Hochschule Karlsruhe Technik und Wirtschaft

University of Applied Sciences

Faculty of Geomatics

DEVELOPMENT OF A GVSIG TUTORIAL FOR STUDENTS

GIS PROJECT WORK

Authors: William Banahene

Dilip Biswas

Nicolai Guth

Dorothea Heim

Janeta Turka

Supervisors: Prof. Dr. H. Saler

Dipl.-Ing. C. Stern

Karlsruhe, 2011

CONTENTS

INTRODUCTION /BY W. BANAHENE/

The University of Applied Sciences in Karlsruhe intends to develop an e-learning application for the open source GIS software gvSIG.

The e-learning application would be used by students of the university who are knowledgeable in GIS software.

Scope of work

Inception activities: The inception of the project began with group meetings on how to execute the project successful within the time frame given. Group members were appointed positions concerning the project.

Analysis of data in gvSIG: The platform for the task was set by installing gvSIG. Data used for the tasks were from GIS I and GIS II assignments from first semester. This task was divided in five subparts, one for each group member.

Implementation of gvSIG results in Flash: The tutorial was designed according to the standards set by the group in Adobe Flash CS4. Also implementation in flash was divided according to individual subparts.

Loading of Flash files, data and self-evaluation in Moodle: The finished shockwave flash movie (.swf) file and the necessary data were uploaded in Moodle. Two to three questions based on the subparts of the tutorial were created in the Moodle question database for the self-evaluation.

E-learning

E-learning has many various definitions, though the main point is to see the difference between e-learning and online education. Learning is just one element of education, so term “online education” should cover a much broader range of services than the term “e-learning”. E-learning focuses more on the content, while online education cover the whole range of educational services (Rekkedal, 2003).

The main e-learning definition that was chosen is the one from Dichanz, professor of education at the German FernUniversität:

E-learning is the collection of teaching and information packages in further education which is available at any time and any place and is delivered to the learners *electronically. They contain units of information, self-testing batteries and tests, which allow a quick self-evaluation for quick placement (Rekkedal, 2003).*

Open source software

Open source software is a type of "free" software to be accessed, used or modified by their user groups and developers. There are many similar terms to describe this kind of software, such as “free software”, “libre software”, “open software”, etc. One key feature to distinguish open source software from other types (such as proprietary software and shareware) is their “free software licenses”, which explicitly define the legal rights to users with freedoms to run, study, change, redistribute, and access the source codes of the licensed software.

All open source software is required to be “licensed”. The procedure of implementing “free software licenses” is necessary to protect their users’ legal rights and to ensure the freedoms of the software.

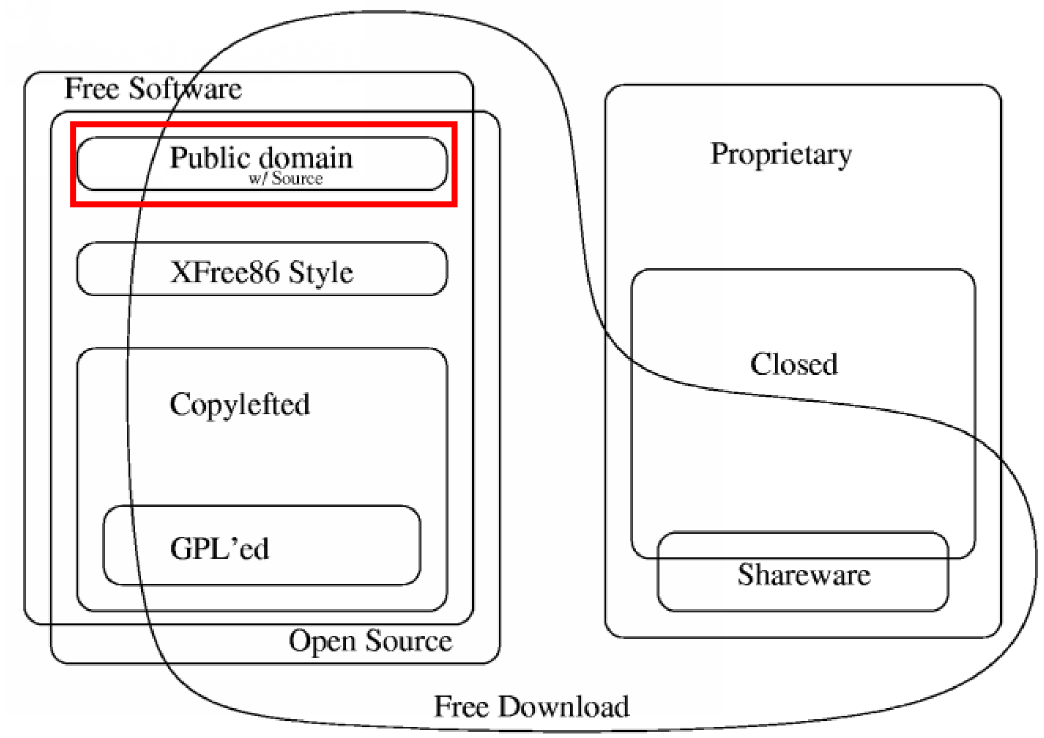


Figure 1, Categories of Free and Non-Free Software by Chao-Kuei Hung.

The red box represents the new boundary of public domain software. Because some public domain software may have no copyright restrictions or licenses, the public domain software has been altered to both include and exclude the open source domain. In GIS software, GRASS GIS is a famous example of public domain software. Public domain software can be re-packaged and sold as proprietary software or licensed as open source software (Valdes, 2008).

Open source software has been categorized into five domains for GIS education:

1. Basic desktop GIS
2. Remote sensing software
3. 3D visualization tools
4. Others (including Web mapping servers and clients, spatial programming tools and libraries, and spatial databases).

Details of open source software have been illustrated for Basic desktop GIS.

Basic desktop GIS

Basic desktop GIS software can provide basic GIS functions, such as data input, map display, spatial query, attribute query and spatial analysis. Most open source desktop GIS software can be installed on multiple operating systems, such as Windows, MacOS, and Linux.

However, one common problem in open source desktop GIS software is the lack of advanced cartographic functions and symbolization. Of course, some offer the capability to export the mapping results as a scalable vector graphic (SVG) format to subsequently edit it in OpenOffice Draw (another open source software) or utilize cartographic support with InkScape.

Examples of Basic dektop GIS softwares are:

• KOSMO (http://www.opengis.es/) (download file size: 108MB - including JRE package) (Available OS: Windows and Linux)

• gvSIG (http://www.gvsig.com) (file size: 89MB) (Available OS: Windows, Linux and MacOS X)

• uDig (http://udig.refractions.net/) (file size: 94 MB, Available OS: Windows, Linus, and MacOS).

• Quantum GIS (QGIS) (http://www.qgis.org/) (http://www.qgis.org/) (file size: 89MB, 1.0.2 LTS Windows Standalone version, available OS: Windows, UNIX, Linux, and MacOS).

• GRASS (http://grass.osgeo.org/) (file size: 88MB, available OS: Windows, MacOS, and Linux).

AIM /BY D. BISWAS/

ArcGIS supports the pioneer role to edit, analyze and model with the spatial data. As gvSIG is open source software, the consumers can easily download from the internet and use it free. Application of the software with multiple languages is also the benefit of gvSIG. With the recent applications of gvSIG in terms of network design, transportation modeling, data mining, settlement and the usage of landuse data analysis, gvSIG is nowadays a best tool for data analysis. gvSIG is a Geographic Information System (GIS), that support the desktop application designed for capturing, storing, handling, analyzing and deploying any kind of referenced geographic information in order to solve complex management and planning problems, user-friendly interface, being able to access the most common formats, both vector and raster data. It features a wide range of tools for working with geographic-like information (query tools, layout creation, geoprocessing, networks).

The aim of the project includes developing a tutorial (e-learning) for the students, who have the basic ideas about GIS. By applying the tutorial, the students will be able to work with gvSIG environment and can be able to perform the self-evaluation themselves from the Moodle platform, so that they can judge their level of knowledge about the software. The basics of gvSIG include the preparation of Map Layout, Geoprocessing, Georeferencing, Digitizing and Digital Elevation Models (DEMs).

In order to fulfill the requirement of the project, a group of five members were formed with consecutive group meetings, weekly reports; weekly meetings with the network learning were performed in order to evaluate the activities among the group members. The activities were divided among the group coordinator, the other members tried to complete the temporal task and finally with the project management tools, the activity records were stored.

The main objective of the tutorial was also outlined as follows:

- easy to understand;
- informative;
- well designed;
- interactivity by using Flash;
- self-evaluation possibilities after completing the tutorial;
- based in moodle;
- ready to use;
- after completing the tutorial, the user should be able to work with gvSIG.

THEORY AND PEDAGOGY OF E-LEARNING /BY J. TURKA/

When creating an e-learning tutorial, the pedagogical background of e-learning solutions and e-teaching itself should be considered.

A web-accessible literature study about pedagogical background of e-learning solutions was carried out. There are many publications concerning e-learning (named also: digital-learning, multimedia-learning, computer-based learning, web-learning, distance-education, online-learning, online-education etc.) and its place in nowadays educational and learning system.

Based on some of them, short overview was made to explain the basic ideas of e-learning.

The „technology" within digital learning techniques should be subservient to the „learning". Technology is a tool and a medium, but not the message itself (Buzzetto-More, 2007). So this shows that there should not be put main focus on technology but the content you broadcast to the learners via this technology. In this case to all the information needed for user to get acquainted with the software.

Definition

E-learning has many various definitions, though the main point is to see the difference between e-learning and online education. Learning is just one element of education, so term "online education" should cover a much broader range of services than the term "e-learning". E-learning focuses more on the content, while online education cover the whole range of educational services (Rekkedal, 2003).

The main e-learning definition that was chosen is the one from Dichanz, professor of education at the German FernUniversität:

> *"E-learning is the collection of teaching and information packages in further education which is available at any time and any place and is delivered to the learners electronically. They contain units of information, self-testing batteries and tests, which allow a quick self-evaluation for quick placement (Rekkedal, 2003)".*

Approaches which are purely prescriptive, i.e. "how to" models are inappropriate, because human behaviour is not predictable. Different learners may be taught by the same instructional method, yet their performances may vary considerably. Learning strategies vary as well (Buzzetto-More, 2007). Even though it is hard to predict different people learning habits, in e-learning courses one common way suitable for any learner should be found.

Different theories, models and forms for e-learning

There exist different theories of effective e-learning. Some of them are joined in special e-learning models and divided in different forms of e-learning concerning what should be taught via e-learning. There are different approaches how to build e-learning course depending on the context. For example, it is not possible to build up e-learning course that teaches biology in the same way used for teaching any software.

The main models and forms of e-learning concerned in this project will be shortly described here.

The Hexa-C Metamodel (HCMm)

The Hexa-C Metamodel (HCMm**)** is a synthesis of contemporary learning theories and practices, whose six elements: cognitive learning, constructivism, components, creativity, customization and collaborative learning should play a role in the design and development of e-learning.

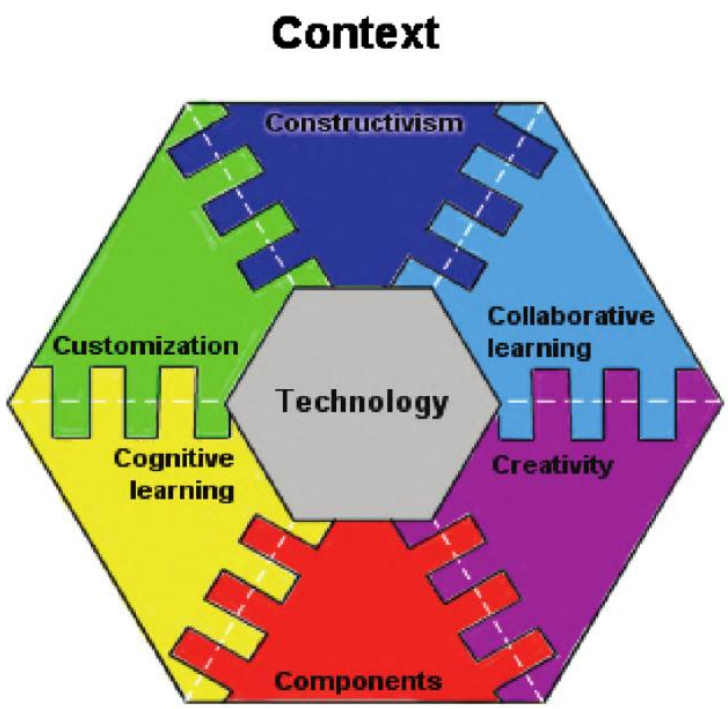


Figure 2, The framework of the Hexa-C Metamodel (Buzzetto-More, 2007)

It is an approach that integrates six selected interrelated concepts from contemporary learning theory into a framework that serves both as design aid and as an evaluation approach for investigating existing resources from the perspective of learning theory.

It is not the idea that every e-learning resource system should conform to all six HCMm elements. Three of the Cs: constructivism, cognitive learning and components are mainly theoretical aspects, while the others: collaborative learning, creativity and customization are practical methods used by educators to foster effective and affective learning (Buzzetto-More, 2007).

Three-component e-learning model

Three-component e-learning model contains that three key components are working collectively to foster meaningful learning and interaction. They form an iterative relationship in which pedagogical models or constructs grounded in the situated cognition view inform the design of E-learning by leading to the specification of instruction and learning strategies that are subsequently enabled or enhanced through the use of learning technologies. Furthermore, as learning technologies become ubiquitous and new technologies continue to emerge bringing forth new affordances, pedagogical practices and social structures are transformed. Therefore, three-component model implies a transformative interaction affecting E-learning. Educators and instructional designers can think of this model as a theory-based or grounded design framework that guides the design of E-learning (Dabbagh, 2005).

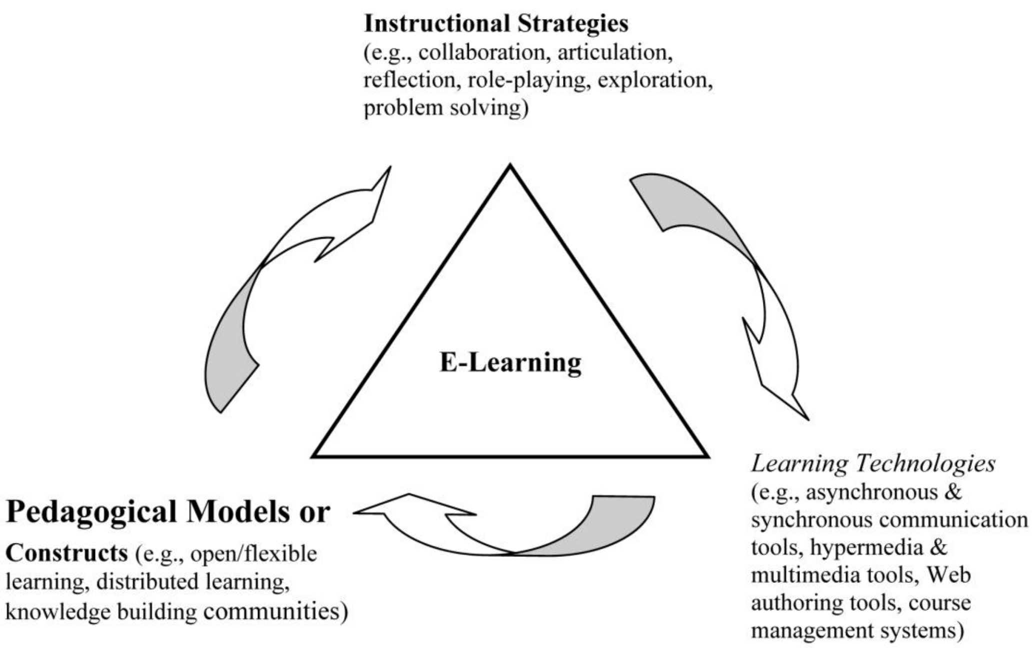


Figure 3, Three-component e-learning model (Dabbagh, 2005)

Forms of e-learning

Forms of e-learning vary according to the use and role of technology as a tutor, an environment or a tool. Zucchermaglia distinguishes between full and empty technologies. A full technology teaches defined content. It contains information to be transferred interactively to learners; for example, a computer-based tutorial. An empty technology, by contrast, is not an instructional system, but a tool or shell that can accept content. It supports exploration and construction, e.g. searches and generation of products using the internet as well as use by learners of commercial software packages for manipulation, documentation, presentation and communication (Buzzetto-More, 2007).

Design

When it comes to full technology tutorials, the main aspect described in most pedagogical literature is the importance of design, because e-learning is very visible. Everything from screen colour to content accuracy to the types of practices is readily available for scrutiny. So design is the main tool to make e-learning effective and appropriate for self-learning.

E-learning definition of design addresses:

The what: training delivered in digital form;

The how: content and instructional methods to help learn the content;

The why: to improve organizational performance by building job-relevant knowledge and skills in workers/or students who may use this in further career (Clark, 2002).

In time the approaches used in e-learning have changed. Technology-centred approach has abandoned in favour of a learner-centred approach. Based on researches have been founded out that it is not the medium that causes learning. Rather are the design of the lesson itself and the best use of instructional methods that make the difference. A learner-centred approach suggests that lessons that accommodate human learning processes regardless of the media involved are designed (Clark, 2002).

In good design there is a distinction between three components – instructional methods, instructional media and media elements:

Instructional methods include the use of techniques such as examples, practise exercises, simulations and analogies;

Instructional media are the delivery agents that contain the content and the instructional methods including computers, workbooks and even instructors.

Media elements refer to the text, graphics and audio used to present content and instructional methods (Clark, 2002).

There have been series of controlled experiments by Richard Meyer and his colleagues at the University of California at Santa Barbara on how to best use audio, text and graphics to optimize learning in multimedia. Based on the psychological aspects, six media element principles can be defined on Mayer's work:

The multimedia principle: adding graphics to words can improve learning;

The contiguity principle: placing text near graphics improves learning;

The modality principle: explaining graphics with audio improves learning;

The redundancy principle: explaining graphics with audio and redundant text can hurt learning;

The coherence principle: using gratuitous visuals, text and sounds can hurt learning;

The personalization principle: use conversational tone and pedagogical agents to increase learning (Clark, 2002).

Pedagogical aspects applied in gvSIG e-learning tutorial

gvSIG e-learning has a form of full technology – web-based tutorial for new software learning with previous knowledge in the field of GIS and other GIS software.

The main instructional methods will be examples, practise exercises, some explanation texts; the instructional media in our case is the Moodle environment available on the web and of course computers with installed software; the media elements will be text, graphics, animations on them ensuring interactivity, small videos, combining with the guidance to the world-wide-web resources for each particular topic to not stick just on our experience and offer multiform guidance and promote self-learning.

Besides that we will also tried to stick to the Mayer's six principles to gain effective design and therefore also learning, so that our e-learning tutorial will be useful for student learning and they will not end like the guy in this cartoon:


"BLAAAH BLAH BLAH BLAH BLAH BLAAAAAH BLA.. CLICK NEXT..."
KLIK
"..CLICK NEXT.. BLAAAH BLAH BLAH BLAH BLAAAAAH BLA.. CLICK NEXT..."
KLIK KLIK
TOM, HOW'D YOU MAKE OUT WITH THAT E-LEARNING COURSE?
I FINISHED IT IN 43 MINUTES!
BUT WHAT WAS IT ABOUT?
UMMMM...... UHHHH.....IT WAS SOMETHING ABOUT "CLICK NEXT"
.

METHODOLOGY

gvSIG /by W. Banahene/

Description of software

In the year 2003, the Regional Council for Infrastructures and Transportation (CIT) called for tender for the development of software for the management of geographic information. The private enterprise Iver won the tender and is now developing the software together with the Valencian Government and the university Jaume I of Castellón. The software is being actualized and published regularly. gvSIG was developed by the European GIS community offering multiple language user interfaces. More than ten different languages (including Deutsch, Spanish, French and Chinese) can be selected to display the menus and tools.

gvSIG is an Open Source GIS software. Its name is a spanish abrievation that stands for 'Generalidad Valencia Sistéma de Información Geográfica'.

gvSIG is a Geographic Information System (GIS), that is, a desktop application designed for capturing, storing, handling, analyzing and deploying any kind of referenced geographic information in order to solve complex management and planning problems.

gvSIG is well known for its flexible GIS data input format. One can use various GIS data formats (both vector and raster) and online resources (such as WMS, WCS and WFS).

gvSIG has nice vector data editing functions. Users can easily digitize lines by snapping vertices to existing nodes and generate correct topology. With an easy-to-configure locator map, gvSIG immediately reveals where you are in your dataset. Some GIS professionals believe that gvSIG is becoming close to replacing ESRI ArcGIS software.

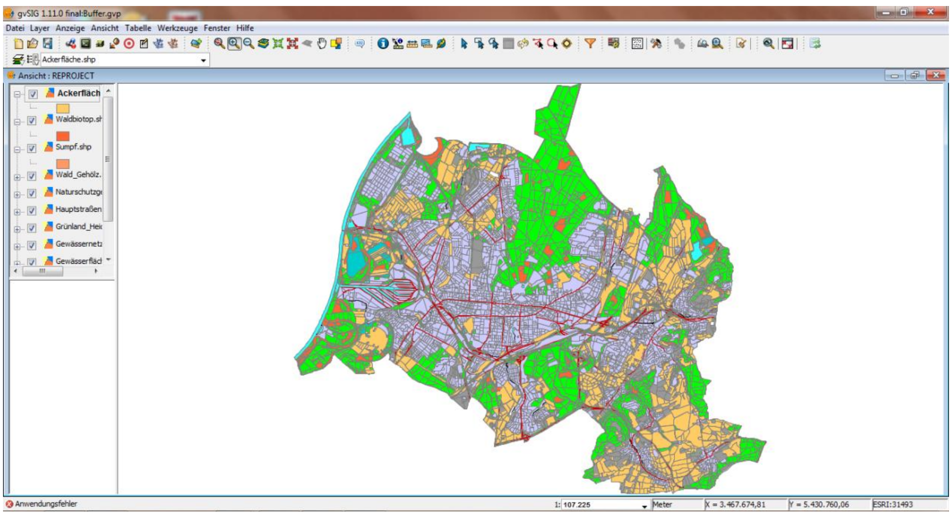


Figure 4: Example of gvSIG interface

Choice of software

gvSIG is developed with Java and is working under the GNU General Public License (GPL). Basically, it is working as a heavy client or desktop GIS.

- Portability: The software is compatible with different working systems, initially Linux and Windows.
- Modularity: gvSIG will be extensible in the future development. New functions may be integrated easily.
- Open Source: The source code of the software is freely available.
- Without licenses: the product is free of charge, without any quantity limits.
- Interoperable: The program must be able to read and write data formats of the leading proprietary software, like ArcView, Microstation or AutoCAD. During the work process, the format doesn't need to be changed. In its version 0.5, it is supporting various raster and vector data formats:
  - Vector formats: shapefile, dxf, dgn, ecw, MrSID
  - Raster formats: jpeg, jp2, tiff, geoTIFF, png, gif
- Subject to standards: It follows the most important standards and guidelines of the branch, e.g. the ones of the Open Geospatial Consortium (OGC) or the European Commission.
- Being available in several languages: Spanish, English UK, English USA, German, French, Italian, Portuguese, Portuguese-Brazilian, Russian, Chinese, Serbian, Swahili, Turkish, Czech, Polish, Romanian, Greek, Basque, Valencian, Gallego.

Following the future tendencies of the OGC, gvSIG is able to read remote data, as they are Web Map Service, Web Coverage Service and Web Feature Service.

Flash /by N. Guth/

To create the gvSIG tutorial, the Software Flash CS4 from Adobe is used. With Flash, it is possible to create interactive animations on a frame base where every frame can in turn consist of movie clips or other elements which consist of more than one frame. Flash has been chosen as there are flash players available for almost every computer platform. Furthermore, only little programming is needed in order to get visually appealing results. Most of the work can be done by putting graphical elements on a frame and add text. The programming is only used for navigation purposes and to control the playback of movies. A typical example of Action Script 3 code can be seen in Figure .

```
1  stop();
2  btn_prev.addEventListener(MouseEvent.MOUSE_DOWN, goto_step1);
3  btn_home.addEventListener(MouseEvent.MOUSE_DOWN, go_home);
4  btn_next.addEventListener(MouseEvent.MOUSE_DOWN, goto_step3);
5
6  btn_georef.addEventListener(MouseEvent.MOUSE_DOWN, goto_georeferencing);
7  function goto_georeferencing(event:MouseEvent):void {
8      gotoAndStop(1, "georef");
9  }
```

Figure 5, Sample of Action Script 3 code

The command in the first line tells Flash to stop the playback at the current frame. Without that, the next frame would automatically be displayed. Lines 2 to 6 define the functions that are called when the respective buttons are pressed in the animation. For this to work, the used name in the code and the instance name of the button in the frame has to be the same. Like in Java, a listener, in this case an EventListener, is added to each button. The argument consists of the type of event which is listened to and the function that is called when the event is triggered. The functions which are linked to the first three buttons are defined in another frame. This means it must be ensured that no functions with the same name exist in different frames or even different scenes. The function referred to in line 6 is defined in lines 7 to 9. The only command in this function tells flash to jump to frame 1 of the scene with the name georef.

Separation into scenes

Scenes in Adobe Flash are like separated parts of a project. As can be seen above, it is possible to jump from one scene to another. For the gvSIG tutorial, each individual tutorial is stored in its own scene. That is due to the fact that each tutorial was developed by a member of the group and these individual Flash files were compiled into a single final Flash file at the end. To finish the compilation, it must be made sure that function names and names for Flash symbols like buttons or movie clips are unique throughout the complete project. The compilation itself is done by copying all frames of all layers from one Flash file and pasting them to a new scene in the final file. After that, the layers must be truncated by hand as they all get the same length after being pasted. The last step is to update the code of the home buttons which have to point to the menu and the code of the menu itself to point to the individual tutorials.

Creation of screen capture videos

To show more complicated actions in the tutorial, short screen capture videos are used. These videos are created with the software CamStudio and converted to flv videos with a tool named Riva FLV Encoder. Both programs are available for free on the internet, the web-links can be found in the references list. With CamStudio, it is possible to define an area on the screen which is recorded and create a video from this. The video must be converted to another data-format as Flash is not able to open the file directly. This is done with the software Riva FLV Encoder. The program recodes the video according to the settings the user takes. In this case, a resolution of 640x480 pixels and a framerate of 15 frames per second with a bitrate of 1600 bits per second are chosen. As the videos do not contain any sound, the encoding of sounds is deactivated. The resolution is set to 640x480 pixels because the stagesize of the Flash-project is set to 800x600 pixels to fit into the moodle screen and the video has to fit on a frame with control elements and text around it. Riva FLV Encoder is used as all other tested programs, including the Adobe Media Encoder CS4 which comes along with Flash CS4, are not able to produce a working flv file from the CamStudio video.

To have a common look in all tutorials, some standards were set regarding the font and fontsize for headings, subheadings and the text. Also, all the tutorials use the same graphics for buttons, for example for navigation or the question- and exclamation marks. These elements were distributed among the group via a Flash file.

Moodle /by N. Guth/

Moodle is an open source online course managing system (CMS). It can be downloaded and used for free from www.moodle.org and thus is widespreaded and has a big community. To set up a moodle service, only a webserver with php-support and access to a database server are necessary. For windows, a complete XAMPP installation package is available with which it is very easy to facilitate moodle. This XAMPP installation consists of Apache as webservice, a php-interpreter for the website and MySQL as database server. After the installation of the environment, where the steps are depending on the used operating system, moodle can be “started” via an internet browser. On Windows, it must be ensured that no other programs use TCP port number 80 as this is used by the webserver. For example, Skype prevents XAMPP from working as it uses this port by default. The first time it is started, an administrator account has to be created.

Within moodle, many modules are available with which a big variety of e-learning scenarios can be covered. For the gvSIG Tutorial, a new course was created. In this course, the flash file with the menu and the tutorials was added as embedded file. The data needed for each tutorial were stored in folders. This way, the user can individually download the data for the currently worked on tutorial. Figure shows the main view of the course after login.

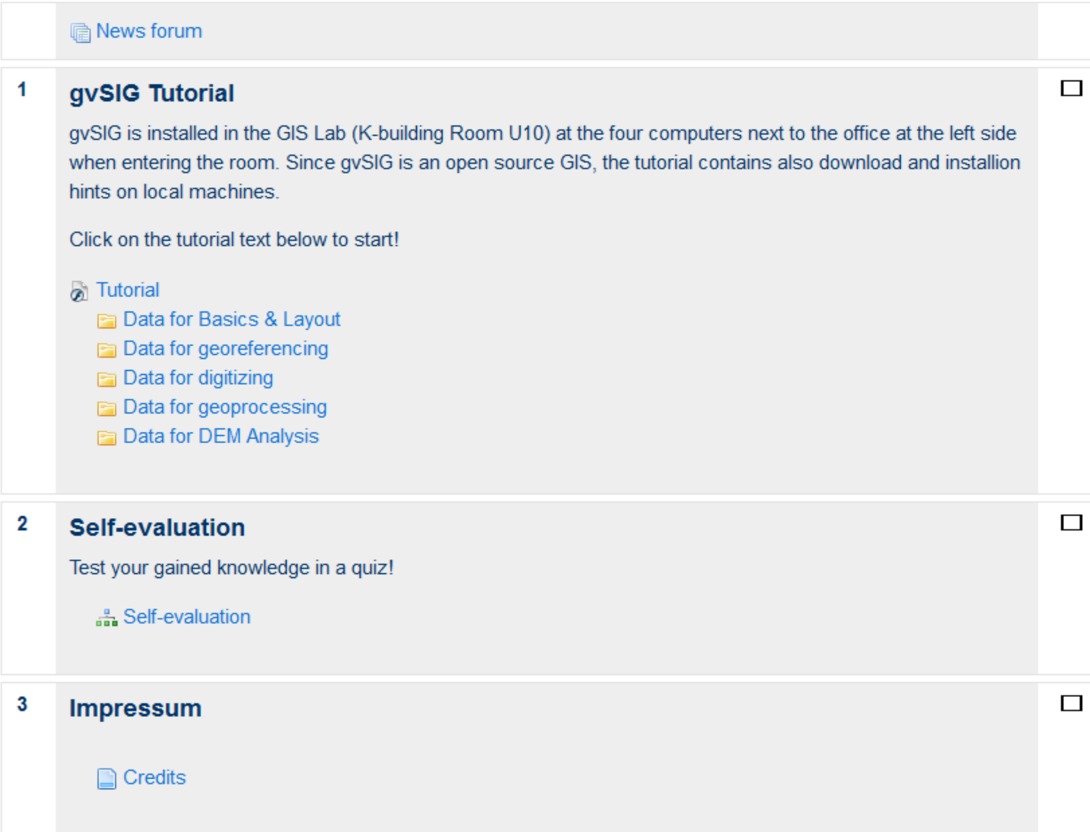


Figure 6, main view of the moodle course

The moodle quiz module

For self-examination, the Moodle's built-in Quiz-module is used. With this module, it is possible to create a question bank where all question-objects are stored. A question object consists of a question and, depending on its type, different possible answers which can be graded individually. For the gvSIG self evaluation, true - false, multiple choice and short answers (sentences or words) are allowed as answers. The order of possible answers is shuffled automatically for multiple choice and true – false questions. For multiple choice questions, it is possible to set multiple correct answers. In this case, all correct radio buttons must be ticked in order to gain 100% of the points. When creating short answer questions, more than one answer can be defined. Each of them can result in different grades. For example, if Geographic Information System is the 100% answer, Geo Information System can be graded with 75%. The result of the quiz can be seen by the course-tutor and statistics are available.

It is possible to setup a quiz in a lot of different ways. For example, the time the student has to finish the quiz or the number of attempts can be set. As the focus of the gvSIG quiz is on self evaluation of the student, there are no restrictions regarding time or number of attempts. There is one single quiz which consists of multiple pages where every page has multiple questions. The questions on each page belong to one part of the tutorial and have been created by the group member which worked on the respective tutorial. The order of the question pages follows the order of the tutorials in the menu. Figure 7 shows the editing windows of the gvSIG quiz. The separation into single pages per topic can be seen in the middle, the available questions from the question bank are on the right side. In this view, the questions can be edited, new ones can be created and added to the quiz and the order of the questions can be set.

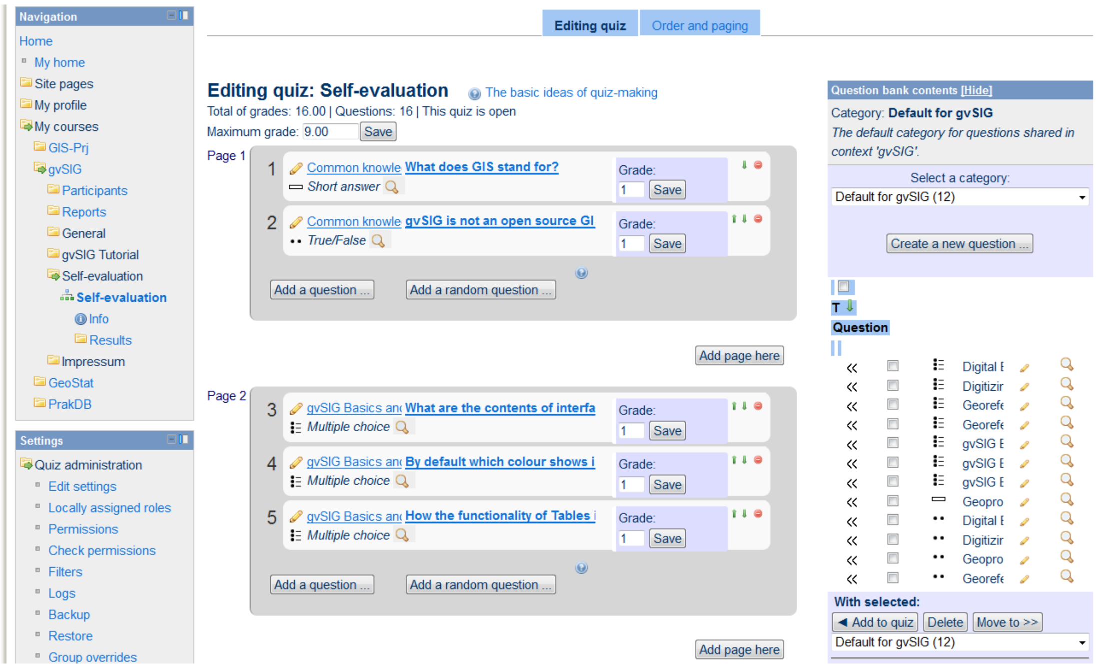


Figure 7, editing page of the gvSIG quiz in moodle

Minor problems

A problem with the embedded Flash is that the representation is depending on the used Web browser and the installed software. With Mozilla-based browsers, no problems appeared, neither at the pool nor at the group member's computers. With Microsoft Internet Explorer, the Flash animation is not displayed in the browser but can only be downloaded. On the other hand, it is only possible to upload files to Moodle with Internet Explorer. With Mozilla, an error occurs and the file is not uploaded.

PROJECT CONTENT

Five topics in gvSIG were selected and tasked as follows:

- Map Layout, Export/Import, Editing Tables (D. Biswas)
- Geo-referencing, Projections & Transformations (J. Turka)
- Geo-processing of vector data (W. Banahene)
- Geo-processing of raster data / DEM (D. Heim)
- Digitizing & Topology (N. Guth)

In addition two more parts to the final tutorial were introduced:

- Tutorial help
- Installation & Downloads

in which the short description about main buttons/interactivity of the tutorial was described as well as short guidelines of installation.

Besides e-learning tutorial in the Moodle environment there is a possibility for user to check his knowledge about gvSIG with self-evaluation test, which consists of questions from each topic stored in question database.

Provided data for completing tutorial tasks are also available for download from Moodle platform. Since the spatial data occupies big storage space, they are stored in separate folders for each tutorial topic. This allows also users, who want to fulfill just separate parts of the tutorial, to download the necessary data.

Basics & layout /by D. Biswas/

This tutorial part consists of explanations of the following aspects:

- user interface;
- gvSIG Projects;
- data source;
- importing data;
- editing symbology;
- editing tables;
- map layout;
- templates;
- publishing;
- remarks.

Interface

Interface is the most useful part of gvSIG, which the users need to get to know very clearly. The graphical interface is intuitive and user friendly and use suitable for any user who is familiar with geographic information system. To know the functionality of Interface, users can divide the whole interface of gvSIG in five different parts (Title Bar, Main Window, Menu Bar, toolbar and status bar)

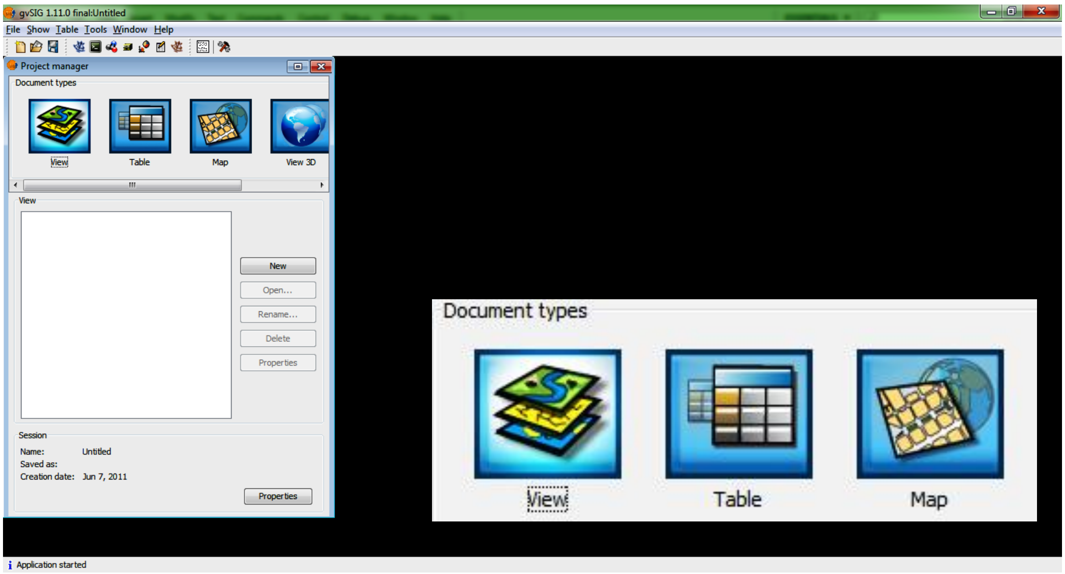


Figure 8: gvSIG Interface

gvSIG Projects

In gvSIG, all the activities are located in one project. This project can be divided into three different documents (View, Table, Map & View3D). Views are the documents in which we work with the graphical or with attribute data. Table is the documents in which the alphanumeric attribute data is stored. Map allows the different cartographic elements which represent the Map is to be visualized (North direction, legend and scale).

It is very necessary to know that the ".gvp" extension do not include spatial data and associated attributes in the shape tables. Instead they save the references to the places the data sources are stored (the path to be followed in the disk in order to find the files). If data changes the updates will be shown in all the projects are used for that. This menu allows to access the project Management options is located in the "File" menu.

In order to fix the properties of the "View" documents, we have set up the properties with the projection, so the base map can support the projection system of gvSIG. In order to that, click the view properties. The units for your view are important, since they are applied as default in the tools, e.g. when something is measured. The background colour is by default white but the user can choose it, as he/she demands. It is necessary to find out the current projects. In order to change it, you should select "ESRI". So that it appears to search the criteria by code or by name. So for Germany "we can select WGS_1984", which shows the different options of WGS_1984. You can select "WGS_1984_UTM_Zone_32N", which should be fixed for the projection of "Baden-Württemberg" of Germany. Figure 9 shows the properties of spatial reference system in gvSIG.

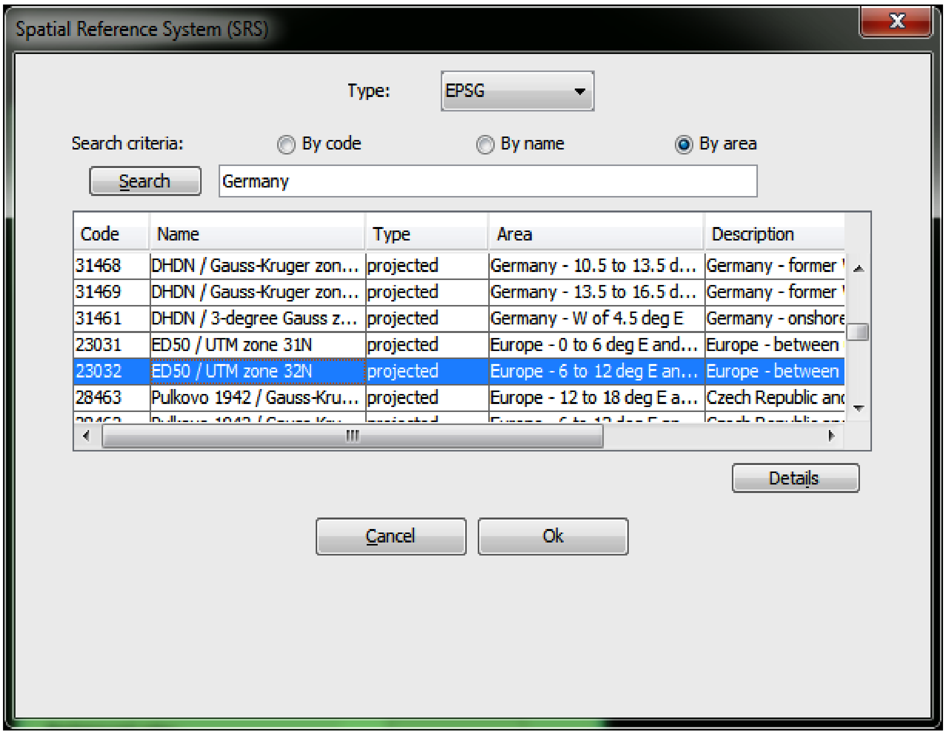


Figure 9: Spatial Reference System of gvSIG

Data Source

Different types of cartographic information can be added to a view. Vector and raster files can be loaded. Each of these groups can contain a wide range of formats.

GIS data: The standard GIS format is the shape, which stores both spatial data and their attributes. A shape (also called "Shape file") is actually three or more files with the same name and different extensions (even though in gvSIG it is handled as one file):

- dbf: Table of attributes.
- shp: Spatial data.
- shx: Spatial data index.

From version 0.5 onwards, gvSIG also has the capacity to access the MySQL Spatial and PostGIS spatial data bases via a new driver which uses JDBC.

CAD data: These are vector drawing files which support the dxf and dgn formats. The CAD files may contain information on points, lines, polygons and texts. From version 0.4 onwards, gvSIG also allows access to the information contained in Autodesk's 2000 dwg files.

WMS data (Web Mapping Service): gvSIG can be used to consult WMS data, i.e. data available on the web. WFS data (Web Feature Service): From version 0.5 onwards, gvSIG can be used to download WFS vector layers from servers that comply with the Open Geospatial Consortium (OGC) Standard.

WCS data (Web Coverage Service): From version 0.4 onwards, gvSIG allows access to remote information based on the OGC's WCS protocol.

GML (Geography Markup Language): From version 1.0 onwards, gvSIG allows GML documents to be displayed and exported. Geography Markup Language (GML) is an XML format to transport and store geographic information whose design is based on specifications produced by the OGC group.

Images: gvSIG can display different raster images (tiff, jpg, ecw, mrsid, etc.). From version 0.4 onwards, gvSIG can save images which have been modified in these formats.

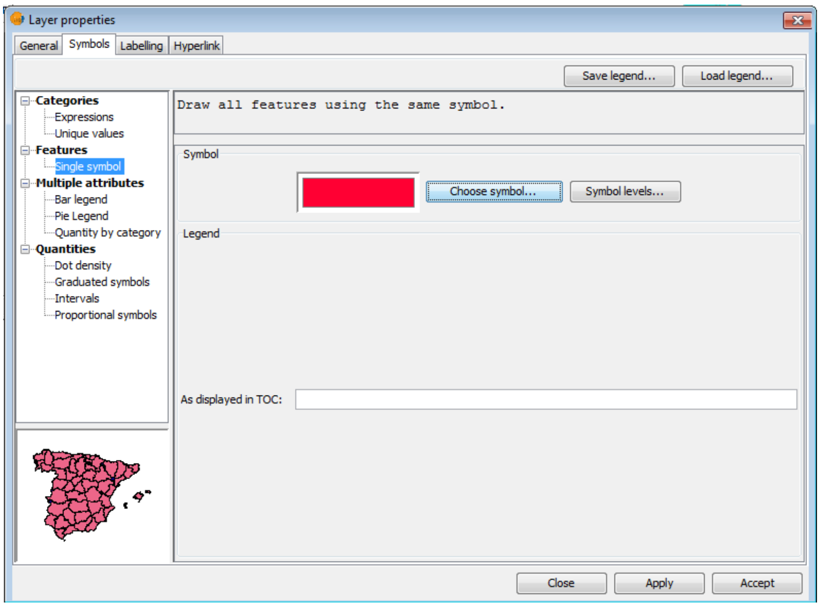
From version 0.5 onwards, "colour palette" (GIFs, 8-bit PNGs, etc.) raster files can be opened and raster files without georeferencing can also be opened. Moreover, this new version supports GIF, BMP y JPEG2000 formats.

Importing Data

Importing data could contain several vector layers like settlement area, lake, rivers, highway, railway and cities of a region. In order to add a new layer you should click "add layer" button of the toolbar. If you click it a pop-up window will appear, here click "Add" again and navigate to the file "Settlement_areas". Please pay attention that the "gvSIG shape driver" is selected in the "File type" box at the bottom. Then select "Settlement_areas" and choose "Open". In the same way you can add raster files, only change the "File type" as "gvSIG Raster driver".

Editing Symbology

It is necessary to visualize the map with respect to different cartographic attributes, which includes the requirements of colour settings. By defaults all layers are displayed with cyan colour. Select a layer with the left mouse, right-click on it and choose "Properties". In the Symbology Tab you can choose different options to symbolize your layers similar to ArcGIS. Then choose a suitable Symbology for all the layers. With the “save legend” button, it is possible to save the Symbology in a legend so that you can use them for other purposes. Figure 10 describes the editing result with the Symbology.



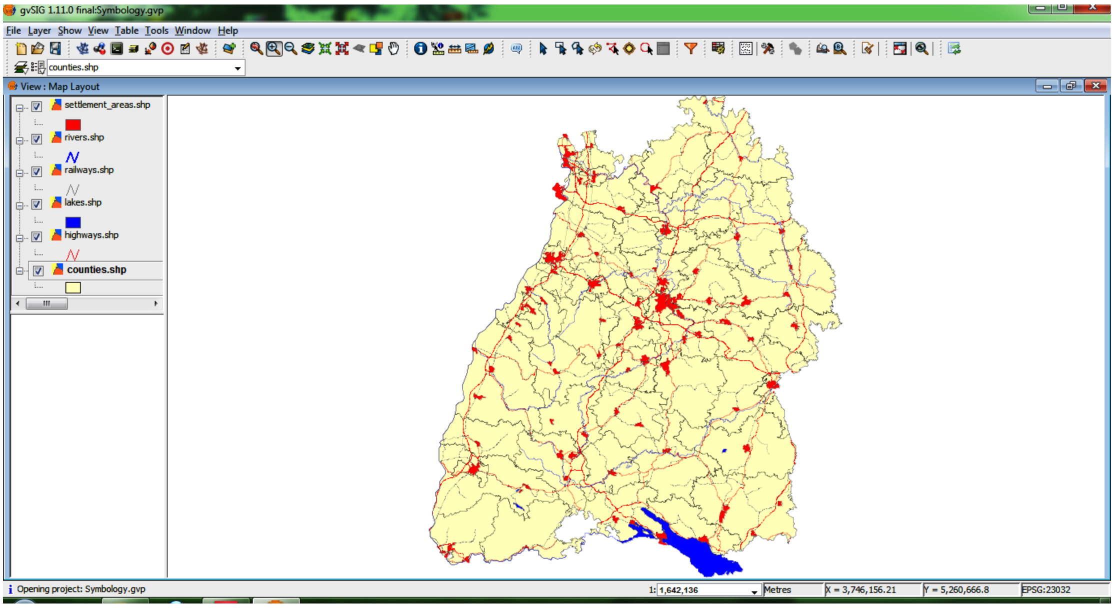


Figure 10: Editing with Symbology in gvSIG

Editing Table

Working with the attribute table is necessary for updating or editing data from the Table. It is necessary to select the layers, upon which you want to work. By selecting “show attribute table of active layer” you can open the attribute table of the selected layer. When a table is open, in the table menu in the menu bar you will find some different options like sorting, jointing tables, summary statistics or printing.

It seems that the main functionality of the tables in gvSIG is the possibility of selections depending on attribute vales. To do this choose the tool “Filter”, either in the toolbar or in the menu bar “Table”. You can select features by their attributes as known from ArcGIS. When the layer is in the editing mode (“right click on it and select “Start Editing”, its name will be highlighted as red”), you can add, delete or rename fields with the option “Manage Fields”. It seems that there is field calculator in gvSIG like in ArcGIS.

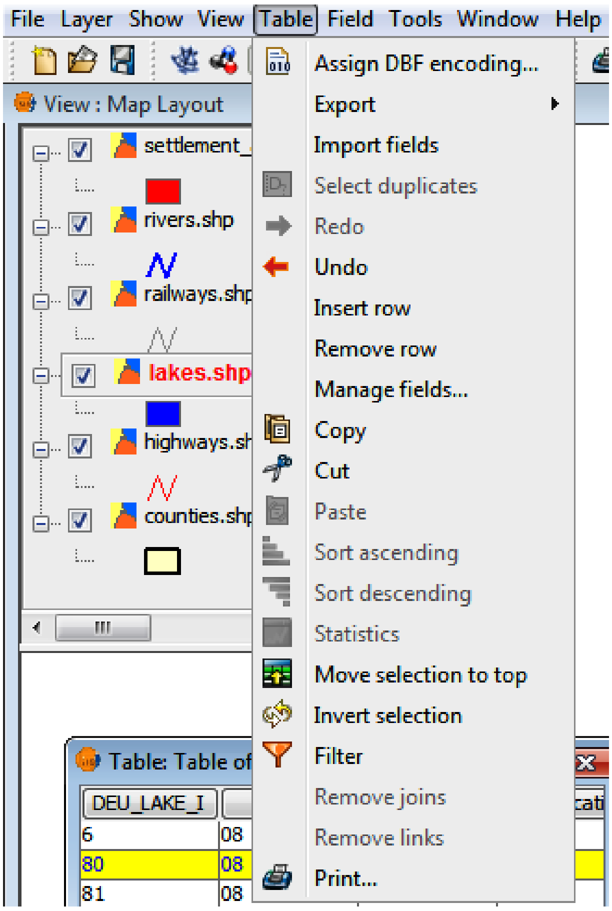


Figure 11: Table properties in gvSIG

Once you open a table inside a view, it will also appear in the project Manager in the “Table” tab. Since the attribute table is not made by the gvSIG developers, but it is an extension by another unknown group of developers, it is only available in Spanish. Even in the English user manual of gvSIG, this part remains untranslated. That is the main reason why it is not further explained in this tutorial.

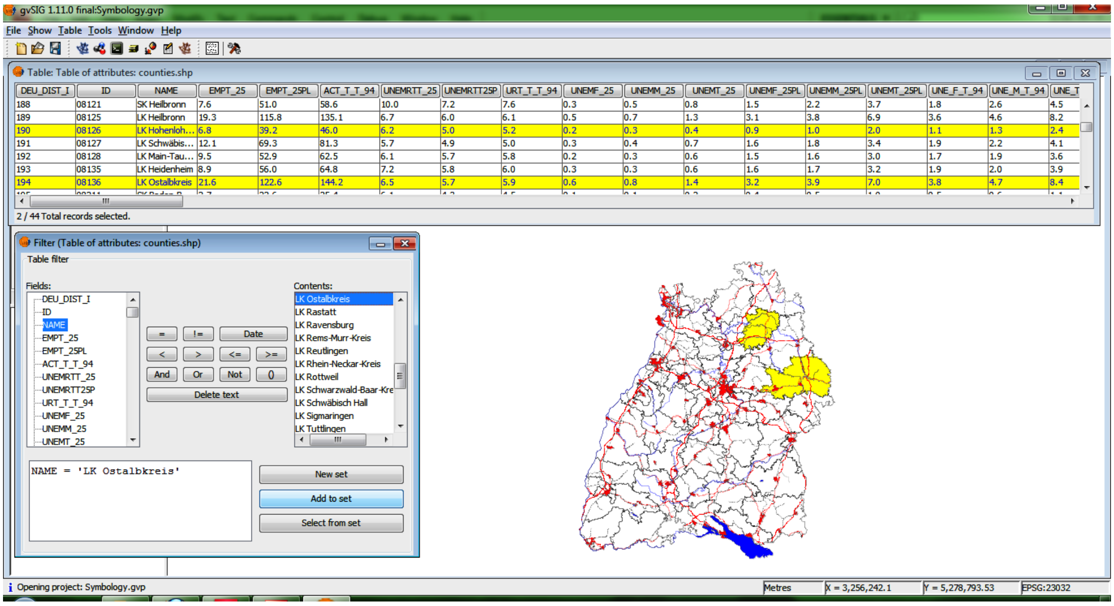


Figure 12 : Editing Tables in gvSIG

Map Layout:

It is very much necessary to visualize the map by creating the Map layout. It includes legend, north direction, title, scale bar, author and source. So it is urgent to select "New" in the Map Tab from the project management. Give a suitable name and open it.

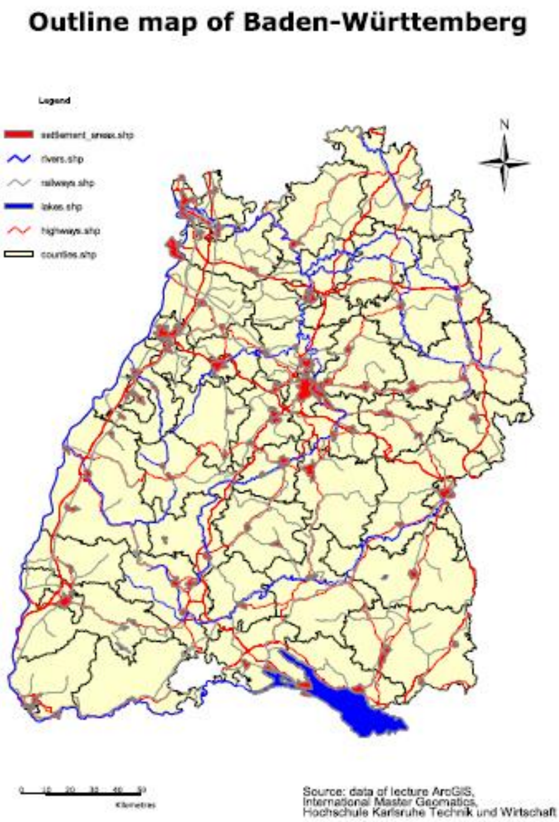


Figure 13: Preparation of Map Layout in gvSIG

Templates

To generate a nice layout through the gvSIG is time consuming. So in order to save it, you can use the "Save as Template". This can be used for next update or according to the users' choice.

Publishing:

When you want to publish your map, you may have different possibilities:-

- Print directly the map layout: In the Menu bar go to "Map > Print", and choose the preferred settings.
- Export the Map as .pdf, go to menu bar find "File - > select Portable Document Format (PDF)".

Geoprocessing /by W. Banahene/

Analysis of data in gvSIG: The platform for the task was set by installing gvSIG.

Data used for the task was "GIS1_TB01e" which contained data sets of the National Institute for Environment, Measurements and Nature Protection, which referred to the urban area of Karlsruhe.

The gvSIG geoprocessing extension allows one to apply a series of standard processes to the vector information layers loaded in the layer tree in a gvSIG view (ToC), which is creating new vector information layers which will provide new information for the source layers.

The following geo-processing tools were used to analyse the data in gvSIG:

- Buffer – This tool generates an "areas of influence" around the vector element geometries (points, lines and polygons) of an "input layer" that is creating a new polygon vector layer.
- Spatial Join – This tool transfers the attributes of one layer to another based on a common element.
- Lateral Buffer –This tool creates a new polygon layer, with polygon buffers of the geometries of the input layer which can have only line type.
- Clip –This tool creates a limit to the working area of a vector layer (points, lines or polygons), and extracts an area of interest from it.

- Intersection –The tool works with two layers, the input layer and the overlay layer. It computes the intersections with the geometries of the overlay layers.
- Union– The tool is similar to the “Intersection” and "Difference" geo-processer in that it operates on two polygon layers to obtain their intersections (this is why these three geo-processers are known as “overlay geo-processers”).
- Convex Hull –This tool calculates the “Convex hull”, or the smallest convex polygon which surrounds all the vector elements in an “input layer”.
- Dissolve –This tool only acts on one “input layer” whose geometry must be a polygon type. The process analyses each polygon in the "input layer" and merges the polygons that have an identical value for a specific field into one polygon.
- Split line –The functionality provided by this geo-process is to cut a line into sections of equal size.
- Merge –This geo-process act on one or several layers, generating a new layer which joins all the geometries in the “input layer”. The "result layer" of this geo-process will keep the attributes of the "input layer" specified by the user.
- XY Shift –This geo-process allows a translation transformation to be applied to all the points, lines and polygons of the geometries in the input layer.
- Reproject –This geo-process changes the projection of the vector elements in the input layer.

Implementation of gvSIG results in Flash

The geo-processing part of the tutorial was designed according to the standards set by the group in Flash. This part of the tutorial was designed based on four geo-processing tools namely union, buffer, xy shift and convex hull tools.

The Frame rate, dimension, background color and ruler units were set with respect to the design standard for each flash movie. To check horizontal and vertical alignments of graphics the grid option was enabled. Graphics, text and buttons were imported into the library to enable easy access to them. The graphics were made up of screen shots of analysis made in gvSIG. The Flash movies were stored in folders according to the frame number in the time line. Contents of the first frame were stored in a folder called” Frame1” which contained layers made up of text, buttons and graphics. To enable mouse events text, buttons and graphics were converted to symbols and given instance names. Various movies were linked with buttons by action script for easy navigation through the tutorial. Videos showing procedures in gvSIG were captured using the" CAM studio" software and converted to flash video files(FLV).The FLV files were imported and embedded into the movies. The viewings of the videos were controlled by buttons via action script.

Uploading of Flash files, data and self-evaluation in Moodle

The finished shockwave flash movie (.swf) file and the vector data was uploaded in Moodle. Two questions based on this aspect of the tutorial were created in Moodle.

Georeferencing & Projections /by J. Turka/

This subpart of the tutorial is dedicated to georeferencing and projections in gvSIG as the spatial reference is one of the most important characteristics within GIS.

The task and procedure first was revised in the ArcGIS as well and the practical exercise was overtaken from GIS I assignment "C Editing" and implemented in the tutorial for gvSIG. Of course, any other georeferencing task can be done by using own data because gvSIG supports also WMS or any other database as reference cartography.

Project steps

The tutorial consist of two parts, as, compared to ArcGIS, gvSIG has two main options for georeferencing based on whether there is a reference cartography available for image to be georeferenced or it is based just on ground control points with known coordinates. These two methods are distinguished in tutorial by naming them as they are called in gvSIG environment – "Georeferencing with reference cartography" and "Georeferencing without reference cartography".

In the beginning of georeferencing part there is short information about georeferencing itself (some theoretic explanations, so the knowledge about georeferencing is also covered in tutorial).

As the launching the georeferencing tool is different as in ArcGIS, it is also described in tutorial.

After in the part of "Georeferencing with reference cartography" the explanation of georeferencing view elements and different views is given, so the user does not get confused; here as well the whole georeferencing procedure is stepwise explained, even though it does not differ much from the one used in ArcGIS or any other software.

Besides the procedure, some special cases are explained as well, for example, the use of transformation algorithms (affine, polynomial), what is “Georeferencing with resampling”.

At last in the part “Georeferencing without reference cartography” a practical task is given so the user can try out by himself if has understood the whole procedure and the main principle of georeferencing. Shortly an overview of the list of steps is given as well, in order user needs some points to stick on by completing the task.

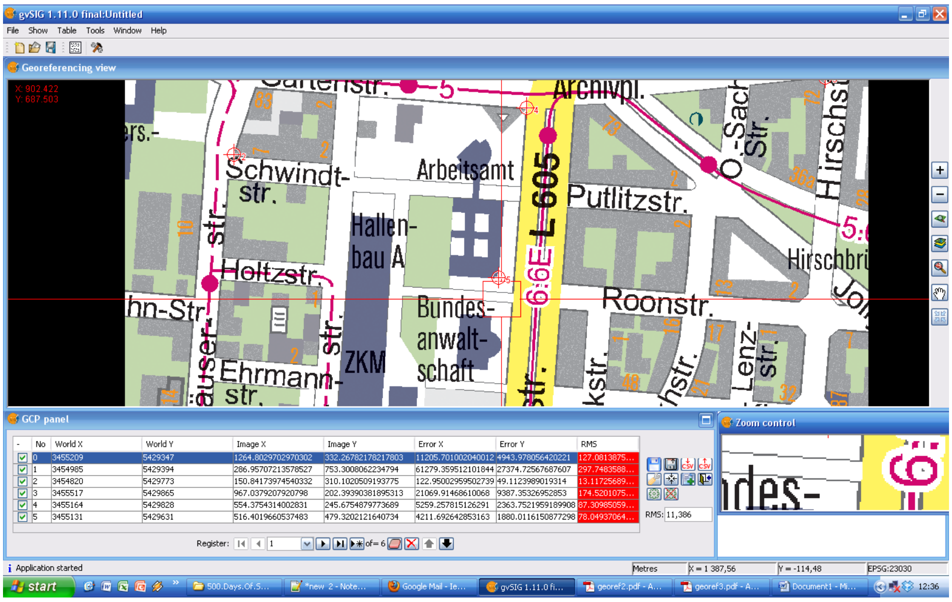


Figure 14: Georeferencing without referencing cartography in gvSIG

Result

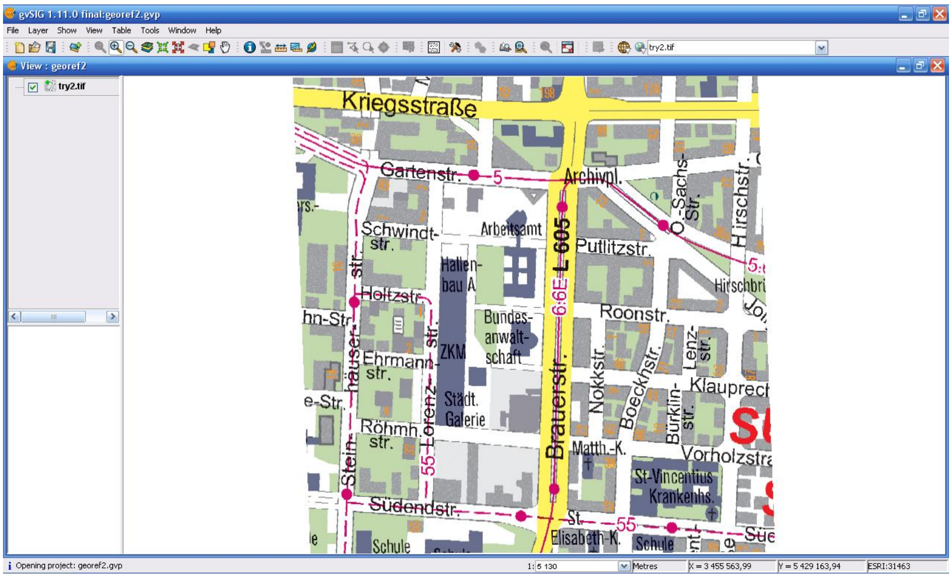


Figure 15: Georeferenced image in gvSIG

Finally the result should be achieved is given so the user can compare. As well the information what could be done further is given, as the result of georeferencing part is used in Digitizing & Topology part.

Digitizing & Topology /by N. Guth/

In this part of the tutorial, the student learns how to digitize a georeferenced map and gets an idea how topological models work. As starting data, only the map of Südweststadt Karlsruhe is necessary. In the first step, the needed shapefiles must be created. This is done by clicking View -> New Layer -> Shapefile in the main menu. In the following dialog, the name of the layer and the geometry type must be specified. The geometry types

- Point
- Multipoint
- Line
- Polygon

are available. In the next step, additional fields can be added. For each field, a data type must be specified. In the last view of the dialog, the path and filename can be chosen and the projection can be altered. After the shapefile is created, a new layer containing the shapefile is added to the table of contents (TOC) in the current view. The red colored name of the layer in the TOC indicates that editing is enabled for the layer. The following shapefiles have to be created:

- Blocks, a polygon shapefile containing the house block
- Special_use, a polygon shapefile containing paved and green areas within the blocks where no buildings are
- Buildings, a polygon shapefile containing the buildings within the blocks
- Public transport, a line shapefile containing the tram lines in the area
- Stops, a point shapefile containing the stops of the trams lines

Georeferencing

In the next step, the base map must be georeferenced. In the tutorial, the coordinates and points are given. When the mouse cursor hovers over the coordinate list, the corresponding point on the map is tagged. The process of georeferencing is not described further, but a link to the tutorial is available. Figure shows the tutorial page with the coordinates, the map and the adjustment which must be taken.

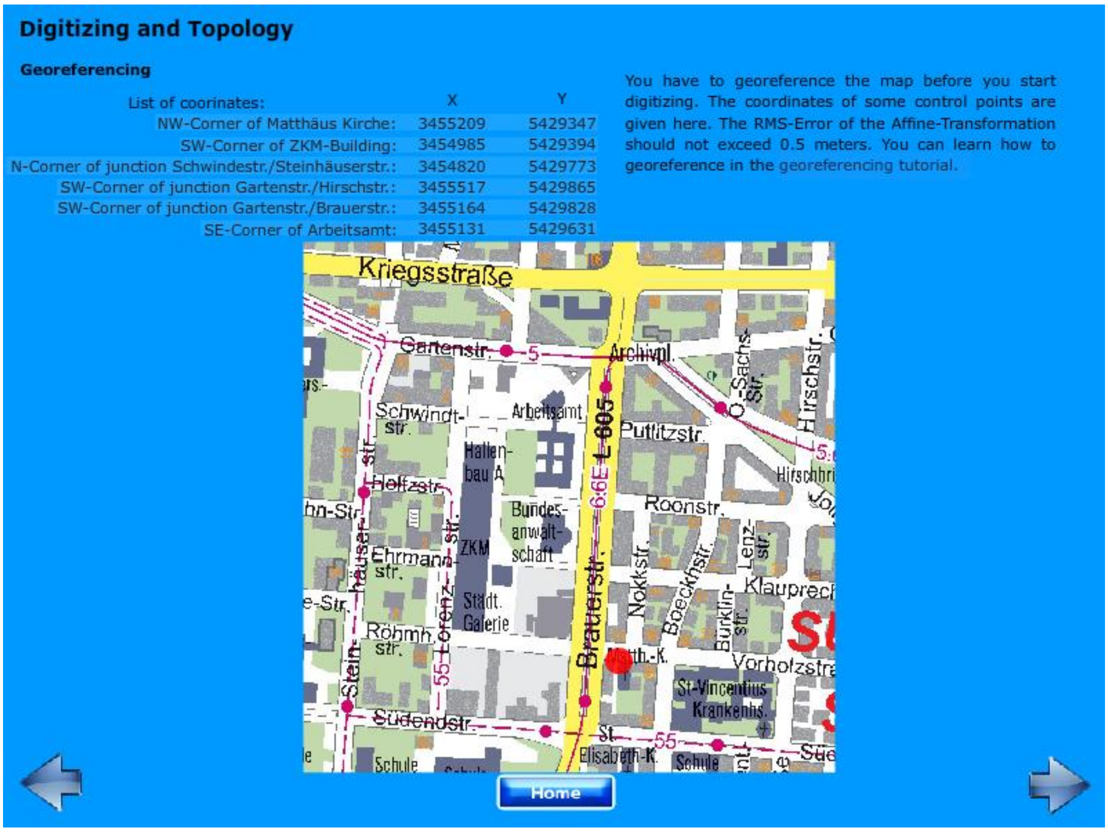


Figure 16: view of the georeferencing page. The first coordinates are tagged in the map.

Topology and digitizing

In the next part, the installation of the topology add-on and the principal usage is described. To show how the add-on is installed, a video is used. In this video the student can see which steps are necessary and where to find the dialog in the menu. The video shows the task very slowly, so it can be easily followed. A special focus is on the non-official state of the topology add-on. This fact is mentioned in the text, too. So the student gets to know that the topology tools are not working properly at the moment and only the topology rules will be described in the next step. It is also mentioned that gvSIG has to be started with administrative rights in Windows 7 or Windows Vista in order to be able to install add-ons.

The definition of the topology-rules is demonstrated with the help of three pictures which show the settings which have to be taken. This is giving the topology a name, defining a cluster tolerance value as radius in which points are considered equal, setting a maximal number of allowable errors in the topology and choosing which layers are included in the topology. In the next screen the rules can be added. Rules are defined as can be seen on the left side in Figure : a rule from a list can be selected which references one or two layers, called A and B. On the right side of the dialog, a small drawing shows what is meant with the rule, in which cases errors occur and in which cases not. Errors are indicated with a red colour. The picture on the right side of Figure shows all rules used in the tutorial and the participating layers.

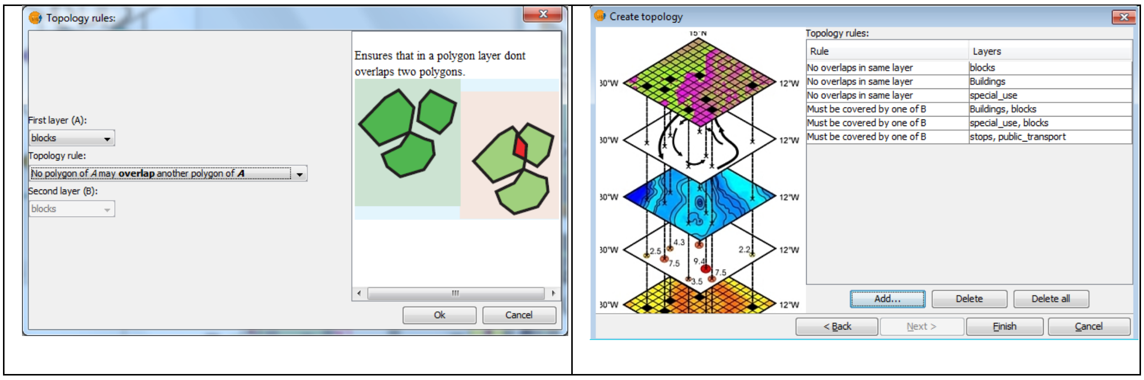


Figure 17: definition of topology rules in gvSIG

On the next frame of the tutorial, the tools used for digitizing are presented. The tools are divided into classes which show what kind of geometries can be digitized with the respective tools. For point features, there is only one tool available, all other tools but the most right one (see Figure) can be used for both line features and polygon features.

Figure 18: digitizing tools

The tool at the right side of the toolbar is the one which is used in the tutorial for the polygonal layers. With this tool, polygons with arbitrary forms can be digitized. The public transportation lines are digitized with the polyline tool (5th from the left) and the tram stops are digitized with the point tool (left).

As no gaps or overlaps are desired in the result and to simplify the digitization, the snapping functionality of gvSIG is shown on the next page. There are two windows where snapping properties can be set, both to be found in the editing settings which can be accessed by right clicking the respective layer. This means that every layer has its own snapping environment, thus the snapping properties must be set for each layer separately. When the editing session is closed, the snapping settings are lost and must be done again the next time. The screens where the settings can be done can be seen in Figure . The picture on the left side shows the dialog as is appears when the editing settings menu item is clicked in the TOC. In this form, the snap tolerance in pixels can be adjusted. This means how far the mouse cursor can be moved to an existing feature. In the middle of the form, the layers to which can be snapped can be activated witch checkboxes. On the right side of Figure , the snapping properties page within the editing settings dialog is displayed. Similar to AutoCAD, the points to which can be snapped can be activated or deactivated and the priority can be set. Thus, it is possible to create a priority list which makes editing easier as the cursor rather snaps to an endpoint, for example, than to the nearest point.

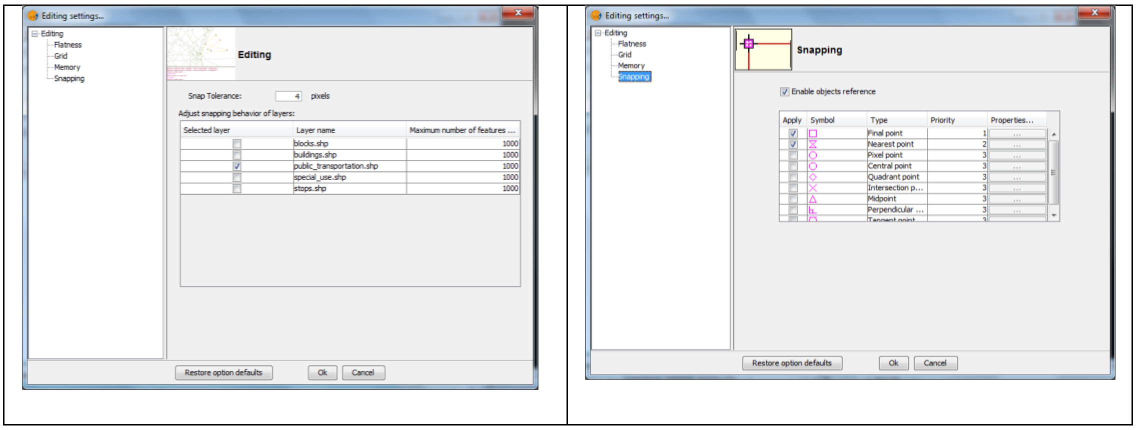


Figure 19: Settings dialogs for the snapping environment of each layer

With the knowledge learned so far, the students should be able to do the digitizing task on their own. All necessary techniques are presented and with the navigation buttons, it is possible to go through the tutorial in any direction to look for information again.

Digital Elevation Model's /by D. Heim/

Project Steps

This subpart of the tutorial is about the basic work and visualization of digital elevation models (DEM) in gvSIG. First, the GIS II assignment "Digital Elevation Models" was repeated using ArcGIS. It consists of the following parts:

- presentation of the DEM with a color ramp
- hillshading
- contour lines
- TIN out of masspoints
- profiles of the terrain
- 3D-visualization of land use data and rectified aerial photography.

Then the task was redone with gvSIG. Both the general user manual and the one of the 3D extension were consulted and it was determined that, in theory, the task can be done in the same manner. The mailing lists of gvSIG and SEXTANTE (company, who developed the analysis tools for gvSIG) which allow to contact gvSIG users and developers were very helpful to solve problems like "Where is the Hillshade tool located (there is no search function in the toolbox)? Does it work properly?" With their help it turned out, that in the latest version of gvSIG there are bugs in the tools reproject raster, shaded relief and Delaunay Triangulation.

This was also confirmed by members of the professional structure of gvSIG. Therefore the parts "Hillshade" and "TIN out of mass points" are not taught in the tutorial. For the color ramp an extension is necessary, which is also not available in the correct version. However, this part remains in the tutorial and is solved by teaching how to create an own individual color ramp. After knowing how the task can be done in gvSIG an interactive Flash tutorial was developed using text, images, buttons and mouse over events. Where many simple actions follow each other videos are used to avoid too many step-by-step instructions. Since the target group consists of students with some GIS experience, the main pages of the tutorial consist of very little text. When a users wishes more explanations or needs help, new info pages are available via buttons.

Result

The DEM tutorial is divided into two parts. According to the DEM task of the GIS II assignments the first part of this tutorial teaches how to load DEM's in a 3D view like it is possible in ArcScene and to drape raster and vector data over the elevation model.

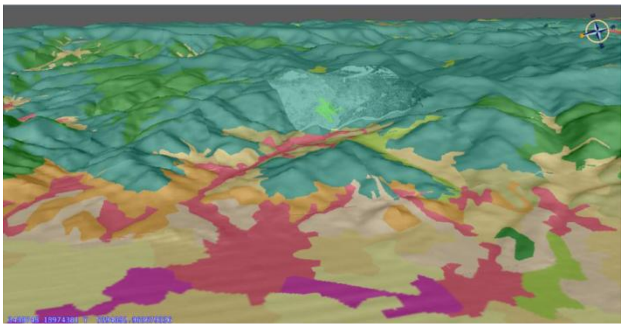


Figure 20: Tutorial "Working with Digital Elevation Models - Basics" shows a DEM of Baden-Württemberg with overlaid landuse layer and an aerial image.

The second part shows possibilities to receive an easy impression of different heights in a two-dimensional view. Here an elevation profile is used as well as the vectorization of the DEM to contour lines. Finally, the visualization with a color ramp is taught.

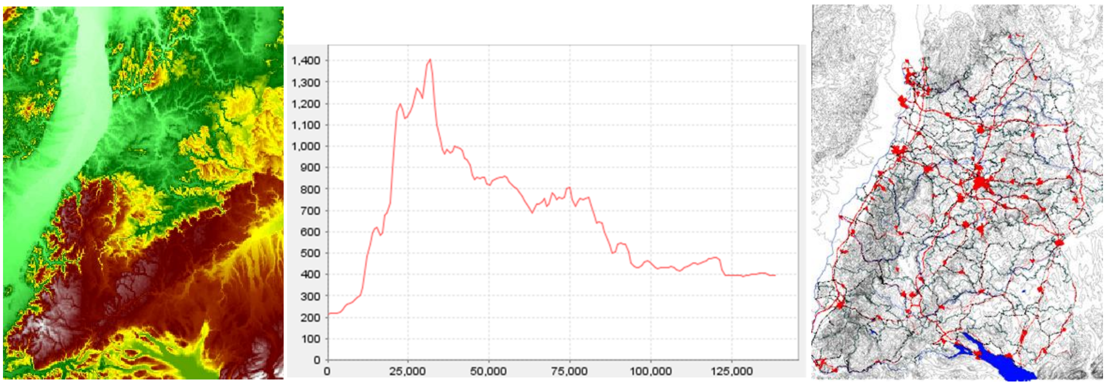


Figure 21: The tutorial "Working with Digital Elevation Models - Visualization of elevations in a 2D View" teaches three different possibilities to receive height information of a DEM in a 2-dimensional way. From left to right: Colour ramp, elevation profile and contour lines.

Test stage

After completing tutorial topics a test stage was carried out. Due to time problems, there was a delay therefore test stage was shortened to six days. Unfortunately, since it was near exam period and final weeks of the project, there was not as much participation as expected. The feedback from those few people, who managed to have a look at tutorial, was positive. Only drawback they gave was about background colour. That has been changed to a higher contrast colour. Otherwise they liked the tutorial and felt it to be useful if in future they might need to use gvSIG software. Maybe it could be published in web not just in Moodle platform so that also other people can access it.

PROJECT MANAGEMENT /BY D. HEIM/

General

In the module "GIS-Project and -Management" not only a project to a GIS-related topic has to be accomplished but also Project Management should be learned and applied. "Project management is about creating an environment and conditions in which a defined goal or objective can be achieved in a controlled manner by a team of people (Haughey, 2010)." To make the project management easier, less time-consuming and much more efficient Microsoft Project is used. This software allows developing and monitoring a project plan to identify time-related main stages consisting of several single tasks, assigning resources, determining costs and deadlines for completion, calculating delays and many other useful aspects. However, only the project planning is done with Microsoft Project, the realized project is documented with a simple Excel table to be able to compare the discrepancies at the end.

For the successfully completed module six credit points are given per student, which correspond 180 working hours. With 10 hours of lecture time and 15 hours of preparation for the oral exam subtracted, 155 working hours remains to perform in 7 weeks (the week from 2. to 9. May 2011 is not counted, since the first group meeting is on 10. May, but the holiday on Pfingsten is counted as a normal working week). This is an amount of time of 22.2 hours per week or a workload of three hours a day. Therefore, in the project planning inside Microsoft Project it is calculated with three hours a day as the general working time (also at the weekend). In the following short descriptions of each week are given. The basic of it is the planned and the realized project progress, which can be found in the appendix.

9. May - 15. May: Preparation

On 10th of May the first meeting with the supervisors took place where the goal and target group of the project were discussed. This kick-off meeting was the initiating and planning for the gvSIG project and the tasks for this week were divided as seen in figure 25 in the appendix. For the processes "Concept" and "Presentation" the attended time was one hour respectively half an hour more than the planned one, but the tasks were completed in the week. The process "Moodle" required an hour less than planned, and the remaining tasks were fulfilled in time. Altogether, the project realization took place as forecast.

16. May - 22. May: Development of gvSIG tasks

With this week the actual execution of the project started. To develop the different gvSIG tasks twelve hours were intended in each case. Afterwards, it was emerged that in four of the five processes less time was needed (only eight or ten hours). In the planning, the weekly processes "Report", "Presentation" and "Project Management" were forgotten. Nevertheless, they were done in time.

23. May - 29. May: Implementation to Flash

The producing of the Flash files was the most time-consuming part of all the work. It was estimated that the workload is about 24 hours per tutorial topic, but there is no strict task-to-do deadline within this week, because it was assumed to spend two weeks for implementing the whole tutorial in Adobe Flash. However, for this week 16 hours workload were scheduled with an exception for the topic "Projections + Georeferencing", because J. Turka, who was responsible for that part, had to work also approximated seven hours on pedagogic of e-learning researches. Comparing the planning and the realization it is conspicuous that the foreseen workload for the Flash implementation was not achieved by far.

Again the remaining weekly processes "Report", "Presentation", "Project Management" as well as the "pedagogical research" were realized in the planned amount of time.

30. May - 05. June: Implementation to Flash

For the finishing of the production of the Adobe Flash files, which is the aim of this week, another eight hours are planned (respectively 16 hours for the topic "Projections + Georeferencing"). However, to compensate the laziness of the last's week, definite more working hours (10 to 16 hours) are necessary. In addition to that, the task to check the possibilities to add a quiz for self-evaluation either in Moodle or Flash, was planned for the next week, but we managed to complete it this week. Concerning this point the project progress is not only in time compared to the planning but even a step ahead.

06. June - 12. June: Preparation for Moodle implementation

This week, another six hours were needed for some corrections in the "Basics & Map Layout" part. Also, the whole tutorial as well as the data and the questions for the self-evaluation were implemented to Moodle. In general, this is not a complicated deal, but since the Flash tutorial is very big – around 100MB – we planned enough time (four hours) to be able to find a way for the uploading if problems should appear. It proved that the uploading wasn't a problem, but since often unplanned improvements were necessary (in total every group member spent 2.5 hours with it) everything had to be uploaded several times. In the end, three hours were taken. The uploading of the data and the inserting of the question database took place as planned.

13. June - 19. June: Moodle implementation & Final Report

It was planned that the complete tutorial can be tested by classmates this week. Due to delays resulting from problems with the combination of the single tutorial parts (5 more hours were necessary), this could not be started until the end of the week. To be able to finish the project in time and have enough time for analyzing the test results, the test stage was shortened from ten to six days. The second aspect of the week considered the final report. A template was developed and each group member was assigned to particular chapters. The deadline for the writing was the 26th of June, therefore the duration spent with this task ranged between two and ten hours between the group members.

20. June - 28. June: Final Report

According to plan, around four hours were spent with further improvements by all group members. The main point of this last project phase was the developing of a final report. Six hours were foreseen, but it required much more. Depending on the individual work of the last week, the time amounted to two up to 13 hours. Two more hours than planned (namely four) were necessary to combine the chapters and another two to burn a DVD with all the projects results.

Summary of Project and Time Management

In total it was planned to complete the whole project in 357.5 hours, but at the end only 328,5 were needed. The small difference of 29 hours proofs that the project management was well planned. This thesis is supported by comparing figure 22 and 23 a trend line would look very similar with low points in the first and the fifth week, a peak around the third week and another one at the in the sixth and seventh week. The amount of time of 155 working hours per group member (derived from the ECTS points as mentioned above) was not needed to fulfill the task.

As seen in figure 22, at the beginning the foreseen working hours were more or less similar for all group members. Toward the end of the project there was much more variability. The reason for it was that differences in the realized work (compare figure 23) were taken into account in the weekly updating of the planning. Especially, this is noticeable in the last two weeks for D. Biswas: since he had less realized working hours than the other members, the planned as well as the realized amount of time was considerably higher at the end.

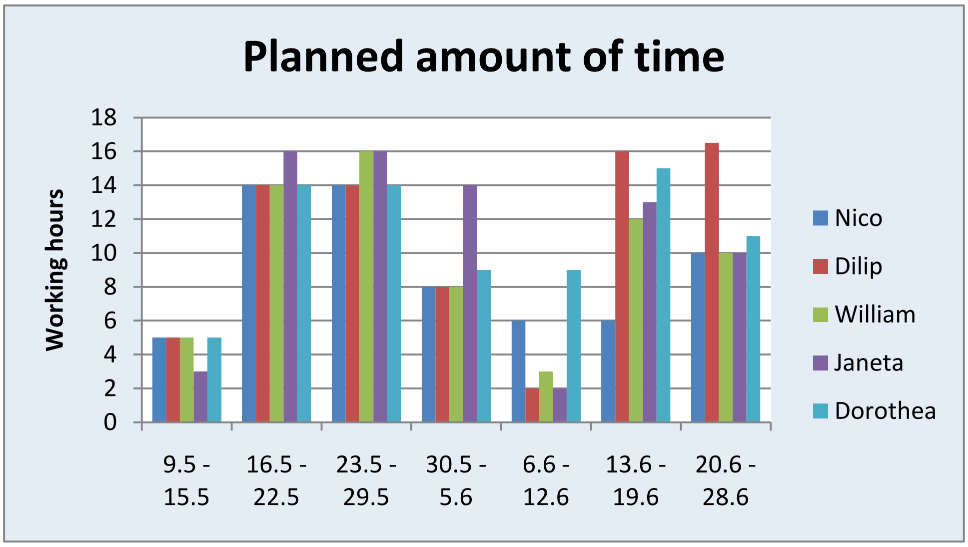


Figure 22: Planned amount of time according to group members.

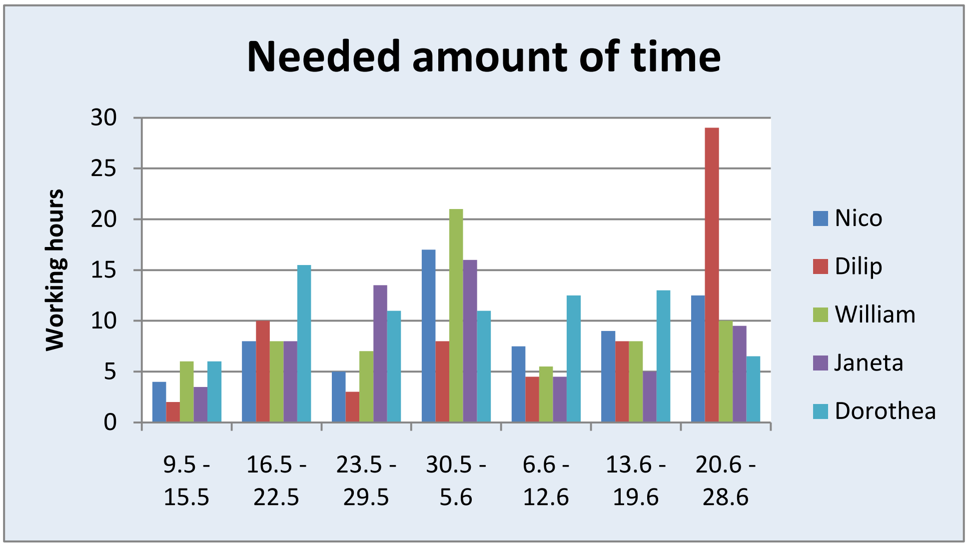


Figure 23: Needed amount of time according to group members.

Finally the personal working hours lay between 63 and 75.5, which indicate that the work was divided relatively equal between all members. Figure 24 shows, that the maximum discrepancy between plan and realization is 14 hours for J. Turka and 11 hours for D. Biswas. N. Guth, W. Banahene and D. Heim worked almost as long as planned.

	N.Guth		D. Biswas		W. Banahene		J. Turka		D. Heim	
	planned	*realized*	*planned*	*realized*	*planned*	*realized*	*planned*	*realized*	*planned*	*realized*
9.5 - 15.5	5	4	5	2	5	6	3	3,5	5	6
16.5 - 22.5	14	8	14	10	14	8	16	8	14	15,5
23.5 - 29.5	14	5	14	3	16	7	16	13,5	14	11
30.5 - 5.6	8	17	8	8	8	21	14	16	9	11
6.6 - 12.6	6	7,5	2	4,5	3	5,5	2	4,5	9	12,5
13.6 - 19.6	6	9	16	8	12	8	13	5	15	13
20.6 - 28.6	10	12,5	16,5	29	10	10	10	9,5	11	6,5
Total	63	63	75,5	64,5	68	65,5	74	60	77	75,5

Figure 24: Overview of time management according to group members.

The project management was very useful, since it forced to think about all the necessary steps and to make a good plan. Also, delays and differences in the assignment of tasks to group members are easily seen so that adjustments could be made.

CONCLUSIONS /BY D. BISWAS/

gvSIG is designed to be easily extendable, allowing continuous application enhancement, as well as enabling the development of tailor-made solutions. It supports Java and is available for Linux, Windows and Mac platforms. It Integrates in the same view both local (files, databases) and remote data through OGC standards. The open source software gvSIG, with GNU/GPL license, allows its free use, distribution, study and improvement.

To develop a tutorial with the software flash is time consuming, huge concentration and expertness to work with both gvSIG, Flash and Moodle. For the future, we recommend to use separated but linked flash documents instead of combining everything in a single file, since this file will be become to huge and make the software very slow.

This tutorial is developed for the students, who have the basic ideas about GIS and is familiar with spatial analysis. For the best use of this software students can use this tutorial to follow the steps and practice with the gvSIG software simultaneously. After this tutorial, one will be able to work freely with preparation of Map Layout (view, legend, north, scale, frames, image, text, graphic grid, templates); Geoprocessing (buffer, intersection, clip, dissolve, union, convex hull, difference, merge, spatial join, XY shift, reprojection, Sextante geoprocessing); Topology (topological building, topological editing, generalization, reverse line direction, polygons from/to lines, lines from/to points, Delaunay triangulation, Thiessen polygons, build); Geoprocessing and the environment of Digitizing and Editing. The users can also evaluate their level of knowledge about the software by answering the questions related to each parts of the tutorial. Development of further tutorials will ensure more expertness, efficient and effectiveness usage of the gvSIG software as well as geographic information system. Disable functionality of some tools in the software and causing the lower system activity are the main two drawbacks of gvSIG. Updating the gvSIG software from the original developers will be more helpful to work with it efficiently.

REFERENCES

Buzzetto-More, N.,A. (2007). Advanced Principles of Effective e-Learning. Informing Science Press.

CamStudio. http://camstudio.org/

Chao-Kuei Hung . Categories of Free and Non-Free Software

Clark, R. (2002). *Six Principles of Effective e-Learning: What Works and Why.* The eLearning Guild's Learning Solution Practical Applications of Technology for Learning e-Magazine (September 10, 2002). http://www.eLearningGuild.com

Dabbagh, N. (2005). *Pedagogical models for E-Learning: A theory-based design framework*. International Journal of Technology in Teaching and Learning, 1(1), 25-44.

Haughey, D. (2010). An Introduction to Project Management. http://www.projectsmart.co.uk/pdf/introduction-to-project-management.pdf

Rekkedal, T. (2003). *Internet Based E-learning, Pedagogy and Support Systems*. NKI Distance Education

Riva FLV Encoder. http://rivavx.com/?encoder

APPENDIX

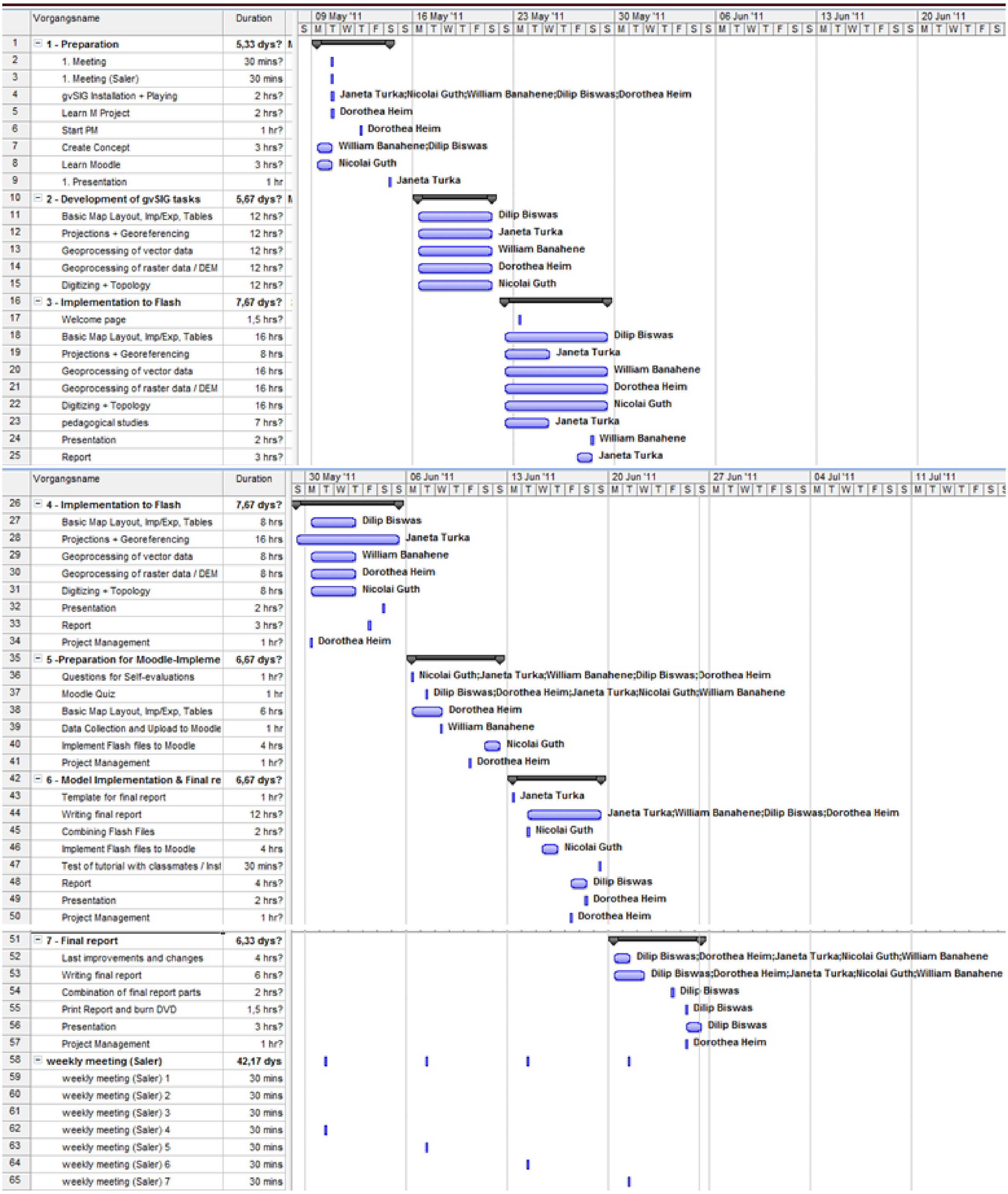


Figure 25: Planned project process in Microsoft Project.

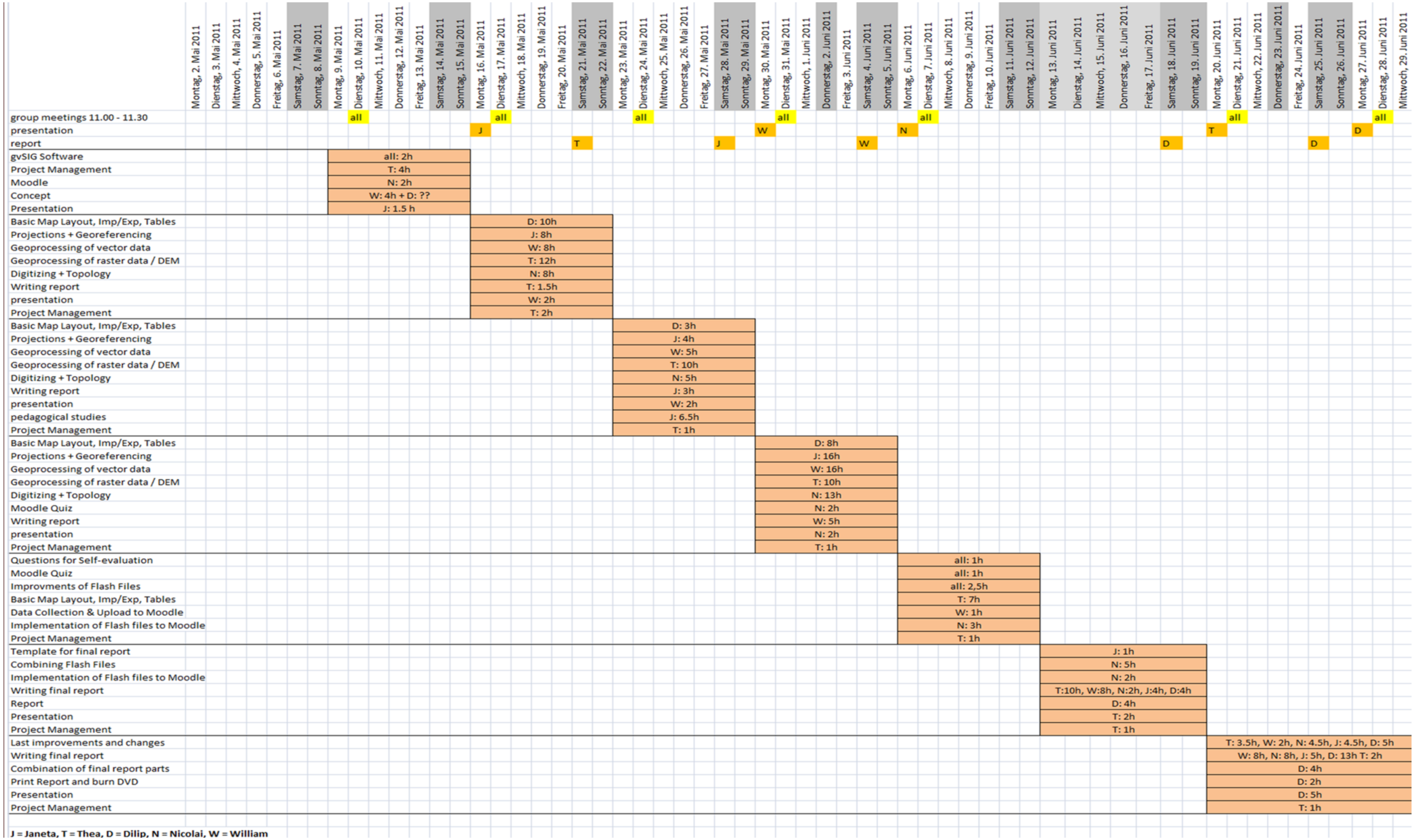


Figure 26: Realized project process.