

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN  
BỘ MÔN CÔNG NGHỆ PHẦN MỀM**

**TRƯƠNG CÔNG THÀNH - 0112369  
TRẦN VĂN TÁNH - 0112365**

**NGHIÊN CỨU CÔNG NGHỆ WEBGIS VÀ XÂY  
DỰNG TRANG WEB DỰ BÁO THỜI TIẾT KHU  
VỰC NAM BỘ**

**KHÓA LUẬN CỬ NHÂN TIN HỌC**

**GIÁO VIÊN HƯỚNG DẪN  
T.S TRƯƠNG MỸ DUNG**

**NIÊN KHÓA 2001 - 2005**





# LỜI CẢM ƠN

Luận văn của chúng em sẽ rất khó hoàn thành nếu không có sự truyền đạt kiến thức quý báu và sự hướng dẫn tận tình của cô Trương Mỹ Dung. Chúng em xin chân thành cảm ơn sự chỉ bảo của các thầy.

Chúng em xin trân trọng cảm ơn quý Thầy cô trong Khoa Công nghệ thông tin trường Đại học Khoa học Tự nhiên Tp.Hồ Chí Minh đã tận tình giảng dạy, truyền đạt những kiến thức quý báu trong suốt bốn năm học làm nền tảng và tạo điều kiện cho chúng em được thực hiện luận văn này.

Đặc biệt xin được gửi lời cảm ơn đến anh Nguyễn Minh Giám, trưởng phòng dự báo Trung tâm khí tượng Thủy văn khu vực Nam bộ. Xin cảm ơn sự hướng dẫn, giúp đỡ nhiệt tình, tạo điều kiện cho chúng em tìm hiểu kiến thức về hệ thống Khí tượng Thủy văn..

Mặc dù đã cố gắng nỗ lực hết sức mình, song chắc chắn luận văn không khỏi còn nhiều thiếu sót. Chúng em rất mong nhận được sự thông cảm và chỉ bảo tận tình của quý Thầy cô và các bạn.

Tp.HCM, 7/2005

Nhóm sinh viên thực hiện

Trương Công Thành- Trần Văn Tánh



Bố cục của luận văn được chia làm các phần sau đây

## **Mở đầu**

Giới thiệu công nghệ GIS, những hạn chế và nhu cầu mở rộng. Các giải pháp và hướng nghiên cứu hiện nay. Đồng thời giới thiệu chung về đề tài, ý nghĩa và các mục tiêu của đề tài.

## **Chương 1: WebGIS- Công nghệ GIS qua mạng**

Giới thiệu về WebGIS. Gồm những khái niệm cơ bản về WebGIS, mô hình hoạt động. Các giải pháp kiến trúc lý thuyết và thực tế trong nước và trên thế giới.

## **Chương 2: MapServer- WebGIS Application**

Tìm hiểu về MapServer, một Application mã nguồn mở rất mạnh hiện nay sử dụng trong công nghệ WebGIS.

## **Chương 3: Bài toán dự báo thời tiết – Biểu diễn trên bản đồ.**

Giới thiệu bài toán dự báo thời tiết, cách giải quyết các yêu cầu cho bài toán dự báo dùng bản đồ.

## **Chương 4: Xây dựng Website Thời tiết Nam bộ.**

Bao gồm các phần Phân tích và Thiết kế website Thời tiết Nam bộ.

## **Chương 5: Cài đặt và thử nghiệm**

Tổ chức dữ liệu, mô hình thiết kế hệ thống. Việc cài đặt, cấu hình hệ thống cho một ứng dụng WebGIS gồm các phần Server, Client, Database...

## **Kết luận**

Tóm tắt lại các vấn đề đã được đặt ra trong luận văn, cách giải quyết, kết quả đạt được và đề ra một số hướng phát triển trong tương lai.

# MỤC LỤC

Danh sách các hình .....	iii
Danh sách các bảng .....	v
Một số khái niệm và thuật ngữ .....	vi
MỞ ĐẦU .....	viii
Chương 1 : WebGIS- Công nghệ GIS qua mạng .....	1
1.1 Bản đồ - Cách biểu diễn thế giới thực .....	1
1.1.1 Khái niệm về bản đồ .....	1
1.1.2 Trái đất quả cầu địa lý .....	1
1.1.3 Cơ sở toán học cho bản đồ .....	4
1.1.4 Các phương pháp thể hiện bản đồ .....	9
1.2 Dữ liệu GIS .....	14
1.2.1 Các dạng dữ liệu của GIS .....	14
1.2.2 Các mô hình dữ liệu được dùng .....	15
1.2.3 Mô hình dữ liệu đồ họa .....	16
1.2.4 Kết nối dữ liệu không gian và dữ liệu thuộc tính .....	24
1.3 Giới thiệu về WebGIS .....	25
1.4 Mô hình xử lý và kiến trúc triển khai WebGIS .....	26
1.4.1 Kiến trúc WebGIS và các bước xử lý .....	26
1.4.2 Các kiến trúc triển khai .....	29
1.5 Các chuẩn trao đổi WebGIS hiện nay .....	36
1.5.1 Vấn đề trong việc trao đổi dữ liệu của hệ thống WebGIS .....	36
1.5.2 Giải pháp cho vấn đề chia sẻ dữ liệu .....	37
Chương 2 : MapServer – WebGIS Application .....	40
2.1 Lược sử phát triển .....	40
2.2 Các thành phần và mô hình xử lý của MapServer .....	41
2.2.1 Các thành phần của MapServer .....	41
2.2.2 Quy trình xử lý .....	44
2.3 Tìm hiểu Mapfile .....	45
2.3.1 Map Object .....	46
2.3.2 Layer Object .....	49
2.3.3 Query Map Object .....	53
2.3.4 Projection Object .....	53
2.4 Xử lý kết nối các loại dữ liệu .....	54
2.4.1 Kết nối dữ liệu mặc định ESRI Shapefiles .....	54
2.4.2 Kết nối dữ liệu Raster .....	54
2.4.3 Kết nối dữ liệu dùng thư viện OGR .....	57
2.4.4 Kết nối dữ liệu dùng WMS .....	61
2.4.5 Kết nối dữ liệu dùng WFS .....	63
Chương 3 : Bài toán dự báo thời tiết – Biểu diễn trên bản đồ .....	66
3.1 Khảo sát hiện trạng .....	66
3.1.1 Giới thiệu bài toán dự báo .....	66
3.1.2 Số liệu thời tiết trong dự báo .....	66

3.1.3 Hệ thống KTTV khu vực Nam bộ.....	68
3.2 Phân tích và xác định yêu cầu .....	70
3.3 Các vấn đề trong bài toán dự báo thời tiết bằng bản đồ .....	71
3.3.1 Thể hiện bản đồ .....	71
3.3.2 Chinh sửa dữ liệu vector MapInfo.....	72
3.3.3 Truy vấn dữ liệu với bản đồ .....	75
Chương 4 : Xây dựng Website Thời tiết Nam bộ .....	77
4.1 Thiết kế kiến trúc hệ thống.....	77
4.2 Xây dựng mô hình Use –case .....	78
4.2.1 Xác định Actor và Use case.....	78
4.2.2 Mô hình Use – case .....	79
4.2.3 Đặc tả Use-case .....	79
4.3 Thiết kế một số màn hình .....	89
4.3.1 Màn hình Chính .....	89
4.3.2 Màn hình Thời tiết trong ngày.....	90
4.3.3 Màn hình Thời tiết vài ngày tới.....	92
4.3.4 Màn hình Tìm kiếm .....	93
4.3.5 Màn hình Góp ý.....	94
4.3.6 Màn hình Liên lạc.....	95
4.3.7 Màn hình Gửi tin .....	96
Chương 5 : Cài đặt và thử nghiệm .....	97
5.1 Tổ chức dữ liệu dự báo .....	97
5.1.1 Hình thức lưu trữ dữ liệu .....	97
5.1.2 Sơ đồ logic.....	97
5.2 Cấu hình và cài đặt hệ thống Server-Client.....	101
5.2.1 Cài đặt trang web.....	101
5.2.2 Cài đặt CSDL Thời tiết.....	104
5.2.3 Cấu hình MapServer.....	105
5.2.4 Cấu hình Client.....	107
5.3 Thử nghiệm.....	108
KẾT LUẬN .....	109
Tài liệu tham khảo .....	110
Phụ lục .....	112
Yêu cầu cấu hình .....	112
Hướng dẫn sử dụng.....	112

## Danh sách các hình

Hình 1-1 Hình dạng ellipsoid của Trái Đất .....	2
Hình 1-2 Các tham số của GEOID .....	2
Hình 1-3 Hệ thống kinh độ và vĩ độ .....	4
Hình 1-4 Phép chiếu hình nón .....	6
Hình 1-5 Phép chiếu phương vị .....	6
Hình 1-6 Phép chiếu hình trụ .....	7
Hình 1-7 Phép chiếu thẳng .....	7
Hình 1-8 Phép chiếu ngang .....	7
Hình 1-9 Phép chiếu nghiêng .....	8
Hình 1-10 Bản đồ đường nét .....	9
Hình 1-11 Bản đồ dạng ảnh .....	9
Hình 1-12 Khái quát hóa theo tỉ lệ .....	13
Hình 1-13 Các dạng dữ liệu GIS .....	14
Hình 1-14 Ví dụ thế giới thực .....	16
Hình 1-15 Biểu diễn thế giới thực bằng Raster .....	16
Hình 1-16 Mô hình Vectơ biểu diễn thế giới thực .....	17
Hình 1-17 Mô hình dữ liệu Raster .....	18
Hình 1-18 Tổ chức CSDL KGian Raster .....	19
Hình 1-19 Tổ chức CSDL KGian Raster .....	19
Hình 1-20 Thể hiện vật thể dạng điểm đường vùng theo tọa độ x, y .....	20
Hình 1-21 Mô hình dữ liệu mì ống ( Spaghetti data model) .....	21
Hình 1-22 Mô hình dữ liệu Tô pô (Topology) .....	22
Hình 1-23 Ảnh chụp bề mặt địa hình .....	23
Hình 1-24 Mô hình dữ liệu vectơ kiểu TIN .....	23
Hình 1-25 Sơ đồ kết nối dữ liệu .....	24
Hình 1-26 Kết nối dữ liệu không gian và thuộc tính .....	25
Hình 1-27 Kiến trúc WebGIS .....	27
Hình 1-28 Các dạng yêu cầu từ phía Client .....	28
Hình 1-29 Cấu hình Server Side .....	30
Hình 1-30 Cấu hình Client side .....	32
Hình 1-31 Tích hợp xử lý GIS vào trình duyệt .....	34
Hình 1-32 Kết hợp Client side và Server side .....	35
Hình 1-33 Dữ liệu GIS trong kiến trúc WebGIS đơn thể .....	36
Hình 1-34 Chia xẻ dữ liệu GIS giữa các nhóm ứng dụng .....	37
Hình 1-35 Các chức năng của một WMS .....	38
Hình 1-36 Các tham số trong chuỗi URL thực hiện chức năng GetMap .....	38
Hình 2-1 Xử lý với file Template đơn giản .....	42
Hình 2-2 Xử lý định dạng file Template .....	43
Hình 2-3 Quy trình xử lý của MapServer .....	44
Hình 2-4 Mô hình đối tượng trong Mapfile .....	45
Hình 2-5 Chồng lấp các layer .....	45
Hình 2-6 Danh sách font sử dụng .....	46
Hình 2-7 Thứ tự được vẽ của các layer .....	49



Hình 3-1 Trao đổi số liệu thời tiết .....	68
Hình 3-2 Hoạt động hệ thống dự báo .....	69
Hình 3-3 Xác định phạm vi bằng hình cố định .....	71
Hình 3-4 Chuyển đổi hệ tọa độ bằng MapInfo .....	72
Hình 3-5 Hiện tượng không khớp khi hiệu chỉnh bản đồ .....	73
Hình 3-6 Nấn chỉnh bản đồ dùng MapInfo .....	74
Hình 3-7 Thêm vùng chọn cho đối tượng tỉnh - tp/hố .....	75
Hình 3-8 Bài toán truy vấn dữ liệu .....	76
Hình 4-1 Kiến trúc hệ thống .....	77
Hình 4-2 Mô hình Use-Case .....	79
Hình 4-3 Màn hình chính .....	89
Hình 4-4 Màn hình thời tiết trong ngày .....	90
Hình 4-5 Các chức năng thao tác với bản đồ .....	90
Hình 4-6 Chọn layer hiển thị .....	91
Hình 4-7 Hướng dẫn thực hiện .....	91
Hình 4-8 Tìm vị trí của tỉnh .....	91
Hình 4-9 Màn hình thời tiết vài ngày tới .....	92
Hình 4-10 Màn hình tìm kiếm .....	93
Hình 4-11 Màn hình Góp ý .....	94
Hình 4-12 Màn hình Liên lạc .....	95
Hình 4-13 Màn hình Gửi tin .....	96
Hình 5-1 Sơ đồ logic dữ liệu .....	97
Hình 5-2 Đặt thuộc tính Chia sẻ thư mục web .....	101
Hình 5-3 Tạo thư mục Virtual Directory .....	102
Hình 5-4 Đặt bí danh (tên trang web) .....	102
Hình 5-5 Chọn thư mục chứa project .....	103
Hình 5-6 Sử dụng Enterispe Manager .....	104
Hình 5-7 Chọn file Backup CSDL .....	105
Hình 5-8 Thiết lập biến môi trường PROJ_LIB .....	106
Hình 5-9 Hiệu chỉnh biến môi trường PATH .....	107

## Danh sách các bảng

Bảng 1-1 Các phương pháp thể hiện bản đồ .....	12
Bảng 1-2 So sánh mô hình dữ liệu Raster và Vector.....	24
Bảng 1-3 Chiến thuật Server-side.....	32
Bảng 1-4 Công việc tại Client với chiến thuật Client side.....	35
Bảng 3-1 Các chức năng trên bản đồ.....	70
Bảng 3-2 Các chức năng dự báo thời tiết .....	70
Bảng 4-1 Nút bấm và chức năng tương ứng.....	91
Bảng 4-2 Các layer hiển thị bản đồ .....	91
Bảng 5-1 Ký hiệu trường dữ liệu.....	98
Bảng 5-2 Danh sách các bảng dữ liệu .....	98
Bảng 5-3 Chi tiết bảng Tinh_TP .....	99
Bảng 5-4 Chi tiết bảng KhuVuc.....	99
Bảng 5-5 Chi tiết bảng Cac_Buoi.....	100
Bảng 5-6 Chi tiết bảng Loai_ThoiTiet .....	100
Bảng 5-7 Chi tiết bảng ThongTin_DuBao .....	101

## Một số khái niệm và thuật ngữ

Khái niệm	Định nghĩa	Ghi chú
GIS (Geography Information System)	Hệ thống thông tin địa lý. Sử dụng công nghệ này là một công nghệ dựa trên máy tính để xây dựng bản đồ, phân tích và xử lý các đối tượng tồn tại và các sự kiện xảy ra trên trái đất.	
Thông tin không gian	Thông tin về những đặc điểm liên quan đến hình dạng, vị trí, quan hệ của các đối tượng địa lý.	Bao gồm hai dạng: <ul style="list-style-type: none"> <li>• Dạng hình học: mô tả các đặc điểm hình dạng, vị trí. Ví dụ như tọa độ của điểm, đường...</li> <li>• Dạng Topology: mô tả quan hệ giữa các đối tượng hình học. Ví dụ như những vùng nào kề với một vùng xác định.</li> </ul>
Thông tin phi không gian (thông tin thuộc tính)	Thông tin về những đặc điểm liên quan đến thống kê, thông tin số, thông tin đặc trưng gán cho mỗi thuộc tính của đối tượng	Ví dụ như tên đường phố, dân số...
Layer	Lớp chứa một nhóm các đối tượng thuần nhất với vị trí của	Các thành phần đồ họa trong cơ sở dữ liệu GIS

	chúng theo hệ tọa độ chung.	thường được mô tả bằng nhiều lớp (layer).
WebGIS	WebGIS là hệ thống thông tin địa lý phân tán trên một mạng các máy tính để tích hợp, hiển thị, trao đổi các thông tin địa lý.	
Raster		
Vector		
Server side		
Client side		
OGC (Open Geographic Consortium)	Tổ chức tập hợp các nhà phát triển nghiên cứu đưa ra các chuẩn cho hệ thống tin địa lý.	
WMS (Web Map Server/Service)	Chuẩn WebGIS hỗ trợ việc trao đổi thông tin địa lý dưới dạng ảnh đồ họa.	
WFS (Web Feature Server/Service)	Chuẩn WebGIS hỗ trợ việc trao đổi thông tin địa lý dưới dạng tài liệu XML.	
MapInfo	Phần mềm làm việc trên hệ thống địa lý lý của hãng MapInfo.	
DNN (DotNetNuke)	Là môi trường hỗ trợ xây dựng trang web sử dụng công nghệ Portal. Mã nguồn mở	

## MỞ ĐẦU

### ❖ Hệ thống thông tin địa lý GIS (Geographic Information System)

Thông tin địa lý được thể hiện chủ yếu dưới dạng bản đồ đã ra đời từ xa xưa. Các bản đồ trước tiên được phác thảo để mô tả vị trí, cảnh quan, địa hình... Bản đồ chủ yếu gồm những điểm và đường. Tuy nhiên bản đồ dạng này thích hợp cho quân đội và các cuộc thám hiểm hơn là được sử dụng như một công cụ khai thác tiềm năng của địa lý.

Bản đồ vẫn tiếp tục được in trên giấy ngay cả khi máy tính đã ra đời một thời gian dài trước đó. Bản đồ in trên giấy bộc lộ những hạn chế như: thời gian xây dựng, đo đạc, tạo lập rất lâu và tốn kém. Lượng thông tin mang trên bản đồ giấy là hạn chế vì nếu mang hết các thông tin lên bản đồ sẽ gây khó đọc. Bên cạnh đó bản đồ giấy không thể cập nhật theo thời gian được vv...

Ý tưởng mô hình hóa không gian lưu trữ vào máy tính, tạo nên bản đồ máy tính. Đó là bản đồ đơn giản có thể mã hóa, lưu trữ trong máy tính, sửa chữa khi cần thiết, có thể hiển thị trên màn hình và in ra giấy.

Tuy nhiên các nhà nghiên cứu nhận thấy nhiều vấn đề địa lý cần phải thu thập một lượng lớn thông tin không phải là bản đồ.

Lúc này khái niệm Hệ thống thông tin địa lý (GIS) ra đời thay thế cho thuật ngữ bản đồ máy tính.

GIS được hình thành từ các ngành khoa học: Địa lý, Bản đồ, Tin học và Toán học. Chỉ đến những năm 80 thì GIS mới có thể phát huy hết khả năng của mình do sự phát triển mạnh mẽ của công nghệ phần cứng

Bắt đầu từ thập niên 80, GIS đã trở nên phổ biến trong các lĩnh vực thương mại, khoa học và quản lý, chúng ta có thể gặp nhiều cách định nghĩa về GIS:

- Là một tập hợp của các phần cứng, phần mềm máy tính cùng với các thông tin địa lý (mô tả không gian). Tập hợp này được thiết kế để có thể thu thập, lưu trữ, cập nhật, thao tác, phân tích, thể hiện tất cả các hình thức thông tin mang tính không gian.
- GIS là một hệ thống máy tính có khả năng lưu trữ và sử dụng dữ liệu mô tả các vị trí (nơi) trên bề mặt trái đất
- Một hệ thống được gọi là GIS nếu nó có các công cụ hỗ trợ cho việc thao tác với dữ liệu không gian - Cơ sở dữ liệu GIS là sự tổng hợp có cấu trúc các dữ liệu số hóa không gian và

phi không gian về các đối tượng bản đồ, mối liên hệ giữa các đối tượng không gian và các tính chất của một vùng của đối tượng

- GIS là từ viết tắt của:

- + G: Geographic - dữ liệu không gian thể hiện vị trí, hình dạng (điểm, tuyến, vùng)
- + I : Information - thuộc tính, không thể hiện vị trí (như mô tả bằng văn bản, số, tên...)
- + S: System - Sự liên kết bên trong giữa các thành phần khác nhau (phần cứng, phần mềm)

Tóm lại, hệ thống thông tin địa lý (Geographical Information System) là một hệ thống phần mềm máy tính được sử dụng trong việc vẽ bản đồ, phân tích các vật thể, hiện tượng tồn tại trên trái đất. Công nghệ GIS tổng hợp các chức năng chung về quản lý dữ liệu như hỏi đáp (query) và phân tích thống kê (statistical analysis) với sự thể hiện trực quan (visualization) và phân tích các vật thể hiện tượng không gian (geographic analysis) trong bản đồ. Sự khác biệt giữa GIS và các hệ thống thông tin thông thường là tính ứng dụng của nó rất rộng trong việc giải thích hiện tượng, dự báo và qui hoạch chiến lược.

### ❖ Các giải pháp và ứng dụng GIS

Các hệ thống thông tin địa lý GIS đều cung cấp các công cụ cho phép tạo lập bản đồ, tổng hợp các thông tin liên quan đến các thực thể trên bản đồ, thể hiện các sự kiện, thể hiện các ý tưởng, giải quyết các bài toán phức tạp trong thực tế. GIS có thể được sử dụng trong nhiều lãnh vực, bởi cá nhân, gia đình, trường học, hay các cơ quan, tổ chức nghiên cứu. Tạo bản đồ và phân tích bản đồ không phải là mới, nhưng GIS đóng vai trò nâng cao chất lượng, độ chính xác và nhanh hơn so với cách làm bằng tay truyền thống. Và, trước khi có GIS, chỉ một số ít người có khả năng sử dụng thông tin địa lý trong việc ra quyết định và giải quyết vấn đề.

Ngày nay, GIS là một công nghệ "đắt giá", có hàng trăm ngàn người trên thế giới đang làm việc với nó. Các nhà chuyên môn của hầu hết các lãnh vực đang dần dần nhận thấy lợi ích trong phương pháp suy nghĩ và làm việc theo phương diện địa lý. GIS không phải chỉ dùng để tạo ra những bản đồ tĩnh, mà nó còn cho phép tạo ra các bản đồ đẹp nhiều màu sắc và hơn thế nữa là khả năng tạo bản đồ động. Khả năng tạo lập bản đồ động giúp người dùng có thể lựa chọn và loại bỏ bất cứ các thành phần nào trên bản đồ

nhằm phân tích một cách nhanh chóng các nhân tố khác biệt ảnh hưởng đến mô hình và ngoài ra giúp việc đưa ra các quyết định đối với những vấn đề phức tạp.

Các giải pháp về GIS thường được chia ra làm hai nhóm chính

- Giải quyết các bài toán phức tạp liên quan đến mạng giao thông  
Các bài toán như tìm kiếm đường đi tối ưu, điều phối lộ trình giao thông... thường được áp dụng trên các hệ thống máy tính lớn, có cấu hình mạnh:

- Hiển thị và tìm kiếm các thông tin bản đồ.

Đây là dạng ứng dụng bản đồ điện tử, cung cấp các khả năng cho phép người sử dụng xem bản đồ và tìm kiếm một số thông tin cần thiết, thường được áp dụng trên các máy tính thông thường và nhỏ.

Với mỗi nhóm trên có rất nhiều ứng dụng GIS. Từ những bài toán tìm đường như tìm đường trong thành phố, tìm đường trên xe buýt cho đến các vấn đề quản lý rừng, quản lý khai thác khoáng sản... và cả ứng dụng vào trong ngành Khí tượng Thủy văn để góp phần dự báo thời tiết..

### ❖ Nhu cầu mở rộng, đưa GIS lên mạng

GIS đã được ứng dụng từ vài thập niên trước đây, nhưng dường như GIS vẫn chưa đến được với mọi người. Lý do là, trước nay các ứng dụng GIS hầu hết chạy trên máy tính đơn. Với những máy tính này cần thiết phải cài đặt các module xử lý GIS (dưới dạng các dll, hay các Active X)...điều này cản trở khả năng ứng dụng GIS rộng rãi.

Ví dụ:

*Khi một người cần biết tuyến xe buýt để di chuyển thì ngoài trừ khi anh ta trang bị một Pocket PC cài ứng dụng Tìm đường xe buýt còn không anh phải trở về nhà hay đến cơ quan tìm đến đúng máy tính được cài ứng dụng này để tìm kiếm thông tin.*

Từ ví dụ này cho thấy với các ứng dụng GIS mang tính cộng đồng hoặc khi cần có thể sử dụng bất kể nơi đâu, thì mô hình ứng dụng chạy trên máy đơn là không đáp ứng được.

Như đã biết Internet ra đời và đã thu ngắn khoảng cách giữa mọi người, và cho phép tìm kiếm thông tin mọi lúc mọi nơi. Mô hình ứng dụng GIS chạy trên nền Internet cho phép mọi người dùng bất kì công cụ nào (máy PC, máy laptop, mobile, Pocket PC...) có thể truy cập Internet tìm kiếm được thông tin mình cần.

### ❖ Mục tiêu của đề tài

Hiện nay, tại nước ta công nghệ GIS không phải là một công nghệ mới. Nhưng hiện chỉ có một số ít viện nghiên cứu, các cơ quan và vài công ty là có nghiên cứu và sử dụng GIS. Về WebGIS số lượng người nghiên cứu thì còn ít hơn.

Mục tiêu của đề tài được chia ra làm hai phần chính như sau

### Công nghệ WebGIS

- Nghiên cứu về bản đồ
- Nghiên cứu về công nghệ WebGIS
- Tìm hiểu MapServer một triển khai công nghệ WebGIS mã nguồn mở

### Xây dựng ứng dụng website Thời tiết trên nền bản đồ sử dụng WebGIS

- Phân tích hiện trạng của hệ thống dự báo Khí tượng Thủy văn khu vực Nam bộ
- Khảo sát các yêu cầu
- Tìm hiểu các loại dữ liệu bản đồ (dạng MapInfo)
- Viết ứng dụng



## Chương 1 : WebGIS- Công nghệ GIS qua mạng

### 1.1 Bản đồ - Cách biểu diễn thế giới thực

#### 1.1.1 Khái niệm về bản đồ

Bản đồ là một mô hình của các thực thể và hiện tượng trên trái đất, trong đó thực thể được thu nhỏ, đơn giản về các hiện tượng được khái quát hóa để thể hiện được trên mặt phẳng bản vẽ. Bản đồ chứa các thông tin về vị trí và các tính chất của vật thể và các hiện tượng mà nó trình bày.

Thế giới thực rất rộng lớn và phức tạp để chúng ta có thể thấy bao quát được. Nếu một phần không gian được chọn để trình bày dưới một tỉ lệ nhỏ hơn thực tế thì chúng ta có thể thấy được cấu trúc và dạng của phần không gian đó dễ hơn nhiều và từ đó có thể hiểu thấu đáo được khu vực nghiên cứu và có thể đưa ra được quyết định đúng đắn (như việc tìm đường đi, việc qui hoạch một tuyến đường, việc tìm kiếm một vị trí thích hợp để xây dựng khu công nghiệp...)

Thông thường bản đồ là một mô hình theo tỉ lệ. Có nghĩa là tỉ lệ của khoảng cách trên bản đồ và khoảng cách trên thực tế sẽ bằng nhau ở mọi vị trí trên bản đồ, mặc dù có một vài sai số không thể tránh khỏi nếu một phần của mặt cầu được thể hiện trên mặt phẳng. Chúng ta thường gặp vấn đề này trong bản đồ có tỉ lệ nhỏ trình bày một khu vực rộng lớn.

Thực chất bản đồ là một hệ thống thông tin về không gian. Chúng ta có thể xem bản đồ và tìm thấy các thông tin mà người vẽ bản đồ muốn truyền tải, ví dụ như bản đồ địa hình, bản đồ dân số, bản đồ quy hoạch sử dụng đất, bản đồ địa chất thủy văn, bản đồ địa chất môi trường...

#### 1.1.2 Trái đất quả cầu địa lý

##### 1.1.2.1 Hình dạng và kích thước trái đất

Bề mặt tự nhiên trái đất rất phức tạp

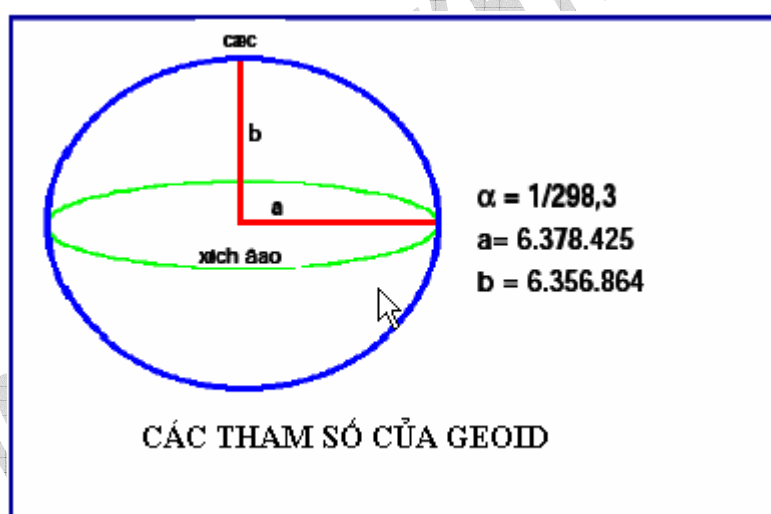
Mặt hình học và không thể biểu thị nó bởi một qui luật xác định, hình dạng trái đất được hình thành và bị chi phối bởi hai lực là lực hấp dẫn và lực ly tâm tạo nên hình dạng ellipsoid của trái đất (hình 1)



Hình 1-1 Hình dạng ellipsoid của Trái Đất

Trong trắc địa người ta dùng mặt geoid, bề mặt này được tạo bởi mặt nước biển trung bình yên tĩnh kéo dài qua các lục địa và hải đảo tạo thành một mặt cong khép kín, có đặc điểm là ở bất kỳ điểm nào nằm trên pháp tuyến cũng trùng với phương dây dọi. Ngoài ra, do tác dụng của trọng lực, sự phân bố không đồng đều của vật chất có tỉ trọng khác nhau trong lớp vỏ của trái đất làm cho bề mặt geoid bị biến đổi phức tạp về mặt hình học.

Như vậy, bề mặt hoàn chỉnh của trái đất không phải là bề mặt đúng toán học, mà chỉ là mặt sần có của chính trái đất. Trong khoa học trắc địa bản đồ, để tiện lợi cho các bài toán đo đạc, người ta lấy mặt ellipsoid tròn xoay có hình dạng và kích thước gần giống mặt geoid làm bề mặt toán học thay cho mặt geoid gọi là ellipsoid trái đất. Ellipsoid có khối lượng bằng khối lượng geoid, tâm của nó trùng với trọng tâm trái đất, mặt phẳng xích đạo trùng với mặt phẳng xích đạo trái đất. Kích thước và hình dạng của ellipsoid trái đất được xác định bởi giá trị các phần tử của nó:



Hình 1-2 Các tham số của GEOID

Nhiều công trình nghiên cứu khoa học nhằm xác định ( $\alpha$ ,  $a$ ,  $b$  của ellipsoid trái đất nhưng kết quả không thống nhất, ở nước ta các trị số của F.N Kraxovski năm 1946 được dùng làm trị số chính thức đo đạc: ( $\alpha = 1/298,3$ ;  $a = 6.378.425$ ;  $b = 6.356.864$ ).

Các số liệu kích thước trái đất được tính như sau:

Bán kính trung bình trái đất: 6.371,166 km.

Độ dài vòng kinh tuyến: 40.008,5 km.

Chu vi xích đạo: 40.075,5 km.

Diện tích bề mặt trái đất 510,2 triệu  $\text{km}^2$

Thể tích trái đất:  $1083 \times 10^3 \text{ km}^3$ .

Tỉ trọng trung bình:  $5,52 \text{ g/cm}^3$ .

Trọng lượng của trái đất:  $5,977 \times 10^{21}$  tấn.

Vì độ dẹt của ellipsoid trái đất nhỏ, nên trong trường hợp đo đạc khu vực nhỏ, người ta có thể coi trái đất như một khối cầu có bán kính gần trùng với trục quay của trái đất,  $R$ , theo F.N Kraxovski là 6371,116 km.

#### 1.1.2.2 Các qui ước về điểm và đường cơ bản để xác định vị trí các đối tượng địa lý trên bề mặt trái đất

##### a. Cực trái đất

Giao điểm giữa bán kính trục nhỏ (trục trái đất) và mặt ellipsoid trái đất gọi là các cực. Trái đất có hai cực là cực Bắc (P) và cực Nam (P').

##### b. Các kinh tuyến

Các mặt phẳng chứa trục trái đất và hai cực là mặt phẳng kinh tuyến.

Giao tuyến giữa mặt phẳng kinh tuyến và mặt ellipsoid trái đất là kinh tuyến.

##### c. Các vĩ tuyến

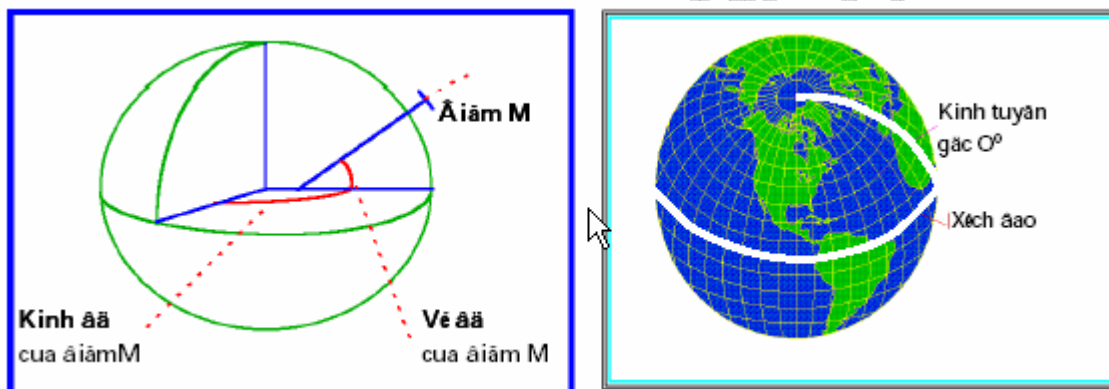
Các mặt phẳng thẳng góc với trục trái đất được gọi là mặt phẳng vĩ tuyến.

Mặt phẳng đi qua tâm trái đất chia trái đất thành hai bán cầu: bán cầu bắc và bán cầu nam, là mặt phẳng xích đạo. Mặt phẳng xích đạo cắt mặt ellipsoid trái đất thành một vòng tròn lớn gọi là xích đạo. Các vòng tròn tạo nên bởi các mặt phẳng song song với mặt phẳng xích đạo gọi là vĩ tuyến.

### 1.1.2.3 Tọa độ địa lý

Tất cả các điểm trên bề mặt ellipsoid trái đất đều được xác định vị trí bằng phương pháp tọa độ. Có nhiều hệ thống tọa độ, trong đó có hệ tọa độ địa lý.

Cơ sở để xác định tọa độ địa lý là kinh tuyến và vĩ tuyến. Tọa độ địa lý một điểm được xác định bằng vĩ độ và kinh độ của điểm đó.



Hình 1-3 Hệ thống kinh độ và vĩ độ

- Vĩ độ địa lý: của một điểm là góc hợp bởi đường dây dọi đi qua điểm đó và mặt phẳng xích đạo. Những vĩ độ được tính từ xích đạo (00) về phía bắc đến 900 gọi là vĩ độ Bắc (N), và về phía nam đến 900 là vĩ độ Nam(S).

- Kinh độ địa lý: của một điểm là góc nhị diện hợp bởi mặt phẳng kinh tuyến gốc và mặt phẳng kinh tuyến đi qua điểm đó. Để tiện xác định vị trí các điểm trên địa cầu, người ta qui định trên địa cầu có 360 đường kinh tuyến các đều nhau. Khoảng cách giữa hai đường kinh tuyến là một cung tròn có góc ở tâm là  $1^{\circ}$ . Hội nghị thiên văn Quốc Tế họp ở Wasington (1884) đã lấy đường kinh tuyến đi qua đài thiên văn Grinwish gần London, thủ đô Anh, làm kinh tuyến gốc (00) thống nhất cho toàn thế giới.

Các kinh độ được tính từ kinh tuyến gốc về phía đông đến 1800 là những kinh độ Đông (E), và về phía tây là những kinh độ tây (W)

Thành phố Hà Nội có tọa độ là 105 052' E và 210 02' N

## 1.1.3 Cơ sở toán học cho bản đồ

### 1.1.3.1 Tỷ lệ

Tỷ lệ bản đồ (map scale) là tỉ số khoảng cách giữa 1 đơn vị đo trên bản đồ so với khoảng cách ngoài thế giới thực.

*Ví dụ* : Tỷ lệ 1 :10.000 được hiểu là 1 cm trên bản đồ tương đương với 100 m trên thực tế.

Việc lựa chọn tỷ lệ bản đồ cần xem xét đến những yếu tố sau:

- Mục tiêu sử dụng
- Độ chính xác yêu cầu
- Yêu cầu của người sử dụng
- Kích thước vùng cần thể hiện lên bản đồ.
- Yếu tố thẩm mỹ

Tỷ lệ bản đồ thường được thể hiện ở ba dạng:

- Dạng số
- Dạng chữ
- Dạng thước tỷ lệ

Việc lựa chọn tỷ lệ thích hợp cũng mang tính tương đối. Đối với bản đồ có tỷ lệ quá lớn, yêu cầu thể hiện chi tiết nhiều hơn, công việc đo đạc thu thập số liệu dữ kiện thông tin phải chi tiết hơn. Đòi hỏi người vẽ bản đồ phải đầu tư công sức nhiều hơn. Thời gian hoàn thành lâu hơn. Giá thành một bản đồ cũng tăng. Trong khi đó, đối với bản đồ có tỷ lệ quá nhỏ thường ít thông tin, khó hiểu.

### 1.1.3.2 Phép chiếu bản đồ

Khi cần vẽ một vùng diện tích có kích thước trong khoảng 30 km x 30 km. Ta xem như độ cong của bề mặt Trái Đất là không đáng kể. Lúc này có thể xem bề mặt Trái Đất là mặt phẳng và thực hiện vẽ trực tiếp

Tuy nhiên khi cần vẽ vùng diện tích lớn hơn vấn đề đặt ra là cần chọn hệ quy chiếu thích hợp.

#### a. Hệ quy chiếu

Hệ quy chiếu được định nghĩa là một cách sắp xếp có hệ thống các kinh tuyến và vĩ tuyến, miêu tả bề mặt cong của địa cầu lên mặt phẳng.

Hệ quy chiếu được phân ra làm các loại sau

- Hệ quy chiếu đồng góc(Conformal projections): góc đo được trên mặt đất bằng với góc trên bản đồ.

- Hệ quy chiếu đồng diện tích (Equivalent projections): Diện tích bề mặt trên mặt đất bằng diện tích trên bản đồ.
- Hệ quy chiếu đồng khoảng cách (Equidistance projections): Khoảng cách từ tâm hệ quy chiếu đến các điểm khác trên bản đồ là thực.
- Các hệ quy chiếu trung gian khác (không thuộc các hệ quy chiếu trên nhưng cho phép thể hiện một khu vực)

b. Phép chiếu bản đồ

Phân làm hai loại

❖ Dựa trên các loại mặt chiếu

- + Phép chiếu hình nón: là phép chiếu mà bề mặt hình học hỗ trợ là hình nón tiếp xúc (chiếu tiếp tuyến) hoặc cắt qua địa cầu (chiếu pháp tuyến).



Hình 1-4 Phép chiếu hình nón

- +Phép chiếu hình phương vị: là phép chiếu mà bề mặt hình học hỗ trợ là mặt phẳng tiếp xúc (chiếu tiếp tuyến) hoặc cắt qua địa cầu (chiếu pháp tuyến)



Hình 1-5 Phép chiếu phương vị

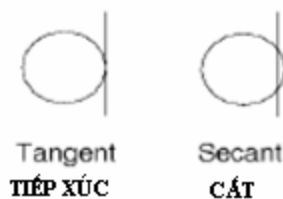
- +Phép chiếu Hình trụ:

là phép chiếu mà bề mặt hình học hỗ trợ là hình trụ tiếp xúc (chiếu tiếp tuyến) hoặc cắt qua địa cầu (chiếu pháp tuyến)



Hình 1-6 Phép chiếu hình trụ

- ❖ Dựa trên vị trí mặt chiếu với trục quả địa cầu
  - + Phép chiếu thẳng (hay phép chiếu đứng): Trục của mặt chiếu (mặt phẳng, nón hay trụ) trùng với trục quay của quả địa cầu.



Hình 1-7 Phép chiếu thẳng

- + Phép chiếu ngang (hay phép chiếu xích đạo):

Đối với phép chiếu phương vị, mặt chiếu hình hỗ trợ tiếp xúc ở một điểm hay một đường bất kỳ trên xích đạo. Ở phép chiếu hình nón và phép chiếu hình trụ, trục của mặt nón và mặt trụ nằm trong mặt phẳng xích đạo, vuông góc với trục quay của quả địa cầu.



Hình 1-8 Phép chiếu ngang

- + Phép chiếu nghiêng: Ở phép chiếu phương vị, mặt phẳng chiếu tiếp xúc với quả địa cầu tại một điểm nào đó giữa xích đạo và cực. Đối với phép chiếu hình nón và hình trụ, trục của mặt nón và mặt trụ có vị trí nghiêng so với mặt phẳng xích đạo.



Hình 1-9 Phép chiếu nghiêng

Ngoài ra trong hệ thống phép chiếu còn có phép chiếu Mercator và phép chiếu Gauss

### 1.1.3.3 Hệ thống phân mảnh và danh pháp bản đồ

Việc phân mảnh bản đồ do điều kiện in ấn không in được bản đồ có kích thước lớn, phân mảnh bản đồ theo hệ thống giúp việc dựng lại bản đồ khi ra ngoài thực địa

Có hai hệ thống phân mảnh bản đồ chính:

- Chia mảnh vuông góc

Khung của bản đồ hoặc trùng với đường của lưới tọa độ vuông góc hoặc theo đường phân chia khác. Bản đồ được chia thành các mảnh hình chữ nhật, đánh số thứ tự theo hàng ngang từ trái sang phải và từ trên xuống dưới theo hàng dọc có sơ đồ kèm theo

- Hệ chia mảnh hình thang

Theo chiều kinh tuyến chia bề mặt trái đất thành 60 dải đánh số từ 1-60, mỗi dải cách nhau  $6^\circ$ . Thứ tự các dải được đánh số lần lượt bắt đầu từ kinh tuyến 180-174 T là dải số 1, 174-168T là dải số 2... dải 60 từ 174 –  $180^\circ$ .

Theo chiều vĩ tuyến từ xích đạo trở về hai cực, cứ  $4^\circ$  chia thành 1 đai có đánh số thứ tự bằng chữ in hoa A,B,C,D...

Như vậy, bề mặt trái đất được chia thành các mảnh hình thang có độ chênh lệch kinh độ  $6^\circ$  và độ chênh lệch vĩ độ là  $4^\circ$ . Mỗi hình thang biểu thị trên một bản đồ 1:1.000.000. Danh pháp của nó được ghi rõ theo đai và dải.

Ví dụ: Bản đồ ghi F-48 là tờ bản đồ có tỷ lệ 1:1.000.000, F biểu thị của đai từ  $20-24^\circ$  vĩ độ, 48 là tên của dải thứ 48 từ kinh tuyến  $102^\circ$  Đ đến  $108^\circ$  Đ.

Nếu tờ bản đồ thể hiện phần bắc bán cầu thì ghi thêm chữ N (north) và ở nam bán cầu thì ghi thêm chữ S (south), ví dụ NF-48.

Lãnh thổ Việt Nam nằm ở trong các đai C,D,E,F và các dải 48,49.

Bản đồ tỷ lệ từ 1:500.000 đến 1:100.000 được chia mảnh và ghi số hiệu theo bản đồ 1:1.000.000.



Mảnh bản đồ tỷ lệ 1:50.000 đến 1:10.000 được chia mảnh và ghi số hiệu theo bản đồ 1:100.000.

Mảnh bản đồ tỷ lệ 1:5.000 đến 1:2.000 thể hiện vùng đất lớn hơn 20km<sup>2</sup> được chia mảnh và ghi số hiệu theo bản đồ 1:100.000.

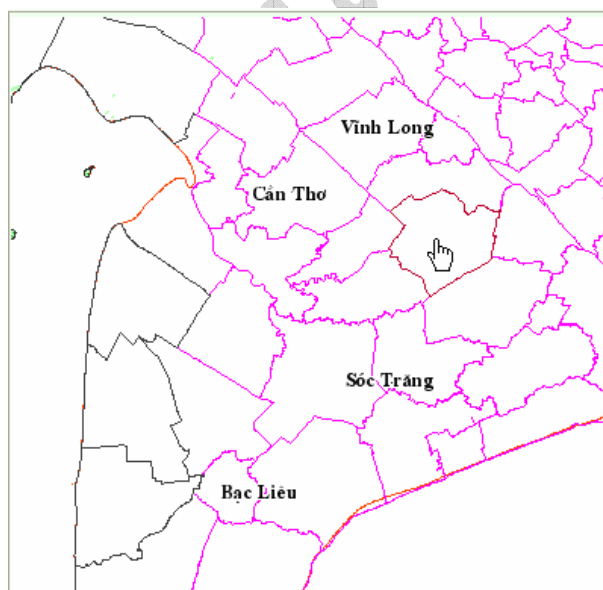
Đối với vùng đất nhỏ hơn 20km<sup>2</sup> ta có thể chia mảnh và ghi số hiệu theo tọa độ ô vuông với kích thước là 40x40 km cho bản đồ tỉ lệ 1:5.000 và 50x50km cho bản đồ tỷ lệ 1:2000 đến 1:500

### 1.1.4 Các phương pháp thể hiện bản đồ

#### 1.1.4.1 Phân loại bản đồ

Bản đồ có 2 dạng chính

Dạng “đường nét” (line map)



Hình 1-10 Bản đồ đường nét

Dạng ảnh (photo and iamge map)



Hình 1-11 Bản đồ dạng ảnh

Bản đồ “đường nét” dùng các kí hiệu, nét vẽ để hiện thông tin một cách tóm lược về khu vực thể hiện. Chủ yếu được vẽ làm bằng thủ công và cộng với sự trợ giúp của máy tính.

Bản đồ ảnh thường là những hình chụp ngoài thực địa từ trên cao (nhà cao tầng, máy bay, vệ tinh...) Người ta thường vẽ thêm “đường nét” để nhấn mạnh các thực thể vào trong bản đồ ảnh. Bản đồ dạng này có ưu điểm là vẽ nhanh, miêu tả được những địa

hình nếu dùng nét vẽ thường khó thể hiện (ví dụ: ao hồ, sa mạc...) Tuy nhiên bản đồ này thường khó khăn trong việc giải đoán các thực thể trên bản đồ.

#### 1.1.4.2 Các thành phần bản đồ

Thành phần của bản đồ rõ ràng liên quan đến mục tiêu sử dụng của nó. Các thành phần trong bản đồ là:

- Thành phần chính (chủ đề chính)
- Thành phần thứ hai (bản đồ nền, thông tin cơ bản của bản đồ)
- Thành phần phụ trợ ( thông tin lẽ như chú thích, tỉ lệ, tiêu đề...)

##### *Thành phần chính*

Là phần chủ đề của bản đồ, ví dụ như địa lý, địa chất, dân số. Đối với bản đồ địa hình, phần chính là tất cả các thông tin được vẽ, bao gồm cả tên của các vùng

##### *Thành phần thứ hai*

Đối với bản đồ chủ đề, thành phần này là phần địa hình, bao gồm lưới tọa độ.

##### *Thành phần phụ trợ*

Bao gồm các thông tin lẽ như tiêu đề, chú thích, thanh tỉ lệ...

#### 1.1.4.3 Độ chính xác của bản đồ

Ba vấn đề của độ chính xác cần quan tâm là:

- Chính xác về vị trí
- Chính xác về chủ đề
- Chính xác về cách thể hiện

##### *Chính xác về vị trí*

Độ chính xác của vị trí được vẽ trên bản đồ liên quan đến vị trí thực tế của nó trên thực tế.

Độ chính xác này ảnh hưởng bởi:

- Phép chiếu
- Độ chính xác của việc thu thập dữ liệu và việc vẽ bản đồ
- Tỉ lệ của bản đồ
- Công cụ và độ ổn định của vật liệu được sử dụng trong việc vẽ bản đồ

##### *Chính xác về chủ đề*

Độ chính xác về chủ đề liên quan đến thông tin chủ đề được thể hiện. Độ chính xác này ảnh hưởng bởi:

- Việc thu thập thông tin thuộc tính: chất lượng của dữ liệu thống kê và phương pháp thống kê
- Việc chuyển đổi dữ liệu: Một phần của vùng đôi khi được dùng để thể hiện cho toàn vùng, ví dụ như trường hợp bản đồ mật độ dân số (một huyện có mật độ 50 người/km<sup>2</sup> không có nghĩa mọi km<sup>2</sup> của huyện đều có 50 người).

#### *Chính xác về cách thể hiện*

Sự thể hiện của các biểu tượng trên bản đồ rất quan trọng, nếu dùng sai biểu tượng thì có thể đánh lạc hướng của người sử dụng, hay làm mờ ranh giới của các vùng trên bản đồ.

#### 1.1.4.4 Các chú giải trên bản đồ - Ngôn ngữ của bản đồ

Ngôn ngữ bản đồ cũng là một loại ngôn ngữ, có các chức năng chính sau:

- Dạng có cấu trúc (hình vẽ) gợi nhớ đối tượng
- Kí hiệu chứa một nội dung về số lượng, chất lượng, cấu trúc của đối tượng cần thể hiện trên bản đồ.
- Kí hiệu trên bản đồ phản ánh vai trò của đối tượng trong không gian và vị trí tương quan của nó với các yếu tố khác.

Hệ thống kí hiệu quy ước bản đồ:

Trên bản đồ ta sử dụng các dạng đồ họa, màu sắc, các loại chữ số và con số. Các kí hiệu trên bản đồ thường được thể hiện dưới dạng.

- Kí hiệu điểm (point)
- Kí hiệu tuyến (polyline)
- Kí hiệu diện tích (region)
- Kí hiệu tượng hình
- Kí hiệu hình học
- Kí hiệu chữ

1.1.4.5 Phương pháp thể hiện thông tin trên bản đồ

Phương pháp	Đối tượng dùng	Cách thể hiện	Thông tin thể hiện
Cartogram	Dạng vùng	Đặt biểu đồ thể hiện mối liên quan của các đặc trưng của hiện tượng vào trong biên của hiện tượng đó	Số lượng, cấu trúc
Nền chất lượng	Dạng vùng	Dùng màu sắc, mẫu tô hay đánh số	Thể hiện các hiện tượng phân bố liên tục trên bề mặt đất (lớp phủ thực vật, loại đất...) hay các hiện tượng phân bố theo khối (dân cư, phân vùng lãnh thổ)
Đường đẳng trị	Dạng điểm	Nối các điểm có cùng chỉ số về số lượng của hiện tượng trên bản đồ (đồng cao độ, đẳng mưa, đẳng nhiệt...)	Các đối tượng có cùng số lượng của hiện tượng
Kí hiệu đường chuyển động	Dạng tuyến hoặc dạng vectơ	Vẽ các mũi tên để thể hiện sự di chuyển	Thể hiện sự di chuyển của các đối tượng hay hiện tượng trên bản đồ
Chấm điểm	Dạng vùng	Chấm điểm cho vùng hiện tượng	Thể hiện sự phân tán của hiện tượng trên một vùng (phân bố dân cư).
Biểu đồ định vị	Dạng điểm	Tạo biểu đồ tương quan (dạng tròn, dạng cột. ) giữa các đặc trưng đo đạc.	Các hiện tượng phân bố liên tục
Kí hiệu	Dạng điểm	Dùng các kí hiệu (hình vẽ, chữ số...) đặt vào vị trí đối tượng	Đặc điểm phân bố, số lượng, chất lượng, cấu trúc...

Bảng 1-1 Các phương pháp thể hiện bản đồ

#### 1.1.4.6 Sự khái quát hóa và phóng đại

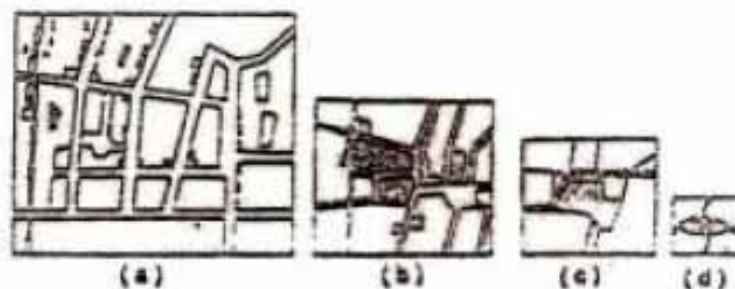
Vì bản đồ là sự thu nhỏ của thế giới thực nên ta không thể trình bày một cách chính xác tuyệt đối hình dạng và kích thước thực thể. Do đó thường người ta dùng hai kỹ thuật sau đây để thể hiện thực thể trên bản đồ:

- *Khái quát hóa* là sự chọn lựa và đơn giản hóa sự thể hiện của thực thể trên bản đồ theo một tỉ lệ và mục đích thích hợp nhằm giúp cho bản đồ dễ đọc.
- *Sự phóng đại* là kỹ thuật nhằm phóng to vật thể cần thể hiện hơn là tỉ lệ thực của nó nhằm giúp cho bản đồ dễ đọc hay nhằm nhấn mạnh vật thể đó.

Sự khái quát hóa yêu cầu chú ý đến các yếu tố sau:

- + Sự chọn lựa: Mục tiêu của bản đồ là yếu tố chính để chọn lựa thực thể nào nên vẽ trên bản đồ. Sự chọn lựa thường liên quan đến tỉ lệ bản đồ.
- + Sự đơn giản hóa: Các thực thể phải được trình bày trên bản đồ nhưng quá nhỏ hay quá phức tạp mà không thể trình bày được chi tiết nếu không bỏ bớt hay đơn giản hoá. Tỉ lệ là yếu tố tham gia chính.
- + Lược bỏ: Để duy trì tính dễ đọc và sạch sẽ của bản đồ, một vài thực thể sẽ không được thể hiện, ngay cả nó rõ ràng. Tỉ lệ vẫn là yếu tố ảnh hưởng chính nhưng yếu tố tự nhiên và địa hình cũng quan trọng.

Mối liên hệ giữa sự khái quát hóa và sự phóng đại rất gần, thực ra chính sự phóng đại hóa là sự khái quát hóa. Ví dụ trong trường hợp bản đồ đường sá tỉ lệ 1/50000, nếu ta vẽ theo đúng tỉ lệ con đường rộng 10m thì nét vẽ thể hiện con đường này chỉ rộng 0.2mm cho tất cả các đoạn rẽ hay đoạn xoắn, nhưng trong bản đồ chúng ta phải dùng nét vẽ 1mm để thể hiện. Tuy nhiên với nét vẽ này ta vẫn không thể hiện chính xác được các đoạn rẽ và đoạn xoắn



Hình 1-12 Khái quát hóa theo tỉ lệ  
a. Tỉ lệ 1:25.000    b. 1:50.000    c. 1:100.000    d. 1:1.000.000

## 1.2 Dữ liệu GIS

### 1.2.1 Các dạng dữ liệu của GIS

Dữ liệu GIS được chia làm 3 dạng: dữ liệu không gian, dữ liệu phi không gian hay dữ liệu thuộc tính và dữ liệu thời gian.

Dữ liệu không gian					Dữ liệu thuộc tính				Thời gian		
Tọa độ x,y			Vị trí tương quan (topology)		Biến	Lớp	Giá trị	Tên	T1	T2	T3
Đ i ể m	Đ ư ờ n g	V ù n g	L ư ờ i	M ạ n g	-	-	-	-			

Hình 1-13 Các dạng dữ liệu GIS (theo J.Dangermon, 1983)

Dữ liệu địa lý mô tả những thực thể có vị trí. Dữ liệu địa lý gồm thông tin vị trí và những thông tin cần quan tâm, được xem như là các thuộc tính của thực thể.

**Dữ liệu không gian** (trả lời cho câu hỏi về vị trí - ở đâu?) được thể hiện trên bản đồ và HTTĐL dưới dạng điểm (point), đường (line), hoặc vùng (polygon). Dữ liệu không gian là dữ liệu về đối tượng mà vị trí của nó được xác định trên bề mặt Trái Đất. Dữ liệu không gian sử dụng trong HTTĐL luôn được xây dựng trên một hệ thống tọa độ.

**Dữ liệu phi không gian** (trả lời cho câu hỏi cái gì?, như thế nào) thể hiện tính chất của đối tượng như chiều dài, rộng của con đường, độ cao của cây rừng, dân cư của thành phố... với dữ liệu phi không gian thì vị trí không quan trọng.

Dữ liệu thời gian (trả lời cho câu hỏi: tồn tại khi nào?) Trên thực tế thì các thông tin không gian (có vị trí tọa độ) và thông tin thuộc tính có thể biến đổi không phụ thuộc vào nhau tương đối theo thời gian.

Ví dụ: Sự di chuyển của các cồn cát làm thay đổi vị trí của chúng nhưng vẫn giữ nguyên thuộc tính “cồn cát”. Hay quá trình xói mòn đất làm thay đổi thuộc tính “độ cao” của quả đồi nhưng lại giữ nguyên vị trí tọa độ của nó.

### 1.2.2 Các mô hình dữ liệu được dùng

Trong hệ thống thông tin địa lý dữ liệu có mối quan hệ với nhau, đặt ra yêu cầu phải được lưu trữ như là một CSDL.

Dưới đây là các mô hình dữ liệu được dùng để lưu trữ sắp xếp dữ liệu bên trong hệ thống thông tin địa lý.

- Mô hình tổng quát
  - Mô hình mì ống – Spaghetti model
- Mô hình dữ liệu cơ bản – Basic data model
  - Vector
  - Raster
- Mô hình không gian ( Spatial model)
- Mô hình bề mặt ( Surface model)
- Mô hình toán học ( Mathematical models)
- Mô hình khái niệm ( Conceptual models)
  - Mô hình Thực thể - mối quan hệ (Entity- Relationship ER)
  - Mô hình Thực thể - mối quan hệ nâng cao (Enhanced Entity – Relationship EER)
  - Mô hình thực thi (An Implmentation model)
  - Mô hình quan hệ (Relational model)
- Mô hình ngữ nghĩa ( Semantic models)
- Mô hình độc quyền

Đây là mô hình của các tổ chức, công ty kinh doanh trên lĩnh vực GIS, có thể kể ra ở đây như: Arc/Info, ERDAS, Geovision, DBMS based...

### 1.2.3 Mô hình dữ liệu đồ họa

Trong thế giới GIS, phần dữ liệu đồ họa dùng để tạo lập nên các bản đồ đóng vai trò quan trọng. Dữ liệu đồ họa này mô tả thế giới thực và được chia làm 2 loại : dữ liệu raster và dữ liệu vector.

Xét ví dụ sau đây dùng dữ liệu dạng raster và vector để mô hình hóa thế giới thực: Một khu vực gồm có dòng sông chảy qua, bên cạnh bờ sông là vài cây cao, chếch về bên trái là một ngôi nhà nhỏ



Hình 1-14 Ví dụ thế giới thực

#### ❖ Biểu diễn bằng Raster

	1	2	3	4	5	6	7	8	9	10
1							R			
2							R			
3						R				
4					R	P				
5					R	P				
6					R	P			H	
7					R	P				
8			R	R						
9			R							
10			R							

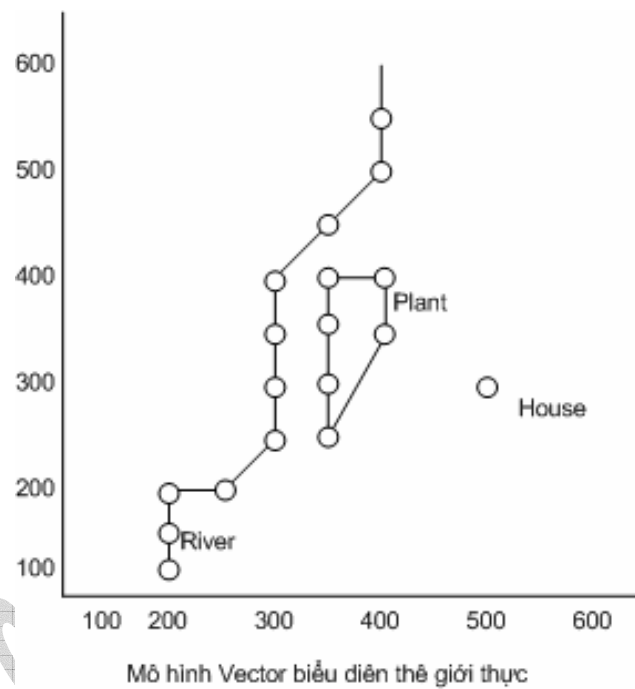
Mô hình Raster biểu diễn cảnh thực trên

Hình 1-15 Biểu diễn thế giới thực bằng Raster

Xem ví dụ trên cho thấy, dòng sông được chia thành nhiều ô nhỏ nằm trên một lưới ô (kí hiệu R), còn cây dọc bên bờ cũng được biểu diễn bằng ô (kí hiệu P)...

#### ❖ Biểu diễn bằng Vector

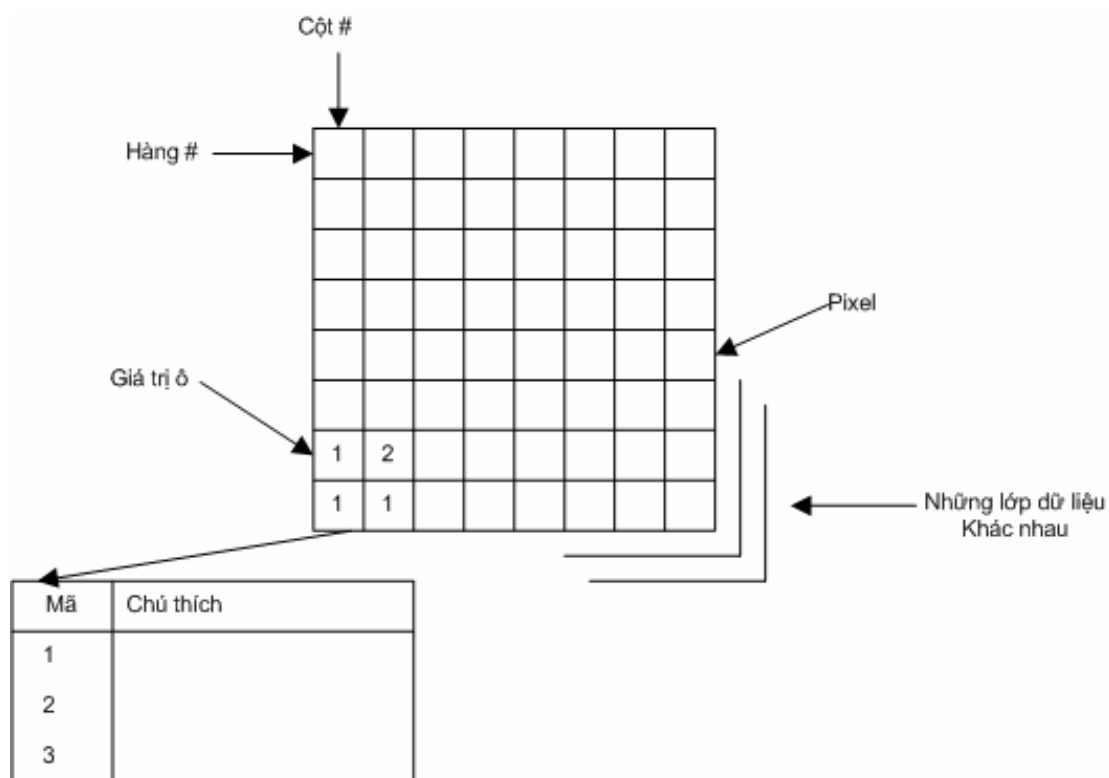




Hình 1-16 Mô hình Vector biểu diễn thế giới thực

Các đối tượng được quy về điểm (House), đường nối các điểm (River) và vùng (một tập các đường nối điểm đầu trùng điểm cuối).

#### 1.2.3.1 Mô hình dữ liệu Raster



Hình 1-17 Mô hình dữ liệu Raster

Đây là hình thức đơn giản nhất để thể hiện dữ liệu không gian, mô hình raster bao gồm một hệ thống ô vuông hoặc ô chữ nhật được gọi là pixel (hay điểm ảnh). Vị trí của mỗi pixel được xác định bởi số hàng và số cột. Giá trị được gán vào pixel tượng trưng cho một thuộc tính mà nó thể hiện.

Hình ảnh được thể hiện càng rõ khi kích thước của pixel hay ô lưới càng nhỏ. Thông số này được gọi là độ tương phản. Xét ví dụ hai ảnh raster có cùng kích thước thì ảnh nào có độ tương phản càng cao (pixel càng nhỏ) thì kích thước càng tăng.

Giả sử 1 pixel thể hiện một diện tích 250m x 250m trên thực tế thì để thể hiện một vùng 1km x 1km cần 4pixel. Trong khi đó nếu 1 pixel thể hiện diện tích 100 m x 100 m thì với diện tích 1km x 1km ta lại cần đến 10 pixel.

Một ảnh thông thường gồm hàng triệu pixel dẫn đến kích thước ảnh là khá lớn, tuy nhiên nhiều pixel gần nhau lại mang cùng giá trị. Từ nhận xét này, người ta đã sử dụng nhiều phương pháp nén khác nhau để nén ảnh raster như : phương pháp Run-Length Encoding, phương pháp Value-Point Encoding và phương pháp Quadrees.

❖ Xây dựng CSDL Raster

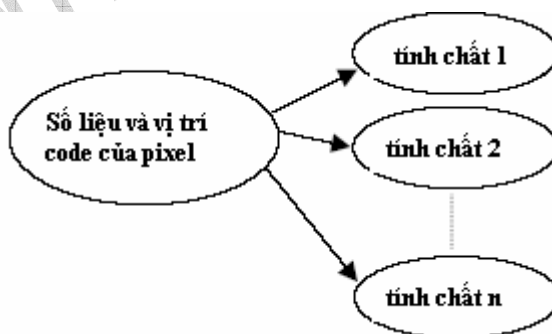
- Mỗi pixel là một đối tượng, có vị trí theo hàng, cột tương ứng trên ảnh, giá trị của pixel cho biết pixel đó thuộc loại đối tượng nào, tính chất của đối tượng đó được lưu trữ ở một cơ sở dữ liệu thuộc tính tương ứng.

- CSDLKG Raster có thể chứa hàng ngàn lớp DLKG (layer).

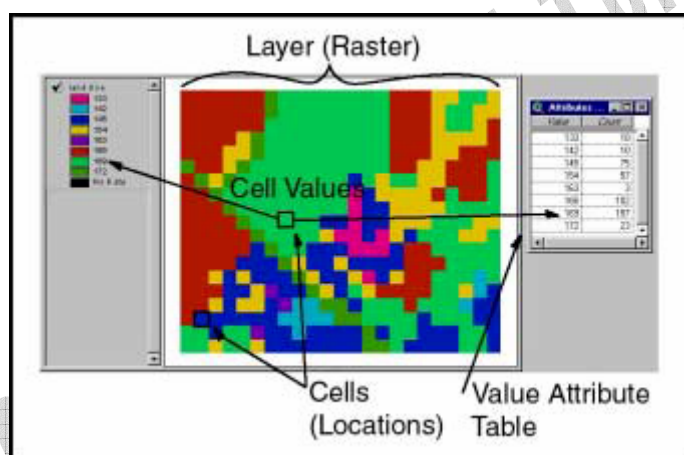
Kiểu giá trị của pixel trong mỗi layer tùy theo việc mã hóa của người sử dụng, có thể là số nguyên, số thực hay ký tự alphabet. Thường giá trị số nguyên thường được dùng làm mã số để liên hệ với bảng DL khác hay làm chú giải để thể hiện bản đồ.

- Để thể hiện một bề mặt liên tục, người ta sử dụng mô hình raster, các bề mặt liên tục này thường thể hiện bề mặt địa hình, mưa, áp suất không khí, nhiệt độ, mật độ dân số...

- Như vậy đối với CSDLKG raster các thông tin được tổ chức như hình dưới đây:



Hình 1-18 Tổ chức CSDL KGian Raster



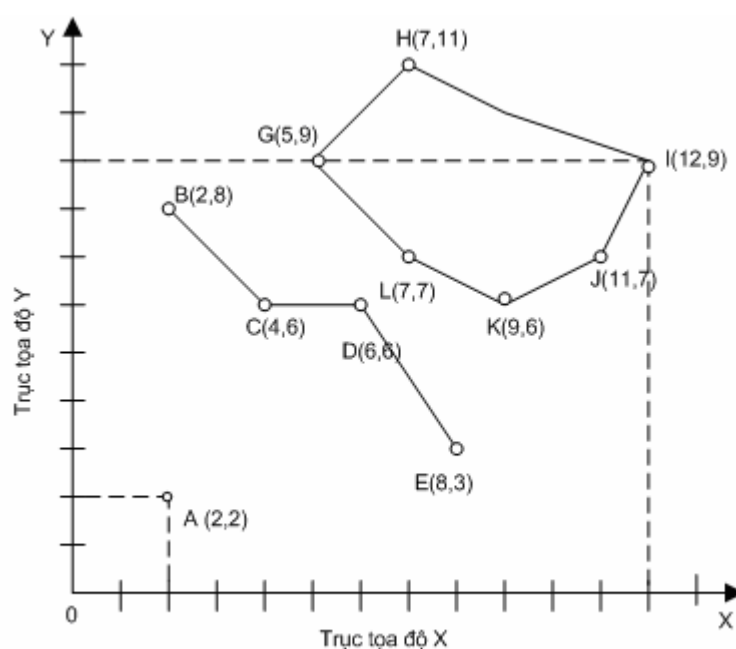
Hình 1-19 Tổ chức CSDL KGian Raster

### 1.2.3.2 Mô hình dữ liệu Vector

Mô hình dữ liệu vector thể hiện vị trí chính xác của vật thể hay hiện tượng trong không gian. Trong mô hình dữ liệu vector, người ta giả sử rằng hệ thống tọa độ là chính xác.

Thực tế, mức độ chính xác bị giới hạn bởi số chữ số dùng để thể hiện một giá trị trong máy tính, tuy nhiên nó chính xác hơn rất nhiều so với mô hình dữ liệu raster.

Vật thể trên trái đất được thể hiện trên bản đồ dựa trên hệ tọa độ hai chiều x,y (Cartesian coordinate system), trên bản đồ vật thể có thể được thể hiện như là các điểm (point), đường (line) hay miền (area). Mô hình dữ liệu vector cũng tương tự như vậy, một vật thể dạng điểm (point feature) được chứa dưới dạng cặp tọa độ x,y; một vật thể dạng đường (line feature) được chứa dưới dạng một chuỗi các cặp tọa độ x,y; một vật thể dạng vùng (area feature) được chứa dưới dạng một chuỗi cặp tọa độ x,y với cặp đầu tọa độ bằng với cặp tọa độ cuối, hay còn gọi là đa giác (polygon). Trong hình [X], các vật thể được số hóa (digitize) bằng các cặp tọa độ x,y. Vị trí của điểm A được thể hiện bởi tọa độ (2,2) và đường BCDE được thể hiện bởi chuỗi tọa độ (2,8),(4,6),(6,6),(8,3). Đa giác được thể hiện bởi một chuỗi tọa độ khác trong đó tọa độ đầu và cuối bằng nhau. Trong thí dụ này đơn vị của các tọa độ là tùy ý. Tuy nhiên trong GIS, vị trí thường được lưu trữ theo một hệ quy chiếu chuẩn như là hệ thống UTM, hệ thống quốc gia hay hệ kinh tuyến, vĩ tuyến.

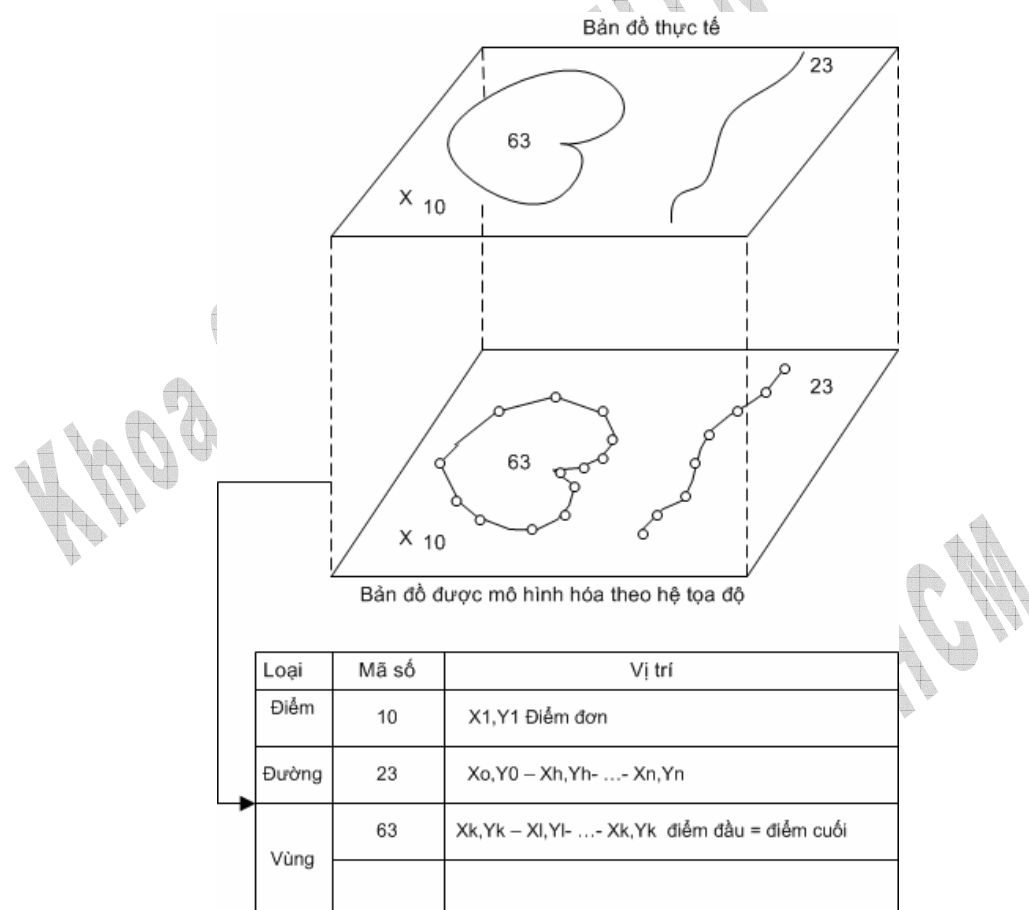


Hình 1-20 Thể hiện vật thể dạng điểm đường vùng theo tọa độ x, y

Trong mô hình dữ liệu vector, tùy theo cách lưu trữ dữ liệu, người ta chia ra thành các mô hình: Spaghetti Data Model, Topological Model, Triangulated Irregular Network (TIN.)

a. Mô hình Spaghetti Data Model

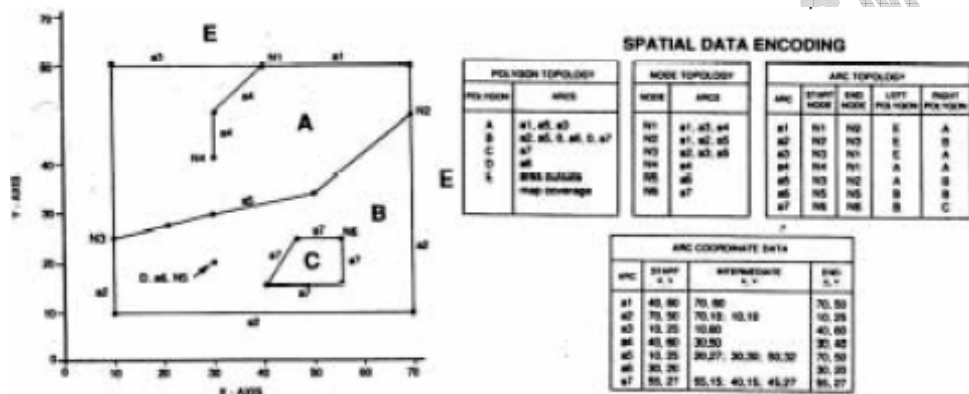
Trong mô hình này tọa độ của các vật thể trên bản đồ được chuyển đổi và ghi nhận vào tập tin dữ liệu theo từng dòng danh sách các cặp tọa độ. Như vậy các cặp tọa độ của cạnh chung của hai đa giác kề nhau phải được lập lại hai lần, mỗi lần cho một đa giác. Cấu trúc của dạng mô hình này rất dễ hiểu, tuy nhiên mối liên hệ của các vật thể trong mô hình không được ghi nhận.



Hình 1-21 Mô hình dữ liệu mì ống ( Spaghetti data model)

Mô hình này không hiệu quả trong phân tích không gian nhưng lại có ích trong việc tái sản xuất bản đồ số mà không cần lưu trữ quan hệ không gian.

b. Topological Model



Hình 1-22 Mô hình dữ liệu Tôpô (Topology)

Mô hình topology được sử dụng rộng rãi trong việc mã hóa các mối quan hệ không gian. Topology là phương pháp toán học được dùng để định nghĩa các quan hệ không gian. Mô hình này còn được gọi là mô hình Arc-Node. Với Arc là một cung gồm chuỗi các điểm bắt đầu và kết thúc tại node. Node là nút, điểm giao nhau của 2 hay nhiều cung. Polygon là chuỗi khép kín của các arc thể hiện ranh giới của vùng.

Topology được ghi trong 3 bảng dữ liệu cho 3 loại yếu tố không gian : polygon, node và arc. Dữ liệu về tọa độ được ghi trong bản thứ 4.

Dữ liệu thuộc tính thường được lưu trữ trong các bảng quan hệ, trong đó 1 trường chứa ID của đối tượng không gian.

▪ Ưu điểm

Phân tích không gian được thực hiện không sử dụng dữ liệu tọa độ, giảm thời gian phân tích.

▪ Nhược điểm

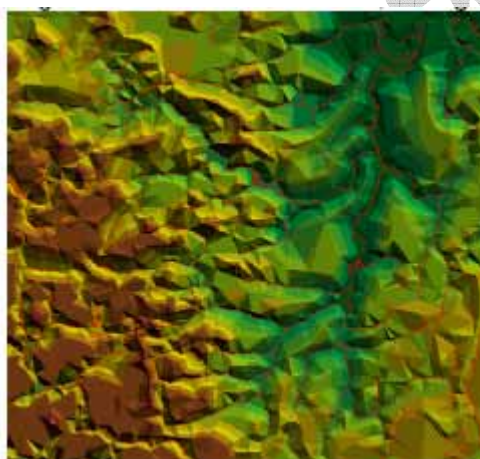
Cập nhật mô hình topology mất nhiều thời gian

c. TIN (Triangular Irregular Network)

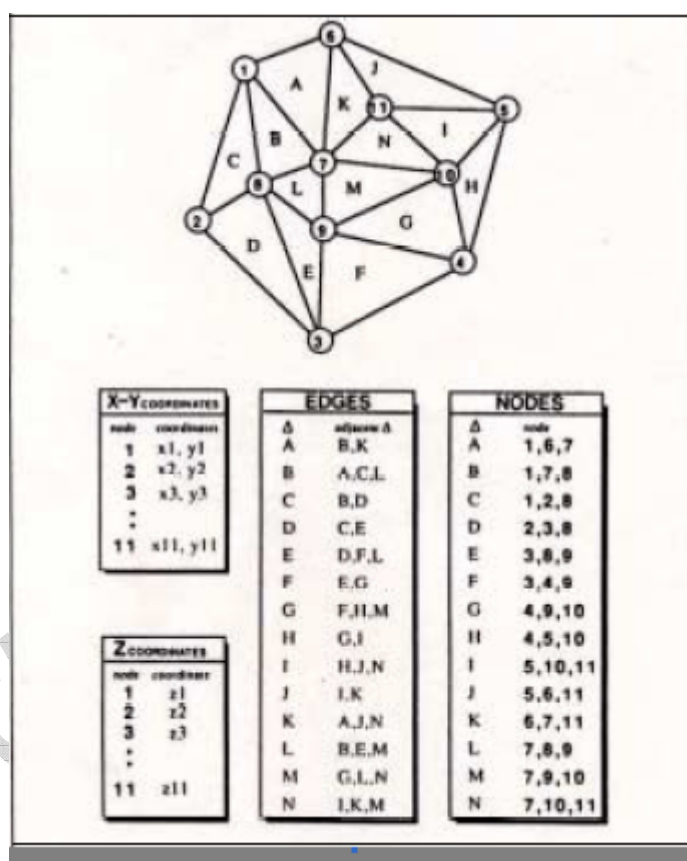
Mô hình dữ liệu kiểu TIN thường được dùng để thể hiện dữ liệu về địa hình. Trong mô hình TIN bề mặt địa hình được thể hiện như là tập hợp các mặt tam giác liên kết với nhau. Trong đó mỗi đỉnh của tam giác được thể hiện bằng tọa độ [x,y, z] với z là giá trị độ cao của bề mặt.

Mỗi mặt của tam giác được gán cho một chữ cái và ba đỉnh của nó được gán bằng chữ số.

Bảng Nodes (nút) thể hiện danh sách đỉnh của từng tam giác, bảng Edges (cạnh) thể hiện danh sách các tam giác nằm xung quanh của từng tam giác, bảng tọa độ X-Y thể hiện tọa độ của các đỉnh, bảng z coordinate thể hiện giá trị z của các đỉnh đó. TIN rất thích hợp trong việc tính toán các thông số của địa hình như độ dốc, hướng dốc.



Hình 1-23 Ảnh chụp bề mặt địa hình



Hình 1-24 Mô hình dữ liệu vectơ kiểu TIN

1.2.3.3 So sánh giữa mô hình dữ liệu Raster và Vector

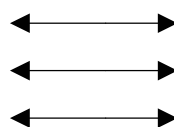
Mô hình Raster	Mô hình Vector
<p><u>Ưu điểm</u></p> <ol style="list-style-type: none"> <li>Đơn giản</li> <li>Thao tác chồng lấp (overlay) dễ dàng</li> <li>Thích hợp cho việc thể hiện dữ liệu phức tạp (đa dạng)</li> <li>Thích hợp cho việc nâng cấp, xử lý ảnh</li> </ol>	<p><u>Ưu điểm</u></p> <ol style="list-style-type: none"> <li>Khả năng dữ liệu được nén tốt dẫn đến kích thước nhỏ hơn so với dữ liệu của mô hình raster.</li> <li>Thể hiện liên hệ hình học do đó thích hợp cho các phân tích về hình học hay phân tích về mạng lưới</li> <li>Thích hợp cho việc số hóa các bản đồ được vẽ bằng tay</li> </ol>
<p><u>Nhược điểm</u></p> <ol style="list-style-type: none"> <li>Khả năng nén kém</li> <li>Không thể hiện rõ liên hệ hình học</li> <li>Thể hiện bản đồ không rõ nét nếu độ tương phản thấp, nhưng nếu dùng độ tương phản cao sẽ làm tăng kích thước file ảnh.</li> </ol>	<p><u>Nhược điểm</u></p> <ol style="list-style-type: none"> <li>Phức tạp</li> <li>Thao tác chồng lấp phức tạp</li> <li>Không thích hợp cho việc thể hiện dữ liệu phức tạp (đa dạng)</li> <li>Không thích hợp cho việc nâng cấp, xử lý ảnh</li> </ol>

Bảng 1-2 So sánh mô hình dữ liệu Raster và Vector

1.2.4 Kết nối dữ liệu không gian và dữ liệu thuộc tính

HTTTĐL sử dụng mô hình raster hoặc vector để mô tả vị trí, còn dữ liệu phi không gian hầu hết được lưu thành file riêng biệt có cấu trúc hoặc sử dụng cơ sở dữ liệu quan hệ để lưu trữ dữ liệu phi không gian.

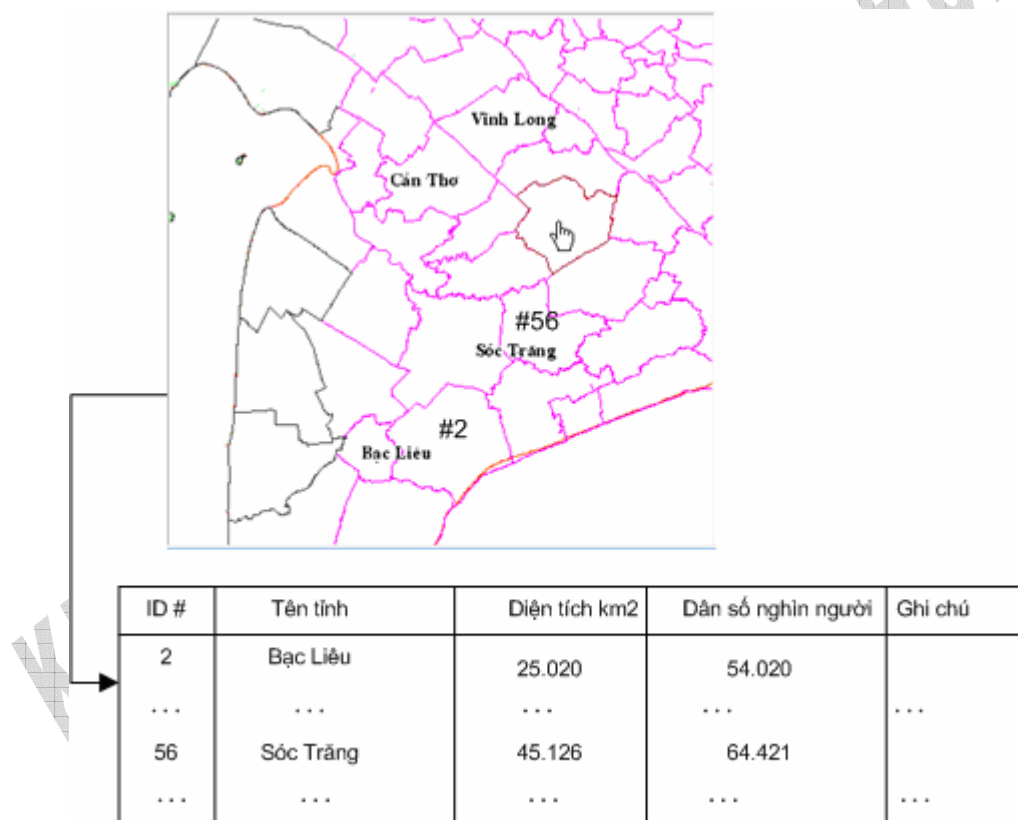
Mô hình liên kết



Hình 1-25 Sơ đồ kết nối dữ liệu



Mỗi liên kết biểu tượng và ý nghĩa được thực hiện bằng cách gán cho bất kì yếu tố địa lý ít nhất một nghĩa xác định, tên hay chỉ số gọi tắt là ID (xác định duy nhất). Dữ liệu phi không gian của yếu tố thường lưu trữ trong một hay nhiều file riêng biệt theo số ID này.



Hình 1-26 Kết nối dữ liệu không gian và thuộc tính

Bên cạnh đó cũng có HTTTĐL chọn cách lưu trữ cả dữ liệu địa lý và dữ liệu thuộc tính vào trong cùng một file có cấu trúc.

### 1.3 Giới thiệu về WebGIS

WebGIS là hệ thống thông tin địa lý phân tán trên một mạng các máy tính để tích hợp, trao đổi các thông tin địa lý trên World Wide Web (Edward, 2000, URL). Trong cách thực hiện nhiệm vụ phân tích GIS, dịch vụ này gần giống như là kiến trúc Client-Server của Web. Xử lý thông tin địa lý được chia ra thành các nhiệm vụ ở phía server và phía client. Điều này cho phép người dùng có thể truy xuất, thao tác và nhận kết quả từ việc khai thác dữ liệu GIS từ trình duyệt web của họ mà không phải trả tiền cho phần mềm GIS.

Một client tiêu biểu là trình duyệt web và server-side bao gồm một Web server có cung cấp một chương trình phần mềm WebGIS. Client thường yêu cầu một ảnh bản đồ

hay vài xử lý thông tin địa lý qua Web đến server ở xa. Server chuyển đổi yêu cầu thành mã nội bộ và gọi những chức năng về GIS bằng cách chuyển tiếp yêu cầu tới phần mềm WebGIS. Phần mềm này trả về kết quả, sau đó kết quả này được định dạng lại cho việc trình bày bởi trình duyệt hay những hàm từ các plug-in hoặc Java applet. Server sau đó trả về kết quả cho client để hiển thị, hoặc gửi dữ liệu và các công cụ phân tích đến client để dùng ở phía client. (Peng 1997 p.5).

Phần lớn sự chú ý gần đây là tập trung vào việc phát triển các chức năng GIS trên Internet. WebGIS có tiềm năng lớn trong việc làm cho thông tin địa lý trở nên hữu dụng và sẵn sàng tới số lượng lớn người dùng trên toàn thế giới. Thách thức lớn của WebGIS là việc tạo ra một hệ thống phần mềm không phụ thuộc vào platform và chạy trên chuẩn giao thức mạng TCP/IP, có nghĩa là khả năng WebGIS được chạy trên bất kì trình duyệt web của bất kì máy tính nào nối mạng Internet. Đối với vấn đề này, các phần mềm GIS phải được thiết kế lại để trở thành ứng dụng WebGIS theo các kỹ thuật mạng Internet.

## **1.4 Mô hình xử lý và kiến trúc triển khai WebGIS**

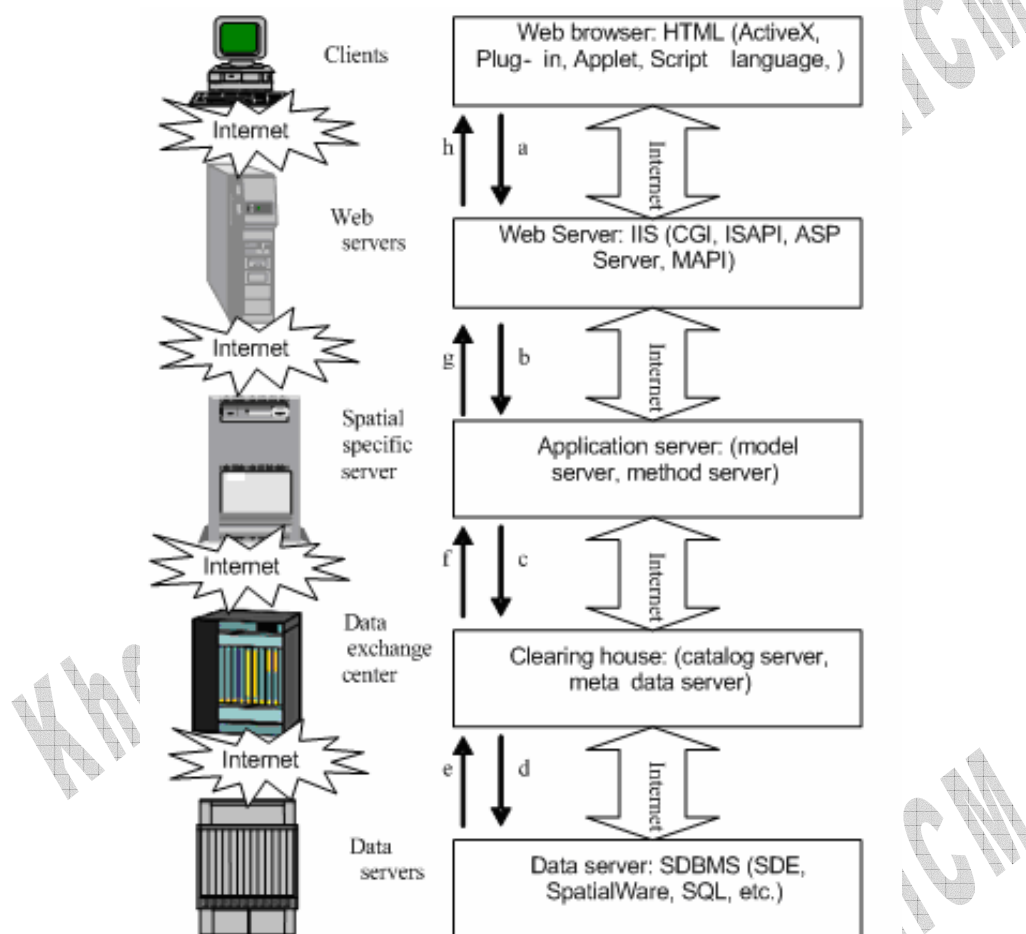
### **1.4.1 Kiến trúc WebGIS và các bước xử lý**

#### **1.4.1.1 Kiến trúc WebGIS**

Kiến trúc xuất bản web của hệ thống tin dữ liệu không gian cũng gần giống như kiến trúc dành cho một hệ thông tin web cơ bản khác, ngoại trừ có ứng dụng GIS sử dụng các kỹ thuật khác. Có nhiều dạng của việc xuất bản web cho thông tin không gian, phần phức tạp nhất sẽ được trình bày ở đây để có cái nhìn tổng quát hơn về kiến trúc của chúng.

Cơ sở dữ liệu không gian sẽ được dùng để quản lý và truy xuất dữ liệu không gian, được đặt trên data server. Nhà kho hay nơi lưu trữ (clearing house) được dùng để lưu trữ và duy trì những siêu dữ liệu (dữ liệu về dữ liệu - metadata) về dữ liệu không gian tại những data server khác nhau. Dựa trên những thành phần quản lý dữ liệu, ứng dụng server và mô hình server được dùng cho ứng dụng hệ thống để tính toán thông tin không gian thông qua các hàm cụ thể. Tất cả kết quả tính toán của ứng dụng server sẽ được gửi đến web server để thêm vào các gói HTML, gửi cho phía client và hiển thị nơi trình duyệt web.

Xem hình minh họa dưới đây. Lưu ý là tất cả các thành phần đều được kết nối nhau thông qua mạng Internet.



Hình 1-27 Kiến trúc WebGIS

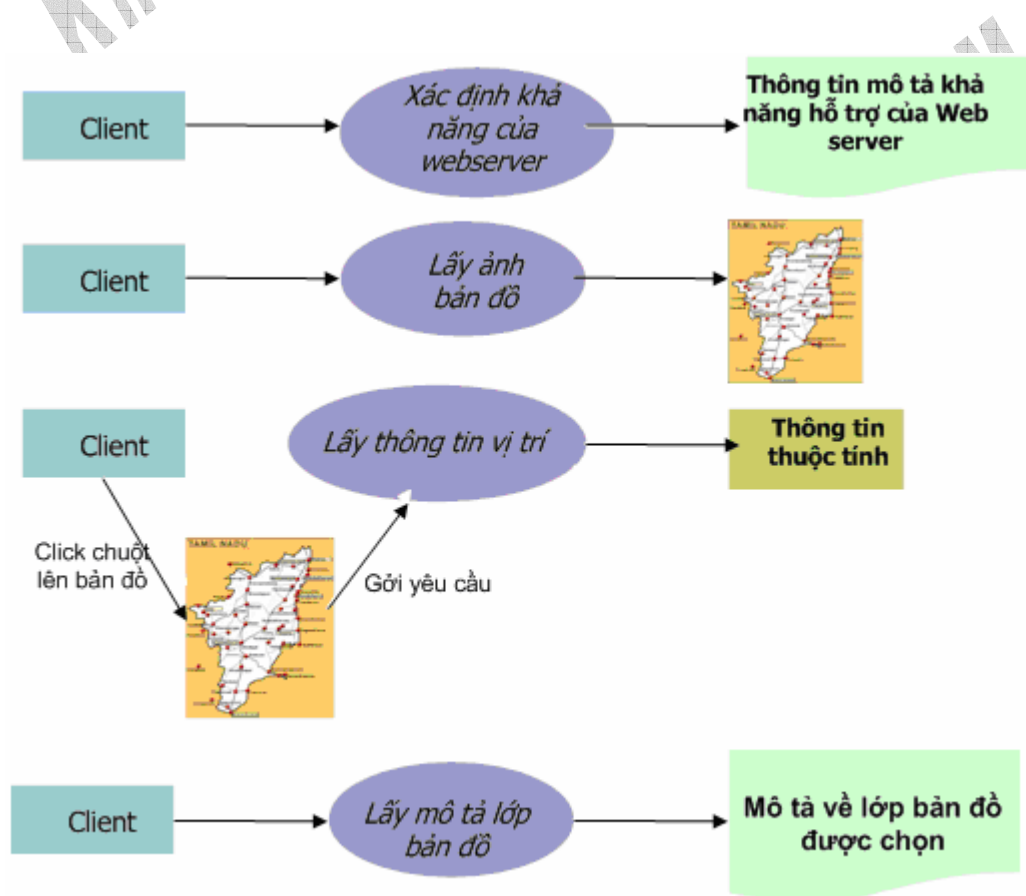
#### 1.4.1.2 Các bước xử lý

Quá trình làm việc với hệ thống web xử lý thông tin không gian được minh họa như trên hình vẽ trên.

Người dùng sử dụng trình duyệt web ở phía client (thường là giao diện đồ họa).

- Client gửi yêu cầu của người sử dụng thông qua giao thức HTTP đến webserver.
- Web server nhận yêu cầu của người dùng gửi đến từ phía client, xử lý và chuyển tiếp yêu cầu đến ứng dụng trên server có liên quan.
- Application server (chính là các ứng dụng GIS) nhận các yêu cầu cụ thể đối với ứng dụng và gọi các hàm có liên quan để tính toán xử lý. Nếu có yêu cầu dữ liệu nó sẽ gửi yêu cầu dữ liệu đến data exchange server(server trao đổi dữ liệu)..

- d. Data exchange server nhận yêu cầu dữ liệu và tìm kiếm vị trí của những dữ liệu này sau đó gửi yêu cầu dữ liệu đến server chứa dữ liệu (data server) tương ứng cần tìm.
- e. Data server dữ liệu tiến hành truy vấn lấy ra dữ liệu cần thiết và trả dữ liệu này về cho data exchange server.
- f. Data exchange server nhận dữ liệu từ nhiều nguồn data server khác nhau nằm rải rác trên mạng. Sắp xếp dữ liệu lại theo logic của yêu cầu dữ liệu, sau đó gửi trả dữ liệu về cho application server.
- g. Application server nhận dữ liệu trả về từ các data exchange server và đưa chúng đến các hàm cần sử dụng, xử lý chúng tại đây và kết quả được trả về cho web server.
- h. Web server nhận về kết quả xử lý, thêm vào các ngữ cảnh web (HTML, PHP..) để có thể hiển thị được trên trình duyệt và cuối cùng gửi trả kết quả về cho trình duyệt dưới dạng các trang web.



Hình 1-28 Các dạng yêu cầu từ phía Client

### 1.4.2 Các kiến trúc triển khai

Trong mô hình hoạt động của WebGIS được chia ra 2 phần : các hoạt động ở phía client – client side và các hoạt động xử lý ở phía server ( server side).

#### ❖ *Client side*

Client side được dùng để hiển thị kết quả đến cho người dùng, nhận các điều khiển trực tiếp từ người dùng và tương tác với web server thông qua trình duyệt web.

Các trình duyệt web sử dụng chủ yếu HTML để định dạng trang web. Thêm vào đó một vài plug-in, ActiveX và các mã Applet được nhúng vào trình duyệt để tăng tính tương tác với người dùng.

#### ❖ *Server side*

Gồm có: Web server, Application server, Data server và Clearinghouse..

Server side có nhiệm vụ lưu trữ dữ liệu không gian, xử lý tính toán và trả về kết quả (dưới dạng hiển thị được) cho client side.

- Web server

Web server được dùng để phục vụ cho các ứng dụng web, web server sử dụng nghi thức HTTP để giao tiếp với trình duyệt web ở phía client. Tất cả các yêu cầu từ phía client đối với ứng dụng web đều được web server nhận và thông dịch và sau đó gọi các chức năng của ứng dụng thông qua các giao tiếp mạng như MAPI, Winsock, namped pipe...

- Application server

Đây là phần chương trình gọi các hàm xử lý GIS, gửi yêu cầu lấy dữ liệu đến clearinghouse

- Data server

Data server là phần cơ bản của hầu hết các hệ thống thông tin với nhiệm vụ quản lý và điều khiển truy cập dữ liệu.

Ban đầu, đa số GIS sử dụng File System để quản lý dữ liệu không gian và DBMS (Database Management System) để quản lý dữ liệu thuộc tính. Ngày nay có nhiều sản phẩm và giải pháp phần mềm thay thế để quản lý dữ liệu không gian và thuộc tính một cách chung nhất

Ví dụ:

SDE của ESRI (1998), SpatialWare của MapInfo (1998)...

Nhìn chung các cơ sở dữ liệu sử dụng đều là các cơ sở dữ liệu quan hệ, và trong tương lai sẽ thay thế bằng cơ sở dữ liệu hướng đối tượng.

- Clearinghouse

Clearinghouse được sử dụng để chứa dữ liệu về dữ liệu không gian được quản lý bởi các data server. Clearinghouse đóng vai trò như một cuốn catalog, clearinghouse tìm kiếm trong catalog này để tìm dữ liệu cần.

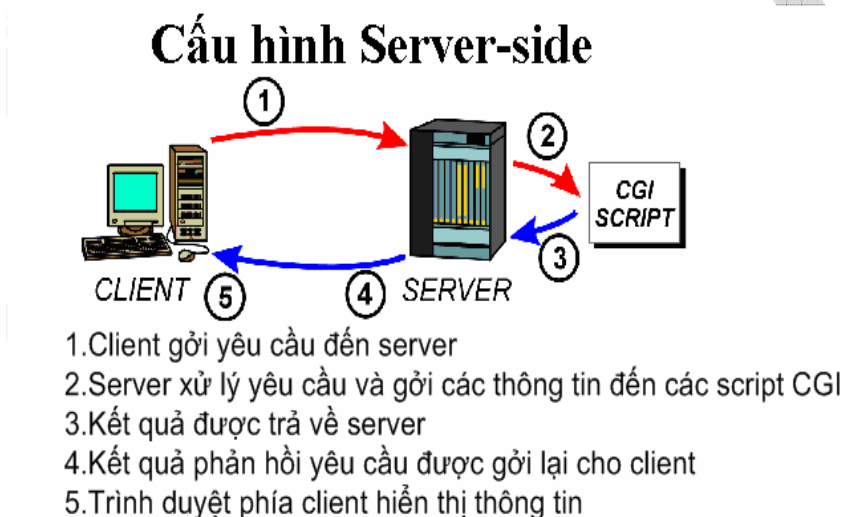
Có 2 chiến thuật lựa chọn, tương ứng với 2 kiểu triển khai, kiểu thứ nhất tập trung công việc chủ yếu cho phía server, kiểu kia ngược lại tập trung công việc cho phía client.

#### 1.4.2.1 Chiến thuật Server – side

Những chiến thuật này tập trung vào việc cung cấp dữ liệu GIS và phân tích “theo yêu cầu” bởi một server đủ mạnh, server này sẽ truy cập dữ liệu và phần mềm cần thiết để xử lý dữ liệu.

Chiến thuật server- side có thể so sánh với mô hình sử dụng máy mainframe để chạy GIS trong một mạng cục bộ. Trong đó, máy client cấu hình không đòi hỏi cao, chỉ cần chạy chương trình để gọi các yêu cầu và hiển thị được các trả lời từ server.

Các bước xử lý như hình vẽ:



Hình 1-29 Cấu hình Server Side

Trong WebGIS đôi khi thuật ngữ “map server” được dùng để chỉ ra rằng chiến thuật áp dụng là server- side. Mà trong đó khi người dùng gửi yêu cầu cần “map” để hiển thị, thì sẽ được “phục vụ” bởi server.

Chiến thuật server-side dựa trên khả năng trình duyệt web của người dùng có thể gửi các yêu cầu đến các phần mềm GIS trên server thông qua Internet

Các chương trình được dùng để nhận và xử lý yêu cầu người dùng có thể được viết bằng các ngôn ngữ như : Perl, Visual Basic, C++... Các chương trình này cũng có thể mua từ các nhà sản xuất để tạo khả năng kết nối tốt hơn đến các hệ xử lý GIS đã tồn tại.

Để có thể giao tiếp với các ứng dụng WebGIS đặt trên server, web server có thể sử dụng các chuẩn giao tiếp phổ biến như CGI (Common Gateway Interface), Java, ISAPI (Internet Server Application Programming Interface), and NSAPI (Netscape Server Application Programming Interface)...

Các thuận lợi và khó khăn của chiến thuật này

▪ *Thuận lợi*

- Với server có cấu hình mạnh được sử dụng, người dùng có thể truy xuất trên tập dữ liệu lớn hơn và phức tạp hơn. Thay vì phải xử lý trên máy client, hầu hết không được cấu hình mạnh và việc truyền dữ liệu lớn qua mạng Internet sẽ gây nhiều khó khăn.
- Cũng với server mạnh, việc phân tích, xử lý các chức năng GIS sẽ được tiến hành nhanh và không đòi hỏi quá nhiều ở người dùng sự am hiểu.
- Kiểm soát được các thao tác của người dùng (chủ yếu là đơn giản) trên dữ liệu và luôn đảm bảo người dùng nhận kết quả đúng từ dữ liệu (do phía client không phải xử lý nhiều).

▪ *Khó khăn*

- Với chiến thuật này thì dù yêu cầu là nhỏ (client hoàn toàn xử lý được) hay lớn, tất cả đều gửi về phía server, và server xử lý xong lại gửi trả về cho client thông qua đường truyền trên mạng.
- Hiệu năng của hệ thống WebGIS sẽ bị ảnh hưởng bởi băng thông và đường truyền mạng Internet giữa server và client. Nhất là khi mà kết quả trả về phải mang chuyên những file lớn.
- Hệ thống WebGIS sử dụng chiến thuật này không tận dụng được khả năng xử lý trên máy client. Chủ yếu client chỉ xử lý gửi yêu cầu và hiển thị kết quả đáp ứng.

Nhìn chung, chiến thuật này áp dụng tốt nhất cho các ứng dụng WebGIS thương mại hay cộng đồng với số lượng lớn người dùng mà không quan tâm đến khả năng xử lý GIS trên các máy người dùng.

Chiến thuật Server-side	
Nhiệm vụ phía Server	Nhiệm vụ Client
Duyệt bản đồ	Hiển thị
Truy vấn	
Phân tích	
Vẽ bản đồ	

Bảng 1-3 Chiến thuật Server-side

#### 1.4.2.2 Chiến thuật Client – side

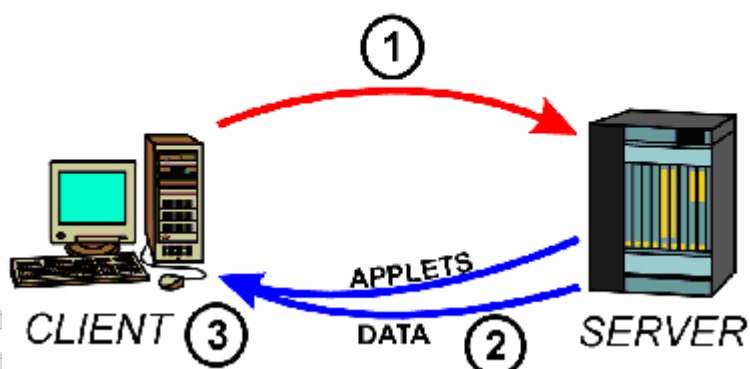
Thay vì để server làm quá nhiều việc, một số chức năng xử lý GIS sẽ được đưa về phía máy người dùng, và tại đây sẽ có một phần dữ liệu được xử lý.

Có 2 dạng triển khai chiến thuật client side như sau:

a. GIS Applet được phân phối đến Client khi có yêu cầu

Trong cách triển khai chiến thuật này các xử lý GIS sẽ được server cung cấp cho phía client dưới dạng các chương trình thực thi nhỏ hoặc là các applet để có thể chạy được ở phía client. Những applet như vậy được phân phối đến client khi client cần nó để xử lý.

#### Cấu hình Client-Side



Hình 1-30 Cấu hình Client side

Các bước xử lý:



- Người dùng tạo ra một yêu cầu từ trình duyệt
- Yêu cầu được chuyển qua Internet đến server (1).
- Server xử lý các yêu cầu
- Kết quả phản hồi trả về bao gồm dữ liệu và các applet cần thiết để người dùng có thể làm việc trên dữ liệu này.

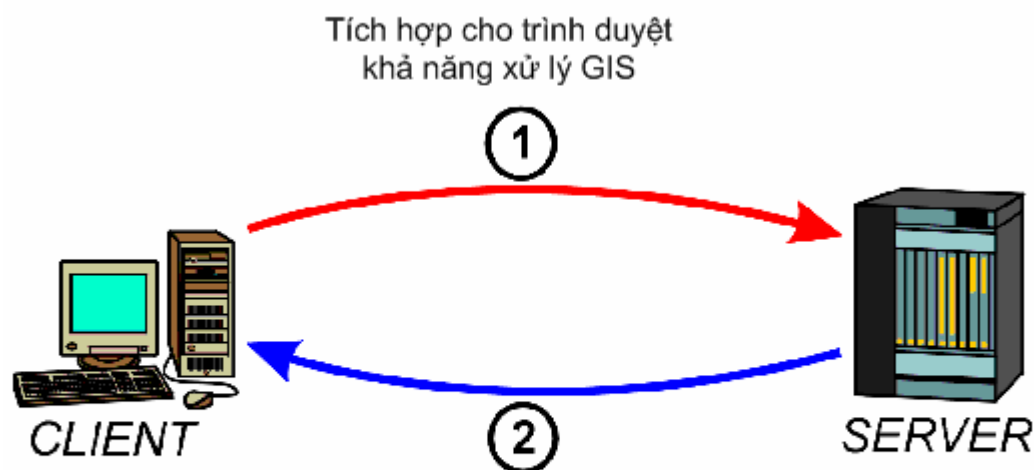
Các applet có thể được viết bằng Java, JavaScript hoặc ở dạng các ActiveX. Như vậy trình duyệt cần được tích hợp các compiler để xử lý các applet này.

*b. GIS Applet và Plug-in cố định ở Client*

Các triển khai trên cần thêm các chức năng xử lý GIS vào trình duyệt. Tuy nhiên việc chuyển dữ liệu và các applet cần thiết qua mạng Internet mất nhiều thời gian nhất là khi mà ứng dụng ít được dùng đến.

Giải pháp cho vấn đề này là cách triển khai sau đây:

- Chuyển các GIS applet đến máy tính client một cách cố định, và không phải chuyển đến mỗi khi cần nữa
- Download và cài đặt cố định các plug-in vào trình duyệt web của client
- Xây dựng một trình duyệt web có tích hợp sẵn phần mềm xử lý GIS để chạy trên client.
- Tích hợp các link đến nguồn tài nguyên dữ liệu khác trên mạng trong mỗi gói dữ liệu tải về.
- Server chỉ được gọi khi client khi cần dữ liệu mới, hoặc dữ liệu cho một ứng dụng mới
- Người dùng được toàn quyền thao tác trên dữ liệu họ dùng và phân tích chúng.



1. Tích hợp các link lấy dữ liệu cho các yêu cầu
2. Dữ liệu được phân phối khi cần

Hình 1-31 Tích hợp xử lý GIS vào trình duyệt

Các thuận lợi và khó khăn của chiến thuật này

- *Thuận lợi*
  - Tận dụng sức mạnh xử lý trên máy người dùng
  - Người dùng được quyền điều khiển trong quá trình phân tích dữ liệu
  - Khi server gửi dữ liệu cần thiết về, người dùng có thể làm việc với dữ liệu này mà không phải gửi đi hay nhận về cái gì khác từ mạng.
- *Khó khăn*
  - Các hồi đáp từ server có thể bao gồm lượng lớn dữ liệu cũng như các applet (cho lần đầu tiên khi trình duyệt tại máy sử dụng chạy ứng dụng WebGIS) có thể dẫn đến sự trì hoãn.
  - Dữ liệu GIS thường lớn và phức tạp dẫn đến sẽ khó xử lý nếu client không được cấu hình mạnh.
  - Người dùng có thể chưa được huấn luyện đầy đủ để thực hiện các chức năng phân tích dữ liệu một cách đúng đắn → không dành cho người dùng bình thường.

Chiến thuật Client-side	
Phân tích	Hiển thị
Vẽ bản đồ	Duyệt bản đồ
	Truy vấn

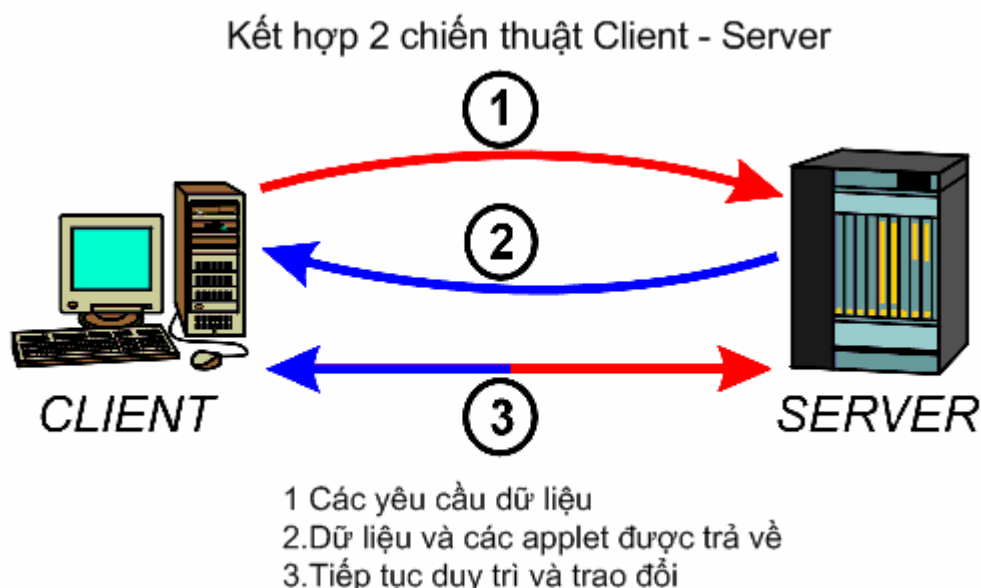
Bảng 1-4 Công việc tại Client với chiến thuật Client side

### 1.4.2.3 Kết hợp hai chiến thuật

Áp dụng thuần túy 2 chiến dịch trên đều có những hạn chế nhất định. Đối với chiến thuật Server-side chất lượng đường truyền sẽ ảnh hưởng đến tốc độ và thời gian truyền giữa yêu cầu và hồi đáp. Trong khi đó với client-side lại phụ thuộc vào cấu hình máy client. Một vài thao tác có thể chậm do đòi hỏi bộ xử lý mạnh không được đáp ứng.

Kết hợp 2 chiến thuật cho ta một giải pháp “lai”, tận dụng được ưu điểm của 2 chiến thuật trên. Những công việc đòi hỏi dữ liệu lớn, tính toán phức tạp giao cho server xử lý. Những công việc đòi hỏi người dùng có quyền điều khiển cao (thao tác bản đồ, v.v..) được giao cho client. Như vậy đòi hỏi thông tin về cấu hình của server và client cần được chia sẻ cho nhau.

Giải pháp này tỏ ra hiệu quả khi mà client thỉnh thoảng mới cần liên lạc với server để lấy dữ liệu.

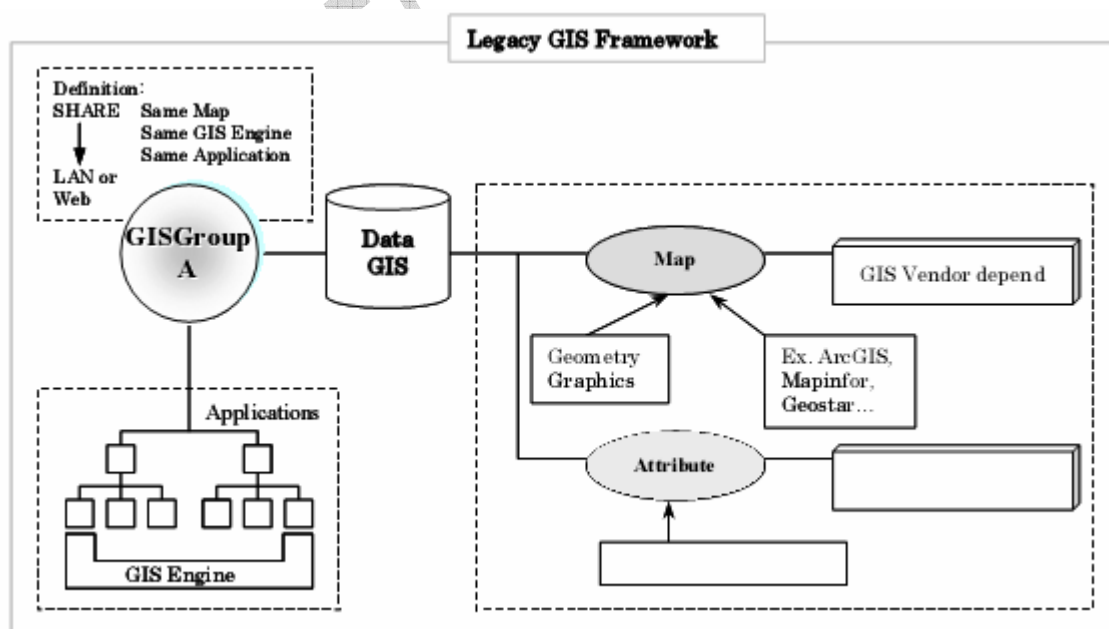


Hình 1-32 Kết hợp Client side và Server side

## 1.5 Các chuẩn trao đổi WebGIS hiện nay

### 1.5.1 Vấn đề trong việc trao đổi dữ liệu của hệ thống WebGIS

Từ nhiều thập niên nay, một số lượng lớn ứng dụng GIS đã được phát triển trong nhiều lĩnh vực như điều khiển, quản lý tài nguyên, giao thông, giáo dục, tài nguyên nước, trong lĩnh vực quân sự... Ứng với mỗi ứng dụng GIS, dữ liệu GIS cũng được tạo ra tương ứng. Thông thường các dữ liệu này sẽ rất lớn và tốn thời gian và công sức để xây dựng. Vấn đề chia sẻ nguồn tài nguyên dữ liệu được đặt ra nhằm giảm thiểu chi phí xây dựng và quản lý dữ liệu GIS.

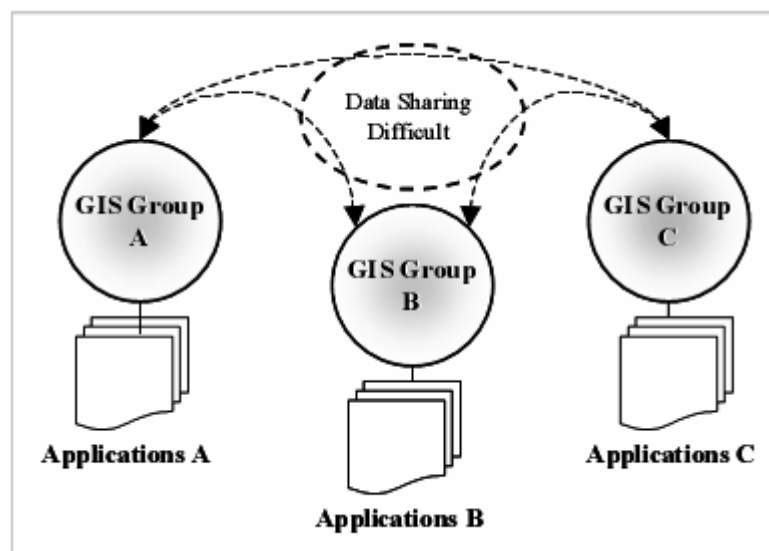


Hình 1-33 Dữ liệu GIS trong kiến trúc WebGIS đơn thể

Trong mô hình kiến trúc trên, hệ thống WebGIS được gọi là đơn thể. Vì trong đó dữ liệu GIS được dùng chỉ cho một nhóm ứng dụng GIS. Trong khi dữ liệu GIS này có thể được sử dụng cho các nhóm ứng dụng GIS khác. Việc chia sẻ dữ liệu GIS này thường khó khăn, các nhóm ứng dụng GIS sẽ dùng chung trên toàn thể dữ liệu này.

Ví dụ:

*Dữ liệu bản đồ thành phố dùng cho nhóm các ứng dụng tìm đường đi trong thành phố. Dữ liệu này cũng có thể được dùng cho các nhóm ứng dụng tìm thông tin nơi chốn như quán ăn nhà hàng hoặc nhóm ứng dụng liên quan đến hệ định vị toàn cầu vv...*



Hình 1-34 Chia sẻ dữ liệu GIS giữa các nhóm ứng dụng

Bài toán chia sẻ dữ liệu được giải quyết bằng 2 chuẩn trao đổi dữ liệu WebGIS, do tổ chức OGC định nghĩa.

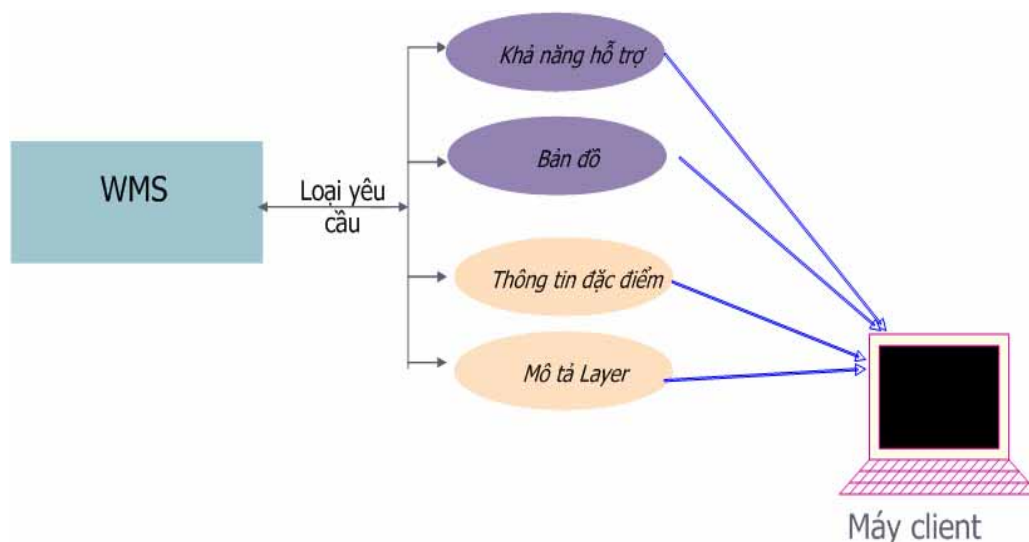
## 1.5.2 Giải pháp cho vấn đề chia sẻ dữ liệu

### 1.5.2.1 Web Map Service / Server

Đây là một chuẩn do tổ chức OGC đưa ra. Trong đó web server sẽ trở thành web map server có service phục vụ cho chia sẻ dữ liệu. Các hoạt động mà client có thể thực hiện thông qua service này gồm : nhận về mô tả các bản đồ, nhận về bản đồ, và các thông tin truy vấn các đặc điểm được thể hiện trên bản đồ.

Chuẩn này không áp dụng cho việc nhận về dữ liệu thô (dữ liệu chưa xử lý dạng thuộc tính hay không gian) mà thường nhận về một ảnh bản đồ dạng đồ họa. Những bản đồ này thường được tạo ra với các định dạng như PNG, GIF, JPEG hoặc cũng có thể là dưới dạng các yếu tố đồ họa như SVG (dạng XML) hoặc là định dạng WebCGM (Web Computer Graphics Metafile).

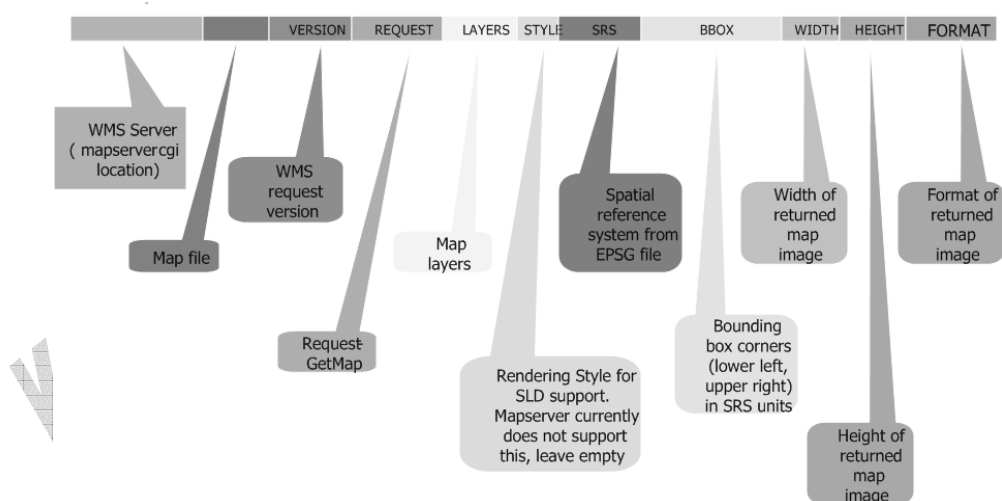
Sau đây là các chức năng của một web map service:



Hình 1-35 Các chức năng của một WMS

- GetCapabilities - Khả năng hỗ trợ (bắt buộc): Client nhận về một mô tả thông tin về WMS, các tham số được chấp nhận và hỗ trợ, bảng mô tả, thường dưới dạng file XML.
- GetMap - Lấy bản đồ (bắt buộc) : Client nhận về một ảnh bản đồ phù hợp với tham số mà client gửi lên server.
- GetFeatureInfo - Lấy thông tin đặc điểm ( không bắt buộc) : Client hỏi thông tin về đặc điểm nào đó (đối tượng) xuất hiện trên bản đồ.

Trình duyệt web phía client có thể thông qua Web Map Service thực hiện các chức năng này bằng cách gửi các yêu cầu dưới dạng một URL. Nội dung của chuỗi URL này phụ thuộc vào công việc được yêu cầu (chỉ ra bởi tham số Request Type).



Hình 1-36 Các tham số trong chuỗi URL thực hiện chức năng GetMap

Ví dụ :

+ Lấy thông tin về WMS Server

*Chuỗi URL :*

<http://mymapserv/cgi-bin/mapserv?map=Tamilnadu.map&request=capabilities>

*Kết quả :*

Nhận về một tài liệu XML mô tả thông tin về WMS Server.

+ Lấy bản đồ của sông Tiền với layer kích thước 500x700.

*Chuỗi URL :*

<http://tnmapserver/cgi-bin/mapserv?map=tienmap.map&VERSION=1.1.1&layers=state,rivers&Height=500&width=700&request=GetMap>

*Kết quả :*

Nhận về một ảnh bản đồ dòng sông của Tiền.

Client của Web map server ở đây có thể là trình duyệt web, hay cũng có thể là một web server có chức năng GIS.

#### 1.5.2.2 Web Feature Service / Server

Đây cũng là chuẩn do OGC đưa ra. Trong đó web server giờ được gọi là Web Feature server có service phục vụ việc chia sẻ dữ liệu. Tuy nhiên thay vì trả về một ảnh bản đồ dạng đồ họa thì Web Feature Server sẽ gửi trả về thông tin không gian và thông tin thuộc tính có liên quan dưới dạng file GML (Geographic Markup Language) một dạng XML và sau đó client sử dụng file XML này làm dữ liệu để tạo ra ảnh bản đồ.

Web Feature Service hoàn toàn tương tự như Web Map Service về các chức năng, cách làm việc chỉ khác nhau dạng dữ liệu trao đổi.

## Chương 2 : MapServer – WebGIS Application

### 2.1 Lược sử phát triển

MapServer là môi trường phát triển cho việc xây dựng các ứng dụng GIS thông qua Internet. Trong mô hình kiến trúc WebGIS, MapServer đóng vai trò là ứng dụng GIS được đặt trên web server. Đây là phần mềm mã nguồn mở được xây dựng với các thành phần cũng ở dạng nguồn mở hoặc phi lợi nhuận.

MapServer được phát triển bởi trường đại học Minesota, bang Minesota Hoa Kỳ, dưới sự tài trợ của NASA thông qua các dự án ForNet và sau đó Terraship. Hiện nay số lượng các nhà phát triển tham gia phát triển ngày càng nhiều trên thế giới.

MapServer có thể chạy trên nhiều môi trường điều mà các phần mềm thương mại khác chưa làm được. MapServer với mã nguồn được viết bằng C++ được biên dịch để có thể chạy trên các version của UNIX/Linux, Microsoft Windows và cả trên MacOS.

Để giao tiếp với các thành phần khác trên môi trường web, MapServer sử dụng chuẩn giao tiếp CGI (Common Gateway Interface).

MapServer hỗ trợ các chuẩn của tổ chức OGC ( tổ chức phát triển các chuẩn WebGIS) như :WMS, WFS, WCS, WMC,SLD, GML v.v..

Hệ thống MapServer bao gồm cả MapScript, cho phép các ngôn ngữ kịch bản khác như PHP, Perl, Python và Java có thể truy xuất các hàm API của MapServer. MapScript cung cấp môi trường thuận lợi cho việc phát triển các ứng dụng tích hợp các dữ liệu phân tán. Ta có thể lấy dữ liệu không gian thông qua các các ngôn ngữ kịch bản kể trên và dựa vào MapScript ta có thể tạo được một ảnh bản đồ.

Ví dụ sử dụng module Perl's DBI, cho phép tích hợp dữ liệu từ cơ sở dữ liệu của nhiều hãng (Oracle, Sybase, MySQL) với dữ liệu GIS truyền thống trong một ảnh bản đồ hoặc trang web.

MapServer không hẳn là một ứng dụng WebGIS hoàn chỉnh tuy nhiên MapServer cung cấp những chức năng cốt lõi đủ mạnh để đáp ứng cho các ứng dụng web khác nhau. Ngoài việc tương tác với các dữ liệu GIS, MapServer còn cho phép người dùng điều khiển và tùy biến việc tạo ra ảnh bản đồ, có thể dưới dạng trang web, file ảnh,report...Nói



cách khác MapServer đóng vai trò như “map engine” được cung cấp nội dung để tạo ảnh bản đồ khi cần đến.

## 2.2 Các thành phần và mô hình xử lý của MapServer

### 2.2.1 Các thành phần của MapServer

Ứng dụng MapServer sử dụng chuẩn giao tiếp CGI để giao tiếp với các thành phần và với HTTP Server. Cho nên đôi khi còn gọi ứng dụng MapServer là ứng dụng MapServer CGI. Do có mã nguồn mở nên cũng có những ứng dụng được biên dịch để có thể dùng MapScript truy xuất trực tiếp các hàm API của MapServer.

Ứng dụng MapServer CGI sử dụng các tài nguyên như sau:

- i. Một HTTP Server như Apache hoặc Internet Information Server (vai trò web server).
- ii. Phần mềm MapServer (vai trò WebGIS application).
- iii. File khởi tạo, dùng để cấu hình và tùy biến các thông số của ứng dụng MapServer (không bắt buộc).
- iv. Một file text được gọi là Mapfile, điều khiển cách tương tác với dữ liệu. Như lấy dữ liệu gì, ở đâu, sắp xếp chúng theo thứ tự nào ?...
- v. Một template file được dùng để định dạng kết quả (đối với trường hợp truy vấn) được trình bày theo định dạng nào trên cửa sổ trình duyệt.
- vi. Tập dữ liệu GIS.

#### 2.2.1.1 Initialization file

MapServer “không có trạng thái” nghĩa là mỗi khi http server nhận được yêu cầu từ client thông qua chuỗi URL thì http server sẽ gọi MapServer chạy và thực thi. Bằng cách sử dụng file khởi tạo (dưới dạng file html hoặc htm ) ta định nghĩa những tham số cơ bản cần thiết để MapServer có thể thực thi được.

Ví dụ :

```
img_file=[img]
img_ext=[mapext]
img_xy=[center]
```

Thông qua chuỗi URL nhận được, sử dụng file khởi tạo này MapServer biết được các tham số cơ bản này. Các tham số này còn được MapServer gửi trả kèm trong kết quả cho phía client.

### 2.2.1.2 Mapfile

Mapfile định nghĩa dữ liệu sẽ được dùng như thế nào trong ứng dụng, cách hiển thị và các tham số cho câu truy vấn. Mapfile có thể xem như là một file cấu hình cho ứng dụng. Mapfile cũng bao gồm cả thông tin về vẽ bản đồ như thế nào, ghi chú bản đồ ra sao và vẽ kết quả của câu truy vấn. Mapfile có phần mở rộng là .map.

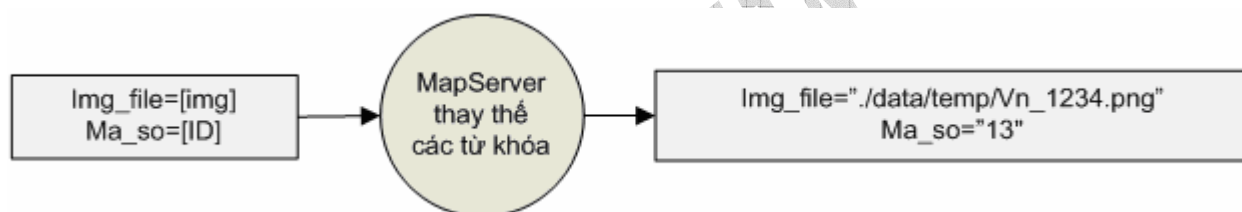
Ta sẽ tìm hiểu Mapfile ở phần sau

### 2.2.1.3 Template File

Template file điều khiển các hình bản đồ và các ghi chú trả về bởi MapServer sẽ xuất hiện trên trang html. Cách làm của MapServer như sau. Trước hết MapServer đọc file template này và nếu gặp các từ khóa hoặc các từ mẫu thì nó sẽ thay thế các giá trị tương ứng với lấy từ chuỗi kết quả trả về, cuối cùng file html này được gửi về cho trình duyệt. Bởi vì template file sẽ được dùng để tạo ra một trang html nên thông thường template file cũng được lưu dưới dạng một trang html với phần mở rộng .html.

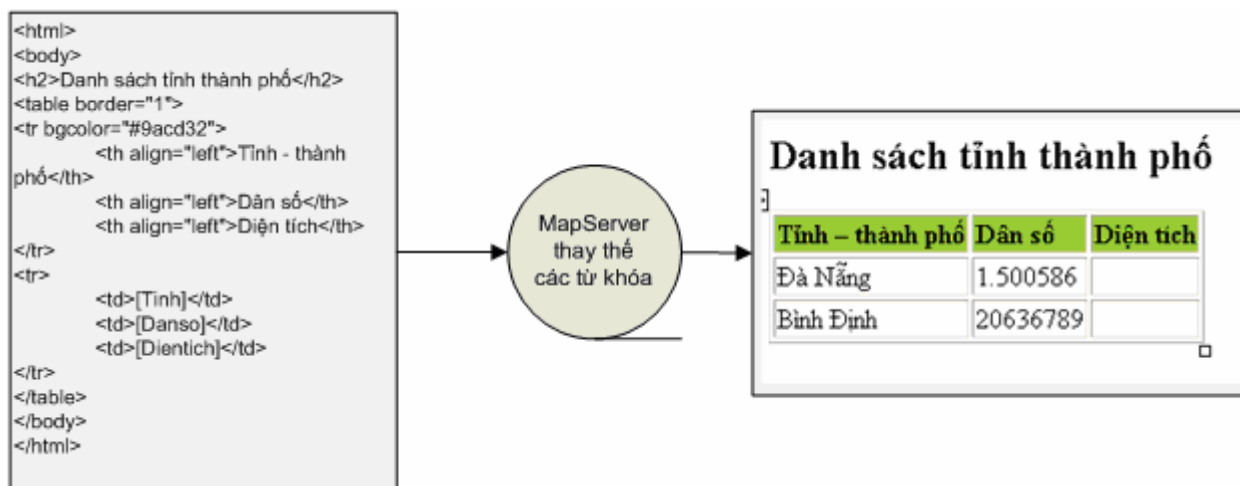
Ví dụ:

Template file đơn giản chỉ chứa các field sẽ được MapServer thay thế



Hình 2-1 Xử lý với file Template đơn giản

Ta cũng có thể định dạng template file



Hình 2-2 Xử lý định dạng file Template

Ngoài ra trình duyệt có thể sử dụng trang html được phát sinh từ template file để hiển thị cho người dùng, cho nên trong template file cũng có thể chứa các đoạn mã các ngôn ngữ script, thẻ html vv... quy định tương tác của người dùng (phóng to, thu nhỏ dịch chuyển...).

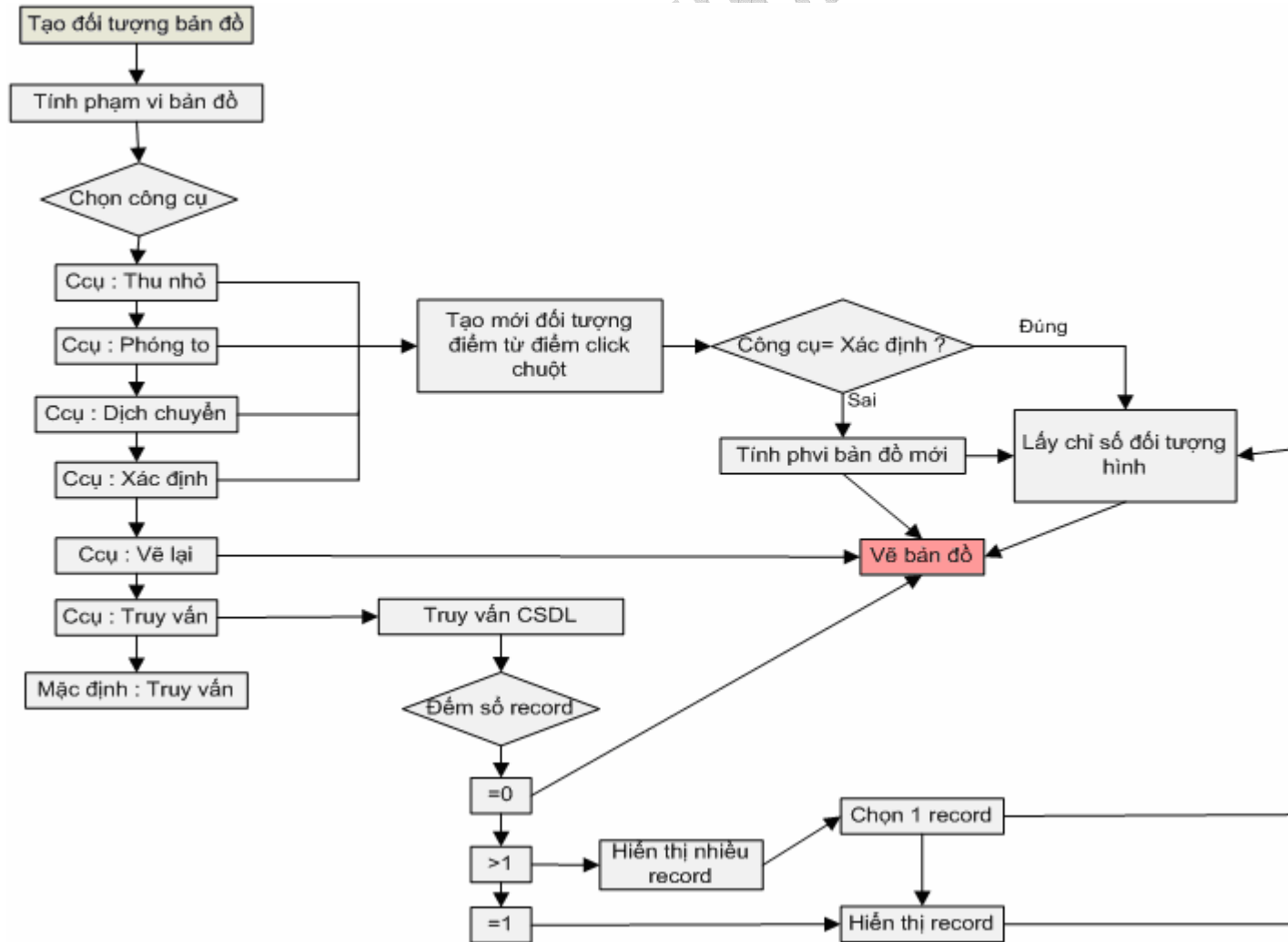
#### 2.2.1.4 GIS Dataset

Về dữ liệu vector, MapServer sử dụng shapefile (của ESRI) làm định dạng dữ liệu mặc định. Bên cạnh đó hầu hết các định dạng dữ liệu vector GIS phổ biến trên thế giới hiện nay, MapServer đều có thể hiểu và tương tác được. Như: PostGIS, ESRI ArcSDE, MapInfo... và nhiều định dạng khác thông qua thư viện OGR.

Về dữ liệu raster, MapServer hỗ trợ TIFF/GeoTIFF, EPPL7 ... và các định dạng khác thông qua thư viện GDAL.

MapServer có mã nguồn mở nên người dùng có thể biên dịch lại MapServer để thêm hay bỏ bớt các định dạng dữ liệu không muốn hỗ trợ.

### 2.2.2 Quy trình xử lý



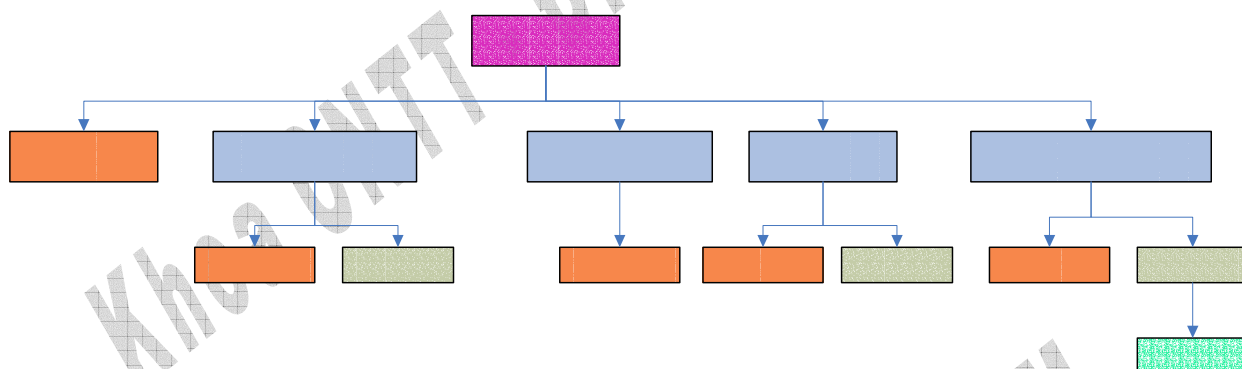
Hình 2-3 Quy trình xử lý của MapServer

## 2.3 Tìm hiểu Mapfile

Mapfile được xem như file cấu hình cho ứng dụng dùng MapServer. Trong phần này ta sẽ tìm hiểu về những đối tượng trong Mapfile, thiết lập các giá trị cho chúng.

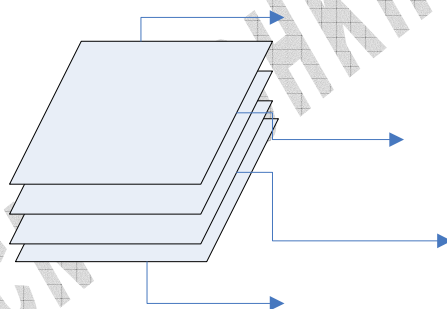
Trong Mapfile có nhiều đối tượng như MAP, PROJECTION, LAYER, CLASS... mỗi đối tượng định nghĩa cách thức tạo nên ảnh bản đồ hoặc đối tượng để MapServer truy xuất dữ liệu cho các câu truy vấn.

❖ Ví dụ :



Hình 2-4 Mô hình đối tượng trong Mapfile

Trong hình vẽ minh họa trên, ảnh bản đồ (đối tượng MAP) là sự chồng lấp do bốn layer tạo thành, trong đó có một layer sử dụng Raster (ảnh đồ họa) còn lại ba layer được vẽ từ dữ liệu vector : polygon, line, annotation...



Hình 2-5 Chồng lấp các layer

Mỗi layer được vẽ, cần được chỉ ra nguồn dữ liệu (vector hay raster), hệ quy chiếu cho mỗi layer... loại đối tượng được vẽ (line, label, polygon...). Tất cả được định nghĩa trong từng đối tượng của mapfile.

### 2.3.1 Map Object

Trong file MapFile thì MAP chính là đối tượng gốc, chứa trong nó các đối tượng khác.

- *CONFIG [key] [value]*

Tham số này được dùng để định nghĩa vị trí đặt file EPSG dùng cho thư viện PROJ.4 ([X]). Giá trị [key] là PROJ\_LIB và [value] là đường dẫn đến EPSG file. Tham số CONFIG được thiết lập để tránh việc phải thiết lập biến môi trường PROJ\_LIB đòi hỏi quyền Admin.

Ví dụ:

```
CONFIG PROJ_LIB /tmp/proj/
```

- *DEBUG [on|off]*

Cho phép thực hiện debug trên các đối tượng map. Ngoài kết quả, MapServer sẽ ghi các kết quả debug vào logfile nếu như logfile được chỉ ra trong tham số LOG của đối tượng WEB.

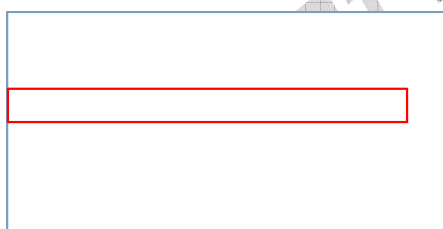
- *EXTENT [minx] [miny] [maxx] [maxy]*

Không gian phạm vi của ảnh bản đồ được tạo ra. Nếu giá trị EXTENT không được gán thì MapServer cũng có thể nội suy ra một giá trị từ dữ liệu và vị trí trung tâm của ảnh bản đồ.

- *FONTSET [filename]*

Tập tin liệt kê danh sách các font được dùng. Định dạng rất đơn giản. Mỗi dòng chứa hai thành phần : một bí danh và một là đường dẫn đến font được phân cách bằng khoảng trắng. Alias đơn giản là tên mà ta dùng để chỉ font này trong mapfile. Các font sử dụng được với MapServer là các True Font Type.

Ví dụ :



Hình 2-6 Danh sách font sử dụng

- *IMAGECOLOR [r] [g] [b]*

Màu được dùng làm background cho ảnh bản đồ. Khi mà thuộc tính Transparency (trong suốt) được chọn, thì màu này sẽ được đánh dấu như là màu trong suốt trong bảng màu. Khi đó thành phần nào của ảnh bản đồ sử dụng màu này để vẽ cũng trong suốt. Vì thế trong khi tạo ảnh bản đồ nếu chọn ảnh bản đồ trong suốt thì nên chọn màu Imagecolor là màu không được dùng để vẽ các thành phần khác trên bản đồ.

- *IMAGETYPE* [*gif|png|jpeg|wbmp|gtiff|swf|userdefined*]

Định dạng ảnh bản đồ được tạo ra.

- *LAYER*

Bắt đầu cho đối tượng LAYER.

- *LEGEND*

Bắt đầu cho đối tượng LEGEND.

- *NAME* [*name*]

Xác định tiền tố cho tên ảnh bản đồ, ảnh các thước tỉ lệ, ghi chú được tạo ra từ Mapfile này.

Ví dụ:

NAME VN\_

Các ảnh bản đồ tạo ra sẽ có tiền tố là VN\_ như VN\_11197048662768.png,

VN\_11197048992800.png, VN\_1119719302224.png...

- *PROJECTION*

Bắt đầu cho đối tượng PROJECTION.

- *QUERYMAP*

Bắt đầu khai báo đối tượng QUERYMAP.

- *REFERENCE*

Bắt đầu đối tượng REFERENCE.

- *RESOLUTION* [*int*]

Định độ phân giải cho ảnh kết quả, độ phân giải sẽ ảnh hưởng đến việc tính toán tỉ lệ. Mặc định là 72.

- *SCALE* [*double*]

Tính toán tỉ lệ của bản đồ.

- *SHAPEPATH* [*filename*]

Đường dẫn đến dữ liệu dạng vector.

- *SIZE [x][y]*

Kích thước theo đơn vị pixel của ảnh bản đồ.

- *STATUS [on|off]*

Trong mapfile ta có thể định nghĩa : ảnh bản đồ, thước tỉ lệ, ảnh tham chiếu (dạng ảnh nhỏ toàn cục). STATUS cho phép ta lựa chọn có kích hoạt ảnh bản đồ không ? Nếu không được kích hoạt MapServer sẽ không tạo ra ảnh bản đồ khi sử dụng mapfile này.

- *SYMBOLSET [filename]*

File name chứa tập hợp các biểu tượng được dùng trên bản đồ. Trên bản đồ, các symbol được dùng để đánh dấu các đối tượng nhằm làm nổi bật và tăng thêm ngữ nghĩa.

- *SYMBOL*

Dấu hiệu bắt đầu của đối tượng SYMBOL.

- *TEMPLATEPATTERN [regular expression]* và *DATAPATTERN [regular expression]*

Trong request được gọi lên từ trình duyệt gồm có 2 dạng tham số là DATA và TEMPLATE. Như đã biết các tham số đều là các từ khóa được MapServer quy định trước và thường khó nhớ. Tuy nhiên bằng cách sử dụng TEMPLATEPATTERN và DATAPATTERN ta có thể định nghĩa một tên khác cho các từ khóa này.

Ví dụ :

[http://terrasip.gis.umn.edu/cgi-](http://terrasip.gis.umn.edu/cgi-bin/mapserv40?map=/data/projects/tutorial/example1-1.map&mode=map)

[bin/mapserv40?map=/data/projects/tutorial/example1-1.map&mode=map](http://terrasip.gis.umn.edu/cgi-bin/mapserv40?map=/data/projects/tutorial/example1-1.map&mode=map)

Bằng cách sử dụng TEMPLATEPATTERN và DATAPATTERN với định nghĩa  
*anh\_ban\_do=map.*

*che\_do\_ban\_do=mode*

Ta có chuỗi URL dễ nhớ hơn

[http://terrasip.gis.umn.edu/cgi-](http://terrasip.gis.umn.edu/cgi-bin/mapserv40?anh_ban_do=/data/projects/tutorial/example1-1.map&che_do_ban_do=map)

[bin/mapserv40?anh\\_ban\\_do=/data/projects/tutorial/example1-](http://terrasip.gis.umn.edu/cgi-bin/mapserv40?anh_ban_do=/data/projects/tutorial/example1-1.map&che_do_ban_do=map)

[1.map&che\\_do\\_ban\\_do=map](http://terrasip.gis.umn.edu/cgi-bin/mapserv40?anh_ban_do=/data/projects/tutorial/example1-1.map&che_do_ban_do=map)

- *TRANSPARENT [on|off]*

Thiết lập nền trong suốt cho ảnh bản đồ hay không? Mặc định là off.

- *UNITS [feet|inches|kilometers|meters|miles|dd]*



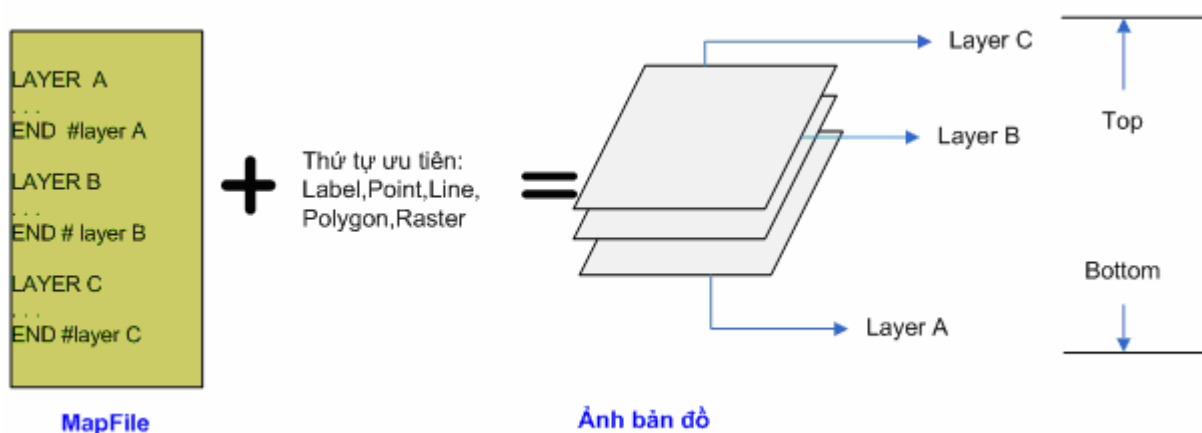
Đơn vị của hệ tọa độ ảnh bản đồ. Được sử dụng cho thước tỉ lệ và các tính toán.

- *WEB*

Dấu hiệu bắt đầu đối tượng WEB.

### 2.3.2 Layer Object

Đây chính là đối tượng được sử dụng nhiều nhất trong một MapFile, mỗi đối tượng layer mô tả một layer được dùng để tạo ra ảnh bản đồ. Các layer được vẽ theo thứ tự xuất hiện trong MapFile (layer đầu tiên ở dưới cùng, layer cuối dùng ở trên).



Hình 2-7 Thứ tự được vẽ của các layer

Các layer được vẽ ra còn được tính thêm thứ tự ưu tiên. Raster độ ưu tiên thấp sẽ được vẽ trước và đặt ở phía dưới, tiếp đến là Vùng (Polygon), Đường (Line), Điểm (Point) và Chú thích (Label). Thứ tự này đảm bảo các layer khi xếp chồng thì không che khuất nhau.

- *CLASS*

Bắt đầu đối tượng CLASS. Trong các định dạng vector, mỗi Layer được vẽ lấy dữ liệu từ một bảng dữ liệu. Mỗi bảng dữ liệu có nhiều trường thuộc tính, mỗi thuộc tính được xem như một CLASS.

- *CLASSITEM [attribute]*

Ứng với tên của trường thuộc tính trong bảng dữ liệu, được định nghĩa trong MapFile.

- *CONNECTIONTYPE [local|sde|ogr|postgis|oraclespatial|wms]*

Kiểu kết nối, mặc định là local. Kiểu kết nối OGR được dùng cho các loại dữ liệu khác ngoài dữ liệu mặc định là shapfile của ESRI. Thực chất OGR là một thư viện được viết

bằng C++, hỗ trợ các kết nối nhiều loại dữ liệu như : MapInfo, Microstation DGN, ArcInfo...

- *CONNECTION [string]*

Câu kết nối CSDL để nhận về dữ liệu đối với dữ liệu nằm trên các server hoặc các hệ DBMS.

Ví dụ :

Câu kết nối đến SDE bao gồm hostname, instance name, database name, username và password được phân cách bằng dấu phẩy.

Câu kết nối đến PostGIS có dạng “user=nobody password=\*\*\*\*\* dbname=dbname host=localhost port=5432”.

Còn câu kết nối đến Oracle : user/pass[@db].

- *DATA [filename]/[sde parameters]/[postgis table/column]/[oracle table/column]*

Tên file đầy đủ của dữ liệu để xử lý. Đối với dữ liệu là shapefiles không cần chỉ rõ phần mở rộng. Đường dẫn có thể là tuyệt đối hoặc tương đối so với giá trị được chỉ ra bởi tham số SHAPEPATH của đối tượng MAP.

Nếu đây là một SDE layer, thì [sde parameters] cần bao gồm cả tên layer cũng như cột dữ liệu địa lý ví dụ : "mylayer,shape".

Nếu đây là một PostGIS layer, thì tham số ở dạng “<columnname> from <tablename>”. Với "columnname" là tên của trường chứa đối tượng địa lý cần thể hiện và "tablename" là tên của bảng dữ liệu cần đọc.

Đối với Oracle, sử dụng “shape FROM table” hoặc là "shape FROM (SELECT statement)” hoặc thậm chí có thể sử dụng các câu truy vấn phức tạp.

- *DUMP [true|false]*

Cho phép Mapserver trả dữ liệu về dưới định dạng GML. Đặc biệt có ích khi sử dụng với hành động WMS GetFeatureInfo. Mặc định là “false”.

- *FILTER [string]*

Tham số này cho phép định nghĩa điều kiện lọc dữ liệu. Đối với dữ liệu shapfiles hoặc các loại dữ liệu được kết nối thông qua OGR, điều kiện lọc dữ liệu đơn giản là một chuỗi điều kiện. Còn đối với các loại dữ liệu được trích xuất từ cơ sở dữ liệu thông qua chuỗi kết nối thì điều kiện lọc chính là chuỗi mệnh đề SQL WHERE. Ví dụ: FILTE "type='road' and size <2"

- *FILTERITEM [attribute]*

Trường dữ liệu dùng cho câu chuỗi lọc dữ liệu FILTER, chỉ dùng cho OGR và shapefiles.

- *MAXSCALE [double]*

Tỉ lệ lớn nhất mà layer được vẽ

- *METADATA*

Được dùng với OGC WMS để định nghĩa nhiều thứ chẳng hạn như tiêu đề của layer, hoặc được dùng để tạo ra các template linh động hơn. Những dữ liệu được đặt trong Metadata sẽ được truy xuất thông qua các tag của tập tin template.

Ví dụ:

METADATA

title "Lớp địa hình "

author "Trung tâm bản đồ Việt Nam"

create date "1/2/1996"

END

- *MINSCALE [double]*

Tỉ lệ nhỏ nhất mà layer được vẽ.

- *NAME [string]*

Tên ngắn cho layer. Giới hạn trong khoảng 20 kí tự. Tên layer được dùng để liên kết giữa giao diện web và mapfile. Ảnh bản đồ là kết quả của nhiều layer chồng lấp lên nhau. Từ giao diện web có thể cho phép người dùng chọn lựa layer hiển thị. Khi đó tên layer được chọn (trong source code) và tên layer được định nghĩa trong mapfile phải là một để có thể hiển thị được. Tên layer nên là duy nhất, trừ khi có một layer khác cùng tên nhưng khác tỉ lệ. Có thể sử dụng GROUP để nhóm các layer lại với nhau.

- *PROJECTION*

Bắt đầu đối tượng PROJECTION

- *STATUS [on|off|default]*

Đặt trạng thái của layer. Nếu giá trị là default, layer luôn được vẽ ra. Thông thường layer phụ có trạng thái là off, và để vẽ layer này ra ta có thể thêm vào chuỗi URL tên layer này.

Ví dụ:

Chuỗi URL : ...& layername="địa hình"& ...

- *TOLERANCEUNIT [double] và TOLERANCE [double]*

Đối với trường hợp cần lấy thông tin đối tượng trên bản đồ, ta cần chỉ ra đối tượng được chọn (thường là bằng cách click chuột). Tuy nhiên không phải khi nào cũng có thể click chính xác vị trí của đối tượng trên bản đồ. TOLERANCE được dùng để quy định phạm vi đối tượng thuộc về với tâm là vị trí click chuột. TOLERANCEUNIT chỉ ra đơn vị của TOLERANCE, mặc định TOLERANCEUNIT có giá trị là 3 pixel.

Ví dụ:

TOLERANCE 3

TOLERANCE 6

Khi người dùng click chuột lên bản đồ. MapServer xác định được vị trí click chuột là (X,Y), sau đó tìm kiếm trong cơ sở dữ liệu xác định có đối tượng nào trong có vị trí trong hình tròn tâm (X,Y), bán kính là 6\*3. Lưu ý là các giá trị đều được qui chuyển về cùng một hệ tọa độ.

- *TRANSPARENCY [integer|alpha]*

Đặt mức độ trong suốt của layer. Giá trị có thể là số nguyên (0 - 100) hoặc là giá trị hằng "ALPHA". Mặc dù tham số này tên là mang nghĩa là trong suốt, nhưng giá trị nguyên mới thật sự là giá trị độ mờ (sáng). Giá trị 0 là trong suốt hoàn toàn.

Giá trị hằng "ALPHA" được dùng khi ảnh bản đồ xuất ra dưới dạng RGB.

- *TRANSFORM [true|false]*

Báo cho MapServer chuyển từ hệ tọa độ địa lý sang hệ tọa độ đồ họa (ảnh đồ họa). Mặc định là True. Đối với hệ tọa độ đồ họa, gốc tọa độ luôn là điểm góc trái trên của ảnh khác với các hệ tọa độ địa lý (mô tả thế giới thực).

- *TYPE [point|line|polygon|circle|annotation|raster|query]*

Quy định các dữ liệu được vẽ ra. Không cần phải cùng loại với dữ liệu. Ví dụ : các đối tượng polygon có thể được vẽ như là một tập các điểm, ngược lại một điểm không thể vẽ như là tập các polygon. Việc chỉ ra TYPE là cần thiết bởi vì đôi khi một file dữ liệu vectơ (shapefile) không chỉ chứa đơn thuần một loại đối tượng, mà có thể nhiều đối tượng. Giả sử có đối tượng Point và Polygon, chọn kiểu TYPE là polygon thì các đối tượng kiểu Point sẽ không được vẽ.

Query chỉ ra rằng layer được truy vấn thông tin, không cần phải vẽ lại. Nếu giá trị là Circle thì 2 điểm (thường là một line) sẽ xác định hình chữ nhật chứa đường tròn.

### 2.3.3 Query Map Object

Định nghĩa cơ chế thực hiện câu lệnh truy vấn từ bản đồ.

- *COLOR [r] [g] [b]*

Khi xác định được đối tượng trên bản đồ, được chọn để truy vấn, MapServer sẽ vẽ lại đối tượng này với màu là Color. Mặc định là Yellow.

- *SIZE [x][y]*

Phạm vi thực hiện truy vấn, ngoài phạm vi này các đối tượng sẽ không được chọn để truy vấn dữ liệu. Mặc định là kích thước của cả bản đồ, được quy định trong đối tượng Map.

- *STATUS [on|off]*

Giá trị off thì sau khi thực hiện truy vấn, ảnh bản đồ sẽ không được vẽ lại.

- *STYLE [normal|hilite|selected]*

Qui định cách thức vẽ lại các đối tượng được chọn cho truy vấn, các đối tượng khác vẫn được vẽ lại như bình thường. Normal : vẽ lại các đối tượng này bình thường theo các giá trị thiết lập cho layer. Hilite : vẽ lại các đối tượng được chọn theo màu COLOR. Selected : chỉ vẽ lại các đối tượng được chọn, các đối tượng khác không được vẽ.

### 2.3.4 Projection Object

Để thiết lập phép chiếu cần xác định một phép chiếu chung cho đối tượng Map, và mỗi layer cũng cần chỉ ra một phép chiếu để vẽ các đối tượng trong layer đó. Đối tượng phép chiếu bao gồm tập các từ khoá của PROJ.4 (thư viện các phép chiếu được dùng hiện nay trên thế giới).

Ví dụ :

Phép chiếu UTM khu vực 15, NAD83:

PROJECTION

"proj=utm"

"ellps=GRS80"

"zone=15"

"north"

"no\_defs"

END

Và phép chiếu địa lý

PROJECTION

"proj=latlong"

END

## 2.4 Xử lý kết nối các loại dữ liệu

Ảnh bản đồ được tạo từ nhiều layer, mỗi layer có thể được vẽ từ các loại dữ liệu khác nhau lấy từ nhiều nguồn khác nhau. MapServer sử dụng các thư viện riêng cho mỗi loại kết nối, hơn nữa với mã nguồn mở MapServer cho phép biên dịch thêm vào các thư viện xử lý dữ liệu.

### 2.4.1 Kết nối dữ liệu mặc định ESRI Shapefiles

Đây là kiểu dữ liệu được hỗ trợ mặc định bởi MapServer, xem phần Tìm hiểu Mapfile mục Layer object để biết cách khai báo đối tượng Layer trong Mapfile.

### 2.4.2 Kết nối dữ liệu Raster

#### 2.4.2.1 Biên dịch thư viện GDAL trên môi trường Window

GDAL có thể được biên dịch trên môi trường Windows sử dụng MS V++6.x và MS Visual Studio.NET (C++) dưới dạng command line theo các bước sau đây :

- i. Chạy file VCVAR32.BAT từ dòng lệnh command line.VCVAR32.BAT là tập tin khởi tạo môi trường Microsoft Studio, đi kèm với trình biên dịch.

Ví dụ :

```
C:\Program Files\Microsoft Visual Studio\VC98\bin\VCVAR32.BAT ( VC++ 6.x)
C:\Program Files\Microsoft Visual Studio .NET 2003\VC7\bin\VCVAR32.BAT ( C++ .NET)
```

- ii. Khi môi trường cài đặt đã được thiết lập, chuyển đến thư mục chứa thư viện GDAL, và thực hiện dòng lệnh sau :  
C:\GDAL> nmake /f makefile.vc
- iii. Khi quá trình build hoàn thành, sử dụng makefile install các files cần thiết để cho thư viện GDAL. Cần thiết lập lại đường dẫn cho BINDIR, DATADIR trong

nmake.opt. Với BINDIR là thư mục chứa các file exe, dll tạo ra khi biên dịch với dòng lệnh dưới đây, DATADIR thư mục chứa các file cần thiết cho GDAL.

```
C:\GDAL> nmake /f makefile.vc install
```

### 2.4.2.2 Khai báo layer dùng dữ liệu Raster

Để sử dụng ảnh raster như một layer, cần phải khai báo một đối tượng layer trong Mapfile và xác định các tham số kèm theo. Đơn giản nhất chỉ cần khai báo ví dụ như sau:

Ví dụ:

```
LAYER
  NAME "Dia_hinh"
  DATA "DiahinhVN.tif"
  TYPE RASTER
  STATUS ON
END
```

Trong đó DATA chỉ ra đường dẫn file ảnh raster, có thể là đường dẫn tuyệt đối hoặc là tương đối theo SHAPEPATH khai báo trong đối tượng Map. Ngoài các tham số như trên một đối tượng layer dạng raster còn có thể có thêm thông tin PROJECTION, METADATA, PROCESSING, MINSCALE, và MAXSCALE.

### 2.4.2.3 Chọn lọc dữ liệu Raster

Dữ liệu Raster dưới dạng các pixel (điểm ảnh) do đó để lọc dữ liệu cần dựa vào giá trị của các pixel, hoặc dựa theo bảng màu giá trị của pixel theo bộ ba giá trị ([“red”], [“green”],[“blue”]). Khai báo chọn lọc dữ liệu raster như ví dụ sau :

Ví dụ :

```
LAYER
  NAME "Dia_hinh"
  DATA "DiahinhVn.tif"
  TYPE RASTER
  STATUS ON
  CLASSITEM "[pixel]"
  # class sử dụng câu điều kiện "0" tương đương với ([pixel]=0)
CLASS
  EXPRESSION "0"
  COLOR 0 0 0
```

```
END
# class sử dụng điều kiện lọc theo giá trị [pixel]
CLASS
    EXPRESSION ([pixel] >= 64 AND [pixel] < 128)
    COLOR 255 0 0
END

# sử dụng giá trị red/green/blue từ bảng màu
CLASS
    NAME "near white"
    EXPRESSION ([red] > 200 AND [green] > 200 AND [blue] > 200)
    COLOR 0 255 0
END

# Lọc các pixel có giá trị tận cùng là 1
CLASS
    EXPRESSION /*1/
    COLOR 0 0 255
END
END
```

Việc chọn lọc dữ liệu thường được tiến hành như sau : đọc và phân loại trước các giá trị pixel vào một bảng gọi là bảng tìm kiếm, khi đó các pixel có giá trị lân cận được xếp gần nhau. Cuối cùng là khi vẽ sẽ đọc được từng nhóm pixel từ bảng tìm kiếm dựa vào các chọn lọc dữ liệu trong định nghĩa các class của đối tượng layer. Cách làm việc như thế này nhanh hơn so với việc đọc và so sánh giá trị từng pixel một của ảnh raster thường lên đến hàng triệu pixel.

Tuy nhiên việc lọc dữ liệu pixel chỉ có thể tiến hành tốt trên các loại raster dạng 8bit (giá trị pixel thuộc 0-255) còn đối với các ảnh raster dạng khác như 16 bit (0- 65535) cần định nghĩa thêm giá trị cho tham số PROCESSING để quy định phạm vi chọn lọc các pixel.

Ví dụ:

```
LAYER
    NAME grid1
    TYPE raster
    STATUS default
```



```
DATA data/float.tif
PROCESSING "SCALE=10,20"
PROCESSING "SCALE_BUCKETS=4"
CLASS
  NAME "red"
  EXPRESSION ([pixel] < -3)
  COLOR 255 0 0
END
CLASS
  NAME "green"
  EXPRESSION ([pixel] >= -3 and [pixel] < 3)
  COLOR 0 255 0
END
CLASS
  NAME "blue"
  EXPRESSION ([pixel] >= 3)
  COLOR 0 0 255
END
END
```

Trong đó PROCESSING “SCALE=10,20” cho chỉ ra khoảng giá trị của những pixel được xét (10 -20). Và PROCESSING “SCALE\_BUCKETS=4” nghĩa là sẽ lấy 4 pixel có giá trị thỏa bất đẳng thức sau :

{

### 2.4.3 Kết nối dữ liệu dùng thư viện OGR

Ban đầu các nhà phát triển dự tính xây dựng thư viện "OpenGIS Simple Features Reference Implementation" và thế là tên OGR xuất hiện nhưng về sau OGR không đi theo hướng phát triển ban đầu nữa mà chuyển thành một thư viện các hàm C++ có thể được gọi độc lập, hoặc có thể giao tiếp thông qua giao tiếp OLEDB để đọc ghi các loại dữ

liệu vector. Lúc này tên OGR vẫn được giữ vì tiền tố OGR xuất hiện trong hầu hết các tên lớp, tên file.

OGR là một thư viện C++ mã nguồn mở cung cấp khả năng đọc/ghi các loại dữ liệu vector khác nhau, trong đó bao gồm cả ESRI shapefile và MapInfo mid/mif và định dạng TAB.

### 2.4.3.1 Các định dạng dữ liệu được hỗ trợ

Vào địa chỉ [http://ogr.maptools.org/ogr\\_formats.html](http://ogr.maptools.org/ogr_formats.html) để xem danh sách cập nhật các định dạng file vector được hỗ trợ, hiện nay vào thời điểm báo cáo này được viết, các định dạng sau đây được hỗ trợ:

ArcInfo Coverages, ESRI Shapefiles, FMEObjects Gateway, IHO S-57 datasets, MapInfo TAB and MIF/MID files, Microstation DGN files, OGD I Vectors, Oracle Spatial, PostgreSQL, SDTS TVP (Topological Vector Profile and Point Profile datasets), TIGER/Line file sets, UK.NTF (National Transfer Format).

### 2.4.3.2 Tích hợp OGR vào MapServer

OGR được tích hợp thẳng vào source code của MapServer, do đó để thực hiện kết nối một layer theo sử dụng OGR chỉ cần thay đổi và khai báo layer trong mapfile. Lúc này tham số DATA của đối tượng LAYER được thay bằng 3 tham số CONNECTIONTYPE OGR, CONNECTION và DATA.

Quy tắc cho 3 tham số này tùy thuộc loại dữ liệu được dùng. Với OGR, nguồn dữ liệu có thể là tập các file có cùng tên ( .shp/.shx/.dbf của ArcView Shapefiles hoặc là tab/.map/.ind/.id của MapInfo TAB files) hoặc cả thư mục (TIGER). Đối với nguồn dữ liệu là tập các file thì chỉ cần chỉ ra tên một file trong tập hợp đó. Các nguồn dữ liệu được kết nối dùng thư viện OGR đôi khi cũng được gọi là nguồn dữ liệu OGR.

### 2.4.3.3 Khai báo layer kết nối OGR trong Map file

```
LAYER
...
CONNECTIONTYPE OGR
CONNECTION "<datasource_name>"
    DATA "<layer_definition>"
...
END
```

Trong đó:

i. *<datasource\_name>* là đường dẫn của nguồn dữ liệu cần đọc. Nếu nguồn dữ liệu dạng tập hợp các file cùng tên thì *datasource\_name* là đường dẫn của một file trong tập hợp đó. Đường dẫn này có thể là tuyệt đối hoặc tương đối theo SHAPEPATH hoặc theo mapfile. Đối với các nguồn dữ liệu được lấy từ một CSDL thì *datasource\_name* bao gồm cả user và password, tên bảng dữ liệu.

Ví dụ :

Kết nối đến Oracle Spatial : “OCI: nick/password@mypass”.

ii. *<layer\_definition>* tên, số lượng hoặc các định nghĩa SQL của layer dùng cho việc đọc nguồn dữ liệu.

- + *Layer Name* : (phân biệt hoa thường) tên layer dùng để chọn layer.
- + *Layer Number* : số thứ tự layer dùng để chọn một layer (bắt đầu từ 0 cho layer đầu tiên). Thường thì layer name sử dụng tiện hơn so với layer number.
- + *Omitted*: nếu không có từ khóa DATA, hàm ý chọn layer 0.
- + *SQL SELECT* : Câu SQL SELECT được dùng, trong trường hợp dữ liệu được lưu trữ trong một CSDL gồm nhiều bảng. Khi đó từ câu SQL này dữ liệu được chọn ra để phát sinh một layer tạm thời sử dụng.

### 2.4.3.4 Ví dụ minh họa

#### MapInfo TAB file

```
LAYER
  NAME "Builtup_Areas_tab"
  TYPE POLYGON
  CONNECTIONTYPE OGR
  CONNECTION "data/tab/092b06_builtup_a.tab"
  STATUS ON
  CLASS
  ...
END

...
END
```

Microstation DGN file sử dụng <layer\_index>

```
LAYER
  NAME "Water_dgn"
  TYPE POLYGON
  CONNECTIONTYPE OGR
  CONNECTION "data/dgn/wat.dgn"
  DATA "0"
  STATUS ON
  CLASS
  ...
  END
...
END
```

TIGER/Line file using <layer\_name>

```
LAYER
  NAME "Roads_tig"
  TYPE line
  CONNECTIONTYPE OGR
  CONNECTION "../data/tiger_michigan"
  DATA "CompleteChain"
  STATUS ON
  CLASS
  ...
  END
END
```

Directory of Shapefiles sử dụng SQL JOIN

```
LAYER
  NAME "Parks_cov"
  TYPE POLYGON
  CONNECTIONTYPE OGR
  CONNECTION "data/shppoly"
  DATA "SELECT eas_id, idlink.Name FROM poly LEFT JOIN idlink ON
poly.eas_id = idlink.eas_id"
  STATUS ON
  CLASSITEM "idlink.Name"
  CLASS
  NAME "At Risk"
  COLOR 0 0 255
```

```
        OUTLINECOLOR 255 0 0
        EXPRESSION "_158_"
    END
    CLASS
        NAME "Normal"
        COLOR 0 255 0
        OUTLINECOLOR 255 0 0
    END
    ...
END
```

### 2.4.4 Kết nối dữ liệu dùng WMS

Thực hiện kết nối dữ liệu theo chuẩn WMS, MapServer khi này đóng vai trò là một client của Web Map Server. Do là client nên MapServer chỉ cần khai báo layer lấy dữ liệu từ WMS là đủ.

#### 2.4.4.1 Lưu trữ file tạm

Bạn cần phải đặt giá trị cho tham số IMAGEPATH trong đối tượng WEB của mapfile chỉ đến một thư mục hợp lệ và có thể ghi được. MapServer sẽ dùng thư mục này để lưu trữ các file tạm được download từ server ở xa khác. Những file tạm này sẽ được tự động xóa đi bởi MapServer.

Khai báo IMAGEPATH

```
MAP
    ...
    WEB
        IMAGEPATH "/tmp/ms_tmp/"
        IMAGEURL ...
    END
    ...
END
```

Để giữ lại các file tạm cho mục đích debug, khai báo :

```
LAYER
    ....
    DEBUG ON
    ...
END
```

### 2.4.4.2 Khai báo layer dùng WMS

Layer dùng dữ liệu từ WMS được gọi là WMS layer. WMS layer được truy xuất thông qua kết nối kiểu WMS.

```
LAYER
  NAME "prov_bound"
  TYPE RASTER
  STATUS ON
  CONNECTION "http://www2.dmsolutions.ca/cgi-bin/mswms_gmap?"
  CONNECTIONTYPE WMS
  METADATA
    "wms_srs" "EPSG:42304 EPSG:42101 EPSG:4269 EPSG:4326
  EPSG:42304"
    "wms_name" "prov_bound"
    "wms_server_version" "1.1.0"
    "wms_formatlist" "image/gif,image/png,image/jpeg,image/wbmp"
    "wms_format" "image/gif"
  END
END
```

Các tham số sau đây là bắt buộc đối với kiểu kết nối WMS.

- + CONNECTIONTYPE WMS
- + Tham số CONNECTION : chuỗi URL xác định địa chỉ của Web Map Server.
- + “wms\_onlineresource” (meta data): được dùng khi tham số CONNECTION không có giá trị.
- + “wms\_srs” (metadata): danh sách các mã phép chiếu EPSG được hỗ trợ bởi Web Map Server. Thông tin này có được từ câu truy vấn GetCapabilities (xem phần Các chuẩn WebGIS).
- + “wms\_name” (metadata): danh sách các layer được lấy từ WMS server, giá trị được dùng để đặt LAYERS và tham số QUERY\_LAYERS WMS URL
- + “wms\_server\_version” : phiên bản của giao thức WMS được hỗ trợ bởi WMS server, và WMS server sẽ dùng cho GetMap.
- + "wms\_format" metadata: Định dạng ảnh dùng để thực hiện yêu cầu GetMap
- + "wms\_formatlist" : Danh sách các định dạng ảnh sử dụng cho GetMap

Chú ý: Nếu cả “wms\_format” và “wms\_formatlist” đều được khai báo trong mapfile, thì wms\_format sẽ được dùng.

### 2.4.5 Kết nối dữ liệu dùng WFS

#### 2.4.5.1 Biên dịch các thư viện hỗ trợ WFS

Để MapServer hỗ trợ WFS, cần biên dịch MapServer với các thư viện sau:

PROJ.4 : thư viện các hệ quy chiếu, phiên bản 4.4.3 trở lên.

GDAL/OGR: thư viện hỗ trợ đọc ghi các dữ liệu vectơ, phiên bản từ 1.1.8 trở lên.

LibCURL: Dùng để giúp MapServer đóng vai trò HTTP client, phiên bản 7.10 trở lên.

#### 2.4.5.2 Lưu trữ file tạm

Cần chỉ ra thư mục trong IMAGEPATH trong mapfile bởi vì MapServer sử dụng thư mục này để lưu trữ file tạm download từ WFS server.

```
MAP
...
WEB
    IMAGEPATH "/tmp/ms_tmp/"
    IMAGEURL ...
END
...
END
```

#### 2.4.5.3 Khai báo layer dùng WFS

Layer dùng cho WFS được khai báo như với các tham số chỉ rõ kết nối tới WFS server sử dụng tham số CONNECTION và tham số METADATA. Các tham số bắt buộc như sau:

- + CONNECTIONTYPE WFS
- + CONNECTION : chuỗi URL đến WFS server.

Ví dụ : [http://www2.dmsolutions.ca/cgi-bin/mswfs\\_gmap?](http://www2.dmsolutions.ca/cgi-bin/mswfs_gmap?)

LAYER phải chứa đối tượng METADATA với các tham số sau :

- + wfs\_connectiontimeout (không bắt buộc) : Thời gian tối đa đợi để WFS trả lời (mặc định là 30 giây)..

- + wfs\_service (bắt buộc) : là WFS
- + wfs\_typename (bắt buộc) : tên layer. Tên layer được xác định khi các client gửi GetCapabilities request lên WFS server
- + wfs\_version (bắt buộc) : phiên bản của WFS, hiện nay là 1.0.0
- + wfs\_request\_method (không bắt buộc) : để thực hiện GET request cần đặt wfs\_request\_method là “GET”

```
...  
METADATA
```

```
  "wfs_filter"    "GET"
```

```
END  
...
```

- + wfs\_filter (không bắt buộc) : chuỗi định nghĩa để lọc dữ liệu trong yêu cầu GetFeature của client đối với WFS server.

```
...
```

```
METADATA
```

```
  "wfs_filter"
```

```
  "<PropertyIsGreaterThan><PropertyName>POP_RANGE</PropertyName><Literal>4
```

```
  </Literal></PropertyIsGreaterThan>"
```

```
END  
...
```

- Ví dụ về layer sử dụng WFS

### LAYER

```
NAME park
```

```
TYPE POLYGON
```

```
STATUS ON
```

```
CONNECTIONTYPE WFS
```

```
CONNECTION "http://www2.dmsolutions.ca/cgi-bin/mswfs_gmap?"
```

```
METADATA
```

```
  "wfs_service"    "WFS"
```

```
  "wfs_typename"  "park"
```

```
  "wfs_version"   "1.0.0"
```

```
  "wfs_request_method" "GET"
```

```
  "wfs_connectiontimeout" "60"
```

```
  "wfs_maxfeatures" "1"
```

```
END
```



```
PROJECTION
  "init=epsg:42304"
END
CLASS
  NAME "Parks"
  COLOR 200 255 0
  OUTLINECOLOR 120 120 120
END
END # Layer
```

Khoa CNTT - ĐHKHTN TP.HCM

Khoa CNTT - ĐHKHTN TP.HCM

## **Chương 3 : Bài toán dự báo thời tiết – Biểu diễn trên bản đồ**

### **3.1 Khảo sát hiện trạng**

#### **3.1.1 Giới thiệu bài toán dự báo**

Ngày nay các Web Site ngày càng nhiều, càng phổ biến, do người dân ngày tiếp cận với tin học, càng chú trọng vào việc tìm kiếm thông tin trên mạng. Muốn nắm bắt thông tin một cách nhanh chóng, tiện lợi nhằm phục vụ cho việc kinh doanh và sinh hoạt trong đời sống hằng ngày.

Tình hình thời tiết trên thế giới và trong nước hiện nay vẫn luôn phức tạp, luôn biến động một cách khó lường, những thảm họa xảy ra một cách bất ngờ để lại những hậu quả nặng nề và thảm khốc. Để cung cấp thông tin thời tiết cho mọi người đã có các phương tiện thông tin đại chúng : phát thanh, truyền hình, bên cạnh đó còn có một số web site ... Tuy nhiên các Web Site về dự báo thời tiết của nước ta nói chung và khu vực Nam Bộ nói riêng là chưa đáp ứng được yêu cầu về cung cấp thông tin. Trên tư tưởng đó chúng em muốn xây dựng một WebSite sử dụng công nghệ bản đồ WebGIS nhằm làm cho trang Web thêm phần sinh động, cung cấp một số chức năng cần thiết cho người dùng.

#### **3.1.2 Số liệu thời tiết trong dự báo**

##### **\*Các loại số liệu**

Dữ liệu gồm 3 dạng :

- a. Số liệu dự báo
- b. Số liệu tức thời
- c. Số liệu lưu trữ

Trong đó số liệu lưu trữ là không cần đáp ứng thời gian thực. Nghĩa là khi có cá nhân hay tổ chức nào có nhu cầu sử dụng số liệu lưu trữ (quá khứ) thì có thể sẽ nhận được kết quả sau một khoảng thời gian nào đó (vài giờ hay vài ngày.)

##### **\*Hình thức lưu trữ:**

Điện tử (lưu vào máy tính )và giấy tờ (có bộ phận quản lý giấy tờ số liệu). Hai hình thức lưu trữ:điện tử tiện sử dụng, tham khảo nhanh. Giấy tờ tiện cho đối chiếu sau này nếu có nghi ngờ về độ chính xác của số liệu.

**\* Quyền hạn trên số liệu**

- Hiện nay chưa có cơ chế phân quyền trên số liệu. Mọi người trong Đài đều có quyền khai thác sử dụng và loại bỏ số liệu (dạng điện tử).

- Về nguyên tắc không được tự ý chuyển giao số liệu ra bên ngoài (các hệ thống, cơ quan, cá nhân ngoài ngành) đặc biệt là đối với một số khu vực nhạy cảm. Cần phải được sự cho phép của lãnh đạo Đài.

**\*Trao đổi số liệu và dự báo**

- Các trạm “vệ tinh” gửi số liệu về Đài Khí tượng Khu vực (1)

- Đài Khí tượng Khu vực gửi số liệu của Khu vực ra bên ngoài, tức là gửi về trung tâm tư liệu KTTV (2)và nhận số liệu về các khu vực quốc tế (3).

- Bắt đầu trao đổi từ 1h sáng trong ngày, và cách 3 tiếng thì lặp lại cho đến hết ngày.

- Số liệu từ các trạm được gửi về theo hệ thống mạng của ngành và được lưu trữ vào một máy tính chung trong mạng. Các máy khác truy cập vào máy chung này để khai thác dữ liệu.

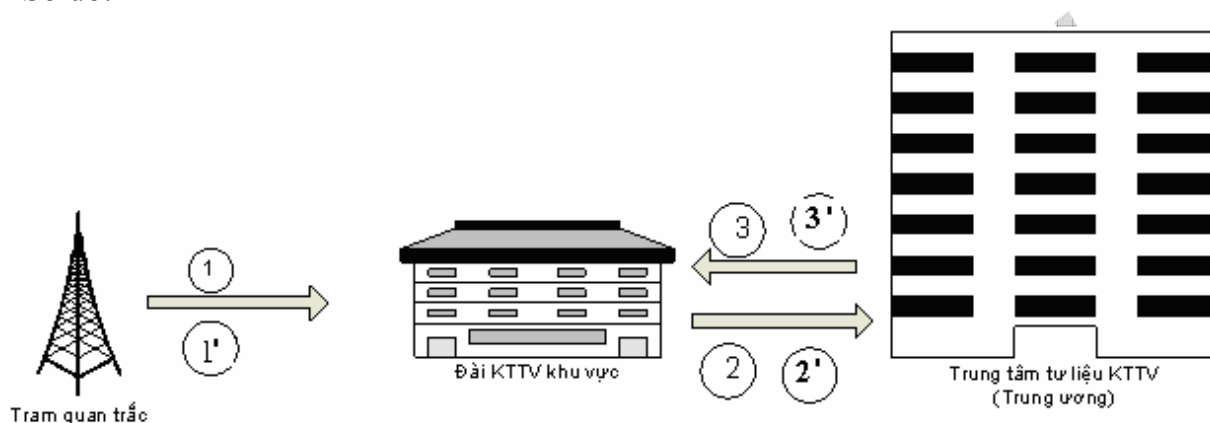
- Số liệu lần lượt đổ về và cứ theo thời gian, số liệu lần sau “ghi đè” lên số liệu lần trước. Những số liệu được nhận được lúc này gọi là “Số liệu tức thời”. Nếu nhận xét thấy số liệu có gì sai sót thì có thể loại bỏ số liệu này (không chỉnh sửa).

Những số liệu tức thời này được sử dụng vào các mô hình dự báo→tạo ra số liệu dự báo.

- Số liệu dự báo được cập nhật cứ 12h 1 lần.

- Đến cuối tháng, các trạm con mới gửi 1 bản (dạng giấy) ghi số liệu, hoặc biểu đồ về Đài khu vực(1’). Sau khi nhận được và nhập liệu vào máy. Đài sẽ tiến hành kiểm định các số liệu này. Sau đó gửi kết quả kiểm định, file số liệu, văn bản do các trạm gửi ra trung tâm tại Hà Nội để duyệt(2’). Nếu được duyệt (3’) số liệu lúc này trở thành số liệu lưu trữ.

Sơ đồ:

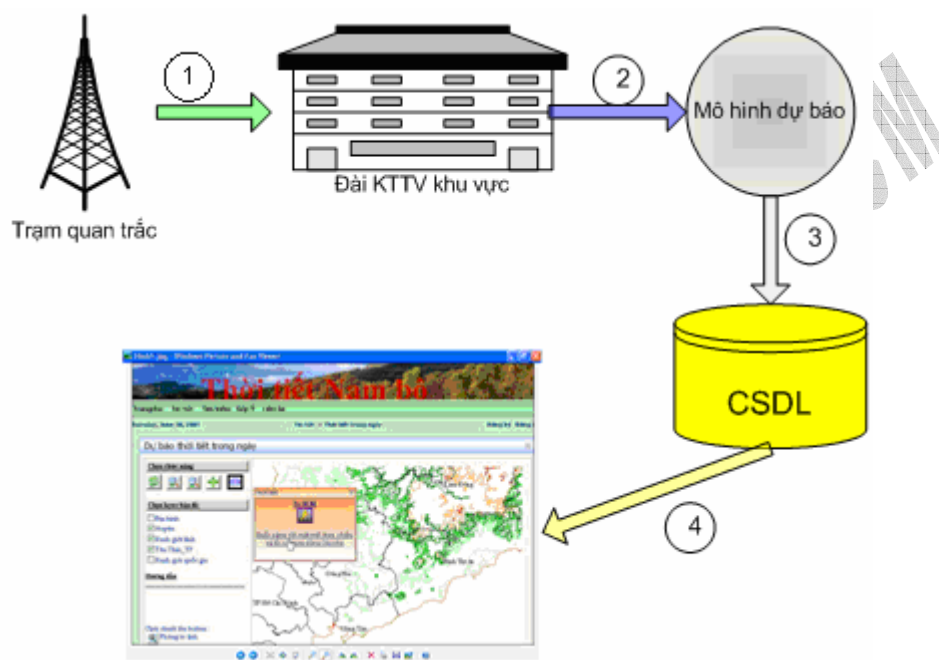


Hình 3-1 Trao đổi số liệu thời tiết

### 3.1.3 Hệ thống KTTV khu vực Nam bộ

- Quy trình hoạt động : (xem thêm phần Trao đổi dữ liệu và dự báo)

- (1) Trạm quan trắc đo đạc và gửi số liệu về Đài KTTV khu vực Nam bộ. Số liệu này được gọi là số liệu thô.
- (2) Sau một khoảng thời gian nhất định, số liệu thô sẽ trở thành dữ liệu đầu vào cho các Mô hình dự báo.
- (3) Kết quả của Mô hình dự báo sẽ là dữ liệu dự báo được chuyển vào CSDL.
- (4) Dữ liệu dự báo sẽ được truy vấn và đưa ra trang web theo yêu cầu.



Hình 3-2 Hoạt động hệ thống dự báo

- Thông tin dự báo

- Thông tin chung: nêu lên dự báo cho vùng hay khu vực gồm : khoảng trung bình nhiệt độ, kết luận nắng hay mưa, sức gió..

- Thông tin chi tiết : cho vùng hay khu vực

- + Nhiệt độ
- + Lượng mưa
- + Sức gió
- + Độ ẩm
- + Áp suất

...

- Các loại dự báo

Số liệu được cập nhật (trạm gửi về đài) theo khoảng 12h/1 lần, nếu có thời tiết đặc biệt thời gian sẽ ngắn lại. Từ số liệu thô, tùy theo mô hình dự báo sử dụng mà ta có hai loại dự báo như sau :

- Dự báo theo ngày
- Dự báo tương lai: theo 3 ngày, hoặc 10 ngày

### 3.2 Phân tích và xác định yêu cầu

Các yêu cầu của ứng dụng Website dự báo thời tiết gắn liền với nhu cầu một người dùng khi sử dụng Website để xem thông tin thời tiết.

- Tình hình thời tiết khu vực trong ngày như thế nào ?
- Thời tiết của tỉnh-thành phố trong ngày như thế nào ?
- Thời tiết một tỉnh-thành phố vào một ngày nào đó như thế nào ?

...

Các chức năng trên trang Web được chia làm 2 nhóm chức năng

- Các chức năng trên bản đồ :

STT	Chức năng	Giải thích
1	Phóng to, thu nhỏ bản đồ	
2	Di chuyển bản đồ	
3	Truy vấn thông tin trên bản đồ	Lấy thông tin của một tỉnh-thành phố khi click chuột vào vùng không gian bản đồ của tỉnh-thành phố đó.
4	Tìm kiếm vị trí của một Tỉnh _TP trên bản đồ	Dịch chuyển tới khu vực chứa tỉnh-thành phố này

Bảng 3-1 Các chức năng trên bản đồ

- Các chức năng dự báo thời tiết:

STT	Chức năng	Giải thích
1	Xem thời tiết trong ngày	Sử dụng các chức năng trên bản đồ để truy xuất thông tin thời tiết
2	Xem thời tiết vài ngày tới	Sẽ cho phép truy xuất thông tin trong ngày và cả quá khứ.
3	Tra cứu thông tin thời tiết	
4	Gửi bài về trạm	Chỉ những User đã đăng ký mới được gửi bài.
5	Góp ý kiến về Trạm	Người góp ý phải có địa chỉ email để liên lạc
6	Liên lạc với Admin	

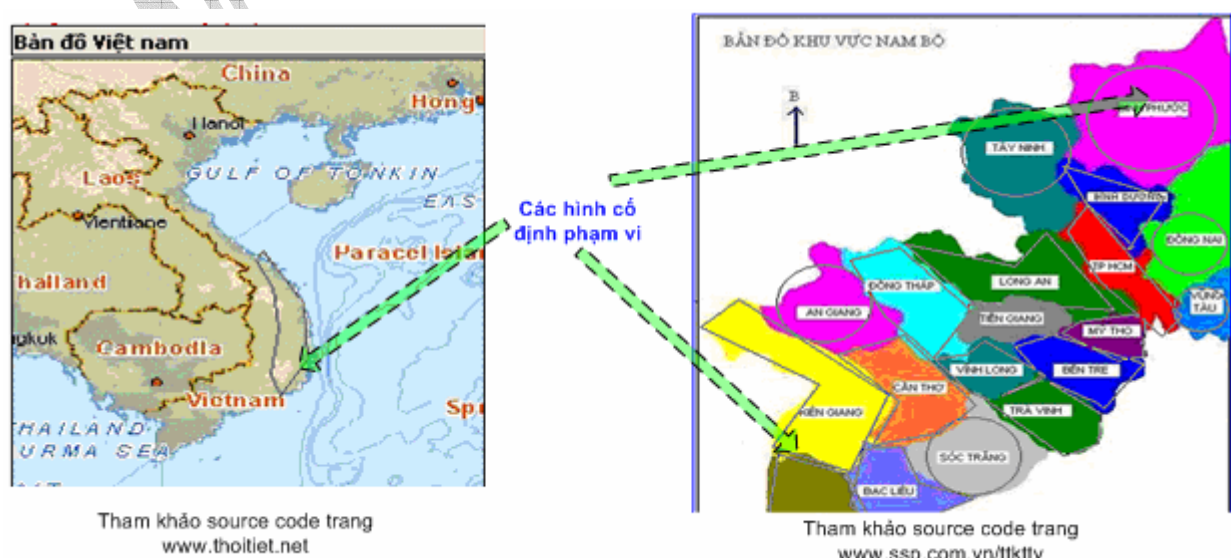
Bảng 3-2 Các chức năng dự báo thời tiết

### 3.3 Các vấn đề trong bài toán dự báo thời tiết bằng bản đồ

#### 3.3.1 Thể hiện bản đồ

Việc dùng bản đồ để minh họa trong dự báo thời tiết được nhiều website dùng đến như [www.ssp.com.vn/ttkttv](http://www.ssp.com.vn/ttkttv). (Trung tâm Khí tượng thủy văn khu vực Nam bộ ), [www.thoitiet.net](http://www.thoitiet.net) (Trung tâm InteCom,chuyên kinh doanh các giải pháp dự báo thời tiết), [www.nchmf.gov.vn](http://www.nchmf.gov.vn), [www.hymettdata.com](http://www.hymettdata.com) (Trung tâm Tư liệu Khí tượng Thủy văn).

Các trang web chủ yếu sử dụng ảnh bản đồ tĩnh, nhiều màu sắc nhưng không thay đổi hình dạng, không cung cấp được các chức năng cơ bản với thao tác bản đồ (phóng to, thu nhỏ, dịch chuyển...). Trên các trang web này đều hỗ trợ chức năng xác định thông tin thời tiết theo điểm click chuột. Giải pháp của các trang sử dụng ảnh bản đồ tĩnh là vẽ lên ảnh bản đồ các hình cố định phạm vi theo từng khu vực,sử dụng JavaScript để xác định tọa độ chuột. Nếu tọa độ chuột thuộc phạm vi hình nào thì sẽ suy ra được khu vực và thực hiện kết nối cơ sở dữ liệu để lấy thông tin dự báo



Hình 3-3 Xác định phạm vi bằng hình cố định

Do sử dụng ảnh tĩnh, và giải quyết việc xác định phạm vi khu vực bằng cách vẽ các hình cố định phạm vi nên bản đồ không thể phóng to, hay thu nhỏ, hay dịch chuyển được.

Do đó chúng em sẽ sử dụng công nghệ WebGIS vào việc thể hiện bản đồ động, nhằm giải quyết những khó khăn mà cách sử dụng bản đồ dạng tĩnh gặp phải. Cụ thể là sử dụng MapServer, một WebGIS application mã nguồn mở hiện nay.

### 3.3.2 Chỉnh sửa dữ liệu vector MapInfo

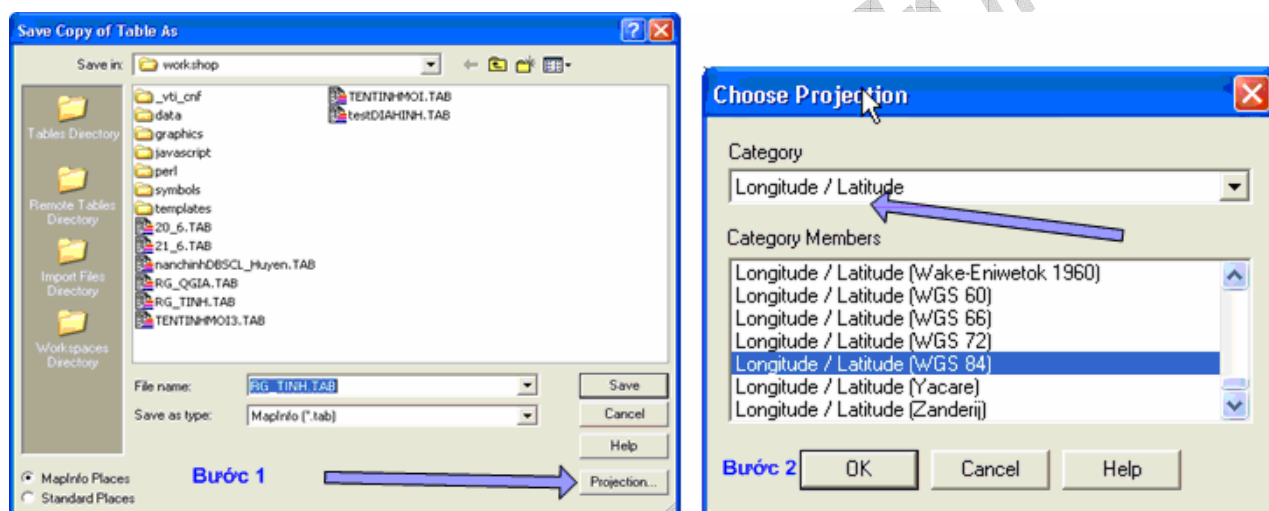
#### 3.3.2.1 Dữ liệu MapInfo

Đây là định dạng phổ biến được dùng trong các cơ quan, tổ chức sử dụng GIS tại Việt Nam. Bản đồ tạo từ dữ liệu MapInfo cũng gồm nhiều layer, mỗi layer thường thể hiện một loại đối tượng ví dụ : địa hình, sông ngòi... Thông tin để tạo ra layer được lưu trữ thành nhóm bốn file (hoặc ba file) cùng tên, ví dụ layer địa hình sẽ được vẽ ra từ việc đọc bốn file là Diahinh.TAB, Diahinh.DAT, Diahinh.IND, Diahinh.ID. Mỗi layer đều có một hệ tọa độ xác định.

#### 3.3.2.2 Chuyển đổi hệ tọa độ

Trong MapInfo, ta có thể mở nhiều layer. Khi layer thứ hai được mở, MapInfo sẽ chuyển đổi hệ tọa độ của layer thứ hai thành hệ tọa độ của layer thứ nhất, lúc đó hai layer cùng tọa độ tạo thành ảnh bản đồ hợp lý. Tuy nhiên khi sử dụng MapServer chỉ có thể xác định duy nhất một hệ tọa độ cho đối tượng ảnh bản đồ (đối tượng Map trong mapfile). Do đó cần chuyển tất cả các layer về cùng một hệ tọa độ.

Cách 1 : Sử dụng MapInfo, mở file TAB tương ứng với layer cần chuyển. Chọn Save Copy As, chọn Projection, chọn Category và CategoryMember

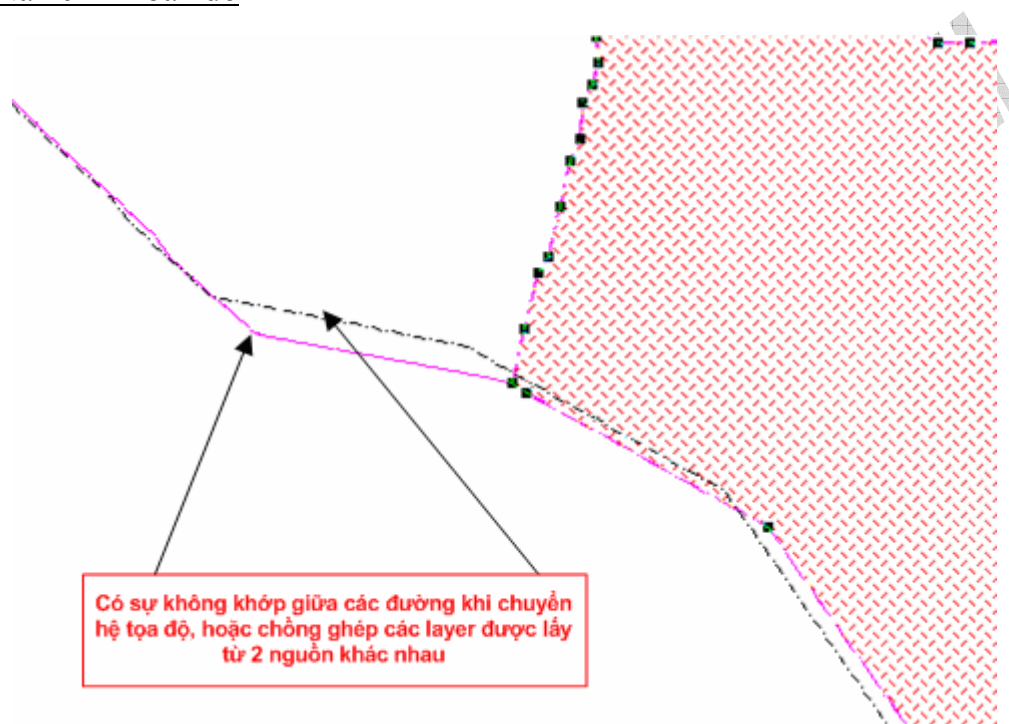


Hình 3-4 Chuyển đổi hệ tọa độ bằng MapInfo

Cách 2: Sử dụng phần mềm DolGIS 6.0 của công ty DolSoft [www.dolsoft.com.vn](http://www.dolsoft.com.vn)



3.3.2.3 Nắn chỉnh bản đồ

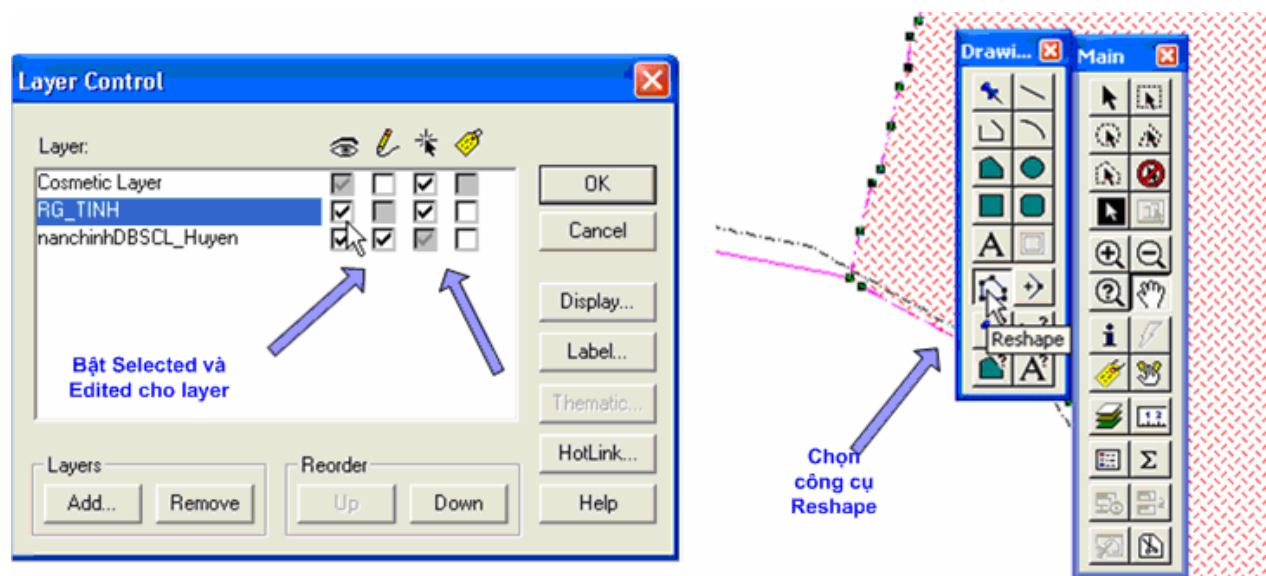


Hình 3-5 Hiện tượng không khớp khi hiệu chỉnh bản đồ

Sau khi chuyển các layer về cùng tọa độ thì có thể sự không khớp giữa các đối tượng điểm, đường, đa giác giữa các layer khi tiến hành chồng lấp các layer. Cần phải làm khớp hay còn gọi là nắn chỉnh bản đồ.



Đặc biệt do dữ liệu được lấy từ hai nguồn khác nhau : Trung tâm Khí tượng Thủy văn (layer địa hình, layer ranh giới các tỉnh, ranh giới quốc gia...), Trung tâm Thủy lợi khu vực Nam bộ (layer các tỉnh Nam bộ, layer các sông...). Nên khi tiến hành chồng lấp bản đồ hiện tượng không khớp xảy ra rất nhiều, tốn nhiều thời gian chỉ để nắn chỉnh bản đồ. Nếu để hiện tượng không khớp giữa các đối tượng thì ảnh bản đồ do chồng lấp các layer sẽ không logic, mất thẩm mỹ đặc biệt trong trường hợp phóng to bản đồ.


Cách thực hiện




Hình 3-6 Nấn chỉnh bản đồ dùng MapInfo

- Cho phép chọn các đối tượng trên layer (Selected) và cho phép chỉnh sửa các đối tượng trên layer (Edited).

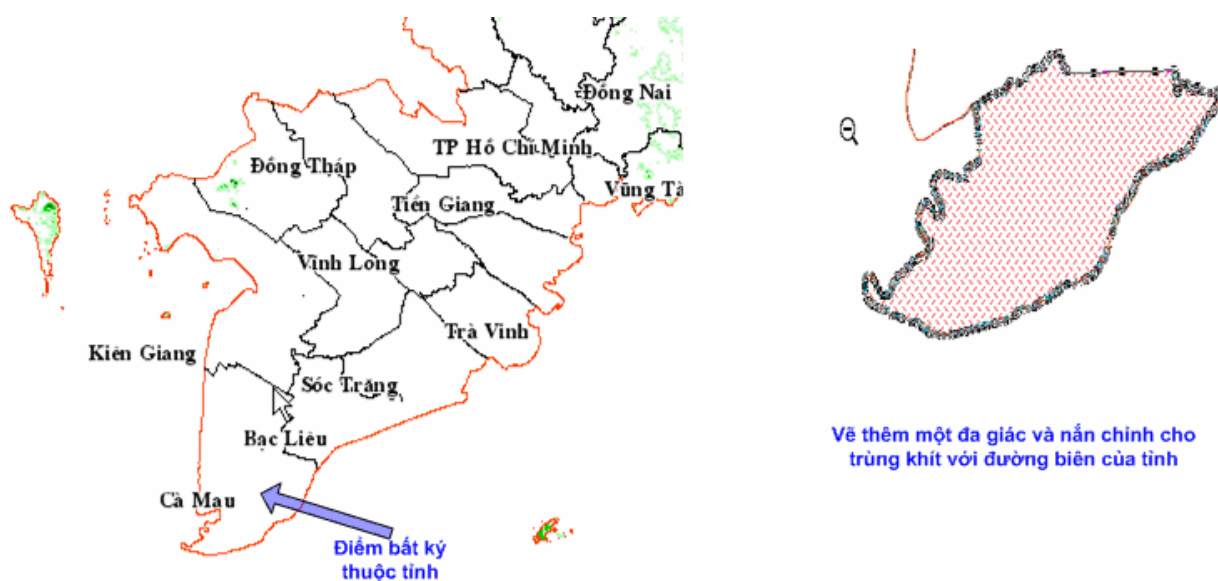
Chọn Layer Control trên thanh Main toolbar hoặc từ Menu, check vào biểu tượng  và .

- Chọn đối tượng cần chỉnh sửa, chọn công cụ Reshape  từ thanh công cụ Drawing

- Chỉnh sửa các đối tượng cho khớp lại, nếu cần có thể thêm các điểm neo bằng công cụ Add Node .

#### 3.3.2.4 Thêm các vùng chọn đối tượng

Với GIS chỉ có thể xác định được đối tượng khi mà người dùng click chuột chính xác vào đối tượng. Trong bài toán dự báo biểu diễn bằng bản đồ, để có tính tiện dụng cao, người dùng có thể chọn bất cứ điểm nào thuộc một tỉnh hay khu vực để nhận được thông tin dự báo về tỉnh hay khu vực đó.



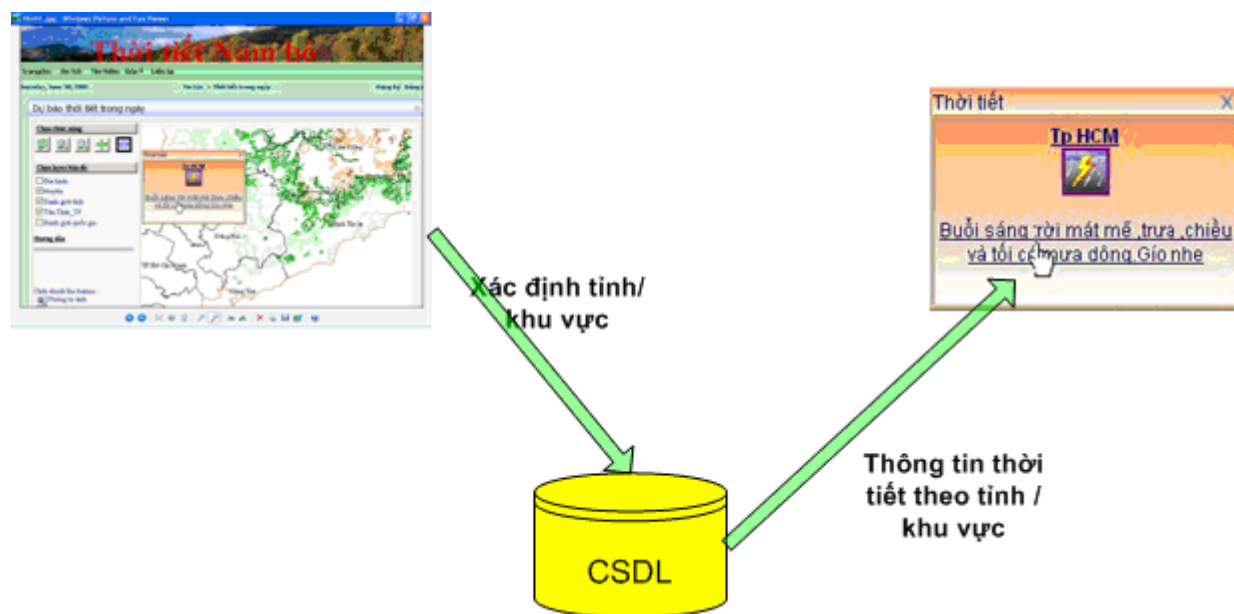
Hình 3-7 Thêm vùng chọn cho đối tượng tỉnh - tpố

Giải pháp là tạo ra một layer gồm các hình đa giác để xác định phạm vi của tỉnh, layer này được chọn tô và màu vẽ trùng với màu nền ảnh bản đồ nên sẽ không thấy được. Công việc này tiến hành cho toàn bộ các tỉnh khu vực Nam bộ

### 3.3.3 Truy vấn dữ liệu với bản đồ

Thông tin thuộc tính mô tả về đối tượng trên bản đồ (thường liên quan đến các yếu tố địa lý) không đủ để giải quyết bài toán dự báo. Hơn nữa, do yêu cầu dự báo thời tiết, số lượng lưu trữ nhiều, thay đổi liên tục. Do đó, dùng một hệ quản trị CSDL là phương án được chọn lựa.

Như vậy bài toán truy vấn dữ liệu với bản đồ được đưa về:



Hình 3-8 Bài toán truy vấn dữ liệu

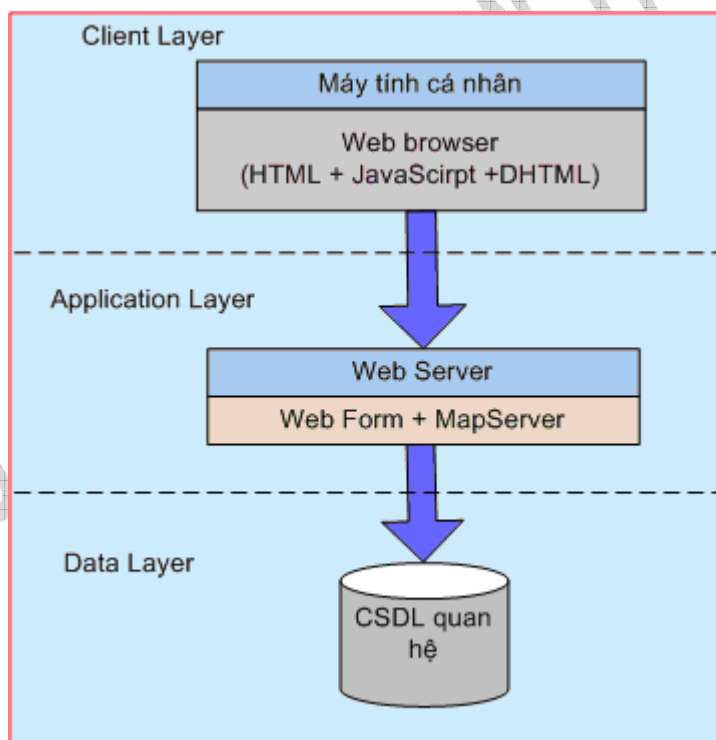
Mô tả :

- Người dùng click chuột trên bản đồ, ta sẽ xác định được tỉnh / khu vực người dùng muốn có thông tin (theo mã tỉnh khu vực ID).
- Truy vấn CSDL (lưu trữ dạng bảng) theo ID xác định.
- Hiện thị kết quả truy vấn CSDL.

## Chương 4 : Xây dựng Website Thời tiết Nam bộ

### 4.1 Thiết kế kiến trúc hệ thống

Hệ thống được thiết kế với 3 tầng khác nhau như sau :



Hình 4-1 Kiến trúc hệ thống

- **Tầng client :** được xây dựng bằng Javascript, HTML và DHTML, thực hiện nhiệm vụ xử lý các thao tác, lưu trữ thông tin ứng với từng người sử dụng, đảm nhận vai trò trung gian, truyền nhận dữ liệu, giữa người sử dụng với web server.
- **Tầng ứng dụng :** chia làm 2 thành phần : WebForm và MapServer. Cả 2 đều được phát triển dựa trên công nghệ .NET (ASP.NET và VB.NET)
  - WebForm : đảm nhận trách nhiệm phát sinh giao diện và các đoạn script để tương tác với client, đóng vai trò trung gian giữa client và MapServer, nó sẽ gửi yêu cầu của client tới MapServer và nhận dữ liệu trả về để gửi ngược cho client.
  - MapServer : xử lý các thao tác về phát sinh bản đồ, phóng to, thu nhỏ, dịch chuyển, tra cứu thông tin trên bản đồ. Nó là thành phần

trung gian giữa WebForm và tầng cơ sở dữ liệu, tiếp nhận yêu cầu từ WebForm rồi gọi xuống tầng cơ sở dữ liệu để rút trích thông tin sau đó tiến hành xử lý rồi trả kết quả về cho WebForm.

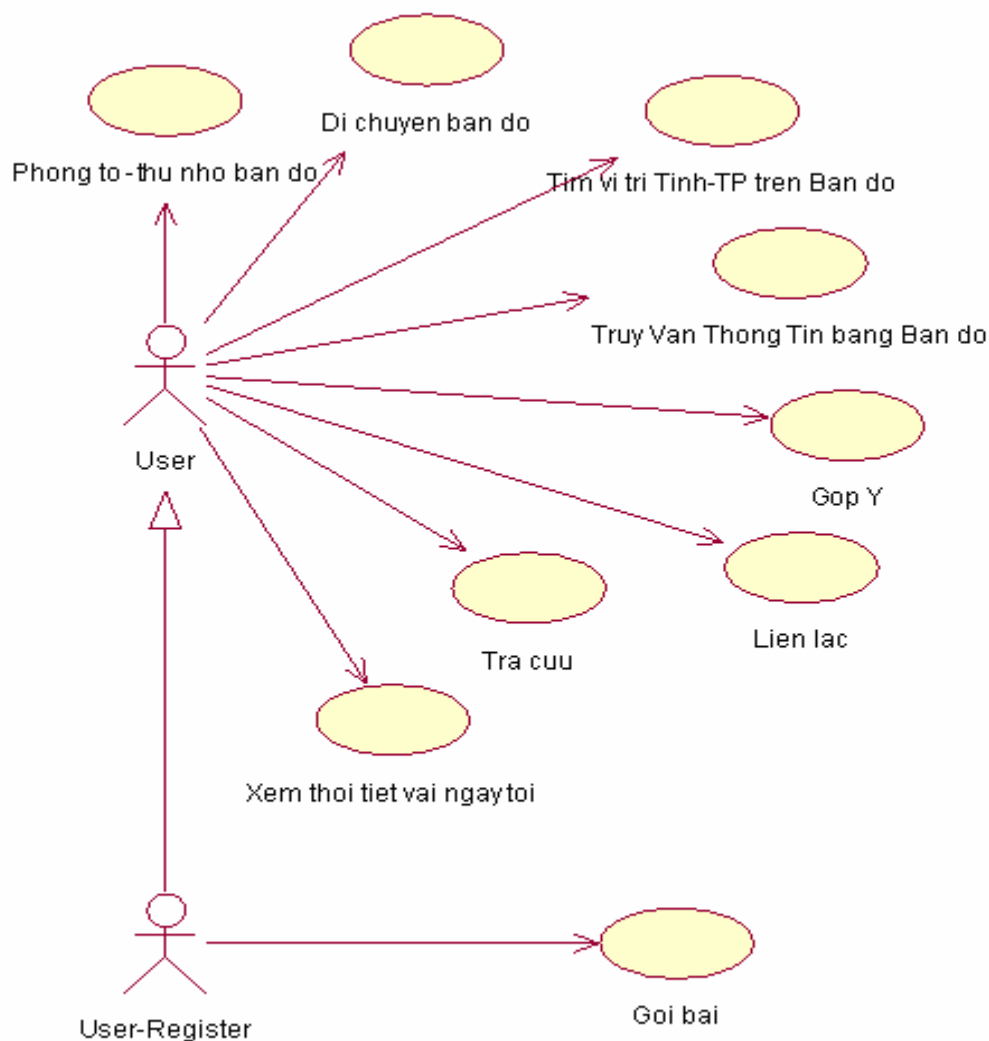
- **Tầng cơ sở dữ liệu** : đóng vai trò trung gian giữa tầng ứng dụng với cơ sở dữ liệu.

## 4.2 Xây dựng mô hình Use –case

### 4.2.1 Xác định Actor và Use case

- Actor
  - User : Tra cứu thông tin, duyệt bản đồ, xem thông tin thời tiết, Góp ý, liên lạc với Admin.
  - User\_Register : có vai trò như một User, ngoài ra còn có quyền gửi bài đến trung tâm.
- Use case
  - Duyệt bản đồ
    - Di chuyển bản đồ
    - Phóng to, thu nhỏ bản đồ
    - Truy vấn thông tin trên bản đồ
    - Tìm vị trí một tỉnh, thành phố trên bản đồ
  - Tra cứu thông tin
    - Tra cứu thông tin thời tiết dựa vào ngày, tháng, năm, tỉnh\_ tp tra cứu
  - Góp ý
    - Gửi mail góp ý với Admin
  - Liên lạc với Admin
    - Liên lạc với Admin thông qua các thông tin cung cấp
  - Gửi bài
    - Gửi bài lên Server
  - Xem thời tiết vài ngày tới
    - Xem thời tiết trong ngày và một vài ngày tới

## 4.2.2 Mô hình Use – case



Hình 4-2 Mô hình Use-Case

### 4.2.3 Đặc tả Use-case

#### 4.2.3.1 Di chuyển bản đồ

- Tóm tắt :  
Use-case này mô tả cách thức người dùng di chuyển bản đồ để xem vùng không gian khác trên bản đồ
- Luồng các sự kiện
  - Luồng sự kiện chính

Use-case này bắt đầu khi người dùng muốn dịch chuyển bản đồ để xem vùng không gian khác trên bản đồ

1. Người dùng click chuột lên button +
2. Hệ thống ghi nhận tình trạng click chuột, thay đổi trạng thái button + sang trạng thái đậm. Chuyển trạng thái các button nào đang ở trạng thái đậm vì được click chuột trước đó về trạng thái bình thường
3. Người dùng click chuột lên bản đồ
4. Hệ thống xác định vị trí Click chuột trên bản đồ, dịch chuyển bản đồ theo chiều ngược lại  
(click chuột càng ở mép ngoài bản đồ chừng nào thì dịch chuyển càng xa và ngược lại )

- Các luồng sự kiện khác

Không có

- Các yêu cầu đặt biệt

Không có

- Điều kiện tiên quyết

Không có

- Post-conditions

Nếu use case thực hiện thành công, hệ thống sẽ hiện thị vùng không gian bản đồ mới ra cho người dùng ngược lại trạng thái bản đồ không thay đổi

- Điểm mở rộng

Không có

#### 4.2.3.2 Phóng to/ Thu nhỏ bản đồ

- Tóm tắt



Use-case này mô tả cách thức người dùng phóng to hay thu nhỏ bản đồ hiển thị.

- Luồng các sự kiện

- Luồng sự kiện chính

Use case này bắt đầu khi có người dùng muốn phóng to hay thu nhỏ bản đồ hiển thị



1. Người dùng Click chuột lên button ZoomIn  hay button ZoomOut 
2. Hệ thống xác định loại button đang chọn, tô đậm button này để người dùng biết button đã được chọn.
3. Người dùng click chuột lên trên bản đồ
4. Hệ thống xác định vị trí click chuột, sau đó tùy theo button được chọn là ZoomIn hay ZoomOut mà hệ thống sẽ phóng to hay thu nhỏ bản đồ tại vị trí được click chuột
5. Người dùng tiếp tục click chuột lên bản đồ để tiếp tục phóng to hay thu nhỏ bản đồ.

- Các luồng sự kiện khác

Không có

- Các yêu cầu đặt biệt

Không có

- Điều kiện tiên quyết

Không có

- Post-conditions

Nếu use case thực hiện thành công hệ thống sẽ hiển thị vùng khung gian bản đồ được phóng to hay thu nhỏ ra cho người dùng, ngược lại trạng thái bản đồ không thay đổi.

- Điểm mở rộng

Không có

#### 4.2.3.3 Tìm kiếm thông tin thời tiết

- Tóm tắt

Use-case này cho phép một người dùng tìm kiếm thông tin thời tiết. Người dùng có thể tìm kiếm thông tin trong quá khứ hay tương lai tùy thuộc vào thời gian tìm kiếm người dùng chọn lựa.

- Luồng các sự kiện

- Luồng sự kiện chính

Use case này bắt đầu khi người dùng muốn tìm kiếm thông tin thời tiết.

1. Người dùng xác định ngày, tháng, năm cần tìm
  2. Người dùng chọn Tỉnh cần tìm kiếm thông tin
  3. Người dùng click vào button tìm kiếm
  4. Hệ thống sẽ xác định thời gian, tỉnh –thành phố cần tìm kiếm, sau đó hiển thị kết quả ra màn hình
- Các luồng sự kiện khác
    1. Người dùng chọn thời gian tìm kiếm quá xa so với hiện tại (cách đây quá nhiều năm ) hệ thống sẽ không tìm thấy thông tin thời tiết, sẽ xuất thông báo tới người dùng
    2. Người dùng không chọn thời gian tìm kiếm, hệ thống sẽ xuất thông báo tới người dùng
    3. Không tìm thấy thông tin thời tiết của một tỉnh-thành phố vào thời gian đó, hệ thống sẽ xuất thông báo tới người dùng
  - Các yêu cầu đặt biệt  
Không có
  - Điều kiện tiên quyết  
Không có
  - Post-conditions  
Nếu use case thực hiện thành công, thông tin thời tiết cần tìm sẽ được hiển thị ra cho người dùng, ngược lại hệ thống sẽ xuất thông báo tới người dùng.
  - Điểm mở rộng  
Không có

#### 4.2.3.4 Gửi báo cáo về trạm

- Tóm tắt  
Use-case này cho phép người dùng (các trạm trung gian) muốn gửi báo cáo về trạm.
- Luồng các sự kiện
  - Luồng sự kiện chính  
Use case này bắt đầu khi người dùng muốn gửi báo cáo về trạm trung tâm

1. Người dùng click lên button **Browse...** để chọn lựa tập tin cần gửi.
  2. Người dùng click lên button **Send**. Tập tin sẽ được gửi lên Server
- Các luồng sự kiện khác
    1. Người dùng không chọn tập tin cần gửi mà click vào button **Send**. Hệ thống sẽ xuất thông báo yêu cầu phải chọn tập tin cần gửi
    2. Người dùng chọn tập tin có kích thước quá lớn sẽ không gửi được. Khi đó server sẽ không nhận được
  - Các yêu cầu đặt biệt  
Không có
  - Điều kiện tiên quyết  
Để có thể gửi bài, người dùng phải đăng nhập vào hệ thống, lúc đó chức năng gửi bài mới hiển thị trên thanh menu chức năng cho phép người dùng truy cập tới Use case này .
  - Post-conditions  
Nếu Use case thực hiện thành công, hệ thống sẽ xuất thông báo tới người dùng
  - Điểm mở rộng  
Không có

#### 4.2.3.5 Gửi góp ý về trang

- Tóm tắt  
Use-case cho phép người dùng góp ý kiến của mình tới trang. Ý kiến này sẽ được gửi vào địa chỉ mail của admin.
- Luồng các sự kiện
  - Luồng sự kiện chính  
Use case này bắt đầu khi người dùng muốn góp ý tới trang
    1. Người dùng nhập vào địa chỉ email của mình
    2. Người dùng nhập vào tên của mình
    3. Người dùng nhập vào chủ đề cần thảo luận
    4. Người dùng nhập vào nội dung thảo luận
    5. Người dùng click vào button **Send**
    6. Hệ thống xử lý và gửi thông điệp này tới địa chỉ của admin.

- Các luồng sự kiện khác
  1. Người dùng không cung cấp đủ thông tin yêu cầu. Hệ thống sẽ yêu cầu người dùng nhập vào
  2. Người dùng click chuột vào button **Cancel** thông điệp sẽ không được gửi
- Các yêu cầu đặt biệt  
Không có
- Điều kiện tiên quyết  
Không có
- Post-conditions  
Nếu use case thực hiện thành công, thông điệp sẽ được gửi tới Admin, đồng thời sẽ có thông báo tới người dùng. Ngược lại sẽ xuất thông báo lỗi.
- Điểm mở rộng  
Khi người dùng đã đăng nhập vào trong hệ thống khi đó hệ thống sẽ cập nhật, lấy địa chỉ email, cùng với tên đã đăng ký của user này đặt vào textbox Email, Name trong Use case, người dùng sẽ không phải nhập lại địa chỉ email, và tên của mình.

#### 4.2.3.6 Liên lạc

- Tóm tắt  
Use-case này mô tả cách người dùng liên lạc với những thành viên của trang Web.
- Luồng các sự kiện
  - Luồng sự kiện chính  
Use case này bắt đầu khi người dùng muốn liên lạc với một trong các thành viên của trang Web
    1. Màn hình cung cấp một số thông tin cần thiết của các thành viên để người dùng liên lạc. Bao gồm địa chỉ Email và điện thoại liên lạc
    2. Khi người dùng click chuột vào một trong các địa chỉ Email trên Microsoft Office Outlook sẽ xuất hiện để người dùng liên lạc.
  - Các luồng sự kiện khác  
Không có
- Các yêu cầu đặt biệt

Không có

- Điều kiện tiên quyết

Không có

- Post-conditions

Không có

- Điểm mở rộng

Không có

#### 4.2.3.7 Tìm vị trí một tỉnh\_ thành phố trên bản đồ

- Tóm tắt

Use-case này mô tả cách người dùng tìm vị trí của một tỉnh-thành phố trên bản đồ. Bản đồ sẽ dịch chuyển tới vùng không gian chứa tỉnh đó cho người dùng đồng thời phóng to một kích thước cố định để nhìn rõ ranh giới của tỉnh –thành phố này.

- Luồng các sự kiện

- Luồng sự kiện chính

Use case này bắt đầu khi người dùng muốn tìm vị trí của một tỉnh –thành phố trên bản đồ

1. Người dùng chọn tên tỉnh hay thành phố cần tìm trong list chứa danh sách các tỉnh-thành phố.
2. Chương trình xử lý và dịch chuyển bản đồ đến vùng không gian chứa tỉnh\_ thành phố vừa chọn, đồng thời phóng to bản đồ một cách vừa phải để người dùng tiện quan sát

- Các luồng sự kiện khác

Không có.

- Các yêu cầu đặt biệt

Không có

- Điều kiện tiên quyết


Không có

- Post-conditions

Nếu use case thực hiện thành công vùng không gian mới của của bản đồ chứa tỉnh-thành phố sẽ được thể hiện, ngược lại trạng thái bản đồ không thay đổi

- Điểm mở rộng  
Không có

#### 4.2.3.8 Truy vấn thông tin

- Tóm tắt  
Use-case này mô tả cách thức người dùng truy vấn thông tin về thời tiết của một tỉnh-thành phố khu vực Nam Bộ
- Luồng các sự kiện
  - Luồng sự kiện chính  
Use case này bắt đầu khi người dùng muốn truy vấn thông tin về thời tiết của một tỉnh-thành phố trên bản đồ
    - 1 Người dùng click chuột lên button **Truy vấn** . Hệ thống sẽ cập nhật trạng thái button bằng cách tô đậm.
    - 2 Người dùng click chuột lên bản đồ. Chương trình xử lý, truy xuất dữ liệu tìm thấy lên một khung hiển thị
    - 3 Người dùng tiếp tục click chuột lên bản đồ để tiếp tục truy vấn thông tin
  - Các luồng sự kiện khác
    - 1 Người dùng click vào vùng không cung cấp thông tin thời tiết, khi đó hệ thống sẽ xuất thông báo tới người dùng.
- Các yêu cầu đặt biệt  
Không có
- Điều kiện tiên quyết  
Không có
- Post-conditions  
Nếu use case thực hiện thành công, một khung hiển thị thông tin hiện ra thông báo tình hình chung của tỉnh-thành phố vừa chọn, từ khung này ta có thể xem tiếp thông tin về thời tiết của tỉnh-thành phố này bằng cách click chuột vào dòng liên kết hiện trên khung.
- Điểm mở rộng  
Không có

#### 4.2.3.9 Xem thời tiết vài ngày tới

- Tóm tắt

Use-case này cho phép người dùng xem thông tin thời tiết trong ngày, vài ngày tới và cả trong quá khứ. Người dùng có thể xem thông tin tổng quát hoặc chi tiết.

- Luồng các sự kiện

- Luồng sự kiện chính

Use case này bắt đầu khi người dùng muốn xem thông tin thời tiết trong nhiều ngày của các tỉnh –thành phố

1 Hệ thống xác định ngày hiện hành sau đó xuất thông tin thời tiết tổng quan của các tỉnh –thành phố.

2 Ở đây có các dòng sự kiện con như sau :

- Người dùng click vào image **Ngày trước** chương trình xác định ngày trước với ngày đang theo dõi, truy xuất thông tin và hiển thị tới người dùng. Tiếp tục thực hiện như vậy nếu người dùng tiếp tục click vào image **Ngày trước**
- Người dùng click vào image **Ngày kế** chương trình xác định ngày tiếp theo ngày đang theo dõi, truy xuất thông tin và hiển thị tới người dùng. Tiếp tục thực hiện như vậy nếu người dùng tiếp tục click vào image **Ngày kế**
- Người dùng click vào liên kết **Chi tiết** chương trình sẽ xác định liên kết trên thuộc tỉnh-thành phố nào sau đó truy xuất thông tin thời tiết một cách chi tiết hơn tới người dùng

- Các luồng sự kiện khác

ở đây sẽ có các dòng sự kiện con sau :

**Ngày trước**

- ✓ Chương trình xác định ngày trước đó so với ngày đang xét
- ✓ Nếu có dữ liệu chương trình sẽ tính toán cho ra dữ liệu của ngày trước đó. Sau đó quay lại luồng sự kiện ban đầu
- ✓ Nếu dữ liệu không còn tồn tại sẽ có thông báo cho người dùng

**Ngày sau**

- ✓ Chương trình xác định ngày sau đó so với ngày đang xét
- ✓ Nếu có dữ liệu chương trình sẽ tính toán cho ra dữ liệu của ngày sau đó. Sau đó quay lại luồng sự kiện ban đầu
- ✓ Nếu dữ liệu không còn tồn tại sẽ có thông báo cho người dùng

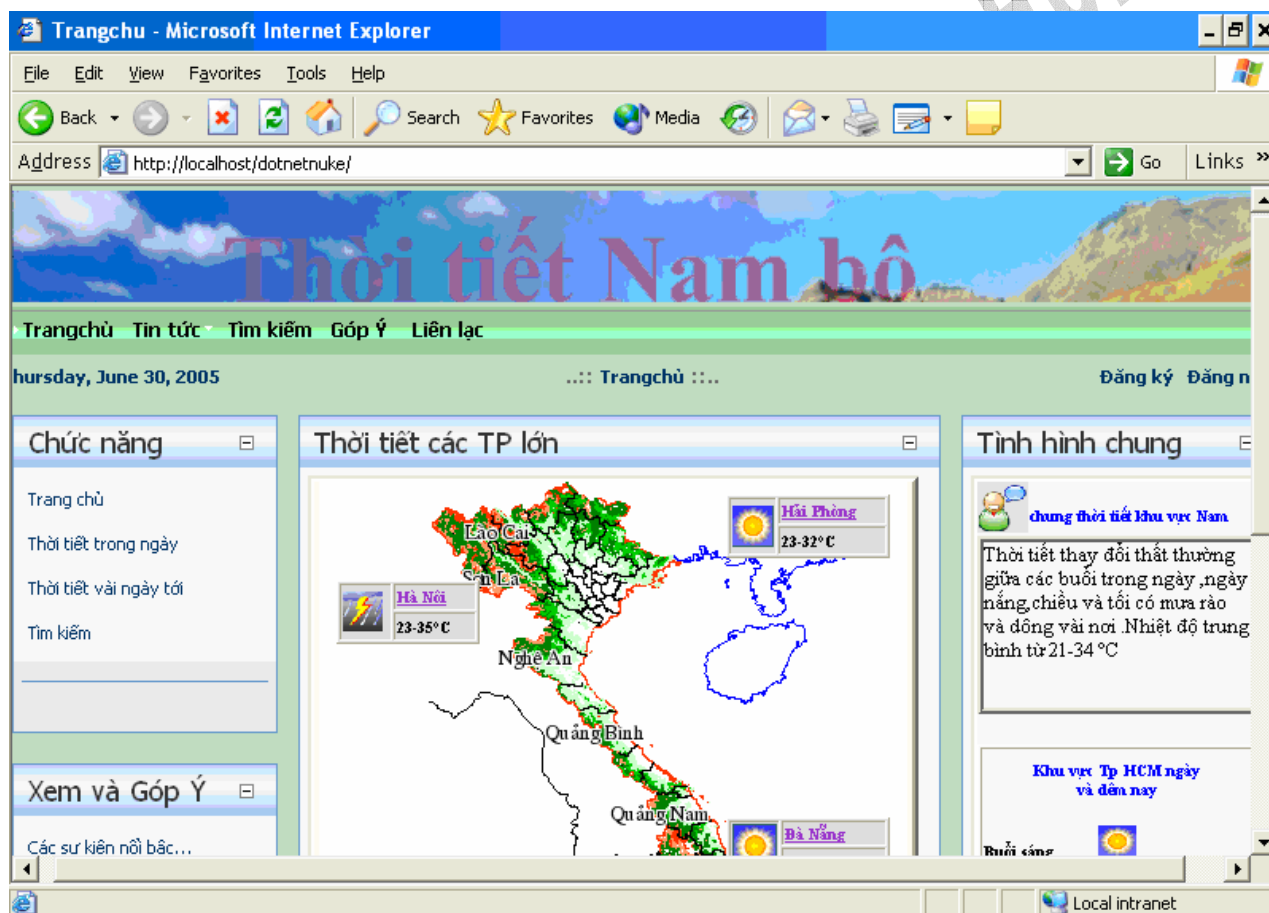
#### Chi tiết

- ✓ Chương trình xác định liên kết vừa được click chuột thuộc tỉnh – thành phố nào. Sau đó truy xuất cơ sở dữ liệu lấy ra thông tin chi tiết và hiển thị cho người dùng
  - ✓ Người dùng click vào button **Trở về** để quay lại luồng sự kiện chính.
- Các yêu cầu đặt biệt  
Không có
  - Điều kiện tiên quyết  
Không có
  - Post-conditions  
Nếu use case thực hiện thành công thông tin thời tiết sẽ xuất ra cho người dùng ngược lại sẽ có thông báo cho người dùng biết.
  - Điểm mở rộng  
Không có



## 4.3 Thiết kế một số màn hình

### 4.3.1 Màn hình Chính



Hình 4-3 Màn hình chính

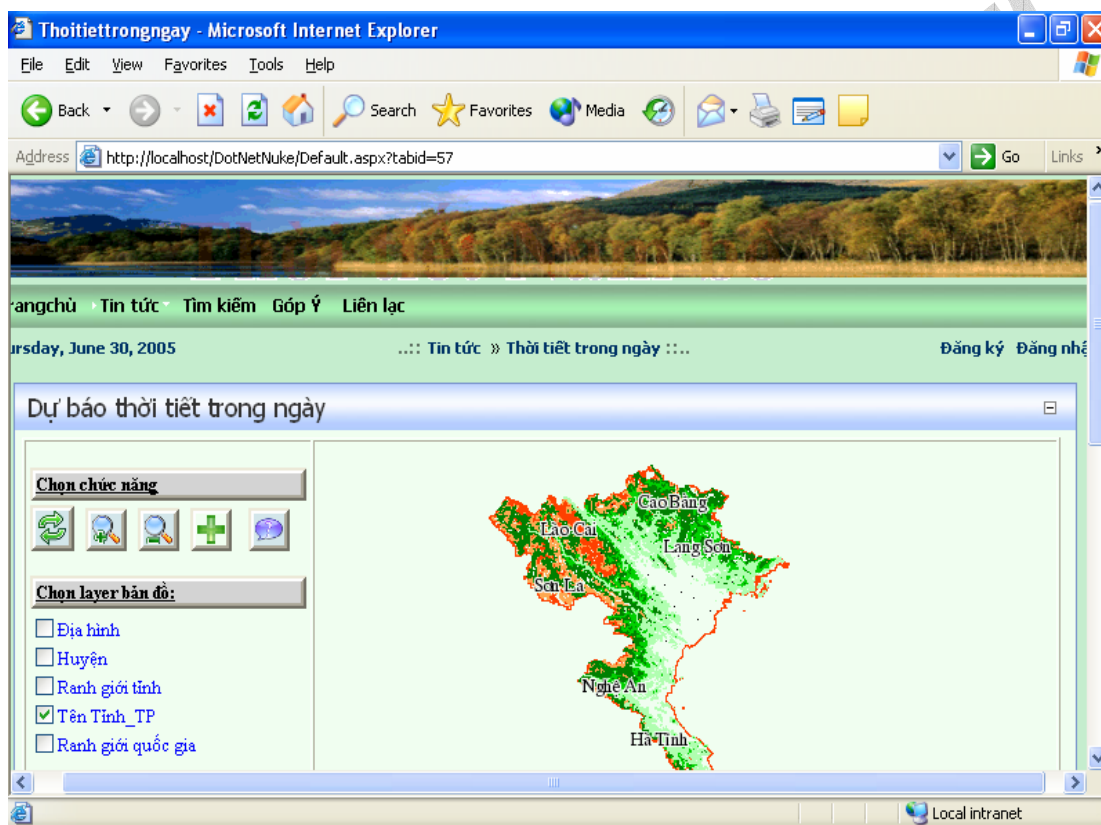
Màn hình được chia làm ba khung (frame) như hình trên :

- Khung bên trái (left panel): gồm các liên kết tới các trang chức năng của chương trình.
- Khung giữa (content panel): gồm bản đồ Việt Nam cùng với thông tin thời tiết chung của một số thành phố lớn của cả nước như Hà Nội, Hải Phòng, Đà Nẵng, Hồ Chí Minh, Cần Thơ.

Thông tin thời tiết của các thành phố này cũng sẽ được hiện ra khi nhấp chuột lên trên thành phố đó.

- Khung bên phải (right panel): chứa thông tin thời tiết chung của khu vực Nam Bộ. Bên dưới là thông tin thời tiết chi tiết các buổi trong ngày của khu vực Thành phố Hồ Chí Minh.

### 4.3.2 Màn hình Thời tiết trong ngày








Hình 4-4 Màn hình thời tiết trong ngày

#### Trang được chia làm hai phần :

- Bên trái là khung chứa các chức năng thao tác trên bản đồ

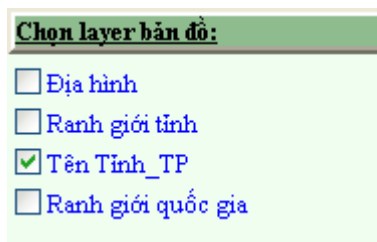


Hình 4-5 Các chức năng thao tác với bản đồ

Nút bấm	Chức năng
	Refresh
	Phóng to bản đồ
	Thu nhỏ bản đồ
	Dịch chuyển bản đồ sang trái, phải, lên trên, xuống dưới.
	Truy vấn thông tin thời tiết khi nhấp chuột lên bản đồ

Bảng 4-1 Nút bấm và chức năng tương ứng

Các chức năng hiển thị bản đồ :

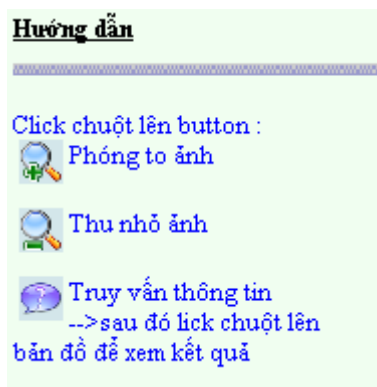


Hình 4-6 Chọn layer hiển thị

Check box	Chức năng
<input checked="" type="checkbox"/> Địa hình	Hiển thị ranh giới địa hình trên bản đồ
<input type="checkbox"/> Ranh giới tỉnh	Hiển thị ranh giới giữa các tỉnh, thành phố
<input type="checkbox"/> Tên Tỉnh_TP	Hiện tên tỉnh, thành phố trên bản đồ
<input type="checkbox"/> Ranh giới quốc gia	Hiện ranh giới giữa các quốc gia

Bảng 4-2 Các layer hiển thị bản đồ

Hướng dẫn thực hiện



Hình 4-7 Hướng dẫn thực hiện

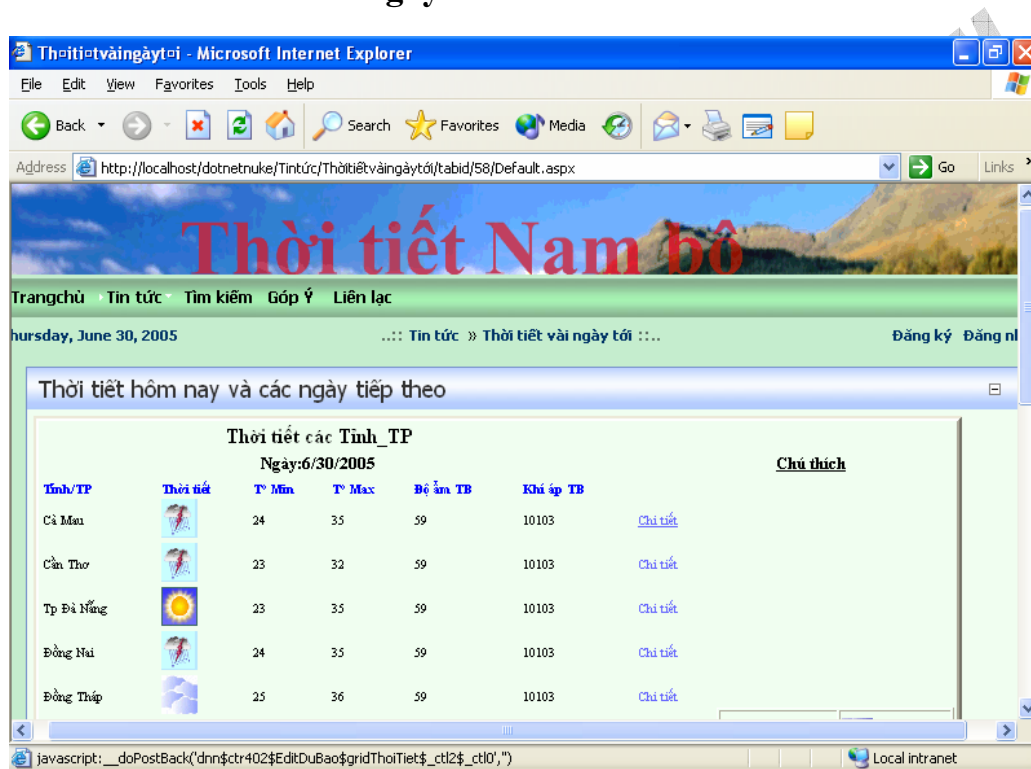
Tìm vị trí của một tỉnh thành phố trên bản đồ





Hình 4-8 Tìm vị trí của tỉnh

- Bên phải là bản đồ thể hiện các trạng thái của bản đồ khi thực hiện các chức năng trên bản đồ

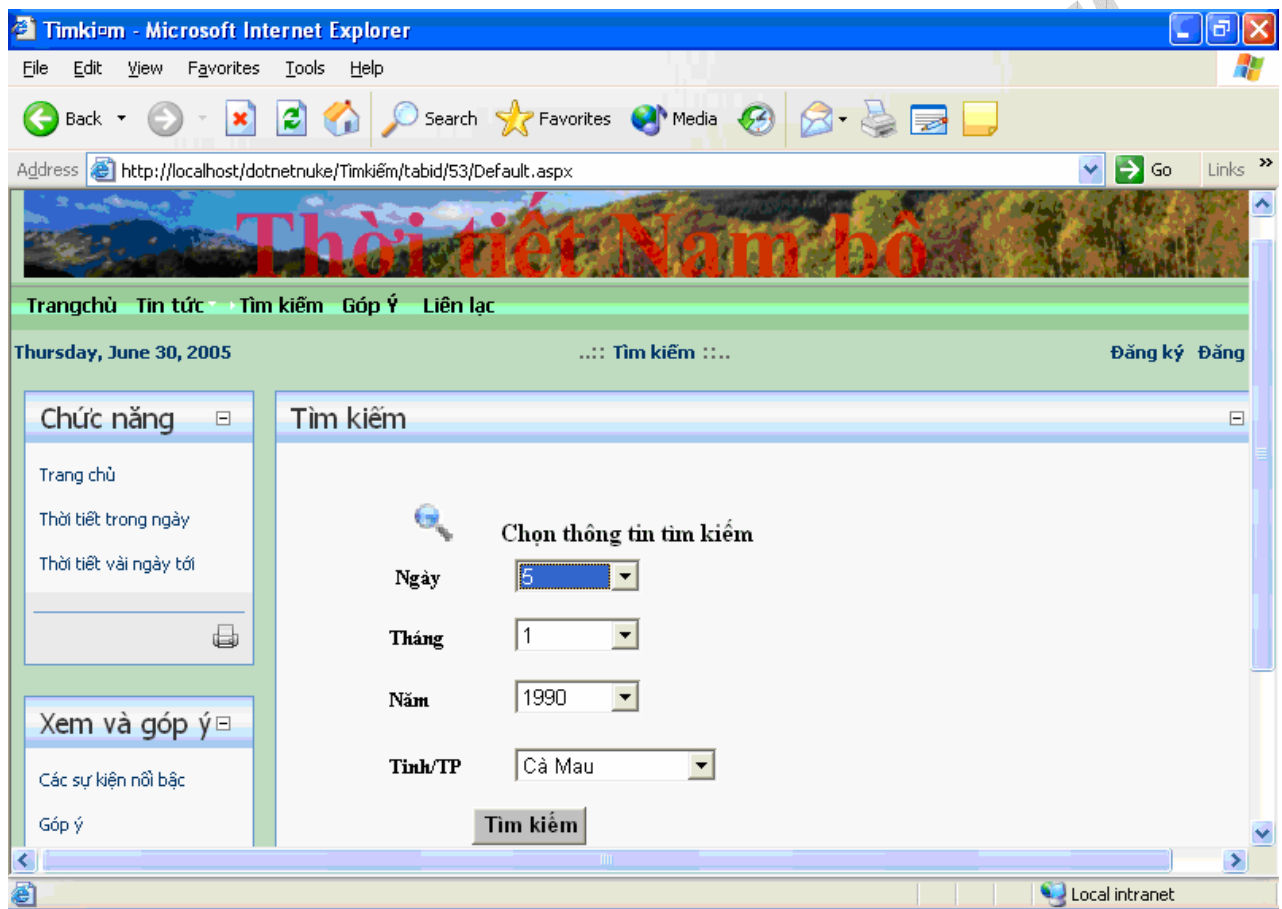
### 4.3.3 Màn hình Thời tiết vài ngày tới



Hình 4-9 Màn hình thời tiết vài ngày tới

Thông tin thời tiết của các tỉnh thành phố khu vực Nam Bộ được hiển thị trong khung lưới. Bạn có thể xem thông tin thời tiết chi tiết bằng cách nhấp chuột vào link '[Chi tiết](#)'. Bạn có thể xem thời tiết ngày tiếp theo bằng cách nhấp vào button , xem ngày trước đó bằng cách nhấp vào button 

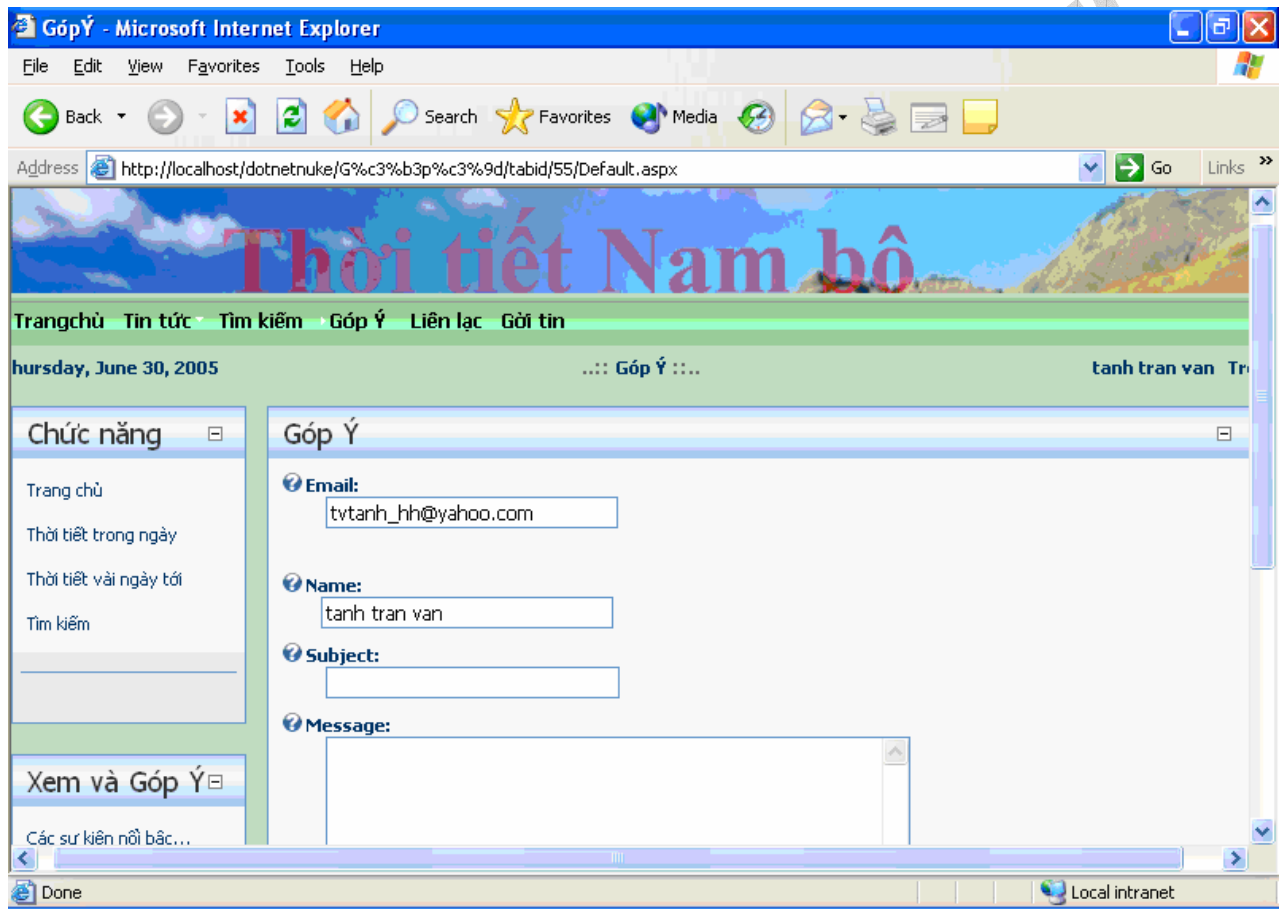
### 4.3.4 Màn hình Tìm kiếm



Hình 4-10 Màn hình tìm kiếm

Chọn ngày, tháng, năm, tỉnh/TP, sau đó nhấp vào button Tìm kiếm thông tin thời tiết cần tìm sẽ hiển thị.

### 4.3.5 Màn hình Góp ý



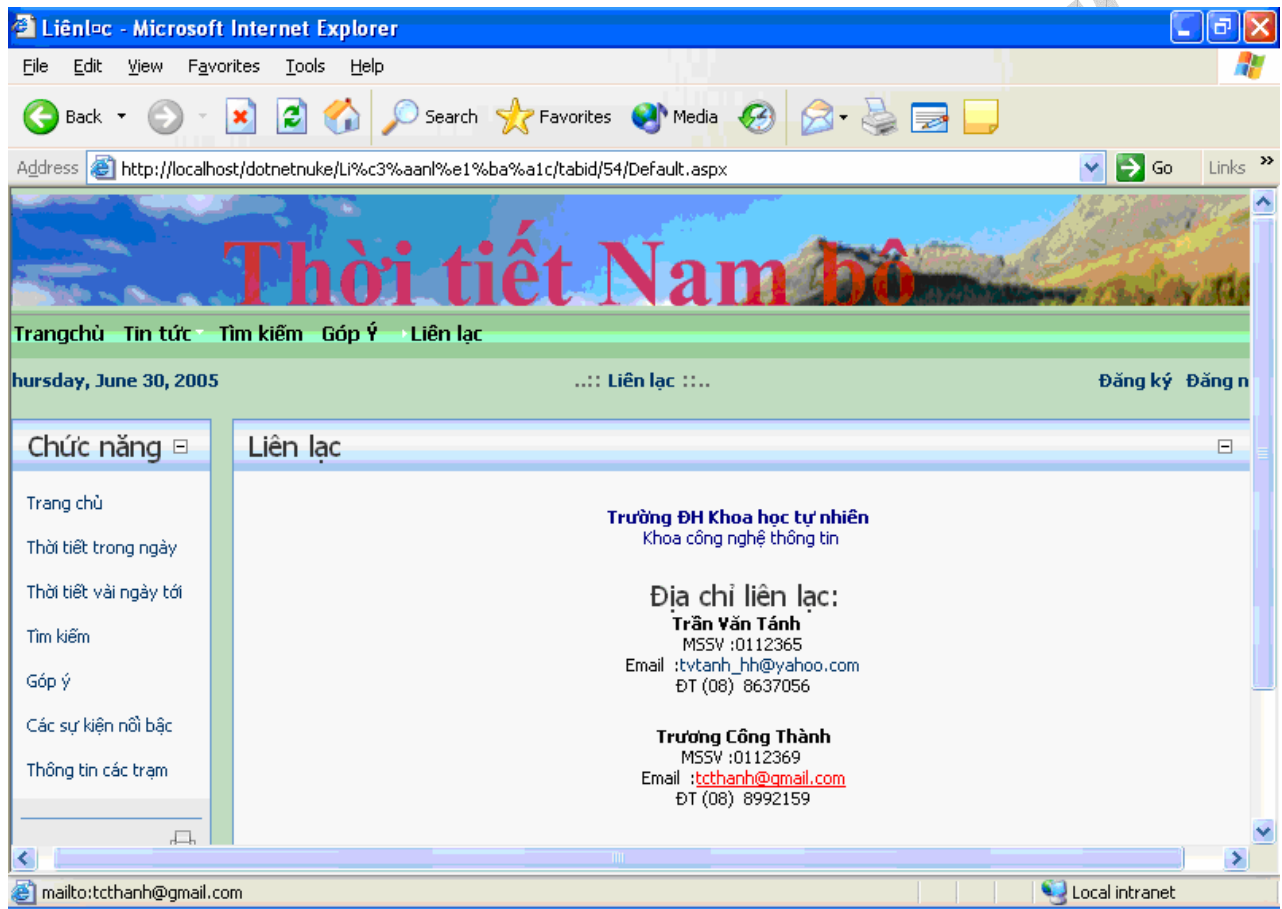
Hình 4-11 Màn hình Góp ý

Nhập các thông tin vào trong các TextBox sau đó click chuột vào button Send



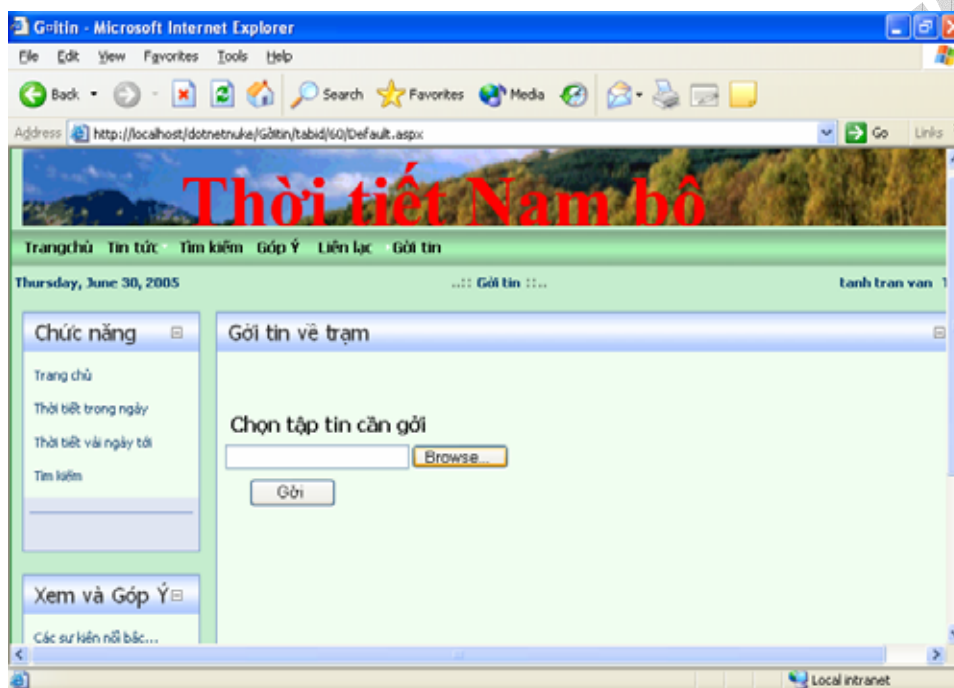
, nội dung message sẽ được gửi đi.

### 4.3.6 Màn hình Liên lạc

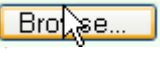


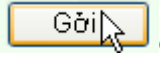
Hình 4-12 Màn hình Liên lạc

### 4.3.7 Màn hình Gửi tin



Hình 4-13 Màn hình Gửi tin

Nhấp vào button  để chọn tập tin cần gửi

Nhấp vào button  để bắt đầu gửi tập tin.



## Chương 5 : Cài đặt và thử nghiệm

### 5.1 Tổ chức dữ liệu dự báo

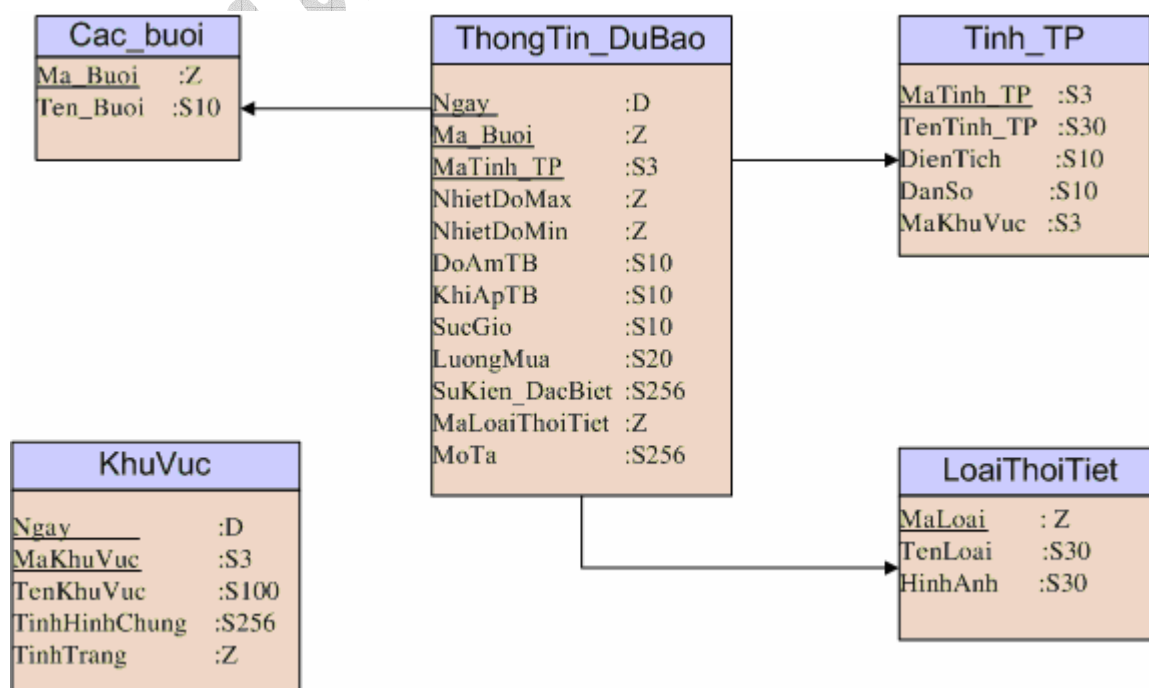
#### 5.1.1 Hình thức lưu trữ dữ liệu

Dữ liệu dự báo được lưu vào CSDL quan hệ để tiện quản lý, truy vấn dữ liệu. CSDL ở đây là SQL Server. Dữ liệu được lưu vào cơ sở dữ liệu là đầu ra của một mô hình dự báo thời tiết, được phân loại và đưa vào CSDL

Hình thức lưu trữ dữ liệu là tập trung.

Do dữ liệu được dùng để dự báo thời tiết nên không xét đến vấn đề lưu trữ dữ liệu dài lâu.

#### 5.1.2 Sơ đồ logic



Hình 5-1 Sơ đồ logic dữ liệu

Giải thích :

Ký hiệu	Ý nghĩa
Z	Kiểu số nguyên
Sn	Kiểu Chuỗi tối đa n ký tự
D	Kiểu ngày

Bảng 5-1 Ký hiệu trường dữ liệu

5.1.2.1 Danh sách các bảng

STT	Tên	Ý nghĩa	Ghi chú
1	Tinh_TP	Bảng lưu trữ thông tin một số tỉnh, thành phố	Chỉ một số tỉnh, thành phố đại diện, không lưu hết các tỉnh, thành phố của cả nước Chủ yếu là khu vực Nam bộ và vài tỉnh thành phố lớn
2	KhuVuc	Bảng lưu trữ các khu vực của cả nước	Chỉ một khu vực Nam Bộ được quan tâm
3	Cac_Buoi	Bảng lưu trữ các buổi trong ngày và cả ngày	
4	LoaiThoiTiet	Bảng lưu trữ thông tin phân loại thời tiết	
5	ThongTin_DuBao	Bảng lưu trữ thông tin để dự báo thời tiết	Chỉ một số thông tin về thời tiết được quan tâm, không cung cấp đầy đủ các yêu cầu về khí tượng thuỷ văn

Bảng 5-2 Danh sách các bảng dữ liệu

5.1.2.2 Mô tả chi tiết các bảng

❖ Tên : Tinh\_TP

Ý nghĩa : Tỉnh –Thành phố

STT	Tên cột	Kiểu	Ràng buộc	Ghi chú
1	MaTinh_TP	Chuỗi 3 ký tự	Khoá chính	
2	TenTinh_TP	Chuỗi 30 ký tự		
3	DienTich	Chuỗi 10 ký		

		tự		
4	DanSo	Chuỗi 10 ký tự		
5	MaKhuVuc	Chuỗi 3 ký tự		

Bảng 5-3 Chi tiết bảng Tinh\_TP

❖ Tên :KhuVuc

Ý nghĩa :Khu vực

STT	Tên cột	Kiểu	Ràng buộc	Ghi chú
1	Ngay	Kiểu ngày (datetime)	Khoá chính	
2	MaKhuVuc	Chuỗi 3 ký tự	Khoá chính	
3	TenKhuVuc	Chuỗi 100 ký tự		
4	TinhHinhChung	Chuỗi 256 ký tự		
5	TinhTrang	Kiểu số nguyên		0 : Bình thường 1 : Cảnh báo 2 : Nguy hiểm 3 :Khẩn cấp

Bảng 5-4 Chi tiết bảng KhuVuc

❖ Tên :Cac\_Buoi

Ý nghĩa :Các buổi trong ngày

STT	Tên cột	Kiểu	Ràng buộc	Ghi chú
1	Ma_Buoi	Số nguyên	Khoá chính	Ma_Buoi chỉ nhận các giá trị sau : 0 : Cả ngày 1 : Buổi sáng 2 :Buổi trưa 3 :Buổi chiều
2	Ten_Buoi	Chuỗi 10 ký tự		

Bảng 5-5 Chi tiết bảng Cac\_Buoi

❖ Tên :LoaiThoiTiet

Ý nghĩa :Loại thời tiết

STT	Tên cột	Kiểu	Ràng buộc	Ghi chú
1	MaLoai	Kiểu số nguyên	Khoá chính	1 :Nắng 2 :Nắng, nhiều mây 3 :Mưa 4 :Mưa dông 5 :Mưa rào và dông 6 :Mây thay đổi 7 : Ít mây 8 :Nhiều mây
2	TenLoai	Chuỗi 30 ký tự		
3	HinhAnh	Chuỗi 30 ký tự		Chứa tên của các file ảnh tương ứng với từng loại thời tiết

Bảng 5-6 Chi tiết bảng Loai ThoiTiet

❖ Tên :ThongTin\_DuBao

Ý nghĩa :Thông tin dự báo

STT	Tên cột	Kiểu	Ràng buộc	Ghi chú
1	Ngay	Kiểu thời gian theo ngày (datetime)	Khoá chính	
2	Ma_Buoi	Số nguyên	Khoá chính	
3	MaTinh_TP	Chuỗi 3 ký tự	Khoá chính	
4	NhietDoMax	Số nguyên		
5	NhietDoMin	Số nguyên		
6	DoAmTB	Chuỗi 10 ký tự		
7	KhiApTB	Chuỗi 10 ký tự		
8	SucGio	Chuỗi 10 ký tự		

9	LuongMua	Chuỗi 20 ký tự		
10	SuKien_DacBiet	Chuỗi 256 ký tự		
11	MaLoaiThoiTiet	Số nguyên		
12	MoTa	Chuỗi 256 ký tự		Mô tả thông tin thời tiết bằng lời

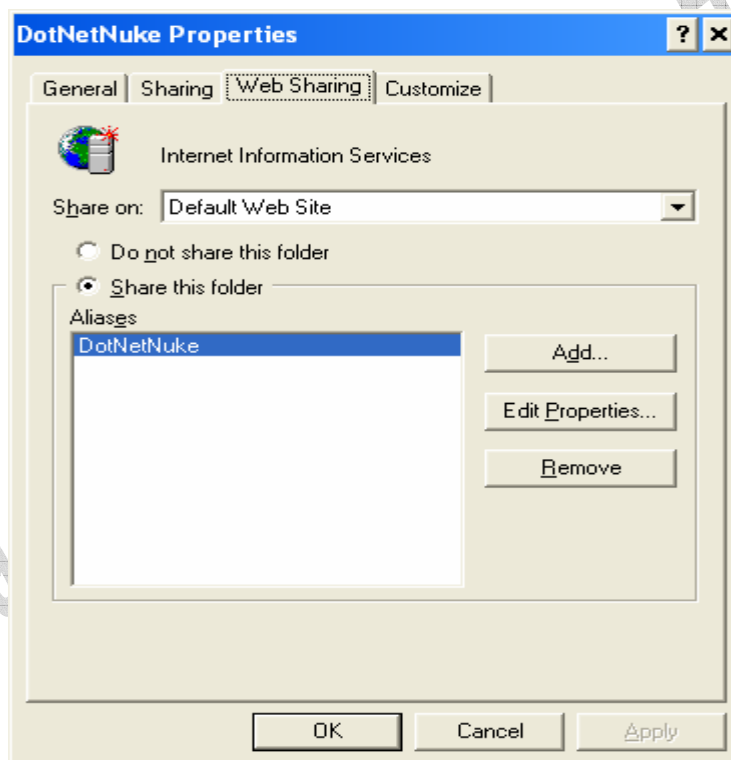
Bảng 5-7 Chi tiết bảng ThôngTin\_DuBao

## 5.2 Cấu hình và cài đặt hệ thống Server-Client

### 5.2.1 Cài đặt trang web.

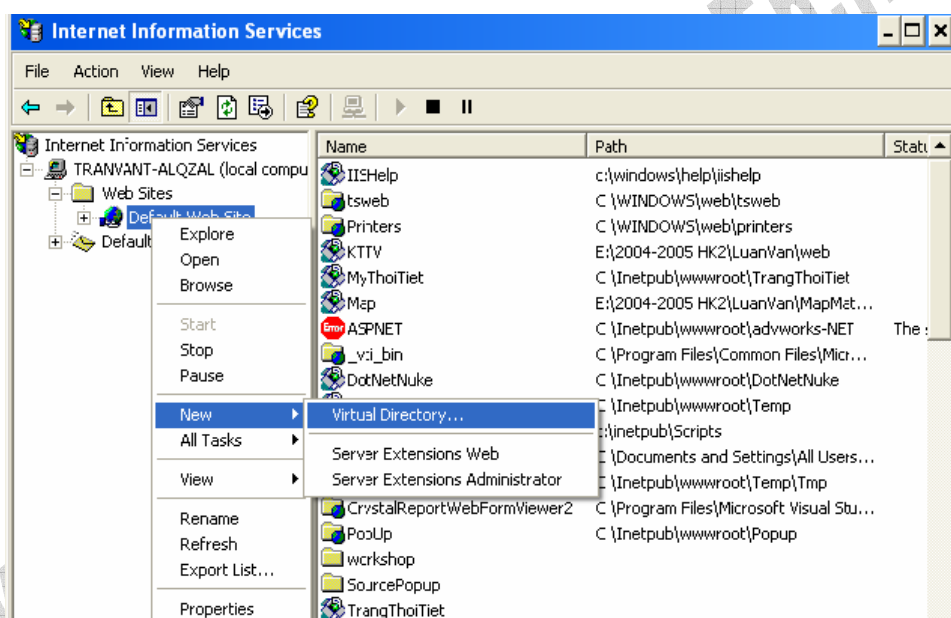
Các bước thực hiện :

- Chép tập tin DotNetNuke.Rar vào trong thư mục 'c:\inetpub\wwwroot\' . Sau đó unzip thành thư mục DotNetNuke.
- Click chuột phải lên thư mục DotNetNuke chọn 'Properties' .  
Trong cửa sổ Properties chọn tab 'Web Sharing' .  
Sau đó check vào radio 'Share this folder ' như hình dưới



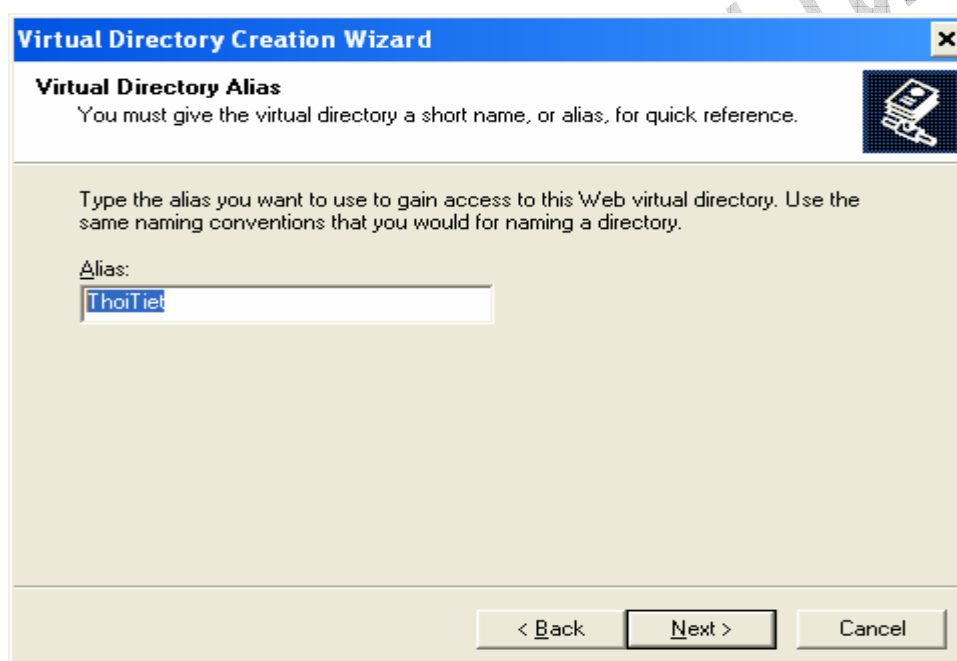
Hình 5-2 Đặt thuộc tính Chia sẻ thư mục web

- Thực hiện ‘Start→Control Panel→Administrative Tools →Internet Information Services’. Trong cửa sổ **Internet Information Services** chọn **Default Web Site**. Bấm chuột phải lên **Default Web Site** chọn **New → Virtual Directory ...** như hình dưới



Hình 5-3 Tạo thư mục Virtual Directory

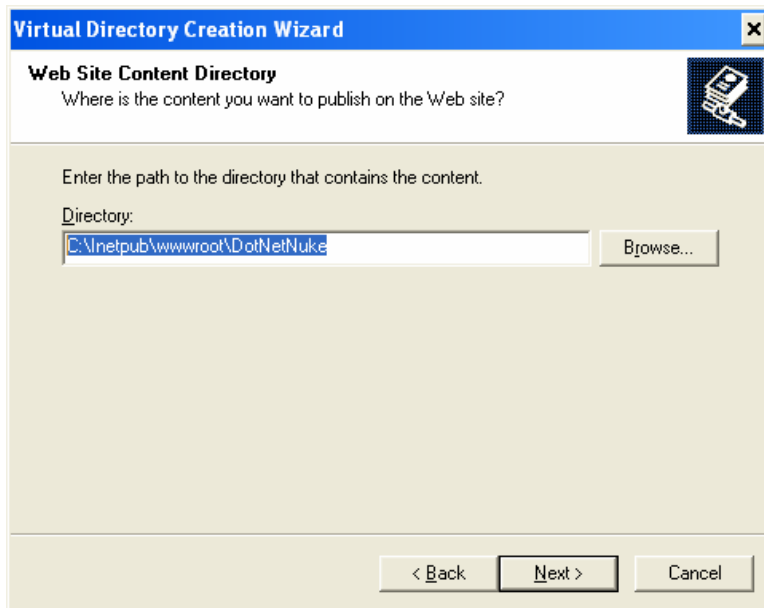
Click **Next** để tiếp tục. Một cửa sổ hiện ra, nhập tên cho Alias chẳng hạn ThoiTiet như hình vẽ :



Hình 5-4 Đặt bí danh (tên trang web)

Click Next để tiếp tục. Một cửa sổ hiện ra như hình vẽ

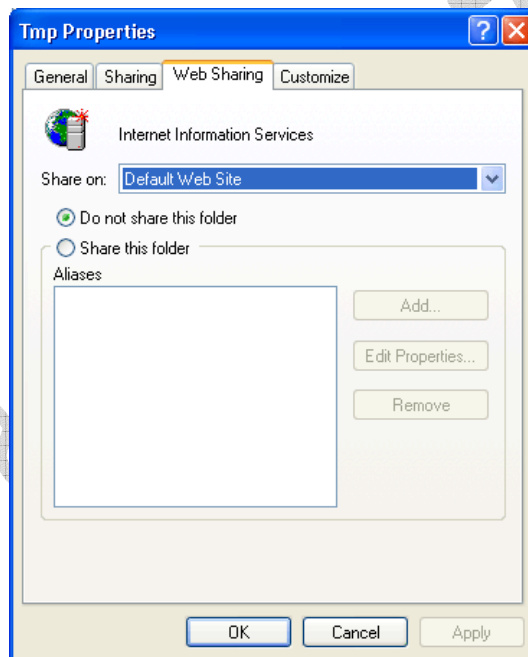
Click lên button **Browse...** chọn đến thư mục chứa project



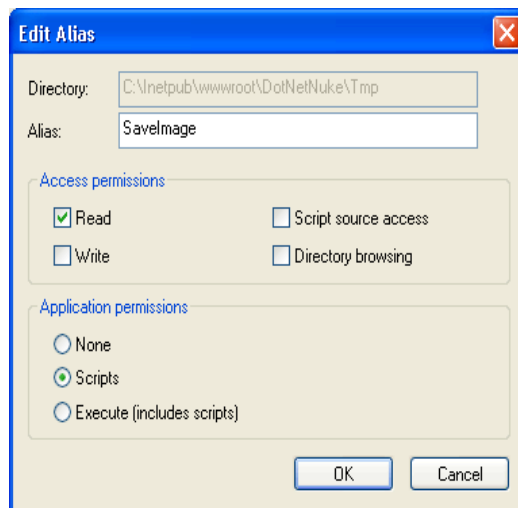
Hình 5-5 Chọn thư mục chứa project

Click **Next** → **Next** → **Finished** để hoàn tất

- Cấu hình thư mục tạm để lưu hình :
  - Vào thư mục DotNetNuke\DataGIS\Tmp.
  - Click chuột phải, trong cửa sổ Properties, chọn tab Web Sharing, chọn mục Share this folder.



- Hiệu chỉnh bí danh chia sẻ trong cửa sổ Edit Alias là SaveImage..

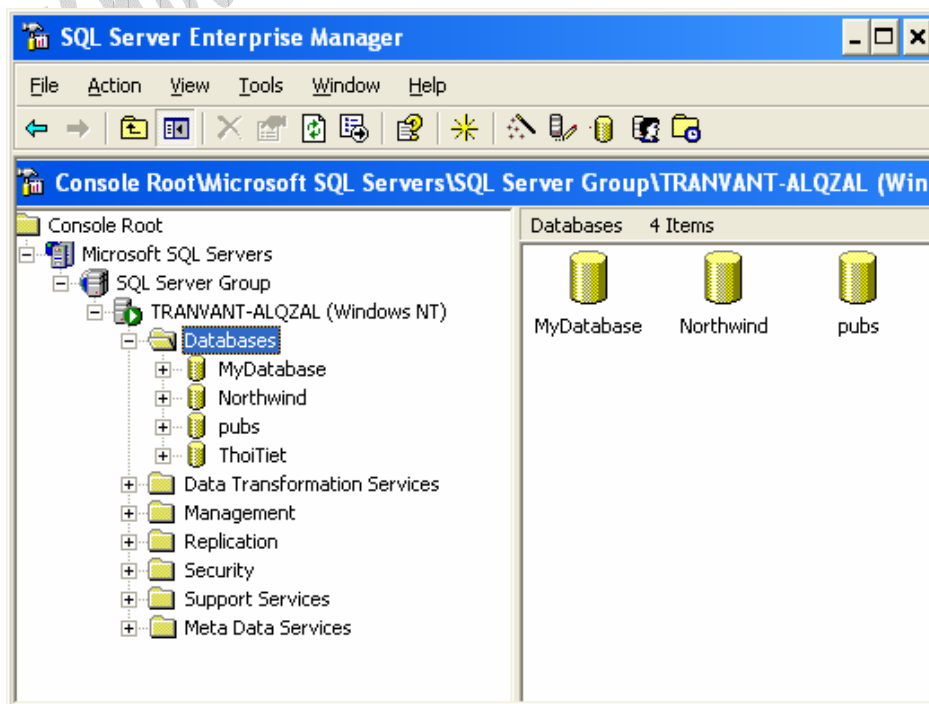


- Chọn OK và Apply để hoàn tất.

### 5.2.2 Cài đặt CSDL Thời tiết

Restore cơ sở dữ liệu đã backup để có thể chạy được ứng dụng

Khởi động Enterprise Manager sau đó vào Database như hình dưới



Hình 5-6 Sử dụng Enterispe Manager

Vào **Tools** chọn **Restore Database...**

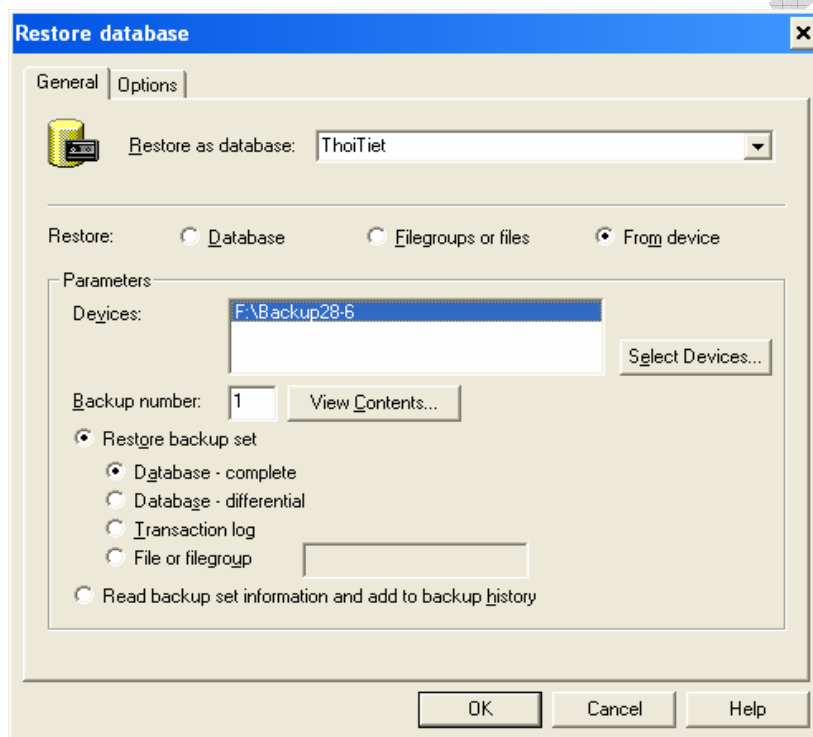
Chúng ta đặt tên cho Database sau khi restore tại combo Restore as Database là

**ThoiTiet**



Để restore từ file ta chọn vào radio button From device, chọn Select devices và add vào đường dẫn đến file backup.

Tại Tab Options ta chọn Force restore over existing database để chép đè lên database nếu nó đã có sẵn.



Hình 5-7 Chọn file Backup CSDL

### 5.2.3 Cấu hình MapServer

#### Nguồn

Chép gói cài đặt MapServer phiên bản mới nhất dành cho hệ điều hành Windows tại trang chủ <http://mapserver.gis.umn.edu/>.

#### Yêu cầu hệ thống:

- Microsoft .NET Framework .1.1 hoặc cao hơn. Download tại :

<http://www.microsoft.com/downloads/details.aspx?FamilyID=262d25e3-f589-4842-8157-034d1e7cf3a3&displaylang=en>

- Có cài một WebServer (tốt nhất nên là IIS hoặc Apache). Ở đây xin minh họa với IIS

#### Các bước cài đặt

- i. Giải nén gói cài đặt vào thư mục trên ổ đĩa cứng, ví dụ : C:\MapServer.

- ii. Chép file mapserv.exe vào thư mục chạy các ứng dụng CGI trên WebServer.

C:\Inetpub\wwwroot\cgi-bin

Hoặc C:\Inetpub\Scripts\

Chép thêm file “msvcr71.dll” là file thư viện của .NET Framework vào cùng thư mục trên

- iii. Chỉ định thư viện các hệ quy chiếu

Cách 1:

Chép thư mục Proj vào vị trí C:\Proj (mặc định MapServer sẽ tìm kiếm thư mục này). Tuy nhiên chỉ áp dụng cho máy localhost, nếu sử dụng host trên Internet thì không thể tạo ra thư mục như thế.

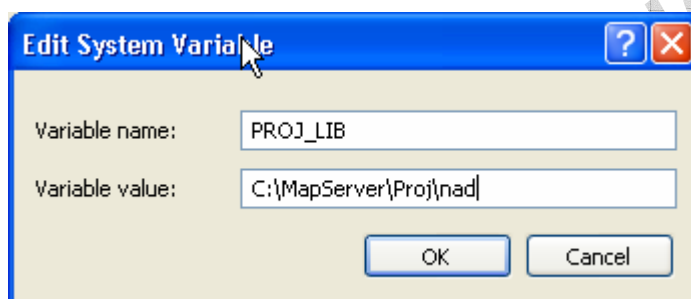
Cách 2:

Tạo mới một biến môi trường với tên là PROJ\_LIB, giá trị là đường dẫn đến thư mục chứa Proj

Để tạo mới một biến môi trường cần phải có quyền Admin, tiến hành như sau:

MyComputer\Properties\Advanced\Enviroment Variables.

Chọn mục System Variables và thiết lập giá trị



Hình 5-8 Thiết lập biến môi trường PROJ\_LIB

Cách 3:

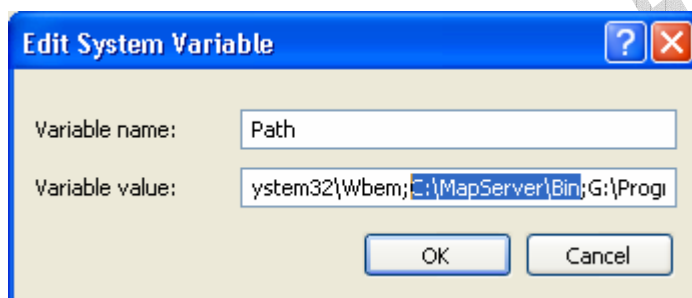
Trong mỗi mapfile, cần xác định tham số CONFIG cho đối tượng Map. Tham số *CONFIG [key] [value]* chỉ định vị trí MapServer tìm kiếm file thư viện, tránh phải tạo biến môi trường PROJ\_LIB đòi hỏi quyền Admin.

Ví dụ:

CONFIG PROJ\_LIB /tmp/proj/

- iv. Chỉ định vị trí các dll liên kết.

Chép tất cả các thư viện liên kết (mở rộng là dll) vào thư mục hệ thống System hoặc System32. Hoặc cập nhật biến môi trường PATH, chỉ đến thư mục chứa các dll này.



Hình 5-9 Hiệu chỉnh biến môi trường PATH

### Test thử

Khởi động lại Web Server. Gõ đường dẫn vào trình duyệt web:

<http://xxx/cgi-bin/mapserv.exe?>

Với xxx là đường dẫn đến Web Server. Nếu kết quả trả về là

"No query information to decode. QUERY\_STRING is set, but empty."

Thì việc cài đặt MapServer đã thành công.

### **5.2.4 Cấu hình Client**

Theo mô hình lựa chọn và với việc sử dụng công nghệ WebGIS nên phía client không phải cấu hình gì cả.

### 5.3 Thử nghiệm

Chương trình được thực thử nghiệm trên máy Pentium(R) 4, CPU 1.8 GHz, 248 MB of Ram kết quả thử nghiệm như sau :

Stt	Tính năng thử nghiệm	Đánh giá
1	Hiển thị bản đồ	Hiển thị bản đồ tương đối tốt. Tuy nhiên hơi bị giật màn hình
2	Phóng to, thu nhỏ bản đồ	Bản đồ được phóng to, thu nhỏ tùy ý, không hạn chế. Ảnh không bị bể khi phóng to hay thu nhỏ
3	Dịch chuyển bản đồ	Bản đồ được dịch chuyển tốt. Tuy nhiên việc dịch chuyển đối với người dùng sẽ khó khăn trong việc định vị chính xác vị trí cần dịch chuyển
4	Tìm vị trí của một tỉnh, thành phố trên bản đồ	Chức năng này được thực hiện tốt. Tỉnh thành phố cần tìm được dịch chuyển tới giữa khung ảnh cho người dùng
4	Truy vấn thông tin thời tiết bằng bản đồ	Chức năng trên được thực hiện tương đối tốt. Tuy nhiên để truy vấn chính xác thông tin cần phóng to hay tìm đến tỉnh, thành phố này sau đó nhấp chuột lên bản đồ.
5	Tra cứu thông tin thời tiết	Tốc độ tìm kiếm nhanh

Các thử nghiệm trên được thực hiện trên localhost do đó kết quả có thể khác trong các thử nghiệm khác trên máy client.

## KẾT LUẬN

### ❖ Nhận xét – Đánh giá

Sau khi thực hiện xong đề tài, chúng em đã đạt được một số kết quả nhất định

- Về phía bản thân :
  - Nắm vững hơn kiến thức về công nghệ :WebGis, DotNet, HTML, JavaScript.
  - Nâng cao tính làm việc theo nhóm, khả năng tìm kiếm tài liệu, thông tin, sử dụng công nghệ DDN, UML.
  - Phát hiện thêm những kiến thức mới lạ, bổ ích ,
- Về phía luận văn :
  - Website hỗ trợ dự báo thời tiết với giao diện trực quan, tiện dụng, tương đối đẹp.
  - Cho phép người dùng theo dõi thông tin về thời tiết một cách nhanh chóng
  - Cho phép người quản trị cập nhật thông tin một cách nhanh chóng, đơn giản với sự bảo mật cao
- Hạn chế
  - Do kế thừa từ DNN ,tốc độ lướt Web sẽ hơi chậm trong những lần truy xuất đầu tiên bởi csdl về thời tiết khá lớn và của nền trang Web
  - Chưa cung cấp khả năng tìm kiếm lại thông tin đã quá cũ một cách nhanh chóng .

### ❖ Hướng phát triển

Do hạn chế về thời gian và khả năng trang web còn một số tính năng cần phát triển :

- Kết hợp các trung tâm dự báo thời tiết trong cả nước để cung cấp thông tin thời tiết của cả nước một cách nhanh chóng, chính xác
- Mở rộng Website thành một dịch vụ web cho phép nhiều đối tượng truy cập.

Xây dựng hệ thống dự báo thời gian thực bằng việc thu dữ liệu GIS trực tuyến từ vệ tinh.

---

## Tài liệu tham khảo

### Tài liệu viết:

- [ 1] Trần Tân Tiên , *Dự báo thời tiết bằng phương pháp số trị*, NXB Đại học Quốc Gia Hà Nội, 1997.
- [ 2] Đài Khí tượng Thủy văn Khu vực Nam bộ, *Danh sách các trạm Đài KTTV Khu vực Nam bộ*, Tài liệu nội bộ, 1998.
- [ 3] Dương Anh Đức, *Giáo trình phân tích và Thiết kế Hướng đối tượng bằng UML*, NXB Thống kê, 2002.
- [ 4] Võ Sỹ Nam – Đỗ Lệnh Hùng Sơn, *Xây dựng một ứng dụng bản đồ trên máy Pocket PC 2002 (Windows CE 3.0) cho phép hiển thị một bản đồ điện tử và cung cấp một số chức năng tìm kiếm thông tin*, Luận văn cử nhân tin học, Đại học Khoa học Tự nhiên Tp.Hồ Chí Minh, 2003.
- [ 5] Lâm Vĩ Quốc – Trần Thị Bích Hạnh, *Hệ thống hỗ trợ tìm đường xe buýt trên Web*, Luận văn cử nhân tin học, Đại học Khoa học tự nhiên Tp. Hồ Chí Minh, 2003.
- [ 6] Jesse Liberty, *Programming C#*, First Edition, O'Reilly, July 2001.
- [ 7] Keith Franklin, *VB.NET for Developers*, Frist Edition, SAMS, America, July 2001.
- [ 8] ESRI, *ESRI Shapefile Technical Description*, July 1998
- [ 9] Nguyễn Trường Sinh (Chủ biên), *Hướng dẫn thiết kế trang web tương tác bằng JavaScript*, NXB Giáo Dục, 2001.

### Website:

- [ 10] Trang web Đài Khí tượng Thủy Văn Nam bộ, <http://kttv-nb.org>
- [ 11] Trang dự báo thời tiết, <http://www.thoitiet.net>
- [ 12] Trung tâm số liệu KTTV Trung ương, <http://hymettdata.com>
- [ 13] MapInfo homepage, <http://mapinfo.com>
- [ 14] MapServer homepage, <http://mapserver.gis.umn.edu>.
- [ 15] MapServer Projects , <http://terraship.gis.umn.edu>.

---

[ 16] Công ty Dolsoft Việt Nam , <http://www.dolsoft.com.vn>

[ 17] WMS Tool in .NET , <http://www.iter.dk>

[ 18] DotNetNuke homepage , <http://www.dotnetnuke.com>

Khoa CNTT - ĐHKHTN TP.HCM

Khoa CNTT - ĐHKHTN TP.HCM

## Phụ lục

### Yêu cầu cấu hình.

#### Server

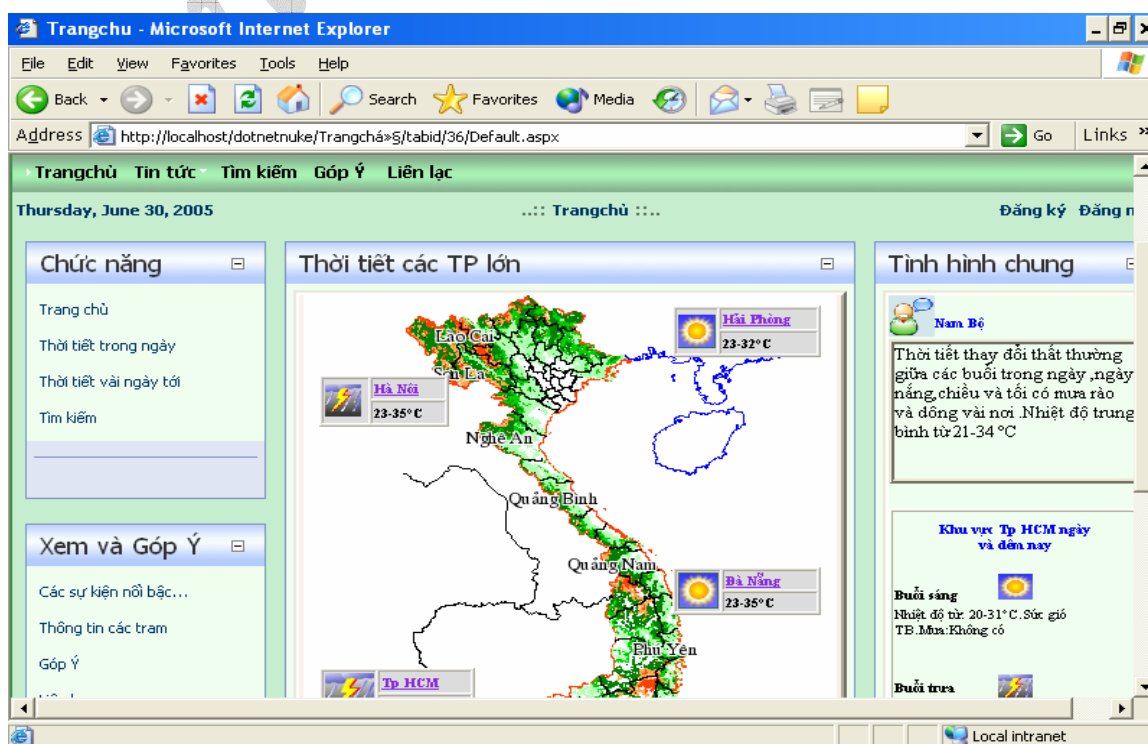
- .NET Framework version 1.1 hoặc cao hơn.
- Web Server : IIS hoặc Apache.
- Hệ quản trị CSDL SQL 2000
- Ổ cứng còn trống trên 160 Mb.

#### Client

- Trình duyệt web như IE 5.0 trở lên, Mozilla, Firefox, Opera, Netscape...

### Hướng dẫn sử dụng

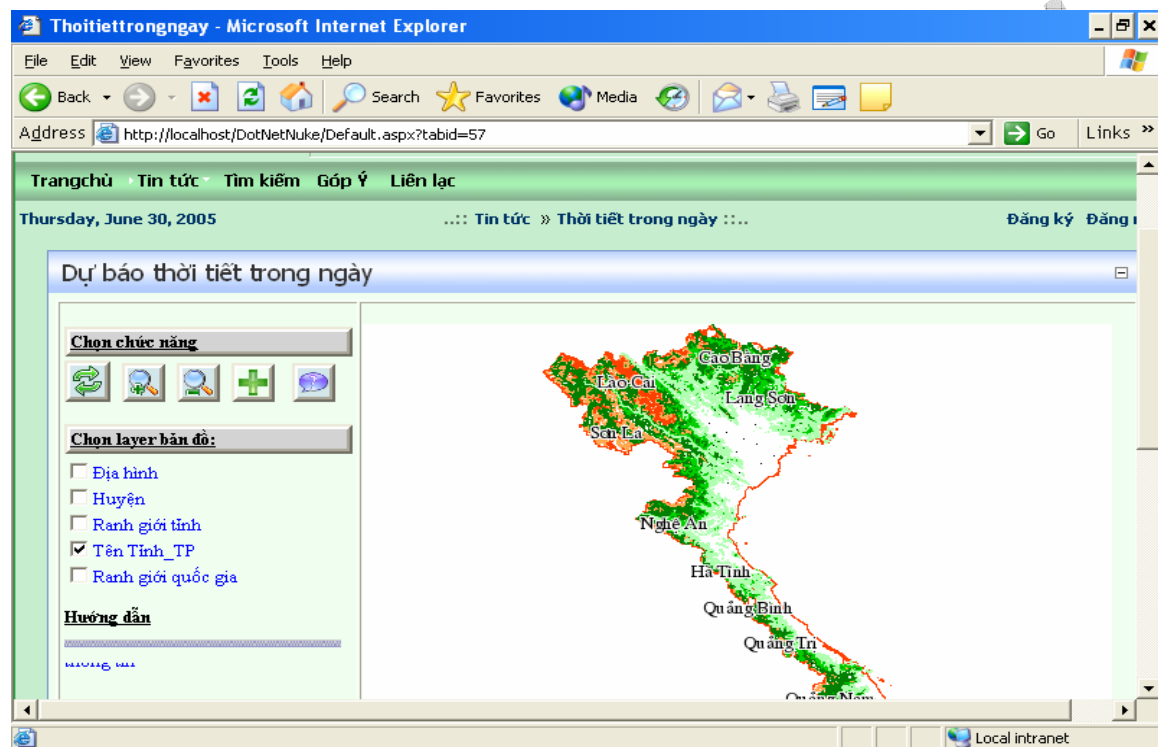
Khi gõ địa chỉ URL (ví dụ: <http://localhost/thoitiet>) .Trang chủ xuất hiện như sau








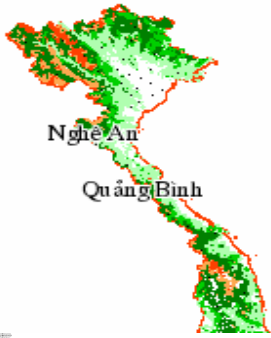


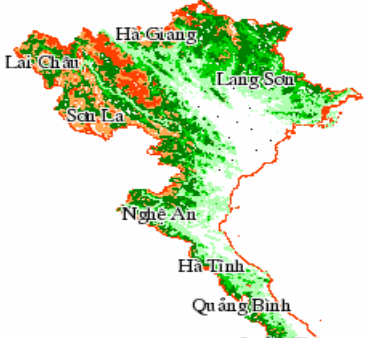



- Để di chuyển tới các trang khác nhấp các liên trong panel bên trái .
- Để xem thông tin thời tiết của các thành phố lớn một các cụ thể hơn nhấp vào tên của tỉnh ,thành phố đó .
- Để xem thời tiết của các tỉnh ,thành phố khác nhấp vào link ‘Thời tiết trong ngày’ hoặc ‘Thời tiết vài ngày tới’ .







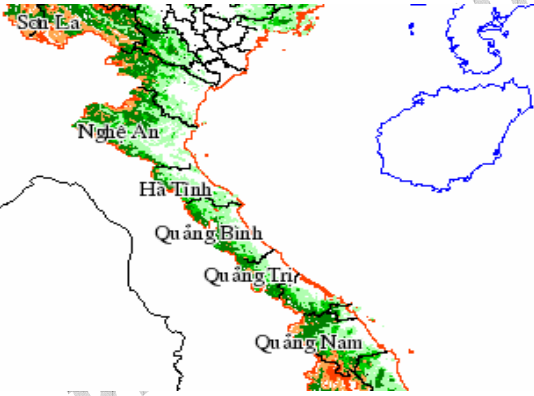



## Trang ‘Thời tiết trong ngày’



<p>Chọn một trong các button trong nhóm button chức năng như hình bên</p>	
<ul style="list-style-type: none"> <li>• Chọn . Button được đánh dấu như hình bên Trạng thái con trỏ chuột được thay đổi thành  khi di chuyển vào vùng bản đồ Nhấp chuột lên bản đồ để phóng to bản đồ</li> </ul>	

<ul style="list-style-type: none"> <li>• Chọn  .Button được đánh dấu như hình bên .Trạng thái con trỏ chuột thay đổi thành  khi di chuyển vào vùng có bản đồ .Nhấp chuột lên bản đồ để thu nhỏ bản đồ .</li> </ul>	<p><b>Chọn chức năng</b></p>  
<ul style="list-style-type: none"> <li>• Chọn  button được đánh dấu như hình vẽ .Chon trỏ chuột thay đổi thành bàn tay .Nhấp chuột lên bản đồ để di chuyển bản đồ . (Trong hình vẽ bên chuột được nhấp bên trái →bản đồ sẽ dịch chuyển về bên phải )</li> </ul>	<p><b>Chọn chức năng</b></p>  
<ul style="list-style-type: none"> <li>• Chọn  button được đánh dấu như hình vẽ .Chon trỏ chuột thay đổi theo .Nhấp vào bản đồ khu vực Nam Bộ để truy vấn thông tin thời tiết</li> </ul>	<p><b>Chọn chức năng</b></p>   <div data-bbox="986 1547 1299 1704" style="border: 1px solid black; padding: 5px;"> <p>Thời tiết</p> <p><b>TP HCM</b></p> <p>Buổi sáng trời mát mẻ ,trưa ,chiều và tối có mưa dông.Gió nhẹ</p> </div>

<ul style="list-style-type: none"> <li>• Chọn các layer muốn hiển thị dưới đây</li> </ul> <div data-bbox="279 315 667 524" style="border: 1px solid black; padding: 5px;"> <p><b>Chọn layer bản đồ:</b></p> <p><input type="checkbox"/> Địa hình</p> <p><input checked="" type="checkbox"/> Huyện</p> <p><input checked="" type="checkbox"/> Ranh giới tỉnh</p> <p><input checked="" type="checkbox"/> Tên Tỉnh_TP</p> <p><input type="checkbox"/> Ranh giới quốc gia</p> </div> <ul style="list-style-type: none"> <li>• Chọn  để thay đổi trạng thái bản đồ hiển thị layer vừa chọn</li> </ul>	<div data-bbox="710 203 1118 309" style="border: 1px solid gray; padding: 2px;"> <p><b>Chọn chức năng</b></p>      </div> 
<p>Muốn dịch chuyển bản đồ đến một tỉnh thành phố nào đó .</p> <p>Chọn tên tỉnh-thành phố trong list</p> <div data-bbox="231 1039 555 1375" style="border: 1px solid gray; padding: 5px;"> <p><b>Tìm vị trí tỉnh\TP</b></p> <p>TP HCM</p> <p>TP HCM</p> <p><b>Vũng Tàu</b></p> <p>Đồng Nai</p> <p>Bình Dương</p> <p>Tiền Giang</p> <p>Đồng Tháp</p> <p>Cần Thơ</p> <p>Cà Mau</p> </div>	<p>Kết quả :</p> 

- Các trang còn lại đơn giản.