Decision Support for Small Earth Dams using Web GIS

David McNamara

UNIVERSITY OF LEEDS

A dissertation submitted in partial fulfillment of the requirements of the Masters Degree in Geographic Information Systems of the University of Leeds

June 2014

Abstract

The benefits of creating small earth dams for retaining water, both surface and subsurface, are multiple. The reserved water can be used for domestic and agricultural applications. Natural ecosystems will benefit as a result of the improved hydrological conditions, often helping to halt or reverse desertification by slowing down the large water cycle and preventing water runoff in the rainy season. Improved water retention improves conditions in drought-affected ASAL regions and lifts the burden of water carrying from women, allowing more time for family and education. (Holzer 2012; Ulbig 2014; Stephens 2010).

This research examines the application of the geographical sciences and GIS as a methodology for assessing models of small earth dams with regard to their utility in retaining water for community use and conservation. Building on the work of a number of researchers, it proposes a GIS-based solution for estimating the scale of earthworks, reservoir volumes, watershed extent and dam construction risk. By formulating these concerns into a single procedural model, it is hoped that students and designers can use this work to accompany the decision making process during site selection and cost/benefit analysis. By comparing this GIS-based model with prior research and guideline formulae from the literature, the accuracy and effectiveness of this model can be evaluated.

Additionally, this paper is intended to be of use as a manual to students who wish to learn more about the development of the tool, with a view to enabling further research and understanding of the spatial analysis procedures described therein. Attention is given to the process of acquiring and preparing source data, with the intention that local elevation data can be sourced and imported into the solution. By utilising open source GIS technologies such as PostGIS, Mapnik and Leaflet, it is demonstrated how such a tool can be developed for the open web, and the output of this project includes a working, open sourced web based tool for designing landscapes of small earth dams.

UNIVERSITY OF LEEDS

SCHOOL OF GEOGRAPHY
UNIVERSITY OF LEEDS

DECLARATION OF ACADEMIC INTEGRITY

This form must be completed, signed and attached to every dissertation submitted for a degree in Geography at the University of Leeds.

I have read the University regulations on cheating and plagiarism and I confirm that this piece of work is my own and does not include any unacknowledged work from any other sources.

I understand that the title of this dissertation and my name, as its author, may be included in an on-line catalogue of School of Geography dissertations. Also, that the mark/grade may be recorded on the dissertation when it is stored in the School of Geography. I agree to this.

Signed.. Date: 30 June 2014

Surname	McNamara
First names	David Anthony

Programme of Study

Postgraduate	MA		MSc	X
Masters Programme Title		**MSc in GIS by ODL (WUN)**		
Other *(give programme title)*				

Dissertation title

Decision Support for Small Earth Dams using Web GIS

Dissertation tutor:	Word length*
Andrew Evans	14,784

* *Word length - exclude bibliography, appendix, tables, diagrams*

June 30, 2014

Acknowledgements

I would like to thank Paul Norman, Andrew Evans, and the academic staff at the University of Leeds for their support and suggestions during the course of the dissertation writing process.

Gratitude also goes to the people of Tamera Healing Biotope in Portugal for their vision and inspiration for this research.

I am indebted to my parents whose financial support and love of the natural environment ultimately resulted in this research.

Thanks finally to Lacey Cranberry whose company in the final stages helped to complete this work with light and love.

Contents

List of Figures

List of Tables

List of Procedures

Technical Vocabulary

ASAL = Arid and Semi-Arid Lands

API = Application Programming Interface

DEM = Digital Elevation Model

FSL = Full supply level

GPS = Global Positioning System

NED = National Elevation Dataset

OGC = Open Geospatial Consortium

SDSS = Spatial Decision Support System

TLS = Terrestrial Laser Scanning

UAV = Unmanned Aerial Vehicle

USGS = United States Geological Survey

Chapter 1

Introduction

> "The conservation of rainwater on land 'in situ' and the conducting away only of the natural surplus of water in a region is *condicio sine qua non*– a condition essential for ensuring environmental security, global stability and the sustenance of economic growth." (Kravčík 2007 p. 93)

1.1 Tamera's Water Retention Landscapes

This research was partly inspired by studying in the Tamera ecovillage in the Alentejo region in Portugal. There, sensitive to the local hydrological processes, the community has created a water landscape landscape comprising of a large lake of 5 hectares and several smaller lakes (Daum 2014).

Engineers and researchers Bernd Müller and Christoph Ulbig teach Tamera's course on *Professional Training in Creating Water Retention Landscapes*. This introductory course covered the principles behind this practice, including exposure to a live build of a small earth dam. In an interview with Ulbig (2014), he said GIS could be of use to "help scale the lakes with regard to the catchment size and the landuse." This leads to an interesting research question: does the literature examine the use of GIS for modelling small earth dams and water retention measures, and if not perhaps there a benefit in exploring how spatial analysis could be of benefit here.

To give some context to the word *small*, Tamera's largest dam is $110m$ in length and rises $8 - 10m$ above the original valley floor. The deepest part of the reservoir is $16m$ and the lake is 4.9 hectares in surface area.

1.2 Research Objectives

The work has a number of objectives. The first is to outline the need for research in this area. Second is to develop a tool that can be used to model small earth dams as part of a water retention landscape. The intention is to study using actual elevation data and to show how users of this system can import local data. This system will be built using open technologies that anyone can use and are free to modify and improve, and as a basis for future work. Special attention will be given to high risk arid and semi-arid lands (ASAL) regions such as East Africa and Mediterranean Europe.

Based on interviews with engineers Müller and Ulbig in Tamera, the use of the geographic sciences was discussed as to how it could be applied with this field of work. It was felt that, although useful, most of the design work and moving of earth needed both experience and local soil appraisal. This favoured a direct approach and attempting to simulate or model such close contact in GIS seemed of dubious benefit.

However, it was agreed that students and early stage designers could make use of the assistance of an educational tool to model such landscapes. Based on a digital elevation model, an interesting question is how would one best represent small earth dams and other works to model factors such as the size of earthworks and the volume of resultant reservoirs. The scope of this work is to output such a tool.

Ulbig also advises that any such system should be combined with on-site experience and "contact to the land." This lends value to the concept of a decision support tool that students and researchers can use in addition to more hands-on skills. While the small earth dam is a key structure of the conservation work, Ulbig notes that dams should always be accompanied with swales, terraces, and vegetation cover.

1.3 The Open Web as a GIS Platform

A further aim of this research is to develop the solution and conduct the analysis using only open source technologies. The rationale behind this is to foster improvement of the tool and continuation of the research. It is felt that working in an open ecology will allow greater adaption and understanding of any work. This concept is developed more in the next chapter.

To reach the largest possible audience, the tool be will developed as a web GIS. This lowers the barrier to entry, as all that is needed to use the system is a computer with a web browser accessing the GIS server. As Dragićević (2004 p. 79) notes, Internet technology "democratises spatial data, open accessibility and effective dissemination". However, the hope is that the spatial analysis procedures can be 'decoupled' from the web interface and used by other tools.

The success of social and environmental GIS system such as Ushahidi (Okolloh 2009) for managing humanitarian crises and Global Forest Watch (2002) show the effectiveness of developing GIS on the open web.

It is naturally of too large a scope to develop a full scale public website like Open Street Map, Global Forest Watch or Ushahidi. Rather the aim is to describe the tools and procedures used to create suitable models for spatial analysis and interfaces for user interaction. This paper exposes GIS students to the use of a rich web application as an interface to powerful open source GIS technology such as PostGIS.

This report also serves as a development guide for those wishing to expand or maintain this system.

1.4 Research Methodology

The literature on small earth dam design manuals are reviewed to select a concrete set of requirements. The existing methodology on earth and water calculations will be considered in order to compare with the results of the procedures developed.

By considering several open source technologies, it is possible to select the correct stack of software to develop the tool and at each stage, documentation is provided.

It is important to consider the final users of the tool and a section of this manual is concerned with data acquisition and processing, with the result of usable and beautiful thematic map images for the web application.

A full procedural model and system architecture will be produced and each spatial analysis procedure will be detailed outside the body of the document.

By documenting and publishing the source of a research tool, the methodologies will be available for review and the results of a typical spatial analysis will be checked against the literature. It is reasoned that the error rate will be a good indicator of the accuracy of the research methodology here.

Formal Research Aim

> To research and develop an open source web GIS to enable students and designers to model landscapes of small earth dams in order to support the decision making process.

Figure 1.1: View of small earth dam and reservoir, Tamera, Portugal

Chapter 2

Literature Review

This chapter reviews the available literature on the design and construction of small earth dams. The design manuals reviewed deal with a range of concerns, from the social context in countries where such dams are built, to technical matters like location and soil analysis. These engineering issues however are out of the context of this study, which looks to answer questions of a purely geospatial nature, and a review of the literature seeks to extract this salient information. The solution developed can be considered a geospatial tool to be used in conjunction with engineering manuals on the design and construction of small earth dams.

By the end of this chapter a picture emerges of some of the geospatial problems that may be faced by the small earth dam designer and of the requirements that need to be addressed by any solution developed.

2.1 GIS as a Decision Support Tool

An objective of this research is to allow a student to use GIS to solve the problem of site selection for small earth dams. To model this scenario is to employ GIS as a Spatial Decision Support System (SDSS). SDSSs are often characterized by good interfaces, easy customisation, and fitting ill-structured problems (Keenan 2003), which is an accurate description of this problem space.

A most fundamental role of GIS is to allow a user to view geospatial data related to an area of interest and all but the most basic systems allow the user to input data and interact. These basic needs form the first requirement of the solution.

> Solution Requirement 1: Allow the user to view a study area, and input a dam crest.

From this point, the focus must be on creating an accurate model of a dam and its resultant reservoir. The model must accurately reflect similar dams that have been built and studied.

2.2 Review of Design Manuals

The reference text for the construction of small dams is the United States Department of the Interior Bureau of Reclamation (USBR 2006) manual on the design of small dams. This comprehensive document provides a complete manual for the design and construction of dams up to 90m high. As well as earthfill dams, it addresses rockfill and concrete dams and provides criteria for selecting which dam type to construct. Full coverage is given to the physical properties of soil and rock, exploration methods, design principles and many examples of successful small earth dam construction (all in the United States). However, as an engineering document, little attention is given to site selection, save for 'seismotectonic investigations'. Stephens (2010) summarizes the USBR manual as a useful reference, but having design and construction procedures not applicable to smaller 'farm' dams which require less-sophisticated techniques and methodologies.

Leaving aside rock and concrete dams, Nisson-Peterson (2006) classifies small earth dams into three categories:

Charco dams are built on flat land in natural depressions and are usually small and hold only a little water, being subject to evaporation.

Hillside Dams are built into the side of hills and are among the simplest to design and construct.

Valley Dams are larger dams, built into a valley and have the highest potential for water retention. They are also the dams with the highest risk of wash-out.

> Nissen-Peterson gives an example of a valley dam – Kimuu dam in Kenya – suffering a wash-out after a 72 hour heavy downpour, despite having been designed by a team of 25 skilled engineers.

Considering the high risk factor versus high water saving potential, this paper will focus solely on decision support and modelling risk in valley dams.

2.2.1 Project Considerations

Nissen-Peterson (2006) identifies four key questions that should be asked before constructing a small earth dam in order to assess if a project is viable. Those questions are:

1. Will the water be clean enough?
2. How much water is needed?
3. How much water will the reservoir provide?
4. What will the project cost?

These four questions form the core of the decision making process and must be examined individually in order to identify the geospatial components of each question.

2.2.2 Provision of Clean Water

Having clean water in a reservoir depends mainly on what is in the catchment area. Human latrines and animal waste can spoil water in the reservoir, making it dangerous for animals and people to drink. The ability to view a dam's catchment area would therefore enable a designer to locate any potential sources of contamination of the water supply, assuming that that person has sufficient local knowledge. From this the first requirement is defined:

Solution Requirement 2: Allow the user to view a dam's catchment area in order to identify sources of water contamination.

2.2.3 Quantification of Water Requirements

Nissen-Peterson's second question asks for a figure on the amount of water needed. For local communities in ASAL areas this is typically a function of the number of people, the head of livestock and the irrigation requirements. [1]

Evaporation is a major factor in determine water requirements. A study by McMurray (2003) found that around 20% of storage capacity was lost to evaporation from the ponds of small farm dams in southern Australia, with more (25-30%) being lost in hotter summers. In Eastern Kenya, Nissen-Peterson estimates the figure to be closer to 50%.

Naturally, the shape of the reservoir will have an impact on how much water is lost to evaporation. Stephens (2010) regards the surface area of the pond to be an important design consideration, with a narrow deep reservoir having a much smaller evaporation loss than a broad shallow one.

Solution Requirement 3: Calculate a reservoir's surface area (in m^2) to allow for an estimate of water loss due to evaporation.

Seepage must also be factored into the water requirements calculation. Nissen-Peterson gives a rule-of-thumb of 25% loss due to seepage but admits that it is difficult to estimate accurately without knowing more about the soil type under the reservoir. [2]

[1] For example, a family of 8 with 35 animals and $\frac{1}{4}$ acre of crops require 200,000 litres of water during the dry season.

[2] Continuing the previous example: Water usage $= 200,000l = 200m^3$ + estimated evaporation loss $= 400m^3$ + estimated seepage loss $= 200m^3$. The family of 8 would need a reservoir of volume $800m^3$.

2.2.4 Estimation of Water Capacity

Having calculated the water requirements, the decision maker needs to determine if a dam will create a reservoir large enough to meet those requirements.

Solution Requirement 4: Calculate a reservoir's volume to allow the user to estimate water capacity.

Stephens (2010) gives a formula for estimating storage capacity to an accuracy of within 20% using a simplification of reality, assuming the water volume to be an inverted pyramid:

$$Q = \frac{LTH'}{6}$$

Where:

- Q is the storage capacity in m^3
- L is the length of the dam wall at full supply level (FSL) in m
- T is the throwback (the distance from the crest to the foot of the dam), in m and approximately in a straight line from the wall
- H' is the maximum height of the dam, in m, at FSL.
- 6 is a factor that can be adjusted (to 5 or 4) with experience and local knowledge

This formula is a useful guideline estimate, but thankfully, creating a reservoir model in a GIS allows for a more accurate calculation.

Freeboard

The freeboard—the clearance between the reservoir at FSL and the crest of the dam— must be factored into any reservoir volume calculation. Helped by a well designed spillway, this clearance prevents overflowing of the dam in rain storm events. Stephens recommends a freeboard height in the range of $0.75m$ to $1.5m$.

Borrow Areas

If the earth used for the embankment or 'borrow area' comes from within the reservoir, its capacity is increased. Stephen's manual instructs that these borrow areas should be given first preference if possible, then to take earth from the valley sides close to the embankment.

Naturally, this is dependent on the availability of the correct type of soil, however the calculation of reservoir capacity can make the assumption that all the earth required to build the embankment comes from inside the reservoir, increasing water capacity by the volume of earthworks.

2.2.5 Project Cost

Nissen-Peterson's fourth question is: *How much will the project cost?* Bluntly, unless the benefits exceed the costs, there is little point in constructing the dam. Leaving aside the advantage of having a ready supply of water, other economic benefits might include the savings in the value of labour and time used to fetch the water, as well as improved crop yields and cash from sale of animals.

The cost of dam construction–whether with shovel, ox or machine–is basically a function of the volume of earthworks. This is a geospatial operation and therefore can in the scope of the decision support tool.

Solution Requirement 5: Calculate the volume of earthworks to allow the user to estimate cost of dam construction.

Stephens' formula for calculating earthworks is as follows:

$$V = 0.216HL(2C + HS)$$

Where:

- V is the volume of earthworks in m^3.
- H is the crest height of the dam in m.
- L is the length of the dam, at crest height H, in m.

- C is the crest width in m.
- S is the combined slope value. For example, if the slopes of the embankment are 1:2.5 and 1:2, S = 4.5.

The formula is based on areal equations for cross-section and longitudinal sections of the embankment, adjusted by a factor (0.216) based on "empirical evidence" (Stephens 2010).

As with the reservoir volume calculation, GIS provides the ability to model the dam itself and possibly improve on the accuracy of this calculation.

Embankment Slopes

The volume of the embankment depends on the dam height, crest length and the width of the dam slopes.

All earth dams require a flatter upstream slope as this allows the earthworks to resist slumping when saturated with water (Stephens 2010). Slumping occurs due to the horizontal thrust exerted by the depth of water, and is a function of water depth, not volume. Therefore, higher dams require a flatter slope to resist the increased force.

The downstream section of the dam—beyond the impervious core—is not saturated with water and therefore not subject to the same pressures. Nelson (1985) recommends a minimum of a 3:1 slope length to height ratio for the upstream slope and 2.5:1 for the downstream slope, decreasing with the dam height (see table 2.1).

Dam Height	Position	Slope Ratio
Below 3m	Upstream	2.5:1
	Downstream	2:1
3m to 6m	Upstream	2.5:1
	Downstream	2.5:1
Over 6m	Upstream	3:1
	Downstream	3:1

Table 2.1: Slope length to dam height ratio
Source: Nelson (1985)

Embankment Core

A calculation of the earthworks volume must also consider the design of the embankment core. Stephens (2010) recommends the width of the crest be at least $2m$ wide, rising to $3m$ wide for dams over $5m$. However, for the purposes of estimating embankments works it is sufficient to assume a $3m$ crest width in all cases.

Additionally, a good dam needs a good core foundation or 'key' and Nissen-Peterson (2006) advises that the key must be excavated through all layers of sand and gravel until it is "at least $0.6m$ into watertight soil". As the depth of the cutoff is dependent on local soil conditions, it is difficult to predict without hyper-local soil maps, so the volume of the key must be omitted from solution's earthworks volume calculation.

2.2.6 Dam Settlement

Due to the natural behaviour of soil particles, some settling will always occur in dams after construction. Factors that affect the amount of settling are soil type, dam height, compaction and failure to allow for soil settlement increases the risk of a dam breach.

Stephens (2010) recommends the design height of the dam be increased by 5-10% to allow for this. Nissen-Peterson (2006) is more conservative and recommends a settlement allowance of 10% for machine-compacted dam walls and up to 30% where the dam is not compacted.

Save for prompting the end-user for the compaction method, there is no way to apply a settlement allowance scale to the dam model, therefore a 'worst case' scenario of 30% can be assumed. This means that any decision regarding the volume of earthworks required during construction should take into account a 30% increase in dam height overall. This applies not only to the dam wall, but also to supporting slopes, as naturally will also naturally settle after time.

The figure of 30% is relative to the depth of earthworks in that particular location. As shown in figure 2.1, the settlement allowance depends on the underlying topology of the terrain. The risk here is that the height of a column of embankment may not be apparent after construction.

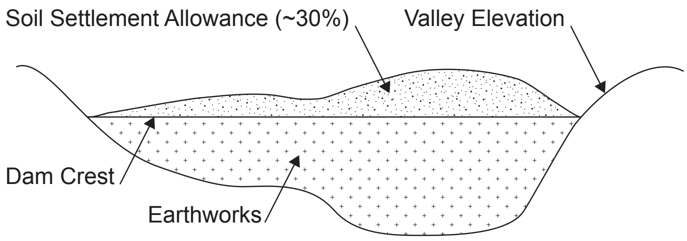


Figure 2.1: Cross-section of earthworks showing soil settlement allowance

There is a strong case here for allowing the decision maker to visualise soil settlement allowance over the whole earthworks, and as this risk has a geospatial component, it can be considered in the scope of this research.

Solution Requirement 6: Allow the user to visualise soil settlement risk over the entirety of the earthworks.

2.2.7 Other Location Considerations

A suitable dam site must be located that can provide satisfactory answers to the above 4 project considerations. However regardless of how much clean water that dam can provide and how much the earthworks cost, the location itself must meet certain general requirements.

Stephens (2010) identifies two such site considerations:

1. Steep valley slopes should be given low priority as dam sites on such slopes are rarely economical.
2. A narrow channel for the dam itself means less earthworks, while side reservoir areas immediately upstream gives a large storage capacity.

Nissen-Peterson (2006) adds the following site consideration:

1. The embankment should be situation at least 100m from any bends in the valley to mitigate against erosion caused by river currents.

While theoretically feasible, it would be difficult to programme a GIS to automatically take the above considerations into account when identifying potential dam sites, and is certainly outside the scope of this study. Instead, elevation data can be displayed in such a way to assist the user in identifying dam sites that satisfy these considerations.

Solution Requirement 7: Display thematic elevation data to the user to allow her to identify steep slopes, narrow channels and valley bends.

2.3 Open source as an Educational Tool

An important goal of this research is to provide an educational resource for students who wish to understand the concepts and methodology in using GIS for small earth dams.

Moore (2002) identifies some benefits of using open source software in education, namely:

- It results in products that supplement and compete with proprietary products.
- It encourages use of standards.
- It nurtures expertise in courseware development.

Developers can develop their own skills and resources appropriate to local needs by availing of the freely available documentation and strong online community in the open source ecosystem (Carmichael & Honour 2002).

'Open Source' software has its roots in the early days of computer science and embodies what Levy (1984) calls the 'hacker ethic'—the desire to creatively modify or improve a computer system. Use of open source software, therefore, encourages students to take code and make their own modifications to add functionality, increase accuracy or improve efficiency. Sharing is permitted and actively encouraged.

While proprietary software usually has a purchase price, open source software is inherently free to use, and such is a lower cost of entry to students who wish to run a GIS. This is especially relevant given the target demographic of this research, many of whom live in developing countries.

Solution Requirement 8: The source code of the solution must be open source, and should facilitate modification and sharing.

2.4 Summary of Findings

This chapter has examined the key literature in this area of research. From that material, the salient GIS-related aspects have been examined and formalized into a set of requirements.

The next chapter will consider the technologies that are available that will assist in the development of the solution to meet these requirements.

Chapter 3

Technology Review

Given the objectives detailed so far, a suitable solution must be designed and developed to meet those goals.

Being an educational tool, open source software is the most suitable choice for developing the solution (see section 2.3). There exists a wide range of open source GIS software which may fulfill the requirements of the project. It is useful to examine some such solutions before choosing a package for the development.

Commercial, closed-source software tends to provide *full-stack* packages providing support at all layers of a software solution. A good example is ESRI's offering of: ArcGIS, a desktop product; ArcGIS Server, the server product, as well as a web application ArcGIS Online. Further components and libraries are provided by the company to manage geodatabases, access catalogs and integrate with web APIs. Although a 'hybrid' ArcGIS / open source model is possible (esri 2011), it is not the objective of this research.

Conversely, open source software tends to be more modular, with each library or component doing one job, and one job well, a philosophy inherited from the early days of UNIX (McIlroy et al. 1978). There are open source desktop GIS software packages: web map servers; spatial database systems; web application frameworks and geospatial libraries. Some of the popular packages are described in later in the chapter.

Prior to that it is useful to review GIS use in similar projects.

3.1 Using GIS to Model Small Earth Dams

3.1.1 A GIS for the San Gabriel Dam

Wang and Ferris (2010) discuss the merits of using a GIS to “archive and display background information to evaluate the structural safety of a dam”. Using the case study of the San Gabriel earth dam in Southern California, they describe the approach taken by the Los Angeles County Department of Public Works to use a GIS to organise, display and share dam safety data. Such data collected included instrument data, geological data and photographs.

The GIS allows the user to “visually correlate these elements on a map”. In the study, data were collected with a particular usefulness to safety analysis and with a view to evaluating hazards that may developer, defining engineering design assumptions and to identify potential failure. The data were also chosen based on availability and quality. For example, in the slope stability analysis, reservoir readings and piezometer levels where inputed to determine safety factors.

The San Gabriel dam is of earth and rocks and measures approximately $95m$ in height and $450m$ in crest length. Although it is not constructed in the rammed-earth manner, and is significantly larger than the dams to which this paper addresses itself, it is a useful case study nonetheless. Crucial to the success of the system was to allow users without specialist GIS software to access the information.

ArcGIS was employed to import and organise the data, which was then exported as static maps to KML (to view in Google Earth) or PDF (to view in Adobe Acrobat).

The paper concludes that visualising the information in a GIS offers many benefits from a paper based system and recommend this approach be applied to other dams. The authors also refer to the availability of free information on the Internet, citing in particular the United States Geological Services (USGS) data sets: the National Elevation Dataset and the National Hydrography Dataset.

This paper is useful when considering institutional processes and procedures, an important factor when convincing stakeholders of the benefits of GIS systems in this industry. However, the paper is short on technical details and does not consider small earth dams.

3.1.2 Estimation of Small Reservoir Storage Capacities in Limpopo River Basin

Sawunyama (2006) estimated small reservoir volumes based on their remotely sensed surface areas. The study was based on 12 small reservoirs in the Limpopo River Basin Zimbabwe, and the results were verified with field studies of the lakes in question.

The findings from the study show that there is a power relationship between remotely sensed surface areas (m^2) and storage capacities of reservoirs ($m3$), given as:

$$Capacity = 0.023083 * Area^{1.3272}$$

The research looks sound, and it was found that 94.6% variation of the storage capacity is explained by surface areas. This was based on the assumed fairly uniform geology and topography where the reservoirs exist. This relationship can be used as a tool in decision-making processes in integrated water resources planning and management, and his findings should prove useful when verifying reservoir area to volume calculations.

For example, applying this formula to Tamera's largest reservoir, introduced in section 1.1 would give the capacity as:

$$Capacity = 0.023083 * 49000^{1.372} = 62840.737512254m^3$$

3.2 GIS and the Open Web

The user of the system must have the ability to input data and view the results of the analysis. The web has proven itself as a suitable platform for offering rich user experiences, and projects such as Global Forest Watch (2002) have shown that rich GIS interfaces can be developed using web technologies. Based on this tool and others like Ushahidi (Okolloh 2009), it seems reasonable to assume that modern web programming can provide every element required for the successful implementation of a web based GIS.

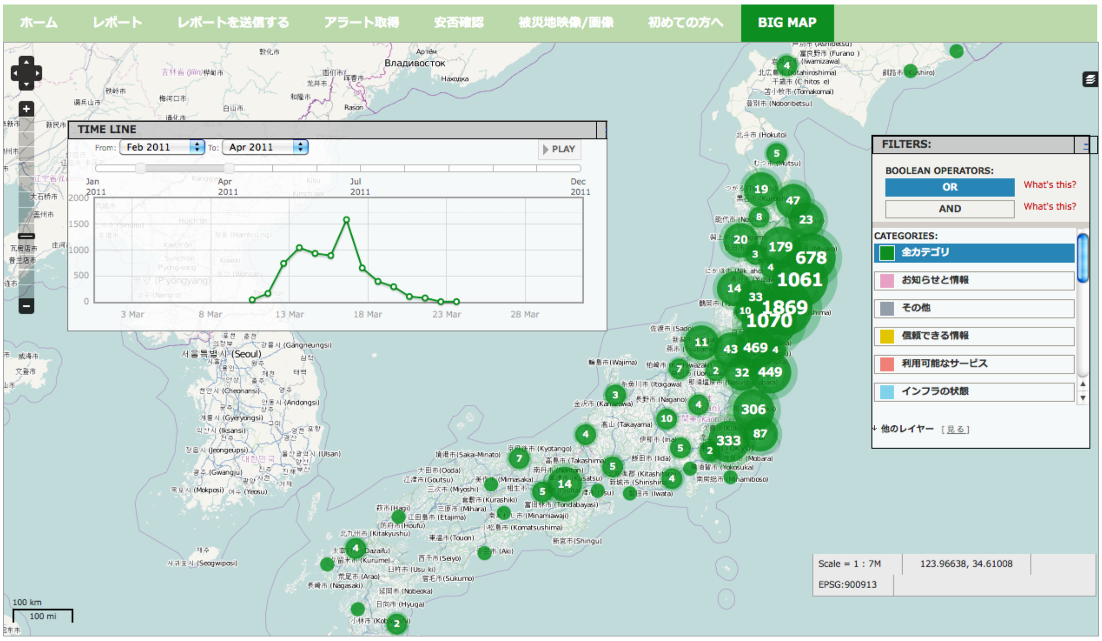


Figure 3.1: Screenshot of the Ushahidi GIS, showing events during the 2011 Japanese earthquake and tsunami. Source: Ushahidi.com

3.2.1 Modern Web Practices

Section 2.3 introduced the benefit of using open source as a means to facilitate the educational application of a tool. Developing to open industry standards has the benefit of encouraging understanding, adoption and modification of the system.

The following general technologies play an important role in delivering the rich web as it is seen today and should be considered as part of a toolset for the development of any open source web app.

HTML5 has led a revolution on the web (Anthes 2012). It includes a range of technologies that allow for rich web apps without the need of browser plugins. For example, HTML5 provides a means for web applications to request the user's geolocation.

Node.js is a platform for "easily building fast scalable network applications" (Node.js 2014). Node.js applications are written in JavaScript, and so has the advantage that the developer may write the front-end and back-end in a single language. Node.js has a large directory of packaged modules (78,095 at time of writing), which makes integration with other libraries easy.

Grunt is a task runner developed by Twitter. Grunt makes the development of web-based applications easier, automating repetitive tasks like packaging styles and scripts and running automated tests (http://gruntjs.com/). Grunt works well with Node.js and has a large ecosystem of plugins.

3.2.2 Review of Open-Source GIS Software

The previous section introduced some general tools and standards for modern web development. The following is a review of such technologies with a GIS flavour.

QGIS is a desktop GIS program that provides data viewing, editing and analysis capabilities. It supports vector and raster data types and provides good integration with other open source tools such as PostGIS, GRASS and MapServer. It is well maintained (the current version is QGIS 2.2 and was released in February 2014) and is used internationally in professional and academic environments (Barton & Brady 2009). As well as viewing data, QGIS can act as a OGC-compatible server (see section 3.2.3), handling WMS and WMF services (as well as the GeoJSON format) which dovetails with the QGIS web client, allowing projects to be seamlessly published to the web (QGIS 2014).

QGIS has the ability to load plugins and there is a wide variety available, including commercially developed plugins for water distribution network simulation (Tsimpiris 2014), which, while looking useful in some domains, is not applicable to the current research.

GRASS GIS (Geographic Resources Analysis Support System) is a desktop client and library capable of handling "raster, topological vector, image processing and graphic data" (Neteler & Mitasova 2008). GRASS has full integration with PostGIS and supports a wide range of data formats via the GDAL and OGR libraries, including ESRI Shapefiles, GeoTIFF and GeoJSON as well as a comprehensive module ecosystem (GRASS 2014). GRASS can be used as a desktop client or can be integrated with other software, such as QGIS.

Geoserver is a GIS server written in Java and designed for high interoperability. It allows users to input, process and publish geospatial data. Geoserver is the reference implementation of the OGC and so naturally has very good support for mapping standards such as WFS, WMS and WCS. Geoserver can read a variety of data formats, including PostGIS, ESRI Shapefiles and GEOTIFF.

Mapnik is an open source mapping toolkit aimed at creating 'beautiful maps' (Pavlenko 2013) including sub-pixel anti-aliasing using the Anti-Grain Geometry (AGG) library. Mapnik utilizes the GDAL library to read a wide range of vector and raster data types, including PostGIS, GeoTIFF and Shapefiles. Mapnik is fast, solid (Miler *et al.* 2010) and used by the OpenStreetMap project as well as others. Mapnik is configured by an XML file or via C++, Python or Node.js language bindings.

Tilelive is a software module to assist in the generation and caching of map tiles, that is the individual square images that are generated from larger maps. For example, if Mapnik was used to merge several raster and vector map layers into a single image, tilelive would be responsible for splicing that single image into a matrix of square tiles at various resolutions. Tilelive fills this gap, providing a single API to server applications wishing to serve tiles. Tilelive delegates the connectivity to sub modules such as tilelive-mapnik. Tilelive is written in JavaScript for Node.js. Tilecache is a caching library that wraps Tilelive.

PostGIS is an open source spatial database based on PostgresSQL. It provides support for importing, storing and querying spatial data in vector and raster form. PostGIS "implements the specified standard" for *Simple Features for SQL* by the OGC (Open Geospatial Consortium 2006a). Being standards compliant means easier integration with third party products, and PostGIS enjoys wide support from various third party open source and proprietary tools including QGIS, Mapnik and Geoserver (PostGIS 2014). The database can use the GDAL and OGR libraries to enable it to import a large range of data formats. As well as adding extra geometry and raster types to PostgresSQL, PostGIS adds functions and operators to allow for manipulation of these spatial types.

A good article on using PostGIS as part of a web GIS stack is Brovelli & Magni's (2003) paper on the developing of an archaeological web GIS using PostGIS and Mapserver.

GDAL is a low-level library for reading and writing raster data formats such as GeoTIFF. It is required by several other software programs listed here, such as PostGIS and QGIS, but may be used on its own via a suite of command line tools. OGR is a related library that deals with vector data formats. GDAL has become so ubiquitous in the open source GIS marketplace that Wilcox (2009 p. 290) of the Open Source Geospatial Foundation opines that it is "open, it provides core functionality, I can't understand how anybody gets anything done without it."

Leaflet is a JavaScript library for creating interactive maps on the web. Leaflet is designed with simplicity, performance and usability in mind (Leaflet 2014). It is a modern, lightweight library making use of standard web technologies such as HTML5 and CSS3. It has a growing plugin ecosystem and is well documented. It has native support for the GeoJSON data format.

OpenLayers , similar to Leaflet, is a JavaScript library which aims to assist development of modern web mapping applications. OpenLayers (2014) is feature-rich, enabling formats such as WFS and GML out of the box, as well as supporting advanced map projections compared to Leaflet (Schütze & Vatterrott 2007). With more complexity comes the price of a steeper learning curve, and Leaflet is favoured by some for it's ease of integration.

3.2.3 OGC Standards

The Open Geospatial Consortium (OGC) is an international body concerned with developing and implementing open standards for geospatial data exchange, processing and services. More than 400 organisations collaborate and contribute towards the standard body's work (McKee 2014).

There are over 30 standards in the OGC baseline, including:

GML and KML are two XML-based standards that define a grammar for expressing geographical features. KML (Keyhole Markup Language) describes geographic features using longitude and latitude coordinates, as well as camera position. GML (Geographic Markup Language) offers a richer set of primitives, including coordinate reference systems, time and coverage.

The Web Map Service (WMS) defines a format that a server may use to deliver spatial information and images over the web. The standard defines three operations: one to return service-level metadata; another to return map images; and a third to return a map legend (Open Geospatial Consortium 2006b). The standard enjoys wide adoption, including support by Geoserver, QGIS and Leaflet.

The Web Map Tile Service (WMTS) is a related standard that aims to standardize the delivery of map 'tiles', that is square chunks of a map that can be individually rendered and cached, saving expensive CPU operations that would require the entire map to be rendered at once. A the time of writing the WMTS has not been adopted by any of the software reviewed here.

The Web Feature Service (WFS) provides a data query mechanism for clients to access and retrieve information at a feature level from a web server. By using standard web protocols for transport and GML for encoding vector data, Peng & Zhang (2004) believe WFS has a great potential for solving two problems with internet GIS; that of interoperability and the presentation of raster images. Peng and Zhang conclude that WFS and GML's greatest advantage is that they are based on XML and are thus efficient and interoperable.

3.2.4 GeoJSON

An interesting geospatial data format has emerged in recent years: GeoJSON. GeoJSON is a schema of JSON a data format borrowing heavily from JavaScript syntax.

Compared to XML, JSON can be considered 'fat-free', with minimal additional characters required to decorate the content (JSON 2014) and is equally interoperable. By extension compared to XML-based geographic data structures, GeoJSON is more lightweight and efficient. As Li (2013) points out GeoJSON is easy for humans to read and write and is easy for machines to parse and generate.

However, being lightweight may come at a cost, as GeoJSON sacrifices some features compared to GML. From the specification (GeoJSON 2008) GeoJSON supports only the basic geometry types: Point, LineString, Polygon, MultiPolygon, etc, with each object containing name-value pairs. GeoJSON also supports specifying a coordinate reference system.

3.3 Selection of Appropriate Technologies

This chapter has examined the current state of play on the open web, with special attention given to open source GIS tools. After this analysis, the most appropriate tools for this research can be chosen. The criteria for choosing the correct libraries are: 1) interoperability and 2) proven strength in open source GIS projects.

For desktop GIS, QGIS looks to be the most fully featured compared to the other offerings. It is interoperable with the other tools in this stack and looks featured enough for most GIS tasks required; importing data, layering, analysis and preparing output. In their review of open source GIS software for water resources management in developing countries, Chen *et al.* (2010 p.273) describe QGIS as "outperforming others" and has a "long list of user contributed plug-ins".

PostGIS is a good choice for the database layer as it enjoys wide integration with QGIS and the other tools and has a proven track record in similar GIS projects such as the web GIS developed by Simeoni *et al.* (2013) for storing and analysing field measurements in river embankments.

GDAL is a dependency of both QGIS and PostGIS and so is required. However, GDAL also provides several standalone command line tools for manipulating geospatial data, such as reprojecting, generating contours, etc.

Mapnik and Tilelive/Tilecache work well together and with the other technologies in this stack. Miler *et al.* (2010) describe the use of Mapnik and Tilecache in the delivery of a fast web mapping solution for Urban Planning and Management of the City of Zagreb, Croatia.

Finally, on the front end, OpenLayers provides more interoperability than Leaflet, and both support GeoJSON. However, Leaflet is considerably lighter and provides a more streamlined API. Together with the suitability of the Leaflet.Draw plugin (Leaflet.draw 2014) in this project, Leaflet appears to be a good choice as a front end mapping library.

All tools support GeoJSON and its easy-to-understand JavaScript-like syntax make it an ideal data exchange format.

This choice of technology will drive the data preparation and the planning and development of the tool, detailed in the next two chapters.

Chapter 4

Data Acquisition and Preparation

> "1:5 000 to 1:12 500 scale photos and 1:25 000 to 1:50 000 scale maps are most suitable for interpretation by eye and stereoscope. Satellite imagery to a suitable scale can also be considered."
>
> – Stephens (2010) of the use of geographic data for the planning of small earth dams.

4.1 Data Acquisition Methods

Various techniques exist for the supply of high precision geospatial data. Even compared to the highest quality photogrammetric data offered by national and commercial remote sensing services (Gesch *et al.* 2002; Airbus 2014), many of these techniques offer very precise data by virtue of being based on the ground or at low altitude. It is worth considering some of these data acquisition techniques in their own right.

Terrestrial Laser Scanning (TLS) is a ground-based technique for collecting high-density 3D geospatial data using a 'point cloud' of data reference points (Khairul 2012). Laser pulses are emitted by the device and their reflections observed by the scanner for their range and intensity. Alkan & Karsidag (2012) have found the accuracy of TLS to be best at close distances with high scanning densities and describe the cost as 'minimum', although specialist equipment is required.

Another form of laser scanning is Airborne Laser Scanning. As may be evident from the name, the laser scanner is mounted on an aircraft which flies over the study area in order to build a point cloud of data. Precision is at least as good as the best NED data Kraus & Pfeifer (1998) describe a study of a forest in Vienna which produced a terrain model with grid spacing of 3.125m and accuracy of $\pm 10cm$. Furthermore their study describes an algorithm to evaluate terrains with tree cover. As with TLS this technique requires specialist laser scanning hardware and the total cost must include the pilot and aircraft fuel.

Khairul (2012) suggests the unmanned aerial vehicle (UAV) as a possible source of low-cost high quality data, stating that the UAV has become popular among researchers looking to solve geospatial problems especially with regarding to decision making. In a study the researcher finds that using UAV images result in an improvement over LiDAR, TLS and ALS, with low cost, less time and less labour needed. Khairul and Anuar (2011) state that an inexpensive digital camera can produce elevation maps with 'moderate' accuracy. The commercial UAV mapping service DroneMapper (dronemapper.com 2014) can produce terrain models with an accuracy of $6cm$ per pixel.

Clearly there are large differences in the accuracy, cost and timeliness of the various acquisition methods, which are summarised in table 4.1.

Method	Precision	Accuracy	Cost	Timeliness
Satellite	10m – 3m	?	?	?
LiDAR	?	?	?	Immediate
TLS	?	Best at close distances	Minimum, but specialist equipment required	Immediate
ALS	3.125m	±10.0cm	Specialist equipment, aircraft rental.	Immediate
UAV	6.0cm	moderate	Low cost	Immediate

Table 4.1: Comparison of elevation data acquisition methods
Source: Various

4.2 Selection of Sample Data

The previous section described various methods for data acquisition that could be utilized by students or designers needed to acquired data for analysis by this system.

Having chosen the type of data acquisition most suitable for the study area, the next step is the actual acquisition of that data. Naturally, this varies depending on the data acquisition method, but must ultimately result in a raster file containing altitude data for the study area. This is true also for the case study areas that were chosen.

To demonstrate use of the system and to select suitable study areas, sample DEM data was downloaded from two sources: The National Elevation Dataset (Gesch *et al.* 2002) and WorldDEM (Airbus 2014).

The two study areas were selected to match closely the context of this work, i.e. to target ASAL regions in particular. From NED, a study area near Napa Valley was chosen, and from WorldDEM, a study area in Saudi Arabia was selected.

The first stage in processing any dataset is to look at the metadata in order to ascertain salient details about the source, precision, accuracy and timestamps. Unless the data is primary source, it should come with some documentation, perhaps in text or PDF format.

4.2.1 National Elevation Dataset

Downloading data from NED comes with a 'Data Dictionary' PDF file which gives details about the bundled data, titled *National Elevation Dataset (NED) 1/9th Metadata Field Definitions*, summarised below:

April, 2013 was the release date.

LiDAR or "other active remote sensing technique" was the method of acquisition.

Approximately $3m$ resolution. After processing the datasets are resampled to a common resolution, i.e. 1/9-arc-second or approximately 3 metres.

1m elevation unit (the vertical or z unit).

Accuracy statistics are available in RMSE units. RMSE is the Root Mean Square Error of the elevation estimates, or the square root of the average of the set of

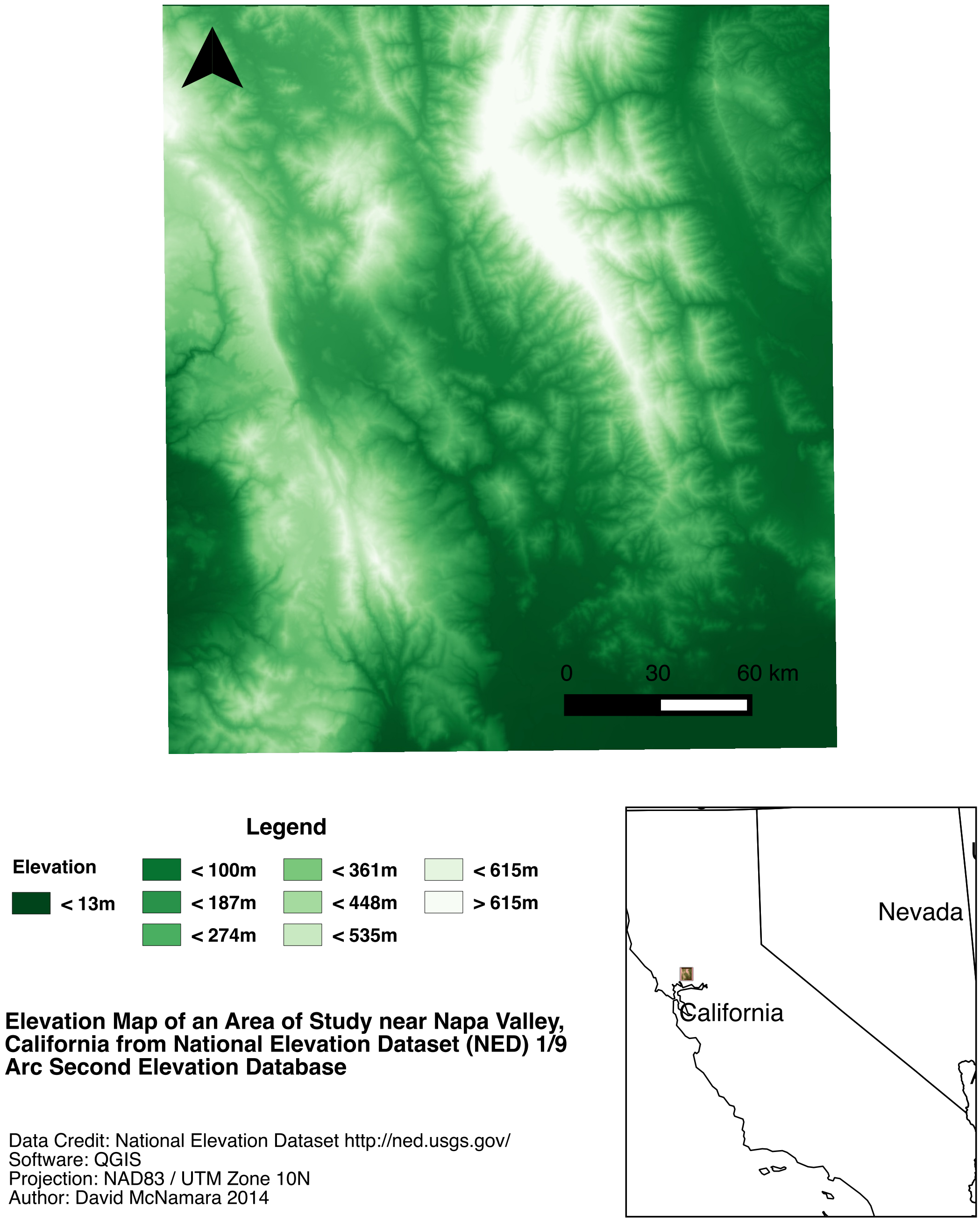


Figure 4.1: National Elevation Dataset study area

squared differences between dataset coordinate values and coordinate values from an independent source of higher accuracy for identical points.

A good deal can also be learned from the raster files themselves, given that well prepared data will have meta data embedded. Using the `gdalinfo` tool to inspect the NED raster files gives the following details:

8112 by 8112 image size.

North American Datum 1983 coordinate reference system.

0.00003 degree pixel size.

A bounding box with 4 coordinates is given.

Statistics such as maximum and minimum altitude are given.

4.2.2 WorldDEM

Similarly, the WorldDEM sample data package comes with a PDF document with the following details about the dataset:

Northcentral Saudi Arabia, Al-Qassim province is the general location. Listing of nearby cities and a description of the land - 'Desert area with agriculture infrastructure (fruit and wheat production)'

576m - 737m elevation range, and a bounding box.

Details about the process of editing the DEM after acquisition: Radar-typical artifacts ('spikes' and 'wells') are removed; small voids are interpolated; lakes and reservoirs are set to a single elevation; Rivers and canals are flattened with monotonic flow.

12m grid spacing

Vertical accuracy is $< 2m$ where slopes $< 20\%$ and $< 4m$ for slopes $> 20\%$. Horizontal accuracy is $< 10m$.

1m elevation unit

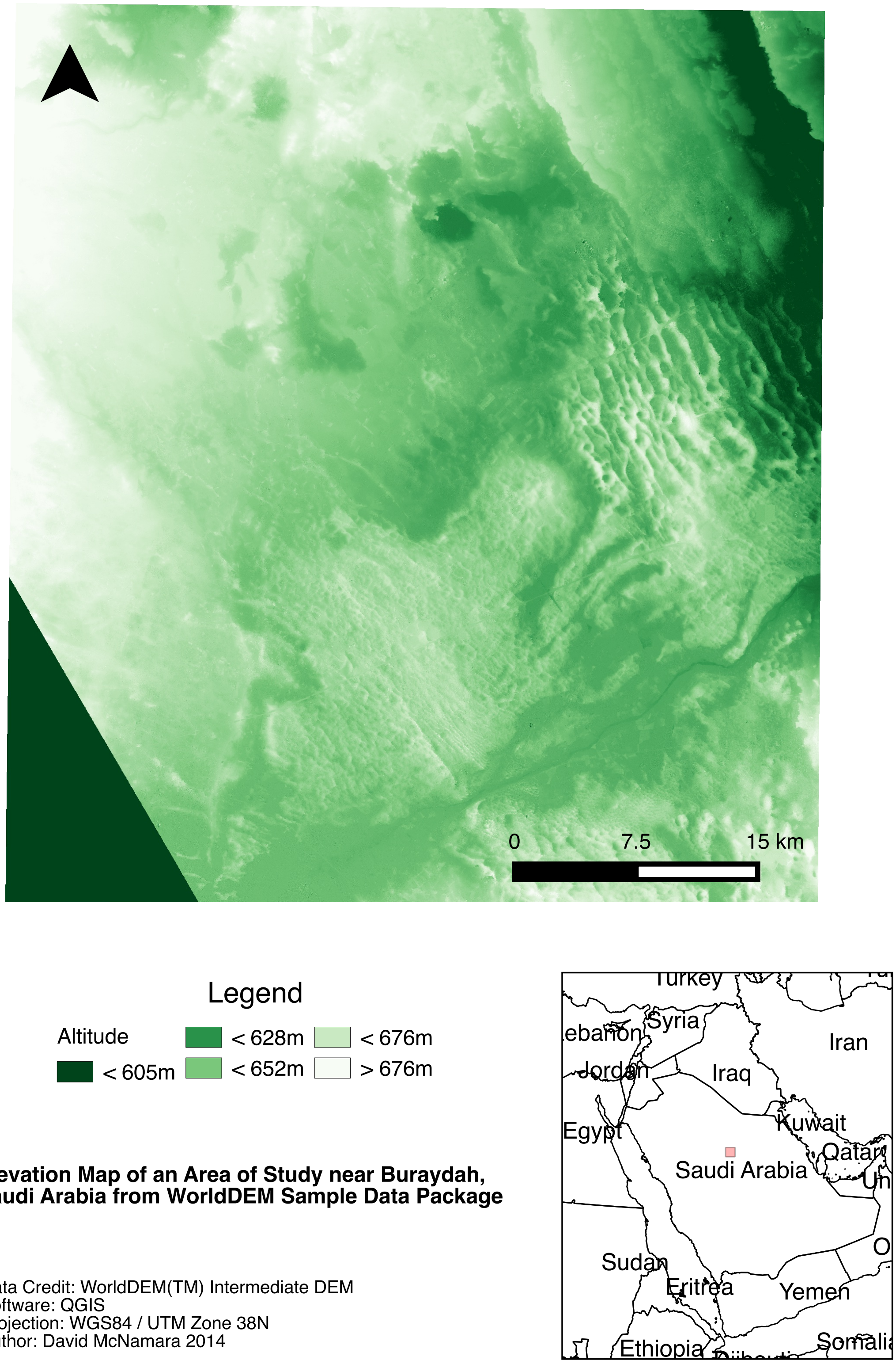


Figure 4.2: WorldDEM study area

4.3 Preparation of Data

Having acquired the source data in raster format, the aim now is to process that data so that it is a) consumable by the user and b) useful in spatial analysis procedures.

Figure 4.3 shows the data import process, each step of which is explained following.

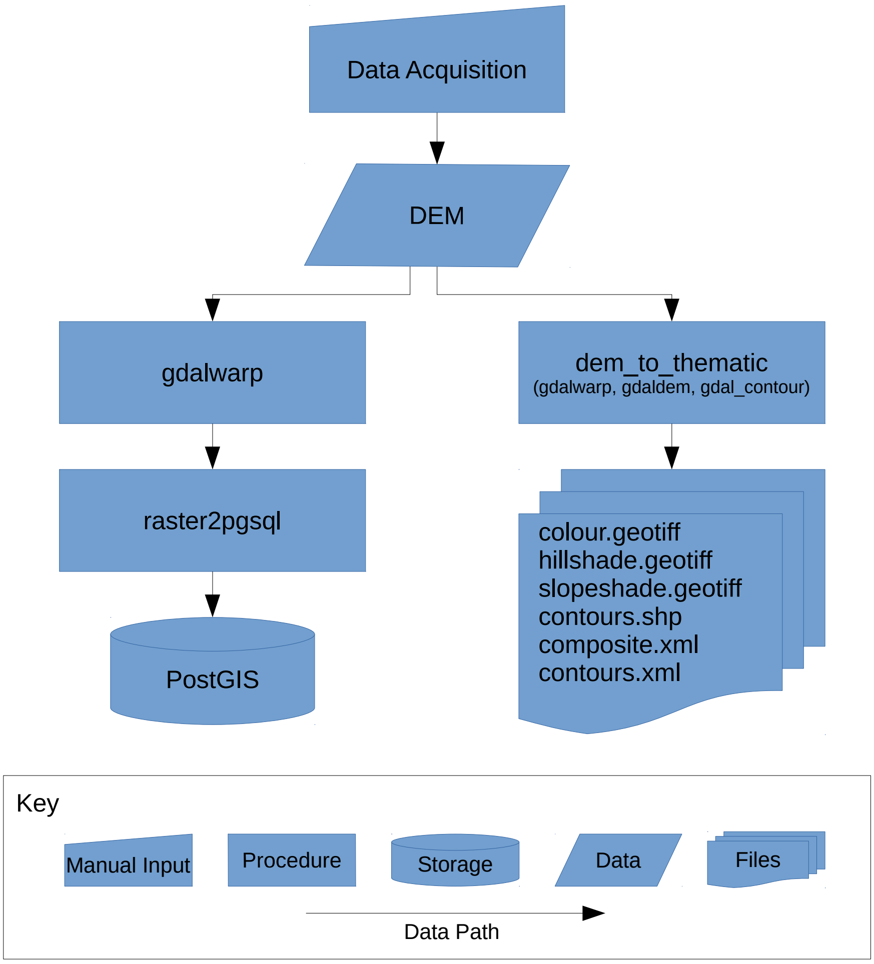


Figure 4.3: Data import procedure.

4.3.1 Reprojecting the Source Raster

With WGS 84 has been as a suitable coordinate reference system to store both raster and vector data in the database. This projection is suitable asit is widely used, has global scope is the reference coordinate system used by GPS. This implies its applications and limitations are well understood.

Some source data will invariably be another coordinate system and will need to be reprojected to WGS 84 before spatial analysis can be done. Although possible to reproject source data each time it is needed, it makes sense for performance reasons to reproject once and store the data as so.

GDAL provides the `gdalwarp` tool for image 'mosaicing, reprojection and warping' (Open Source Geospatial Consortium 2014). The command for reprojecting the NED raster from North American Datum 1983 to WGS 84 is as detailed in procedure 4.1.

Command

```
gdalwarp -s_srs EPSG:4269 -t_srs EPSG:4326 -r bilinear ned-nad83.img napa.img
```

Options

gdalwarp The executable itself.
-s_srs EPSG:4269 The source spatial reference system, NAD83 using the EPSG definition.
-t_srs EPSG:4326 The target spatial reference system, WGS 84
-r bilinear The resampling method. Bilinear is a reasonably fast method that gives good interpolation results.
ned-nad83.img The name of the input raster file.
napa.img The name of the output raster image file. In this case napa was chosen as study area is near Napa Valley in California.

Procedure 4.1: Use of `gdalwarp` to reproject a source raster

4.3.2 Importing the Raster to PostGIS

Having reprojected the raster to WGS 84, it can be imported into the database. The command to import the raster image should take into account the database schema defined – the table and column parameters should correspond to the data and column names specified when creating the database (the schema of which is listed in appendix D. A sample of import command is described in procedure 4.2.

Command

```
raster2pgsql -s 4326 -a -n area_name napa.tif areas | psql gis
```

Options

raster2pgsql The actual executable that does the import. This executable is provided by PostGIS for this purpose.
-s 4326 The spatial reference system used by the raster: EPSG code 4326, i.e. WGS 84.
-a Appends raster to an existing table. This option is used if the table has already been created.
-n area_name The name of the filename column. This has the effect of putting 'napa' into the column 'area_name'. This is useful for naming a study area.
napa.tif The source raster file.
areas The name of the table to import the raster into.
— psql gis This directs the output of the raster2pgsql (which is sql, given everything goes correctly) into PostGIS, assuming a database called 'gis' and with correct database credentials.

Procedure 4.2: Use of `raster2pgsql` to import a source raster

4.3.3 Preparing The Thematic Maps

The previous section described the process for importing the raster image into the database. The name 'image' in this case can be misleading as it is not suitable for viewing as an image proper; grid values represent altitudes not colour ranges.

In order to present the elevation model to the user, the grid data must be reclassified to a colour index. In its most simple form, a reclassification would map elevation data to grayscale, with the point of lowest altitude mapped to white and the point of highest altitude being black (or vice versa). However, this would not make for compelling viewing nor would it be particularly informative in giving hints to the user as to the 'lay of the land', a requirement listed in section 2.2.7.

Various techniques exist for processing elevation data in order to create a more appealing cartographic relief map for user consumption.

One common technique is to create a hillshade or shaded relief. This works by simulating a shadow thrown upon a raised relief map. Hillshading software typically allows for the light source to be shown from any angle, however by convention the light source appears to come from the north east (Imhof 2007).

Hillshading provides some nice indications of crests and valleys, but the output is still in greyscale. It would be beneficial to use colour as another means of transmitting information. Colour relief or hypsometric tints depict elevation as bands of colour, enhancing elevation zones so map readers can better see differences in relief (Buckley 2008). A colour map is selected to indicate what ground cover may exist at that elevation, for example white for high peaks, brown for rocks and scree further down, gradually changing to lush green in the valleys.

Sandvik (2012) specifies such a colour map for use with gdal and suggests combining it with a slope map where flat terrain is more white and steep terrain tends towards black. He also lists a mapnik configuration file that creates a suitable blend of hillshade, colour relief and slope into a single thematic map.

Sandvik's work is the inspiration for the data data processing procedure defined in procedure 4.3. An example output of the procedure for a study area near Napa Valley, California is shown in figure 4.4.

Definition

Name	dem_to_thematic.sh
Inputs	study_area_raster as GeoTIFF in EPSG:4326 projection
Outputs	hillshade as GeoTIFF, slopeshade as GeoTIFF, colour_relief as GeoTIFF, contours as ShapeFile, composite.xml and contours.xml mapnik configuration files.

Procedure

1. Use `gdalwarp` command to map from input map projection (i.e. EPSG:4326 ('WGS 84') to EPSG:4269 ('Web Mercator')).
2. Use `gdaldem color-relief` command to generate a color relief map based on suitable altitude to colour mappings.
3. Use `gdaldem hillshade` command to generate hillshade.
4. Use `gdaldem slope` command to generate slope raster, then `gdaldem color-relief` on that raster to map slope to a colour index.
5. Use `gdal_contour` command to generate a contour map of the input raster, at 25m contour line intervals, writing altitude into the *height* column of the attribute table.
6. Write composite.xml mapnik configuration file with suitable opacity mix for the three raster layers and contour layer.
7. Write contours.xml mapnik configuration file with suitable styling for contours lines and altitude labels.

Procedure 4.3: Processing a DEM to a thematic map

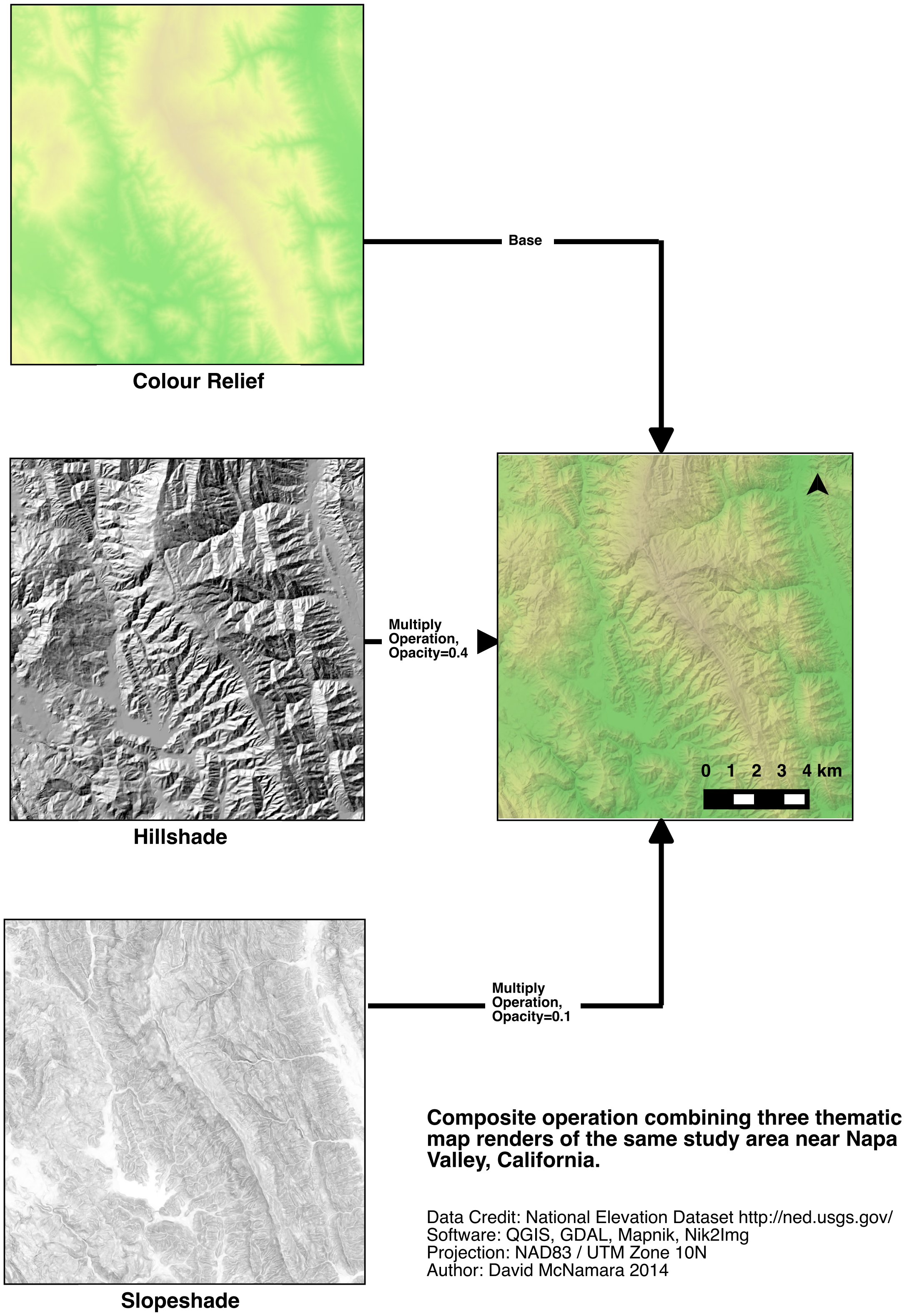


Figure 4.4: Sample output of composite map

4.3.4 Preparing The Tile Layers

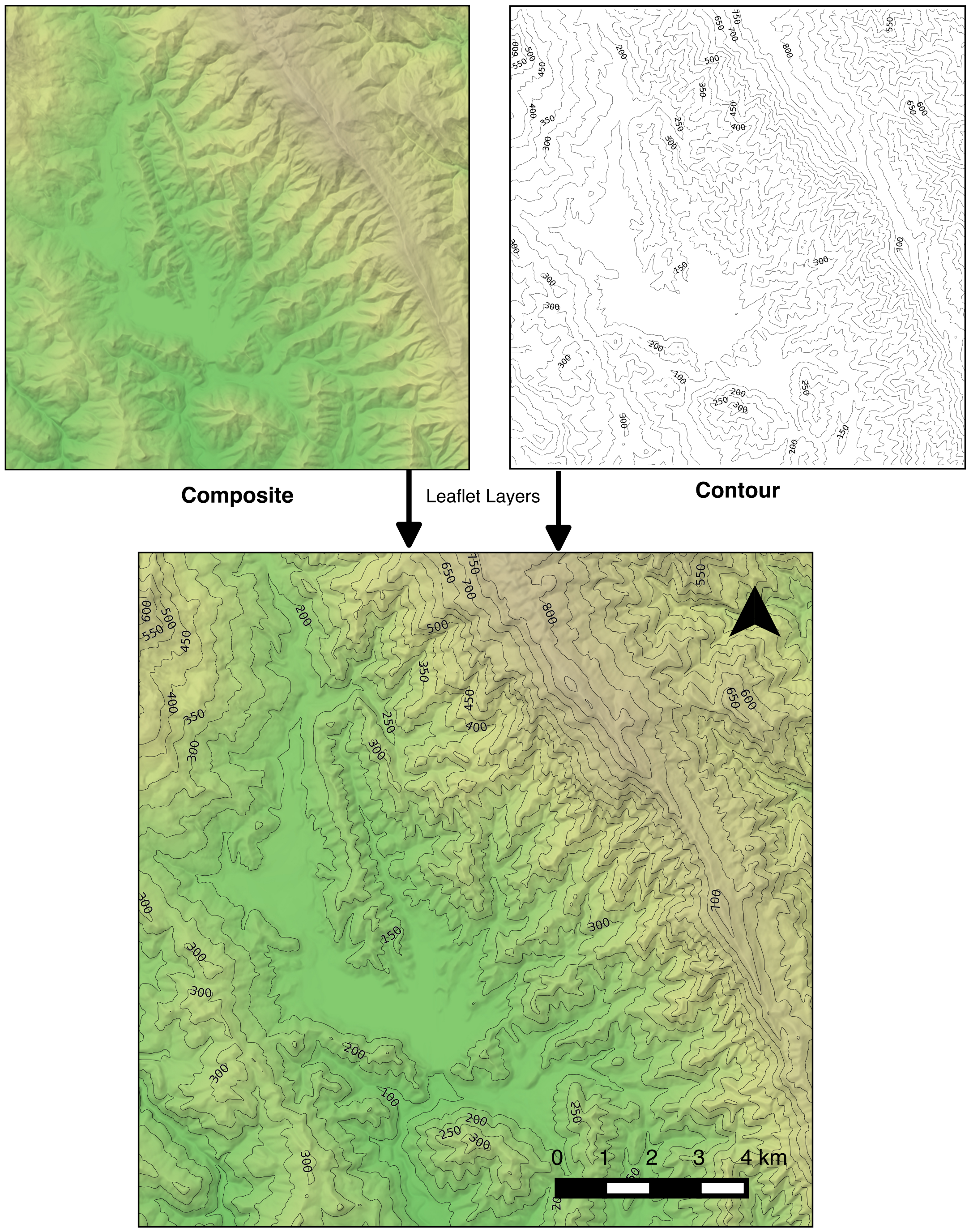
Once added to the database with the script, the new study area will automatically appear in the web application. The details of how this happens are listed in section 5.2.

Note that for each study area, two layers are rendered by the client: the composite of colour relief, hillshade and slopeshade as detailed in the previous section; and the contour map. These layers are of type raster and vector respectively, so should be maintained separately should the client wish to optimize rendering with regard to each layer type. A sample output is given in figure 4.5.

At this point, additional metadata should be saved for each study area, as this information has to be displayed to the user for informative and legal reasons:

1. *description* should be a textual description about the location of the study area. This will be presented to the user, e.g. “Napa Valley, California”.

2. *attribution* should list any attribution or license data that must be displayed for legal reasons when the map is reproduced, e.g. “National Elevation Dataset 2013”.



Map showing output of 2 leaflet layers: a mapnik composite of colour relief, hillshade and slopeshade; and a mapnik rendering of a contour shape file.

Data Credit: National Elevation Dataset http://ned.usgs.gov/
Software: QGIS, GDAL, Mapnik, Nik2Img, Leaflet
Projection: WGS84 Web Mercator (Auxiliary Sphere)
Author: David McNamara 2014

Figure 4.5: Sample output of contour layer and composite map layer

Chapter 5

Solution Development

Given the requirements defined in chapter 2, a solution must be developed to satisfy those. A successful GIS should also aim to be user-friendly, aesthetically pleasing and intuitive to use. As the GIS is open source, code legibility and elegance is also important and thought should be given to students who wish to inspect or maintain the source code.

This chapter examines the requirements with the aim of planning and developing a solution to satisfy them. A procedure model is designed and each spatial analysis procedure is detailed. The chapter concludes with a description of the system architecture.

5.1 Analysis of Requirements

Below are the system requirements established in chapter 2:

1. Allow the user to view a study area, and input a dam crest up to $100m$ in length.
2. Allow the user to view a dam's catchment area in order to identify sources of water contamination.
3. Calculate a reservoir's surface area (in m^2) to allow for an estimate of water loss due to evaporation.
4. Calculate a reservoir's volume to allow the user to estimate water capacity.

5. Calculate the volume of earthworks to allow the user to estimate cost of dam construction.
6. Allow the user to visualise soil settlement risk over the entirety of the earthworks.
7. Display thematic elevation data to the user to allow her to identify steep slopes, narrow channels and valley bends.
8. The source code of the solution must be open source, and should facilitate modification and sharing.

With the exception of the final two requirements—which have been addressed in previous chapters—the remaining requirements all look to be the end result of some sort of spatial analysis or data processing procedure. Each procedure cannot be viewed in isolation and depend on the output of others. For example, the procedure to calculate the reservoir volume depends on the output of the procedure to calculate the reservoir surface area.

An analysis of the flow of data and procedure dependencies resulted in the procedural model diagram in figure 5.1. This flow chart shows the system inputs, procedures and outputs. From this, individual procedures can be isolated and the remainder of this section addresses the procedures implemented to fulfill each requirement

5.1.1 Allow the user to view a study area, and input a dam crest up to $100m$ in length

The previous chapter described the data acquirement and processing, which resulted in thematic maps for the study areas. These maps are ready to be viewed by a user in QGIS or via the web interface (described in section 5.2).

In the web interface, the user is given the ability to define one or more dam crests. A dam crest is a single line segment with a start point and end point. In this model, curved or multiline dam crests are not supported.

Given that the start point can be any point on the study area as input by the user, the end point must be a nearby point on the map with the same altitude. The

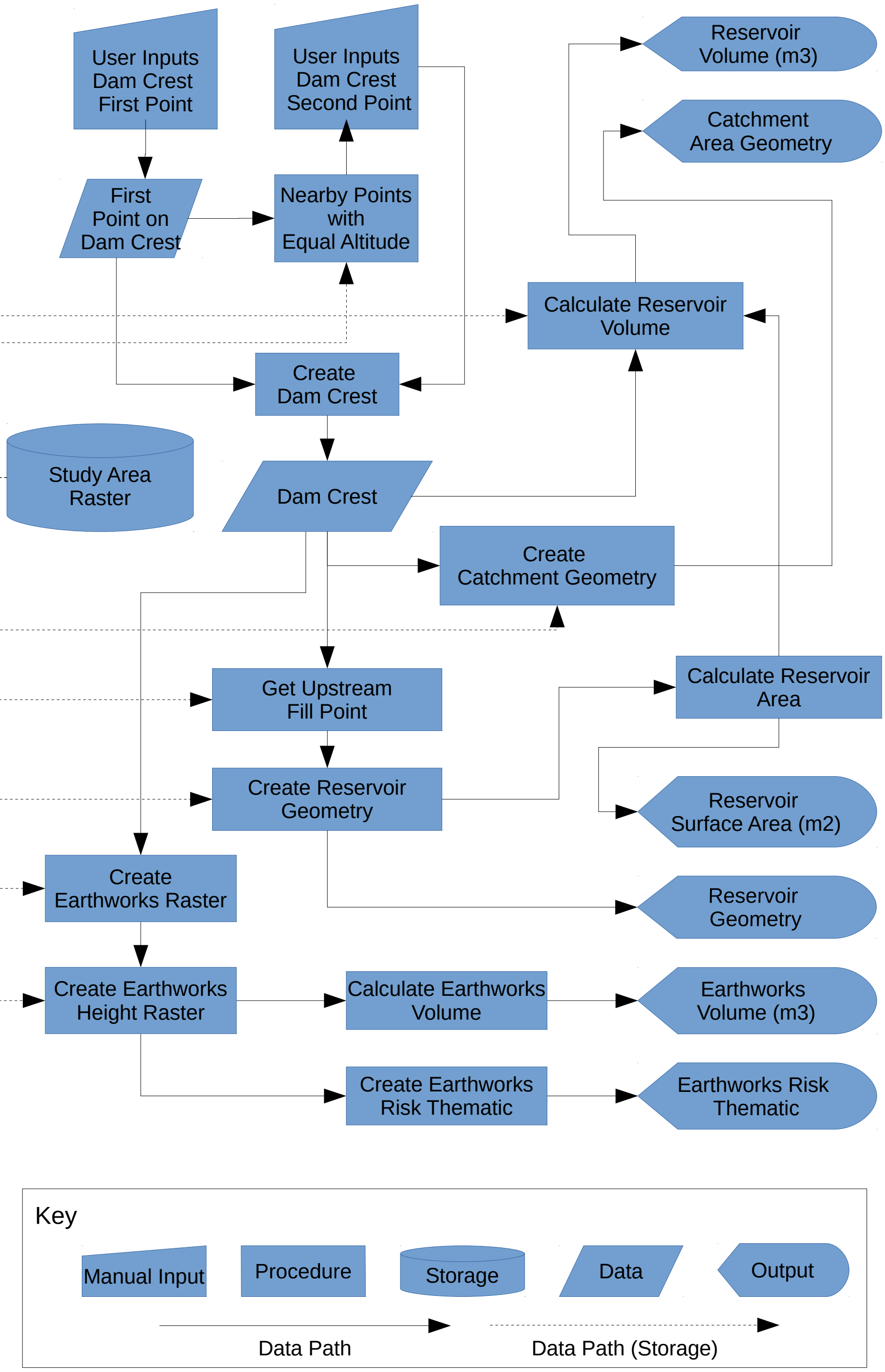


Figure 5.1: The procedural model
Some data points and procedures have been omitted for clarity.

user must be assisted in choosing the end point so as to enable the creation of a valid dam crest.

Procedure 5.1 describes a PostGIS procedure to create a table of points near a given point in a study area.

Definition

Name	points_nearby_equal_altitude
Inputs	ref_point as geometry, study_area as raster
Outputs	nearby_points as table of geometries

Procedure

1. Get altitude of ref_point on raster.
2. Create buffer around ref_point.
3. Clip study area raster to buffer.
4. Perform map algebra on clipped study area, returning only points with similar altitude (see note 2) to ref_point.
5. Convert points from raster co-ordinates to real world points and return.

Notes

1. The procedure is limited to a fixed buffer around the start point. As stated in section 2.2.7, only small dams of maximum length $100m$ are of interest, and this buffer provides a good approximation. The buffer serves to limit the number of points returned and to ensure the procedure executes optimally.
2. A $1m$ matching threshold is specified in the algorithm. This means that points within $1m$ altitude of the start point are returned. Depending on the precision of the input raster, attempting to match points with an exactly equal altitude may not return any points. Subsequent procedures must take into account that the start and end points of the dam crest may differ in altitude by up to $1m$.

Procedure 5.1: Nearby points with equal altitude

The user should be given feedback when creating the dam crest as to the final length of the dam. As this information must be presented to the user as she moves the mouse to examine potential dam end points, it would be suboptimal to request a server-side procedure for distance every time the mouse is moved. Therefore, this procedure 5.2) is best performed on the client side.

Definition

Name	distance
Inputs	point1 as geometry
Outputs	Distance as integer in metres

Procedure

1. Use Leaflet's distanceTo() function from point1 to point2.
2. Return distance.

Notes

1. The Leaflet library (as described in section 3.2.2) implements a distanceTo function on each point. The function returns the distance in metres to another point using the Haversine formula (Leaflet 2014).

Procedure 5.2: Distance between two points

5.1.2 Allow the user to view a dam's catchment area in order to identify sources of water contamination

As described in section 2.2.2, the catchment area or watershed is essential in identifying upstream points of contamination. From a water safety aspect, this may affect the viability of the reservoir.

Hydrology literature references several algorithms for determining the upstream catchment area from a specified point on an elevation model. ArcGIS (2014) defines a watershed as "the upslope area that contributes flow to a common outlet as concentrated drainage," and calculates this using a steepest downslope neighbor Single Flow Direction (SFD) algorithm, i.e. water flows from one cell to another in the direction of the steepest slope.

GRASS's watershed function uses the A^T algorithm described by Ehlschlaeger (1989) which promises "more accurate results in areas of low slope".

A very simple algorithm is Multiple Flow Direction (MFD). This assumes water will flow to all neighbouring cells of a lowest slope, which is not as accurate as the above methods but can be considered suitable for indicative purposes in an application such as this. The benefit of this algorithm is that it is relatively simple to understand and implement: for a given point, water can be assumed to flow from any neighbouring point with a higher altitude. Using a recursive function, a rough watershed can be derived. An implementation of this algorithm is in procedure 5.3.

Definition

Name	create_watershed
Inputs	dam_crest as geometry, study_area as raster
Outputs	watershed as geometry

Procedure

1. Make an empty watershed raster using width, height and cell size as study_area, all cells have value = 0.
2. For each point on the dam_crest: Get position in raster of that point and altitude of study_area at that point.
3. Starting with this point, 'fill' the watershed with the recursive function:
4. (a) If point is already in watershed, return.
   (b) If altitude at this point is less than altitude passed to function, return.
   (c) Set cell value in watershed raster at this position = 1.
   (d) Step into this recursive function with altitude of this point and position of point one grid unit to north.
   (e) Step into this recursive function with altitude of this point and position of point one grid unit to south.
   (f) Step into this recursive function with altitude of this point and position of point one grid unit to east.
   (g) Step into this recursive function with altitude of this point and position of point one grid unit to west.
5. Convert water raster to geometry where value = 1 and return.

Procedure 5.3: Create catchment geometry

5.1.3 Calculate a reservoir's surface area (in m^2) to allow for an estimate of water loss due to evaporation.

In order for the designer to factor water evaporation into the decision making process, estimating the reservoir's surface area is essential. To calculate the surface area of the lake (see procedure 5.8), the lake geometry must first be modelled. This in turn requires the earthworks to be in place.

As described in section 2.2.5, the earthworks consists of the dam wall, the upstream slope and the downstream slope. The inclines of the slopes are dependent on the dam height.

Procedure 5.4 aims to create a raster model of the dam in isolation, i.e. with reference to, but separate from the surrounding area. The reason for this is to allow for further processing and calculations. The raster defines data points containing crest and slope altitudes where the dam is present, and NODATA elsewhere.

The altitude of a point on the dam crest can be calculated using the following formula:

$$altitude\_at\_point = altitude\_at\_crest - (distance\_from\_dam/consequent\_of\_slope\_ratio)$$

For example, for a $6m$ high dam with dam crest altitude of $100m$, the altitude of the upstream slope at $12m$ away from the crest would be:

$$altitude\_at\_point = 100 - (12/3) = 96m$$

Procedure 5.5 calculates the maximum dam height, essential for choosing the correct slope ratio. The maximum height is defined as the height in metres between the altitude at either point on the dam crest and the lowest point on the elevation map as traced by the dam crest.

Once the earthworks are modelled, it is possible to determine the extent of the reservoir that will form 'behind' the dam. To find which side is behind, procedure 5.6 describes a method of finding the upstream side of the dam.

With human eyes, it is relatively obvious where a lake will form should a dam be placed on a landscape, but to define an algorithm for such requires some thought. The procedure uses a version of the common 'flood fill' algorithm. The essence of this algorithm is a recursive function which takes a point and 'grows' outwards given a particular evaluation expression at each point in the raster.

While traditional flood fill algorithms replace one colour with another in a study raster, this variation does not replace values in the study raster. Instead, a second raster is used to record positions in the study raster that have an altitude lower than the dam altitude. The function stops recursing when it encounters a point in the study raster that is of a higher altitude than the dam altitude.

An illustration of the reservoir flood fill algorithm is shown in figure 5.2 and the procedure to create the lake geometry is described in procedure 5.7.

Definition

Name	create_dam_raster
Inputs	dam_crest as geometry, study_area as raster
Outputs	dam as raster

Procedure

1. Get altitude from study_area of first point in dam crest.
2. Create raster from dam_crest geometry, setting altitude on points where dam_crest is present and NODATA elsewhere.
3. Get upstream_fill_point using procedure 5.6
4. Starting with upstream_fill_point, fill the upstream dam slope using this recursive procedure:
5. (a) If mask at point is set, return.
   (b) If altitude at point is higher than dam crest, return.
   (c) Calculate altitude of slope at point using formula as above.
   (d) If altitude of slope is greater than altitude of study area at that point, then set altitude.
   (e) Set mask at point.
   (f) Step into this recursive function with point one grid unit to north.
   (g) Step into this recursive function with point one grid unit to south.
   (h) Step into this recursive function with point one grid unit to east.
   (i) Step into this recursive function with point one grid unit to west.
   (j) Return.
6. Repeat recursive procedure for downstream fill point.
7. Return dam as raster union of dam crest, upstream slope and downstream slope.

Notes

1. The precision of the output raster is equal to the precision of the study area raster. Issues with precision are discussed in section 4.1.

Procedure 5.4: Create an earthworks raster

5.1.4 Calculate a reservoir's volume to allow the user to estimate water capacity

The reservoir's volume is indicative of how much usable water is retained above ground by the dam. A formula to estimate water volume given some input factors is given in section 2.2.4, however a more accurate figure may be obtained by performed spatial analysis on the reservoir model itself.

Similar to the calculation in section 2.2.4, the algorithm (procedure 5.9) works by calculating the height of water in each 'slice' of a reservoir raster, summing the total and multiplying by the cell size. This algorithm assumes the lake is at maximum

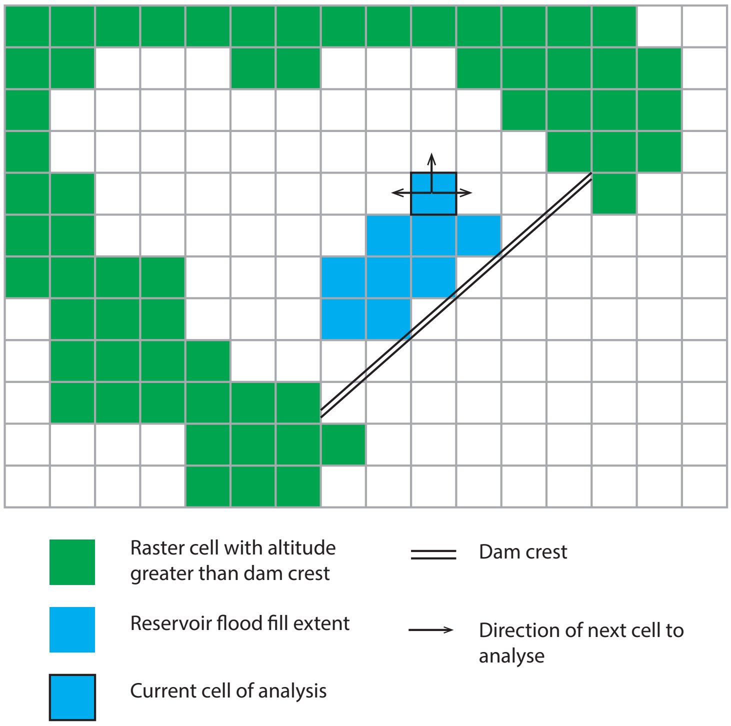


Figure 5.2: The reservoir flood fill algorithm

volume (up to the freeboard line).

5.1.5 Calculate the volume of earthworks to allow the user to estimate cost of dam construction

A key requirement of the system is to accurately calculate the volume of earthworks required to construct a particular dam. A formula is given in section 2.2.5, estimating volume based on dam height and length, however it may be possible to improve on this accuracy given a more accurate model of the dam as per the previous procedures.

By calculating the volume of earthworks in each grid cell 'slice' of the dam raster and then summing theses values, a total amount is derived.

The aim of procedure 5.10 is to estimate how much earth must be moved during construction (rather than final, settled volume), so must therefore use the output of the procedure to calculate earthworks adding the 30% allowance for soil settlement.

Definition

Name	dam_height
Inputs	dam_crest as geometry, study_area as raster
Outputs	height as double precision

Procedure

1. Create dam_crest raster by converting the dam crest geometry to a raster using the altitude of the study area at the first point of the crest.
2. Using map algebra, create dam_height raster with each cell having the value rast2 - rast1 where rast1 is the altitude of that cell in the study_area raster, and rast2 is the altitude of that cell in the dam_crest raster.
3. Get the maximum value from dam_height raster.

Procedure 5.5: Calculating dam height

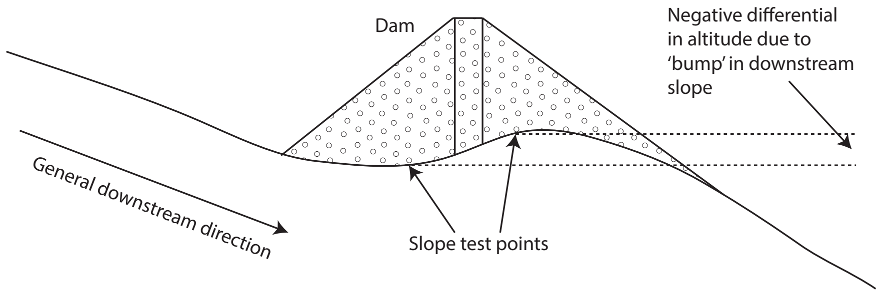


Figure 5.3: Example of potential failure in upstream fill point procedure

5.1.6 Allow the user to visualise soil settlement risk over the entirety of the earthworks

Procedure 5.12 creates a raster image of the earthworks needed for dam crest and slopes. The raster defines the absolute altitude for the earthworks in each raster grid point and is in isolation from the study area (i.e. a cell's value being NoData where no earthworks present).

However, the project requirements mean it is necessary to calculate a similar raster containing earthworks heights, i.e. the difference in altitude between the study area terrain and the earthworks. This raster must take into account the 30% soil settlement allowance explained in section 2.2.6. Figure 5.4 shows an illustration of this issue and the differences between earthworks before and after soil settlement.

Definition

Name	get_upstream_fill_point, get_downstream_fill_point
Inputs	dam_crest as geometry, study_area as raster
Outputs	point as geometry

Procedure

1. Take midpoint of dam_crest.
2. Create line, perpendicular to dam_crest and through its midpoint.
3. Get altitude of point with 0.6 interpolation along this line.
4. Get altitude of point with 0.4 interpolation along this line.
5. The upstream point is that point with the higher altitude.

Notes

1. This procedure may fail in cases where there is not a uniform slope under the proposed dam (see illustration in figure 5.3). In this case it will be evident to the user that the procedure has failed because the lake will not form correctly.
2. This procedure will fail in cases where the dam crest does not trace a concave path through the elevation model. This behaviour is undefined, and it will be obvious to the user that the operation has not completed successfully (the lake will not form correctly).

Procedure 5.6: Locating upstream fill point

5.2 System Architecture

The procedures described in the previous section are implemented in PostGIS as stored procedures in the PL/pgSQL language. This means they can be independently executed by any GIS capable of interfacing with PostGIS.

However, a system requirement is to allow a user to access the system via a web interface. To achieve this, a stack of layers is implemented on top of the PostGIS procedures. The system architecture is outlined in figure 5.5.

A 'thin' server layer is implemented in Node.js (see section 3.2) using the Express web application framework (ExpressJS 2014), which plugs into Node.JS. This layer receives API requests from the client and executes the appropriate SQL query to execute the PostGIS stored procedure. This layer also directs map tile requests to the Mapnik rendering engine (via tilecache and tilelive).

The web client application is implemented in HTML5 & JavaScript and is tasked with taking the user interaction (clicks, etc) and constructing the appropriate API request to send to the server, as well as displaying the results of any request.

Definition

Name	create_lake
Inputs	dam_crest as geometry, study_area as raster
Outputs	lake as geometry

Procedure

1. Determine upstream fill point (as in procedure 5.6)
2. Determine altitude of lake = altitude of study_area at first point on dam_crest - 1m freeboard.
3. Create composite raster of study_area and earthworks.
4. Create an empty 2-bit raster, lake_raster, with same dimensions, scale and bounds as the composite raster. Add a raster band with all values = 0.
5. Starting with the upstream fill point, 'fill' the lake using the recursive function:
6. (a) If point is in lake, return.
   (b) If altitude of study area at point is greater than altitude of lake, return.
   (c) Update lake_raster, setting this point = 1.
   (d) Step into this recursive function with point one grid unit to north.
   (e) Step into this recursive function with point one grid unit to south.
   (f) Step into this recursive function with point one grid unit to east.
   (g) Step into this recursive function with point one grid unit to west.
7. Create the lake vector geometry from lake_raster where cell values = 1.

Notes

1. There is an assumption that there is no 'bump' under the dam (see figure 5.3).
2. To prevent out-of-memory errors in PostGIS, it is assumed that the lake fits inside a certain buffer
3. It is obvious to user when algorithm doesn't work.

Procedure 5.7: Creating reservoir geometry

The API uses a mixture of JSON and GeoJSON (see section 3.2.4) and its methods are listed in appendix B.

The client application was developed to a set of UI Mockups, listed in appendix C. The mockups were developed first to ensure that the user of the system was able to perform all actions to satisfy the requirements of the system. The actual implementation of the web application was based on the UI mockups.

Definition

Name	get_lake_area
Inputs	lake as geometry
Outputs	lake area as double precision (in m^2)

Procedure

1. Reproject lake geometry to a spatial reference system with metres as map units. In PostGIS, casting to a geography type has this effect.
2. Use PostGIS's ST_Area() function to calculate area.

Procedure 5.8: Calculating reservoir area

Definition

Name	get_lake_volume
Inputs	lake as geometry, dam_crest as geometry, study_area as raster
Outputs	volume as double precision (in m^3)

Procedure

1. Calculate raster cell width in metres.
2. Calculate raster cell height in metres.
3. Create a lake raster from the lake geometry with lake altitude = altitude of study area at first point on dam crest 1m (the freeboard—see section 2.2.4
4. Generate a series for each non-null cell value in the lake raster.
5. Use SUM() to get total height of water in the lake.
6. Multiply this value by the cell width and height to give maximum water volume.

Procedure 5.9: Calculating reservoir volume

Definition

Name	get_dam_volume
Inputs	dam_height_raster as raster
Outputs	volume as double precision (in m^3)

Procedure

1. Calculate raster cell width in metres.
2. Calculate raster cell height in metres.
3. Generate a series for each non-null cell value in the input raster.
4. Use PostGIS SUM() to get total height of earthworks.
5. Multiply this value by the cell width and height to give total earthworks volume.

Notes

1. Calculating cell width and height in metres requires reprojection of the raster to a spatial reference system whose map units are metres. In PostGIS, casting a geometry type object to geography has this effect. Therefore to calculate cell size, two neighbouring cells are cast to a geography type and then the distance formula is used. This distance is then compared with the source raster's meta data for accuracy.

Procedure 5.10: Calculate earthworks volume

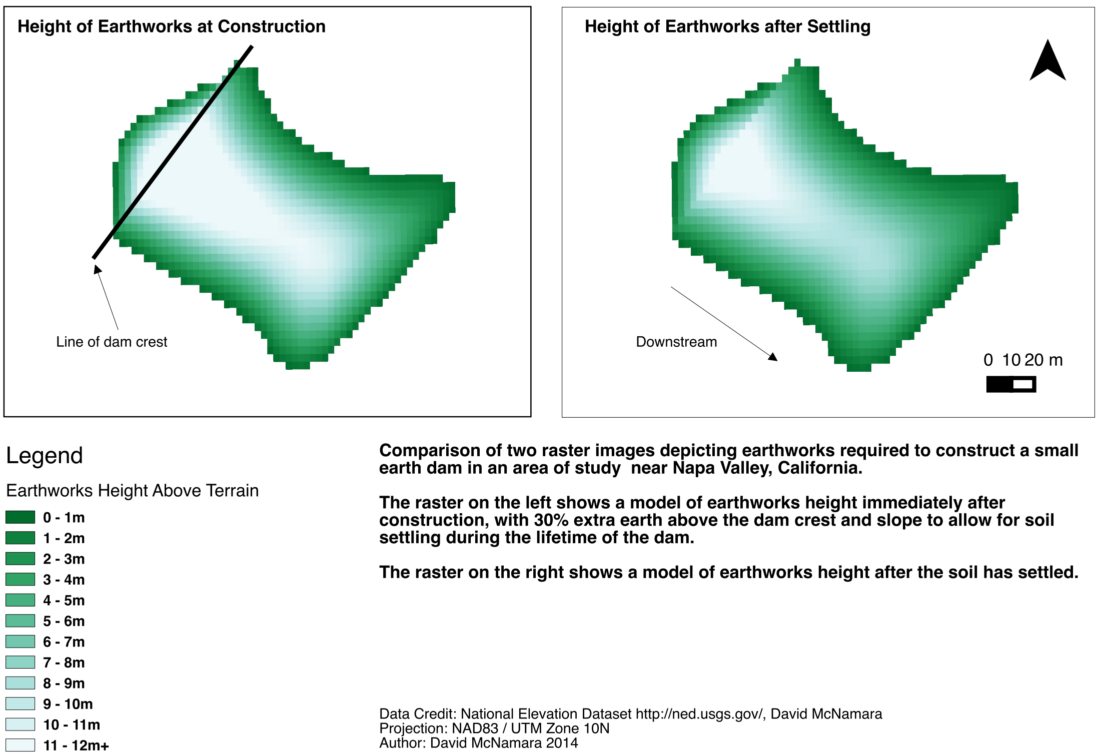


Figure 5.4: Comparison of dam height rasters

Definition

Name	create_dam_height_raster
Inputs	dam_raster as raster, study_area as raster
Outputs	dam_height as raster

Procedure

1. Perform map algebra on dam_raster + study_area
2. For each cell position, calculate height = altitude for cell position in dam_raster - altitude for cell position in study_area
3. Multiply height * 1.3
4. Return output raster

Notes

1. The height of each cell position is multiplied by 1.3 to take into account the factor of 30% soil settlement after construction (see section 2.2.6).

Procedure 5.11: Creating an earthworks height raster

Definition

Name	dam_as_png
Inputs	dam_height raster from 5.11
Outputs	thematic earthworks height image in .png format, image bounds in world coordinations

Procedure

1. Resize dam_height raster to a standard size, say 500x500 pixels.
2. Create a colour map to translate altitude to a RGB colour value.
3. Use PostGIS ST_AsPNG function to create a .png image with the colour mapped dam height raster, translating values to their nearest colour.
4. Encode the .png image as base64 (ASCII) to allow for transmission over the JSON API.
5. Also return the bounds of the dam_height raster in the response. This will allow the front-end to correctly place the .png image.

Procedure 5.12: Creating the earthworks risk thematic

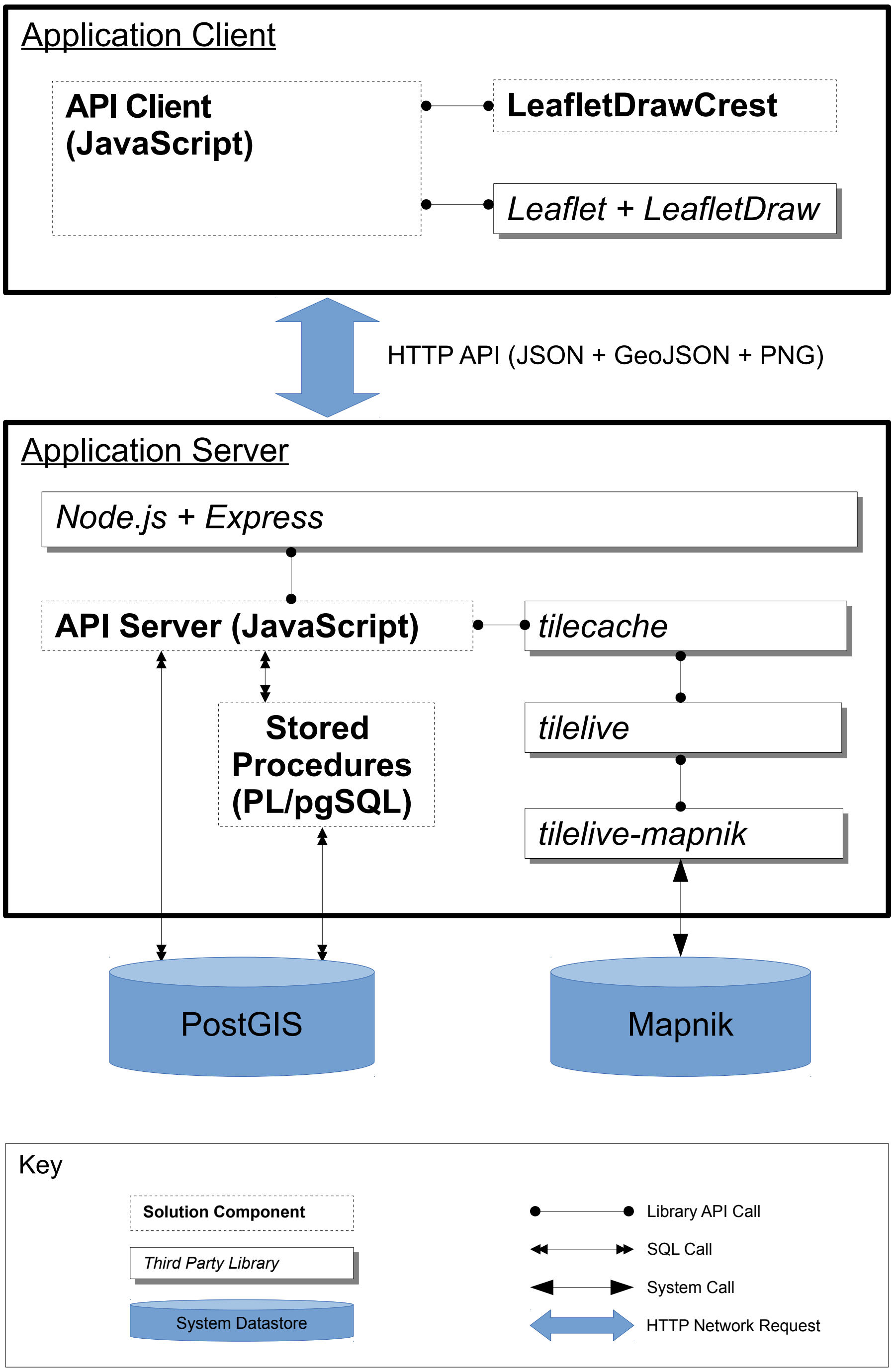


Figure 5.5: Solution architecture

Chapter 6

Results

6.1 Fulfillment of Requirements

A critical measure of the success of this work is the completion of a solution that fulfills the requirements established in chapter 2. Indeed, the tool satisfies those requirements and was actually used to create the models and to execute the spatial procedures for testing; the results of which are presented in this chapter.

The source code of the solution is available for researchers to deploy and further develop, or simply to use over the web as a decision support tool. Two screenshots of the tool are given in figures 6.1 and 6.2. These show examples of the tool use to define a dam crest and to inspect dam and reservoir statistics.

Additionally, it is intended that the spatial analysis procedures can be used outside the web frontend. QGIS was used to prepare the thematic map infigure 6.3, showing the results of the analysis for a single study dam in the Napa Valley study area.

From these outputs, it is obvious that the user of the system can successfully model landscapes of small earth dams and view results to aid in the decision making process.

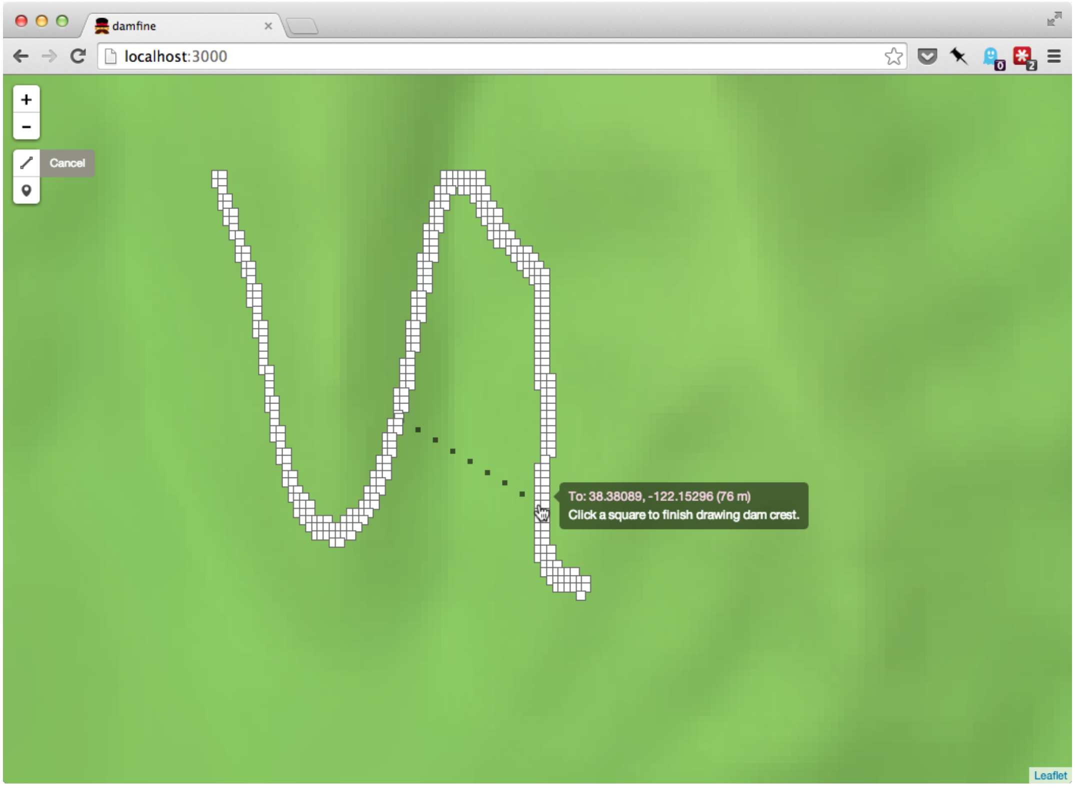


Figure 6.1: Screenshot of tool showing user being prompted to finish drawing dam crest

6.2 Data Analysis

6.2.1 Reservoir Volume

Procedure 5.9 details the steps necessary to calculate the volume of the new reservoir after a dam model has been defined. To analysis the accuracy of this calculation, it can be compared against the formula given by Stephens (2010) (see section 2.2.4).

Table 6.1 shows the reservoir volumes for a sample of 10 dams created by the system in the Napa Valley study area. This table compares the results of the reservoir volume calculation in this work with Stephens' method from section 2.2.4. An illustration of the figures is shown in figure 6.4. Table 6.2 and figure 6.5 shows the results of the same comparison for the Buraydah study area.

Table 6.3 compares reservoir volume calculations for dams in both Napa and Buraydah study areas against Sawunyama's volume estimate formula from section 3.1.2.

Compared with Stephens' formula, error rates of 17% and 5% imply that the

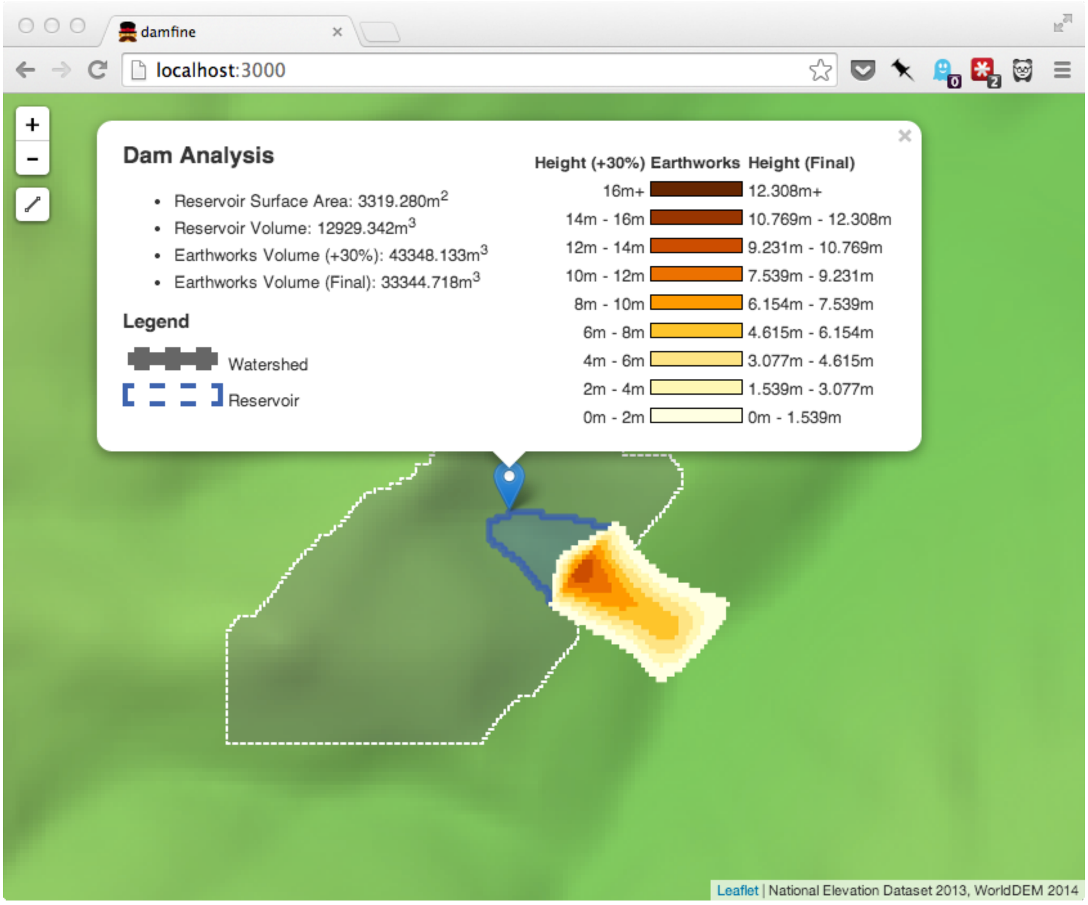


Figure 6.2: Screenshot of tool showing final dam analysis

GIS methodology described here is reasonably accurate. However, there is little correlation in results when compared to Sawunyama's methodology. One would expect close results when compared with one methodology would favour similarly accurate results with another. However this does not seem to be the case and could be due to a number of variables. For instance, Sawunyama's research was limited to a small study area in one region, whereas Stephens has attempted to formulate a more general case. This could explain some of the discrepancy in testing and is a good subject for further investigation.

6.2.2 Earthworks

Procedure 5.10 describes a process to estimate earthworks volumes, i.e. the amount of earth that must be moved to complete works at the site. To evaluate the accuracy of the model and calculation, the results can be compared with Stephens' estimation

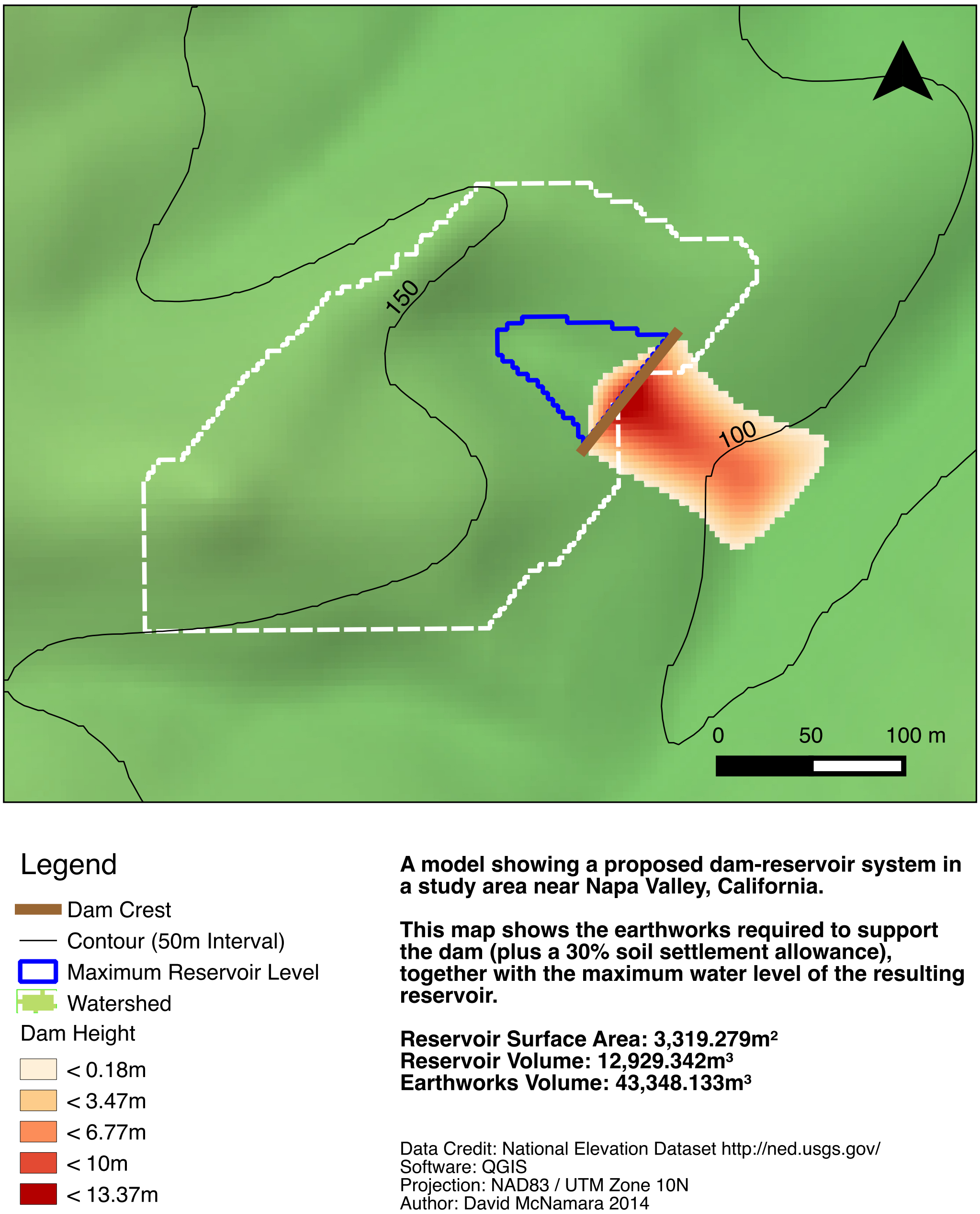


Figure 6.3: Sample analysis prepared in QGIS

Dam	GIS Method (m^3)	Stephens Method (m^3)	Error
1	17,159.63	19,463.25	+13.42%
2	10,862.47	13,422.70	+23.57%
3	8,867.63	10,957.52	+23.57%
4	12,929.34	14,409.10	+11.44%
5	112.56	497.69	+342.14%
6	10,775.44	12,340.36	+14.52%
7	11,299.07	10,218.51	+9.56%
8	2,479.06	5,939.93	+139.60%
9	8,586.46	9,640.73	+12.28%
10	894.78	1,596.44	+78.42%
Total	83,966.46	98,486.22	+17.29%

Table 6.1: Analysis of reservoir volume calculation for 10 sample dams in Napa Valley

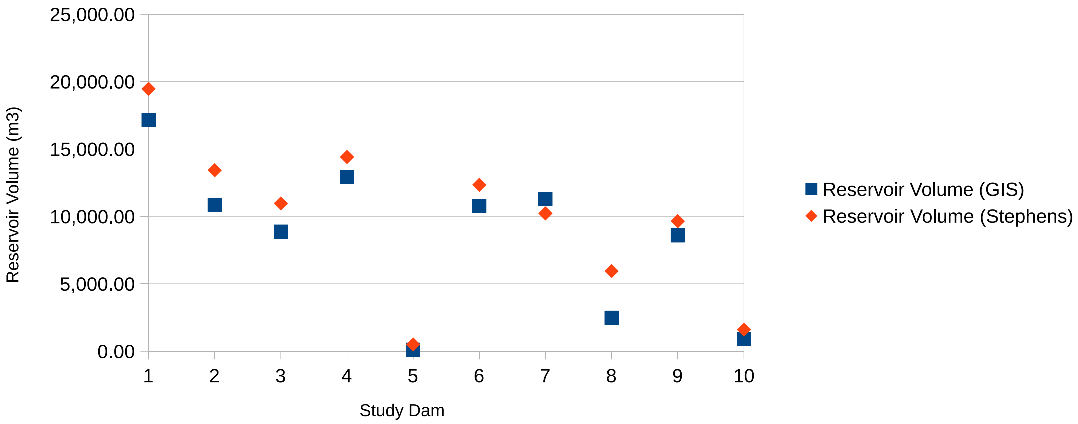


Figure 6.4: Comparison of reservoir volume calculation for 10 sample dams in Napa Valley

formula from section 2.2.5.

Table 6.4 and table 6.5 list the results of the earthworks volume calculation for the Napa Valley and Buraydah study areas. An illustration of those results are in figure 6.7 and figure 6.8 respectively.

A difference in the accuracy of the results of 50% and 33% show a reasonable correlation between Stephens' estimate and the GIS method described here. However, without further testing with field research during dam construction it is difficult to assert any improvement in accuracy over the reference method.

Dam	GIS Method (m^3)	Stephens Method (m^3)	Error
11	497,653.35	449,856.63	+9.60%
12	21,486.01	29,772.83	+38.57%
13	2,490.10	4,412.57	+77.20%
14	49,952.84	28,819.05	-42.31%
15	79,290.22	113,044.80	+42.57%
16	51,240.62	48,076.06	-6.18%
17	66,880.96	73,617.67	+10.07%
18	102,644.93	69,317.11	-32.47%
19	270,517.63	178,118.06	-34.16%
20	106,986.71	190,523.53	+78.08%
Total	1,249,143.39	1,185,558.30	-5.09%

Table 6.2: Analysis of reservoir volume calculation for 10 sample dams in Buraydah

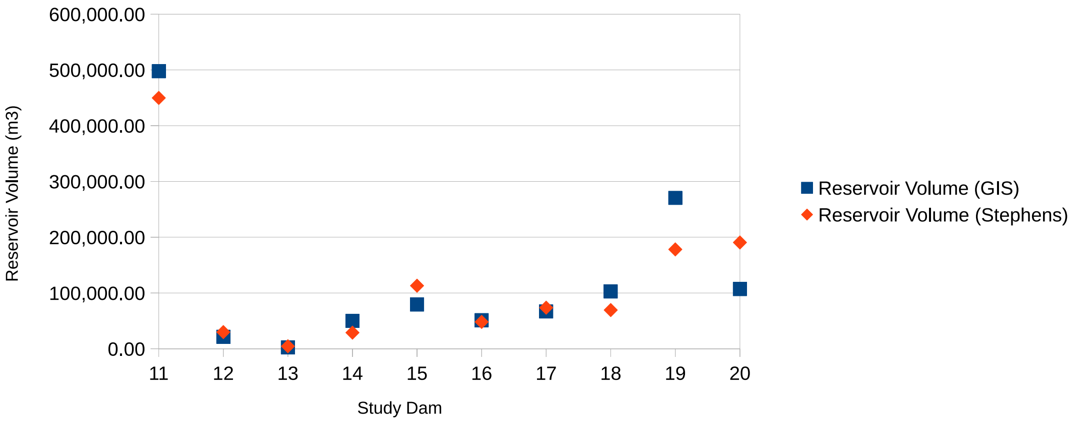


Figure 6.5: Comparison of reservoir volume calculation for 10 sample dams in Buraydah

Dam	Area (m^2)	Sawunyama Method (m^3)	GIS Method (m^3)
1	3,855.54	1,920.44	17,159.63
2	2,967.93	1,341.22	10,862.47
3	2,672.07	1,161.25	8,867.63
4	3,319.28	1,563.74	12,929.34
5	110.90	14.76	112.56
6	2,829.25	1,255.98	10,775.44
7	4,615.54	2,458.14	11,299.07
8	1,174.85	376.10	2,479.06
9	2,024.03	793.27	8,586.46
10	434.37	96.04	894.78
11	47,544.73	60,294.37	497,653.35
12	4,091.17	2,083.28	21,486.01
13	1,092.98	340.62	2,490.10
14	7,237.54	4,556.71	49,952.84
15	30,857.24	33,318.87	79,290.22
16	26,236.70	26,670.77	51,240.62
17	37,289.53	43,202.56	66,880.96
18	31,297.22	33,972.40	102,644.93
19	44,804.41	55,578.18	270,517.63
20	31,557.70	34,360.91	106,986.71
Total	286,012.97	305,359.61	1,333,109.84

Table 6.3: Analysis of reservoir volume calculation: GIS method compared with Sawunyama method

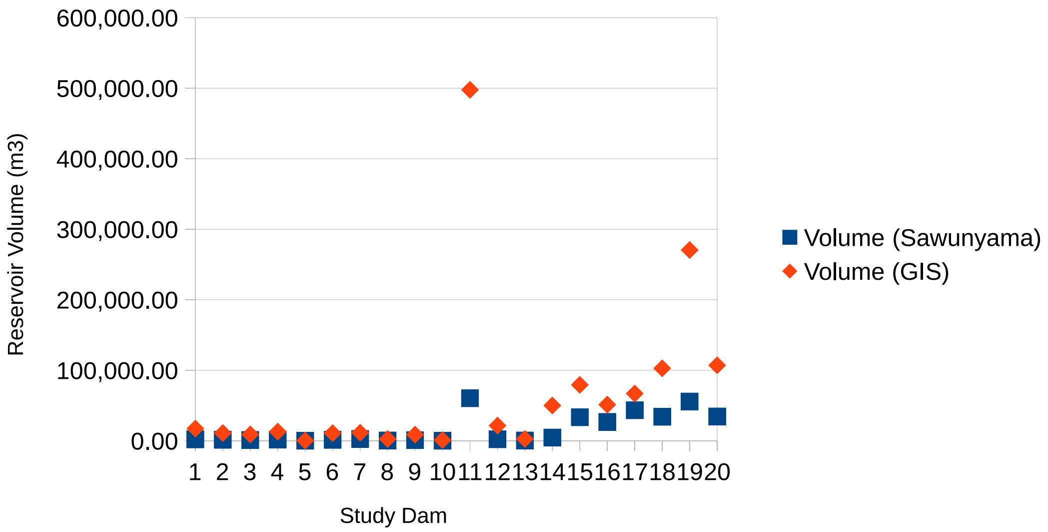


Figure 6.6: Illustration of reservoir calculation: GIS method compared with Sawunyama method

Dam	GIS Method (m^3)	Stephens Method (m^3)	Error
1	47,114.17	20,146.43	-57.24%
2	27,813.99	13,133.14	-52.78%
3	23,287.69	10,008.82	-57.02%
4	33,344.72	14,146.10	-57.58%
5	282.43	688.57	+143.80%
6	18,770.27	12,322.57	-34.35%
7	4,586.86	5,936.48	+29.42%
8	8,981.03	5,500.78	-38.75%
9	22,835.28	13,084.59	-42.70%
10	4,846.48	2,346.28	-51.59%
Total	191,862.92	97,313.76	-49.28%

Table 6.4: Analysis of earthworks calculation for 10 sample dams in Napa Valley

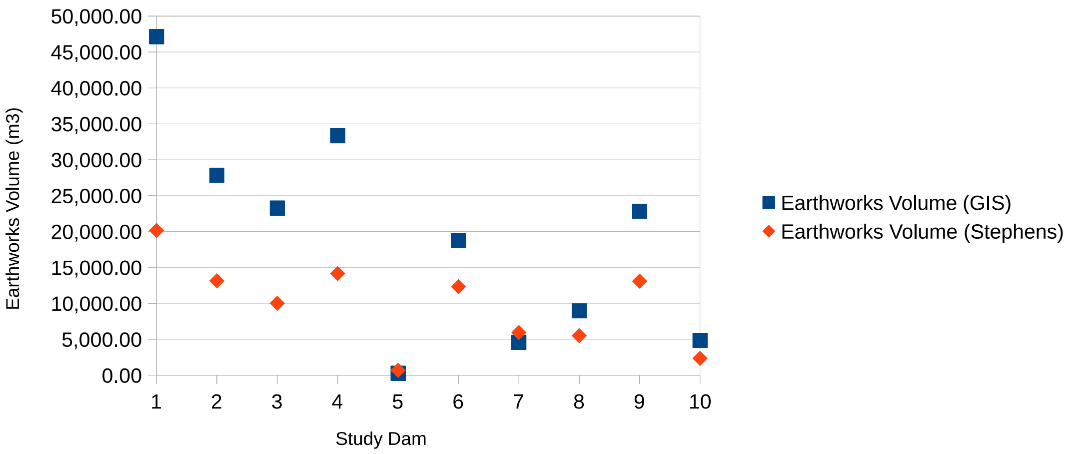


Figure 6.7: Comparison of earthworks calculation for 10 sample dams in Napa Valley

Dam	GIS Method (m^3)	Stephens Method (m^3)	Error
11	122,889.13	171,287.99	+39.38%
12	26,646.68	20,987.35	-21.24%
13	4,215.15	2,891.40	-31.40%
14	24,666.65	16,184.65	-34.39%
15	909.82	14,979.53	+1,546.43%
16	10,866.85	11,685.80	+7.54%
17	7,863.03	10,098.74	+28.43%
18	7,520.67	16,043.70	+113.33%
19	36,856.03	46,824.71	+27.05%
20	14,113.30	30,724.72	+117.70%
Total	256,547.33	341,708.59	+33.20%

Table 6.5: Analysis of earthworks calculation for 10 sample dams in Buraydah

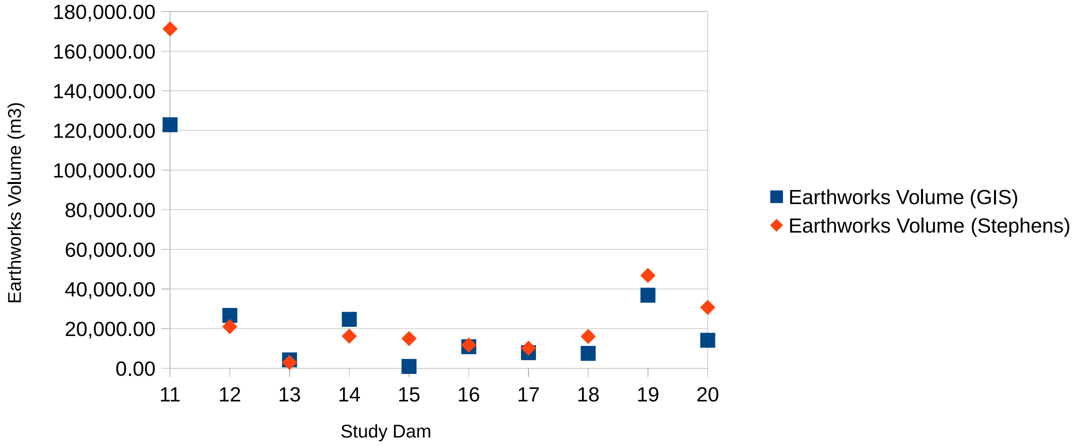


Figure 6.8: Comparison of earthworks calculation for 10 sample dams in Buraydah

Chapter 7

Conclusion

7.1 Summary of Research

This work has aimed to apply a research methodology to the geospatial concerns of creating landscapes of small earth dams as a retention measure. The implementation of the work was performed as a technical handbook to a student or landscape designer that would find value in this system.

At the core of the system are a set of well documented and reasoned spatial analysis procedures. A student wishing to understand the system must familiarize themselves with the PL/pgSQL language (although having a steep learning curve) is an ideal skill in GIS development.

A somewhat tangential objective of the work was to develop the GIS with open source tools and technology. It was felt that doing it in such a manner would encourage adaptation and improvement of the software. The source code has been published on the web (see Appendix D) under a MIT license (MIT 1998)

Finally, the work has resulted in a useful tool, freely available to use and share. The final user interface certainly meets the objectives of the research, and was used to produce the data analysed in the previous chapter. There is certainly more work to do to produce a friendly tool that non-programmers can use with ease, and it is hoped that there will be interest in continuing this work the future.

7.2 Critical Appraisal

The results of the spatial analysis procedures were compared with calculations from the literature. With a small 5% error rate in one study area, it can be said that the data model and spatial analysis look sound. However the earthworks volume calculation had an error rate of 49%, which suggests that there could be further investigation there. Perhaps the most logical continuation of this research would be to more accurately corroborate the results with measurements during a dam construction and of the resultant reservoir.

On one hand, it can be considered that this research simply reformulates existing calculations albeit using GIS methodology. That may be true in one respect, but documenting and validating spatial analysis adds value to that research. Packaging it as a educational tool add value.

Some failings of the project were due to time constraints and could be considered out of scope of the research in any case. For example, allowing for curved dams would more accurately model earthworks, but the complexity would be exponential compared to straight, single-segment dams.

A considerable amount of time was invested in learning the PL/pgSQL language to program the spatial analysis procedures. While the documentation is generally very good, the extent of community knowledge—so prevalent in many open source projects—seemed small. A student or GIS developer wishing to continue this work would need to familiarise themselves with this technology, which may be difficult depending their on background and skill-set.

In some cases, considerable effort was put into developing procedures that didn't produce good results. As failures they are still valuable research material and so are listed in the source code in Appendix D. The watershed analysis, for instance, is perhaps a naive implementation and would benefit from further iteration. Indeed that may make for a research project in its own respect.

In hindsight, the scope of developing a tool as well as the methodology was perhaps too large. The results could have been produced with less effort using QGIS, ArcGIS or r (2014). However, there is value and satisfaction in learning and documenting the developing process in such an endeavor. It was an objective of the research to produce a web GIS, and that was met. Moreover, the procedures can be 'decoupled' from the GIS and used directly in PostGIS or via QGIS (the PostGIS functions and schema are included in the source code).

7.3 Further Work

It would have been interesting to examine further the use of high-precision elevation data especially from UAV mapping (see section 4.1), as indications are that this will be a high-growth area for amateur surveyors, perhaps particularly in ASAL regions such as this research is targeting. Some effort was made to look into sample data from DroneMapper (2014) but the data was unsuitable by the tool in its current form. This could indicate that an issue in the software, memory issues or some unknown problem. Perhaps if the decision was taken to use another GIS software—open source or otherwise—then processing high-precision data would not have been an issue.

Further investigation into the results of the spatial analysis procedures is a topic for future research. It would be worthwhile examining the discrepancy in results compared to Stephens and Sawunyama's research. Statistical analysis such as regression testing would be beneficial in determining which factor affected the results. With this knowledge, the accuracy of procedural model could be improved.

With more time, a study of the design and construction of new small earth dams in ASAL regions could be undertaken and the results analysed for accuracy. However to do so on a fair scale, this would take several years. Additionally, a more advanced model could be developed. As Ulbig (2014) points out, dams should always be accompanied with swales, terraces and vegetation cover.

7.4 Benefits to Society and Environment

It has been evident, through this and other works, that benefit to community and environment comes with water security and it is hoped that this work will stand some part in advancing the hydrological and social benefit of understanding these processes. The discipline of GIS, and open source GIS in particular, has an important role to play in creating local knowledge of earthworks for water retention. By working with engineers and other professionals, it can be demonstrated that GIS-based estimates of earthworks and storage capacity and other issues can assist in the decision making process when planning a small earth dam, especially in arid and semi-arid lands.

Bibliography

Airbus (2014) *WorldDEM* [Online] URL: http://www.astrium-geo.com/worlddem/ (accessed December 2013).

Anthes, G. (2012) HTML5 leads a web revolution. *Communications of the ACM* 55, 16. doi:10.1145/2209249.2209256

ArcGIS (2014) *How Watershed Works* [Online] URL: http://resources.arcgis.com/en/help/main/10.1/index.html#/How_Watershed_works/009z00000068000000/ (accessed March 2014).

Barton, K., Brady, C. (2012) *An International Surface Collection and Remote Sensing Field School on the Hill of Slane, County Meath, Ireland. Presented at the Aerial archaeology, remote sensing and the archaeological process.* Aerial Archaeology Research Group, Eötvös Loránd University, Budapest, Hungary.

Brovelli, M.A., Magni, D. (2003) An archaeological Web GIS application based on Mapserver and PostGIS. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences* 34, 8994.

Carmichael, P., Honour, L. (2002) Open source as appropriate technology for global education. *International Journal of Educational Development* 22, 4753.

Chen, D., Shams, S., Carmona-Moreno, C., Leone, A. (2010) Assessment of open source GIS software for water resources management in developing countries. *Journal of Hydro-environment Research* 4, 253264. doi:10.1016/j.jher.2010.04.017

Daum, S. (2014) *Water quality and sustainability of the water retention landscape at Tamera ecovillage, south Portugal.* Tübingen.

Dragićević, S. (2004) The potential of Web-based GIS. *Journal Geographic Systems* 6, 7981. doi:10.1007/s10109-004-0133-4

dronemapper.com (2014) *DroneMapper Aerial Imagery Processing and GIS Services* [Online] DroneMapper Aerial Imagery Processing and GIS Services. URL: http://www.dronemapper.com/ (accessed December 2013).

Ehlschlaeger, C. (1989) Using the AT Search Algorithm to Develop Hydrologic Models from Digital Elevation Data. *Proceedings of International Geographic Information Systems (IGIS) Symposium* 89 275281.

esri (2011) *Open Source Technology and Esri* [Online] URL: http://www.esri.com/news/arcnews/spring11articles/open-source-technology-and-esri.html (accessed January 2014).

Gesch, D., Oimoen, M., Greenlee, S., Nelson, C., Steuck, M., Tyler, D. (2002) The national elevation dataset. *Photogrammetric engineering and remote sensing* 68, 532.

Global Forest Watch (2002) *Global Forest Watch.* World Resources Institute, Washington, DC Available [Online] URL: http://www.globalforestwatch.org (accessed March 2014).

Holzer, S. (2012) *Desert or Paradise.* Chelsea Green, Vermont.

Imhof, E. (2007) *Cartographic Relief Presentation.* Esri Press, Redlands, Calif.: S.l.

JSON (2014) JSON: *The Fat-Free Alternative to XML* [Online] JSON: The Fat-Free Alternative to XML. URL: http://www.json.org/xml.html (accessed January 2014).

Keenan, P.B. (2003) Spatial decision support systems. Decision making support systems: Achievements and challenges for the new decade 2839.

Kraus, K., Pfeifer, N. (1998) Determination of terrain models in wooded areas with airborne laser scanner data. *ISPRS Journal of Photogrammetry and remote Sensing* 53, 193203.

Kravčík, M. (2007) *Water for the recovery of the climate: A new water paradigm.* TypoPress.

Leaflet (2014) *Leaflet Reference* [Online] URL: http://leafletjs.com/reference.html (accessed January 2014).

Leaflet.draw (2014) *Leaflet.draw* [Online]
URL: https://github.com/Leaflet/Leaflet.draw (accessed January 2014).

Li, M. (2013) Architecture of WebGIS based on Web2. 0, in: *Conference Anthology, IEEE*. IEEE, pp. 14.

McKee, L. (2014) *OGC History (detailed)* [Online] OGC History (detailed). URL: http://www.opengeospatial.org/ogc/historylong (accessed February 2014).

McMurray, D. (2003) Assessment of water use from farm dams in the Mount Lofty Ranges, South Australia. *Department of Water, Land and Biodiversity Conservation.*

Miler, M., Odobašić, D., Medak, D. (2010) *Efficient Web-GIS Solution based on Open Source Technologies: Case-Study of Urban Planning and Management of the City of Zagreb, Croatia* [Online] URL:
http://bib.irb.hr/prikazi-rad?lang=en&rad=482403 (accessed March 2014).

MIT (1998) The MIT License (MIT). URL: http://opensource.org/licenses/MIT

Moore, A. (2002) *Open-Source Learning.* Educause.

Nelson, K.D. (1985) *Design and construction of small earth dams.* 127 pp.

Neteler, M., Mitasova, H. (2008) *Open source GIS a GRASS GIS Approach.* Springer, New York, NY.

Nissen-Petersen, E. (2006) *Water from Small Dams.* ASAL Consultants Ltd. for the Danish International Development Assistance (Danida), Kenya.

Node.js (2014) *Node.js* [Online] Node.js. URL: http://nodejs.org/ (accessed December 2013).

Okolloh, O. (2009) *Ushahidi, or testimony: Web 2.0 tools for crowdsourcing crisis information. Participatory learning and action* 59, 6570.

Open Geospatial Consortium, 2006a. *OGC Implementing/Compliant Product Details: PostGIS / PostgreSQL Product Details* [Online] URL: http://www.opengeospatial.org/resource/products/details/?pid=509 (accessed April 2014).

Open Geospatial Consortium, 2006b. *OpenGIS Web Map Service version 1.3.0.* Open Geospatial Consortium (OGC).

Open Source Geospatial Consortium (2014) GDAL.

OpenLayers (2014) *OpenLayers Documentation* [Online] OpenLayers Documentation. URL: http://trac.osgeo.org/openlayers/wiki/Documentation (accessed January 2014).

Pavlenko, A. (2013) *mapnik* [Online] URL: http://mapnik.org/ (accessed January 2014).

Peng, Z.-R., Zhang, C. (2004) The roles of geography markup language (GML), scalable vector graphics (SVG), and Web feature service (WFS) specifications in the development of Internet geographic information systems (GIS). *Journal of Geographical Systems* 6. doi:10.1007/s10109-004-0129-0

PostGIS (2014) *PostGIS Spatial and Geographic objects for PostgreSQL* [Online] URL: http://postgis.net/ (accessed January 2014).

QGIS (2014) QGIS Documentation [Online] QGIS Documentation. URL: http://www.qgis.org/en/docs/index.html (accessed January 2014).

r (2014) *The R Project for Statistical Computing* [Online] URL: http://www.r-project.org/index.html (accessed April 2014).

Sandvik, B. (2012) *thematic mapping blog: Creating color relief and slope shading with gdaldem.* [WWW Document] URL: http://blog.thematicmapping.org/2012/06/creating-color-relief-and-slope-shading.html (accessed February 2014).

Sawunyama, T., Senzanje, A., Mhizha, A. (2006) Estimation of small reservoir storage capacities in Limpopo River Basin using geographical information systems (GIS) and remotely sensed surface areas: Case of Mzingwane catchment. *Physics and Chemistry of the Earth*, Parts A/B/C 31, 935943. doi:10.1016/j.pce.2006.08.008

Schütze, E., Vatterrott, I.H.-R. (2007) *Current state of technology and potential of Smart Map Browsing in web browsers.* Thesis, University of applied sciences Bremen

Simeoni, L., Zatelli, P., Floretta, C. (2013) Field measurements in river embankments: validation and management with spatial database and webGIS. *Natural Hazards* 71, 14531473. doi:10.1007/s11069-013-0955-9

Stephens, T. (2010) *Manual on small earth dams.* Food and Agriculture Organization of the United Nations, Rome.

Tsimpiris, I. (2014) *New QGIS plugin for water management* [Online] URL: http://www.gisblog.com/new-qgis-plugin-for-water-management/ (accessed February 2014).

Ulbig, C. (2014) Interview with Christoph Ulbig of Tamera Healing Biotope in April 2014.

USBR (1987) *Design of small dams.* US Department of the Interior, Bureau of Reclamation.

Wang, C.L., Ferris, N.M. (2010) *A GIS Based Approach in Assessing Embankment Dams.*

Wilcox, M. (2009) *Porting to the Symbian Platform: Open Mobile Development in C/C++*, 1 edition. ed. Wiley, Chichester, U.K.

Appendix A

Interview Transcript

This interview was conducted with Christoph Ulbig of Tamera Healing Biotope in April 2014 over email.:

Interviewer: In what ways can the building of small earth dams help to reverse desertification in arid and semi-arid regions of the world?

Ulbig: Strong and rare rain events can be buffered for infiltration and aquifer recharge. Water can be directly used for bordering vegetation and irrigation and domestic/animal use, if higher in altitude without pumping (gravity). No need for (deep) boreholes. Remember dams should always be combined with terraces, swales, and vegetation cover. Small earthdams can be made easily from handwork to small local available machinery and do not need external planning.

Interviewer: How do you imagine GIS (Geographic Information Systems) or other technologies can assist in this industry/practice?

Ulbig: Planning of watershed management and regional planning. If data is available: geology soil type, infiltration, runoff rate, GIS can help to scale the lakes with regard to the catchment size and the landuse. This can avoid flood events and help to reach more infiltration and aquifer recharge. GIS can map the best suited locations in a district.

Interviewer: As an engineer/designer of water retention landscapes, what do you consider to be the greatest risk to the stability/safety of the earthworks during and after construction?

Ulbig: In small and not steep earthdams the risks are generally low. Risk occur if wrong material is used, or good material is not available.

Interviewer: Do you think students of this discipline would find benefit in a tool that would allow them to model water retention landscapes on a computer?

Ulbig: Yes if it is combined with on-site experience and contact to the land. As an education tool it is very valuable.

Appendix B

API

The following is a list of API methods exposed by the server. URL parameters prefixed by ':' are replaced with the appropriate value by the client.

`/:study_area/points_at_level_near/:lat/:lng/`

Returns list of points of similar latitude near :lat, :lng

`/:study_area/save_dam_crest/:startlat/:startlng/:endlat/:endlng`

Creates a new dam. Returns the id of the dam.

`/:study_area/create_lake/:dam_id`

Creates a reservoir for the dam. Returns the reservoir geometry as GeoJSON.

`/:study_area/get_lake_area/:dam_id`

Returns the reservoir area in m^2

`/:study_area/create_dam/:dam_id`

Creates a model of the earthworks for a dam. Returns an earthworks height raster as png, and the bounds of the raster image.

`/:study_area/get_dam_meta/:dam_id`

Returns meta statistics for the dam/reservoir: dam volume, reservoir area and volume.

`/:study_area/create_watershed/:dam_id`

Estimates the watershed for the dam. Returns geometry of the watershed.

`/:study_area/:z/:x/:y`

Returns the tile map image for the specified coordinates and zoom.

Appendix C

UI Mockups

Creating mockups of the user interface was a valuable part of the process to determine if the UI would allow the system to meet all the requirements (see section 5.2).

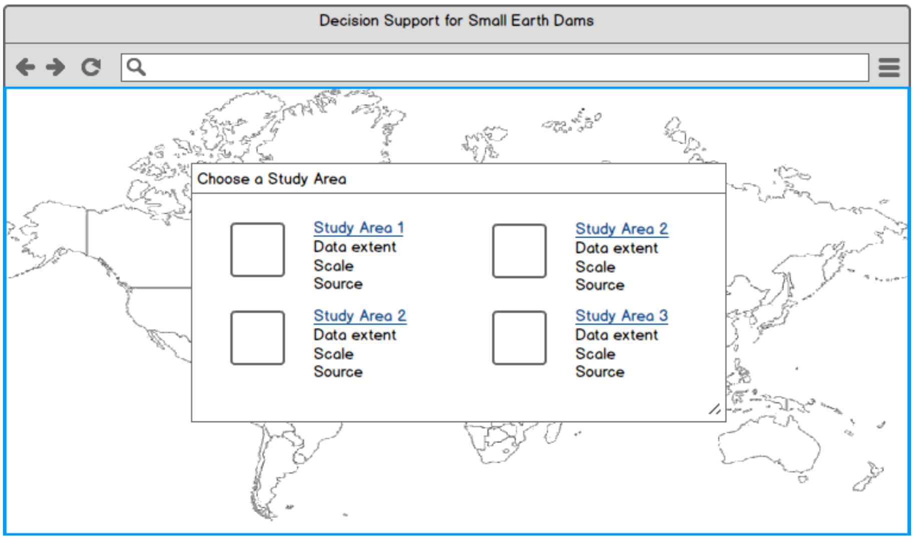


Figure C.1: UI Mockup: Choose Study Area

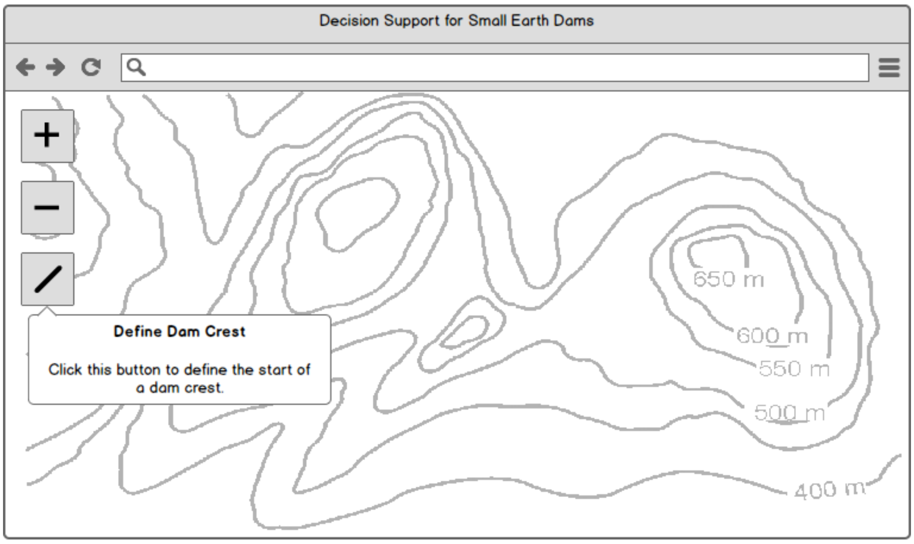


Figure C.2: UI Mockup: View Study Area

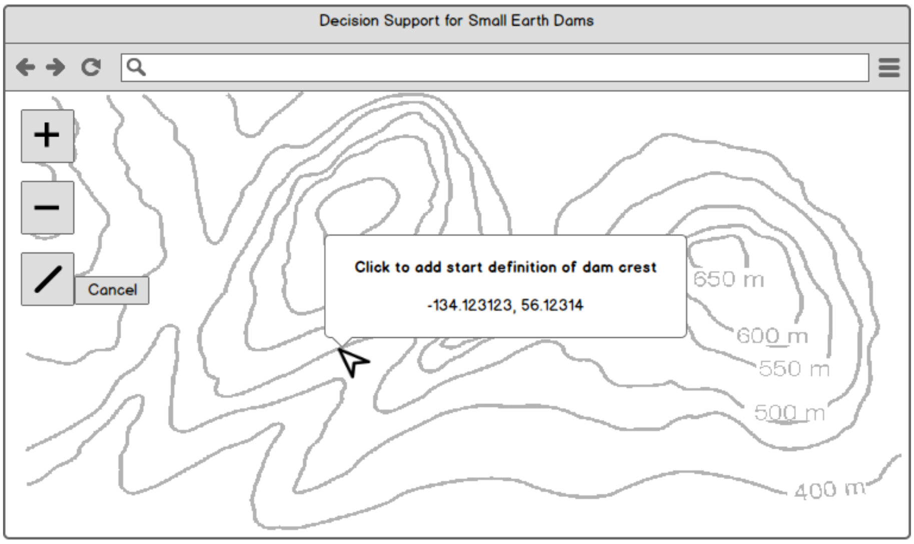


Figure C.3: UI Mockup: Click to Start Definition Dam Crest

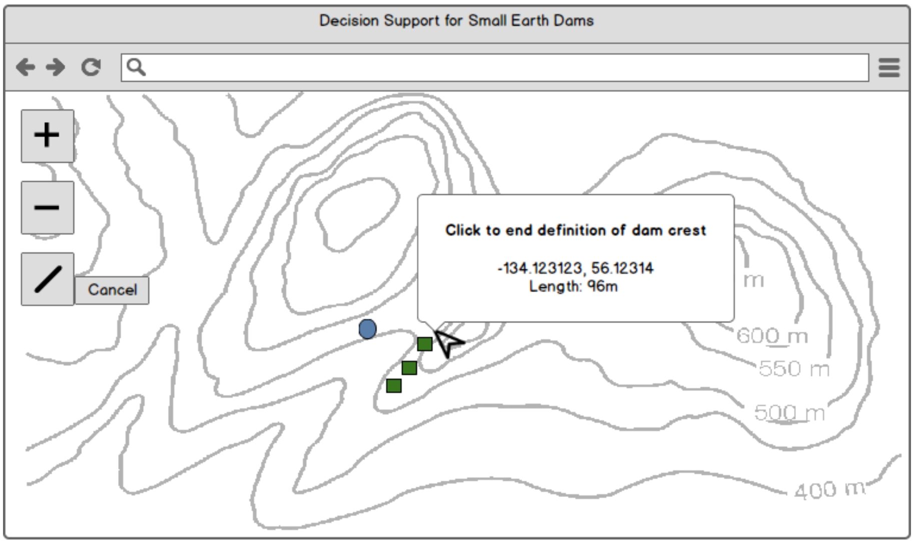


Figure C.4: UI Mockup: Click to End Definition Dam Crest

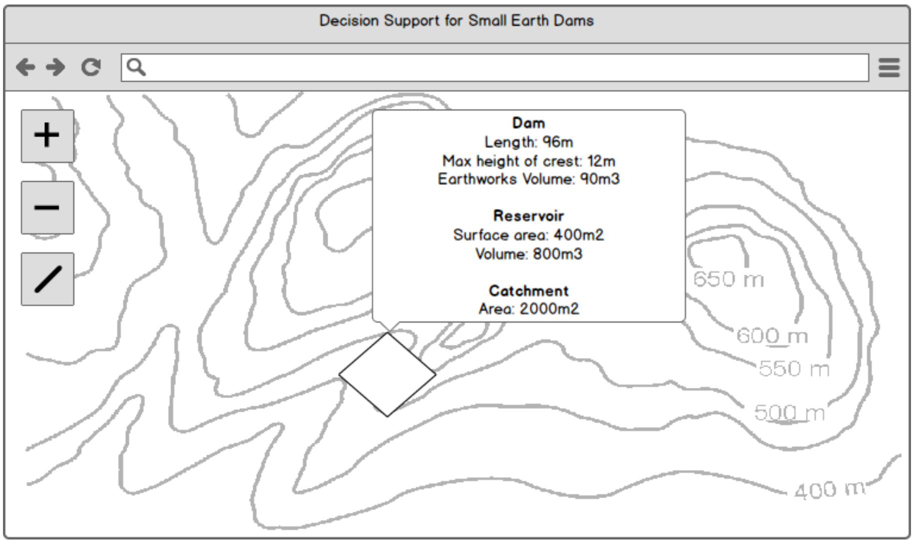


Figure C.5: UI Mockup: Dam Stats

Appendix D

Source Code

The source code is available at:

```
http://github.com/mackers/simdam
```