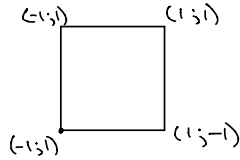


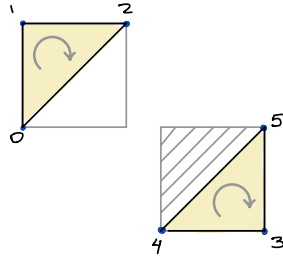
Geometry
(способ задания топологии
графиков примитива)



① **Triangle List** (перечисл. вершин по часовой стрелке)

Vertex Buffer

VB ind: 0	(-1, -1)
1	(-1, 1)
2	(1, 1)
3	(1, -1)
4	(-1, -1)
5	(1, 1)



② **Indexed Triangle List** (экономит по памяти, т.к. ind - 4 байт)

Index Buffer

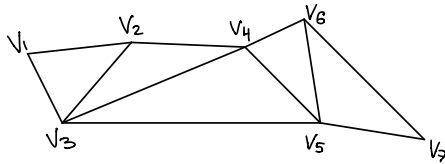
IB ind: 0	0
1	1
2	2
3	3
4	0
5	2

Vertex Buffer

VB ind: 0	(-1, -1)
1	(-1, 1)
2	(1, 1)
3	(1, -1)

③ **Strip**

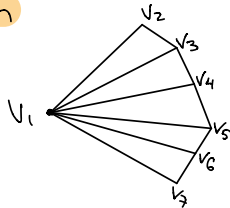
послед. вершин сама опр. мн-во Δ



1. первые 3 верш. — первый Δ
2. отбр. первую верш.
3. след. 3 верш. — второй Δ и т.д.

важно помнить, что меняется порядок обхода вершин

④ **Fan**



алгоритм как у Strip, только начинаем отбрасывать с V_2 .

Материал можно описать:

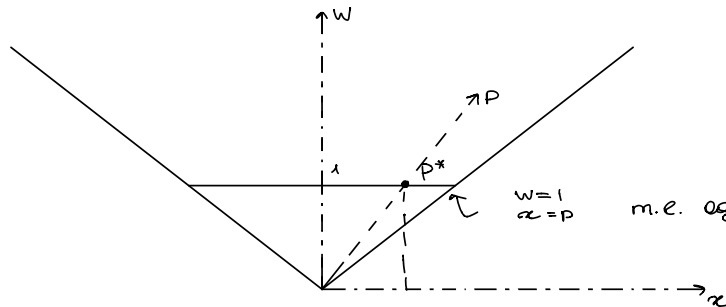
- цветом
- текстурой
- коэф. освещённости

Формат вершин:

- position
- normal (нормаль пов-ти в данной точке) (нормаль ешё при осв-ти)
- color (diffuse)
- color (specular)
- texture coordinate (n)

Transformation

в к.т. исп. 4-х мерная система координат - **однородная система к.** (homogeneous)



выполняется центральное проецирование

$w=1$
 $x=p$ т.е. одна координ.

z координ - глубина

Переход из 4-х мерного пр-ва в 3-х мерное:

$$P = [x, y, z, w]$$

$$P' = [x/w, y/w, z/w, 1]$$

z-buffer работает с этой координ.

Translate (T):

матрица сдвига

$$\begin{bmatrix} x' & y' & z' & 1 \end{bmatrix} = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ T_x & T_y & T_z & 1 \end{bmatrix}$$

умножение вектора на вектор вогн. за один такт

Scale (S)

$$\begin{bmatrix} x' & y' & z' & 1 \end{bmatrix} = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotate (R_x)

вращ.

$$\begin{bmatrix} x' & y' & z' & 1 \end{bmatrix} = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

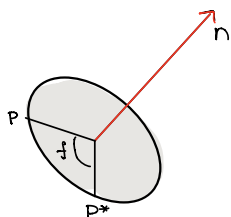
$$R = R_z * R_y * R_x$$

$$W = R * S * T$$

Угол Эйлера для поворота не опр. однозначно вращ.

Однозначно вращение опр. 4-ми углами.

- Нужно:
- указать ось, вокруг которой вращать (вектор \Rightarrow 3 числа)
 - угол (4 числа)



Но вообще используют **кватернион**

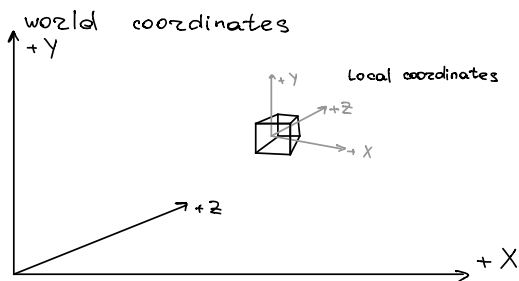
$$q = (\bar{n}, f) = (\bar{n} \cdot \sin(f/2), \cos(f/2))$$

$$\bar{n} = [x \ y \ z]$$

$$\begin{bmatrix} 1 - 2 \cdot (y^2 + z^2) & 2 \cdot (x \cdot y - w \cdot z) & 2 \cdot (x \cdot z + w \cdot y) \\ 2 \cdot (x \cdot y - w \cdot z) & 1 - 2 \cdot (x^2 + z^2) & 2 \cdot (y \cdot z - w \cdot x) \\ 2 \cdot (x \cdot z + w \cdot y) & 2 \cdot (y \cdot z - w \cdot x) & 1 - 2 \cdot (x^2 + y^2) \end{bmatrix}$$

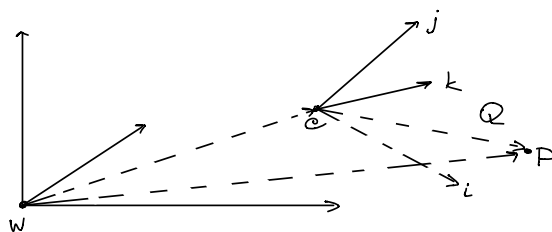
матр. вращ. по
заданному кват.-ну

Преобразование в с.к. камеры



$$P_w = P_c + W$$

мировые с.к. и мест. с.к. камеры



$$P = C + Q$$

$$P = C + Q_x i + Q_y j + Q_z k$$

$$W = \begin{bmatrix} i_x & i_y & i_z & 0 \\ j_x & j_y & j_z & 0 \\ k_x & k_y & k_z & 0 \\ C_x & C_y & C_z & 0 \end{bmatrix} \text{ вектора лок. сист. к.}$$

$$P = Q * W$$

преоб. из лок. в мир.

из мод. в ок:

$$V = W^{-1}$$

$$W = \begin{bmatrix} R & 0 \\ c & 1 \end{bmatrix}$$

$$W^{-1} = \begin{bmatrix} R^{-1} & 0 \\ -cR^{-1} & 1 \end{bmatrix}$$

Projection

