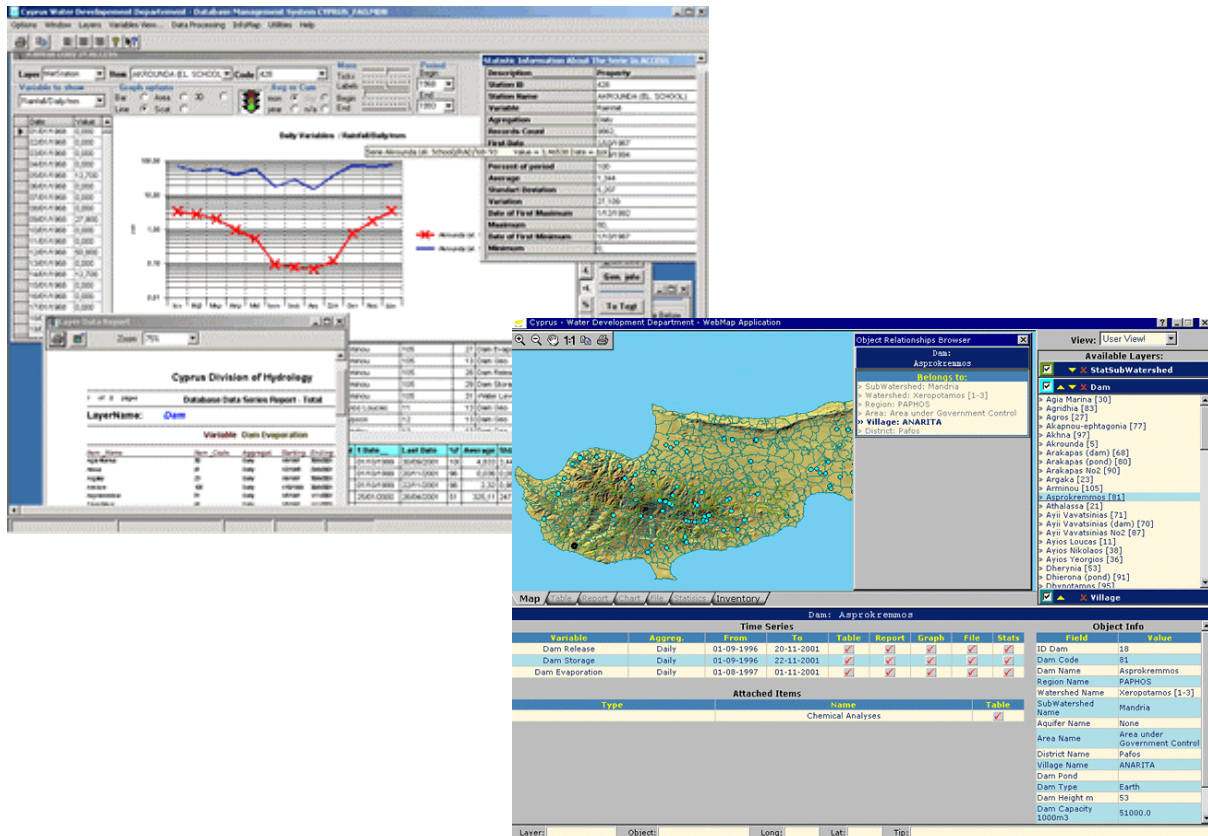


Ministry of Agriculture, Natural Resources and Environment of the Republic of Cyprus
Water Development Department

Food and Agriculture Organisation of the United Nations
Land and Water Development Division

TCP/CYP/8921

WATER RESOURCES DATA MANAGEMENT SYSTEM



Dejan Lekić, Eram Artinyan
FAO - International consultants

December 2001

Abbreviations

Cdpack – CD package, provides semi-automatic re-creation of ICD.

DBIB - Database Inventory Browser, providing low-level access to the database objects, displays general database inventories.

DBMS - Database Manager System, database application for database maintenance, management and administration.

DHTML – Dynamic HTML, combination of client and server side scripts making a dynamic user-friendly web interface.

DOH – Division of Hydrology.

DOM – Document Object Model, description of the HTML document structure.

FAODB – FAO Project Database.

GIS – Geographical Informational System.

HTML – “Hypertext Mark-up Language”, used for Web page creation.

HTTP – Hypertext Transfer Protocol, Internet protocol for ASCII data transfer.

ICD – Installation CD, provides automatic installation of entire system on one computer for testing and presentation purposes.

IE – Internet Explorer, Web Browser developed by Microsoft.

IFRAME – HTML tag, provides page embedding.

IS – Informational System.

LAN – Local Area Network.

MB – Mega Bytes.

MI – MapInfo, GIS desktop application developed by MapInfo.

MIF&MID – Pair of ASCII files with MIF (MapInfo Interchange Format) and MID extensions used as an intermediate step when passing geographical data from MI to the FAODB.

ODBC – Stands for “open database connectivity”, set of programs (drivers) providing access to different types of databases.

PHP - Stands for "PHP: Hypertext Preprocessor", a widely used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.

RAM – Random Access Memory, computer memory storage.

SQL – Structured Query Language, language for making database queries.

SVG - Scalable Vector Graphics, vector graphics description language, rendering engine has to be installed as a browser plug-in.

UTM – Universal Transverse Mercator, geographical projection.

VB – Visual Basic, programming language.

VML - Vector Mark-up Language, vector graphics description language, rendering engine preinstalled as part of IE.

WAAP - Web Access Administrator Panel, interface for creating, editing and deleting users of WMA, provides user-level security for database access.

WDA - Web Database Application, client-server application providing remote access to the database using Internet technologies.

WDD – Water Development Department.

WMA - WebMap Application, main database Web application, provides full database access using GIS-like interface.

WRDMS – Water Resources Data Management System.

WWW – World Wide Web, Internet network.

CONTENTS

1. Introduction	3
2. Objective of the mission	4
3. Description of activities	6
3.1. Analysis of available data, information, knowledge, software and hardware, requested user policy	6
3.2. Logical definition of Information System	8
3.3. Installation of chosen software components	8
3.4. Information System design and creation	9
3.4.1. Logical Database Structure	10
3.4.2. Physical Database Structure	10
3.4.3. Database Manager Application design and development	12
3.4.4. Web Database Application (WDA) design and creation	14
3.4.5. Data preparation	15
3.5. IS testing and debugging	19
3.6. Training workshops for IS administrators and for end users	19
4. Results	20
4.1. The Database (E.Artinyan; D.Lekic)	20
4.2. The WWW site (D.Lekic)	21
4.3. Database Manager Application (E.Artinyan)	22
4.4. Web Database Application (D.Lekic)	23
4.5. Installation CD (ICD) design and creation (D.Lekic)	24
4.6. CDpack creation (D.Lekic)	25
5. Conclusion and recommendations	26
6. Programme of the mission and people met	27
Annexes	28

1. Introduction

The mission main objective “Set-up of Water resources data management system” was initially divided in two parts and two international consultants were engaged to accomplish the objectives proposed by FAO. Dejan Lekic was specially engaged for the creation of some components of Information System (IS) such are WWW Site, Web Database Application and some other modules. The main objective of Eram Artinyan was to build the Database Logical and Physical Structure and to design the Database Manager Application as main tool for Database maintenance and feeding. This report will try to give a description of each of those particular IS components, mentioning each time the author or authors of the particular part of it.

It is important to emphasise that the entire system, developed in a collaboration of both consultants, consists of numerous parts that cannot be easily split in a sense of used technologies and system components, meaning that some parts have been built in a very close collaboration between the authors.

2. Objective of the mission

The main goal of the mission was to define and create a Water Resources Data Management system (WRDMS) as well as to train employees of the Water Development Department for future use of the established system.

WRDMS itself, can be defined as an Information System that will provide a centralised data storage designed for multi-user, network (client-server) oriented environment, updateable with incoming data either in form of new time series or in a form of new geographical data (layers or objects within existing layers). The designed system has to be extensible and flexible, providing the infrastructure for further development and improvement. The design of WRDMS can be also viewed as a process of information/data warehousing, reorganising and storing in a database system that will afford fast and accurate database access for the users in Water Development Department (WDD) as well as to possible remote users in other departments of the Ministry of Agriculture, Natural Resources and Environment of the Republic of Cyprus. We could group the needed characteristics of the WRDMS as follows:

- It should incorporate the geographical data available at the Division of Hydrology as MapInfo outputs: hydrological regions boundaries, watershed boundaries, river vector presentation and any kind of map containing point objects (hydrological and rainfall stations etc.).
- It should include any kind of data that could be attached to those geographical objects: data series, flat tables, documents, images etc.
- Easy understandable system, low cost system maintenance and low upgrade price have to be provided.
- User-friendly Database Manager program has to be created.
- Ensure high availability and robustness as well for local network usage (intranet) and in future in not very loaded Internet usage.
- Provide good capabilities for export of data through the Web Database Interface and good import and editing tools in the Database Manager application.
- Provide support for updating of geographical data, as importing (adding) new geographical objects not used at the design level. For example, pipeline networks, irrigated areas, roads etc.

Set-up and Development of a World Wide Web (WWW) Intranet/Internet Site has been defined as a part of Mr Lekic mission objective mainly in a sense of a main database interface for all the users of the WRDMS.

Advanced use of several existing software packages, such are MapInfo, MS Excel, MS Access and others as well as WRDMS creation using numerous programming languages led to the definition of the last mission objective – training of personnel and technology/knowledge transfer to the employees of WDD.

During the mission, employees of WDD defined additional objectives. One of them was the creation of an Installation CD (ICD), that will contain a complete set-up of the entire IS, meaning the Database, DBMS, WDA and all the custom software, script engines and documentation included. That additional task imposed implementation of some major changes in the IS philosophy since the system design was based on the idea of the typical client-server architecture. In order to accomplish this additional task, Mr. Lekic created a special software

package, providing a possibility of installing the entire system as a standalone application on a computer that doesn't have a direct access to the main Database/Web Server. Moreover, following the idea that in the future, an additional CD has to be created containing an updated version of the database, another task has been defined: CD package (CDpack) that will provide a semi-automatic recreation of Installation CD, with the updated version of the database.

Another persistent issue was the very different format of the supplied data for database input. This led Mr Artinyan to develop additional routines in order to import data faster.

Mission objectives were accomplished in close cooperation between the two consultants: Mr. Eram Artinyan took responsibility of creating a logical and physical Database structure, import/export utilities and the Database Management System Software (DBMS), while Mr. Dejan Lekic was responsible for the logical database structure (GIS information storing), GIS data preparation and import/export component, ICD, CDpack and Web Database Application (WDA) representing the main database interface and at the same time the main result of this mission.

3. Description of activities

Establishing of WRDMS, as a very complex technological solution, requires several types of activities. Each of them has to provide an intermediate result used as input to the next phase. Generally, the activities can be divided in five parts/tasks:

- Analysis of available data, information, knowledge, software and hardware (duration 8 days);
- Logical definition of Information System (5 days);
- Installation of chosen software components (2 days);
- Information System design and development (35 days);
- Data preparation and import (20 days);
- IS testing and debugging (10 days);
- Training workshops for IS administrators and for end users (5 days).

Some of the activities were simultaneous; therefore, the total amount of days goes over the mission timeframe.

3.1. Analysis of available data, information, knowledge, software and hardware, requested user policy.

This task is required as a prerequisite for logical definition of IS in a sense of indexing all the existing data and information in various forms. Analysis of software packages (MapInfo, MS Access, etc.) already extensively used by the staff of WDD provided an input in a process of making an appropriate configuration of IS that will use the experience and knowledge of WDD personnel. Collection of information about hardware usage as well as about operational system installed and pre-selected software components on the computers in the local area network of WDD (LAN WDD) was also needed as a set of constraints that resulted in selecting suitable software components in IS development (Apache Web server, PHP, etc.). Generally, this task can be described as:

- Analyses of available data – time series:
 - Variable types (rainfall, stream flow discharge, etc.);
 - Aggregation levels (daily, hourly, monthly, etc.);
 - File format of stored data (ASCII files, Excel, etc.).
 - Structure quality of the data (well structured with date-time stamp, poorly structured etc.).
 - Data volume (Mb)
- Analyses of available data – GIS data:
 - Layer types;
 - Available Object definition tables;
 - Definition of relationships between Objects within layers.
- Analyses of available software and hardware:
 - Hardware configuration of computers used in the Department;
 - Hardware configuration of LAN WDD;
 - Hardware configuration of the main server;
 - Prevailing operating system installed;
 - Set of software packages already used by WDD staff.
- Analyses of current knowledge and responsibilities of the local staff

- How many and who are the employees that could work with relatively complex routines?
- What is the average and higher knowledge level of the Database System users?
- Who will be designated as system administrator?
- Analyses of the need of implementing user security policies
 - How many different levels of user access are needed?
 - What should be the differences between the user levels?
 - Who will apply the user security policy?

Information mentioned above was obtained through numerous discussions with employees of WDD during the first few weeks of the mission. During the first two weeks of the mission, analysis of software and hardware used in Division of Hydrology (DOH) was made. Several conclusions were made, used as set of predefined conditions for development of future system, implicating some constraints on the component selection. Established facts about existing hardware and used software are as follows:

- Operating System => Windows 2000 Professional (Windows 98);
- Internet Browser => Internet Explorer 5.0 and higher;
- Used Software => MS Access 2000, MS Excel 2000, MS Word 2000, MapInfo 5.0 and higher;
- Hardware => Pentium I, II, III and IV, equipped with wide range of RAM and hard drive capacities.

The result of other above-mentioned analyses could be grouped into few main topics:

- Data availability and quality:
 - Very good geographical data support, achieved with MapInfo software usage. This includes Watersheds, Rivers, Administrative limits, Rainfall and Hydrologic station coordinates etc.
 - Relatively small volume of data series, up to 50 MB, excluding the rainfall daily series that are coming from the Meteorological Department.
 - Full set of data about water needs, supply and water use by village, crops and livestock within the villages.
 - Chemical data maintained as Access application, without established link between station identification numbers used in DOH and those for chemical analyses.
 - Long, uninterrupted and well-formatted series available about Streamflow Discharge, Rainfall monthly and daily, Borehole Water Level. Bad formatted (missing evident date stamp in the file) data about Dam Storage, Release and Evaporation.
- Software, hardware and network capabilities
 - Very good software availability of Microsoft products: Windows 2000 and MSOffice 2000. The local personnel widely use Microsoft products, have good knowledge and understanding of PC technologies, but there are no high level specialists to maintain complex database solutions like SQL-Server or Oracle.
 - Very good availability and knowledge of MapInfo software.
 - Fast and reliable local network available, powered by a fast network server. Network server, mainly used as file server, with backup maintenance facilities and could be used to run Web Server applications.
- Need of Database User policy

- The local staff and the National Project Director maintained the decision to have no more than two levels of user access with “read-write” rights to the database file, for people using the Database Manager application. For users of the Web Database Interface separate user access administration could be maintained, having in mind that the implemented technology for Web access is “read-only”.

Those results brought us to the conclusion that (as it was previously suggested in the “Terms of reference”) the most appropriate format of the database for chosen level of information system development is Access (Jet DBMS). The database file should be created in a shared folder on a network server running Apache Web Server, accessing the database through ODBC calls. The common user should access the data from the database through Microsoft Internet Explorer, like accessing any Web site on the Internet, but the system administrators will use the Database Manager application to import data into the database, edit and update the objects and data. This solution is the cheapest one since it doesn’t require purchase of any additional software. It’s also the most robust as it uses “read-only” connection for the common user and only administrators have the right to write into the database. This solution could be used from up to 250 users through Internet Explorer as part of truly “client-server” system, but only few persons should receive the database password of the Administrator account and use it through the Database Manager application, which connects to the database file in the “file-server” environment.

It was also found that centralised database doesn't exist and that the usage of the local network is basically reduced to the file and folder sharing. Main use of Local Area Network Server is for backup and shared file storage purposes using data tape backup system installed. Local Network is connected only with the other Division of WDD located in the same building. Permanent Internet connection is not available. Instead, dial-up Internet connection can be made only by using one of two computers equipped with 56 Kbps modems.

3.2. Logical definition of Information System

Resulting from the analysis made in the previous step IS definition in a term of logical structure has been performed. Several components were defined, ranging from WRDMS importing utilities, through database management software, client-server database interface, ending with exporting utilities that closes the circle of data and information flow from WRDMS administrator to common user. Implemented technology (*Image 1.*) provides a wide set of possibilities for data storing, editing, retrieving, visualising and manipulating and it will be described latter in this document.

3.3. Installation of chosen software components

Web Server, Server Side scripting engine and linking with the database system has been done as a part of this particular activity. Apache Web Server (www.apache.org), leading Open Source Web server, version 1.3.22, has been installed on the server of Division of Hydrology, providing an Internet/Intranet Web Site for WDD. Another component, PHP (pre hypertext processor) engine, with precompiled ODBC module (www.php.net), one of the leading Server Side programming languages, version 4.0.4, has been also installed. In addition, as a component that is needed for connection between PHP and the database, the latest ODBC drivers were installed on the server. Installation of all these custom components, upgraded the server of the Division of Hydrology from the classical server used to support Local Area Network connections to the Web & Database server, keeping the previous functionalities at the same time.

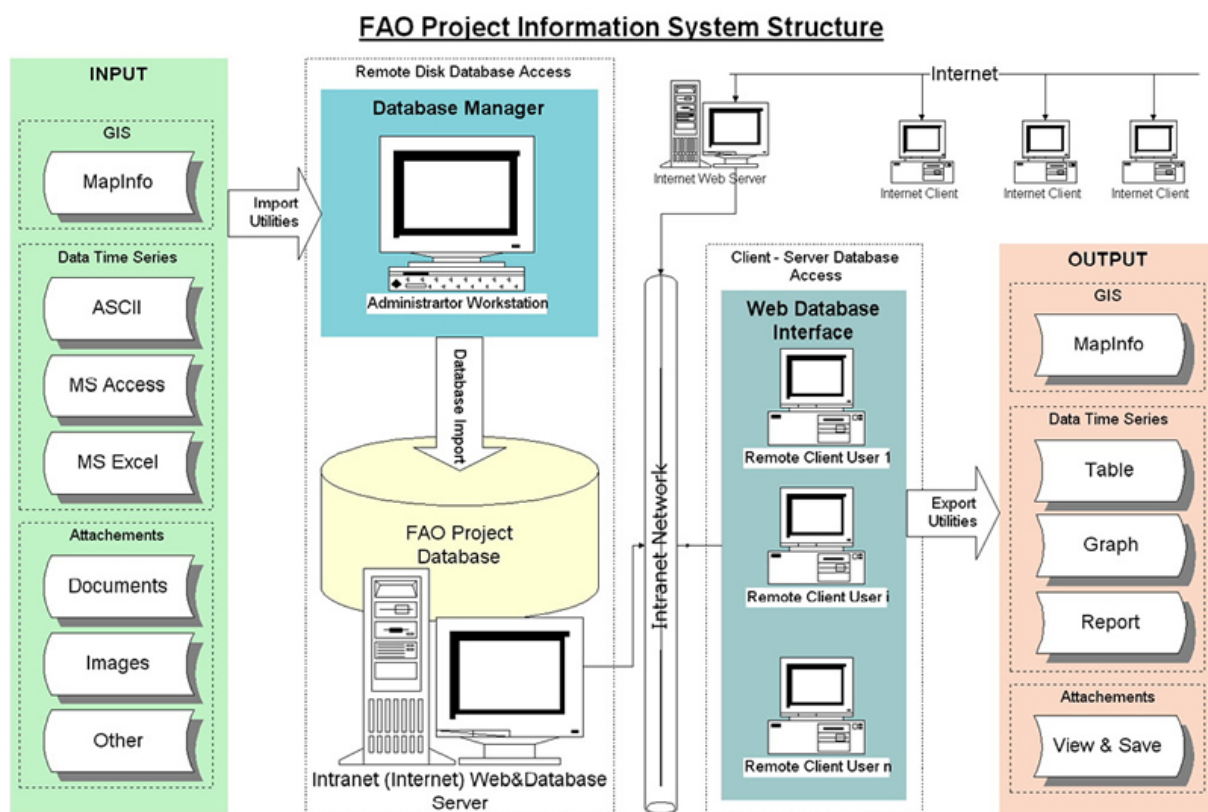


Image 1. Logical Structure of IS

3.4. Information System design and creation

Definition of the logical IS structure, led to the work on the physical structure of IS, precisely in the creation of software that will have the ability to perform the tasks defined in the previous step. The following list defines the tasks accomplished as well as the components created accompanied with consultant name in parenthesis:

- Time series preparation, formatting [E.Artinyan];
- Geographical data preparation [D.Lekic];
- MS Access Database logical and physical structure [E.Artinyan, D.Lekic];
- Database Manager program design and development, including the following topics:
 - Database Import/Export utilities for all the data (time series, attached items, etc.) except for the geographical data [E.Artinyan];
 - Database Import/Export utilities for geographical data [D.Lekic, E.Artinyan];
 - Database Management Software with administrator utilities [E.Artinyan];
- Web Database Application with remote user exporting utilities [D.Lekic];
- Installation CD, containing entire IS as a standalone installable product and CDpack, package for creating installation CD [D.Lekic].

Preparation of the data (first two tasks) was accomplished with valuable help and feedback of WDD staff. The development of IS components was done in the same hardware and software environment where the entire system will be used in the future, therefore the entire system was immediately available to the users at the moment of the final version production.

Schematic view on the system, used languages and technologies is given as Image 2.

FAO Project Information System Technology Overview

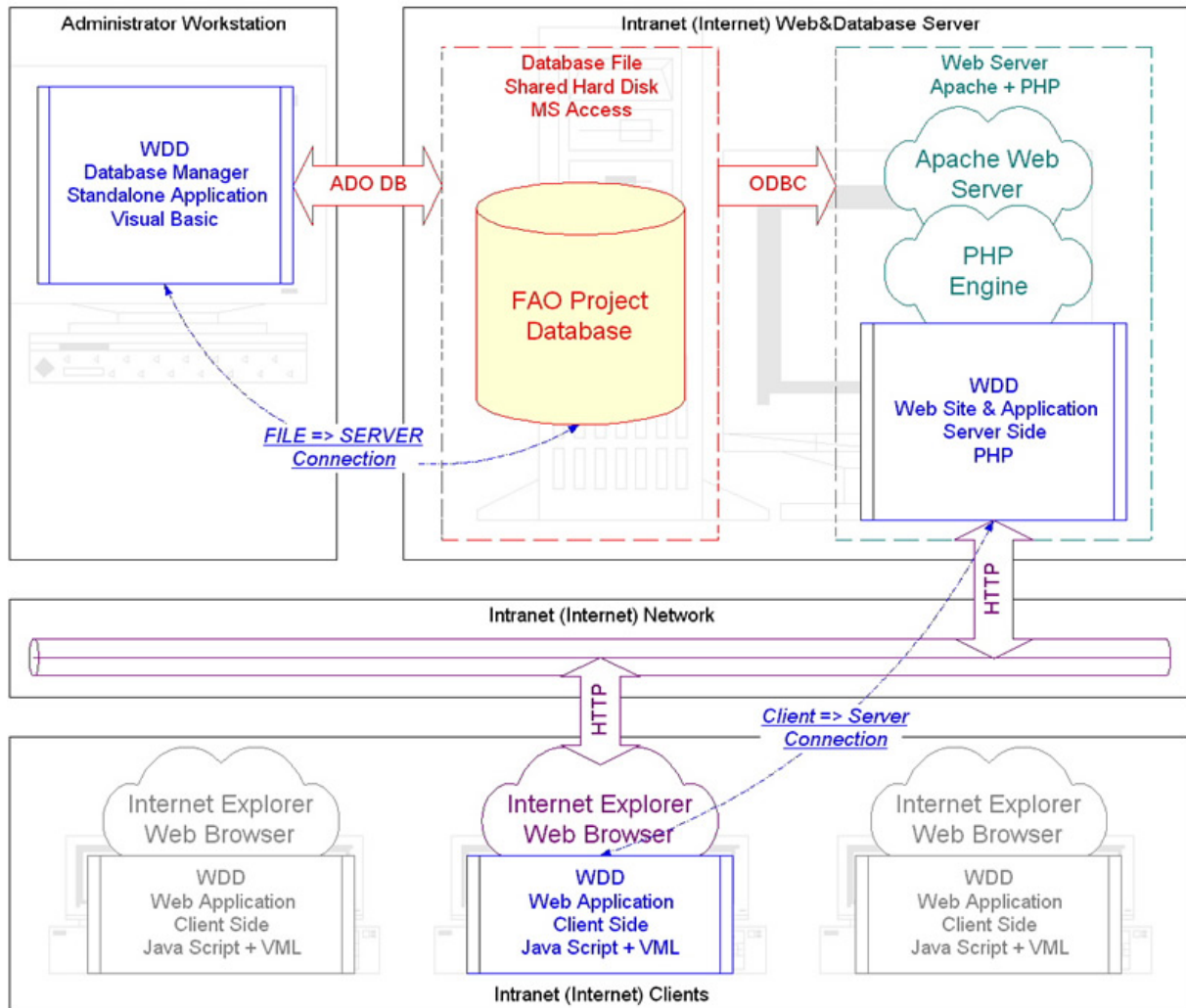


Image 2. Technologies and languages used in IS development

3.4.1. Logical Database Structure

The logical database structure is composed by three distinctive parts:

- The geographical data (Spatial Data). In further references, we will use the term “Layer” for the entire subset of the same type (i.e.. Rivers Layer, Watershed Layer etc.) and the term “Item” for a single object within the subset.
- The data attached to the single “Item” is making the second part of the logical structure. This could be “Data Series”, any kind of flat tables containing data about single item –“Table” and any kind of documents attached to a single item – “Attachment”
- The Meta-data. That is describing the availability of data for each item within each layer inside the database.

3.4.2. Physical Database Structure

A physical database structure has been designed using written procedures within the Database Manager Application. All the tables, views and relations are built automatically and there is no need for any kind of manual operation on the database (using MS Access package). Physical data and geographical objects database storage is defined as follows:

- The Spatial Data is represented within the physical structure with the table [*Layers*] (*Image 3.*) and one additional table for each Layer containing the description fields for each Item in the Layer – [*Layer_Desc*]. The [*Layers*] table contains names of layer’s children and layer’s parents, the “Vector Mark-up Language” (VML) image of that Layer.

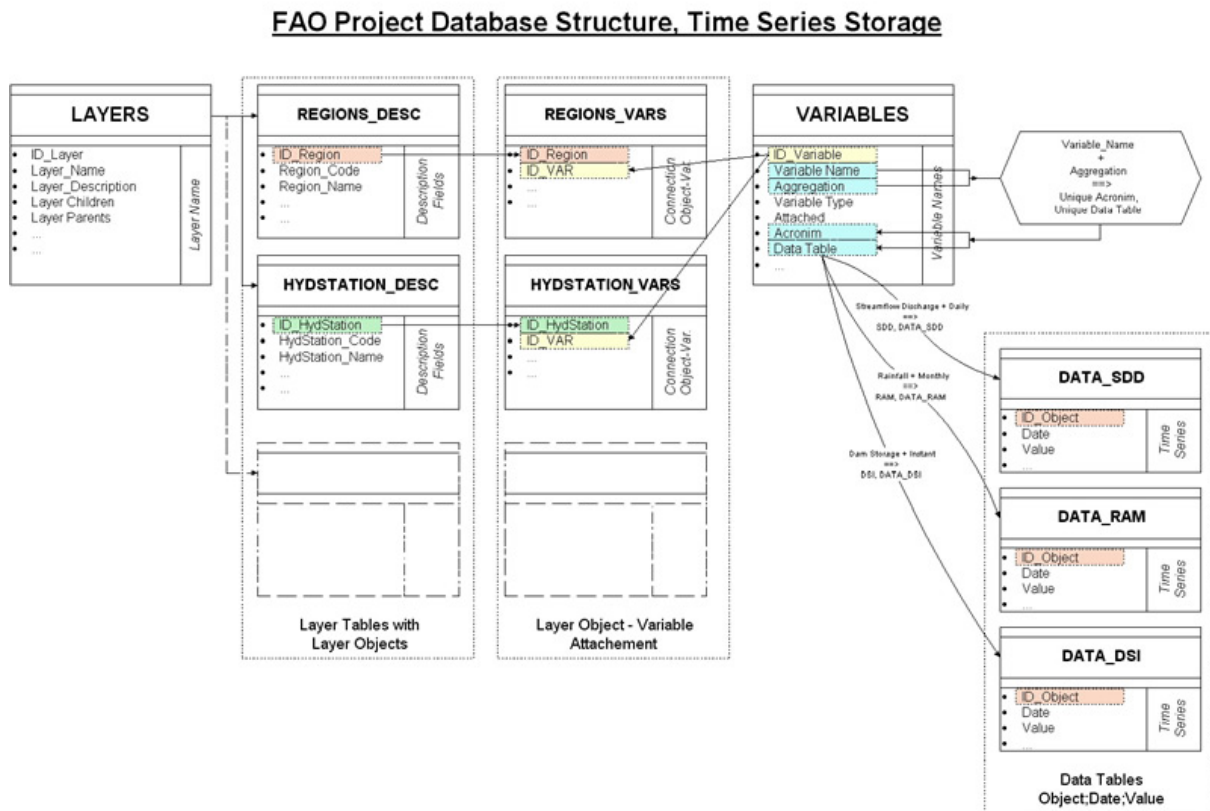


Image 3. Physical Database structure

- The Meta-Data is stored in [*Layer_Vars*] tables – one for each Layer and one common table [*VARIABLES*]. [*VARIABLES*] table contains the description of the variable (the variable name, type, aggregation level, to which a layer should be attached, column names and types of the linked table, name of the table containing that variable). Different variables can be attached to a single Layer using database relation resulting in the fact that removing a Layer produces cascade removing off the connection to the attached variables.
- The Data Storage is maintained from a structure of tables usually with names [*DATA_XXX*], where *XXX* is the acronym of the variable contained in that table. Those tables have direct link with the [*Layer_Desc*] tables through their first column – the *id_object*. That means that data could be added only for existing Layer Items. More than that, if for some reasons one or more Layer Items or an entire Layer is removed, automatic removal of all the data attached to this Layer Item or to the entire Layer is invoked.
- Additionally several “views” and “procedures” are created in order to show the linked tables in one single table (or grid). By instance, such views are the Layer Inventory views that are showing together all the possible combinations of linked geographical

objects (Layers). Other example is the Data Inventory view that shows the availability of attached data to each item of the Layer.

3.4.3. Database Manager Application design and development

3.4.3.1. Geographical Data Import/Export tool

Initially invoked tool, when user creates an empty database structure. As any data in the database must be attached to a geographical object, the Layer object should exist in the database before the actual Data Import. The first job is to design (on a separate sheet) the Layer Objects Relationships Scheme (see Image 3). Then, the administrator has to import all the Layers in the database, starting with the topmost parent Layer.

Therefore, the tool was designed with four basic guidelines:

- To maintain bi-directional compatibility with MapInfo export/import format.
- To provide high robustness and checking level against possible user errors.
- To provide import, but also update support for geographical layer information in the database.
- To create the vector mark-up language (VML) image of the imported layer immediately after the import procedure finished.

In order to store the geographical objects in the database such MS Access, there has to be a transitional step that will translate the object definition from GIS format to the tabular representation of the same data. For that purpose, a special module was developed, first using PHP language, the latter translated to Visual Basic and incorporated as a part of Mr. Artinyan's Database Management Application. The basic idea is to convert two types of information available in GIS software, object coordinates and object attributes to some kind of format that will be suitable for storage in tabular form. More than that, since the WDA (Web Database Application) displays vector representation of objects within Web Browser using VML (Vector Mark-up Language, standard for displaying vector objects on the Web proposed by Microsoft), two ways of geographical objects database storing were created. First one, using MapInfo interchange format (MIF, producing two files for each layer, extension .mif – containing coordinates and extension .mid – containing tabular data), was designed to use files exported from MapInfo, parse those files, extract all the available information from them and store the data in prepared database tables. Second way of storing was made to store geographical coordinates of objects within one layer as exactly one database field. That particular field contains VML representation of all the objects that belong to one layer. The main reason for this “double” storing is to increase the speed of vector object creation for remote user that access the database through Web database interface. Another interesting feature of this module has to be mentioned. While the object coordinates are stored in database tables in their original format, UTM or Latitude/Longitude (providing export from the database in the same projection), geographical coordinates, stored in VML fields, are always recalculated to Latitude/Longitude system, since that is the only one that can be used for geographical object rendering within Web Browser. This means that the module itself contains a subroutine for geographical translation from UTM to Lat/Long projection.

3.4.3.2. Generic Data Import/Export tool

This part of the Data Manager program (E. Artinyan) is providing full range of capabilities for import and export of any kind of data attached to the geographical objects in the database (Image 4.). As mentioned in the “Terms of reference”, the tool provides the capability of data export with respect to the geographic coordinates of the station (the user could export the coordinates together with the data records) or the code of the geographic item when it is a polygon type object (i.e. Village, Region or Watershed).

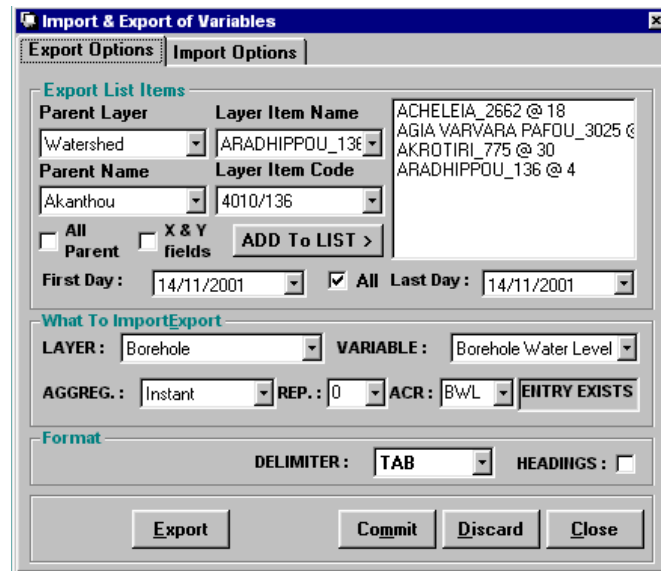


Image 4. Generic import/export tool

3.4.3.3. Administrator Utilities

Part of the Data Manager program dealing with Layer Description tables, Meta-Data editing, Variables management and user policy.

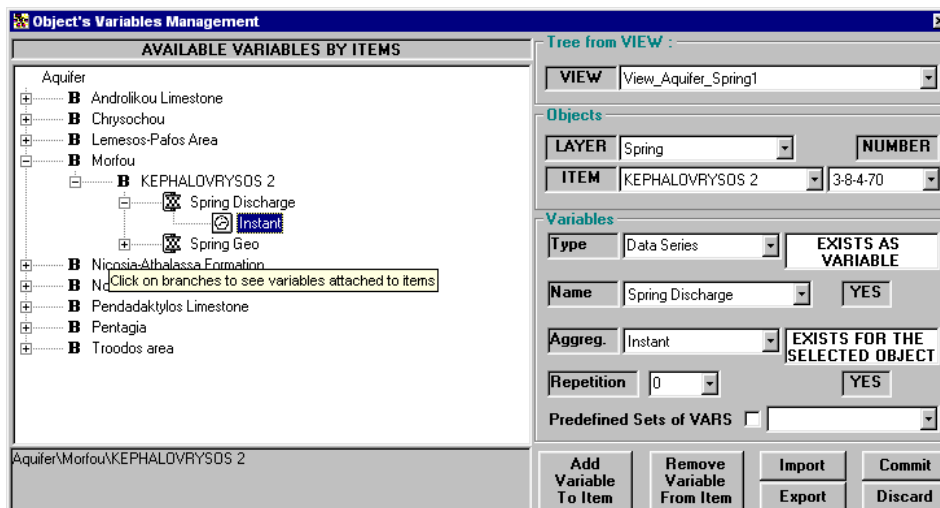


Image 5. Object's variables management window

- Editing Geographical data already included in the database is providing high flexibility, but in the same time is maintaining the data integrity. If some item name is changed in the parent layer description table, the change is propagated wherever that name is used in a children layer description tables. For instance, when the user

changes the name of an Aquifer in the Aquifers description table all the boreholes having this Aquifer name in their description are receiving the same Aquifer name.

- Editing the Meta-Data is permitting to maintain tight link between data and data holders in the database. Without declaring variables for a particular item, the user couldn't add data for this item. In the particular case when a layer item is deleted the data attached to it are deleted also.

User policy is providing two levels of access:

- Administrator (Owner) access level, with rights to build a new database, to add and edit geographical Layers as well as Meta-Data editing rights.
- Normal user, which should only add and edit non-geographical data as data series etc.

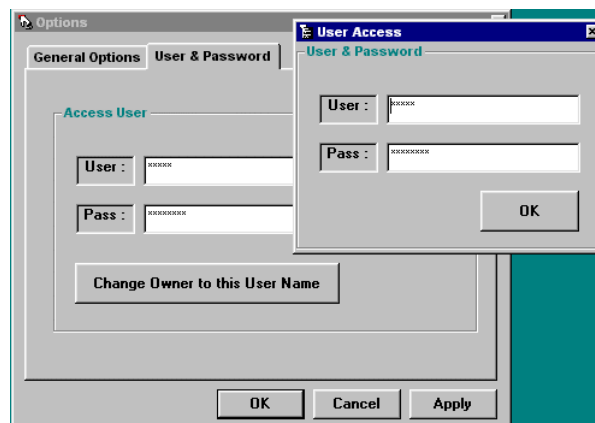


Image 6. User log-on window

3.4.4. Web Database Application (WDA) design and creation

Defined as an Internet, Dynamic HTML application that will be used for common user database access, the WDA is based on the idea that the map (GIS) interface represents the simplest way of raw data access. Led by this fact, a user-friendly interface has been built, providing data visualisation as a result of straightforward user action over the database items displayed as objects on the map. Operational, reacting on both left and right mouse buttons, WDA provides execution and visualisation of a wide variety of automatic database queries, database tables and views.

The technology chosen for WDA creation is well known client-server architecture, using the so-called "thin" client, since it uses Internet Browser as the engine for running on the client side. For that reason, there is no need for any intervention on the client side. The user has just to point his browser to the Intranet/Internet Web Site of the DOH. Programming languages selection came as a logical result of the concept that the newly created Information System has to be as cheap as possible, maintaining its operability and robustness at the same time. On the client side, JavaScript was preferred over VBScript since it provides better opportunity for further development of the same application for the Netscape browsers family. Between two currently available languages for vector image visualisation within Internet Browser, VML and SVG, VML was chosen because of a fact that it doesn't need any additional browser plug-in in order to be used. On the server side, Apache Web Server as very reliable software was selected, powered with PHP language as an engine for running server side scripts, providing dynamical HTML pages creation in a response to user driven events. A special technique that includes usage of particular HTML object, IFRAME, was pre-selected as the

way of making the WDA interactive, avoiding the necessity of refreshing the entire WDA interface every time the user interacts with it.

Following the request about user level access, mentioned earlier, an administrator panel was projected as a way of creating, editing and deleting of users that will access the database through WDA.

Resulting from the need of another level of database access that will allow the simplest possible view on the database inventory, another component of WDA was conceived – a Database Inventory Browser module that will act as a dynamic database catalogue creator (*Image 7*).



Image 7. Database Inventory Browser, buttons for catalogue creation

3.4.5. Data preparation

3.4.5.1 Geographical data preparation (D. Lekic)

This activities required MapInfo Layer preparation, redefinition of Layer Object description tables in MapInfo, redefinition of internal Object code for each object within one layer, unification of layer geographic projection etc. All these steps required extensive use of several software packages such is MapInfo, MS Access, MS Excel, etc.

Creation of the list of available data sets either in a sense of data series or geographical data information led to the necessity of unifying the format and the structure of these data. Geographical data were provided as a set of layers, stored in separate files, MapInfo format. There were several geographical projections used for GIS representation: classical Latitude/Longitude system, Universal Transverse Mercator (UTM) system (zone 36, ED=50) and custom UTM system for Cyprus. Unification of the geographical projection has been done and all the maps were converted to the same, prevailing, UTM (zone 36), projection.

Redefinition of the columns describing each object within the particular layer was also applied, as the implementation of the rule that each object has to be defined by unique *Object Code* and unique *Object Name*.

During the mission, several layers were redefined and added in the list of available layers so the final list of layers stored in the database contains sixteen layers containing various numbers of objects. These layers followed with layer types and descriptions are:

- Country (Polygon - contour of the island, Master layer containing all other layers);
- Aquifer (Polygon - contours of major aquifers on Cyprus, defined by Name and ID);
- Region (Polygon - contours of defined Hydrological regions representing predefined sets of watersheds);
- Watershed (Polygon - contours of main catchment areas of Cyprus);
- SubWatershed (Polygon - contours of arbitrary defined zones of each watershed);
- River (Polyline - contours of streams, only the rivers that have measuring station were considered);

- Area (Polygon - Cyprus specific feature, defining area under governmental control, under Turkish occupation, etc.)
- Borehole (Point - position of the underground water measurement point);
- HydStation (Point - position of measurement point on the river, weir);
- StatSubWatershed (Polygon – contour of the sub watershed attached to the particular Hydstation, catchment of the HydStation);
- Dam (Point - position of the dam on the river or the pond, water collecting hydrological construction);
- Spring (Point - position of the spring);
- MetStation (Point - position of the measurement point for meteorological parameters such as rainfall, temperature, humidity, etc.);
- District (Polygon - contour of political, administrative item);
- Village (Polygon - contour of political, administrative item, several villages make a district);
- Village Centre (Point – position of administrative centre for each village).

MS Access database system, as one of the simplest database environments currently available on the market, doesn't provide a way for storing geographical objects within the database. Knowing that, special organisational scheme for translating geographical object relations to the relations between MS Access tables has been established. As a precondition for this task, a layer objects relationship scheme (Image 8.) was created.

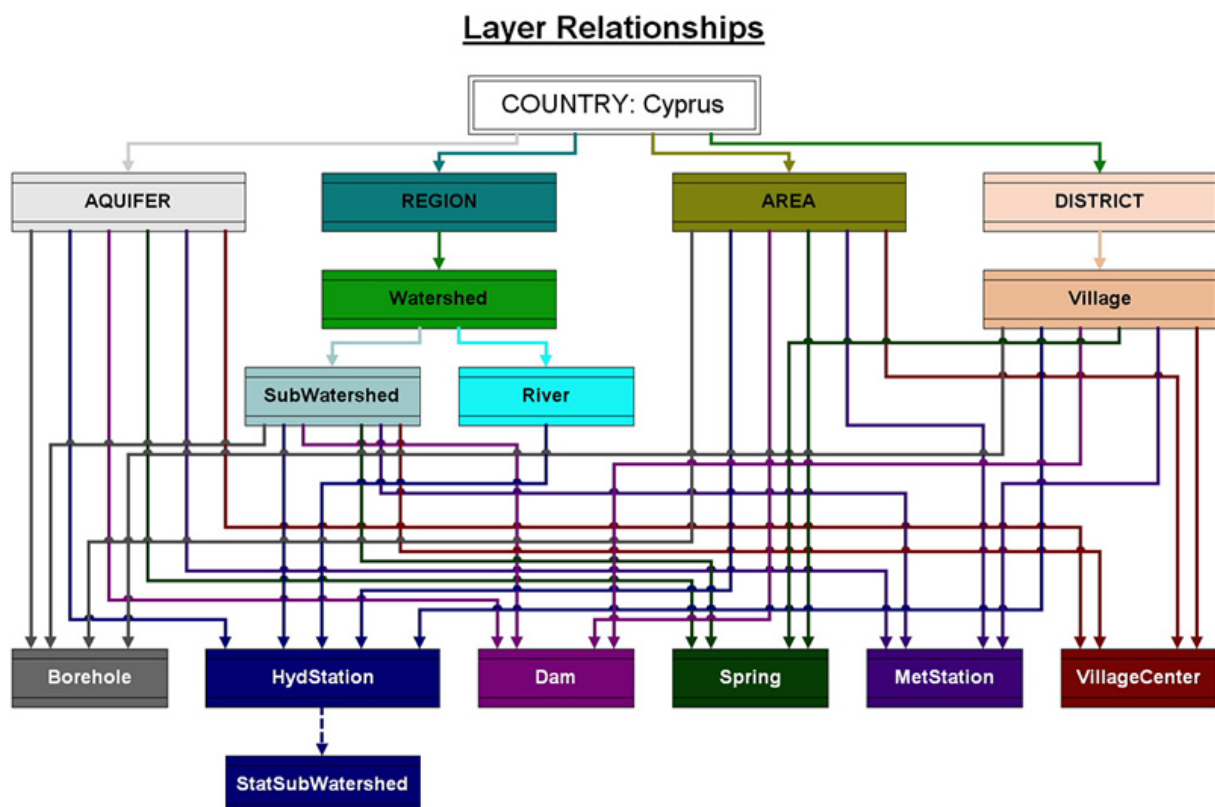


Image 8. Layer Objects relationships Scheme

It can be easily understood that the coloured lines connecting rectangles (Layers) on the image represent the relationships between objects belonging to connected layers. For example, an object in the Layer *Spring* belongs to four upper objects in four different “parent”

layers: *Aquifer*, *SubWatershed*, *Area* and *Village*. Since some of mentioned parents have their own parents, this connection is extended through *Watershed* to *Region* layer (from *SubWatershed* layer) as well as to *District* layer (from *Village* layer). All the relations are going up to the master Layer, in this case *Country Cyprus* (master layer represents a closed contour containing all the objects in all the layers). Following the presented idea of making a system that will keep geographical relations, the need of adding some extra fields in each MapInfo layer table definition was required. Each of the tables defining objects within one layer had to have columns that will provide explicit linking between that object and the entire set of parent objects within parent layers. As it is explained in the example above, the table containing layer *Dam*, had to be extended with fields containing names of all the parents of selected Layer: *Aquifer*, *SubWatershed*, *Watershed*, *Region*, *Area*, *Village* and *District*. This extension of columns describing objects within layers has been applied to fifteen layers mentioned above, excluding *Country* layer (it doesn't have parents). After remaking the layer table structure, newly added fields were filled using the built in feature of MapInfo that provides automatic calculation of *belongs/contains* geometric relationships.

Geographical data organised by the way explained above were ready for importing in the database. As shown (*Image 9.*), the database table that keeps relations between layers has a very simple layout.

	ID_LAYER	LAYER_TYPE	LAYER_NAME	CHILDREN	PARENTS	NESTED
+	1	Polygon	Country	Region;Aquifer;Area;District		
▶	2	Polygon	Aquifer	Borehole;HydStation;Dam;Spring;MetStation;VillageCenter	Country	
+	3	Polygon	Area	Borehole;HydStation;Dam;Spring;MetStation;VillageCenter	Country	
+	4	Polygon	District	Village	Country	
+	5	Polygon	Village	Borehole;HydStation;Dam;Spring;MetStation;VillageCenter	District	
+	6	Polygon	Region	Watershed	Country	
+	7	Polygon	Watershed	SubWatershed;River	Region	
+	8	Polygon	SubWatershed	Borehole;HydStation;Dam;Spring;MetStation;VillageCenter	Watershed	
+	9	Line	River	HydStation	Watershed	
+	10	Point	VillageCenter		Village;Area;SubWatershed;Aquifer	

Image 9. Database table [LAYERS] describing layers relations

Two table fields are used as key fields for storing relations: *Children* and *Parents* fields. The fields containing a list of children/parent layers are separated with semicolon. These fields are used later in database queries giving a view on geometrical relations between objects.

3.4.5.2. Data series preparation and import (E.Artinyan)

After the database main structure was built and geographical (Spatial Data) layers imported it was possible to import any kind of data into the database. Although some of the data were in appropriate format for any software import and export (i.e. Streamflow Discharge series and

Borehole Water level series), few types of variables were in too complicated excel sheets and were not possible to convert easily into the needed simple import format.

The import format for non-geographical data is as follows:

- One column for the id number of the layer item to which the data should be attached
 - This id is a string variable, so it could contain any alphanumeric characters.
- One column for the date-time stamp, in case it is a Data Series or Table Series variable type.
 - This column is of Date type, so it must have the format DD/MM/YYYY. The most important is to use four digits years in order to avoid 2k errors.
- One or many columns containing variable's data attached to the item from the first column and having the corresponding date-time stamp from the second column. They should be of the type defined in the moment of Variable's definition.
- Optionally a last column with the so-called repetition number (this is permitting to have several records with the same date-time stamp for the same variable and item id). Repetition number is used in cases when several quality sets exist for the same Data Series.

The columns could be separated with tabs, semi-columns, spaces or commas.

Therefore, all the data should be more or less pre-processed in order to reach the criteria mentioned above. The most complicated and difficult to convert was the data about Dam Storage, Dam Evaporation and Dam Release.

The following data were prepared and imported into the database:

- Data Series
 - Streamflow Discharge Daily
 - Spring Discharge Instant
 - Rainfall Monthly
 - Rainfall Daily
 - Evaporation Monthly
 - Borehole Water Level Instant
 - Dam Release Daily
 - Dam Storage Daily
 - Dam Evaporation Daily
- Table Data
 - Village Data (Population, Water Demand for different use, by crop type, livestock type, etc)
 - Dam Chemical Analyses
 - Spring Chemical Analyses
- Attached Documents¹
 - Aquifer documents

¹ In the database file is included not the document itself, but the information linking specific document to specific layer item. The document is copied to the server when the user is attaching it to the layer item using the Data Manager application.

3.5. IS testing and debugging

Having in mind the complexity of such kind of systems containing several dependant components working in the same time on the server and on the client computers, extensive debugging and testing was applied.

The Web Application (D. Lekic) has been tested on the following operation systems: Windows NT, Windows 98 and Windows 2000. It was also tested using several versions of Internet Explorer installed on the client. Bugs and errors found during application development were fixed and the application was operational at the end of the mission.

The Data Manager application (E.Artinyan) was installed and tested on Windows 2000 and Windows 98. Few bugs were discovered during installation on Windows 98 platform and these were corrected.

3.6. Training workshops for IS administrators and for end users

Usage of advanced technologies and the complexity of the system imposed the need of organising several presentations of the developed system. Several times, during application development, the IS and its components were presented internally to the staff of the Division of Hydrology (DOH). Feedback gathered during these presentations was used in the development and almost all of the suggestions were used as a source for new application features. During the last few days of the mission the training course was provided for several persons from DOH, giving them an overview on all the important features of both applications. At the end of the mission, presentation of the IS was also made on the WDD level, explaining the general features and possible usage of developed IS for a wide set of users and Divisions within WDD.

4. Results

Following the activities described in the previous topic, four main results were achieved. The first and fundamental one is the database itself. It is the “root” for all further developed tools, containing all the data: geographical (Spatial Data), data series and documents linked in a strong relational structure. The Web Site is giving full access to the database contents through a developed interactive mapping interface to anybody who has the right access permissions. The Data Manager Application is used to maintain the Database internal structure, to update and to edit geographical and time series data and serves for document uploading. Installation CD and CDpack creation can be viewed as a fourth, final set of results.

4.1. The Database (E.Artinyan; D.Lekic)

- Data Series

Data Series Name	Num. of records	Period	Num. of objects
Streamflow Discharge Daily	612000	1965-1999	85
Spring Discharge Instant	14000	1903-1997	61
Rainfall Monthly	85000	1916-2000	134
Rainfall Daily	115000	1916-1995	6
Evaporation Monthly	7500	1970-2001	34
Borehole Water Level Instant	8244	1951-2001	73
Dam Release Daily	31000	1996-2001	17
Dam Storage Daily	30000	1996-2001	17
Dam Evaporation Daily	30000	1996-2001	17

- Table Data

Table Data Name	Num. of records	Num. of objects
Village Data (Population, Water Demand for different use, by crop type, livestock type, etc)	405	405
Dam Chemical Analyses	25	9
Spring Chemical Analyses	17	5

- Attached Documents

Attachment Name	
Aquifer documents	6 rows

- Layer Relationship Views (*Image 10.*) – 50 views automatically created

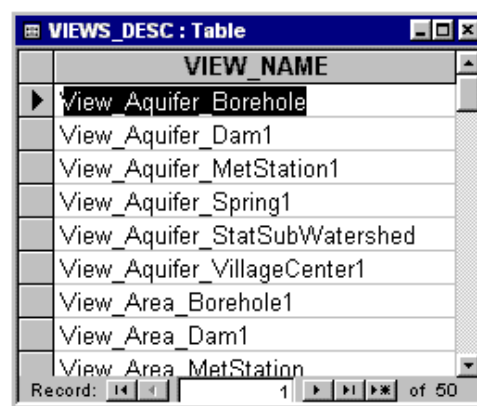


Image 10. Database table containing the list of available views

- 16 Layers, Layer Description Tables and Data Inventory Views
- Meta-Data description Tables: [*Variables*] and 18 [*Layer_Var*] tables
- 33 Variables of 4 main types: Geo, Data Series, Table and Tables Series
- 33 Data tables for the storage of each of these variables
- 157 Automatically created relationships that maintain the internal referential integrity between the layers having parent/children relationship and between layers and the variable's data attached to them.
- Unlimited number of layers² (different types of spatial data) in the database
- Upper limit of 140 variables of any type attached to one layer of point type.
- Upper limit of 20 variables of any type attached to one layer of polygon or line type.

4.2. The WWW site (D.Lekic)

Basic World Wide Web site was created, currently providing access only for the users in Local Area Network (*Image 11.*).

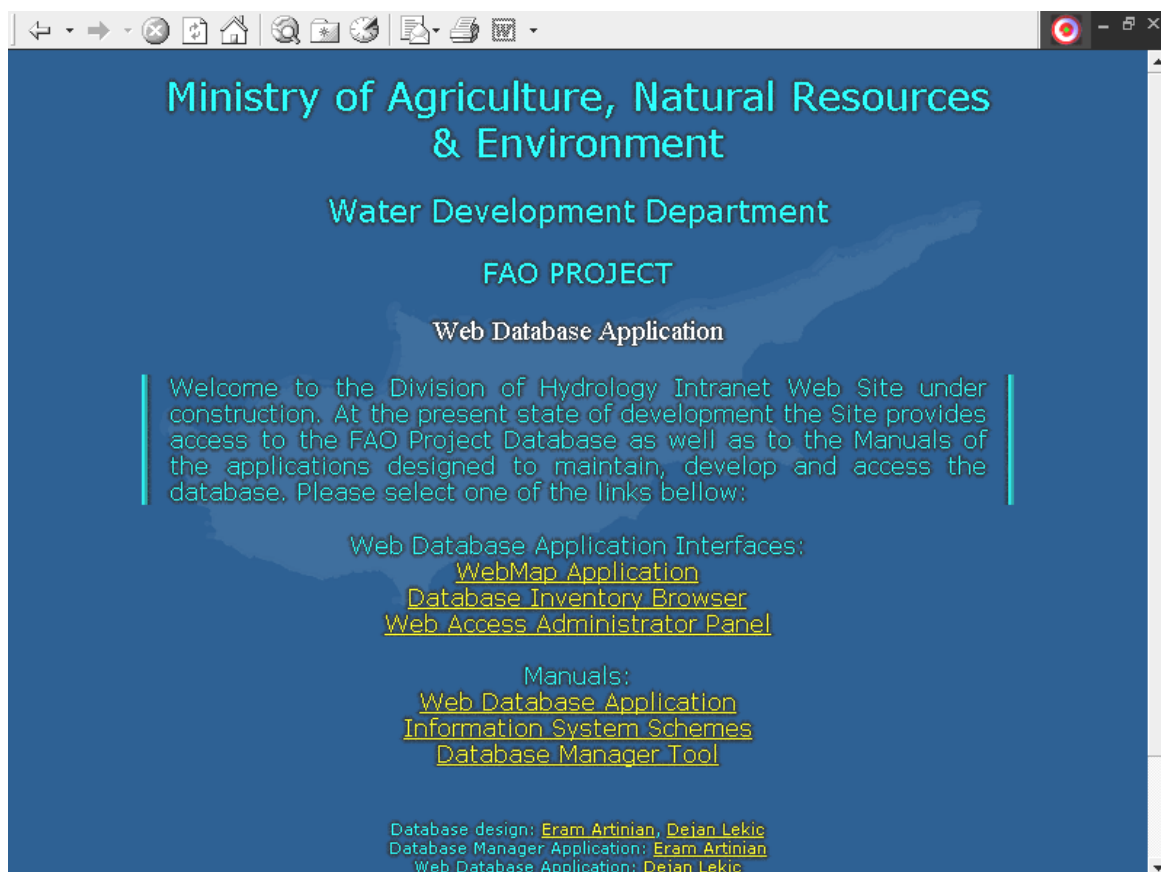


Image 11. Home Page of WDD Web Site

In the future, following the WDD computer network expansion, the Web site can be automatically promoted to the Division Intranet/Internet Web site without the need of any major changes in server or network configuration. The Web site currently provides users with links to different parts of WDA as well as to Application Manuals and four Images displaying IS schemes, facilitating understanding of the technology applied.

² The limitations could appear as a result of hardware overloading or in case of exceeding allowed application execution time.

4.3. Database Manager Application (E.Artinyan)

The Database Manager application (*Image 12.*) requires up to 10 minutes installation before normal operation. It reaches the database through the network or on the local hard disk. As the database file is accessed directly, the network connection should be good enough to not experience wait time while importing or exporting of the data.

The export tool gives high flexibility to choose what to export and in which format. It includes X-Y coordinates and the item's code in the exported ASCII file. This is especially useful when the exported data will be used for modelling purpose or data series analyses.

The application permits creation and maintenance of the internal structure of the database. It allows import and export of geographical data as well as any kind of data attached to the spatial objects.

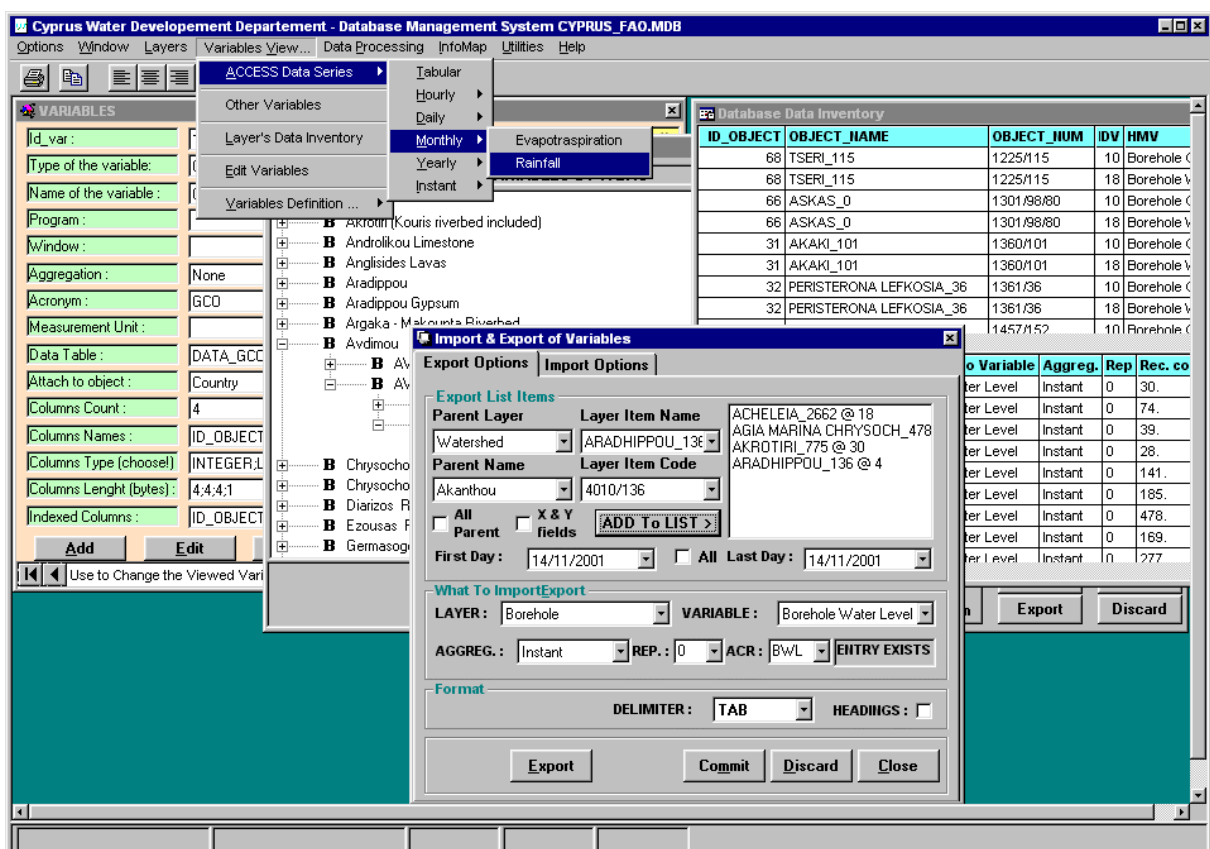


Image 12. Database Manager Application main interface

The views and inventory queries as well as the VML image of the layer are automatically created with the layer import. The application stores information about imported and exported data in special log files. User policy is simple but robust, corresponding to the level of development of the entire IS.

Complete application Manual attached as annex of this report will give the complete view on application features, description of capabilities and possible usage.

4.4. Web Database Application (D.Lekic)

Web Database Application (WDA) represents an application that provides remote user database access using “thin” client, meaning that it uses Web Browser as the application container. In this particular case, Web Browser has to be one of MS Internet Explorer browsers, version 5.0 or higher, usually installed with the operation system itself, meaning that there is no need for any kind of administrator intervention, software installation or change of configuration on the user machine, before he/she can access the database using WDA. Generally, WDA consists of three parts:

- Database Inventory Browser (DIB);
- Web Access Administrator Panel (WAAP) and
- Web Map Application (WMA).

All three parts were created using the same technology. On the server side PHP scripts reside, creating dynamical HTML pages as a response to user interaction. On the client side, JavaScript combined with DOM scripting (Document Object Model) was used to provide application interface interactivity with or without internal interaction with the server. This technology is usually known as DHTML (Dynamic HTML) powered with server side scripts.

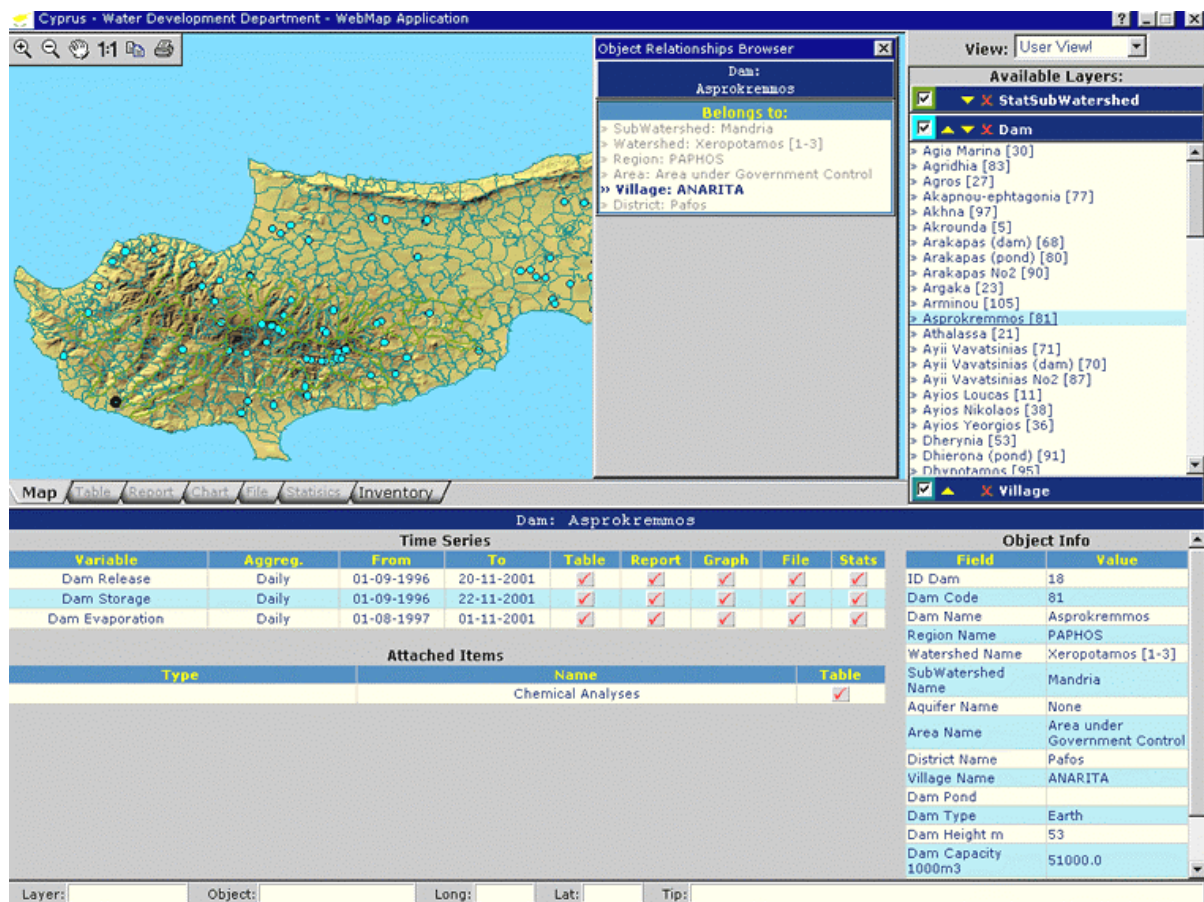


Image 13. WMA application interface

- The first part provides an overview on the objects stored in the database, variables + aggregations combination available, attachment of those variables to particular objects and finally interconnection between objects in different layers, as explained before.

All users can freely access this part of application, since the data themselves cannot be obtained.

- The second part, WAAP is used for creation, editing and maintaining of users that has to be provided with full database access using WMA. It consists of several panels that facilitate some administrative tasks; such are creation of a new user, deleting of existing user, and view on the user activity over the database, etc.
- The third part, the main database interface (*Image 13.*) provides a way of accessing hydrological (or any other kind) data attached to geographical objects through very well known mapping interface.

All the application features, description of capabilities and possible usage are well explained in the Annex of this report and it will not be specially given here. Only few basic principles will be explained.

The User, by making interaction with database interface (mouse click, mouse selection, etc.) moves through the set of panels with the final goal of accessing the data themselves. During application run time, the user is always provided with two ways of geographical object visualisation, on the map and in a form of list of objects. The data itself can be visualised in form of HTML table, Chart in VML format (*Image 14.*), MS Excel File, ASCII file, MS Word Report, etc. All of the mentioned forms of display support functions common to the Windows operating system such as copy, paste, print and other. The visualisation of the data shouldn't be seen as the only possible usage of this application, since the most powerful part of it is a module called "Object Relationship Browser" providing a user with a way of browsing through layers using object relations stored in the database.

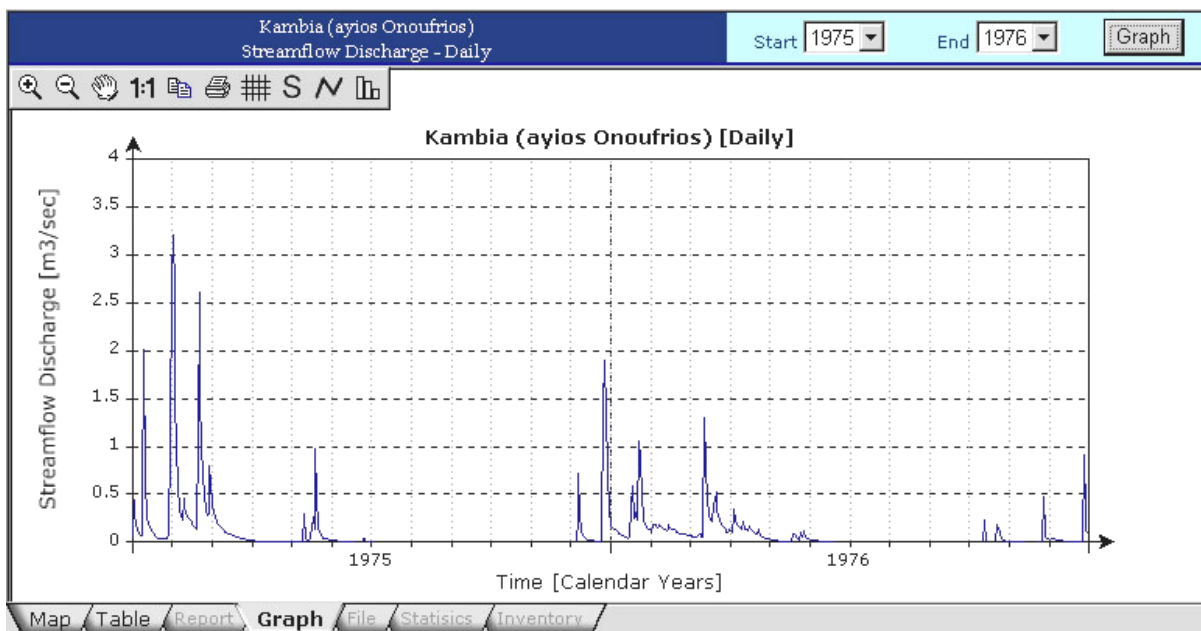


Image 14. Data Chart in VML format

4.5. Installation CD (ICD) design and creation (D.Lekic)

Defined as an additional task during the mission, design and creation of the installation CD required use of some atypical technologies. Mentioned above, the entire system structure was based on the idea of client-server computing, excluding the possibility of using it as a

standalone application installed on one computer, disconnected from the network. The present state of the computer network in WDD was used as a reason for including this additional item in the list of mission tasks. In reality, WDD consists of several divisions, situated at several locations in Nicosia, currently not connected in a common LAN. Following this fact ICD creation and design, providing an opportunity for disconnected clients to test and use developed applications, was included in the task list and successfully accomplished. ICD contains Apache Web Server, PHP script engine, main Database, Web access security database and set-up packages of both Database Management Software and Web Database Application. It can be installed on separated computer, providing server and client components at the same time. Original CD was delivered to the Division of Hydrology and it can be obtained through contact with Mr. Panayotis Skordis Deputy Project Coordinator with the WDD.

4.6. CDpack creation (D.Lekic)

Resulting from the previous task, satisfying the need for a system that can automatically create an updated ICD (updated with the new version of the database), a CDpack was created. It is a CD with a predefined structure of files and folders, which should be used as a template for ICD creation in the future. Advanced user has to copy the entire contents to the hard disk, replace the database file in the predefined structure and finally, after running a procedure that recreates the ICD structure, to copy (write) the created ICD from hard disk to the empty CD. The procedure that has to be followed in order to accomplish this task is also documented and furnished as the database administrator manual.

5. Conclusion and recommendations

Mission objectives were fully accomplished in a period of two months. Web Site, importing/exporting utilities and Web Database Application were created as components of Information System designed for Water Development Department. Unfortunately, the duration of the mission did not provide enough time for application debugging in real conditions. Therefore, all the user suggestions will be appreciated as well as any feedback indicating application bottlenecks, bugs or problems will be very valuable for the system designer as an input in the future application development.

Usage of the main WRDMS system, in the sense of filling the main database with all the available data and database access through Intranet connection is highly recommended. The standalone application available on the ICD should be used only as a demo or as a teaching tool for users disconnected from the network. Since the standalone version contains all the applications normally running on both client and a server side, a computer with the standalone application installed can be easily overloaded. Minimal recommended hardware configuration can be defined as a Pentium III 866 MHz processor computer equipped with at least 256 MB of RAM.

It is also recommended that the computer network between different divisions of WDD should be established as soon as possible in order to provide database access to the wider set of users.

Permanent Internet connection can be seen as a training resource by itself for the staff of DOH, since the utilisation of new, advanced technologies cannot be imagined without permanent software updating. Having in mind that almost all the components used in IS development belong to the Open Source (freely distributed and used software) community, importance of direct Internet connection is even more highlighted, meaning that new versions of used components can be easily downloaded and installed.

6. Programme of the mission and people met

The mission was accomplished in two parts. The first part, of duration of two weeks, between September 21 and October 5 2001, was mainly used for gathering all the needed information for IS development as well as for Logical IS structure definition. During the six weeks of the second part of the mission, from 26.10.2001 to 05.12.2001 all the major tasks were fulfilled with taking in account valuable input provided by the people met and consulted during the mission:

M. Panayotis Skordis, Executive Engineer, Division of Hydrology (WDD), Deputy Project Coordinator

M. Iacovos Iacovides, Chief Water Engineer Water Development Department, National Project Coordinator

M. Adonis Georgiou , Hydrogeologist, Division of Hydrology (WDD)

Ms Marilena Panaretou, Hydrologist, Division of Hydrology (WDD)

M. Christos Ioannou, Head of Division of Hydrology (WDD)

M. Christodolos Artemis, Director, WDD

Annexes

1. Database Manager Application Manual
2. Web Database Application Manual



Cyprus Data Manager Program

Manual

Overview

Three distinct parts compose the [FAO Project Information System](#): the database file (Access format), the Database Manager (CypFAOdbm executable) and the Web Database Interface, which is giving the final output for the clients.

[CypFAOdbm](#) is an application designed for the creation, maintenance and usage of the Cyprus Water Development Department (WDD) Database through user-friendly windows [Interface](#). The main objective of the program is to ensure the creation of the data structures: Tables, [Views](#), and [Links](#) between them in an appropriate manner according to the various needs of a national hydrologic service. The database format chosen is Access. Although it is recommended to use MSAccess® to have a more detailed view on the database structure, the application makes all the necessary jobs without the need of MSAccess® installation. The main features of the program are as follows:

- The application design is closely linked to the MapInfo export/import format. The program has import/export utilities from MapInfo MID/MIF ASCII format. It allows to have unique storage of geographical objects (vector format) in the Jet database, which could be used directly from Web applications or standalone applications, or after been exported back to the native MapInfo format – from MapInfo users.

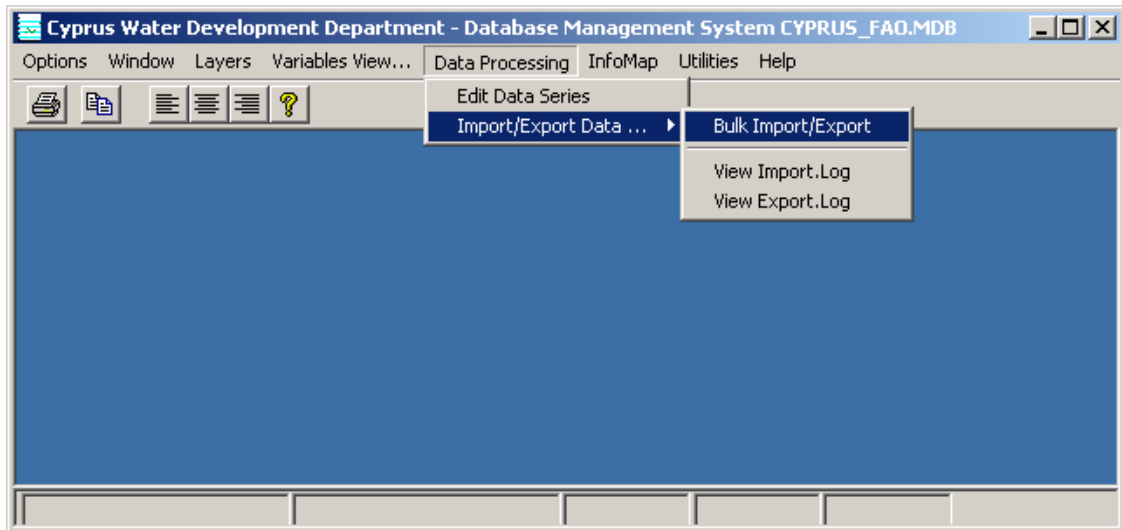
- Automatic creation of internal database structures after importing a new layer from MapInfo®. This is maintaining the internal logical (and physical) integrity with respect of the geographical relationships settled in MapInfo application or similar software that has MapInfo® export format included.
- Full compatibility with the [Web Database Interface](#) developed by Mr. Dejan Lekic for the Cyprus Water Development Department (WDD) - Division of Hydrology (DOH).
- Highly automates meta-data maintenance and reporting.
- Automates import/export routines for data import/export from the database.
- A developed linear graphics visual interface for simple data series analyses
- Full set of inventories on the geographical objects, relations between them and the data attached to them as data series or any kind of documents (MSExcel®, MSWord®, Images etc.)
- Editing tools for any tables in the database, except geographical coordinates.
- Printing facilities from the Linear Graphics window, any Grid and Text controls.
- Export to the clipboard for further usage in Windows systems.
- Internal usage of compacting tool for database file compacting and repairing.

CypFAOdbm is built according to the last conceptions of database access and maintenance ADO. Some internal objects as Views and Procedures couldn't be seen through the Access2000 interface although they exist and work through the program interface.

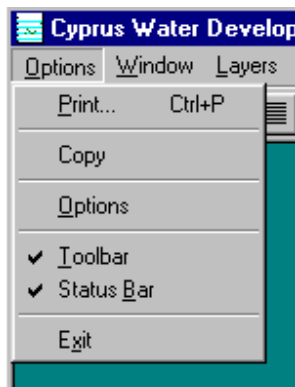
Main Menu Interface

The Main Menu Interface contains:

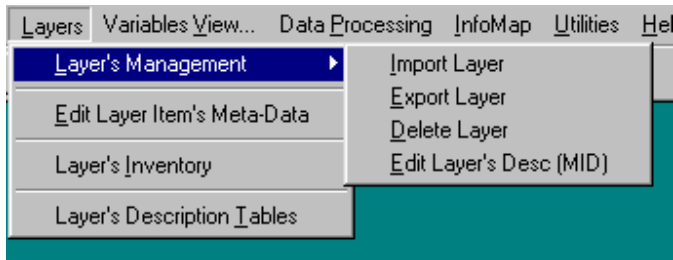
- Topmost row with menus, that are of the most common Windows program type
- Fast access button bar for printing, copying to the clipboard, text formatting etc.
- Status bar, on which is displayed the current status of the Jet connection



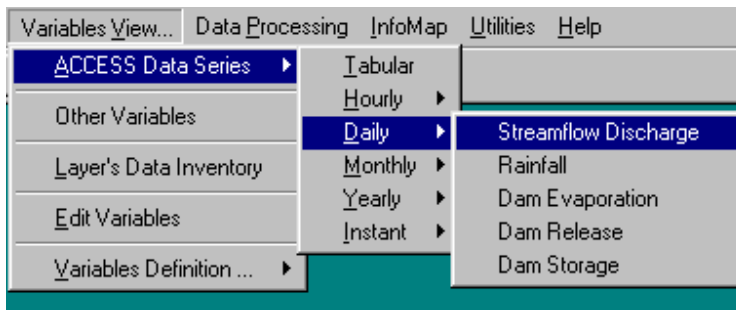
- The menu contains 8 items, in which are grouped related tools. By instance under the main menu item “Data Processing” are appearing the submenus “Edit Data Series” and “Import/Export Data ”.
- The “Options” main menu contains the “Print” facility, “Copy” facility, it allows hiding and showing of the buttons bar and the status bar. The main purpose of the “Options” submenu utility is to set the initialization parameters for the normal work of the program. Those are stored in an ini file and don’t need to be changed often.



- The “Window” menu entry is used to arrange the current active windows inside the main panel of the application. Those could be maximizing, minimizing and so on.
- “Layers” entry is linked to the main usage of the program: import, export, delete and current maintenance of geographical objects (layers) inside the Jet database.

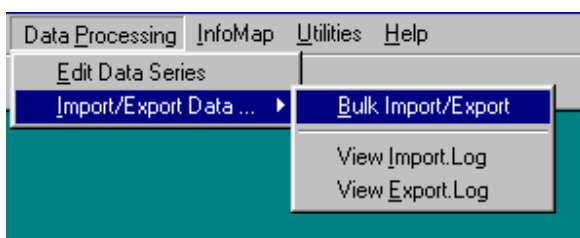


- “Variables View...” entry is permitting to see the variables attached to the objects in the database. They could be of type ‘Data series’ when a date field is present, ‘Table’, which are composed from several variables in one single table, ‘Table series’ – table with unique date field for a set of variables (i.e. Chemical analysis) and ‘Attachments’, as documents, pictures and so on.

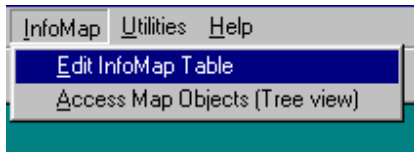


This entry contains also procedures for data inventory consultation and, printing of total or annual reports on the data about a selected layer.

- “Data processing” is containing tools for data series editing as well as the export/import routines. Here one can consult the log files with recorded input/output operations to the database.



- ‘InfoMap’ entry serves to maintain the [InfoMap] table. It contains also a tool for a simple mapping selection combined with a tree selection. This gives the user the possibility to find fast the object on the map and then to see all the data attached to it.



- 'Utilities' entry is mainly for file maintenance of the database. Through it one could compact or repair the Access file or in case of need to create a new (empty from data, but with appropriate structure) database file.

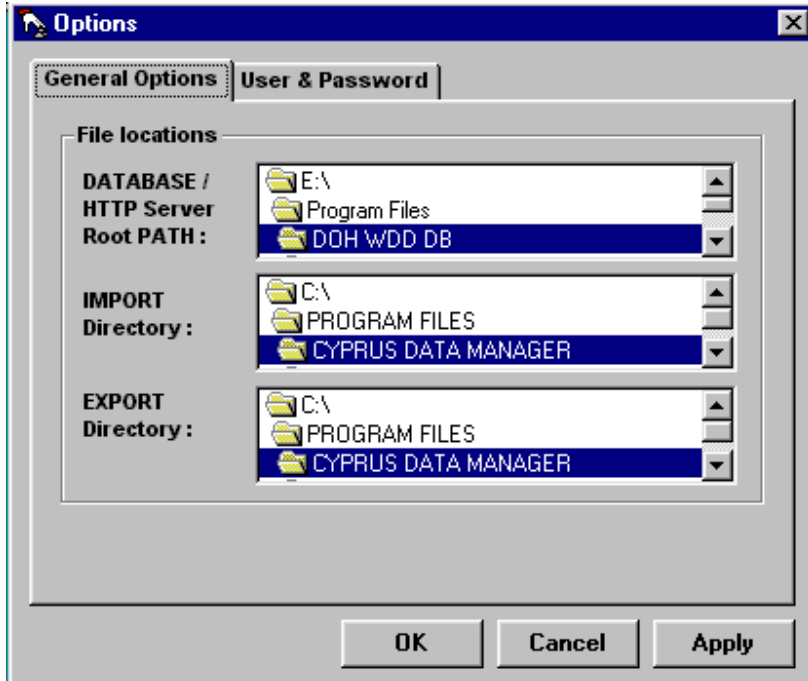


- 'Help' entry is giving access to the program's help file.

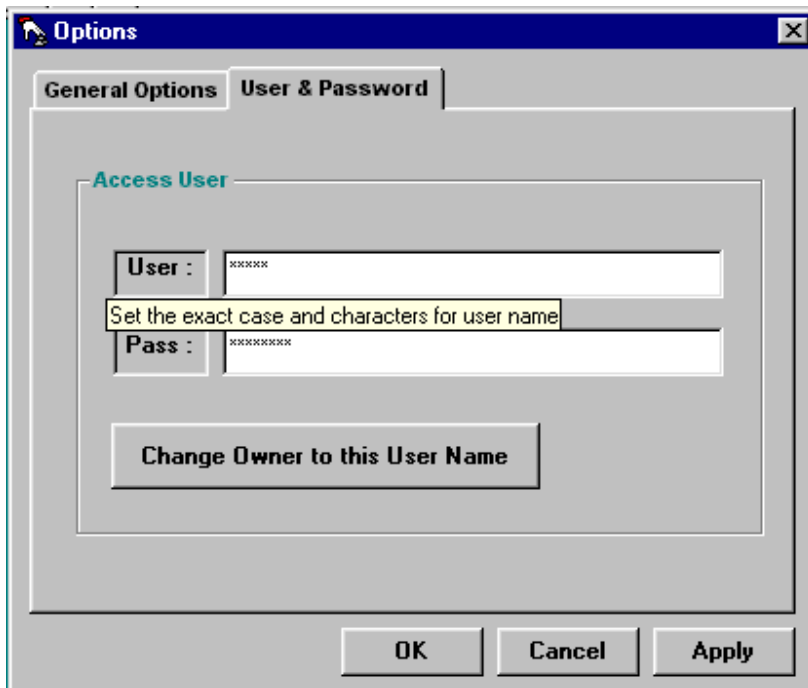
Options Window

The "Options" Window contains:

- The setting of the HTTP server and the path to the Database file. These two are closely connected as the intranet Web server is operating on the Access database file locally. So choosing the path to the database, at the same time you are choosing the path to the server. Usually, in the most common situation this is another PC running Windows NT or Windows 2000 server. It means that the relative path from the HTTP server root directory to the Database file is fixed and it is '\DATABASE'. By this way choosing the Database file directory – which should end with '\DATABASE', you are also choosing the server root path. If the path is situated on a drive other than C: hold the {SHIFT} key and click the <HTTP server path> box. This will bring the windows Open File Dialog in which you could search your Database directory. Once you selected it, press OK. It is also important to remember that the Database file name is fixed to CYPRUS_FAO.MDB.



- You are allowed to choose the default path for file export and the default path for file import. Those are important as you won't want to spend time on changing directories each time you want to export from or import to the database. If some of these are on a drive other than "C:," you have to edit manually the file "Data_Base.ini" in your application path in order to change them.



- The second panel of the window contains two fields providing the user with access to his user name and password. Without the right user name and password you will not be able to connect to the database. The database password should not be changed, unless you use the MSAccess user interface. This is not allowed, because the Web server access is working with it's own security, which is maintained outside the Jet user security policy. If you change the password, your Web data access applications will not work. If you are launching the application for the very first time you could see several panels saying that the user or password are incorrect. In this case close all of them, then reopen the "Options" window and set the proper user name and password.
- The application is maintaining two levels of access to the database objects. The password for the two levels is the same, but the user name is different. In fact only one user could open the application and use "Layer Management", "Edit Layer Item's Meta-Data", "Edit Variables" and "Edit InfoMap Table" menu items, all the other users who know the password have no rights to create or to delete geographical objects in the database, although they can still view them. The "administrator" could change his user name (that is acting like a password), using the "Options" second panel. If the user enters the program with the administrator's user name, he is allowed to change the user name of the administrator. To do that click on the <Change Owner> button.
- After closing the "Options" window the program asks again for the User and Password. If you are sure you entered them in the right way just click OK.

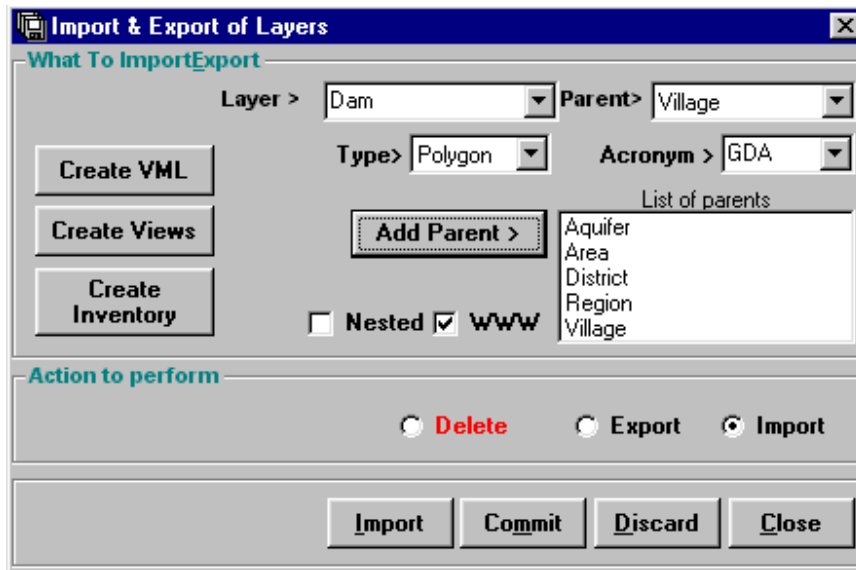
Layer Import/Export

This window is making possible to create all the structure needed in the Access database file, to store the tabular data of a geographical layer object as well as the geographical data (point and polygon coordinates). The imported data should be previously prepared within MapInfo application. The preparation consists in:

1. Putting together in the Layer Description Table all the very important, but static information about the layer objects. Those are: the Layer Item Name, the Layer Item unique code (string), the immediate geographical [parent](#) name (column), all of the higher [parent](#)'s names (columns) which are directly or through another layer linked to the actual layer. By instance in the table describing data about Dams the following columns should be present: SubWatershed_Name, Watershed_Name, Region_Name, as well as the columns for Village_Name, Aquifer_Name, Area_Name and District_Name. Other columns such as X and Y coordinates for point layers, elevation for point layers or mean elevation for polygon objects as watersheds should be present as well, describing additional features of each object in the layer. The name of that table in the resulting Access

data file will be composed from the Layer name (ex. dam or spring) as follows: “Dams_Desc” and “Springs_Desc”. After the import procedure is finished this table will be linked to all the [parent](#)’s description tables. That means that the names for parents in the imported file should match exactly those that are already in the database file. For example you could not import a layer for dam representation if one dam is linked to a parent not existing in the corresponding parent layer in the database or is not linked at all. In cases where some child items are outside the defined [parent](#) one must create a void parent with the name “None” to handle not linked child items. This is bringing the very simple idea, that the importation of layers in the database should be made starting with the topmost parent layer and ending with the child layers not connected between themselves unless indirectly, if they have common parent. It is very useful to prepare before any importation a [Hierarchical Tree](#) representation of the [Geo-Referenced Structure](#) of the database.

2. The layer names have to be carefully chosen, as they will be used for field captions in the database tables, for the names of the different [views](#), as well for the [inventory query](#) names. They must be descriptive but not too long.
3. It is not allowed to have as Layer Item names strings containing the following characters:
 - a. (‘);(“);(&) and comma.
 - b. Strings in non Latin alphabet (some of the import/export routines are loosing non-Latin characters or they are not appearing properly in the program’s windows)
 - c. The above mentioned is true for any text imported in the database, unless it will not be used within the Database application nor in Web database interface.
4. It is a very good idea to collect the entire layer and item’s information working with MapInfo before exporting them for database import. Although database import allows the updating of geographical information, this process is more complicated than importing a new layer and could cause some unpredictable results. In case you want to update the geographical information of some already existing in the database layer, you should first export the layer from the database (the result is a couple of files, for example for spring they will be: SPRING.MID and SPRING.MIF), then to work with them in MapInfo in order to make changes and after to re-import them in the database.



5. Working with the Import Option you have to:
- Write in the top-middle box the new layer's name. By default this should be the file name that contains the geographical data, exported from MapInfo. It's good idea to write it starting with Uppercase. Don't forget that this name should match exactly the one mentioned inside the MID file as layer name.
 - Write in the acronym box a 3-letter string. This will be used to identify the new variable that will be created automatically to handle the geographical data for the layer. In fact the data table will have the name "DATA_XXX", where XXX will match your acronym. For instance geographical data about dam's layer could have an acronym GDA. This should be written with UPPERCASE letters.
 - Chose from the list the parent's names and for each of them click on the button <Add Parent>, they will appear in the list. If you added something unneeded click twice on its item in the list to remove it.
 - Choose the layer type from the list <Layer Type>. You must be careful, as the default position of the list is **Polygon**, while you may wish to import a **Point** type layer. The option for River layer or Pipeline layer should be **Line**.
 - If the layer is uniquely linked with a parent one, but in the same time it is not logically related to upper parents it is *nested*. In this case check the <Nested> check box.
 - If the layer should be stored in the database but not used for Web Database Interface uncheck the <WWW> check box

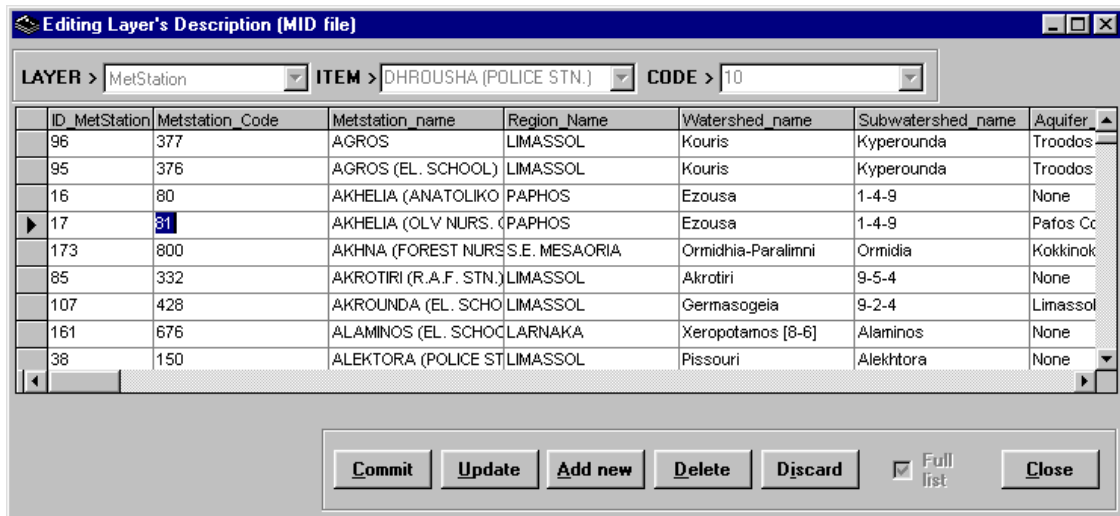
- g. At the end press the <Import> button in order to launch the import procedure. This could take some time, especially for huge layers with a lot of points or many polygons.
 - h. If the importation is successful you will be prompted twice to confirm the operation. Be careful and press “OK” only when it is not an Error Message. If the operation fails for some reason (improperly filled MID file, missing parent name, etc.) you will be prompted to Discard changes, including to delete data tables. In this case you should answer “OK” or “Yes” as if some tables were already created they will stay in the database as garbage. **In case you are updating** the layer you could loose data series or other data connected to it. Because of that you **must make a full export** of your data **before starting to update** the geographical data of your layer.
6. In order to export your geographical data in MapInfo format you should select the export option box, select the layer you want to export then press the <Export> button. This will create in your [Export Directory](#) the MID and MIF files in MapInfo format.
 7. In order to definitively delete a layer you should select the delete option box, select the layer from the list and press <Delete>.

Layer Editing

Through the “[Layer](#)” main menu item you have access to the “[Edit Layer’s Desc](#)” and to the “[Edit Layer Item’s Meta-Data](#)” submenus. The first one is permitting to edit the item’s description. The second has the role to manage the layer item’s meta-data.

Editing Layer Items Description:

- Choose using the <LAYER> and <ITEM> lists, the layer name and item that you want to change. You could either chose the item by name or by code from the <CODE> list.



- Click on the <EDIT> button.
- **Deleting an item is launching the cascade deleting of all the associated items in the database and the data related to them!**
- Make the necessary changes knowing that they will propagate within the database if they are made in a column linked to other tables. Such columns are all the columns with layer names. You should not change the column with caption ID_[LAYER]. This column is used for the unique internal database indexing of the rows.
- In order to confirm changes in the current cell it is enough to change the cell. When you are working on a new added row, you have to click in another row. Don't click on the empty row if don't plan to add a new record.
- Because of the differences between regional parameters describing the decimal separator symbol, writing numbers with decimal point should be done from the right side of the keyboard (numeric keypad).
- Adding a new row in the table is equivalent to adding a geographical item, that could have attached data series or other documents, but will not appear on the map, as it has no geographic information associated with it. The next step should be to add the geographical information using the "[Layer Import](#)" utility. In this case adding the information here seems not to be useful.
- In order to confirm all the changes you have made press the <Update> button. It has the same effect as pressing <Commit>
- In case you don't want to confirm changes and go back to the previous state of the database press <Discard> button.
 - The first column is designated to contain the ID number. It is an internal unique number for the database and it is attribute not of the layer item but

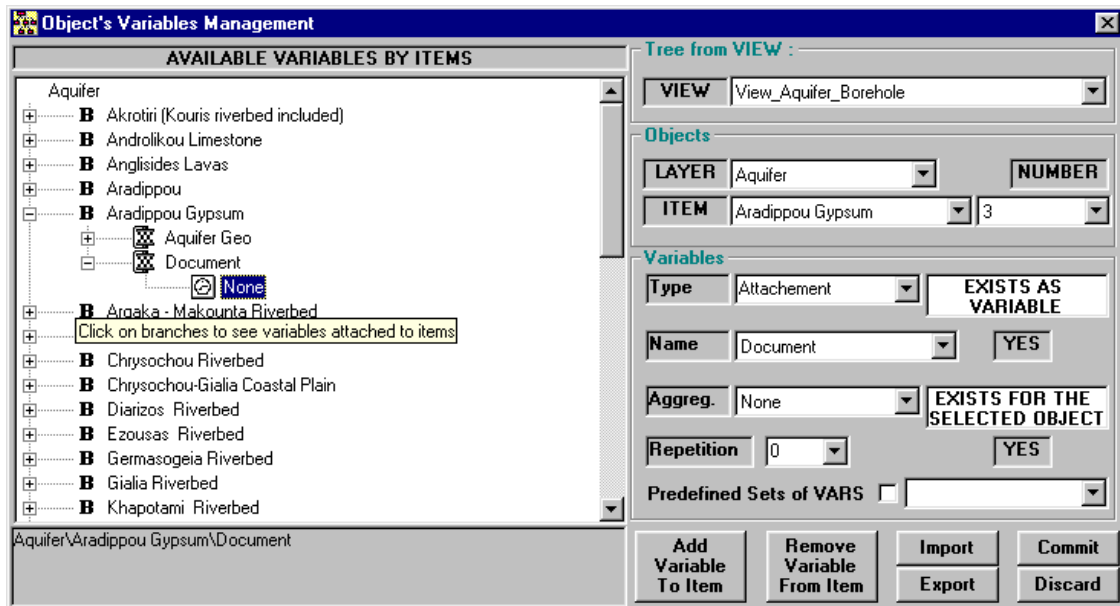
of the row in the data table that contains it's description. You should not try to fill or to edit the values in that column. They are filled automatically when you add a new row.

- Especially for layer item names some characters should be avoided as this field serves to make the links between tables and are not supported by database systems for that purpose. Such characters are (&), ('), ("). Instead of them use (_) or (-).

Layer Meta-Data Editing

Layer Meta-Data represents the link between the Variables and the Layer Items that have those Variables attached to them. In the database, for each Layer in the moment of the importing, a table [Layer_Vars] is created. When a user decides to attach some data to some of the items in this layer, he should specify in the table [Layer_Vars], the id number of the layer, the id number of the variable and the repetition number. The first two numbers are useful for the program, but don't mean anything for the user. The first one is giving only, for internal purpose, the row where the layer item description is. The second is also for internal purpose and is giving the id of the variable. The program is made to hide all these details from the user, but at the same time to give him a simple way to attach variables to the layer items and to see and edit the attached variables. This is done through the Layer Meta Data Editing window.

- In order to see the variables attached to a certain layer item, click on the list <LAYER> and choose the right layer. Then from the list below <ITEM> choose the right item. The next step is to scroll through the list in the panel <Variables>. One of them is <Name>; the second is <Aggregation>. When you chose a name and aggregation for the variable, you will see if it is attached to the selected layer item. In case it is, you will see the label <EXISTS AS VARIABLE> set to <YES> and the label <EXISTS FOR THE OBJECT> set to <YES>.



- In order to attach a variable to an object you have to fill the <Repetition> box and press the button <Add Variable to Item>. The repetition is a number associated with the variable row, that permits to have several rows with the same Id item, date and value (in case of data series). This is made in order to provide support for data series with different quality index or for other similar purposes.
- In lot of cases after having imported a layer with a lot of items, you need to attach one or more variables to several items or to all the layer items. This facility is provided through the button <Import>. The links between the item numbers and the needed variables are described in a [Var_Layer.txt](#) file that should be read from the program. If you need to correct few or more of the links between the items and the variables, you could use the button <Export>, which will produce the same file in the right format. Then you could edit it and re import using the <Import> button. The option <[Predefined set of VARS](#)> has similar role.
- Removing a variable from a layer item is deleting all the data related to this item about the variable, which is chosen in the <Name> and <Aggregation > lists.

Layer Inventory

- Layer Inventory is a simple window showing the multiple links between the geographical objects in the database. Immediately after the layer is imported, the program evaluate all the possible links between the existing layers and make a full set of views linking them from the top-most “parent” (except Country) to the lower “child”.
- This tool is permitting to locate the items easily by allowing the user to sort by columns and to group two or more columns together by drag and drop operation.

District_Code	District_Name	Village_Code	Village_Name	VillageCenter_Code	VillageCenter_Name
1	Lefkosia	1021	KATO	1021	KATO LAKATAMEIA
		1200	KAMPI	1200	KAMPI
		1201	FARMAKAS	1201	FARMAKAS
		1202	APLIKI OREINIS	1202	APLIKI OREINIS
		1203	LAZANIAS	1203	LAZANIAS
		1204	GOURRI	1204	GOURRI
		1205	FIKARDOU	1205	FIKARDOU
		1206	AGIOS	1206	AGIOS EPIFANIOS OREI
		1207	KALO CHORIO	1207	KALO CHORIO OREINIS
		1209	KLIROU	1209	KLIROU
		1300	PALAICHORI	1300	PALAICHORI MORFOU
		1301	ASKAS	1301	ASKAS
		1302	ALONA	1302	ALONA

Select View: View District VillageCenter1 Remake Views Close

- To move a column to another location press its caption with the mouse and drag it with the mouse to the new location.
- To sort the grid by column content, click on its caption. One click is sorting the grid by ascending order of the chosen column, the second by descending order.
- The grid could be printed using the <Printer> button under the main menu or invoking the printer through the menu. It could also be copied to the clipboard if the user presses the <Copy> button. The copied into the clipboard grid could be used in Word or Excel. If you need to print less than the entire table – select with the mouse the necessary rows then press the <Print> button.
- The button <Remake Views> should not be used unless you want to force remaking of all the views. The program is remaking the views each time you import a new layer.
- This grid is not for editing. If you want to edit some of the items names or other features, use the tool “[Layer Editing](#)”

Layer Description

This tool is providing the possibility to view a single layer description table. It shows all the available information that is linked to one layer item in one row. Its format is exactly the same as it was in the MID file exported from MapInfo.

Watershed Layer Description				
ID_Watershed	Watershed_Code	Watershed_name	Region_name	WS_Area_sq_km
1	6-1	Pedhieos	MESAORIA	867.14001
2	6-2	Lefkoniko	MESAORIA	163.24899
3	6-3	Syngrasi	MESAORIA	79.786201
4	6-5	Yialias	MESAORIA	602.59302
5	4-1	Orga	KYRENIA	19.5341
6	4-2	Panagra	KYRENIA	41.519199
7	4-3	Lapithos-Karavas	KYRENIA	75.655197
8	4-4	Kyrenia	KYRENIA	64.288803
9	4-5	Klepini-Kharcha	KYRENIA	78.459198
10	4-6	Ayios Amvrosios	KYRENIA	47.569302
11	4-7	Akanthou	KYRENIA	55.738602
12	4-8	Phlamoudhi	KYRENIA	50.632401
13	5-1	Platanisso	KARPAS	33.073898
14	5-2	Yialousa	KARPAS	66.011398

Layer name: Watershed [Close]

Use the list <Layer Name> to select which layer Description table you wish to explore.

The columns could be grouped and sorted as well as copied to the clipboard or printed. For details see the previous chapter. Clicking on the captions is sorting the grid by the selected column value.

Variable Definitions

Variables are the second main element of the Database System. The layer items are the geographical areas or lines or points, defined in the database. They are the objects to which Variables have to be attached. Variables could be of any type. The Data Manager application is using five major types of Variables:

- Geo
 - The Geo variable type has only one usage – to store the geographical information of the specified layer item. It has 3 columns: one to store the particular item code, another for the geographical information (coordinates of points of lines or polygons) and a third for the style with which that object is drawn on the map. You don't have to create this type or to edit it, as it's automatically done by the application before the very first layer is imported in the database.
- Data Series
 - Data series represent a time series of one variable linked to particular object. That means that each row of the data series must contain a field for the item's code, a field for the date and a field for the value of the variable.

- Data series in the Data Manager could be of 5 aggregation types: Instant and InstantH (that means not regularly measured in time), Hourly, Daily, Monthly or Yearly. The program assumes that each of these series contains date field with day, month and year information. Only the Hourly and InstantH Data Series contains in addition hour information. The day for the Monthly Data Series is by default the first. For the Yearly Data Series the month is January and the day is 1. Instant Data Series don't contain hour information.
- Data Series Values are all by default of type Single or Float. That means that they can store negative and positive numbers from $3.4E-38$ through $3.4E+38$, although in some grid representation the part after the decimal point is cut up to three digits, the data in the database contains the number of digits after the decimal point that you provided while importing the data series.
- Examples of Data Series are Streamflow Discharge Daily, Rainfall Monthly etc.
- If the user selects the Data Series type from the interface the fields for the column names and type are filled automatically.
- Table Series
 - Table Series is a group of two or more Data Series linked to the same object and having the same time stamp. By instance the several chemical analyses that are made the same day for one point (dam, river gauge, tap water etc.) grouped together is giving one row in the Table Series. This row has in its first column the id of the point, in the second the day of the analyses and in the following columns the full list of measured values.
 - Table series are designed to store better a lot of different data about one point measured the same date. Usually variables of this type have Instant time step, although it could be each of the five types mentioned above.
- Table
 - The Table variable type is the most flexible. It does not have to contain a date-time stamp. The only required column is the first, which should contain the id of the object (item) to which the data belong. In the Table data type, the statistics and text descriptions of a particular Village or other geographical item could be stored. For example those are the demographic characteristics of the village, water demand and water use for the different groups of users (domestic, tourist, industry, various crops)
- Attachment
 - The Attachment variable type is designed to store the various documents, images or any other computer manipulated objects (maps, video, sound records). It is necessary to understand that in the database the links between those documents to the particular item are specified, but the documents themselves are copied to the directory of the Web server, from where the Web Application could retrieve them. In other words, they are

not physically existing in the database, but the applications know where they are on the server and to which geographical object they are belonging.

Working with Variables Definitions

- Adding a variable consists in specifying the required fields that describe its properties.
 - Type of the variable: Data Series, Table Series, Table or Attachment. This could be chosen from the list at the right of the text box <Type of the variable>
 - Name of the variable:
 - For Data Series, type has to be chosen according to the type of data. For example Streamflow Discharge
 - For Attachment, type has to be chosen between ‘Document’, ‘Image’ or ‘Other’
 - For Table and Table Series the name depends on the Item’s name. By instance it could be ‘Village general’ or ‘Chemical Analyses’

The screenshot shows a dialog box titled "VARIABLES" with the following fields and controls:

Id_var :	31	< Don't fill ! (autoincrement)
Type of the variable:	Data Series	Data Series
Name of the variable :	Dam Storage	Dam Storage
Program :		
Window :		< Optional
Aggregation :	Instant	
Acronym :	DSI	< To fill (ex. PRD) PrecipDaily
Measurement Unit :	m3*10^6	m3*10^6
Data Table :	DATA_DSI	< To fill (ex.DATA_PRD)
Attach to object :	Dam	Dam
Columns Count :	4	
Columns Names :	ID_OBJECT;DAT;VALU;REP	1+ Check fields before Update
Columns Type (choose!)	INTEGER;DATE;SINGLE;BYTE	add+
Columns Lenght (bytes) :	4;8;4;1	reset
Indexed Columns :	ID_OBJECT;DAT;REP	Make Table

Buttons at the bottom: Add, Edit, Delete, Commit, Discard, Close.

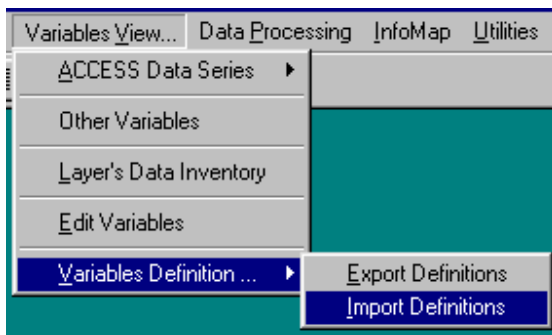
Footer: Use to Change the Viewed Variable

- Aggregation: this is specified only for Data Series type and Table Series type. It could be chosen from the list at the right from the text box <Aggregation>
- Acronym: the Acronym is short description of the variable name that could have some relation with the aggregation. By instance the Acronym for Streamflow Discharge Daily could be SDD, but it remains the user's choice.
- Measurement unit: this could be chosen from the list or written in the text box.
- Data table name to be created to store this Variable
- To which layer you wish to attach the variable. Chose it from the list at the right of the text box <Attach to object>
- Filling the column characteristics:
 - You could fill them manually or use the buttons <1+> <last+> <add+> and <reset>
 - If you use the buttons: press <1+> for the first column, then for each next column chooses the right type in the list and then press <add+>. In this case you should only type the name of the column in the <Column Names> text box. At the end press <last+> to add the repetition number column. This procedure could be more effective than typing the names and types, as it fills automatically the right column count, the right types and length for numeric and text columns.
 - If you have to enter a lot of column names, it's a very good idea to use Excel to prepare the names and the types. If you already have a table with the specified data containing captions, you could save the Excel file in semi-colon separated ASCII file then to copy paste the first row into the <Column Names> text box.
 - The INTEGER data type is for numbers above 32768, for the integer numbers below use the SMALLINT type. For the integers below 255 use the BYTE type. Boolean values are using BIT type. LONGTEXT is for storing strings above 255 characters, otherwise use TEXT type.
- Don't forget that one column should be added first for the item's ID (must have the name "ID_OBJECT" and type INTEGER) and one at the end for the repetition number (must have the name "REP" and type BYTE). This is valid for each type of variables. For Data Series and Table Series a second column is required to store the date stamp. It should have the name "DAT" and be of type DATE.
- Specifying the fields for the unique index. The obligatory fields here are:

- “ID_OBJECT” and “REP” for non Data Series and Tables Series
 - “ID_OBJECT”, “DAT” and “REP” for Data Series and Table Series
 - After you fill the required fields press the <Update> button and then the <Commit> button. If you omit or not properly fill some fields the application will ask you to make the required changes in the fields.
- If you want to edit an already created variable, you could change only few things:
 - The name
 - The acronym
 - The aggregation of data series could be changed but if you have data in the tables it will be interpreted as it is with the updated time step, although the data will not be changed.
 - The measurement unit
- Deleting a variable is achieved by pressing the <Delete> button. In this case you are deleting the Variable Definition, but also all the rows containing this Variable name of this Aggregation, attached to the current Layer.

Exporting and Importing Variable definitions

Exporting of variables definitions is achieved through the menu item “Variables View” > “Variables Definition” > “Export/Import Definitions”



- The export operation is producing a comma delimited ASCII file with the name VARIABLES.txt in your Export directory. The numeric fields are exported without enclosing; the text fields are enclosed with quotation marks (ex. “Aquifer”); the Null fields are exported enclosed with the sharp sign (ex. #Null#)
- To import Variable Definitions you have to prepare previously the VARIABLES.txt file according to the above rules. It could be easier to make an export, and then to edit the file according to the new variables you want to define in the database and then to import it using the menu item <Import Definitions>.

An example of [VARIABLES.txt](#) is given at the end of this document.

Data Inventory

The menu entry “Layer’s Data Inventory” is opening a window through which is available a set of different reports on the data stored in the database:

To work with that window just choose the layer name that you want to analyze the data in the list <Layer Name>.

- The upper grid is giving full view on all the attached variables to the chosen layer items.

The screenshot shows a software window titled "Database Data Inventory". It contains two data grids and a control panel on the right.

Upper Grid:

ID_OBJECT	OBJECT_NAME	OBJECT_NUM	IDV	HMV	VAT
68	TSERI_115	1225/115	10	Borehole Geo	Geo
68	TSERI_115	1225/115	18	Borehole Water Level	Data Se
66	ASKAS_0	1301/98/80	10	Borehole Geo	Geo
66	ASKAS_0	1301/98/80	18	Borehole Water Level	Data Se
31	AKAKI_101	1360/101	10	Borehole Geo	Geo
31	AKAKI_101	1360/101	18	Borehole Water Level	Data Se
32	PERISTERONA LEFKOSIA_36	1361/36	10	Borehole Geo	Geo

Lower Grid:

Item Number	Item Name	Hydro Meteo Variable	Aggreg.	Rep	Rec. count	1 Date__	Last Date	%F	Average	St
1225/115	TSERI_115	Borehole Water Level	Instant	0	30,	26-04-1984	28-03-2001	0	159,433	7,2
1301/98/80	ASKAS_0	Borehole Water Level	Instant	0	74,	16-12-1984	05-03-2001	0	893,82	6,8
1360/101	AKAKI_101	Borehole Water Level	Instant	0	39,	26-04-1978	12-12-2000	0	213,903	2,7
1361/36	PERISTERON	Borehole Water Level	Instant	0	28,	26-03-1985	21-03-2001	0	138,443	4,4
1457/152	KATO PYRO	Borehole Water Level	Instant	0	141,	05-10-1987	02-03-2001	0	18,318	7,0
1457/154	KATO PYRO	Borehole Water Level	Instant	0	185,	04-03-1986	03-07-2001	0	4,513	5,0
1461/13	PACHYAMMO	Borehole Water Level	Instant	0	478,	24-02-1986	01-11-2001	0	1,055	0,4

Control Panel (Right):

- Clear Before New Analyse
- Layer name: Borehole
- Make Views Inventory
- Analyse Inventory
- Total Report
- HTML
- Annual Basis Report
- Close

- The lower grid that is filled only if you press the <Analyze Inventory> button is giving the information about the stored data series in the database, that are related to the chosen layer. Here one could find the number of records of each data series by items, as well as few other statistics made through an SQL query (for example Minimum and Maximum of the data series, Average, St. Deviation, Date of the Maximum and of the Minimum etc.)
- The user can sort both grids by the values in one column. The grids could be exported to the clipboard or printed.
- The button <Total report> is making an extraction of the above grid, which could be printed in an ASCII for HTML file.

Layer Data Report

Zoom 100%

Water Development Department

1 of 2 pages Database Data Series Report - Total

LayerName: *Borehole*

Variable *Borehole Water Level*

<i>Item_Name</i>	<i>Item_Code</i>	<i>Aggregat</i>	<i>Starting</i>	<i>Ending</i>	<i>Count</i>
ACHELEIA_2662	6014/2662	Instant	01-01-1976	12-09-2001	306
AGIA MARINA CHRYSOCH_462	6365/462	Instant	01-01-1987	01-05-2001	62
AGIA MARINA CHRYSOCH_478	6365/478	Instant	01-01-1987	01-05-2001	69
AGIA VARVARA PAFOU_3025	6106/3025	Instant	01-01-1976	12-10-2001	304
AGIOS THEODOROS LARN_38	4307/38	Instant	01-06-1974	12-04-2001	93
AGROS_0	5366/21/82	Instant	11-12-1984	04-04-2001	69

Pages: 1

- The check box <Make new report queries> is used only in case you want to force queries creation, before their execution. This is an optional possibility.
- The check box <Clear before New Analyze> is used when you have made an analysis having the lower grid filled and before the next analysis you want to clear it instead of having the second analysis following the first in the grid.
 - Pressing the button <Annual Basis Report> is producing a printable report with the number of records per year for each variable related to the chosen layer. The last year in the report could be chosen from the list below the button <Annual Report>.

Water Development Department - Layer Annual report

Zoom 100%

1 of 3 pages Database Data Series Report per Annual basis

LayerName: *Borehole* **Variable:** *Borehole* **Aggreg:** *Instant*

<i>Item_Name</i>	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
ACHELEIA_2662	12	12	12	12	12	12	12	12	11	12	12	12	12	12
AGIA MARINA CHRYSOCH_462	NON	11	7	6	6	6	6	4	4	2	1	2	2	2
AGIA MARINA CHRYSOCH_478	NON	12	7	6	7	6	6	5	5	2	2	3	3	2
AGIA VARVARA PAFOU_3025	12	12	12	12	12	12	11	12	12	12	12	10	12	11
AGIOS THEODOROS LARN_38	1	3	NON	8	5	NON	NON	NON	5	2	7	4	1	NON
AGROS_0	7	7	8	7	4	3	5	3	3	4	2	2	2	2
AKAKI_101	3	2	2	2	2	2	2	2	1	1	1	1	1	1
AKROTIRI_775	12	12	12	12	11	12	12	12	12	12	11	11	10	NON

Pages: 1

Data Series – Tabular View

The tabular view is the most common view on the Data Series

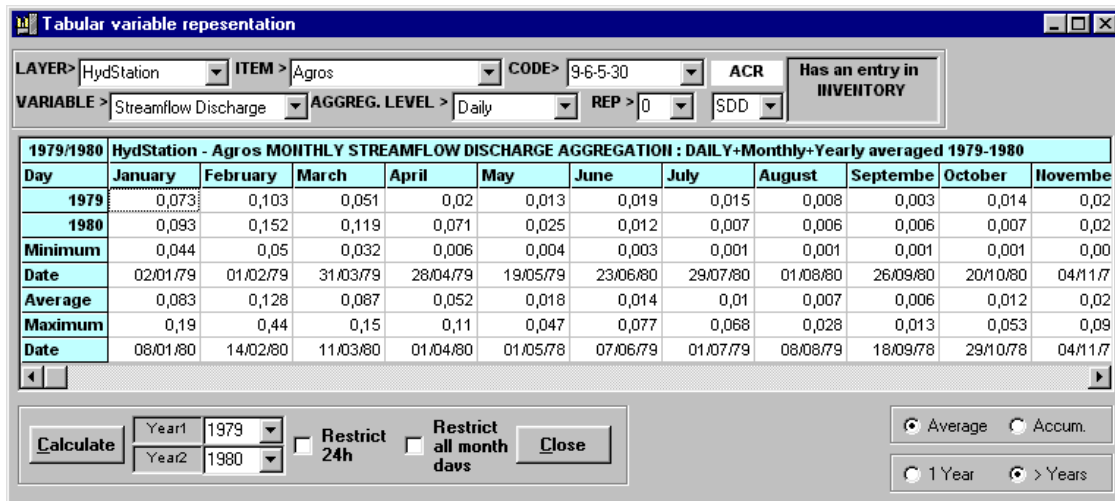
Using it one could produce an annual grid with daily values, monthly averaged, maximum and minimum as well as yearly aggregated values. It's important to know that those grid and statistics are evaluated over a calendar year.

The screenshot shows a software window titled "Tabular variable representation" with a data table. The window has a menu bar and a toolbar. The data table is titled "1979 HydStation - Agros ANNUAL TABLE : DAILY STREAMFLOW DISCHARGE AGGREGATION - DAILY+Monthly+Yearly averaged". The table has columns for months (January to October) and a "Hove" column. The rows include days of the month (26-31), Minimum, Date Min, Average, Maximum, Date Max, Max Year, and Min Year. The bottom of the window has a "Calculate" button, "Year1" and "Year2" dropdowns (set to 1979 and 1980), "Restrict 24h" and "Restrict all month days" checkboxes, a "Close" button, and radio buttons for "Average" (selected), "Accum.", "1 Year" (selected), and "> Years".

Day	January	February	March	April	May	June	July	August	Septembe	October	Hove
26	0,057	0,061	0,048	0,013	0,014	0,008	0,009	0,003	0,003	0,02	
27	0,057	0,065	0,043	0,009	0,014	0,014	0,002	0,002	0,003	0,017	
28	0,055	0,059	0,039	0,006	0,02	0,019	0,002	0,002	0,003	0,014	
29	0,053		0,039	0,008	0,014	0,014	0,003	0,002	0,002	0,011	
30	0,053		0,036	0,015	0,014	0,046	0,01	0,001	0,004	0,009	
31	0,053		0,032		0,014		0,008	0,001		0,008	
Minimum	0,044	0,05	0,032	0,006	0,004	0,003	0,002	0,001	0,002	0,003	
Date Min	02/01/79	01/02/79	31/03/79	28/04/79	19/05/79	23/06/79	14/07/79	30/08/79	01/09/79	19/10/79	04
Average	0,073	0,103	0,051	0,02	0,013	0,019	0,015	0,008	0,003	0,014	
Maximum	0,12	0,4	0,064	0,034	0,025	0,077	0,068	0,028	0,009	0,036	
Date Max	08/01/79	09/02/79	12/03/79	01/04/79	24/05/79	07/06/79	01/07/79	08/08/79	05/09/79	23/10/79	07
Max Year	0,4	09/02/79	CALENDAR YEAR								
Min Year	0,001	30/08/79	CALENDAR YEAR								

It's not possible to produce annual grid with daily values starting from monthly values.

The second option is to produce multi annual grid with monthly averaged (accumulated) values. In this case the user produces a grid with 12 rows and so many years he has chosen from the <Year1> and <Year2> lists. This option is chosen when the user clicks on the <> Years> option button.



When using multi annual grid you will receive 12 monthly aggregated values per year and average, maximum and minimum below the grid. At the right side the user will have 5 additional columns per year: the yearly average (accumulate) maximum, minimum and the corresponding dates. All the statistics above are made for a calendar year.

Data Series – Linear Graph

Linear Graph is designed to show Data Series data type.

Here you could make several operations as Editing, Aggregating, Exporting Aggregated values, Exporting the graph or its data to the clipboard etc.

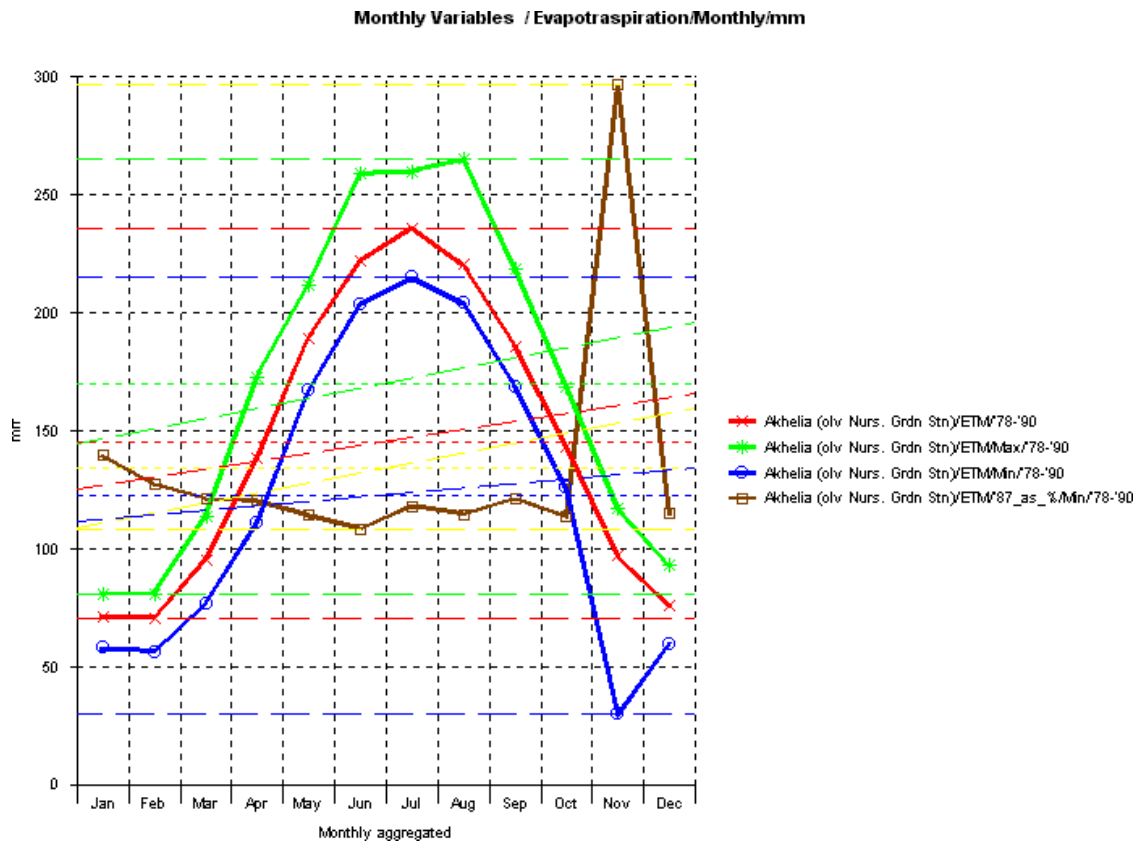
Editing has to be made in the grid. If you change a cell in the grid you have to press <Update> button. After the editing press <Commit> to confirm all the changes you made.

To get help on some button or check box, just cross over it with the mouse. All of the controls inside this form provide tool tip information.

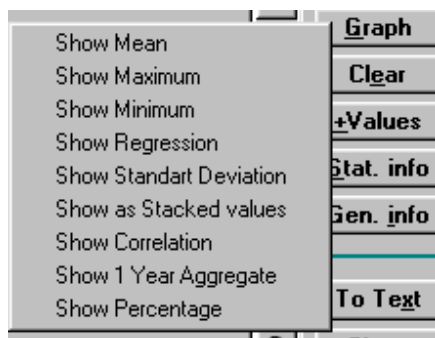
The graph has few integrated features:

- Changing the line shape is achieved with holding Ctrl & Shift keys and clicking the chosen series in the legend;
- Changing the series appearance (to bars or areas) is achieved with holding Ctrl key and clicking the chosen series in the legend;
- Changing the marker shape is achieved holding the Shift key and clicking the series in the legend;
- Clicking the series in the legend without holding Ctrl or Shift is allowing to change its color;

- To change the title hold Ctrl key and click it with the mouse;
- To see the point's data, click on the point in the graph, then the grid marker is positioned at the corresponding row;



- Clicking the <Add Graph> button with the right mouse button is showing a menu with several options:



- Choosing “Show Mean”, “Show Maximum” and “Show minimum” is adding to the actual graph the corresponding lines, but for the next drawing and only in case you have chosen an aggregation level from the frame <Avg or Cum> you will get the corresponding values. Notice that choosing Maximum is overlapping Mean

- and Minimum. That means that if you want to extract Minimums (Monthly, Yearly) over a given period, you should uncheck “Show Maximum”.
- b. The aggregations chosen from the frame <Avg or Accum> could be performed by two ways.
- If you check the < m > option button you will receive an annual monthly graph with 12 values that are either monthly averaged values over all the period, either monthly maximums or minimums over all the period. In order to be accurate you should take care to choose the right period (having full set of data) using the <Period> frame. After you selected the needed period refresh the grid by reselecting the Layer item from the list <Item>
 - If you check the < A > option, only in case you are performing annual aggregation, you will get a graph with 365 daily values (Averages, Maximums or Minimums) corresponding to the Average Year – which is composed from the averages of each day or to extracted over all the period maximum or minimum values for each day.
 - If you don't check either < A > or < m > boxes you should get a graph with monthly averaged or yearly averaged (or max or min) values.
- c. If you check the “Actual Year” menu item you will get the averaged (or max or min) values extracted only from the year specified in the list <Actual Year>. By this way one could perform comparisons between a long period and a particular year.
- d. If you specify “Show Percentage” you will get a series with the calculated percentage of the last added series over the previous added one.

Other Variables

From the menu item “Variables View...” one could choose the sub menu item “Other Variables”. That's bringing up the tool for visualize data, add, edit or delete records for variables that are not of type Data Series. Such variable types are Table type, Table Series and Attachment type.

The tool is the same for the three types and the actions to perform in order to add, edit or delete data are similar except in few particular cases.

Edit Data Table

LAYER > Village ITEM > LEFKOSIA CODE > 1000 VAR > Village General/None/0

Display Grid Graph of selected data Edit Grid

ITEM CODE	ITEM NAME	POPULATION2000	ANIMAL_HUSBANDRY	NO_OF_TOU	NO_OF_TOU	DOMESTIC_Y	IRRIGATION	TOURIST_VW	AREA
1012	STROVOLOS	57734	25161	-	-	4530682	61402	-	-
1000	LEFKOSIA	52731	18859	1862	952	4138045	343775	636198	-
4000	LARNAKA	48825	2538	3204	1339	3831523	16425	789262	-
1021	KATO LAKATAMEIA	23466	16212	-	-	1841516	159346	-	-
6000	PAFOS	21928	2832	11504	4346	1720769	882385	1839978	-
1013	AGLANGEA	19613	14129	51	27	1539142	7800	17425	-
5022	KATO POLEMIDIA	17932	161172	-	-	1407215	1273083	-	-
1010	AGIOS DOMETIOS	13584	-	-	-	1066007	4500	-	-
5011	MESA CEITONIA	12938	1566	352	176	1015290	199013	82389	-
1023	LATSIA	11228	6821	-	-	881081	411142	-	-
1011	EKKOMI LEFKOSIAS	11146	-	165	71	874659	66336	56376	-
3101	PARALIMNI	8551	52549	15969	7670	671052	1009268	1593379	-
4010	ARADHIPPOU	8091	743861	-	-	634954	748422	-	-
5012	AGIOS ATHANASIOS	7774	2438	288	144	610072	161393	67409	-
5013	GERMASOGEIA	6621	1156	5890	2388	519573	960322	1378609	-

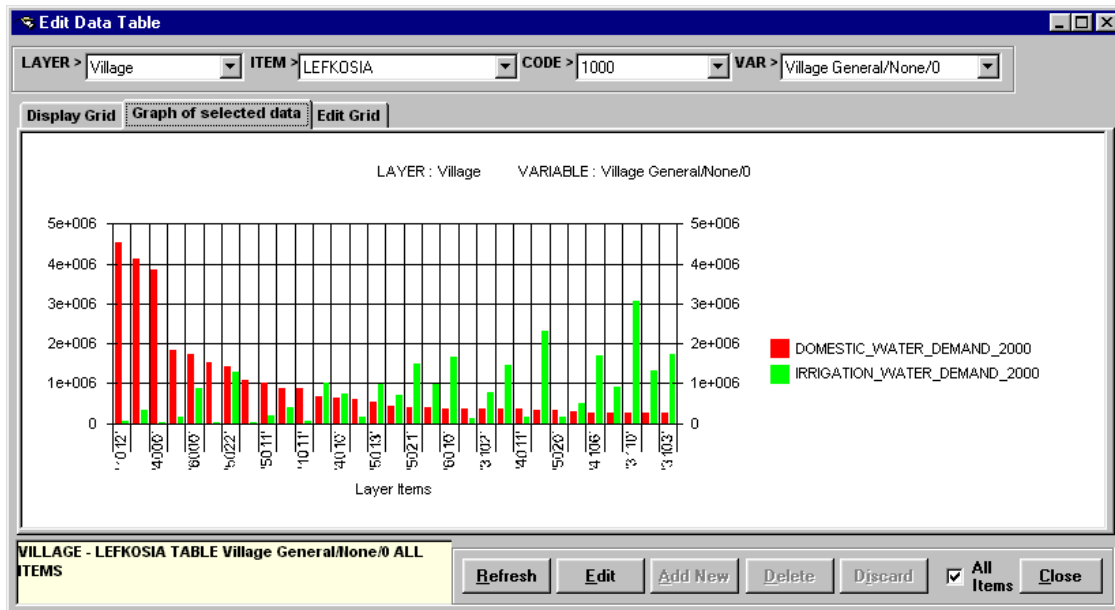
VILLAGE - LEFKOSIA TABLE Village General/None/0 ALL ITEMS

Refresh Edit Add New Delete Discard All Items Close

In order to get the information chose the right layer and item name from the lists on the top of the window. Then click the <Refresh> button. You could choose to have one or several items' information in the same grid. If you want to see all the items' information check the box <All items' below the grid. Notice that in this mode you could not edit the data, but you could build interesting charts based on several layer items data values.

The visualizing panel is the first one. The second is the chart panel. The editing panel is the last. So when you don't want to edit you are using only the first two panels. Otherwise click on the button <EDIT> to get into the editing panel.

While visualizing, if you work with Tables series, you can select a rectangular area within the visualize grid (first panel). The selected area will appear as a column graph in the second panel. So before you select, it's possible to group some columns that you are interested to have in the graph, using drag and drop operation. This could be achieved clicking with the mouse on the first row and dragging the column to its new destination. Before selecting you could also sort the column content clicking twice at the first (with captions) row.



While editing, don't fill or edit the first column. This is valid for all the cases. This column is storing the internal id of the object to which the data is related. The program fills it automatically.

Because of the differences between regional parameters describing the decimal separator symbol, writing numbers with decimal point should be done from the right side of the keyboard (numeric keypad).

To view attached document simply click twice on its name in the cell. The cells with attached documents have captions DOC_NAME, IMAGE_NAME or OTHER_NAME.

When you edit Attachment data type table, you can insert a new document after clicking twice with the mouse in the cell with caption DOC_NAME, IMAGE_NAME or OTHER_NAME.

This is bringing a "File Open" dialog box in which you have to choose the right document to be sent to the file (http) server. Before sending the document don't omit to fill the other fields as description and type. Their content is optional.

Edit Data Series

Editing of data series could be achieved in the tool Linear Graph and through the specialized menu entry "Data Processing" – "Edit Data Series"

The window for editing is very similar to that for editing tables and table series.

ID_OBJECT	DAT	VALU	REP
20	01/01/01	1	0
20	02/01/01	2,1	0
20	03/01/01	0,9	0
20	04/01/01	0,9	0
20	05/01/01	0,7	0

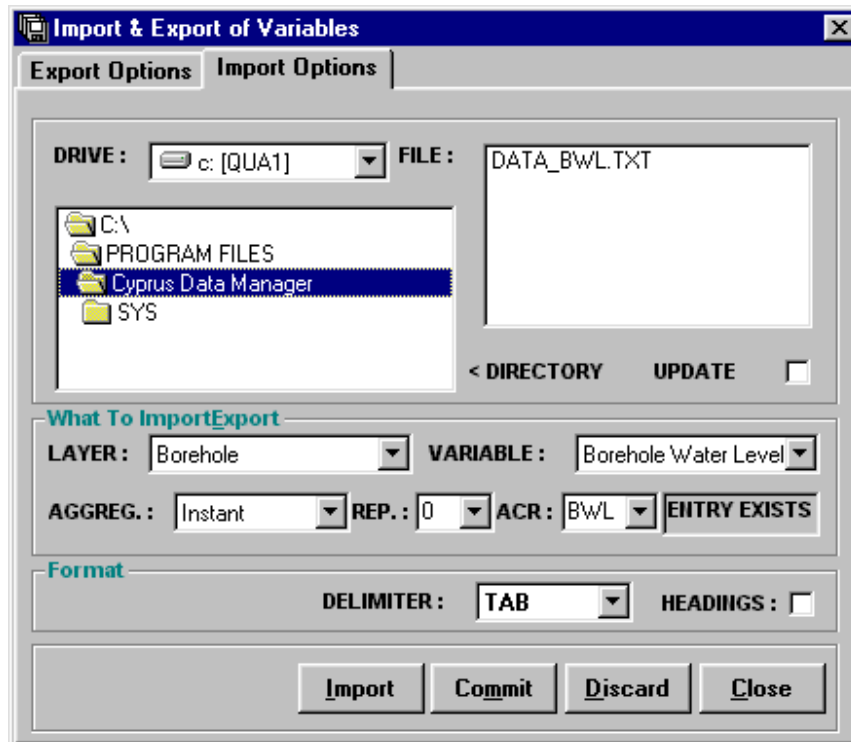
However, there are few differences:

- The user could work on the whole period of available data or on one year. The edited year has to be chosen from the list <Year>
- It is not possible to merge several items' data in one grid for editing
- Because of the differences between regional parameters describing the decimal separator symbol, writing numbers with decimal point should be done from the right side of the keyboard (numeric keypad).

Data Import

To import new data series or other data in the database:

- Prepare the data in a text file separated by the chosen delimiter. The first column has to be the code of the item to which the data is attached. In case of data series or Table series the second column should be a date field. Take care that the last column is always filled with data or is empty string between two tab characters.



- Choose the right Layer name, variable name and aggregation.
- Choose the right file name from the list <FILE>
- Press the <Import> button
- If the row combination CODE + DATE exists in the database the user will be warned for duplicated data and only the not-duplicated rows will be imported.
- In case the user wants to change the data existing in the database, he should use the <Update> check box.

Data Export

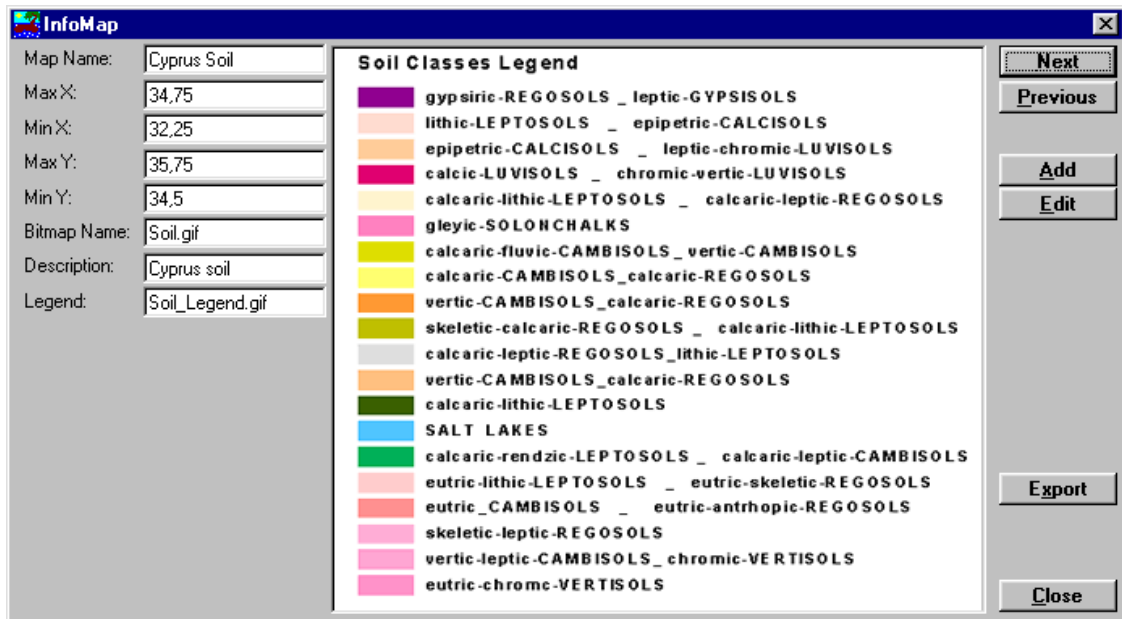
The exporting tool provides a full set of facilities:

- To add in the exported file the coordinates of the points together with the item's code and the values of the data. In case of data series the date is immediately following the coordinates or the code.
- To get all the included in a layer's parent items in the exported data check the <All Parent> box and press <Add to List>.

- To make a list of all the items contained in the database without regard which is the parent, hold Shift key and press <Add to list> button.
- To get all the data inside a time period specify the start and the end in the <First Day> and <Last Day> boxes. The selected period will include all the days from the first and last months, so it is not necessary to choose a particular day. If you want all the data without regard of the period check the <All> box.
- In case when the export list is empty, the exported data will contain all the items data about this variable.
- Press <Export> button to make the export file.

InfoMap table Editing

To add a new raster image to the database click the <Add> button. Write the coordinates that are written above and then drag and drop from the explorer window the name of the file for the raster image.



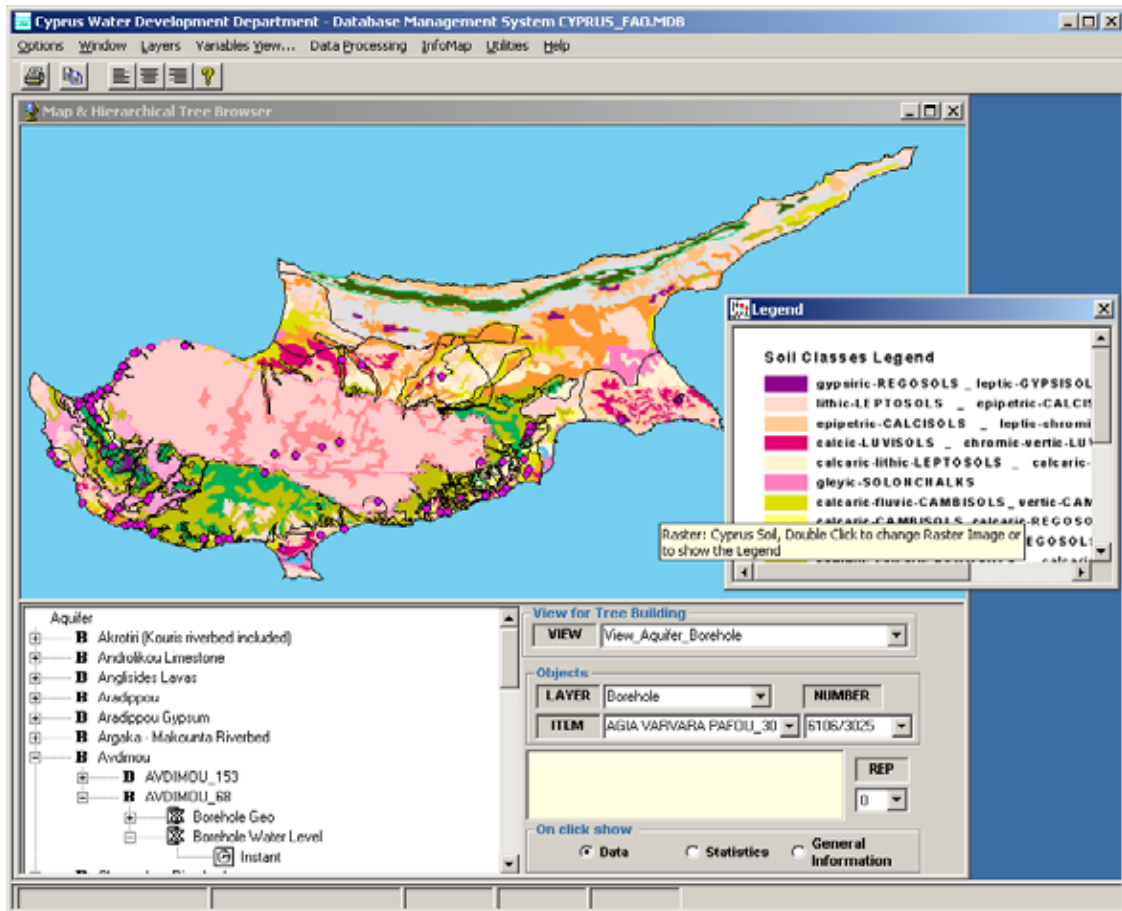
To add a legend to this raster background image drag and drop from the explorer the name of the file with the legend image.

Type the names of the background raster and of the legend in the description fields: <Map Name> and <Description>.

Click the <Update> button

Using Map Access to the Data

Using the Map access to the data is most often intuitive and will work for Data Series and other types of data. Choose the right information you want to get from the <On click show> frame. Then click on the map to get the object's ID and open its tree branch. Clicking on the tree twice will bring either the data window of statistics window or the general information window.



Database administration tools

The database administration has two main parts:

- Database file creation and care
- User policy maintenance.

The database has to be created only when the system has to be installed for the very first time or for some other reason the entire process of database set-up has to be repeated. In such case the user should provide a path in some of his directories that ends with "FAO_DATABASE\Database". Afterwards he has to open the menu item that is shown below and to press the item "Create New Data Base".



The program will ask for the destination directory. In this moment the user should provide the directory name where the database will be created.

After the database file is created it is necessary to perform the usual backup operations as often as the local administrator decides. The repairing and compacting operations could be performed using the MSAccess user interface or from within the application itself using the “Compact & Repair “ menu item.

The users of the database are:

- The main administrator who has the right to edit, add, delete Layer information and add, edit and delete declared variables. He has also the right to compact and repair the database through the application
- The ordinary users who have the right to add, edit and delete data (data series, attached documents etc)

The password for both users is the same, but the name of the Admin user is acting as a second password, which is allowing having different rights on the objects in the database.

When a new database is created the creator is given the right to become “owner” of the database, so in the further use with the database he could work with layer objects inside. The other users should know the password in order to edit data series or other data, but they don't use a name to access the database.

If the Admin user wishes to change his “User Name” he must use the second panel of the [“Options”](#) window, and once entering with the right Admin user name and password, change it and press <Change Owner to this User button.

Notice that the program asks twice for the user name and password.

The term “parent” is used for a layer that contains other layers. (Ex. ‘Watershed’ contains ‘Subwatershed’; ‘District’ contains ‘Village’; ‘Village’ contains ‘Village Center’). In the same terminology ‘Village Center’ is “child” of ‘Village’ and ‘District’.

View in the database terminology is the result of a query on one or several tables that gives a common view on some of their fields in one table. View could be read only or read write but don't allow parameters. In this case views are automatically created after the layer import putting together all the linked layers from each branch of the belonging hierarchical tree. For example, after having imported Aquifer, Region (hydrological), Watershed and Borehole we should have the following views: View_Aquifer_Borehole and View_Region_Borehole. The first will contain linked together in one table the Aquifer's name and code and the Borehole's names and codes. The second will contain the same for Region, Watershed and Borehole. That means that importation only of Borehole layer into the database is launching the creation of two Views. When a new layer is imported into the database, all the existing *views* are updated and new ones are

created. This could take some time, which is depending on the number of layers in the database.

Inventory queries are also *views* but they are combining data from layer description tables, meta-data tables and tables containing no geographical data, for instance 'Data series'. Those are automatically created (you could make them also by pressing a button) after the layer import. Such queries are Inv_Dam, Inv_Spring, and Inv_Watershed etc.

Procedure is a query that could make some changes in the database tables; sometimes it requires an input parameter.

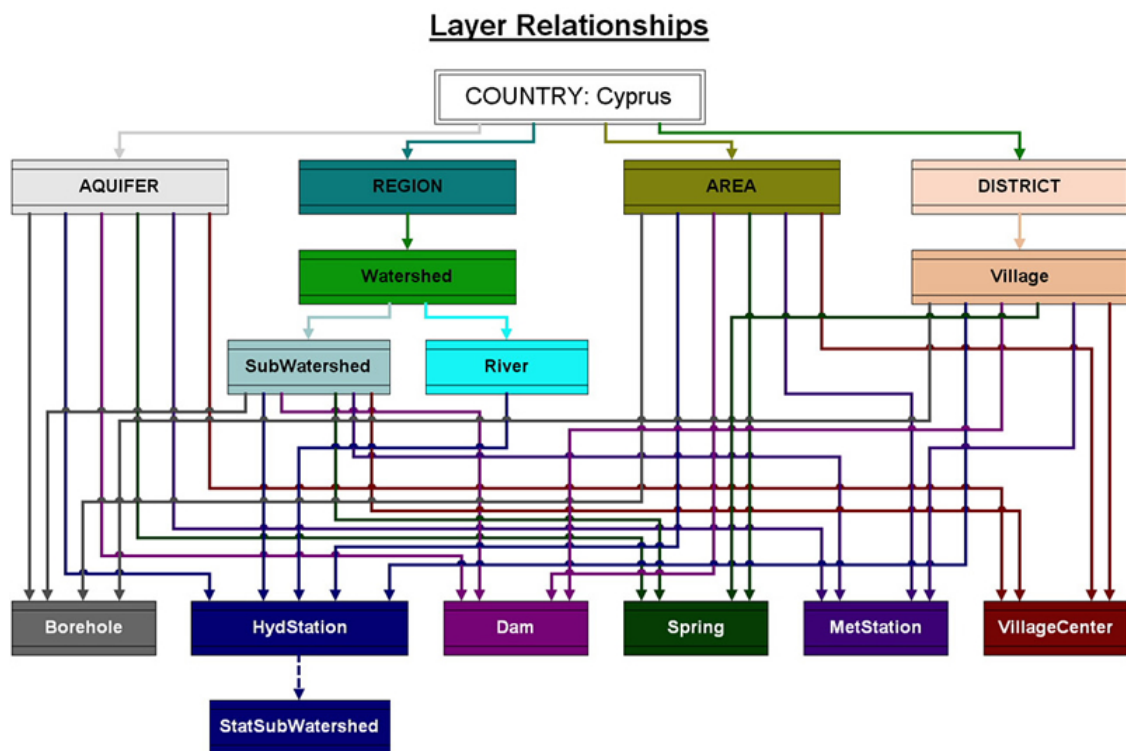
In some particular cases you may want to do full export of your database to ASCII files. There are few steps to do that, but still they require manual operation:

- From the “Layers” > “Layers Management” > “Export Layer” menu chain load the “[Export Layer](#)” window. Then select the name for each layer you have in the database and press the <Export> button. The files names will be &.MID and &.MIF for each exported layer, where & is the layer name.
- From the “Layers” > “[Edit Layer Item's Meta-Data](#)” menu chain load the “Item's Variables Management” Window. For each of your layers that you could choose from the list <LAYER> you should press the <Export> button. The file name is like “VAR_&.TXT”, where & is the layer's name.
- From the “Variables View” > “Variables Definition” > “[Export Definitions](#)”, launch the export definitions routine that will write all your variables definitions in an ASCII file. The file name is [VARIABLES.TXT](#)
- From the “Variables View” > “Data Inventory” launches the “[Data Inventory](#)” window. For each of your layers select and print the upper grid. This is the full inventory of declared variables for the layer.
- From the “Data Processing” > “Import/Export Data” > “Bulk Import/Export” menu chain launch the “[Bulk Import/Export](#)” window. For each of your layers choose the valid variables (from the printed inventories). Then hold <SHIFT> key and press <Add to List > button. Press <Export> button.
- From “Info Map” > “Edit Info Map” menu chain, choose “[Edit InfoMap](#)”. From this window press <Export> button. This will create InfoMap.txt

In the opposite case, when you want to import entire database having already made the full export you should take the actions in this order:

- Import all the needed layers using the “[Import Layer](#)” tool starting from the topmost parent and going down through the [tree](#), but passing carefully each level of “parents” that have following “children”.

- Import the [Variable Definitions](#).
- For each layer use the “[Edit Layer Item’s Meta-Data](#)” tool to import the layer’s meta-data.
- Using the [Data/Import](#) tool for each layer – scroll through the available variables and import the previously exported data file.
- Go to the “[InfoMap](#)” menu item and add the missing raster and legend file names and other attributes that you could read in the exported InfoMap.txt



HydStation_Num	GHS	RAH	SDD	SDM	SDY
1-1-3-95	1	0	1	0	1
1-1-7-95	1	1	1	0	0
1-2-4-95	1	0	1	1	0
1-2-7-90	1	0	1	0	0
1-3-5-05	1	1	1	1	1
1-3-8-55	1	0	1	0	0
1-3-8-60	1	0	1	1	0

The table above presents an example of the [Meta-Data](#) ASCII file that could be read from the program in order to attach several variables to several layer items in the database. One item is identified by one row. The first row contains the layer name (+ the string “_NUM”) followed by the acronyms representing the variables that we want to attach to each item. The acronyms are chosen freely in the moment of the variable creation but here we must supply the right acronym representing the needed variables. The rows below contain the id number of the item, as it exists in the database (and in MapInfo application) followed by 1, if we want to link the variable corresponding to the acronym or 0 in the opposite case. The columns must be separated by tab character. If you prepare the file in Excel, use the export (tab separated option) txt option. The example above means that:

- If we used the [<Export>](#) button: that are the variables attached to the items in the HydStation layer.
- If we plan to use the [<Import>](#) button: that are the links we want to create in the database related to the layer HydStation.

The example shows few acronyms for Rainfall Hourly, Streamflow Discharge Daily, Streamflow Discharge Monthly and so on variables. The second column (with acronym GHS) shows that we have geographical information for each of the items in the sample.

The [Predefined set of variables](#) is a similar text file, but without the first column. It has only one row with the acronyms of the variables that we want to attach to one single item, the one that is chosen in the tool windows list [<ITEM>](#). In addition to the acronyms (because of the need to avoid errors) it should contain as a second row the id numbers of the variables. This option is useful when we want to link several variables to one single item without preparing the above grid. The file should have an extension VAR in order to appear in the list. Check the box [<predefined sets>](#) to make the program read the files and fill the list. Then if you click [<Add Variable to Item>](#) button all the variables corresponding to the acronyms below will be linked to the one chosen layer item.

GHS	RAH	SDD	SDM	SDY
1	3	4	6	5

This is an example of VARIABLES.txt file:

```
16,"Geo","Spring
Geo",#NULL#,#NULL#,"None","GSP",#NULL#,"DATA_GSP","Spring",4,"ID_OBJEC
T;GEO;STYLE;REP","INTEGER;LONGTEXT;LONGTEXT;BYTE","4;4;4;1","ID_OB
JECT;REP"
```

```
17,"Geo","Dam
Geo",#NULL#,#NULL#,"None","GDA",#NULL#,"DATA_GDA","Dam",4,"ID_OBJEC
```

T;GEO;STYLE;REP", "INTEGER;LONGTEXT;LONGTEXT;BYTE", "4;4;4;1", "ID_OBJECT;REP"

18,"Data Series", "Borehole Water Level", #NULL#, #NULL#, "Instant", "BWL", "m", "DATA_BWL", "Borehole", 4, "ID_OBJECT;DAT;VALU;REP", "INTEGER;DATE;SINGLE;BYTE", "4;8;4;1", "ID_OBJECT;DAT;REP"

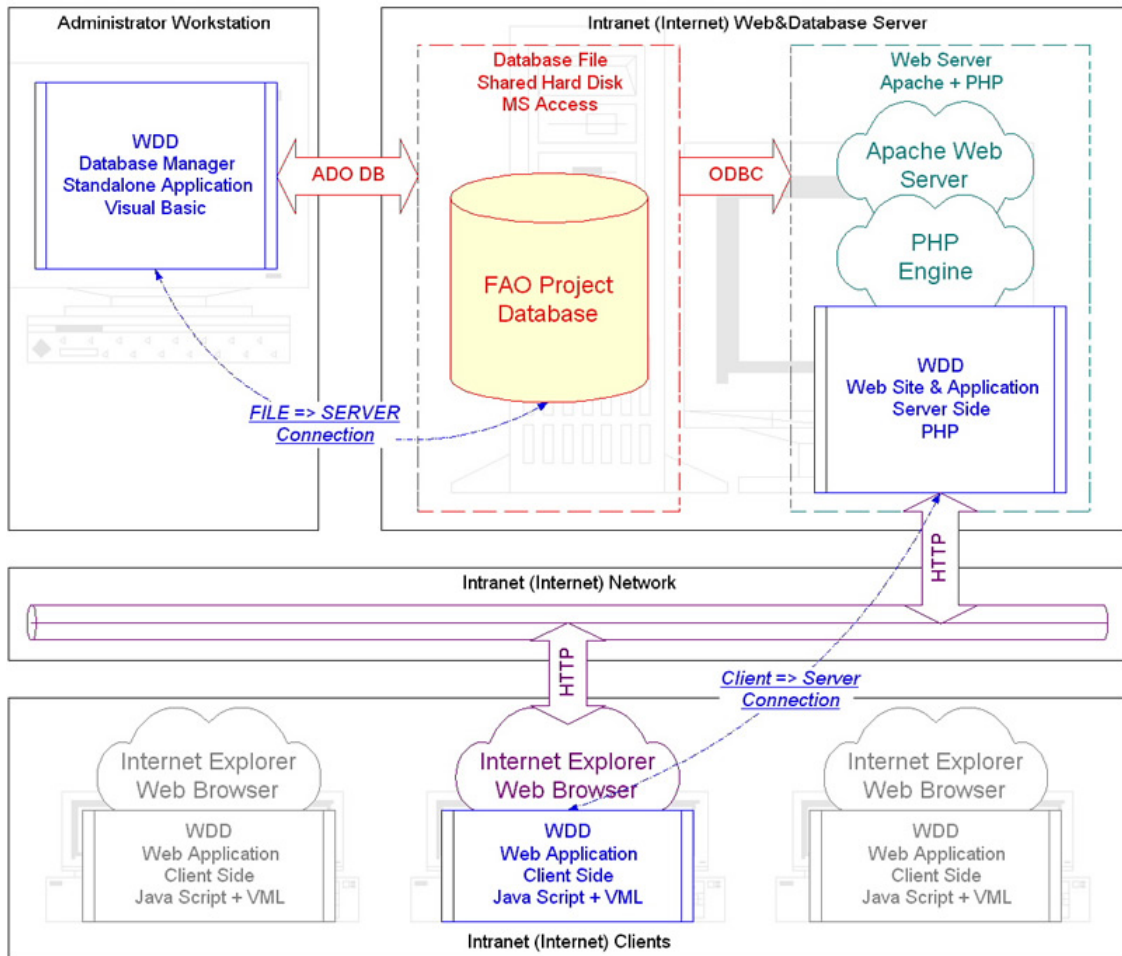
19,"Data Series", "Streamflow Discharge", #NULL#, #NULL#, "Daily", "SDD", "m3/sec", "DATA_SDD", "HydStation", 4, "ID_OBJECT;DAT;VALU;REP", "INTEGER;DATE;SINGLE;BYTE", "4;8;4;1", "ID_OBJECT;DAT;REP"

20,"Data Series", "Water Level", #NULL#, #NULL#, "Hourly", "WLH", "cm", "DATA_WLH", "Dam", 4, "ID_OBJECT;DAT;VALU;REP", "INTEGER;DATE;SINGLE;BYTE", "4;8;4;1", "ID_OBJECT;DAT;REP"

FAO Project Information System Overview

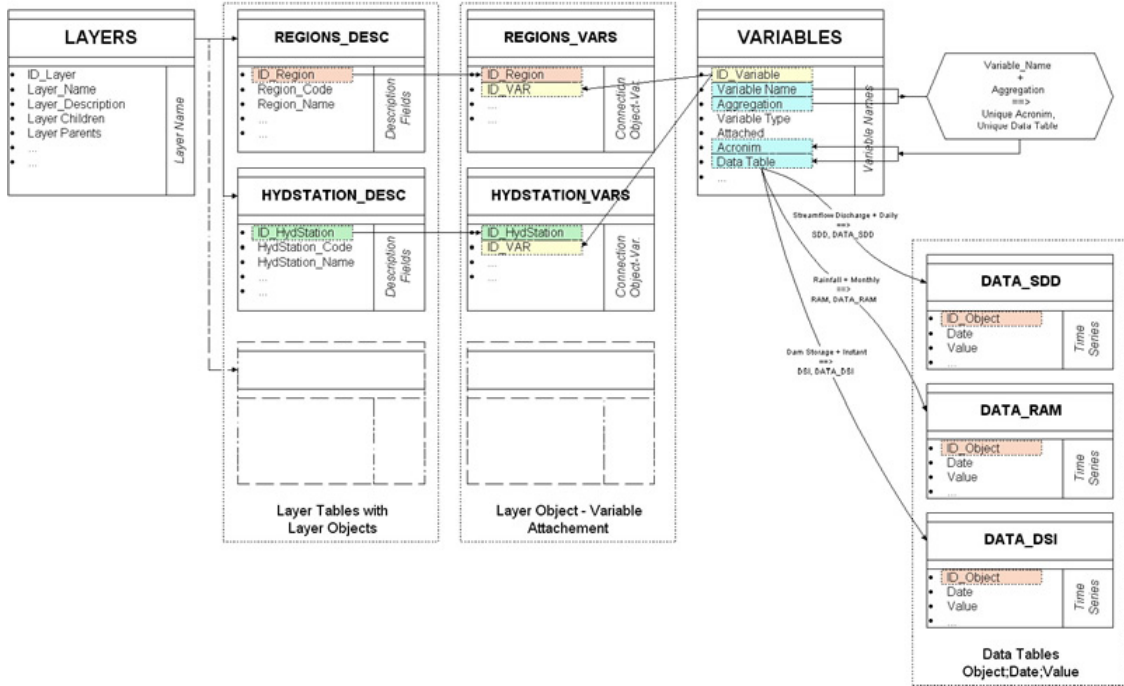
- Three distinct parts compose the FAO Project Information System. The database file (Access format), the Database Manager (CypFAOdbm application) and the Web Database Interface, which is giving the final output for the clients.
- When the database is accessed through a Web Interface Application, which is active on the server, it produces a Client-Server configuration. If the database file is accessed using a desktop application as MSAccess® or CypFAOdbm (Data Manager) it produces a File Server system.

FAO Project Information System Technology Overview



Database Structure

FAO Project Database Structure, Time Series Storage





Web Database Application

component of FAO Project Information System
developed for Water Development Department of Cyprus

Dejan Lekic, B.Sc.

Contents:

List of Abbreviations

1. Introduction
 2. Used Technology Overview
 3. User Interface
 - 3.1. Database Inventory Browser
 - 3.2. Web Access Administrator Panel
 - 3.3. WebMap Application
 - 3.3.1. Multi Tab Frame
 - 3.3.2. Map TabFrame with Map Tools
 - 3.3.3. Table TabFrame
 - 3.3.4. Report TabFrame
 - 3.3.5. Table TabFrame
 - 3.3.6. Chart TabFrame with Chart Tools
 - 3.3.7. File TabFrame
 - 3.3.8. Statistics TabFrame
 - 3.4. Layers Frame
 - 3.5. Selected Object Title Frame
 - 3.6. Inventory Frame
 - 3.7. Full Info Frame
 - 3.8. Status Line Frame
 - 3.9. Title Frame
 - 3.10. Object Relationships Browser Floating Frame
 - 3.11. Custom View Dialog
 - 3.12. Layer Object Edit Dialog
 4. Known Limitations
 5. Acknowledgments
-

Abbreviations

DBIB - Database Inventory Browser, providing low-level access to the database objects, displays general database inventories.

DBM - Database Manager, database application for database maintenance, management and administration.

DHTML - Dynamic HTML, combination of client and server side scripts making a dynamic user-friendly web interface.

FAODB - FAO Project Database.

GIS - Geographical Informational System.

HTML - "Hypertext Mark-up Language", used for Web page creation.

IE - Internet Explorer, Web Browser developed by Microsoft.

MI - MapInfo, GIS desktop application developed by MapInfo.

MIF&MID - Pair of ASCII files with MIF (MapInfo Interchange Format) and MID extensions used as an intermediate step when passing geographical data from MI to the FAODB.

ODBC - Stands for "open database connectivity", set of programs (drivers) providing access to different types of databases.

PHP - Stands for "PHP: Hypertext Preprocessor", a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.

SVG - Scalable Vector Graphics, vector graphics description language, rendering engine has to be installed as a browser plug-in.

VML - Vector Mark-up Language, vector graphics description language, rendering engine preinstalled as part of IE.

WAAP - Web Access Administrator Panel, interface for creating, editing and deleting users of WMA, provides user-level security for database access.

WDA - Web Database Application, client-server application providing remote access to the database using Internet technologies.

WMA - WebMap Application, main database Web application, provides full database access using GIS-like interface.

1. Introduction

Web Database Application (WDA) is a Client-Server type of application that provides remote access to the FAO Project Database (FAODB). Clients are accessing the database using their Internet Browser (MS Internet Explorer 5.0 or higher), without need of installing any kind of software on their computers. This approach provides the information system designer a wide set of possibilities since the code for the application exists on the server only, meaning that all the changes or software enhancements are instantly available to the client users. The application itself is a combination of different technologies, programming languages and already existing software packages interconnected in a system that simultaneously runs on both sides of it, on the Web&Database Server and on the Client computer. The connection made through this application doesn't have the possibility of altering the database itself, meaning that the main purpose of the application is to make the database accessible for numerous remote users either on Intranet or Internet, through user-friendly interface. The application itself consists of three parts: Database Inventory Browser (DBIB), Web Access Administrator Panel (WAAP) and the main WebMap Application (WMA). The first part doesn't require user logging since it just gives the view on the database tables and connections between them, currently stored in the database. Using DBIB users can easily browse through all the available database views, but they cannot access the data itself in any form. For the direct data access, the database administrator has to create a particular user, with his proper username and password that will be used latter for logging in on the WMA. The third, main part is a fully featured Web application, combining features from different types of applications such as GIS, Spreadsheet and other common Windows packages. User is provided by several interfaces with features such as Object and Layer selection, manipulation and reordering as well as direct access to the inventories of the Data attached to Objects. Final result of user interaction can be defined as a creation of the Data view in common formats like tables, graphs and data statistics. Data export is available as well, for ASCII and MS Excel file format in a form of data array and for MS Word file format as Tabular Report.

2. Used Technology Overview

The technology used for the application design consists of two parts: Server side and Client side. On the Server four components were created or installed:

1. Database, MS ACCESS Format, created and maintained through Database Manager (DBM) other software component built as a part of the Information System improvement, version 2000, created by Mr. Eram Artinian;
2. Apache Web Server (www.apache.org), leading Open Source Web server, version 1.3.22, installed;
3. PHP engine with precompiled ODBC module (www.php.net), one of the leading Server Side programming languages, version 4.0.4, installed;
4. WDA, set of PHP server side scripts and HTML files used to make the application interface, connect to the database and respond to the events invoked by client user.

The Client can be defined as so called Thin Client since the application doesn't require any kind of user or administrator intervention on the Client computer. The idea of using Thin Client imposed the need of acquiring the information about client computers in the Local Area Network. Presumption that the Client users will have one of the available Windows platforms has been made on the basis of the fact that currently all of them are using Windows 98 or Windows 2000. The fact that Internet Explorer Web Browser is usually installed together with operating system itself led to the choice of the following application structure on the Client Side:

1. Internet Explorer (IE), version 5.0 or higher, preinstalled as part of Windows OS;
2. Java Script engine, version 1.2, preinstalled as part of IE;
3. VML (Vector Markup Language) rendering engine, preinstalled as part of IE;
4. WDA, Java Scripts and HTML files (known as DHTML) used to make the application interface, providing client side interactivity with or without connection to the server.

It is important to emphasize that all the remote users can access the data in the database using WDA on the real-time basis. In the moment when the database Administrator, using DBM, updates, deletes or adds the data or any other kind of attachments, object or layer, these changes will be immediately available to the client users. All the interfaces of the application are built in-fly, meaning that they follow the settings on the user computer such as screen resolution, chosen parameters for data

display, etc. *Image 1.* gives a complete overview on the Information System technology that has been applied.

FAO Project Information System Technology Overview

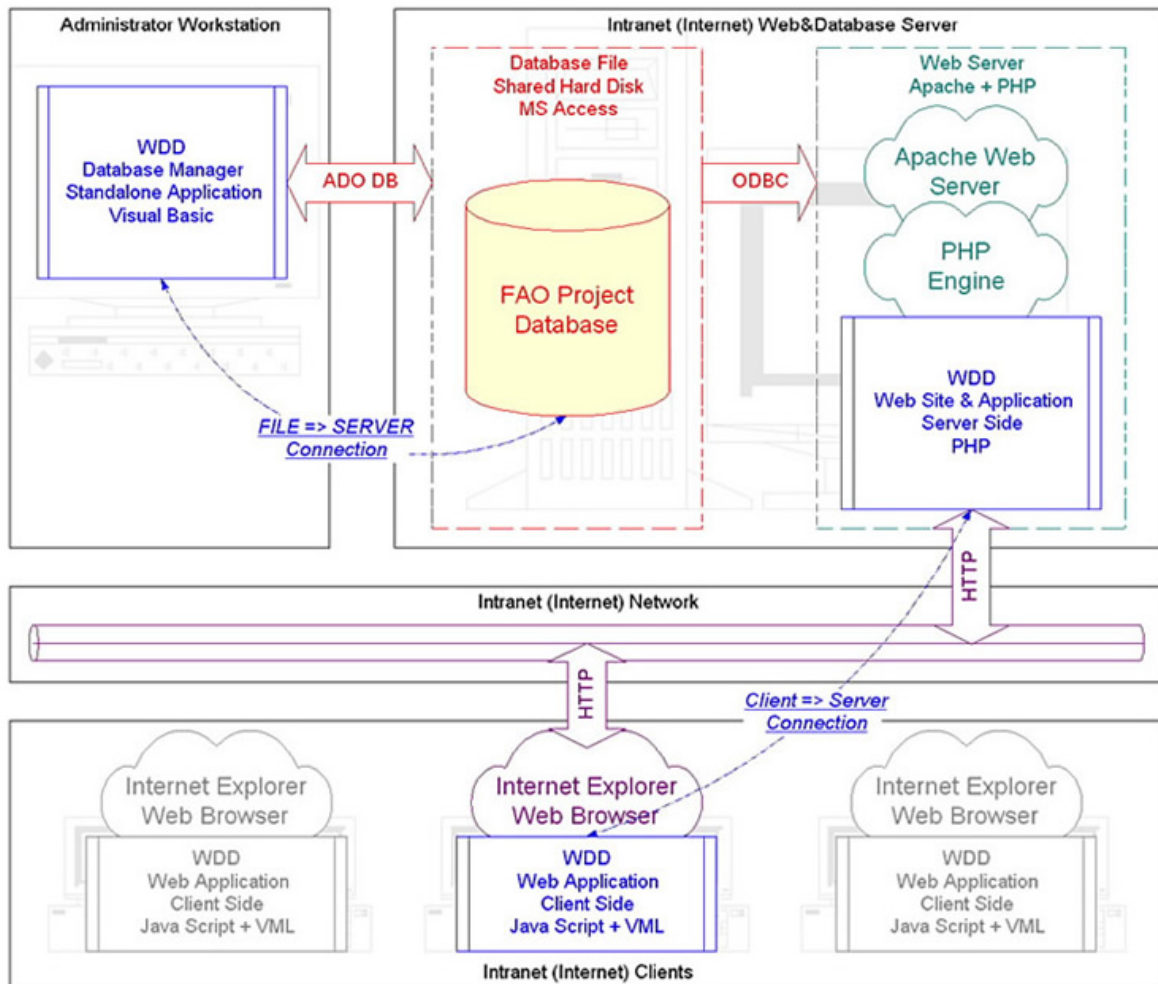


Image 1. Information System Technology Overview

3. User Interface

3.1. Database Inventory Browser

Database Inventory Browser (DIB) represents the simplest way of accessing the database. It gives the general overview on the objects and the data stored in the database. The interface of DIB consists of three frames (*Image 2*).

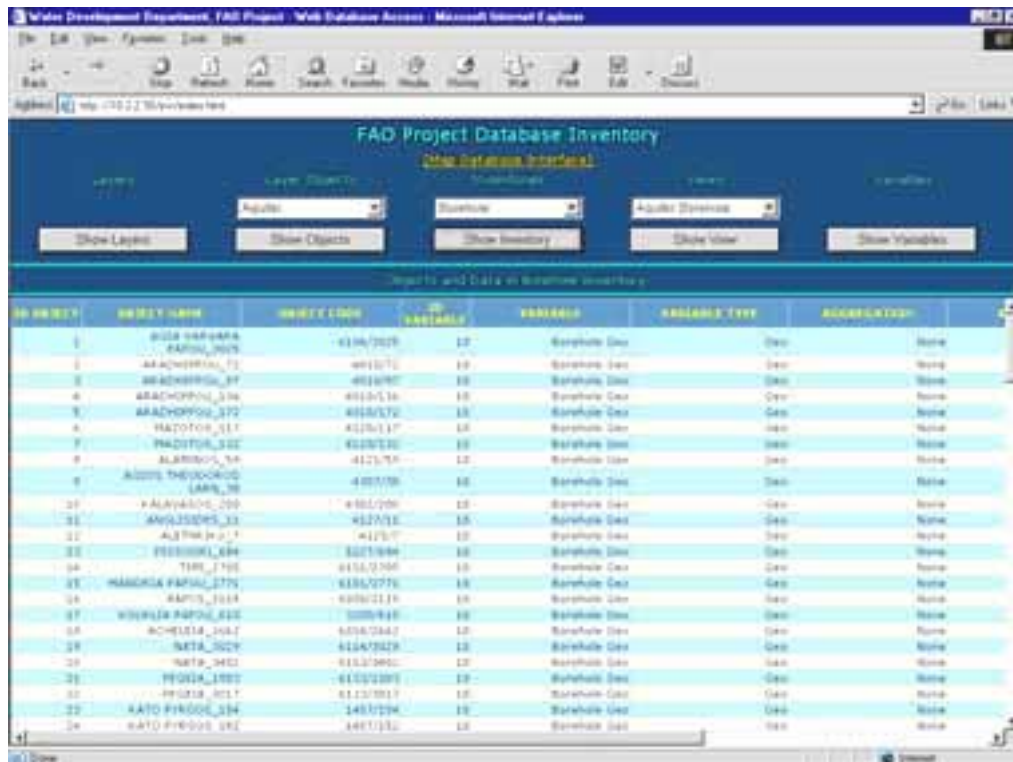


Image 2. DIB Full Screen Shot

On the top of the screen five buttons are situated, some of them accompanied with drop-down boxes (*Image 3*).



Image 3. DIB Interface Detail, Buttons & Drop-down boxes

Each of them invokes generation of the table in the lowest frame, giving a view to the chosen database set of information. Proper title of the table follows the procedure in the middle of the interface. Five buttons are:

- **Show Layer**
Provides access to all currently available Layers (in term of MapInfo Layers) as well as to their relationships and other information;
- **Show Objects** (requires selection from drop-down list)
After selecting one layer name from the drop-down box, creates view on the Description table of the chosen layer. Each layer has it's own description table so the fields displayed as header of the table depend on the user's choice.
- **Show Inventory** (requires selection from drop-down list)
Makes the query on the database, searching for the connection between the Objects in chosen Layer and the available Variables in the database. The resulting table shows only the Objects that have some sort of attached Variable.
- **Show View** (requires selection from drop-down list)
Drop-down box contains the list of available views automatically created by DBM software. The views are created as tabular representation of the Layer Relationships Scheme (available later in the help file as *Image 14*).
- **Show Variables**
Lists all currently available variables in the database. This list includes variable name, type (Geo - Geographical Coordinates, Data Series, Attachment, etc.), aggregation level, unit and the other.

Typical result displayed in the bottom frame gives a tabular representation of a query made as a response to the choice made by user on the upper frame. As an example, *Image 4* gives a result of database querying with Borehole + Inventory combination.

ID OBJECT	OBJECT NAME	OBJECT CODE	ID VARIABLE	VARIABLE	VARIABLE TYPE	AGGREGATION
1	AGIA VARVARA PAFOU_3025	6106/3025	10	Borehole Geo	Geo	None
2	ARADHIPPOU_72	4010/72	10	Borehole Geo	Geo	None
3	ARADHIPPOU_97	4010/97	10	Borehole Geo	Geo	None
4	ARADHIPPOU_136	4010/136	10	Borehole Geo	Geo	None
5	ARADHIPPOU_172	4010/172	10	Borehole Geo	Geo	None
6	MAZOTOS_117	4120/117	10	Borehole Geo	Geo	None
7	MAZOTOS_132	4120/132	10	Borehole Geo	Geo	None
8	ALAMINOS_59	4121/59	10	Borehole Geo	Geo	None
9	AGIOS THEODOROS LARN_38	4307/38	10	Borehole Geo	Geo	None
10	KALAVASOS_200	4302/200	10	Borehole Geo	Geo	None
11	ANGLISIDES_11	4127/11	10	Borehole Geo	Geo	None
12	ALETHRIKO_7	4125/7	10	Borehole Geo	Geo	None
13	PISSOURI_684	5227/684	10	Borehole Geo	Geo	None
14	TIMI_2705	6101/2705	10	Borehole Geo	Geo	None
15	MANDRIA PAFOU_2771	6101/2771	10	Borehole Geo	Geo	None
16	PAFOS_2115	6000/2115	10	Borehole Geo	Geo	None
17	KOUKLIA PAFOU_610	3200/610	10	Borehole Geo	Geo	None
18	ACHELEIA_2662	6014/2662	10	Borehole Geo	Geo	None
19	NATA_3029	6114/3029	10	Borehole Geo	Geo	None
20	NATA_3402	6113/3402	10	Borehole Geo	Geo	None
21	PEGEIA_1803	6133/1803	10	Borehole Geo	Geo	None
22	PEGEIA_3017	6133/3017	10	Borehole Geo	Geo	None
23	KATO PYRGOS_154	1457/154	10	Borehole Geo	Geo	None
24	KATO PYRGOS_152	1457/152	10	Borehole Geo	Geo	None

Image 4. Borehole + Inventory Database Query result

Headers of the table displayed on the screen are sensible on the user click, sorting the table on the chosen column name. This kind of presentation is used to give a view on the database inventory without

using the complex interface as in WMA. However it can be used as a first step for understanding the database relations, structure and as an index of available items in it. The tables produced by this application, after copying and pasting in appropriate software can be also used as draft reports on the database contents.

3.2. Web Access Administrator Panel

For the administration of the Web access through the WebMap application, Web Access Administrator Panel has to be used (*Image 5*).



Image 5. Web Access Administrator Panel

In order to get the Access Panel Interface the FAODB Administrator has to log in with the special user name and password, different from "normal" users. That "Administrator" password will provide him the ability to create, edit and delete users as well as to observe current and past user activity using WMA. The Panel consists of five frames, changeable using tabs on the top of the Panel. These five tabs are:

- **Users**
Listing all the users currently registered for the database access using WMA. Users are presented in a tabular form with attributes such as Name, Family Name, Position and others. User information can be edited and deleted using this frame by clicking on and images respectively. Deletion from the list will disallow any future database access for deleted user.
- **Log**
Log panel is actually the viewer of the log database table. Using

calendar object, FAOBD Administrator has the access to the listings of the users that have logged in for the selected date. As shown on the [Image 6](#), the report contains columns such as User name, Log in date, time, IP address of the remote computer, etc. Mouse Click on the column headers sorts the table on the chosen column.

FAO Project Database Administrator Panel

Users Log New User Administrator Log Off


Logs List

December 2001

Mon	Tue	Wed	Thu	Fri	Sat	Sun
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

NAME	FAMILY NAME	LOGIN DATE	LOGIN TIME	IP NUMBER	BROWSER
Adonis	Georgiou	12-05-2001	14:20	10.2.2.217	Mozilla/4.0 (compa
Adonis	Georgiou	12-05-2001	13:55	10.2.2.217	Mozilla/4.0 (compa
Guest	Guest	12-05-2001	10:41	10.2.2.211	Mozilla/4.0 (compa
Dejan	Lekic	12-05-2001	21:22	10.2.2.223	Mozilla/4.0 (compa
Dejan	Lekic	12-05-2001	09:08	10.2.2.211	Mozilla/4.0 (compa
Dejan	Lekic	12-05-2001	10:02	10.2.2.211	Mozilla/4.0 (compa
Dejan	Lekic	12-05-2001	11:26	10.2.2.211	Mozilla/4.0 (compa
Dejan	Lekic	12-05-2001	11:29	10.2.2.211	Mozilla/4.0 (compa
Dejan	Lekic	12-05-2001	12:02	10.2.2.211	Mozilla/4.0 (compa
Dejan	Lekic	12-05-2001	11:51	10.2.2.211	Mozilla/4.0 (compa

Image 6. Log Frame (with 5th December 2001 as a selected date)

- New User**
 Provides the Form for registering a new user for WMA database access. Note that the new user cannot be added without filling four requested fields: Name, Family Name, Username and Password. Also, these fields cannot contain characters other than alphanumeric + underscore. The minimum length for username and password field is set on four characters.
- Administrator**
 Frame has exactly the same layout as Users frame with the difference that it is used to organize Administrator's accounts. Same features as for Users frame are available.
- Log Off**
 Log Off frame for WAAP application; the same result can be made if the Administrator closes the window by clicking on  button on the upper right side of the frame.

3.3. WebMap Application

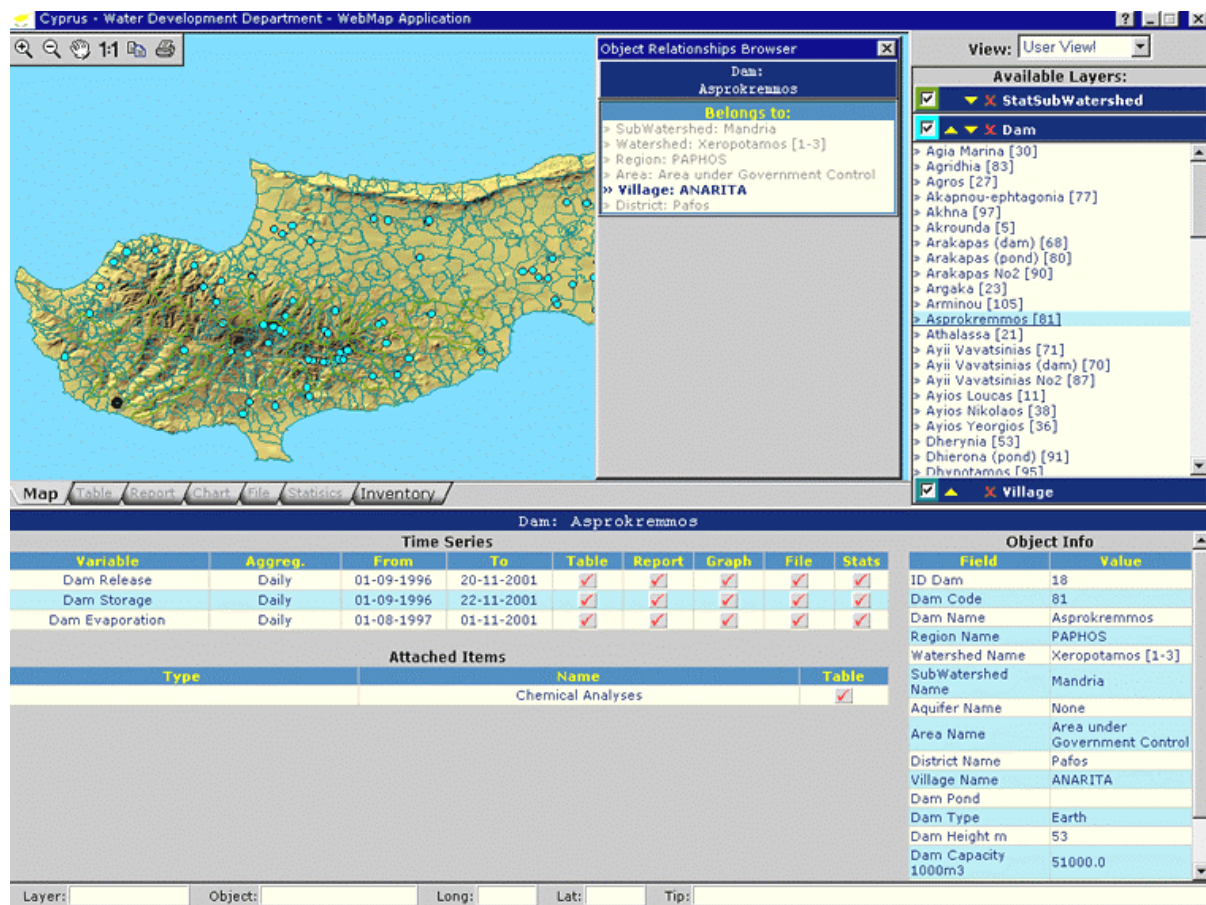
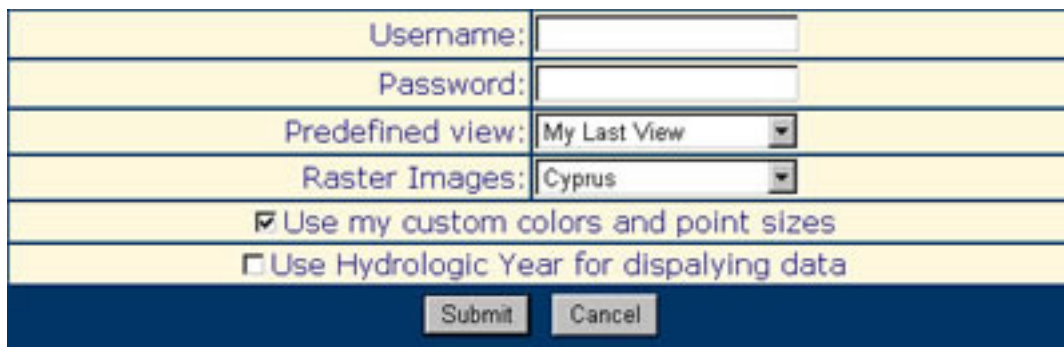


Image 7. WMA user interface

The main application of WDA is WebMap Application (*Image 7*). It is an application based on the idea that the easiest way for common user database access is through the Map interface, showing the geographical representation of the Objects in the database. For Map creation purpose a special kind of procedure has been created. First part of the procedure is implemented in the DBM software, making a geographical data transfer from existing MapInfo files (MIF&MID, MapInfo Interchange Format - ASCII file exported from MI) to the previously prepared data tables. These MI files, defined as one layer per file, are stored in FAODB in two ways. While the first one just represents a differently organized copy of MIF&MID files, the other one is a special kind of Vector Image format, known as VML (Vector Markup Language).

VML is one of the standards (the other one, SVG - Scalable Vector Graphics, proposed from W3 Web standards consortium is still available only as a plug-in) currently accepted as a technological solution for displaying vector images inside the Web Browser and IE installation package includes the rendering engine for it. The second procedure, part of WMA, recreates the Map representation in a form of Layers containing geographical Objects, attaching user driven events to them.

Before using the application the user has to provide valid user name and password on the Log In page. In case that user forgets his logging parameters, the database administrator should be contacted and asked to provide a new combination of username and password. On the logging screen two drop-down menus are available as well as two check boxes (*Image 8*).



Username:	<input type="text"/>
Password:	<input type="password"/>
Predefined view:	My Last View
Raster Images:	Cyprus
<input checked="" type="checkbox"/> Use my custom colors and point sizes	
<input type="checkbox"/> Use Hydrologic Year for displaying data	
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

Image 8. Log In page with start up settings

First drop down menu "Predefined view" gives the choice derived from the set of basic layers stored in the database, "My Last View" in case that user already visited WMA, as his last view, or "Custom View" providing the start up dialog window for choosing the list of visible layers.

Raster Image selection is also available, giving a set of background bitmap images georeferenced through the database available as a background of the Map that will be created.

Two check boxes, "Use my custom colors and point sizes" and "Use Hydrological Year for data display" enable customization of the application on the user level. Layer Objects colors could be redefined during application run time and client can reuse his personal settings by checking the appropriate check box. A second check box allows selection of Hydrological instead of Calendar Year for all methods of data display. This feature was specially added as a result of particular needs of the Division of Hydrology.

Submit Button invokes creation of two pages, the page with information about the user stays in the background and a new, full screen size WMA interface appears on the screen.

Generally WebMap user interface can be divided in several parts:

3.3.1. Multi Tab Frame

Initially this frame opens the *Map* Tab with a map containing geographic objects in layers selected by user. Also, at start up time all the Tabs (*Image 9*) are disabled except the first one. They will be enabled as a reaction to user actions, for example, *Chart* Tab will be enabled when

user tries to visualize the data in a form of Chart. Map frame is invoked by default. All other frames are invoked from Object Inventory Frame, except Inventory tab that requires a call form Layer Inventory Frame. Mouse Click on the enabled tabs invokes rising of selected TabFrame. Generally, following TabFrames are available:



Image 9. Tabs at start up time (Map Tab enable and preselected)

3.3.2. Map TabFrame with Map Tools

TabFrame is one of the essential parts of the application. It provides clients with a fully interactive map, reacting on user events, such as positioning mouse over the object, left mouse click, right mouse click, giving tool tips about object under the mouse cursor in case that user keeps the mouse for few seconds over the object. Not only vector objects are active, background image can be also changed and replaced through user intervention. The application handles following user invoked mouse events on the map Objects:

- I. Mouse Over
 - => Partially Selects Object (increases the line width, increases opaque from 0 to 30%) ;
 - => Fills the **Status Line** Frame with Object Attributes (Layer Name and Object Name) and Tip for the user.
- II. Mouse Out
 - => Deselects partially selected Object in previous step;
 - => Empties **Status Line** Frame.
- III. Mouse Move
 - => Fills the **Status Line** Frame with geographic coordinates (Latitude, Longitude, decimal degrees) of the mouse cursor.
- IV. Left Mouse Click (*Image 10.*)
 - => Fully Selects Object (increases the line width, increases opaque to 100%, all the objects behind the selected become invisible);
 - => Invokes opening of corresponding *Layer* block in **Layers** Frame, followed by opening of Objects list;
 - => Selection of the selected Object in the list opened above;
 - => Fills *Object* Inventory in **Inventory** Frame with the information about Variables attached to selected Object;
 - => Fills **Full Info** Frame with the general information about selected Object, obtained from the Description Table.
- V. Right Mouse Click
 - => All the actions described for Left Mouse Click;
 - => Opens **Object Relationships Browser**.
- VI. Mouse Stays
 - => Opens a tool tip with the information about Object under the mouse cursor.

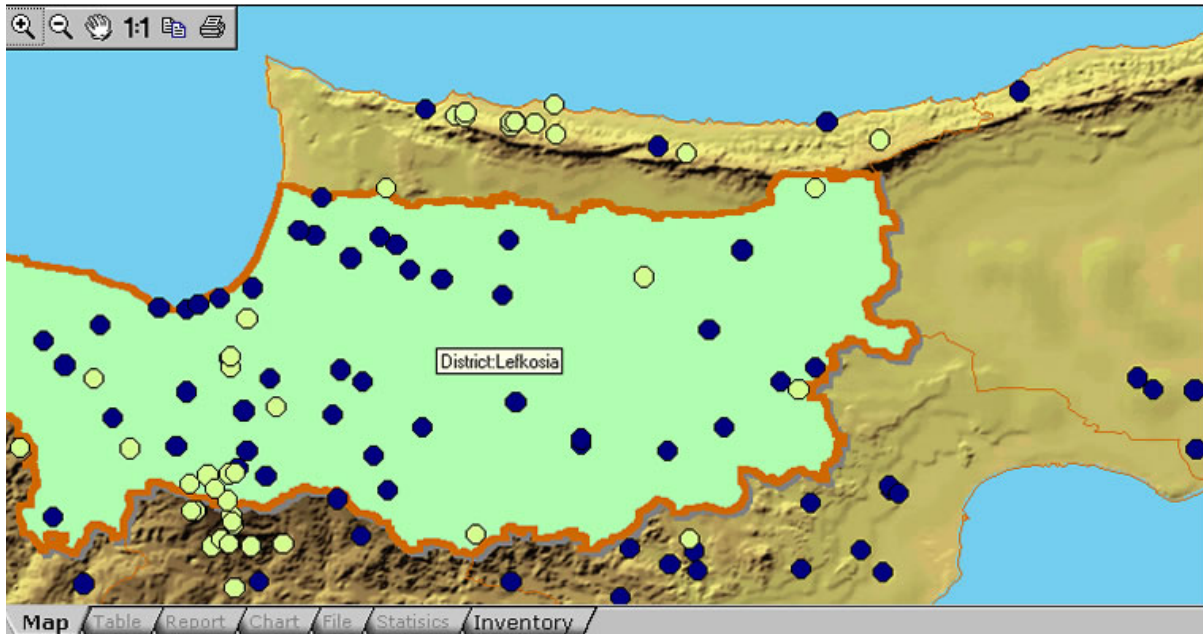








Image 10. Typical user action, zoomed map with Lefkosia District selected

On the other hand, using **Map** Tools, the user has an option to use several tools over the Map. These tools are:

- I.  Zoom Plus
=> Click and Drag on the Map draws a transparent region selected for zooming on.
- II.  Zoom Minus
=> Reduces Zoom level for one step (default zoom counts for level one).
- III.  Move
=> Click and Drag to move the map (zoom level has to be greater than one).
- IV.  One to one
=> Resets Zoom level to default (level one).
- V.  Copy
=> Copies entire Map (not the zoomed view!!!) to the clipboard. Can be pasted in any MS Office application.
- VI.  Print
=> Prints the Map on the selected printer (Landscape paper orientation is suggested).

3.3.3. Table TabFrame

Used for data visualization in tabular form. Consists of two parts, upper containing the Table title and drop-down box for Year selection and lower for table display. Five types of aggregations can be displayed. Yearly monthly and instant with the same feature that the Table frame displays

entire data set, without the need of year selection while daily and hourly aggregation demands year selection in upper part. Simple data statistics like minimum, average, maximum as well as the same values for entire selected period are also provided usually at the bottom of the table. Table can also be copied using built in features of IE such as select+copy or select+print.

3.3.4. Report TabFrame

One of the simplest frames providing just an intermediate step for automatic MS Word format Report creation. User has to pick one year from the list on the top left part of the frame and to press button on the right. New window pops up opening MS Word application embedded inside IE, containing predefined Report Template filled with selected data (*Image 11*).

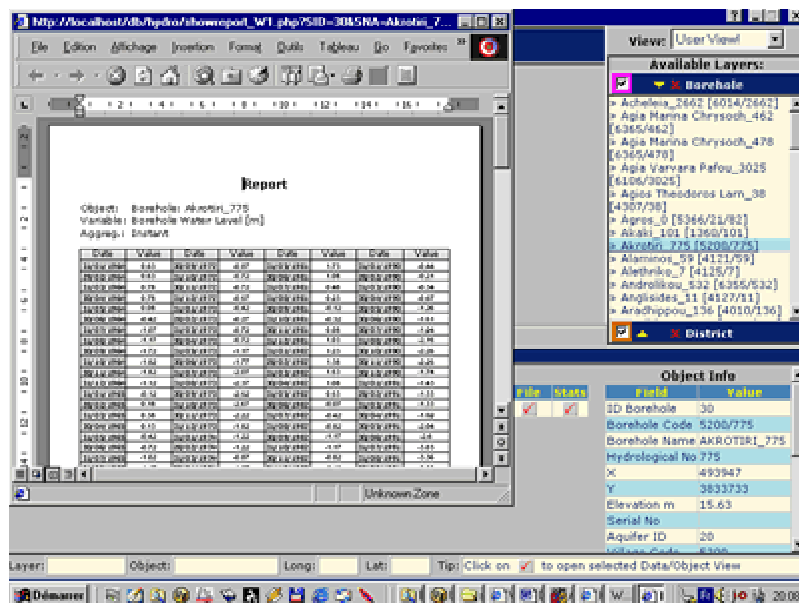






Image 11. Screen shot of the application overlaid with IE window containing MS Word Report

3.3.5. Chart TabFrame with Chart Tools

Frame with maximum features among TabFrames is the Chart Frame accompanied with Chart Tools toolbar. It is a container for graphical presentation of the data, obtained in a process of data selection. Aggregations suitable for this Frame are instant, daily, monthly and yearly aggregation. By default daily and instant aggregation are displayed as line charts while monthly and yearly use bar graph. Anyhow, default presentation can be changed later using toolbar. Displayed chart follows user choice on Hydrological Year vs. Calendar Year, meaning that October is the first month for displayed Hydrological Year in opposition to January for Calendar One. Another interesting feature is available, positioning mouse over the data line or bar displays the date and the value for

underlying chart point. Using this feature user can easily retrieve points of interest.

Chart toolbar give some additional options. In comparison to Map toolbar, several supplementary tools are available. Entire set of tools with their description follows:

- I.  First five tools have exactly the same characteristics as those on the Map Tools.
- II.  Statistics
=> Toggles basic statistics display on the graph.
- III.  Line
=> Switches graph type to Line.
- IV.  Line
=> Switches graph type to Bar.

3.3.6. File TabFrame

File Frame is very similar to the Report frame differing in the fact that it doesn't fill any kind of predefined template for data display. Instead of that it may provide two types of output, flat ASCII file and MS Excel file. Both formats contain data presented as two-column matrix, with the date (time) in the first and the corresponding value of the chosen variable in the second. As for Report Frame, the above mentioned files appear in the newly created window.

3.3.7. Statistics TabFrame

Using this frame it is possible to obtain statistics about the whole selected data series, maximum, average and minimum. For some aggregation levels, such as daily and monthly, data series filling is available as well ([Image 12](#)).

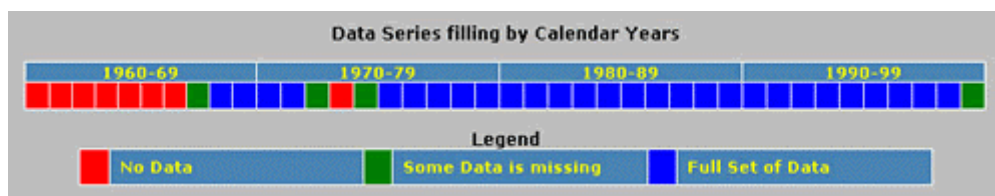


Image 12. Data Series filing form

3.3.8. Inventory TabFrame

Last selectable Tab, Inventory, has a similar way of presentation as Table TabFrame. The difference lies in the fact that it doesn't display the data itself. Instead, it is used for presenting various tabular presentation of Database views, Object description tables, Layer Object Inventories, etc.

Another difference is that the headers of the columns react on user action, ordering displayed table using selected column. Three types of tables fill this Frame: Layer Objects table, Layer Objects Data Inventory Table and Layer Views Inventory table. If a Layer has additional variable of type *Table* attached additional tabular representations will be available.

3.4. Layers Frame

The user tool for manipulating and managing Layers displayed on the map is situated in *Layers Frame*. Its structure consists of three interconnected parts, each of them representing an essential way in a process of Object querying and data selection. Through this Frame the user can add and remove Layers on Map Frame, change their vertical order, switch between predefined Map Views or custom ones, control Layer visibility on the Map and finally generate the list of available Objects for the selected Frame. User interaction with this frame produces results in almost all the other frames. Results and ways to invoke them are explained below as a specific feature of each Frame part.

Some of the components are static, like *View SubFrame*, others are dynamic, created and destroyed as a result of user interaction. These three components are (Image 13):

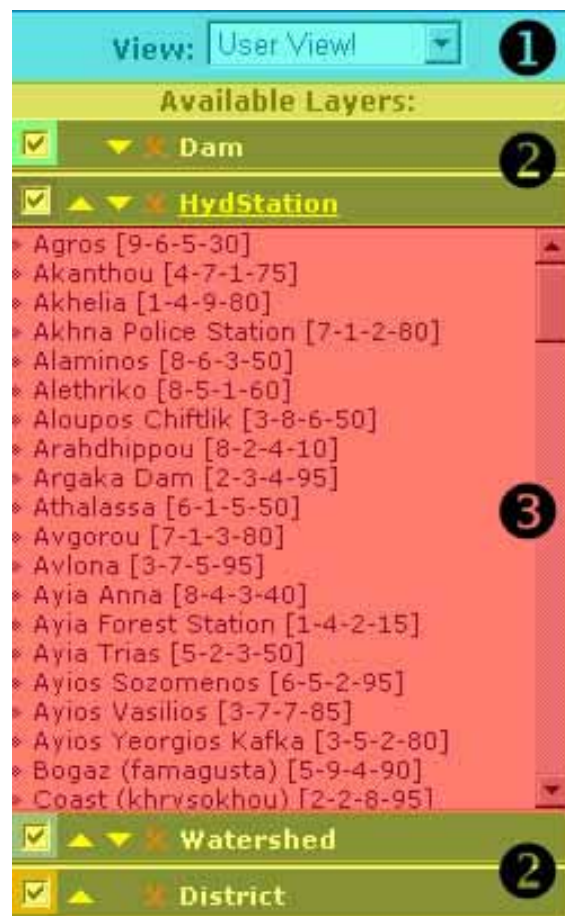






Image 13. Layers Frame with enumerated parts

- **View SubFrame**
 Situated at the top of the parent Frame, contains drop-down list with several choices. It should be used for selection of Layers that will be displayed on the Map as well as for creation of *Layer Blocks* (part 2.). Currently four predefined views are available: Aquifer, Region, Area and District. They are dynamically created, meaning that they are the result of parent/child querying of database structure, presenting the lowest layers on the Map, containing all other layers existing in the database. At the present moment layers parent/child relationships scheme looks like displayed on *Image 14*. Two other choices are accessible: Custom View and User View. The first one invokes opening of *Custom View Dialog*, offering the complete list of layers. The second one is a shortcut to the last user defined Custom View. For example, if user creates his Custom View selecting Area and HydStation layers and later changes it to one of predefined views, he/she can always switch back to the last created Custom view.
- **Layer Block**
 Presented as vertically ordered list of horizontal cells, *Layer Block* contains layers chosen to be displayed on the Map. Each layer is accompanied with several buttons and other interactive elements such as:
 1.  Visibility Check Box
 => Controls Layer Visibility. If unchecked, the layer becomes invisible on the map, its Object List disappears and background color of Layer Block switches to gray.
Important Note: Surrounding color of check box corresponds to the main layer object color on the map. For point objects it's the fill color and for line and polygon objects it is the color of their contour!!!
 2.  Up-Down Image Buttons
 => Vertical Layers order control. Clicking on them moves selected Layer up and down on the map, reorders *Layer blocks*. It is important to stress that, by default order, layers containing polygon objects are created as lowest layers on the map. Putting them on the top may result in loss of reaction to mouse events for the objects in underlying layers.
 3.  Delete Image Button
 => Deletes Layer from the list, deletes Layer from the Map. Major difference from switching layer visibility on/off is in the fact that by pressing this button, the user really removes the layer from the map and the list. To get it back again he/she has to reopen Custom View Dialog and reselect deleted layer.
 4.  Layer Name Link
 => Click on Layer Name produces generation of *Layer Objects List* as well as it invokes opening of Layer Inventory in

Inventory Frame.

Important Note: If Layer Object List is already opened, it will just close the list, it will not open Layer Inventory.

- **Layer Object Lists**

As one of the steps in database querying, *Layer Objects List* is a viewpoint on Objects for selected Layer. It is created on-fly, at the time when user clicks on the Layer Name Link. It contains the list of Objects stored in the database, ordered alphabetically. The list items are colored in two ways, blue when container layer visibility is on and gray in opposite. All the list items also react on user interaction:



1. **MouseOver**
 - => Underlines list item;
 - => Partially Selects Object on the Map (increases the line width, increases opacity from 0 to 30%) ;
 - => Fills the **Status Line** Frame with Object Attributes (Layer Name and Object Name) and Tip for the user.
2. **Mouse Out**
 - => Deselects partially selected Object on the Map in previous step;
 - => Empties **Status Line** Frame.
3. **Left Mouse Click**
 - => Fully Selects Object on the Map (increases the line width, increases opacity to 100%, all the objects behind the selected become invisible);
 - => Selects list item;
 - => Fills *Object Inventory* in **Inventory** Frame with the information about Variables attached to selected Object;
 - => Fills **Full Info** Frame with the general information about selected Object, obtained from the Description Table.
4. **Right Mouse Click**
 - => All the actions described for Left Mouse Click;
 - => Opens **Object Relationships Browser**.




Important Note: Depending on client screen resolution the *Layers* Frame can be overloaded with *Layer* Blocks and some of them will be invisible since the Frame will automatically resize itself, increasing its height, but going under the *Full Info* Frame. In that case *Layer Objects List* will become very short and scrolling will be needed for accessing more than few items. It is recommended that the users with lower screen resolution reduce the number of opened layers in order to avoid this inconvenience.

3.5. Selected Object Title Frame

Provides a title for two frames below. Always contains the name of the selected Object or Layer. User cannot interact with this frame.


3.6. Inventory Frame

Situated below *Selected Object Title* Frame it is used as a container for displaying tables of Layer/Object Inventories. Inventory can be defined as a database generated list of views representing either the relationship between Objects in different Layers (for Layer Inventory) or list of Objects with Data Series, Table or other kind of attachment (Objects Inventory). Lists are displayed as tables with colored rows to facilitate user's view on them. Several kinds of separated lists are created depending on type of selected Layer/Object and Variables attached to it. All the rows are accompanied with two to five cells on the right side containing image button , giving a set of possible steps to user. Clicking on one of these buttons will always invoke some action in the *Multi Tab* Frame – raising of the corresponding Tab (for example:  button is clicked in the Table column => Table frame raises) or appearance of the dialog box asking user interaction. Two types of contents appear in the *Inventory* Frame:

- *Layer Inventory*
Invoked by click on the *Layer Name Link* within *Layers* Frame.
Contains up to three tables:
 1. Layer Properties
Table always contains two rows:
=> MapInfo Layer Definition, click on  in Download column invokes creation and downloading of MIF&MID MapInfo Layer Files that have to be saved on client computer. Procedure for saving and importing MIF&MID files follows:
 - a. In the 1st file download message box choose "Save this file to disk";
 - b. In the save as message box choose the directory to store the mif file in and save the mif file;
 - c. In the 2nd file download message box choose "Save this file to disk";
 - d. In the "save as message" box choose the same directory as before to store the mid files and save the mid file with the same name as the mif file;
 - e. Open MapInfo, choose Table > Import;
 - f. Choose the mif file and click open.
 - g. In the Import into Table box click save to store the MapInfo tab-file
 - => Objects Definition, click on  in Edit column opens *Object Edit* Dialog.
- 2. Layer Inventory
Two rows are always available: MapInfo Layer Definition and Objects Definition. Depending on data attached to the objects of selected Layer two additional tables can be available: Layer Inventory and Layer Views Inventory. Click on  image

button invokes appropriate action, for example creation of selected Inventory in *Inventory* TabFrame;


3. Layer Views Inventory

Contains set of rows dynamically created from the relations between Layers/Objects in the database (*Image 14*). Click on button  to get view on selected table in *Inventory* TabFrame.

- *Object* Inventory

Invoked by click on the Object Name within *Layer Objects* List. Provides a two table view on Object joined with Time Series or Attached Items.

1. Time Series Table

=> Provides a final step for Hydrological and other data access. Columns available for user interaction are: Table, Report, Chart, File and Stats containing  when the appropriate view can be created for selected Object + Variables + Aggregation combination. Click on Image button raises corresponding TabFrame.

2. Attached Items Table

The same as previous one, except that Attached items can be only viewed as a table list inside *Table* TabFrame.

3.7. Full *Info* Frame




On the bottom right side of the application, *Full Info* Frame displays data collected from the Descriptions Table of the selected Layer+Object. The list of displayed fields varies from layer to layer in function of the available description fields in the database. It can be used as a source of Object information that may be selected, copied and printed using IE built in mouse right click menu.

3.8. Status *Line* Frame

Automatically filled Form with five dynamic fields:

1. Layer Field - Displays name of the Layer containing selected Object while the user interacts with Map;
2. Object Field - Displays name of the Object while the user interacts with Map;
3. Latitude - Displays Latitude of mouse cursor while the user interacts with Map;
4. Longitude - Displays Longitude of mouse cursor while the user interacts with Map;
5. Tip - Field containing a helpful advice for each activated frame while user mouse cursor moves over it.

3.9. Title Frame

As in all windows applications, it contains Title of the application as well as buttons for window closing  and minimizing . It also contains help button  that invokes this Help file to appear.

Important Note: For proper WMA functioning the user should always use the Close button for quitting the application!!!

3.10. Object Relationships Browser Floating Frame (invisible on the application start up)

Part of Application that uses all the power of the designed database structure is stored in this frame. For better understanding of the features provided by this frame it is needed to overview the database Layer relations scheme displayed as *Image 14*.

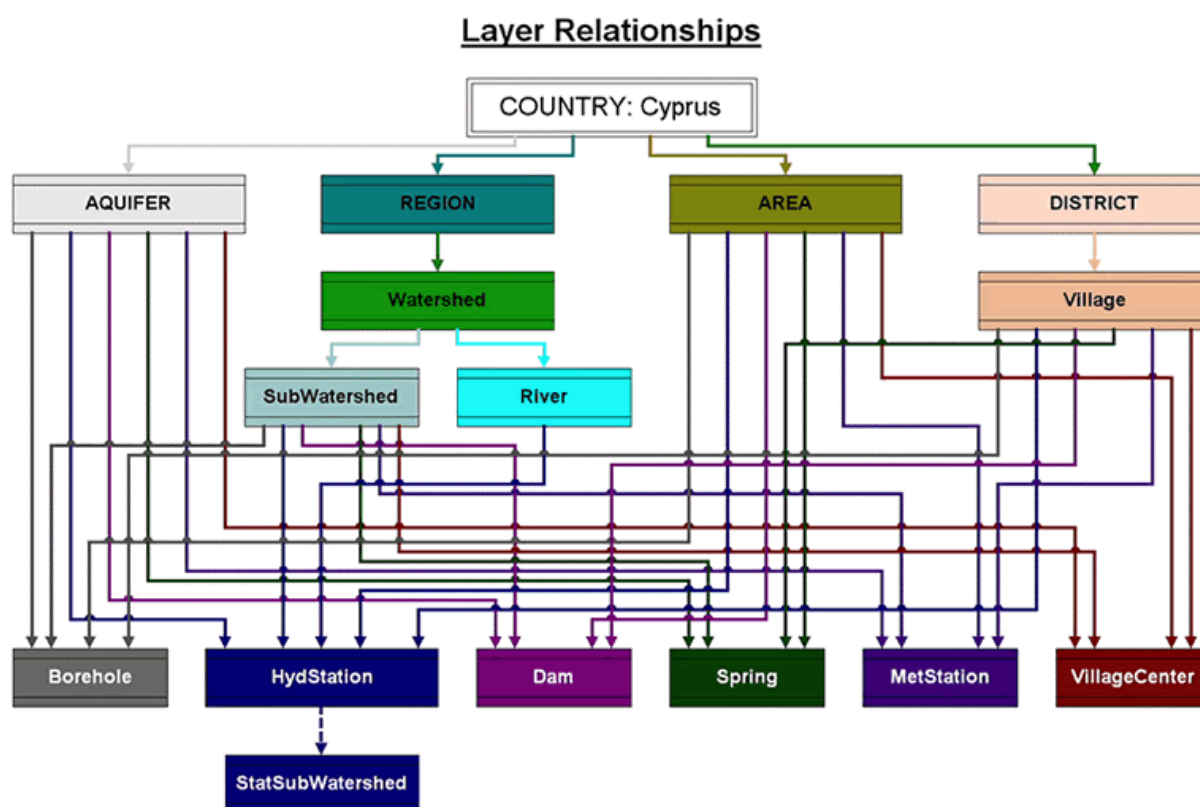


Image 14. Layers Relationships

As displayed on the image, the layer hierarchy starts with one master layer, in this case **Country: Cyprus**. Four basic layers follow it: Aquifer, Region, Area and District. These four Layers have the same characteristic that they are overlaying entire territory of the island (for Aquifers Layer, object None is created in order to fulfil this constraint). The following level

contains Watersheds and Villages then followed by SubWatersheds, Rivers etc. Lines connecting colored rectangles (representing a particular Layer) represent the relationship that can be defined as **Belonging** or **Containing**, depending the starting side of observation. For example, it is obvious that Objects in the Spring Layer can directly belong only to four parent Layers (Aquifer, SubWatershed, Area and Village). At the same time, since SubWatershed Layer is a child of Watershed Layer and Region Layer respectively, it means that automatically one Spring belongs to those two as well. Using this approach all the Objects inside Layers have their appropriate parent or children Objects in other Layers that have a relationship (colored line) as a connection between container Layers.

This feature incorporated through database relations inside the database enabled the creation of a specialized tool for browsing between objects in different layers connected through their Layer's relationships - *Objects Relationships Browser*. At the application start up, this frame is invisible (to avoid application window cluttering) and it can be invoked either by mouse right click on the object on the map or mouse right click on the item in *Layer Objects* List. After the frame creation, the user will have the resulting relationships organized as displayed on Image 15.

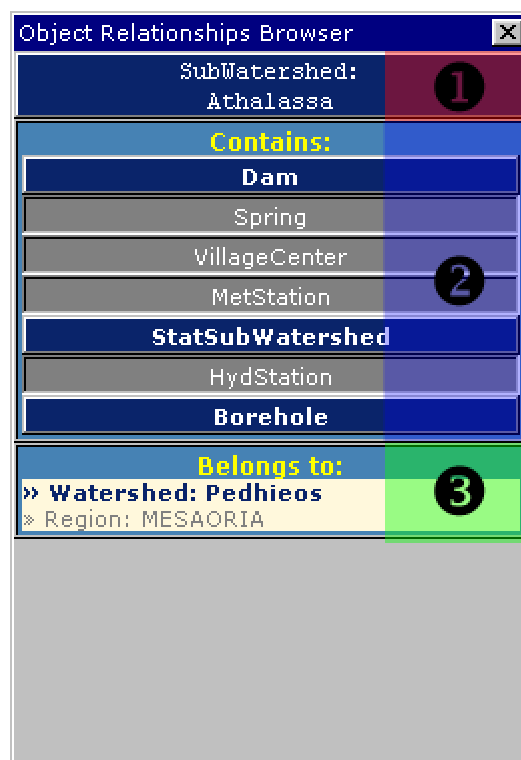


Image 15. Objects Relationships Browser with SubWatershed:Athalassa selected

Three parts can be observed:

1. Title
=> Displays Object Layer Name and Object Name;

2. **Contains** panel

=> Displays all the Children Layers of the selected Layer (derived from selected Object). The blocks are displayed using two colors, blue - for layers currently visible on the map and gray - for layers either invisible or not available on the map. Blocks with Layer names will respond on click by opening the exact list of children (if they exists) in clicked Layer belonging to the Object displayed on the title (*Image 15a*)

3. **Belongs to** panel

=> Display all the parent Objects in a form of a pair

LayerName:ObjectName. Blue color is again used for objects currently visible on the map while gray is used for invisible or unavailable Objects.

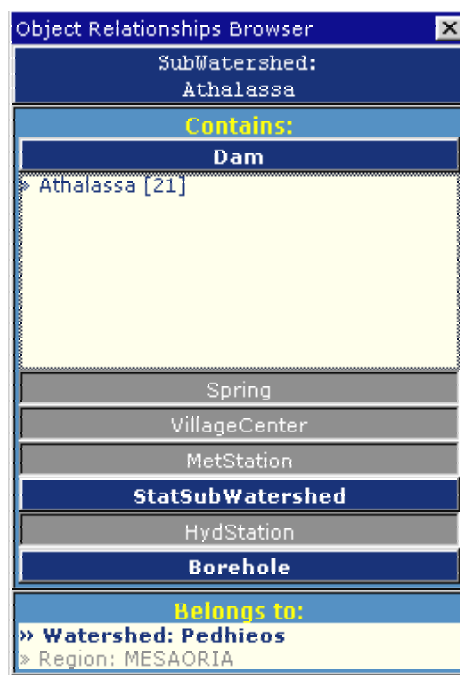


Image 15a. Objects Relationships Browser after opening Dam Tab, showing the list of Dams contained in parent Object SubWatershed:Athalassa

The most advanced feature is that a mouse right click to any Object name within this frame (no matter if user clicks on **Contains** or **Belongs to** list) reopens the same frame, now showing relationships on the next level. By instance, if a user clicked on the Object Name Athalassa (from the list bellow Dam Block, *Image 15a*), the following result will be generated (*Image 15b.*):



Image 15b. Objects Relationships Browser after Dam:Athalassa has been clicked by the user resulting in the list of objects to which Dam:Athalassa belongs to

This process of re-querying the relations between Objects and Layers can be continued as much as needed. By this way the user can get a very specific overview on the Objects stored in the database. This tool can be used for browsing through Layers, following relationship lines, up and down, from parents to children and vice versa. Knowing that, the user can even make a connection between Objects disconnected in the relationships Scheme (*Image 14*). For example the following path can be considered: *select:District - contains:VillageCenter - belongs to:Aquifer*, giving the intersection between selected District and Aquifer obtained at the end.

It is worth to mention that all the Objects Lists inside this frame also react on mouse left click, producing the same result as the same action over *Map TabFrame* or *Layer Objects List*, described above.

3.11. Custom View Dialog

Invoked either from Log In Page or from View SubFrame, provides the list of all available layers accompanied with check boxes. Already selected layers appear with dark blue background while the others appear with gray background color. After making layer selection by checking or unchecking attached check boxes, Submit button leads to the recreation of the Map from the server, with selected layers included.

3.12. Layer *Object Edit* Dialog

For creating user-customized maps, *Layer Object Edit* Dialog is designed. Polygon and line objects can be customized in sense of their fill and line color, while for the point objects even the size can be chosen from one of six levels. The user has to choose from the drop-down list for changing object size or to click on the cell containing one of Objects colors (Fill or Stroke) to obtain color selection pop up (*Image 16*). Pressing on button Apply changes selected attributes of selected Layer Objects on Map and Layer Frame.

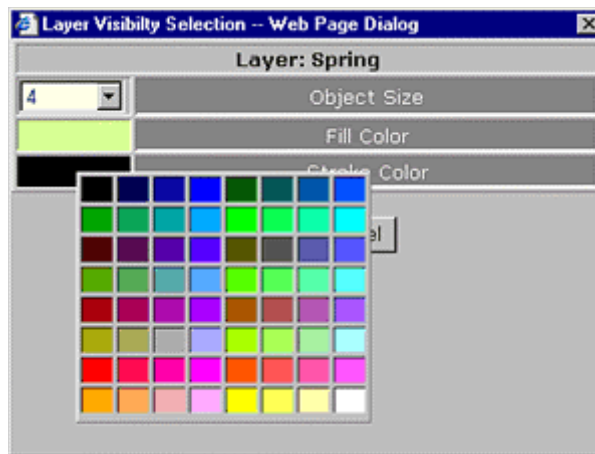


Image 16. Layer Object Edit Dialog with color selection invoked

4. Known Limitations

- Application reaction on user Interaction over map interface strongly depends on the speed of the client computer. Recommended hardware configuration is Pentium 600 MHz with at least 128 MB RAM and screen resolution 800x600 or higher.
- Application start up as well as regeneration of Map during application run time depends on the network speed. As the application is primarily developed for Intranet use, recommended LAN capacity is 100Mbs.
- The Application was tested on Windows 98, Windows NT 4.0 and Windows 2000 Professional as OS and on Internet Explorer version 5.0, 5.5 and 6.0.

5. Acknowledgments

This application is made by:

Dejan Lekic, B.Sc.
e-mail: lekid@beotel.yu
Studentski Trg 17
11000 Belgrade
Yugoslavia

The following people have provided much valuable feedback and ideas during application development:

Mr. Eram Artinian,
Mr. Gerald Dorflinger,
Ms. Marilena Panaretou,
Mr. Iacovos Iacovides,
Mr. Panayiotis Skordis,
Mr. Adonis Georgiou and the other employees of the Division of Hydrology of the Water Development Department.

Special thanks to my wife Maja and to my sister Ana who both provided irreplaceable love, support and care during work on the Application development and my stay in Cyprus.

*Dejan Lekic
Nicosia, December 2001*