System Manuals BASEMENT

User Manual - BASEmesh

VAW - ETH Zurich

Contents

1 Current Version

VERSION 2.6

March, 2016

Project Team

Software Development, Documentation and Test (alphabetical)

D. Ehrbar, MSc. ETH Civil Eng.

M. Facchini, MSc. Environmental Eng.

M. Gerber, BSc. Software Eng.

S. Kammerer, MSc. ETH Environmental Eng.

S. Peter, MSc. ETH Civil Eng.

Dr. D. Vanzo, MSc. Environmental Eng.

L. Vonwiller, MSc. ETH Environmental Eng.

Project Supervisor

Dr. D. Vetsch, Dipl. Ing. ETH

Scientific Advisor

Dr. A. Siviglia, MSc

Director VAW

Prof. Dr. R. Boes

Art Design and Layout: W. Thürig, D. Vetsch

Former Project Members

em. Prof. Dr.-Ing. H.-E. Minor, Director of VAW 1998-2008

Dr. R. Fäh, Dipl. Ing. ETH, Scientific Supervisor, 2002-2013

Dr.-Ing. D. Farshi, MSc., Software Development, 2002-2007

Dr. R. Veprek, Dipl. Rech. Wiss. ETH, Software Development, 2009-2010

R. Müller, Dipl. Ing. EPFL, Software Development, 2003-2012

P. Rousselot, Dipl. Rech. Wiss. ETH, Software Development, 2006-2014

Dr. C. Volz, Dipl.-Ing. Umwelttechnik, Software Development, 2007-2015

Commissioned and co-financed by

Swiss Federal Office for the Environment (FOEN)

Contact

basement@ethz.ch

http://www.basement.ethz.ch

**Laboratory of Hydraulics,
Hydrology and Glaciology**

ETH

**Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich**

Vetsch D., Siviglia A., Ehrbar D., Facchini M., Gerber M., Kammerer S., Peter S., Vanzo D., Vonwiller L., Volz C., Farshi D., Mueller R., Rousselot P., Veprek R., Faeh R.

Citation Advice

For System Manuals:

Vetsch D., Siviglia A., Ehrbar D., Facchini M., Gerber M., Kammerer S., Peter S., Vonwiller L., Volz C., Farshi D., Mueller R., Rousselot P., Veprek R., Faeh R. 2016. System Manuals of BASEMENT, Version 2.5. Laboratory of Hydraulics, Glaciology and Hydrology (VAW). ETH Zurich. Available from http://www.basement.ethz.ch. [date of access].

For Website:

BASEMENT – Basic Simulation Environment for Computation of Environmental Flow and Natural Hazard Simulation, 2016. http://www.basement.ethz.ch

For Software:

BASEMENT – Basic Simulation Environment for Computation of Environmental Flow and Natural Hazard Simulation. Version 2.5. © ETH Zurich, VAW, Vetsch D., Siviglia A., Ehrbar D., Facchini M., Gerber M., Kammerer S., Peter S., Vanzo D., Vonwiller L., Volz C., Farshi D., Mueller R., Rousselot P., Veprek R., Faeh R., 2006-2016.

Preface to Versions 1.0 – 1.3

The development of computer programs for solving demanding hydraulic or hydrological problems has an almost thirty-year tradition at VAW. Many projects have been carried out with the application of "home-made" numerical codes and were successfully finished. The according software development and its applications were primarily promoted by the individual initiative of scientific associates of VAW and financed by federal instances or the private sector. Most often, the programs were tailored for a specific application and adapted to fulfil costumer needs. Consequently, the software grew in functionality but with little documentation. Due to limited temporal and personal resources to absolve an according project, a single point of knowledge concerning the details of the software was inevitable in most of the cases.

In 2002, the applied numerics group of VAW was invited by the Swiss federal office for water and geology (BWG, nowadays Swiss Federal Office for the Environment FOEN) to offer for participation in the trans-disciplinary "Rhone-Thur" project. With the idea to build up a new software tool based on the knowledge gained by former numerical codes - while eliminating their shortcomings and expanding their functionality - a proposal was submitted. The bidding being successful a partnership in terms of co-financing was established. By the end of 2002, a newly formed team took up the work to build the so-called "BASic EnvironMENT for simulation of environmental flow and natural hazard simulation – BASEMENT".

From the beginning, the objectives for the new project were ambitious: developing a software system from scratch, containing all the experience of many years as well as state-of-the-art numerics with general applicability and providing the ability to simulate sediment transport. Additionally, professional documentation is a must. As to meet all these demands, a part wise reengineering of existing codes (Floris, 2dmb) has been carried out, while merging it with modern and new numerical approaches. From a software-technical point of view, an object-oriented approach has been chosen, with the aim to provide reusability, reliability, robustness, extensibility and maintainability of the software to be developed.

After four years of designing, implementing and testing, the software system BASEMENT has reached a state to go public. The documentation at hand confirms the invested diligence to create a transparent software system of high quality. The software, in terms of an executable computer program, and its documentation are available free of charge. It can be used by anyone who wants to run numerical simulations of rivers and sediment transport – either for training or for commercial purposes.

The further development of the software tends to new approaches for sediment transport simulation, carried out within the scope of scientific studies on one hand side. On the other hand, effectiveness and composite modelling are the goals. On either side, a reliable software system BASEMENT will have to meet expectations of the practical engineer and the scientist at the same time.

Minor

em. Prof. Dr.-Ing. H.-E. Minor
Member of the steering committee of Rhone-Thur Project 2002-2007
Director of VAW, 1998-2008

October, 2006

Preface to Versions 1.4

The work since the first release of the software in October 2006 was exciting and challenging. To go public is paired with interests and demands of users – although user support for the software never was intended. But interchange with users is definitely one of the most crucial

factors of successful software development. Feedback from academic or professional users conveys a different point of view and enables the development team to achieve costumer proximity as well as to consolidate experience. Accordingly, the project team tried to meet the demands as effectively as possible. In version 1.3 of BASEMENT, which was released in April 2007, there were some errors fixed, a few new features added and the documentation was completed. Since then, many things have changed: on the personnel, on the project as well as on the software technical level.

In summer 2007 one of our main software developers, Dr. Davood Farshi, left VAW and changed to an international hydraulic consultant. Dr. Farshi supported our team from 2002 to 2007 as a profound numeric specialist and was mainly involved in the development of BASEplane. At his own request, he is still engaged in the development of BASEMENT as external advisor and tester. Dr. Farshi's position in the project team was reoccupied by Christian Volz, an environmental engineer from southern Germany. Mr. Volz has broad experience in numerical modelling as well as object-oriented programming.

On the project level the framework slightly changed. The initial scope within BASEMENT was developed, the "Rhone-Thur" project, has been finalized by the end of 2007. The sequel is called "Integrales Flussgebietsmanagement". It has the same co-financer as its predecessor, namely the Swiss federal office for the environment (FOEN), and basically the same participating institutions (EAWAG, WSL, LCH(EPFL) and VAW(ETHZ)). The funding runs until the end of 2011. Due to the retirement of Prof. Dr.-Ing. H.-E. Minor in summer 2008, our laboratory is solely represented in the project committee by Dr. R. Fäh at the moment.

The emphases of the new proposal for the further development of BASEMENT are advanced topics of hydraulics and sediment transport, such as secondary currents and lateral erosion. Furthermore, the efficiency of the software should be increased by the implementation of appropriate parallelisation and coupling approaches.

Since the last minor release a long time passed, which was mainly consumed by a general revision of the software. After five years of development a diligent consolidation was expedient. In addition, the coincidence of a new team member offered an unbiased reflection of the source code. All in all it was very worthwhile.

Last but not least, there are numerous bugs fixed and some new features in the current version. Mainly the efficiency of the software has been improved. The first stage of parallelisation is completed. The current implementation of the code includes the OpenMP interface which allows for parallel execution of the basic computation loops. In other words, the software is now able to exploit the power of current multi-core processors with a convincing speedup. Furthermore, the revision of some data structures and output routines as well as the application of an optimised compiler led to a reduction in execution time.

Concerning sediment transport, the one-dimensional model BASEchain now supports the modelling of fine material, either as suspended or bed load. Also the advanced models for boundary conditions are worth mentioning. On the one hand, it is now possible to model domain boundaries with momentum and on the other hand, special boundary conditions

inside the computational region, such as a weir or a gate, are implemented. The fact, that the version 1.4 of BASEMENT is also available for the Linux operating system the first time, rounds off the new additions and features of the software package at hand.

Summarised one may say that the release 1.4 of BASEMENT is a major release due to all the different kinds of changes, but it's still a minor release concerning the new features – let's call it a "major minor" release. We are looking forward to Version 2.0 of BASEMENT, which is planned for next year.

D. Vetsch

Project Supervisor

October, 2008

Preface to Version 2.0

Four years ago, in spring 2006, the first version of the software system BASEMENT was completed and ready for internal use. In autumn of the same year, the first official version 1.1 of the software was released and made available as free download on the project website www.basement.ethz.ch. Since then, the functionality of the program has been enhanced and the international user community has grown gradually. Over the last years, BASEMENT has become a reliable tool for professional investigations, especially within the scope of flood prevention, and for scientific studies. Furthermore, the software is part and parcel of the lecture "Numerical Models in Hydraulic Engineering" to ensure education of young engineers in the field of hydrodynamic numerical simulation. The lecture is held on a regular basis by VAW staff for master students of civil and environmental engineering at ETH Zurich.
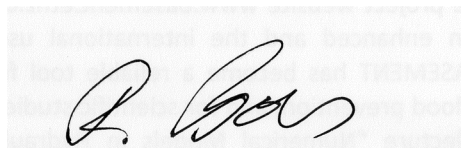
In February 2009, I have become the successor of Prof. em. Dr.-Ing. H.-E. Minor as Director of the Laboratory of Hydraulics, Hydrology and Glaciology (VAW) at ETH Zurich. In the meantime, I have joined the project committee "Integrales Flussgebietsmanagement"as a further representative of VAW besides Dr. R. Faeh.

Furthermore, there are some changes concerning the personnel of the project team of BASEMENT to mention. Lukas Vonwiller joined the team last autumn after having obtained his master's degree at ETH Zurich. Within the scope of his master thesis at the VAW, he studied the hydrodynamics and ecological impact of floods at the river Flaz using BASEMENT. Some of his experiences with the application of BASEMENT and selected results are documented in the new tutorial on 2-D simulations in the user manual UIV. His current duties are the application and testing of the software in terms of project work. We were also very lucky being able to engage Dr. Ratko Veprek as a distinguished software engineer for a limited period of time. His contributions to the software, such as flow control of river systems, computational efficiency and the graphical user interface, just to name a few, are of great value. Unfortunately he will leave us by the day of the release to take on a post doctoral position abroad.

According to the announcement in the preface to version 1.4, the second major version of BASEMENT is released with little delay but with all the more important improvements and substantial new features. First of all, the new version 2.0 of the program comes with a graphical user interface (GUI), which allows running or stopping simulations and tracking the progress. Furthermore, the model setup and configuration, i.e. the assembling of the command file, is completely integrated into the GUI. The user is guided through the setup and any input is validated directly. In addition, the integrated help function, which is based on the command file reference, provides detailed information on the meaning of input parameters. This gives way to a clearer model setup compared to the rather fault-prone manual text editing, which is still available and also accessible through the GUI. Another main feature of the new GUI is the editing of the topography for BASEchain. Besides the GUI based setup, interpolation and thinning out of model cross sections, a graphical viewer helps the user to check the configuration and subdivision. For this reason, the new version of BASEMENT comes with its own topography file format for BASEchain. The new format has a clear structure similar to the style of the command file.

Moreover, the visualisation of actual results during a simulation with BASEviz has been improved and is now more interactive, i.e. the simulation can be paused, continued or the variable shown can be switched. Other improvements concern computational efficiency and sediment transport, especially gravitational bed load transport. Please refer to the release notes in the section "introduction and installation" of this manual for further details about new features and bug fixes.

The software system BASEMENT in its current version 2.0 has reached the point to be termed as a state of the art numerical modelling tool for flow and sediment transport in rivers. The incorporated well established or new numerical approaches, software technical features like parallelization or the coupling of sub domains, advanced features for sediment transport and flow control are making it a reliable tool for professional as well as scientific applications. With the new GUI another hurdle has been cleared and a new era of the software in terms of usability has begun. We are looking forward to the further development as well as upcoming releases of BASEMENT and we are curious about how the software will establish itself in the future.

Prof. Dr. R. Boes
Committee Member of Project "Integrales Flussgebietsmanagement"
Director of VAW

May, 2010

BASEMENT SOFTWARE LICENSE

between

ETH

Rämistrasse 101

8092 Zürich

Represented by Prof. Dr. Robert Boes

VAW

(Licensor)

and

Licensee

1. Definition of the Software

The Software system BASEMENT is composed of the executable (binary) file BASEMENT and its documentation files (System Manuals), together herein after referred to as "Software". Not included is the source code.

Its purpose is the simulation of water flow, sediment and pollutant transport and according interaction in consideration of movable boundaries and morphological changes.

2. License of ETH

ETH hereby grants a single, non-exclusive, world-wide, royalty-free license to use Software to the licensee subject to all the terms and conditions of this Agreement.

3. The scope of the license

a. Use

The licensee may use the Software:

- according to the intended purpose of the Software as defined in provision 1

- by the licensee and his employees

- for commercial and non-commercial purposes

The generation of essential temporary backups is allowed.

b. Reproduction / Modification

Neither reproduction (other than plain backup copies) nor modification is permitted with the following exceptions:

Decoding according to article 21 URG [Bundesgesetz über das Urheberrecht, SR 231.1)

If the licensee intends to access the program with other interoperative programs according to article 21 URG, he is to contact licensor explaining his requirement.

If the licensor neither provides according support for the interoperative programs nor makes the necessary source code available within 30 days, licensee is entitled, after reminding the licensor once, to obtain the information for the above mentioned intentions by source code generation through decompilation.

c. Adaptation

On his own risk, the licensee has the right to parameterize the Software or to access the Software with interoperable programs within the aforementioned scope of the licence.

d. Distribution of Software to sub licensees

Licensee may transfer this Software in its original form to sub licensees. Sub licensees have to agree to all terms and conditions of this Agreement. It is prohibited to impose any further restrictions on the sub licensees' exercise of the rights granted herein.

No fees may be charged for use, reproduction, modification or distribution of this Software, neither in unmodified nor incorporated forms, with the exception of a fee for the physical act of transferring a copy or for an additional warranty protection.

4. Obligations of licensee

a. Copyright Notice

Software as well as interactively generated output must conspicuously and appropriately quote the following copyright notices:

Copyright by ETH Zurich, VAW, 2006-2016

Vetsch D., Siviglia A., Ehrbar D., Facchini M., Gerber M., Kammerer S., Peter S., Vanzo D., Vonwiller L., Volz C., Farshi D., Mueller R., Rousselot P., Veprek R., Faeh R.

5. Intellectual property and other rights

The licensee obtains all rights granted in this Agreement and retains all rights to results from the use of the Software.

Ownership, intellectual property rights and all other rights in and to the Software shall remain with ETH (licensor).

6. Installation, maintenance, support, upgrades or new releases

a. Installation

The licensee may download the Software from the web page http://www.basement.ethz.ch or access it from the distributed CD.

b. Maintenance, support, upgrades or new releases

ETH doesn't have any obligation of maintenance, support, upgrades or new releases, and disclaims all costs associated with service, repair or correction.

7. Warranty

ETH does not make any warranty concerning the:

- warranty of merchantability, satisfactory quality and fitness for a particular purpose

- warranty of accuracy of results, of the quality and performance of the Software;

- warranty of noninfringement of intellectual property rights of third parties.

8. Liability

ETH disclaims all liabilities. ETH shall not have any liability for any direct or indirect damage except for the provisions of the applicable law (article 100 OR [Schweizerisches Obligationenrecht]).

9. Termination

This Agreement may be terminated by ETH at any time, in case of a fundamental breach of the provisions of this Agreement by the licensee.

10. No transfer of rights and duties

Rights and duties derived from this Agreement shall not be transferred to third parties without the written acceptance of the licensor. In particular, the Software cannot be sold, licensed or rented out to third parties by the licensee.

11. No implied grant of rights

The parties shall not infer from this Agreement any other rights, including licenses, than those that are explicitly stated herein.

12. Severability

If any provisions of this Agreement will become invalid or unenforceable, such invalidity or enforceability shall not affect the other provisions of Agreement. These shall remain in full force and effect, provided that the basic intent of the parties is preserved. The parties will in good faith negotiate substitute provisions to replace invalid or unenforceable provisions which reflect the original intentions of the parties as closely as possible and maintain the economic balance between the parties.

13. Applicable law

This Agreement as well as any and all matters arising out of it shall exclusively be governed by and interpreted in accordance with the laws of , excluding its principles of conflict of laws.

14. Jurisdiction

If any dispute, controversy or difference arises between the Parties in connection with this Agreement, the parties shall first attempt to settle it amicably.
Should settlement not be achieved, the Courts of Zurich-City shall have exclusive jurisdiction. This provision shall only apply to licenses between ETH and foreign licensees

By using this software you indicate your acceptance.

(License version: 2016-03-07)

THIRD PARTY SOFTWARE COPYRIGHT NOTICES

The Visualization Toolkit (VTK)

Copyright (c) 1993-2008 Ken Martin, Will Schroeder, Bill Lorensen. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither name of Ken Martin, Will Schroeder, or Bill Lorensen nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

CVM Class Library

Copyright (c) 1992-2010 Sergei Nikolaev

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the

Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Qt Toolkit - Cross-platform application and UI framework

The Qt Toolkit is Copyright (C) 2015 Digia Plc and/or its subsidiary(-ies) and other contributors.

Contact: http://www.qt-project.org/legal

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Qwt - Qt Widgets for Technical Applications

BASEMENT is based in part on the work of the Qwt project (http://qwt.sf.net).

CGNS – CFD General Notation System

cgnslib 4.5 - http://cgns.sourceforge.net

2 BASEmesh Manual

2.1 Use of QGIS plugin BASEmesh for grid generation

In order to provide a free and open source solution for the creation of computational meshes (Pre-Processing) and to visualize simulation results (Post-Processing) the plugin BASEmesh for the free and open source geographic information system (GIS) software QGIS was developed.

We hope to enforce this powerful free GIS-solution in future to allow for a workflow which is completely based on non-commercial software. A further advantage of this workflow is that the involved programs are portable and are available for Windows-OS as well as Linux-OS.

2.1.1 Introduction

The plugin BASEmesh is a tool for creating computational meshes, based on the advanced mesh generator TRIANGLE by Jonathan R. Shewchuk. Furthermore, features for loading and editing existing meshes are provided. At the moment, BASEmesh only supports the automated creation of meshes consisting of triangular elements. Meshes containing quadrilateral elements, however, can be imported, manually edited and exported.

Beside 2D meshes, the plugin is also able to convert 1D meshes of different formats into each other (at the moment conversions are possible between BASEchain and HEC-RAS). The meshes created by BASEmesh can directly be used for computations with the basic simulation environment software BASEMENT. Post-processing, i.e. the visualization of simulation results, is not directly supported by BASEmesh. Still, there are various options for displaying and processing BASEMENT results in QGIS, which is briefly explained in Section 2.1.9 and detailed in the Post-Processing tutorial.

The manual at hand introduces the features of BASEmesh and the basics of its usage. No information is given about the usage of GIS software in general. Some basic knowledge in applying GIS software might be acquired using QGIS' excellent documentation. To get in touch with BASEmesh and QGIS fast and easy, a step by step tutorial. is provided as well and can be downloaded as separate document. This manual provides the reader with an overview of the capabilities of BASEmesh, the installation procedure, and a short description of the

BASEmesh components that enable creating, importing, editing, and exporting computational meshes.

2.1.1.1 Installation

BASEmesh is at present available on a specific Plugin repository which has to be connected manually in the QGIS plugin manager. In contrast to other plugins, it is not available via the official QGIS plugin repository which is set as default in every QGIS installation.

To install BASEmesh, follow these steps:

(1) Start QGIS

(2) Load the QGIS plugin manager by choosing *Manage and Install Plugins. . .* in the menu *Plugins* in the QGIS main toolbar

(3) Go to *Settings* (you should now see the connection to the official QGIS-plugin repository)

(4) Click on *Add. . .* and give a name, e.g. 'BASEmesh repository'

(5) Enter the repository address: http://people.ee.ethz.ch/~basement/qgis_plugins/qgis_ plugins.xml (If you copy this adress, the 'tilde (~)' must be set correctly, otherwise the plugin will not be found)

(6) Press *OK*

(7) The additional repository should now be visible (make sure that the *Status* is *connected*)

(8) Go to *All* in the menu of the plugin manager and search for 'BASEmesh'

(9) Choose the BASEmesh plugin (if several are available, choose the one with the highest version number) and press *Install plugin*

If everything worked, you should be able to access BASEmesh via the toolbar and via the menu *Plugins* (see Figure 2.1).
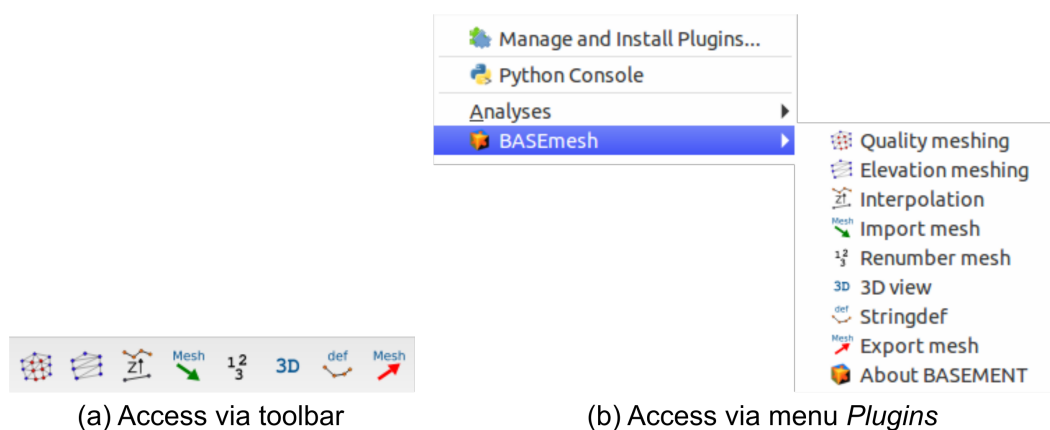

(a) Access via toolbar   (b) Access via menu *Plugins*

Figure 2.1 *BASEmesh representation in QGIS after successful installation*

2.1.1.2   Components of BASEmesh

The philosophy of creating a computational mesh within BASEmesh is as follows (see Figure 2.2):

Generate a quality mesh Here you produce the actual computational grid ('quality mesh') with all its attributes you define, i.e. domain area, element sizes, mesh holes, breaklines or enforced vertices. This mesh does not contain any topographic information. The quality of this mesh influences the results of your numerical analysis, e.g. stability, computation time, accuracy etc.

Provide an elevation model The topography of your computational domain is usually described by points in a text file containing the (x,y,z)-coordinates or by a raster dataset. In case of xyz-data the elevation model has to be formulated as triangulated irregular network (TIN), called *elevation mesh* within BASEmesh.

Interpolation The topographical information contained in the elevation model will be interpolated on the quality mesh, i.e. an elevation value is assigned to each node of the computational grid.

In addition to mesh creation the possibility of manipulating an existing mesh is provided as well in BASEmesh (see Figure 2.2). Starting from a mesh you have created previously or you have imported to QGIS, you can edit the following:

Material indices This attribute is used in BASEMENT to distinguish between different materials which can be used to define friction, soil composition etc.

Combine meshes Multiple meshes can be combined to one big mesh, by carefully applying manual meshing.

Renumber your mesh After applying manual changes to your mesh, you need to make sure that the internal representation of the mesh is correct. To this end BASEmesh provides a tool called *Renumber Mesh*, where all the manual changes are incorporated into a valid mesh structure (see Section 2.1.1.3).

Find *stringdef* nodes After changes of the computational mesh the node IDs may change as well, and all *stringdef* definitions of BASEMENT have to be updated. To this end BASEmseh provides a tool called *Stringdef*.
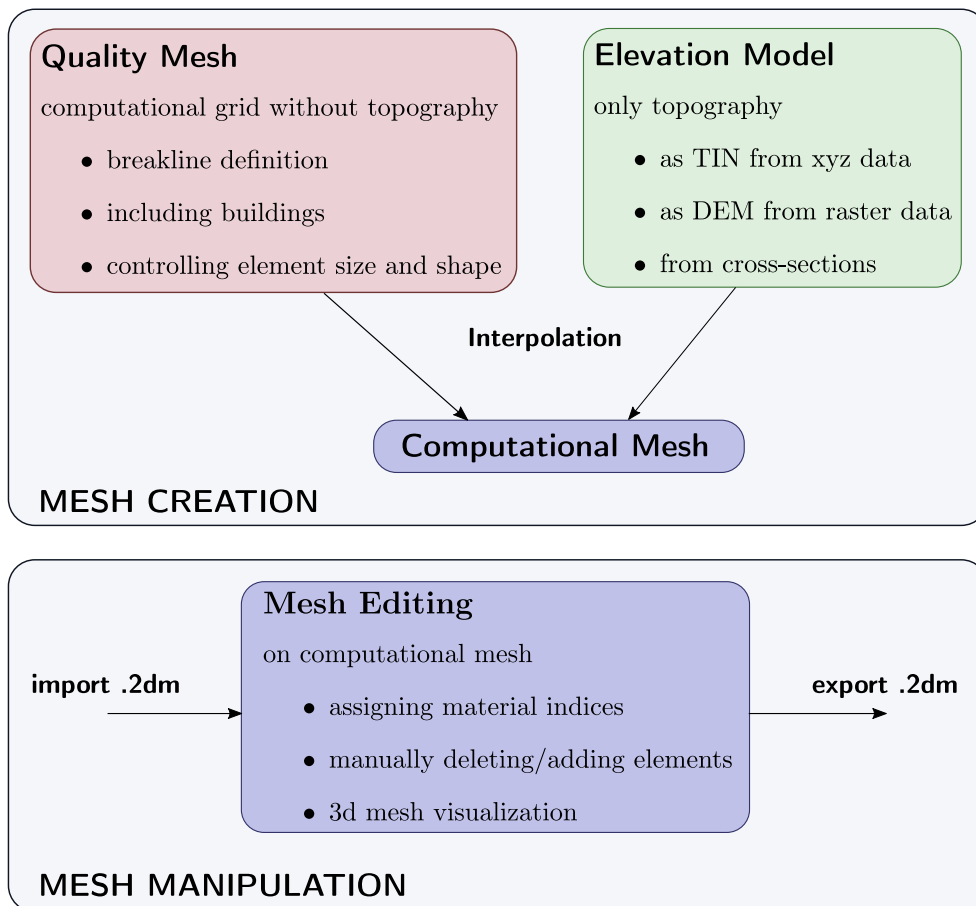
Figure 2.2 *Pre-processing components of BASEmesh*

2.1.1.3 Mesh respresentation in BASEmesh

Meshes generated by BASEmesh follow a specific structure. When manually editing a mesh it is of high importance to maintain this structure, otherwise it is not possible to export the mesh to a 2dm - grid which can be used for computational purposes. The mesh structure of BASEmesh is defined as follows (see Figure 2.3):

- In BASEmesh, meshes are represented by two vector layers: (1) '*meshName*_nodes' containing only the nodes of the mesh and (2) '*meshName*_elements' containing only the elements of the mesh.

- '*meshName*_nodes' contains point features (2-dimensional geome). The attributes for each feature are *NODE_ID*, *X*, *Y*, and *Z*. Please note thate the elevation information is only contained in the attribute table of the vector file.

- '*meshName*_elements' contains polygon features (2-dimensional). The attributes for each feature are *ELEMENT_ID*, *NODE1*, *NODE2*, *NODE3*, *NODE4*, and *MATID*. The attributes *NODEx* stand for the *NODE_ID*'s within the layer '*meshName*_nodes' and define the mesh connectivity.

- The *MATID* attribute is used to set material indices for each element within the mesh. BASEMENT uses these indices to set up zones with different characteristics in the mesh, e.g. to set different friction values in the river bed and in the flood plain of the domain.

- In case of triangular elements the attribute *NODE4* is -1. In case of quadrilateral elements, however, the *NODE4* attribute has to be specified.

Within QGIS, it is possible to create elements without defining the attributes *NODEx*. In this case, the tool *Renumber Mesh* must be used to update the connectivity of the mesh before exporting to a 2dm - grid.

Node 4	
NODE_ID	4
X	635000
Y	278000
Z	486.5

Element 3	
ELEMENT_ID	3
NODE1	8
NODE2	5
NODE3	9
NODE4	-1
MATID	1
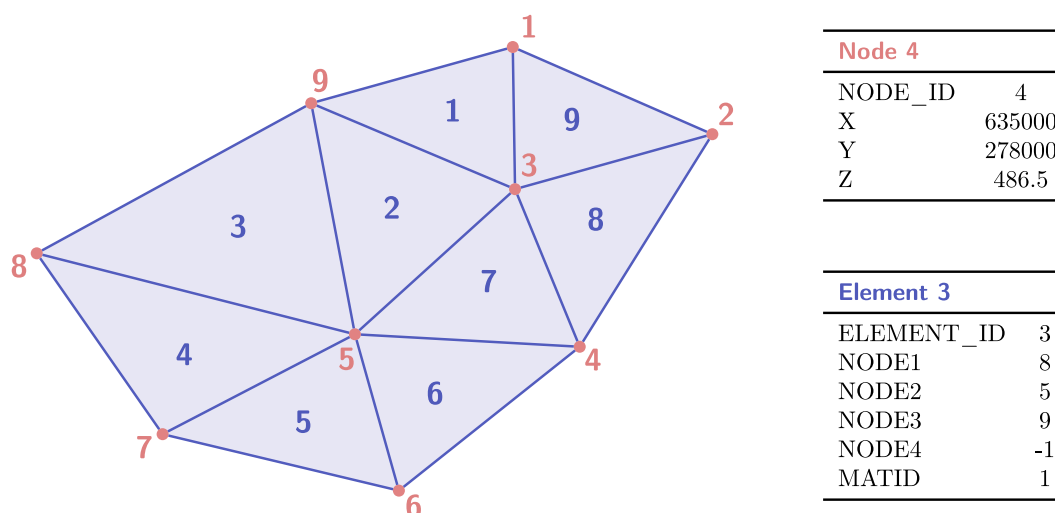
Figure 2.3 *Example of internal mesh representation with nodes as point layer (red) and elements as polygon layer (blue).*

2.1.2   Elevation Meshing

The elevation mesh consists of triangulated elevation points and corresponding breaklines. It represents the topography of the computational domain and organizes the elevation data into a geometric structure which can be used for future elevation interpolations. The elevation mesh needs to be distinguished from the quality mesh, since it does not consider any quality aspects of the mesh elements and therefore should not be used for computations. The following data sources are combined:

1. *Elevation points* with x,y,z-coordinates. These elevation points may be taken from DTMs, cross section profiles, field measurements and other data sources. As input, all elevation points need to be combined into one single point shapefile.

2. *Breaklines* defining distinct interruptions of the the surface slopes (e.g. dyke crests or river side walls). Breaklines are given as line shapefile and connect elevation points with each other. During mesh generation, no triangles crossing the breaklines will be

created. Be aware that all end points or vertices of a breakline must have corresponding elevation points!

3. *Model boundary* defining the extents of the elevation mesh. It must be given as polygon shapefile, which connects elevation points with each other. Be aware that all vertices of the model boundary polygon must have corresponding elevation points!

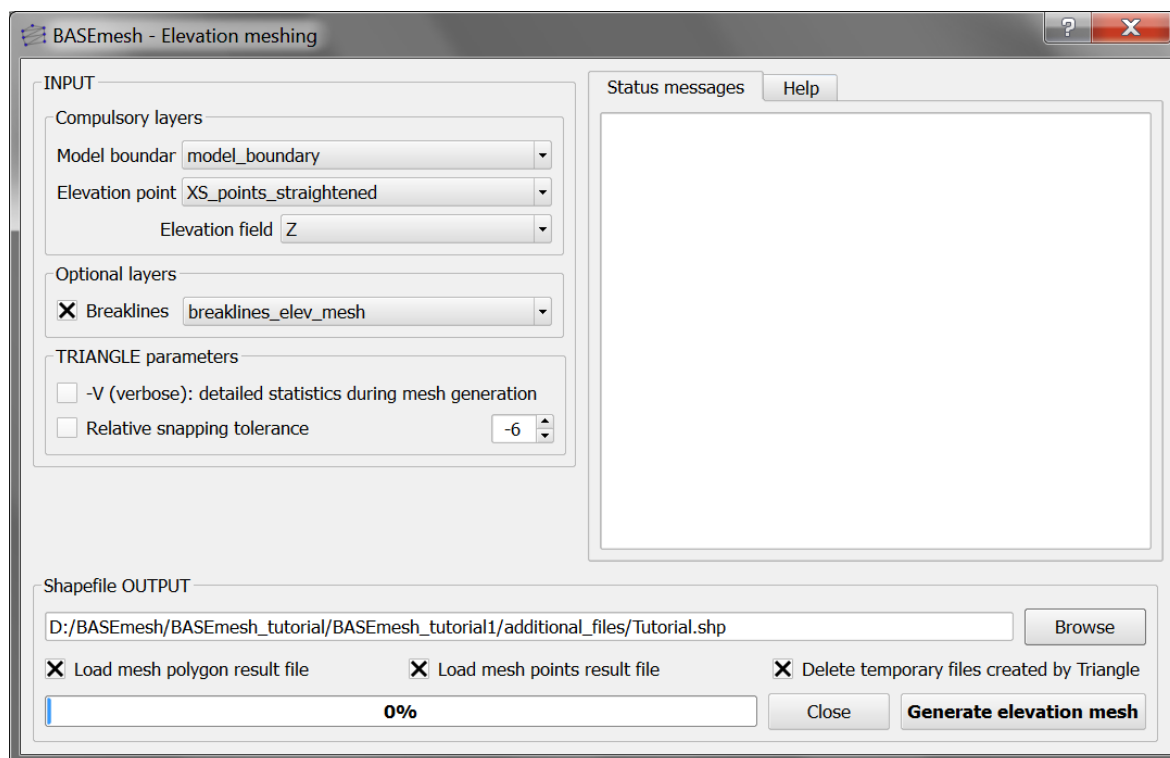The elevation mesh is generated automatically with the program Triangle, without any parameters to be set.

Figure 2.4 *User interface for the creation of elevation meshes.*

2.1.3 Quality Meshing

The quality mesh defines the computational elements and nodes as they are used for the numerical computations. In contrast to DTM-models or TIN of elevation data, it has high demands concering the quality of the elements (element size, element angles, shape, etc.), which directly influence the quality, stability and performance of the computations. It is important to have smooth transitions between elements with different sizes and to prevent distorted elements with small angles or very small element sizes. In general, a high-quality mesh is one which is nice to look at! The quality mesh contains all geometric information (x-y plane), but has no topographical information (z plane). Quality mesh generation involves mainly two steps:

1. Specification of all geometric information about the computational domain:

- *Model boundary*: extent of the computational domain.

- *Break lines*: distinct interruptions of the the surface slopes (dyke crest, river side walls, ...) which shall be preserved in the computational mesh.

- *Holes*: parts within the mesh which are excluded from modelling (e.g. buildings). These parts are defined by special points (layer region_points) surrounded by breaklines.

- *Vertices*: enforced geometric points in the mesh (e.g. measurement points).

2. Generation of triangles respecting specific quality criteria using the free program TRIANGLE. Parameters of major importance are:

- *Maximum area* constraints: definition of the mesh density using *maximum area* contraints for the triangular mesh elements. The *maximum area* is defined as attribut in the layer region_points and holds for a specific *Region* surrounded by breaklines.

- *Dividing constraints*: With this attribute in the layer *Breaklines* one can enforce a certain number of mesh elements along a breakline. This is of major importance for the use of inner boundaries in BASEMENT, where an equal number of mesh elements at the uptream and downstream interface is required.

- *Minimum triangle angle*: no elements with angles smaller than the minimum angle specified are generated (smaller angles lead to less elements, while larger angles lead to more elements).

- *Relative snapping tolerance*: defines, how far two point coordinates may be located apart to still be considered at the same location. The default value is 10E-6. Increasing this tolerance can help to avoid problems due to improper snapping of vertices (line or polygon features) and points in OGIS.

Since the quality mesh has no topographical information, it is usually necessary to interpolate data from an elevation model on the mesh nodes before using it for computations.
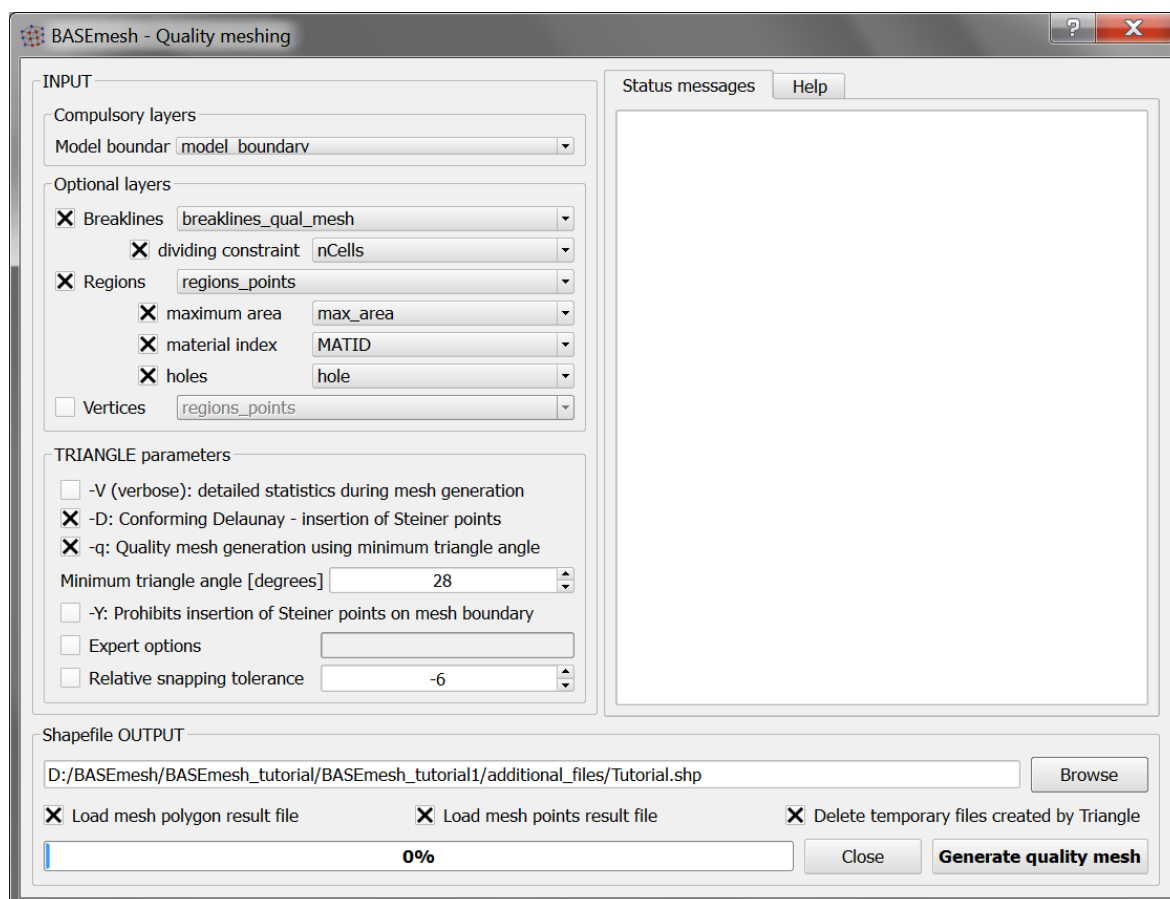
Figure 2.5 *User interface for the creation of quality meshes.*

2.1.4   Interpolation of Elevation Data

An important task in mesh generation is the interpolation of elevation data onto the nodes of the quality mesh. As a result, the final computational mesh is obtained, which then can be exported and used for simulations. The elevation data serving as input can be provided in two different elevation model types:

1. *Elevation mesh* triangulated from pointwise elevation data (TIN). The routine identifies the coordinates of each quality mesh node and determines any underlying elevation mesh element. Two methods are used for data interpolation:

    a) If an underlying elevation mesh element is found, the elevation of the quality mesh node is interpolated at its x-y-coordinates. This is the normal case, since the elevation mesh usually covers the whole computational domain. Nodes interpolated with this method are marked by a 1 in the element - field of the node attribute table. If the quality mesh node is located at the exact coordinates of an elevation mesh node, its height value is preserved exactly.

    b) If no underlying elevation mesh element is found, the quality node elevation is set to that of the nearest node of the elevation mesh. This is the case if quality

mesh nodes lie outside the domain covered by the elevation mesh or when holes are present in the elevation mesh. It may lead to incorrect quality mesh ndoe elevations. Hence, it is recommended to choose a bigger domain for elevation meshing than for quality meshing. Nodes interpolated this way are marked by a 0 in the element - field of the result attribute table and are named 'with special treatment' in the QGIS status messages.

2. *Digital elevation map* as raster data which contains the topography as DTM. The raster elevation data is directly mapped on the computational mesh nodes without interpolation. If no corresponding raster cell is found, the elevation is set to '-999'.
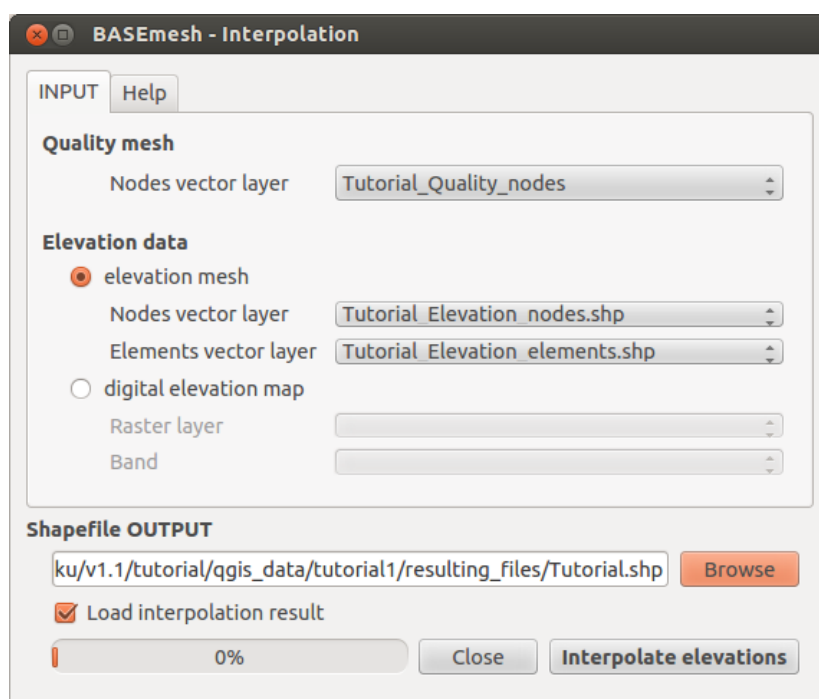

Figure 2.6 *User interface for the interpolation of topographical data onto quality meshes.*

2.1.5   Visualizing meshes in 3D

During the process of mesh generation you can have a look at your mesh using the 3D viewer. The intention is to only have a quick glance to the third dimension of your mesh to detect potential errors or shortcomings in your mesh visually.

The 3D view tool requires the specification of a polygon shapefile with mesh elements and a point shapefile with the corresponding mesh nodes. The mesh nodes need to have elevation attributes and the mesh elements need to have material indices specified in their attribute table. Changing the *Elevation scaling factor* will stretch (>1.0) or squeeze (<1.0) the third dimension of your mesh view.

The window popping up shows your mesh (if only parts of your mesh is visible, zoom out). You can change the view as follows:

- zooming in and out: scroll using the wheel of your mouse

- shifting: drag and drop using the wheel button of your mouse

- rotating: drag and drop using left-click of your mouse (the orbit centre is located in the centre of your mesh)
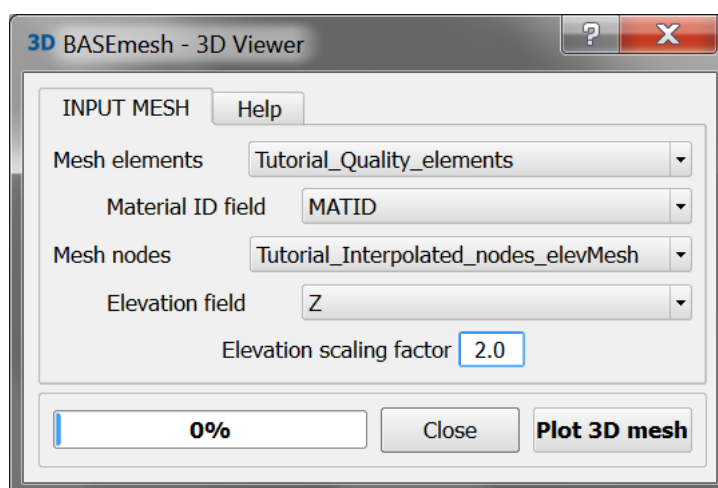

Figure 2.7 User interface for 3D view.

2.1.6   Renumbering Meshes

An unstructered mesh generally consists of a list of nodes, a list of elements and the mesh connectivity logic behind. The elements are formed by three or four nodes and dispose of a certain orientation (here: clockwise orientation). If a mesh is generated via BASEmesh using *quality meshing* or *elevation meshing*, then the mesh connectivity is always in a valid state. However, if the mesh is modified afterwards, e.g. by deleting or adding elements/nodes or swapping edges, then the mesh connectivity becomes invalid. Therefore, after modifications, the mesh connectivity has to be updated. This task is performed using the *Renumber Mesh* tool:

1. The mesh connectivity is checked and updated. For modified or added elements, the corresponding node ID-numbers are searched by comparisons of x-y coordinates. Furthermore, the clockwise element orientation is established by re-ordering the element nodes if necessary.

2. Nodes and elements are renumbered in a way that their ID-numbers begin ascending from '1' without gaps.

Renumbering requires a mesh as input data, i.e. a polygon shapefile containing the mesh elements and a point shapefile containing the mesh nodes. In addition, all elements must have material indices specified in their attribute table and all nodes must have elevation attributes. After renumbering, the mesh is in a valid state and can be exported for computations.
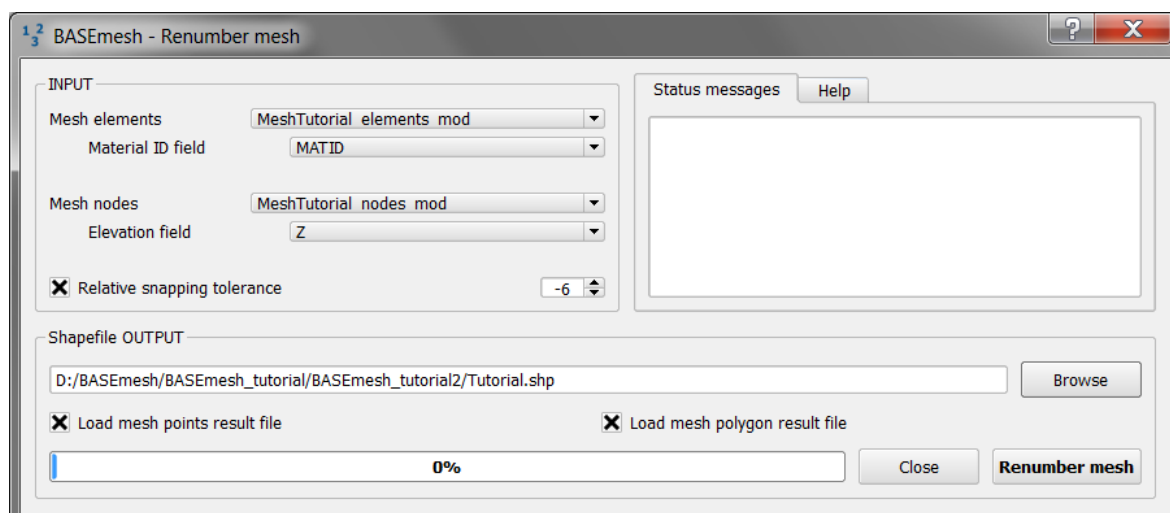
Figure 2.8 *User interface for renumbering meshes.*

2.1.7   Finding Node IDs for *stringdef*

In BASEMENT a list of node IDs is defined as *stringdef*. They can be used to define a boundary condition or an output along these nodes. The IDs correspond to the node IDs of the computational mesh. As soon as there are changes in this mesh, or only in its connectivity, the IDs will change as well and the list of node IDs inside of BASEMENT have to be updated. This task is automated with the tool *Stringdef* of BASEmesh:

1. For each line feature with a non-empty stringdef-field, all nodes that are located exactly on that line, are listed in a text file (BASEMENT-like stringdef block). The content of the stringdef-field represents the stringdef name.

2. The stringdef line features are favorably included in the layer where the breaklines of the quality mesh are defined. To distinguish between regular breaklines and stringdef lines, the stringdef-field can be used (empty vs. non-empty).

3. The upstream direction of the generated stringdefs is *right*. Therefore the line feature has to be drawn from the left riverbank to the right riverbank.

The Stringdef tool requires a point layer (i.e. the computational nodes) and a line layer (i.e. the breaklines with at least one stringdef line feature) as input. The resulting list(s) of node IDs is written as text file output that is user defined.
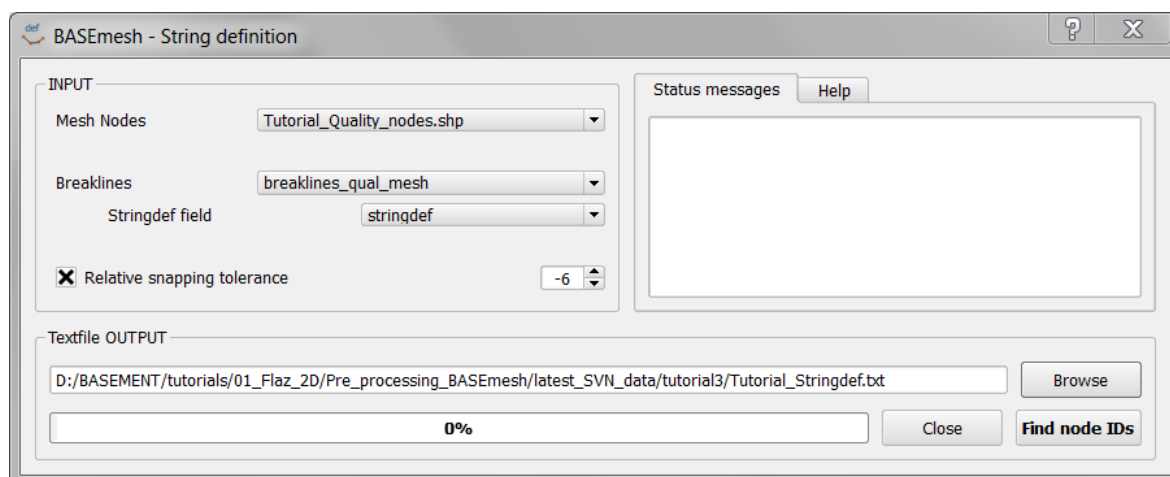
Figure 2.9 *User interface for Stringdef.*

2.1.8   Import and Export of mesh - files

2D mesh files:

The *.2dm*-format is an ASCII data format for unstructured meshes specified for the Mesh Module of software SMS (www.aquaveo.com) . Due to its simplicity, this data format is used by a variety of surface water modelling programs, as e.g. BASEMENT. A *.2dm*-file contains the node list with all coordinates and the elements with the mesh connectivity.

The main task of BASEmesh is the generation of a computational mesh in *.2dm*-format, which can be directly used for numerical simulations with BASEMENT. Therefore it is necessary to export the generated mesh to the .2dm-format. Furthermore, it is also often useful to visualize an existing *.2dm*-mesh, to change its material indices, to extract node or element numbers or to modify other mesh properties. For these purposes, BASEmesh provides options for importing and exporting *.2dm* - meshes.

1D mesh files:

Beside 2D meshes, BASEmesh is also able to import and export 1D grid files. These file formats can be converted into each other. At the moment, two file formats are supported:

- BASEchain geometry files (*.bmg*), and

- HEC-RAS geometry files (*.g01*).

These 1D functions are useful for visualization purposes of 1D grids, as for example the lateral coupling of 1D and 2D models. Furthermore, the conversion utility allows for using the HEC-RAS software for pre-processing of 1D geometry files (e.g. extracting cross sections from digital elevation models (DEM) or interpolations of cross sections).

Import mesh

After specification of the locations of the input and output files, the 2D / 1D mesh-file can be imported into QGIS. The mesh data are stored in shape-format within QGIS.

- 2D: Two shapefiles are generated with (a) element data (suffix: '_elements.shp') and (b) node data ('_nodes.shp'), containing the mesh connectivity information and material indices.

- 1D: Two shapefiles are generate with (a) cross section data (suffix: '_crossSections.shp') and (b) point data ('_points.shp'), containing the cross section data and the cross section points.
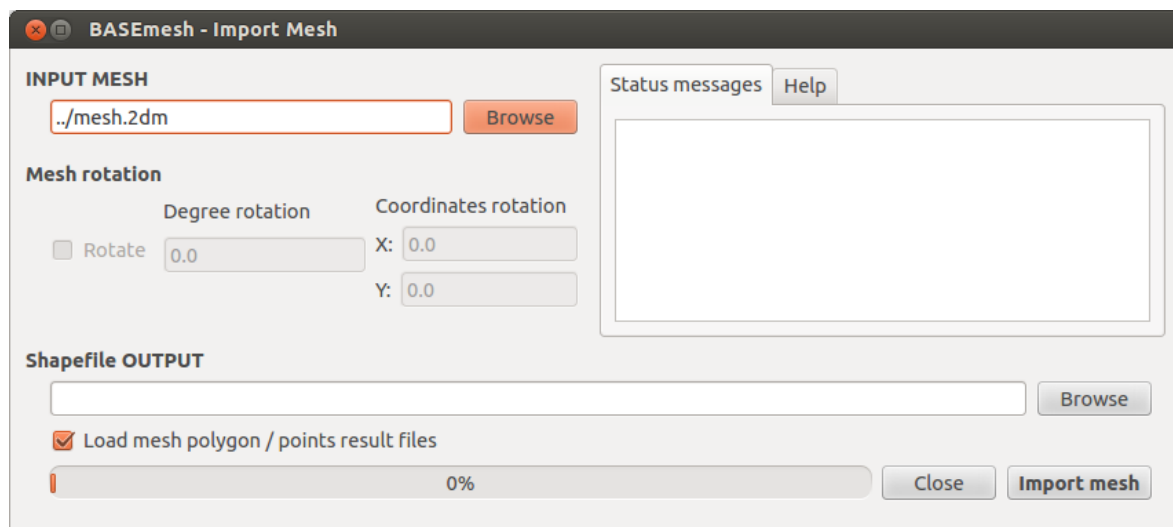

Figure 2.10 *Importing an existing mesh from a .2dm-file.*

Export mesh

- 2D: Export into the 2D mesh-format requires the specification of a polygon shapefile with mesh elements and a point shapefile with the corresponding mesh nodes. The mesh elements need to have material indices specified in their attribute table and the mesh nodes need to have elevation attributes.

- 1D: Export into the 1D mesh-formats requires the specification of the file format and the cross section shapefile.
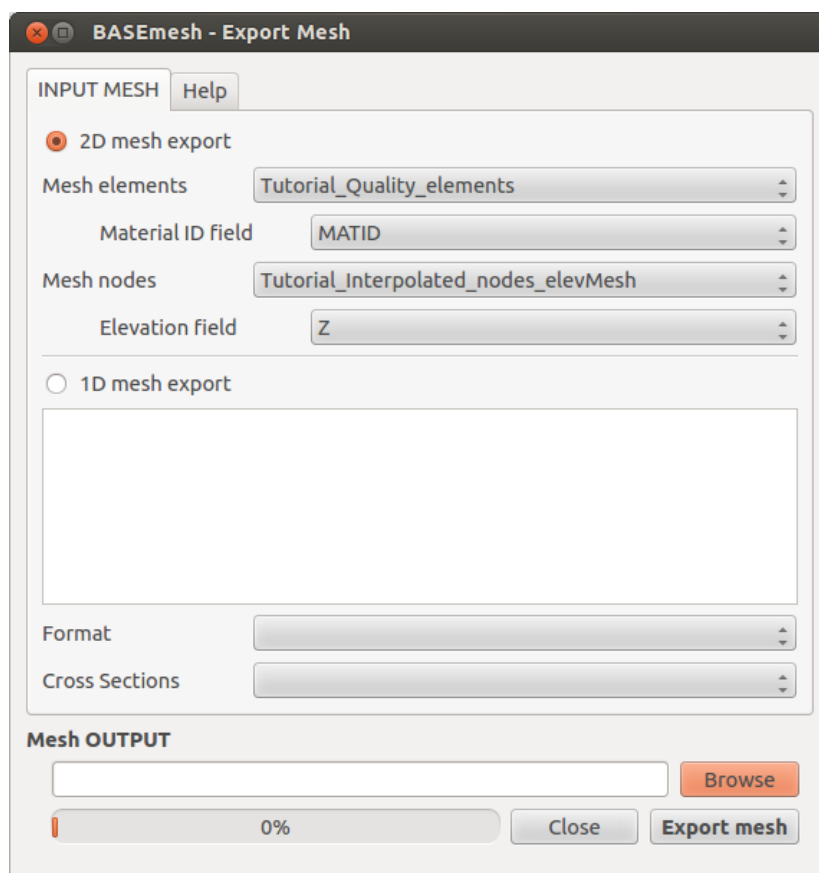
Figure 2.11 *Exporting a mesh from QGIS to file.*

2.1.9   Visualization of Simulation Results using QGIS (Post-Processing)

2.1.9.1   Visualization with the Crayfish plugin

Fast and easy visualizations of time dependent BASEMENT simulation results can be achieved using the Crayfish plugin developed by Lutra Consulting. The plugin is based on a fast engine which can deal with large and time dependent data sets in an efficient way. The plugin allows for plotting simulations results as adjustable color maps and offers additional visualization options like velocity vectors. Furthermore time dependent 2D results can be exported as animation.

How to do the Post-Processing using the Crayfish plugin is explained in detail in the Post-Processing tutorial.

2.1.9.2   Visualization with QGIS operations

The BASEMENT simulation results can also be visualized using QGIS core functions, mainly via two approaches:

1. **Shapefiles:** BASEMENT has the option to write results files in the shape-format with each time step as column in the attribute table. These data files can be directly loaded

into QGIS and visualized/colorized. A disadvantage of this visualization method is, however, that there is no smooth color transition between the discrete elements.

2. **Raster data:** Output data of BASEMENT (either as shapefiles or as ASCII-x-y-z data files) can be rasterized in QGIS. Hereby, the pointwise data values are interpolated on a raster. To rasterize the pointwise data values you can use the QGIS option *Raster → Interpolation*, which offers multiple interpolation algorithms, like TIN or inverse distance weighting. To colorize the raster data, open the properties dialog and select the option singleband pseudocolor and choose your preferred color map. Raster data are generally nice to work with, since GIS-applications have many and powerful tools to cope with raster data in various ways, e.g. to create contour lines or cross section profiles.
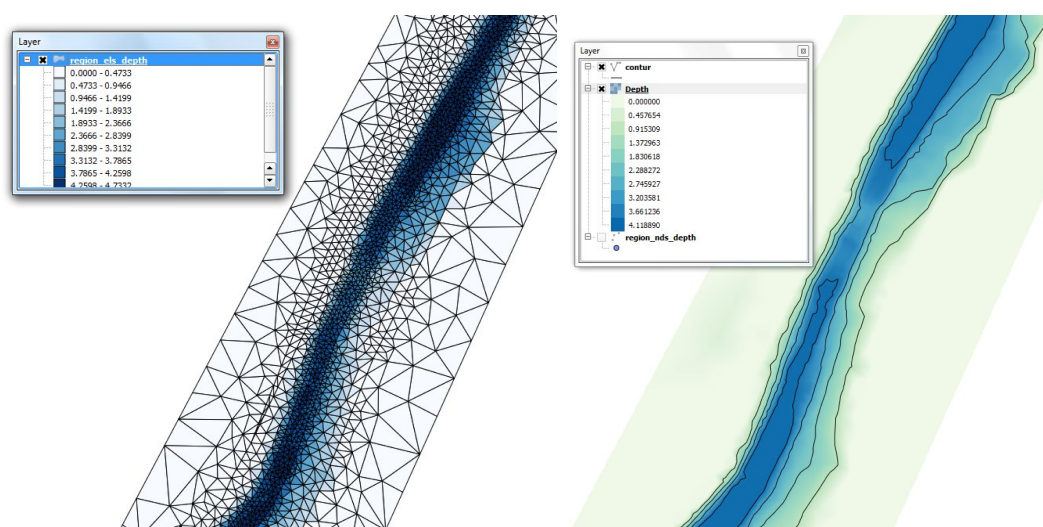

Figure 2.12 *Screenshot of data visualization using shapefiles (left) and raster data (right).*

It is important to note that output can be generated as 'element - centered' data (in the middle of elements) and as 'node - centered' data (at nodes). Most data values are computed on the elements in BASEMENT. Therefore 'element - centered' data usually equals exactly the computed values, whereas 'node - centered' data is interpolated onto the node coordinates.

Post-processing with Shapefiles or Raster data is usually more time consuming than using the Crayfish plugin, especially if time dependent data shall be visualized. However, it may be a good choice especially if high-quality plots and graphics of a specific simulation timestep are needed and the powerful mapping tools of QGIS shall be used.