Представяне на тримерни (3D) обекти

Основни проблеми при създаването на 3D графични обекти:

- Представяне на 3D обекти (модели на 3D обекти)
- Геометрични трансформации на 3D обекти
- Построяване на изгледа
- Проекции
- Определяне на видимите части на 3D обекти
- Алгоритми за изобразяване на повърхности (*surface rendering*)

1 Основни методи за представяне на 3D графични обекти

- Равнинни и квадратични повърхнини
- Сплайнови повърхнини
- Процедурни методи – фрактали и моделиране на поведението на частици (particle systems)
- Методи на физическо моделиране

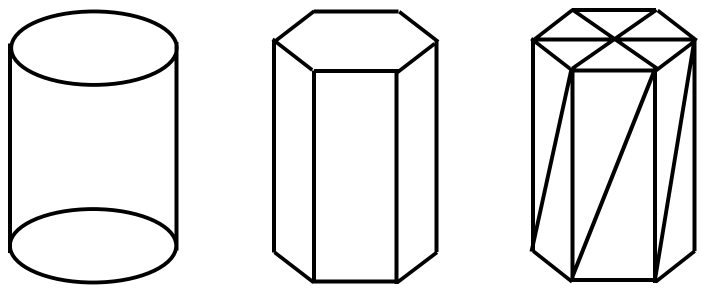
Методите се делят на две основни категории:

1) Представяне на границите на обекта - 3D обект се описва като множество от повърхнини, които отделят вътрешността на обекта от околната среда;
2) Представяне с разбиване на пространството, заемано от обекта, на множество от малки и непокриващи се твърди тела.

2 Плътни многостенници

3D обект се описва като множество от стени, всяка представляваща равнинен многоъгълник – “стандартен графичен обект”.

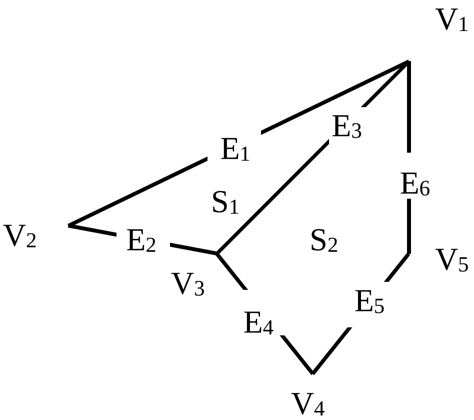
- При геометрични многостенници (призми, пирамиди,...) описанието е математически точно;
- При други повърхности се прави апроксимация с многоъгълници – мрежа от многоъгълници (*polygon mesh*).



2.1 Структури от данни за представяне на многостенници

Използват се два типа данни:

- Геометрични – координати на върхове и топологични данни за свързването им в ръбове и стени
- Атрибутни – параметри, определящи отражателните способности на повърхността, степента на прозрачност, текстура и др.



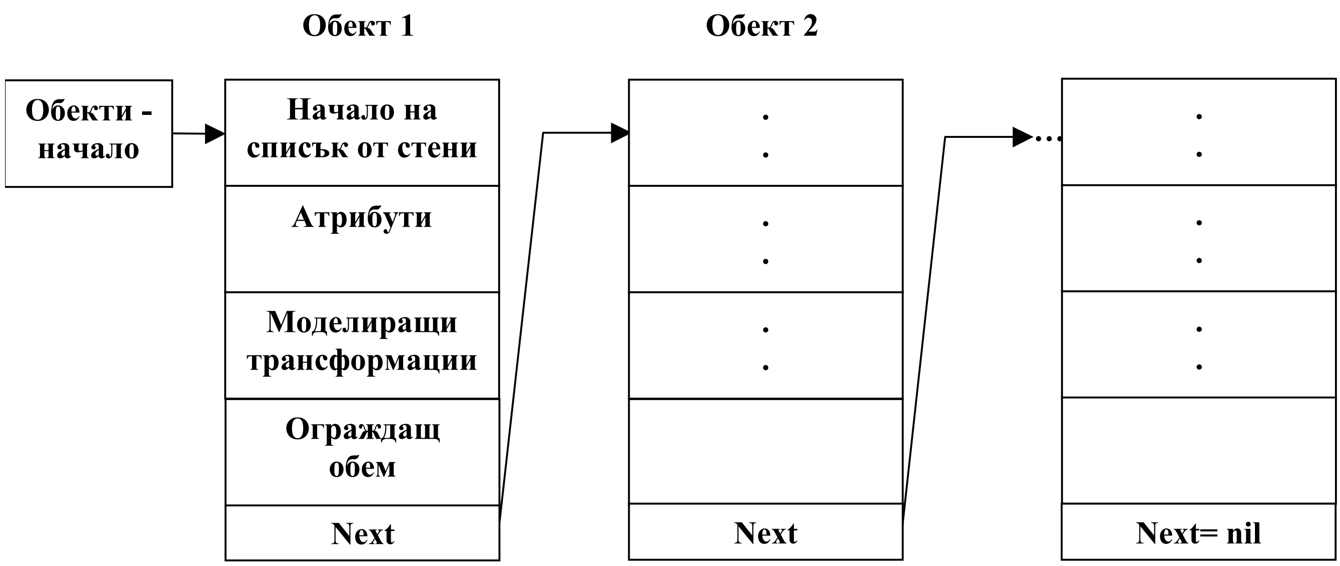
Върхове
$V_1: \; x_1, y_1, z_1$
$V_2: \; x_2, y_2, z_2$
$V_3: \; x_3, y_3, z_3$
$V_4: \; x_4, y_4, z_4$
$V_5: \; x_5, y_5, z_5$

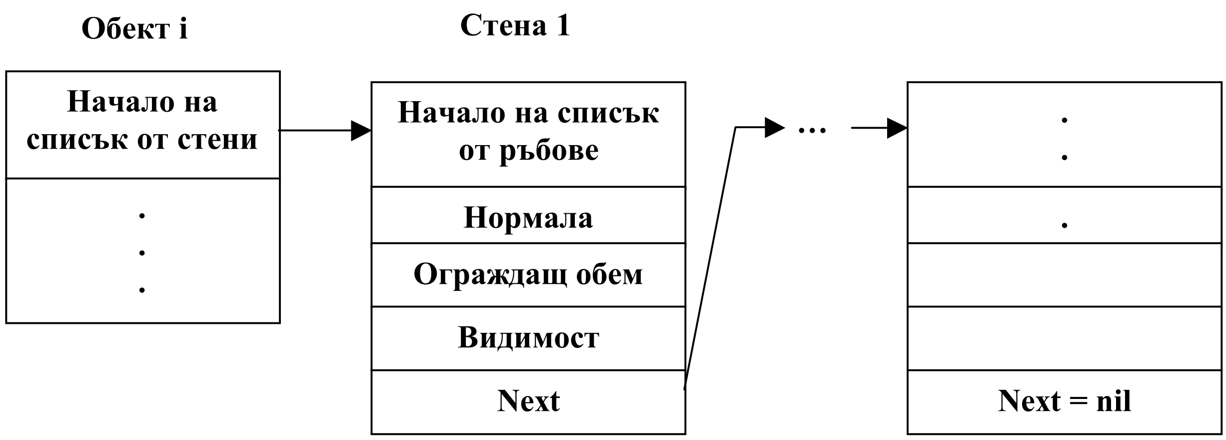
Ръбове
$E_1: \; V_1, V_2$
$E_2: \; V_2, V_3$
$E_3: \; V_3, V_1$
$E_4: \; V_3, V_4$
$E_5: \; V_4, V_5$
$E_6: \; V_5, V_1$

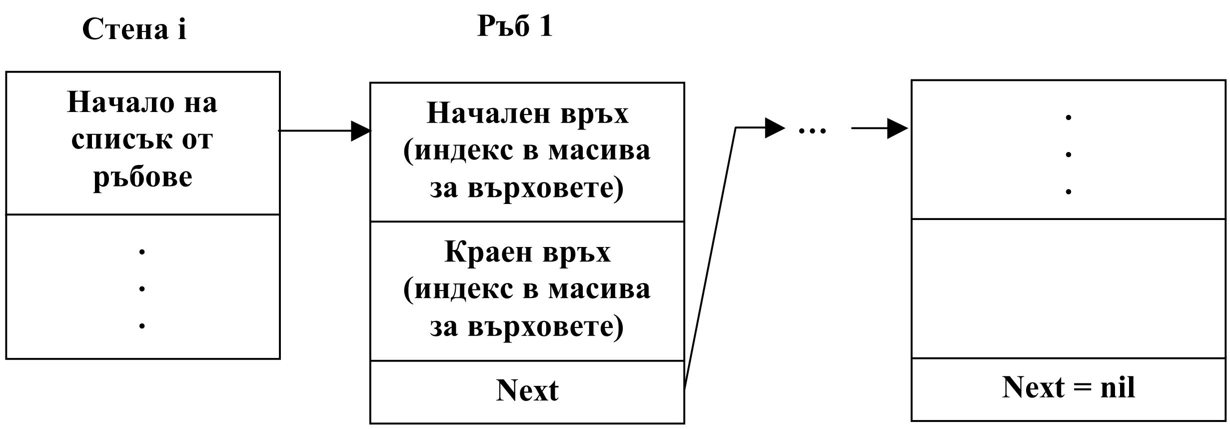
Стени
$S_1: \; E_1, E_2, E_3$
$S_2: \; E_3, E_4, E_5, E_6$

Масив за върховете

Връх 1		Връх n
(x,y,z) в моделираща координатна система	…	
(x,y,z) в световна координатна система	…	
(x,y) в екранни координати	…	







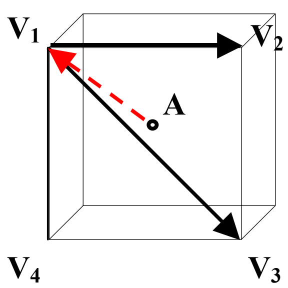
Ако страните са само **триъгълници**, всяка стена може да се опише с масив от 3 точки – полето за “начало на списък от ръбове” да се замени с масив от 3 елемента, цели числа, индекси в масива за върховете.

Примерна структура на файл за съхраняване на данни за обект:

```
120 240                          { Брой върхове, брой стени          }
0.90000   0.00000   0.00000     { (x,y,z) координати на всеки връх }
0.84271   0.17634   0.00000
0.69271   0.28532   0.00000
0.50729   0.28532   0.00000
...
1    2    12                     { Стена 1 – върхове на стени като  }
1    12   11                     { индекси в масив за върховете     }
2    3    13
...
```

2.2 Външни нормали на многостенници

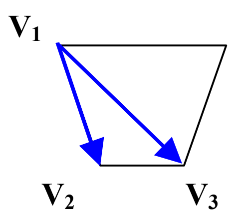
2.2.1 Изпъкнали многостенници



1. Избират се 3 последователи върха от стена $V_1,\ V_2,\ V_3$;
2. Изчислява се нормала в точка V_1:
   $$n = (V_2 - V_1) \times (V_3 - V_1);$$
3. Изчислява се вътрешна точка на тялото $A = \frac{1}{k}\sum_{i=1}^{k} V_i$, където k е броят на върховете на тялото;
4. Изчислява се вектор, насочен извън тялото и минаващ през точка V_1:
   $$V = V_1 - A$$
5. Ако $n.V > 0$, то n е външна нормала, в противен случай $(-n)$ е външна нормала.

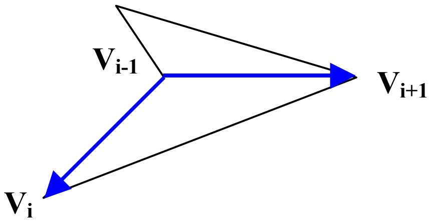
2.2.2 Произволни мрежи от многоъгълници

Върховете на всяка стена се номерират по посока, обратна на часовниковата стрелка, когато стената се наблюдава откъм външната страна.



Тогава 3 последователи върха $V_1,\ V_2,\ V_3$ определят външна нормала
(при **изпъкнал** многоъгълник):

$$n = (V_2 - V_1) \times (V_3 - V_1)$$

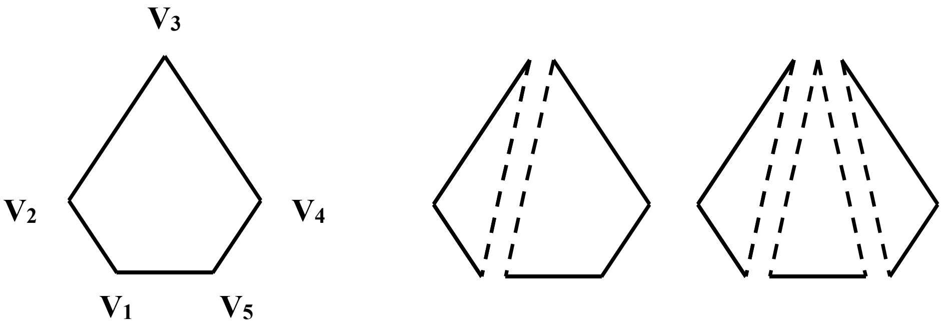


Ако многоъгълникът е **вдлъбнат**, може да се намери най-левият му връх V_i и двата съседни V_{i-1} и V_{i+1} и тогава външната нормала в точка V_{i-1} е:

$$n = (V_i - V_{i-1}) \times (V_{i+1} - V_{i-1})$$

2.3 Разделяне на стените на триъгълници (triangulaion)

Изпъкнали многоъгълници



Върховете $V_1, V_2, ..., V_n, V_{n+1} (= V_1)$ се обхождат по тройки от съседни върхове.

Пример:

$V_1, V_2, ..., V_5, V_6 (= V_1)$

V_1, V_2, V_3 образуват триъгълник, V_2 се изключва от списъка на върховете

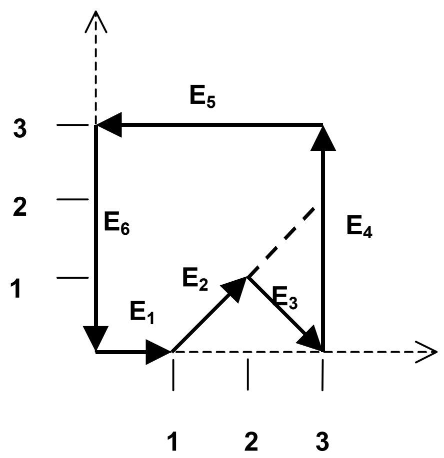
V_3, V_4, V_5 образуват триъгълник, V_4 се изключва от списъка на върховете

V_3, V_5, V_6 образуват триъгълник, V_5 се изключва от списъка на върховете

Остават V_3 *и* V_6, край

2.4 Разделяне на вдлъбнати многоъгълници

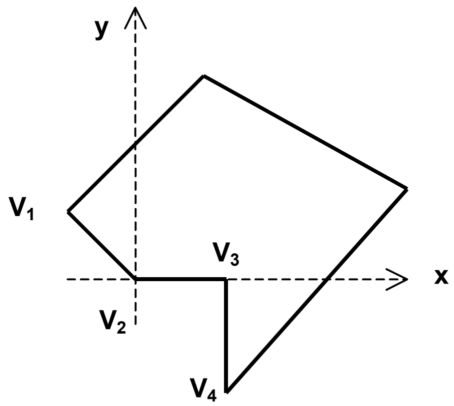
2.4.1 Векторен метод за разделяне на вдлъбнати многоъгълници



$$E_1 = (1,0,0) \qquad E_2 = (1,1,0)$$
$$E_3 = (1,-1,0) \qquad E_4 = (0,3,0)$$
$$E_5 = (-3,0,0) \qquad E_6 = (0,-3,0)$$
$$z_{ij} = E_{ix}E_{jy} - E_{jx}E_{iy}$$

$E_1 \times E_2 = (0,0,1)$; $E_2 \times E_3 = (0,0,-2)$ $\Rightarrow$ Вдлъбнат, разделя се по продължението на E_2 .

2.4.2 Ротационен метод за разделяне на вдлъбнати многоъгълници



- Обхождат се ръбовете по посока, обратна на часовниковата стрелка;
- Всеки връх V_i се пренася в т.О;
- Ротация по посока на часовниковата стрелка, така че връх V_{i+1} да легне на оста x ;
- Ако следващият връх V_{i+2} е под оста x , многоъгълникът е вдлъбнат и се разделя на две по оста x ;
- Получените многоъгълници се подлагат отново на теста.

2.5 Алгоритми за многостенници

2.5.1 Тест за принадлежност на точка в изпъкнал обем

Тялото се представя чрез множество от равнинни многоъгълници, всеки зададен с координатите на върховете си.

```
Inside := true;
for всяка равнина от тялото do
  begin
  Избират се три върха V1, V2, V3 от равнината
  Изчислява се нормалата n=(V2-V1)x(V3-V1)
  Избира се връх Vi от произволна друга равнина
  if SIGN(n.(P-V1)) <> SIGN(n.(Vi-V1))
    then Inside := false;
  end;
```

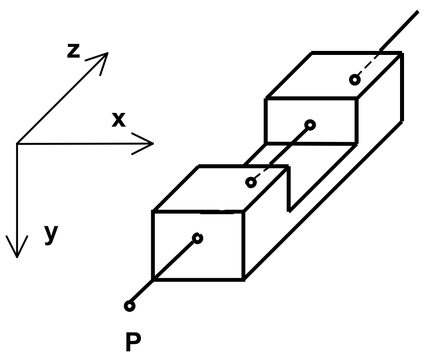
2.5.2 Изрязване на отсечка спрямо изпъкнал обем

Зададени са:

- Отсечка с двете си крайни точки P1 и P2
- Тяло, оградено от равнинни многоъгълници в масива Face
- Функция OUTSIDE(Точка, Равнина), определяща дали точката е външна спрямо равнината
- Функция INTERSECTION(Точка1, Точка2, Равнина), определяща точка на пресичане на равнината и правата, минаваща през двете точки P1 и P2

```
Clipped := false; i:=1;
repeat
  if OUTSIDE(P1, Face[i]) and OUTSIDE(P2, Face[i])
      then Clipped := true
  else if  not OUTSIDE(P1, Face[i]) and OUTSIDE(P2, Face[i])
                 then P2 := Intersection(P1, P2, Face[i])
    else if OUTSIDE(P1, Face[i]) and  not OUTSIDE(P2, Face[i])
                 then P1 := Intersection(P1, P2, Face[i]);
  i:=i+1;
until Clipped or (i>броя на равнините);
{ В P1,P2 остава частта от отсечката, която лежи изцяло вътре в изпъкналия обем}
```

2.5.3 Тест за принадлежност на точка в неизпъкнал обем



```
Избира се права P+tk през точка P
(k е единичен вектор по положителната посока на оста z)
NoOfIntersections := 0;
for всяка стена от тялото do
  begin
  Изчислява се t за точката на пресичане на правата P+tk и равнината на стената
  if t>0 then
    begin
      Изчисляват се координатите на точката на пресичане
      Проверяват се [x,y] координатите на точката на пресичане
      за включване в многоъгълника, определен от [x,y] координатите на стената
      if Точката е вътре в многоъгълника then
          inc(NoOfIntersections);
    end;
  end;
Inside := ODD(NoOfIntersections);
```

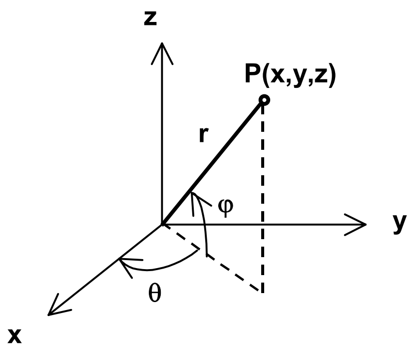
2.5.4 Изрязване на отсечка спрямо неизпъкнал обем

Отсечката е зададена с крайните си точки P_1 и P_2, уравнението й е $L = P_1 + t(P_2 - P_1)$.

- Намират се точките на пресичане на отсечката и стените и се подреждат по нарастващи стойности на t : $Q_1, Q_2, ..., Q_k$
- Ако t е нечетно, P_1 е вътре в обема, P_1Q_1 е вътре в обема, Q_1Q_2 е вън от обема, Q_2Q_3 е вътре и т.н.

3 Квадратични повърхнини

3.1 Сфера



Аналитично уравнение:

$$x^2 + y^2 + z^2 = r^2$$

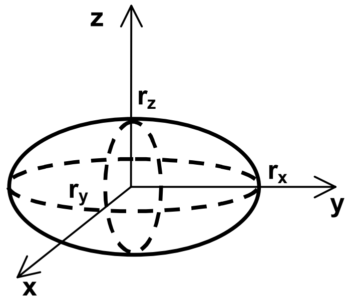
Параметрично уравнение:

$$x = r\cos\varphi\cos\theta \qquad -\pi/2 \le \varphi \le \pi/2$$

$$y = r\cos\varphi\sin\theta \qquad -\pi \le \theta \le \pi$$

$$z = r\sin\varphi$$

3.2 Елипсоид



Аналитично уравнение:

$$\left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2 + \left(\frac{z}{r_z}\right)^2 = 1$$

Параметрично уравнение:

$$x = r_x \cos\varphi \cos\theta \quad -\pi/2 \le \varphi \le \pi/2$$

$$y = r_y \cos\varphi \sin\theta \quad -\pi \le \theta \le \pi$$

$$z = r_z \sin\varphi$$

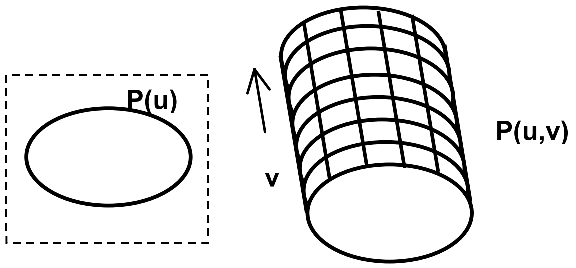
3.3 Хиперболоид

$$\left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2 - \left(\frac{z}{r_z}\right)^2 = 1$$ - еднополюсен

$$\left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2 - \left(\frac{z}{r_z}\right)^2 = -1$$ - двуполюсен

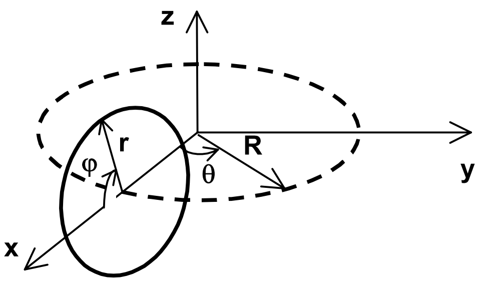
4 Омитащи повърхнини

4.1 Създаване на омитащи повъхнини чрез транслация



4.2 Създаване на омитащи повърхнини чрез ротация

Пример: Създаване на тор чрез ротация на малка окръжност около ос по периметъра на друга окръжност



Параметрични уравнения:

$$x = (R + r\cos\varphi)\cos\theta$$

$$y = (R + r\cos\varphi)\sin\theta$$

$$z = r\sin\varphi$$