

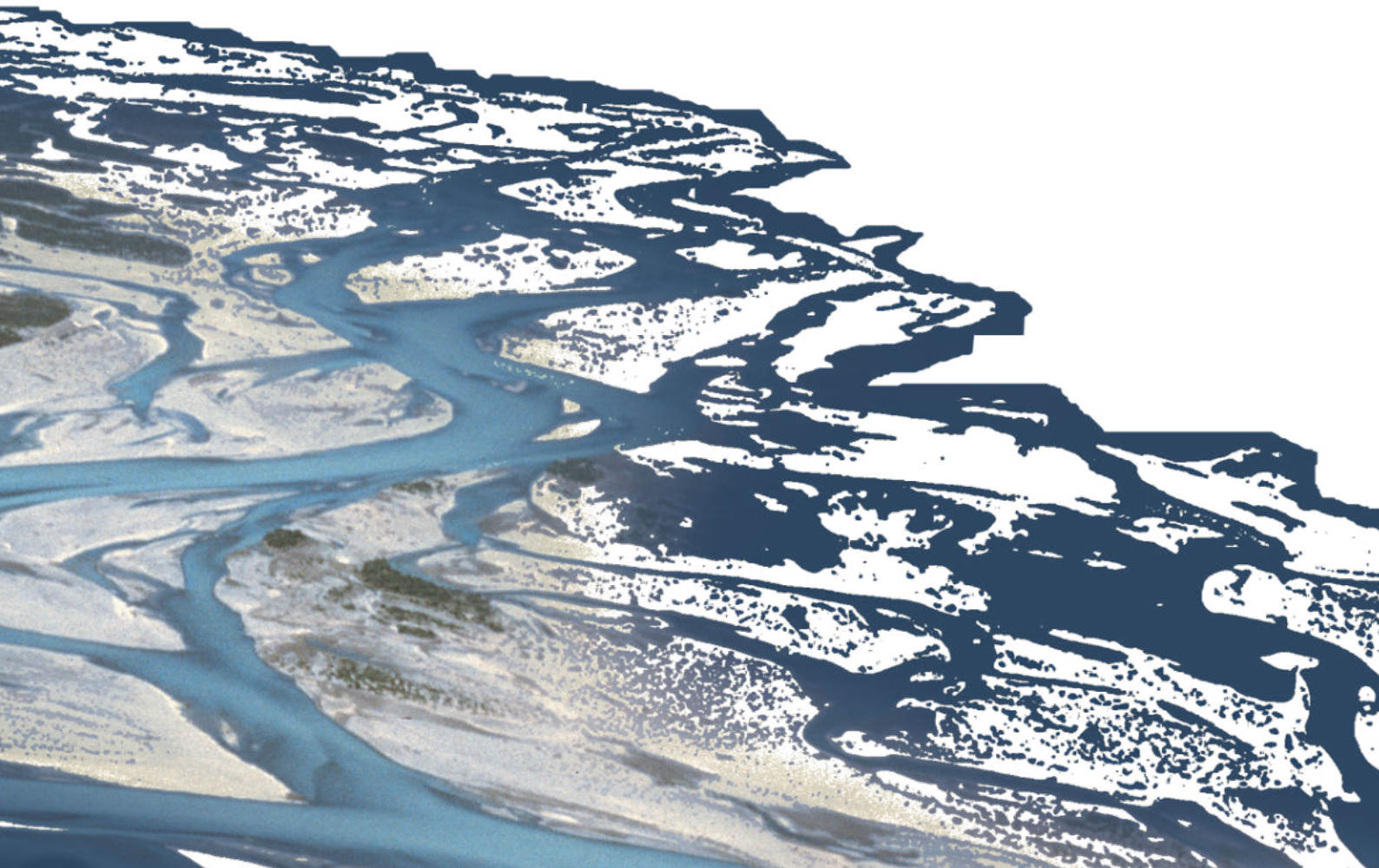


BASEMENT

**BASIC SIMULATION ENVIRONMENT
FOR MODELLING OF ENVIRONMENTAL
FLOWS AND NATURAL HAZARDS**

SYSTEM MANUALS

**VERSION 4.1.0
JUNE 2024**



Preamble

VERSION 4.1.0

June 2024

Credits

Contributors

Over the years, many enthusiastic engineers and developers have contributed to the development, testing and documentation of BASEMENT. An up-to-date overview of the current development team, along with current and former contributors, can be found on our website:

<https://basement.ethz.ch/people>

Commissioned and co-financed by

Swiss Federal Office for the Environment (FOEN)

Contact

Website: <https://www.basement.ethz.ch>

User forum: <https://people.ee.ethz.ch/~basement/forum>

ETH Zurich / Laboratory of Hydraulics, Glaciology and Hydrology (VAW)

For list of contributors see <https://www.basement.ethz.ch>



Laboratory of Hydraulics,
Hydrology and Glaciology



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Citation Advice

For System Manuals:

Vetsch D., Siviglia A., Bacigaluppi P., Brown, A., Bürgler M., Caponi F., Conde D., Gerke E., Halso M., Kammerer S., Koch A., Peter S., Vanzo D., Vonwiller L., Weberndorfer M. 2024. System Manuals of BASEMENT, Version 4.1.0. Laboratory of Hydraulics, Glaciology and Hydrology (VAW). ETH Zurich. Available from <https://www.basement.ethz.ch>. [date of access].

For Website:

BASEMENT – Basic Simulation Environment for Computation of Environmental Flow and Natural Hazard Simulation, 2024. <https://www.basement.ethz.ch>

For Software:

BASEMENT – Basic Simulation Environment for Computation of Environmental Flows and Natural Hazard Simulation. Version 4.1.0 ETH Zurich, VAW, 2024.

License

End-User License Agreement (EULA)

THIS EULA IS INTENDED FOR COMMERCIAL AND NON-COMMERCIAL PURPOSES. FOR QUESTIONS RELATED TO THIS AGREEMENT PLEASE CONTACT: Dr. David Vetsch, basement@ethz.ch

This End-User License Agreement (“EULA”) is a legal agreement between you (“You”) (an individual or acting on behalf of a company) and ETH Zurich, Raemistrasse 101, 8092 Zurich (Switzerland) (“ETH Zurich”) for the binary software code of **BASEMENT** and associated media, and may include “online” or electronic documentation (“SOFTWARE”).

The SOFTWARE simulates water flow, sediment and scalar transport in rivers and according interaction in consideration of movable boundaries and morphological changes. Further information and description of the SOFTWARE is available here: <https://basement.ethz.ch/>

The SOFTWARE is protected by copyright laws. The SOFTWARE is hereby licensed, not sold.

In order to install and use the SOFTWARE, You must indicate agreement with the following terms and conditions by clicking “ACCEPT” at the end of this EULA during the installation process.

1 LICENSE GRANT

- (i) ETH Zurich hereby grants to You, and in case You are acting on behalf of a company also to the employees of such company, a free-of-charge, single, non-exclusive, world-wide, non-transferable, non-sublicensable right to install, execute and display the SOFTWARE on device(s) running a validly licensed copy of the operating system for which the SOFTWARE was designed. Such rights are granted for commercial and non-commercial purposes.
- (ii) With respect to electronic documents included with the SOFTWARE, You may make an unlimited number of copies (either in hardcopy or electronic form), provided that such copies shall be used only for internal purposes and are not republished or distributed to any third party.

2 USE OF RESULTS GENERATED BY THE SOFTWARE

You are allowed to use the content generated by the SOFTWARE (“SOFTWARE RESULTS”) for commercial and non-commercial purposes.

Note that any attribution (e.g. ETH Zurich logo) on the SOFTWARE RESULTS must be retained. You are not allowed to alter, cancel or fade, after a few seconds, such attribution.

3 THIRD PARTY CODE

The SOFTWARE may contain other program code from third parties. A list of other third party code and libraries used by this SOFTWARE is available here: <https://basement.ethz.ch/about/thirdpartysoftware>

Their license applies to such third party code and libraries contained herein. Refer to the above internet site for the licenses and copyrights.

4 DURATION OF LICENSE AND TERMINATION

This EULA enters into effect on the date of acceptance of this EULA by You. This EULA,

- (i) may be terminated by ETH Zurich at any time for any reason;
- (ii) will terminate automatically without notice from ETH Zurich if (a) You fail to comply with any term(s) of this EULA or (b) You refuse, after the notification in accordance with clause 10 (i), to accept the new EULA term and conditions provided by ETH Zurich;
- (iii) is terminated as soon as You cease to use the SOFTWARE and destroy all copies, full or partial, of the SOFTWARE;

Upon termination pursuant to (i) and (ii), you must cease all use of the SOFTWARE and destroy all copies, full or partial, of the SOFTWARE.

5 OBLIGATIONS OF YOU

- (i) You may not remove or alter any copyright notices on any and all copies of the SOFTWARE.
- (ii) You may not distribute or assign the SOFTWARE or any copy thereof to third parties. You may not rent, lease, sell, lend, transfer, redistribute, or sublicense the SOFTWARE to any third party.
- (iii) You may not reverse engineer, decompile or disassemble the SOFTWARE, except and only to the extent that such activity is expressly permitted by applicable law despite this limitation.
- (iv) You may not reproduce, modify or adapt the SOFTWARE, except and only to the extent that such activity is expressly permitted by applicable law despite this limitation.
- (v) You must comply with all applicable laws.

6 OWNERSHIP

Except as expressly licensed to You in this EULA, ETH Zurich and its licensors retains all right, title, and interest in and to the SOFTWARE. All title and copyrights in and to the SOFTWARE (including but not limited to any images, photographs, animations, video, audio, music, text, and “applets” incorporated into the SOFTWARE), the accompanying materials, and any copies of the SOFTWARE are owned by ETH Zurich and its licensors. The SOFTWARE is protected by copyright laws. Therefore, You must treat the SOFTWARE like any other copyrighted material. All rights not expressly granted are reserved by ETH Zurich.

7 MAINTENANCE, SUPPORT, UPGRADES OR NEW RELEASES

ETH Zurich has no obligation to provide maintenance, support, upgrades, new releases, enhancements or modifications and disclaims all costs associated with service, repair or correction of the SOFTWARE. If any supplemental software code is provided to You by ETH Zurich, this supplemental software code shall be considered part of the SOFTWARE and is subject to the terms and conditions of this EULA if not otherwise explicitly written. It is expressly acknowledged by You that no rights to receive maintenance, support, upgrades, new releases, enhancements or modifications may be derived from this EULA.

8 NO WARRANTY

YOU EXPRESSLY ACKNOWLEDGE AND AGREE THAT USE OF THE SOFTWARE IS AT YOUR SOLE RISK AND THAT THE ENTIRE RISK AS TO SATISFACTORY QUALITY, PERFORMANCE, ACCURACY, AND EFFORT IS WITH YOU. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE SOFTWARE AND ANY SERVICES PERFORMED OR PROVIDED BY THE SOFTWARE ARE PROVIDED “AS IS” AND “AS AVAILABLE”, WITH ALL FAULTS AND WITHOUT WARRANTY OF ANY KIND, AND ETH ZURICH HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH RESPECT TO THE SOFTWARE AND ANY SERVICES, EITHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES AND/OR CONDITIONS OF MERCHANTABILITY, OF SATISFACTORY QUALITY, OF FITNESS FOR A PARTICULAR PURPOSE, OF ACCURACY, OF QUIET ENJOYMENT, AND OF NON-INFRINGEMENT OF THIRD-PARTY RIGHTS. ETH ZURICH DOES NOT WARRANT AGAINST INTERFERENCE WITH YOUR ENJOYMENT OF THE SOFTWARE, THAT THE FUNCTIONS CONTAINED IN OR SERVICES PERFORMED OR PROVIDED BY THE SOFTWARE WILL MEET YOUR REQUIREMENTS, THAT THE OPERATION OF THE SOFTWARE OR SERVICES WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE SOFTWARE OR SERVICES WILL BE CORRECTED. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY ETH ZURICH OR ITS AUTHORIZED REPRESENTATIVE SHALL CREATE A WARRANTY. SHOULD THE SOFTWARE OR SERVICES PROVE DEFECTIVE, YOU ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR, OR CORRECTION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES OR LIMITATIONS ON APPLICABLE STATUTORY RIGHTS OF A CONSUMER, SO THE ABOVE EXCLUSION AND LIMITATIONS MAY NOT APPLY TO YOU.

9 LIABILITY

In no event shall ETH Zurich be liable for any damages (including, without limitation, lost profits, business interruption, or lost information) arising from the use of or inability

to use the SOFTWARE through You or the employees of the company You are legally representing. In no event will ETH Zurich be liable for loss of data or for indirect, special, incidental, consequential (including loss of profit), or other damages based in contract, tort or otherwise.

The above limitation of liability shall not be construed to amend or limit any party's statutory liability.

10 GENERAL PROVISIONS

- (i) ETH Zurich reserves the right to change the terms and conditions of this EULA at any point in time. In such event, ETH Zurich will notify You in due time of the changes to the terms of the EULA.
- (ii) Rights and duties derived from this EULA shall not be transferred to third parties without the written acceptance of the ETH Zurich.
- (iii) You shall not infer from this EULA any other rights, including licenses, than those that are explicitly stated herein.
- (iv) This EULA shall exclusively be governed by and interpreted in accordance with the laws of Switzerland, without reference to its conflict of laws principles. The exclusive place of jurisdiction is Zurich (Switzerland).

11 ACKNOWLEDGMENT

You acknowledge that you have read this EULA, understand it, and had an opportunity to seek independent legal advice prior to agreeing to it. In consideration of ETH Zurich agreeing to provide the SOFTWARE, You agree to be bound by the terms and conditions of this EULA. You further agree that it is the complete and exclusive statement of the agreement between you and ETH Zurich, which supersedes any proposal or prior agreement, oral or written, and any other communication between you and ETH Zurich relating to the subject of this EULA.

Notice:

Third party software copyright notices and third party software licenses can be found in the appendix.

**BASIC SIMULATION ENVIRONMENT
FOR MODELLING OF ENVIRONMENTAL
FLOWS AND NATURAL HAZARDS**

TUTORIALS

**VERSION 4.1.0
JUNE 2024**



BASEMENT

Contents

1	Introduction to BASEmesh	5
1.1	Introduction to BASEmesh	5
1.2	Mesh Generation based on Pointwise Elevation Data	7
1.2.1	Project Settings	7
1.2.2	Coordinate Reference System Configuration	8
1.2.3	Loading Input Data for Elevation Model	8
1.2.4	Converting Input Elevation Data to 3D Points	9
1.2.5	Creating 3D Lines for Elevation Meshing	11
1.2.6	Loading the Breaklines	14
1.2.7	Editing the Model Boundary	14
1.2.8	Creation of the Elevation Mesh	16
1.2.8.1	Creation of the Elevation Mesh using 3D points	16
1.2.8.2	Creation of the Elevation Mesh using 3D Lines	17
1.2.9	Adaption of the Breaklines for Quality Meshing	17
1.2.10	Creation of Region Marker Points	20
1.2.11	Creation of Quality Mesh	22
1.2.12	Generation of the Computational Mesh	22
1.2.13	Viewing the computational mesh	23
1.3	BASEmesh Tips and Tricks	26
1.3.1	Recommended Plugins	26
1.3.2	Tips and Tricks	26
2	Post-Processing of 2D simulation results	31
2.1	Introduction	31
2.2	2D result visualization with QGIS	31
2.2.1	Input data	31
2.2.2	About Crayfish	32
2.2.3	Installation	32
2.2.4	Load and visualize data	33
2.2.4.1	Project Settings	33
2.2.4.2	Loading results of unsteady flow simulation (river Flaz)	33
2.2.4.3	Display properties	35
2.2.4.4	Export results to raster	37
2.2.4.5	Profile tool	38
2.2.5	Creating maps and animations	41
2.2.5.1	Customized layout (QGIS print Layout)	42
2.2.5.2	Export animation	45
2.3	3D result visualization with Paraview	48

2.3.1	Input data	48
2.3.2	About Paraview	48
2.3.3	Installation	48
2.3.4	User Interface	49
2.3.5	Import Data	50
2.3.6	ParaView Filters	50
2.3.6.1	Overview	50
2.3.6.2	Applying Filters to data	51
2.3.7	Exporting figures and animations	56
2.3.7.1	Save Screenshot (picture)	56
2.3.7.2	Save Animation (movie)	57
3	Hydrodynamics and sediment transport at the river Flaz with BASEHPC (2D)	59
3.1	Hydrodynamics and sediment transport at the river Flaz with BASEHPC (2D)	59
3.1.1	Introduction	59
3.1.2	Computational Mesh	59
3.1.3	Hydraulics	60
3.1.3.1	Setup the Configuration File model.json	60
3.1.3.2	Setup the Configuration File simulation.json	63
3.1.3.3	Set up the Configuration File results.json	63
3.1.3.4	Steady Flow Simulation	64
3.1.3.5	Model Calibration	64
3.1.3.6	Unsteady Flow Simulation	64
3.1.4	Morphology with uniform sediment transport	66
3.1.4.1	Setup the Configuration File model.json	66
3.1.5	Morphology with mixed-size sediment transport	71
3.1.5.1	Setup the Configuration File model.json	71
3.1.5.2	Setup the Configuration File simulation.json	75
3.1.5.3	Results	75
3.1.6	Passive tracers	75
3.1.6.1	Setup the Configuration File model.json	75
3.1.6.2	Setup the Configuration File simulation.json	78
3.1.6.3	Set up the Configuration File results.json	78
4	Hydrodynamics and sediment transport at the river Flaz with BASEMD (2D)	81
4.1	Hydrodynamics and sediment transport at the river Flaz with BASEMD (2D)	81
4.1.1	Introduction	81
4.1.1.1	Case study description	81
4.1.1.2	Tutorial structure	82
4.1.2	Computational grid	82
4.1.3	Setting up the model file	84
4.1.3.1	BASEMD	84
4.1.3.2	Physical properties	85
4.1.3.3	Two dimensional domain	85
4.1.3.4	Geometry	85

4.1.3.5	Define the hydraulics	86
4.1.3.6	Write the setup file	90
4.1.3.7	Perform hydraulic simulations	90
4.1.4	Morphological simulation with single-grain bed load transport	95
4.1.4.1	Define the morphological information	96
4.1.4.2	Bedload	100
4.1.4.3	Gravitational transport	102
4.1.4.4	Define the output	102
4.1.4.5	Perform morphological simulation with single-grain bed load transport	103
4.1.5	Morphological simulation with multi-grain bed load transport	103
4.1.5.1	Morphological parameters	103
4.1.5.2	Grain size distribution	106
4.1.5.3	Grain mixture	106
4.1.5.4	Define the soil composition	107
4.1.5.5	Bed load boundary condition	108
4.1.5.6	Bed load formula	108
4.1.5.7	Define the output	109
4.1.6	Perform morphological simulation with multi-grain bed load transport	109
5	Hydrodynamics and sediment transport at the river Thur with BASEMD (1D)	111
5.1	Hydrodynamics and sediment transport at the river Thur with BASEMD (1D)	111
5.1.1	Introduction	111
5.1.1.1	General description	111
5.1.1.2	Used features	111
5.1.1.3	Purpose	112
5.1.2	Setting up the topography file	112
5.1.2.1	Cross sections	112
5.1.2.2	Definition of different cross section zones	114
5.1.2.3	Friction values	114
5.1.2.4	Computation of water surface elevation	116
5.1.2.5	Characterization of the sediments	116
5.1.2.6	Define flowing zones	117
5.1.3	Setting up the model file	118
5.1.3.1	Setup	118
5.1.3.2	Domain	118
5.1.3.3	Define the physical properties	118
5.1.3.4	One dimensional simulation	118
5.1.4	Perform hydraulic simulations	121
5.1.4.1	Write the setup file	121
5.1.4.2	Setting up the simulation file	121
5.1.4.3	Perform steady flow simulation	122
5.1.4.4	Results of the steady flow simulation	122
5.1.4.5	Perform simulation of the floods (Thur2)	123
5.1.5	Edit the model file for bed load transport	125
5.1.5.1	Define the bed material	126
5.1.5.2	Soil assignment	127
5.1.5.3	Define general parameters for sediment transport	127

5.1.5.4	Define specific parameters for bed load transport	128
5.1.5.5	Define bed load transport formula	128
5.1.5.6	Define boundary conditions for bed load	128
5.1.5.7	Generate a “geometry” file	129
5.1.6	Perform bed load simulation (Thur 3)	129
6	Dynamics of longitudinal bed profile due to local river widening with BASEMD (1D)	131
6.1	Dynamics of longitudinal bed profile due to local river widening with BASEMD (1D)	131
6.1.1	Introduction	131
6.1.1.1	General description	131
6.1.1.2	Purpose	132
6.1.1.3	Used features	133
6.1.1.4	Parameter variation	134
6.1.2	Model setup	134
6.1.2.1	Definition of 1D topography	134
6.1.2.2	Determination of the upstream sediment boundary conditions	135
6.1.3	Results of numerical simulations	136
6.1.3.1	Temporal evolution of longitudinal bed profile	136
6.1.3.2	Width and length of the widening	136
6.1.3.3	Discharge and grain diameter	136
7	Laterally coupled 1D-2D hydrodynamic simulation with BASEMD	141
7.1	Laterally coupled 1D-2D hydrodynamic simulation with BASEMD	141
7.1.1	Introduction	141
7.1.2	Set-up of model file	142
7.1.2.1	General remarks on mesh creation	142
7.1.2.2	BASEchain (1D) river model	144
7.1.2.3	BASEplane (2D) floodplain model	145
7.1.2.4	Lateral coupling set-up	146
7.1.3	Coupling connections between 1D and 2D subdomains	146
7.1.3.1	Definition of coupling connections	146
7.1.3.2	Automatic generation of coupling connections	147
7.1.4	Perform coupled simulation	149
8	Vegetation dynamics and bedload transport with BASEHPC	151
8.1	Vegetation dynamics and bedload transport with BASEHPC	151
8.1.1	Introduction	151
8.1.2	Computational Mesh	151
8.1.3	Morphology with Vegetation	151
8.1.3.1	Setup the Configuration File model.json	152
8.1.3.2	Setup the Configuration File simulation.json	154
8.1.3.3	Set up the Configuration File results.json	155
8.1.3.4	Results	155
9	References	157

Introduction to BASEmesh

1.1 Introduction to BASEmesh

What is the goal of these tutorials? The following tutorials introduce to the creation of computational meshes using BASEmesh and the integrated mesh generator Triangle. Besides mesh generation, features for loading and editing existing meshes are presented. Not all features of BASEmesh can be covered, but using the available tutorials will give you an impression of its workflow and its capabilities. For specific questions, a help function describing the necessary input layers and parameters is included in every tool of BASEmesh and can be accessed under the tab Help. Furthermore, these tutorials introduce to several basic GIS - operations using QGIS. As all features and aspects of QGIS cannot be covered, we recommend the excellent documentation of [QGIS](#) for specific features or tasks.

What is the philosophy of mesh creation with BASEmesh? BASEmesh is a free and open source pre- and postprocessing QGIS plugin. It generates meshes for the numerical simulation software [BASEMENT](#) using Jonathan Richard Shewchuk's mesh generator 'Triangle'. The focus of BASEmesh is on the automatic generation of unstructured meshes based on specific quality criteria. BASEmesh follows the philosophy of separating the tasks of high-quality mesh generation and the generation/use of elevation models.

What is the workflow of BASEmesh? Please take a look at the figure below to see the workflow in BASEmesh. There are 3 main steps of the mesh generation process: (1) Create Elevation Mesh (2) Create Quality Mesh (3) Create Computational Mesh

For further details, please refer to Section 2.2 "2D grid generation with BASEmesh QGIS plugin" of the BASEMENT User Manual.

This tutorial covers the following BASEmesh workflow:

- Tutorial 1: Creation and export of a computational mesh for BASEMENT using pointwise elevation data.

Future tutorials will cover the following workflows in BASEmesh:

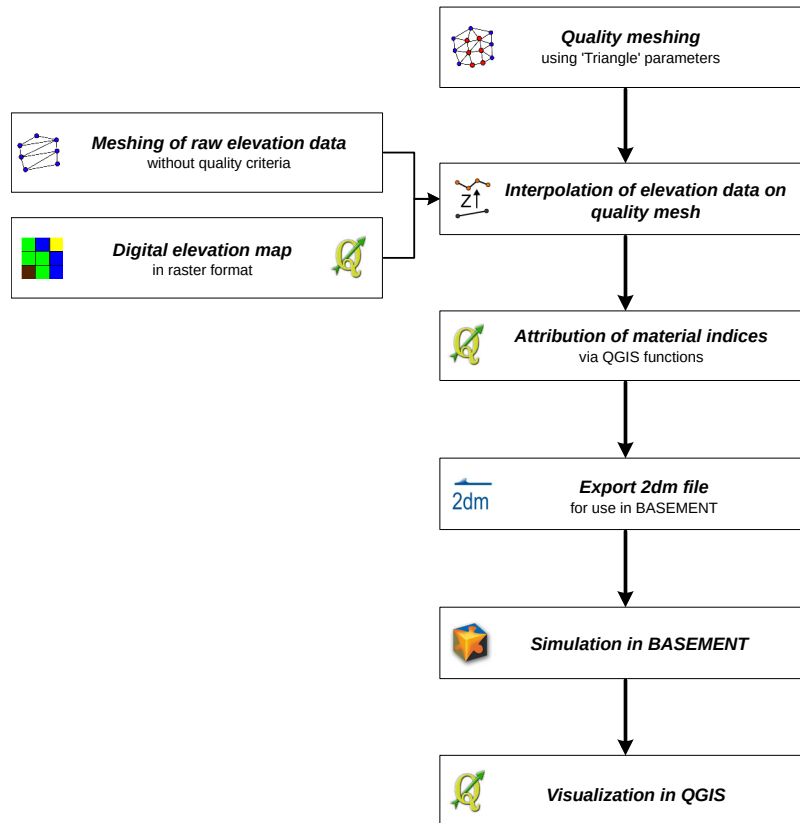


Figure 1.1 General workflow for the creation of a computational mesh with BASEmesh.

- Tutorial 2: Modifying of a computational mesh and using raster elevation data.
- Tutorial 3: Using dividing constraints along boundary cross sections and setting up a BASEMENT simulation.
- Tutorial 4: Create 1D cross sections with HEC-RAS and use BASEmesh to convert them into BASEMENT format.

1.2 Mesh Generation based on Pointwise Elevation Data

The following tutorial illustrates the consecutive steps to create a high - quality computational mesh based on pointwise elevation data stored in a text file. The elevation data in this tutorial is represented by cross section data, gathered in a river restoration project in Switzerland by terrestrial survey. All other files have been edited or created based on this elevation data. Therefore this tutorial exemplifies the mesh generation based on given river cross section data. Another typical task, e.g. for flood simulations with overland flow, is the generation of meshes based on a digital elevation model (DEM) in raster format .

Input Data The data needed to complete this tutorial comes as ZIP - file and needs to be extracted to a location of your choice. All screenshots and figures in this document were taken from QGIS version 3.16 Hannover. All data files have to be loaded into the QGIS - project before executing the different tools, as it is not possible to select files directly on the hard drive by browsing. Furthermore, those files must be activated in the QGIS Table of Contents (TOC- this abbreviation will be used often in this tutorial) on the left side of the screen. To prevent the selection of wrong shapetypes, the available fields are populated with the corresponding data type.

Rule of thumb Only data that is displayed on the map can be used for meshing.

Only 2 shapefiles are needed as input data for this tutorial. All other input files are created using BASEmesh and other QGIS functions. In case of difficulties, the result files for each step are additionally provided in the subfolder called 'additional_files'.

1.2.1 Project Settings

- (1) Start QGIS and make sure that the plugin BASEmesh is successfully installed (see Section 2.2.1 of the BASEMENT User Manual).

It is advisable to create a QGIS project with a meaningful name. The project's name is used as basis for most of the files created with BASEmesh.

- (2) Go to *Project* → *Project Properties*.
- (3) Under *General* → *General settings* you will find the field *Project title*. Enter a name of your choice. Here the name 'Tutorial' was chosen.

In this tutorial we need the Swiss projection 'CH1903 / LV03'. In the following steps, the project's coordinate reference system (CRS) will be changed.

- (4) Under *CRS* you can see the coordinate reference system settings for this project. Check *Enable 'on the fly CRS' transformation* only to change the reference system.
- (5) Enter the EPSG code '21781' in the field *Filter*.

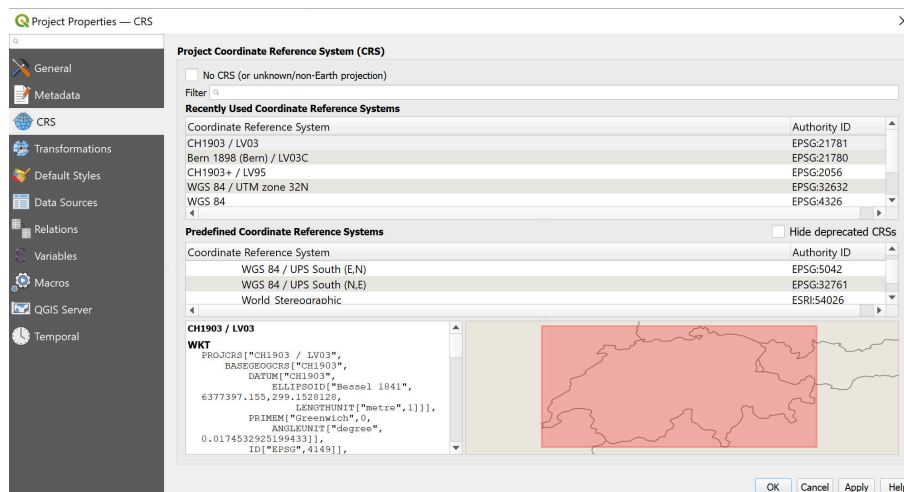


Figure 1.2 Project Properties, CRS

- (6) Select the coordinate reference system ‘CH1903 / LV03’.
- (7) Click *OK*.
- (8) Again go to *Project* → *Project Properties* → *CRS*.
- (9) Close the project properties window. If everything went well, you should see the chosen project name at the title of the QGIS main window and the EPSG code of your coordinate system at the lower right corner of the QGIS desktop.
- (10) Now save the project: Go to *Project* → *Save*. The project name chosen before is automatically proposed.

1.2.2 Coordinate Reference System Configuration

Now we need to check how QGIS determines the coordinate system of added layers. This may vary between QGIS versions and operating systems.

- (1) Go to *Settings* → *Options*
- (2) Under *CRS* fill in the dialog as shown in Figure 1.3. Make sure that *Use project CRS* under *CRS for new layers* is checked.
- (3) Close the dialog.

Now that the project and coordinate reference system properties have been set successfully, we can start adding data to the current project.

1.2.3 Loading Input Data for Elevation Model

In this tutorial the elevation model is represented by a triangulated irregular network TIN, in the following referred to as ‘elevation mesh’. Thus, our first step is loading the elevation data into our QGIS project. This data originates from cross sectional data and is stored in a delimited text file, where the values are separated by a comma. We will load this text file and convert it to a shapefile, which is more suitable for working in a GIS environment.

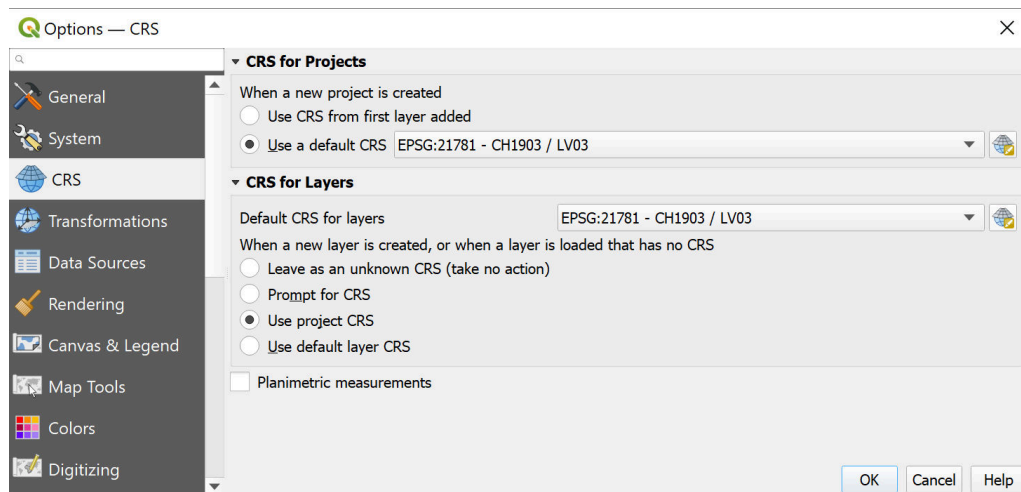


Figure 1.3 Settings, CRS

- (1) Go to *Layer* → *Add layer* → *Add Delimited Text Layer*.
- (2) Fill in the dialog that shows up (see Figure 1.4): Browse for the file containing the elevation data, ‘XS_points_straightened.txt’ provided with this tutorial. Be sure to select *Comma* as delimiter. Otherwise QGIS will not be able to separate the values found in the file. Under *Geometry definition* select *Point coordinates* and verify if the X, Y, and Z coordinates are set correctly.
- (3) Be sure to check *Use spatial Index*.
- (4) At the bottom of the dialog a preview of the file content using the selected delimiter and coordinate fields is given. If everything seems to be correct and according to Figure 1.4, click *OK*.
- (5) After successful import you will see the river cross sections displayed in the QGIS map canvas. The flow direction is from the bottom left corner to the upper right corner.

1.2.4 Converting Input Elevation Data to 3D Points

The loaded elevation data must be converted into a format that can be used by BASEmesh. We will first convert the data into a shapefile.

- (1) Right-click on layer ‘XS_points_straightened’ in the TOC.
- (2) Go to *Export* → *Save Features As...*
- (3) Fill in the dialog according to Figure 1.5 and click *OK*.
- (4) You will now see two layers in the TOC with the same name. The layer that represents the data from the text file can now be deleted by right-clicking on it in the TOC and selecting *Remove Layer*. If both layers have the same symbology, you can check its properties by right-clicking and selecting *Properties*.

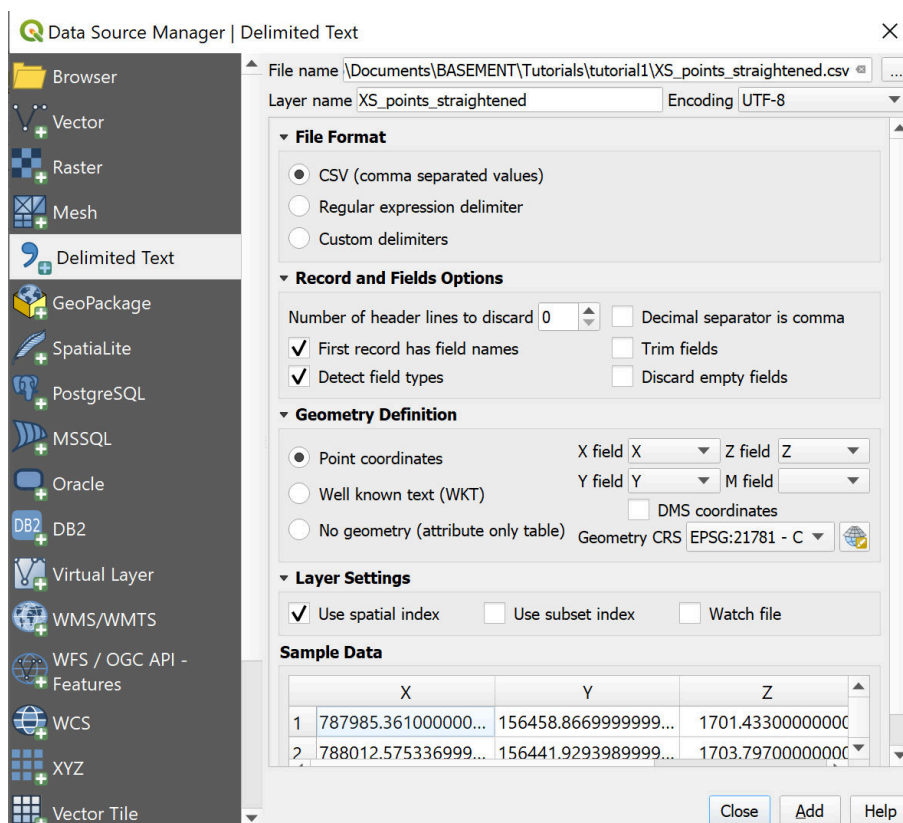


Figure 1.4 Add delimited text layer

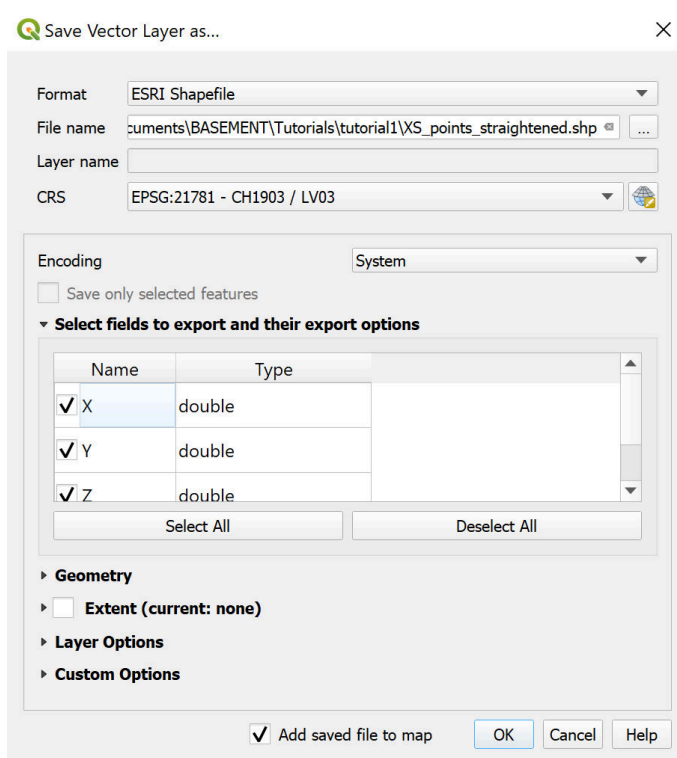


Figure 1.5 Save vector layer

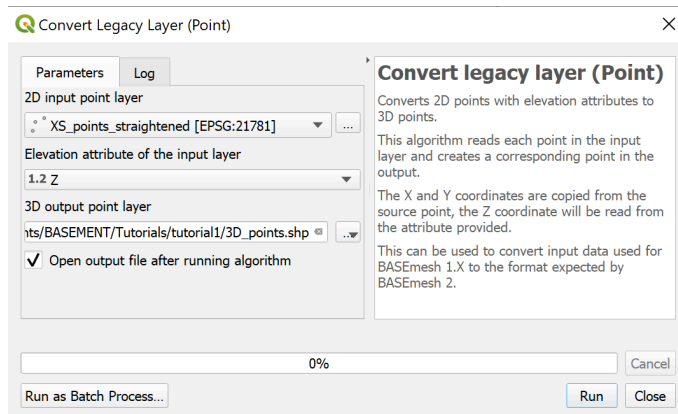


Figure 1.6 Convert points to 3D

The shapefile that we just generated contains 2D points. Now we will convert the 2D points into 3D points.

- (1) Open the BASEmesh “Convert legacy layer (Point)” tool. *Processing Toolbox* → *BASEmesh* → *Converters* → *Convert legacy layer (point)*.
- (2) Fill in the dialog according to Figure 1.6 and click run.

This 3D points file can be used in the BASEmesh *Elevation Meshing* tool.

1.2.5 Creating 3D Lines for Elevation Meshing

We will now use the elevation points to create 3D lines, which can also be used by the BASEmesh *Elevation Meshing* tool. First, we create a digital elevation map using the points shapefile.

- (1) Open the Natural neighbour tool from SAGA.
- (2) Fill in the dialog according to Figure 1.7 and click run (this may take a few minutes)

Now we will use the elevation points to create 2D lines, which represent each cross section of the river.

- (1) Open the QGIS points to path tool.
- (2) Fill in the dialog according to Figure 1.8 and click run (this may take a few minutes)

You will now see lines representing each cross section, plotted on top of the DEM, as in Figure 1.9. The 2D lines can now be converted to 3D lines, using the DEM.

- (1) Open the QGIS drape tool
- (2) Fill in the dialog according to Figure 1.10 and click run

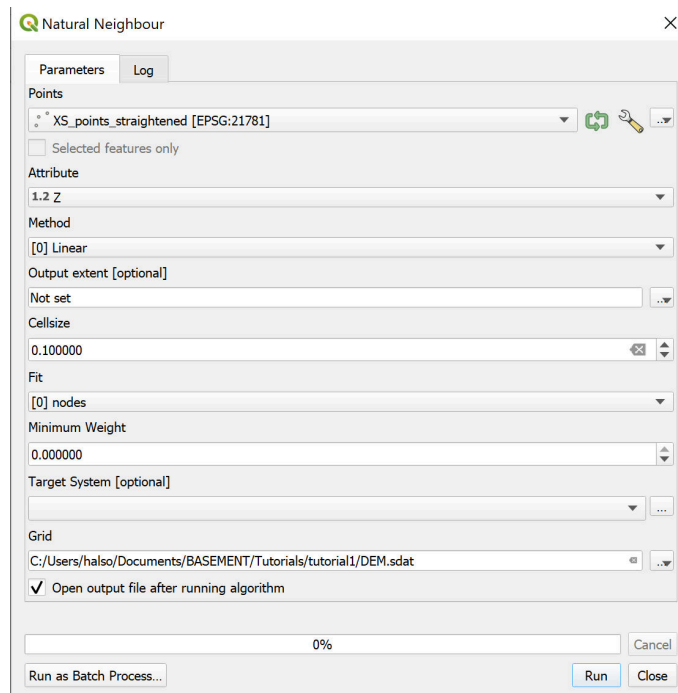


Figure 1.7 Natural Neighbor

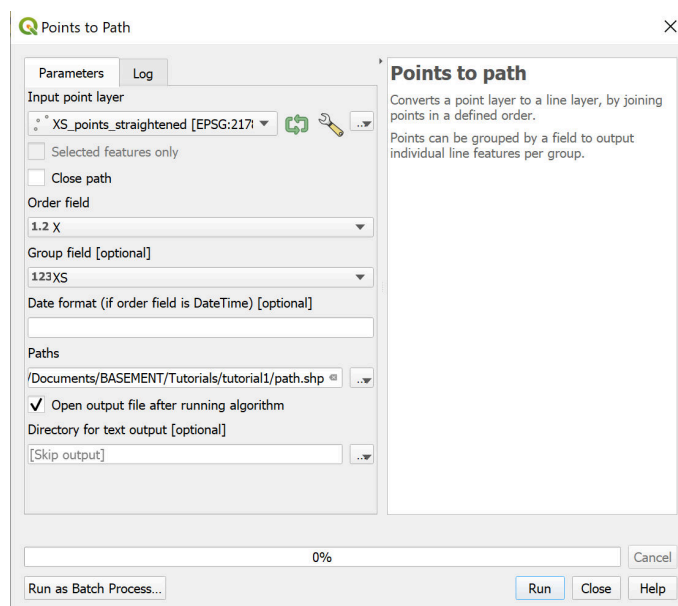


Figure 1.8 Points to Path

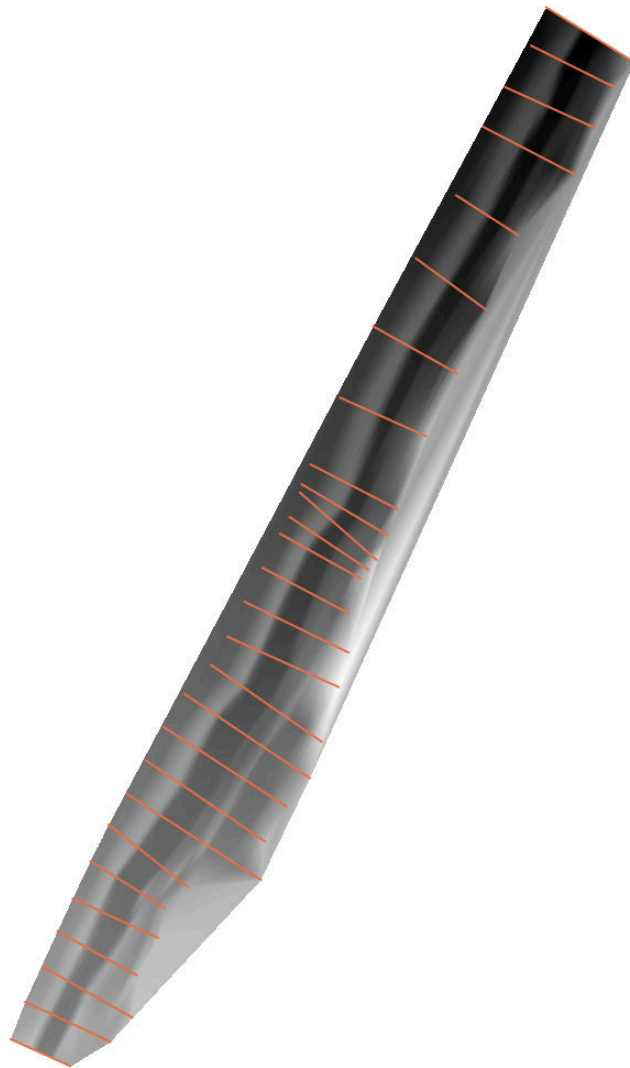


Figure 1.9 2DLines on DEM

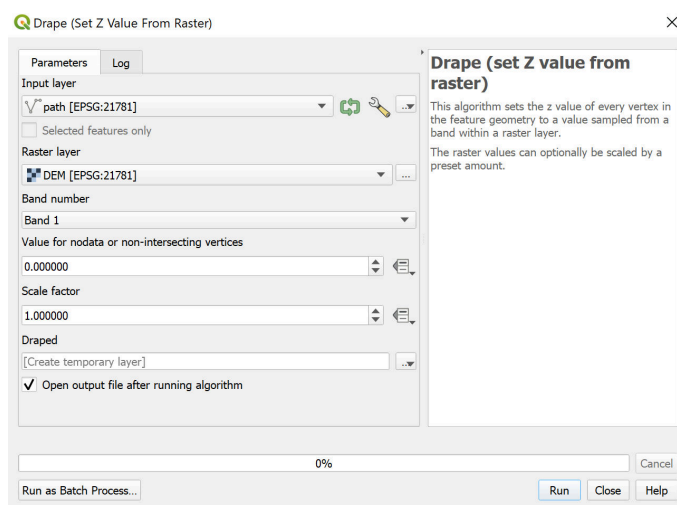


Figure 1.10 Tutorial 1.1 Drape

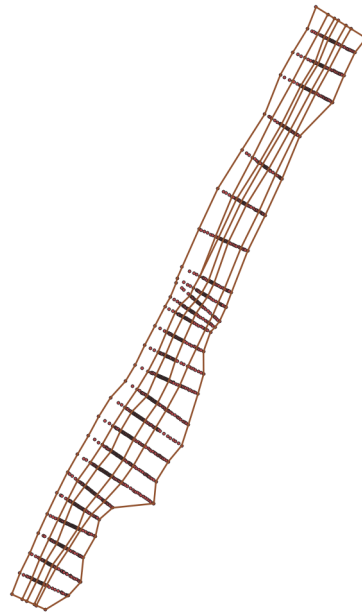


Figure 1.11 Breaklines

1.2.6 Loading the Breaklines

- (1) Go to *Layer* → *Add Layer* → *Add Vector Layer...* (or Ctrl-Shift-V).
- (2) Browse for the shapefile provided with this tutorial ‘breaklines.shp’.
- (3) Click *Add*.

The color schemes of newly added layers are randomly chosen by QGIS. You can change them by double-clicking on a layer and selecting *Symbology*.

- (4) Pull the newly added layer ‘breaklines’ below the initial file ‘XS_points_straightened’. Your QGIS canvas should now look like in Figure 1.11.

This breaklines shapefile contains several features, including one labeled ‘model_boundary’. This feature gives the extents of the computational domain. Please note that for generating a computational mesh, the model boundary should not lie outside of the elevation points. Otherwise, there might be interpolation errors in the following meshing steps. A suitable model boundary can be created by using the *Convex hull(s)* feature of QGIS. For more information, please refer to the Tips and Tricks (Section 1.3) of this Tutorial.

1.2.7 Editing the Model Boundary

All vertices of the model boundary polygon **must** lie within the extents of the elevation points. Otherwise there might be interpolation errors in the following steps. In our example, a portion of the model boundary does not satisfy this criteria (see lower right region of the model data). Therefore we have to move/add nodes of the boundary layer:

- (1) Right-click on layer ‘breaklines’ in the TOC and go to *Toggle Editing*. The layer can now be edited.

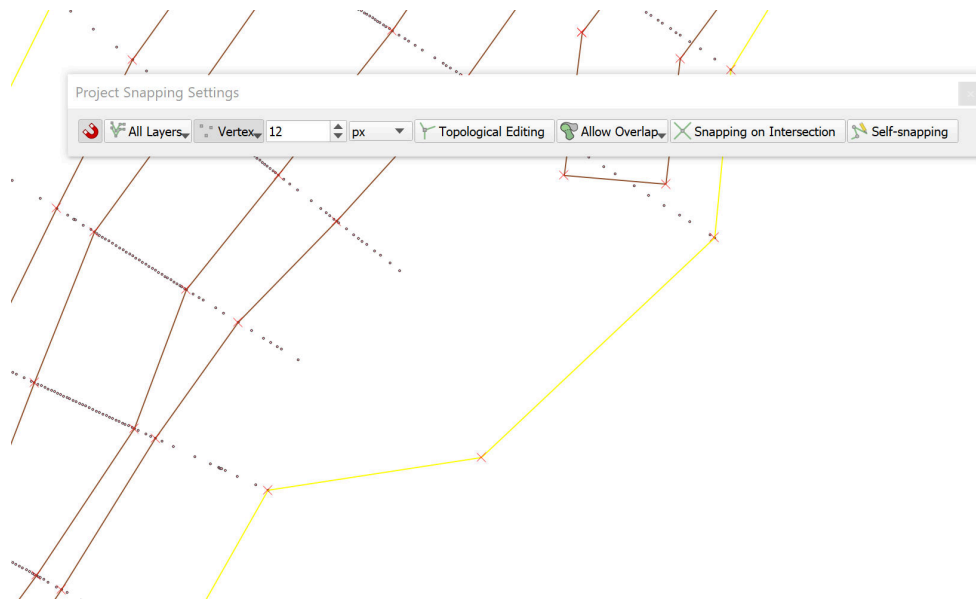


Figure 1.12 Snapping

- (2) Zoom into the area of interest. You should see small red 'x' for each polygon vertex.

We must ensure that the new/moved vertex will lie on an elevation point:

- (3) Go to *Project* → *Snapping Options*. Fill in the form as shown in Figure 1.12 and click *OK*. Depending on your QGIS version and installation, the window with snapping and digitizing options might be docked. In this case, your edits are applied immediately and there are no *Apply* or *OK* - buttons available.
- (4) Go to *Edit* → *Vertex Tool*.
- (5) Click on the vertex of the boundary which is free. The polygon feature gets selected (all vertices turn to a red square).
- (6) Drag & drop the free vertex to a neighbouring elevation points (see Figure 1.12).

We need to add a vertex to the polygon to include the remaining cross section end-point into the boundary:

- (7) Hover over the segment somewhere between the two boundary vertices where you want to have the new one. In the middle of the segment, a small plus sign (+) will appear. Click on that plus sign to create a new node. (Be careful with clicking: as you may move the entire segment by mistake. This can lead to meshing errors or very fine triangulations in the following steps.)
- (8) Again, drag and drop the vertex to the wished position.
- (9) We are done with editing, go to *Layer* → *Toggle Editing*. Click *Save*.

The breakline shapefile is now ready for Quality Meshing. But first, the Region Markers must be defined.

1.2.8 Creation of the Elevation Mesh


Using the data that we have loaded and processed, we can now create the elevation model as a triangulated irregular network using the plugin BASEmesh. We call it *Elevation Meshing*. There are 3 methods for Elevation Meshing:

- (1) Creation of the Elevation Mesh with 3D points
- (2) Creation of the Elevation Mesh with 3D lines
- (3) Using a DEM as the Elevation Layer

In this tutorial, we will show how to perform the first 2 methods.

1.2.8.1 Creation of the Elevation Mesh using 3D points

First, we will create an elevation mesh using the 3D lines that were created in Section 1.2.4.

- (1) Go to *Plugins* → *BASEmesh* → *Elevation meshing*, or click the respective button  in the toolbar.

On the left side of the dialog (Figure 1.13) you can define the input layers. See Section 2.2 “2D grid generation with BASEmesh QGIS plugin” of the BASEMENT User Manual for further explanations. On the right side status messages as well as Triangle’s output messages are displayed during meshing. Tool-specific help can be found in the *Help*-tab.

- (2) Select the different fields according to Figure 1.13.
- (3) Choose an output filename. The default is the name of the project.
- (4) Click *run*.

This newly generated elevation mesh is an intermediate step and will be used as basis for the further interpolation of the elevation data. Due to its low mesh quality, it should **not** be used as computational mesh for any numerical simulations!

At this point it is advisable to check whether the *Elevation meshing* worked correctly. To do so:

- (5) Right-click on the new layer ‘Tutorial_Elevation_nodes.shp’ in the TOC and go to *Open Attribute Table*.
- (6) Check if an elevation (column *Z*) has been assigned to every single node of the layer.
- (7) In case of a *Z*-value equals 0, you defined a vertex in your layer ‘model_boundary’ and/or ‘breaklines_elev_mesh’ which doesn’t lie exactly on an elevation point.
- (8) Check the vertices at the location of the respective points missing an elevation and make sure to use the snapping option.

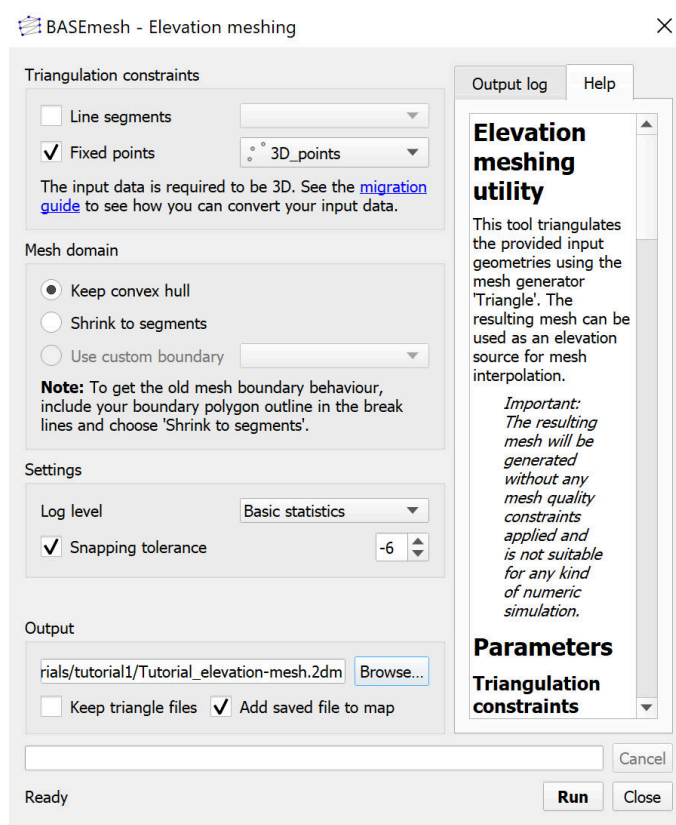



Figure 1.13 BASEmesh Elevation Meshing

- (9) If necessary, Repeat the *Elevation meshing* (steps 1 - 4).

In case the problem cannot be solved by redoing the snapping, try to vary the *Relative snapping Tolerance* in the *Elevation meshing* dialog (Figure 1.13). Increase the tolerance and try a value of -3 for example. Check the Tipps and Tricks (Section 1.3) for further explanations on this important parameter.

1.2.8.2 Creation of the Elevation Mesh using 3D Lines

Now we will create an elevation mesh using the 3d points that were created in Section 1.2.4.

- (1) Go to *Plugins* → *BASEmesh* → *Elevation meshing*, or click the respective button  in the toolbar.
- (2) Select the different fields according to Figure 1.14.
- (3) Follow the same steps as in the previous section.

1.2.9 Adaption of the Breaklines for Quality Meshing

For most tasks, quality meshing requires the same basic breaklines as elevation meshing does. Nevertheless, some content might be added, e.g. building outlines or lines along which

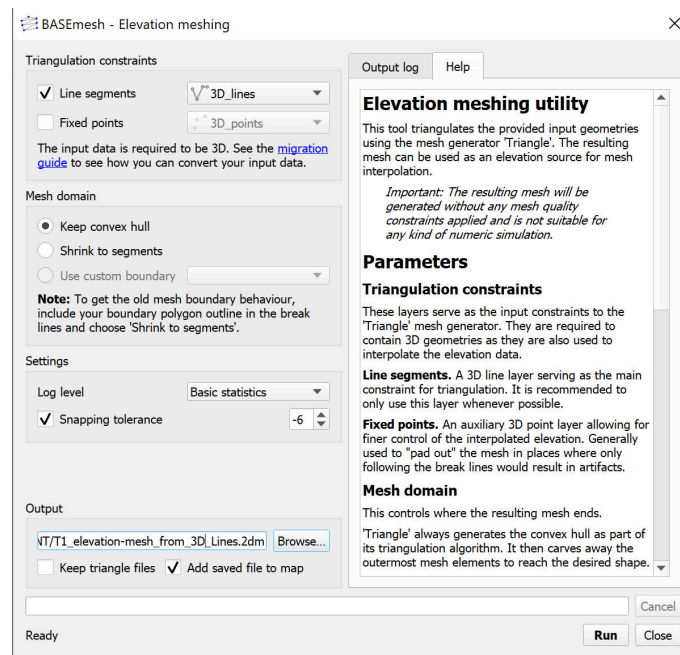


Figure 1.14 BASEmesh Elevation Meshing with 3D Lines

we wish to have special outputs from the future numerical computations. In the following, the breakline layer used before will be duplicated and a building outline will be added:

- (1) Right-click on the layer 'breaklines' in the TOC and go to *Export* → *Save Feature as...*
- (2) In the next dialog choose an appropriate name and location for the new layer (e.g. 'breaklines_w_building.shp'). Be sure to check *Add saved file to map* at the bottom of the dialog.
- (3) Load the shapefile 'building.shp' into QGIS (provided with this tutorial).
- (4) Select the layer 'building' in the TOC.
- (5) Go to *View* → *Select* → *Select Single Feature*.
- (6) Click on the building outline in the map (it should get colored differently).
- (7) Go to *Edit* → *Copy Features* (or Ctrl-C).
- (8) Right-click on the layer 'breaklines_qual_mesh' in the TOC and go to *Toggle Editing*.
- (9) Go to *Edit* → *Paste Features* (or Ctrl-V).
- (10) As copying the new line feature is completed, select *Edit Toggle Editing*. QGIS asks to save the changes. Click *Save*.
- (11) Having activated only the 'breaklines_w_building' layer in the TOC, the lower part of your model should now look like Figure 1.15. If the building outline is still highlighted, select *View Select Deselect Features from all Layers*.

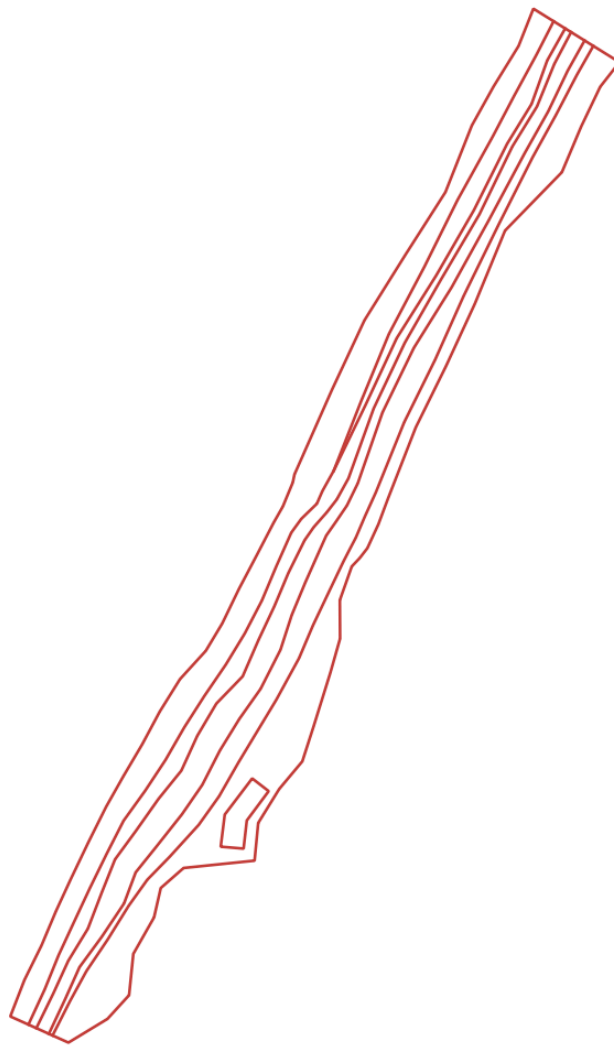


Figure 1.15 Breaklines and building

1.2.10 Creation of Region Marker Points

In this new Point-Shapefile, three attributes are defined that will be used for the *Quality meshing* (Figure 1.18) afterwards: *maximum area*, *material index* and *holes*. The attributes have to be set individually for each region, which is embraced by breaklines or boundaries and are specified by placing a point into this area. Be careful, a misplaced region marker point can lead to very fine and computationally intensive meshes or wrong definition of Material Indexes respectively.

maximum area: The area attribute of the layer constraints the maximum area of the elements created during *Quality meshing*. If left blank, no maximum area constraints will be taken into account for this particular region (see Figure 1.17)

material index: This attribute determines the material index of a certain region. If left blank, the material index is by default set to 1. The material indices are used in BASEMENT to group elements into zones with similar properties, e.g. to set different friction values or soil properties in certain mesh regions. These material indices are stored in the attribute field *MATID* of the mesh elements layer during mesh generation.

holes: These points define regions that will be neglected during meshing. These areas will be cut out and therefore not be integrated in the final mesh, preventing water flow through these regions.

- (1) Create a new point layer: Go to Layer → Create Layer → New Shapefile Layer... .
- (2) Fill in the form as shown in Figure 1.16. Be sure to define the correct CRS (EPSG:21781).
- (3) Add an attribute for the *maximum area* with name e.g. ‘max_area’ and type ‘Decimal Number’ (e.g. *Length*=10 and *Precision*=3).
- (4) Add a second attribute for the *material index* with name ‘MATID’ and type ‘Whole Number’ (e.g. *Length*=3).
- (5) Add a third attribute for *holes* with name ‘hole’ and type ‘Whole Number’ (e.g. *Length*=1*).
- (6) Optionally add another attribute with name ‘Type’ and type ‘Text data’ (e.g. *Length*=20) to assign a specific description for each region (e.g. ‘River bed’, ...)
- (7) Save the layer with name ‘regions_points’.
- (8) Right-click on the layer ‘regions_points’ in the TOC and go to Toggle Editing.
- (9) Go to *Edit* → *Add Feature*
- (10) Add six features (points) inside the regions displayed in Figure 1.17
- (11) Click somewhere inside the particular region. Enter an arbitrary *id* and fill in the attributes like shown in Figure 1.17
- (12) Go to Layer → Toggle Editing and save the changes.

The value of the attributes ‘max_area’ (left) and ‘MATID’ (right) are used as label. In this example three different region ‘Types’ have been defined: ‘river_bed’, ‘embankment’ and ‘surrounding’.

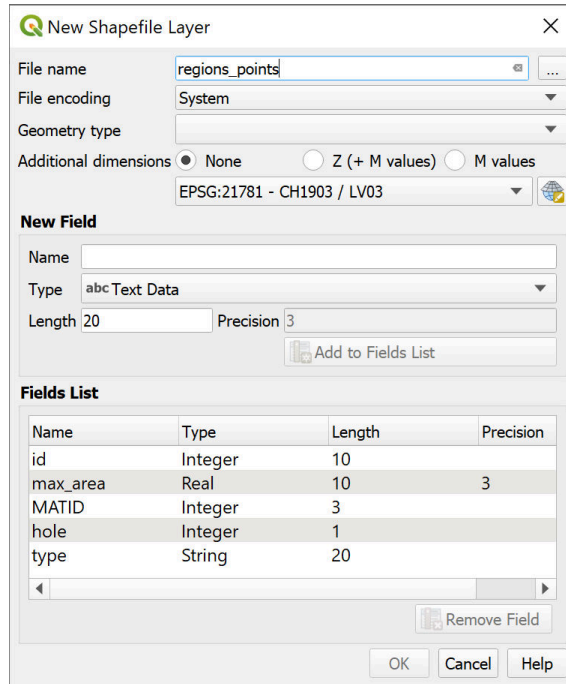


Figure 1.16 New point layer 'region_points'

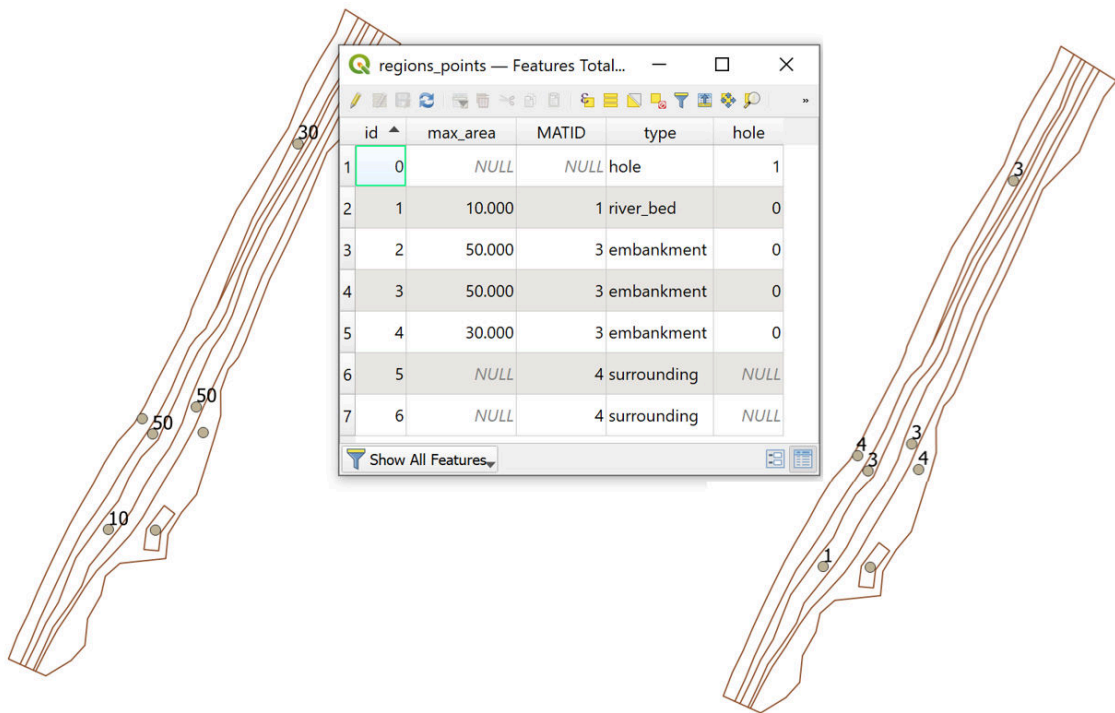



Figure 1.17 Region marker points

1.2.11 Creation of Quality Mesh

We can now use BASEmesh to generate a mesh with high quality properties by controlling cell sizes, using breaklines and holes and other parameters. Please note that the quality mesh created in this section does not incorporate any elevation data. This information will be added in the next step of the tutorial.

- (1) Go to *Plugins* → *BASEmesh* → *Quality meshing*, or click the respective button  in the toolbar.

On the left side of the dialog (Figure 1.18) you can define the input layers. See Section 3.2 “2D grid generation with BASEmesh QGIS plugin” of the BASEMENT User Manual for further explanations. On the right side status messages as well as Triangle’s output messages are displayed during meshing. Tool-specific help can be found in the *Help*-tab.

- (2) Select the different fields according to Figure 1.18
- (3) Check the optional button for *Breaklines*, and *Regions*. Within the Layer ‘region_points’ check the attributes *maximum area*, *material index* and *holes* like shown in Figure 1.18.


In this tutorial a *minimum triangle angle* of 28 degrees was chosen. This means that no elements with angles smaller than 28 degrees are created. Therefore a smaller value leads to a smaller number of elements in the mesh, while a larger value leads to a higher number of elements but less distorted triangles.

- (4) Finally, click on *run* to generate the quality mesh.

If the mesh contains regions with almost infinitely dense triangulation, again check the snapping of your model boundary and the breaklines used for quality meshing or increase the *Relative snapping tolerance* (see Tips and Tricks Section 1.3).

1.2.12 Generation of the Computational Mesh

The quality mesh generated in the previous step still lacks any elevation data. Before it can be exported and used for simulations, elevation data has to be interpolated on the nodes of the quality mesh. For this purpose, the elevation mesh created in a previous step will be used. Alternatively, one could also use raster data as source for the elevation model.

- (1) Go to *Plugins* → *BASEmesh* → *Interpolation*, or click the respective button  in the toolbar. A new dialog will open (Figure 1.19).
- (2) As input for *Mesh layer to interpolate* choose the quality mesh.
- (3) To make the computational mesh using an elevation mesh, check the radio button *Interpolation via elevation mesh*, and select the desired elevation mesh. If you wanted to make the computational mesh using a DEM, you could check the radio button *Interpolation via DEM (raster)* and select the DEM.

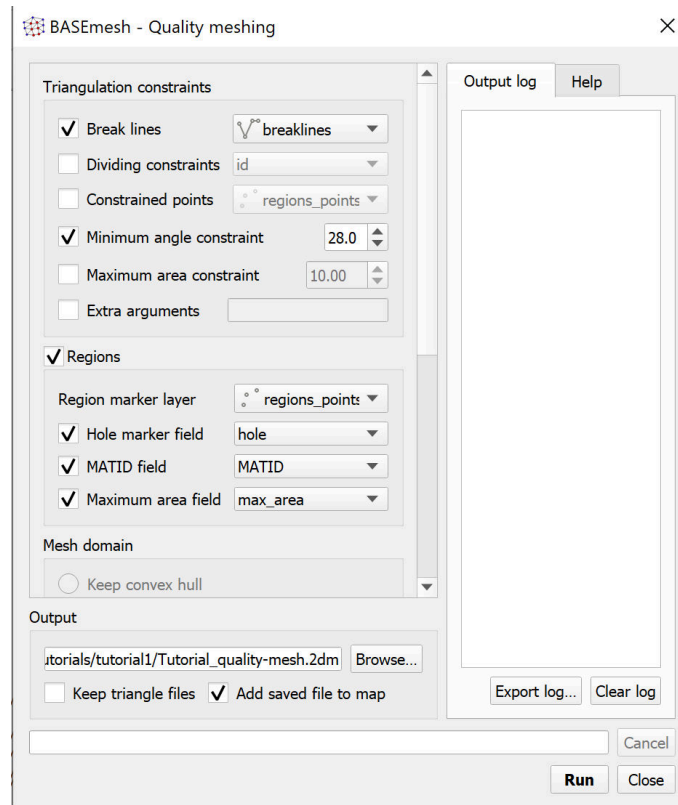


Figure 1.18 BASEmesh Quality Meshing

- (4) Click on *Browse* and input a name for the computational mesh.
- (5) Click on *run*. On Windows systems, the interpolation dialog might be marked as Not Responding. On Linux systems, the interpolation dialog might be darkened. This does not necessarily mean that the process stopped or had errors: The system is simply busy.

In case of a large number of elements ($>10'000$), the interpolation might take up several minutes. In general, the interpolation from raster data is faster than the interpolation from the elevation mesh. For every node of the quality mesh, the elevation has been interpolated based on the elevation mesh given. Therefore, the locations of the newly created interpolated nodes are identical to the nodes of the quality mesh, but contain the interpolated elevation data as attribute. The result can be checked for plausibility by labeling the elevation nodes layer and the interpolation result layer with their elevation attributes and comparing the values.

1.2.13 Viewing the computational mesh

During the interpolation step, elevation data has been added to the mesh. With this additional information we are now able to view both the elevation data and the mesh itself. This can be useful to check for plausibility.

- (1) The elevation data of the computational mesh will be visible. To view the mesh, right click on the computational mesh in the layers toolbar, select *Properties* → *Symbology*

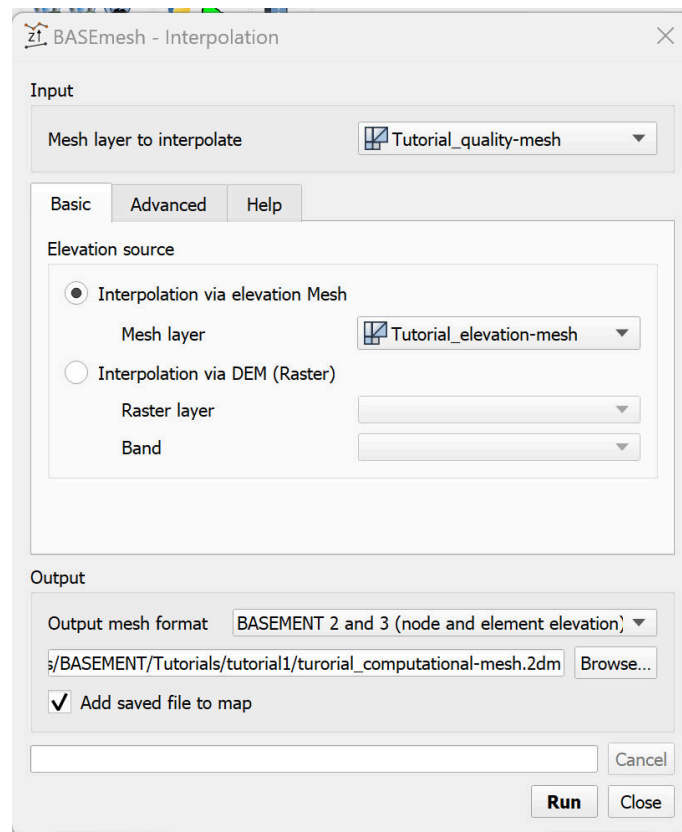


Figure 1.19 BASEmesh Interpolation

→ select *Native Mesh Rendering* → *OK*. The computational mesh for this tutorial should look as it does in Figure 1.20.

- (2) The computational mesh is now ready for use in a BASEMENT simulation.

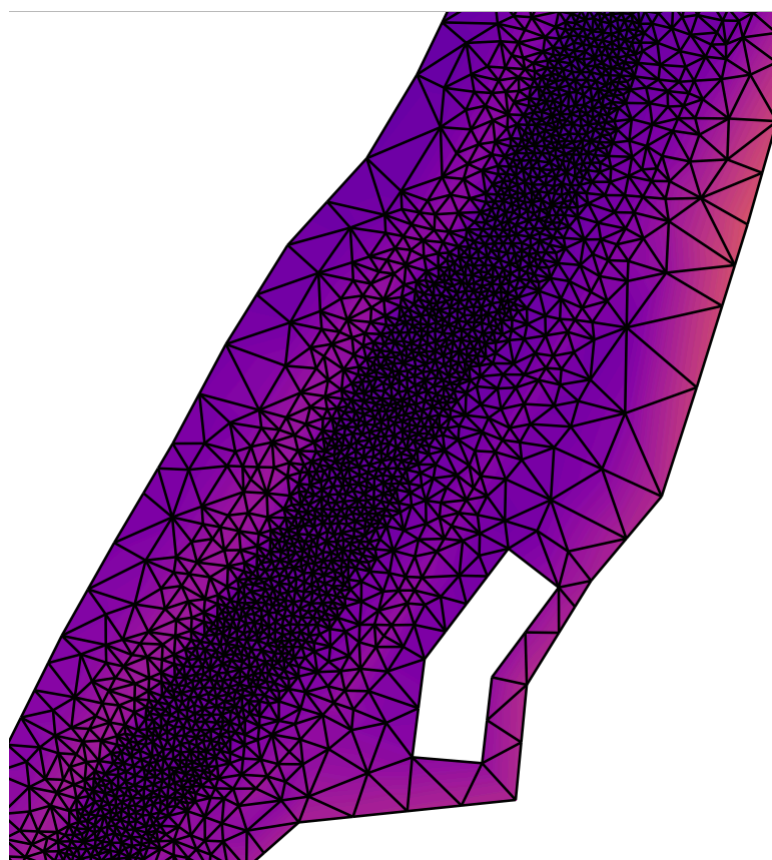


Figure 1.20 Computational mesh

1.3 BASEmesh Tips and Tricks

1.3.1 Recommended Plugins

QGIS offers the possibility to extend its features using plugins from various sources. We recommend in particular the following plugins.

Core plugins:

fTools This is the main plugin for many common vector-based GIS tasks. Core plugin, which can be activated in the plugin manager.

Spatial Query Plugin Core plugin of QGIS, which can directly be activated in the QGIS plugin manager without downloading. *Spatial Query* is comparable to ArcGIS' *Select by Location* feature and enables the user to select features in a target layer with reference to another layer. Possible operators are: contains / equals / overlap / crosses / intersect / touches / within.

Select Within Another spatial query plugin that runs through each geometry you want to select from and tests if the centroid falls within the selected geometry. This plugin is especially useful to select mesh elements for the assignment of material indices.

Processing Spatial data processing framework, which gives access to a large number of analysis algorithms. It can also connect external algorithms from other GIS packages, such as GRASS, SAGA or Orfeo Toolbox. One of the main features is the graphic modeler, where frequent workflows can be graphically represented by the user and executed automatically.

Python plugins:

Python plugins are mostly contributed by the worldwide QGIS user community. Most of them are stored in the official repository and available as stable or experimental versions.

Crayfish plugin Visualizes result data on structured or unstructured meshes using color maps and vectors. Crayfish is able to directly visualize result data from the numerical simulation software BASEMENT (see Post-Processing tutorial Section 2.1).

Point to One Converts a series of points to lines or polygons based on a common attribute or a sequence field.

Point Sampling Tool Samples polygon attributes and raster values from multiple layers at specified sampling points.

Dxf2ShpConverter Conversion and import of dxf files.

Interpolation plugin Very useful for interpolating raster maps of given elevation data, e.g. point clouds.

1.3.2 Tips and Tricks

The following list contains some basic tips and tricks for problem handling and the daily work with BASEmesh.

Creation of polygons out of intersection polylines: In many cases, polygons must be created out of intersecting polylines, e.g. for the definition of model boundary polygons based on breaklines. For completing this task, two features are available in QGIS:

- via fTools: Select *Vector* → *Geometry Tools* → *Lines to polygons*. Quite simple menu, no parameters or options can be chosen (standard option). Prone to errors with complex geometries.
- via the Processing toolbox: Select *Processing* → *Toolbox* → *QGIS geotools* → *Vector geometry tools* → *Polygonize*. Complex geometries are handled well by this tool, multiple options and a log are available.

Creation of a model boundary based on elevation data: When creating an elevation mesh, all vertices of a model boundary polygon must lie on elevation points. Otherwise, there might be interpolation errors in the following meshing steps. Therefore, it is advisable to create a model boundary using the *Convex hull(s)* - feature of QGIS. Below is a short example using the input data provided with Section 1.2.

- Select *Vector* → *Geoprocessing Tools* → *Convex hull(s)*
- As *Input vector layer* select 'XS_points_straightened'.
- Check the radio button *Create single minimum convex hull*.
- Select an appropriate name for the output shapefile.
- Check *Add result to canvas*.
- Select *OK*, after completion close the dialog. Your result should now look like Figure 1.21 .

What to do if Triangle creates (almost) infinitely dense triangulations:




- Polylines that are used for the segmentation of boundary polygons must end in vertices of the polygons. Otherwise, Triangle creates an almost infinitely dense triangulation to ensure its angle and area criteria.
- Be aware that using the *vertex editing tool* generates a new vertex point at each double-click. This can easily lead to the generation of two or more points at the same location or very near to each other. In such a case, Triangle creates a very dense triangulation due to its angle and area criteria.
- Prevent situations where breaklines (and corresponding vertices) have very short distances to each other. The resulting mesh will be very fine in these regions. Meshing sometimes requires to manually adapt the input data and to make compromises between mesh quality and accuracy in certain regions.
- Use the *Relative snapping tolerance* provided in the BASEmesh dialogues *Elevation meshing* , *Quality meshing*  and *String definition* . The mesh shown in Figure 1.22 on the right was generated with the default *Relative Snapping Tolerance* of -6. This led to a very dense triangulation at the intersection of breakline and



Figure 1.21 QGIS Convex hull

model boundary since the breakline is not exactly snapped to the vertex of the model boundary. Furthermore MATID 3 was not considered for the embankment as desired. To prevent such behaviour, increase the tolerance to -3 for example. This allows you to generate a proper mesh like shown in Figure 1.22 on the left.

Do not group layer in the table of contents (TOC): Grouping layers in the TOC may lead to errors because the resulting shapefiles cannot be generated after triangulation.

Conversion of 3D shapefiles to 2.5D shapefiles with elevation attributes This special kind of shapefile is sometimes used in ArcGIS. The 3rd dimension value is not stored in the attribute table (as in 2.5D shapefiles), but in the geometry definition itself. The x and y coordinates of such shapefiles can be displayed in QGIS, but the elevation information is inaccessible and lost. In the following, three workflows are illustrated how to convert 3D shapefiles to 2.5D shapefiles:

- *GDAL - ogr2ogr*: On Linux systems, this GDAL command line tool is directly accessible on the shell. On Windows systems, the tool can be used in the OSGEO4W-shell. For further information, please visit the [GDAL homepage](#)
- *Spatialite*:
 - Open QGIS, add your shapefile and save it as Spatialite (or alternatively use the QGIS DB Manager to drag and drop the shapefile to an already existing Spatialite Database).

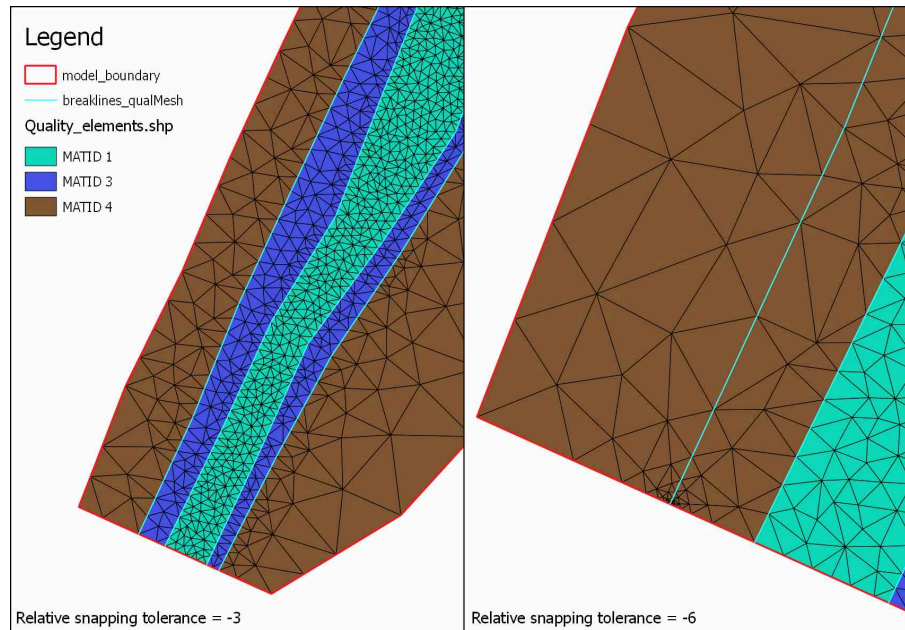


Figure 1.22 Quality meshing using the Parameter ‘Relative snapping tolerance’

- Add the newly created Spatialite vector and through the QGIS vector properties (‘fields’ tab) add a new column, that will store the z-values.
 - In the QGIS DB Manager SQL Window use this command: `update tablename set columnname = st_z(st_pointn(geom,1)).`
 - If at the end of the process you really need to have a shapefile (shame on you) then just use the ‘save as...’ function of QGIS.
- *PostGIS:*
 - Import the shapefile to your PostGIS database
 - `ogr2ogr -f "PostgreSQL" PG:"host=yourhost user=user dbname=dbname password=*****" shapename.shp`
 - Connect to your database
 - `psql -h yourhost -U user dbname`
 - Add a column
 - `ALTER TABLE tablename ADD COLUMN columnname numeric(19,11);`
 - Fill the column with the z-values
 - `update tablename set columnname=st_z(ST_PointN(wkb_geometry,1));`
 - Save result as shapefile

2

Post-Processing of 2D simulation results

2.1 Introduction

In the following tutorials it is demonstrated how to do the post-processing of 2D results generated by a BASEMENT simulation. For this purpose the free and open source GIS software [QGIS](#) is applied.

In addition, the free and open source application [ParaView](#) is used to generate 3D views of the BASEMENT simulation results.

2.2 2D result visualization with QGIS

2.2.1 Input data

The data needed to complete this tutorial comes as ZIP-file and needs to be extracted to a location of your choice. All screenshots and figures in this document were generated with QGIS version 3.16 ‘Hannover’. Along with the computational grid (Flaz_mesh.2dm), the results of the unsteady flow simulation of the tutorial ‘Hydrodynamics and sediment transport at the river Flaz’ (Section 4.1) are used for visualization. In the *SPECIAL_OUTPUT* Block of the BASEMENT model file select *format* ‘sms’ to generate result files that can be visualized in QGIS (see Figure 2.1). QGIS can handle both types *node_centered* and *element_centred* results. Be aware that most of the data (e.g. depth, velocity, wse,...) is calculated on the elements during simulation. When choosing type *node_centered*, these results are interpolated to the nodes by BASEMENT and written to the solution files (*.sol).

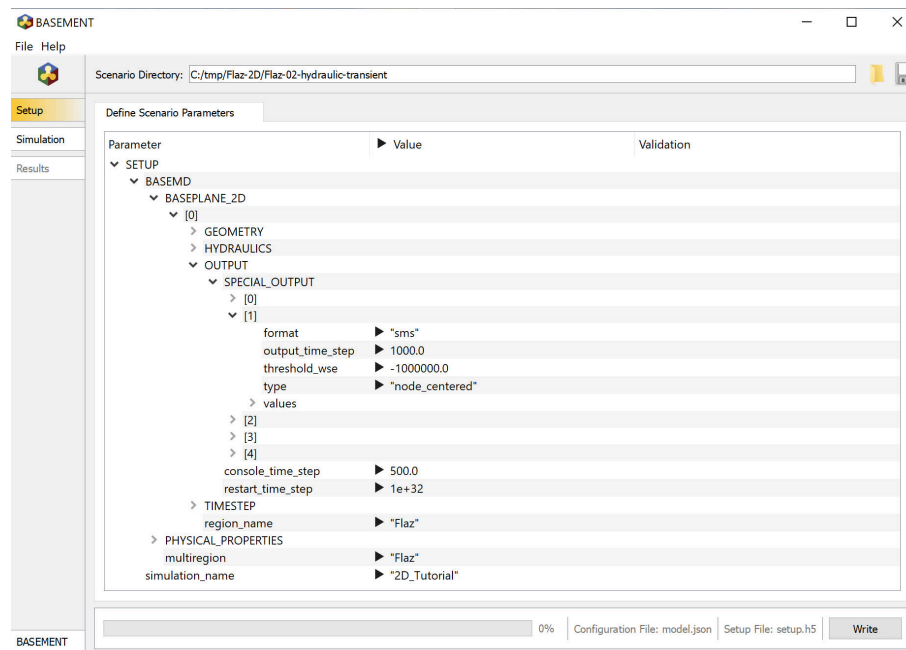


Figure 2.1 Settings defined in the SPECIAL_OUTPUT Block of BASEMENT to generate time dependend result-files (*.sol) of a 2D simulation

2.2.2 About Crayfish

Crayfish is a plugin (extension) developed by Lutra Consulting for the free and open source GIS platform QGIS. The Crayfish plugin aspires to be a complete set of post-processing tools to support numerical modelling within QGIS. With the functionalities of Crayfish, users can load time varying mesh into QGIS. In more recent editions of QGIS version 3, the Crayfish functionalities come incorporated automatically into the QGIS core. Currently, QGIS supports a number of hydraulic modelling software packages including BASEMENT.

The functionalities of Crayfish allow for loading and rendering of results directly, rather than converting them to other GIS formats before viewing. This allows users to flick quickly through the various output steps in the result files and to create animations.


2.2.3 Installation

For recent editions of QGIS version 3, the Crayfish functionalities come incorporated automatically into the QGIS core. Therefore, installation of a separate plugin is not needed.

For older versions of QGIS, Crayfish can be installed from the official QGIS plugin repository. It requires binary libraries specific to your platform. Lutra consulting distributes Windows (32-bit and 64-bit) DLL files and Linux libraries (64-bit) for Debian-based distributions (Ubuntu, Debian, Linux Mint, etc). If your operating system is one of these, the plugin should automatically download the required libraries. To install Crayfish from the QGIS plugin repository, 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) In the tab *All* type ‘Crayfish’ into the search field and click *Install plugin*

Once Crayfish is installed successfully, a new icon  should be added to your Layers toolbar.

2.2.4 Load and visualize data

2.2.4.1 Project Settings


In order to display the provided geodata correctly, some settings regarding the coordinate reference system (CRS) have to be defined. In this tutorial we use the same CRS (CH1903/LV03) like in the tutorial of BASEMesh (Section 1.1). With the following steps, the project’s CRS can be changed:

- (1) Go to *Project* → *Properties*.
- (2) Under the tab *CRS* you can see the coordinate reference system settings for the current project.
- (3) Enter EPSG code ‘21781’ into the field *Filter*.
- (4) Select the coordinate reference system ‘CH1903 / LV03’ and click *OK*.
- (5) Again go to *Project* → *Project Properties* → *CRS*.
- (6) Define a *Project title* under the tab *General* → General Settings*.
- (7) Close the Project Properties window. If everything went well, you should see the chosen project name at the title of the QGIS main window and the EPSG code of your coordinate system at the lower right corner of the QGIS desktop.

2.2.4.2 Loading results of unsteady flow simulation (river Flaz)

The provided data is stored in the ‘QGIS’ subfolders ‘background_data’, ‘Flaz_mesh’ and ‘2D_results’. You can either use these data or visualize your own mesh and/or simulation results of the BASEMENT 2D tutorial ‘Hydrodynamics and sediment transport at the river Flaz’.

Loading geodata

- (1) Go to *Layer* → *Add Layer* → *Add Raster Layer* (Alternatively press Ctrl+Shift+R or use the *Add Raster Layer* icon  in the Manage Layers toolbar).
- (2) Browse to the folder ‘background_data’ and load the areal image ‘Flaz_si25.tif’.

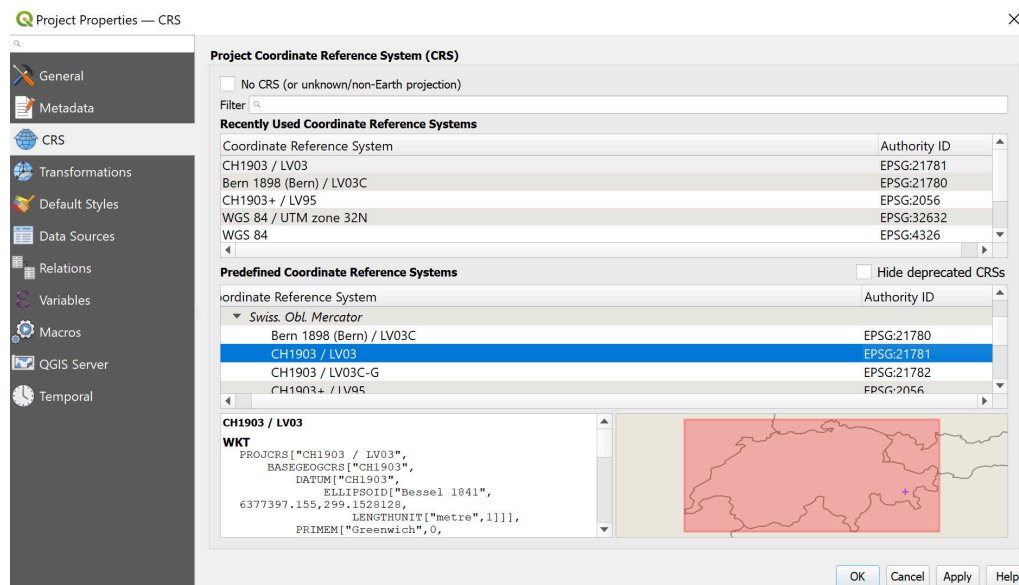




Figure 2.2 Project Properties, CRS

- (3) Go to *Layer* → *Add Layer* → *Add Vector Layer* (Alternatively press Ctrl+Shift+V or use the *Add Vector Layer* icon  in the Manage Layers toolbar).
- (4) Browse to the folder ‘Flaz_mesh_data’ and load the provided shapefiles (*.shp).

The color schemes of newly added layers are randomly chosen by QGIS. You can change them by right-clicking on a layer → *Style* → select a color from the palette. QGIS displays the loaded layers according to their order in the Layers control panel.

Adding Results layers

To view 2D results, first, we load the computational mesh of the 2D simulation. If you use your own data, make sure to load the latest mesh file (*.2dm) you used for your simulations. Otherwise the node ids of the mesh won’t match those of the result files and the visualization won’t be correct.

- (1) Go to *Layer* → *Add Layer* → *Add Mesh Layer* or use the *Add Raster Layer* icon  in the Manage Layers toolbar).
- (2) Browse to the folder ‘03_2D_results’ and load the mesh ‘Flaz_mesh.2dm’.

Now load the result files of the 2D simulation:

- (1) Double-click on the mesh layer. Go to *Source* → *Available Datasets Layer* → *Assign extra dataset to mesh*
- (2) Browse to the folder ‘03_2D_results’ and load the solution files ‘Flaz_nds_depth.sol’ and ‘Flaz_nds_velocity.sol’

The appropriate mesh file should be loaded automatically when the user opens a .dat or .sol file. If the mesh file is named differently, users will be prompted to locate the mesh file

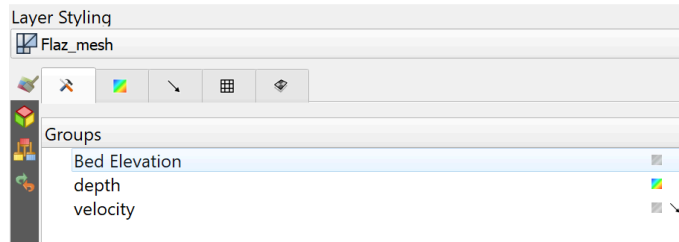


Figure 2.3 Layer Styling Panel Groups

manually. In this case, browse to the location of your mesh and double-click on the file (*.2dm).

If a quantity contains time-varying data, you will be able to browse through the output times defined by the ‘output_time_step’ in the BASEMENT command file (see Figure 2.1) and change the timing through the time control.

2.2.4.3 Display properties

The style of the displayed quantities can be modified in the Layer Styling panel. On the 1st tab of the Layer Styling Panel, you can select which layer associated with your 2dm that you would like to edit. To select a layer, click on the icon to the right of the layer name (see Figure 2.3).

Contour styles

Styling of contours can be done in the 2nd tab of the Layer Styling Panel (see Figure 2.4). In this tab you can:

- Create your own color ramp
- Set the interpolation in the color ramp
- Save and load color settings
- Fill values below or above min/max

Vector Styles

To activate the vectors, *Go to the 1st tab of the Layer Styling Panel → Click on the arrow to the right of ‘velocity’*. Vector styling of the velocity vectors can then be done in the 3rd tab of the Layer Styling Panel (Figure 2.5).

In this tab you can:

- Set the color, length, and width of the vectors
- Filter by min/max
- Set vector head size
- set an arbitrary grid, where vectors will be displayed by activating *Display on User Grid*. This is particularly useful to view a smoother vector interpolation.

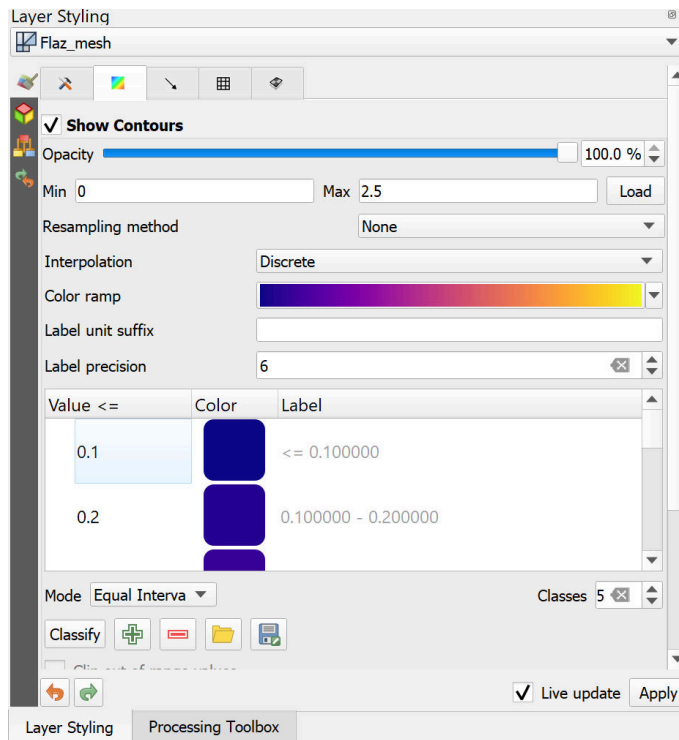


Figure 2.4 Layer Styling Panel

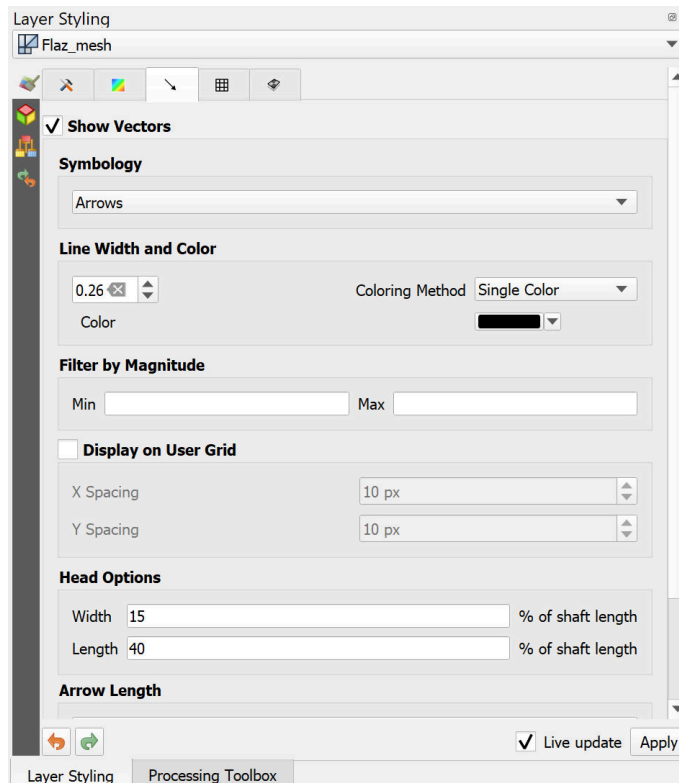


Figure 2.5 Vector Options window of the Layer Styling panel

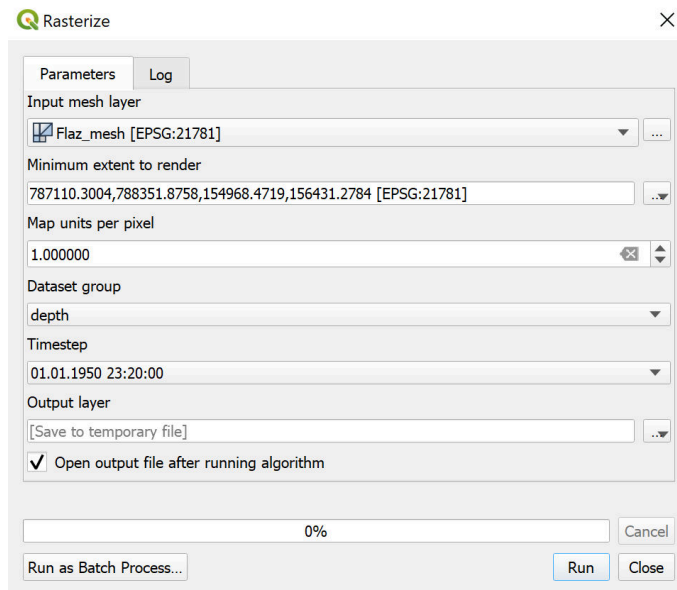


Figure 2.6 Rasterize

2.2.4.4 Export results to raster

For further results analysis, it might be useful to export the simulation results into raster format. Crayfish offers the possibility to directly resample the results data at a desired output time step on a desired output resolution.

- (1) In the *Processing Toolbox*, search for ‘Rasterize.’ Select the Crayfish Rasterize tool. (Figure 2.6)
- (2) Set the “Minimum extent to render” based on *Calculate from layer* → *Flaz_model_boundary* (make sure QGIS window is zoomed to show entire model domain).
- (3) Select the mesh layer that contains the results data.
- (4) Select the desired data, timestep, and grid resolution.
- (5) Click OK .

- (1) *Right click on the newly created layer* → *Click Export* → *Save as...*
- (2) Save the raster layer as a GeoTIFF (Figure 2.7)

The generated raster is stored in Geo-Tiff format (*.tif) and will cover the extent defined by the model boundary. The flexible raster format allows further post processing steps of the simulation results. After having exported both quantities depth and velocity, you can easily calculate the product depth * velocity using the QGIS *Raster Calculator* for example. This might be of particular interest when generating flood hazard maps for example.

- (1) Go to *Raster* → *Raster Calculator*.

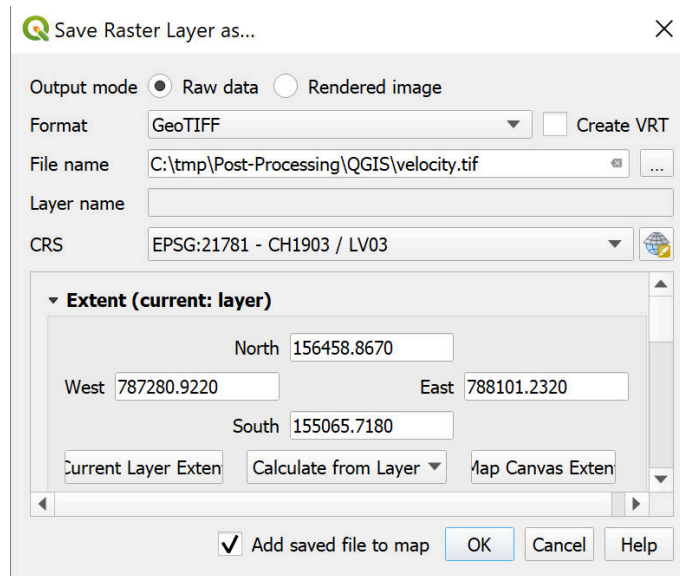


Figure 2.7 Save as GeoTIFF

- (2) Specify the *Raster calculator expression* as shown in Figure 2.8 by double-clicking on the *Raster bands* depth and velocity in combination with the required *Operators*.
- (3) Define the *Output layer* and press *OK*.

The new calculated raster dataset will cover the same extent like the two input datasets and will be stored in Geo-TIFF format as well.

2.2.4.5 Profile tool

If you want to analyse the temporal change of a variable along a certain cross section of a river reach during a BASEMENT simulation, in principle there are two options:

- Define a *STRINGDEF* along the desired location within your computational grid and chose a *SPECIAL_OUTPUT* of type *stringdef_history* with the required *stringdef_values*.
- Use the QGIS plugin ‘Profile Tool’ and analyse the results of a Crayfish layer.

The advantage of option two is a fast and easy evaluation at any location of the modelled domain, without having defined any *stringdef_history* outputs beforehand.

Installation

The Profile Tool plugin can be installed via the official QGIS repository:

- (1) Load the QGIS plugin manager by choosing *Manage and Install Plugins...* in the menu *Plugins* in the QGIS main toolbar.
- (2) In the tab *All* type ‘Profile tool’ into the search field and click *Install plugin*.

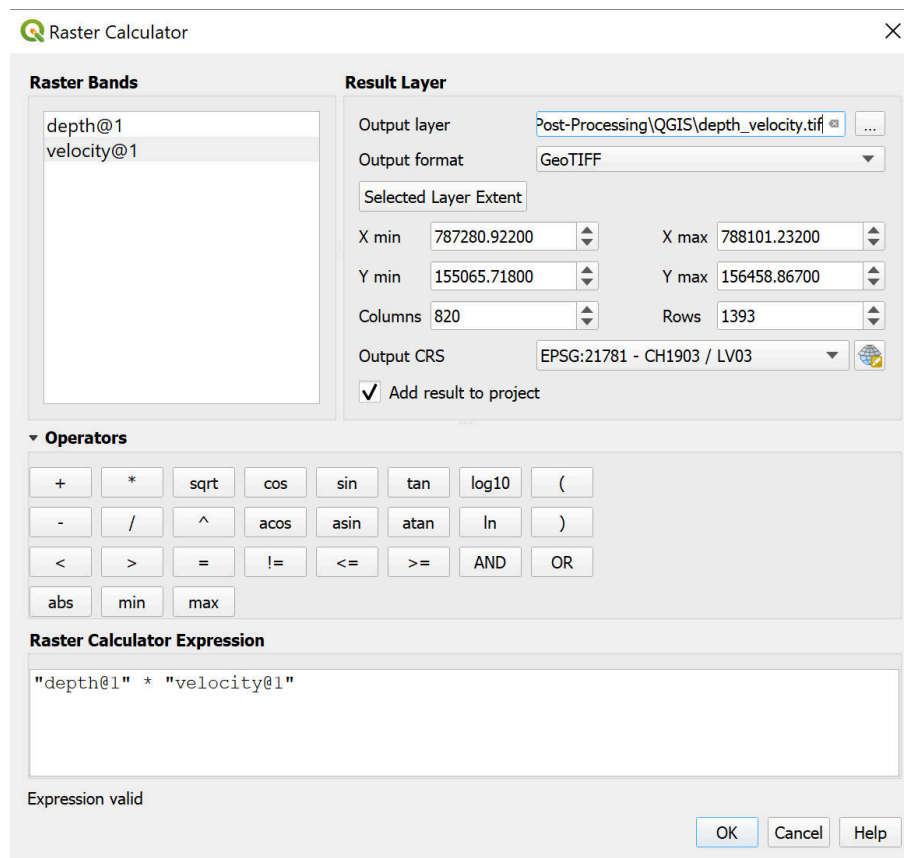


Figure 2.8 Applying QGIS Raster Calculator

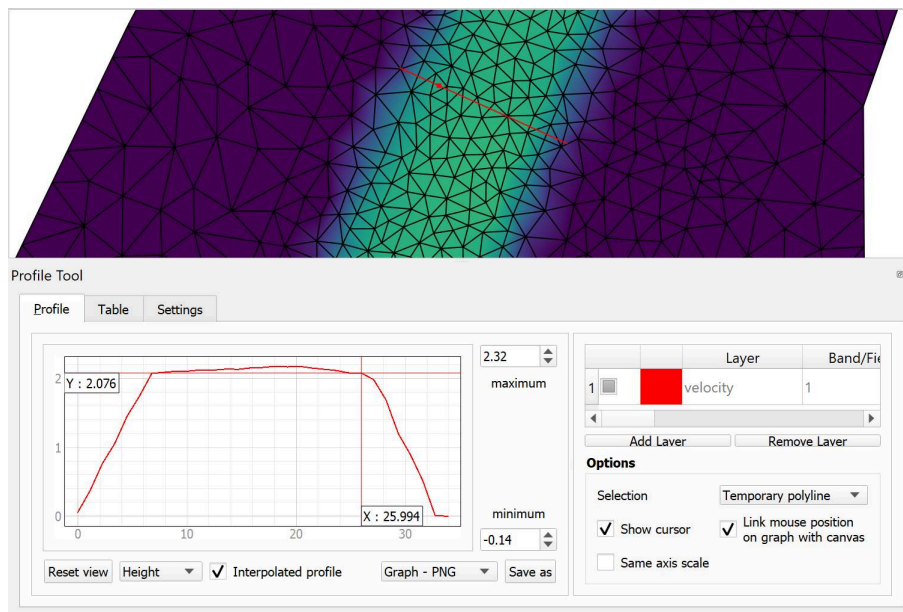



Figure 2.9 Velocity profile along a temporary polyline generated with the ‘Profile Tool’ plugin


Once the ‘Profile Tool’ is installed successfully, a new icon  should be added to your Layers toolbar.

Usage

In general, there are two options to use the ‘Profile Tool’:

1. Using a temporary polyline:

With this option, you can draw an arbitrary polyline at the location of your choice. Along this line the data of the selected *Quantity* will be extracted.

- (1) Click on the *Terrain profile* icon  and the Profile Tool window pops up.
- (2) In the field *Selection* chose *Temporary polyline* from the dropdown menu.
- (3) Use the black cross to draw a red line along the desired location of your displayed *Quantity* (see Figure 2.9).
- (4) Start drawing the line with a click and finish the line with a double-click

Now the profile of the chosen quantity should be displayed in the *Profile Tool* window (see Figure 2.9 below). From there you can save the graph in different formats or export the raw data via the tab *Table*.

2. Using a selected polyline:

With this option you can chose an existing polyline feature e.g. a certain cross section to extract the desired quantities.

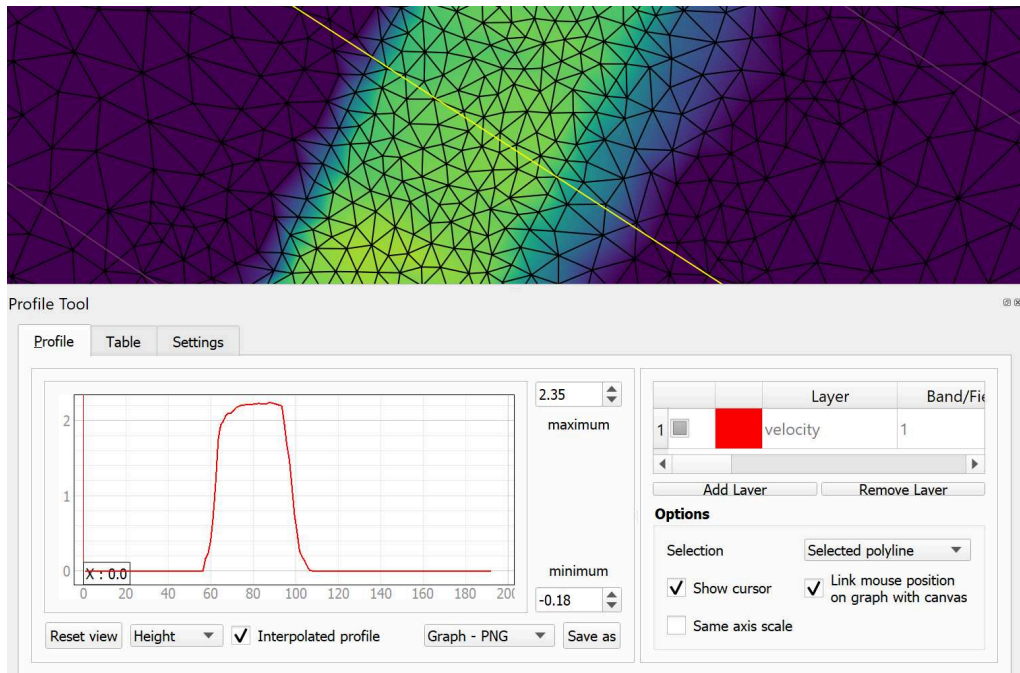




Figure 2.10 Velocity profile along the red line drawn at the location of cross section ‘CS10’ generated with the ‘Profile Tool’ plugin

- (1) First make sure to *Deselect Features from All Layers* pressing the respective toolbar button 
- (2) Again click on the *Terrain profile* icon  and the Profile Tool window pops up.
- (3) In the field *Selection* now chose *Selected polyline* from the dropdown menu.
- (4) Right-click on the layer ‘Flaz_cross_sections’ and go to *Open Attribute Table*.
- (5) Select the feature with *name* ‘CS10’ and left-click into the map displayed in the QGIS main canvas.

Again the profile of the chosen quantity should be displayed in the *Profile Tool* window (see Figure 2.10). Note that the profile also contains entries with value 0 since the extent of the cross section is wider than the wetted main channel.

2.2.5 Creating maps and animations

An important step of the post processing of simulation results is the creation of maps for example of flow depths or velocities to be published in reports or presentations. For this purpose QGIS offers a powerful tool called ‘Print Layout’ which allows you to take your GIS layers and package them to create professional maps including legend, scale bar or text boxes. It is possible to display and export BASEMENT results layers using the QGIS ‘Print Layout’. This tutorial can’t cover the whole functionality of the ‘Print Layout’, but some of the main features, needed to create proper figures and animations of 2D simulation results are explained subsequently. Check the online [User Guide](#) of the software QGIS for further documentation.

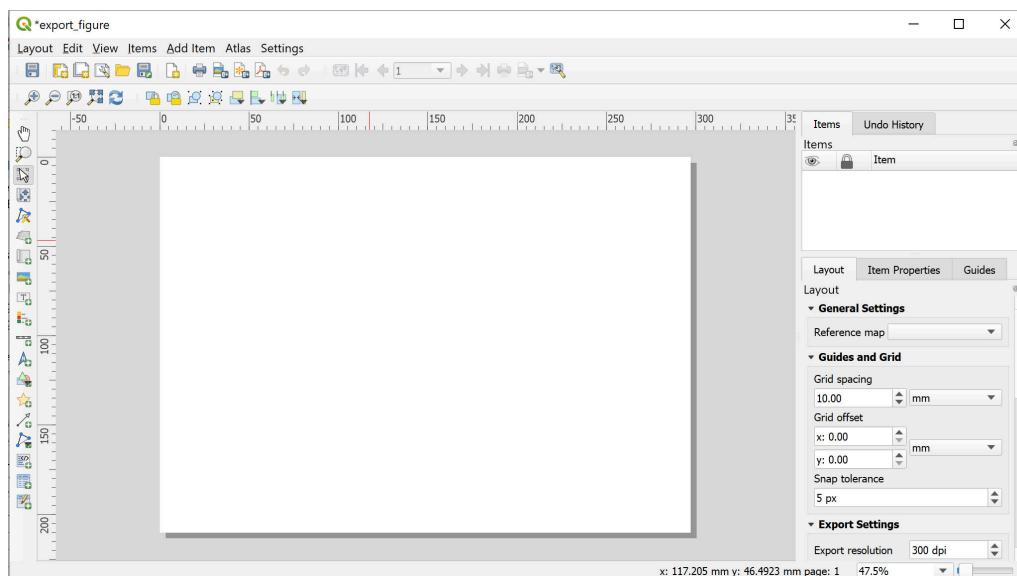



Figure 2.11 QGIS Print Layout with some of the basic control features


2.2.5.1 Customized layout (QGIS print Layout)

After loading all the layers to be exported in a map into the main QGIS canvas, we can start to assemble our map. Note, that only layers which are checked / enabled in the *Layers control panel* of the main canvas can be displayed in the *Print Layout*.


- (1) Go to *Project* → *New Print Layout* or press the New Layout icon  in the toolbar.
- (2) You will be prompted to enter a title for the layout e.g. 'export_figure'. If left blank, a default name will be applied.

A new Print Layout opens up (see Figure 2.11). The main control buttons to add and modify maps can be found in the toolbar on the left site. You can also specify the layout and export resolution in the tab *Layout* on the right site.

After defining a proper layout a new map can be added:

- (1) Press the *Add map button* , hold the left mouse button and drag a rectangle where you want to insert the map.

The rectangle window will be rendered with the currently active map from the main QGIS canvas. In this example the 'depth' results shall be displayed together with the background areal image, breaklines and mesh elements (see Figure 2.12). The rendered map might not cover the desired extent of interest. To alter the displayed map extent and/or zoom:

- (1) Select *Move item content*  to pan the map in the window to desired position.
- (2) Once *Move item content* is active, you can change the map zoom using the mouse wheel.

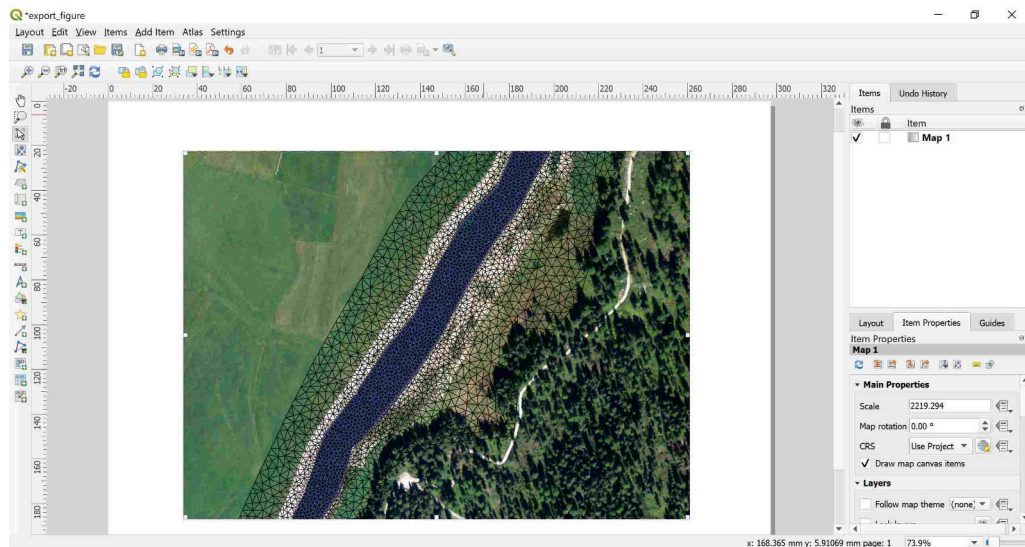








Figure 2.12 QGIS Print Layout with displayed results layer ‘depth’. Orthophoto: © 2016 swisstopo (JD100041)

The map is now listed as ‘Map 1’ in the *Item* dialogue box. Further map properties can be defined via the tab *Item properties*. Additionally, we want to add a legend to our map:

- (1) Press *Add legend* , hold the left mouse button and drag a rectangle where you want to insert the legend.
- (2) The legend will appear as new *Item* ‘Legend’ in the *Items* dialogue box (see Figure 2.13). Go to *Item properties* in order to modify the legend entries.
- (3) Under *Legend items* uncheck/disable the *auto update* function. Now you can remove the legend items that shall not be displayed using the *minus* button . In this example everything but the ‘Flaz_mesh’ and the ‘model_boundary’ is removed. The order of the legend items can be changed with the *up*  and *down*  buttons.

So far no units of the ‘depth’ are displayed in the legend. Furthermore, we want to add scale bar and a north arrow to our map:

- (1) Click on ‘depth’ in the *Legend items* dialogue box and press *edit* .
- (2) Type ‘depth [m]’ and press *OK*
- (3) With the additional *Item properties* you can modify Fonts size and style of the legend entries as well as the size of the symbols or spacing of the columns and rows.
- (4) To insert a scale bar press the *Add new scalebar* button  and click to the map at the desired position.
- (5) A new *Item* ‘scale bar’ appears in the *Items* dialogue box and again properties of the scale bar can be modified via the *Item properties* tab.

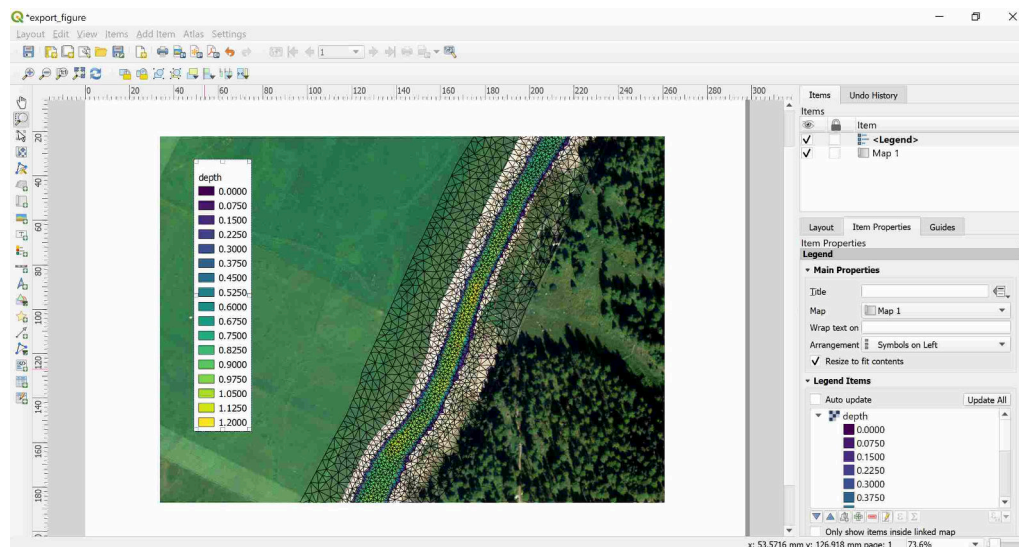






Figure 2.13 QGIS print layout with displayed results layer ‘depth’ after inserting the legend. Orthophoto: © 2016 swisstopo (JD100041)

- (6) To insert a north arrow, press the *Add image* button  and click to the desired location.
- (7) A new *Item* ‘picture’ is listed in the Items dialogue box.
- (8) A new *Item* ‘Picture’ appears in the Items dialogue box. In the *Item properties* tab you will find a preview of different arrow styles offered by QGIS, it is also possible to *Add* and *Remove Image search paths* manually.

If everything went well, your Print Layout should now look like shown in Figure 2.14. The map is now ready to be exported or printed. You can add additional features like text or images to the map at your convenience. Just use the *Add image*  or *Add label* . You can save your map properties as template for further QGIS projects.

(image.png) Go to *Layout* → *Save as Template* or use the button  in the main toolbar.

In general the Print Layout is stored together with the main QGIS Project. If you open a project in which you defined a Print Layout in, you will always be able to access it via the main QGIS canvas.

- (1) Go to *Project* → *Layout manager* where you can find a table of all print layouts defined within the main QGIS Project.

For the export of your map several formats are available in the Print Layout.

- (1) Go to *Laout* → *Export as Image* to export the map as raster image e.g. png, jpeg, tif, . . .
- (2) Alternatively chose export as PDF or export as SVG to save the map in vector format.

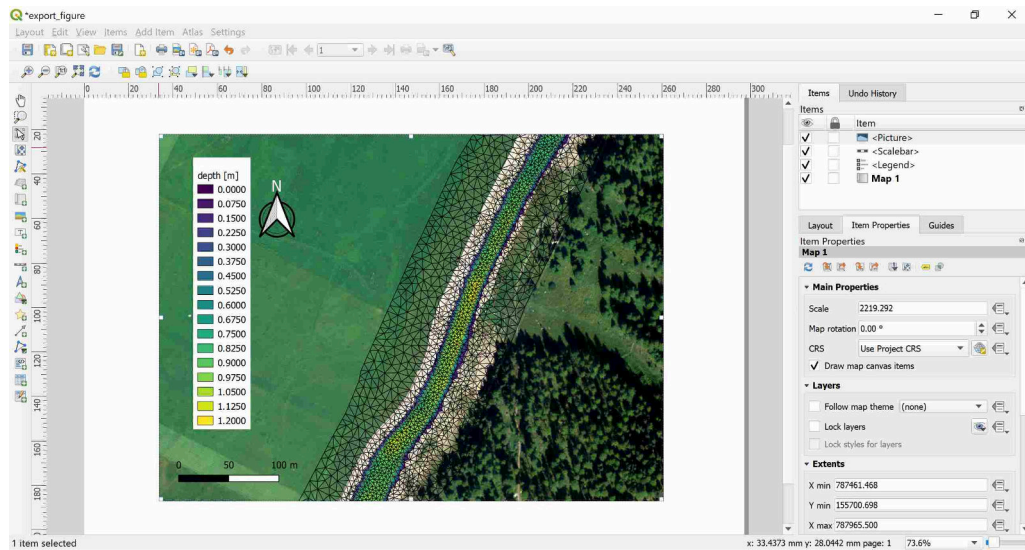


Figure 2.14 QGIS print Layout with displayed results layer ‘depth’ after inserting legend, scalebar and north arrow. Orthophoto: © 2016 swisstopo (JD100041)

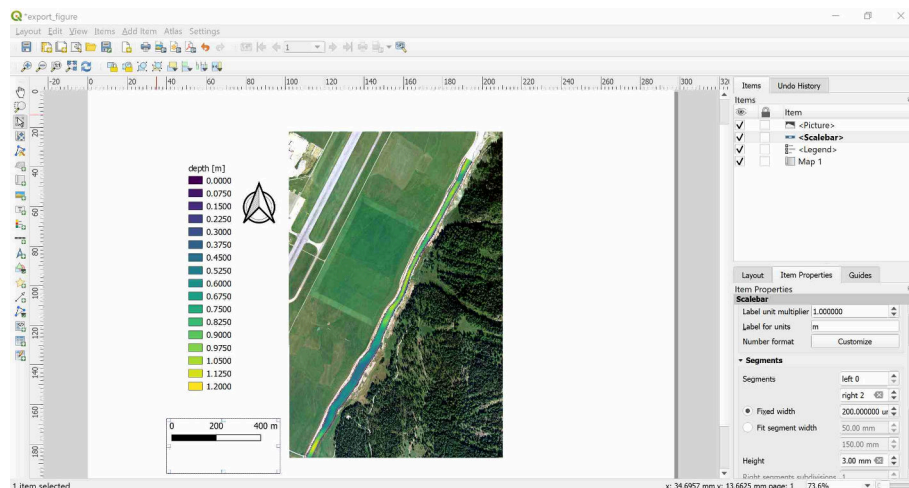




Figure 2.15 QGIS print layout with displayed results layer ‘depth’ zoom to the full extent of the model for export as animation. Orthophoto: © 2016 swisstopo (JD100041)

2.2.5.2 Export animation


If your simulation results are time dependent like those provided in this tutorial, you might not just want to export selected time steps as maps, but rather create an animation over the whole run time of the simulation. Before exporting your time-dependent result, you have to define the layout of your animation in the QGIS Print Layout.

We can continue from the Print Layout of the previous section (Section 2.2.5.1). Using the *Move item content* , zoom the map out to see the entire model domain. You may want to change the size of the scale bar, which can be done in the *Item properties* tab. You may also set the paper size and orientation: *Right click on the map* → select “Page Properties”.

In order to display the runtime of the simulation some additional steps are required:

- (1) Press *Add label*  and add a text field below the Legend.
- (2) Go to *Item properties* and type ‘run time:’ into the *Main properties* field.
- (3) Define *Font size*, *Frame*, and *Background* as desired.
- (4) Again press *Add new label* and insert a second text field to the right of the first.
- (5) Don’t change the default expression ‘Lorem ipsum’ in the Main properties field.
- (6) Go down to *Item ID* and type ‘time’ into the *ID* field

Now the text field is linked to the simulation runtime, which will be displayed during the animation.

- (1) Go to *Layout* → *Save as Template* or use the button  in the main toolbar.

The customized layout of the animation is saved as QGIS layout template (*.qpt) and will be used during the animation export.

The final export of the animation is done from the QGIS main canvas:

- (1) Right click the ‘Flaz_mesh’ layer, select *Export animation...* and a new window will appear (see Figure 2.16).
- (2) In the tab *General* you have to specify the start and end times of the animation, the frame rate in frames per second (fps), and the output directory.
- (3) In the tab *Layout* chose *Custom layout (.qpt)* and browse to the location where you saved the Print Layout template (step 56).
- (4) In the tab *Video* you can chose the Quality of the animation. The choice will affect the time to generate the animation as well as the resulting file size.
- (5) Under *Video encoding utility (FFmpeg) to use:* chose custom.
- (6) Press *OK* to begin generation of the animation.

A window may pop up saying that “The tool for video creation (FFmpeg) is missing ...” (Figure 2.17). Follow the instructions depending on your system (Windows or Linux).

- (1) Finally press *OK*. QGIS will inform you that the export of the simulation was successful.

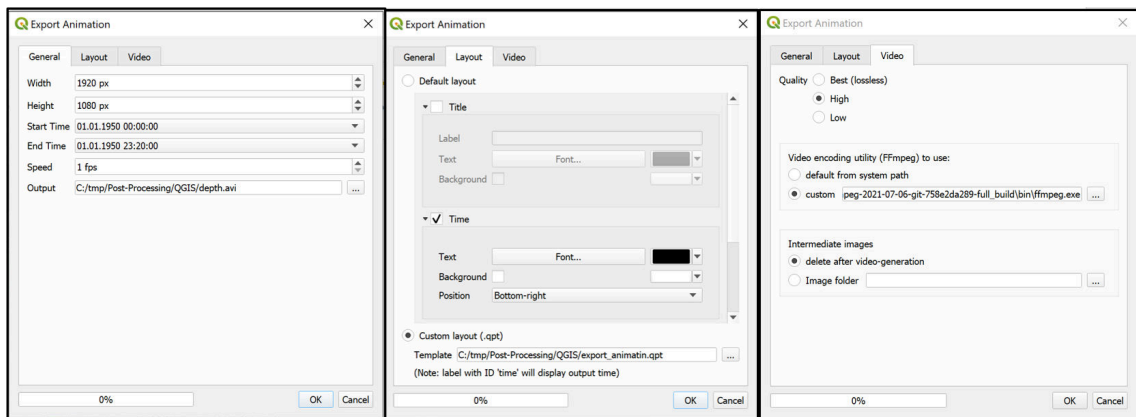


Figure 2.16 QGIS Export Animation window

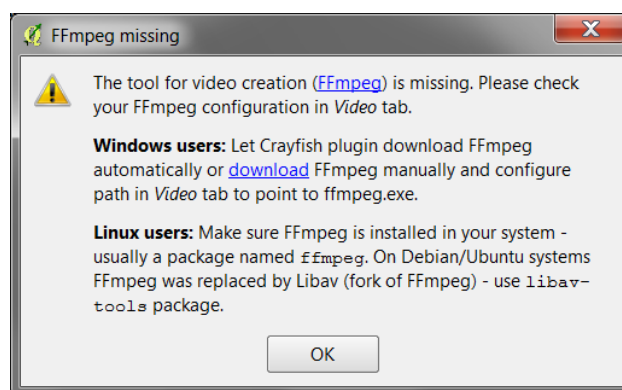


Figure 2.17 QGIS Export Animation FFmpeg

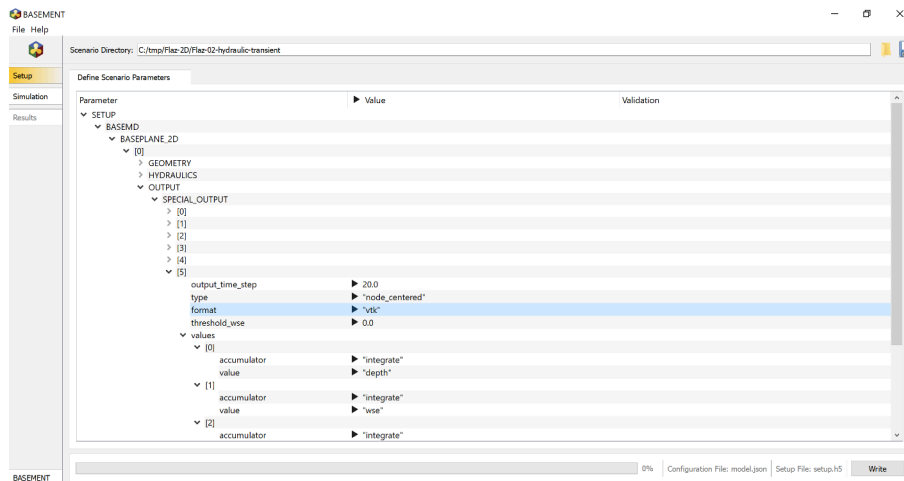


Figure 2.18 Settings defined in the `SPECIAL_OUTPUT` Block of BASEMENT to generate time dependend result-files (*.vtk) of a 2D simulation

2.3 3D result visualization with Paraview

2.3.1 Input data

The data needed to complete this tutorial comes as ZIP-file and needs to be extracted to a location of your choice. All screenshots and figures in this document were generated with ParaView version 4.3.

The result files (*.vtk) used for visualization in this tutorial come from the River Flaz. In the `SPECIAL_OUTPUT` Block of the BASEMENT command file select *format* 'vtk' to generate result files that can be visualized with ParaView (see Figure 2.18).

2.3.2 About Paraview

ParaView is an open source application for visualizing 2- and 3-dimensional data sets. The size of the data sets ParaView can handle varies widely depending on the architecture on which the application is running. The platforms supported by ParaView range from single-processor workstations to multiple-processor distributed-memory supercomputers or workstation clusters. Using a parallel machine, ParaView can process very large data sets in parallel and later collect the results.

More Information

In this tutorial, only a very small part of the enormous functionality of ParaView will be covered. The documentation as well as a number of tutorials is available online as [Public Wiki](#). ParaView also offers an online help that can be accessed by clicking the help button



in the application.

2.3.3 Installation

To use ParaView on your own computer simply go to the [Download page](#) of ParaView. The software is available for Windows, Linux and Mac.

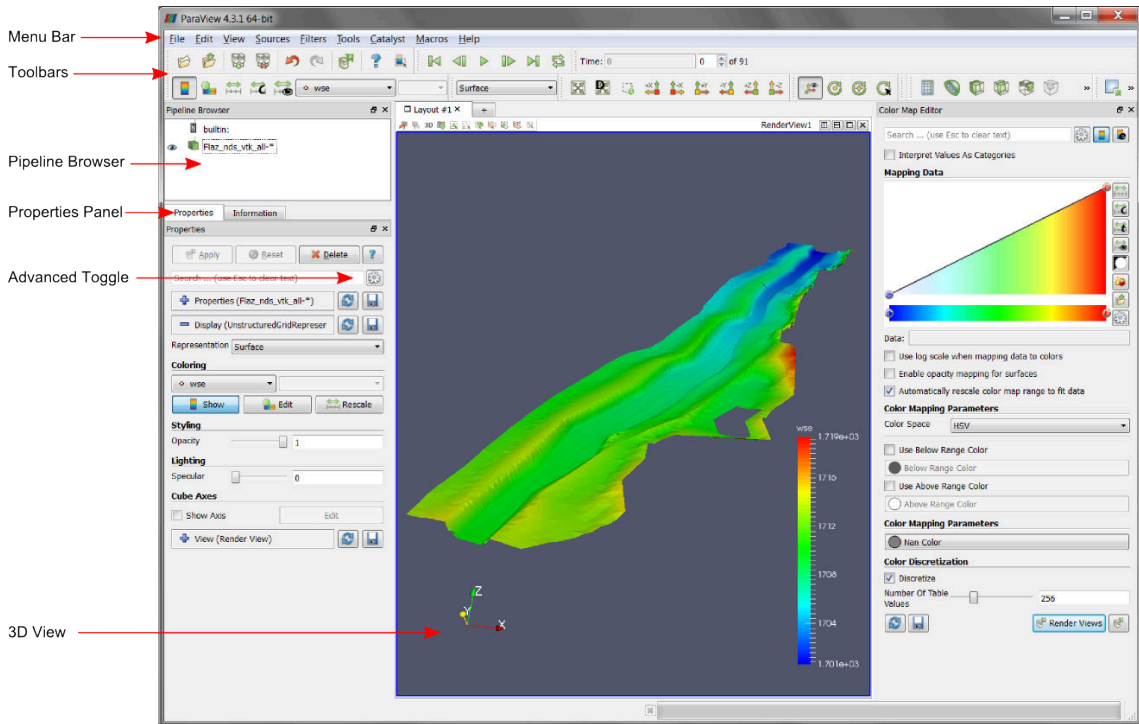


Figure 2.19 ParaView GUI

2.3.4 User Interface

The layout shown in Figure 2.19 is the default layout given when ParaView is first started. The GUI comprises the following components:

Menu Bar As with any other program, the menu bar allows you to access the majority of the features.

Toolbars The toolbars provide quick access to the most commonly used features within ParaView.

Pipeline Browser ParaView manages the reading and filtering of data with a pipeline. The pipeline browser allows you to view the pipeline structure and select pipeline objects for visualization. To toggle visibility of single objects, click on the ‘eye-symbol’.

Properties Panel The properties panel allows you to view and change the parameters of the current pipeline object. Use the *Toggle advanced properties* button to show and hide advanced options. The properties are by default coupled with an **Information** tab that shows a basic summary of the data produced by the pipeline object.

3D View The remainder of the GUI is used to present the data so that you may view, interact with and explore your data. This area is initially populated with a 3D view that will provide a geometrical representation of the data.

Note that the GUI layout is highly configurable. To toggle the use of a toolbar, go to *View* → *Toolbars* and check/uncheck the available options.

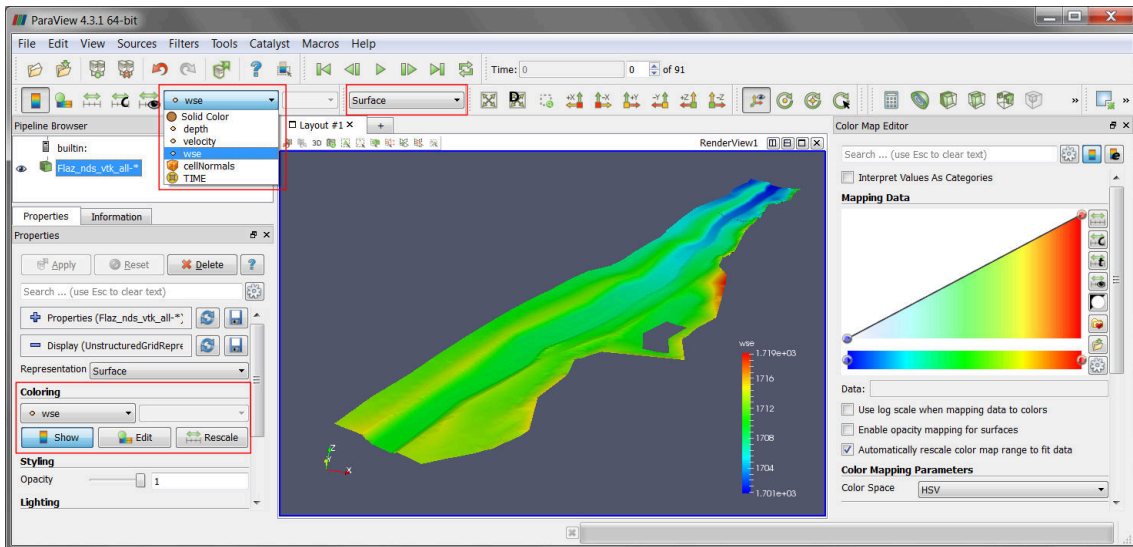



Figure 2.20 ParaView GUI with 3D view of the domain Flaz

2.3.5 Import Data

The provided results are stored in the subfolder ‘ParaView’. Although the folder contains 91 vtk-file representing 91 output timesteps, we can load the results as one single data package:

- (1) Go to *File* → *Open* or use the *Open* button  in the toolbar.
- (2) Browse to the ‘ParaView’ folder, select the file ‘Flaz_nds_vtk_all-..vtk’ and press *OK*.

Note that opening a file is a two step process in ParaView, so you don’t see any data yet.

- (3) In the *Properties Panel*, click on *Apply* to load the data.


Now a 3D preview of the domain Flaz should be rendered to the 3D view. From the dropdown in the toolbar, we can select, which quantity shall be visualized. Alternatively, the quantities can also be selected under *Coloring* in the *Properties* panel. In this example the quantities ‘depth’, ‘velocity’ and ‘wse’ are available (see Figure 2.20). With the option ‘Solid Color’, just the geometry of the domain without any time depending quantity can be displayed. To visualize the mesh elements, select ‘Surface with edges’ instead of ‘Surface’ from the dropdown menu. The color of the mesh elements can be modified under *Edge Styling*, which is enabled by toggeling the advanced properties (see Figure 2.20).


2.3.6 ParaView Filters

2.3.6.1 Overview


In ParaView a large number of so called *Filters* can be applied. *Filters* are functional units that process the data in order to generate, extract or derive features from the data. Here


are some of the most common, which are available by clicking the respective icon in the filters toolbar.


Calculator  Evaluates a user-defined expression on a per-point or cell basis.


Contour  Extracts the points, curves, or surfaces where a scalar field is equal to a user-defined value. This surface is often also called an isosurface.

Clip  Intersects the geometry with a half space. The effect is to remove all the geometry on one side of a user-defined plane.

Slice  Intersects the geometry with a plane. The effect is similar to clipping except that all that remains is the geometry where the plane is located.

Threshold  Extracts cells that lie within a specified range of a scalar field.

Extract Subset  Extracts a subset of a grid by defining either a volume of interest or a sampling rate.

Glyph  Places a glyph, a simple shape, on each point of the mesh. The glyphs may be oriented by a vector and scaled by a vector or scalar. With this *Filter* it is possible to visualize velocity vector for example.

For further information about available *Filters*, please refer to the [documentation](#) of ParaView.

2.3.6.2 Applying Filters to data

After introducing the basic concept of *Filters*, we are now ready to apply three different *Filters* to our dataset. The goal is, to generate a 3D view of the time depending water surface elevation (wse) superimposed on the mesh geometry. First we want to display the geometry with vertical exaggeration for better contrast.

- (1) In the toolbar chose ‘Solid Color’ and ‘Surface’ from the dropdown menu (see Figure 2.21)
- (2) Go to *Transforming* in the *Properties* panel (advanced properties) and alter the *Scale* in the third column (z-direction) from 1 to 2 and press Enter.
- (3) Reload the 3D view by clicking on the eye symbol in the *Pipeline Browser*.

Filter: Warp by Scalar

This *Filter* moves point coordinates along a vector by a distance determined by a user defined scalar for the active data set.

- (4) Select the data set ‘Flaz_nds_vtk_all-*’ from the *Pipeline Browser*.
- (5) In the menu bar go to *Filters* → *Alphabetical* and select *Warp By Scalar*.
- (6) In the *Properties* panel chose ‘depth’ as *Scalars* and press *Apply* (see Figure 2.22)

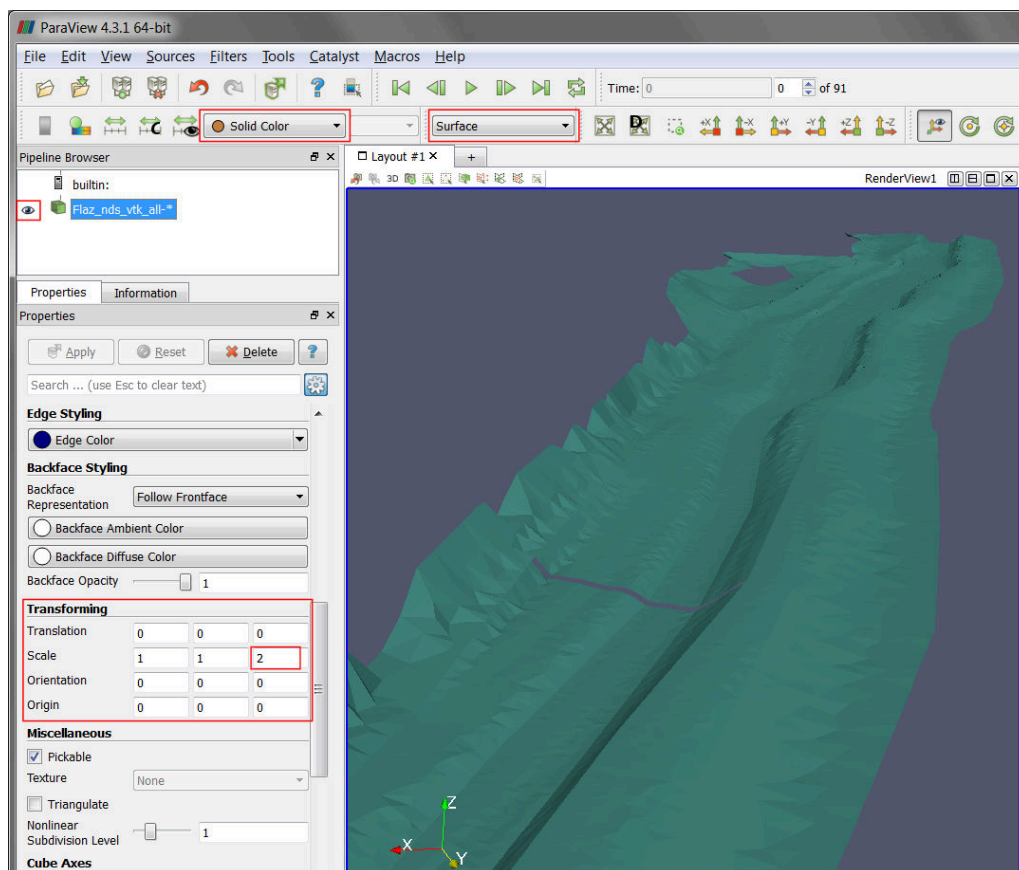


Figure 2.21 Vertical exaggeration of solid mesh

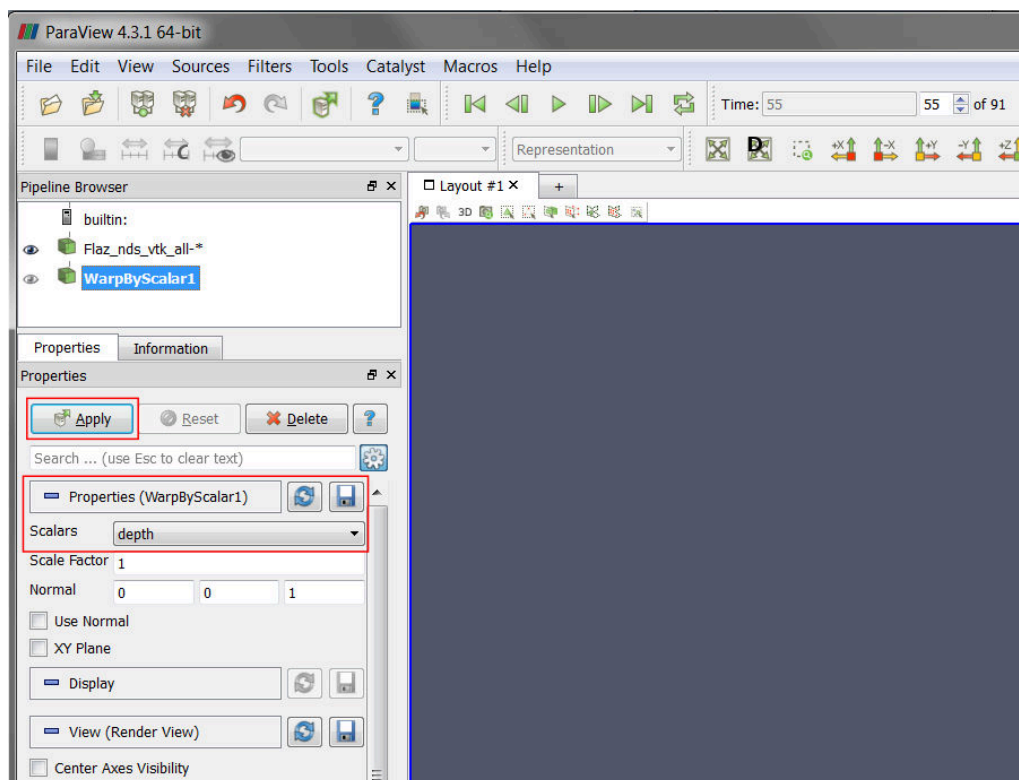


Figure 2.22 Filter, Warp by Scalar

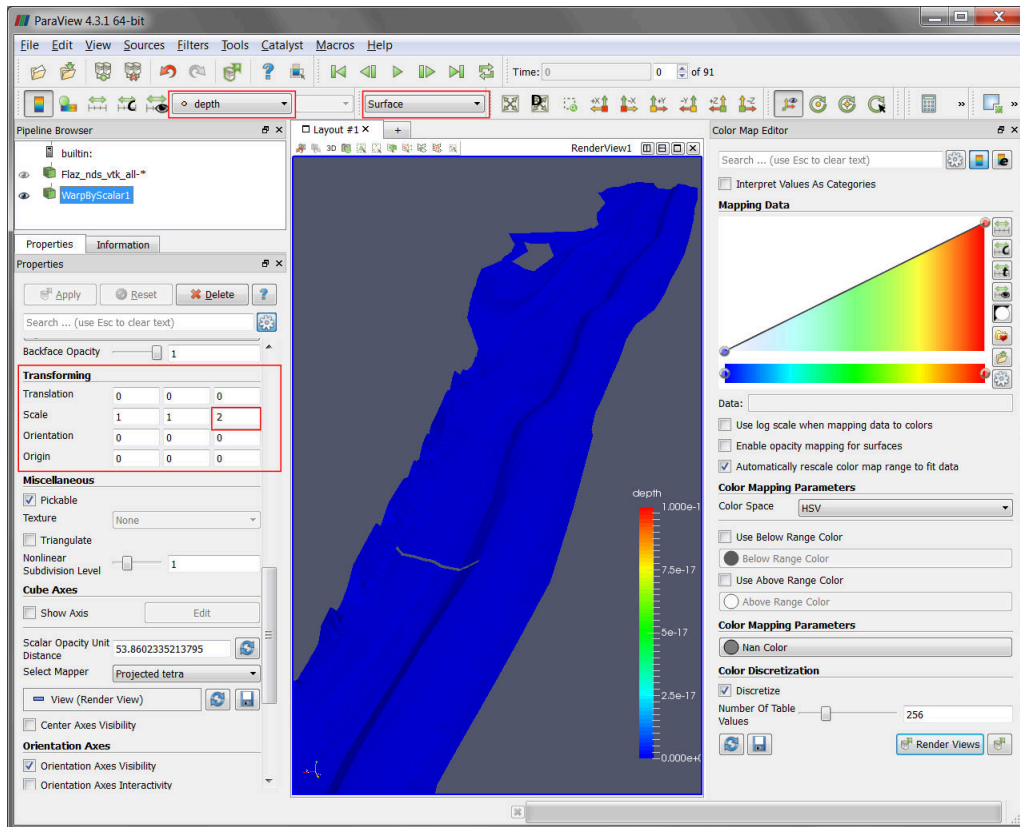



Figure 2.23 Warp by Scalar, properties

- (7) Select the data set 'WarpByScalar1' from the *Pipeline Browser*.
- (8) In the toolbar chose 'depth' and 'Surface' from the dropdown menu (see Figure 2.23)
- (9) Again alter the *Scale* in z-direction from 1 to 2.

Filter: Threshold

Applying a *Threshold Filter*, we are able to blank out elements with a water depth below a certain threshold:

- (10) Select the data set 'WarpByScalar1' from the *Pipeline Browser*.
- (11) In the menu bar go to *Filters* → *Alphabetical* and select *Threshold* or directly access the *Filter* pressing the respective button  in the toolbar.
- (12) In the *Properties* panel for 'Threshold1' chose 'depth' as *Scalars* and specify the *Minimum* and *Maximum* as shown in Figure 2.24, than press *Apply*.
- (13) Repeat the steps 1-3 for the layer 'Threshold1' in order to adjust the coloring and the vertical exaggeration.
- (14) Toggle visibility of the layers 'Flaz_nds_vtk_all-*' and 'Threshold1' in the *Pipeline Browser*.

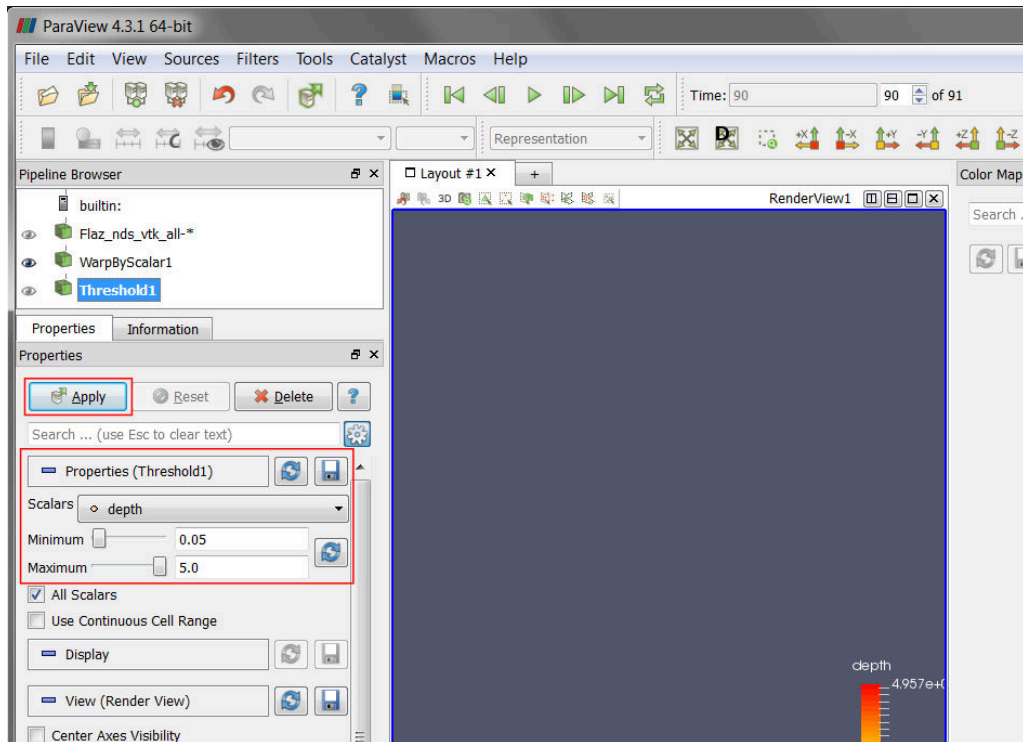


Figure 2.24 Filter, Threshold

Annotate Time Filter

So far our 3D preview of the simulation results is missing a time annotation, which shows the current simulation runtime of the selected output timestep. To add this information:

- (15) Select the data set ‘Threshold1’ from the *Pipeline Browser*.
- (16) In the menu bar go to *Filters* → *Alphabetical* and select *Annotate Time Filter*
- (17) Specify the *Properties* like shown in the *Properties* panel of Figure 2.25 and press *Apply*.

Note that the definition of *Format*, *Shift* and *Scale* is highly dependent on the data you want to visualize. For the provided data a runtime of 1800 seconds and an output timestep of 20 seconds have been defined in the *SPECIAL_OUTPUT* Block of the BASEMENT command file. This leads to 90 time steps you can browse through with the time control buttons in the toolbar (see Figure 2.25).

- Format %4.0f: Double with a width of 4 characters and a precision of 0 decimal places.
- Shift 0: Time annotation starts at 0.
- Scale 20: The timespan between the single output steps in 20 seconds.

If everything went well, your 3D view should no like like Figure 2.26. At the beginning (Time: 0) only the geometry is visible due to the initial condition ‘dry’. From time step 1

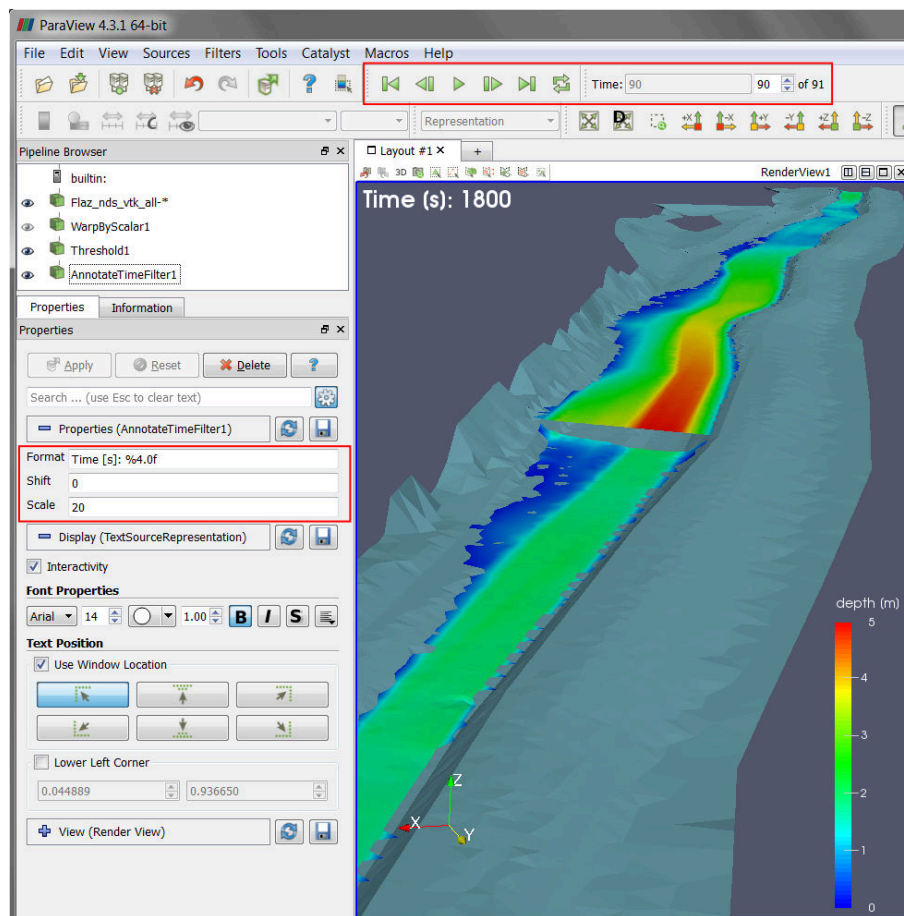


Figure 2.25 Annotation Time Filter

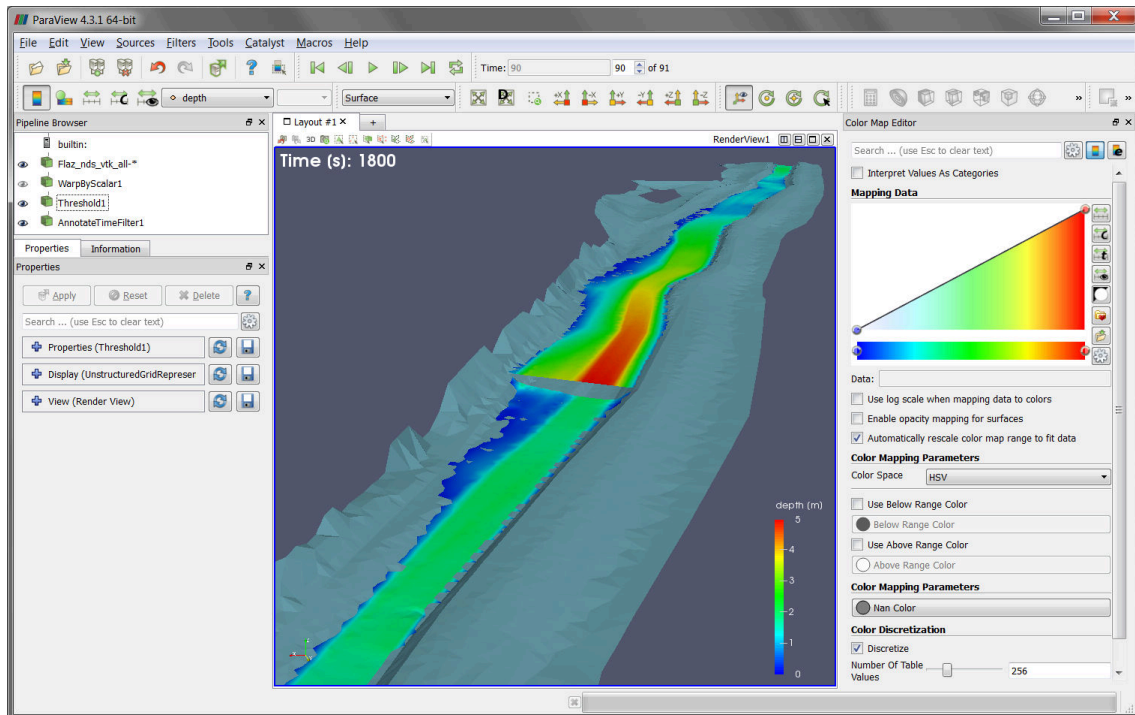


Figure 2.26 Water depth at the steady state of the simulation

to 27 water enters the domain at the upstream boundary and a backwater profile appears upstream of the inner weir boundary. After time step 28 the weir is overtopped and water is visible also downstream of the weir boundary.

2.3.7 Exporting figures and animations

One of the most important products of any visualization is screenshots and movies that can be used for reports and presentations.

2.3.7.1 Save Screenshot (picture)

- (1) Go to *File* → *Save Screenshot* and a new window with several controls will appear. (see Figure 2.27)

The *Select resolution for the image to save* entries allows you to create an image that is larger (or smaller) than the current size of the 3D view.

- (2) Press *OK* and specify a location for the file to be exported. From the dropdown you can select several file types. It is recommended to save images as PNG file though.

The save screenshot option saves the image as a raster graphic, which is natural for rendered images and very efficient for large data sets. However, elements like text and other labels often become disfigured. These elements are better represented as vector graphics, which use geometrical primitives to draw the shapes. To create a vector graphic:

- (3) Go to *File* → *Export Scene* and specify an output location. Several file types are available e.g. *.pdf or *.svg.

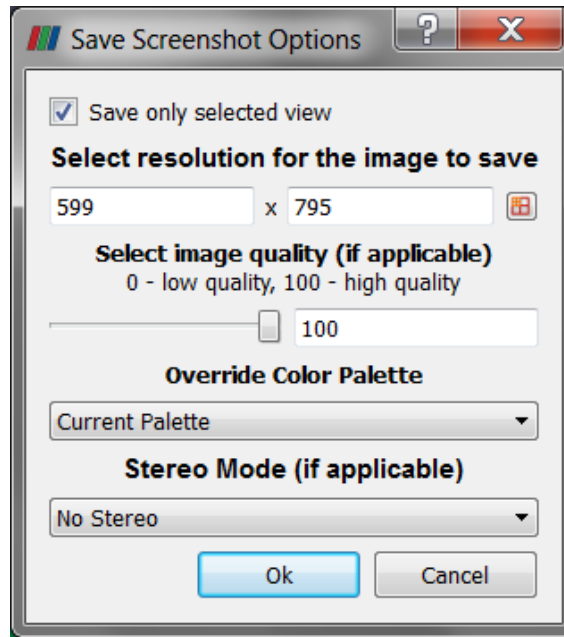


Figure 2.27 Export Screenshot

2.3.7.2 Save Animation (movie)

- (1) Go to *File* → *Save Animation* and a new window with several controls will appear. (see Figure 2.28)

Again, you can create an animation that is larger (or smaller) than the current size of the 3D view modifying the *Resolution (pixels)* entries. It is advisable to reduce the *Frame Rate (fps)* from 15 to 8, otherwise the animation might run to fast.

- (2) Press *Save Animation* and specify an output location.
- (3) Select AVI files (*.avi) as file type and press *OK*.

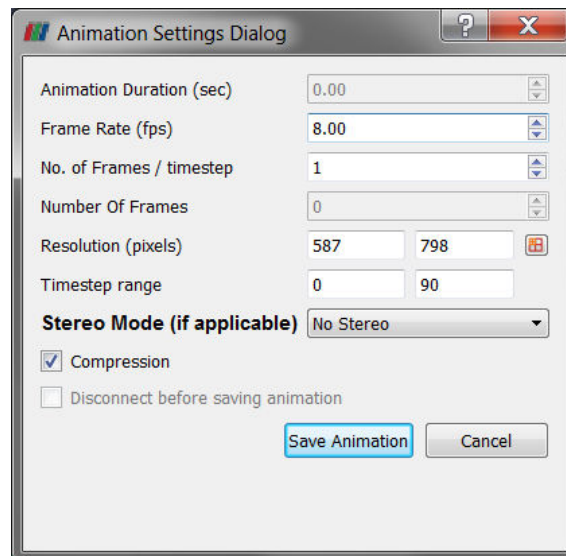


Figure 2.28 Export Animation

Hydrodynamics and sediment transport at the river Flaz with BASEHPC (2D)

3.1 Hydrodynamics and sediment transport at the river Flaz with BASEHPC (2D)

3.1.1 Introduction

The river Flaz is located in the canton Graubünden in Switzerland. A reach of 1.5 km long is taken as example for this tutorial. The reach contains a widening section created to protect the village of Samadan from flood event. The aim of this tutorial is to show the setup of the three configuration files for the numerical simulation with BASEMENT. First, a hydraulic simulation is performed to obtain a calibrated model at steady state. Then a morphological simulation is performed adding the morphological part to the result of the hydraulic simulation.

3.1.2 Computational Mesh

The computational mesh of the Flaz is imported from the tutorial of BASEMENT version 2.8. The 2dm file has been modified as explain in the pre-processing tutorial for small meshes in the User manual. There are two stringdefs for the inflow and outflow boundaries. The mesh has 14'457 cells, 7'446 vertices and the interpolation method “weighted” is used to convert the mesh from version 2.8 to a 3 compatible computational mesh. Figure 3.1 shows the bottom elevation of the river Flaz used in this tutorial.

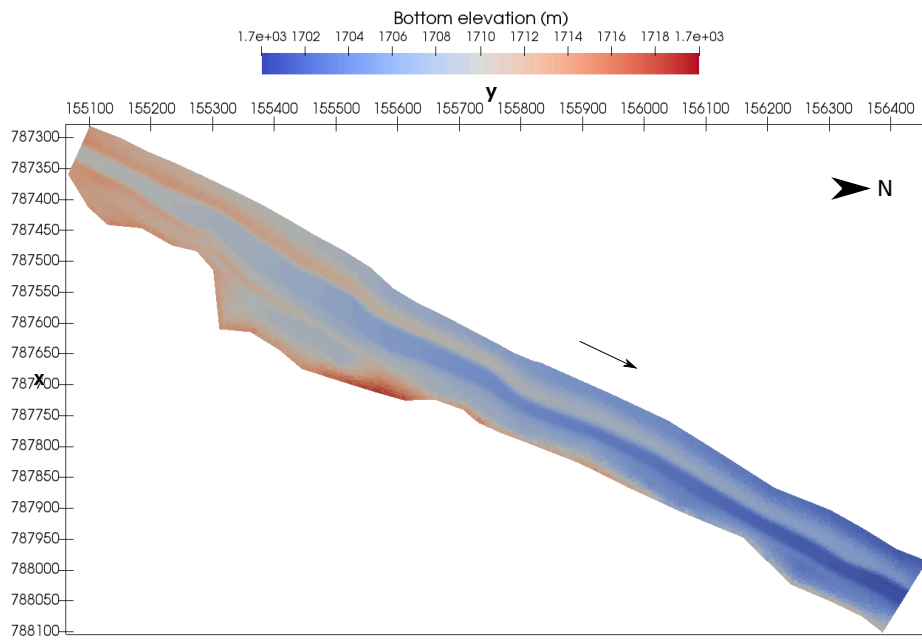


Figure 3.1 Planar view of the initial bottom elevation of the river Flaz

3.1.3 Hydraulics

The configuration files (model.json, simulation.json and results.json) can be created and modified with the graphical user interface (GUI) or any text editor. The configuration files are saved in one folder and the simulation will automatically generate a new folder called “run” which contains the output.

3.1.3.1 Setup the Configuration File model.json

The configuration file model.json for a hydrodynamic simulation has the following structure:

```
{
  "SETUP":{
    "simulation_name":"Flaz_steady_state",
    "BASEHPC": {
      "PHYSICAL_PROPERTIES": {...},
      "BASEPLANE_2D": {
        "GEOMETRY": {...},
        "HYDRAULICS": {
          "PARAMETER": {...},
          "FRICTION": {...},
          "TURBULENCE": {...},
          "BOUNDARY": {...},
          "INITIAL": {...}
        }
      }
    }
  }
}
```

```
}

```

The blocks `PHYSICAL_PROPERTIES` and `BASEPLANE_2D` are mandatory. The physical property is the gravity and the components of the `BASEPLANE_2D` contain information about the domain (`GEOMETRY`) and the simulation type (`HYDRAULICS`).

```
"PHYSICAL_PROPERTIES": {
  "gravity": 9.81
}
```

The `GEOMETRY` part contains the path to the mesh file and different subsections as the interpolation method, a list of `STRINGDEF` for boundary conditions and a list of `REGIONDEF` to assign the friction, external sources and different interpolation methods.

```
"GEOMETRY": {
  "mesh_file": "Flaz_mesh.2dm",
  "INTERPOLATION": {
    "method": "weighted"
  },
  "STRINGDEF": [...],
  "REGIONDEF": [...]
}
```

```
"STRINGDEF": [
  { "name": "Inflow",
    "upstream_direction": "left"},
  { "name": "Outflow",
    "upstream_direction": "left"}
],
"REGIONDEF": [
  {
    "name": "one",
    "index": [1]},
  {
    "name": "two",
    "index": [2]},
  {
    "name": "three",
    "index": [3]},
  ...
]
```

The `HYDRAULICS` block contains the subsections `PARAMETER` for the hydraulic simulation only, `FRICTION` for each region, `TURBULENCE` parameters, `BOUNDARY` for the flow conditions and `INITIAL` for the condition at time $t=0.0$.

```
"PARAMETER": {
  "CFL": 0.95,
```

```
"minimum_water_depth":0.002,
"fluid_density": 1000.0,
"max_time_step": 100
}

"FRICITION": {
  "type": "strickler",
  "default_friction": 30,
  "regions": [
    {"region_name": "one",
     "friction": 28.0},
    {"region_name": "two",
     "friction": 30.0},
    {"region_name": "three",
     "friction": 35.0},
    ...
  ]
}

"TURBULENCE": {
  "INITIAL": {
    "type": "zero"
  },
  "PARAMETER": {
    "turbulence_start": 0,
  },
  "REYNOLDS_STRESS": {
    "closure_model": "mix_length",
    "constants": {
      "turb_visc_max": 0.05
    }
  }
}

"BOUNDARY":{
  "STANDARD": [
    {
      "name": "Inflow",
      "string_name": "Inflow",
      "type": "uniform_in",
      "discharge_file": "Inflow_stationary.txt",
      "slope": 0.02
    },
    {
      "name": "Outflow",
      "string_name": "Outflow",
      "type": "uniform_out",
      "slope": 0.02
    }
  ]
}
```



```

    ]
  }

  "INITIAL":{
    "type": "dry"
  }

```

The hydrograph is defined constant in a file “Inflow_stationary.txt”, where the time in seconds (left) and discharge in m^3/s (right) is indicated.

```

0.0, 50.0
3600.0, 50.0

```

3.1.3.2 Setup the Configuration File simulation.json

The configuration file simulation.json defines the simulation time parameters (seconds) in the block TIME and the different output types inside the OUTPUT block.

```

{
  "SIMULATION":{
    "TIME": {
      "start": 0.0,
      "end": 3600,
      "out": 600
    },
    "OUTPUT": [
      "water_surface",
      "flow_velocity_abs",
      "ns_hyd_discharge",
      "bottom_elevation",
      "spec_discharge",
      "water_depth"
    ]
  }
}

```

3.1.3.3 Set up the Configuration File results.json

The configuration file results.json defines the output format in the block EXPORT. Currently, xmdf is the only output format available.

```

{
  "RESULTS": {
    "EXPORT": [
      {"format": "xmdf"}
    ]
  }
}

```

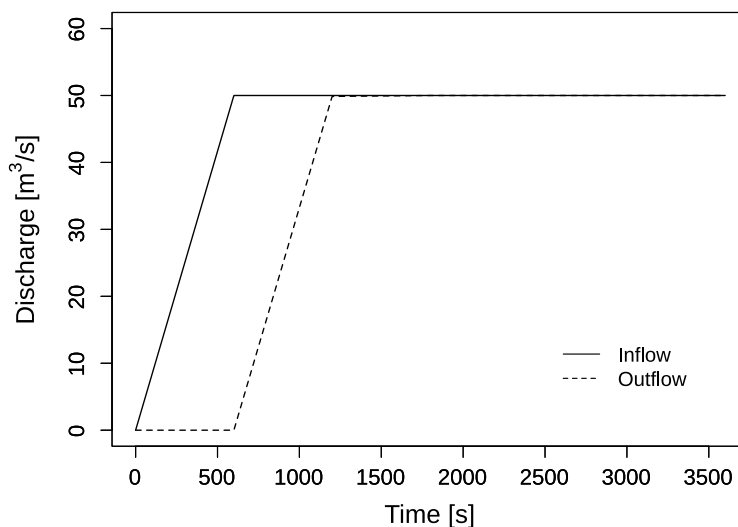


Figure 3.2 Inflow and outflow hydrograph at boundaries

3.1.3.4 Steady Flow Simulation

The simulation results are stored inside the results.h5 binary. By calling the python script BMv3NodestringResults.py available on the [BASEMENT Website](#), the discharge values at boundaries are extracted and stored in the Discharge.csv file, where the rows are the time steps and the columns represents the stringdef in their definition order (inside the .2dm file). The steady state is reached after 1200 seconds (Figure 3.2).

3.1.3.5 Model Calibration

The calibration of the friction value is done by comparing the water surface elevation between BASEMENT version 3 and version 2.8 at a cross section located in the middle of the channel (Figure 3.3). The water surface elevation and the water depth values along the cross section were obtained using the software ParaView.

The resulting water depth and water surface elevation are compared in Figure 3.4. The steady flow simulation of BASEMENT version 3 provides similar results to those obtained with BASEMENT version 2.8. There is no need to modify the friction value defined in Section 3.1.3.1.

3.1.3.6 Unsteady Flow Simulation

The hydrograph based on the flood event of July 2004 provides unsteady flow conditions for the numerical simulation. The results of the steady flow simulation are stored in the binary Flaz_steady_state_results.h5 inside the run/ folder and taken as initial state. The other parameters defined in Section 3.1.3.1 don't change, except for the boundary block where the new discharge file (Inflow_transient.txt) replaces the stationary hydrograph. The initial block:

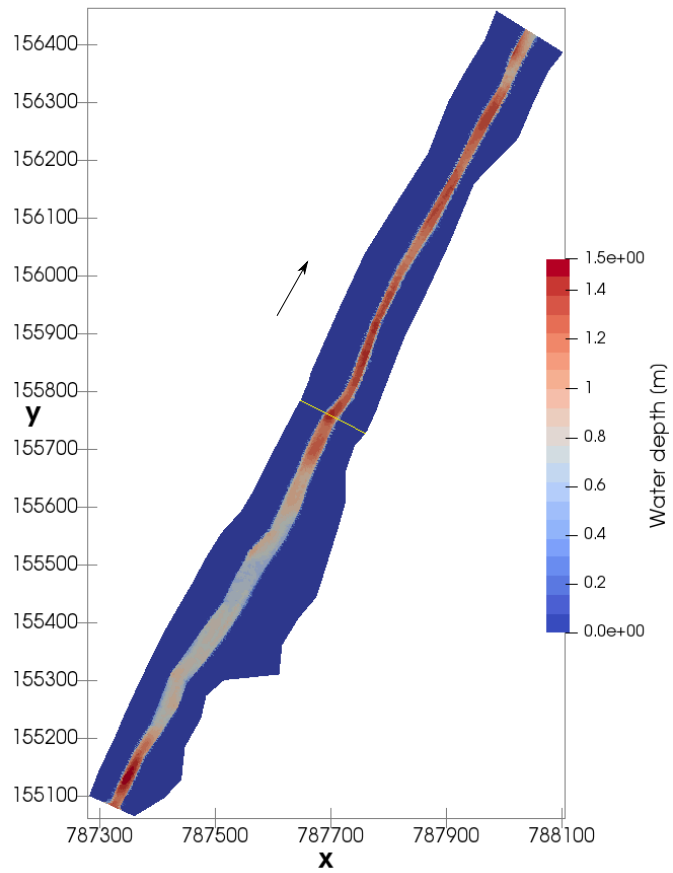


Figure 3.3 Water depth at the end of the steady flow simulation ($t=3600$ s) with the cross section location (yellow line)

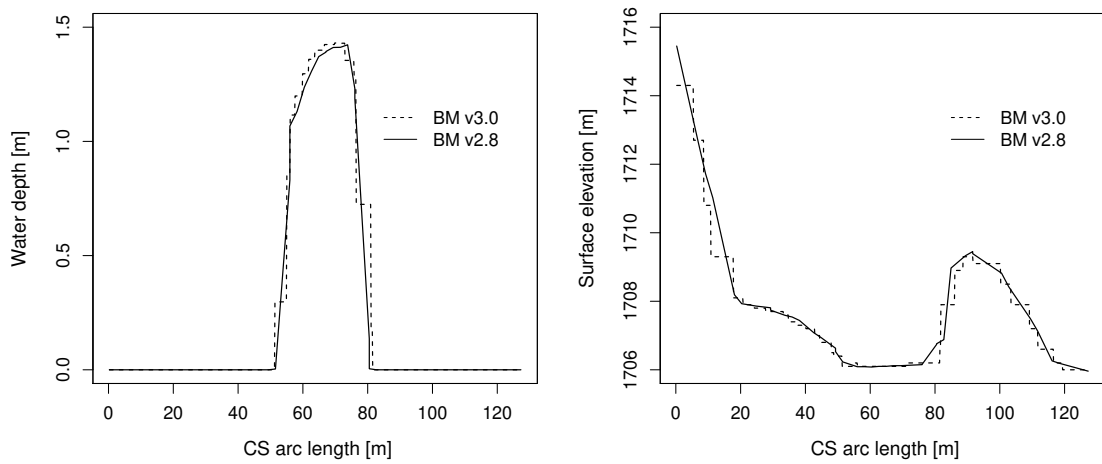


Figure 3.4 Cross sectional view of the water depth and surface elevation facing the opposite flow direction at the end of the steady flow simulation ($t=3600$ s)

```
"INITIAL":{
  "type": "continue",
  "file": "../run/Flaz_steady_state_results.h5",
  "time": 3000.0
}
```

The run time in the configuration file simulation.json is increased to the last value of the discharge file (Inflow_transient.txt). A higher “end” value will take the last discharge value written on the hydrograph to run the simulation. The starting time corresponds to the smallest time value of the discharge file Inflow_transient.txt.

```
"TIME": {
  "start": 0.0,
  "end": 82000,
  "out": 2000
}
```

After changing the discharge file, modifying the initial block in the setup and the time of the simulation, the model is ready to proceed with the numerical simulation.

3.1.4 Morphology with uniform sediment transport

3.1.4.1 Setup the Configuration File model.json

The unsteady flow simulation is now converted into a uniform sediment morphodynamic simulation with a morphology block and starting from the results of the steady flow simulation. The inflow hydrograph and the initial blocks are the same as described in Section 3.1.3.6. Additionally, the block morphology is defined inside the model.json file with the following structure

```
{
  "SETUP":{
    "simulation_name":"Flaz_unsteady_morph",
    "BASEHPC": {
      "PHYSICAL_PROPERTIES": {...},
      "BASEPLANE_2D": {
        "GEOMETRY": {...},
        "HYDRAULICS": {...},
        "MORPHOLOGY": {
          "INITIAL":{...},
          "PARAMETER": {...},
          "BEDMATERIAL": {...},
          "BEDLOAD": {...}
          "SUSLOAD": {...}
        }
      }
    }
  }
}
```

Inside the morphology block, the initial conditions look like:

```
"INITIAL":{
  "type": "mesh"
}
```

The morphology parameters defines the density of sediments, the porosity and the time at which the morphodynamic simulation starts. Further, the morphodynamic solver can be defined. For uniform sediment simulations, we recommend to always use the HLLC solver.

```
"PARAMETER": {
  "morphodynamic_start": 0.0,
  "sediment_porosity": 0.4,
  "sediment_density" : 2650.0,
  "morphodynamic_solver": "HLLC"
}
```

The bed material block inside the morphology block can be used to characterize the substrate. The structure of the block is described below. The parameter, soil definition and fixed bed blocks are optional, and either the soil assignment or the initial condition block is required:

```
"BEDMATERIAL": {
  "PARAMETER": {...},
  "GRAIN_CLASS": {...},
  "MIXTURE": {...},
  "SOILDEF": {...},
  "SOIL_ASSIGNMENT": {...},
  "INITIAL": {...},
  "FIXED_BED": {...}
}
```

The bed material parameters include the accuracy for fix bed correction defining the maximal overshoot below the fix bed elevation (*correction_accuracy*) and the maximum number of iterations for fix bed correction (*max_iteration*).

```
"BEDMATERIAL": {
  ...
  "PARAMETER": {
    "correction_accuracy": 0.001,
    "max_iteration": 10
  },
  ...
}
```

The bed material is composed of uniform grains with one diameter (m).

```
"BEDMATERIAL": {
  ...
  "GRAIN_CLASS": {
    "diameters": [0.050]
  },
  ...
}
```

Since we have only one grain size, the volume fraction of *mixture_uniform* is equal to 100%.

```
"BEDMATERIAL": {
  ...
  "MIXTURE": [
    {
      "name": "mixture_uniform",
      "volume_fraction": [100]
    }
  ],
  ...
}
```

The soil layers and the according sediment mixture are defined in the SOILDEF-block. For a single-grain simulation it is not important how many layers are defined. The negative relative elevation defines the thickness of the layers. The default value of -999.0 results in an infinite substrate (no fixed bed). Setting a *z_rel* value larger than -999.0 results in the definition of a fixed bed.

```
"BEDMATERIAL": {
  ...
  "SOILDEF": [
    {
      "name": "soil1",
      "LAYER": [
        {
          "z_rel": -999.0,
          "mixture_name": "mixture_uniform"
        }
      ]
    }
  ],
  ...
}
```

Soils can subsequently be assigned to different regions of the computational domain within the SOIL_ASSIGNMENT block. Here, we define *soil1* as the default soil for the entire computational domain.

```
"BEDMATERIAL": {
  ...
  "SOIL_ASSIGNMENT": {
    "default_soil": "soil1"
  },
  ...
}
```

Note that there are three ways to define a fixed bed elevation: (1) via regions in the FIXED_BED block, or (2) via a separate .2dm mesh file in the FIXED_BED block, or (3) via layers in the SOILDEF block, as mentioned above. If fixed bed elevations are defined both in the FIXED_BED and SOILDEF blocks, definitions in FIXED_BED block overwrite fixed bed information from the SOILDEF block. In this example, we initialize the soil without a fixed bed ($z_{rel} = -999.0$) and instead define the fixed bed via the FIXED_BED block. Fixed bed elevations are assigned to different regions to prevent a high erosion. The erosion is unlimited if the fix bed is not defined.

```
"BEDMATERIAL": {
  "FIXED_BED": {
    "type": "region_defined",
    "regions": [
      {"region_name": "one",
       "z_rel": -0.8},
      {"region_name": "two",
       "z_rel": 0.0},
      {"region_name": "three",
       "z_rel": 0.0},
      ...
    ]
  }
}
```

In the bedload block, the bedload transport formula is chosen and the morphological boundary conditions are defined. The boundary condition is defined as `transport_capacity` at the inflow boundary. The value of the sediment flux is averaged over the stringdef length and equally distributed (same value) among the edges. Additionally, the mixture of the sediment entering the domain has to be specified. The outflow boundary has been reduced to the bed width composed of 6 nodes only. Therefore, the nodes located on the channel levee of the outflowing stringdefs have been removed from the .2dm file (computational mesh) manually using a text editor. The stringdef definition remains unchanged and the outflow boundary is defined as `equilibrium_out` with the reference bottom elevation set to 1700.68 m. Additionally, lateral transport due to local slope bed slope is considered with a default value of 1.5 (see Reference Manual).

```
"BEDLOAD": {
  "FORMULA": {
    "type": "MPM_like",
    "factor": 1.0,
    "coefficient": 3.2,
  }
}
```

```

    "exponent":1.6,
    "theta_critical_approach": "constant"
    "critical_value":0.047
  },
  "BOUNDARY":{
    "STANDARD": [
      {
        "name": "inflow_MOR",
        "string_name": "Inflow",
        "type": "transport_capacity",
        "boundary_factor": 0.8,
        "mixture_name": "mixture_inflow"
      },
      {
        "name": "outflow_MOR",
        "string_name": "Outflow",
        "type": "equilibrium_out",
        "reference_bed_elevation": 1700.68
      }
    ]
  },
  "DIRECTION": {
    "LATERAL_SLOPE": {
      "factor": 1.5
    }
  }
}

```

An additional suspended load block can be defined as well. The erosion and deposition laws are chosen and the morphological boundary conditions are defined. The boundary condition is defined as an input concentration at the inflow boundary. The outflow boundary is a simple zero-gradient outflow.

```

"SUSLOAD": {
  "PARAMETER": {
    "suspended_diffusivity": [0.1],
  },
  "BOUNDARY": {
    "STANDARD": [
      {
        "name": "inflow_SUS",
        "string_name": "Inflow",
        "type": "concentration_in",
        "data1": 0.0001
      },
      {
        "name": "outflow_SUS",
        "string_name": "Outflow",
        "type": "zero_gradient_out"
      }
    ]
  }
}

```



```

    }
  ]
},
"INITIAL": {
  "type": "zero"
},
"SETTLING_VELOCITY": {
  "closure_model": "van_rijn"
},
"SHEAR_PARTITION": {
  "default_friction_susp": 67.0
},
"EROSION_RATE": {
  "closure_model": "van_rijn",
  "constants": {
    "critical_shields": 0.125
  }
},
"DEPOSITION_RATE": {
  "closure_model": "lin",
  "constants": {
    "param_m": 1.0
  }
}
}
}

```

3.1.5 Morphology with mixed-size sediment transport

3.1.5.1 Setup the Configuration File `model.json`

In order to avoid duplication, only the modifications of the `model.json` file, compared to the single-grain simulation, are detailed here.

In the PARAMETER-block important parameters for the morphological simulation are defined. For mixed-size sediment transport simulations, the *Upwind* solver must be used. We recommend the use of an upwind factor of 0.7 to 0.8. Further, the thickness of the bed load control volume is an important calibration parameter, which influences significantly the grain sorting process. Here, the constant active layer thickness is set to a constant thickness of 0.1 m.

```

"PARAMETER": {
  ...
  "morphodynamic_solver": "Upwind",
  "upwind_factor": 0.8,
  "ACTIVE_LAYER": {
    "active_layer_type": "constant",
    "active_layer_thickness": 0.1
  }
}
}

```

The grain size distribution of the bed material is discretized with six grain classes. They have to be defined in ascending order from the smallest to the largest grain.

```
"BEDMATERIAL": {
  ...
  "GRAIN_CLASS": {
    "diameters": [0.001, 0.005, 0.015, 0.044, 0.082, 0.150]
  },
  ...
}
```

In the MIXTURE block the volume fraction of the different mixtures are defined. The three river sections are considered with different sediment mixtures. Furthermore, a mixture for the inflow is defined.

```
"BEDMATERIAL": {
  ...
  "MIXTURE": [
    {
      "name": "mixture_inflow",
      "volume_fraction": [15, 15, 23, 27, 10, 10]
    },
    {
      "name": "mixture_element_roughness",
      "volume_fraction": [17, 11, 14, 27, 14, 17]
    },
    {
      "name": "mixture_widening",
      "volume_fraction": [21, 13, 16, 25, 11, 14]
    },
    {
      "name": "mixture_alt_bars",
      "volume_fraction": [27, 14, 14, 20, 14, 11]
    }
  ],
  ...
}
```

The soil layers with the corresponding sediment mixture are defined in the SOILDEF block. The soil can be defined with up to two layers of different material, but to keep it simple we assume a single layer. The negative bottom elevation defines the thickness of the layer. We use this especially for the river bed near the upper boundary condition to avoid uncontrolled erosion. A fully mobile bed (no fixed bed) is defined for the two soils *soil_widerning* and *soil_alt_bar* by using the default value of -999.0 for *z_rel* of the lowest layer. Furthermore the embankments are kept fix because the main focus is on set on the river bed morphology. The soils *soil_fix_20*, *soil_fix_40* and *soil_element_roughness* are defined to have a gradual transition from the fixed bed to the movable bed.

```
"BEDMATERIAL": {
```

```
...
"SOILDEF": [
  {
    "name": "soil_element_roughness",
    "LAYER": [
      {"z_rel": -0.8,
       "mixture_name": "mixture_element_roughness"}
    ]
  },
  {
    "name": "soil_widening",
    "LAYER": [
      {"z_rel": -2.0,
       "mixture_name": "mixture_widening"}
    ]
  },
  {
    "name": "soil_alt_bars",
    "LAYER": [
      {
        "z_rel": -2.0,
        "mixture_name": "mixture_alt_bars"
      }
    ]
  },
  {
    "name": "soil_fix",
    "LAYER": [
      {
        "z_rel": -0.0,
        "mixture_name": "mixture_element_roughness"
      }
    ]
  },
  {
    "name": "soil_fix_20",
    "LAYER": [
      {
        "z_rel": -0.2,
        "mixture_name": "mixture_element_roughness"
      }
    ]
  },
  {
    "name": "soil_fix_40",
    "LAYER": [
      {
        "z_rel": -0.4,
        "mixture_name": "mixture_element_roughness"
      }
    ]
  }
]
```

```

    }
  ]
}
],
}

```

Soils can subsequently be assigned to different regions of the computational domain within the SOIL_ASSIGNMENT block. Here, we define *soil_fix* as the default soil, but also individually assign the defined soils to the various regions of the mesh.

```

"BEDMATERIAL": {
  ...
  "SOIL_ASSIGNMENT": {
    "default_soil": "soil_fix",
    "regions": [
      {
        "region_name": "one",
        "soil_name": "soil_element_roughness"
      },
      {
        "region_name": "two",
        "soil_name": "soil_fix"
      },
      {
        "region_name": "three",
        "soil_name": "soil_fix"
      },
      ...
    ]
  }
}

```

Since we already defined the fixed bed elevations during the soil definition, we do not require the fixed bed block.

In the bedload block, the bedload transport formula MPM_multi is chosen, which extends formula of Meyer-Peter and Müller (1948) for sediment mixtures by considering hiding and exposition effects with the approach of Ashida and Michiue (1971). At the inflow boundary, the sediment mixture *mixture_inflow* is specified for any sediment entering the domain.

```

"BEDLOAD": {
  "FORMULA": {
    "type": "MPM_multi",
    ...
  },
  "BOUNDARY": {
    "STANDARD": [
      {
        "name": "inflow_MOR",

```

```

        "string_name": "Inflow",
        "type": "transport_capacity",
        "boundary_factor": 0.8,
        "mixture_name": "mixture_inflow"
    },
    ...
]
},
...
}

```

3.1.5.2 Setup the Configuration File simulation.json

The simulation time defined in the simulation.json file is the same as for the uniform sediment transport simulation and eleven additional outputs are defined (active layer thickness, active layer fractions, fractions of the upper and lower sublayer, elevation of the lower edge of the sublayers, the arithmetic and geometric mean grain size of the active layer, and the characteristic diameters d_{16} , d_{50} , d_{84} , and d_{90} of the active layer.

```

{
  "OUTPUT": [
    "...",
    "active_layer_thickness",
    "active_layer_fractions",
    "sublayer_fractions0",
    "sublayer_fractions1",
    "sublayer_elevations",
    "dm_arithmetic",
    "dm_geometric",
    "d_16",
    "d_50",
    "d_84",
    "d_90"
  ]
}

```

3.1.5.3 Results

The morphological changes of the river bed are observed on Figure 3.5. The software ParaView was used for the post-processing of the output file (.xdmf).

3.1.6 Passive tracers

3.1.6.1 Setup the Configuration File model.json

This section shows how the flow simulation could be augmented with scalar advection with a TRACERS block. A maximum of 5 tracers are presently supported. Additionally, the block tracers is defined inside the model.json file with the following structure:

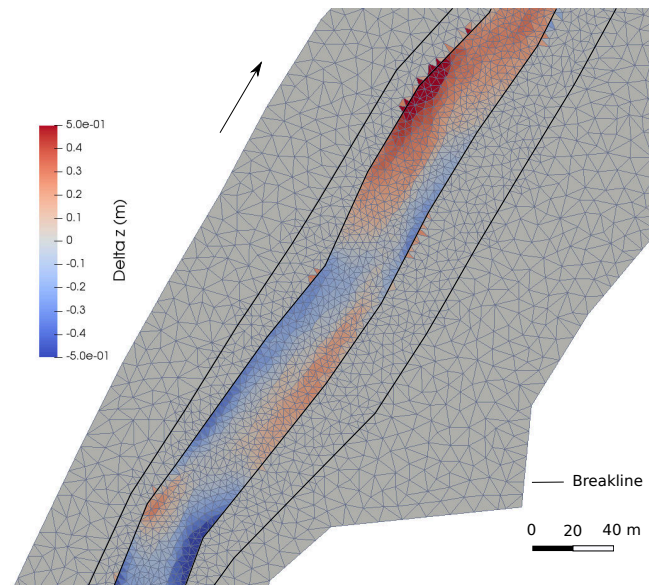


Figure 3.5 Planar view of the widening part of the river Flaz with the change in bed elevation (Δz) due to the flood event at the end of the morphodynamic simulation ($t=84'000$ s)

The configuration file model.json for a tracers simulation has the following structure:

```
{
  "SETUP":{
    "simulation_name":"Flaz_tracers",
    "BASEHPC": {
      "PHYSICAL_PROPERTIES": {...},
      "BASEPLANE_2D": {
        "GEOMETRY": {...},
        "TRACERS": {
          "PARAMETER": {...},
          "BOUNDARY": {...},
          "INITIAL": {...},
          "SOURCE": {...}
        }
      }
    }
  }
}
```

The TRACERS block contains the subsections PARAMETER, BOUNDARY and INITIAL:

```
"PARAMETER": {
  "tracers_start": 0.0,
  "num_tracers": 3,
}

"BOUNDARY":{
```

```

"STANDARD": [
  {
    "name": "tracer_inflow",
    "string_name": "Inflow",
    "type": "discharge_in",
    "tracer1_file": "Inflow_tracer1.txt",
    "tracer2_file": "Inflow_tracer2.txt",
    "tracer3_file": "Inflow_tracer3.txt"
  },
  {
    "name": "Outflow",
    "string_name": "Outflow",
    "type": "zero_gradient_out"
  }
]
}

```

The tracer discharge series are of the same format of the hydrograph: two columns with time (first col.) and imposed quantity (discharge or concentration) (second col.). To specify constant values use “tracerX” followed by the intended value, instead of “tracerX_file”.

```

"INITIAL":{
  "type": "zero"
}

```

```

"INITIAL":{
  "type": "uniform",
  "tracers": [0.0,0.25,0.5]
}

```

```

"INITIAL":{
  "type": "region_defined",
  "regions": [
    {"region_name": "one",
     "tracers": [0.0,0.1,0.2]},
    {"region_name": "two",
     "tracers": [0.4,0.3,0.2]}
  ]
}

```

The initial conditions are prescribed in array format with the “tracers” keyword. The supported types of initial conditions with this input method are “uniform” and “region_defined”. The remaining options are “zero”, with no additional inputs, and “continue” accepting “file” and “time” parameters.

A source block within the tracers block would look like:

```

"SOURCE": [
  {

```

```

    "name": "total_exact1",
    "type": "total",
    "data1_file": "./scalars1.dat",
    "data2_file": "./scalars2.dat",
    "data3_file": "./scalars3.dat",
    "data4_file": "./scalars4.dat",
    "data5_file": "./scalars5.dat",
    "region_name": "one",
    "sink": "exact"
  }
]

```

And, as in the boundaries case, to specify constant values use “dataX” followed by the intended value, instead of “dataX_file”.

3.1.6.2 Setup the Configuration File simulation.json

The configuration file simulation.json defines the simulation time parameters (seconds) in the block TIME and the different output types inside the OUTPUT block.

```

{
  "SIMULATION":{
    "TIME": {
      "start": 0.0,
      "end": 3600,
      "out": 600
    },
    "OUTPUT": [
      "water_surface",
      "flow_velocity_abs",
      "ns_hyd_discharge",
      "bottom_elevation",
      "spec_discharge",
      "water_depth",
      "tracer1",
      "tracer2",
      "tracer3"
    ]
  }
}

```

3.1.6.3 Set up the Configuration File results.json

The configuration file results.json defines the output format in the block EXPORT. Currently, xmdf is the only output format available.

```

{
  "RESULTS": {

```



```
    "EXPORT": [  
      {"format": "xdmf"}  
    ]  
  }  
}
```

Hydrodynamics and sediment transport at the river Flaz with BASEMD (2D)

4.1 Hydrodynamics and sediment transport at the river Flaz with BASEMD (2D)

4.1.1 Introduction

This tutorial gives an introduction to the capabilities of the 2D modelling module BASEplane of BASEMENT. It provides a step-by-step guidance on how build up a model for BASEplane.

4.1.1.1 Case study description

The tutorial for the 2D modelling module is based on an extract of the case study of the River Flaz in Graubünden. Within the framework of a high water protection project for the village Samedan a completely new section of the River Flaz was built. On a length of 4.1 km morphologically different kind of river subsections can be distinguished (Figure 4.1). The numerical modelling of the whole domain is carried out within the river monitoring project Flaz of the Laboratory of Hydraulics, Hydrology and Glaciology (VAW). In order to reduce the model size (and thus computational running time) only the three most interesting sub-sections are modelled in this tutorial, such as the lower part of the section enriched with roughness elements, the widening part and the part with alternating bars shown in Figure 4.1.

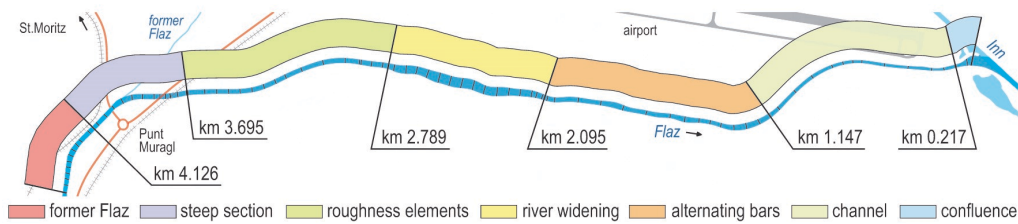


Figure 4.1 Different morphological river subsections of the new section of the river Flaz

4.1.1.2 Tutorial structure

In a first step the important properties of the mesh file are shown. The tutorial is designed to run `BASE_MD_` with the help of the free and open source Geographic Information System `QGIS` as pre- and post-processor. The computational grid used in this tutorial (`Flaz_mesh.2dm`) covers the same perimeter like the one generated in the tutorial of `BASEmesh` (see Section 1.1). However the position of the breaklines differs slightly from the version of the `BASEmesh` tutorial. Furthermore the material indices have been changed for the definition of friction factors and soil properties.

How to visualize the results of the 2D simulations using `QGIS` and how to generate the graphics shown in this tutorial is demonstrated in the post-processing tutorial (Section 2.1). The main focus of this tutorial lies on the setup the command file for the numerical simulation with `BASEMENT`.

The structure of this tutorial reflects the recommended steps to go from a hydraulic simulation to a morphodynamic simulation. A hydraulic simulation is initially set up. Based on this simulation the morphological part can be added with a single-grain model. An outline of outputs is given and possible visualization is shown for each step.

4.1.2 Computational grid

The computational mesh is generated with the `QGIS` plugin `BASEmesh`. Here, only a few important features and characteristics of the computational mesh for modelling with `BASEMENT` are mentioned. The mesh discretizes the topography of the river in such a way that the important topographical information is maintained. Breaklines in the mesh are ensuring that important features of the topography such as the river bed and dike crests are represented correctly (Figure 4.2).

An important feature is the assignment of the material indexes to the different groups of elements. By the material index different properties such as the friction factor and the soil properties can be assigned. The material index is mainly used to assign the friction factor to the different river sections separately. For example it is usual to assign different values for the main channel, embankments, and surrounding floodplain (Figure 4.3).

The mesh file is saved as `Flaz_mesh.2dm` and has the following structure:

```
MESH2D
.
E3T    eN    n1    n2    n3    eMi    (triangle element)
.
ND     ni   x    y    z
.
```

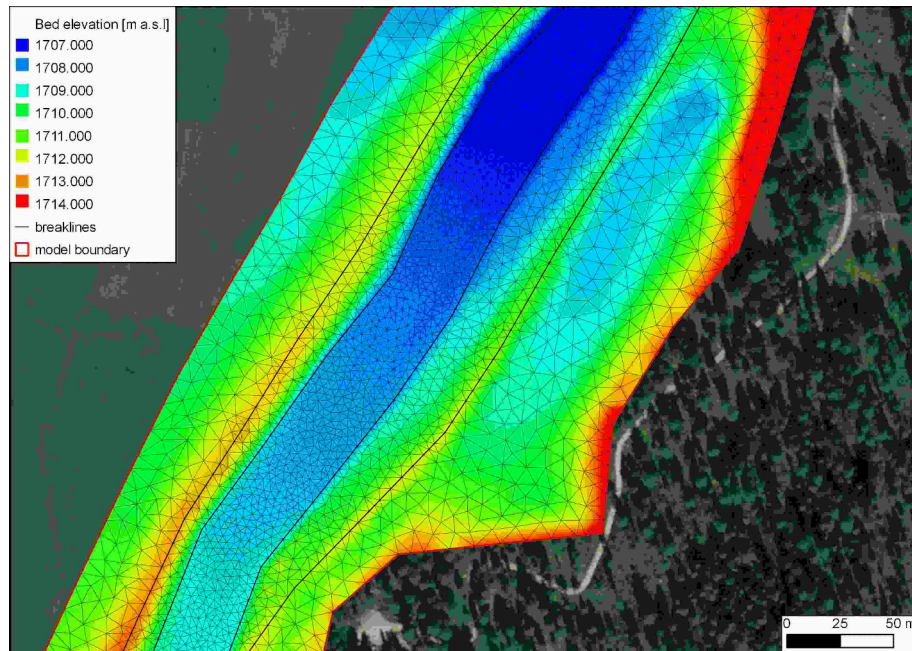


Figure 4.2 Computational mesh generated with BASEmesh for a section in the widening part. The bold black lines are breaklines for the mesh.

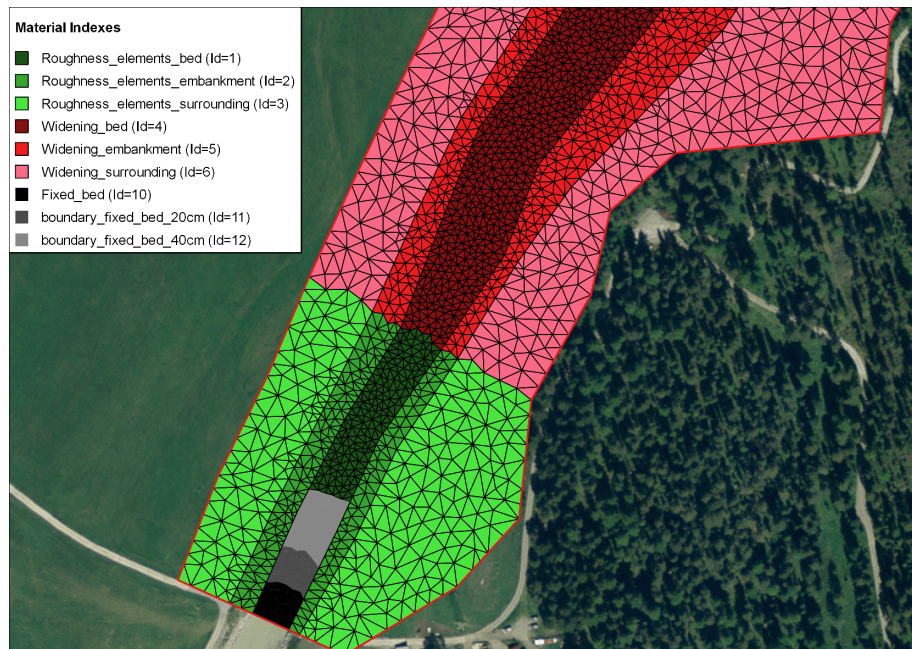


Figure 4.3 Material indices used for assignment of the friction factor and the soil properties.

Where E3T is the flags for the triangular elements; eN denotes the element number; n1, n2 and n3 denote the node numbers of the element and eMi is the material index of the element. The elements are defined in a counter-clockwise direction. The coordinates of the nodes are defined in the second block. ND is the flag for a node; ni denotes node number and x, y and z are the coordinates of the node.

4.1.3 Setting up the model file

The configurations files (`model.json`, `simulation.json` and `results.json`) can be created and modified with the graphical user interface (GUI) or any text editor. The configuration files must all be located in one folder.

Open the model setup file `model.json` via the menu of the BASEMENT GUI (File → Open Scenario Directory → select the folder that contains the configuration files). The structure of the model setup file looks like this:

```

SETUP
{
  "simulation_name": "2D_Tutorial"
  BASEMD
  {
    multiregion = Flaz
    PHYSICAL_PROPERTIES
    {...}
    BASEPLANE_2D
    {
      region_name = Flaz
      GEOMETRY
      {...}
      HYDRAULICS
      {...}
      TIMESTEP
      {...}
      MORPHOLOGY
      {...}
      OUTPUT
      {...}
    }
  }
}

```

4.1.3.1 BASEMD

The BASEMD-block includes all necessary blocks for a simulation.

```

BASEMD
{
  multiregion = Flaz

```

```

    PARALLEL
    {...}
    PHYSICAL_PROPERTIES
    {...}
    BASEPLANE_2D
    {...}
}

```

4.1.3.2 Physical properties

The physical properties are global constants in a project.

```

PHYSICAL_PROPERTIES
{
    gravity = 9.81           // [m/s2]
    viscosity = 1.0e-6      // [m2/s]
    rho_fluid = 1000        // [kg/m3]
}

```

4.1.3.3 Two dimensional domain

The *BASEPLANE_2D*-block within the BASEMD-block contains all information concerning the two dimensional model domain.

```

BASEPLANE_2D
{
    region_name = Flaz
    GEOMETRY
    {...}
    HYDRAULICS
    {...}
    TIMESTEP
    {...}
    MORPHOLOGY
    {...}
    OUTPUT
    {...}
}

```

4.1.3.4 Geometry

The GEOMETRY-block defines the mesh file and necessary strings of nodes. Strings are used for inflow and outflow boundaries and can also be used for discharge control. The node ids of the inflow and outflow string can be read out from the mesh in Figure 4.4 and Figure 4.5 respectively.

```

GEOMETRY

```

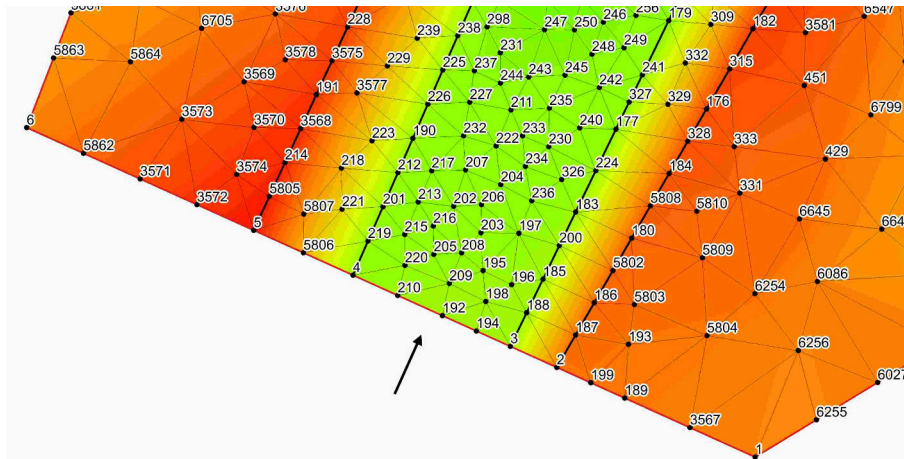


Figure 4.4 Node id numbers for the definition in the `STRING_DEF` block for the inflow boundary.

```
{
  type = 2dm
  file = Flaz_mesh.2dm

  STRINGDEF
  {
    name = Inflow
    node_ids = ( 2 3 194 192 210 4 5806 5 )
    upstream_direction = left
  }
  STRINGDEF
  {
    name = Outflow
    node_ids = ( 38 5568 5569 5570 37 5565 5538 5521 5551 36 6004 35 )
    upstream_direction = left
  }
}
```

4.1.3.5 Define the hydraulics

The `HYDRAULICS`-block includes all the information necessary for the hydraulic part of the simulation. This block is divided into the following sub-blocks:

```
HYDRAULICS
{
  BOUNDARY
  {...}
  INITIAL
  {...}
  FRICTION
  {...}
  PARAMETER
```

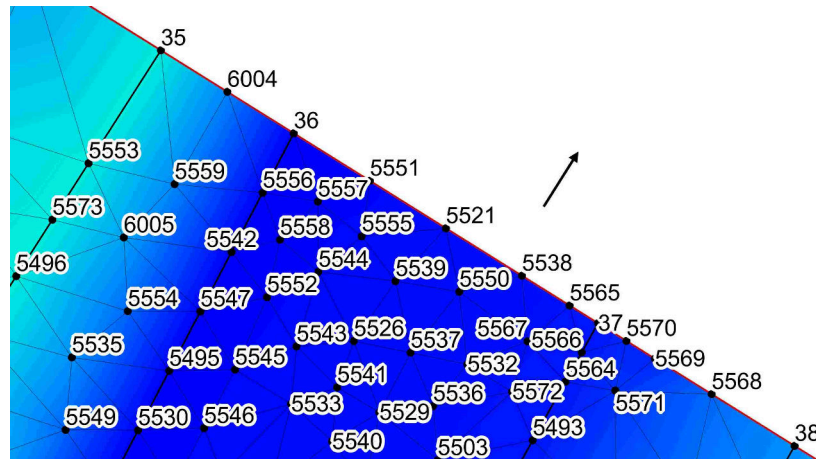



Figure 4.5 Node id numbers for the definition in the `STRING_DEF` block for the outflow boundary.

```
{...}
}
```

4.1.3.5.1 Hydraulic boundary conditions

For the upper (inflow) and lower (outflow) boundary condition we have to refer to the predefined `STRINGDEFs` (see Section 4.1.3.4). If the boundary condition is not defined explicitly a wall boundary is considered for those edges. Except for the explicitly defined inflow and outflow boundary the model boundary is basically an impermeable wall.

The inlet boundary condition is defined across the predefined string *Inflow*. The hydraulic condition at the boundary is set by the use of a hydrograph and a corresponding slope. The normal slope is used in order to calculate the normal flow depths and the normal flow velocities at the boundary and can be considered as a calibration parameter.

```
BOUNDARY
{
  type = hydrograph
  string_name = Inflow
  file = Inflow_stationary.txt
  slope = 10.0           // [per mill]
}
```

The hydrograph is saved in a text file *Inflow_stationary.txt* in which the first column is the time and the space separated second column is the discharge (Figure 4.6). As a first step, a steady inflow hydrograph is an appropriate choice in order to test the mass conservation of the model. After a certain run time, depending on the size of the model domain, the outflow should counterbalance the inflow. There should be no uncontrolled mass loss within the model domain.

Usually the discharge is taken as the mean annual discharge or the beginning discharge of a flood event. In this case a steady discharge of $50 \text{ m}^3/\text{s}$ is chosen in order to continue later on with a flood hydrograph which starts in this range.

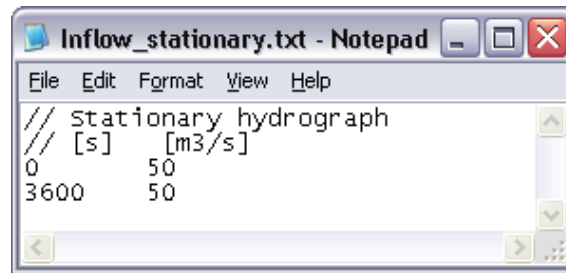


Figure 4.6 Stationary hydrograph file saved as *Inflow_stationary.txt*.

The outlet boundary condition is defined across the predefined string *Outflow*. The normal slope is used in order to calculate the normal flow depths and the normal flow velocities at the boundary and can be considered as a calibration parameter. A sensitivity analysis of this parameter should always be done. In any case the upper and lower model boundary should be far away enough from the river section of interest, in order to minimise the influence of the boundary conditions.

```
BOUNDARY
{
    type = hqrelation
    string_name = Outflow
    slope = 2.0          // [per mill]
}
```

4.1.3.5.2 Initial condition

The INITIAL-block defines the flow variables at the beginning of the simulation. In a very first step the simulation is started with a dry initial condition.

```
INITIAL
{
    type = dry
}
```

4.1.3.5.3 Friction

The FRICTION-block defines everything related to the friction term in the shallow water equations. Within the computational mesh, a material index is assigned to all elements. By the use of this material index (see Figure 4.3) a friction factor can be assigned. The default friction is used whenever there is no friction assigned to an element.

```
FRICTION
{
    type = strickler
    default_friction = 30
    input_type = index_table
    index = ( 1 2 3 4 5 6 7 8 9 10 11 12 )
    friction = ( 28 30 35 30 30 30 32 32 35 28 28 28 )
    wall_friction = off
}
```

4.1.3.5.4 Computational parameters

The PARAMETER-block defines the control parameters for the numerical simulation of the hydraulic part. The numerical simulation is performed using explicit time integration and the exact Riemann solver for flux computation. The elements with a water depth below the minimum water depth will be considered as dry elements due to stability reasons.

```
PARAMETER
{
    simulation_scheme = exp
    riemann_solver = exact
    minimum_water_depth = 0.05
}
```

4.1.3.5.5 Define the timestep

The simulation is performed with a total runtime of 3000 seconds. Later on, it has to be tested that after this runtime the flow in the model domain has reached a steady state, meaning that the outflow counterbalances the inflow (see Section 4.1.3.7).

```
TIMESTEP
{
    start_time = 0.0
    total_run_time = 3000
    CFL = 0.95
    minimum_time_step = 0.0001
}
```

4.1.3.5.6 Define the output

In the OUTPUT-block, the desired output has to be defined. During the simulation, output can also be visualized with BASEviz. In order to visualize the 2D results with QGIS, chose the output format 'sms'. For ParaView visualization chose type 'vtk'.

```
OUTPUT
{
    console_time_step = 100
    SPECIAL_OUTPUT
    {
        type = BASEviz
        output_time_step = 10
    }
    SPECIAL_OUTPUT
    {
        format = sms
        type = node_centered
        values = ( depth velocity wse )
        output_time_step = 500
    }
}
```

```

SPECIAL_OUTPUT
{
    format = vtk
    type = node_centered
    values = ( depth velocity wse )
    output_time_step = 500
}
SPECIAL_OUTPUT
{
    type = balance
    balance_values = ( timestep )
    output_time_step = 100
}
SPECIAL_OUTPUT
{
    type = boundary_history
    boundary_values = ( Q )
    history_one_file = yes
    output_time_step = 100
}
}

```

4.1.3.6 Write the setup file

Once all the inputs have been set in the in the *model.json* file, a setup file must be written. Click *write* in the bottom right corner of the GUI. This will combine the model file with the computational mesh and external data, and store it in a binary file named *setup.h5*.

4.1.3.7 Perform hydraulic simulations

4.1.3.7.1 Setting up the simulation file

To set up the simulation file, switch to the *simulation* tab on the left. Select whether the simulation should be run on a single core (select *Single-threaded on CPU*) or on multiple cores (selected *Multi-threaded on CPU*). If multiple cores are selected, you must enter the *number of CPU cores*. This selection should depend on the number of threads your computer has available.

4.1.3.7.2 Perform steady flow simulation

Run the simulation with the *Run* button of the BASEMENT window. If the SPECIAL_OUTPUT of the type *BASEviz* is chosen press the keyboard button *p* to start the simulation. Be aware that the configuration files, computational mesh, and all other input files have to be in the same folder. The output files are stored in this same folder. In order to check the mass conservation of the model, the files *Flaz_bnd_Inflow_th.dat* and *Flaz_bnd_Outflow_th.dat* are used. After approximately 1600 seconds the outflow counterbalances the steady inflow (Figure 4.7).

In the file *Flaz_balance.dat* the run time of the simulation, the computational time steps and the element which is limiting the computational time step are stored. The identification

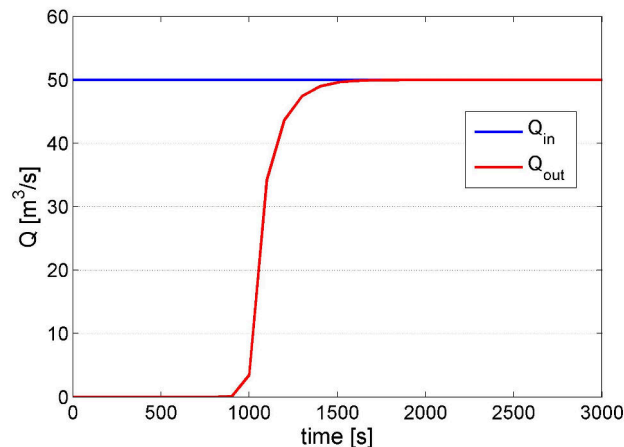


Figure 4.7 Steady inflow hydrograph and outflow hydrograph.

of the limiting element allows for improvement of the mesh. In order to identify the element location, load the computational mesh into QGIS.

The solution files with the ending *.sol* can be imported into the program QGIS (see Section 2.1) and the water depth and flow velocities can be visualized as shown in Figure 4.8. At the end of the simulation the flow variables of the last time step ($t = 3000$ s) are stored in the *Flaz_restart.h5* binary file. This file can be used later on to continue the simulation or to use it as an initial condition for a new simulation.

4.1.3.7.3 Perform unsteady flow simulation

The unsteady flow simulation is based on the flood event of July 2004 (Figure 4.10). Compared to the steady flow simulation the command file needs some minor changes. First, the last time step of the steady simulation is taken as initial condition for the unsteady simulation. Therefore the file *Flaz_restart.h5* from the steady simulation can be renamed and saved for example as *Initial_Condition.h5*. This file now can be used as an initial condition for the unsteady flow simulation. Therefore we have to consider two things:

1. Choose the solution time to be used as initial condition (note: restart file may contain several solution times).
2. Define the start time for the new simulation.

In this case we set the tag *restart_solution_time* equal to the solution of the last time step of the stationary simulation (*restart_solution_time* = 3000.027 sec). In order to start the simulation from time = 0 again, we have to set the *start_time* in the TIMESTEP block to zero. Be aware that if the *start_time* flag is not defined, the *restart_solution_time*, which is defined in the INITIAL block, will be used as start time (in this case 3000.027 sec).

```
INITIAL
{
  type = continue
  file = Initial_Condition.h5
  restart_solution_time = 3000.027
```

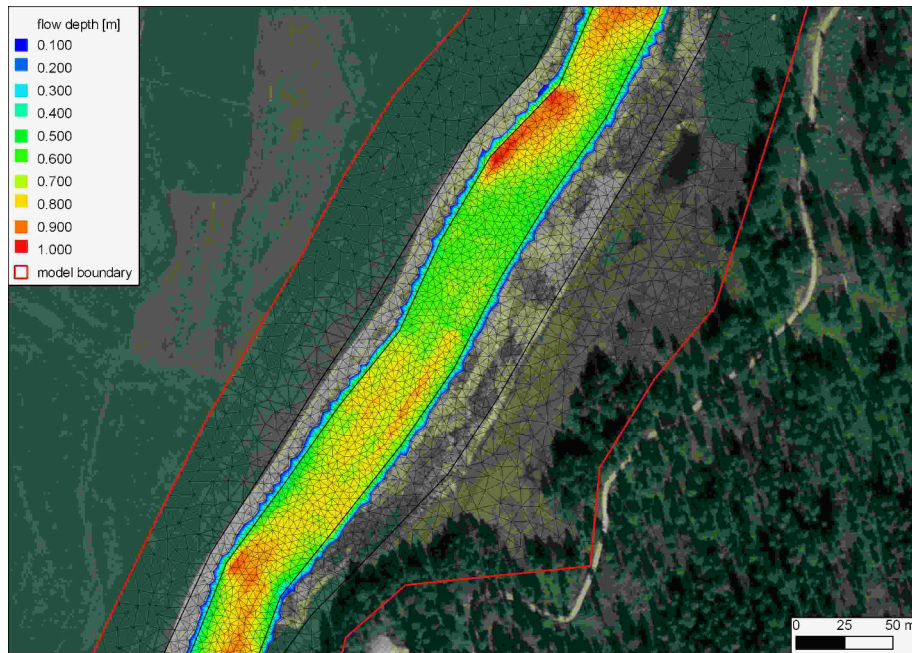


Figure 4.8 Flow depth at the steady state of the model.

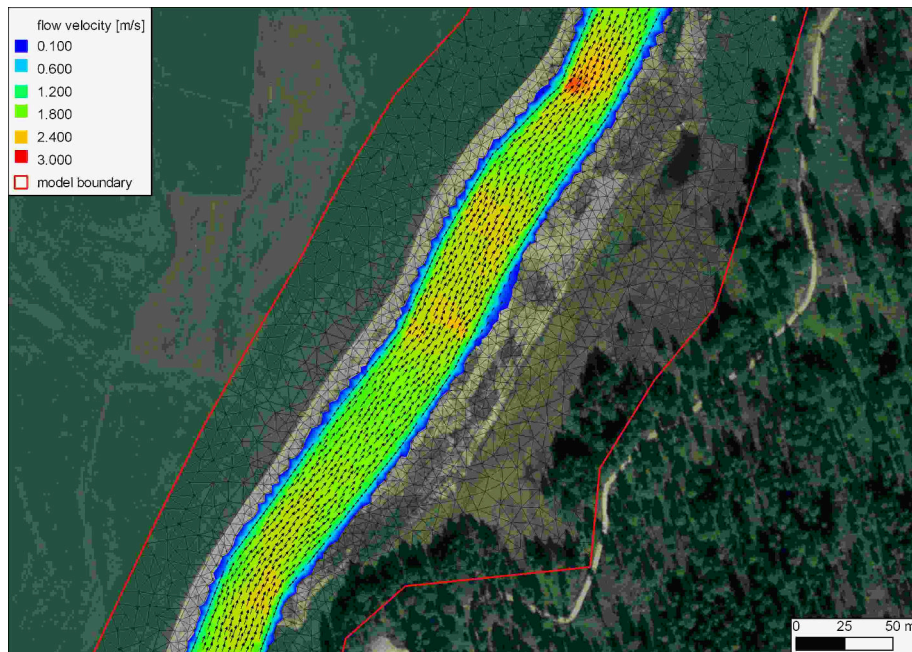


Figure 4.9 Flow velocity and velocity vectors at the steady state of the model.

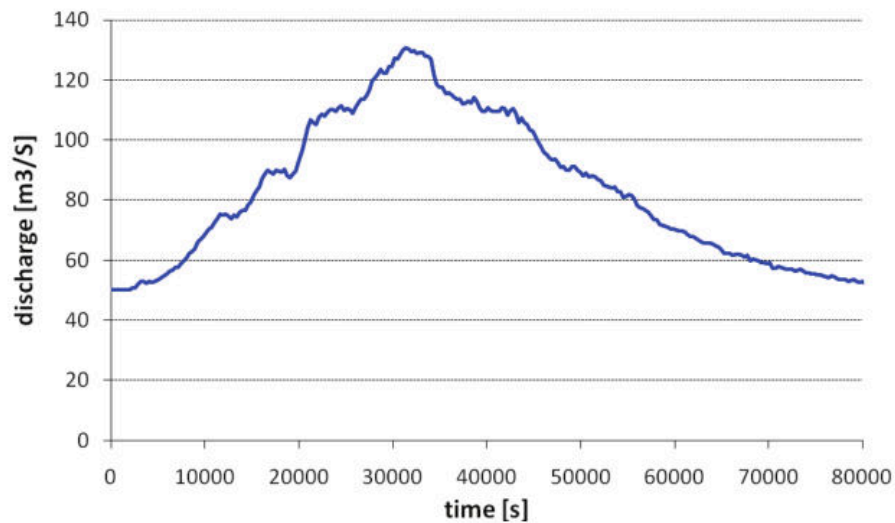


Figure 4.10 Hydrograph of the flood event of July 2004.

```

}

TIMESTEP
{
  start_time = 0.0
  total_run_time = 3000
  CFL = 0.95
  minimum_time_step = 0.0001
}

```

Furthermore the inflow hydrograph of the flood has to be defined and assigned to the upper boundary condition. The hydrograph of the flood shown in Figure 4.10 is saved in the text file *Inflow_transient.txt* with contents as depicted in Figure 4.11. Be aware that the final time defined in this file has to be the same or larger than the computation time. The upper BOUNDARY-block changes to:

```

BOUNDARY
{
  type = hydrograph
  string_name = Inflow
  file = Inflow_transient.txt
  slope = 10.0 // [per mill]
}

```

In the TIMESTEP-block the total run time of the simulation is increased to 84'000 seconds in order to capture the whole flood event:

```

TIMESTEP
{
  ...
  total_run_time = 84000
}

```

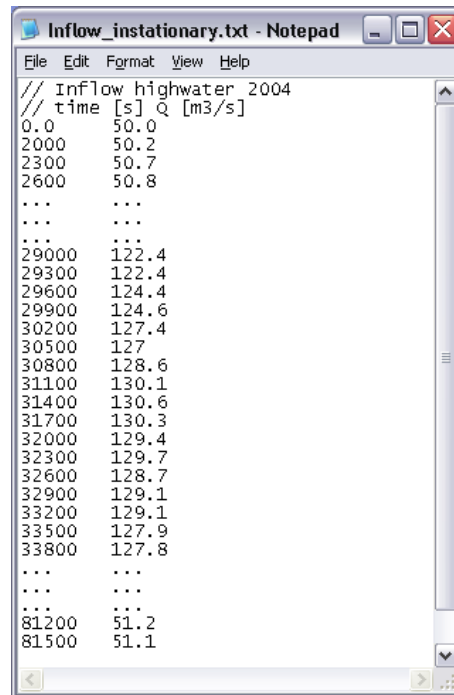


Figure 4.11 Inflow hydrograph stored in the file *Inflow_transient.txt*. Note that the points (...) are just illustrative in order to show the first and last line of the file.

...
}

Last but not least, the OUPUT-block has to be adjusted to the needs of the simulation. For the unsteady simulation, the output time step for the SPECIAL_OUTPUT format sms might be reduced in order to capture the maximum values of the output variables .

```

OUTPUT
{
  output_time_step = 2000
  console_time_step = 500
  SPECIAL_OUTPUT
  {
    format = sms
    type = node_centered
    values = ( depth velocity wse)
    output_time_step = 500
  }
  SPECIAL_OUTPUT
  {
    type = balance
    balance_values = ( timestep )
    output_time_step = 1000
  }
  SPECIAL_OUTPUT
  {

```

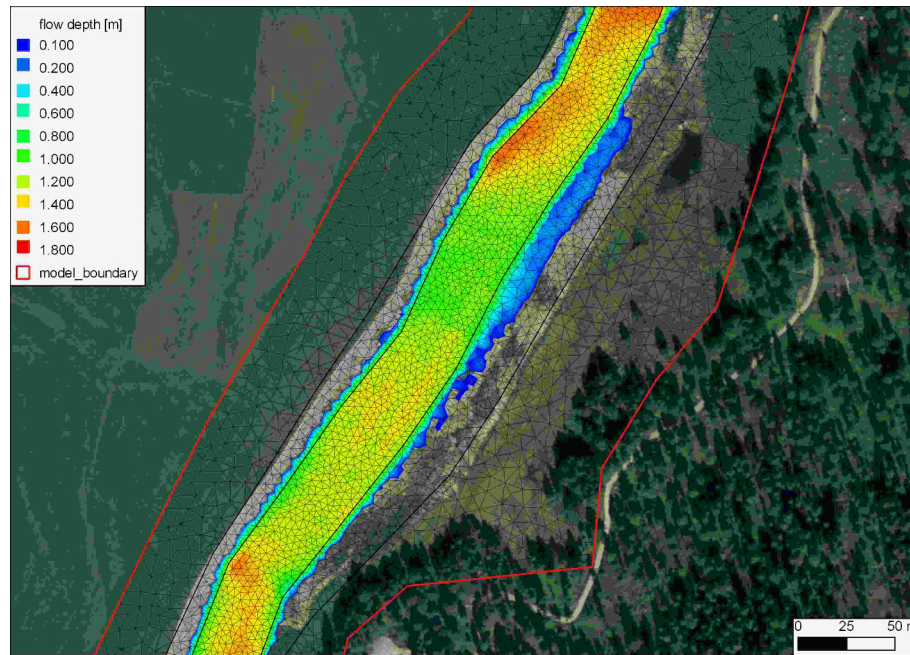



Figure 4.12 Maximal flow depth of the unsteady flow simulation observed at the flood peak after a run time of about 9 hours.

```

    type = boundary_history
    boundary_values = ( Q )
    history_one_file = yes
    output_time_step = 400
  }
}
```

Open the model file from the BASEMENT GUI (File → Open Scenario Directory → select the folder that contains the configuration files). Run the simulation with the *Run* button of the BASEMENT window. The maxima values of the flow depths and flow velocity vectors can be visualized using QGIS as shown in Figure 4.12 and Figure 4.13.

4.1.3.7.4 Calibration of the hydraulic model

The hydraulic model can be calibrated for example based on flood level marks by comparing the modelled water surface elevations with the flood level marks. Usually the calibration parameter is the bed roughness introduced with the Strickler value. The calibration procedure may need several adjustments and is an iterative process. The demonstration of the calibration is not part of this tutorial. It should be mentioned that it is important to have a calibrated hydraulic model either for further hydraulic modelling or for morphological modelling in a further step.

4.1.4 Morphological simulation with single-grain bed load transport

The MORPHOLOGY-block is not compulsory. If this block is not defined the simulation is purely hydraulic. The command file of the unsteady hydraulic simulation has to be completed for the single-grain bed load transport as shown in this section.

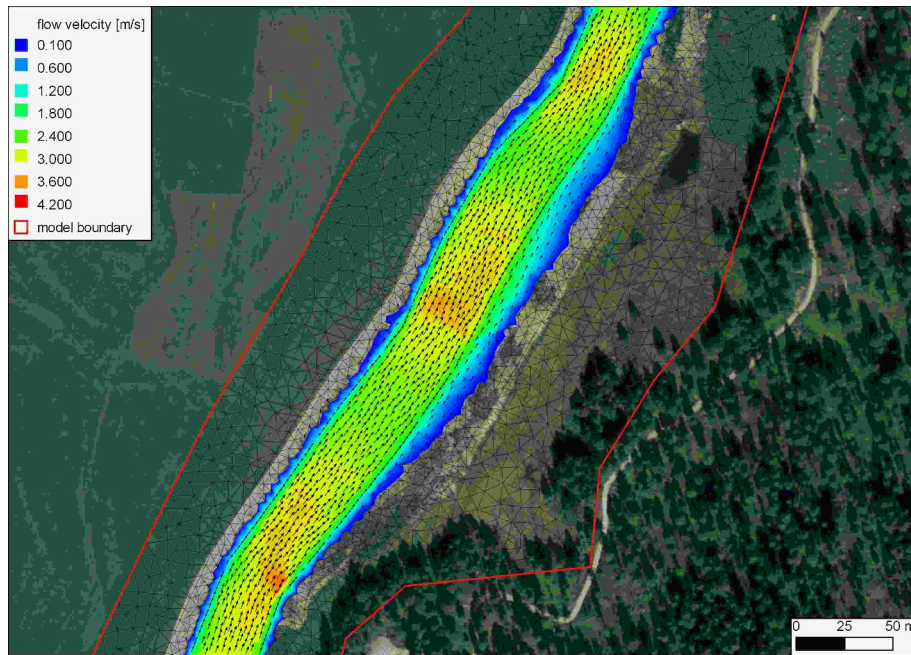


Figure 4.13 Maximal flow velocity and velocity vectors of the unsteady flow simulation observed at the flood peak after a run time of about 9 hours.

The morphological simulation is based on the flood event in July 2004. Therefore a single-grain bed load transport is added to the unsteady hydraulic simulation in Section 4.1.3.7.3. In the HYDRAULIC-block a small change has to be done in order to define the boundary string *Inflow_sed* for the bed load inflow. Thus a new STRINGDEF-block is added within the GEOMETRY-block as follows:

```

GEOMETRY
{
  ...
  STRINGDEF
  {
    name = Inflow_sed
    node_ids = (3 194 192 210 4)
    upstream_direction = left
  }
  ...
}

```

4.1.4.1 Define the morphological information

The necessary information for the morphological part of the simulation is defined in the *MORPHOLOGY*-block.

```

MORPHOLOGY
{
  PARAMETER
  {...}
}

```

```

    INITIAL
    {...}
    BEDMATERIAL
    {...}
    BEDLOAD
    {...}
    GRAVITATIONAL_TRANSPORT
    {...}
}

```

4.1.4.1.1 Morphological parameters

In the PARAMETER-block important parameters for the morphological simulation are defined. The bed load control volume is chosen to be constant with a thickness of 0.1 m.

```

PARAMETER
{
    porosity = 40                // [%]
    density = 2650               // [kg/m3]
    control_volume_type = constant
    control_volume_thickness = 0.1 // [m]
}

```

4.1.4.1.2 Initial conditions

The initial bed elevation is defined in most cases as the actual topography.

4.1.4.1.3 Bed material

In the BEDMATERIAL-block the grain classes, the composition, the thickness of the soil layers, the level of the fixed bed and the assignment of the soil to the mesh is defined in several sub-blocks.

```

BEDMATERIAL
{
    GRAIN_CLASS
    {...}
    MIXTURE
    {...}
    SOIL_DEF
    {...}
    FIXED_BED
    {...}
    SOIL_ASSIGNMENT
    {...}
}

```

4.1.4.1.3.1 Grain size distribution

The single-grain simulation is performed with only one grain class of a given diameter, e.g. the mean grain diameter.

```
GRAIN_CLASS
{
    diameters = ( 50 ) // [mm]
}
```

4.1.4.1.3.2 Grain mixture

Since we have only one grain size, the volume fraction is equal to 100%.

```
MIXTURE
{
    name = single_grain
    volume_fraction = ( 100 ) // [%]
}
```

4.1.4.1.3.3 Define the soil composition

The soil layers and the according sediment mixture are defined in the SOIL_DEF-block. For a single-grain simulation it is not important how many layers are defined. The negative bottom elevation defines the thickness of the layer. Below the last layer a fixed bed is assumed. If no LAYER-block is defined, then automatically a fixed bed on the surface is assumed. We use this especially for the river bed near the upper boundary condition to avoid uncontrolled erosion. Furthermore the embankments are kept fixed because the main focus is on the river bed morphology. The two soils *soil_fix_20* and *soil_fix_40* are defined to have a gradual transition from the fixed bed to the movable bed. Anyway, the river section with the roughness elements cannot be modelled accurately, because single roughness elements which are more or less fixed stones cannot be discretized within the computational mesh. They have to be modelled with an increased bed roughness instead.

```
SOIL_DEF
{
    name = soil_element_roughness
    LAYER
    {
        bottom_elevation = -0.8 // fixed bed 0.8 m below the surface
        mixture = single_grain
    }
}
SOIL_DEF
{
    name = soil_widening
    LAYER
    {
        bottom_elevation = -2.0 // fixed bed 2.0 m below the surface
```

```

    mixture = single_grain
  }
}

SOIL_DEF
{
  name = soil_alt_bars
  LAYER
  {
    bottom_elevation = -2.0    // fixed bed 2.0 m below the surface
    mixture = single_grain
  }
}
SOIL_DEF
{
  name = soil_fix_20
  LAYER
  {
    bottom_elevation = -0.2    // fixed bed 0.2 m below the surface
    mixture = single_grain
  }
}
SOIL_DEF
{
  name = soil_fix_40
  LAYER
  {
    bottom_elevation = -0.4    // fixed bed 0.4 m below the surface
    mixture = single_grain
  }
}
SOIL_DEF
{
  name = soil_fix            // fixed bed
}

```

4.1.4.1.3.4 Fixed bed elevation

There are several possibilities to define a fixed bed. In the `FIXED_BED`-block, the elevations of areas with fixed bed can be defined either with a separate mesh file containing the fixed bed elevations or with specific fixed bed elevations for some selected nodes. Furthermore a fixed bed can be implemented in the `SOIL_DEF`-block (Section 4.1.4.1.3.3). If there is no layer defined, a fixed bed will be assumed. In any case, a fixed bed is assumed below the last layer. In this tutorial the `FIXED_BED`-block is used as an example to define a fixed bed for a single node. This can be used to consider a big stone for example. A fixed node (node id 8956) is implemented by giving `zb_fix` a value smaller or equal to -100.

```
FIXED_BED
```

```
{
  type = nodes
  node_ids = ( 5486 )
  zb_fix = ( -100 )
}
```

4.1.4.1.4 Assignment of the defined soil types

The soil types defined in the SOIL_DEF-blocks (Section 4.1.4.1.3.3) are assigned to the elements of the mesh by the material index.

```
SOIL_ASSIGNMENT
{
  type = index_table
  index = ( 1 2 3 4 5 6 7 8 9 10 11 12 )
  soil = ( soil_element_roughness soil_fix soil_fix soil_widening
  soil_fix soil_widening soil_altBars soil_fix soil_fix soil_fix
  soil_fix_20 soil_fix_40 )
}
```

4.1.4.2 Bedload

In the BEDLOAD-block all needed data for bedload transport as well as boundary conditions are defined in several sub-blocks.

```
BEDLOAD
{
  PARAMETER
  {...}
  FORMULA
  {...}
  BOUNDARY
  {...}
  DIRECTION
  {...}
}
```

4.1.4.2.1 Bed load parameter

The control parameters for the bed load simulation are defined in the PARAMETER-block. Since the *limit_bedload_wetted* tag is turned off, the bed load is computed not only in completely wetted cells but in partially wetted cells as well. Averaging the bedload fluxes over the sediment cells is turned off. This option is computational less expensive and less diffusive.

```
PARAMETER
{
  limit_bedload_wetted = off
  use_cell_averaged_bedload_flux = off
}
```

4.1.4.2.2 Bed load formula

The bed load transport is computed with the Meyer-Peter and Mueller's (mpm) formula.

```

FORMULA
{
    bedload_formula = mpm
    bedload_factor = 0.4
    theta_critic_index = ( 1 2 3 4 5 6 7 8 9 10 11 12 )
    theta_critic = ( 0.04 0.04 0.04 0.04 0.04 0.04 0.04 0.04 0.04 0.04 0.04 0.04
0.04 0.04)
}

```

4.1.4.2.3 Bed load boundary condition

The bed load input is handled with a boundary condition which determines the transport capacity at the inflow cross section. The *IODown* is the only downstream boundary condition available for sediment transport at the moment. All sediment entering the last computational cell will leave the cell over the downstream boundary.

```

BOUNDARY
{
    type = transport_capacity
    string_name = Inflow_sed
    mixture = single_grain
    factor = 0.7
}

```

```

BOUNDARY
{
    type = IODown
    string_name = Outflow
}

```

4.1.4.2.4 Bedload direction

This block contains general settings dealing with the adaption of the bedload transport direction. The lateral transport caused by a *lateral_bed_slope* with respect to the main flow direction is taken into account. The *lateral_index* defines the regions of the computational mesh, where lateral transport should be considered.

```

DIRECTION
{
    lateral_transport_type = lateral_bed_slope
    lateral_transport_factor = 1.5
    lateral_index = ( 1 2 3 4 5 6 7 8 9 10 11 12 )
}

```

4.1.4.3 Gravitational transport

In the GRAVITATIONAL_TRANSPORT-block the parameters for gravitation induced transport are defined. The gravitational transport can be limited to elements which are fully wetted or can be considered for all elements. Over the material index the scope and the applied angles for the gravitational transport can be defined. Note that for soils with a fixed bed the gravitational transport is not active. In this model gravitational transport is applied only for the mesh elements with index 6. This enables river bed widening due to bank collaps on parts of the right embankment of the widening zone.

```
GRAVITATIONAL_TRANSPORT
{
  index = (6)
  angle_failure_dry = (30)
  angle_failure_wetted = (15)
  angle_failure_deposited = (10)
  gravity_transport_on_cells = partially wetted
  angle_wetted_criterion = partially_wetted
}
```

4.1.4.4 Define the output

The desired output of the simulation has to be defined explicitly in the OUTPUT-block. The *output_time_step* defines the time steps of the results. The *console_time_step* defines the time step to appear in the BASEMENT window during simulation. Specific output modes have to be defined in the repeatable SPECIAL_OUTPUT-blocks. Inside this block the *output_time_step* defines the output time step for this particular output. A detailed overview of all possible output types, values, format types and more can be found in the [Input block structure](#).

```
OUTPUT
{
  console_time_step = 1000
  SPECIAL_OUTPUT
  {
    type = BASEviz
    output_time_step = 5
  }
  SPECIAL_OUTPUT
  {
    format = sms
    type = node_centered
    values = ( depth wse velocity deltaz z_node )
    output_time_step = 1000
  }
  SPECIAL_OUTPUT
  {
    type = balance
    balance_values = (sediment timestep)
```



```

        output_time_step = 1000
    }
    SPECIAL_OUTPUT
    {
        type = element_history
        element_ids = (3517 10729)
        output_time_step = 1000
        element_values = (depth velocity z_element)
    }
    SPECIAL_OUTPUT
    {
        type = boundary_history
        boundary_values = ( Q Qsed )
        history_one_file = yes
        output_time_step = 1000
    }
}

```

4.1.4.5 Perform morphological simulation with single-grain bed load transport

Open the *model.json* file via the menu of the BASEMENT GUI (File → Open Scenario Directory → select the folder that contains the configuration files). Run the simulation with the Run button of the BASEMENT window.

The output files with the ending *.sol* can be visualized using QGIS . The bed elevation after the flood event is shown in Figure 4.14. Two cross sections are defined and the bed elevation along these cross sections before and after the flood event is compared using the QGIS plugin *Profile tool* (Figure 4.15). The morphological changes (*deltaz*) due to the flood event are shown in Figure 4.16 .The position of the simulated depositions matches quite well with the position of the banks observed in the aerial image.

4.1.5 Morphological simulation with multi-grain bed load transport

In order to avoid duplication, only the modifications of the *model.json* file, compared to the single-grain simulation, are detailed here. A modification that is not considered here is the possibility to use the grain size distribution to determine the bed friction in the FRICTION-Block. To simplify this tutorial, the friction is defined with the Strickler value. Generally it is suggested to try both options and to choose the most suitable for your model purpose.

4.1.5.1 Morphological parameters

In the PARAMETER-block important parameters for the morphological simulation are defined. In multi-grain simulations the thickness of the bed load control volume is an important calibration parameter. This parameter influences significantly the grain sorting process.

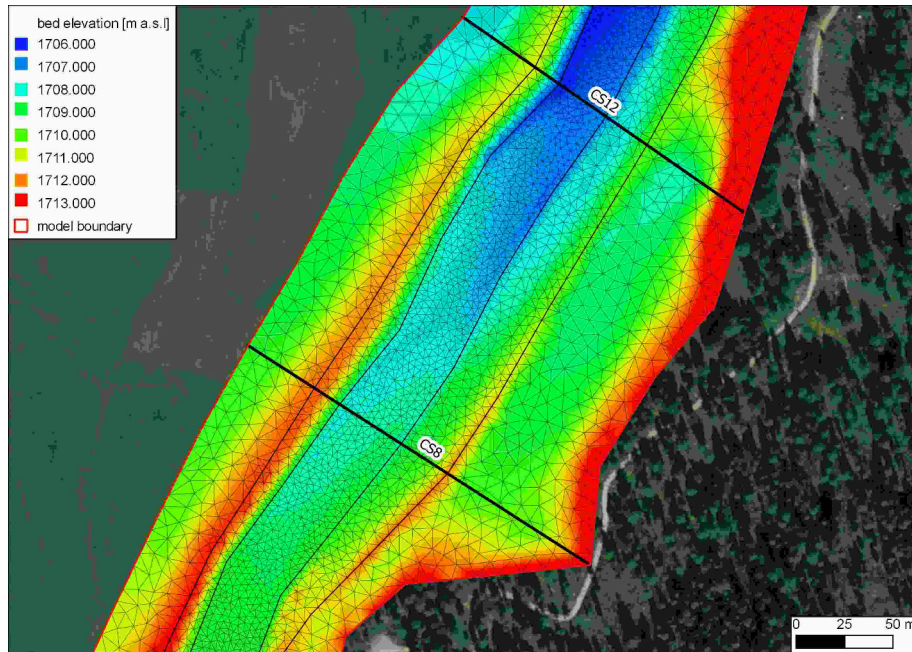


Figure 4.14 Modeled bed elevation (z_{bed}) and two cross sections defined in the widening part.

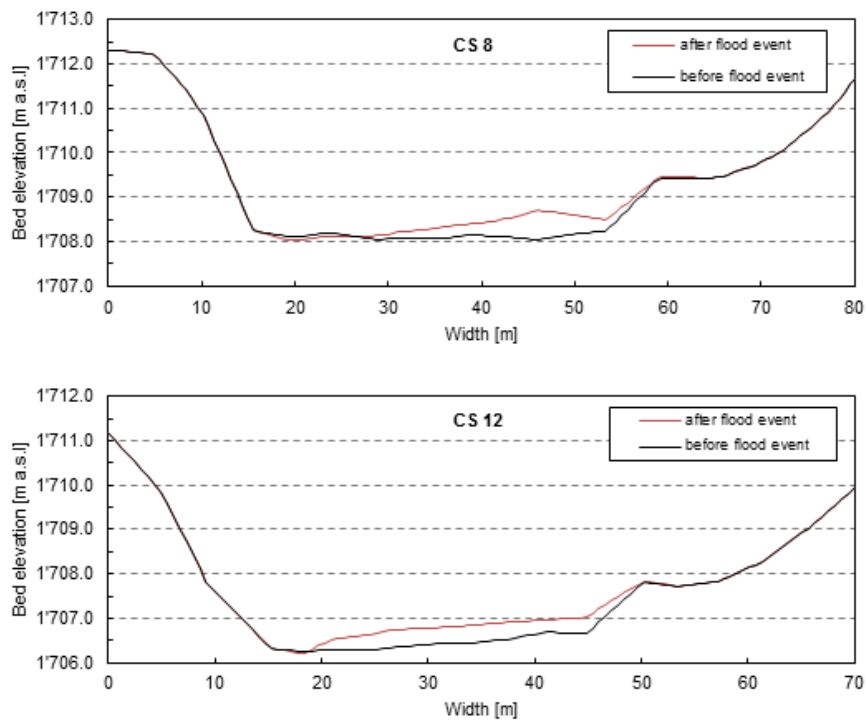


Figure 4.15 Comparison of the river bed before and after the flood event in cross-section CS 8 and CS 12.

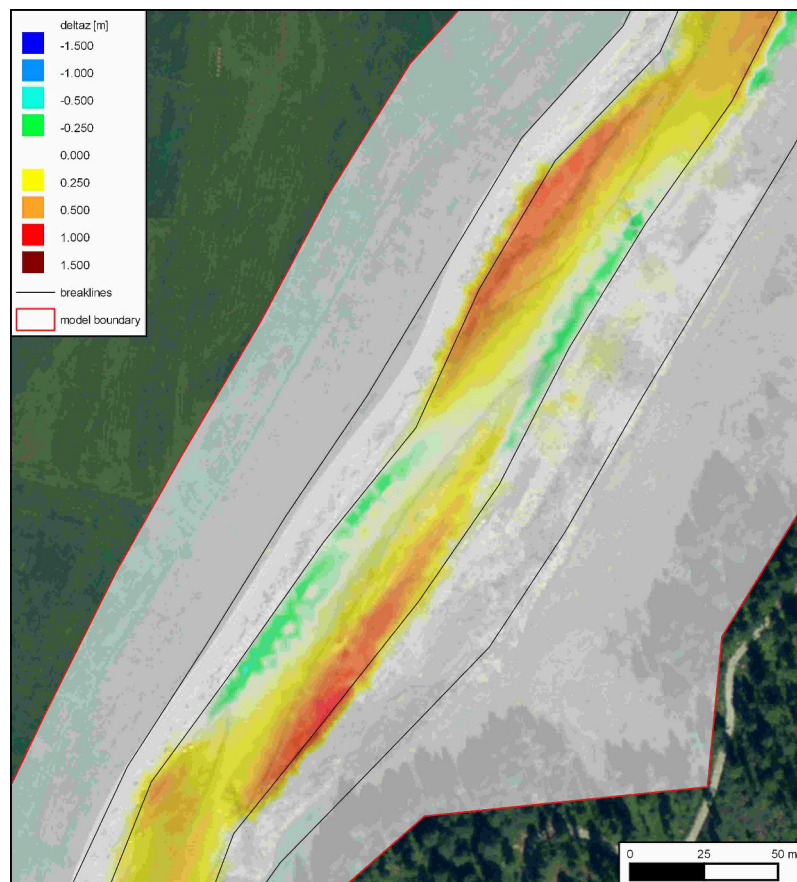


Figure 4.16 Changes of the morphology (deltaz) due to the flood event with the single-grain model. The red colour range represents deposition and the green/blue colour range shows erosion.

```

PARAMETER
{
    ...
    control_volume_type = constant
    control_volume_thickness = ( 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 ) // [m]
    control_volume_thickness_index = (1 2 3 4 5 6 7 8 9 10 11 12)
}

```

4.1.5.2 Grain size distribution

The grain size distribution is discretized with six grain classes. They have to be defined in ascending order from the smallest to the largest grain.

```

GRAIN_CLASS
{
    diameters = ( 1 5 15 44 82 150 ) // [mm]
}

```

4.1.5.3 Grain mixture

In the MIXTURE-block the volume fraction of the different mixtures are defined. The three river sections are considered with different sediment mixtures. Furthermore a mixture for the inflow is defined.

```

MIXTURE
{
    name = mixture_inflow
    volume_fraction = ( 15 15 23 27 10 10 )
}
MIXTURE
{
    name = mixture_element_roughness
    volume_fraction = ( 17 11 14 27 14 17 )
}
MIXTURE
{
    name = mixture_widening
    volume_fraction = ( 21 13 16 25 11 14 )
}
MIXTURE
{
    name = mixture_alt_bars
    volume_fraction = ( 27 14 14 20 14 11 )
}

```

4.1.5.4 Define the soil composition

The soil layers with the corresponding sediment mixture are defined in the SOIL_DEF-block. The soil can be defined with several layers of different material, but to keep it simple we assume a single layer. The negative bottom elevation defines the thickness of the layer. Below the last layer a fixed bed is assumed. If no LAYER-block is defined then automatically a fixed bed on the surface is assumed. We use this especially for the river bed near the upper boundary condition to avoid uncontrolled erosion. Furthermore the embankments are kept fix because the main focus is on set on the river bed morphology. The two soils *soil_fix_20* and *soil_fix_40* are defined to have a gradual transition from the fixed bed to the movable bed.

```
SOIL_DEF
{
  name = soil_element_roughness
  LAYER
  {
    bottom_elevation = -0.8    // fixed bed 0.8 m below the surface
    mixture = mixture_element_roughness
  }
}
SOIL_DEF
{
  name = soil_widening
  LAYER
  {
    bottom_elevation = -2.0    // fixed bed 2.0 m below the surface
    mixture = mixture_widening
  }
}
SOIL_DEF
{
  name = soil_alt_bars
  LAYER
  {
    bottom_elevation = -2.0    // fixed bed 2.0 m below the surface
    mixture = mixture_alt_bars
  }
}
SOIL_DEF
{
  name = soil_fix    // fixed bed
}
SOIL_DEF
{
  name = soil_fix_20
  LAYER
  {
    bottom_elevation = -0.2    // fixed bed 0.2 m below the surface
```

```
    mixture = mixture_element_roughness
  }
}

SOIL_DEF
{
  name = soil_fix_40
  LAYER
  {
    bottom_elevation = -0.4    // fixed bed 0.4 m below the surface
    mixture = mixture_element_roughness
  }
}
```

4.1.5.5 Bed load boundary condition

The bed load input is regulated with a boundary condition which determines the transport capacity at the cross section defined. The factor for the bed load at the boundary is an important calibration parameter and depends on the transport formula. Therefore this factor is different for single-grain and multi-grain simulations. The outflow boundary is handled as in the single-grain simulation.

```
BOUNDARY
{
  type = transport_capacity
  string_name = Inflow_sed
  mixture = mixture_inflow
  factor = 0.7
}
BOUNDARY
{
  type = IODown
  string_name = Outflow
}
```

4.1.5.6 Bed load formula

For the sediment transport computation different bed load transport formulas are available. In this tutorial the formula of Meyer-Peter and Mueller for multiple grain classes is chosen. It is suggested to try different sediment transport formulas.

```
FORMULA
{
  bedload_formula = mpm_multi
  bedload_factor = 0.5
}
```

4.1.5.7 Define the output

The desired output of the simulation has to be defined explicitly in the OUTPUT-block. The specific output is defined in the repeatable SPECIAL_OUTPUT-blocks. For the multi-grain simulation some additional output may be interesting such as for example the grain size distribution in selected nodes. This way grain sorting effects can be observed. A detailed overview of all possible output types, values, format types and more is given in help buttons in the Command File Editor of BASEMENT.

```
OUTPUT
{
  console_time_step = 1000
  SPECIAL_OUTPUT
  {
    format = sms
    type = node_centered
    values = ( depth deltax z_node )
    output_time_step = 1000
  }
  SPECIAL_OUTPUT
  {
    type = node_history
    node_values = ( grain_size )
    node_ids = (1138 3235)
    history_one_file = yes
    output_time_step = 4000
  }
  SPECIAL_OUTPUT
  {
    type = boundary_history
    boundary_values = ( Q Qsed )
    history_one_file = yes
    output_time_step = 1000
  }
}
```

4.1.6 Perform morphological simulation with multi-grain bed load transport

Open the *model.json* file via the menu of the BASEMENT GUI (File → Open Scenario Directory → select the folder that contains the configuration files). Run the simulation with the Run button of the BASEMENT window.

The morphological changes *deltax* are shown in Figure 4.17. Here the multi-grain model is not compared quantitatively with the single-grain model. Nevertheless the qualitative comparison is indicating a quite similar behaviour (Figure 4.16 and Figure 4.17). At this state much more details could be investigated such as the grain class fractions, the hiding-and-exposure function (*hiding_exponent*), the amount of grain classes etc. Further important calibration parameters are the critical dimensional shear stress, the bed load factor and the bed load inflow controlled with the bed load factor at the boundary.

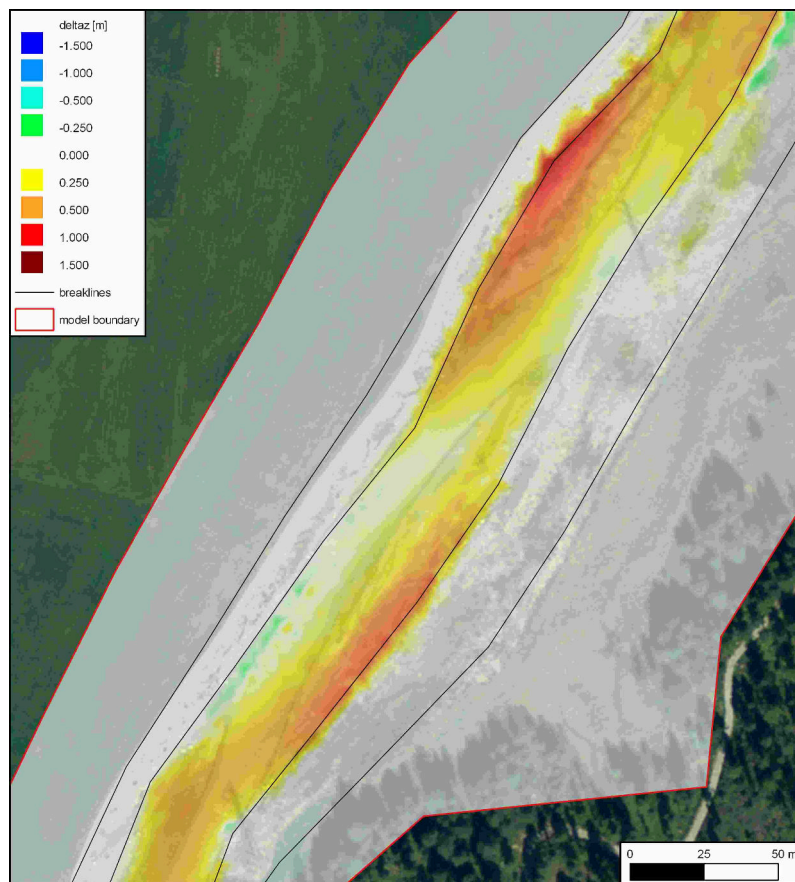


Figure 4.17 Changes of the morphology (deltaz) due to the flood event with the multi-grain model. The red colour range represents deposition and the green/blue colour range shows erosion

Hydrodynamics and sediment transport at the river Thur with BASEMD (1D)

5.1 Hydrodynamics and sediment transport at the river Thur with BASEMD (1D)

5.1.1 Introduction

This tutorial gives an introduction to the capabilities of the 1D modelling module BASEchain of BASEMENT. It provides a step-by-step guidance on how build up a model for BASEchain.

5.1.1.1 General description

This Tutorial describes the necessary steps for the simulation of hydrodynamics and bed load in a specific section of the river Thur. In the considered section a river widening has been realized during the last years. It's located in Altikon and illustrated in the figure below. The flow direction is from right to left. The bed modification over a year including an important flood will be simulated.

5.1.1.2 Used features

In this tutorial the following points will be treated:

- Preparation of the needed input files;
- Simulation of a steady flow to use for the following simulations;
- Use of composite cross sections;



Figure 5.1 View of the simulated river section

- Simulation of bed load with formula of Meyer-Peter Müller;
- Use of dry initial condition;
- Use of a file to define the initial conditions;
- Use of the following boundary conditions:
 - Inflow hydrograph
 - Inflow of sediment in/out
 - Outflow h-q relation
 - Outflow of sediment in/out
- Representation of the results.

5.1.1.3 Purpose

In the year 2005, intensive rainfall led to a large flood event. The aim of the simulations in this tutorial is to study, which influence this flood had on the channel geometry of the river Thur.

5.1.2 Setting up the topography file

5.1.2.1 Cross sections

The data of the topography are available in the form of cross section measurements, where each measured point is given by its x, y and z coordinates. This is an extract from the raw data:

x	y	z
698578.504	272450.223	376.841
698578.494	272446.999	374.991
698578.32	272444.286	373.748
698577.929	272441.889	373.229
698578.081	272439.244	372.207
698578.533	272437.229	371.544

x	y	z
698578.56	272434.366	370.869
698578.612	272431.522	370.766
698578.86	272429.064	370.401
698579.201	272426.526	370.388
698579.323	272424.56	370.617
698578.937	272422.91	370.341
698579.208	272421.645	370.436
...		

These data have to be separated into groups belonging to one cross section and then transformed in a way to have a $z(y)$ relation, where the smallest y is the extreme point on the left river side.

Example:

y	z
0	376.264
1.455081097	376.327
4.349044033	377.804
5.134094857	378.133
6.803278107	378.238
8.241452785	378.227
9.123103693	377.965
10.23346129	377.395
11.41786604	376.664
12.57016281	376.221
14.53240603	376.21
16.34176138	376.215
17.09961488	375.99
18.74552432	375.296
...	

Additionally the distance from the upstream end of the channel (first cross section) has to be determined for each cross section.

The obtained geometry points can be introduced in the topography by copy pasting it directly into the node coordinates field in the grid file editor (Figure 5.2). The editor will translate the column wise data into the proper syntax. Another, more efficient way is to use the python scripts available on <http://www.basement.ethz.ch> to transform topography data in excel format into the BASEMENT format in a first step.

The minimum information we have to provide for each cross section besides the node coordinates are the cross section name and the distance coordinate measured from upstream to downstream in km.

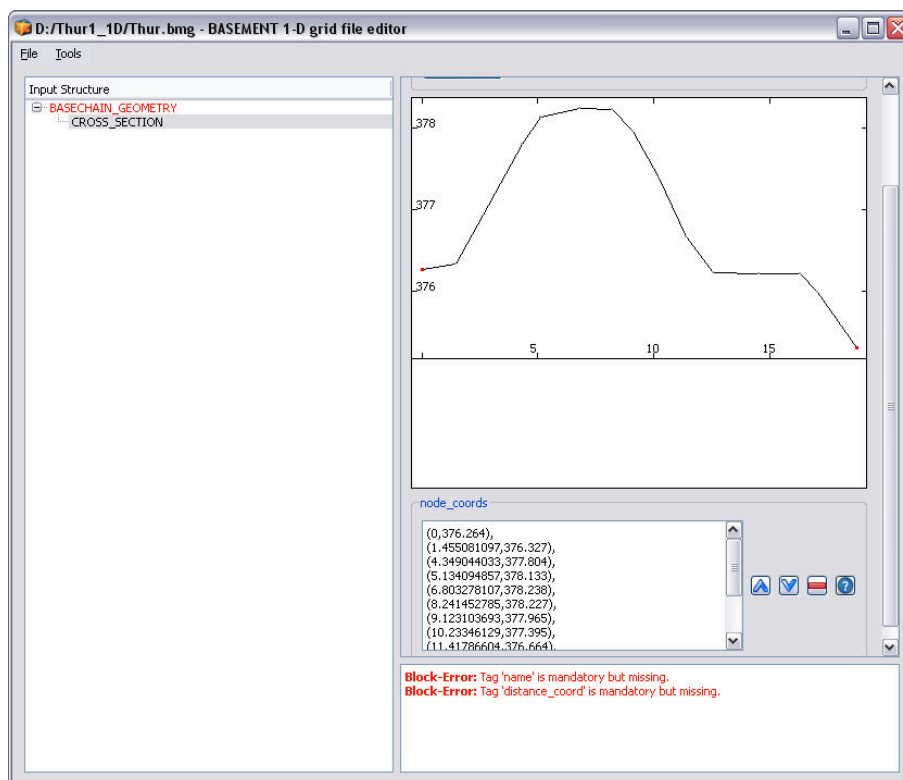


Figure 5.2 Cross section points inserted in the topography editor

5.1.2.2 Definition of different cross section zones

To reduce some drawbacks of the 1-D simulation, in the present case it is useful to define a main channel and flood plains, as well as the bed bottom which is limiting the bed load transport.

In Figure 5.3, the flood plains are given by the part of the cross section not defined as the main channel. The soils by their indexes. The keys 2 or 1 refer to the type of soil which is defined later in the command file. Here we use only one soil for the whole bottom, but it is also possible to add several soils of the same type or of different types as shown in Figure 5.3. Further, different friction values can be defined for different parts of the cross section. The active range should span from the left to the right dike. Points outside the active range are simply ignored (Figure 5.4).

The graphical view of the cross section data helps to identify the correct point and set the ranges to the correct lateral node coordinates. For convenience, one can switch into the text editor mode of the input file by choosing from the *Tools* Menu the option *Edit Raw*.

5.1.2.3 Friction values

For the friction determination the Strickler approach is used. This is declared in the command file by setting the *type* in the `FRICITION` block within the `HYDRAULICS` section to *strickler*. In this case, Strickler *k*-values have to be defined for the different regions. The banks of the main channel are partially covered with small bushes. The flood plains are covered with grass, stones and sand, but there are also zones with trees.

The following k_{str} values are used:

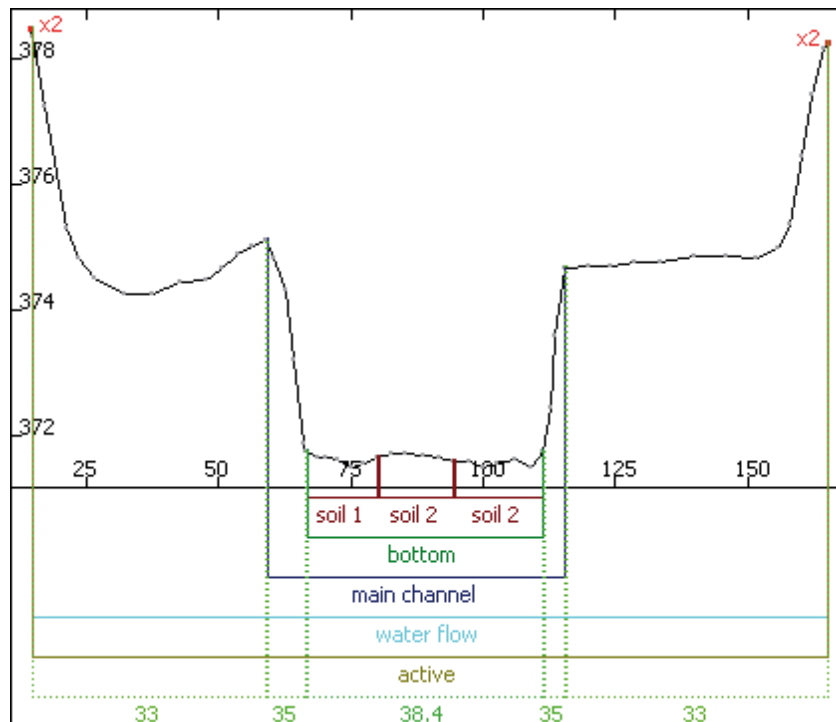


Figure 5.3 Delimitation of cross section zones

name
CS2

distance_coord
0.049821

main_channel_range
60.523 117.128

friction_coefficients
33, 35, 38.4, 35, 33

friction_ranges
(10.233,60.523),
(60.523,67.274),
(67.274,109.984),
(109.984,117.128),
(117.128,167.344)

node_coords
(10.233,377.395),
(11.417,376.664),
(12.570,376.221),
(14.532,376.21),
(16.341,376.215),
(17.099,375.99),
(18.745,375.296)

Figure 5.4 Definition of cross section properties

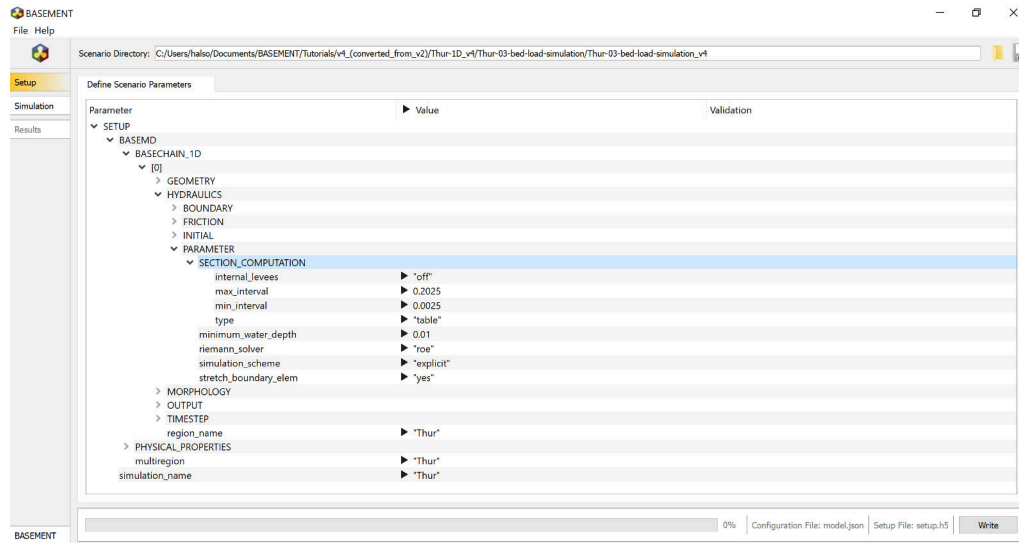


Figure 5.5 Definition of table values

- Banks of main channel: $k_{Str} = 35 \text{ [m}^{1/3}/\text{s]}$
- Flood plains: $k_{Str} = 33 \text{ [m}^{1/3}/\text{s]}$.

For the bed bottom the following transformation, based on the grain characteristics of the sediment is used ($d_{90} = 5 \text{ cm}$):

$$\text{Bottom: } k_{Str} = \frac{23}{\sqrt[6]{d_{90}}} = 38 \text{ [m}^{1/3}/\text{s]}.$$

In BASEMENT internally, the cross section is represented by slices, defined by the segment between two nodes. Each slice has its own properties. Therefore we have to provide so called ranges to assign the friction values to the respective slices. The ranges can be defined either referring to node coordinates (note that you have to match the coordinates exactly) or by referring to slice indexes, starting at index 1 from left to right.

5.1.2.4 Computation of water surface elevation

As it is much quicker, the use of tables for the computation of the water surface elevation and other hydraulic variables is chosen for this example. In the case of tables, all properties are pre-computed for a given set of points and only updated in case of a non-negligible change of the soil. This is accomplished using the block `SECTION_COMPUTATION` in `PARAMETER`. As all variables are calculated for several water surface elevations, the maximum and minimum intervals between the different levels have to be set accordingly. The default spacing is given by `max_interval - min_interval`. Whenever the bed changes, the table is updated accordingly.

5.1.2.5 Characterization of the sediments

Two types of ground will be defined:

1. the ground is not erodible;

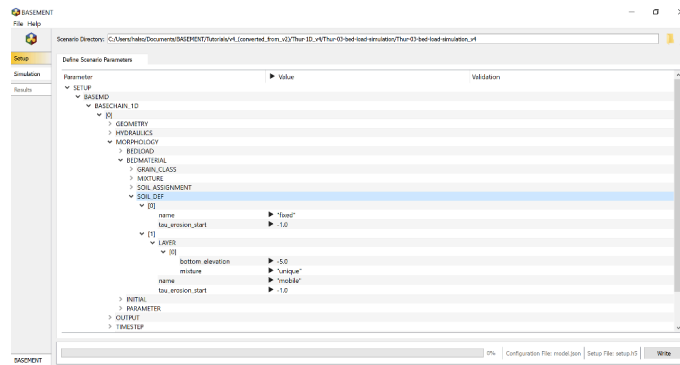


Figure 5.6 Declaration of soil types

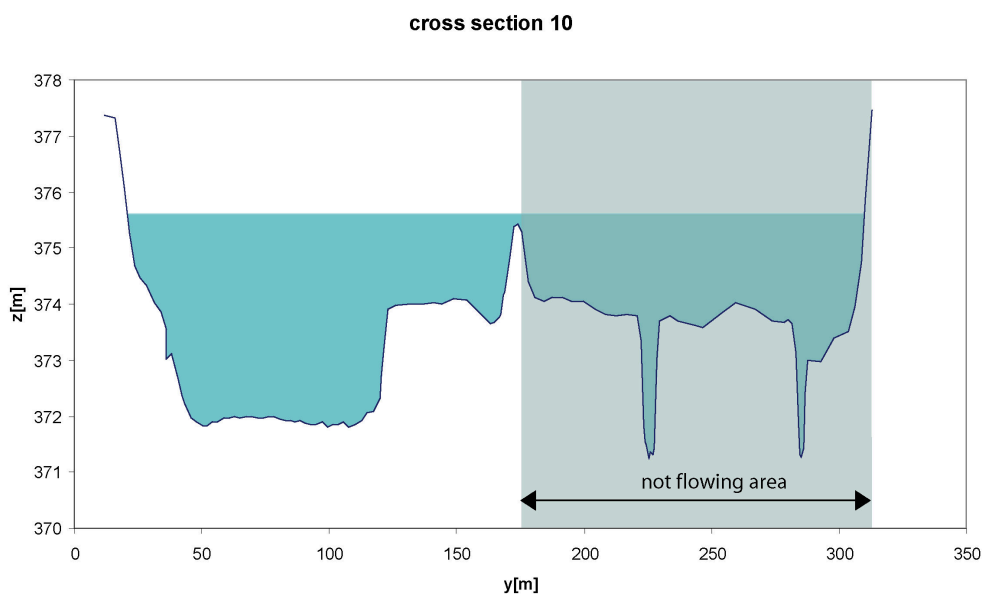


Figure 5.7 Definition of non flowing areas

- the ground is composed by sediments with a mean diameter of 2.5 cm.

In the topography file the codes of the different types are assigned to different cross sections. A code can be set in the cross sections by creating a new sub block SOIL_DEF, where the index is assigned to the respective soil index in the command file and the span of the soil is defined via the range it extends.

5.1.2.6 Define flowing zones

The 1-D model considers the flow velocity to be the same over the whole width of the cross section: This is obviously not true, especially for cross sections where important zones are behind a sort of dike, like it occurs very often at the Thur. This effect has an important influence on the bed load transport. For this reason, regions where the water does not flow are declared using the `water_flow_range` tag. The next figure shows an example of a cross section with the different zones. Of course, we only mention this here. The tutorial topology already contains the required ranges.

5.1.3 Setting up the model file

5.1.3.1 Setup

The first configuration file is called “model.json”. Open the model file from the BASEMENT GUI (File → Open Scenario Directory → select the folder that contains the configuration files).

The first step is to define a project by its name:

```
SETUP
{
  "simulation_name": "Thur"
}
```

5.1.3.2 Domain

A domain is defined by including all relevant parameters for this computation. The first parameter is the name of the computation region.

```
BASEMD
{
  multiregion = Thur
  [...]
}
```

5.1.3.3 Define the physical properties

The Physical properties normally do not change from one project to another.

```
PHYSICAL_PROPERTIES
{
  gravity = 9.81
  viscosity = 0.000001004
  rho_fluid = 1000
}
```

5.1.3.4 One dimensional simulation

The next step is to declare a *BASECHAIN_1D* block. This will make the program execute a 1-D simulation. The name of the computational region is given here.

```
BASECHAIN_1D
{
  region_name = Thur
}
```


5.1.3.4.1 Define the geometry

The next block defines in which file the topography is stored and the type of geometry file used. The cross section names are listed from upstream to downstream.

```
GEOMETRY
{
    type = basement
    file = ThurTopo.bmg
    cross_section_order = ( CS1 CS2 CS3...CS54 CS55 )
}
```

5.1.3.4.2 Define hydraulic information

All information concerning the hydraulic simulation is declared in the block *HYDRAULICS*.

```
HYDRAULICS
{
    [...]
}
```

5.1.3.4.2.1 Define the upper boundary condition

The upper boundary condition is defined by a hydrograph, which is stored in a separate file. Indications are given about the precision required between the discharge corresponding to the iteratively determined area and the given discharge, as well as the maximum number of iterations allowed to reach this precision. The slope of the first cross section must be given in per mil (the last 3 values are used only in case of supercritical flow).

Create a hydrograph file named *ThurSteadyHydrograph.txt*:

```
// T    Q
0       30
100000 30
```

Then, add the upper boundary block:

```
BOUNDARY
{
    type = hydrograph
    string = upstream
    file = ThurSteadyHydrograph.txt
    precision = 0.001
    number_of_iterations =100
    slope = 0.93
}
```

5.1.3.4.2.2 Define the lower boundary condition

The lower boundary is an h-q-relation which is calculated internally. Again, we have to define the boundary condition with a specific slope.

```
BOUNDARY
{
    type = hqrelation
    string = downstream
    slope = 1.5
}
```

5.1.3.4.2.3 Define initial condition

The channel is considered to be initially dry. Note that starting with a dry channel is for the depth-average equations a numerically delicate problem. So this option will require some care when we set the numerical parameters.

```
INITIAL
{
    type = dry
}
```

5.1.3.4.2.4 Define default friction values

The declaration of a default friction type and a default friction value are mandatory. The friction values are overwritten by the values declared in the topography file.

```
FRICTION
{
    type = strickler
    default_friction = 35
}
```

5.1.3.4.2.5 Declare parameters for hydraulic computation

In the *PARAMETER* block the relevant parameters for the hydraulic simulation are defined. With the flag *minimum_water_depth*, the water depth is defined for the case where the channel is considered to be dry. The simulation is using an explicit Euler scheme using a Roe solver for the Riemann problem at the edges of the control volumes.

```
PARAMETER
{
    minimum_water_depth = 0.01
    simulation_scheme = explicit
    riemann_solver = roe
}
```

5.1.3.4.3 Define time step information

For the first computation the simulation time is set to 150000 s. For a computation on a dry bed, a small initial time step should be chosen. It is used only at the very beginning, as there is no flow in the channel from which the time step could be deduced. The maximum time step should be bigger than all time steps computed during the simulation.

```
TIMESTEP
{
    total_run_time = 150000
    initial_time_step = 1.0
    maximum_time_step = 60.0
    CFL = 0.95
}
```

5.1.3.4.4 Define output

If only standard output is needed, only the time step for file printing and for console printing have to be defined. If Tecplot software is available, it is also very useful to generate a tecplot file.

```
OUTPUT
{
    output_time_step = 100
    console_time_step = 100
    SPECIAL_OUTPUT
    {
        type = tecplot_all
        output_time_step = 100
    }
}
```

5.1.4 Perform hydraulic simulations

5.1.4.1 Write the setup file

Once all the inputs have been set in the in the *model.json* file, a setup file must be written. Click *write* in the bottom right corner of the GUI. This will combine the model file with the computational mesh and external data, and store it in a binary file named *setup.h5*.

5.1.4.2 Setting up the simulation file

To set up the simulation file, switch to the *simulation* tab on the left. Select whether the simulation should be run on a single core (select *Single-threaded on CPU*) or on multiple cores (selected *Multi-threaded on CPU*). If multiple cores are selected, you must enter the *number of CPU cores*. This selection should depend on the number of threads your computer has available.

5.1.4.3 Perform steady flow simulation

Run the simulation with the *Run* button of the BASEMENT window. If the SPECIAL_OUTPUT of the type *BASEviz* is chosen press the keyboard button p to start the simulation.

5.1.4.4 Results of the steady flow simulation

When the simulation has finished, several results files will be produced in “.dat” form. *In the Basement window, you may ignore the “Results” tab on the left side. This tab is only active for 2D simulations.* remains open the main output file named “Thur_out.dat” using a text-editor such as notepad++ or gedit. Scroll to the end of the file, where the output data of the last time step is listed. Check the discharges Q which are given in the column labeled “Mass_Flux_[m³/s]”. If the discharges correspond to the steady inflow discharge of 30 m³/s at all edges between the cross sections, then the steady-state has been reached. You should also have a look at the other output values to see if there are suspicious values, which indicate an error somewhere in your model setup.

Edge_number	Mass_Flux_[m ³ /s]	Momentum_Flux_[m ⁴ /s ²]	...
0	30	21.0328	...
1	30	20.7677	...
2	30	20.8081	...
3	30	18.4419	...
4	30	16.5479	...
5	30	18.1862	...
6	30	19.3961	...
7	30	21.1823	...
8	30	21.8026	...
...			

If the steady-state has been reached, then we use this result for subsequent simulations (hot start). For this purpose you can use the file “Thur_restart_150000.dat”, which contains the results of the last time step. It is recommended to rename the restart file (e.g. “ThurInitial.dat”) in order to use it as an initial condition in further simulations.

You should also have a look at the file named “ThurTopo_tables.txt” and verify whether there have been any errors or strange values resulting from your topography file. (Of course, the grid file you have here in the tutorial should not have any errors. It’s just a good exercise to check your geometry if the program does not work as expected).

Now, take from the main output file “Thur_out.dat” the columns named “distance_[m]”, “z_talweg_[m_asl]”, “wse_[m_asl]” and “energy_head_[m_asl]” for the last time step and use them to plot a longitudinal profile (e.g. with Excel).

As it seems that there is no problem, we can proceed to the next step.

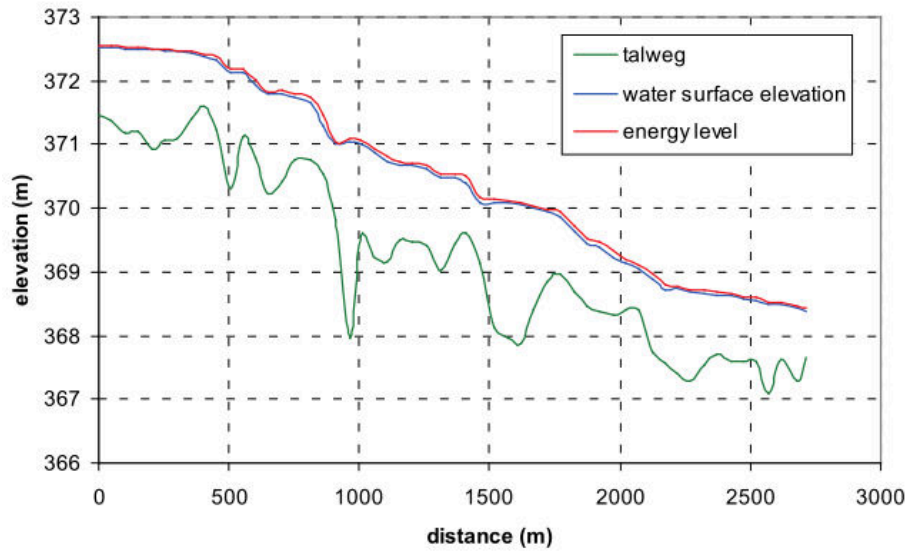


Figure 5.8 Longitudinal profile

5.1.4.5 Perform simulation of the floods (Thur2)

The zip file “Thur-02-flood-simulation.zip” contains the files for simulation of the flood. The “ThurInitial.dat” file contains the results of the last time step of the steady flow simulation, for use as a restart for this unsteady flow simulation.

The “ThurHydrograph.txt” contains the hydrograph of the flood event in 2005.

T	Q
0	30.648
3600	34.05
7200	37.305
10800	39.707
14400	41.18
18000	41.916
21600	42.275
25200	42.654
28800	43.646
32400	45.444
...	

For the upstream boundary condition, change the data file into “ThurHydrograph.txt”.

```
BOUNDARY
{
  type = hydrograph
  area = upstream
  file = ThurHydrograph.txt
  precision = 0.001
  number_of_iterations = 100
  slope = 0.93
}
```

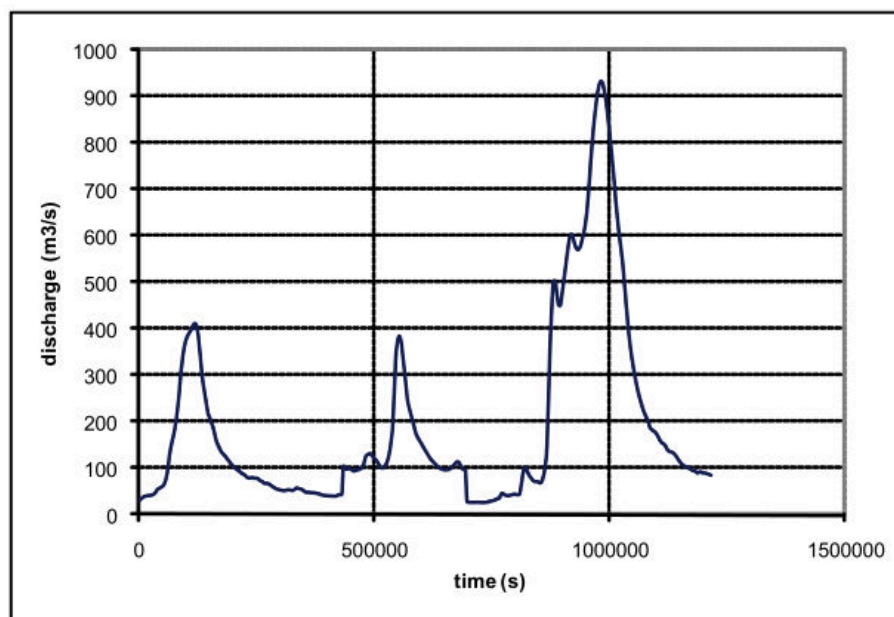


Figure 5.9 Hydrograph of the flood event of 2005

The renamed restart file “ThurInitial.dat” shall be input as an initial condition for the new simulation. Therefore we have to consider two things:

1. Choose the restart file
2. Define the start time of the new simulation.

```
INITIAL
{
  type = continue
  file = ThurInitial.dat
}
```

Because we want to restart from time zero we set the flag *start_time* in the *TIMESTEP* block to zero. Be aware that if the *tag start_time* is not defined, the time in the restart-file is used (i.e. # time: 150000 sec).

The total running time of the simulation is increased to 338 hours (1216800 sec.). So we change the settings in the *TIMESTEP* block accordingly.

```
TIMESTEP
{
  start_time = 0.0
  total_run_time = 1216800
  CFL = 0.95
  maximum_time_step = 60.0
}
```

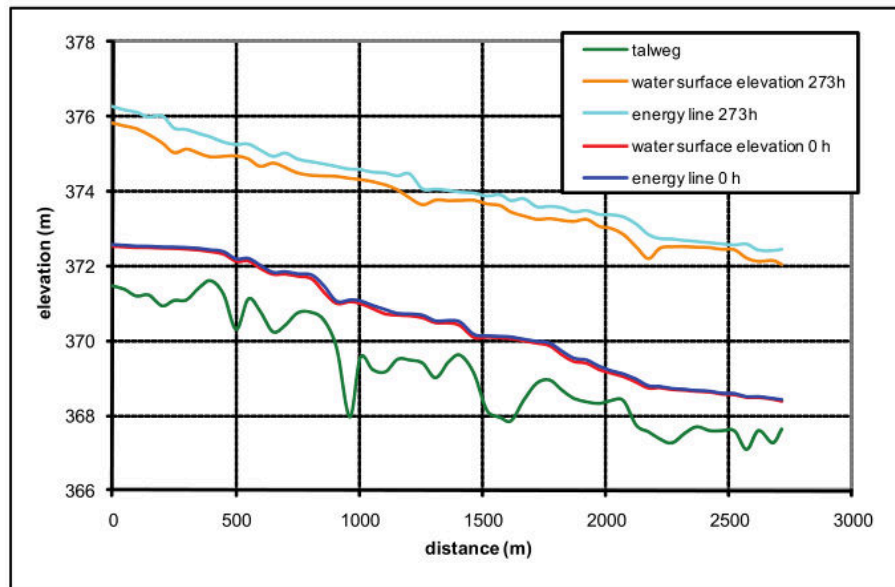


Figure 5.10 Longitudinal profile for maximum discharge

The output should be plotted less often. Therefore, we change the console and output time.

```
OUTPUT
{
  output_time_step = 1000
  console_time_step = 1000
}
```

Now we can run the simulation by writing the *setup.h5* file, and then clicking on the Run button in the simulation tab. When the simulation is finished, have a look at the main output file “Thur_out.dat”. Take the columns of “z_talweg_[m_asl]”, “wse_[m_asl]” and “energy_head_[m_asl]” for the time of maximal discharge (ca. 933 m³/s) at ca. 982.000 s (= 273 hours) and plot them over the “distance_[m]” (see Figure 5.10).

Then we plot some interesting cross sections with their water surface elevation, for the same time (see Figure 5.11). This can help to see what happens, and which parts of the cross sections are touched by the flood. For this purpose the cross section geometry and can be taken from the topography file and the water surface elevation from the main output file. Alternatively a monitoring point of type “geometry” could be used.

5.1.5 Edit the model file for bed load transport

The zip file “Thur-03-bed-load-simulation.zip” contains the files for simulation of the flood with bed load transport. Information about bed load is grouped in the block MORPHOLOGY. In the block BASECHAIN-1D, chose MORPHOLOGY and press Add Block. Go in the block MORPHOLOGY.

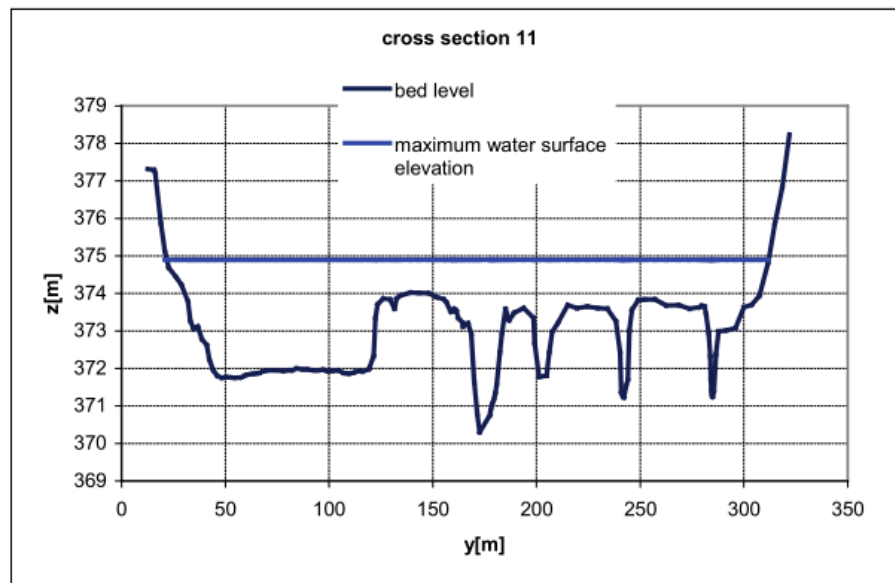


Figure 5.11 Example of resulting water surface elevation

5.1.5.1 Define the bed material

The simulation is executed with a single grain class with mean diameter = 2.5 cm. This means that you have to define one grain class and one mixture. Add a block of type `GRAIN_CLASS` and one of type `MIXTURE`. In the `GRAIN_CLASS` block add the diameter.

In the `MIXTURE` block add the name and the volume fraction.

```
BEDMATERIAL
{
  GRAIN_CLASS
  {
    diameters = ( 25 )
  }
  MIXTURE
  {
    name = unique
    volume_fraction = ( 100 )
  }
  [...]
}
```

Two types of soils are defined: one which is fixed (code 1) and one with a sub layer of 5 m thickness which is attributed to the bed bottom where bed load takes place (code 2). Add twice a block of type `SOIL_DEF`. The first one needs only a name as it has no layers of material. In the second one add a `LAYER` block. Then give the layer a bottom elevation and a mixture.

```
SOIL_DEF
```

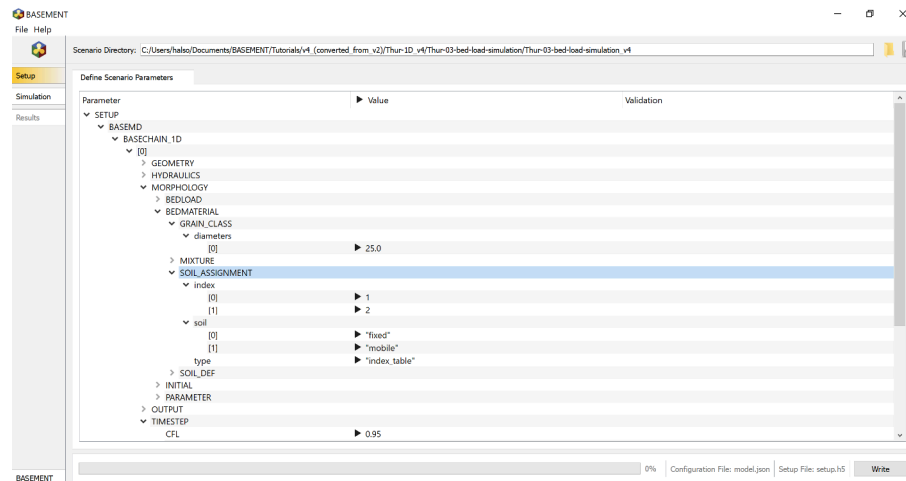



Figure 5.12 Soil assignment in the user interface

```

{
  name = fixed
}
SOIL_DEF
{
  name = mobile
  LAYER
  {
    bottom_elevation = -5.0
    mixture = unique
  }
}

```

5.1.5.2 Soil assignment

The names of the described soils have now to be assigned to the soil codes used in the topography file. Add a `SOIL_ASSIGNMENT` block and there the attributes `type`, `index` and `soil`. The first value in the index window has to correspond to the first name in the soil window etc.

5.1.5.3 Define general parameters for sediment transport

The porosity and the density of the material are standard values.

The control volume is set to a constant thickness of 20 cm (varying this parameter influences the grain sorting processes in the simulation). The tables for the hydraulic computation will be updated each time when the bed level has changed more than 5 cm.

```

PARAMETER
{
  porosity = 37
  control_volume_type = constant
  control_volume_thickness = 0.2
}

```

```
density = 2650
max_dz_table = 0.05
}
```

5.1.5.4 Define specific parameters for bed load transport

The parameter for upwind scheme is set to 1.0 and the 'velocity_area' which defines which part of the cross section area is taken for the velocity computation for bedload transport, is changed to 'main'.

```
PARAMETER
{
  upwind = 1
  velocity_area = main
}
```

5.1.5.5 Define bed load transport formula

The Meyer-Peter and Müller bed load approach will be applied without adjusting the calculated transport capacity (bedload_factor = 1.0). For the critical angle a standard value has been chosen.

```
PARAMETER
{
  bedload_formula = mpm
  bedload_factor = 1
  angle_of_repose = 30
}
```

5.1.5.6 Define boundary conditions for bed load

At the downstream boundary it is considered that the quantity of sediment which enters the last element leaves it by the boundary.

```
BOUNDARY
{
  type = IODown
  string = downstream
}
```

At the upper boundary, the observed modification of the bed level before and after the floods is very small. For this reason it can be assumed that at the upstream boundary there is as much sediment coming in, as is transported out of the first element.

```
BOUNDARY
{
  type = IOUp
  string = upstream
}
```

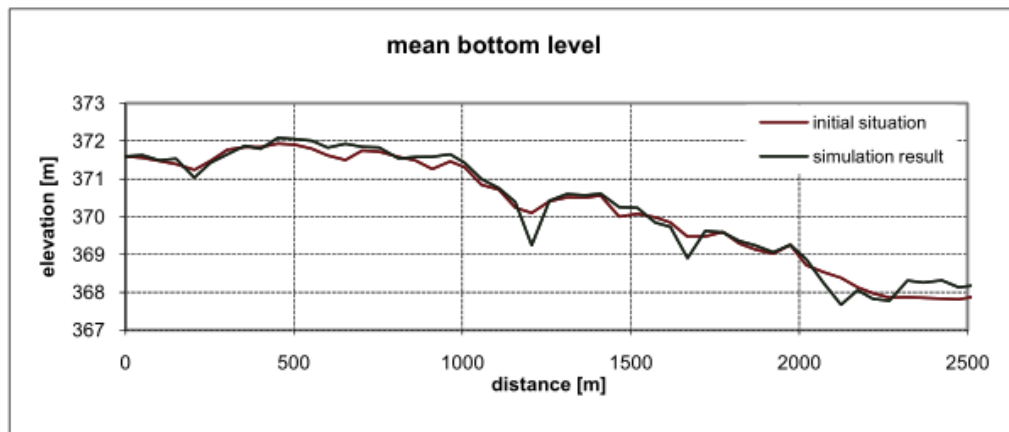


Figure 5.13 Longitudinal profile of mean bottom level

5.1.5.7 Generate a “geometry” file

To see how the geometry of cross section 14 changes during the flood add a `SPECIAL_OUTPUT` block to the `OUTPUT` block. Set the type of the `SPECIAL_OUTPUT` to monitor.

```
OUTPUT
{
  output_time_step = 1000
  console_time_step = 1000
  SPECIAL_OUTPUT
  {
    type = monitor
    output_time_step = 1000
    cross_sections = ( CS14 )
    geometry = ( time )
  }
}
```

5.1.6 Perform bed load simulation (Thur 3)

Run the simulation of the flood with bedload transport. When the simulation has finished, look at the “Thur_out.dat” file, take the columns of distance and mean bottom level of the start and end situation and make a longitudinal profile of it (Figure 5.13).

Additionally open the topology file of cross section 14 “ThurCS14_tec.dat” and plot the old and new geometry of this cross section to see the morphodynamical changes (Figure 5.14).

Obviously this is only a first run for exercise. This computation now needs calibration and validation before it can be used to make prediction of future bed evolution.

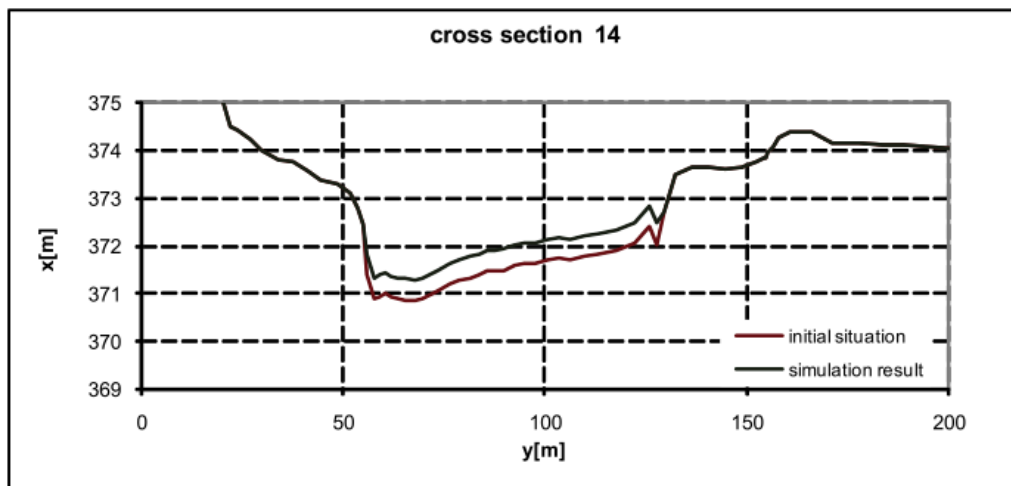


Figure 5.14 Transversal cross section profile

6

Dynamics of longitudinal bed profile due to local river widening with BASEMD (1D)

6.1 Dynamics of longitudinal bed profile due to local river widening with BASEMD (1D)

6.1.1 Introduction

In this tutorial the capabilities of *BASEchain* in modelling the evolution of the longitudinal bed profile due to a local symmetric widening is shown and discussed. The general concepts of the 1D simulation with *BASEMENT* and how to set up the configuration files are demonstrated in the previous tutorial ‘Hydrodynamics and sediment transport at the river Thur’ (Section 5.1).

6.1.1.1 General description

In the context of river engineering, local river widenings are a measure for river bed stabilisation and ecological restoration. The example river widening can be divided into three main parts (Figure 6.1): Upstream channel (L_{US}) with a certain channel width (W_C), transition zone between channel and widening (L_T), widening with length (L_W) and width (W_W) and the downstream channel (L_{DS}).

The new channel geometry due to the widening affects the hydraulic conditions in the particular river reach. This causes morphological processes that lead to a new equilibrium state of the system (Figure 6.2). Starting from an initially plane bed, erosion occurs upstream of the widening due to a temporary flow acceleration towards the widening (Scour_US). A significantly larger erosion can be observed at the downstream end of the widening (Scour_DS) as a result of the bed load deficit caused by the deposition inside

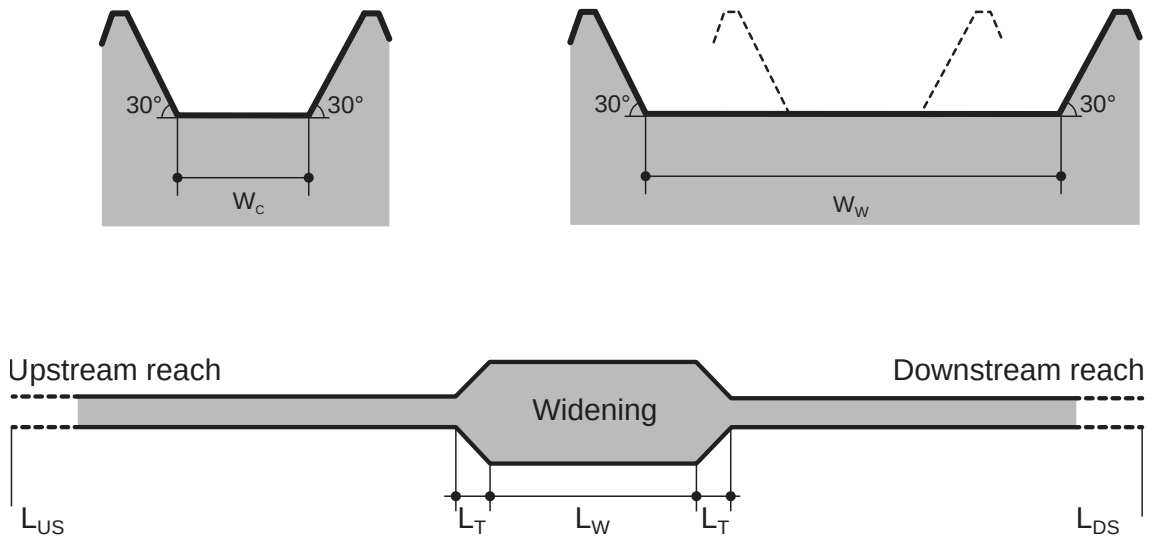


Figure 6.1 Sketch of symmetric widening used for the numerical simulations in this tutorial.

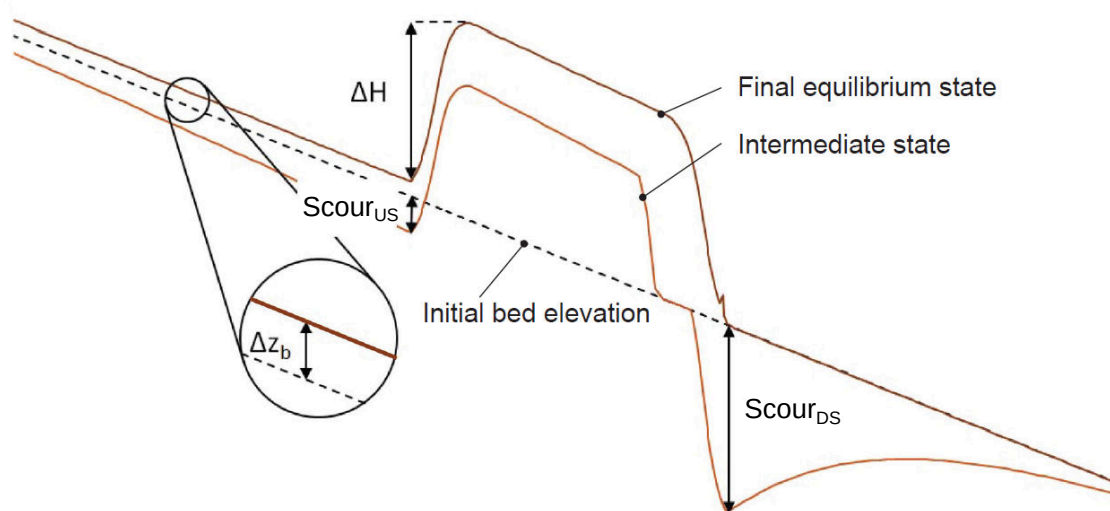


Figure 6.2 Temporal evolution of the longitudinal bed profile due to a local river widening.

the widening. This deposition (ΔH) increases and migrates in downstream direction until the end of the widening is reached. The material slowly fills up the scour whereby the downstream bed tends towards its initial state again. The longitudinal slope within the widening increases until a new equilibrium slope is reached, providing enough transport capacity to convey the entire bedload discharge downstream. The widening may induce an uplift (Δz_b) of the river bed in the upstream reach. However, before the depositions in the widening, reach equilibrium state, erosion may be observed in the upstream channel.

6.1.1.2 Purpose

Based on the general conditions of the river widening at the River Thur near Altikon (ZH, Figure 6.3), numerical 1D simulations with simplified geometry are performed in order to analyse the effects of an artificial river widening on the evolution of the longitudinal bed profile (mean bottom elevation). Besides the temporal evolution, the final equilibrium



Figure 6.3 River widening at the River Thur near Altikon (ZH) in 2014, source: C. Herrmann, BHAteam Ingenieure AG, Frauenfeld

state of the longitudinal bed profile shall be determined by means of a parameter variation for different length of the widening, width of the widening, discharge and mean grain size diameter.

6.1.1.3 Used features

BASEMENT version 4.0 is used for the simulations in this tutorial. All relevant files to run the simulations including the topographies for the different scenarios are provided. Bedload transport is calculated using the Meyer-Peter and Müller (MPM) formula for a single grain configuration. In general default values of the program are used. Two important parameters, which have been modified, are described in Table 6.1.

Table 6.1 *BASEMENT* parameters used for this tutorial

Parameter	Description	Default value	Used value
<i>max dz table</i>	Change in bottom elevation to trigger recomputation of tables [tables1D]	0.1 [m]	0.01 [m]
<i>upwind</i>	Weighting for sediment flux calculation on edge		0.7 [-]

6.1.1.4 Parameter variation

Starting from the default configuration of the reference geometry, the parameters are altered according to Table 6.2. The input files and *BASEMENT* command files (*.bmc) for each run are provided and can be identified by their run-number.

Table 6.2 Summary of the simulation runs for different parameter combinations

Parameter	run	W_W [m]	L_W [m ²]	Q [m ³ /s]	d [mm]
	default	100	500	800	20
Width variation	1	150	500	800	20
	2	200	500	800	20
Lenght variation	3	100	250	800	20
	4	100	750	800	20
	5	100	1000	800	20
Discharge variation	6	100	500	200	20
	7	100	500	400	20
	8	100	500	1600	20
Diameter variation	9	100	500	800	5
	10	100	500	800	10
	11	100	500	800	40

6.1.2 Model setup

6.1.2.1 Definition of 1D topography

All the topography files (.bmg) used in this tutorial are provided to the user. A uniform value for the Strickler roughness is chosen for all cross sections. The *bottom_range*, which defines the part of the cross section considered for the sediment transport calculations, spans from the left to the right embankment toe (Figure 6.4). This means that deposition and erosion only occur at the bottom, the embankments are fixed. A mobile bed (layer soil 1) with a thickness of 2 m is defined at the bottom.

The parameter variation is performed based on the reference geometry as specified in Table 6.3:

Table 6.3 Default configuration and variation of the investigated parameters.

Type	Symbol	Units	Description	Default value	Variation
Reference geometry	L_{US}	m	Length upstream channel	3000	
	L_T	m	Length transition zones (US/DS)	150	
	L_W	m	Length widening	5000	250-1000

Type	Symbol	Units	Description	Default value	Variation
	L_{DS}	m	Length downstream channel	3000	
	W_C	m	Width channel	50	
	W_W	m	Width widening	100	100-200
	S_C	%	Channel slope	0.15	
	S_W	%	Initial slope widening	0.15	
Additional variables	Q	m ³ /s	Steady discharge	800	200-1600
	Q_S	m ³ /s	Steady sediment discharge	0.178	$f(Q, d)$
	d	mm	Grain diameter (single grain)	20	5-40
	$k_{St,C}$	m ^{1/3} /s	Strickler roughness channel	34	
	$k_{St,W}$	m ^{1/3} /s	Strickler roughness widening	34	

6.1.2.2 Determination of the upstream sediment boundary conditions

The equilibrium sediment discharge at the inflow model boundary is determined for each discharge Q using a topography with a slope equal to the reference geometry but without river widening. The sediment discharge is calculated for different discharge and grain diameters (see Table 6.4) by *BASEMENT* using the boundary condition of type *transport_capacity*. The resulting sediment discharges are used for the parameter variation simulations afterwards. This approach was chosen to assure a constant sediment inflow independent of local erosions or depositions when running the final simulations including the local widening.

Table 6.4 Upstream sediment boundary inflow Q_S defined for different stationary discharges Q and mean grain size diameters d .

	Q [m ³ /s]	d [mm]	Q_S [m ³ /s]
Discharge variation	800	20	$Q_{S,Q800} = 0.178$
	200	20	$Q_{S,Q200} = 0.020$
	400	20	$Q_{S,Q400} = 0.043$
	1600	20	$Q_{S,Q1600} = 0.363$
Diameter variation	800	5	$Q_{S,Q800,d5} = 0.282$
	800	10	$Q_{S,Q800,d10} = 0.245$
	800	40	$Q_{S,Q800,d40} = 0.067$

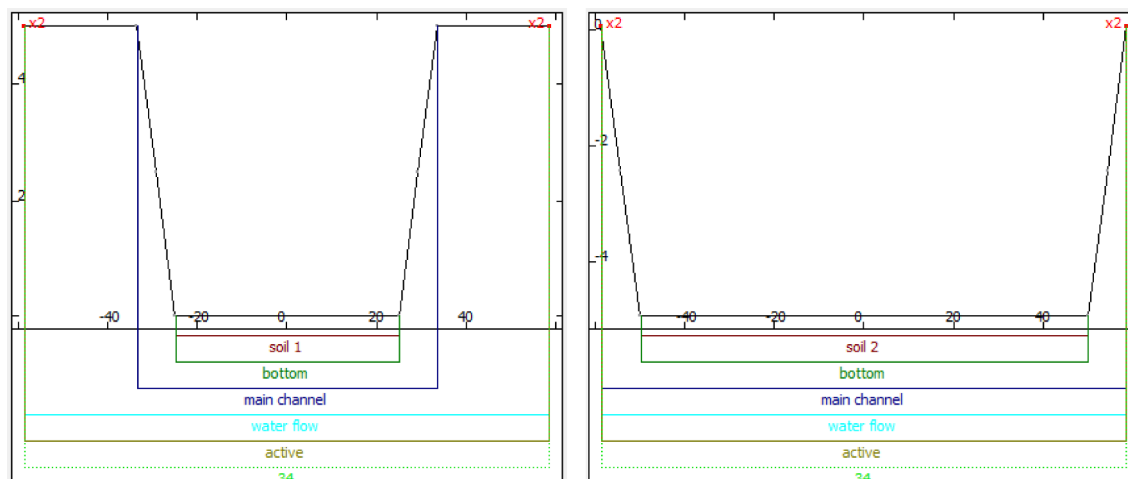


Figure 6.4 Cross sections of the BASEMENT topography representing the channel (left) and the widening (right) to illustrate the geometry and the bottom/soil definition.

6.1.3 Results of numerical simulations

6.1.3.1 Temporal evolution of longitudinal bed profile

The basic morphological processes described in the introduction can be reproduced generally well with *BASEMENT*. The results with default configuration are shown in Figure 6.5. Steady state river bed conditions are reached after a total simulation runtime of about one year.

6.1.3.2 Width and length of the widening

The effect of the width and the length of the widening (geometric parameters W_W and L_W) on the longitudinal bed profile and equilibrium slope is shown in Figure 6.6 and Figure 6.7. The corresponding bottom offset ΔH and upstream bed level change Δz_b is shown in Figure 6.8. For increasing width, ΔH rises significantly and reaches about 2.7 m for $W_W = 200$ m. The new equilibrium slope inside the widening is steeper than the initial slope (Figure 6.6). This causes a bed level change (aggradation) of 0.3 m upstream of the widening. The effect of the river widening length on ΔH and Δz_b is rather small as the slope in the widening remains almost constant. Δz_b increases slightly upstream with increasing length of the widening.

6.1.3.3 Discharge and grain diameter

The effect of the discharge and the grain diameter (parameters Q and d) on the longitudinal bed profile and the equilibrium slope is shown in Figure 6.9 and Figure 6.10. The corresponding bottom offset ΔH and upstream bed level change Δz_b are shown in Figure 6.11. For one geometry but varying discharge, big differences regarding the final topography are observed. For small discharges (200 and 400 m^3/s), ΔH reaches only about 1 m but the slope in the widening is much steeper than the initial one. Therefore the bed level change upstream of the widening is quite large. For very high values of Q however, the slope in the widening is smaller than in the channel and Δz_b becomes negative. The

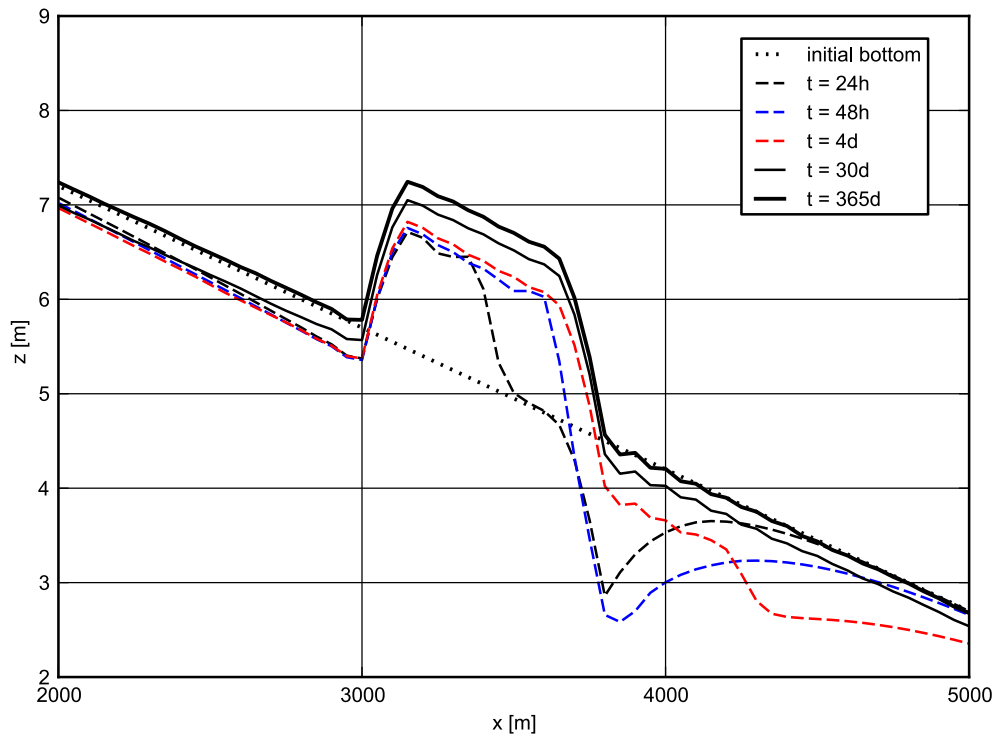


Figure 6.5 Temporal evolution of the mean bottom elevation of the reference geometry with default configuration.

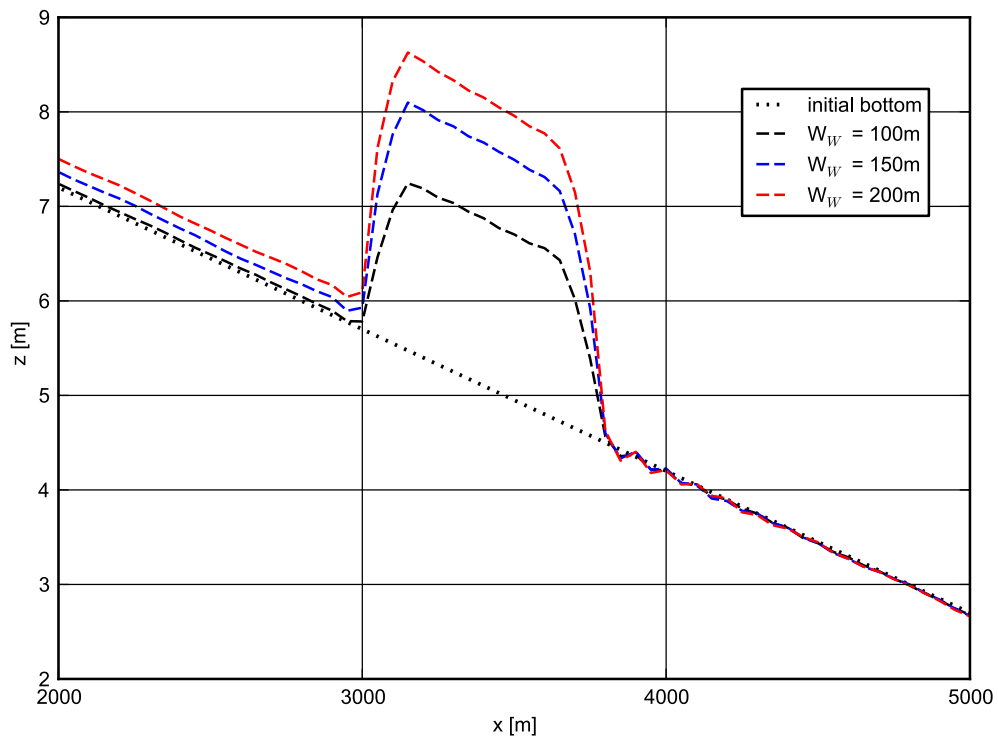


Figure 6.6 Longitudinal bed profile of the initial and resulting steady state for different widening width W_W .

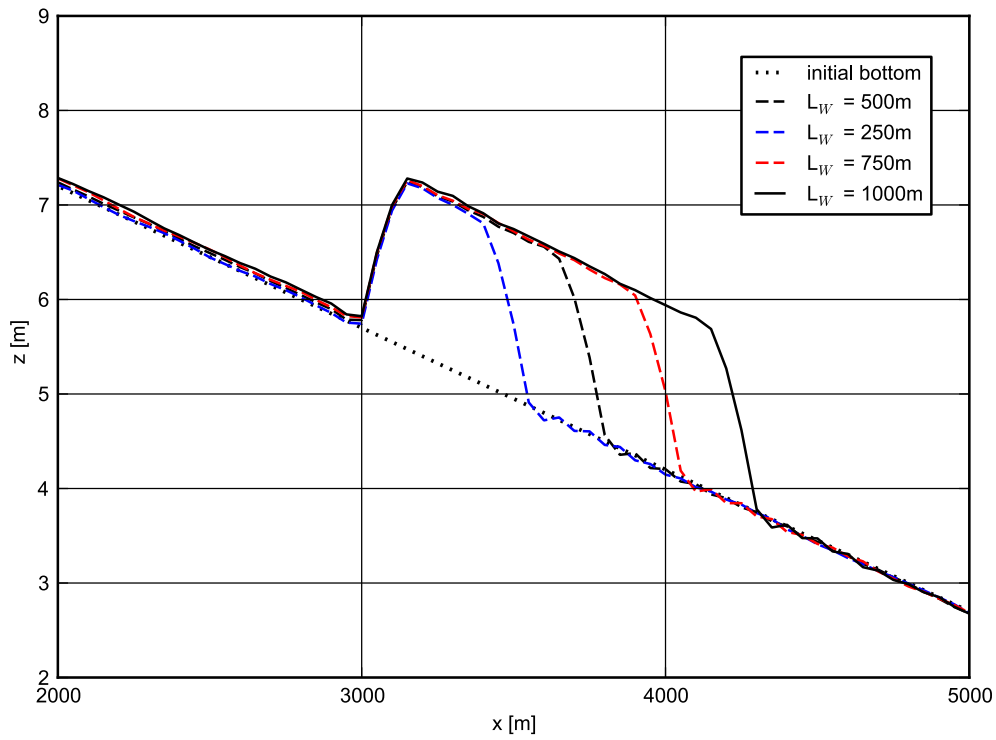


Figure 6.7 Longitudinal bed profile of the initial and resulting steady state for different widening length L_W .

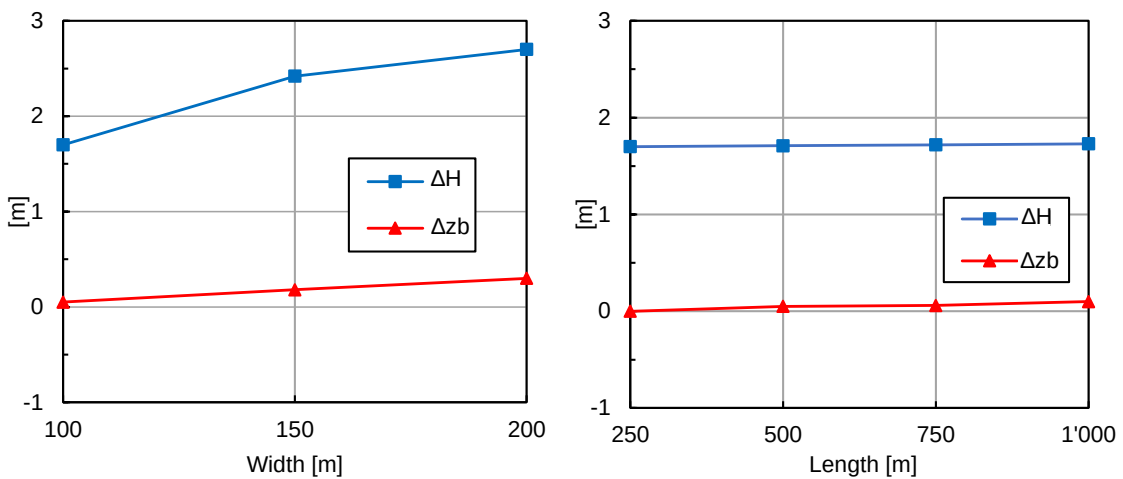


Figure 6.8 Bottom offset ΔH and upstream displacement Δz_b for different W_W (left) and L_W (right), parameter variation according to table 6.2.

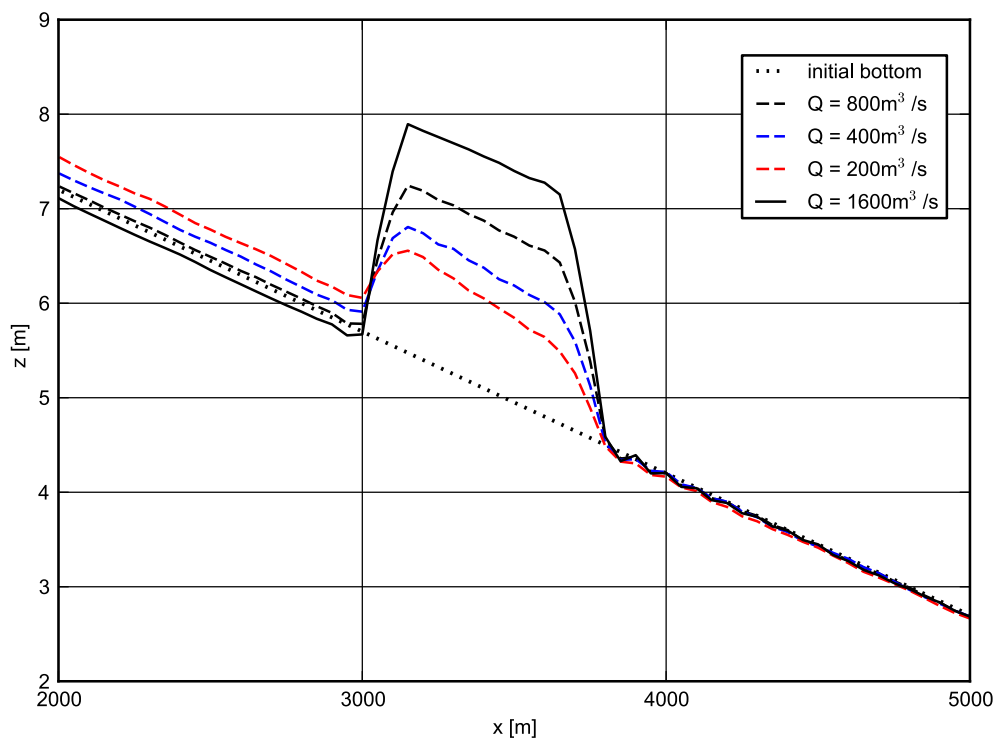


Figure 6.9 Longitudinal bed profile of the initial and resulting steady state for different discharges Q .

same effect can be observed for small grain diameters ($d = 5$ mm). Note that these extreme values for discharge and grain diameters do not correspond to the real conditions at the river Thur and can therefore be considered as hypothetical configurations. In the case of a coarser grain diameter of 40 mm, the bed slope in the widening get larger than for the default configuration and again depositions in the upstream reach are observed.

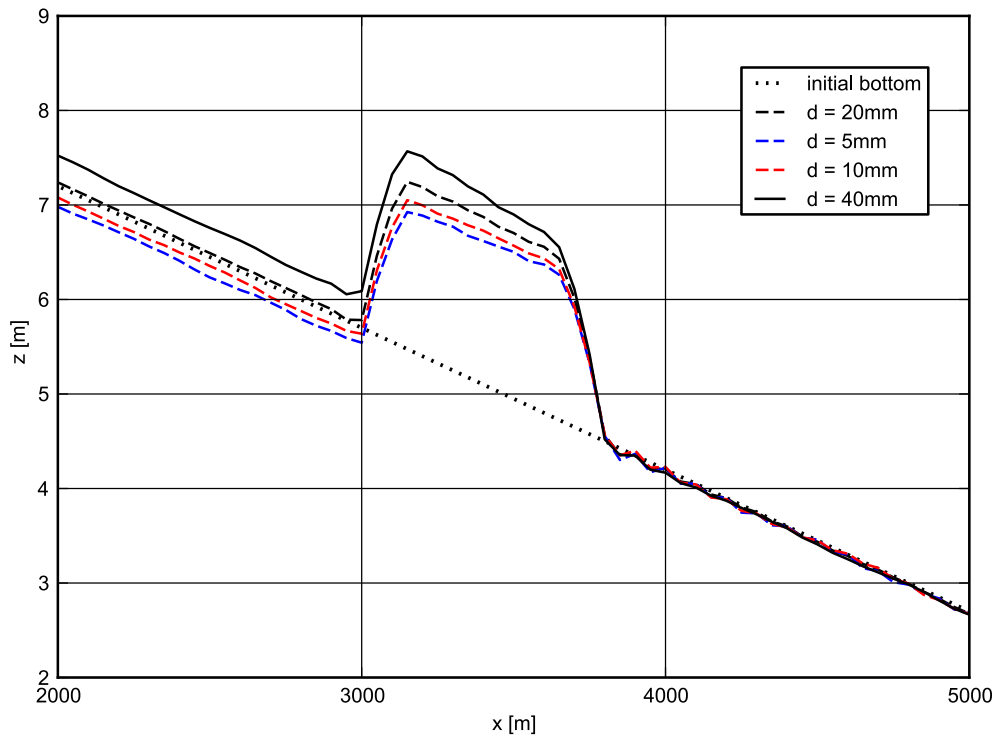


Figure 6.10 Longitudinal bed profile of the initial and resulting steady state for different mean grain size diameters d .

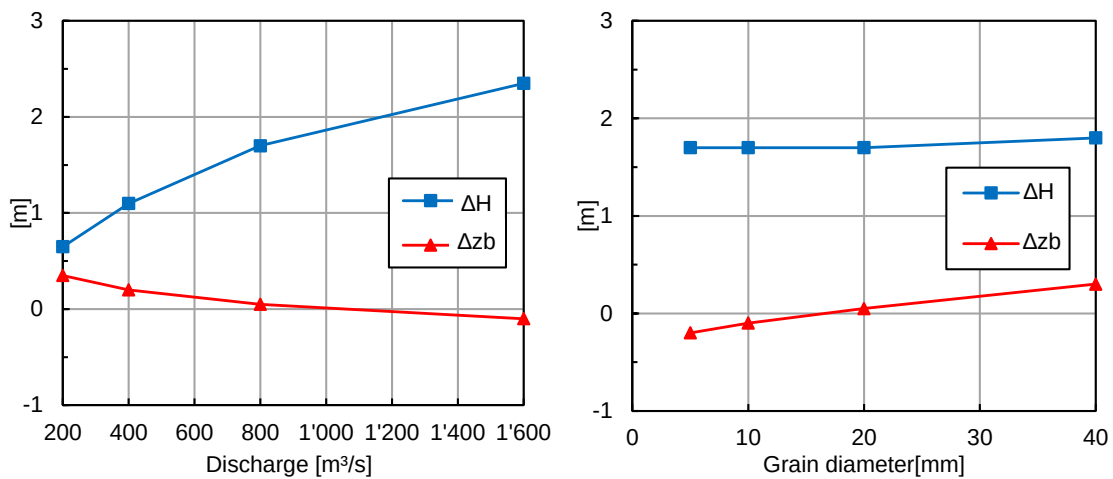


Figure 6.11 Bottom offset ΔH and upstream displacement Δz_b for different discharges Q (left) and mean grain size diameter d (right) parameter variation according to table 6.2.

Laterally coupled 1D-2D hydrodynamic simulation with BASEMD

7.1 Laterally coupled 1D-2D hydrodynamic simulation with BASEMD

7.1.1 Introduction

BASEMENT offers the possibility to combine BASEchain (1D) and BASEplane (2D) domains in coupled simulations. So-called “laterally coupled” simulations simulate the channelized river flow with a 1D approach, while the overland flow in the floodplains, in contrast, is simulated with a 2D approach. Using this coupling concept, the advantages of 1D models (like efficiency, simplicity, representation of 1D flow structures (weirs, gates)) and the advantages of 2D models (like differing flow directions and flow paths, complex topographies) shall be combined.

Coupled 1D / 2D simulations are an interesting modelling approach, especially in case of spatially extended scenarios with a clearly defined main channel and overflow into 2D floodplains with complex flow paths or topographies (like urban areas). However, be aware that this concept also suffers from the limitations of both model types (like the 1D flow assumptions, neglect of momentum exchange, etc.) and should be applied with caution.

The model coupling takes place laterally along the 1D river. If the water level exceeds the left or right river dyke crest, the water overtops the dyke and flows into the 2D floodplain. The water exchange is internally modelled as external sources, i.e. the water is removed from the 1D model and added to the 2D model as external sources, thereby neglecting the momentum exchange. In a similar way, water from the 2D floodplain flows back, if its water level exceeds the dyke crest and the water level of the 1D domain.

Water exchange takes place at defined connections between 2D boundary edges and 1D

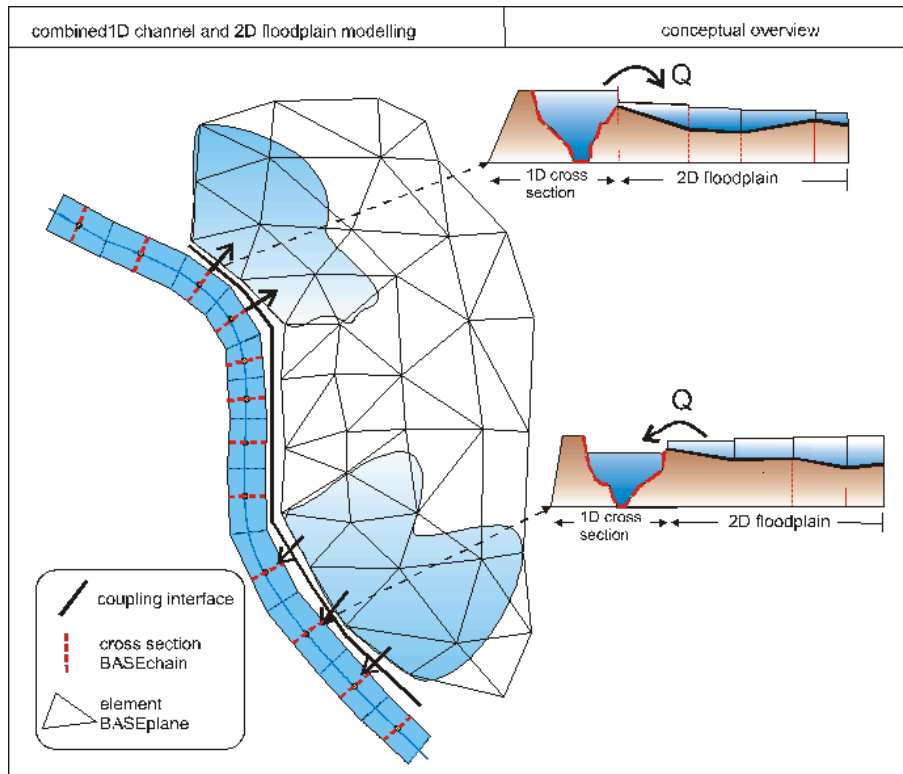


Figure 7.1 Conceptual sketch of lateral coupling of 1D BASEchain (river channel) and 2D BASEplane (overland flow) sub-domains. Water is exchanged as weir overflow over the river dykes.

cross sections. The overtopping flow is determined using simple 1D weir formulas. To apply these weir formulas, water levels are needed from both models. The 1D water level is hereby interpolated between two cross-sections and the 2D water level is taken from the corresponding 2D cell.

This tutorial briefly introduces the use of this lateral coupling approach based on a simple scenario. Only hydraulic computations are performed, since the lateral coupling supports no morphological computations up-to-now. The tutorial makes use of the QGIS-plugin BASEmesh (see BASEmesh tutorial Section 1.1) for visual checks, which is recommended for setting-up lateral coupling simulations.

7.1.2 Set-up of model file

7.1.2.1 General remarks on mesh creation

Using the 1D / 2D coupling, parts of the domain are represented with the 1D model and other parts with the 2D model. It is important hereby, that both meshes do not overlap, since this would create artificial and unphysical storage volumes and flow paths. In a similar way, there should be no gaps between the model meshes. In this tutorial, both model meshes are adapted to fit to each other along the right river dyke as illustrated in Figure 7.2.

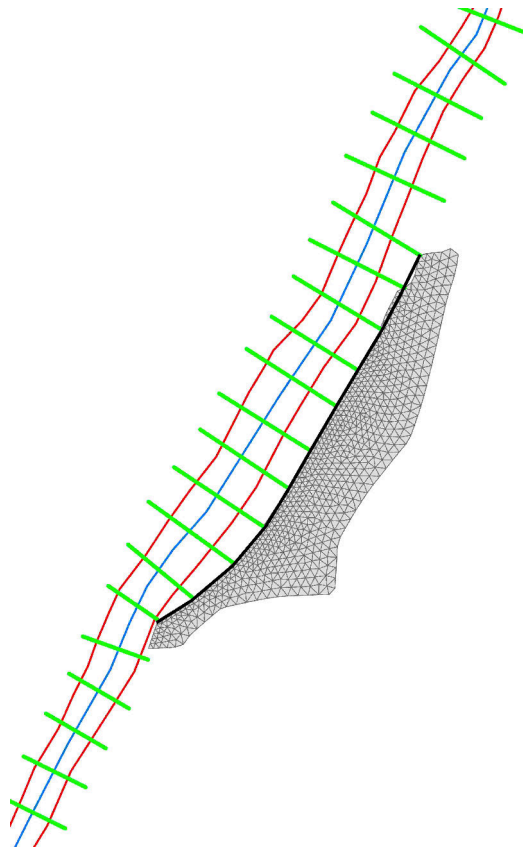


Figure 7.2 1D cross-sections (green lines), 2D mesh (gray) and dyke break-line (black).
Cross-sections and 2D mesh are connected along the river dyke, without gaps or overlapping areas. Flow direction is from bottom left to top right.

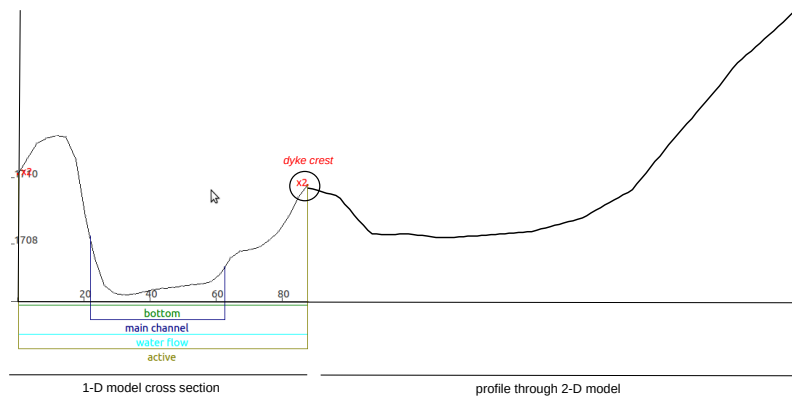


Figure 7.3 Connection of 1D cross section (left) with 2D model (transversal profile, right) at the dyke location. Water overtopping takes place via the coupling connection.

7.1.2.2 BASEchain (1D) river model

First, the 1D model of the river is set-up, analogous to the steps of the Thur tutorial (see Section 5.1) outlined above.

```
BASECHAIN_1D {
    region_name = river_flaz_1D
    GEOMETRY {
        ...
    }
    TIMESTEP {
        ...
    }
    HYDRAULICS {
        ...
    }
}
```

The topography file (*.bmg) contains the cross section points, the main channel definitions, and the active range definitions. This data was obtained here by cutting profiles through a digital terrain model. Of special importance for the lateral coupling is the accurate definition of the active range (defined by the left and right dykes), because the water exchange into the 2D floodplain takes place along these dykes. Figure 7.3 shows the definition of the right dyke in the 1D cross-section and the connection to the 2D mesh (shown as transversal profile).

The adaptation (“fitting”) of the model meshes along the river dyke was achieved by cutting the 1D cross-sections (*active_range*) at the river dyke location and by using the dyke as break-line for the boundary of the 2D model.

In this scenario, a steady-state inflow of $700 \text{ m}^3/\text{s}$ is defined for the 1D model using a hydrograph inflow boundary. This discharge is large enough to overtop the dyke and to flood the 2D floodplain. The outflow boundary is modelled as a zero-gradient boundary. The simulation is run until steady-state conditions are reached, which are used later on as initial conditions for the coupled simulation.

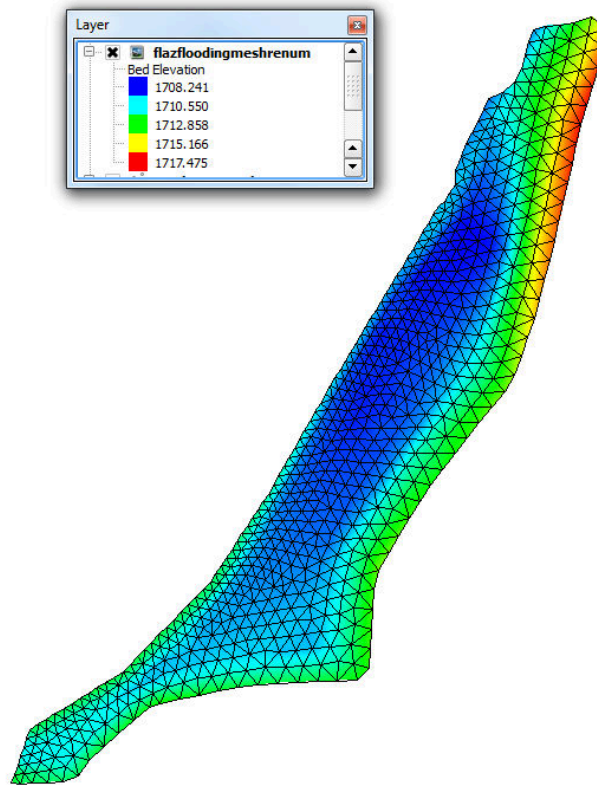


Figure 7.4 Unstructured 2D mesh of the floodplain located to the right of the 1D river channel.

7.1.2.3 BASEplane (2D) floodplain model

The floodplain outside of the main river channel is modelled with a 2D BASEplane sub-domain. Figure 7.4 illustrates the corresponding 2D mesh.

The 2D floodplain model is quite simple to set-up – it has neither inflow boundaries nor outflow boundaries, i.e. water cannot enter or leave the domain, except via the lateral coupling. The model domain is dry at simulation start.

```

BASEPLANE_2D {
  region_name = floodplain_flaz_2D
  GEOMETRY {
    ...
  }
  HYDRAULICS {
    ...
  }
  TIMESTEP {
    ...
  }
}

```

7.1.2.4 Lateral coupling set-up

The command file of the coupled simulation must contain both the BASEchain and BASEplane sub-domain definitions. These are coupled with each other by defining an additional COUPLINGS-block:

```
DOMAIN {
  BASECHAIN_1D {
    ...
  }
  BASEPLANE_2D {
    ...
  }
}
COUPLINGS {
  PARAMETER {
    max_time_level = 1
  }
  COUPLING_LATERAL {
    basechain = flaz_river_1D
    baseplane = flaz_floodplain_2D
    connections_filename = connections.dat
    c_m = 0.6
    weir_from_dyke1D = no
  }
}
}
```

The COUPLINGS-block has just two inner blocks in this scenario. The PARAMETER-block is used to define the time level at which the exchange takes place (for details see {#sec:afmodelcoupling} (Section 8.2) in the User manual). Usually, this parameter is set to 1 for lateral coupling, meaning that both sub-domains are executed with the same (minimum) time step size and water exchange takes place after each time step. This setting is recommended for lateral coupling to prevent or mitigate oscillations.

The second inner block is the COUPLING_LATERAL-block. Its main settings are the names of the coupled BASEchain and the BASEplane sub-domains and the filename of the coupling connections file (see Section 7.1.3). Additional parameters are the empirical coefficients for the weir / side-weir formulas. Furthermore, one must specify whether the weir crest elevation (= the dyke crest) is taken from the 2D mesh elevations (*weir_from_dyke1D* = *no*, default) or is interpolated in between the 1D cross-sections at the connection location. Determining the weir crest elevation from the 2D mesh topography is often advantageous, especially if the 2D mesh topography is based on high-resolution DEM models. Interpolating between 1D cross-sections, in contrast, may neglect local depressions or dyke openings.

7.1.3 Coupling connections between 1D and 2D subdomains

7.1.3.1 Definition of coupling connections

The dynamic interactions between the 1D and 2D models take place via a list of defined coupling connections. These connections are defined by the name of a 1D cross-section

and the corresponding node numbers of the connected 2D boundary edge. Cross-sections thereby may have multiple 2D boundary edges attached to it (1:n-relation), whereas each 2D boundary edge can only be connected to a single cross-section (1:1-relation).

The geometrical connections must be listed in an ASCII-file, which is read at the start of the simulation. The file contains the following information and has the shown data format:

Cross section name	dyke location	nodeID 1	nodeID 2
CS1	1	44	67
CS1	1	67	43
CS2	1	47	46
...			

1. Column = name of cross-section
2. Column = location of the dyke (0 = left, 1 = right; seen in flow direction)
3. Column = node 1 (of 2-D boundary edge)
4. Column = node 2 (of 2-D boundary edge)

In principle, it is possible to define these connections manually by filling out the connections file. In practice, however, this is usually unfeasible, due to the large number of connections along the river reach. To ease the determination of the coupling connections between both meshes, BASEMENT offers an automatic detection mechanism.

7.1.3.2 Automatic generation of coupling connections

The automatic detection and generation of coupling connections is based on a search algorithm. It searches from the left and right dyke of each 1D cross section for the nearest 2D boundary edges within a specified distance.

```
COUPLING_LATERAL {
    ...
    connections_automatic = yes
    connections_distance = 30
    connections_index = ( 6 )
    ...
}
```

To use the automatic detection, insert the tag `connections_automatic = yes` in the `COUPLING_LATERAL` block. Furthermore, you need to set a `connection_distance`, which defines the search distance in which it is searched for 2D boundary edges (If you choose the distance too small, some 2D boundary edges may not be connected to a cross-section!). In addition, you can limit the detection of 2D boundary edges to a list of cell material indices. Using this option, you can easily prevent some areas from being automatically connected and from taking part in the lateral coupling by setting corresponding material indices.

The general workflow of the automatic connection detection is as follows:

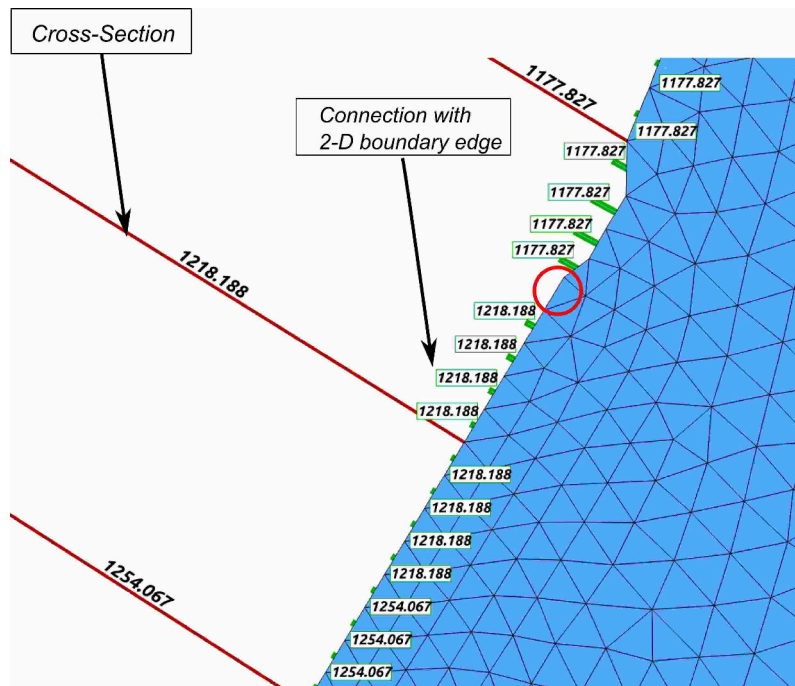


Figure 7.5 1D cross sections (red) connected to 2D boundary edges (blue mesh) via coupling connections (green). The labels show the cross-section names and indicate the corresponding mapping of the 2D edge \rightarrow 1D cross-section. The red circle indicates a 2D boundary edge without a coupling connection, which is not taking part in the lateral coupling.

1. Open the model file in BASEMENT and set the tag: `connections_automatic = yes`.
2. Write the `setup.h5` file. On the simulation tab, Run BASEMENT. The program will automatically search for the connections and write the detected connections to file and then terminates. The simulation will then abort with the error `ERROR -> Please check the created lateral connections in the file 'lateral_connections_created.dat'` and restart with `'connections_automatic = no'` using the created file. Note: Automatic creation of lateral connections may have failed. This error is normal. Three files are written related to generation of the connections:
 - `lateral_connections_created.dat` = the coupling connection file,
 - `lateral_connections_stats.shp` = shapefile for visualizing the generated connections using GIS-software,
 - `lateral_connections_stats.txt` = tab-delimited text file for visualizing the generated connections with other software.
3. In the model file, set the tag: `connections_automatic = no`. Set the tag: `connections_filename = yes`.

The connections are shown as green lines in Figure 7.5 and are drawn from the middle of each 2D boundary edge orthogonal to the imaginary line between the dykes of two neighbored cross-sections. (These green lines should be as short as possible in order to minimize gaps and overlapping areas of the 1D and 2D meshes).

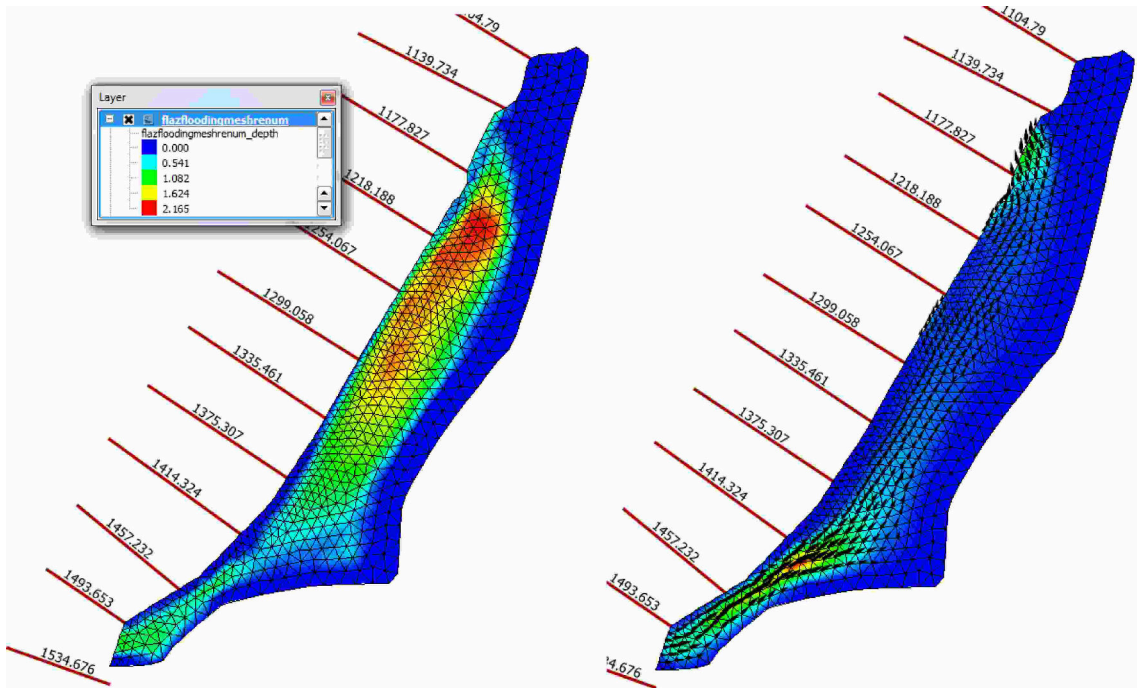


Figure 7.6 Computed water depths (left) and velocity vectors (right) in the floodplain at steady-state conditions. Water enters the floodplain at the lower left and leaves it at the upper right.

If the connections are set-up correctly in the connections file, then enter its name in the corresponding tag in the COUPLING_LATERAL-block and de-activate the automatic generation of connections:

```
connections_filename = lateral_connections_created.dat
connections_automatic = no
```

7.1.4 Perform coupled simulation

Run the simulation with the BASEchain and BASEplane sub-domains connected via the COUPLINGS-block. Be aware that the mesh files, the model file, the coupling connections file and all other input files have to be in the same folder. The defined outputs will also be written into the same folder. The computed water depths (left) and velocities (right) of the steady-state simulation are shown in Figure 7.6. The water enters the floodplain at the lower left, flows through the floodplain and leaves it at the upper right by overtopping the dyke into the river channel.

To check if the water mass balance is fulfilled, we can simply check the inflow and outflow of the 1D model. Both must have the same value, since the water discharges leaving and entering the 1D model over the dykes cancel out to zero during steady-state conditions.

Vegetation dynamics and bedload transport with BASEHPC

8.1 Vegetation dynamics and bedload transport with BASEHPC

8.1.1 Introduction

The aim of this tutorial is to show the specific setup of the configuration files for numerical simulations with BASEMENT that include vegetation and bed load transport. For this purpose, the model is applied to a 1 km-long reach of the Alpine Rhine River located at the boundary between Switzerland and Lichtenstein. This particular river reach is characterized by an alternating sequence of gravel bars, which are hotspot for riparian vegetation growth. This tutorial presents the simulation of the flood event of 2005.

8.1.2 Computational Mesh

The computational mesh of the Alpine Rhine river was created ad-hoc to show the main model functionalities with the vegetation module activated and for application with BASEMENT version 3. The mesh has 9799 number of cells with 5149 vertices. The MATID was defined on the gravel bars, the main channel, and the banks to differentiate the bed properties. Figure 8.1 shows the bed elevation of the river used in the simulation.

8.1.3 Morphology with Vegetation

The simulation in the tutorial includes a steady-state hydrodynamic simulation and an unsteady morphodynamic simulation with vegetation. The first one is used to get the initial conditions for the hydraulic variables. The unsteady simulation

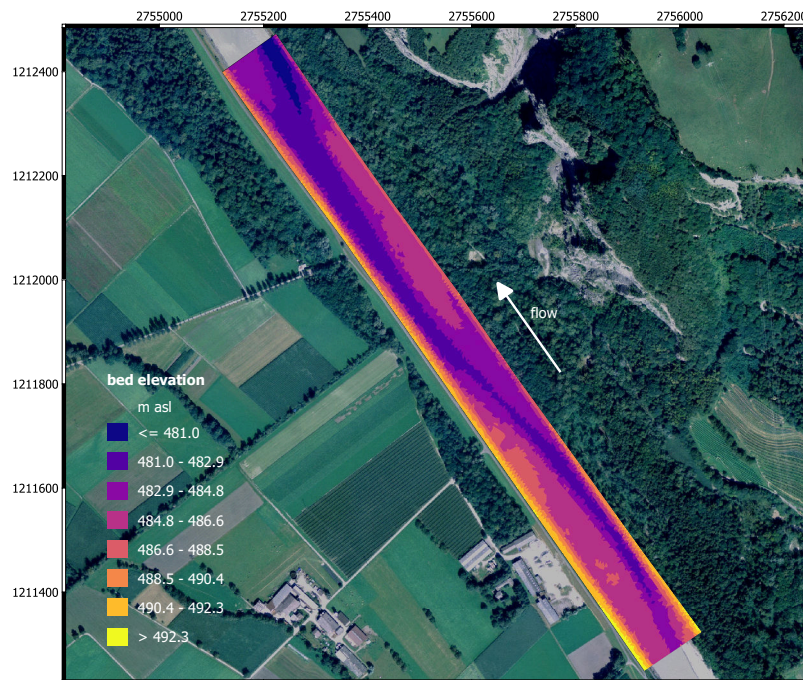


Figure 8.1 Planar view of the initial bottom elevation of the Alpine Rhine river

The setup of the configuration files follows the same procedure seen in the Flaz tutorial for hydraulics and morphology (Section 3.1.3.1 and Section 3.1.5.1). Here only the vegetation part is shown in detail.

8.1.3.1 Setup the Configuration File model.json

The model is set up with the blocks HYDRAULICS, MORPHOLOGY, and VEGETATION activated. The structure of the VEGETATION block in the file model.json is:

```
{
  "SETUP":{
    "simulation_name":"tutorial_morpho_veg_flood",
    "BASEHPC": {
      "PHYSICAL_PROPERTIES": {...},
      "BASEPLANE_2D": {
        "GEOMETRY": {...},
        "HYDRAULICS": {...},
        "MORPHOLOGY": {...},
        "VEGETATION": {
          "INITIAL": {},
          "PARAMETERS": {}
        }
      }
    }
  }
}
```

The blocks INITIAL and PARAMETERS are both mandatory. The initial conditions for vegetation can be set up using a restart file from a previous simulation (as for the hydraulics) or with input values for the aboveground, belowground biomass, and the rooting depth. In the latter case, the structure is:

```
"INITIAL": {
  "type": "region_defined",
  "regions": [
    {
      "aboveground_biomass": 0.4,
      "belowground_biomass": 0.6,
      "rooting_depth": 1.5,
      "region_name": "bar_up"
    },
    {
      "aboveground_biomass": 0.8,
      "belowground_biomass": 0.2,
      "rooting_depth": 0.5,
      "region_name": "bar_dw"
    },
    {
      "aboveground_biomass": 0.0,
      "belowground_biomass": 0.0,
      "rooting_depth": 0.0,
      "region_name": "channel"
    },
    ...
  ]
}
```

In this tutorial the vegetation was placed only in the gravel bars and using different characteristics for each of the bar (Figure 8.2). It is important that the sum of the aboveground and belowground biomass equal 1 at maximum. The biomass values are dimensionless, while the rooting depth is expressed in meter.

A number of parameters are needed to run simulation with vegetation. These are all collected in the block PARAMETERS, which has the following elements:

```
"PARAMETER": {
  "burial_factor": 1,
  "plant_height_exp": 1,
  "plant_height_fact": 1,
  "uprooting_factor": 1,
  "veg_strickler_fact": 15,
  "veg_theta_critical_fact": 0.02
}
```

The parameters that are mandatory are the Strickler coefficient, the critical Shield parameter valid for completely vegetated bed, and the parameters for calculating the plant height

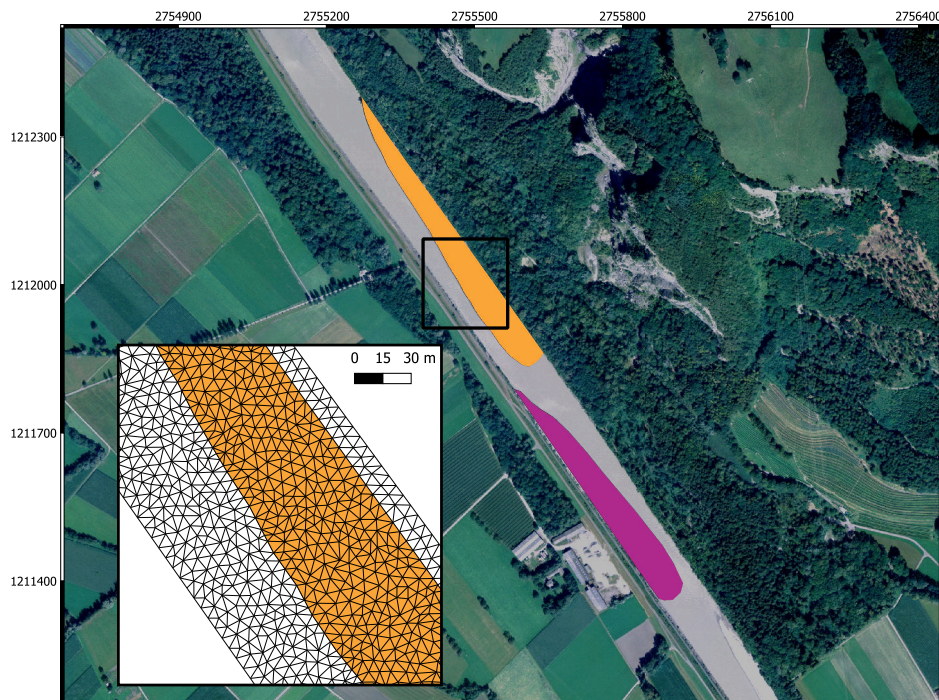


Figure 8.2 Initial location of vegetation on the two bars and an extract of the computational mesh used.

starting from the aboveground biomass (“plant_height_exp” and “plant_height_fact”). The others are by default set to 1. It is important to note that vegetation should be used in combination with the “MPM”, “MPM_like”, or “MPM_multi” formula for bedload transport and a “strickler” type friction closure for the hydrodynamics.

8.1.3.2 Setup the Configuration File simulation.json

The configuration file simulation.json defines the simulation time parameters (seconds) in the block TIME and the different output types inside the OUTPUT block.

```
{
  "SIMULATION": {
    "OUTPUT": [
      "bottom_elevation",
      "flow_velocity",
      "friction_chezy",
      "water_surface",
      "delta_z",
      "water_depth",
      "theta_critical",
      "trsp_capacity",
      "aboveground_biomass",
      "belowground_biomass",
      "rooting_depth"
    ],
  },
}
```

```
"TIME": {
  "end": 111600.0,
  "out": 11160.0,
  "start": 0.0
},
"TIMESTEP": {
  "init": 0.02,
  "minimum": 0.0001
}
}
```

There are three additional available outputs for vegetation: the aboveground biomass, the belowground biomass, and the rooting depth.

8.1.3.3 Set up the Configuration File `results.json`

The configuration file `results.json` defines the output format in the block `EXPORT`. Currently, `xmfd` is the only output format available.

```
{
  "RESULTS": {
    "EXPORT": [
      {"format": "xmfd"}
    ]
  }
}
```

8.1.3.4 Results

An example of the results of the simulation of the 2005 flood in the Alpine Rhine river considering vegetation is shown in Figure 8.3. Vegetation is removed in most parts of the bars as a result of uprooting, while the aboveground biomass of surviving vegetation changed because of erosion and deposition processes, which modify biomass redistribution.

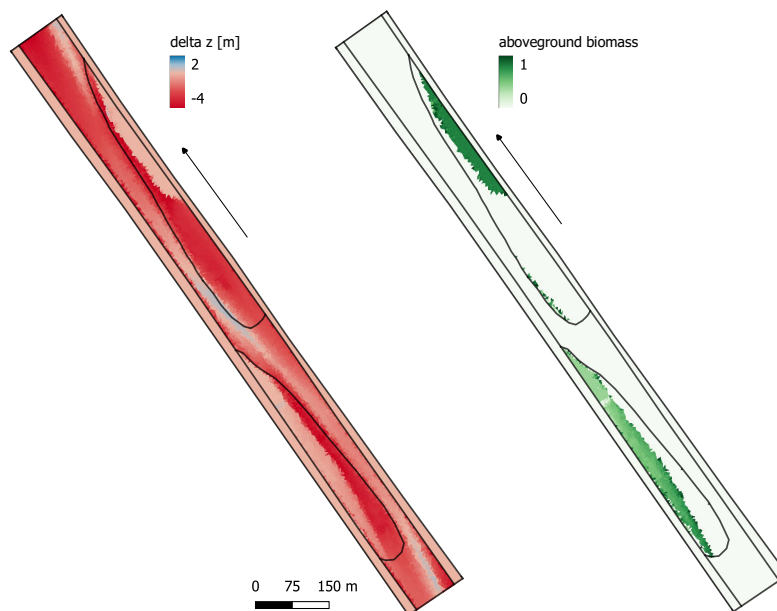


Figure 8.3 Results for the bed elevation and the aboveground biomass after the flood (at $t=111600$ s). Breaklines are highlighted in black.

9

References

**BASIC SIMULATION ENVIRONMENT
FOR MODELLING OF ENVIRONMENTAL
FLOWS AND NATURAL HAZARDS**

APPENDIX

**VERSION 4.1.0
JUNE 2024**



BASEMENT

Third Party Software

1.1 Third party software licenses

Abseil

Apache License

Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Brotli

Copyright (c) 2009, 2010, 2013-2016 by the Brotli Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Bzip2

This program, "bzip2", the associated library "libbzip2", and all documentation, are copyright (C) 1996-2019 Julian R Seward. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
3. Altered source versions must be plainly marked as such, and must

not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``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 AUTHOR 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.

Julian Seward, jseward@acm.org
bzip2/libbzip2 version 1.0.8 of 13 July 2019

Curl

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2022, Daniel Stenberg, <daniel@haxx.se>, and many contributors, see the THANKS file.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

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 AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Double-conversion

Copyright 2006-2011, the V8 project authors. 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 the name of Google Inc. nor the names of its 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 COPYRIGHT OWNER 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.

Egl-registry

Copyright (c) 2008-2018 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE 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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

Copyright 2013-2020 The Khronos Group Inc.
Copyright 2007-2020 The Khronos Group Inc.

SPDX-License-Identifier: Apache-2.0

Eigen3

Eigen is primarily MPL2 licensed. See COPYING.MPL2 and these links:
<http://www.mozilla.org/MPL/2.0/>
<http://www.mozilla.org/MPL/2.0/FAQ.html>

Some files contain third-party code under BSD or LGPL licenses, whence the other COPYING.* files here.

All the LGPL code is either LGPL 2.1-only, or LGPL 2.1-or-later.
For this reason, the COPYING.LGPL file contains the LGPL 2.1 text.

If you want to guarantee that the Eigen code that you are #including is licensed under the MPL2 and possibly more permissive licenses (like BSD), #define this preprocessor symbol:

```
EIGEN_MPL2_ONLY
```

For example, with most compilers, you could add this to your project CXXFLAGS:
-DEIGEN_MPL2_ONLY

This will cause a compilation error to be generated if you #include any code that is LGPL licensed.

Copyright (c) 2011, Intel Corporation. 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 the name of Intel Corporation nor the names of its 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 COPYRIGHT OWNER 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.

Minpack Copyright Notice (1999) University of Chicago. All rights reserved

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. 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.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND. THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER

SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE, EVEN IF ANY OF SAID PARTIES HAS BEEN WARNED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

Expat

Copyright (c) 1998-2000 Thai Open Source Software Center Ltd and Clark Cooper
Copyright (c) 2001-2019 Expat maintainers

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Freetype

FREETYPE LICENSES

The FreeType 2 font engine is copyrighted work and cannot be used legally without a software license. In order to make this project usable to a vast majority of developers, we distribute it under two mutually exclusive open-source licenses.

This means that *you* must choose *one* of the two licenses described below, then obey all its terms and conditions when using FreeType 2 in any of your projects or products.

- The FreeType License, found in the file ``docs/FTL.TXT``, which is similar to the original BSD license *with* an advertising clause that forces you to explicitly cite the FreeType project in your product's documentation. All details are in the license file. This license is suited to products which don't use the GNU General Public License.

Note that this license is compatible to the GNU General Public License version 3, but not version 2.

- The GNU General Public License version 2, found in ``docs/GPLv2.TXT`` (any later version can be used also), for programs which already use the GPL. Note that the FTL is incompatible with GPLv2 due to its advertisement clause.

The contributed BDF and PCF drivers come with a license similar to that of the X Window System. It is compatible to the above two licenses (see files ``src/bdf/README`` and ``src/pcf/README``). The same holds for the source code files ``src/base/fthash.c`` and ``include/freetype/internal/fthash.h``; they were part of the BDF driver in earlier FreeType versions.

The gzip module uses the zlib license (see ``src/gzip/zlib.h``) which too is compatible to the above two licenses.

The MD5 checksum support (only used for debugging in development builds) is in the public domain.

--- end of LICENSE.TXT ---

Glew

The OpenGL Extension Wrangler Library
Copyright (C) 2002-2007, Milan Ikits <milan ikits[at]ieee org>
Copyright (C) 2002-2007, Marcelo E. Magallon <mmagallo[at]debian org>
Copyright (C) 2002, Lev Povalahev
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.
- * The name of the author 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 COPYRIGHT OWNER 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.

Mesa 3-D graphics library
Version: 7.0

Copyright (C) 1999-2007 Brian Paul All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL BRIAN PAUL BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2007 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to

permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE 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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

Glibc-queue

Copyright (C) 1991-2015 Free Software Foundation, Inc.

The GNU C 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.

The GNU C 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 the GNU C Library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

* All code incorporated from 4.4 BSD is distributed under the following license:

Copyright (C) 1991 Regents of the University of California.
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. 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.
3. [This condition was removed.]
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS 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 REGENTS 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.

Gtest

Copyright 2008, Google Inc.

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 the name of Google Inc. nor the names of its 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 COPYRIGHT OWNER 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.

Harfbuzz

HarfBuzz is licensed under the so-called "Old MIT" license. Details follow. For parts of HarfBuzz that are licensed under different licenses see individual files names COPYING in subdirectories where applicable.

Copyright © 2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020 Google, Inc.
 Copyright © 2018,2019,2020 Ebrahim Byagowi
 Copyright © 2019,2020 Facebook, Inc.
 Copyright © 2012 Mozilla Foundation
 Copyright © 2011 Codethink Limited
 Copyright © 2008,2010 Nokia Corporation and/or its subsidiary(-ies)
 Copyright © 2009 Keith Stribley
 Copyright © 2009 Martin Hosken and SIL International
 Copyright © 2007 Chris Wilson
 Copyright © 2005,2006,2020,2021 Behdad Esfahbod
 Copyright © 2005 David Turner
 Copyright © 2004,2007,2008,2009,2010 Red Hat, Inc.
 Copyright © 1998-2004 David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

Hdf5

Copyright Notice and License Terms for
HDF5 (Hierarchical Data Format 5) Software Library and Utilities

HDF5 (Hierarchical Data Format 5) Software Library and Utilities
Copyright 2006 by The HDF Group.

NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities
Copyright 1998-2006 by The Board of Trustees of the University of Illinois.

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted for any purpose (including commercial purposes)
provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions, and the following disclaimer in the documentation
and/or materials provided with the distribution.
3. Neither the name of The HDF Group, the name of the University, nor the
name of any Contributor may be used to endorse or promote products derived
from this software without specific prior written permission from
The HDF Group, the University, or the Contributor, respectively.

DISCLAIMER:

THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS
"AS IS" WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. IN NO
EVENT SHALL THE HDF GROUP OR THE CONTRIBUTORS BE LIABLE FOR ANY DAMAGES
SUFFERED BY THE USERS ARISING OUT OF THE USE OF THIS SOFTWARE, EVEN IF
ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You are under no obligation whatsoever to provide any bug fixes, patches, or
upgrades to the features, functionality or performance of the source code
("Enhancements") to anyone; however, if you choose to make your Enhancements
available either publicly, or directly to The HDF Group, without imposing a
separate written license agreement for such Enhancements, then you hereby
grant the following license: a non-exclusive, royalty-free perpetual license
to install, use, modify, prepare derivative works, incorporate into other
computer software, distribute, and sublicense such enhancements or derivative
works thereof, in binary and source code form.

Limited portions of HDF5 were developed by Lawrence Berkeley National
Laboratory (LBNL). LBNL's Copyright Notice and Licensing Terms can be
found here: COPYING_LBNL_HDF5 file in this directory or at
http://support.hdfgroup.org/ftp/HDF5/releases/COPYING_LBNL_HDF5.

Contributors: National Center for Supercomputing Applications (NCSA) at
the University of Illinois, Fortner Software, Unidata Program Center
(netCDF), The Independent JPEG Group (JPEG), Jean-loup Gailly and Mark Adler
(gzip), and Digital Equipment Corporation (DEC).

Portions of HDF5 were developed with support from the Lawrence Berkeley
National Laboratory (LBNL) and the United States Department of Energy
under Prime Contract No. DE-AC02-05CH11231.

Portions of HDF5 were developed with support from Lawrence Livermore National Laboratory and the United States Department of Energy under Prime Contract No. DE-AC52-07NA27344.

Portions of HDF5 were developed with support from the University of California, Lawrence Livermore National Laboratory (UC LLNL). The following statement applies to those portions of the product and must be retained in any redistribution of source code, binaries, documentation, and/or accompanying materials:

This work was partially produced at the University of California, Lawrence Livermore National Laboratory (UC LLNL) under contract no. W-7405-ENG-48 (Contract 48) between the U.S. Department of Energy (DOE) and The Regents of the University of California (University) for the operation of UC LLNL.

DISCLAIMER:

THIS WORK WAS PREPARED AS AN ACCOUNT OF WORK SPONSORED BY AN AGENCY OF THE UNITED STATES GOVERNMENT. NEITHER THE UNITED STATES GOVERNMENT NOR THE UNIVERSITY OF CALIFORNIA NOR ANY OF THEIR EMPLOYEES, MAKES ANY WARRANTY, EXPRESS OR IMPLIED, OR ASSUMES ANY LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF ANY INFORMATION, APPARATUS, PRODUCT, OR PROCESS DISCLOSED, OR REPRESENTS THAT ITS USE WOULD NOT INFRINGE PRIVATELY- OWNED RIGHTS. REFERENCE HEREIN TO ANY SPECIFIC COMMERCIAL PRODUCTS, PROCESS, OR SERVICE BY TRADE NAME, TRADEMARK, MANUFACTURER, OR OTHERWISE, DOES NOT NECESSARILY CONSTITUTE OR IMPLY ITS ENDORSEMENT, RECOMMENDATION, OR FAVORING BY THE UNITED STATES GOVERNMENT OR THE UNIVERSITY OF CALIFORNIA. THE VIEWS AND OPINIONS OF AUTHORS EXPRESSED HEREIN DO NOT NECESSARILY STATE OR REFLECT THOSE OF THE UNITED STATES GOVERNMENT OR THE UNIVERSITY OF CALIFORNIA, AND SHALL NOT BE USED FOR ADVERTISING OR PRODUCT ENDORSEMENT PURPOSES.

HighFive

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.

Icu

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

See Terms of Use <<https://www.unicode.org/copyright.html>>
for definitions of Unicode Inc.'s Data Files and Software.

NOTICE TO USER: Carefully read the following legal agreement.
BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S
DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"),
YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE
TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE
THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright © 1991-2022 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in <https://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE 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 AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

Third-Party Software Licenses

This section contains third-party software notices and/or additional
terms for licensed third-party software components included within ICU
libraries.

ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies of
the Software and that both the above copyright notice(s) and this

permission notice appear in supporting documentation.

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 AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Chinese/Japanese Word Break Dictionary Data (cjdict.txt)

```
# The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# 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 the name of Google Inc. nor the names of its
# 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 COPYRIGHT OWNER 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.
#
#
# The word list in cjdict.txt are generated by combining three word lists
# listed below with further processing for compound word breaking. The
# frequency is generated with an iterative training against Google web
# corpora.
#
# * Libtabe (Chinese)
# - https://sourceforge.net/project/?group_id=1519
# - Its license terms and conditions are shown below.
```

```

#
# * IPADIC (Japanese)
#   - http://chasen.aist-nara.ac.jp/chasen/distribution.html
#   - Its license terms and conditions are shown below.
#
# -----COPYING.libtabe ---- BEGIN-----
#
# /*
# * Copyright (c) 1999 TaBE Project.
# * Copyright (c) 1999 Pai-Hsiang Hsiao.
# * 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 the name of the TaBE Project nor the names of its
# * 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
# * REGENTS 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.
# */
#
# /*
# * Copyright (c) 1999 Computer Systems and Communication Lab,
# * Institute of Information Science, Academia
# * Sinica. 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 the name of the Computer Systems and Communication Lab
# * nor the names of its 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
# * REGENTS 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.
# */
#
# Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
# University of Illinois
# c-tsai4@uiuc.edu http://casper.beckman.uiuc.edu/~c-tsai4
# -----COPYING.libtabe-----END-----
#
# -----COPYING.ipadic-----BEGIN-----
#
# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology. All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software. The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
# that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
# on, or be attached to, the Program, which is distributed substantially
# in the same form as set out herein and that such intended
# distribution, if actually made, will neither violate or otherwise
# contravene any of the laws and regulations of the countries having
# jurisdiction over the User or the intended distribution itself.
#
# NO WARRANTY
#
# The program was produced on an experimental basis in the course of the
# research and development conducted during the project and is provided
# to users as so produced on an experimental basis. Accordingly, the
# program is provided without any warranty whatsoever, whether express,
# implied, statutory or otherwise. The term "warranty" used herein
# includes, but is not limited to, any warranty of the quality,
# performance, merchantability and fitness for a particular purpose of
# the program and the nonexistence of any infringement or violation of
# any right of any third party.
#
# Each user of the program will agree and understand, and be deemed to
# have agreed and understood, that there is no warranty whatsoever for
# the program and, accordingly, the entire risk arising from or
# otherwise connected with the program is assumed by the user.
#
# Therefore, neither ICOT, the copyright holder, or any other
# organization that participated in or was otherwise related to the
# development of the program and their respective officials, directors,
# officers and other employees shall be held liable for any and all
# damages, including, without limitation, general, special, incidental
# and consequential damages, arising out of or otherwise in connection
# with the use or inability to use the program or any product, material
# or result produced or otherwise obtained by using the program,
# regardless of whether they have been advised of, or otherwise had
# knowledge of, the possibility of such damages at any time during the

```

```

# project or thereafter. Each user will be deemed to have agreed to the
# foregoing by his or her commencement of use of the program. The term
# "use" as used herein includes, but is not limited to, the use,
# modification, copying and distribution of the program and the
# production of secondary products from the program.
#
# In the case where the program, whether in its original form or
# modified, was distributed or delivered to or received by a user from
# any person, organization or entity other than ICOT, unless it makes or
# grants independently of ICOT any specific warranty to the user in
# writing, such person, organization or entity, will also be exempted
# from and not be held liable to the user for any such damages as noted
# above as far as the program is concerned.
#
# -----COPYING.ipadic-----END-----

```

Lao Word Break Dictionary Data (laodict.txt)

```

# Copyright (C) 2016 and later: Unicode, Inc. and others.
# License & terms of use: http://www.unicode.org/copyright.html
# Copyright (c) 2015 International Business Machines Corporation
# and others. All Rights Reserved.
#
# Project: https://github.com/rober42539/lao-dictionary
# Dictionary: https://github.com/rober42539/lao-dictionary/laodict.txt
# License: https://github.com/rober42539/lao-dictionary/LICENSE.txt
#         (copied below)
#
# This file is derived from the above dictionary version of Nov 22, 2020
# -----
# Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
# 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.
#
# 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
# COPYRIGHT HOLDER 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.
# -----

```

Burmese Word Break Dictionary Data (burmesedict.txt)

```

# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
#
# This list is part of a project hosted at:
#   github.com/kanyawtech/myanmar-karen-word-lists
#
# -----
# Copyright (c) 2013, LeRoy Benjamin Sharon

```

```

# 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 the name Myanmar Karen Word Lists, nor the names of its
# 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 COPYRIGHT HOLDER 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.
# -----

```

Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

```

# 7. Database Ownership
#
# The TZ database itself is not an IETF Contribution or an IETF
# document. Rather it is a pre-existing and regularly updated work
# that is in the public domain, and is intended to remain in the
# public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do
# not apply to the TZ Database or contributions that individuals make
# to it. Should any claims be made and substantiated against the TZ
# Database, the organization that is providing the IANA
# Considerations defined in this RFC, under the memorandum of
# understanding with the IETF, currently ICANN, may act in accordance
# with all competent court orders. No ownership claims will be made
# by ICANN or the IETF Trust on the database or the code. Any person
# making a contribution to the database or code waives all rights to
# future claims in that contribution or in the TZ Database.

```

Google double-conversion

Copyright 2006-2011, the V8 project authors. 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 the name of Google Inc. nor the names of its 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 COPYRIGHT OWNER 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.

File: aclocal.m4 (only for ICU4C)
Section: pkg.m4 - Macros to locate and utilise pkg-config.

Copyright © 2004 Scott James Remnant <scott@netsplit.com>.
Copyright © 2012-2015 Dan Nicholson <dbn.lists@gmail.com>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program 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 General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

As a special exception to the GNU General Public License, if you distribute this file as part of a program that contains a configuration script generated by Autoconf, you may include it under the same distribution terms that you use for the rest of that program.

(The condition for the exception is fulfilled because ICU4C includes a configuration script generated by Autoconf, namely the `configure` script.)

File: config.guess (only for ICU4C)

This file is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

This program 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 General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, see <<https://www.gnu.org/licenses/>>.

As a special exception to the GNU General Public License, if you

distribute this file as part of a program that contains a configuration script generated by Autoconf, you may include it under the same distribution terms that you use for the rest of that program. This Exception is an additional permission under section 7 of the GNU General Public License, version 3 ("GPLv3").

(The condition for the exception is fulfilled because ICU4C includes a configuration script generated by Autoconf, namely the `configure` script.)

File: install-sh (only for ICU4C)

Copyright 1991 by the Massachusetts Institute of Technology

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Jsoncpp

The JsonCpp library's source code, including accompanying documentation, tests and demonstration applications, are licensed under the following conditions...

Baptiste Lepilleur and The JsonCpp Authors explicitly disclaim copyright in all jurisdictions which recognize such a disclaimer. In such jurisdictions, this software is released into the Public Domain.

In jurisdictions which do not recognize Public Domain property (e.g. Germany as of 2010), this software is Copyright (c) 2007-2010 by Baptiste Lepilleur and The JsonCpp Authors, and is released under the terms of the MIT License (see below).

In jurisdictions which recognize Public Domain property, the user of this software may choose to accept it either as 1) Public Domain, 2) under the conditions of the MIT License (see below), or 3) under the terms of dual Public Domain/MIT License conditions described here, as they choose.

The MIT License is about as close to Public Domain as a license can get, and is described in clear, concise terms at:

http://en.wikipedia.org/wiki/MIT_License

The full text of the MIT License follows:

=====
Copyright (c) 2007-2010 Baptiste Lepilleur and The JsonCpp Authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====
(END LICENSE TEXT)

The MIT license is compatible with both the GPL and commercial software, affording one all of the rights of Public Domain with the minor nuisance of being required to keep the above copyright notice and license text in the source code. Note also that by accepting the Public Domain "license" you can re-license your copy using whatever license you like.

Libharu

Copyright (C) 1999-2006 Takeshi Kanno
Copyright (C) 2007-2009 Antony Dovgal

This software is provided 'as-is', without any express or implied warranty.

In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Libiconv

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for

this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only

works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or

table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
- b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
- c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
- d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license

restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that

system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 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
Library General Public License for more details.
```

```
You should have received a copy of the GNU Library 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
```

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.
```

```
<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
```

That's all there is to it!

Libjpeg-turbo

libjpeg-turbo Licenses =====

libjpeg-turbo is covered by three compatible BSD-style open source licenses:

- The IJG (Independent JPEG Group) License, which is listed in [README.ijg] (README.ijg)

```
This license applies to the libjpeg API library and associated programs
(any code inherited from libjpeg, and any modifications to that code.)
```

- The Modified (3-clause) BSD License, which is listed below

```
This license covers the TurboJPEG API library and associated programs, as
well as the build system.
```

- The [zlib License] (<https://opensource.org/licenses/Zlib>)

```
This license is a subset of the other two, and it covers the libjpeg-turbo
SIMD extensions.
```

Complying with the libjpeg-turbo Licenses =====

This section provides a roll-up of the libjpeg-turbo licensing terms, to the best of our understanding.

1. If you are distributing a modified version of the libjpeg-turbo source, then:

1. You cannot alter or remove any existing copyright or license notices from the source.

```
**Origin**  
- Clause 1 of the IJG License  
- Clause 1 of the Modified BSD License  
- Clauses 1 and 3 of the zlib License
```

2. You must add your own copyright notice to the header of each source file you modified, so others can tell that you modified that file (if there is not an existing copyright header in that file, then you can simply add a notice stating that you modified the file.)

```
**Origin**  
- Clause 1 of the IJG License  
- Clause 2 of the zlib License
```

3. You must include the IJG README file, and you must not alter any of the copyright or license text in that file.

```
**Origin**  
- Clause 1 of the IJG License
```

2. If you are distributing only libjpeg-turbo binaries without the source, or if you are distributing an application that statically links with libjpeg-turbo, then:

1. Your product documentation must include a message stating:

```
This software is based in part on the work of the Independent JPEG  
Group.
```

```
**Origin**  
- Clause 2 of the IJG license
```

2. If your binary distribution includes or uses the TurboJPEG API, then your product documentation must include the text of the Modified BSD License (see below.)

```
**Origin**  
- Clause 2 of the Modified BSD License
```

3. You cannot use the name of the IJG or The libjpeg-turbo Project or the contributors thereof in advertising, publicity, etc.

```
**Origin**  
- IJG License  
- Clause 3 of the Modified BSD License
```

4. The IJG and The libjpeg-turbo Project do not warrant libjpeg-turbo to be free of defects, nor do we accept any liability for undesirable consequences resulting from your use of the software.

```
**Origin**  
- IJG License  
- Modified BSD License  
- zlib License
```

```
The Modified (3-clause) BSD License  
=====
```

```
Copyright (C)2009-2022 D. R. Commander. All Rights Reserved.  
Copyright (C)2015 Viktor Szathmary. 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 the name of the libjpeg-turbo Project nor the names of its 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 COPYRIGHT HOLDERS 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.

Why Three Licenses?

=====

The zlib License could have been used instead of the Modified (3-clause) BSD License, and since the IJG License effectively subsumes the distribution conditions of the zlib License, this would have effectively placed libjpeg-turbo binary distributions under the IJG License. However, the IJG License specifically refers to the Independent JPEG Group and does not extend attribution and endorsement protections to other entities. Thus, it was desirable to choose a license that granted us the same protections for new code that were granted to the IJG for code derived from their software.

Liblzma

XZ Utils Licensing

=====

Different licenses apply to different files in this package. Here is a rough summary of which licenses apply to which parts of this package (but check the individual files to be sure!):

- liblzma is in the public domain.
- xz, xzdec, and lzmadec command line tools are in the public domain unless GNU getopt_long had to be compiled and linked in from the lib directory. The getopt_long code is under GNU LGPLv2.1+.
- The scripts to grep, diff, and view compressed files have been adapted from gzip. These scripts and their documentation are under GNU GPLv2+.
- All the documentation in the doc directory and most of the XZ Utils specific documentation files in other directories are in the public domain.
- Translated messages are in the public domain.
- The build system contains public domain files, and files that are under GNU GPLv2+ or GNU GPLv3+. None of these files end up in the binaries being built.
- Test files and test code in the tests directory, and debugging utilities in the debug directory are in the public domain.
- The extra directory may contain public domain files, and files that are under various free software licenses.

You can do whatever you want with the files that have been put into the public domain. If you find public domain legally problematic, take the previous sentence as a license grant. If you still find the lack of copyright legally problematic, you have too many lawyers.

As usual, this software is provided "as is", without any warranty.

If you copy significant amounts of public domain code from XZ Utils into your project, acknowledging this somewhere in your software is polite (especially if it is proprietary, non-free software), but naturally it is not legally required. Here is an example of a good notice to put into "about box" or into documentation:

This software includes code from XZ Utils <<https://tukaani.org/xz/>>.

The following license texts are included in the following files:

- COPYING.LGPLv2.1: GNU Lesser General Public License version 2.1
- COPYING.GPLv2: GNU General Public License version 2
- COPYING.GPLv3: GNU General Public License version 3

Note that the toolchain (compiler, linker etc.) may add some code pieces that are copyrighted. Thus, it is possible that e.g. liblzma binary wouldn't actually be in the public domain in its entirety even though it contains no copyrighted code from the XZ Utils source package.

If you have questions, don't hesitate to ask the author(s) for more information.

Libogg

Copyright (c) 2002, Xiph.org Foundation

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 the name of the Xiph.org Foundation nor the names of its 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 FOUNDATION 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.

Libpng

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE
=====

PNG Reference Library License version 2

-
- * Copyright (c) 1995-2019 The PNG Reference Library Authors.
 - * Copyright (c) 2018-2019 Cosmin Truta.
 - * Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson.
 - * Copyright (c) 1996-1997 Andreas Dilger.
 - * Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and non-infringement. In no event shall the Copyright owners, 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, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabudhe
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the

list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

Libtheora

Copyright (C) 2002-2009 Xiph.org Foundation

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 the name of the Xiph.org Foundation nor the names of its 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 FOUNDATION 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.

Libxml2

Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar licence but with different Copyright notices) all the files are:

Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Lz4

LZ4 Library
Copyright (c) 2011-2016, Yann Collet
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.

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 COPYRIGHT HOLDER 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.

Netcdf-c

Copyright 2018 Unidata

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. 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.
3. Neither the name of the copyright holder nor the names of its 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 COPYRIGHT HOLDER 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.

Nlohmann-json

MIT License

Copyright (c) 2013-2022 Niels Lohmann

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Op2

BSD 2-Clause License - <http://www.opensource.org/licenses/bsd-license.php>

This file is part of the OP2 distribution.

Copyright (c) 2011, Gihan Mudalige, Istvan Reguly, Mike Giles, and others.
Please see the AUTHORS file in the main source directory for details.
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.
- * The name of Mike Giles may not 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 COPYRIGHT HOLDER 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.

OpenMesh

OpenMesh is licensed under the 3-clause BSD License. See the LICENSE file for the complete license.

```

/* ===== *
 * * *
 * OpenMesh *
 * Copyright (c) 2001-2015, RWTH-Aachen University *
 * Department of Computer Graphics and Multimedia *
 * All rights reserved. *
 * www.openmesh.org *
 * * *
 *-----*
 * This file is part of OpenMesh. *
 *-----*
 * * *
 * Redistribution and use in source and binary forms, with or without *
 * modification, are permitted provided that the following conditions *
 * are met: *
 * * *
 * 1. Redistributions of source code must retain the above copyright notice, *
 * this list of conditions and the following disclaimer. *
 * * *
 * 2. 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. *
 * * *
 * 3. Neither the name of the copyright holder nor the names of its *
 * 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 COPYRIGHT HOLDER *
 * 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.      *
*                                                                      *
* =====*\
```

Openssl

Apache License

Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Pcre2

PCRE2 LICENCE

PCRE2 is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Releases 10.00 and above of PCRE2 are distributed under the terms of the "BSD" licence, as specified below, with one exemption for certain binary redistributions. The documentation for PCRE2, supplied in the "doc" directory, is distributed under the same terms as the software itself. The data in the testdata directory is not copyrighted and is in the public domain.

The basic library functions are written in C and are freestanding. Also included in the distribution is a just-in-time compiler that can be used to optimize pattern matching. This is an optional feature that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS

Written by: Philip Hazel
Email local part: Philip.Hazel

Email domain: gmail.com

Retired from University of Cambridge Computing Service,
Cambridge, England.

Copyright (c) 1997-2021 University of Cambridge
All rights reserved.

PCRE2 JUST-IN-TIME COMPILATION SUPPORT

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2010-2021 Zoltan Herczeg
All rights reserved.

STACK-LESS JUST-IN-TIME COMPILER

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2009-2021 Zoltan Herczeg
All rights reserved.

THE "BSD" LICENCE

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 notices,
this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright
notices, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
- * Neither the name of the University of Cambridge 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 COPYRIGHT OWNER 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.

EXEMPTION FOR BINARY LIBRARY-LIKE PACKAGES

The second condition in the BSD licence (covering binary redistributions) does
not apply all the way down a chain of software. If binary package A includes
PCRE2, it must respect the condition, but if package B is software that
includes package A, the condition is not imposed on package B unless it uses
PCRE2 independently.

End

Pegt1-2

The MIT License (MIT)

Copyright (c) 2007-2020 Dr. Colin Hirsch and Daniel Frey

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Pkgconf

Copyright (c) 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018
pkgconf authors (see AUTHORS file in source directory).

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

This software is provided 'as is' and without any warranty, express or implied. In no event shall the authors be liable for any damages arising from the use of this software.

Proj

All source, data files and other contents of the PROJ package are available under the following terms. Note that the PROJ 4.3 and earlier was "public domain" as is common with US government work, but apparently this is not a well defined legal term in many countries. Frank Warmerdam placed everything under the following MIT style license because he believed it is effectively the same as public domain, allowing anyone to use the code as they wish, including making proprietary derivatives.

Initial PROJ 4.3 public domain code was put as Frank Warmerdam as copyright holder, but he didn't mean to imply he did the work. Essentially all work was done by Gerald Evenden.

Copyright information can be found in source files.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Pugixml

MIT License

Copyright (c) 2006-2022 Arseny Kapoulkine

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Qt5-base

GNU LESSER GENERAL PUBLIC LICENSE

The Qt Toolkit is Copyright (C) 2016 The Qt Company Ltd.
Contact: <http://www.qt.io/licensing/>

You may use, distribute and copy the Qt Toolkit under the terms of GNU Lesser General Public License version 3, which is displayed below. This license makes reference to the version 3 of the GNU General Public License, which you can find in the LICENSE.GPL3 file.

GNU LESSER GENERAL PUBLIC LICENSE Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <<http://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies of this licensedocument, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or
- b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Qwt

Qwt License

Version 1.0, January 1, 2003

The Qwt library and included programs are provided under the terms of the GNU LESSER GENERAL PUBLIC LICENSE (LGPL) with the following exceptions:

1. Widgets that are subclassed from Qwt widgets do not constitute a derivative work.
2. Static linking of applications and widgets to the Qwt library does not constitute a derivative work and does not require the author to provide source code for the application or widget, use the shared Qwt libraries, or link their applications or widgets against a user-supplied version of Qwt.

If you link the application or widget to a modified version of Qwt, then the changes to Qwt must be provided under the terms of the LGPL in sections 1, 2, and 4.

3. You do not have to provide a copy of the Qwt license with programs that are linked to the Qwt library, nor do you have to identify the Qwt license in your program or documentation as required by section 6 of the LGPL.

However, programs must still identify their use of Qwt. The following example statement can be included in user documentation to satisfy this requirement:

```
[program/widget] is based in part on the work of
the Qwt project (http://qwt.sf.net).
```

GNU LESSER GENERAL PUBLIC LICENSE Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You

can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free

library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion

of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
- b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if

the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

- c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
- d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
- e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR

PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
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., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
```

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.
```

```
<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
```

That's all there is to it!

Shapelib

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

```
Copyright (C) 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies
```


of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a

fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany

it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
- b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials

specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not

excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN

WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 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
Library General Public License for more details.
```

```
You should have received a copy of the GNU Library General Public
License along with this library; if not, write to the Free
Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
```

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.
```

```
<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
```

That's all there is to it!

SQLite3

SQLite is in the Public Domain.
<http://www.sqlite.org/copyright.html>

Tecio

Tecplot, Inc. LICENSE AGREEMENT FOR Tecplot's TecIO ("TecIO")

TecIO is a software library provided by Tecplot, Inc. to enable software developed by others to write data in Tecplot's proprietary binary file

formats, .plt and .szplt, and to read Tecplot binary data in .plt and .szplt format. TecIO is included with Tecplot 360 EX and may also be downloaded from <http://www.tecplot.com/downloads/tecio-library/>.

This license applies to versions of the TecIO library distributed with Tecplot 360 EX 2016 R2 and later and covers both the serial and parallel (MPI) versions of the library.

1. This LICENSE AGREEMENT is between Tecplot, Inc. ("Tecplot"), and the Individual or Organization ("Licensee") accessing and otherwise using TecIO software in source or binary form and its associated documentation.
2. Licensee acknowledges that this is only a limited nonexclusive license. Tecplot is and remains the owner of all titles, rights, and interests in TecIO Software. Title to TecIO and all copies thereof remain with Tecplot. The Materials are copyrighted and are protected by United States copyright laws and international treaty provisions. Licensee will not remove any copyright notice from the Materials. Tecplot does not grant any express or implied right to you under Tecplot patents, copyrights, trademarks, or trade secret information.
3. Subject to the terms and conditions of this License Agreement, Tecplot hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use TecIO alone or in any derivative version, provided, however, that Tecplot's License Agreement and Tecplot's notice of copyright, i.e., "Copyright © 1988-2016 Tecplot, Inc. All rights reserved worldwide." are retained in TecIO alone or in any derivative version prepared by Licensee.
4. In the event Licensee prepares a derivative work that is based on or incorporates TecIO or any part thereof, and wants to publish the derivative work as provided herein, Licensee hereby agrees to provide to all end users of any such work a brief summary of all changes made to TecIO, and to convey to Tecplot a copy of the modified TecIO source code within 30 days after publication of any work containing any such changes.
5. Tecplot is making TecIO available to Licensee on an "AS IS" basis. NO OTHER WARRANTIES. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, TECPLOT, INC. AND ITS SUPPLIERS DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NONINFRINGEMENT, WITH REGARD TO THE SOFTWARE, AND THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES. APART FROM THE WARRANTIES STATED ABOVE, TECPLOT, INC. MAKES NO WARRANTY THAT THE SOFTWARE OR SERVICES WILL: MEET REQUIREMENTS; PROVIDE UNINTERRUPTED, TIMELY, SECURE, OR ERROR-FREE, USE OF COMPUTERS OR NETWORKS; PROVIDE RESULTS WHICH ARE ACCURATE OR RELIABLE; MEET EXPECTATIONS, OR; CORRECT ANY ERRORS IN THE SOFTWARE. TECPLOT, INC. SHALL NOT BE RESPONSIBLE FOR MISUSE OF THE SOFTWARE OR ANY LOSS OF DATA. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS, WHICH VARY FROM STATE/JURISDICTION TO STATE/JURISDICTION.
6. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL TECPLOT, INC. OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE OR THE PROVISION OF OR FAILURE TO PROVIDE SUPPORT SERVICES, EVEN IF TECPLOT, INC. HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS SHALL BE TRUE EVEN IN THE EVENT OF THE FAILURE OF AN AGREED REMEDY. IN ANY CASE, TECPLOT, INC.'S ENTIRE LIABILITY FOR CLAIMS ARISING OUT OF USE OF THE SOFTWARE, SERVICE OR ARISING FROM ANY PROVISION OF THIS AGREEMENT SHALL BE LIMITED TO THE AMOUNT ACTUALLY PAID BY LICENSEE FOR THE SOFTWARE OR SERVICE COMPLAINED OF. BECAUSE SOME STATES AND JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.
7. This License Agreement will automatically terminate upon a material breach of its terms and conditions.
8. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between Tecplot and Licensee. This

License Agreement does not grant permission to use Tecplot trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

9. By copying, installing or otherwise using TecIO, Licensee agrees to be bound by the terms and conditions of this License Agreement.

Tclap

Copyright (c) 2003 Michael E. Smoot
Copyright (c) 2004 Daniel Aarno
Copyright (c) 2017 Google Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Tiff

Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Utfcpp

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.

Vcpkg

Copyright (c) Microsoft Corporation

All rights reserved.

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Vtk

/*****

Program: Visualization Toolkit
Module: Copyright.txt

Copyright (c) 1993-2015 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.

=====*/

VtkexodusII

Copyright (c) 2005-2017 National Technology & Engineering Solutions of Sandia, LLC (NTESS). Under the terms of Contract DE-NA0003525 with NTESS, the U.S. Government retains certain rights in this software.

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 the name of NTESS nor the names of its 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 COPYRIGHT OWNER 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.

Zlib

ZLIB DATA COMPRESSION LIBRARY

zlib 1.2.12 is a general purpose data compression library. All the code is thread safe. The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files <http://tools.ietf.org/html/rfc1950> (zlib format), [rfc1951](http://tools.ietf.org/html/rfc1951) (deflate format) and [rfc1952](http://tools.ietf.org/html/rfc1952) (gzip format).

All functions of the compression library are documented in the file `zlib.h` (volunteer to write man pages welcome, contact zlib@gzip.org). A usage example of the library is given in the file `test/example.c` which also tests that the library is working correctly. Another example is given in the file `test/minigzip.c`. The compression library itself is composed of all source files in the root directory.

To compile all files and run the test program, follow the instructions given at the top of `Makefile.in`. In short `./configure; make test`, and if that goes well, `make install` should work for most flavors of Unix. For Windows, use

one of the special makefiles in win32/ or contrib/vstudio/ . For VMS, use make_vms.com.

Questions about zlib should be sent to <zlib@zip.org>, or to Gilles Vollant <info@winimage.com> for the Windows DLL version. The zlib home page is <http://zlib.net/> . Before reporting a problem, please check this site to verify that you have the latest version of zlib; otherwise get the latest version and check whether the problem still exists or not.

PLEASE read the zlib FAQ http://zlib.net/zlib_faq.html before asking for help.

Mark Nelson <markn@ieee.org> wrote an article about zlib for the Jan. 1997 issue of Dr. Dobbs's Journal; a copy of the article is available at <http://marknelson.us/1997/01/01/zlib-engine/> .

The changes made in version 1.2.12 are documented in the file ChangeLog.

Unsupported third party contributions are provided in directory contrib/ .

zlib is available in Java using the java.util.zip package, documented at <http://java.sun.com/developer/technicalArticles/Programming/compression/> .

A Perl interface to zlib written by Paul Marquess <pmqs@cpan.org> is available at CPAN (Comprehensive Perl Archive Network) sites, including <http://search.cpan.org/~pmqs/IO-Compress-Zlib/> .

A Python interface to zlib written by A.M. Kuchling <amk@amk.ca> is available in Python 1.5 and later versions, see <http://docs.python.org/library/zlib.html> .

zlib is built into tcl: <http://wiki.tcl.tk/4610> .

An experimental package to read and write files in .zip format, written on top of zlib by Gilles Vollant <info@winimage.com>, is available in the contrib/minizip directory of zlib.

Notes for some targets:

- For Windows DLL versions, please see win32/DLL_FAQ.txt
- For 64-bit Irix, deflate.c must be compiled without any optimization. With -O, one libpng test fails. The test works in 32 bit mode (with the -n32 compiler flag). The compiler bug has been reported to SGI.
- zlib doesn't work with gcc 2.6.3 on a DEC 3000/300LX under OSF/1 2.1 it works when compiled with cc.
- On Digital Unix 4.0D (formerly OSF/1) on AlphaServer, the cc option -std1 is necessary to get gzprintf working correctly. This is done by configure.
- zlib doesn't work on HP-UX 9.05 with some versions of /bin/cc. It works with other compilers. Use "make test" to check your compiler.
- gzdopen is not supported on RISCOS or BEOS.
- For PalmOs, see <http://palmzlib.sourceforge.net/>

Acknowledgments:

The deflate format used by zlib was defined by Phil Katz. The deflate and zlib specifications were written by L. Peter Deutsch. Thanks to all the people who reported problems and suggested various improvements in zlib; they are too numerous to cite here.

Copyright notice:

(C) 1995-2022 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code. We make all contributions to and distributions of this project solely in our personal capacity, and are not conveying any rights to any intellectual property of any third parties.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes. Please read the FAQ for more information on the distribution of modified source versions.

Zstd

ZSTD is dual licensed under BSD and GPLv2.

BSD License

For Zstandard software

Copyright (c) 2016-present, Facebook, Inc. 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 the name Facebook nor the names of its 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 COPYRIGHT HOLDER 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.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not

covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three

years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other

circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
```

```
This program 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 General Public License for more details.
```

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

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.2 Open source licenses

Apache-2.0

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

FTL

The FreeType Project LICENSE

2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction

=====

The FreeType Project is distributed in several archive packages; some of them may contain, in addition to the FreeType font engine, various tools and contributions which rely on, or relate to, the FreeType Project.

This license applies to all files found in such packages, and which do not fall under their own explicit license. The license affects thus the FreeType font engine, the test programs, documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

- o We don't promise that this software works. However, we will be interested in any kind of bug reports. ('as is' distribution)
- o You can use this software for whatever you want, in parts or full form, without having to pay us. ('royalty-free' usage)
- o You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. ('credits')

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

```
""  
Portions of this software are copyright © <year> The FreeType  
Project (www.freetype.org). All rights reserved.  
""
```

Please replace <year> with the value from the FreeType version you actually use.

Legal Terms

=====

0. Definitions

Throughout this license, the terms 'package', 'FreeType Project', and 'FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the 'FreeType Project', be they named as alpha, beta or final release.

'You' refers to the licensee, or person using the project, where 'using' is a generic term including compiling the project's source code as well as linking it to form a 'program' or 'executable'. This program is referred to as 'a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive.

If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- o Redistribution of source code must retain this license file (FTL.TXT) unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.
- o Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: `FreeType Project', `FreeType Engine', `FreeType library', or `FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts

There are two mailing lists related to FreeType:

o freetype@nongnu.org

Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

o freetype-devel@nongnu.org

Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at

<https://www.freetype.org>

GPL-2.0

GNU GENERAL PUBLIC LICENSE Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software

patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of

this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the

original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY

FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
```

```
This program 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 General Public License for more details.
```

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

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GPL-3.0

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<https://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we

stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an

implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you

receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided,

in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims

owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you

to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the

Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
```

```
This program 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 General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see [<https://www.gnu.org/licenses/>](https://www.gnu.org/licenses/).

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read [<https://www.gnu.org/licenses/why-not-lgpl.html>](https://www.gnu.org/licenses/why-not-lgpl.html).

LGPL-2.1

GNU LESSER GENERAL PUBLIC LICENSE Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts

as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked

with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are

prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free

Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
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
```

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.
```

```
<signature of Ty Coon>, 1 April 1990
```

Ty Coon, President of Vice

That's all there is to it!

MPL-2.0

Mozilla Public License Version 2.0
=====

1. Definitions -----

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

means any form of the work other than Source Code Form.

1.7. "Larger Work"

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having

made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- (b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- (a) for any code that a Contributor has removed from Covered Software; or
- (b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- (c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- (a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- (b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to

the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

```
*****
*
* 6. Disclaimer of Warranty
* -----
*
* Covered Software is provided under this License on an "as is"
* basis, without warranty of any kind, either expressed, implied, or
* statutory, including, without limitation, warranties that the
* Covered Software is free of defects, merchantable, fit for a
* particular purpose or non-infringing. The entire risk as to the
* quality and performance of the Covered Software is with You.
* Should any Covered Software prove defective in any respect, You
* (not any Contributor) assume the cost of any necessary servicing,
*
```

```
* repair, or correction. This disclaimer of warranty constitutes an *
* essential part of this License. No use of any Covered Software is *
* authorized under this License except under this disclaimer. *
*
```

```
*****
```

```
*****
```

* 7. Limitation of Liability *

```
* ----- *
```

```
*
* Under no circumstances and under no legal theory, whether tort *
* (including negligence), contract, or otherwise, shall any *
* Contributor, or anyone who distributes Covered Software as *
* permitted above, be liable to You for any direct, indirect, *
* special, incidental, or consequential damages of any character *
* including, without limitation, damages for lost profits, loss of *
* goodwill, work stoppage, computer failure or malfunction, or any *
* and all other commercial damages or losses, even if such party *
* shall have been informed of the possibility of such damages. This *
* limitation of liability shall not apply to liability for death or *
* personal injury resulting from such party's negligence to the *
* extent applicable law prohibits such limitation. Some *
* jurisdictions do not allow the exclusion or limitation of *
* incidental or consequential damages, so this exclusion and *
* limitation may not apply to You. *
```

```
*****
```

8. Litigation

```
-----
```

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

```
-----
```

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

```
-----
```

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a

modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

Zlib

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.