

**CẢI TIẾN MÔ HÌNH UDM  
TRONG ỨNG DỤNG QUẢN LÝ KHU DÂN CƯ MỚI**

*Nguyễn Gia Tuấn Anh  
Trường Đại học Hùng Vương*

**TÓM TẮT**

*Bài báo trình bày tóm tắt các mô hình dữ liệu GIS 3D, phân tích một mô hình GIS 3D hiện đại, phục vụ cho các chủ đề quản lý đô thị. Bài báo nêu các khó khăn của mô hình khi triển khai ứng dụng quản lý khu dân cư, lý giải các khó khăn và đề xuất 02 cải tiến: Cải tiến cách biểu diễn trên một số đối tượng 3D-Body và cải tiến cách biểu diễn trên một số đối tượng 2D-Surface. Ngoài ra, bài báo chứng minh bằng phương pháp toán học các cải tiến này là hữu ích cho lưu trữ và biểu diễn các đối tượng Body và Surface. Sự hữu ích này làm gia tăng hiệu quả cho câu truy vấn. Cuối cùng là phần thực nghiệm cài đặt cấu trúc dữ liệu mới và dùng VB.NET để hiển thị các khối 3D.*

*Từ khóa: 3D, GIS, mô hình dữ liệu.*

**1. Giới thiệu**

**1.1. Sơ lược GIS 3D**

Chúng ta đang sống trong thế giới 3 chiều. Các ứng dụng công nghệ thông tin cũng cần phản ánh đúng về thế giới thực, ứng dụng GIS không thể là một ngoại lệ. Đó cũng là lí do hình thành GIS 3D. GIS 3D được phát triển trên nền GIS 2D. Khi phát triển các ứng dụng GIS 3D trên nền GIS 2D, chúng ta gặp phải một số khó khăn sau: dữ liệu thiếu chiều thứ 3, phương pháp tiếp cận GIS 2D theo mô hình dữ liệu quan hệ, bị hạn chế bởi các quy định về dạng chuẩn, mô hình quan hệ không chấp nhận một số khái niệm mới rất cần cho đặc trưng GIS 3D. Vì thế, GIS 3D phải chọn con đường cho chính mình.

Xây dựng một hệ thống GIS 3D, chúng ta cần giải quyết một số chủ đề quan trọng sau [9, 10]:

- Nhập dữ liệu đầu vào: bao gồm các kiểu dữ liệu đa dạng như vector, raster; phù hợp với các thủ tục nhập liệu phong phú gồm thủ công, bán thủ công, tự động.
- Thao tác dữ liệu: gồm chuyển dữ liệu qua lại giữa các hệ tọa độ; chuyển dữ liệu giữa các độ đo khác nhau; làm trơn mượt dữ liệu.
- Phân tích: Xem như là phần lõi hệ thống, liên quan đến các thao tác topology, độ đo.

- Trình bày dữ liệu: Các thông tin, kết quả được trình bày dưới dạng bản đồ, biểu mẫu, báo cáo, đồ thị...

- Mô hình dữ liệu: phải biểu diễn được các đối tượng trong thực tế và mối quan hệ giữa các đối tượng.

Các đối tượng trong GIS được biểu diễn dưới 2 thành phần: phi không gian và không gian. Các thành phần phi không gian sử dụng các kiểu dữ liệu thông thường: chuỗi, số. Các thành phần không gian cần sử dụng các kiểu dữ liệu đặc biệt. Các ứng dụng về GIS 3D là khá rộng, có thể kể một vài lĩnh vực điển hình sau [4]: Hệ nghiên cứu sinh thái, hệ giám sát môi trường, hệ thống hải dương học, hệ khai thác khoáng sản, hệ quản lý đô thị,... Tốc độ đô thị hóa diễn ra ngày càng nhanh chóng đã tạo ra áp lực lên bài toán quản lý đô thị [3]. Các mô hình dữ liệu GIS 3D khác nhau có các thuận lợi và khó khăn khác nhau khi nhúng chúng vào một ứng dụng cụ thể. Mục đích của bài báo này là trình bày tóm tắt các mô hình dữ liệu 3D đã biết, phân tích các đặc điểm chính. Phân tích kỹ một mô hình UDM, nhận định những trở ngại của mô hình với bài toán quản lý các khu dân cư mới và nêu các cải tiến trên mô hình này. Các cải tiến nhằm đạt các mục tiêu: giảm lược thể tích dữ liệu, giảm lược các Node, các Line dư thừa và như thế nó giúp trực tiếp trong việc nâng cao tốc độ truy vấn dữ liệu cho tìm kiếm và hiển thị dữ liệu. Ngoài ra, cách cải tiến về biểu diễn các đối tượng 3D còn giúp việc tính toán thể tích, chiều cao trở nên dễ dàng, chính xác.

Bài báo được trình bày theo dàn ý, phần 1.2 mô tả tóm tắt các mô hình dữ liệu GIS 3D, phần 1.3 trình bày các nhận xét về mô hình UDM, phần 1.4 mô tả các đặc điểm về dữ liệu không gian của bài toán quản lý khu dân cư mới. Phần 2 mô tả 3 cải tiến và chứng minh bằng toán học ích lợi của cải tiến về khối lượng lưu trữ, loại bỏ các dữ liệu thừa trong việc lưu trữ và hiển thị. Phần 3 mô tả thực nghiệm cài đặt cấu trúc dữ liệu sau các cải tiến lên hệ quản trị CSDL Oracle 10g và dùng VB.NET để hiển thị các đối tượng 3D.

## ***1.2. Một số mô hình dữ liệu GIS 3D***

### **3D-FDS (Format Data Structure)**

Mô hình do Molenaar đề xuất 1990 và được Rikker cùng các đồng nghiệp phát triển 1993 [1, 8], lấy 4 đối tượng cơ sở là Body, Surface, Line, Point. Mô hình dùng các nguyên tố: Arc, Edge, Face, cộng thêm một số quy tắc cho mô hình: Arc phải là một đoạn thẳng, Arc và Face không giao nhau. Edge, Face phải là hai chiều, Surface có đường biên và có thể có vài Surface không lồng nhau bên trong. Body có đường biên và có thể có vài Body không lồng nhau bên trong, Arc và Node có thể tồn tại bên trong Face hay Body.

### **TEN (Tetrahedral Network )**

Mô hình do Pilouk đề nghị 1996 [1, 2, 8], dựa trên 4 đối tượng cơ bản Point,

Line, Surface, Body. Các thành phần nguyên tố gồm: Arc, Node, Triangle. Một Body được tạo bởi các Tetra, một Surface được tạo bởi các Face, một Line được tạo bởi các Arc. Node là một thành phần của Arc, Arc là một thành phần của Triangle, Triangle là một thành phần của Tetrahedron, các ngoại lệ không xem xét.

### **Mô hình OO (Object Oriented)**

Mô hình do Dela Losa, Cervelle đề xuất 1999 [1, 2]. Mô hình có thể biểu diễn và quản lý các lỗ hổng 2D và đường hầm 3D, có thể bảo quản và hỗ trợ các đối tượng không gian phức tạp. Mô hình được xây dựng trên 4 đối tượng cơ sở: 0-Simplex, 1-Simplex, 2-Simplex, Volume và sử dụng 3 đối tượng nguyên tố: Node, Arc, Face. Hướng của Face cần được lưu trữ. Một Simplex là đối tượng hình học cơ bản trong chiều đã cho. Mỗi chiều có 1 phần tử nhỏ nhất gọi là Simplex. Simplex của n chiều gọi là n-Simplex

### **Mô hình SSM- Zlatanova (Simplified Spatial Model.)**

Mô hình do Zlatanova đề xuất 2000 [1, 2, 8], mô hình tập trung vào việc thực hiện các câu truy vấn hiển thị hình dạng trên ứng dụng GIS Web. Mô hình chỉ sử dụng 2 đối tượng nguyên tố: Node, face. Mô hình dùng 4 đối tượng cơ bản: Point, Line, Surface, Body, nó không sử dụng nguyên tố 1D: Arc, xem Arc là một phần của 2 hay nhiều Face. Face phải là lồi, có hướng. Các mối quan hệ: Node nằm trong Face thể hiện bởi quan hệ NodeFace, Face nằm trong Body thể hiện trong quan hệ FaceBody. Hướng của Face cần lưu trữ và thứ tự các Node tạo Face cần thể hiện trong các quan hệ.

### **Mô hình SOMAS (Solid Object Management System)**

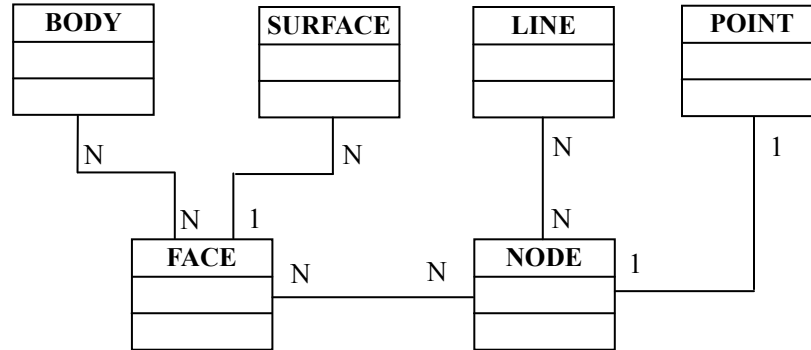
Mô hình do Plund đề xuất năm 2001 [2], tiếp cận theo phương pháp hướng đối tượng. Bao gồm 4 thực thể cơ bản: Point, Line, Polygon, Solid. Mô hình gồm 3 đối tượng nguyên tố: Vertex, Edge, Face, nó bảo quản cấu trúc của các đối tượng rõ ràng. Mô hình tuy dễ hiểu nhưng không được triển khai trong một hệ quản trị cơ sở dữ liệu nào.

### **UDM (Urban data Model)**

Mô hình theo đề nghị của Coors 2003 [7], đây là mô hình dữ liệu 3D [5] dựa trên 4 đối tượng cơ bản Point, Line, Surface, Body và 02 nguyên tố Node, Face. Đối tượng Arc không được đề cập ở mô hình này. UDM là một mô hình dữ liệu GIS 3D biểu diễn dữ liệu cho các ứng dụng quản lý đô thị. UDM có 2 ưu điểm lớn: hiển thị tốt bề mặt các đối tượng và khối lượng dữ liệu lưu trữ khá nhỏ. Mô hình dựa trên nền tảng của việc tam giác hóa.

Node được mô tả bởi 3 thuộc tính X,Y, Z. Line được biểu diễn ít nhất qua 2 Node cũng có thể qua nhiều Node. Mỗi Face định nghĩa bằng nhiều Node trên cùng mặt phẳng, trong thực tế khi biểu diễn Face người ta dùng tam giác để biểu diễn, nghĩa là 1 Face chỉ có 3 Node. Một Surface được tạo từ nhiều Face. Một Body được tạo bởi các

Face, ngoài ra Body còn sử dụng 2 cặp điểm  $(X1, Y1, Z1)$ ,  $(X2, Y2, Z2)$  lưu trữ hình hộp có thể tích nhỏ nhất bao quanh một Body. Việc sử dụng 2 cặp điểm này nhằm xây dựng chỉ mục, phục vụ cho bài toán truy xuất không gian. Các Face tạo nên một Body được lưu trữ một cách tường minh nhờ quan hệ FACEBODY. Mô hình UDM (hình1) đã giản lược một số quan hệ Arc-Node, Arc-Face nhờ bỏ đối tượng Arc.



**Hình 1. Mô hình UDM**

**Biểu diễn mô hình UDM ở dạng các quan hệ, ta có:**

**NODE (#IDN, X, Y, Z).** Mô tả: Mỗi Node gồm một mã số Node duy nhất, dùng phân biệt với các Node khác. Mỗi Node trong không gian 3 chiều được biểu diễn bởi 3 tọa độ X, Y, Z trong hệ tọa độ Oxyz.

**FACE (#IDF, IDN1, IDN2, IDN3, IDS).** Mô tả: Mỗi Face gồm một mã số Face duy nhất. Một Face là một tam giác được tạo từ 3 đỉnh gọi là 3 Node, có các mã số IDN1, IDN2, IDN3. Do vậy, có thể giản lược quan hệ NODEFACE. Một Face sẽ thuộc một Surface IDS. Một đa giác, phụ thuộc vào dạng lồi hay lõm, khi được chia thành các tam giác sẽ có các giải thuật khác nhau [5].

**BODY (#IDB, X1,Y1, Z1, X2, Y2, Z2).** Mô tả: Một Body có một mã số IDB duy nhất, ngoài ra một Body còn được sử dụng 2 cặp điểm P1  $(X1,Y1, Z1)$ , P2  $(X2, Y2, Z2)$  nhằm biểu diễn một hình hộp chữ nhật có thể tích nhỏ nhất bao lấy Body. Cặp điểm P1, P2 sẽ phục vụ cho công việc truy xuất không gian sau này.

**SURFACE (#IDS, DESC).** Mô tả: Mỗi Surface có một mã số IDS và các thông tin khác trình bày trong cột DESC.

**FACEBODY (#IDB, # IDF).** Mô tả: Một Body được tạo từ nhiều Face và một Face có thể thuộc nhiều Body.

**POINT (#IDP, IDN, DESC).** Mô tả: Mỗi Point có một mã số IDP, các thông tin khác trình bày trong cột DESC. IDN là khóa ngoại.

**LINE (#IDL, DESC).** Mô tả: Mỗi Line có một mã số IDL và các thông tin khác trình bày trong cột DESC.

**LINENODE (#IDL, #IDN, SEQ).** Mô tả: Một Line có thể tạo từ nhiều Node và

ít nhất là 2 Node. Các Node trên một Line là có thứ tự, thứ tự được lưu trữ trên cột SEQ.

Bảng so sánh, các mô hình [6].

Năm	Tác giả	Mô hình	Ứng dụng	Các đối tượng	Các nguyên tố	Thuận lợi
1990	Molenaar	3DFDS	Ứng dụng GIS 3D	Point, Line, Surface, Body	Node, Edge, Face	Dễ dàng tách bạch đối tượng không gian và phi không gian
1996	Pilouk	TEN	Ô nhiễm về khí	Point, Line, Surface, Body	Node, Arc, Triangle, Tetra	Hiện thị bề mặt tốt, mô tả được các vật thể phức tạp
2000	Zlatanova	SSM	Web GIS 3D	Point, Line, Surface, Body	Node, Face	Khối lượng lưu trữ vừa phải
2001	Pfund	SOMAS	Ứng dụng GIS 3D	Point, Line, Polygon, Solid	Vertex, Edge, Face	Bảo quản cấu trúc của các đối tượng rõ ràng. Dễ hiểu
2002	ShiWenzhong và đồng nghiệp	OO3D	Quản lý đô thị	Point, Line, Surface, Volumn	Node, Segment, Triangle	Mô tả tốt các vật thể phức tạp
2003	Coors	UDM	Quản lý đô thị	Point, Line, Surface, Body	Node, Triangle	Hiện thị bề mặt tốt, khối lượng lưu trữ vừa phải

### 1.3. Hai nhận định về UDM

**Nhận định 1.** Do mô hình dựa trên nền tảng tam giác hóa, nếu một Surface là phẳng thì việc tam giác hóa sẽ có 2 nhược điểm: khối lượng dữ liệu lớn và nó tạo ra một số đoạn thẳng không có thực vừa mất chi phí cho lưu trữ, vừa mất chi phí cho hiển thị và mất chi phí cho I/O.

**Nhận định 2.** Vì mô hình biểu diễn các đối tượng 3D bởi tập các tam giác của các mặt ngoài tạo nên đối tượng 3D. Nếu đối tượng đó là các hình trụ, hình lăng trụ đứng thì mô hình UDM có 4 nhược điểm: khối lượng dữ liệu lớn; nó tạo ra một số đoạn

thẳng không có thực; tạo ra một số Node không có trong thực tế và không thể tính thể tích các đối tượng 3D. Việc tạo ra các Node, các đoạn thẳng không thực không những làm tăng kích thước dữ liệu, tăng các chi phí I/O mà còn làm cho việc hiển thị các đối tượng phức tạp và không giống thực tế.

Dựa vào 2 nhận định quan trọng trên và một số đặc trưng chính về thuộc tính không gian của bài toán quản lý khu dân cư mới trong phần 1.4, bài báo đã đề xuất các cải tiến trong 2.1, 2.2 và 2.3.

#### **1.4. Một số đặc trưng không gian của khu dân cư mới**

- Các công trình được xây dựng trên một mặt bằng được xem là phẳng.
- Không gian được quy hoạch tốt về kiến trúc, tuân thủ các quy định trong xây dựng.

- Khu dân cư bao gồm công trình: chung cư, nhà liền kề, nhà biệt thự, công viên, cây xanh, trường học, công trình thể thao, bệnh viện. Các công trình hạ tầng khác: đường nội bộ, đường ống nước, đường cống thoát nước, đường dây điện, đường cáp internet, đường gaz.

- Tỷ lệ số căn hộ là chung cư có tỷ lệ vượt trội so với nhà liền kề, nhà biệt thự.

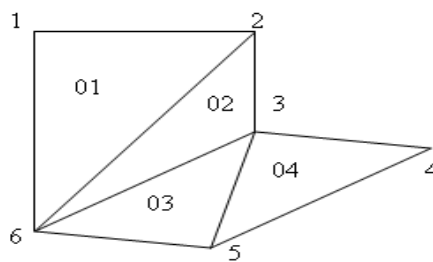
- Các căn hộ có hình dạng là các hình lăng trụ đứng. Số các căn hộ có hình dạng, kích cỡ giống nhau (khác nhau về vị trí) là khá lớn.

- Các ứng GIS 3D tại Việt Nam về lĩnh vực này là hoàn toàn mới.

## **2. Các cải tiến**

### **2.1. Cải tiến 1**

Phát biểu: Với một Surface là phẳng, không áp dụng phương pháp tam giác hóa cho Surface. Surface sẽ dùng phương pháp mới để lưu trữ dựa vào các đỉnh và thứ tự của nó. Mỗi đỉnh được mô tả bởi một Node. Dữ liệu Surface biểu diễn trong quan hệ SUR-PL-NODE (#IDS, #IDN, SEQ). Trong đó, SEQ cho biết thứ tự của các Node tạo thành Surface. Thực tế Surface là: sân thể thao, hồ bơi, công viên. Ví dụ: Cho một công viên có mặt S1 gồm 6 ( $n=6$ ) đỉnh 1, 2, 3, 4, 5, 6. Nếu tam giác hóa S ta có 4 mặt: 01, 02, 03, 04.



**Hình 2.** S1 được chia thành 4 tam giác theo UDM

Theo UDM, S1 biểu diễn trên quan hệ FACE (#IDF, IDN1, IDN2, IDN3, IDS)

FACE

IDF	IDN1	IDN2	IDN3	IDS
01	.	.	.	S1
02	.	.	.	S1
03	.	.	.	S1
04	.	.	.	S1

Trong quan hệ FACE, nếu mỗi cột có kích thước 2 byte, số byte cần biểu diễn S1:  $10 \times 4 = 40$  byte ( $40 = 10n - 20$ ). Và theo UDM, nó tạo ra 3 ( $3 = n - 3$ ) cạnh dư thừa: 62, 63, 53. Nếu dùng phương pháp sau cải tiến 1, S1 biểu diễn trong quan hệ SUR-PL-NODE (#IDS, #IDN, SEQ) và dữ liệu của S1:

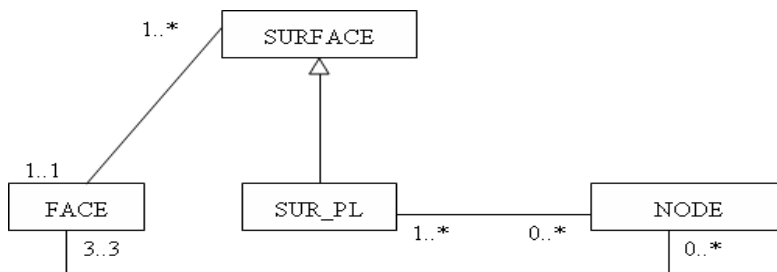
SUR-PL-NODE

IDS	IDN	SEQ
S1	1	1
S1	2	2
S1	3	3
S1	4	4
S1	5	5
S1	6	6

Trong quan hệ SUR-PL-NODE, nếu mỗi cột có kích thước 2 byte, số byte cần biểu diễn S1:  $6 \times 6 = 36$  bytes ( $36 = 6n$ ).

Tổng quát: Nếu một đa giác có  $n$  cạnh, số byte cần thiết để biểu diễn đa giác theo phương pháp tam giác hóa:  $(10n - 20)$  byte. Ngược lại, số byte cần thiết khi sử dụng cải tiến 1 :  $6n$ .

Mô hình dữ liệu sau cải tiến 1



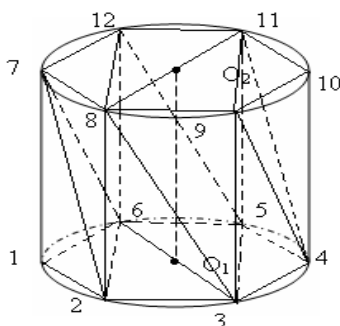
Hình 3. Một phần mô hình UDM sau cải tiến Surface

Với một đa giác có số cạnh là  $n$ , cải tiến 1 có các ưu điểm:

- Khối lượng dữ liệu lưu trữ rút gọn:  $(4n - 20)$  byte. ( $4n - 20 = 10n - 20 - 6n$ ).
- Giảm lược việc trình bày  $(n - 3)$  đoạn thẳng không có thực trong thực tế, chúng được hình thành do quá trình tam giác hóa.

## 2.2. Cải tiến 2

Phát biểu: Các vật thể khối có dạng là khối trụ đứng hay nằm, sẽ lưu trữ với cách thức riêng dựa vào bán kính đáy, chiều cao và tâm vòng tròn đáy trong quan hệ BODY-CYL (#IDB, IDN, RA, HE). Trong đó, RA: bán kính, HE: chiều cao, IDN: tâm vòng tròn đáy cho hình trụ đứng (hay vòng tròn trái cho hình trụ nằm). Thực tế, Cylinder là các bể chứa nước lớn của các chung cư. Ví dụ: cho bể chứa nước dạng hình trụ B1, giả sử tam giác hóa bề mặt hình trụ bằng 6 mặt như hình 4.



**Hình 4.** Tam giác hóa hình trụ bằng 6 mặt

Với phương pháp biểu diễn cũ, B1 được biểu diễn trong quan hệ BODYFACE (#IDB, # IDF)

BODYFACE

IDB	IDF
B1	1
B1	2
...	...
B1	20

Biểu diễn B1 trong BODYFACE gồm 20 dòng: 12 dòng mô tả các mặt bên và 8 dòng cho 2 đáy. Giả sử mỗi cột là 2 byte, số byte cần thiết biểu diễn B1:  $4 \times 20 = 80$  byte ( $80 = 16n - 16$ ). Đồng thời, UDM tạo ra 12 ( $12 = 2n$ ) Node và 28 ( $28 = 6n - 8$ ) cạnh. Các Node và cạnh này là không có thực. Theo cách biểu diễn sau cải tiến 2, B1 biểu diễn trong BODY-CYL (#IDB, IDN, RA, HE) là:



## BODY-CYL

IDB	IDN	RA	HE
B1	1	R1	H2

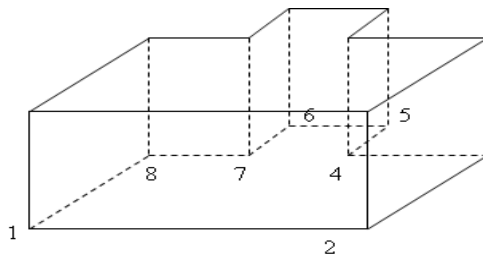
Giả sử mỗi cột trong BODY-CYL là 2 byte, số byte để biểu diễn B1: 8 byte.

Tổng quát, với hình trụ đứng B1, nếu dùng n mặt để tam giác hóa mặt bên của hình lăng trụ đứng theo UDM, thì cải tiến 2 có các ưu điểm :

- Khối lượng dữ liệu rút gọn:  $(16n - 24)$  byte. ( $16n - 24 = 16n - 16 - 8$ ).
- Rút gọn số Node không cần thiết:  $2n$ .
- Giảm lược việc trình bày (vẽ) một số đoạn thẳng không có thực trong thực tế mà được hình thành do quá trình tam giác hóa:  $6n-8$ .
- Tính toán thể tích hình trụ chính xác nhờ công thức toán học.

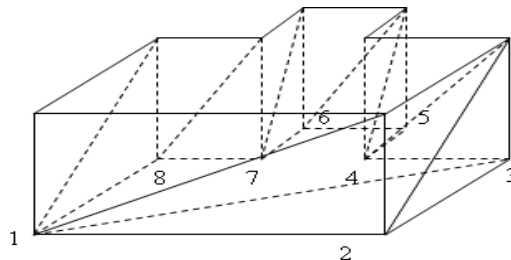
### 2.3. Cải tiến 3

Phát biểu: Với vật thể khối có dạng là lăng trụ đứng, nó sẽ lưu trữ với cách thức mới dựa vào mặt đáy và độ cao với quan hệ BODY-PYR (#IDB, IDS, HE). Trong đó, HE: chiều cao hình lăng trụ đứng, IDS: mô tả đa giác đáy. Thực tế, các lăng trụ đứng là các căn hộ.



**Hình 5.** Hình lăng trụ đứng, đáy là đa giác 8 cạnh

Ví dụ: Cho căn hộ dạng hình lăng trụ đứng B1 như hình 5, có mặt đáy là một đa giác 8 ( $n = 8$ ) cạnh.



**Hình 6.** Tam giác hóa bề mặt hình lăng trụ đứng

Nếu biểu diễn theo phương pháp cũ bằng cách tam giác hóa, B1 có dạng như hình 6. B1 được biểu diễn trong quan hệ BODYFACE (#IDB, #IDF).

BODYFACE

IDB	IDF
B1	1
B1	2
...	...
B1	28

Biểu diễn B1 trong BODYFACE gồm 28 dòng: 16 dòng mô tả các mặt bên, 12 dòng mô tả 2 đáy. Giả sử mỗi cột là 2 byte, số byte cần thiết:  $4 \times 28 = 112$  byte ( $112 = 16n - 16$ ) và nó tạo 18 ( $18 = 3n - 6$ ) cạnh không có trong thực tế. Nếu biểu diễn B1 theo cải tiến 3, B1 biểu diễn trong BODY\_PYR (#IDB, IDS, HE) là:

BODY\_PYR

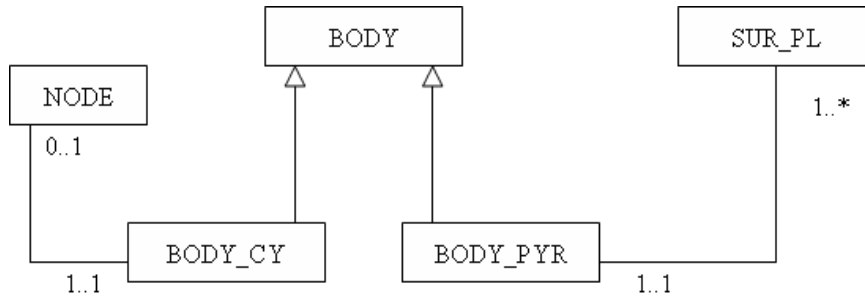
IDB	IDS	HE
B1	S1	H1.

Giả sử mỗi cột trong BODY\_PYR là 2 byte, số byte cần thiết biểu diễn B1 là 6 byte.

Tổng quát, với hình lăng trụ đứng B1, có mặt đáy là n cạnh. Cải tiến 3 có các ưu điểm so với UDM:

- Khối lượng dữ liệu rút gọn:  $(16n - 22)$  byte. ( $16n - 22 = 16n - 6 - 16$ )
- Rút gọn số Node không cần thiết: n.
- Giảm lược việc trình bày (vẽ) một số đoạn thẳng không có thực trong thực tế mà được hình thành do quá trình tam giác hóa  $3n - 6$ . ( $3n - 6 = 2n - 6 + n$ ).

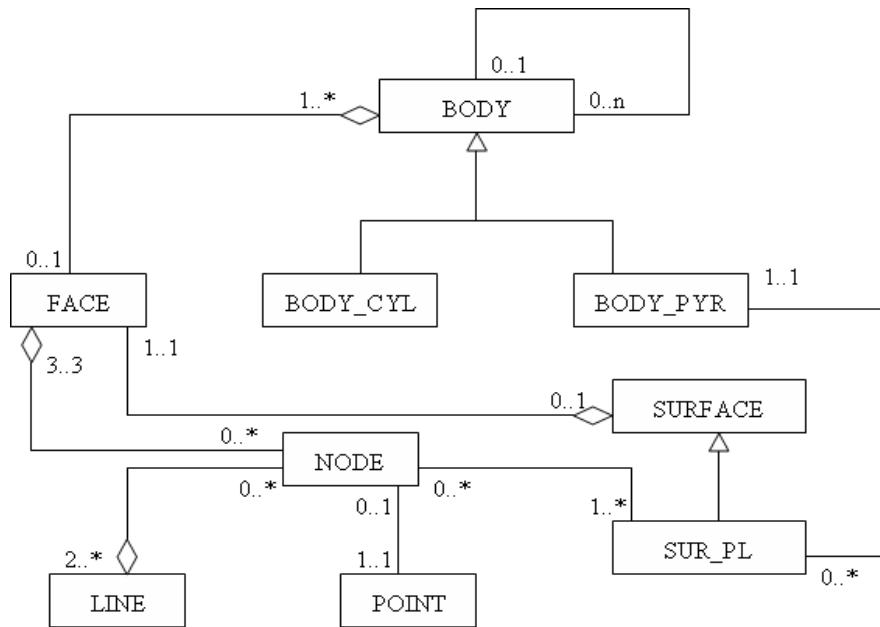
Mô hình dữ liệu sau các cải tiến 2, 3 (hình 7)



Hình 7. Một phần mô hình UDM sau cải tiến 2, 3

2.4. Mô hình UDM sau các cải tiến 1, 2, 3

Mô hình UDM sau 3 cải tiến 1, 2, 3 và bổ sung mối liên kết topology về quan hệ cha con giữa 2 Body có dạng sau (hình 8).



**Hình 8.** Mô hình UDM sau cải tiến 1, 2, 3

Trong đó, mối quan hệ đệ quy giữa Body và Body diễn đạt ngữ nghĩa: nằm trong. Một Body B1 có thể nằm trong 1 Body B2, B2 là Body có thể tích nhỏ nhất chứa B1 và nó có thể chứa nhiều Body.

Phân rã mô hình, ta có các quan hệ sau:

BODY (#IDB, X1, Y1, Z1, X2, Y2, Z2, IDB2)

BODY-CYL (#IDB, X1, Y1, Z1, X2, Y2, Z2, IDN, RA, HE, CYL\_TYPE, , IDB2)

BODY-PYR (#IDB, X1, Y1, Z1, X2, Y2, Z2, IDS, HE1, HE2, IDB2)

BODYFACE (#IDB, #IDF)

SURFACE (#IDS, DESC)

SUR-PL-NODE (#IDS, # IDN, SEQ)

LINE (#IDL, DESC)

LINENODE (#IDL, # IDN, SEQ)

NODE (#IDN, X, Y, Z)

POINT (#IDP, DESC, IDN)

FACE (#IDF, IDN1, IDN2, IDN3, IDS, IDB)

### 3. Thực nghiệm

Phần 3 là đoạn chương trình và hình ảnh minh họa cho việc cài đặt cơ sở dữ liệu

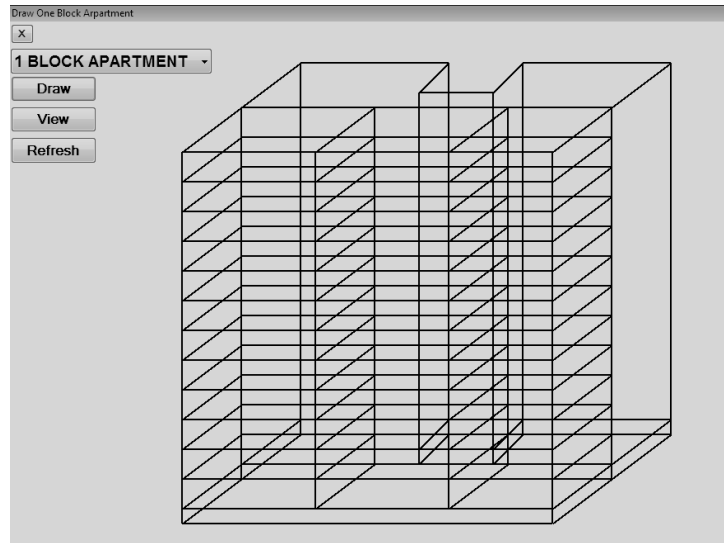
mới sau các cải tiến. Hệ quản trị CSDL được sử dụng là Oracle 10g, ngôn ngữ minh họa là VB.NET. Dữ liệu phục vụ cho việc minh họa là một chung cư 10 tầng, mỗi tầng ở phía trước có 3 căn hộ. Đoạn mã tạo bảng BODY-PYR chứa các căn hộ có hình lăng trụ đứng là:

```
create table BODY-PYR
(
  idb      number primary key,
  name     varchar2(50), height1 number,
  idb2     number , shape  mdsys.sdo_geometry );
```

Đoạn mã minh họa việc thêm dữ liệu vào BODY-PYR:

```
insert into  BODY-PYR  values(150,'1 cylinder block', 26, 235,
mdsys.sdo_geometry (2003, null, null, mdsys.sdo_elem_info_array(1,1003,1),
mdsys.sdo_ordinate_array(25,0, 33,6, 23,6, 21,4, 16,4, 18,6, 8,6, 0,0, 25,0)));
```

Đọc dữ liệu từ Oracle 10g và hiển thị lên màn hình minh họa bởi hình 9 bằng VB.NET



**Hình 9.** Chung cư 12 tầng minh họa bởi VB.NET

#### 4. Kết luận

Trên đây, bài báo đã trình bày tóm tắt các mô hình dữ liệu GIS 3D, mô tả chi tiết về mô hình UDM, một mô hình hiện đại được nhúng cho các ứng dụng quản lý đô thị. Khi nhúng mô hình này vào một ứng dụng cụ thể: quản lý các khu dân cư mới, bài báo dựa vào một số đặc điểm ứng dụng kết hợp với 2 nhận định về mô hình UDM để đề nghị 3 cải tiến: 1. Cách biểu diễn mới với các Surface là mặt phẳng. 2. Cách biểu diễn mới các Body là hình trụ đứng và nằm. 3. Cách biểu diễn các Body là hình lăng trụ đứng. Đồng thời chứng minh các cải tiến này là hữu ích trên 4 khía cạnh:

- Rút gọn kích thước lưu trữ của dữ liệu không gian.
- Giảm chi phí xử lý việc hiển thị dữ liệu không gian.
- Hiển thị đúng kích thước của thể giới thực với các đối tượng là mặt phẳng, hình trụ, hình lăng trụ.
- Một số độ đo: thể tích, chiều cao, bán kính được lưu trữ tường minh hay được tính toán chính xác nhờ công thức đơn giản của toán học.

Các cải tiến cũng gợi ý, nếu các Body là các khối hình cầu, hình nón, hình nón cụt cũng nên có cách tiếp cận như các cải tiến 2, 3 thay vì sử dụng phương pháp cũ của UDM là tam giác hóa các bề mặt biên.

### **TÀI LIỆU THAM KHẢO**

- [1]. Alias Abdul-Radman, Morakot Pilouk, *Spatial Data Modelling for 3D GIS*, Springer, (2007), 24-43.
- [2]. Alias Abdul-Radman, *Developing 3D Topological Model for 3D GIS*, Reseach Project 12, 2005.
- [3]. Ale Raza, *Object Oriented Temporal GIS for Urban Applications*, ITC Publication, 2001.
- [4]. Meng, X. and Liu, G., *Development of 3D GIS modeling Technology*, IFIP International Federation for Information Processing, Volume 259, 2008.
- [5]. Philip J. Schneider, *Geometric Tools for Computer graphics*, Morgan Kaufmann Publisher, 2003.
- [6]. Peter Van Oosterom, Jantien Storer, Wilko Quak, Sisi Zlatanova, *The balance between Geometry and Topology*, Proceedings of 10th international symposium on spatial data handling, Canada, 2002.
- [7]. Volker Coors, *3D-GIS in networking environments*, Computers, Environment and Urban Systems, (2003), 345-357.
- [8]. Zlatanova, S, PhD Thesis, *3D GIS for Urban Development*, ITC The Netherlands (2000), 71-107.
- [9]. Zlatanova, S & A.A. Rahman & W. Shi, *Topology models and framework for 3D spatial objects*, Journal of Computer & Geosciences, 2004.
- [10]. Zlatanova, S. A & A. Rahman And W. Shi, *Topology for 3D spatial objects*, Journal of Geospatial Engineering, 2002.

# INNOVATING URBAN DATA MODEL FOR NEW URBAN AREA MANAGEMENT

Nguyen Gia Tuan Anh  
Hung Vuong University

## SUMMARY

*To increase the effectiveness of the query, we need simple storage method, reducing the number of access I/O, the operations unnecessary for data presenting. Urban data model, when implemented on a specific application: Spatial data representing for the new urban area has some disadvantages: size of data storage, the number of nodes and lines are drawn but they are not in reality. To overcome the disadvantages, we propose some improvements on the UDM: to reorganize the structure of data for a number of objects 3D, 2D. The proposal based on comments on mathematics and practice of objects 3D, 2D in application: Spatial data representing for the new urban area. The proposals are proved by mathematics that they are better than the old model in storage and representation of Surface and Body objects. The paper describes a demo program with new data structure using VB.NET to show 3D objects.*

**Keywords:** 3D, GIS, data model.