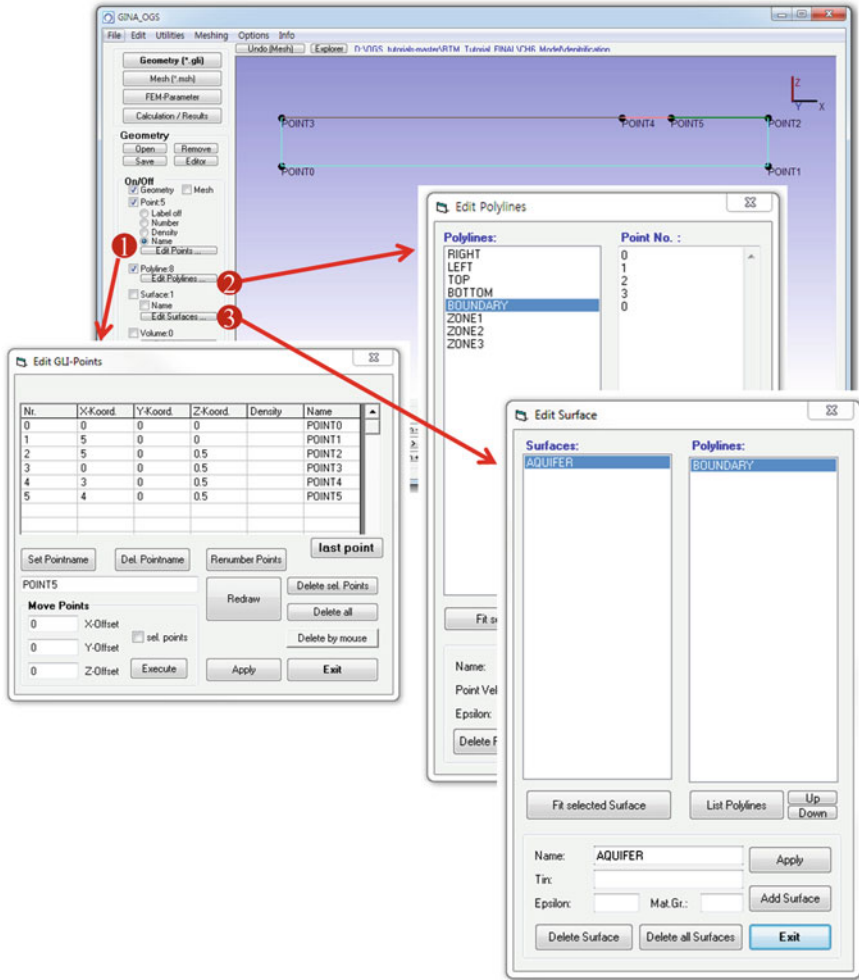
Fig. 6.3 2D geometry by *GINA*

Listing 6.1 Geometry file

```
#POINTS
0 0 0 0     $NAME  POINT0
1 5 0 0     $NAME  POINT1
2 5 0 0.5   $NAME  POINT2
3 0 0 0.5   $NAME  POINT3
4 3 0 0.5   $NAME  POINT4
5 4 0 0.5   $NAME  POINT5

#POLYLINE
$NAME
TOP
$POINTS
2
```

```
3
#POLYLINE
$NAME
BOTTOM
$POINTS
0
1
#POLYLINE
$NAME
RIGHT
$POINTS
1
2
#POLYLINE
$NAME
LEFT
$POINTS
0
3

[...]

#SURFACE
$NAME
AQUIFER
$POLYLINES
BOUNDARY
#STOP
```

6.4.1.2   Mesh Generation

After the geometry is created, switch the button on Mesh(*.msh). Then, use
the pull-down menu at the top of the window and click Meshing →Creation
Elements from Geometry. Choose the Quad and click Exit (Fig. 6.4-①).
You can build line-, tri-, or quad elements from the existing geometry. Make
sure that an option Delete Geometry should be deactivated. Otherwise, your
whole geometry will be deleted after element creation. Click Meshing →Refine
Elements. Here, you can refine an existing mesh. For this case study, we choose
homogeneous meshing with a relatively small element size of 0.25 m (x) and 0.05 m
(y). Thus, the x-direction is divided into 20 units (for 0.25 m) and the y-direction
into 10 units (for 0.05 m) by using Even. The last two steps are Refine and Exit
(Fig. 6.4-②). The meshing procedure may take several seconds depending on the
options and your CPU (Fig. 6.5).

Finally, save the mesh file as "*denitrification.msh*" under File →Save at
→Mesh. Take care that all files (geometry, mesh and any other following ones) have
to be saved in one directory and same name but with different file ending.
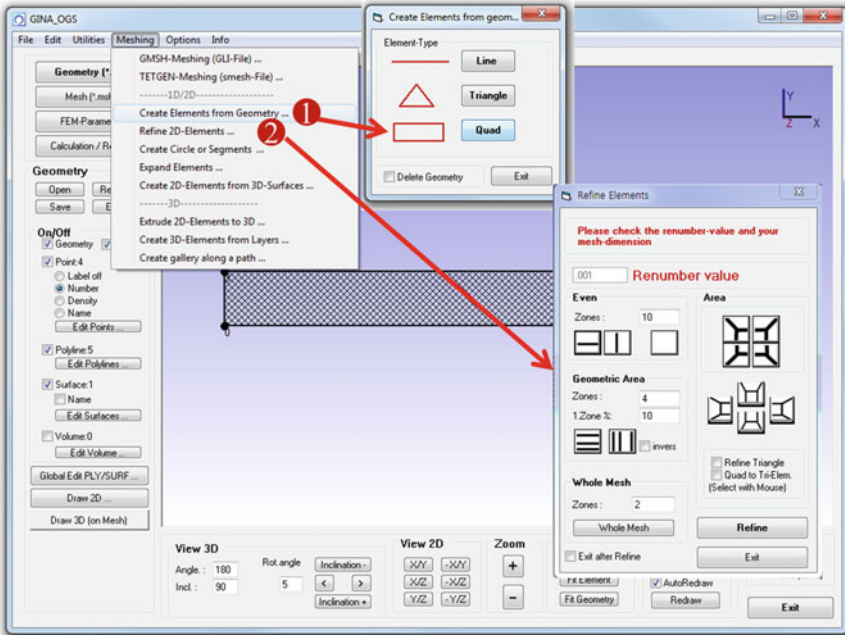
Fig. 6.4 2D mesh generation by *GINA*
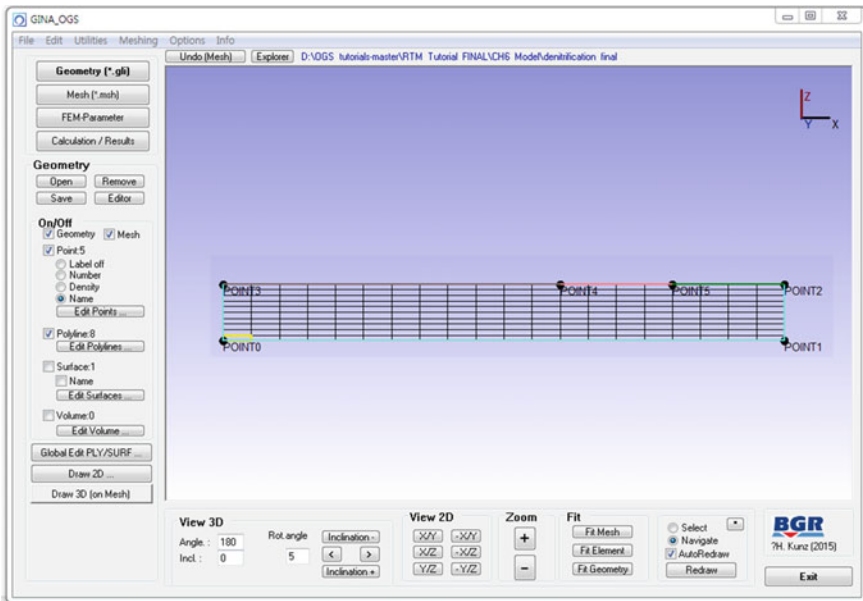
Fig. 6.5 Final 2D model

Table 6.4 Assigning names for geometric objects

Polyline Name	Point No.	Polyline Name	Point No.
TOP	2 3	BOTTOM	0 1
RIGHT	1 2	LEFT	0 3
BOUNDARY	0 1 2 3 0	ZONE1	3 4
ZONE2	4 5	ZONE3	2 5
Surface Name	Polyline Name		
AQUIFER	BOUNDARY		

Listing 6.2 Mesh file

```
#FEM_MSH
$PCS_TYPE
GROUNDWATER_FLOW
$NODES
231
0  0  0  0
1  .25  0  0
2  .25  0  .05
[..]
228  5  0  .4
229  5  0  .45
230  5  0  .5
$ELEMENTS
200
0  0  quad  0  1  2  3
1  0  quad  3  2  4  5
2  0  quad  5  4  6  7
[..]
197  0  quad  216  227  228  217
198  0  quad  217  228  229  218
199  0  quad  218  229  230  219
#STOP
```

6.4.1.3 Process Definition

All processes have to be defined in the process definition (PCS) file. In our simulation, one water flow (GROUNDWATER_FLOW) and 18 mass transport processes have to be listed in the PCS file. The number or mass transport processes MASS_TRANSPORT must have the same number of components which will be defined in MCP file (Sect. 6.4.1.8). Thus, please add MASS_TRANSPORT PCS block 18 times below the groundwater flow definition GROUNDWATER_FLOW. Note that the #PROCESS keyword indicates the beginning of process related numerical setting. Each data block ends with the beginning of new PCS block with the #PROCESS, or with the file input terminator #STOP.

Please set up the PCS file by creating "*denitrification.pcs*" with *Notepad++*.

Listing 6.3 Process File

```
#PROCESS
$PCS_TYPE
GROUNDWATER_FLOW

#PROCESS        ; comp 1
$PCS_TYPE
MASS_TRANSPORT

#PROCESS        ; comp 2
$PCS_TYPE
MASS_TRANSPORT

[..]

#PROCESS        ; comp 18
$PCS_TYPE
MASS_TRANSPORT

#STOP
```

6.4.1.4 Numerics

The numerical file defines and controls a numerical solver. Since GROUNDWATER_
FLOW and MASS_TRANSPORT processes are used in this simulation, two sets of
numeric data blocks are required. The sub-keywords ELE_MASS_LUMPING and
FEM_FCT help to increase numerical stability of the solution.

Listing 6.4 Numeric File

```
#NUMERICS
$PCS_TYPE
GROUNDWATER_FLOW
$LINEAR_SOLVER
; method error_tolerance max_iterations theta precond storage
2      6 1.e-014        2000             1.0   1       2

#NUMERICS
$PCS_TYPE
MASS_TRANSPORT
$LINEAR_SOLVER
; method error_tolerance max_iterations theta precond storage
2      6 1.e-014        2000             0.5   1       2
$ELE_MASS_LUMPING
1
$FEM_FCT
1 0
#STOP
```

6.4.1.5 Time Stepping

Here, we can define the time discretization. A constant time step size of 30 hours
with a total simulation time of 3000 hours is established. Since the default setting

is SECOND, we need an additional sub-keyword $TIME_UNIT to change the time unit. The available time unit is SECOND, MINUTE, HOUR, DAY, MONTH, and YEAR.

Please set up the time stepping by creating a "*denitrification.tim*" file with *Notepad++*. The following code has to be written (Listing 6.5).

Listing 6.5 Time step control file

```
#TIME_STEPPING
$PCS_TYPE
GROUNDWATER_FLOW
$TIME_STEPS
100 30
$TIME_END
3000
$TIME_START
0.0
$TIME_UNIT
HOUR

#TIME_STEPPING
$PCS_TYPE
MASS_TRANSPORT
$TIME_STEPS
100 30
$TIME_END
3000
$TIME_START
0.0
$TIME_UNIT
HOUR

#STOP
```

6.4.1.6 Initial Condition

The initial condition file is needed to determine the values of primary variables at the starting time of the simulation. Open *Notepad++* and create a new file called "*denitrification.ic*". Please add the following lines based on the format listed in the Listing 6.6.

Listing 6.6 Initial condition file

```
#INITIAL_CONDITION
$PCS_TYPE
GROUNDWATER_FLOW
$PRIMARY_VARIABLE
HEAD
$GEO_TYPE
DOMAIN
$DIS_TYPE
CONSTANT 70

#INITIAL_CONDITION
$PRIMARY_VARIABLE
Na
$GEO_TYPE
```

```
DOMAIN
$DIS_TYPE
CONSTANT 3.67E-04

[..]

$PRIMARY_VARIABLE
Pyrite
$GEO_TYPE
DOMAIN
$DIS_TYPE
CONSTANT 0.0035
#STOP
```

The primary variables of GROUNDWATER_FLOW and MASS_TRANSPORT are hydraulic HEAD and solute species (e.g., Na, N(5), and so on) which defined in the component property file, respectively.

We assume that an initial head of 70m for the entire model domain. Therefore, the $GEO_TYPE is DOMAIN and the required parameter is CONSTANT 70 at the GROUNDWATER_FLOW block. The sub-keyword DIS_TYPE specifies the way of how data are distributed over the defined geometric object. Under DIS_TYPE, first parameter indicates the distribution type (CONSTANT or CONSTANT_NEUMANN), and second parameter gives its value. The distribution type CONSTANT assigns the same value to every node belonging to the given geometric object whereas CONSTANT_NEUMANN assigns the value multiplied by node area/node length to each node.

To determine the initial distribution of mobile and immobile species in the model, MASS_TRANSPORT blocks are required. The initial concentration is based on the field measurement (Hessian Ried, Germany) conducted by Preiß (2013) and Knipp (2012) shown in detail in Table 6.5. Only one parameter can be defined as a primary variable per IC block. Thus, total 18 initial data blocks are needed including pe, pH, and nitrate tracer (chemically inert).

Note that we assume the concentration of nitrate and nitrate tracer are zero before the simulation. This does not reflect the reality; however, it allows identifying the distribution of reactive species and calculating nitrate and reaction products over time without having to consider the past nitrate input and associated chemical reactions.

6.4.1.7 Boundary Condition

The flow direction is from right to left with a regional hydraulic gradient of 0.001. To do this, we assign the hydraulic head 70.005 m at the right model boundary (POLYLINE_RIGHT). Note that nitrate is entering from the top boundary with various concentrations of source zones as shown in Fig. 6.1. That means three boundary condition blocks are needed for the each source zone. Tracer is chemically inert and used to illustrate the nitrate configuration developing from purely physical transport in the absence of any reactive processes. Thus, please include three additional bound-

Table 6.5 Water chemistry and reactants used for initial and boundary condition including *PHREEQC* notation (modified from Jang et al., (2017))

Initial condition			
pe	5.1		
pH	7.8		
Aqueous components	$(mol \cdot L^{-1})$		
Na	3.67E-4	Ca	2.54E-3
Mg	8.64E-4	K	4.99E-5
C(+4) as HCO_3	4.80E-3	Cl	7.44E-4
S(6) as SO_4	9.47E-4	N(5) as NO_3	-
O(0) as O_2	-	Fe(2) as Fe^{2+}	1.02E-5
Tracer(chemically inert)	-		
Solid phase $(mol \cdot kg^{-1})$			
Pyrite (FeS_2)	3.50E-3	Goethite $(FeOOH)$	-
Calcite $(CaCO_3)$	5.00E-6	-	-
Boundary condition			
	ZONE1 $(\cdot L^{-1})$	ZONE2 $(mol \cdot L^{-1})$	ZONE3 $(mol \cdot L^{-1})$
N(5) as NO_3	1.45E-3	1.15E-3	6.62E-4
O(0) as O_2	2.24E-4	2.24E-4	2.24E-4
Tracer	1.45E-3	1.15E-3	6.62E-4

ary condition blocks for the tracer with the same concentration of nitrate. Oxygen is assumed to have a uniform distribution with a concentration of 2.24E-4 $mol \cdot L^{-1}$

Please set up the boundary condition by creating a "*denitrification.bc*" file with *Notepad++*. The following code has to be entered into the file (Listing. 6.7).

Listing 6.7 Boundary condition file

```
#BOUNDARY_CONDITION
$PCS_TYPE
GROUNDWATER_FLOW
$PRIMARY_VARIABLE
HEAD
$GEO_TYPE
POLYLINE RIGHT
$DIS_TYPE
CONSTANT 70.005

#BOUNDARY_CONDITION
$PCS_TYPE
GROUNDWATER_FLOW
$PRIMARY_VARIABLE
HEAD
$GEO_TYPE
POLYLINE LEFT
$DIS_TYPE
CONSTANT 70

#BOUNDARY_CONDITION
$PCS_TYPE
```

```
MASS_TRANSPORT
$PRIMARY_VARIABLE
N(5)
$GEO_TYPE
POLYLINE ZONE1
$DIS_TYPE
CONSTANT  1.45E-3

#BOUNDARY_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
N(5)
$GEO_TYPE
POLYLINE ZONE2
$DIS_TYPE
CONSTANT  1.15E-3

#BOUNDARY_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
N(5)
$GEO_TYPE
POLYLINE ZONE3
$DIS_TYPE
CONSTANT  6.62E-4

[...]

#BOUNDARY_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
O(0)
$GEO_TYPE
POLYLINE TOP
$DIS_TYPE
CONSTANT  2.24E-4
#STOP
```

6.4.1.8   Component Properties Definition

For each species, a data block has to be added in a "*denitrification.mcp*" file.
PHREEQC notation for chemical formulae can be found in Table 6.6. Five immobile
components should be defined as immobile ($MOBILE = 0) whereas other mobile
components are transport in the aqueous phase ($MOBILE = 1). For the molecular
diffusion, a value of 1E-9 $m^2 \cdot s^{-1}$ is assumed for the each component.

Listing 6.8 Component properties file

```
#COMPONENT_PROPERTIES
$NAME
N(5)
$MOBILE
1   ; MOBILE-Flag: 0=imMOBILEe, 1=MOBILEe/transported
$DIFFUSION
1  1E-9            ; diffusion model type, diffusion
    constant
```

```
#COMPONENT_PROPERTIES
$NAME
N(0)
$MOBILE
1    ; MOBILE-Flag: 0=imMOBILEe, 1=MOBILEe/transported
$DIFFUSION
1   1E-9              ; diffusion model type, diffusion
    constant

[..]

#COMPONENT_PROPERTIES
$NAME
Pyrite ;
$MOBILE
0    ; MOBILE-Flag: 0=imMOBILEe, 1=MOBILEe/transported
$DIFFUSION
0            ; diffusion model type, diffusion constant
#STOP
```

Table 6.6 Mobile and immobile components

PHREEQC notation	
Mobile components	
Na	Na^+
Ca	Ca^{+2}
Mg	Mg^{+2}
K	K^+
Fe(2)	Fe^{+2}
Fe(3)	Fe^{+3}
C(+4)	HCO_3^{-2}
Cl	Cl^-
S(6)	SO_4^{-2}
N(5)	NO_3^-
N(0)	N_2
O(0)	O_2
Tracer	Nitrate tracer
Immobile components	
pe	
pH	
Pyrite	FeS_2
Goethite	$FeOOH$
Calcite	$CaCO_3$

6.4.1.9 Porous Media and Material Properties

In this simulation, we assume that the whole domain is one material group. The material parameters required for the simulation can be defined in the file "*denitrification.mmp*" (Listing 6.9). For the porosity and tortuosity, values of 0.34 and 1, respectively are assumed. Note that for groundwater flow processes (GROUNDWATER_FLOW), the hydraulic conductivity in $m \cdot s^{-1}$ is used instead of the permeability but the keyword says $PERMEABILITY_TENSOR. Our model assumes an isotropic case with a hydraulic conductivity of 1.55E-4 $m \cdot s^{-1}$. And longitudinal dispersivity (α_l) and transversal dispersivity (α_t) are 0.1 and 0.01 m, respectively.

Listing 6.9 Medium properties File

```
#MEDIUM_PROPERTIES
$NAME
Aquifer
$GEOMETRY_DIMENSION
2
$GEOMETRY_AREA
1.000000e+000
$POROSITY
1   0.34
$VOL_MAT
1   0.66
$TORTUOSITY
1    1.000000e+000
$PERMEABILITY_TENSOR
ISOTROPIC 1.55E-4
$MASS_DISPERSION
1   0.1   0.01
$DENSITY
1   1716.0
#STOP
```

For the mass transport simulation, the material density of the aquifer and fluid information are also required. The fluid properties file contains the description of one or more fluids. The unit of density and viscosity are $kg \cdot m^{-3}$ and $kg \cdot m^{-1} \cdot s^{-1}$, respectively.

Please create "*denitrification.msp*" and "*denitrification.mfp*" files and add the following lines shown in the Listing 6.10 and 6.11, respectively.

Listing 6.10 Solid properties

```
#SOLID_PROPERTIES
$DENSITY
1 1.716E+003
#STOP
```

Listing 6.11 Fluid properties

```
#FLUID_PROPERTIES
$FLUID_TYPE
LIQUID
$PCS_TYPE
HEAD
$DENSITY
```

```
1 1000.0
$VISCOSITY
1 1.0e-3
#STOP
```

6.4.1.10   PHREEQC Calculation

In the *OGS#IPhreeqc* simulation, *PHREEQC* input definition file is based on the
PHREEQC input file. The concentration of each component does not affect the
system. Instead, the real concentration of the component has to be defined in the
initial condition (*denitrification.ic*) and boundary condition file (*denitrification.bc*).
Note that the nitrate tracer is not listed in the *PHREEQC* input file since it does not
include any reactive processes.

Please set up the *PHREEQC* input file by creating a "*denitrification.pqc*" file with
Notepad++. The following code has to be entered into the file.

Listing 6.12 PHREEQC input definition

```
SOLUTION 1
units mol/kgw
temp 25.000000
pH 7.8 charge # comp 1
pe 5.1  # comp 2
Na 3.67E-4 # comp 3
Ca 2.54E-3 # comp 4
Mg 8.64E-4 # comp 5
K 4.99E-5 # comp 6
C(+4) 4.80E-3 # comp 7
Cl 7.44E-4 # comp 8
S(6) 9.47E-4 # comp 9
N(5) 0 # comp 10
N(0) 0 # comp 11
O(0) 0 # comp 12
Fe(2) 1.02E-5 # comp 13
Fe(3) 0 # comp 14
#ende

EQUILIBRIUM_PHASES 1
Calcite   0 5.00E-5   # comp 15
Goethite  0 0         # comp 16
#ende

KINETICS
Pyrite   # comp 17
-formula   FeS2   1
-m         3.50E-3
-m0        3.50E-3
-tol       1e-008
#ende

RATES
Pyrite
-start
1 A = 15e3*m0
10 if SI("Pyrite") >0 then goto 100
20 fH = MOL("H+")
30 fFe2 = (1 + TOT("Fe(2)"))
40 if MOL("O2") < 1e-6 then goto 80
```

```
50  rO2 = 10^-8.19*MOL("O2")^0.5*fH^-0.11
60  rO2_Fe3=10^-5.78 *TOT("Fe(3)")^0.92*fFe2^-0.43
70  goto 90
80  rem  mol/dm3/s
81  rFe3= 10^-8.84*TOT("Fe(3)")^0.28*fFe2^-0.52*fH^-0.3
82  rNO3 = TOT("N(5)")^0.5 *fH^-0.11
90  rate = A* (m/m0)^0.67 *(rO2 + rO2_Fe3 + rFe3+ rNO3) * (1-
    SR("Pyrite"))
100 Save rate*time
-end
#ende

USER_PUNCH
10 PUNCH TOT("N(5)"), TOT("N(0)"), TOT("Fe(2)"), TOT("Fe(3)")
   , TOT("O(0)"), TOT("S(6)")
20 PUNCH -LA("H+"), -LA("e-")
30 PUNCH KIN("Pyrite"), EQUI("Goethite"), EQUI("Calcite")
#ende
END
#STOP
```

Kinetic rate expressions of geochemical reactions can be calculated with the embedded BASIC interpreter in *PHREEQC*. In the data block RATES, the mathematical expression of the kinetic reactions is defined (described detailed in Sect. 6.3.3). Do not forget to add #comp at the right-hand side of each component and #ende after each *PHREEQC* module.

6.4.1.11 Output

In the output file, we can define how and when simulation results are written. As shown in the following Listing 6.13, two output blocks are required. Each of them starts with the keyword #OUTPUT.

For the groundwater flow simulation, we want to get hydraulic HEAD [m] over the entire domain with the VTK format. Under the keyword TIM_TYPE, specification STEPS 10 means that the simulation results will be printed once in every ten steps. This output block ends with the beginning of the next output block, i.e., #OUTPUT. In the second output block, the concentration of the transport parameters can be defined by adding the corresponding names in the sub-keyword $NOD_VALUES. The parameter should be listed one below the other. Again, the results will be printed once in every ten steps. The output file ends with the file input terminator #STOP.

Please open *Notepad++* and opens a new form and add the following lines. Subsequently, save the file as "*denitrification.out*"

Listing 6.13 Output File

```
#OUTPUT
$PCS_TYPE
GROUNDWATER_FLOW
$GEO_TYPE
DOMAIN
$ELE_VALUES
HEAD
$DAT_TYPE
VTK
```

```
$TIM_TYPE
STEPS 10

#OUTPUT
$PCS_TYPE
MASS_TRANSPORT
$NOD_VALUES
N(5)
N(0)
[..]
Tracer
$GEO_TYPE
DOMAIN
$DAT_TYPE
VTK
$TIM_TYPE
STEPS 10
#STOP
```

6.4.2 Model Execution

When all input files (Table 6.3) are ready, please add the executable *ogs.exe* to the folder containing the input files. Open the executable and enter `denitrification` in the command line (Fig. 6.6). The simulation starts and results files will appear (`denitrification_GROUNDWATER_FLOW00*.vtk` and `denitrification_MASS_TRANSPORT00*.vtk`) in your data folder.
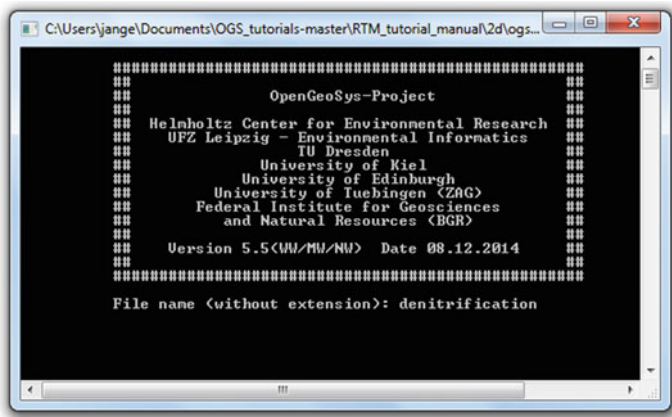
Fig. 6.6 Command window of the *OpenGeoSys* executable
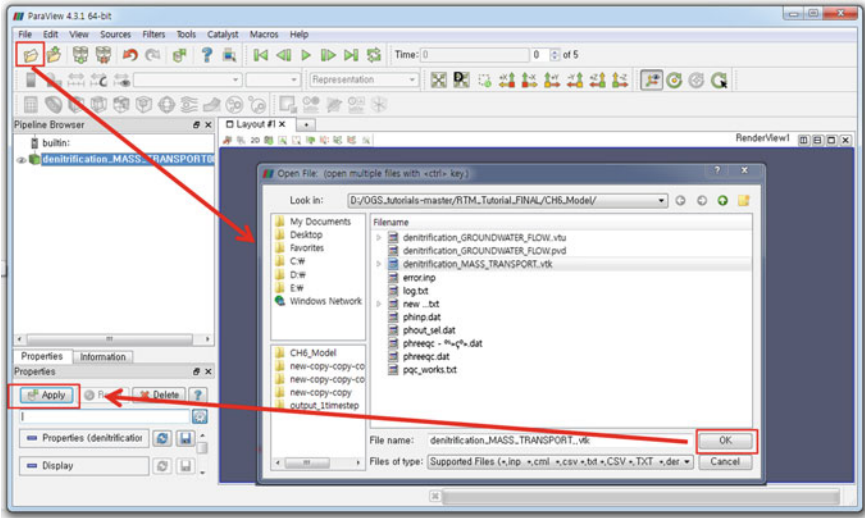
Fig. 6.7 Open the file in *ParaView*
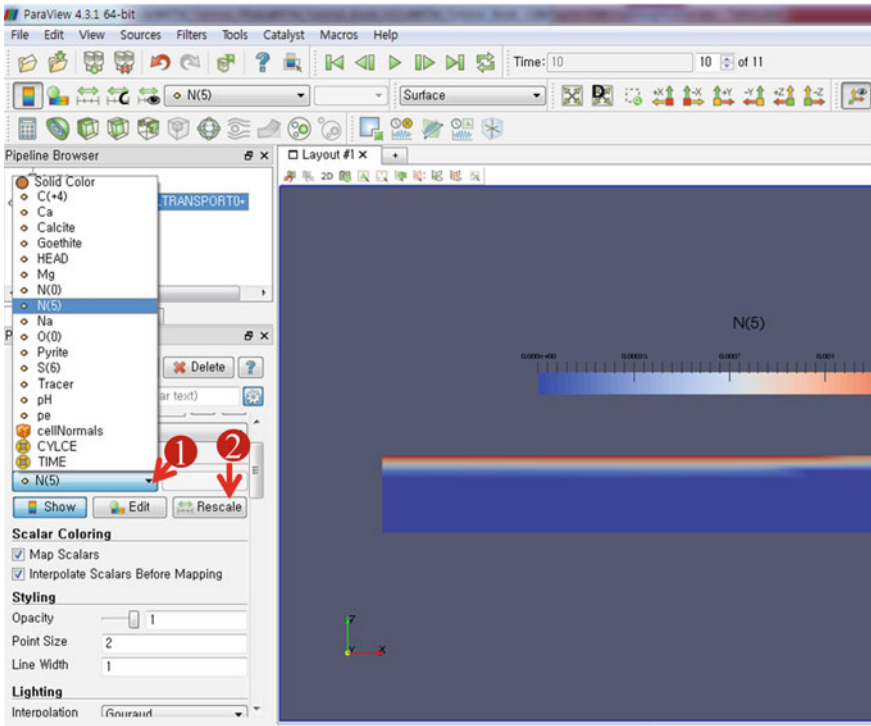
Fig. 6.8 Visualization of the mass transport

6.4.3   Post-processing

For the visualization of simulation result, *ParaView* can be applied. Open the *ParaView* and click on File →Open in the Main Toolbar. Locate the *OGS* project folder and open the file *.vtk which will appear in the pipeline browser. Click on the data set and press Apply in the properties window (Fig. 6.7). The model domain should appear in the layout window.

To visualize the mass transport, for example, select "denitrification_ MASS_TRANSPORT..vtk". Then, you can choose components to visualize in the pipeline browser Fig. 6.8(1). After you have selected the component, click Rescale Fig. 6.8(2).

6.5   Simulation Results

Simulation results are shown in Figs. 6.9 and 6.10. Within the upper part of the aquifer, a sharp boundary is observed where high nitrate transitions into low-nitrate water. As the oxygen in the solution is quickly consumed with depth, conditions become thermodynamically favorable for denitrifying NO_3 to N_2 where the presence of pyrite is ubiquitous throughout the domain. The depth where nitrate is depleted coincides with the depth where the concentration of sediment pyrite starts to increase (Figs. 6.9-a and 6.10-c). Excess nitrogen is comparatively a conservative product of nitrate reduction and has been used as a natural tracer to identify or quantify denitrification, especially in deep, confined groundwater system. In this simulation, it is assumed that the initial groundwater system does not contain any free nitrogen species. Thus, the amount of nitrogen gas (N_2) produced can be referred in terms of equivalent reduced nitrate by denitrification (Fig. 6.9-b).
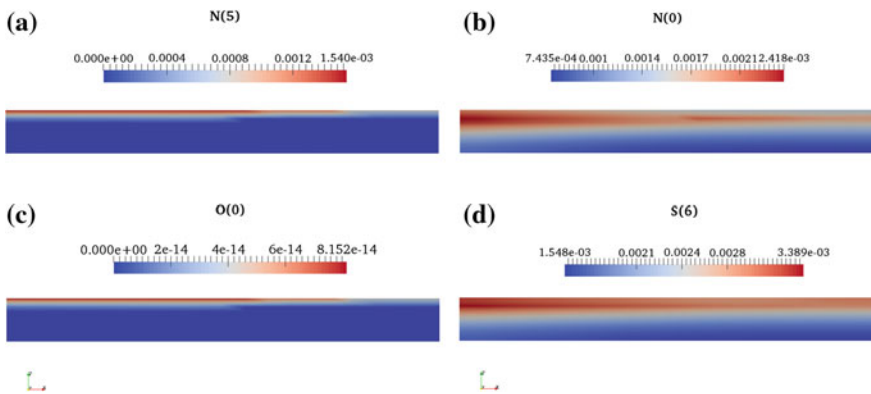
Fig. 6.9 Simulation results : Spatial distribution of **(a)** NO_3, **(b)** N_2,**(c)** O_2, and **(d)** SO_4 at the end of the simulation

(a) HEAD

(b) Tracer
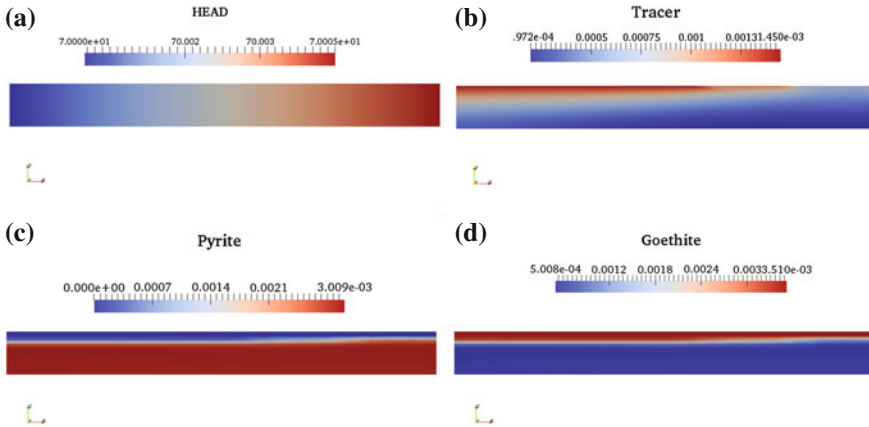
(c) Pyrite

(d) Goethite

Fig. 6.10 Simulation results : **(a)** groundwater flow, spatial distribution of **(b)** tracer, **(c)** pyrite, and **(d)** goethite at the end of the simulation

As nitrate is reduced by pyrite oxidation reaction, sulfate (SO_4^{2+}) and ferrous ion (Fe^{2+}) concentration are increased, and the produced ferrous ion is oxidized to ferric ion (Fe^{3+}). Then, ferric ion is precipitated as ferric oxide or oxyhydroxide such as goethite (Fig. 6.10-d). The nitrate tracer shows the nitrate configuration developing from purely physical transport in the absence of reactive processes. The simulated conservative nitrate tracer mainly reflects the groundwater flow and changes in the nitrate contents of recharge (Fig. 6.10-b).

References

C. Kludt, F.-A. Weber, A. Bergmann, K. Knöller, G. Berthold, C. Schüth. Identifizierung der nitratab-bauprozesse und prognose des nitratabbaupotenzials in den sedimenten des hessischen rieds. Grundwasser, **21**(3) 227–241 Sep 2016 ISSN 1432-1165, https://doi.org/10.1007/s00767-015-0317-5

E. Jang, W. He, H. Savoy, P. Dietrich, O. Kolditz, Y. Rubin, C. Schüth, T. Kalbacher, Identifying the influential aquifer heterogeneity factor on nitrate reduction processes by numerical simulation. Adv. in Water Resour. **99**, 38–52 (2017), https://doi.org/10.1016/j.advwatres.2016.11.007

A. M. Fan , V. E. Steinberg, Health implications of nitrate and nitrite in drinking water: An update on methemoglobinemia occurrence and reproductive and developmental toxicity. Regul. Toxicol. Pharmacol. **23**(1), 35 – 43 (1996). ISSN 0273-2300, http://dx.doi.org/10.1006/rtph.1996.0006

A. J. P. Smolders, E. C. H. E. T. Lucassen, R. Bobbink, J. G. M. Roelofs, L. P. M. Lamers, How nitrate leaching from agricultural lands provokes phosphate eutrophication in groundwater fed wetlands: the sulphur bridge. B.geochem. **98**(1) 1–7, Apr 2010. ISSN 1573-515X, https://doi.org/10.1007/s10533-009-9387-8

S. F. Korom, Natural denitrification in the saturated zone: A review. Water Resour. Res. 28(6), 1657–1668 (1992). ISSN 1944-7973, http://doi.org/10.1029/92WR00252

M. O. Rivett, S. R. Buss, P. Morgan, J. W. N. Smith, C. D. Bemment, Nitrate attenuation in groundwater: A review of biogeochemical controlling processes. Water Res. **42**(16), 4215 – 4232 (2008). ISSN 0043-1354, http://www.sciencedirect.com/science/article/pii/S0043135408002984

S.J. Bailey, A. Vanhatalo, P.G. Winyard, A.M. Jones, The nitrate-nitrite-nitric oxide pathway: Its role in human exercise physiology. European Journal of Sport Science **12**(4), 309–320 (2012), http://doi.org/10.1080/17461391.2011.635705

C. Torrentó, J. Cama, J. Urmeneta, N. Otero, A. Soler, Denitrification of groundwater with pyrite and thiobacillus denitrificans. Chem. Geol. **278**(1), 80 – 91 (2010). ISSN 0009-2541, http://www.sciencedirect.com/science/article/pii/S0009254110003165

Y.-C. Zhang, H. Prommer, H.P. Broers, C.P. Slomp, J. Greskoviak, B. Van der Grift, P. Van Cappellen, Model-based integration and analysis of biogeochemical and isotopic dynamics in a nitrate-polluted pyritic aquifer. Environ. sci. technol. **47**(18), 10415–10422 (2013), http://pubs.acs.org/doi/abs/10.1021/es4023909

J. Bear, Y. Bachmat, *Introduction to modeling of transport phenomena in porous media*, vol. 4 (Springer, Berlin, 2012)

D. L. Parkhurst , C. A. J. Appelo. User's guide to phreeqc (version 2) : a computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. Technical report, U.S. department of the interior u.s. geological survey, 1999, http://pubs.er.usgs.gov/publication/wri994259

M. A. Williamson, J. D. Rimstidt, The kinetics and electrochemical rate-determining step of aqueous pyrite oxidation. Geochimica et Cosmochimica Acta, **58**(24), 5443 – 5454 (1994). ISSN 0016-7037, http://www.sciencedirect.com/science/article/pii/0016703794902410

P. Eckert , C. A. J. Appelo. Hydrogeochemical modeling of enhanced benzene, toluene, ethylbenzene, xylene (btex) remediation with nitrate. Water Resour. Res. **38**(8), 5–1–5–11 (2002). ISSN 1944-7973, http://dx.doi.org/10.1029/2001WR000692

I. Preiß. Anwendbarkeit einer screeningmethode zur bestimmung des nitratabbaupotenzials mittels redoxprofilmessungen in grundwassermessstellen im hessischen ried. Master's thesis, Technische Universität Darmstadt. (2013)

E. Knipp. Charakterisierung des nitratabbaupotenzials an bohrkernproben aus dem hessischen ried -methodenvalidierung zur lokalisierung regionaler abbauhorizonte. Master's thesis, Technische Universität Darmstadt. (2012)

Chapter 7
Application: Treatment Wetlands

Johannes Boog

7.1 Introduction

This chapter describes the implementation of a 2D steady-state reactive transport model (RTM) of a treatment wetland. Treatment wetlands are technologies used for handling wastewater and other types of contaminated water. The focus of this tutorial will be on modeling secondary treatment of domestic wastewater by a passive horizontal subsurface flow (HSSF) treatment wetland. This translates into a modeling problem of variably saturated flow, transport of particulates and solutes, as well as, biological degradation of organic carbon, nitrogen, and sulfur. The treatment wetland that will be simulated was part of an experiment by García et al., (2005) and already in the focus of modeling studies by Llorens et al., (2011), Langergraber and Simunek (2012).

In this tutorial, we will show how to implement and execute an RTM with a coupling of *OGS#IPhreeqc* step by step.

7.2 Background

Untreated and non-adequately treated wastewater is still a major threat to human health, environmental safeguarding and the protection of freshwater resources, especially because of the discharge of reactive nitrogen (WWAP 2017). Treatment wetlands have been proven as promising technologies for treating wastewater of various

origins such as domestic and municipal effluents, industrial wastewater (Kadlec and Wallace 2009; Wu et al., 2015), agricultural runoff (Vymazal and Brezinova 2015), as well as, contaminated groundwater (van Afferden et al., 2011). Treatment wetlands can be characterized as man-made biogeochemical reaction systems that mimic natural processes to treat wastewater in an ecological and sustainable manner. The main treatment wetland type for handling domestic wastewater in temperate climates is subsurface flow (SSF) systems (Vymazal 2010) (Fig. 7.1).

SSF, especially when intensified with active aeration, combine high treatment performance regarding wastewater key pollutants such as organic carbon and nitrogen with process robustness, low maintenance costs, and a minimum requirement of infrastructure. Therefore, SSF are suitable to be used in decentralized or semi-centralized sanitation concepts. To support further optimization and deepen the process knowledge of SSF wetlands, process-based numerical modeling is a promising tool (Langergraber 2011). However, only a few process-based numerical models for wetlands that cover a different range of processes are available (Langergraber 2017). Unfortunately, the most advanced models such as BIO_PORE (Samso and Garcia 2013a), CWM1 (Langergraber et al., 2009), and CWM2 (Langergraber and Simunek 2005) are implemented in closed-source codes of commercially available software.

Subsurface flow constructed treatment wetlands are a complex interplay of the solid matrix, water flow, microbial organism, and plants (Fig. 7.1). The main drivers of the treatment processes are microbes and plants. The ratio of their contribution strongly depends on the inflow water quality and loading rate (influx). In highly loaded systems, such as horizontal subsurface flow (HSSF) wetlands for secondary treatment of domestic wastewater, the effect of plants is considered of minor importance with respect to the removal of bulk organic carbon and nitrogen. In such context, biodegradation is the dominating treatment processes.

Biodegradation in passive (no intensification) HSSF wetlands is mainly characterized by anoxic and anaerobe, and to some extent, aerobe processes (García et al., 2010). The reasons are saturated conditions in the wetland body, low oxygen transfer rates, and a low content of dissolved oxygen in the inflowing wastewater. In this case, the dominant chemical species are reduced forms of organic carbon, nitrogen, and sulfur. With respect to biodegradation, we will focus on simulating the transformation of organic carbon, nitrogen, and sulfur only. The effect of plants will not be taken into consideration.
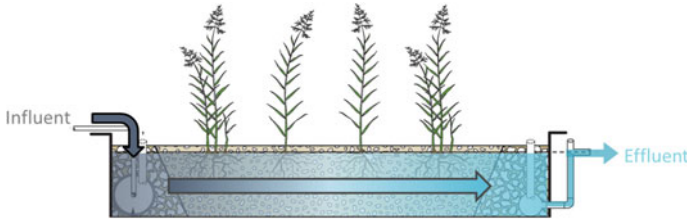
Fig. 7.1 Schematic of a horizontal subsurface flow treatment wetland according to Nivala et al., (2013)

7.3 Model Components and Governing Equations

7.3.1 Experimental System

The treatment wetland to model measures $10.3\,\mathrm{m}$ in length, $5.3\,\mathrm{m}$ in width, and $0.6\,\mathrm{m}$ in depth with an average water level at $0.5\,\mathrm{m}$. Gravel ($d_{10} = 60\,\mathrm{mm}$) with a porosity of 0.41 was used as main media. The system was loaded with primary settled domestic sewage at a rate of $36\,\mathrm{mm\,d^{-1}}$ and planted with *Phragmites Australis*.

7.3.2 Water Flow and Mass Transport

Water flow in variable saturated porous media can be modeled using the Richard's Equation in its pressure based form (from Kolditz et al., 2012b):

$$\epsilon \rho_w \frac{\partial S}{\partial p_c} \frac{\partial p_c}{\partial t} + \epsilon \nabla S q = Q_w \tag{7.1}$$

Here, ϵ is the porosity, ρ_w the water density, S denotes the dimensionless water saturation, p_c the capillary pressure and p_w the water pressure, t the time, q is the Darcy flux term, and Q_w is a source–sink term. The Darcy term is computed as:

$$q = \rho_w \frac{k_{rel}k}{\mu_w} (\nabla p_w - \rho_w g) \tag{7.2}$$

with, k the intrinsic permeability, k_{rel} the relative permeability, μ_w the water viscosity, and g the gravity acceleration. The relationship between capillary pressure and water saturation is described using the van Genuchten parametrization:

$$p_c = \frac{\rho_w g}{\alpha} \left[S_{eff}^{-\frac{1}{m}} - 1 \right]^{\frac{1}{n}} \tag{7.3}$$

were, α is a parameter related to the air entry pressure, n is the pore size distribution index, $m = 1 - \left(\frac{1}{n}\right)$ and S_{eff} is the effective saturation. The relationship between k_{rel} and S_{eff} is described as:

$$k_{rel} = S_{eff}^{\frac{1}{2}} \left[1 - \left(1 - S_{eff}^{\frac{1}{m}} \right)^m \right]^2 \tag{7.4}$$

The effective saturation is computed as:

$$S_{eff} = \frac{S - S_{res}}{S_{max} - S_{res}} \tag{7.5}$$

Hereby, S_{max} and S_{res} are the maximum and residual saturation. Mass transport of a mobile component i is computed using the advection–dispersion equation (ADE) (Bear and Bachmat, 1990) :

$$\epsilon \frac{\partial S C_i}{\partial t} = \epsilon \nabla S \left(q C_i + D \nabla C_i + D^* \nabla C_i \right) + Q_i + r_i \tag{7.6}$$

where C_i is the concentration of the i-th mobile component, q is the Darcy flux term, D is the mechanic dispersion coefficient, D^* is the diffusion coefficient, Q_i is a source–sink term, and r_i is the reaction term of the i-th component. As diffusion is not considered in this model, the diffusion term $D^* \nabla C_i$ can be neglected for all components. For an immobile component k such as bacteria, the right-hand side of the ADE reduces to the reaction term r_k. Following the sequential non-iterative (SNIA) coupling scheme of *OGS#IPhreeqc*, the reaction term r_i in the ADE is decoupled from solving the transport parts of the ADE. Transport and reaction steps are then solved subsequently always starting with the transport step.

7.3.3   Biodegradation

This tutorial uses the biokinetic model CWM1 that has already been implemented in other codes to model biodegradation in HSSF wetlands (Llorens et al., 2011; Langergraber and Simunek 2012; Samso and Garcia 2013a). CWM1 is described in detail in Langergraber et al., (2009). Basically, it models the fate of dissolved oxygen, organic carbon, organic nitrogen, ammonia and oxidized nitrogen, dihydrogen sulfide and sulfate, as well as hydrolysis of particulate organic matter and the growth and decay of heterotrophic, fermenting, autotrophic, methanogenic, and sulfate-reducing and sulfide-oxidizing bacteria (Figs. 7.1 and 7.2).

Hydrolysis and bacterial growth are modeled using monod-type kinetics including substrate consumption and inhibition terms as follows:

Table 7.1 Components of the biokinetic model CWM1

Description	Parameter	Unit
soluble		
Dissolved oxygen.	So	g O$_2$
Fermentable, readily biodegradable soluble COD	Sf	g COD
Fermentation products as acetate	Sa	g COD
Inert soluble COD.	Si	g COD
Ammonium and ammonia nitrogen	Snh	g N
Nitrate nitrogen	Sno	g N
Sulfate sulfur	Sso4	g S
Dihydrogensulfide sulfur	Sh2s	g S
particulate		
Slowly biodegradable particulate COD.	Xs	g COD
Inert particulate COD.	Xi	g COD
Heterotrophic bacteria	Xh	g COD
Autotrophic nitrifying bacteria	Xa	g COD
Acetotrophic methanogenic bacteria	Xamb	g COD
Fermenting bacteria	Xfb	g COD
Acetotrophic sulfate reducing bacteria	Xarsb	g COD
Sulfide oxidizing bacteria	Xsob	g COD
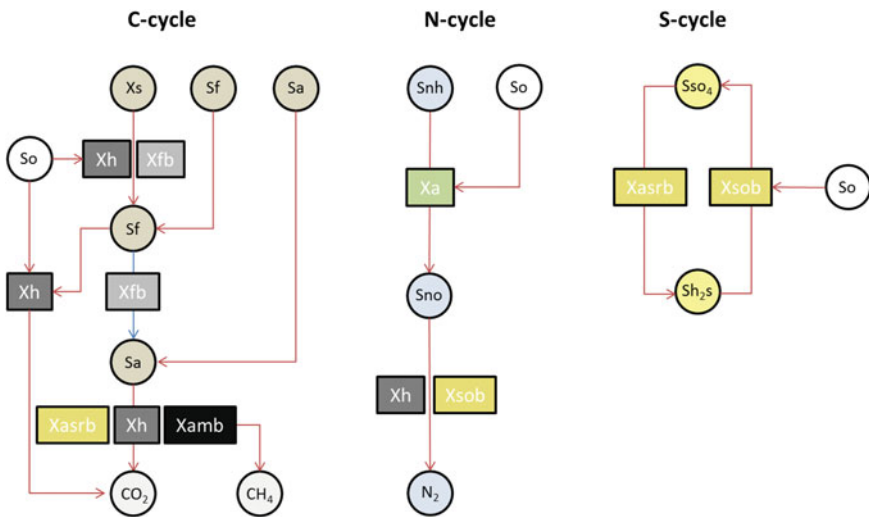
Fig. 7.2 Reaction network of substrates and products (cycles), and bacteria (boxes) of the biokinetic model CWM1 according to (Samso and Garcia, 2013b) CO$_2$, CH$_4$, and N$_2$ are drawn here for facilitating the conceptual understanding, however, their production is not included in the model

Table 7.2 Required Input Files

File	Description
wetland.geo	GMSH geometry file
wetland.gli	OGS geometry file
wetland.msh	Finite element mesh
wetland.pcs	Processes definition
wetland.num	Numerical methods
wetland.tim	Time discretization
wetland.mmp	Porous medium properties
wetland.mfp	Fluid properties
wetland.msp	Solid material properties
wetland.mcp	RT component properties
phreeqc.dat	Phreeqc database
wetland.pqc	Phreeqc control
wetland.rei	Reaction interface definition
wetland.ic	Initial conditions
wetland.st	Source terms
wetland.bc	Boundary conditions
wetland.out	Output specification

$$r_{k,growth} = \mu \cdot \left[\prod_{i=1}^{n} \left(\frac{S_i}{K_{Si} + S_i} \right) \right] \left[\prod_{j=1}^{m} \left(\frac{K_j}{K_j + S_j} \right) \right] \cdot X_k \qquad (7.7)$$

Here, μ is the maximum growth rate of bacteria X_k in h^{-1}, C_i is the concentration of the i-th substrate species in $mg\,L^{-1}$, K_{Si} is the corresponding saturation or inhibition term in $mg\,L^{-1}$, and X_k the concentration of bacteria type k in $mg\,L^{-1}$.

Bacterial decay is assumed to follow first-order rate equation:

$$r_{k,decay} = b_k \cdot X_k \qquad (7.8)$$

with X_k being the k-th bacterial type in $mg\,L^{-1}$ and the corresponding decay rate b_k in h^{-1} (Table 7.2).

7.4 Practical Model Setup

7.4.1 Preprocessing

7.4.1.1 Domain Geometry and Mesh

The 2D model domain is 10.3 m by 0.6 m and split into a 0.3 m long mixing zone at the system front and a main zone for treatment. The element size is 0.1 m. To generate a mesh with *GMSH*, you have to write an input file *wetland.geo* that defines the domain geometry, first. Define six points, then seven lines through these points, and create loops of these lines to define two surface structures.

Listing 7.1 GMSH geometry file *wetland.geo*

```
// set element size and points
lc=0.1;
Point(1)     =    {0,0,0,lc};
Point(2)     =    {0.3, 0, 0, lc};
Point(3)     =    {10.3,0,0,lc};
Point(4)     =    {10.3,0,0.6,lc};
Point(5)     =    {0.3,0,0.6,lc};
Point(6)     =    {0,0,0.6,lc};

// set lines
Line(7)      =    {1,2};
Line(8)      =    {2,5};
Line(9)      =    {5,6};
Line(10)     =    {6,1};
Line(11)     =    {2,3};
Line(12)     =    {3,4};
Line(13)     =    {4,5};

// combine lines to loops for subsequent surfaces
Line Loop(14)={7,8,9,10};
Line Loop(15)={11,12,13,-8};

// set surfaces
Plane Surface(16)={14};
Plane Surface(17)={15};
Physical Surface("mixing")={16};
Physical Surface("main")={17};
```

This file has to be opened with *GMSH* then. To create the mesh click on `Mesh →
2d`, to save it as click on `File → Save As...` and save as *wetlandpre.msh*
(Fig. 7.3).

Now, you have to import the mesh file into the *OGS EXPLORER* to transform the format in a readable form for *OGS*. Open the *OGS EXPLORER* and click on `File → Import Files → GMSH Files...` to import *wetlandpre.msh*. Now click on the mesh name , then on `Tools → Remove Mesh Elements
→ Element Type → Line → OK` to remove any line elements. Right-click on the mesh name and on `Edit material groups... → Condense
material groups to smallest possible range → OK` to adjust
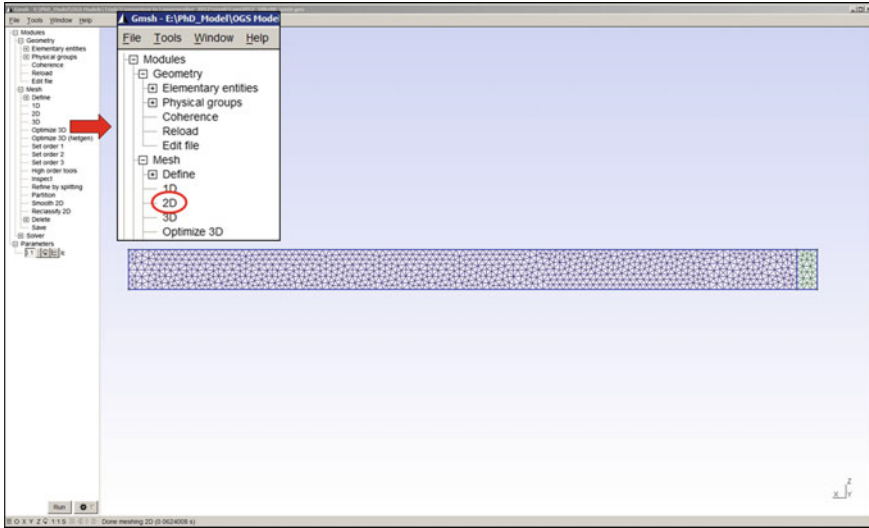
Fig. 7.3 Schematic of a horizontal subsurface flow treatment wetland

that material groups will start with number zero. Now click the symbol `Save mesh...` and save it as *wetland.msh*. The file *wetland.msh* should look like:

Listing 7.2 Finite element mesh file *wetland.msh*

```
#FEM_MSH                // Mesh keyword
$PCS_TYPE
NO_PCS
$NODES                  // list of nodes
928
0 0 0 0
1 0.3 0 0
2 10.3 0 0
...
$ELEMENTS               // list of elements
1636
0 0 tri 225 7 6
1 0 tri 233 225 6
2 0 tri 225 232 7
...
#STOP
```

Now, you have to create the *OGS* geometry file (*wetland.gli*) to define geometric entities for setting initial and boundary conditions later on . You have to define several points to set up the domain and an outflow point (`POINT3`) using `#POINTS`; an inflow line using `#POLYLINE`. To account for the initially water saturated part (and the unsaturated one) of the domain, you have to define two surfaces. This is done in two steps. At first, define a loop via `#POLYLINE`, then define a surface from the loop via `#SURFACE`. In all cases, you have to define the accuracy of the geometric entities with respect to the mesh nodes via `$EPSILON` (set it to 0.001).

Listing 7.3 OGS geometry file *wetland.gli*

```
#POINTS
0 0.00000000000000 0.00000000000000 0.00000000000000 $NAME POINT0
1 0.300000000000 0.00000000000000 0.00000000000000 $NAME POINT1
2 10.3000000000000 0.00000000000000 0.00000000000000 $NAME POINT2
3 10.3000000000000 0.00000000000000 0.05000000000000 $NAME POINT3
4 10.3000000000000 0.00000000000000 0.50000000000000 $NAME POINT4
5 10.3000000000000 0.00000000000000 0.60000000000000 $NAME POINT5
6 0.3000000000000 0.00000000000000 0.600000000000 $NAME POINT6
7 0.0000000000000 0.00000000000000 0.600000000000 $NAME POINT7
8 0.0000000000000 0.00000000000000 0.50000000000000 $NAME POINT8
9 0.0000000000000 0.00000000000000 0.3000000000000 $NAME POINT9
10 0.0000000000000 0.000000000000000 0.4000000000000 $NAME POINT10
11 0.3000000000000 0.00000000000000 0.5000000000000 $NAME POINT11

#POLYLINE                          // define a polyline by pre-defined points
$NAME
top_inflow
$EPSILON
0.001
$POINTS
6
7

#POLYLINE                          // define the loop for a surface
$NAME
pSaturated
$EPSILON
0.001
$POINTS
0
2
4
8
0

#POLYLINE
$NAME
pUnsaturated
$EPSILON
0.001
$POINTS
4
5
7
8
4

#SURFACE
$NAME
sUnsaturated
$EPSILON
0.001
$POLYLINES
pUnsaturated

#SURFACE                          // now define the surface form the loop
$NAME
sSaturated
$EPSILON
0.001
$POLYLINES
pSaturated
#STOP
```

7.4.1.2 Process Definition

All processes that will be simulated have to be listed in the file *wetland.pcs*. You have to define each process and its primary variable. That means, one process for water flow (RICHARDS_FLOW) and 19 processes for transport of solutes and particulates (MASS_TRANSPORT): the 16 components of CWM1 (Fig. 7.1) plus a conservative tracer, as well as, pH, and pe. It is always recommendable to include a conservative tracer for inspecting the flow behavior. PH and pe are necessary components for the coupling of *OGS* and IPhreeqc.

Listing 7.4 Process definition file *wetland.pcs*

```
#PROCESS
$PCS_TYPE
RICHARDS_FLOW
$PRIMARY_VARIABLE
PRESSURE1
$RELOAD
1 1

#PROCESS ; ConservativeTracer   comp 0
$PCS_TYPE
MASS_TRANSPORT

#PROCESS ; pH   comp 1
$PCS_TYPE
MASS_TRANSPORT

#PROCESS     ; pe    comp 2
$PCS_TYPE
MASS_TRANSPORT
...
#STOP
```

7.4.1.3 Time Discretization

The simulation time is set to 50 d with 2000 time steps of a constant time step size of 36 min (2160 s) for flow and transport. Note that you have to enter the values in seconds.

Listing 7.5 Time discretization file *wetland.tim*

```
#TIME_STEPPING
$PCS_TYPE
RICHARDS_FLOW
$TIME_START
0.0
$TIME_STEPS
2000 2160
$TIME_END
4320000
$TIME_UNIT
SECOND

#TIME_STEPPING
$PCS_TYPE
```

```
      MASS_TRANSPORT
      $TIME_START
      0.0
      $TIME_STEPS
      2000 2160
      $TIME_END
      4320000
      $TIME_UNIT
      SECOND

      #STOP
```

7.4.1.4   Numerical Methods Definition

To solve the differential equation system for flow and transport, numerical method has
to be defined for water flow and mass transport as follows (*wetland.num*). For water
flow (RICHARDS_FLOW), a linear and a nonlinear solver will have to be set; for trans-
port (MASS_TRANSPORT) a linear solver and mass lumping ($MASS_LUMPING)
are set.

Listing 7.6   Numerical methods file *wetland.num*

```
      #NUMERICS
      $PCS_TYPE
      RICHARDS_FLOW
      $LINEAR_SOLVER
      ; method error_tolerance max_iterations theta precond storage
      2 1    1.0e-014        100             1.0    100     4
      $NON_LINEAR_ITERATION
      ;type -- error_method -- max_iterations -- relaxation --
          tolerance(s)
      NEWTON ERNORM 10 0 1e-6
      $ELE_GAUSS_POINTS
      3

      #NUMERICS
      $PCS_TYPE
      MASS_TRANSPORT
      $LINEAR_SOLVER
      ; method error_tolerance max_iterations theta precond storage
      2       6 1.e-014          1000            0.5   1       2
      $ELE_GAUSS_POINTS
      3
      $ELE_MASS_LUMPING
      1
      $FEM_FCT
      2 1

      #STOP
```

7.4.1.5   Porous Media and Material Properties

All material input parameters have to be assigned using the different material input
files. The solid matrix of the wetland system consists of gravel with defined hydraulic

and material properties (Table 7.3). You have to create a file to account for the solid material (the gravel matrix) *wetland.msp* and one for the fluid properties *wetland.mfp*. Hydraulic properties of the solid matrix have to be written in the file *wetland.mmp*.

Listing 7.7 Solid properties file *wetland.msp*

```
#SOLID_PROPERTIES          // properties of the mixing zone
$DENSITY
1 1.8000e+003
#SOLID_PROPERTIES          // properties of the main zone
$DENSITY
1 1.8000e+003
#STOP
```

Listing 7.8 Fluid properties file *wetland.mfp*

```
#FLUID_PROPERTIES
$FLUID_TYPE
LIQUID
$DENSITY
1 0.9997026e+03
$VISCOSITY
1 1e-003
#STOP
```

Listing 7.9 Porous medium properties file *wetland.mmp*

```
#MEDIUM_PROPERTIES          // properties of the mixing zone
$GEOMETRY_DIMENSION
2
$GEOMETRY_AREA
5.3
$POROSITY
1 0.45
$PERMEABILITY_TENSOR
ISOTROPIC 4.166e-10
$PERMEABILITY_SATURATION
4   0.01 1.0 0.75
$CAPILLARY_PRESSURE
4 14.5
$MASS_DISPERSION
1   0.14 0.01
$DENSITY
1   1800.0
$TORTUOSITY
1 1.0

#MEDIUM_PROPERTIES          // properties of the main zone
$GEOMETRY_DIMENSION
2
$GEOMETRY_AREA
5.3
$POROSITY
1 0.41
$PERMEABILITY_TENSOR
ISOTROPIC 8.333e-011
$PERMEABILITY_SATURATION
4   0.1 1.0 0.75
$CAPILLARY_PRESSURE
4 14.5
$MASS_DISPERSION
```

Table 7.3 Porous Medium Properties

Parameter	Symbol	Value	Unit
Porosity$_{mixingzone}$	ϵ_{mix}	0.41	
Porosity$_{mainzone}$	ϵ_{main}	0.45	
Sat$_{residual}$	S_{res}	0.1	
Sat$_{max}$	S_{max}	1	
m	m	0.75	
alpha	α	14.5	
Permeability	k	8.33E-11	m^2
Dispersion$_{longitudinal}$	D_T	0.01	m^2 s^{-1}
Dispersion$_{transversal}$	D_L	0.14	m^2 s^{-1}
Tortuosity	τ	1	
Gravel density	ρ_{gravel}	1800	kg m^{-3}
Water density	ρ_{water}	999.7026	kg m^{-3}
Water viscosity	η	1.00E-03	pas

```
1  0.14 0.01
$DENSITY
1  1800.0
$TORTUOSITY
1 1.0


#STOP
```

7.4.1.6 Mass Transport Component Definition

All components that take part in transport and reactions have to be listed in the file
wetland.mcp: 16 biokinetic components (Fig. 7.1), a conservative tracer, pH, and
pe. You have to define a component name ($NAME), define if the component can
be transported or not ($MOBILE) and set the diffusion model and coefficient. All
components, except pH, pe and bacteria, have to be defined as mobile. This tutorial
does not consider diffusion processes–you have to set $DIFFUSION to zero for all
components.

Listing 7.10 RT component properties file *wetland.mcp*

```
    // conservative tracer

    #COMPONENT_PROPERTIES   ; comp 0
    $NAME
    Tracer
    $MOBILE
    1                     ;MOBILE-Flag: 0=imMOBILEe, 1=MOBILEe/transported
    $DIFFUSION
    0   0.0               ; diffusion model type, diffusion constant
```

```
    // Phreeqc necessary components

    #COMPONENT_PROPERTIES   ; comp 1
    $NAME
    pH
    $MOBILE
    0
    $DIFFUSION
    0   0.0

    #COMPONENT_PROPERTIES   ; comp 2
    $NAME
    pe
    $MOBILE
    0
    $DIFFUSION
    0   0.0


    // Reactive (biokinetic) components

    #COMPONENT_PROPERTIES   ; comp 13
    $NAME
    Sf
    $MOBILE
    1
    $DIFFUSION
    0 0.0

    #COMPONENT_PROPERTIES   ; comp 11
    $NAME
    Xh
    $MOBILE
    0
    $DIFFUSION
    0   0.0

    ...

    #STOP
```

7.4.1.7   Biokinetic Reaction Definition

All relevant information regarding the biokinetic reaction computations have to be
written in the input file for *PHREEQC wetland.pqc* and the corresponding database
file *phreeqc.dat*. At first, the reactive model components, to say reactants, that are
listed in the file *wetland.mcp* have to be defined in Phreeqc. This is done by adding
the relevant information (element name, species, name, alkalinity, molar mass, rela-
tion to secondary species) to the keywords SOLUTION_MASTER_SPECIES and
SOLUTION_SPECIES in the Phreeqc database file *phreeqc.dat*–not into the *wet-
land.pqc* file. The keyword CALCULATE_VALUES contains values of rate constants,
etc., and has to be added to the database *phreeqc.dat* as well. The temporal unit of
the reaction parameters in CWM1 is a day, as *PHREEQC* computation is done in
seconds, values have to be converted to second first.

Listing 7.11 Phreeqc database file *phreeqc.dat*

```
SOLUTION_MASTER_SPECIES          // define the master species
Tracer  Tracer     0     1    1
Do      Do         0     1    1
Sf      Sf    0    1     1
Sa      Sa         0     1    1
...

SOLUTION_SPECIES                 // define the secondary species
Tracer = Tracer
        log_k    0
Do = Do
        log_k     0
Sf = Sf
        log_k     0
...

CALCULATE_VALUES                 // define additional reaction parameters
Kh                               // Hydrolysis rate constant Kh
-start
1     value = 3                  // value in 1/d
2     value = value /(24*3600   // convert to 1/s
3     SAVE value
-end
...

END
```

The input file *wetland.pqc* itself should only contain the keywords: SOLUTION
which lists the reaction component that take part, KINETICS and RATES which
define the reaction network and rates; PRINT and SELECTED_OUTPUT which
define output formats and PUNCH which includes the output of all reaction com-
ponents. It is quite important to note that *a)* the model equations in CWM1 are
based on concentration ($g\,L^{-1}$), contrarily, computations in Phreeqc are based on
molality ($mol\,kg^{-1}$). The molar mass of all components is, therefore, set to 1.0 as a
work-around.

Listing 7.12 Phreeqc control file *wetland.pqc*

```
SOLUTION 1               // definition of the reactive components taking
    part in Phreeqc computations
units   mol/kgw       // units, temp, density, pH and pe
temp    20.000000
density 1
pH              7.0000 charge    # comp 1
pe              4.0              # comp 2
Do              9.0e-3           # comp 3
Sf              0.5e-3           # comp 4
Sa              1.03-3           # comp 5
Sin             0.5e-3           # comp 6
Snh             0.5e-3           # comp 7
Sno             40.0e-3          # comp 8
Sulfide         10.00e-3         # comp 9
Sso             10.0e-3          # comp 10
Xs              0.5e-3           # comp 11
Xi              0.1e-3           # comp 12
Xh              1.36e-4          # comp 13
Xa              1.36e-4          # comp 14
Xfb             1.36e-4          # comp 15
Xamb            1.36e-4          # comp 16
Xasrb           1.36e-4          # comp 17
```

```
Xsob              1.36e-4              # comp 18
-water         1                      # kg
#ende


KINETICS
#
Aerobic_Growth_Xh_on_Sf      // define a kinetic reactant
-formula  Do  -0.587301587301587 Sf  -1.58730158730159 Snh  -0.022380952
           Xh  1    # Do: 1-1/Y_h   Sf: -1/Y_h   Snh: v_5,2   Xh 1
-m0           1
-tol     1e-08
#ende

RATES           // define reaction rates of a kinetic reactant
#
Aerobic_Growth_Xh_on_Sf
-start

# get molalities and convert to concentration in g/l
10 Sf = MOL("Sf")
20 So2 = MOL("Do")
30 Sa = MOL("Sa")
40 Snh = MOL("Snh")
70 Sh2S = MOL("Sulfide")
90 Xh = MOL("Xh")

# compute reaction term for specific component
110 r_Sf = Sf/(CALC_VALUE("Ksf")+ Sf)
120 r_So2 = So2/(CALC_VALUE("Koh")+ So2)
130 r_Sa = Sf/(Sf + Sa)
140 r_Snh = Snh /(CALC_VALUE("Knhh")+ Snh)
170 r_Sh2s = CALC_VALUE("Kh2s_h") / (CALC_VALUE("Kh2s_h") + Sh2S )

# compute rate
200 rates = CALC_VALUE("muh") * r_Sf * r_So2 * r_Sa * r_Snh *  r_Sh2s *
     Xh
# compute mass change per time step
300 dGrowth= rates*time
310 dGrowth = dGrowth
400 SAVE dGrowth
-end
#ende

PRINT
-reset true
-selected_output true
#ende

SELECTED_OUTPUT
-file phout_sel.dat
-high_precision
-reset false
#ende

USER_PUNCH          // insert all reactive components
20 PUNCH TOT("Do"), TOT("Sf"), TOT("Sa"), TOT("Sin"), TOT("Snh"), TOT("
    Sno
    "), TOT("Sulfide"), TOT("Sso"), TOT("Xs"), TOT("Xi"), TOT("Xh"),
    TOT
    ("Xa"), TOT("Xfb"), TOT("Xamb"), TOT("Xasrb"), TOT("Xsob")
#ende

END
#STOP
```

You also have to define additional key parameters for the interphase of *OGS* to *PHREEQC* in the file *wetland.rei*. You have to define the unit of molality (moles per weight, $MOL_PER), a constant water concentration, as well as, the pressure and temperature to 1 bar and 293.15 K.

Listing 7.13 Reaction interface file *wetland.rei*

```
#REACTION_INTERFACE
$MOL_PER                    // set molality mol/kg
WEIGHT    ; VOLUME
$WATER_CONCENTRATION
CONSTANT  ; VARIABLE
$PRESSURE
CONSTANT 1.0   ; bar
$TEMPERATURE
CONSTANT 293.15   ; Kelvin

#STOP
```

7.4.1.8 Initial Conditions

It is assumed that an initial water table 0.5 m exists (saturation of 0.5 m). This results in a linear pressure distribution from the top to the bottom of the saturated layer (sSaturated, define in *wetland.gli*) and has to be considered as initial conditions for the water pressure. Initial concentrations for the nonreactive tracer is set to zero in the DOMAIN. Initial concentrations of Sf, Sa, Sin, Snh, Sso, Xs, and Xi are similar to the boundary conditions (Table 7.4) while all other reactive components, except pH (set to 7.0) and pe (4.0), are set to $0.1\,\mathrm{mg\,L^{-1}}$.

Listing 7.14 Initial conditions file *wetland.ic*

```
#INITIAL_CONDITION
$PCS_TYPE
RICHARDS_FLOW
$PRIMARY_VARIABLE
PRESSURE1
$GEO_TYPE
SURFACE sSaturated
$DIS_TYPE
GRADIENT 1 0 4905

#INITIAL_CONDITION
$PCS_TYPE
RICHARDS_FLOW
$PRIMARY_VARIABLE
PRESSURE1
$GEO_TYPE
SURFACE sUnsaturated
$DIS_TYPE
CONSTANT 0.0

#INITIAL_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
Tracer
```

```
$GEO_TYPE
DOMAIN
$DIS_TYPE
CONSTANT  0.0

#INITIAL_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
Do
$GEO_TYPE
DOMAIN
$DIS_TYPE
CONSTANT  0.86e-3

...
#STOP
```

7.4.1.9   Boundary Conditions and Source Terms

Boundary conditions and source terms have to be added for individual processes, by specifying the process type ($PCS_TYPE) and its primary variable ($PRIMARY _VARIABLE). At the inflow (POLYLINE top_inflow), there is one *Dirichlet* boundary condition for water pressure (0.0 Pa), as well as, a *Dirichlet* condition for each solute reactant, as well as, slowly biodegradable COD (Xs) and inert particulate COD (Xi) (Table 7.4). Another *Dirichlet* condition fixes the pressure of water flow in the middle of the outflow (POINT3) at a head of 0.45 m (4415 Pa). The loading rate of the wetland is 36.0 mm d^{-1}, this translates into a source term of 7.5819 m^{2}s^{-1}

Table 7.4 Model Boundary Conditions

Parameter	Symbol	Value	Unit
Influent load	q	36.0	mm d^{-1}
Conservative tracer	Tracer	2.0	g L^{-1}
Dissolved oxygen	Do	0.86e-03	g L^{-1}
Readily biodegradable organic matter	Sf	170.0e-03	g L^{-1}
Organic matter as Acetate	Sa	27.0e-03	g L^{-1}
Inert soluble organic matter	Sin	31.0e-03	g L^{-1}
Ammonia nitrogen	Snh	57.0e-03	g L^{-1}
Sulfate sulfur	Sso	34e.0-03	g L^{-1}
Slowly biodegradable particulate Carbon	Xs	33.0e-03	g L^{-1}
Inert particulate organic matter	Xi	13.0e-03	g L^{-1}

along the inflow (POLYLINE top_inflow) defined in the file *wetland.st*. The width of the inflow is automatically taken into consideration with the defined width under $GEOMETRY_AREA in the file *wetland.mmp*.

Listing 7.15 Boundary conditions file *wetland.bc*

```
// BC water flow

#BOUNDARY_CONDITION
$PCS_TYPE
RICHARDS_FLOW
$PRIMARY_VARIABLE
PRESSURE1
$GEO_TYPE
POINT POINT3
$DIS_TYPE
CONSTANT 4415

#BOUNDARY_CONDITION
$PCS_TYPE
RICHARDS_FLOW
$PRIMARY_VARIABLE
PRESSURE1
$GEO_TYPE
POLYLINE top_inflow
$DIS_TYPE
CONSTANT 0


// BC Mass Transport

#BOUNDARY_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
Do
$GEO_TYPE
POLYLINE top_inflow
$DIS_TYPE
CONSTANT 0.86e-3

...
#STOP
```

Listing 7.16 Source term file *wetland.st*

```
#SOURCE_TERM
$PCS_TYPE
RICHARDS_FLOW
$PRIMARY_VARIABLE
PRESSURE1
$GEO_TYPE
POLYLINE top_inflow
$DIS_TYPE
CONSTANT_NEUMANN    7.75819-05

#STOP
```

The component tracer is added as an impulse at simulation start. Therefore, the boundary condition of this component is time dependent. You will have to define a boundary condition in the file *wetland.bc* with the sub-keyword $TIME_TYPE

and an additional file *wetland.rfd* which contains information about the boundary condition values at individual time steps. Simply, the value in *wetland.bc* is multiplied by the values of the referred curve in *wetland.rfd*. Here, 98 g of tracer will be added at simulation start. That means $2 \, \mathrm{g} \, \mathrm{L}^{-1}$ of tracer will have to be added in the first time step (0–2160 s).

Listing 7.17 Boundary condition for the tracer in *wetland.bc*

```
#BOUNDARY_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
Tracer
$GEO_TYPE
POLYLINE top_inflow
$DIS_TYPE
CONSTANT 2.0
$TIM_TYPE        // activates time dependent bc
CURVE 1          // use values of curve 1 from *.rfd file

#STOP
```

Listing 7.18 Time series of the tracer boundary condition *wetland.rfd*

```
#CURVES
0  1      // bc at time 0 sec is one
2160 1      // bc at time 2160 sec is one
4320 0      // bc at time 4320 sec is zero
43200000 0  // bc at time 43200000 sec is still zero

#STOP
```

7.4.1.10 Output Format Definition

Finally, the output data for subsequent visualization has to be defined in the file *wetland.out*. Values of the pressure, water velocity, and all reactive components at every mesh node of the domain at every time step have to be exported. You have to define an output block for each process (here: either RICHARDS_FLOW or MASS_TRANSPORT); listing each variable under $NOD_VALUES to export its nodal values; defining the geometric entity the nodes belong to ($GEO_TYPE) and defining the time steps increment at which output values should be written ($TIME_TYPE)

Listing 7.19 Output file *wetland.out*

```
#OUTPUT
$PCS_TYPE
RICHARDS_FLOW
$NOD_VALUES
PRESSURE1
SATURATION1
VELOCITY_X1
VELOCITY_Y1
VELOCITY_Z1
$GEO_TYPE
```

```
      DOMAIN
    $DAT_TYPE
      PVD
    $TIM_TYPE
      STEPS 1

    #OUTPUT
    $PCS_TYPE
      MASS_TRANSPORT
    $NOD_VALUES
      Tracer
      Do
      Sf
      Sa
      Sin
      Snh
      Sno
      Sulfide
      Sso
      Xs
      Xi
      Xh
      Xa
      Xfb
      Xamb
      Xasrb
      Xsob
    $GEO_TYPE
      DOMAIN
    $DAT_TYPE
      PVD
    $TIM_TYPE
      STEPS 1

    #STOP
```

7.4.2 Model Execution

You have to put all input files into one directory. It is mandatory to name the files all the same, otherwise, the simulation will not start as files with other names than the specified will not be considered for program execution. Use the name *wetland*. Additionally, the *OGS* executable *ogs.exe* has to be added. At the time of this writing, it is recommended to use the *OGS* version 5.5. It is recommended to run a simulation project as this one in several steps with increasing complexity. At first, to establish a steady-state water flow field run the model for a few time steps with a smaller time step size and without MASS_TRANSPORT processes. To exclude MASS_TRANSPORT processes, simply leave all the MASS_TRANSPORT blocs out of the process file *wetland.pcs*, time step file *wetland.tim* and numeric file *wetland.num*; remove all IC, BC and output values related to mass transport; remove the files *wetland.mcp* and *wetland.pqc*. Then, change the number of time steps and the size for RICHARDS_FLOW under the keyword $TIME_STEPS in the file *wetland.tim* to ten and 360, then save the file. Now just click on the *ogs.exe*, the command line should open, type wetland and press ENTER (Fig. 7.4). *OGS* will run and subsequently create output files. Check
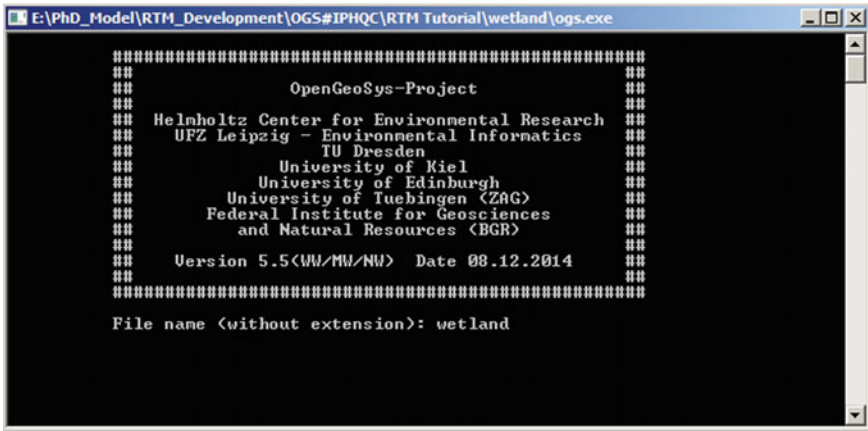
Fig. 7.4 Title view of *ogs.exe*, enter wetland and press ENTER
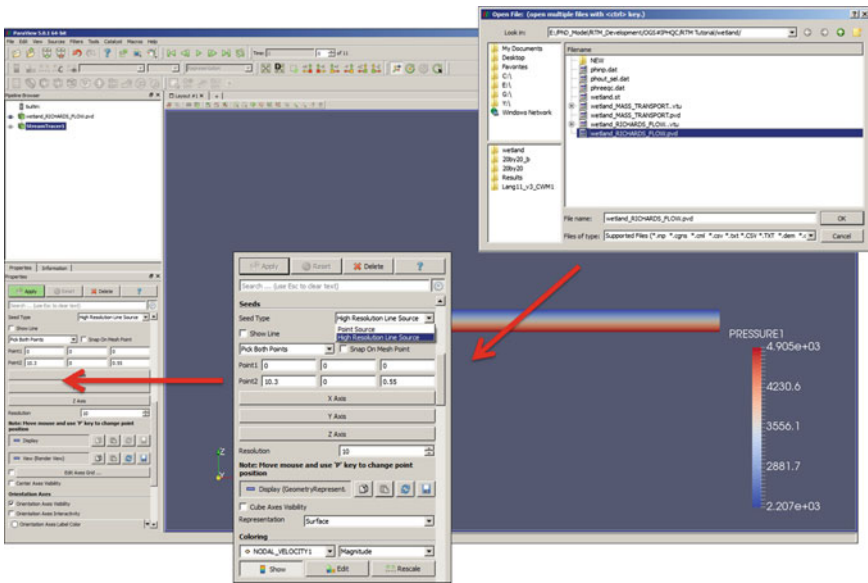
Fig. 7.5 Visualization of water flow in *ParaView*

the development of a steady-state flow field by executing the steps in the subsequent chapter (compare your results with Fig. 7.8).

If the model runs into steady state, you can retype the time stepping to the prior defined conditions (TIME_STEPS 100 and 4320) and change the values of the keyword $RELOAD under RICHARDS_FLOW in the process file *wetland.pcs* to two. This tells *OGS* to use the results of the last time step of the previous simulation
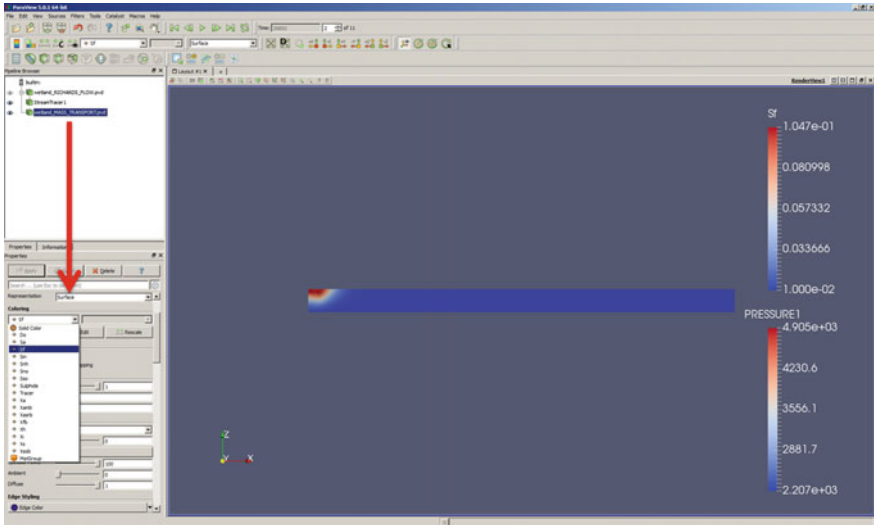
Fig. 7.6 Visualization of substrate and bacteria distribution in *ParaView*
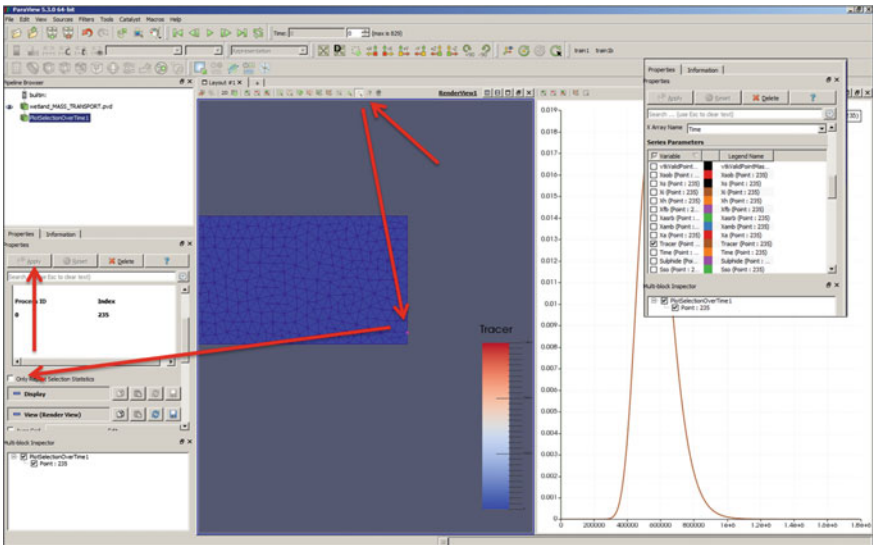
Fig. 7.7 Plot times series of individual nodes in *ParaView*

computed as starting conditions, to say, *OGS* now bypasses computing initial conditions. Now include the MASS_TRANSPORT processes again. The second run should include transport of all components but still exclude biokinetic reactions to verify the transport model. To achieve this, simply leave the Phreeqc input file *wetland.pqc* out of the model folder. If you now click on the *ogs.exe*, type wetland and press ENTER,
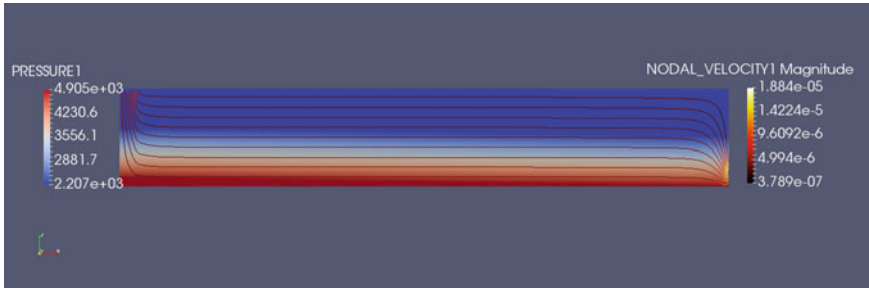
Fig. 7.8 Pressure distribution and streamlines in *ParaView*

the transport model will be started. The time series of the tracer at a point close to the outlet should be similar as in Fig. 7.11. All other components should exhibit a horizontal line. Finally, copy the *wetland.pqc* file back into the model project folder and execute the model again. Now, biokinetic reactions will be included. You will notice this in the output of *ogs.exe*...and in the computation time as well, ;-).

Listing 7.20 Change $TIME_STEPS to 10 and the step size to 360 to compute steady-state flow *wetland.tim*

```
#TIME_STEPPING
$PCS_TYPE
RICHARDS_FLOW
$TIME_START
0.0
$TIME_STEPS
10 360

...

#STOP
```

Listing 7.21 Change $RELOAD to 2 after computing steady-state water flow *wetland.pcs*

```
#PROCESS
$PCS_TYPE
RICHARDS_FLOW
$PRIMARY_VARIABLE
PRESSURE1
$RELOAD
2

...
#STOP
```

7.4.3   Post-processing

If the simulation is finished, open the program *ParaView*. Click on `File` →
`Open` choose the directory and click on the file *wetland_RICHARDS_FLOW.pvd*
(Fig. 7.5). Now, the filename should appear under `builtin:` in the pipeline browser.
Click on `Set view direction to +Y` in the toolbar menu to set the right
perspective. Then click on `wetland_RICHARDS_FLOW.pvd` in the pipeline
browser and on `Stream Tracer` in the toolbar menu. Scroll down to `Seeds`
in the `Properties` field and set `Seed Type` → `High Resolution Line`
`Source`, set `Resolution` → `10` and uncheck the tick mark `Show Line`, click
on `Apply`. Now click on `Play` in the toolbar menu. You should see the simulation
progress in the model domain, with developing distribution of *PRESSURE1* and a
streamline field. The streamlines should look like in Fig. 7.8.

To visualize the distribution of substrate components and bacteria types, again,
click on `File` → `Open` choose the directory and click on the file *wetland_MASS*
_TRANSPORT.pvd (Fig. 7.6). Now, click on *wetland_MASS_ TRANSPORT.pvd* in
the pipeline browser and choose the coloring of interest in the `Properties` field.
With `Coloring`, you can select the `MASS_TRANSPORT` component to visualize.
For screenshots click on `File` → `Save screenshot`.

If you want to plot concentration time series of certain components, such as the
tracer, click on `Interactive Select Points On` in the taskbar at the top
of the layout window (Fig. 7.7). Now, select the first node above the corner at the
bottom right, then click on `Plot Selection Over Time` and unmark the tick
at `Report Selection Statistics Only` in the `Properties` field and
click on `Apply`. A new view showing the times series of all components will show
up. Simply scroll down the `Properties` field and select or unselect your desired
components (Fig. 7.8).

7.5   Simulation Results

The tutorial simulation consists of the fate of a conservative tracer, as well as, several
bacteria and substrate types in a passive horizontal subsurface flow treatment wet-
land over 50 d. At the end of the simulation, different patterns of bacterial groups and
substrate components develop within the simulated system (Figs. 7.9 and 7.10). The
tracer time series shows a breakthrough curve typically observed in subsurface flow
treatment wetlands and can be interpreted as a verification of the transport model
due to the absence of any visible oscillations (Fig. 7.9). Temporal development of the
effluent concentrations show an initial transition into a rather steady state in the first
10 d (Fig. 7.11). The reason is that the wetland domain was filled with wastewater
similar to the influent at simulation start-up. The initial Do, Sf, and Xs in the domain
were then rapidly consumed by heterotrophic and fermenting bacteria resulting in
a decrease of these substances and an increase of Sa in the effluent. After 50 d of
simulation bacteria mainly developed at the inflow region (Fig. 7.10), as this is the
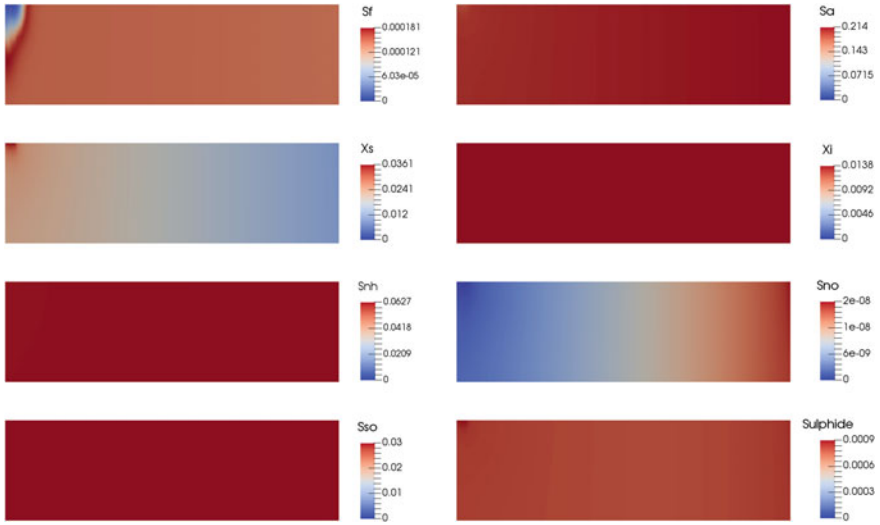
Fig. 7.9 Substrate component distribution across the modeled wetland domain after 50 d of simulation
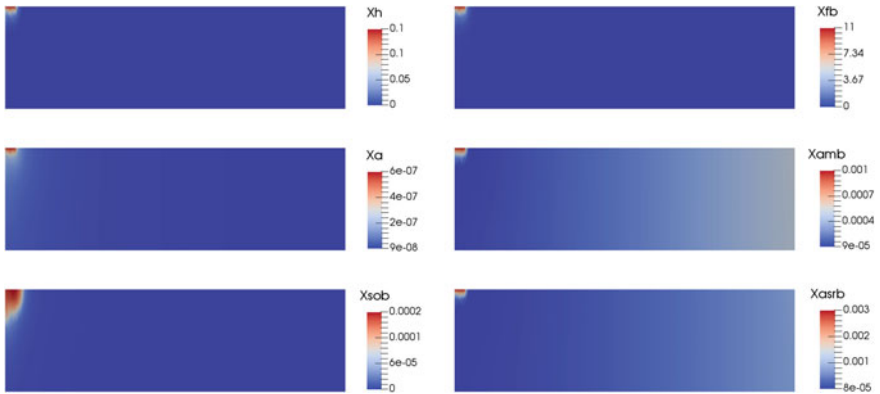
Fig. 7.10 Bacteria distribution across the modeled wetland domain after 50 d of simulation

source of new substrate. However, fermenting bacteria are dominating the population due to their independence of oxygen and fast growth rate among the anaerobe bacteria. Sa, as the fermentation product of Xfb, is the predominant form of COD in the whole wetland domain and still present in high concentrations throughout the wetland. It shows that the capacity of methanogenic bacteria (Xamb) is not enough to mineralize all available Sa. This is probably due to the low growth rate of Xamb and the short simulation time of 50 d only. Bacterial patterns would look different if the simulation will cover a longer period. In general, the microbial character of the simulated wetland can be described as anaerobic which is typical for passive horizontal subsurface flow wetlands used for secondary wastewater treatment.
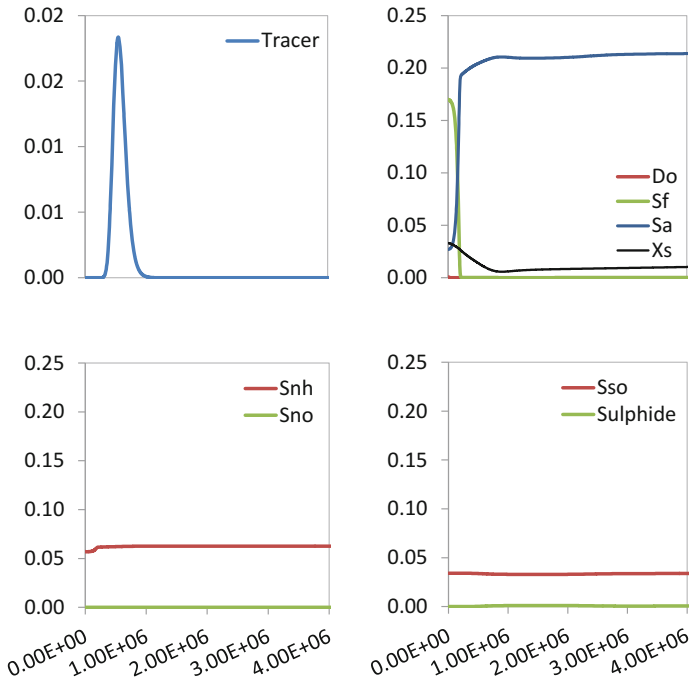
Fig. 7.11 Development of the effluent quality during 50 d of simulation

References

J. García, P. Aguirre, J. Barragán, R. Mujeriego, V. Matamoros, J.M. Bayona, Effect of key design parameters on the efficiency of horizontal subsurface flow constructed wetlands. Ecol. Eng. **25**(4), 405–418 (2005), https://doi.org/10.1016/j.ecoleng.2005.06.010

E. Llorens, M.W. Saaltink, J. García, CWM1 implementation in retrasocodebright: first results using horizontal subsurface flow constructed wetland data. Chem. Eng. J. **166**(1), 224–232 (2011), https://doi.org/10.1016/j.cej.2010.10.065

G. Langergraber, J. Simunek, Reactive transport modeling of subsurface flow constructed wetlands using the hydrus wetland module. Vadose Zone J. **11** (2) (2012), https://doi.org/10.2136/vzj2011.0104

WWAP, The united nations world water development report 2017: Wastewater: the untapped resource (2017)

R.H. Kadlec, S.D. Wallace, *Treatment Wetlands*, 2nd edn. (CRC Press, Boca Raton, 2009)

S. Wu, S. Wallace, H. Brix, P. Kuschk, W.K. Kirui, F. Masi, R. Dong, Treatment of industrial effluents in constructed wetlands: challenges, operational strategies and overall performance. Environ. Pollut. **201**, 107–120 (2015), https://doi.org/10.1016/j.envpol.2015.03.006

J. Vymazal, T. Brezinova, The use of constructed wetlands for removal of pesticides from agricultural runoff and drainage: a review. Environ. Int. **75**, 11–20 (2015), https://doi.org/10.1016/j.envint.2014.10.026, http://www.sciencedirect.com/science/article/pii/S0160412014003201

M. van Afferden, K.Z. Rahman, P. Mosig, C. De Biase, M. Thullner, S.E. Oswald, R.A. Müller, Remediation of groundwater contaminated with mtbe and benzene: the potential of vertical-flow

soil filter systems. Water Res. **45**(16), 5063–5074 (2011), https://doi.org/10.1016/j.watres.2011.07.010

J. Vymazal, Constructed wetlands for wastewater treatment. Water **2**(3), 530–549 (2010), https://doi.org/10.3390/w2030530, http://www.mdpi.com/2073-4441/2/3/530/htm

G. Langergraber, Numerical modelling: a tool for better constructed wetland design? Water Science and Technology **64**(1), 14–21 (2011), https://doi.org/10.2166/wst.2011.520

G. Langergraber, Applying process-based models for subsurface flow treatment wetlands: recent developments and challenges. *Water*, **9**(1) (2017), https://doi.org/10.3390/w9010005

R. Samso, J. Garcia, Bio_pore, a mathematical model to simulate biofilm growth and water quality improvement in porous media: application and calibration for constructed wetlands. Ecological Engineering **54**, 116–127 (2013a), https://doi.org/10.1016/j.ecoleng.2013.01.021, http://www.sciencedirect.com/science/article/pii/S0925857413000360

R. Samso, J. Garcia, Bacteria distribution and dynamics in constructed wetlands based on modelling results. Sci. Total Environ. **461**, 430–440 (2013b)

G. Langergraber, D.P.L. Rousseau, J. Garcia, J. Mena, Cwm1: a general model to describe biokinetic processes in subsurface flow constructed wetlands. Water Sci. Technol. **59**(9), 1687–1697 (2009), https://doi.org/10.2166/wst.2009.131

G. Langergraber, J. Simunek, Modeling variably saturated water flow and multicomponent reactive transport in constructed wetlands. Vadose Zone J. **4**(4), 924–938 (2005). ISSN 1539-1663, https://doi.org/10.2136/vzj2004.0166

J. García, D.P.L. Rousseau, J. Morató, E. Lesage, V. Matamoros, J.M. Bayona, Contaminant removal processes in subsurface-flow constructed wetlands: a review. Crit. Rev. Enviro. Sci. Technol. **40**(7), 561–661 (2010), https://doi.org/10.1080/10643380802471076

O. Kolditz, H. Shao, U.J. Görke, T. Kalbacher, S. Bauer, C.I. McDermott, W. Wang, Thermo-hydro-mechanical-chemical processes in fractured-porous media: Benchmarks and examples, in AGU Fall. Meeting Abstracts **1**, 1161 (2012b)

J. Nivala, T. Headley, S. Wallace, K. Bernhard, H. Brix, M. V. Afferden, R. A. Müller, Comparative analysis of constructed wetlands: The design and construction of the ecotechnology research facility in Langenreichenbach, Germany. Ecological Engineering **61**, 527–543 (2013), https://doi.org/10.1016/j.ecoleng.2013.01.035

Appendix A
OpenGeoSys v5 Keywords Description

This section provides a wrap-up compendium of the OGS v5 keywords used in this tutorial. A more comprehensive compilation of the keywords with examples you can find at www.opengeosys.org/help/documentation/ and https://svn.ufz.de/ogs/wiki/public/doc-auto.

A.1 PCS—Process Definition

Keyword	#PROCESS
	Indicate the beginning of a PCS block. Each block identifies one process and its primary variable. It ends with the beginning of a new PCS block, i.e., #PROCESS, or with the file input terminator #STOP
Sub-keyword	$PCS_TYPE
	Define the definition of the process. Only one parameter can be defined per PCS block
Parameter	char
	The parameter is defined one line below the $PCS_TYPE. The keywords and according primary variables are listed as below
	GROUNDWATER_FLOW HEAD
	LIQUID_FLOW PRESSURE1
	RICHARDS_FLOW PRESSURE1
	AIR_FLOW PRESSURE1, TEMPERATURE1
	MULTI_PHASE_FLOW PRESSURE1, PRESSURE2
	PS_GLOBAL PRESSURE1, SATURATION2
	HEAT_TRANSPORT TEMPERATURE1
	DEFORMATION DISPLACEMENT_X1, DISPLACEMENT_Y1, DISPLACEMENT_Z1
	MASS_TRANSPORT Solute species which defined in the component property file (*.mcp)
	OVERLAND_FLOW HEAD
	FLUID_MOMENTUM VELOCITY1_X, VELOCITY1_Y, VELOCITY1_Z
	RANDOM_WALK with particles in *.pct file

Listing A.1 PCS keyword

```
    # PROCESS             // process keyword
    $ PCS_TYPE            //process type sub-keyword
    GROUNDWATER_FLOW     // specified process

    # PROCESS
    $ PCS_TYPE            // process type sub-keyword
    MASS_TRANSPORT       // specified process
    # STOP               // end of input data
```

A.2   GLI—Geometry Condition

Keyword	#POINTS
	A list of points including the point number and the Cartesian coordinates is defined under #POINT. Optionally, a point name can be assigned to each point with the sub-keyword $NAME
Parameters	int double double double
	The first parameter defines the point number, and the following three values are the Cartesian coordinates X Y Z.
Sub-keyword	$NAME
	Define a name of points. Note that this sub-keyword is only required if the specific points is used for boundary and initial conditions, source terms, and output.
Parameters	char
	Names can be written after the coordinate definition with the sub-keyword $NAME
Keyword	#POLYLINE
	Define a topological description of a polyline or polygon. Each polyline definition includes a name and a list of point's number. Polygons are defined in the same way as polylines, with the only difference that the fist and last point are identical.
Sub-keyword	$NAME
	Defines a name of polyline.
Parameters	char
	The names are written in the subsequent line under $NAME.
Sub-keyword	$POINTS
	The order of points defines the topology, i.e., the point-to-point connection.
Parameter	Points of polylines
Keyword	#SURFACE
	Surfaces are built from polylines specified in sub-keyword $POLYLINE.
Sub-keyword	$NAME
	Defines a name of surface.
Parameter	char
	The names are written in the subsequent line under $NAME.
Sub-keyword	$POLYLINE
	Surfaces should be completely closed by the set of polylines.
Parameter	char
	The name of the polyline is written in the subsequent line under $POLYLINE.

Listing A.2 GLI keyword

```
#POINTS               // points keyword
0 0 0 0 $NAME POINT0  // point number | x | y | z | point name
1 1 0 0 $NAME POINT1
2 1 1 0 $NAME POINT2
[...]
#POLYLINE             // polyline keyword
$NAME                 // polyline name sub-keyword
TOP                   // polyline name
$POINTS               // polyline points sub-keyword
0                     // point of polyline
1                     // dito
[...]
#POLYLINE             // polyline keyword
$NAME                 // polyline name sub-keyword
BOUNDARY              // polyline name
$POINTS               // polyline points sub-keyword
0                     // point ofpolyline
1                     // dito
2                     // dito
0                     // dito; polygon, first and last point are
    identical
#SURFAE               // surface keyword
$NAME                 // surface name sub-keyword
catchment             // name of surface
$POLYLINES            // surface polyline sub-keyword
BOUNDARY              // polyline of surface
#STOP                 // end of input data
```

A.3 MSH—Finite Element Mesh

Keyword	#FEM_MSH
	The header of the mesh file.
Sub-keywords	$NODES
	Define the definition of all node number and coordinates (X Y Z).
Parameter	int double double double
	The first parameter (int) defines the node index. The numbering starts always with "0" and must be completed and without gaps. The following three values (double double double) are the Cartesian coordinates.
Sub-keyword	$ELEMENTS
	Introduces the topological definition of element types and their connectivity
Parameter	int int char int int int int int int int int
	The first parameter (int) defines the element index. The numbering starts always with "0" and must be complete and without gaps. The second parameter (int) assigns a material number to the element. Therefore, the element is directly linked with the material properties given in the material medium property file (*.mmp). Again, we start counting from zero, i.e., "0" is linked with the first material property definition of the MMP file. The third parameter (char) defines the element type. OGS supports the following element type keywords: line (line) tri (triangle) quad (rectangle) tet (tetrahedra) pris (prism) hex (hexahedra).

Listing A.3 MSH keyword

```
#FEM_MSH        // file/object keyword
$NODES          // node sub-keyword
61              // number of grid nodes
0 0 0 0         // node number x y z
1 0 0 1         // dito
[...]
59 0 0 59
60 0 0 60
$ELEMENTS       // element sub-keyword
60              // number of elements
0 0 line 0 1    //element number|material group|type|element node numbers
1 0 line 1 2    // dito
[...]
59 0 line 59 60   // dito
#STOP           // end of input data
```

A.4   NUM—Numerical Properties

Keyword	#NUMERICS
	Define the beginning of process-related numerical settings. It ends with the beginning of a new NUM block, i.e., #NUMERICS, or with the file input terminator #STOP.
Sub-keyword	$PCS_TYPE
	Defines specification of the process.
Parameter	Only one parameter can be defined here. The possible parameters are the OGS process type keywords (see process types listed at the PCS Sect. A.1). In Listing A.1, only groundwater flow is considered.
Sub-keyword	$LINEAR_SOLVER
	Set up a numerical solver for linear problems.
Parameters	int int double int double int int
	The first parameter int specifies the types of the linear solver, i.e., the solver method that will be used to solve the above-defined processes. Linear type is given in Table A.1
	The second parameter int enables to define a criterion for convergence
	1. Absolutely error $\|\|r\|\| < \epsilon$ 2. $\|\|r\|\| < \epsilon\|\|b\|\|$ 3. $\|\|rn\|\| < \epsilon\|\|rn-1\|\|$ 4. if $\|\|rn\|\| < 1$ then $\|\|rn\|\| < \epsilon\|\|rn-1\|\|$ else $\|\|r\|\| < \epsilon$ 5. $\|\|rn\|\| < \epsilon\|\|x\|\|$ 6. $\|\|rn\|\| < \epsilon \max \|\|rn-1\|\|, \|\|x\|\|, \|\|b\|\|$
	The third parameter enables to define an error tolerance ϵ
	The fourth parameter is the maximal number of linear solver iterations one time step
	The fifth parameter is theta, a relaxation number for the Laplacian in the range of $\theta \in [0, 1]$
	The sixth parameter allows setup preconditioning. The available preconditioners are: 0 No preconditioner, 1 Jacobi preconditioner, 100 ILU preconditioner.
	The seventh parameter defines the storage model as: 2 unsymmetrical matrix, 4 symmetrical matrix.

Listing A.4 NUM keyword

```
#NUMERICS           // process keyword
$PCS_TYPE           // process type sub-keyword, see PCS above
GROUNDWATER_FLOW
$LINEAR_SOLVER      // linear solver type sub-keyword, see table below
2 5 1.e-014 1000 1.0 100 4        //
#STOP               // end of input data
```

A.5   TIM—Time Discretization

Keyword	#TIME_STEPPING
	Define the beginning of the time stepping specifications. It ends with the beginning of a new TIM block, i.e., #TIME_STEPPING, or with the file input terminator #STOP.
Sub-keyword	$PCS_TYPE
	Here, it is required to define to which simulated process the time step scheme belongs.
Parameter	The possible parameters are the OGS process type keywords (see process types listed at the PCS Sect. A.1).
Sub-keyword	$TIME_STEPS
	Define the number and size of time step.
Parameter	The first parameter defines the number of time steps and the second values is the time step size. Listing A.5 shows the definition of 1000 time steps, each with a length of 1 day.
Sub-keyword	$TIME_STARTS
	Set the beginning of the simulation time.
Parameter	The parameter must be written in the subsequent line after the keyword. In Listing A.5, simulation starts at the time 0.0
Sub-keyword	$TIME_END
	Define when simulation ends. The simulations stops at this specific time only if enough time steps have been defined by $TIME_STEPS, otherwise it ends with the last defined time step.
Parameter	The parameter is defined after the keyword. In Listing A.5, the simulation ends after 1000 days.
Sub-keyword	$TIME_UNIT
	Define the time unit. The default setting is SECOND.
Parameter	The possible time unit is SECOND, MINUTE, HOUR, DAY, MONTH, or YEAR. In Listing A.5, the time unit is day.

Listing A.5 TIM keyword

```
#TIME_STEPPING      // timt stepping keyword
$PCS_TYPE           // process sub-keyword
GROUNDWATER_FLOW    // specified process
$TIME_STEPS         // time steps sub-keyword
1000 1     // number of times steps | times step length
$TIME_START         // starting time sub-keyword
0.0                 // starting time value
$TIME_END           // end time sub-keyword
1000                // end time value
$TIME_UNIT          // specified time unit
DAY                 // SECOND, DAY, YEAR
#STOP               // end of input data
```

Table A.1 Linear solver

1st parameter	Solver
1	SpGAUSS, direct solver
2	SpBICGSTAB
3	SpBICG
4	SpQMRCGSTAB
5	SpCG
6	SpCGNR
7	CGS
8	SpRichard
9	SpJOR
10	SpSOR

A.6 IC—Initial Condition

Keyword	#INITIAL_CONDITION
	Each component needs one block. It ends with the beginning of a new IC block or with the file input terminator #STOP.
Sub-keywords	$PCS_TYPE
	Define the process
Parameter	The possible processes are listed in Sect. A.1
Sub-keyword	$PRIMARY_VARIABLE
	Define the primary variable
Parameter	The available primary variables are listed in Sect. A.1
Sub-keyword	$GEO_TYPE
	Define the geometric entity of the source term
Parameter	Type (POINT, POLYLINE, SURFACE, DOMAIN) and name such as POINT1
Sub-keyword	$DIS_TYPE
	Define how data are distributed over the defined geometric object, e.g., domain or surface, i.e., a certain defined value or the average of the value can be assigned to each node. There are several types (CONSTANT, GRADIENT)
Parameter	chr double
	CONSTANT defines a constant value
	GRADIENT defines a gradient by entering a relationship, a start, and an end value.

Listing A.6 IC keyword

```
#INITIAL_CONDITION
$PCS_TYPE           // process type sub-keyword
RICHARDS_FLOW       // specified process
$PRIMARY_VARIABLE   // primary variable sub-keyword
PRESSURE1               // specified primary variable
$GEO_TYPE           // geometry type sub-keyword
DOMAIN              // specified geometry type | geometry name
$DIS_TYPE           // initial condition type sub-keyword
GRADIENT 1 0 9810   // initial condition type | model (linear) | start value | end
    value

#INITIAL_CONDITION
$PCS_TYPE
MASS_TRANSPORT
$PRIMARY_VARIABLE
Tracer              // specified primary variable
$GEO_TYPE           // geometry type sub-keyword
DOMAIN                  // specified geometry type | geometry name
$DIS_TYPE              // initial condition type sub-keyword
CONSTANT   0   // initial condition type | value
[...]
#STOP
```

A.7 BC—Boundary Condition

Keyword	#BOUNDARY_CONDITION
	Define the beginning of a BC file description. It ends with the beginning of a new BC block or with the file input terminator #STOP.
Sub-keyword	$PCS_TYPE
	Define the process for which to add a source term
Parameter	see PCS
Sub-keyword	$PRIMARY_VARIABLE
	Define the primary variable for which to add a source term
Parameters	see PCS
Sub-keyword	$GEO_TYPE
	Define the geometric entity of the source term
Parameter	chr chr
	Type (POINT, POLYLINE, SURFACE, DOMAIN) and name such as POINT1
Sub-keyword	$DIS_TYPE
	Define how the bc is distributed over the defined geometric object
	see IC
Sub-keyword	$TIM_TYPE
	Define a time-dependent boundary condition, such as a tracer injection at time X.
Parameter	chr int
	CURVE and a number indicating the time function defined in the file *.rfd*. The value of the bc will be multiplied with the value defined in the *.rfd* file at the specified time steps.

Note that CONSTANT_NEUMANN cannot be used for boundary conditions.

Listing A.7 BC keyword

```
#BOUNDARY_CONDITION
$PCS_TYPE             // process type sub-keyword
GROUNDWATER_FLOW      // specified process
$PRIMARY_VARIABLE     // primary variable sub-keyword
HEAD                  // specified primary variable
$GEO_TYPE             // geometry type sub-keyword
POLYLINE RIGHT        // specified geometry type | geometry
    name
$DIS_TYPE             // boundary condition type sub-keyword
CONSTANT   10.5       // boundary condition type | value

#BOUNDARY_CONDITION
$PCS_TYPE             // process type sub-keyword
MASS_TRANSPORT
$PRIMARY_VARIABLE
Tracer
$GEO_TYPE             // geometry type sub-keyword
POLYLINE TOP          // specified geometry type | geometry
    name
$DIS_TYPE             // boundary condition type sub-keyword
CONSTANT  1.5e-2      // boundary condition type | value
$TIM_TYPE             // sub-keyword for time dependent bc
CURVE 1               // define the time function to use
[...]

#STOP
```

A.8   ST—Source/Sink Terms

Keyword	#SOURCE_TERM
	Start new source term block. It ends with the beginning of a new ST block or with the file input terminator #STOP.
Sub-keywords	$PCS_TYPE
	Define the process for which to add a source term
Parameter	see PCS
Sub-keyword	$PRIMARY_VARIABLE
	Define the primary variable for which to add a source term
Parameter	see PCS
Sub-keyword	$GEO_TYPE
	Define the geometric entity of the source term
Parameter	see IC
Sub-keyword	$DIS_TYPE
	Define how the source term values will be distributed on the defined geometric entity
Parameter	see IC

Listing A.8 ST keyword

```
#SOURCE_TERM          // source term keyword
$PCS_TYPE             // process type sub-keyword
RICHARDS_FLOW         // specified process
$PRIMARY_VARIABLE     // primary variable sub-keyword
PRESSURE1             // specified primary variable
$GEO_TYPE             // geometry type sub-keyword
POINT POINT0          // specified geometry type | geometry
    name
$DIS_TYPE             // boundary condition type sub-keyword
CONSTANT 1E-6          // source term type | value
#STOP                 // end of input data
```

A.9   MCP—Component Properties

Listing A.9 MCP keyword

```
#COMPONENT_PROPERTIES
$NAME
N(5)
$MOBILE          // 1=MOBILE/transported
1
$DIFFUSION
1  0.3e-9         // constant diffusion ceofficient, set to 0.3e-9
#COMPONENT_PROPERTIES
$NAME
Pyrite ;
$MOBILE
0                // 0=imMOBILEe
$DIFFUSION
0                //diffusion model type, diffusion constant
#COMPONENT_PROPERTIES
[..]
#STOP
```

A.10   MFP—Fluid Properties

Listing A.10 MFP keyword

```
#FLUID_PROPERTIES
$FLUID_TYPE     // sub-keyword to set the fluid type
LIQUID          // liquid fluid selected
$DENSITY        // desnsity sub-keyword
1 0.9997026e+03 // density model 1 (constant) chosen,
                // density set to 0.9997026e+03 kg /m3
$VISCOSITY      // viscosity sub-keyword
1 1.308e-003    // viscosity model 1 (constant) chosen,
                // viscosity set to 1.308e-003 kg /m /s
#STOP
```

Keyword	#COMPONENT_PROPERTIES
	Start a new component bloc, every component must have one bloc. It ends with the beginning of a new MCP block or with the file input terminator #STOP.
Sub-keyword	$NAME
	Set the component name
Parameter	chr
Sub-keyword	$MOBILE
	Define whether the component is mobile or immobile
Parameter	int
	0 = immobile, 1 = mobile
Sub-keyword	$DIFFUSION
	Define the diffusion model type and value
Parameter	int double
	-1 No diffusion specified 0 User-defined function 1 Constant diffusion coefficient in $\mathrm{m^2 s^{-1}}$ 10 Temperature-dependent laws

Keyword	#FLUID_PROPERTIES
	Define the beginning of a MFP file description. It contains the description of one or more fluids.
Sub-keyword	$DENSITY
	Define a fluid density model
Parameters	int double
	The parameter (int) indicates the density model type (see a Table A.2).
	The second parameter (double) gives the density value ($kg \cdot m^{-3}$)
Sub-keyword	$VISCOSITY
	Define a fluid viscosity model
Parameters	int double
	The first parameter (int) defines the viscosity model type and the second value (double) gives the viscosity of the fluid.

Table A.2 Density models

Model	Meaning	Formula	Parameters
0	Curve	RFD file	
1	Constant value	ρ_0	value of ρ_0
2	Pressure dependent	$\rho(p) = \rho_0(1 + \beta_p(p - p_0))$	ρ_0, β_p, p_0
3	Salinity dependent	$\rho(C) = \rho_0(1 + \beta_C(C - C_0))$	ρ_0, β_p, C_0
4	Temperature dependent	$\rho(p) = \rho_0(1 + \beta_T(T - T_0))$	ρ_0, β_T, T_0
...	...	...	...

A.11   MSP—Solid Properties

Keyword	#SOLID_PROPERTIES
	Indicate the start of a new bloc to define properties of the solid phase.
Sub-keyword	$DENSITY
	Define the density model of a solid phase.
Parameters	see MFP

Listing A.11 MSP keyword

```
#SOLID_PROPERTIES // solid properties keyword
$DENSITY          // solid density sub-keyword
  1 2500          // type (1: constant value) | value
#STOP             // end of input data
```

A.12   MMP—Porous Medium Properties

Keyword	#MEDIUM_PROPERTIES
	Indicate the beginning of the material property description.
Sub-keyword	$GEOMETRY_DIMENSION
	Define the geometric dimension of the model domain.
Sub-keyword	$GEOMETRY_AREA
	Define a constant thickness [m] in z-direction for a two-dimensional groundwater flow model.
Parameter	double
	thickness in meter(Note that it does not work for vertical models, i.e., in x- or y-directions.
Sub-keyword	$POROSITY
	Specify the porosity of a medium (esp., for reactive mass transport)
Parameter	int double
	The first parameter defines a porosity model type and the second value is the porosity of the material properties.
Sub-keyword	$STORAGE
	This keyword specifies the storativity of a material, i.e., the volumetric specific storage which is the volume of water that the aquifer released from storage.
Parameter	int double
	The first parameter defines a storage model type (1 = constant) and the second values indicate that the storage.
Parameters	
Sub-keyword	$PERMEABILITY_TENSOR
	Define the intrinsic permeability. Both isotropic and anisotropic are supported.
Parameters	char double...
	The first parameter indicates whether ISOTROPIC, ORTHOTROPIC, or ANISOTROPIC. For the isotropic cases, one permeability value is used for all direction. However the orthotropic case, three values are required for the X-, Y-, and Z-direction. The anisotropic case needs to define a 3 *3 permeability tensor. For groundwater flow processes (GROUNDWATER_FLOW), the hydraulic conductivity ($m \cdot s^{-1}$) is used instead of the permeability.

Listing A.12 MMP keyword

```
#MEDIUM_PROPERTIES        // solid properties keyword
 $GEOMETRY_DIMENSION      // dimension sub-keyword
  1                       // 1: one-dimensional problem
 $GEOMETRY_AREA           // geometry area sub-keyword
  1.0                     // value in square meter if 1D
 $POROSITY                // porosity sub-keyword
  1   0.10                // type (1: constant value) | value
 $STORAGE                 // storativity sub-keyword
  1   0.0                 // type (1: constant value) | value
 $TORTUOSITY              // tortuosity sub-keyword
  1   1.000000e+000       // type (1: constant value) | value
 $PERMEABILITY_TENSOR     // permeability sub-keyword
  ISOTROPIC    1.0e-15    // tensor type (ISOTROPIC) | value(s)
 #STOP                    // end of input data
```

It is worth to mention that the model value is the specific storage divided by fluid density and gravity acceleration constant in Pa^{-1} for the liquid flow processes.

A.13   OUT—Output Parameters

Keyword	#OUTPUT
	Indicate the beginning of an output block which defines the data that will be exported during the simulation run. One output blocks ends with the beginning of the following output blocks, i.e., #OUTPUT, or with the file input terminator #STOP.
Sub-keyword	$PCS_TYPE
Description	Specify the process
Parameters	The parameter is one line below the keyword.
	see PCS
Sub-keyword	$NOD_VALUES
	Define what node related values will be written as output.
Parameter	chr
	The parameter depends on the processes.
	see $ELE_VALUES
Sub-keyword	$GEO_TYPE
	Define the geometric entity of which data will be exported such as the whole DOMAIN or a POINT.
Parameter	chr chr
	The first parameter indicates the geometric object and the second parameter is the name of the object as defined in the corresponding GLI file.
	see IC
Sub-keywords	$DAT_TYPE
	Assign the ASCII data format of the output files as either Tecplot or VTK.
Parameter	chr
	VTK or TECPLOT.
Sub-keyword	$TIM_TYPE
	Define at which time step increments output values will be written.
Parameter	int
	The parameters can be specified in the next line and include the STEPS. STEPS indicates the input of the output increment frequency given by the following integral number n > 0. The output will be written every n th time step. This number defines if every time), or every 101st time steps (STEPS 101) .. is written

Listing A.13 OUT keyword

```
#OUTPUT              // output keyword
 $PCS_TYPE           // process sub-keyword
  RICHARDS_FLOW      // specified process
 $NOD_VALUES         // nodal values sub-keyword
  PRESSURE1          // specified nodal values
 $GEO_TYPE           // geometry type sub-keyword
  POLYLINE bottom    // geometry type and name
 $DAT_TYPE           // specified output format
  VTK                // VTK specified
 $TIM_TYPE           // output times sub-keyword
  STEPS 1            // output at every time step
#STOP                // end of input data
```