Knowledge Media Institute

Visualisation of entity distribution in very large scale spatial and geographic information systems

Jiri Komzak, Marc Eisenstadt

KMI-TR-116

June, 2001

www.kmi.open.ac.uk/papers/kmi-tr-116.pdf

The Open University

Visualisation of entity distribution in very large scale spatial and geographic information systems

Jiří Komzák, Marc Eisenstadt

VISUALISATION OF ENTITY DISTRIBUTION IN VERY LARGE SCALE SPATIAL AND GEOGRAPHIC INFORMATION SYSTEMS

Jiri Komzak, Marc Eisenstadt

(komzaj1@sgi.felk.cvut.cz, M.Eisenstadt@open.ac.uk)

Abstract
The aim of this paper is to summarise entity distribution visualisation problems in very large-scale spatial and geographic information systems. The motivation for this theme has arisen when solving a problem of geographic-spatial information system for Open University oriented on visualisation of geographic and logical distribution of university students and administrative organisation. At the beginning, the paper describes algorithms and data structures used by current geographic and spatial information systems. These are visualisation and mapping techniques, user interface problems, clustering analysis, used data structures and network services. Several existing systems are mentioned as examples. In the second part of the paper, suggested approaches and improvements are discussed. These should provide new possibilities, mainly the scalability of system when working with very large numbers of entities.

Keywords
geographic and spatial information system, scalability, clustering, visualisation, query preview

1 Introduction

Open University is a university with distant education programs. It means that its students are spread out all around the world. This brings some problems, which do not appear at standard universities.

The student society (or community) of such university is similar to cybersociety in several ways. Students do not know one another and do not know, who study programs and interests of other students. Thus, they have no easy possibility to confer their problems with their colleagues. Therefore, it is useful to provide real time search for their colleagues, which are interested in the same problems and are just on-line, and like.

On the other side, the university management needs the overview about university students and administrative organisation. Their visualisation in a map (see figure Fig. 1 for example of implemented distribution visualisation) is very suitable for this purpose and can help for example for better distribution of consulting centres. On the other hand, the display of typical courses studied by students in the same time or specialisation, that they are interested in, can be helpful for a better study organisation.

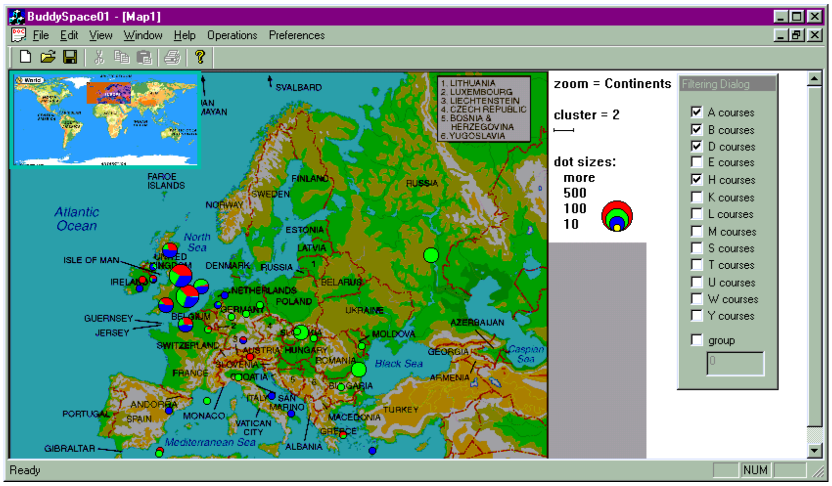

Fig. 1: Buddy Space – student distribution in Europe

The mentioned reasons lead KMi (Knowledge Media institute) of Open University to implement a system providing following functionality:

- student visualisation in a map in different levels of detail
- clustering of students for better lucidity of display and different levels of detail
- visualisation of further administrative information in map with students
- filtration of students according to their courses (and possibly other criteria)
- visualisation of students on a pseudospace given by their interests or courses
- visualisation of student behaviour patterns in time
- search of students according to fuzzy (approximate) queries
- display of on-line students
- visualisation of student communication patterns
- network service (client login) through internet – client-server architecture of the system

These features require implementation of a geographic-spatial information system. This system has to include student database, store used maps, communication log files, administrative university information, and so on.

The system should be built on client-server architecture. While the server will be located in administrative centre of Open University, the client will be a computer used by OU students and staff.

The main problem is scalability of the system, which will not only manage very large amount of data, but also serve huge amount of clients.

This paper is organised into two main chapters, the first one (2) describes currently used techniques and the second chapter (3) explains suggested solutions.

2 Present solutions

When designing a geographical or spatial system, a set of problems has to be solved. This chapter describes present solutions and algorithms used in geographic and spatial information systems. Further, it discusses their suitability for solving our project. It starts with short introduction into what a GIS is and description of its main components. After that, the main problems are described in more detail starting with mapping and user interface. Then the clustering problem is discussed, followed by data filtering and analysis, scalability problems and network services. Data structures used for mentioned techniques are mentioned in the next chapter. Finally, the last part is dedicated to related work.

Spatial and geographic information systems are computer-based tools used for mapping and analysing phenomenon and events. While a GIS deals with geographic phenomenon and events on Earth, a spatial system is not strictly limited on a certain limited problem. It allows working with any data in space and furthermore in pseudospace, which is defined only by a visualisation of some phenomenon. These systems combine common database operations, statistical and other analysis with special spatial analysis and visualisation. Above all, the spatial approach brings new possibilities (like spatial questions answering) and makes the systems more valuable.

Due to their capabilities, the systems can be used in many disciplines, in that it is possible to map concrete spatial problems and take advantages of spatial analysis. Generally, a spatial or geographic information system is understood as a system, which provides these operations: data capture, data management, data manipulation and analysis, and the presentation of results in both graphic and report form. The main goal of a spatial and geographic information system is then to incorporate spatial data, manage it, analyse it, and answer spatial questions.

A spatial or geographic information system can be divided into four main functional subsystems according to just mentioned functions:

- a data input subsystem
- a data storage and retrieval subsystem
- a data manipulation and analysis subsystem
- a data output and display subsystem

Network service is also an important part of information systems. Problems with service of huge amount of clients, network bandwidth, and so on are connected with it.

The systems differ in the level of support of particular functions. While typical geographic systems focus mainly on data processing, other on information collecting and visualisation. A spatial group is made by spatial information systems, which allows mapping of data in non-geographic spaces. In this case, the choice of data processing, which enables visualisation, and type of visualisation is important. This group includes for example systems for mapping cyberspace and information spaces.

2.1 Maps

One of the main purposes of our project is to provide a possibility to discover entity distribution, in either geographic space or following some nonspatial properties. This chapter discusses different types of visualisations with respect to mapping techniques into either geographic or logical spaces. Mapping of any phenomenon gives the structural, spatial or logical view of its structure. Then the space of a map can be given by a geographic distribution of entities, but also by logical organisation or relationships among them. Entities' relationships are based on cultural, political, economic or social organisation. Maps can be divided into two main groups according to which measure they use for spatial distribution. The first of them uses the geographic space and the second one is built on some other properties. The way of displaying entities in map is another criteria for map classification.

2.1.1 Geographic maps

Geographic maps display the physical – geographic space. Since we live on the planet Earth, the maps usually portray the earth surface or a part of it. The way they display three-dimensional spherical earth in two-dimensional map is described by projection. There exist many different map projections. Each of them maps the reality into two-dimensional picture in different way.

Map projections always cause some distortion in conformality, distance, direction, scale and area. Figure Fig. 2 shows differences in maps projected by several map projections. Particular methods differ in size of this distortion in each way, i.e. protection of conformality, distances, directions, scale and/or areas.

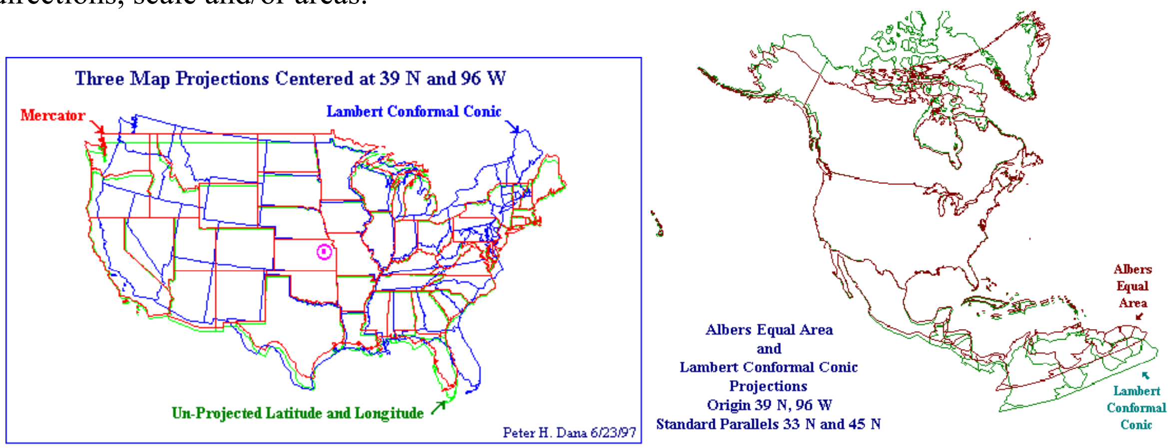

Fig. 2: Distortions caused by different map projections (taken from [2])

Just these determine map projection's positives and negatives and its suitability for certain purpose. Mentioned properties can be explained as follows [2]:

- Conformality - When the scale of a map at any point on the map is the same in any direction and angles are correct, the projection is conformal. Meridians (lines of longitude) and parallels (lines of latitude) intersect at right angles. Shape is preserved locally without twisting or bending on conformal maps. In order to represent angles correctly, relative sizes of areas necessarily will be mis-represented.
- Distance - A map is equidistant when it portrays distances from the centre of the projection to any other place on the map.
- Direction - A map preserves direction when azimuths (angles from a point on a line to another point) are portrayed correctly in all directions.
- Scale - Scale is the relationship between a distance portrayed on a map and the same distance on the Earth.
- Area - When a map portrays areas over the entire map so that all mapped areas have the same proportional relationship to the areas on the Earth that they represent, the map is an equal-area map. It means that any regions compared to each other on the map are the correct sizes relative to each other. In order to represent sizes correctly, the true shapes of the areas necessarily will be distorted.

Map projections can be divided into four classes according to configuration of projection plane and the projected sphere. Each group produces different appearance of the grid on the projected planar surface (see figure Fig. 3). Projections in the first group are called cylindrical, because they project spherical surface onto a cylinder. Conic projections, which project spherical surface onto a cone, form the second group. The third group consists of azimuthal projections projecting surface of sphere onto a plane. There are several projections, which differ in mutual position of the sphere and projection solid, in each group. All other projections such as unprojected ones (e.g. rectangular longitude and latitude grid) falls into the fourth group.

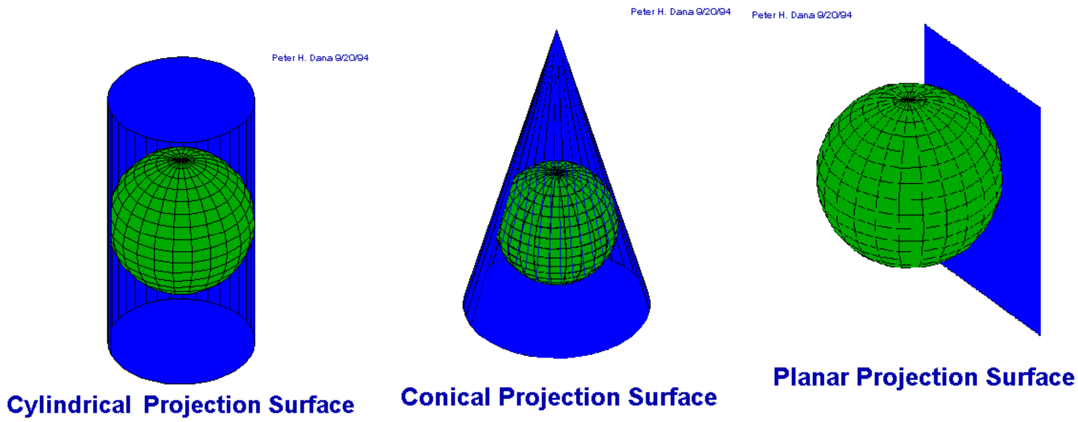

Fig. 3: Different types of projection

Which kind of accuracy we choose to sacrifice depends on the purpose of our map. D.H. Maling (1992) suggests that tropical zones are best mapped using a cylindrical projection. Polar regions are shown best by an azimuthal projection, and countries or areas in the middle latitudes are best mapped using a conical projection. This is because in cylindrical projections distortion increases towards the poles, while in conic projections are true along a chosen line of latitude intermediate between the pole and the equator. Azimuthal projections have a distortion pattern that increases away from the centre point.

Let us discuss some projections with their properties and their suitable usage.

Gall Orthographic (Peters) projection (Fig. 4) is a cylindrical equal-area projection that deemphasizes area exaggerations in high latitudes by shifting the standard parallels to 45 or 47 degrees. It is suitable for thematic maps to display world statistics such as distribution of resources, vegetation, population, or industrial activity.

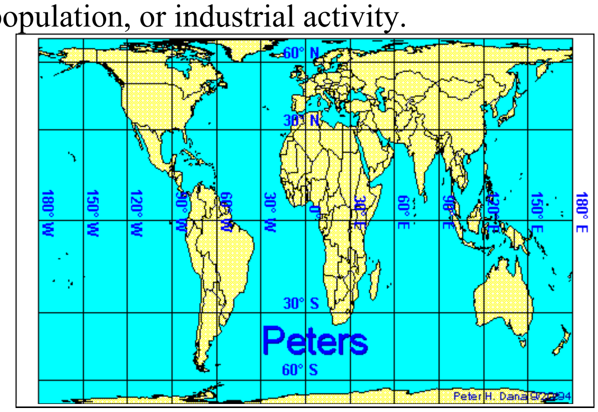

Fig. 4: Gall Orthographic (Peters) projection

Mercator projection (Fig. 5) is a cylindrical conformal projection, which has straight meridians and parallels that intersect at right angles. Meridians are evenly spaced vertical lines, which never converge, the horizontal parallels must be drawn farther and farther apart at higher latitudes to maintain a correct relationship. Scale is true at the equator or at two standard parallels equidistant from the equator. The projection is often used for marine navigation because all straight lines on the map are lines of constant azimuth. On any other projection, a route whose direction does not change would be represented as a curved line. It is not suitable for whole world maps because relative sizes of landmasses are profoundly misrepresented, and also because the poles cannot be shown.

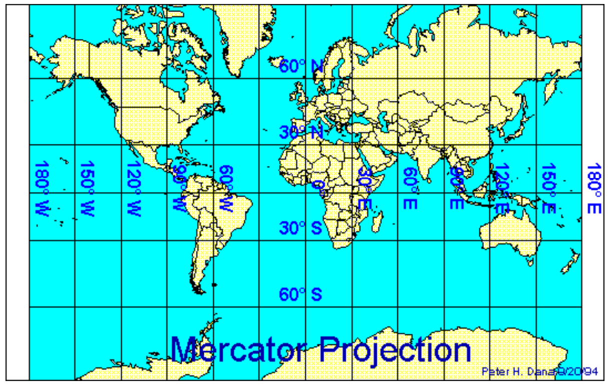

Fig. 5: Mercator projection

Pseudocylindrical projections resemble cylindrical projections, with straight and parallel latitude lines and equally spaced meridians, but the other meridians are curves.

Mollweide projection (Fig. 6) is a pseudocylindrical equal-area projection used for world maps. The central meridian is straight. The 90th meridians are circular arcs. Parallels are straight, but unequally spaced. Scale is true only along the standard parallels of 40:44 N and 40:44 S. It is suitable for thematic maps to display world statistics such as distribution of resources, vegetation, population, or industrial activity.

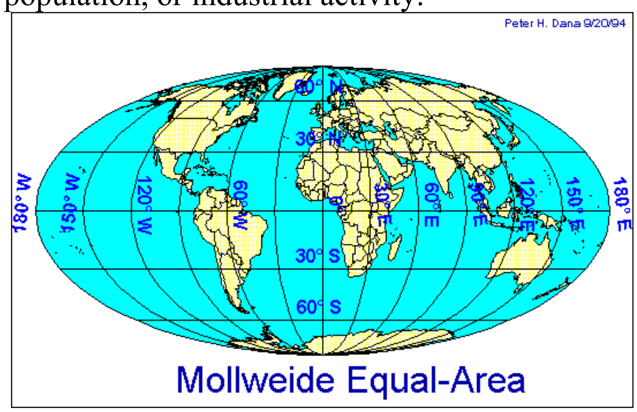

Fig. 6: Mollweide projection

Eckert IV projection (Fig. 7), used for world maps, is a pseudocylindrical and equal-area. The central meridian is straight, the 180th meridians are semi-circles, other meridians are elliptical. Scale is true along the parallel at 40:30 North and South. It is very suitable for thematic maps to display world statistics.

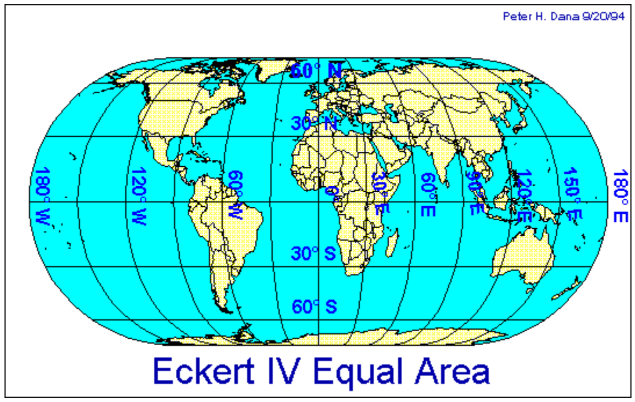

Fig. 7: Eckert IV projection

Sinusoidal projection (Fig. 8) is a pseudocylindrical projection, which has straight parallels at right angles to a central meridian. Other meridians are sinusoidal curves. Scale is true only on the central meridian and the parallels. Often used in countries with a larger north-south than east-west extent. It is suitable for thematic maps to display world statistics.

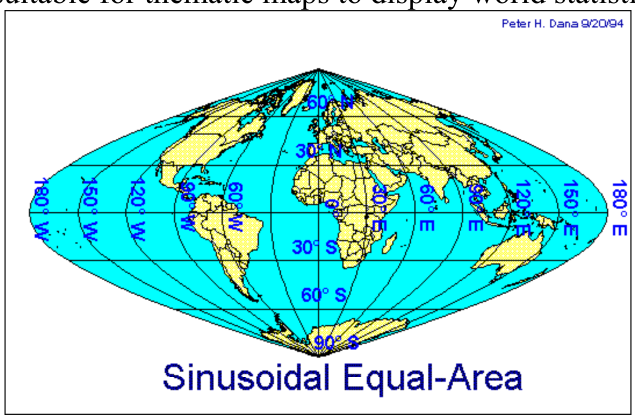

Fig. 8: Sinusoidal projection

Albers equal-area conic projection (Fig. 9) distorts scale and distance except along standard parallels. Areas are proportional and directions are true in limited areas. This projection is best suited for landmasses that extend more in the east-to-west orientation than those lying north to south.

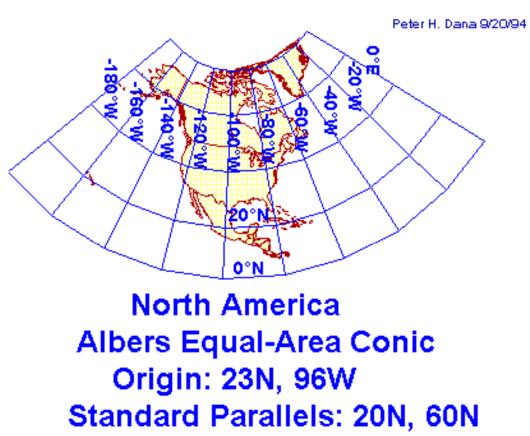

Fig. 9: Albers Equal-Area Conic projection

Azimuthal Equidistant projections (Fig. 10) are neither equal-area nor conformal. They are sometimes used to show air-route distances. Distances measured from the centre are true. Distortion of other properties increases away from the centre point.

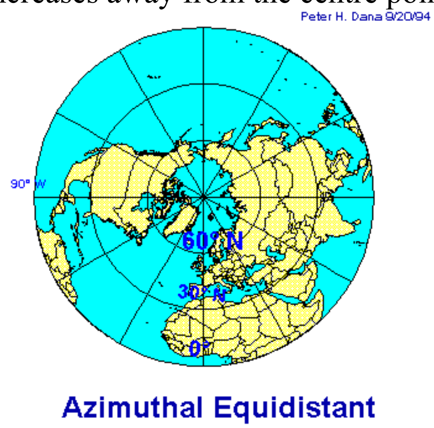

Fig. 10: Azimuthal Equidistant projection

Bonne projection (Fig. 11) is a conic equal-area. All parallels are divided truly and the connecting curves make the meridians. Scale is true along the central meridian and along all parallels. It is used for decorative maps of the northern continents, as Asia, Europe, and North America.

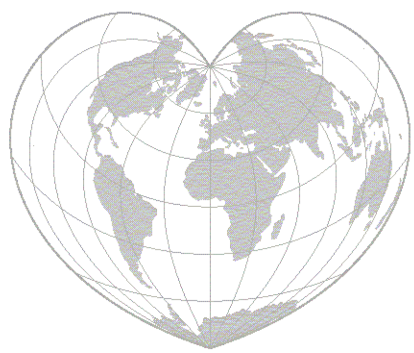

Fig. 11: Bonne projection

Another important information about map is its coverage, which describes the kinds of information included in the map, such as international boundaries, coastlines, roads, airports, or latitude and longitude.

Since our system will visualise mainly geographic distribution of entities, their distances from the closest administrative centre, and so on, the preservation of areas (equal-area) is the main requirement on used map projection. Matching up to other mentioned requirements in settled areas is additional advantage. Next important property of method is the computational complexity of conversion into its coordinates. According to these reasons, Gall (Peters) and Eckert IV projections were chosen.

When mapping only parts of the world, it is useful to follow the assertion mentioned at the beginning of this chapter (Tropical zones are best mapped using a cylindrical projection. Polar regions are shown best by an azimuthal projection, and countries or areas in the middle latitudes are best mapped using a conical projection.) Besides, as the projected area is getting smaller, also the distortion caused by projection is getting smaller. It means, that the

projection with the smallest computational complexity is suitable for displaying of small areas.
Our system should support plug-ins for import of other desired map projections. Thus, user can choose his favourite map projection.

2.1.2 Data representation

Construction of a projection grid was discussed in the previous part. It provides the framework for spatial distribution of entities. Entity visualisation is described in this chapter.
Each map is constructed with an aim to mediate a concrete message, to display a specific distribution and relationships among entities or describe some phenomenon. Thus not all entities or real objects are essential for the message understanding and furthermore they can confuse the message. Similarly, the real shape or appearance of entities is not necessarily important. The question is, which map or visualisation adequately displays key features. When displaying maps with small scales, usually much more entities in bigger detail and probably in real shapes are displayed. On the opposite side small entities, details and unimportant entity features are hidden as the map scale grows. Generally the map generalises and classifies the mapped phenomenon. Appropriate symbols or icons usually represent entities or groups of them in a map with large scale. The colour and size of symbols can represent number and other properties of entities. The selection of appropriate symbols, their sizes, and other visual attributes is generally not easy and is not strictly defined. Thus, only several used ways of displaying entities are mentioned.
Maps can be divided into several groups according to the typical technique they use for visualisation of entities and their properties and relationships. The first such technique accumulates the values for areas and then displays the summary values for example as a colour of region. These maps are called choroplet maps. Figure Fig. 12 shows choroplet map of International connectivity. The map appearance and provided information are highly dependent on selected region size. Thus, they are not always suitable for visualisation especially when the visualised value is not distributed uniformly in whole region.

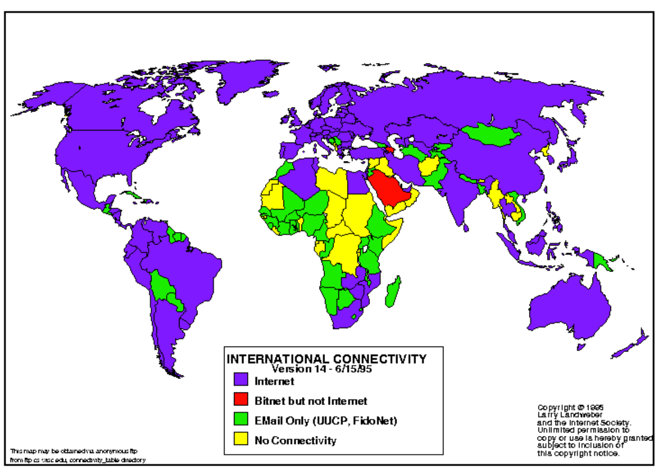

Fig. 12: Choroplet map - International connectivity (source: ftp://ftp.cs.wisc.edu/connectivity_table/)

When mapping infrastructure or network topology (transport, Internet, economic connections etc.), the most important entities, that should be visualised, are nodes and their connections. Such type of entity representation is called arc-node. Since the real shape of arcs is usually not important (or known), arcs are displayed as straight lines. Both node and arc properties are displayed in this type of maps. They are displayed using colour-coding and different arc widths and node sizes. Arc-node map of Israel Academic Network is displayed in Fig. 13 [19].

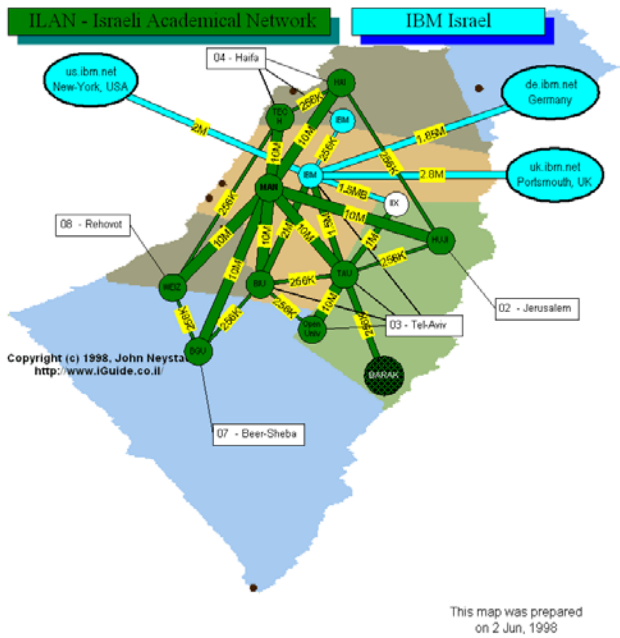

Fig. 13: Arc-node map - Israel Academic Network

Occurrence of entities and their attributes can be displayed in node maps, also called icon maps. They code different attributes of entities by symbol shapes, colours and shapes (also called icons). Figure Fig. 14 shows example of such map produces by Internet Weather Report system [15].

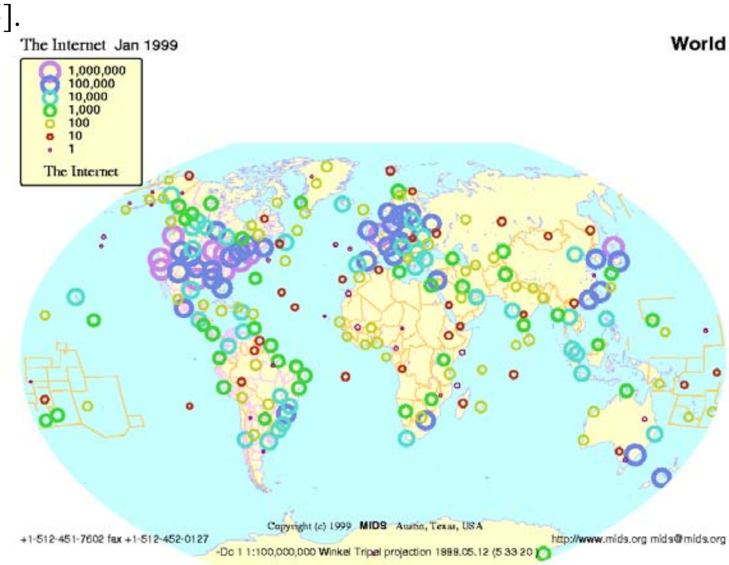

Fig. 14: Dot map – Internet Weather Report - MIDS

Symbol sizes and their big numbers can cause overlapping of symbols and so loss of information. Clustering with appropriately chosen cluster size (see chapter 2.3) can solve this problem as shown in Fig. 15 (numbers of entities are given by bar heights) [24].

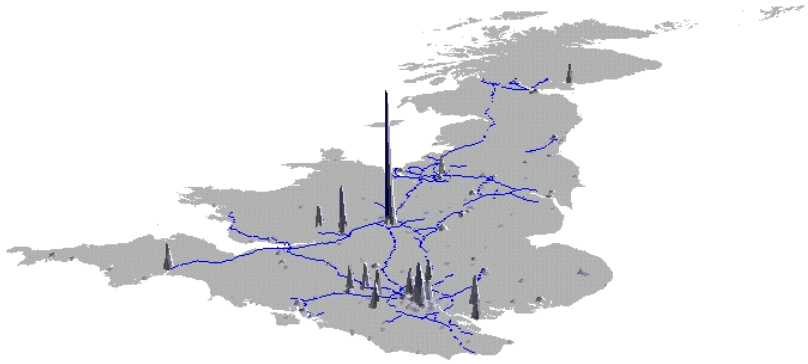

Fig. 15: Dot Map - Internet Address Space

When exploring a map, it is useful to have possibility to get some further information about entities displayed in the map. Reader can want to change the map scale, change type of visualisation or get some specific information about concrete entity. Type of this information can be in general map, other visual, audio or even multimedial and even can be included inside the map and provided on a reader's request. Thus, the map is getting a hypermedia document. For example, UK university sensitive map displayed in Fig. 16 includes hyperlinks to web pages of universities.

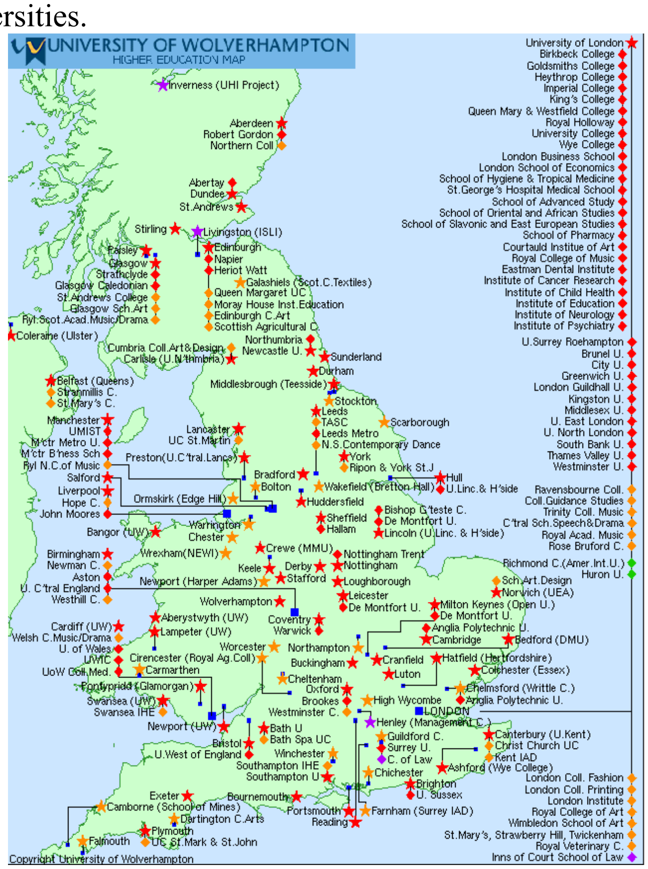

Fig. 16: UK university sensitive map

For purposes of our project, it is suitable to use representation of entity clusters to prevent overlapping of symbols. Symbol size and colour coding encode number of entities and their properties. Choroplet map is suitable only for statistical display numbers of students for each state or smaller administrative area. Arc-node map type is suitable for visualisation of entity (student) relationships (communication). The interactivity of map is important for example for displaying of additional information about shown entities or for zooming into desired location. Similarly, the support for handicapped people should be implemented as well (for example text zooming or reading for visually handicapped or blind people).

2.1.3 Logical maps

Our system should provide not only geographic visualisation of students, but also their distribution into courses given by their study programs, time trends of their studies and other visualisations in spaces given by student properties.

In contrast to the mapping of entities geographic distribution, in this part, we discuss attempts to create other types of spatialisation. Data with no inherent spatial properties is mapped onto a defined spatial framework to be better understood. The selection of map contains and its form is based on the map purpose similarly to geographic maps. The non-geographic space can be based several data properties. In the first case, the logical infrastructure (computer network structure, structure of employees) of data is used to construct topological maps. Information spaces are constructed according to topic of information in data. Finally, other properties of data, such as time or age, can be used as the base of map space.

A typical example of topological map is a map of computer network based on IP addresses (shown in Fig. 17 [27]) or the hierarchical structure of the network. Then main nodes representing servers are distributed in space and their clients surround them.

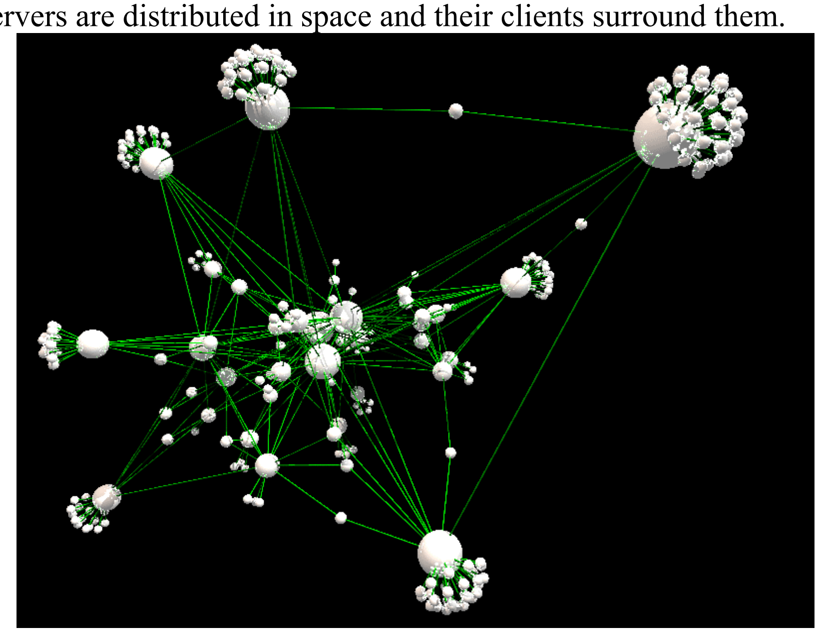

Fig. 17: Three dimensional Graph Visualisation HyperSpace by Andrew Wood, Nick Drew, Russel Beale and Bob Hendley

In contrast to infrastructure topology spatialisation, the creation and mapping of information spaces involves the application of spatial metaphors. Most of spatialisations are designed to improve navigation through an information space and enable people to find the data they are searching for more easily. Such spatialisatins specialise on a particular type of users and type of users' thinking. Figure Fig. 18 shows a visualisation of interest of members of an electronic community made by Visual Who [6].

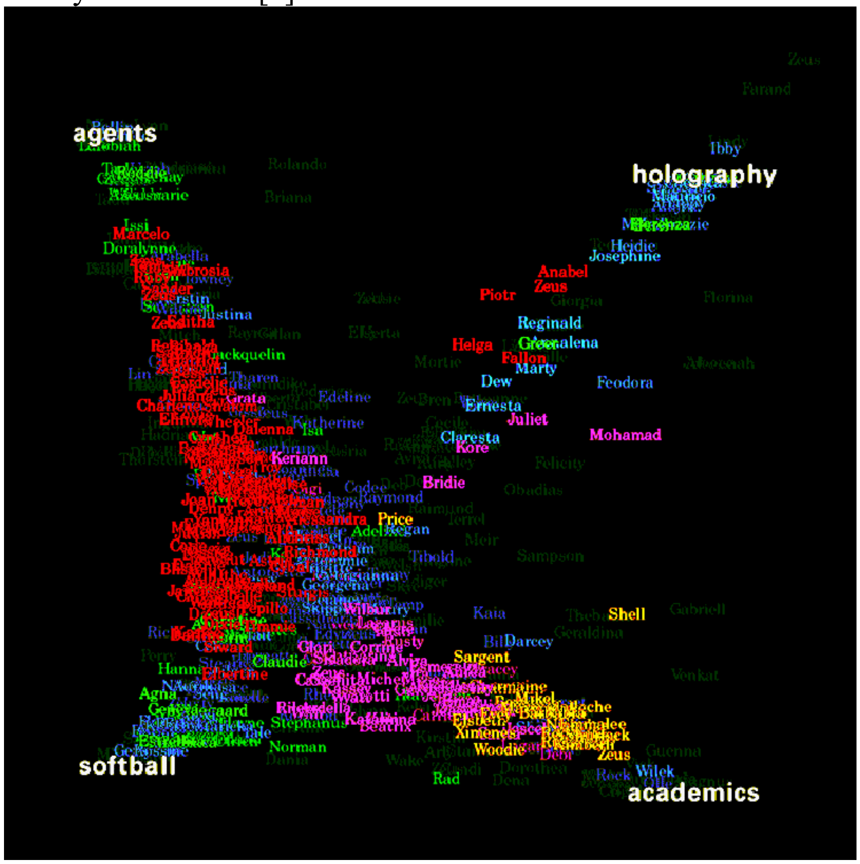

Fig. 18: Visual Who by Judith Donath

The last group of spatialisations is based on some other measure. This measure can be time, age, time of studies, membership in some groups, contacts with some people, borrowing some books and so on. This information can be important for orientation in a large volume of data. Therefore, it is useful to use them for navigation in data. Usage of time is a special case, because either spatialisation can be based on time only or time can be only another dimension usually used for animation.

Just visualisation of student distribution into courses or study programs is one from displays provided by our system. Also representation of time trends of student behaviour is important and will be supported by the system.

2.2 User interface

Interface is understood as the way in which program communicates with its user and vice versa. It is very important part of each program, which can make the work with program simple and pleasant or in the opposite side very complicated and discouraging. Some used interface features, especially those important for interactive, work are mentioned in this chapter.

The aim of each interface is to provide data in the most understandable way and allow the simplest and fastest user input. The previous chapter mentioned ways of data visualisation, now let us look at integration of visualised data into program environment and program control.

Palantir is a system, which collects statistics about web site accesses. Since similar analysis is part of our system, Palantir can be taken as a appropriate example of user interface (see Fig. 19).

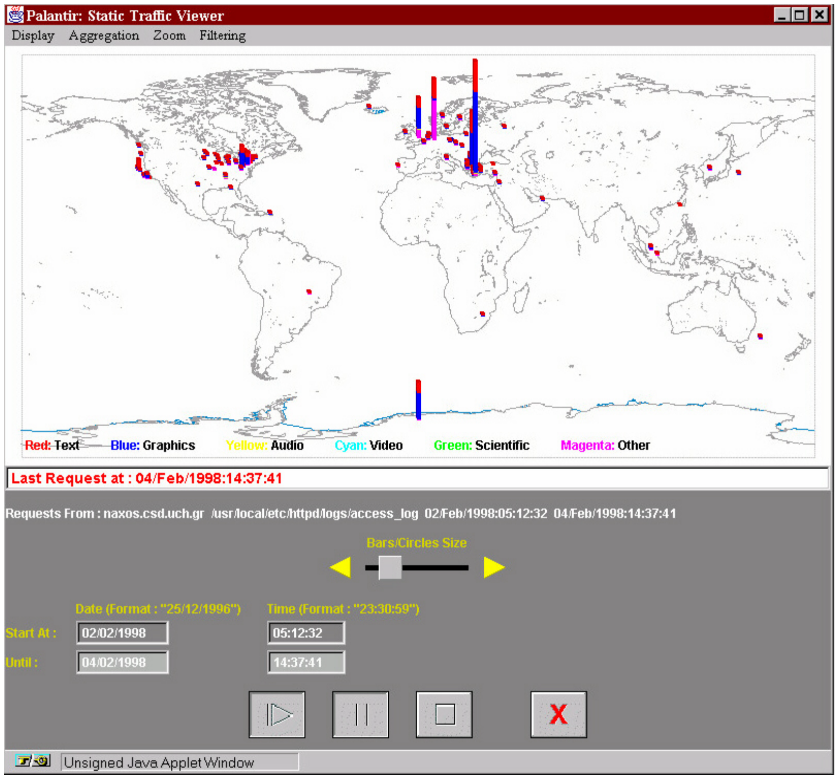

Fig. 19: Palantir 's Static Traffic Viewer Window

The Palantir WWW traffic visualisation application's view [21] can be taken as an example of a simple geographic visualisation tool. The viewer window consists of three parts. The first part displays a geographical map, which is initially set to the whole world, with colour bars or circles. The bars or circles are split and coloured according to numbers of entities in each group belonging into each particular location. There is a key explaining colour coding at the bottom of this part. The viewer menu is situated at the most top of the window. A menu enables to enter more detail commands. The last part, the bottom one, contains information about currently shown map and some basic controls. Sliders, buttons and fill-in boxes are used as the program controls.

The traditional approach to querying, as can be seen in the previous example, is to use a fill-in interface. However, the entered query can return zero hits or a very large number of hits. Such situations can lead to user frustration. It is difficult for the user to estimate how much data is available on a given topic and how to increase or reduce result set sizes. Thus, the presentation of overview and previews of abstracted metadata [22] can solve this situation. It allows users interactive control of visual query parameters that generate rapid, animated and visual displays of database search results and so rapid and dynamic elimination of undesired data. As users adjust sliders or buttons, results are updated rapidly (within 100 ms). The reduced volume of the abstracted metadata allows queries to be previewed and refined locally by the user before they are submitted over the network. This approach can be shown at the Restaurant Finder application (shown by Fig. 20).

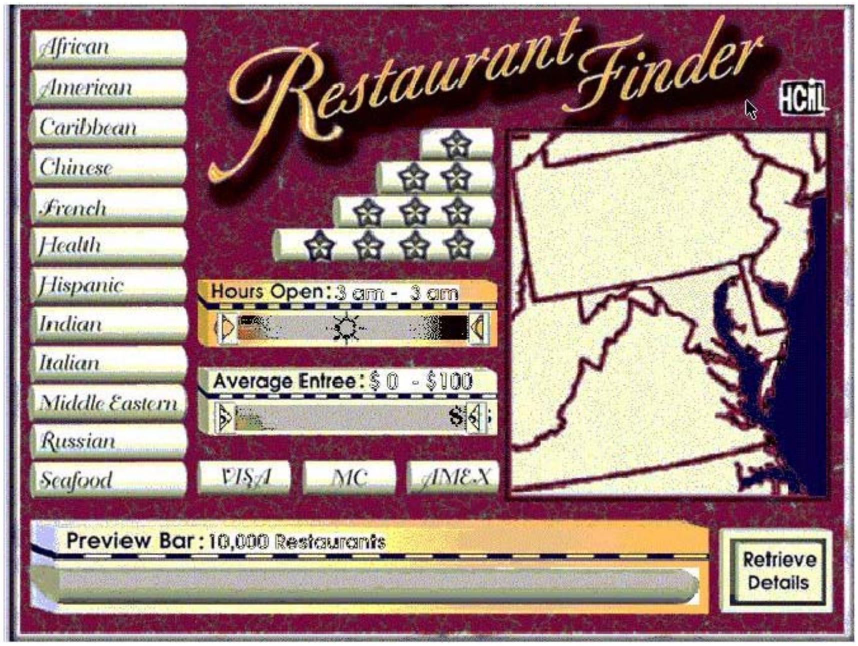

Fig. 20: Restaurant Finder

The most important part of query preview interface is a single-screen overview of all datasets. The selection of desired dataset is done using controls of several high-level attributes (for example location, years of coverage, and general topic area). These high-level attributes have to be chosen by staff and should be the most salient ones for majority of users. As a next example the Dynamic Queries for EOSDIS (see Fig. 21) can be taken. In this case, the granularity of the attributes is deliberately kept rather crude (or large), in order to be able to represent the data in a single overview screen.

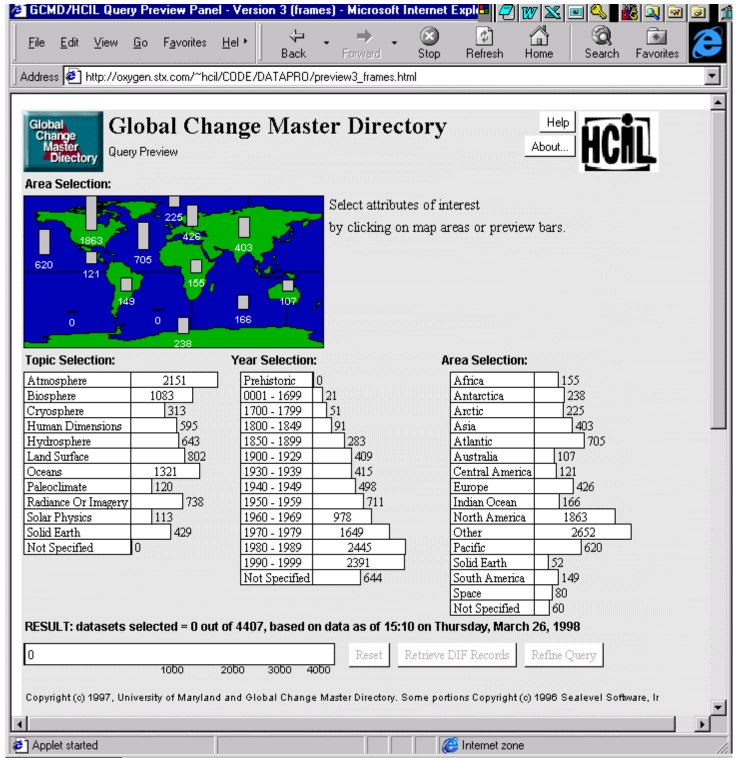

Fig. 21 : Dynamic Queries and Query Previews for networked information systems: the case of NASA EOSDIS

Let us summarise the important features of dynamic queries. They are made in two steps, at first the desired criteria is specified in query preview and after that the request is submitted to the network and the data is refined. Users can then continue to refine their queries with additional, more specific, criteria. The immediate and continuous feedback is important for comfortable query selection by pointing, not typing (sliders and check boxes are used). The query is controlled incrementally and visually represented during the process. The query construction is reversible.

As mentioned above, it is quite congruous to enable interaction directly through the displayed map, to make it clickable. There is a lot of different action, which can be called after the click. The map can be planned or zoomed. Some detail information about chosen region can be provided (textual, audio, video or multimedia). It can be opened either in a floating window or in a window dedicated to these purposes.

Another aspect of user interface is the type of communication with computer. Since now, only the keyboard or mouse control was assumed. For example, voice control can bring new possibilities. Simple commands can be given by voice and complemented by pointing in a map. Thus voice commands with location descriptions through pen-based pointing or marking can be the most effective way to elicit brief input during map-based tasks. This type of interface is called a multimodal interface design [20].

The query preview interface is suitable for our application, because it allows fast reduction of huge amount of entities. Direct interaction through a displayed map will be used for entering of data immediately related to the map (coordinates, positioning, etc.). The support for handicapped people is needed due to the style of OU education, which is suitable also for these people.

2.3 Clustering

Huge number of entities has to be displayed in our system. When displaying them, symbols representing entities overlap especially for very large scales. Clustering the entities can solve this problem. It reduces the number of entities and represents them by clusters. The bases for clustering are logical or spatial distances or both (for example close cities in the same country). The distance has to be defined for any two entities and describes the relationship between them.

When using hierarchical clustering, a hierarchical structure completely describing relationships between entities can be constructed by a technique called hierarchical clustering. Then any suitable distance threshold can be chosen for specific situation. The hierarchical clustering groups entities contiguous in some defined way, thus it can be used for statistical analysis as well.

The basic algorithm of hierarchical clustering can be described as follows [28].

INPUT: m objects with $\frac{m(m-1)}{2}$ pairs of distances, $\{d_{i,j} \mid 1 \le i \le j \le m\}$. It is assumed that self-distances are all 0 and $d_{i,j} = d_{j,i}$.

STEP 1: Find the smallest distance $d_{i,j}$, $i < j$.

STEP 2: Agglomerate objects i and j. That is, i and j are replaced by a new object, denoted by $i \cup j$. The distance of this new object from the remaining m-2 objects is defined to be $d_{i \cup j,k} = f(d_{i,j}, d_{i,k}, d_{j,k})$, where f is a function described bellow. Then distances $d_{i,k}$ and $d_{j,k}$ for all possible k are deleted.

STEP 3: The new object $i \cup j$ is renamed to i, and j disappears. The use of the smaller index is arbitrary. The new object, i, is called a cluster, and we denote by $|i|$ the number of primary objects in this cluster. We also allow single objects to be called clusters (of size 1).

STEP 4: If 2 or more clusters remain, repeat steps 1-3.

Steps 1 through 3 are applied m-1 times before the algorithm terminates.

The function f can be defined one of these ways:

- Single link – nearest neighbour: $d_{i \cup j,k} = \min\{d_{i,k}, d_{j,k}\}$
- Complete link – diameter: $d_{i \cup j,k} = \max\{d_{i,k}, d_{j,k}\}$
- Average link – nearest neighbour: $d_{i \cup j,k} = \frac{|i| d_{i,k} + |j| d_{j,k}}{|i| + |j|}$
- McQuitty's method: $d_{i \cup j,k} = \frac{d_{i,k} + d_{j,k}}{2}$

The methods of distance calculation differ in their meaning. Single link defines distance from cluster as distance from its closest entity. In contrast, Complete link counts it as distance from the furthest entity of cluster. The distance of average link method is based on centre of gravity of the created cluster. Finally, the McQuitty's method just averages distances from both parts of the new cluster.

Figure Fig. 22 shows an example of cluster information display in a map (an ArcMap view [7]). Since the information is accumulated for each group of entities, it is easier to see the attribute distribution.

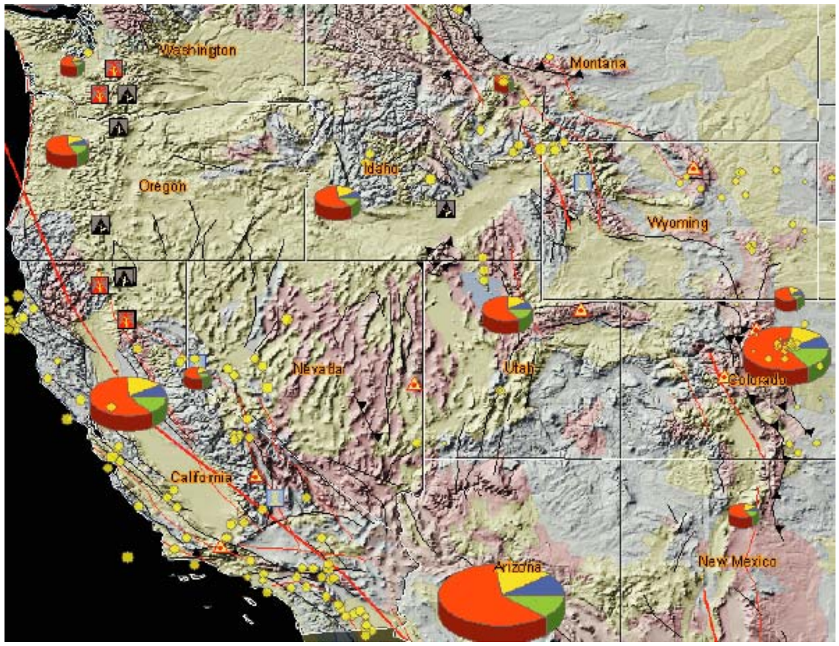

Fig. 22: Dislpay of cluster information in ArcMap

The Average link method is most suitable for our purposes, because it uses the centre of gravity of cluster, into that the created cluster is displayed. A new algorithm with improved asymptotical time complexity was designed for our system (see [12]).

2.4 Data filtering and analysis

A subsystem providing data filtering and analysis is an important part of each geographic or spatial information system, including ours. This chapter describes the way of entering filter rules and filtering. Then data analysis and decision making support are mentioned.

Filtering rules consist of requirements on parts of entities. Entity properties are represented by items of entity record. Then, the requirement can be for example equality to a value, or occurrence in defined interval. Filtering rules are inferred from user requirements or according to intern system rules (for example when computing a more complex measure). Typical user interface items were already described in chapter 2.2. Text fields, sliders, check boxes, radio buttons, list boxes etc. are those used for filters entering. Previews (discussed in chapter 2.5) provide better user control of filter entering and its impact on resulting entities.

Filtering process depends on used database (discussed in chapter 2.7). Relational databases are most frequently used. Filtering is done by a query into the database using given filter.

Filtering is not the only one way of data processing made in information systems. They can implement more complex operations with data such as statistical measures calculation, search for dependencies, and so on. The spatial nature of data is important in geographic and spatial information systems. Set of implemented operations depends on application field of concrete system.

Next possible group of analysis is provided by expert systems or system for support of decision making. They deduce their inferences on bases of derivation rules and they are able to explain their results. At the same time, they allow usage of fuzzy values, it means that they are able to work with ambiguous data and give weights to their results according to their trustworthiness.

All mentioned techniques except expert system will be supported also in our system.

2.5 Scalability

When working with huge volume of data, problems occur with speed of processing and volume of data, which can be handled by system without increasing response time above some acceptable threshold. The problem has to be faced also when designing our system, because the Open University has millions of students. Methods, which allow reaching scalability, are described in this chapter.

Scalability is understood as capability to handle large amount of data. It can be also defined as the property of algorithm to scale up to large sizes of processed data with linear grow of required resources.

In geographic information systems, there are several fields, which are time consuming and therefore there is a need to achieve scalability of used techniques. The first problem is query processing on a database server. The second one is response time on client side, which includes transfer of data from server to users and techniques used for better query formulation.

The database server of a GIS in most cases uses a commercial database engine usually based on a relational database using SQL. Such engines use different techniques to speed up their responses on queries. The techniques are based on a database preprocessing. Such preprocessing creates indeces or clusters used then for faster data access.

The GIS response on client request depends on amount of data, which has to be transmitted from server to client. Accurate query formulation can reduce undesired data receiving and so reduce the response time and need to repeat queries. As discussed in chapter 2.2, the system of previews can be used for query formulation and number of obtained entities control [4]. Let us discuss the computation complexity in more detail.

When using the system of previews, the preview table has to be sent to client at first. The size of the preview table and computation time grows exponentially with the number of attributes and attribute values. Several situations can occur. When the number of records is small, a hybrid solution can be used. It lists IDs of all individuals. Otherwise, the size of preview query table has to be reduced. One of the ways is reducing of number of attributes or their values or selectable values. The second possibility uses only range queries. Then only several arrays of data are transmitted and specific algorithms allow determining numbers of entities in selections (for further information see [22]). The third way is to use only binary preview, which represent only existence or absence of entities, not number of them. In such a case, a compressed binary array can be used for preview data transmission. This technique allows any combination of selected attributes.

Figure Fig. 23 shows query formulation using dynamic queries and query previews (the user interface is discussed in more detail in chapter 2.2).

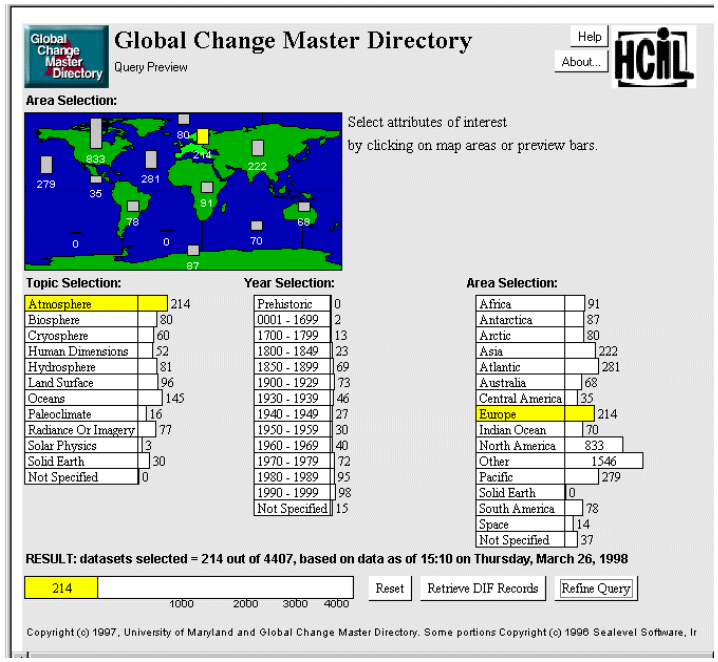

Fig. 23: Dynamic Queries and Query Previews for networked information systems: the case of NASA EOSDIS

System of previews is suitable also for usage in our system, similarly as the division of database into clusters.

2.6 Network service

The architecture of our system is planned to be client-server. It means that a server will be a computer administrated by Open University and containing student database and other necessary information (maps, etc.). Clients will be computers of students or OU staff. Service will be done over a network (Internet). Approaches to network services are discussed in this chapter.

A web browser can be a simple example of client application. The application displays documents downloaded from different servers. Since the document can contain other embedded documents (typically images), it is necessary to download all files needed for given document display. When visiting a particular server often, it is unnecessary to download all the documents or better their parts (especially images) every time, because they usually stay unchanged. Therefore, local copies of these files (html documents, images, etc.) are stored. When requiring display of a file, at first the date of local copy is compared with date of file on server and only when they differ, the file is downloaded. Otherwise, the local copy is used and the downloading is not necessary.

Traffic visualisation Tool for World Wide Web called Palantir [21] was already mentioned in chapter 2.2. This tool supports geographic visualisation of web servers traffic. Log files from web servers and maps stored on own server are used by the system as source of data. The system architecture is shown in figure Fig. 24.

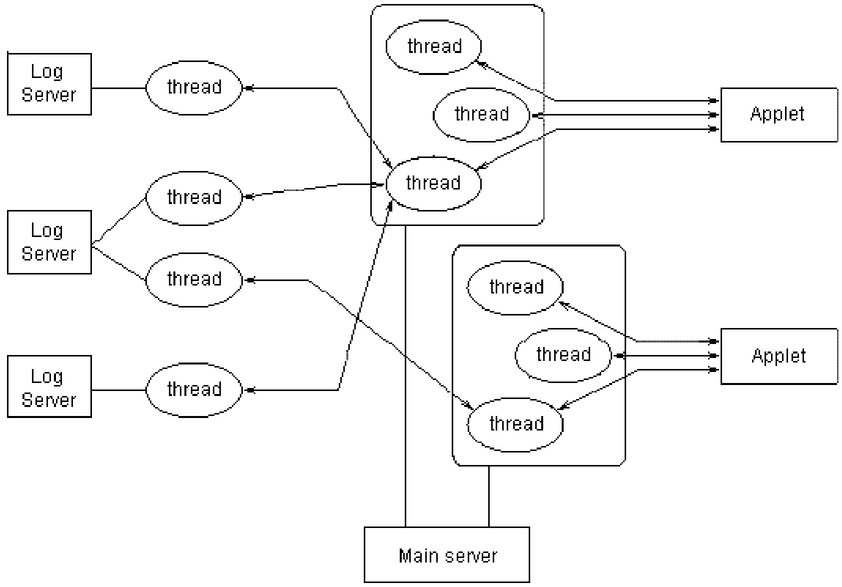

Fig. 24: Architecture of Palanir visualisation tool

The system consists of three main components. The first one consists of log servers. A log server is application, which reads log file and send its contains to main server, whenever it requests it. The second component is made by applets. The applets are client applications with user interface. Main server is the third and most important component. It manages map database and serves applet requests. The main server consists of three threads. The first of them communicates with log servers. The second thread communicates with applets and sends them resulting geographical maps on their request. The third thread provides IP address conversion to geographic position to enable its display in a map.

As already mentioned in chapter 2.5, system of previews brings improvement of user control of query definition and decrease of necessary network communication. Thus, it is reasonable to mention it also concerning network services. Figure Fig. 25 illustrates service in query preview in networked information system.

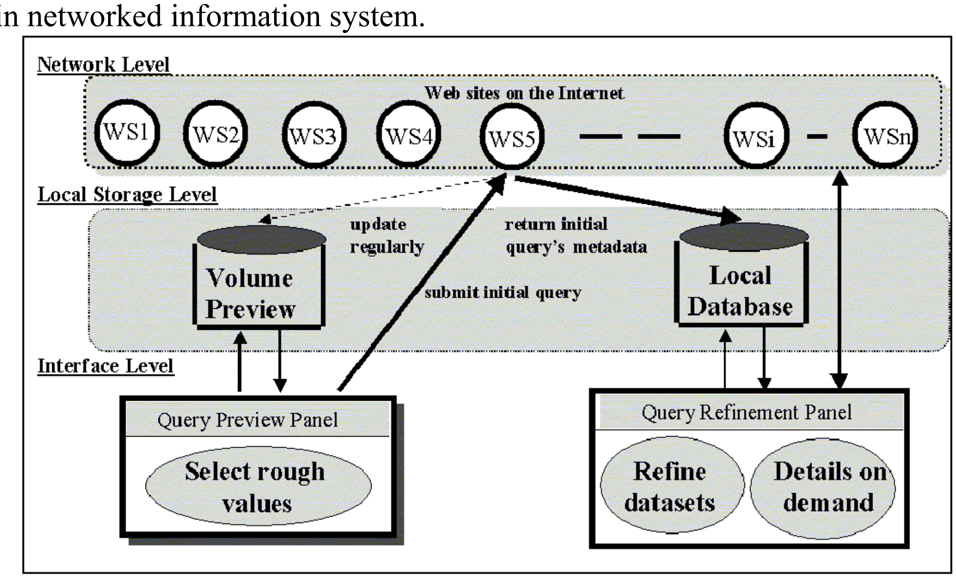

Fig. 25: Architecture of Query Preview in Networked Information Systems

In the first phase, the coarse selection is done by small number of attributes with help of preview information. The resolution of attributes is small, but the system is able to provide information about numbers of corresponding entities. When the user is satisfied with the query formulation, a request is send to database over the network. The database then provides more detail information. Refine datasets allow further specification of requirements in reduced scope but with finer resolution and all attributes. Complete information can be obtained in any phase.
Transmission bandwidth, client display resources and requirements as well as processing power differs from one client to another. Thus, the server response has to be suited to client and network capabilities and user requirements. All data, which cannot be displayed or processed by client, are removed by server before transmission. The server side reduction brings saves of transmission bandwidth and client processing power. Required transmission bandwidth can be further reduces by a compression with optional and adjustable data loss [23].
Documents in our system will be updated once a long time (after enrolment of new student, etc.), therefore it is suitable to use a cache system similar to the one used by web browsers. Similarly to Palantir, our server will read logs of client communication among clients and investigate communication patterns included in their contents. Separation of client request execution into several separated parts is suitable for our system too. Since database will contain huge amount of data, usage of system of previews is suitable. Due to big diversity of client capability including computing power, screen resolution and connection speed, it is suitable to adapt answers according to user requirements and capabilities.

2.7 Data structures

When handling geographical data, several data structures are used. They can be divided into several groups according to their purpose. The first group includes structures used for storing of spatial data, attribute data and maps. The second group consists of data structures, which help for faster query processing (for example indices for faster access into database). Structures used by communication protocol between server and client belong into the third group. They are capable to carry queries and responses.
Let us discuss the data storage in a usual GIS. A GIS stores information about the world using thematic organisation and thus categorise data into vertical thematic layers. The definition of layers is fully dependent on database purpose and users requirements. This concept is simple but extremely powerful and versatile and allows solving many real-world problems. An example of layer organisation is shown in Fig. 26.

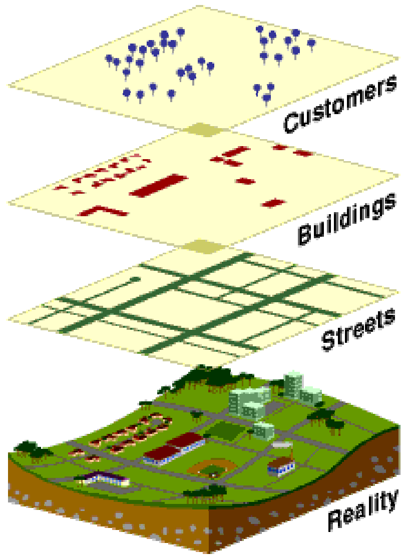

Fig. 26 : Layer organisation

Similarly to data in traditional map, there are two basic data types utilised by GIS. The spatial data describes either relative or absolute location of geographic features. In contrast, attribute data also often called tabular data describes characteristics of the spatial features. The nature of these characteristics can be quantitative and/or qualitative.

A GIS stores digital spatial data in two basic ways. The figure Fig. 27 illustrates difference between raster and vector form.

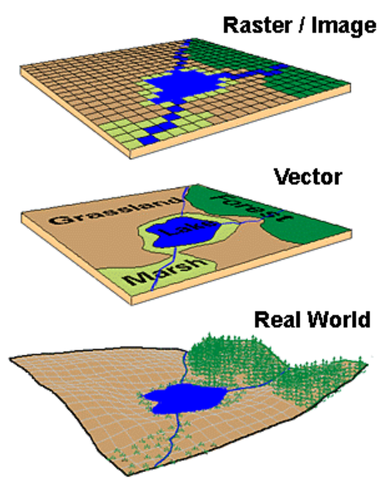

Fig. 27: Types of spatial data storing

The raster form is based on grid (cell or tessellation) data structure. Thus, the geographical data is divided either into regular cells, where the position is identified by row and column, or into an irregular tessellation. The size of cell in a tessellated data depends on accuracy required by user. Since the geographic coordinates of each cell are given by its position in tessellation grid, there is no need to store them explicitly.

The vector form uses vectors for storing spatial features. Vector data is defined by sequence of vertices consisting of X and Y coordinates. The vertex coodrinates determine its spatial position. Vector lines, referred to as arcs, are represented by a string of vertices terminated by nodes. Polygonal features consist of set of closed coordinate pairs. Since the storage of vertex

coordinates for each feature and feature connectivity is crucial for vector representation, the topological data structure is often used. It allows easy derivation of spatial relationships between features.
Attribute data, in a GIS, is stored different way. Databases used for its storage use variety of different data models. These models are tabular, hierarchical, network, relational, and object oriented. The last three are the most commonly implemented. GIS data can also be reflected in an external commercial database.
The organisation of data in horizontal layers is called spatial indexing. The spatial data partitioning is one from a variety of different techniques, which allow speed up the handling of spatial features. It creates more manageable subsets or tiles, which are then indexed using a mathematical function, for example quad-trees or R (rectangular) trees. Thus, the quick search and retrieval of data is possible.
Basic data structures described in this chapter will be used in our system. Besides, other structures will be added to enable storing of system specific data and especially to speed up service (see chapter 3.7). A quadtree will be used for map indexing and a hierarchical clustering tree for student indexing.

2.8 Miscellaneous other related work

2.8.1 General features

Let us start with short description of general capabilities of geographic and spatial information systems. As mentioned already at the beginning of chapter 2, spatial and geographic information systems are used especially for mapping and analysis of phenomenon and events. This reason leads such systems to support following features: data capture, data management, data manipulation and analysis, and the presentation of results in both graphic and report form. A spatial or geographic system can be divided into four main functional subsystems according to just mentioned functions. The first of them is the data input subsystem, which allows user to capture, collect, and transform spatial and thematic data into digital form. Hard copy maps, aerial photographs, remotely sensed images, reports, survey documents usually serve as the data inputs. The second subsystem is the data storage and retrieval subsystem. It manages both spatial and attribute data, in a form which permits it to be quickly retrieved by the user for analysis, and permits rapid and accurate updates to be made to the database. The maintaining of attribute data is usually done by a database management system (DBMS). On the opposite side, spatial data is usually encoded and maintained in a proprietary file format. The third subsystem provides data manipulation and analysis. It is commonly thought of as the most important part of each GIS. It allows the user to define and execute spatial and attribute procedures to generate derived information. Finally, the data output subsystem allows the user to generate graphic displays, normally maps, and tabular reports representing derived information products.
Some systems add the ability of network service of client requests from more computers. Such systems face problems with service of huge amount of clients, speed and reliability of network, and so on.
Each particular system is focused on certain branch or concrete problem. Thus, systems differ in support of particular operations. While some of them focus on data collecting and displaying, other support huge data analysis.

2.8.2 Concrete systems

Several concrete systems are mentioned and their function and features described in this chapter.

ArcGIS by ESRI [7]

The ArcGIS is an integrated geographic information system consisting of three main parts. They are ArcGIS Desktop software (integrated GIS applications), ArcSDE gateway (an interface for managing geodatabases in a database management system (DBMS)), and ArcIMS software (Internet-based GIS for distributing data and services). Thus, the ArcGIS modular architecture provides a framework for GIS solutions. It is scaleable from a single desktop to a distributed heterogeneous computer network of workstations and servers (the services for multiuser editing and Internet deployment can be added - see Fig. 28).

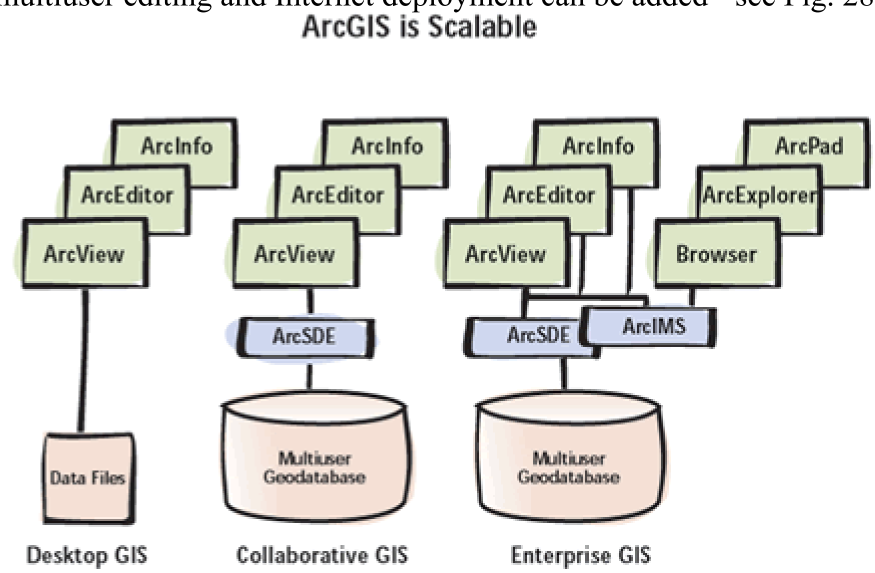

Fig. 28: Modular architecture of ArcGIS

The system provides tools for editing and data automation, mapping and map based tasks, data management, geographic analysis, and deploying data and applications on the Internet.

It supports several geographic data models (file based and geodatabase model using relational databse).

ArcGIS Desktop includes integrated applications ArcMap, ArcCatalog, and Arc Toolbox and it allows to use spatial data and resources through ArcIMS services on the Internet. *ArcMap* (see Fig. 29) provides cartography, map analysis, and editing. It offers different ways to view a map or layout view and perform a broad range of GIS tasks.

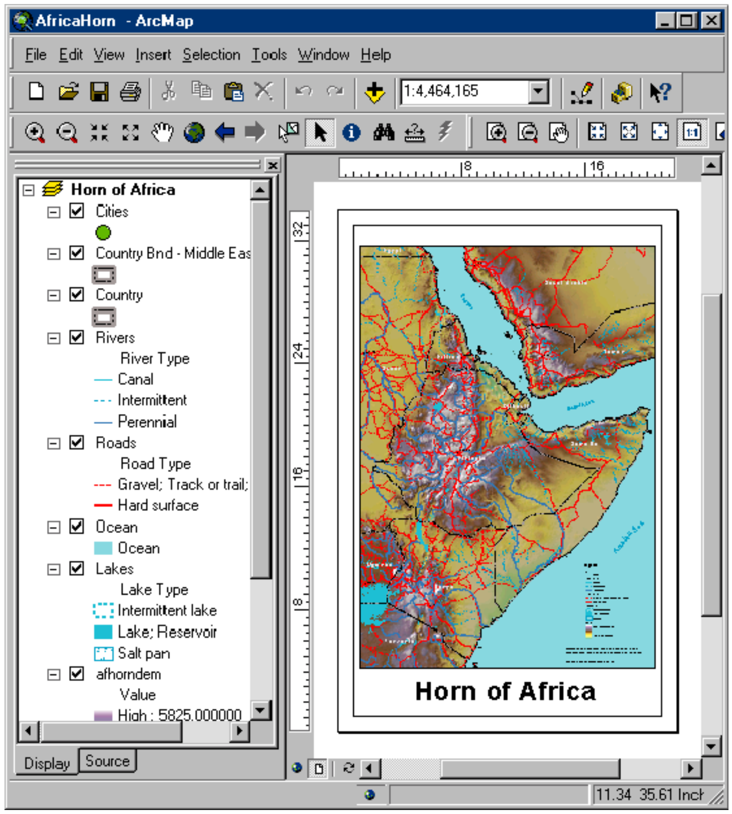

Fig. 29: ArcMap

ArcCatalog (see Fig. 30) performs managing of all GIS data, it means browsing and finding, recording and viewing, and defining schema structure for data layers.

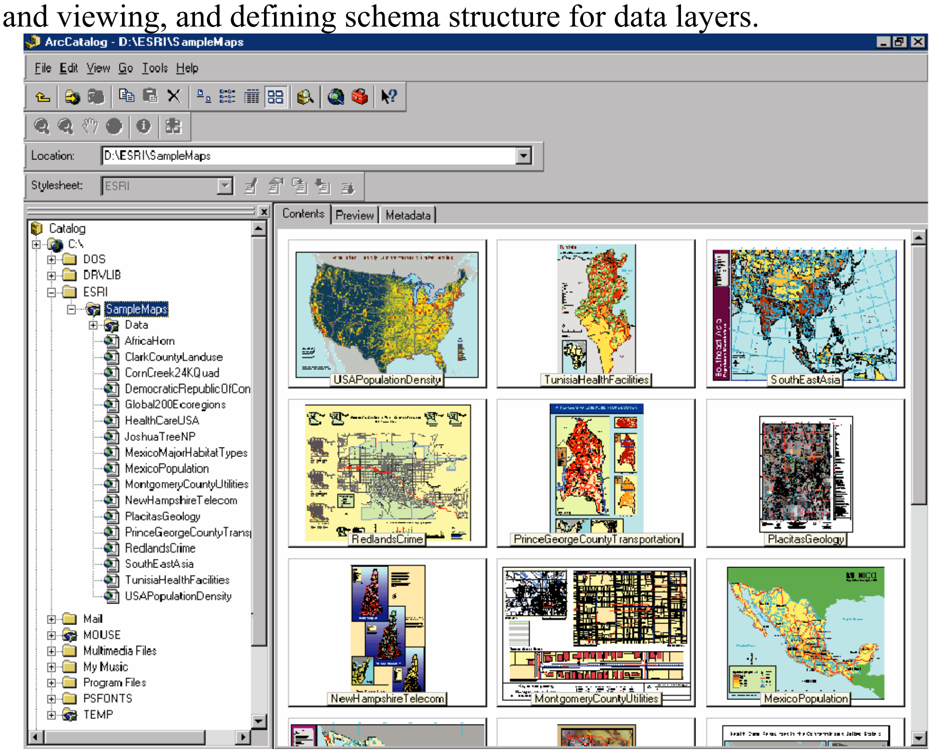

Fig. 30: ArcCatalog view

ArcToolbox can be used for data conversions and geoprocessing.

The most important features of ArcGIS Desktop software are: advanced editing tools, high-quality cartography, Internet-enabled, on-the-fly projection, geocoding, wizard-driven tools, support for metadata standards using XML, COM-based customization, extensible architecture, direct read of more than 40 data formats.

MS MapPoint [16]

MapPoint is a desktop mapping software. It does not provide any spatial analysis or query functionality and it is closed system without possibility of adding information into maps. Thus, it is not a traditional GIS. Its main goal is to visualise data. It focuses on three fields: locate addresses and display them in a map, help identify business trends and integrate visualised maps into other MS Office products (for example MS PowerPoint) as well as export them into HTML. Business trends identification (see Fig. 31) is based on data imported from another application (for example MS Excel or MS Access) and then matched using street address, ZIP code, state and country name or latitude/longitude.

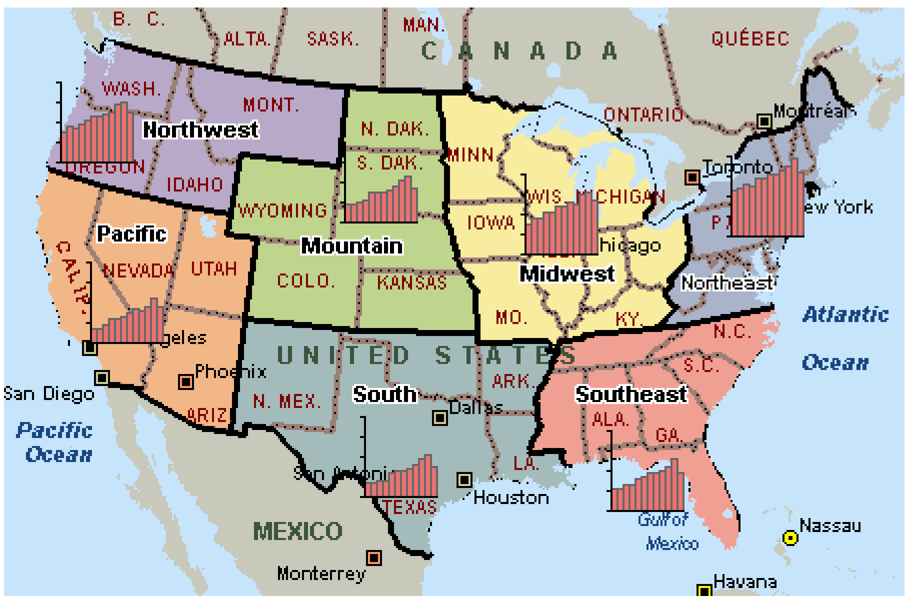

Fig. 31: MapPoint – business trend identification

MapPoint can't perform routing between locations, but it contains tool for manual marking and subsequent measuring of distances. It also provides drawing of areas with certain distance from a given place (see Fig. 32).

Fig. 32: MapPoint – route planning

MapPoint uses high quality geographic (detailed stree-level information) and demographic (for example incomes, age) datasets. The user interface consists of three parts: main map, overview map, and legend.

The Internet Weather Report [15] scans and presents conditions inside the Internet. It measures round trip time (latency) from offices in Austin, Texas to thousands of Internet domains world-wide, currently every four hours, six times a day, seven days a week, using ICMP ECHO (ping). This time is then visualised in geographic maps. The figure Fig. 33 shows its result for Europe.

Fig. 33: Internet Weather Report

The visualisation is done by node map. When visualising, it uses two attributes of entities, explained in the legend in the upper left of each map. The circle diameter indicates time delay. Short times are displayed by small circles and long times by very large circles. The second attribute is number of sites represented by the circle. It is encoded by the colour of circle.

It adds time as a next dimension and uses animated GIF for storing the animations.

System used for **Visualizing the Global Topology of the MBone** [18] is a next example. The MBone is the Internet's multicast backbone. Its topology is not optimal due to its exponential grow without any central authority. This is the reason for its topology visualisation (Fig. 34) and analysis.

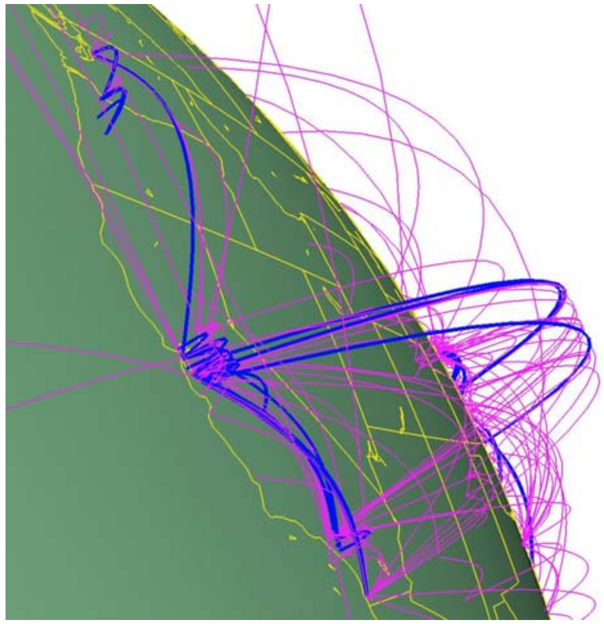

Fig. 34: Global topology of Internet MBone

The system represents tunnel structure using an arc-node map. The map is made as a three-dimensional view on a globe. Nodes are given by longitude and latitude of Mbone routers. The system resolves geographic coordinates of nodes from their IP addresses. Data visualisation techniques such as grouping and thresholding allow further analysis of specific aspects of the MBone topology. The 3D maps are distributed through the World-Wide Web using VRML file format.

Real-Time Geographic Visualization of World Wide Web Traffic [13] displays World Wide Web traffic in three different ways. They are time tunnels, scattercubes, and geographic displays. It uses global perspective for a summary view and a simple flat projection for local views. Virtual reality Avatar is used as the user interface (see Fig. 35). This work is rooted in information visualisation and statistical graphics with emphasis on interactive exploration.

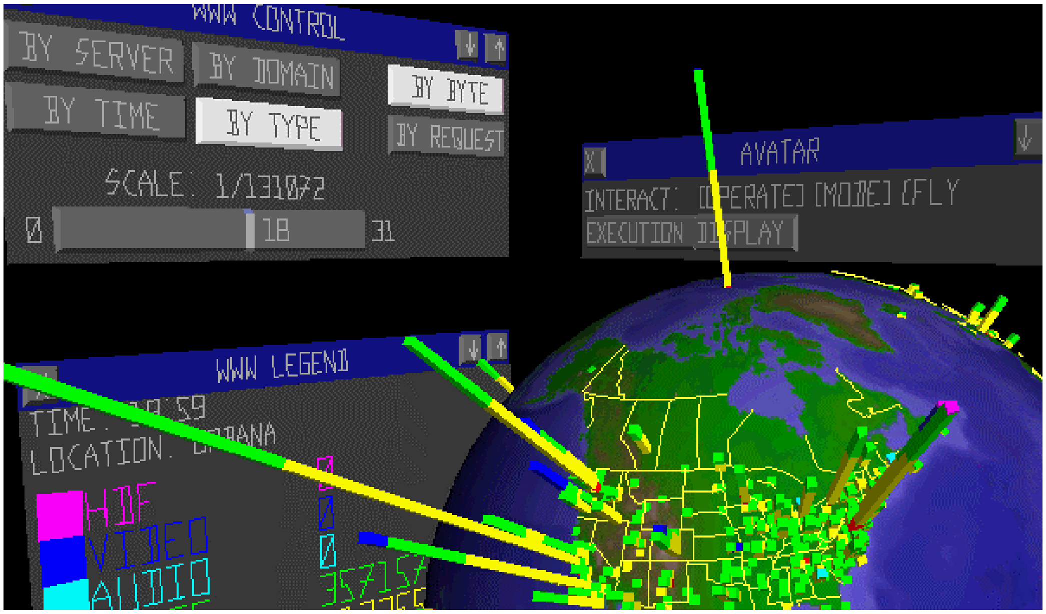

Fig. 35: Real-Time Geographic Visualization of World Wide Web Traffic

Palantir [21] is another visualisation tool for the World Wide Web traffic. This tool is able to animate world wide web traffic. It displays the origin and magnitude of a Web server's hits either in real-time or in batch mode. It can synthesize the traffic to several Web server's so as to get a global view of the hits in a multi-site organization (see Fig. 36).

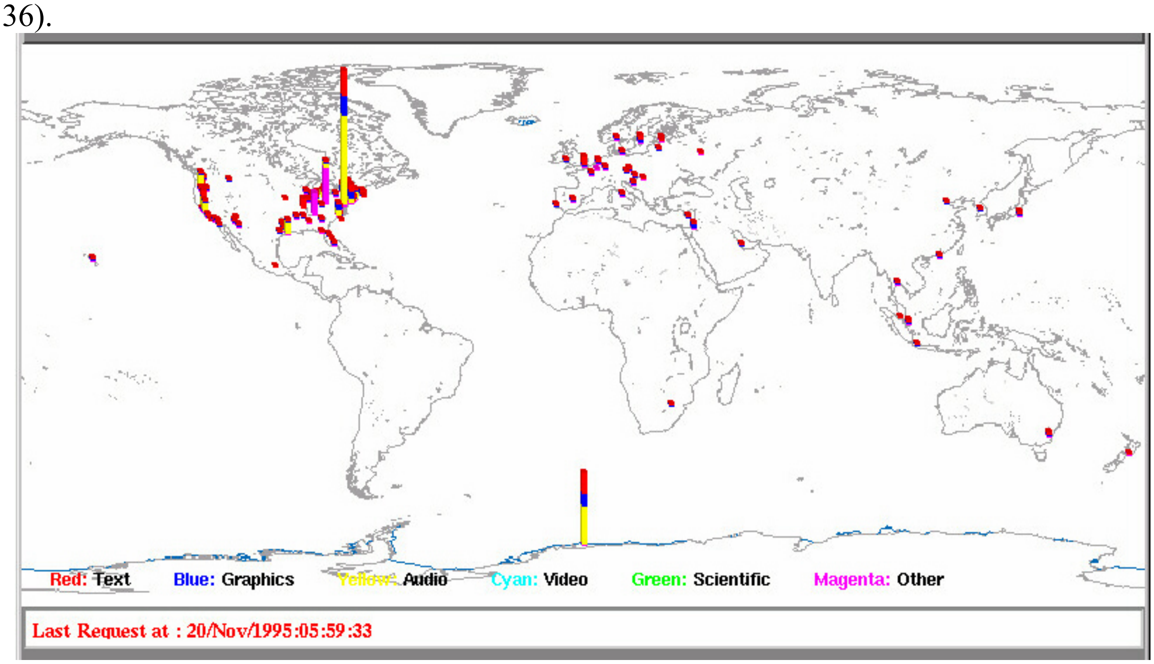

Fig. 36: Palantir

2.8.3 Suggested contribution

All mentioned systems still have a number of unsolved problems. Scaleability up to millions of symbols can be an example. This includes near-real-time clustering of such huge datasets

(made for visualisation or data analysis) and data updating as well. Another unsupported feature is usage of user-defined „logical“maps or office layout plans.
For more details of suggested solutions, see chapter 3.

3 Suggested solutions

As mentioned in chapter 1, a project has to be solved for KMi at OU. This project includes geographic and spatial system based on client-server architecture. The system architecture, as suggested, is depicted in figure Fig. 37.

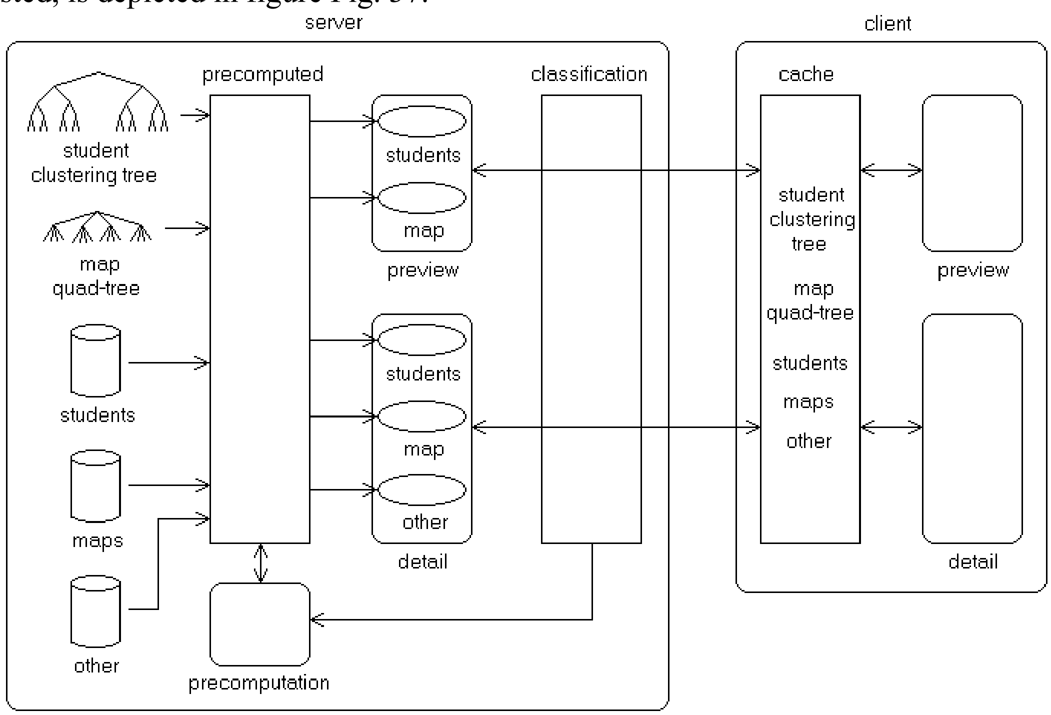

Fig. 37: Suggested system architecture

The system architecture has to be client-server type. The client should enable browsing through maps, entering queries and exploring their results. The system response should be as fast as possible. For these purposes, the client uses system of previews for querying to allow user to formulate queries. A local cache for stores recent data and avoids unnecessary file transmissions. On the other hand the server has to response as fast as possible on client queries. It uses precalculation based on user classification and behaviour prediction. Hierarchical clustering tree for entities and quad-tree for maps allow fast access to desired data and maps. The server response is adapted to client and network capabilities and user requirements.
Following chapters discuss the architecture and its parts in more detail.

3.1 Maps

The system will provide display of several map types. Several types of commonly used geographic map projections can be included into the first type. The system architecture should support additional of modules with additional geographic projections.
The system will also allow display in defined pseudospace. In our implementation, it will obviously be the space of studied courses and student membership in a certain study program or interest in certain science area. In both spaces, geographic and pseudospace, it is possible

to display not only entities, but also their relationships or interaction (for example message sending).
For some purposes, it is suitable to study time patterns of entity behaviour, therefore the system will support also time dimension in form of display with time axis and usage of time for animation.
Context map is an important part of user interface. It allows better orientation of user, because it displays larger area than a main map. At the same time, it can be used for faster position changes and selection of displayed area (see figure Fig. 38).
Each map should have an explaining key also included in our system. The key will inform about meaning of used symbols, attribute display, map detail and briefly describe map contents, it means used filter or query (see figure Fig. 38).

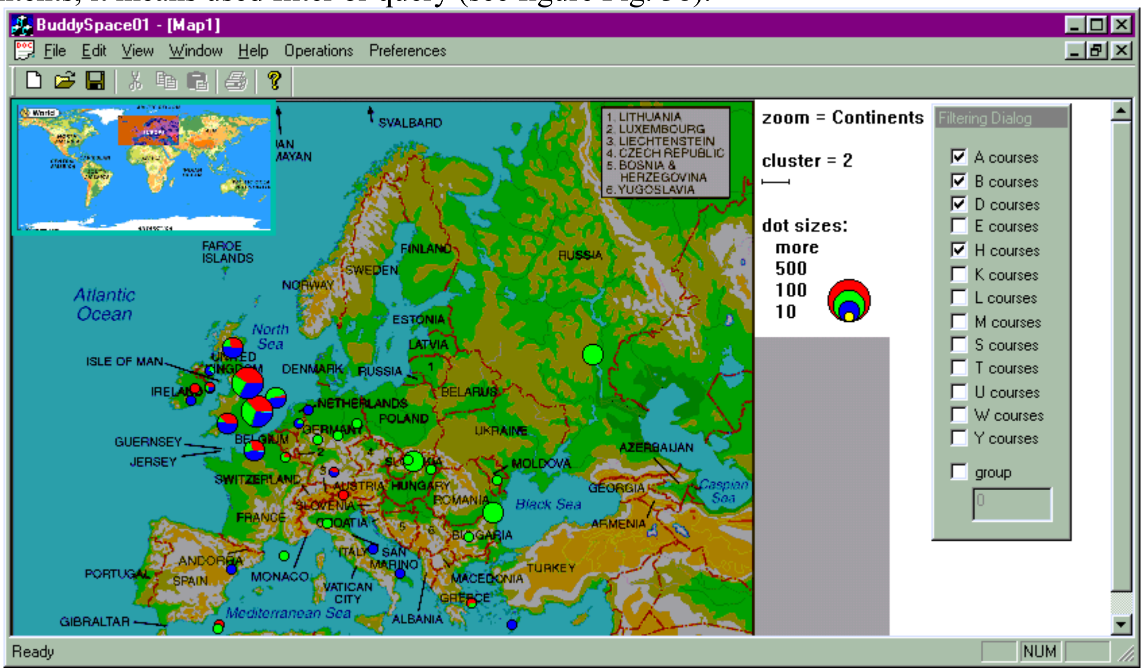

Fig. 38: System user interface.

Entities or their clusters will be displayed using symbols of adjustable type. The basic symbol ability is to visualise entity attribute. Symbols can has different type for example pie charts, bar charts or icons. The symbol size represents number of entities in cluster. Separation of attribute values in entity cluster is shown for example by colour division of symbol. Colour coding and/or symbol size can also display entity propriety, it means entity difference from desired characteristics. For example, a strong coloured entity corresponds exactly, while dull grey entities are only similar.

3.2 User interface

User work with the system consists of several steps. Main two steps, which have to be distinguished, are entering of specific query in special mode and results browsing. The query entering allows query specification and testing of effect of particular attribute changes on previewed results. Visualisation of entity details can be done in browsing.
Query entering mode is based on system of previews. It is possible to adjust selected attributes using sliders and probe their influence on query results. Number and distribution of

matching entities are shown in map (using clustering tree). Besides the exact numbers are displayed in number form. The fast response is important.
The results display mode shows query results in detail. At the same time, it allows to change display properties. Each entity or cluster details can be displayed on users request in desired form. In addition, map detail changes and planning over map are provided in this mode, which can possibly cause further queries.
Since not all users have the same hardware equipment, concerning display resolution and network bandwidth, the system environment will allow to set parameters to make the system response time fast enough and allow display appropriate to user requirements and possibilities. These parameters include for example display resolution, window size, client computer performance, and network bandwidth.

3.3 Data filtering and analysis

Filtering is an important part of the designed system. It should provide basic filtering directly based on entity (student) properties, such as studied courses or on-line status. More analysis, that is complex, has to be done, when using fuzzy queries. This appears in case of similarity searches. Such user question can for example: "Who studies similar courses as I do?" or "I'd like to know students, who communicate with some certain people, which are interested in certain problem."
Three different ways of query entering can be specified. Direct formulation is the first one. In the second case, the query or filter is automatically chosen depending on user profile and context. Question based on similarity to a given entity or set of them (including communication) represents the third way.
The query formulation is done in a system of previews to speed up communication and provide full control of query result. The system of previews is discussed in more detail in chapters 3.2 and 3.5.
Adaptive change of filter scope in areas with different changes of explored values is another important topic of research. This can allow user to formulate queries that are more precise and focus on explored phenomenon.

3.4 Clustering

As mentioned in chapter 2.3, clustering groups in some way close entities. It can be used for visualisation, preview data transmission, and search for spatially close entities.
Entity (student) postcode is used for its position determination in our system. Postal code characterises position very well in many countries. Besides, it is easy accessible, because it is a part of address. Postcode structure is hierarchical in many countries; therefore adding of its parts incrementally increases segmentation and position accuracy. Then, the top most segmentation can be done on country level.
At first, it is suitable for clustering algorithm speed up to cluster entities inside the smallest regions or regions given by an appropriately selected part of postcode. After that, the subtrees built for particular regions are joined by clustering algorithm. Thus, the clustering tree is built hierarchically up to countries or continents. The main positive of this approach is the reduction of number of entities clustered in one time. This brings rapid speed up of the algorithm.
It is possible to make changes without the need of complete rebuild of clustering tree due to its hierarchical construction from separately clustered parts. Any changes, like addition of a node or its deletion, cause changes only in subtrees, which contain the node. Thus, the need of

change propagation into neighbouring subtrees is avoided on each used "separation" level of clustering.
Further speed up will be achieved by suggested algorithm, which significantly reduces number of checked potentially closest neighbours. This algorithm uses a linear list for each dimension. The list contains sorted entities (or pointers on them) according to their coordinate in given dimension. Guaranteed distance, in that the closest neighbour is, can be determined from the closest neighbours in each list. Then, the search is reduced only on neighbours in lists, which are no further then the guaranteed distance. This algorithm brings improvement of asymptotical time complexity (for detail description see [12]).
Although, techniques for clustering tree building speed up were mentioned, the process is still quite computationally intensive. Therefore, it is suitable to use precomputation of this tree and store it. This idea is supported also by the fact, that the majority of spatial data does not change often and thus it is worthless to build the tree again every time. Possible changes can be stored either directly into the tree or into an incremental file.

3.5 Scalability

The aim is to design a system, which is able to display huge scope of entity numbers. It means, it should be able to manage up to millions of entities. For this purpose, tricks (for example background processing during idle time and computation redistribution between server and client computer) have to be used. Precalculation is next usable technique. It can be done for some typical client behaviour (can be prestored) and/or for behaviour predicted according client behaviour until this time (in idle time). System of previews is another suitable technique, which allows easy query formulation with control of fitting entity number during the formulation process (see chapter 2.5).
Basic scalability should be guaranteed using hierarchical clustering structure, which provides two main tasks. The first one is work with only selected detail level. This is easily done by selection of minimal distance between two clusters. The second one is separation of data, which is mutually remote or distant. Thus, there is no need to work with the whole set, but only related entities. Considering that clustering trees for geographic position and similarity of studied courses will be built (as it will be in our system), it will be possible to work only with the intersection of relevant subtrees in both trees.
Next possibility, how to increase service of client, is to use precomputed results. Prediction of required operations can be done according to long time observation of client behaviour and creation of groups – typical client categories. Alternatively, it can be done dynamically according to concrete behaviour of particular user. Then, both ways differ in the way of computation. In the first case, the operations (or their parts) for particular categories can be precomputed and prestored. In the second case, it is necessary to predict behaviour during the client work, thus the precomputation can be done only in idle time. While the first case requires precise position and query match, in the second case, the client behaviour can be predicted according to similarity of operation sequence and the map and data do not need to match. Precomputation of time consuming operations with small size results is more suitable then that of large size results. Thus, the suitability of precomputation depends on data type and application.
As mentioned already in chapter 2.5, system of previews is an important feature, which makes the work with huge volumes of data easier. In our system, the system of preview will be based on information stored in a clustering tree, which collects data (entity numbers with some property) for each area. Preview information will be computed from one layer of clustering

tree with suitable clustering distance. Then, clusters can be filtered and numbers of entities with given properties can be checked inside them.
Usage of all these techniques in the same time should bring a scaleable solution.

3.6 Network service

The system consists of two parts. Database containing all information and map archive lie on the server side, which serves client requests. Clients query about data stored in database and require their display in an appropriate map.
Every time, only incremental data is transmitted. It means, that a cache exists on client side, in that the transmitted data are stored for later use. This data are persistent also into next sessions. Clustering tree and quad-tree for map storing are main structures (see chapter 3.7). Since data is transmitted incrementally, the trees have to be built incrementally on client side. Used cache on client side will store different types of data separately.
Server response on a client query then consists of several parts. They can be divided into structural data and actual data. The structural data holds information about actual data integration into data structure (clustering tree and quad-tree for maps). While the actual data consists of entity record or map texture. An entity record includes information about its position, its attributes (in our case studied courses, age, sex, and so on), available actions with entity (allowing interaction) and complementary information.
User query processing consists of several steps. At first, user specifies his/her query using system of preview on client side. Then, the query is sent to server. The server selects entities from database and filters them. At the same time, it selects appropriate map for their display. All this data are sent back to the client. Then, the client generates symbols, their connections and other visualisation structures based on data obtained by server. In the end, the client displays everything according to user requirements.
Just mentioned concludes, that the query processing on server side can be divided into two independent parts. While the extraction of entity data involves a database query or possibly clustering tree pass, the generation of appropriate map is done by map quad-tree pass and read of corresponding texture (map image file). See figure Fig. 37 for suggested system architecture.
Client service depends on several factors. As mentioned in chapter 3.5, client is classified into a group according to its behaviour. Then, actions are precomputed and some not specified settings are filled automatically according to the group. The response form is also dependent on client requirements. These requirements include display resolution, window size, response time, and so on. When not all the requirements can be met, their optimisation is done.

3.7 Data structures

Data structures used in designed system depend on its purpose (as mentioned in chapter 2.7). Our system should provide entity and their relationships visualisation in a geographical map. Thus, the map data representation and storage of entities and their relationships were separated.
A combination of extern commercial database and clustering tree, which will be administrated directly by the system, will be used for storage of entity data. Extern relational database will contain complete entity database. Detail information about entities can be obtained by a query into the database and can be further filtered, whenever the information in clustering tree will not be sufficient. The main function of clustering tree is to represent entity distribution structure. Thus, it allows the separation of complete database into parts and so speed up

queries into it. There can exist more clustering trees in the system. A clustering tree can be constructed for each visualisation space or even for an important set of properties. Then it enables better orientation in entity organisation and speeds up processing with respect to these properties. Clustering tree and its construction were described already in chapter 3.4.
The raster form was chosen for map representation. Since only the appropriate part of map is sent to client, it is suitable to combine it from tiles, which are stored separately. A quad-tree is used for fast access to desired tile in desired level of detail. One level of detail is stored in each quad-tree layer. It is possible to create any level of detail (which is not stored) using a special combination (transparency) technique on client side.
Cluster position has to be checked when clustering entities to avoid placing them into locations, which are not suitable in some way (for example oceans). "Forbidden areas" have to be stored for each map for this purpose. When placing a cluster into a map, the cluster centre of gravity has to be checked, if it does not belong into a forbidden area. If it does, the closest boundary of forbidden area (land) has to be found. It is defined as the place with the smallest weighted distance from the centre of gravity.
Thus, either a map is stored in one layer or it can be stored in more layers according to system purposes (optional displays of roads, cities with different population sizes, and so on). Other information about Open University organisation (for example catchment areas, administration centres, and used study language) can be stored in further layers.
One of the other stored and visualised structures can be entity relationships (for example communication among students). A graph or a net can be used for this purpose.

4 Evaluation

The system evaluation will be done iteratively with help of beta-testers (in the early phase), and testers later.
The system users can be divided into two classes. The first class is made by OU administrators. They will explore geographic distribution of administrative centres, distribution of students from different regions into different courses, and so on. Students belong into the second group. They will look for their colleagues with defined proprieties and communicate on peer-to-peer level.
The thesis will include user comments, analysis of the benefits and suggested improvements.

5 Conclusion and future work

Techniques and algorithms used by spatial and geographic information systems for entity distribution visualisation were described with stress to thier scalability. These were visualisation and mapping techniques, user interface problems, clustering analysis, used data structures and network services. Further, several existing systems were mentioned as examples.
In the second part of the paper, suggested approaches and improvements were discussed and the system architecture described. The suggested system should provide new possibilities, mainly the scalability of system when working with very large numbers of entities.

6 References

[1] D. J. Buckley: The GIS Primer - An Introduction to Geographic Information Systems. February 1997. http://www.innovativegis.com/education/primer/primer.html

[2] P. H. Dana: Map Projection Overview. Department of Geography, University of Texas at Austin, 1995, http://www.colorado.edu/geography/gcraft/notes/mapproj/mapproj_f.html

[3] G. Dang, C. North: Dynamaps: Dynamic Queries on Choropleth Maps. http://www.cs.umd.edu/hcil/census/techreport.htm

[4] K. Doan, C. Plaisant, B. Shneiderman and T. Bruns: Interface and Data Architecture for Query Preview in Networked Information Systems. Technical report (April 1997) - revised version in *ACM TOIS as Practice and Experience Paper*. A short early version also appeared in *SIGMOD Record, Vol.26*, No.1, pp. 75--81 March 1997, as Previews for Networked Information Systems: A Case Study with NASA Environmental Data. (http://www.cs.umd.edu/local-cgi-bin/hcil/rr.pl?number=97-09)

[5] M. Dodge and R. Kitchin: *Mapping cyberspace*. Routledge 2001

[6] J. S. Donath: Visual Who: Animating the affinities and activities of an electronic community. *ACM Multimedia 95 - Electronic Proceedings*. (http://smg.media.mit.edu/people/judith/VisualWho/VisualWho.html)

[7] ESRI: ArcGIS 8.1. http://www.esri.com/software/arcgis/index.html

[8] GeoComm International Corporation: Data Projections. GeoComm International Corporation, http://data.geocomm.com/helpdesk/projections.html

[9] Debbie Harmon: Microsoft MapPoint 2000 Review. GeoPlace, http://www.geoplace.com/bg/1999/productreview/699qt.asp

[10] HCIL: Dynamic Queries and Query Previews for networked information systems: the case of NASA EOSDIS. http://www.cs.umd.edu/hcil/eosdis/

[11] JMP: Projection Animation. Hammond World Atlas Corporation, http://www.hammondmap.com

[12] J. Komzak: Hierarchical clustering speed up using position lists and data position hierarchy. KMI-TR-112, Knowledge Media Institute, Open University, Milton Keynes, UK, June 2001 (http://kmi.open.ac.uk/publications/techreports.html)

[13] S. E. Lamm and D. A. Reed: Real-Time Geographic Visualization of World Wide Web Traffic. *Fifth International World Wide Web Conference* May 6-10, 1996 (http://www5conf.inria.fr/fich_html/papers/P49/Overview.html)

[14] R. Laurini, D. Thompson: *Fundamentals of Spatial Information Systems*. Academic Press 1992

[15] Matrix.Net: Internet Weather Report (IWR). http://www.mids.org/weather/

[16] Microsoft: Microsoft MapPoint. http://www.microsoft.com/office/mappoint/default.htm

[17] M. Monmonier: *Proc mapy lzou. (How to lie with maps, 2nd Edition.)* Computer Press, Prague 2000 (The University of Chicago 1996)

[18] T. Munzner and E. Hoffman and K. Claffy and B. Fenner: Visualizing the Global Topology of the MBone. *Proceedings of the 1996 IEEE Symposium on Information Visualization*, pp. 85-92, October 28-29 1996, (http://www-graphics.stanford.edu/papers/mbone/)

[19] J. Neystadt and N. Har'El: iGuide. http://www.iguide.co.il/maps

[20] S. Oviatt: Multimodal Interfaces for Dynamic Interactive Maps. http://www.cs.nwu.edu/~bquinn/multimodal/papers/oviattMaps/

[21] N. Papadakakis, E. P. Markatos and A. E. Papathanasiou: Palantir: A Visualization Tool for the World Wide Web. http://archvlsi.ics.forth.gr/html_papers/INET98_Palantir/

[22] C. Plaisant, M. Venkatraman, K. Ngamkajornwiwat, R. Barth, B. Harberts, W. Feng: Refining query previews techniques for data with multivalued attributes: The case of NASA EOSDIS. in *Proc. of Advanced Digital Libraries 99 (ADL'99)*, IEEE Computer Society Press (May 1999) 50-59, (http://www.cs.umd.edu/local-cgi-bin/hcil/rr.pl?number=98-17)

[23] U. Rauschenbach, R. Schultz, H. Schumann: Quality and Resource Controlled Transmission of Images. Urban, B. (ed.): *"Multimedia '96" Proceedings of the EUROGRAPHICS workshop in Rostock, Germany, May 28-30, 1996*, Springer Wien New York, 1996, (http://wwwicg.informatik.uni-rostock.de/Projekte/MoVi/Publications/EGMM96/)

[24] N. Shiode and M. Dodge: Internet Address Space. http://www.geog.ucl.ac.uk/casa/martin/internetspace/

[25] B. Shneiderman: Dynamic queries, starfield displays, and the path to Spotfire. 1999, http://www.cs.umd.edu/hcil/spotfire/

[26] T. Strothotte: *Computational Visualization Graphics*, Abstraction, and Interactivity. Springer 1998

[27] A. Wood, N. Drew, R. Beale and B. Hendley: Three dimensional Graph Visualisation HyperSpace. http://www.cs.bham.ac.uk/~amw/hyperspace/www95/

[28] M. Zuker: Hierarchical Clustering Algorithm. Center for Computational Biology, Washington University in St. Louis, 2000-02-25, http://www.ibc.wustl.edu/~zuker/Bio-5495/multalign-html/node7.html

Acknowledgement

The Knowledge Media Institute at The Open University (UK) provided support for this work. The research has been performed in the frame of PhD study at the Department of Computer Science and Engineering (Czech Technical University in Prague – Czech Republic) under supervision of Pavel Slavik.