The Design of Relational Database for Multipurpose WebGIS Applications

E Sutanta and E K Nurnawati

Department of Informatics Engineering, Institut Sains & Teknologi AKPRIND Yogyakarta
Kalisahak Street #28, Yogyakarta, Indonesia

E-mail: edhy_sst@akprind.ac.id, ernakumala@akprind.ac.id

Abstract. GIS concepts and applications evolved from a desktop-based into web-based and mobile. These changes require a new approach in the GIS database, from spatial data separation and attribute data into a database structure approach. The problem is the database for GIS is generally designed just for specific users, objects, and territory scopes, so it can’t be applied to other applications. This paper proposed a relational database model (RDBM) design that can accommodate new changes and needs in the user's aspect, the objects that are processed and displayed, and the scope of the region in GIS Web Applications (WebGIS). The study was conducted in 3 stages, namely a preliminary study to obtain an overview of the concept, design, and application of RDBM on various WebGIS. Next, identified potential problems in database design for WebGIS, and finally proposed an alternative solution in the form of a new database design template that could accommodate the possibility of new changes and needs. Database design is done using MySQL. The database design process for WebGIS applications needs to anticipate the possible growth and development of different user group data, object categorization to subcategories, and the scope of the object area. Designs that meet these criteria can be done from the beginning of the design process, that is by changing two things. First, change the view of the database designer from the user view level to the conceptual view level so that the database design results can be used by several different applications. Second, expand the view of the database designer from the specific scope into a broader scope in the user aspect, object categorization, and territory coverage. Thus, the database design obtained will be in accordance with the RDBM concept.

1. Introduction

GIS concepts and applications continue to evolve. GIS is a system designed to work with spatially-referenced data or geographic coordinates, having the ability to receive, manage, analyze, and display data [1]. The first computer-based GIS application was CGIS (Canadian GIS) implemented by the Department of Energy, Mining and Resources, Ottawa, Ontario, Canada in 1967 [2]. The development of microcomputer has spurred ESRI, CARIS, MapInfo, and other vendors incorporating spatial data separation approaches and attributes on GIS applications into database structures. Industrial developments in the 1980s and 1990s enabled GIS to be applied to UNIX workstations and personal computers, and since then GIS applications have been developed for diverse fields and objects [3]. ICT alterations have had an impact on GIS applications, from desktop-based, changed to web-based applications (WebGIS), and are currently mobile-based. In general, GIS applications can be divided into two groups, namely desktop-based and distributed GIS. Distributed GIS consists of WebGIS and Mobile GIS applications [2].
The availability of easily accessible online maps with no geographical restrictions and a wide range of attractive display options is one of the key drivers of WebGIS application development. For example, the results of fast-tracking on the internet using the keyword "WebGIS application" found as many as 66,400 posts. If keyword replaced using "WebGIS application example" found 17,000 posts, on the other hand, changed using the keyword "WebGIS application research" found 13,400 words. Furthermore, specifically if using the keyword "WebGIS application thesis" found 6,330 posts. The result shows that WebGIS application has become an interesting topic for many developers. Some examples of GIS applications for college have been developed by[3] and [4]; a tourist attraction or culture found on [5], [6], [7], [8], [9], [10], and [11]; layout by [12]; industry by [13] and [14]; health services on [15] and [16]; Other objects such as houses of worship, boarding houses, cargo agents are found on [17], [18], [19], and [20]. Most of these applications are WebGIS and other mobile-based, mostly developed using open source software and a few other paid apps. WebGIS applications are also widely developed by government agencies in Indonesia, i.e., found at http://gis.pusair-pu.go.id/, http://kewilayahan.bantulkab.go.id/, and https://wartacara.com/ukm/.

GIS applications outline has the same model with computer-based processing, including input, process, and output. The sources of input data on GIS can come from tables, reports, field measurements, digital data, maps, satellite images, aerial photographs, and others. The entire data entered is stored in the GIS database. GIS database processing in principle is to store the inputted data into the database, manipulate the data as needed, and retrieve data to be displayed as the outcome of the application. There are several form of GIS application, i.e., maps, tables, reports, digital information, and other forms according to user needs.

Like the GIS desktop application, WebGIS requires database support that contains graphical data (vector and raster) and attributes (contains tables of tables that relate to graphical databases). Currently, database management for GIS applications is mostly done by utilizing DBMS (Database Management System) software. The main reason for the convenience store (store) and retrieve (retrieve) data. While other reasons are the innate nature of the DBMS, including related to the problem of physical data independency, data redundancy, logical data independency, access speed, data standardization, and data integrity [21]. The database approach in GIS applications also facilitates management and avoids errors due to data loss or duplication.

The most important support of the database in a GIS application is to provide the attribute data query function to obtain a spatial position or vice versa, then display it on the map. In the Relational Data Base Model (RDBM) approach, attribute data in GIS is stored in tables (called relation), each table is composed of some columns with unique names (called attributes) [22]. Each row of data in a RDBM is called a record that can be identified by using a unique primary key (PK). Thus, the query function to obtain spatially or otherwise, the position can be fulfilled by connecting between tables in GIS database using foreign key (FK). Furthermore, the relationship between spatial data and attributes is performed using the unique code provided in each of these data.

A common problem with WebGIS database design in those examples is that databases are developed for the scope of a particular area or region of a specific nature and the type and a limited number of users. As a result, the developed databases will be difficult to accommodate changes in new needs, such as expanding fields or coverage areas to be processed and displayed or adding users to WebGIS applications. The developed databases can only be overcome since the early stages of WebGIS application development, especially in the database design phase. Attempts to resolve the issue were once proposed on [23] through the integration of several database designs, which include tourist attractions, schools, cultures, hotels, and inns, as well as government offices, into a new database design which is then tested on mobile applications. In the integrated database design, all objects are stored in the same table in the database and are differentiated by object code. The proposed database for mobile applications is composed of ten classes, i.e., province, regency, district, category, object, object_detail, user, user_level, log_activity, and change_password. The classes are then implemented into 13 database tables, i.e., province, regency, district, pos_code, user, user_level, pict, object, object_detail, category,
Proposed on [24] emphasizes on how to obtain a flexible database design that can accommodate the growth and development of data so that it can be implemented for various GIS applications. The proposal is a database design consisting of 11 tables, i.e., village, district, regency, province, sub_category, category, user, user_level, user_status, object, and object_status Relationships between tables are shown in FK definitions in user tables, objects, sub_category, village, district, and regency. The integrity of the referential database is implemented through ONUPDATECASCADE and ONDELETECASCADE constraints. Approach on [23] and [24] is an integration solution that is done at the back end level [25]. The integration process begins with a schema matching stage that can be done with many different strategies, e.g., hybrid as done on [26].

The novelty of the approach in this study lies in the expansion of user views from one specific WebGIS application into a wider range of WebGIS applications and the expansion of the designer's view. From design to only one specific WebGIS application becomes more widespread on the merger of multiple WebGIS applications. Thus the database design can be used for a wider scale in the aspect of the user, object category, and region.

2. Methods
The study was conducted in 3 stages, namely a preliminary study to obtain an overview of the concept, design, and application of RDBM on various GIS Web applications. We then identified potential problems in database design for WebGIS, and finally proposed an alternative solution in the form of a new database design template that could accommodate the possibility of new changes and needs. The steps in this study are as follows:

2.1 Preliminary study to obtain an overview of the concept, design, and applications of RDBM on various WebGIS applications.
2.2 Identified potential problems in database design on various WebGIS applications.
2.3 Proposed a new database design template for various WebGIS applications.
2.4 Comparing the old and new database designs for data growth on user groups, object categories, and scope of the object areas.

3. Result & Discussion
3.1 Minimum Database Requirement in WebGIS Applications
In theory, database needs for WebGIS applications in the form of two types of data namely spatial and attributes data. Spatial data is represented by the coordinate position data of each stored object, in which the location of each object is identified and associated with the attribute data. The coordinate position data of each object shows the position of latitude and longitude. In RDBM, attribute data is represented by the detailed values of data stored in linked tables in the database. The simplest database design for WebGIS applications is to store spatial data for one particular object type and attribute data consisting of administrator user data and detailed object-related data in a particular region. The simplest representation of the needs of the database can be defined, e.g., given the name webgis01 as shown in the following schema:

a. Table structure for object:
CREATE TABLE IF NOT EXISTS `object` (`obj_code` char(10) NOT NULL DEFAULT '', `obj_latitude` int(11) NOT NULL, `obj_longitude` int(11) NOT NULL, `user_name` varchar(10) NOT NULL);

b. Table structure for object_attribute:
CREATE TABLE IF NOT EXISTS `object_attribute` (`obj_code` char(10) NOT NULL DEFAULT '', `obj_name` char(50) NOT NULL, `obj_description` varchar(100) NOT NULL);

c. Table structure for user:
CREATE TABLE IF NOT EXISTS `user` (`user_name` varchar(10) NOT NULL DEFAULT '', `user_password` varchar(10) NOT NULL DEFAULT '', `user_email` varchar(50) NOT NULL);
3.2. Problems on webgis1 Database

There are many problems on webgis1 database i.e.:

1. User data can only be increased on the record number.
2. Data object can only be increased on the record number.
3. Data object can only be increased on the record number.
4. Data objects represent a category of objects, it is difficult to adding of new categories.
5. Data object is not possible to do a track by area/region certain.

It is necessary to design an appropriate database table structure so that the detailed values stored can meet the needs of users, both current and future. Such needs changes can occur in the type and number of users, the scope of the application, and/or the scope of its territory.

3.3. The webgis2 Database Design Improvement

The proposed WebGIS database design improvement called webgis2 is made by changing the approach of design that is specific to the user, the object category, and the particular region. webgis2 being a more general design that allows for use by multiple groups of users, for various categories of objects, and for wider territory/different. The improvement can be achieved in two ways at a time. The first is to change the database designer's viewpoint to a conceptual view position so that the database design results can be used by several different applications. Both extend the view of the database designer so that the database is designed to be used for a wider scale in the user, object, and region aspects. Figure 1 shows a database design approach often used in applications such as webgis1, while Figure 2 shows a new database design approach for proposed webgis2. In Figure 1 it appears that each WebGIS application uses separate files with other applications, while in Figure 2 shows a database used by some WebGIS applications. This approach fits in with the stable database concept as found on [27], [22].

![Figure 1. The database design approach on webgis1](image)

![Figure 2. The database design approach on webgis2](image)

Some of the considerations required when drawing up new database design improvement proposals for WebGIS applications are as follows:

1. Need to add status data on user data to record user status. The type of status should be anticipated to not only contain "active" and "not active" because there is a possibility of adding a new user but not yet verified "not active yet". For this change to be done more easily, it is necessary to add a reference table to record the status code and its description.

2. Need to add object category data. Each data object is associated with the appropriate subcategory or category to facilitate the search. For this change to be done more easily, we need to add a reference table to record the category code and its description. If required, sub-category data can be broken down into sub-categories, and so on.

3. Need to add data object area. The division and encoding of the object area can be done by referring to the division of existing and commonly used areas, eg provinces, districts, subdistricts, and villages.
In order to make the area change easier, you need to add the reference table of the region code and its description.

For WebGIS applications covering areas in Indonesia, territorial codes have been formally established by the Government of the Republic of Indonesia in the Minister of Home Affairs Regulation number 66 of 2011, on the Code and Regional Data in the Information System of Population Administration (ISPA) which applies nationally. In the regulation the regions is encoded as province: char[2], regency: char[4], district: char[6], and village: char[10]. Examples of the use of area code in ISPA are 34 (DIY Province), 3401 (Kulon Progo District), 340106 (Sentolo regency), and 34010620008 (Banguncipto villages). The webgis02 database design is displayed on the following schema:

a. Table structure for `district`:
   CREATE TABLE IF NOT EXISTS 'district' ('district_code' char(6) NOT NULL, 'district_name' varchar(50) NOT NULL, 'regency_code' char(4) NOT NULL);

b. Table structure for `object`:
   CREATE TABLE IF NOT EXISTS 'object' ('object_code' char(10) NOT NULL DEFAULT '', 'object_name' varchar(50) NOT NULL, 'object_address' varchar(50) NOT NULL, 'object_contact' varchar(50) NOT NULL, 'object_description' varchar(100) NOT NULL, 'object_latitude' decimal(9,6) NOT NULL, 'object_longitude' decimal(9,6) NOT NULL, 'object_picture' blob NOT NULL, 'village_code' char(10) NOT NULL, 'sub_sub_category_code' char(4) NOT NULL, 'user_name' varchar(10) NOT NULL);

c. Table structure for `object_status`:
   CREATE TABLE IF NOT EXISTS 'object_status' ('object_status_code' char(1) NOT NULL, 'object_status_name' varchar(12) NOT NULL);

d. Table structure for `province`:
   CREATE TABLE IF NOT EXISTS 'province' ('province_code' char(2) NOT NULL, 'province_name' varchar(50) NOT NULL);

e. Table structure for `regency`:
   CREATE TABLE IF NOT EXISTS 'regency' ('regency_code' char(4) NOT NULL, 'regency_name' varchar(50) NOT NULL, 'province_code' char(2) NOT NULL);

f. Table structure for `sub_category`:
   CREATE TABLE IF NOT EXISTS 'sub_category' ('sub_category_code' char(4) NOT NULL, 'sub_category_name' varchar(20) NOT NULL, 'category_code' char(2) NOT NULL);

g. Table structure for `sub_sub_category`:
   CREATE TABLE IF NOT EXISTS 'sub_sub_category' ('sub_sub_category_code' char(6) NOT NULL, 'sub_category_code' char(4) NOT NULL, 'sub_sub_category_name' varchar(20) NOT NULL);

h. Table structure for `user`:
   CREATE TABLE IF NOT EXISTS 'user' ('user_name' varchar(10) NOT NULL DEFAULT '', 'user_password' varchar(10) NOT NULL, 'user_email' varchar(50) NOT NULL, 'user_phone' varchar(12) NOT NULL, 'user_level_code' char(1) NOT NULL, 'user_status_code' char(1) NOT NULL);

i. Table structure for `user_level`:
   CREATE TABLE IF NOT EXISTS 'user_level' ('user_level_code' char(1) NOT NULL, 'user_level_name' varchar(18) NOT NULL);

j. Table structure for `user_status`:
   CREATE TABLE IF NOT EXISTS 'user_status' ('user_status_code' char(1) NOT NULL, 'user_status_name' varchar(12) NOT NULL);

k. Table structure for `village`:
   CREATE TABLE IF NOT EXISTS 'village' ('village_code' char(10) NOT NULL, 'village_name' varchar(50) NOT NULL, 'district_code' char(6) NOT NULL);

3.4. The webgis2 and webgis2 Comparison

Design on webgis2 allows to accommodate the growth and development of different user group data, object categorization to sub-categories, and the scope of the object area. User data can grow and expand from only one predetermined level category (in webgis1) can be expanded with reference tables
user_status and user_level (in webgis2) so that it can be applied to the status and level of the wider user. Data objects can grow and expand from only one category of objects (in webgis1) can be expanded with reference tables district, object_status, province, regency, sub_category, sub_sub_category, user_level, user_status, and village (in webgis2) so that it can be applied to wider region. In some cases, attribute data in the WebGIS database is also equipped with photos, videos, and other supporting data. This can be done by adding it to the object. Please follow these instructions as carefully as possible so all articles within a conference have the same style to the title page. This paragraph follows a section title so it should not be indented.

4. Conclusion
The database design process for WebGIS applications needs to anticipate the possible growth and development of different user group data, object categorization to subcategories, and the scope of the object area. Designs that meet these criteria can be done from the beginning of the design process, that is by changing two things. First, change the view of the database designer from the user view level to the conceptual view level so that the database design results can be used by several different applications. Second, expand the view of the database designer from the specific scope into a broader scope in the user aspect, object categorization, and territory coverage. Thus, the database design obtained will be in accordance with the RDBM concept.

5. Acknowledgments
The authors would like to thanks the Directorate of Research and Community Service, Directorate General of Research and Development Strengthening at the Ministry of Research, Technology and Higher Education of the Republic of Indonesia under Fundamental Research Grant Contract No.: B/1435.21/L5/RA.00/2019 and Institut Sains & Teknologi AKPRIND Yogyakarta for supporting facilities for this research.

6. References
[1] Charter D 2016 Desain dan Applikasi GIS (Jakarta: PT. Elex Media Komputindo)
[2] Irwansyah E 2016 Sistem Informasi Geografis: Prinsip Dasar dan Pengembangan Aplikasi (Yogyakarta: DigiBooks)
[3] Sutanta E 2010 Membangun SIG Perguruan Tinggi DIY dengan CMS Joomla dan Google Map Information Management and Technology (IMATEC) 1 37
[4] Suratmo D Widyastuti N and Suraya 2014 Membangun Sistem Informasi Geografis Perguruan Tinggi Daerah Istimewa Yogyakarta Berbasis Web Script 2 68
[5] Prabowo E T Sholeh M and Iswahyudi C 2013 Sistem Informasi Geografis dalam Pencarian Lokasi Museum di Daerah Istimewa Yogyakarta Script 1 13
[6] Sholeh M and Haryanto R 2014 Sistem Informasi Berbasis Geografi untuk Pendukung Promosi Wisata Budaya di Kota Yogyakarta Jarlit 10 64
[7] Zuliyanto Hamzah A and Suraya 2014 Sistem Informasi Lokasi Wisata Candi di DIY Berbasis GIS Script 1 184
[8] Wijayanto P Sutanta E and Nurnawati E K 2014 Aplikasi Sistem Informasi Lokasi Hotel Berbintang di Yogyakarta Berbasis Webgis Script 2 50
[9] Setiadi T and Yuliansyah H 2015 Kampung Wisata Online Berbasis SIG sebagai Upaya Meningkatkan Partisipasi Warga dalam Mengelola dan Mempromosikan Pariwisata Kota, Jarlit 11 85
[10] Suraya and Sholeh M 2015 e-Museum Sebagai Media Memperkenalkan Cagar Budaya di Kalangan Masyarakat Jarlit 11 24
[11] Kusumaningsih R Y R Kurniawan T and Pradhityo S 2016 Aplikasi Pengembangan Sistem Informasi Bangunan Cagar Budaya di Kota Yogyakarta Berbasis Web Mobile dan Location Based Service Jarlit 12 53
[12] Andika L A A and Ariandi M 2015 Sistem Informasi Geografis Ruang Terbuka Hijau
Kawasan Perkotaan (RTHKP) Palembang, Student Colloquium Sistem Informasi & Teknik Informatika (SCSITI) Universitas Bina Dharma Palembang

[13] Ariandono Y Hamzah A and Nurnawati E K 2014 Sistem Informasi Lokasi Industri di Kabupaten Serang Berbasis Geographic Information System (GIS) Script 1 176

[14] Wibowo H Lestari U and Triyono J 2014 Sistem Informasi Potensi Industri di Kabupaten Bantul Berbasis Geographic Information System dan Location Based Service Script 1 72

[15] Hege Y B L Lestari U and Nurnawati E K 2014 Sistem Informasi Geografis (SIG) Pelayanan Kesehatan di Kotamadya Yogyakarta Berbasis Web Script 1

[16] Nurnawati E K Muryanto J 2014 Aplikasi Mobile Berbasis Lokasi untuk Penyedia Lokasi Layanan Kesehatan di Yogyakarta Proceeding of Seminar Nasional Aplikasi Sains & Teknologi (SNAST) (Yogyakarta: IST AKPRIND Yogyakarta) A-293

[17] Sari W E Sholeh M and Hamzah A 2013 Penerapan Jquery Mobile dan Php Data Object pada Aplikasi Pencarian Lokasi Tempat Ibadah di Yogyakarta Script 1 69

[18] Pratikto H S Suraya and Sutanta E 2014 Sistem Pencarian dan Pemesanan Rumah Kos Menggunakan Sistem Informasi Geografi (SIG) Script 1 110

[19] Suharto U H Triyono J and Sutanta E 2014 Aplikasi Pencarian Lokasi Agen TIKI (Titipan Kilat) di Yogyakarta untuk Android, Final Project (Yogyakarta: IST AKPRIND)

[20] Ardianto J Lestari U and Triyono J 2016 Sistem Informasi Geografis Pemetaan Area Persawahan Desa Gantung Kabupaten Belitung Timur Script 3

[21] Sutanta E 2010 Basis Data Dalam Tinjauan Konseptual (Yogyakarta: Andi)

[22] Silberschatz A Korth H F and Sudarshan S 2011 Database System Concepts (6th ed.) (New York: The McGraw-Hill Companies, Inc.)

[23] Nurnawati E K and Ermawati 2018 Design of Integrated Database on Mobile Information System: A Study of Yogyakarta Smart City App IOP Conf. Series: Materials Science and Engineering 306 1-11

[24] Sutanta E and Kumalasanti R A 2017 Desain Database yang Skalabel untuk Berbagai Aplikasi Berbasis GIS, Proceeding of Seminar Nasional Science dan Teknologi Indonesia (SENSEI) (Jember: Universitas Muhammadiyah Jember) 1-9

[25] Sutanta E Wardoyo R Mustofa K and Winarko E 2016 Survey: Models and Prototypes of Schema Matching International Journal of Electrical & Computer Engineering (IJECE) 6 1011

[26] Sutanta E Wardoyo R Mustofa K and Winarko E 2016 A Hybrid Model Schema Matching Using Constraint-Based and Instance-Based International Journal of Electrical and Computer Engineering (IJECE) 6 1048

[27] Elmasri R and Navathe S B 2011 Fundamentas of Database Systems (6th ed.) (Massachusetts: Addison-Wesley)