International Journal of Advanced Multidisciplinary Research (IJAMR)
ISSN: 2393-8870
www.ijarm.com

Research Articles
Thematic Mapping Engine using GIS Technology

D.Kavitha[1*], S. Aruchamy[2], S. P. Karthikeyan[3] and K. Kumaraswamy[4]

[1]Department of Geography, Bharathidasan University, Tiruchirappalli, Tamil nadu, India
[2]Professor, Department of Geography, Bharathidasan University, Tiruchirappalli, Tamil nadu, India
[3]Scientist-D, Open Technology Centre, National Informatics Centre, Chennai, Tamil nadu, India
[4]Professor and Head, Department of Geography, Bharathidasan University, Tiruchirappalli, Tamil nadu, India
*Corresponding author:*dkavithas39@gmail.com*

Keywords

Thematic for cropping pattern,
WebGIS,
Cartographic toolbox,
GIS technology

Abstract

Development of dynamic thematic for cropping pattern using WebGIS is the energetic or force in actual operation. Dynamic cropping pattern to create a crop production environment that is changing. Such mapping environments are now easier to create with recent advances in web technologies and standards. The user can, to a large extent, determine what information is to be displayed and in what context. This project aimed to provide data, visualization tools and a cartographic toolbox to the user in a web-based interface. the core subject of the study in which The Thematic Mapping Engine was developed to demonstrate how these techniques could be utilised in a web application. A key component to this approach is the utilization of GIS technology. GIS provides the visual integration of all the data sources tied to crops and allows users to identify the cropping pattern information. This approach uses GIS to incorporate spatial information such as Cropping Concentration and Crop Combination Regions. The addition of spatial information to process allows the user to consider within- field they navigate different layers.

Introduction

The aim of this research is to determine whether Keyhole Markup Language (KML) can be used for thematic mapping. As a proof-of-concept, the work culminates in the implementation of a fully functioning open source application, the Thematic Mapping Engine. The term neogeography is commonly applied to the set of technologies and techniques presented in this document. Neogeography combines the complex techniques of cartography and GIS and places them within the reach of users and developers (Turner, 2006; Walsh, 2008; Davis, 2007).

"Every now and again a web based service comes along that takes our breath away, Bjørn Sandvik's Thematic Mapping Engine is one of those services. His Thematic Mapping Engine enables you to visualise global statistics on Google Earth in a way that only a few years ago would have been a showstopper using high end tools such as ESRI's ArcGlobe."

Dr Andrew Hudson-Smith, Digital Urban 1 July 2008.The Thematic Mapping Engine was developed using open source software, and it is released as an open source project. The goal was to develop a low-cost solution suitable for non-profits and public benefit organisations. The application also demonstrates what it is possible to achieve using open source tools, open standards and datasets in the public domain.

Cartography and GIS have both emerged as major capabilities on the web. Distributed maps are different from traditional static maps in that they link information from various sources and provide a user-defined environment (Crampton, 2001).Such mapping environments are now easier to create with recent advances in web technologies and standards. The user can, to a large extent, determine what information is to be displayed and in what context. This project aimed to provide data, visualisation tools and a cartographic toolbox to the user in a web-based interface.

A step-wise approach was chosen in order to meet this aim:

Data preparation

The first step involved finding appropriate statistical and spatial data that could be combined and used for thematic mapping. The data had to be gathered from various sources and stored in a database.

Thematic mapping with KML

A scripting language (PHP) was used to query the database and transform the result into KML documents representing various thematic mapping techniques.

Thematic Mapping Engine

The Thematic Mapping Engine was developed to demonstrate how these techniques could be utilised in a web application.

Evaluation

The various techniques were evaluated after gaining feedback from people using the Thematic Mapping Engine.

The methods of thematic mapping are well described in the cartographic literature,but it was hard to find books and journal articles describing the use of KML and geobrowsers for this purpose. There is a lot of development going on, but little has so far been described in the academic literature. The focus was therefore shifted towards the “blogosphere”3. Blogs are now widely used among “geeks” and professionals alike, to present their own work and perspectives and to comment on other people’s work. Bloggers actively review and comment on the latest trends and developments, and this turned out to be a valuable information source for this project.

The Thematic Mapping Engine

The Research Paper explains how Keyhole Markup Language (KML) and geobrowsers can be used for thematic mapping (Sandvik, 2008). The experiments show that KML has a great potential for thematic mapping, even though the techniques are complicated to use for non-programmers. By embedding the techniques in a Thematic Mapping Engine, it was possible to hide this complexity. TME allows the user to create thematic maps through an easy to use web interface, or by writing a few lines of code.

Requirements

The Thematic Mapping Engine requires the following software (all are open sourceand available free of charge)

Software Requirements

PHP (Version 5)

PHP is a computer scripting language originally designed for producing dynamic web pages PHP, a server technology that allows for rapid application development of interactive websites, is the chief application logic layer in the open source configuration.

Apache HTTP Server

The Apache HTTP Server is an open source web server, developed and maintained by the Apache Software Foundation.

MySQL (Version 5)

MySQL is a relational database management system (RDBMS), especially popular for web applications.

Hardware Requirements

Hardware CPU
RAM, Hardisk
Operating System - Linux ,Windows
Web Browser

The TME web Interface

With the TME web interface, thematic maps can be created in a web browser, without a single line of code. This is achieved through an interactive web form where the user can select between statistical indicators and various thematic mapping techniques. Mapping parameters, like the colour and size, can be readily changed. The form returns a KMZ file which can be visualised directly in the web browser using the new Google Earth plug-in, or downloaded to a computer.

How the web interface works

The TME web interface can be characterised as a Rich Internet Application (RIA), a web application that has the features and functionality of traditional desktop applications (Loosley, 2006). In a traditional web application, all processing is done on the web server and a new web page is downloaded each time the user clicks. RIAs transfer the processing necessary for the user interface to the web browser, but keep the bulk of the data back on the web server. The web interface was developed using HTML, JavaScript, Ext JS, Google Earth API and AJAX techniques. By combining these technologies, it was possible to create a responsive user experience. The web interface was developed using HTML, JavaScript, Ext JS, Google Earth API and AJAX techniques. By combining these technologies, it was possible to create a responsive user experience.

Ext JS is a cross-browser JavaScript library for building desktop-like web applications. Ext is dual licensed under the General Public License (GPL), which TME uses, and a commercial license. By using Ext JS, it was possible to build interactive form elements which responded to various events initialized by the user or the program flow. The validity check

of the user inputs is performed in the web browser and not on the server. Ext JS also controls the AJAX based communication between the web browser and the web server.
AJAX is a group of interrelated web development techniques used for creating interactive web applications. By using AJAX, it is possible to retrieve data from the web server asynchronously in the background without interfering with the display and behaviour of the existing web page. Despite its name, XML is not required as the data-interchange format.
TME uses the JSON encoding to transfer data from the web server to the web browser, as it is more readily generated and parsed by programming languagesIn May 2008, Google launched the Google Earth Plug-in together with a free JavaScript API (Rademacher, 2008). This made it possible to embed Google Earth in a web interface, instead of having to switch between two applications (the web browser and Google Earth).

When the web interface is loaded by the web browser, an AJAX request is automatically fired to the web browser. The browser is asking for a list of all available statistical indicators, and this list is returned by the web server. This information is added to the first drop-down box in the web form.

TME Application Programming Interface (API)

The Thematic Mapping Engine can also be used as an application programming interface (API). This allows thematic maps to be created with a few lines of PHP code. Existing or new applications can use this API to add thematic mapping functionality.

TME DataConnector class

The object-oriented programming (OOP) features introduced in PHP 5 were used to create an application that could be readily extended . All queries to the MySQL database are kept in a separate class (TME_MySQL_DataConnector.php). A different database can be used by adding a new PHP data connector class.

The DataConnector class contains three methods:
getIndicators ()
This method returns a list of the statistical indicators in the database.
getIndicatorYears ($indicatorID)
This method returns a list of available years for one indicator.
getDataStore ($indicatorID, $year, $region)

This method retrieves the spatial features and statistical values required to create a thematic map. The data is returned in a multidimensional array. The spatial features (e.g. countries) and the statistical values are kept separately in the array, as not all features have values, and not all values have associated features. Name, longitude, latitude and geometry (i.e. border) is added for each feature. Various metadata are added, together with the statistical values.

TME Thematic Map class

The multidimensional array retrieved from the *DataConnector* class is passed on to the *ThematicMap* class together with the mapping parameters. Figure 3.7 shows a flow chart of how the KML document is created when the *getKML* method is called. Firstly, the shared styles are defined. The shared styles will be different for each thematic mapping technique. An exception is choropleth and prism maps, which have the same shared styles. There are two *for-each* loops. The outermost loop runs through each of the years. A new KML *Folder* element is added for each year. The inner loop runs throughout all the features (e.g. countries) present in the data store. A KML *Placemark* element is added for each feature. Within this element, feature specific styles and the feature itself are added. How this is done for each thematic mapping technique is explained in section.

Data preparation

Due to special interest and availability, international statistics were used for the thematic mapping experiments. Three types of data were gathered: a spatial dataset containing world borders, statistical data from the United Nations, and international country codes. The latter was used to link statistical values to the spatial features. Although this is principally global data, the techniques developed here are scale independent.

Using open data

Since KML is an open, human-readable format, it was required to use data provided under open and non-restrictive licensing terms. The following requirements had to be met:
Access for all. Anyone with an internet connection should be able to access themapping examples. The *terms of use* of the Google mapping technology also states that “your service must be freely accessible to end users”.

Using vector data. Many data licenses do not readily permit use in an open interface as data providers are concerned about backward engineering. Using an open standard like KML would violate such licensing terms.

Allow redistribution. Many data providers forbid redistribution of their data, which is basically what this project is doing by using KML and publicly available geobrowsers.

An important goal of this project was also to show the benefits of using public domain (“open”) data.

UN statistics

UNdata is a new internet-based data service which offers free access to a wide range of global statistics through a single entry point. The database service is offered by the United Nations Statistics Division (UNSD). UNdata enables users to access a large number of UN databases, either by browsing the data series, or through a keyword search. The non-restrictive licensing terms made this data service an ideal source for this project.

World borders dataset

A world borders dataset was needed for choropleth mapping and to calculate the positions (centroids) of the proportional symbols. Ideally, border data and country statistics should be from the same source, to ensure that the areas are identical. which reflects the cartography practice of the UN Cartographic Section. Unfortunately, this dataset is only accessible to the UN community, due to sensitivity linked to international boundaries.

A world borders Shapefile meeting the above requirements was downloaded from the Mapping Hacks website. This dataset was originally derived by SchylerErle from public domain sources.

Added geographic coordinate system: GCS_WGS_1984
Polygons representing one country/area were merged into one feature.
Various feature changes to make the dataset more compatible with the ISO 3611-1 country codes used by the United Nations.
Region and sub-region codes from the UN Statistical Division were added.
Longitude/latitude values for each country were added.

Database

The world borders dataset and statistics from UNdata were stored in a MySQL database. MySQL was chosen more because of its availability than its spatial capabilities. MySQL has limited support for spatial data, but it was sufficient for this project. An alternative would be to use the PostgreSQL database with the PostGIS spatial extension, but this would exclude many potential users since PostGIS is seldom pre-installed by hosting providers.

Uploading spatial data

The GDAL/OGR library was used to upload the world borders Shapefile to the MySQL database. This library contains the *ogr2ogr* utility program which converts simple features data between various file and database formats. The following command was used to upload the features (including the attribute table):

The country borders were stored in the database using the *MultiPolygon* data type. This data type was needed because many countries consist of several polygons (i.e. land areas/islands). Some of the polygons are complex (i.e. contains holes) because of enclaves. The longitude/latitude position for each country was stored in two table columns. An alternative would be to use a spatial data type (GEOMETRY POINT). The above command also created two extra tables (*geometry_columns* and *spatial_ref_system*). These tables are required according to the Open GIS Simple Features Implementation Specification for SQL.

Uploading statistical data

The statistical indicators were downloaded from UNdata in an XML format, and uploaded to the MySQL database through a tailored PHP script.The script above reads and parses an XML file. Since it was impossible to download indicator metadata from UNdata, this information had to be entered manually in *phpMyAdmin*.

Querying and transforming spatial data

MySQL stores spatial data in an internal geometry format (MySQL, 2008). The data can only be retrieved in this internal format or as *Well-Known Text* (WKT) or *Well- Known Binary* (WKB) representations (OGC, 1999). WKT/WKB predates the Geographical Markup Language (GML), which is now a more commonly used format. The coordinates returned from the MySQL database had 6 or more decimal points. This is far more than needed, and does not reflect the precision of the simplified borders.

Thematic mapping techniques for KML

This section describes how the KML standard was utilised for thematic mapping.

KML icon images can be scaled and colourised by using the *IconStyle* element. Only one image is thereby needed to create symbols in different colours and sizes. This reduces the total file size and improves the performance.

The symbol image is referenced in a shared style:Examples have included choropleth maps, bar charts, prism charts, Collada objects, map legends,including examples of animated time series. If you're into the techy part of this stuff, the series is well worth a look for the ideas you'll get from it, and the downloadable example files. But for non-techy types, the prospect of learning how to do this might have been a bit intimidating. Heck, I find it intimidating, and I know at least a little about this stuff. But Bjorn has taken his examples to the next level, by creating an online tool that can convert datasets to thematic Google Earth maps on the fly, the Thematic Mapping Engine"

Proportional image icons

KML icons are used to visualise various point data, and custom icons can be added by referencing an image stored on the local file system or a remote web server.Two symbols, a circle and a square, were created using Adobe Photoshop Elements. A shadow effect was added to give the icons a slightly 3-D appearance. The symbols are white on a transparent background, and saved as PNG files.

KML icon images can be scaled and colourised by using the *IconStyle* element. Only one image is thereby needed to create symbols in different colours and sizes. This reduces the total file size and improves the performance. The symbol image is referenced in a shared style:

When a KML *Point* element is contained by a *Placemark* element, the point itself determines the position of the *Placemark*'s image icon (Google, 2008a). This is achieved by defining the longitude, latitude and (optional) altitude within the *coordinates* element.

The drawback of this method is that the map is, so far, only viewable in Google Earth. Other geobrowsers are unable properly to scale and colourise the icon images (seeFigure 3.10).

When comparing the results from the Google Earth plug-in with the Google Earth desktop program, there is a noticeable difference: the icons are much bigger in the plug-in. The reason is different viewport sizes, and it can be considered as a Google Earth bug. There are two different ways of adjusting the size of the planet in Google Earth. The left hand visualisation in figure 3.11 shows the earth in a zoomed out view. The circle images are scaled properly. The other way of changing the size of the planet is by adjusting the Google Earth window. The problem is that the circle images maintain their size while the planet shrinks or expands. The KML *Icon* element is here used in a way that was probably not intended.

Proportional 3-D Collada objects

One unique feature to KML is the ability to embed 3-D models or objects into the KML file (Turner, 2006). The 3-D objects used in this project were downloaded from 3D Warehouse, an online repository for 3D models. The objects were edited in Google Sketch Up to make them more suitable for thematic mapping: The *Link* element specifies the Collada object to load. The *Location* element specifies the coordinates of the object's origin in latitude, longitude and altitude. The *Scale* element scales the object along the x, y and z axes in the object's coordinate space. As for bars, the z (height) dimension could represent a different statistical indicator than the x/y dimension. Collada objects can also be placed on top of each other by specifying an increasing altitude value (e.g. to create stacked bars).

Bar maps

The regular polygons described above can be turned into bars by adding an altitude value for each coordinate tuple (vertex). Altitude values are in metres above sea level,and should be directly proportional to the statistical value.

The *extrude* and *altitudeMode* elements are needed to create a 3-D bar. The *extrude* element specifies that the polygon is connected to the ground. When the *altitudeMode* is *absolute*, the altitude of each coordinate tuple is relative to sea level, regardless of the actual elevation of the terrain beneath the element. When bars are rendered on top of a 3-D landscape, the *relative To Ground* attribute should be used.

Choropleth maps

The thematic mapping techniques presented so far are all using the longitude/latitude position (centroid) for each feature. A choropleth map requires the geometry representing the border of the feature (e.g. country). The fill colour is specified for each feature, since the polygons are colourised according to a statistical value.

Prism maps

As regular polygons can be turned into 3-D bars by adding an altitude value for each coordinate tuple , irregular polygons can be turned into prisms.The *extrude* and *altitudeMode* elements have to be specified for each polygon contained by a *MultiGeometry* element. Figures 16 and 38 show examples of a prism map displayed in Google Earth. When 3-D prism maps are rendered in Google Earth, holes appear in the polygons representing large countries with low values on a statistical indicator (i.e. those with a low altitude value).

Map legend

As symbol size varies with scale (zoom level), useful symbol legends are difficult to create. An alternative method is to duplicate the symbology by supporting a colour legend for all thematic mapping techniques, also for proportional symbol maps.

The colour legend informs the user about the range of values (min and max), and where the different symbols are positioned in this range. By using an unclassified scheme, a unique visual shade is assigned to each unique data value. The colour scale can also be classed by using equal intervals or quantiles.

Data preparation

The first step involved finding appropriate statistical and spatial data that could be combined and used for thematic mapping. The data had to be gathered from various sources and stored in a database.

Thematic mapping with KML

A scripting language (PHP) was used to query the database and transform the result into KML documents representing various thematic mapping techniques.

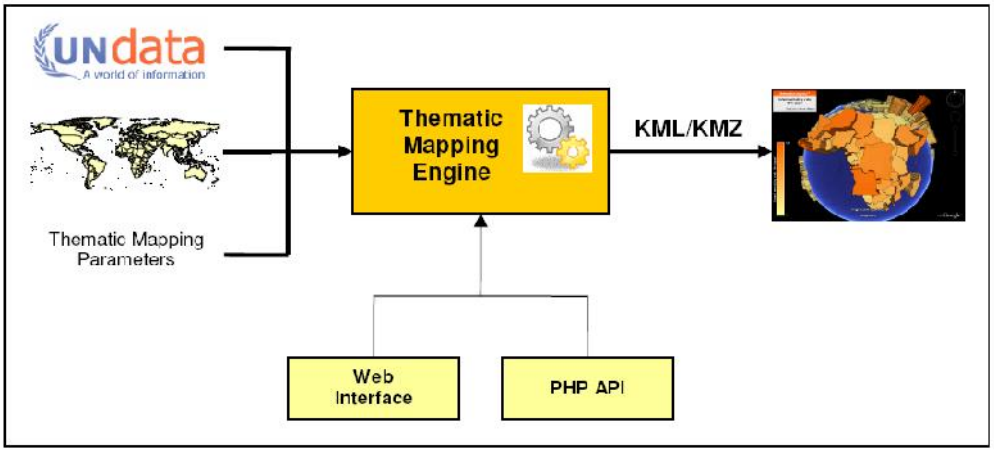


Figure 3.1: The interfaces of the Thematic Mapping Engine.

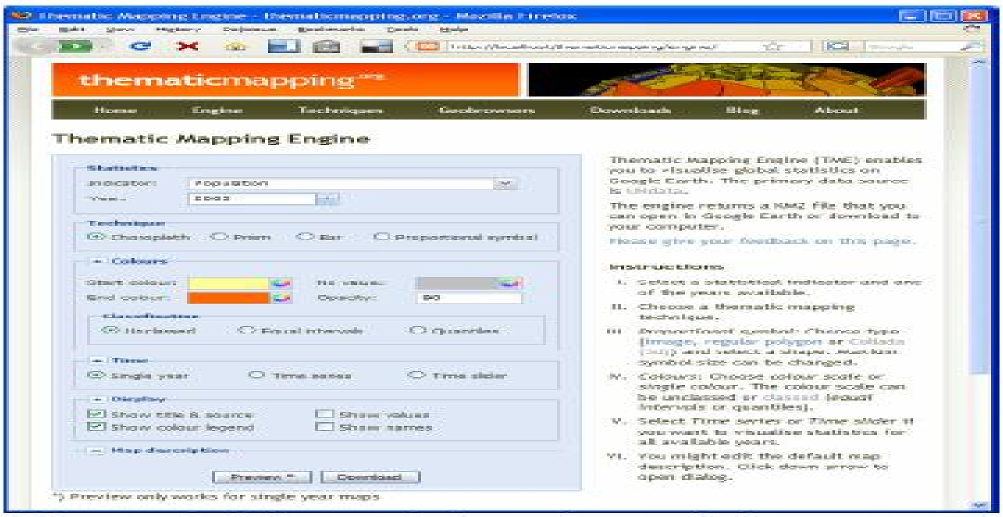


Figure 3.2 The web interface of the Thematic Mapping Engine

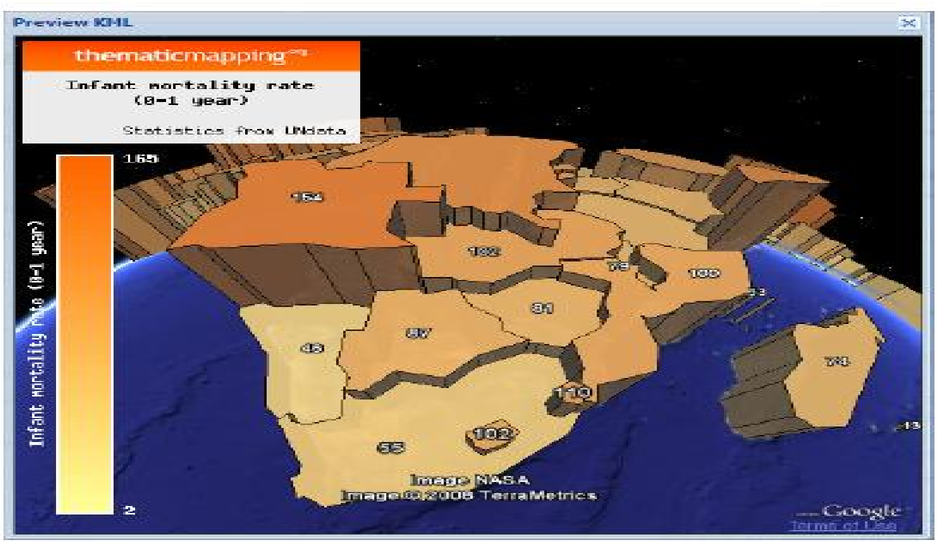


Figure 3.3: Prism map shown with the Google Earth Plug-in.

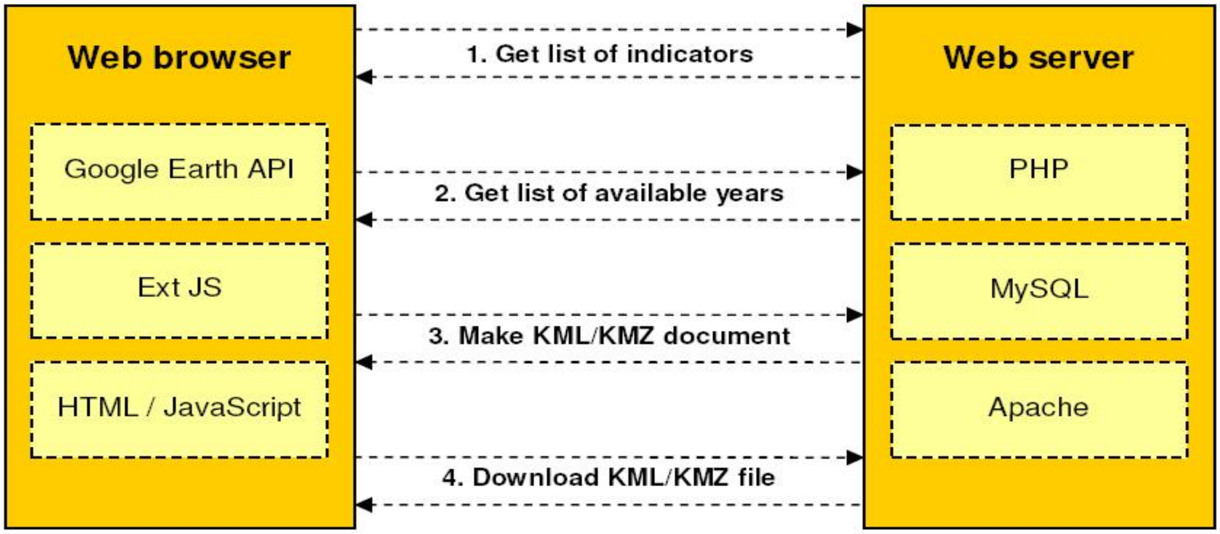


Figure 3.4 AJAX based communication between web browser and web server.

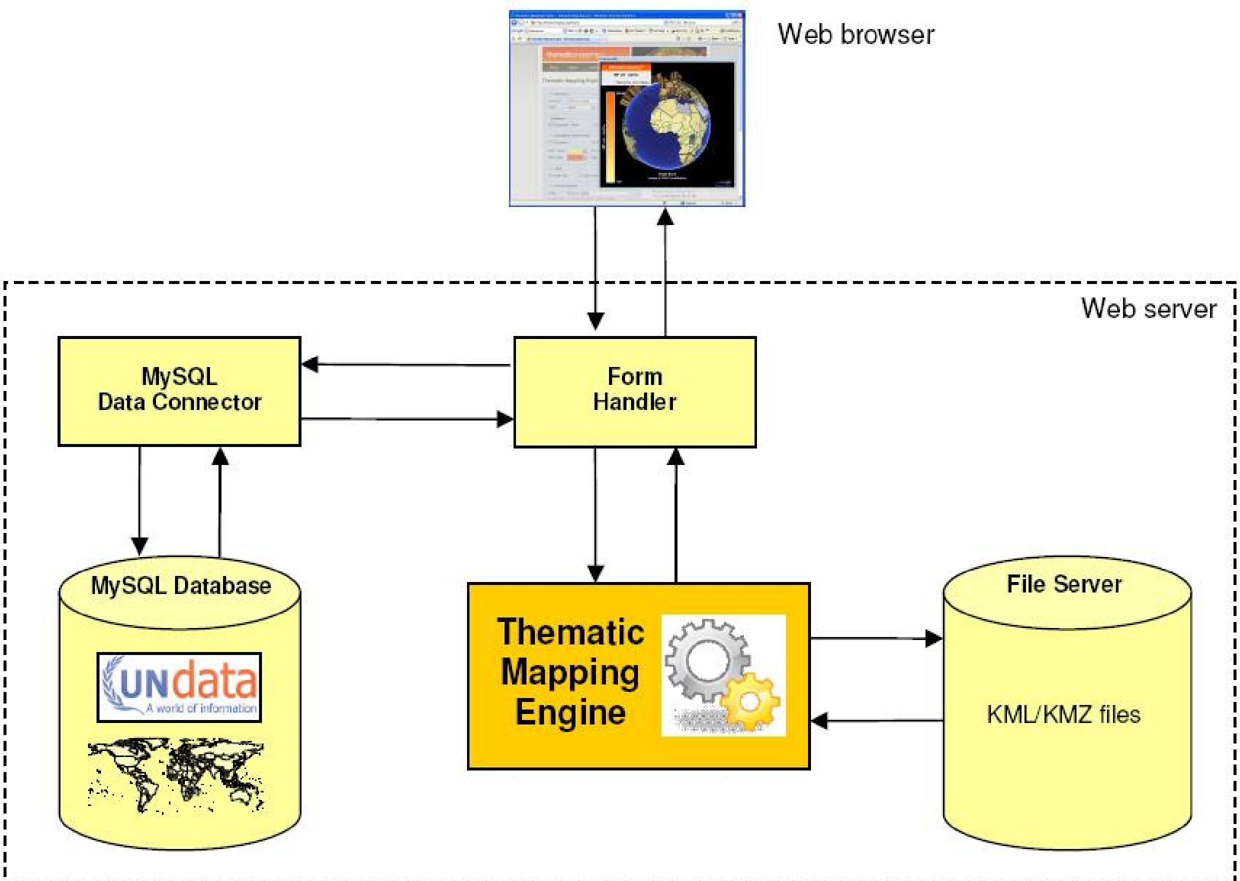


Figure 3.5 TME web server infrastructure

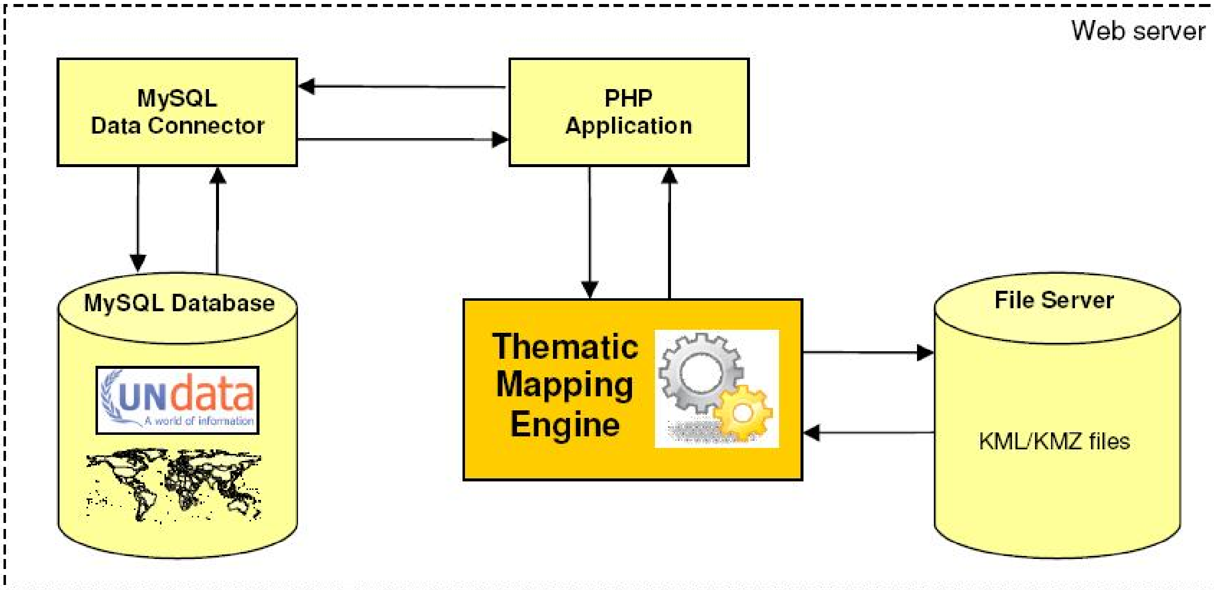
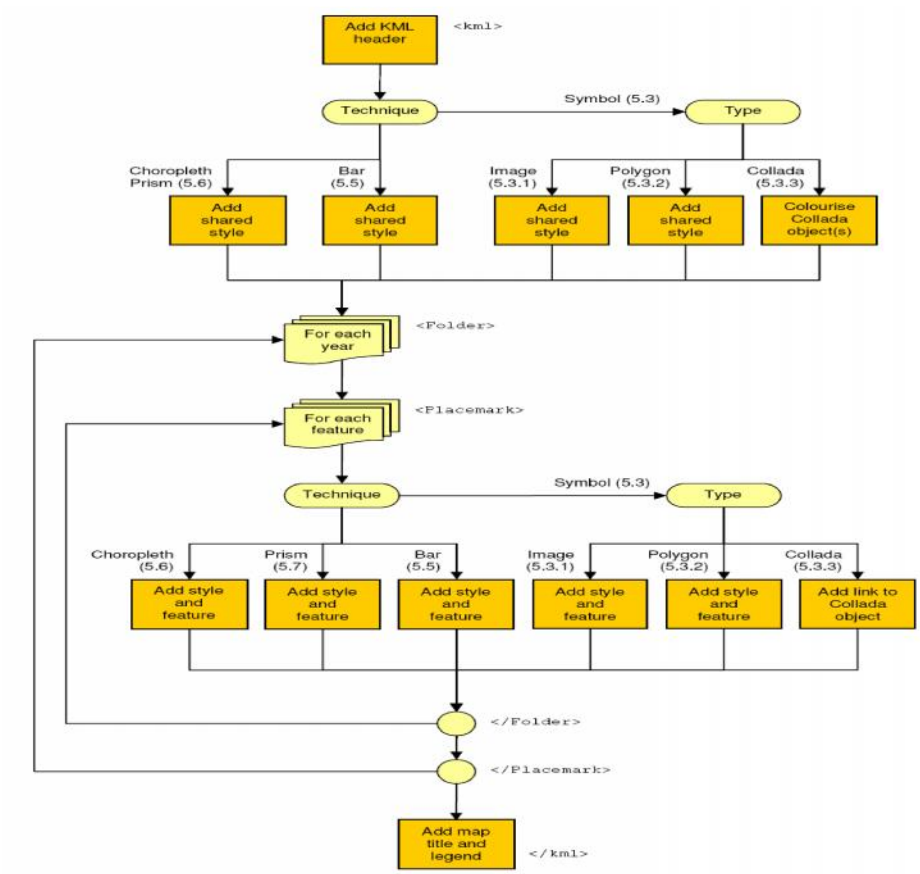


Figure 3.6 TME Application Programming Interface (API). The Form Handler script is here replaced by a custom PHP script.



3.7 Flow chart showing how a KML document is created

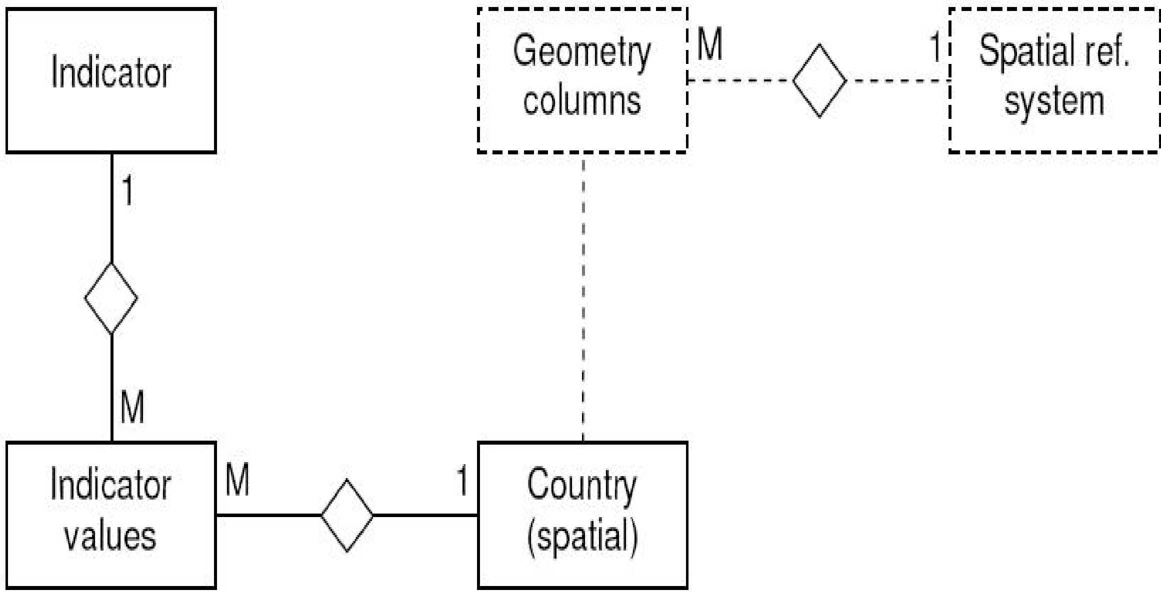


Figure 3.8 Entity-Relationship (ER) diagram showing the database structure.

Table3.1 Thematic mapping techniques

.

Thematic mapping techniques	
Choropleth maps	The thematic mapping techniques presented so far are all using the longitude/latitude position (centroid) for each feature. A choropleth map requires the geometry representing the border of the feature (e.g. country).
Bar maps	The regular polygons described above can be turned into bars by adding an altitude value for each coordinate tuple (vertex). Altitude values are in metres above sea level, and should be directly proportional to the statistical value.
Prism maps	As regular polygons can be turned into 3-D bars by adding an altitude value for each coordinate tuple , irregular polygons can be turned into prisms.
Proportional symbols	
Image, Regular polygon or 3-D object	KML has no built-in support for regular polygons so these have to be generated by calculating the longitude/latitude location for each vertex of the polygon.
colour	The colour scale can be unclassed or classed (*equal intervals* or *quantiles*). The number of Classes can be changed (2-9 classes).
Time	Select Time series or Time slider to visualize statistics for all available years. In statistics, signal processing, and many other fields, a time series is a sequence of data points, measured typically at successive times, spaced at (often uniform) time intervals.
Equal Intervals	Where the range of data values are "divided into equal size sub-ranges
Quantiles	where each class receives an equal number of features;
Display	Select information elements that should be displayed on the map.
Map description	The default title, description and source of the map can be changed.

Figure 3.9 : Proportional symbol map in Google Earth showing the relative population in each country of the world.

Figure 3.10 : The same KML file shown in Microsoft Virtual Earth. The image icons are not scaled or colourised.

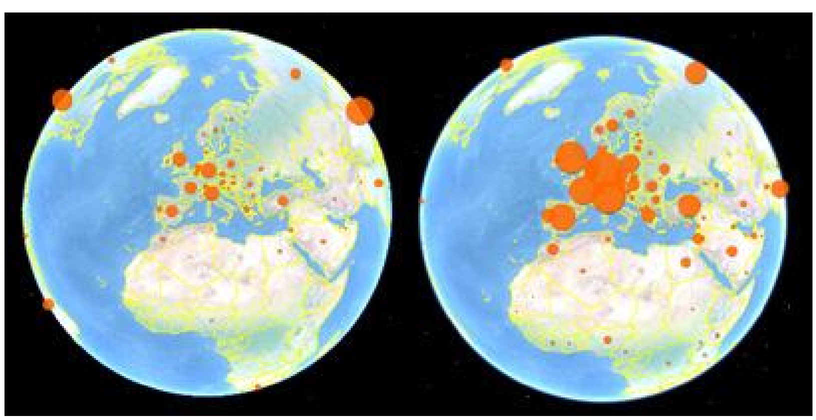


Figure 3.11: Proportional image icons in Google Earth.

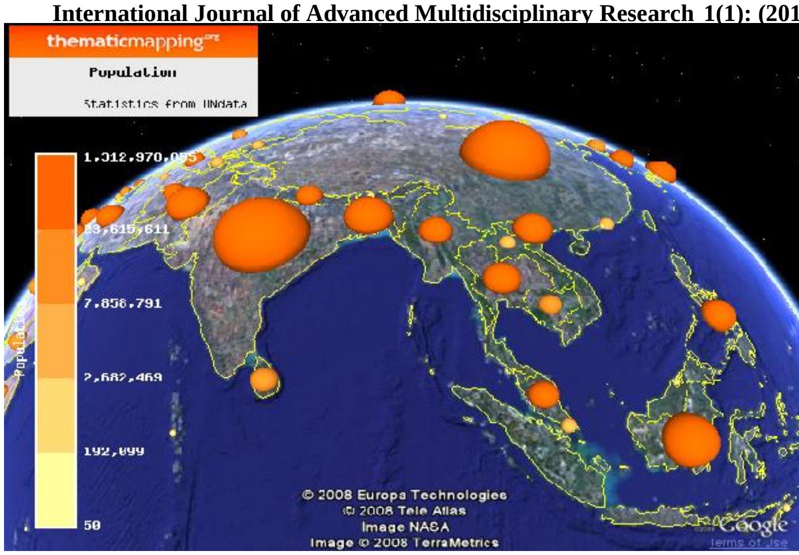


Figure 3.12: Population (2005) in Southeast Asia visualised with 3-D domes.

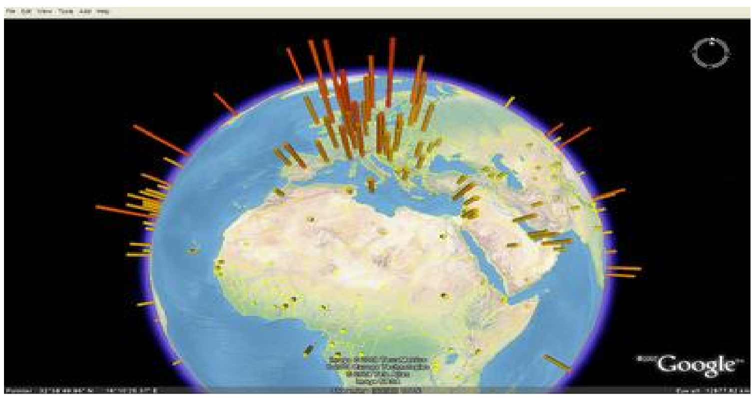


Figure 3.13: 3-D bar

Import shape files into the postgreSQL database. This library contains the shp2pgsql utility program which converts simple features data between various file and database formats. The states or districts borders were stored in the database using the MultiPolygon datatype (postgresql). The longitude/latitude position for each country was stored in two table columns. An alternative would be to use a spatial data type (GEOMETRY POINT).
The above command also created two extra tables (*geometry columns* and *spatial_ref_system*). The statistical indicators were downloaded from UNdata in an XML format, anduploaded to the Post gresql database through a tailored PHP scrip . The script above reads and parses an XML file. Since it was impossible to download indicator metadata from UNdata, this information had to be entered manually in *phpMyAdmin*.

Thematic Mapping Engine

The Thematic Mapping Engine was developed to demonstrate how these techniques could be utilised in a web application.
The Thematic Mapping Engine works on a high level. The engine takes statistical data (attributes), spatia0l features and thematic mapping parameters as input and returns a KML/KMZ file. This file can be viewed in Google Earth, or other geobrowsers supporting the KML standard.TME can be accessed from a web interface .

References

Ayyar, N. P. (1969). *Crop Regions of Madhya Pradesh*. Geographical Review of India.

Alesheikh, A. A., and Helai, H. A. (2000). *Web GIS: Technologies and Its Applications*.Symposium on Geospatial Theory, processing and applications,Ottawa 2002.

Bhatia, S. S. (1965). *Crop Concerntration and Diversification*, Economic Geography, Vol.47(1), pp. 38-56.

Burrough, P. A., and McDonnell, M. F. (1998). *Principles of Geographical Information System*. Oxford University Press, New York.

Doyle, A. (1999). *Web Map Server Interface Specfication.* OpenGIS project Document, URL: http://www.opengis.org.

Hussain, M. (1972). *Patterns of Crop Concentration in Uttar Pradesh.* Geographical Review of India, Vol.32(3), pp. 169-185.

Lakshminarayana, V., and Rajagopal, S. P. (1997). *Optimum Cropping pattern for basin in India.* Journal or Irrigation and Drainage Engineering ASCE, Vol-103, pp. 53-70.

Majid Hussain, (1996). *Systematic Agricultural Geography*. Rawat Publication, New Delhi.

Nagarathinam, S. R. (1990). *Agroclimatic Factors and TheiImpacts on Cropping Pattern*. University of Madras, Chennai.

Raghavan, V., and Santitamont, P. (2002). *Implementing Web GIS Application Using Open Source Software.* GIS Development Publication, India.

Sandvik, B. (2008). *Research Paper: Using KML for Thematic Mapping*", M.Sc., GIS Dissertation, University of Edinburgh.

Slocum, T. A., McMaster, R. B., Kessler, F. C., and Howard, H. H., (2005). *Thematic Cartography and Geographic Visualization.* Second Edition, Person Education Inc.

Turner, A. J. (2006). *Introduction to Neogeography.* O'Reilly Media Inc.

Zhong-Ren Peng and Ming-Hsiang Tsou, (2003). *Internet GIS*. John Wiley & Sons.

URL References:

http://groups.google.com/group/Google-Maps-API
http://drjill.net/?tag=thematic-mapping
http://www.gisdevelopment.net/application/agriculture/cropping/index.htm,
http://www.thaibicindia.org.in/study/tamil_nadu/Overview.pd
http://www.trichy.tn.nic.in/distprof.htm
http://rbidocs.rbi.org.in/rdocs/Publications/PDFs/35598.pdf
http://www.tn.gov.in/crop/overview.htm
http://en.wikipedia.org/wiki/Tiruchirapalli_District
http://www2.dpi.qld.gov.au/fieldcrops/10904.html
http://www.indianetzone.com/6/tiruchirapalli.htm
http://en.wikipedia.org/wiki/Agricultural
http://tnmaps.tn.nic.in/tamil

