



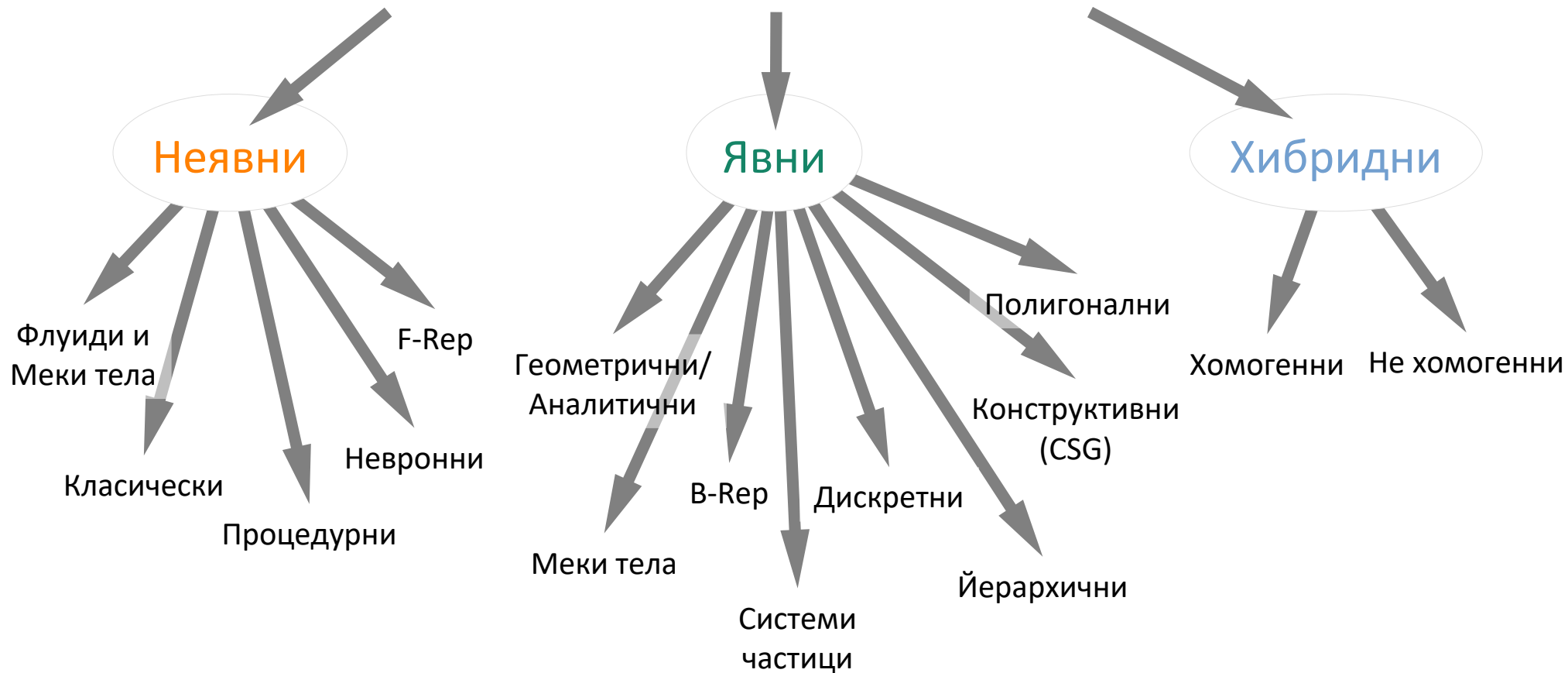
Компютърна Графика и Презентации

Представящи схеми

доц. д-р Александър Пенев

Класификация

Класификация на представящите схеми в КГ



Представящи Схеми



Схеми на Представяне

- ❖ Екземпляри на чисти примитиви (Parameterized primitive instancing);
- ❖ Изброяване на заетото пространство (Spatial occupancy enumeration);
- ❖ Разбиване на клетки (Cell decomposition);
- ❖ Кодиране с осмично дърво (Octree);
- ❖ Конструктивна геометрия (CSG);
- ❖ Описание на границата (V-Rep);
- ❖ Полигонални (Polygonal – meshes);
- ❖ Облаци от точки (3D Point clouds);
- ❖ Параметрични (NURBS, Bezier, B-Splines,...);
- ❖ Неявни / Функционално представяне (F-Rep, SDF, ...);
- ❖ Запълващи продукции (Sweeping);
- ❖ Метатопчета (Metaballs);
- ❖ Системи частици (Particle systems);
- ❖ Невронни (NeRF, DeepSDF, ...);
- ❖ Флуиди и Меки тела (Soft body);
- ❖ Йерархични – Скелет и др. (Armature)
- ❖ Хибридни схеми.



Екземпляри на чисти примитиви

(Parameterized primitive instancing)



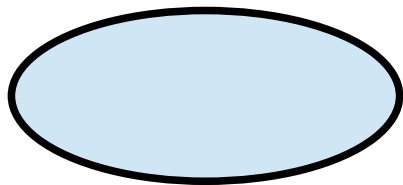
Екземпляри на чисти примитиви

Тези схеми на представяне се основават на идеята за фамилия от обекти, като всеки член на фамилията се различава от останалите чрез стойностите на няколко параметъра.



Екземпляри на чисти примитиви

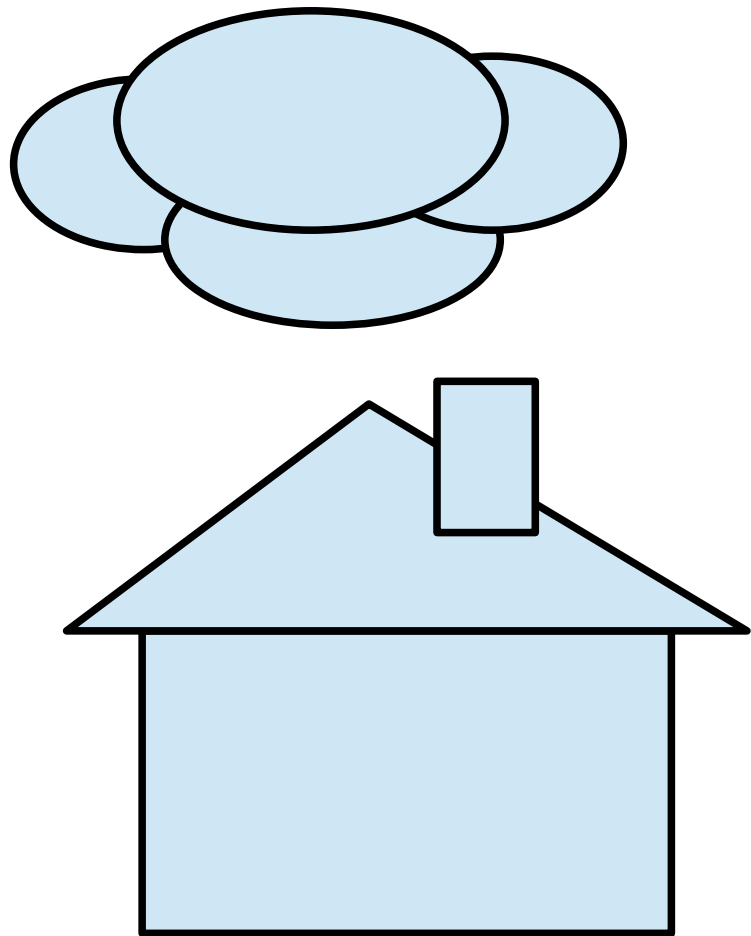
- ❖ Генериращ примитив – всяка фамилия от обекти. Представя се като име последвано от N-торка от променливи величини;
- ❖ Индивидуален обект – всеки обект от фамилията се нарича екземпляр на примитива и се представя като N+1-торка от стойности (или име последвано от N стойности).



ELLIPSE(A, B)

(ELLIPSE, 5, 2) или ELLIPSE(5, 2)

Пример



Елипса (1.00, 3.00, 4.44, 5.26)

Елипса (3.00, 4.00, 7.44, 6.26)

Елипса (5.60, 2.70, 9.00, 5.00)

Елипса (2.37, 2.10, 7.50, 5.00)

Правоъгълник (2.70, 10.30, 9.70, 14.30)

Триъгълник (6.20, 7.30, 1.70, 10.30,
10.70, 10.30)

Правоъгълник (6.60, 7.00, 7.90, 9.00)

Недостатъци на схемата

- ❖ Липсата на средство за комбиниране на екземпляри на примитиви, с цел получаване на нови по-сложни обекти;
- ❖ Трудността на създаване на алгоритми за обработка на тези представяния, защото при добавяне на нов вид примитив, обикновено се налага преработката на всички алгоритми (но въпреки това в повечето случаи новите алгоритми са просто разширение на старите с поддръжка и на новия примитив).

В практиката

- ❖ Схемите са сбити и лесни за използване;
- ❖ Удобни само, ако дефиниционната им област може да бъде сведена до каталог от малко на брой фамилии;
- ❖ Използват се съвместно със схеми, базирани на някаква конструктивна операция;
- ❖ В 2D се използват правоъгълници, елипси, линии, многоъгълници и др. В 3D се използват паралелепипеди, елипсоиди, сфери, линии, равнини, цилиндри, конуси, пирамиди, многостени и др.

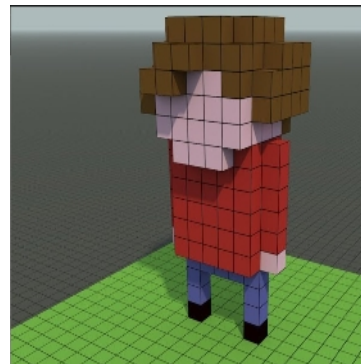
Изброяване на заетото пространство (Spatial occupancy enumeration)

Изброяване на заетото пространство

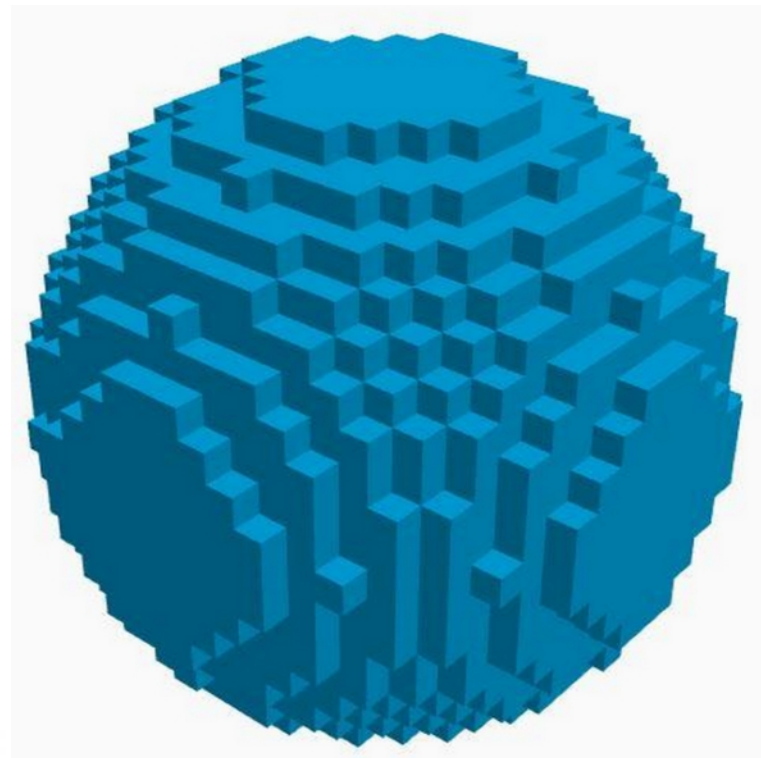
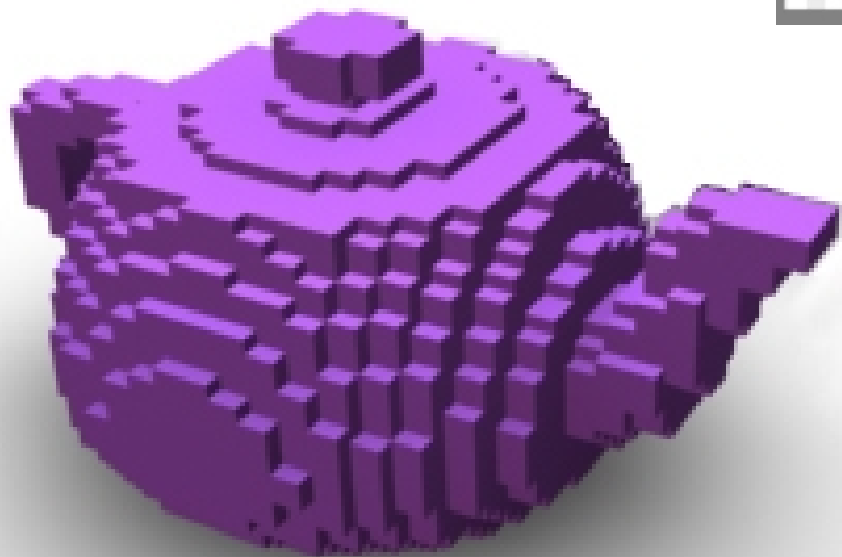
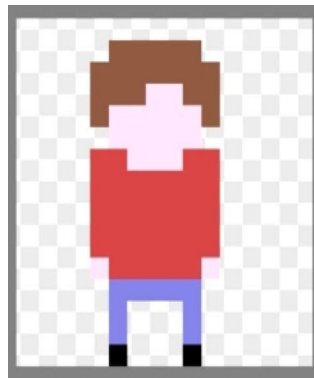
Всяко представяне на тяло, чрез тези схеми представлява списък на пространствени клетки, които тялото заема.

Пространствените клетки обикновено се наричат воксели
(voxel = **v**olumetric **pixel**).

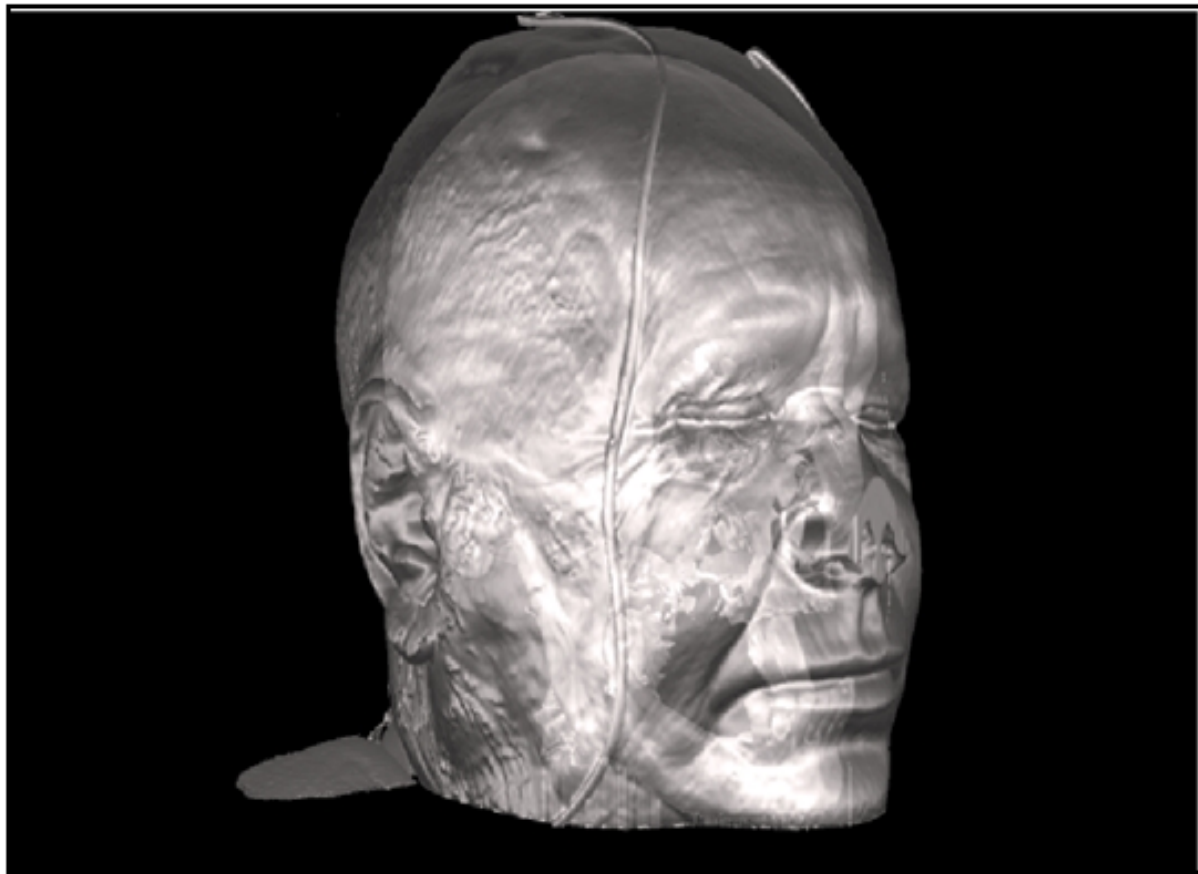
В двумерния (2D) случай клетките се наричат пискели
(pixel = **p**ictures **e**lement).



Примери



Пример – 3D изображение получено от томограф



Получените данни от томографа (благодарение на сложен физичен процес и математическа обработка) могат да се визуализират, да се правят различни разреди и анализи и др.

Свойства

- ❖ Кубове са наредени в пространствена мрежа (растер);
- ❖ Всяко представяне на тяло чрез такава схема е (ненаредено) множество от наредени тройка числа;
- ❖ Към тройката числа може да има и други визуални характеристики, като плътност, цвят и др. Ако се описват всички клетки в отнапред известната мрежа, може да се използва подхода (bitmap/битмап) при който координатите не се съхраняват, а се съхраняват само визуалните характеристики, но винаги в определена последователност;
- ❖ Тези схеми имат твърде много общи свойства със схемите, базирани на разбиване на телата на части с непресичаща се вътрешност.

Разбиване на клетки

(Cell decomposition)

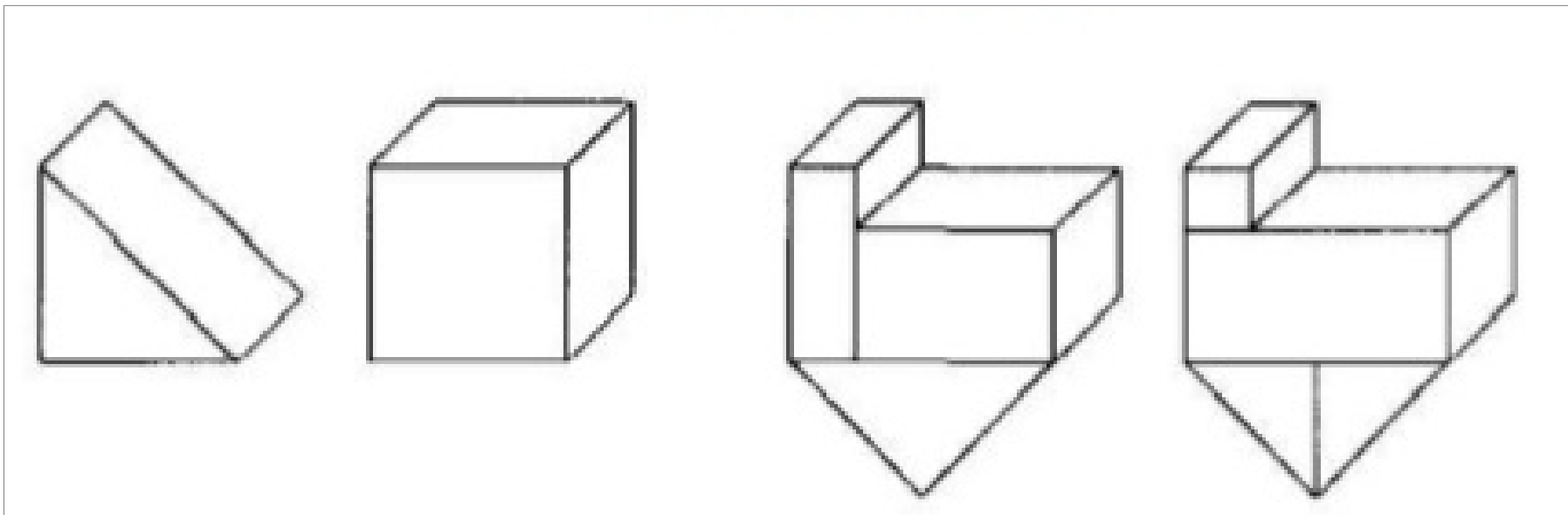
Разбиване на клетки

Всяко представяне на тяло, чрез тези схеми представлява списък на пространствени клетки, които тялото заема.

Пространствените клетки може да бъдат с различна форма и размери.



Пример



Обърнете внимание, че представянето с тези схеми може да не е еднозначно. Тялото в дясно на примера може да се разбие по много начини на различни клетки, но това обикновено не е проблем.

Свойства

- ❖ Схемите **Изброяване на заетото пространство** са частен случай на схемите **Разбиване на клетки**;
- ❖ Понеже клетките може да са с произволна форма, то за разлика от подхода в изброяване на заетото пространство, е необходимо да се описва геометрията на целите клетки (например всички върхове на многостена, вместо само центъра му). Това в повечето случаи води до използване на повече памет;
- ❖ Често за клетки се използват тетраедри;
- ❖ Използват се за създаване на специални ефекти, симулации и други.



Кодиране с осмично дърво

(Octree)

Кодиране с осмично дърво

Всяко представяне на тяло, чрез тези схеми може да се представи с дървовидна структура (осмично дърво), като всяко ниво в дървото може да се разгледа като различна детайлизация на модела.

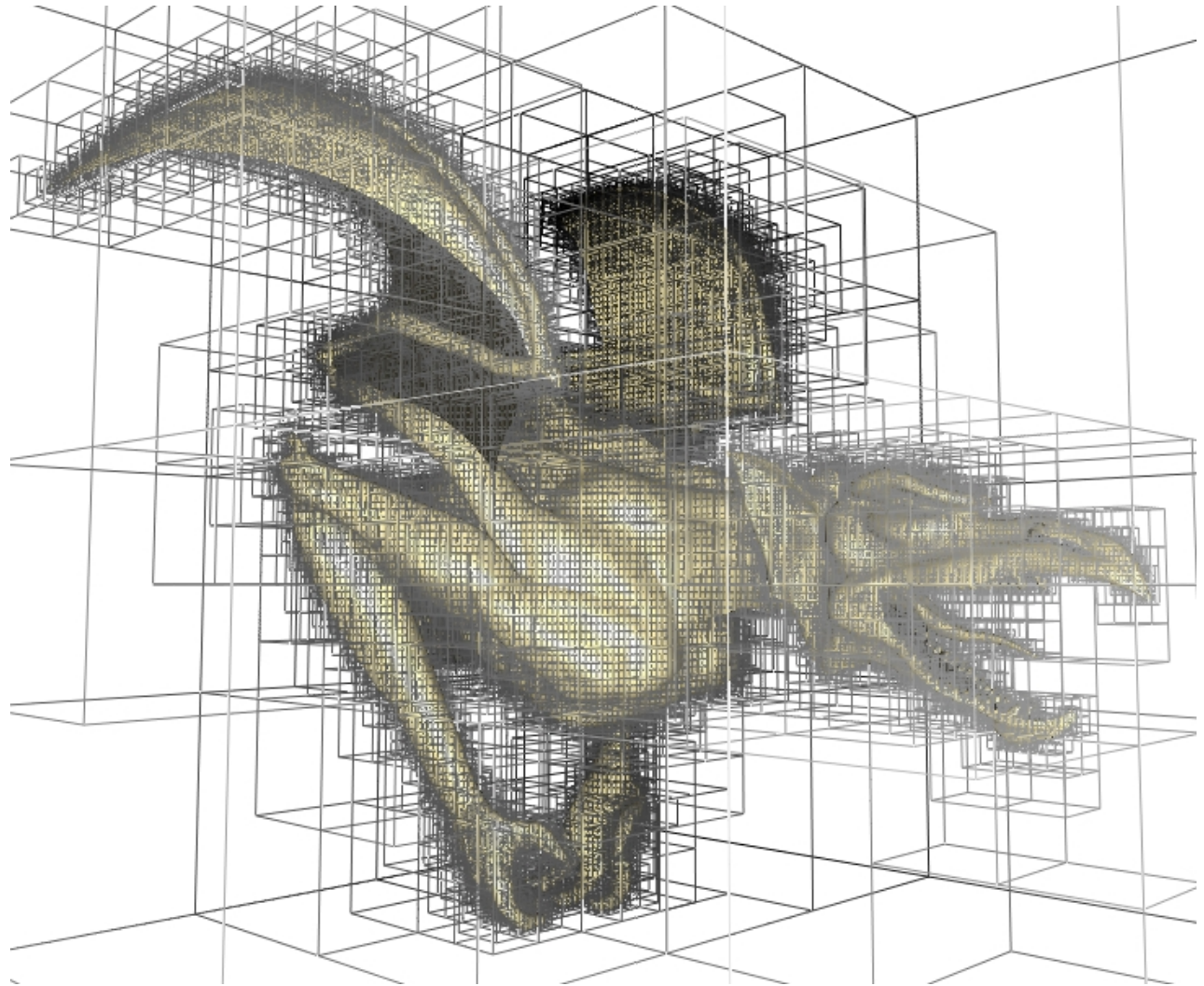
Всеки възел в дървото може да е:
Празен (**E**mpy), Запълнен (**F**ull) или
Частично запълнен (**P**artial).

E и **F** са листа в дървото, а **P** е под дърво.



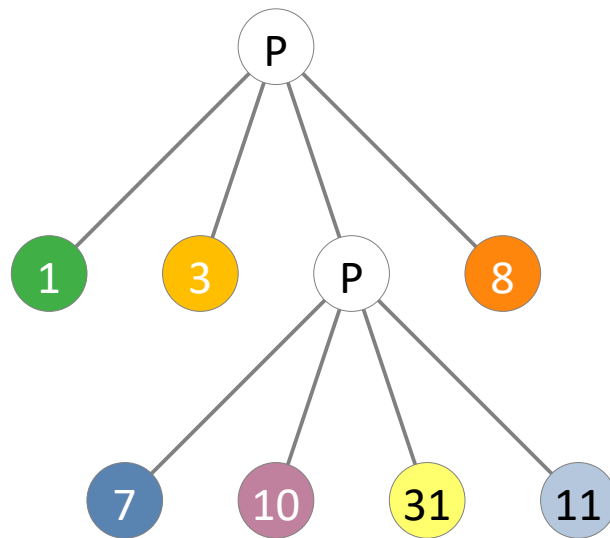
Пример

Описанието на тялото започва от един голям куб, след което той се дели на 8 октанта, всеки от които е или пълен **F** или празен **E** (при тези случаи приключваме описанието на съответния подкуб). В противен случай разделянето продължава.

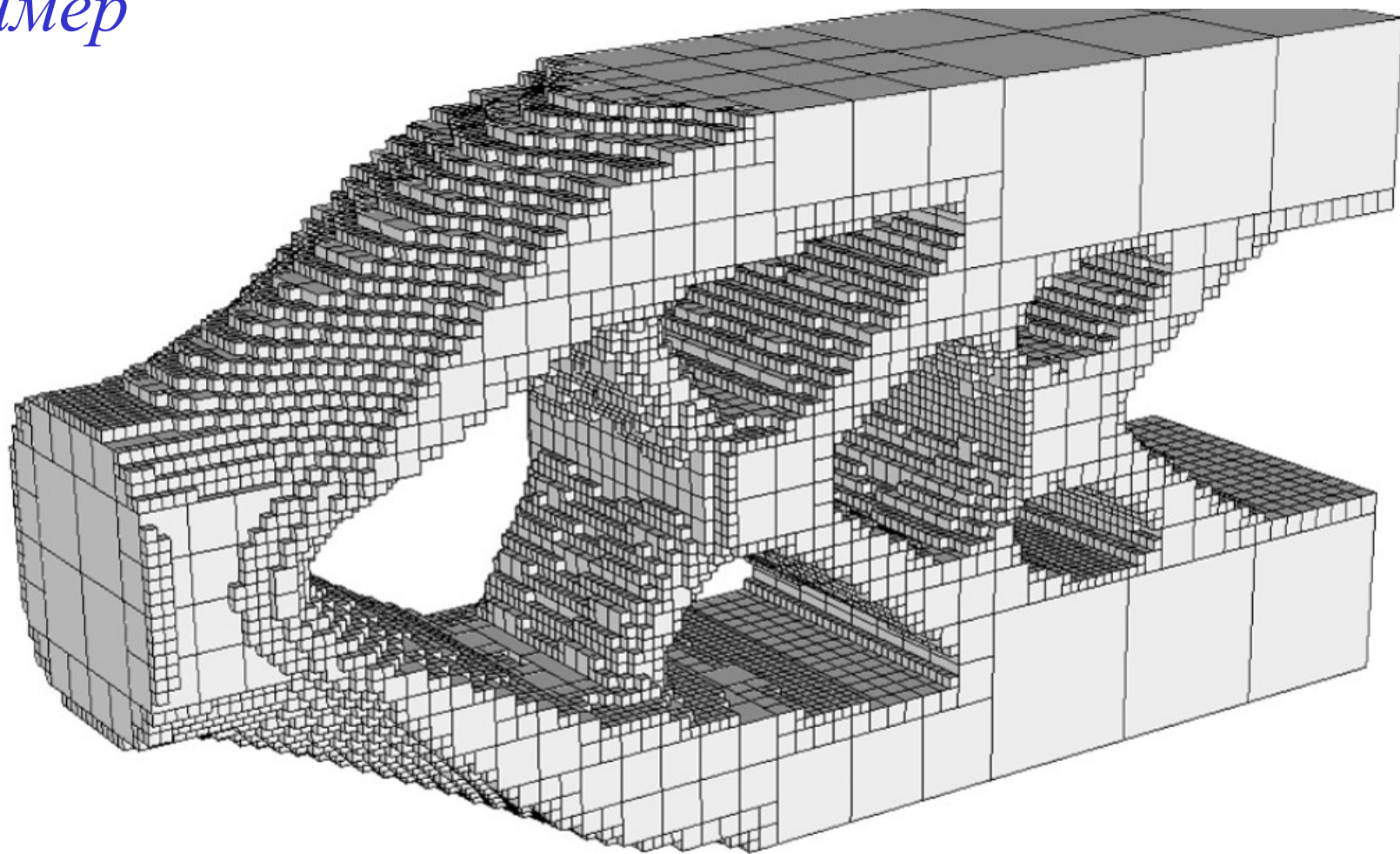


Пример (четвъртично дърво)

1	1	3	3
1	1	3	3
7	10	8	8
31	11	8	8



Пример



Свойства

- ❖ В двумерния вариант схемата се нарича кодиране с четвъртично дърво. Едномерният – двоично дърво.
- ❖ Обикновено представянето е по-сбито в сравнение с Изброяване на заетото пространство;
- ❖ Много често осмичното дърво (както и други подобни на него дървета) се използват за ускоряващи структури в някои алгоритми за визуализация и др.;

Конструктивна геометрия *(на твърди тела)* *(Constructive Solid Geometry, CSG)*

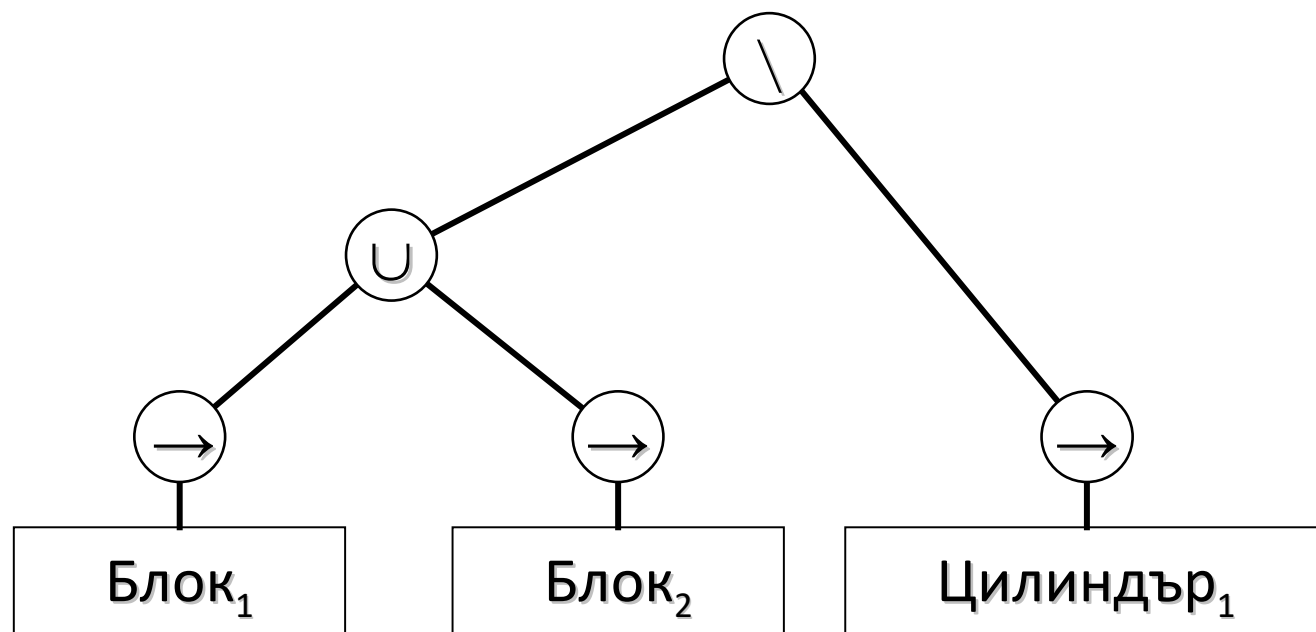
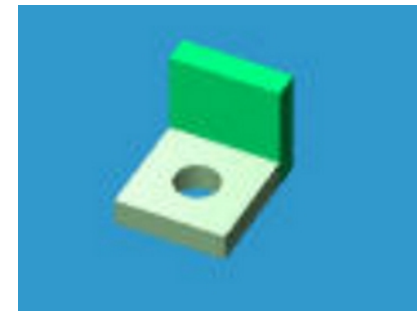
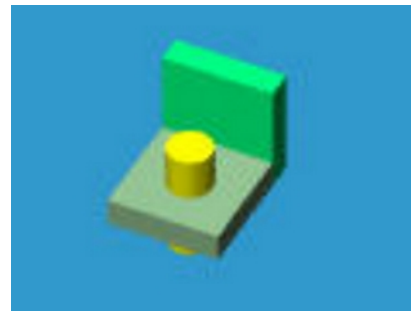
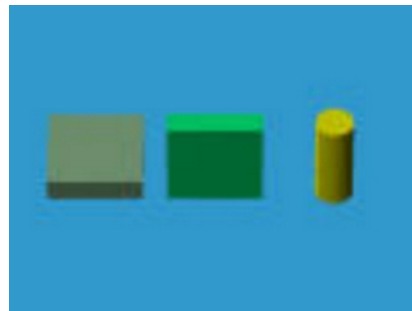
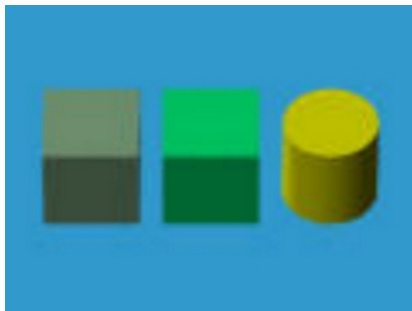


Конструктивна геометрия

Метод за моделиране на (твърди) тела, който комбинира използването на прости форми, наречени примитиви за изграждането на сложни модели посредством булевите оператори: Обединение, Разлика и Сечение.



Конструктивна геометрия на твърди тела



Конструктивно дърво:

Състои се от възли с примитиви, възли с трансформации и възли с теоретико-множествени операции.

Свойства

- ❖ За лесно генериране на описание се използва и конструктивно дърво;
- ❖ Схемите са подходящи за лесно създаване на модели от човек;
- ❖ Много често се използват в CAD/CAM системите;

Описание на границата

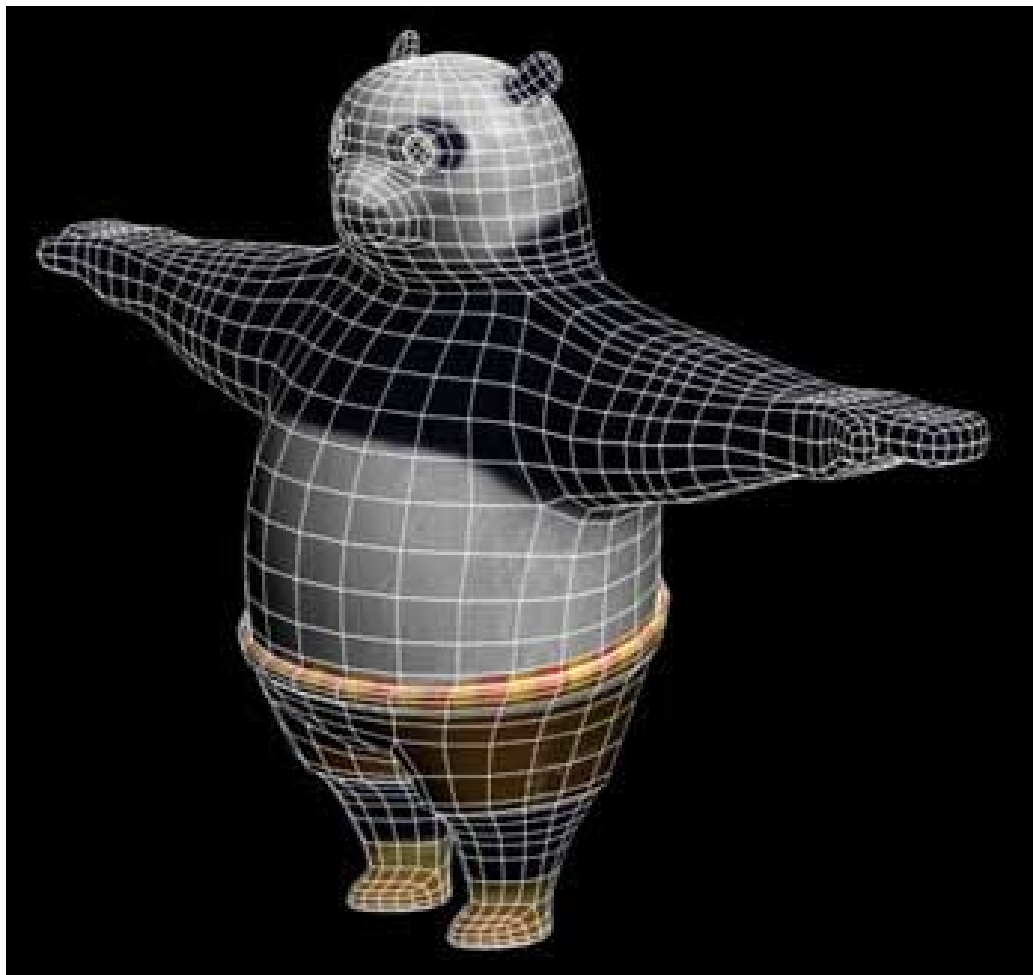
(Boundary Representation, B-Rep)

Описание на границата

Моделите се представят като обединение на някакви пространствени лица (наречени още „кръпки“), описващи границата на модела.

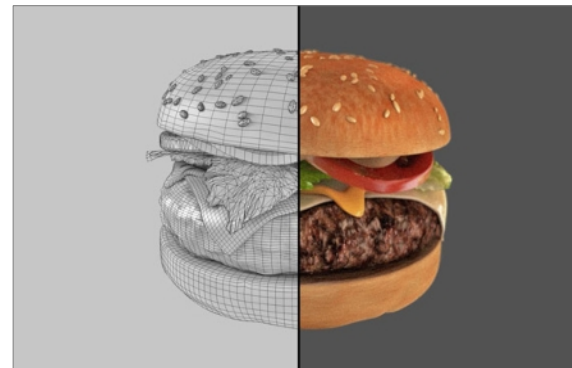
Най-често за такива лица се използват триъгълници или четириъгълници. Този подход е много често срещан подвид на описание на границата, наречен Полигонални модели.

Пример 1



Свойства

- ❖ Една от най-често използваните представящи схеми в момента;
- ❖ Схемите са подходящи за лесно създаване на модели от човек;
- ❖ Съществуват ефективни алгоритми за обработка и визуализация.



Полигонални

(Triangle Mesh, Quad Mesh, Subdivision surfaces, ...)



Полигонални

Полигоналните модели са важен вид „Описание на границата“. Използват т.нар. „mesh“-структури за създаване на 3D модел. Това са съвкупности от помалки компоненти като върхове, ръбове и лица (триъгълници, четириъгълници и други).

Основният компонент на меша е върхът – точка в тримерното пространство. Ръб се образува, когато два върха са свързани с права линия част от границата на обекта. Лице се описва като три или повече подходящо свързани ръба образувачи двумерна повърхност.



Предимства и Недостатъци

- ❖ Много лесни за използване в софтуерите за моделиране;
- ❖ Много ефективни и лесни за визуализация, като за подобен вид модели има хардуерно ускорени алгоритми, като например Z-Buffer и др.;
- ❖ Обикновено са необходими малко на брой полигони, за да се създадат разнообразни форми, особено ако формите не са криви повърхности.
- ❖ Моделирането с полигони отнема много време, особено за сложна геометрия;
- ❖ Използването на полигони в процеса на изграждане на модел не винаги е точно и води до груби приближения на създаваните модели или до необходимост прекомерно увеличаване на броя на полигоните (при криви, гладки повърхности).

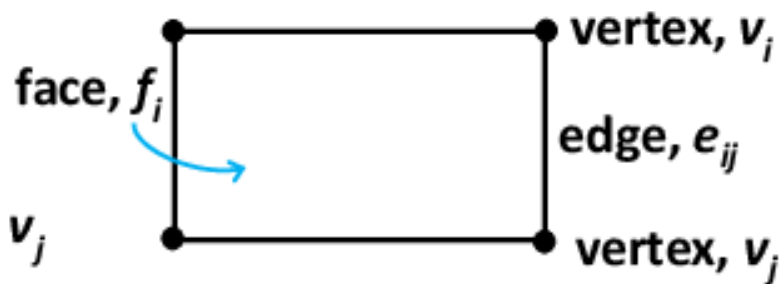


Триъгълен и четириъгълен меш

Triangular Face:



Quadrilateral Face:



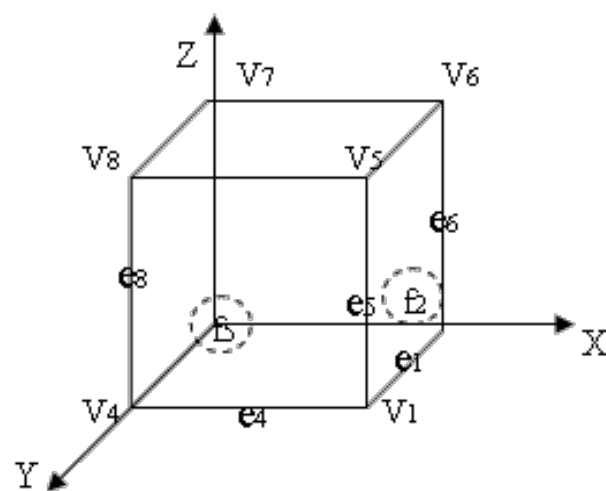
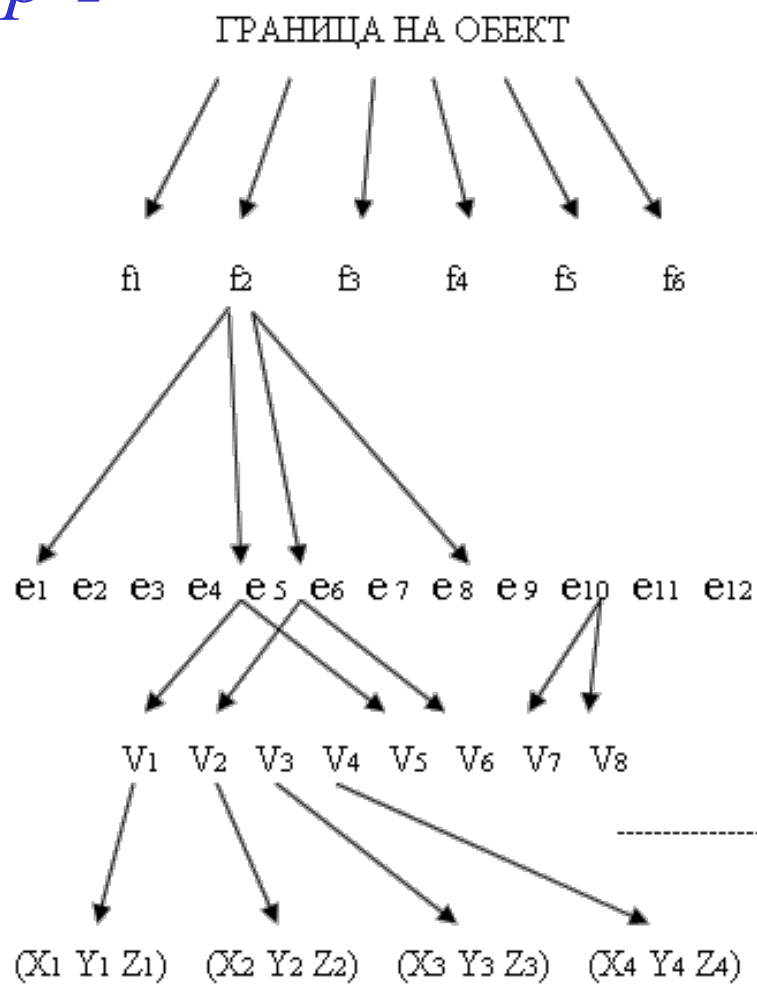
Triangular Mesh:



Quadrilateral Mesh:



Пример 1



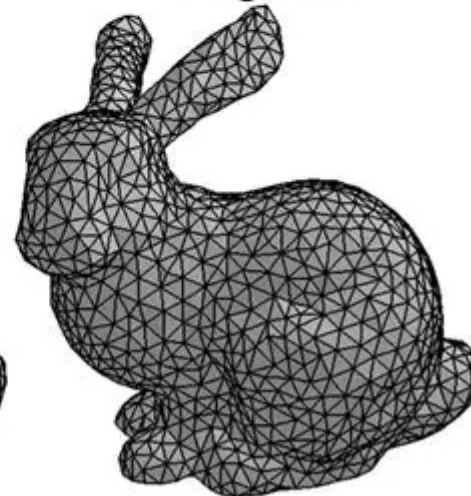
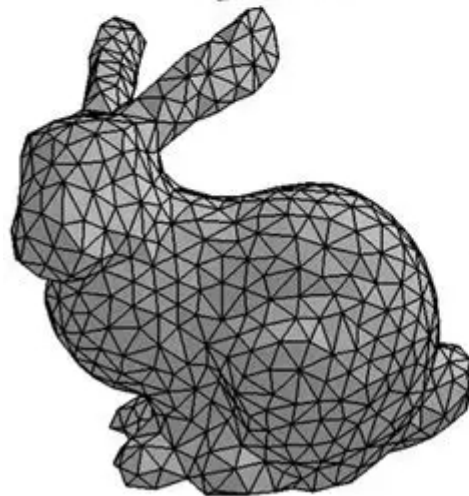
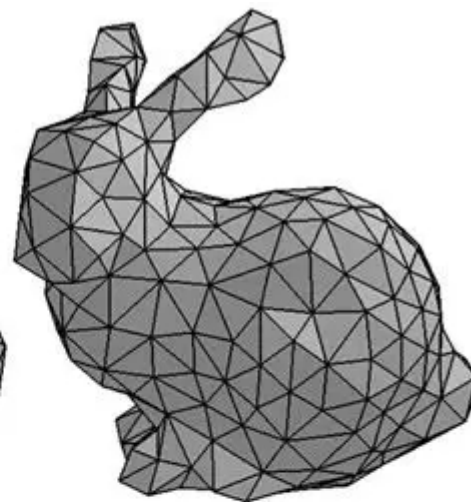
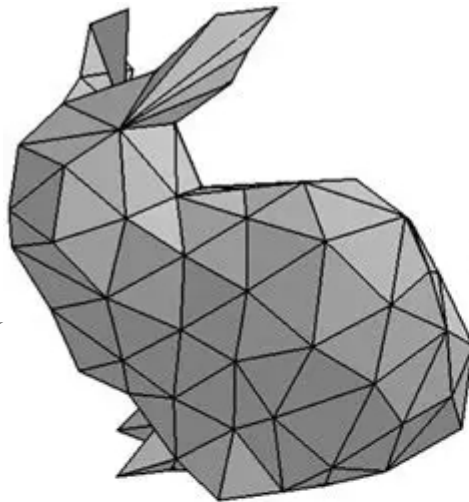
КОМБИНАТОРНИ
СТРУКТУРИ

МЕТРИЧНА
ИНФОРМАЦИЯ



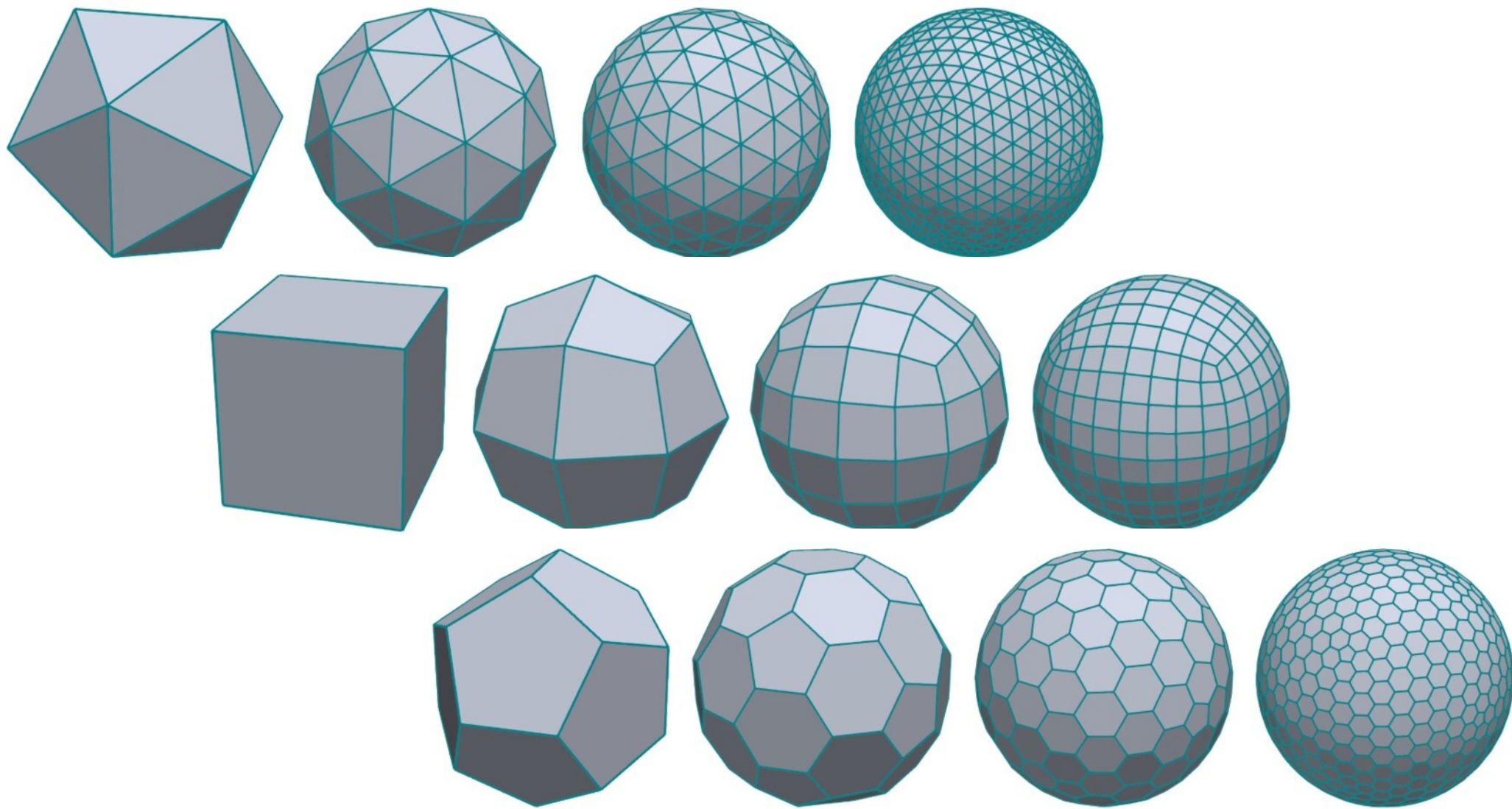
Пример 2

Low-poly



High-poly





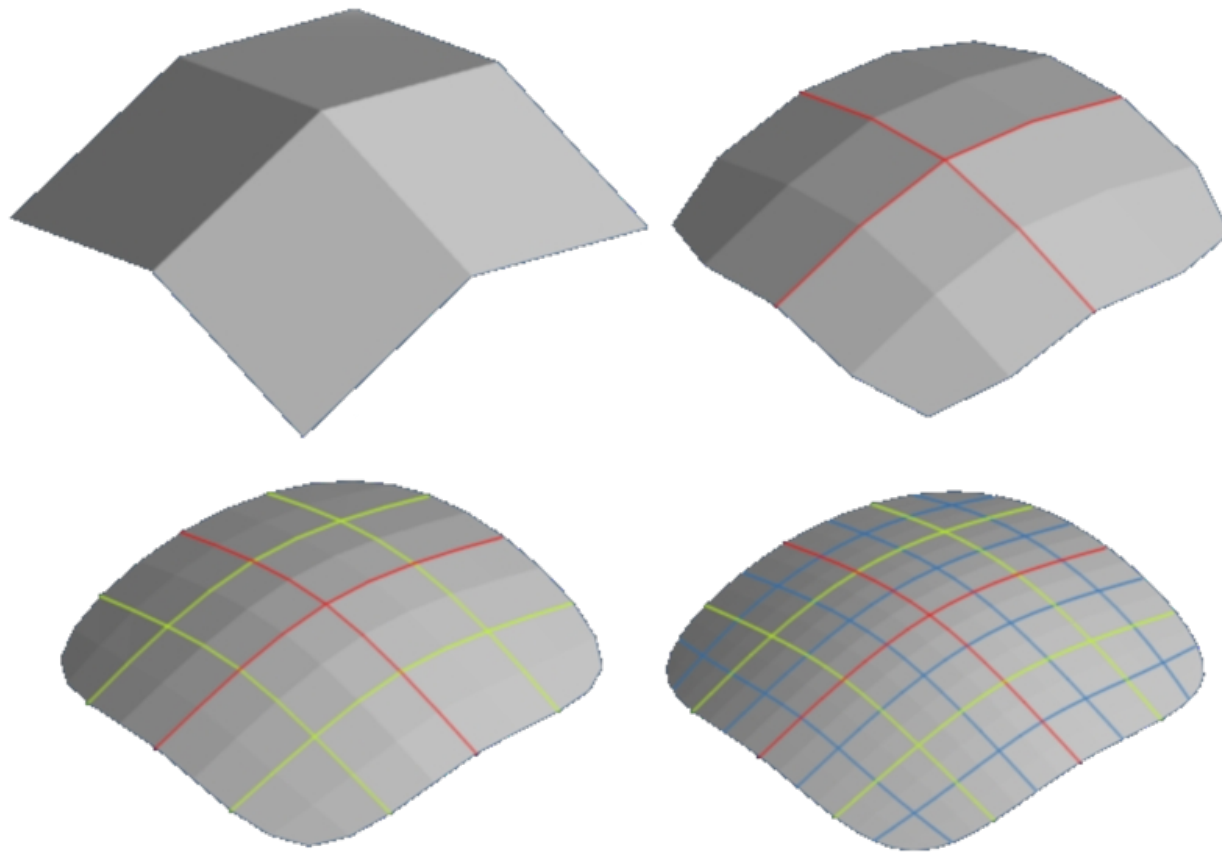
Subdivision Surface

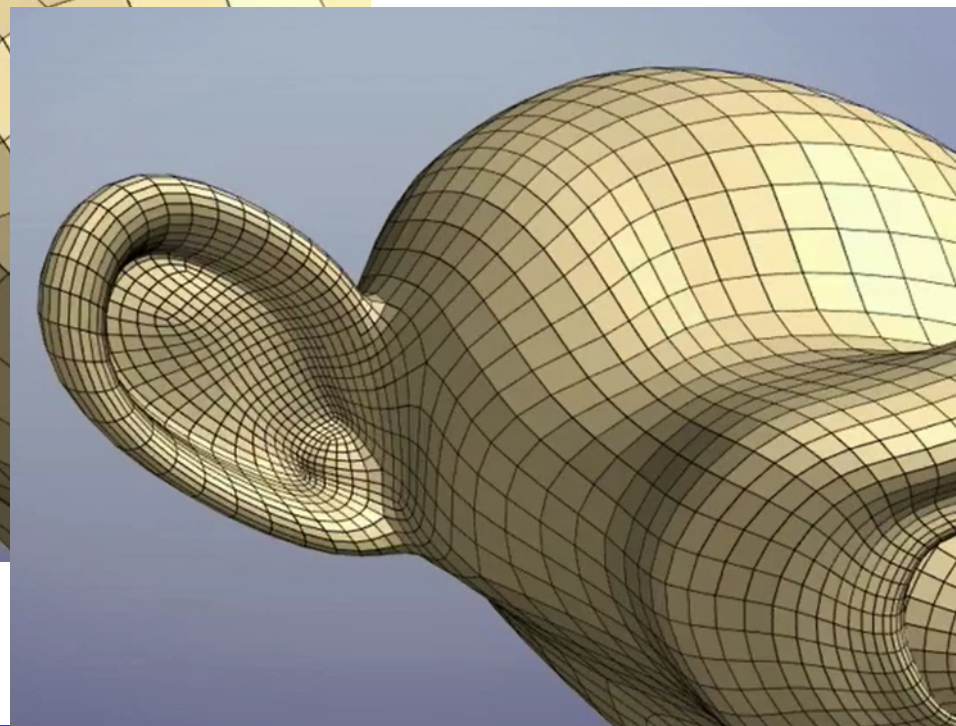
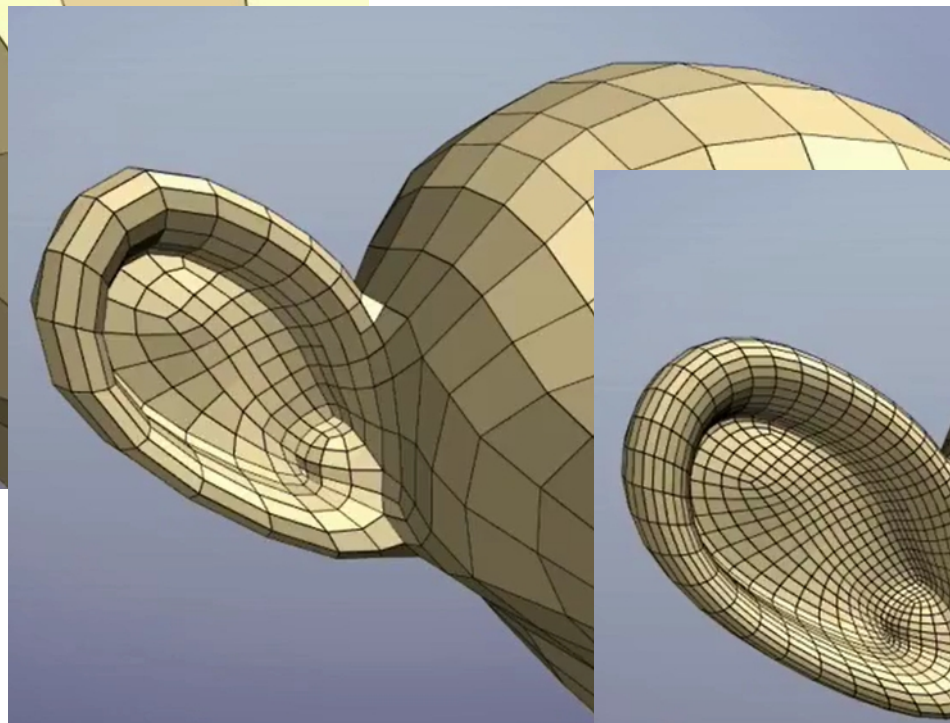
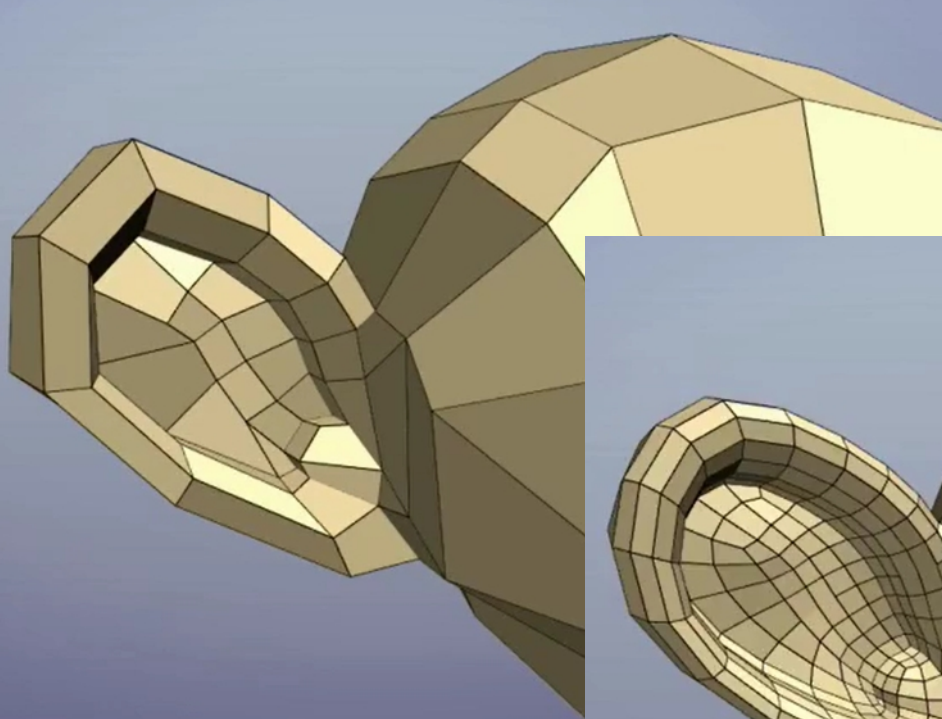
Subdivision Surface (SubD Surface или Subsurf) е модел на крива повърхност, представена чрез спецификацията на по-груба полигонална мрежа и получена чрез рекурсивен алгоритъм на разделяне.

По-рядко се използва прост алгоритъм за добавяне на геометрия към мрежа чрез подразделяне на повърхностите на по-малки, без да се променя общата форма или обем.



Пример Catmull–Clark Subdivision Surface





Облаци от точки

(3D Point clouds, PCL)

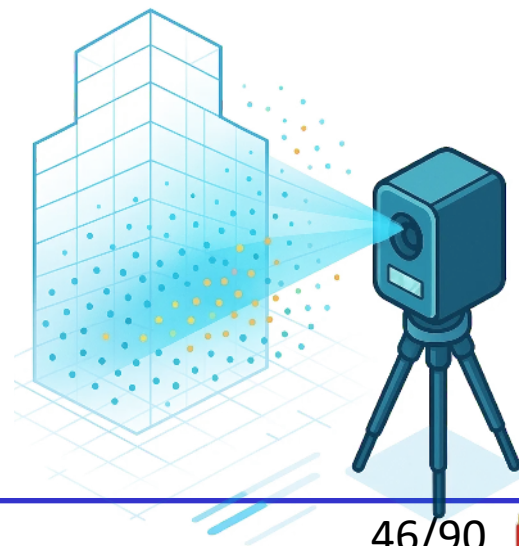


Облаци от точки

Облак от точки е дискретен набор от точки в пространството, описващ 3D форма или обект.

Точките могат да съдържат данни, различни от позицията, като например RGB цветове, нормали, времеви марки и други.

Облаците от точки обикновено се създават от 3D скенери, от фотограметричен софтуер и др.



Пример – Lidar заснемане



Пример – сегментация върху облак от точки



Параметрични

(Parametric representations)
(NURBS, B-Splines, Bezier...)

Параметрични представяния

Кривите/Телата се описват с параметрични функции, които явно описват точки от кривите или повърхностите

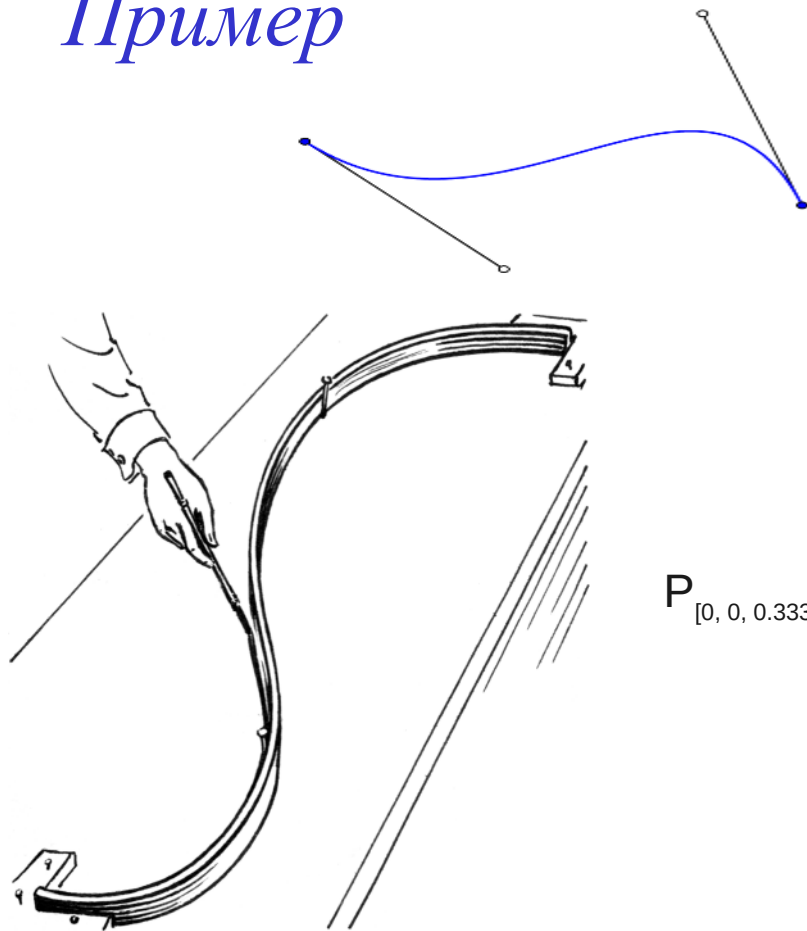
$$F(u) \rightarrow (X, Y)$$

или

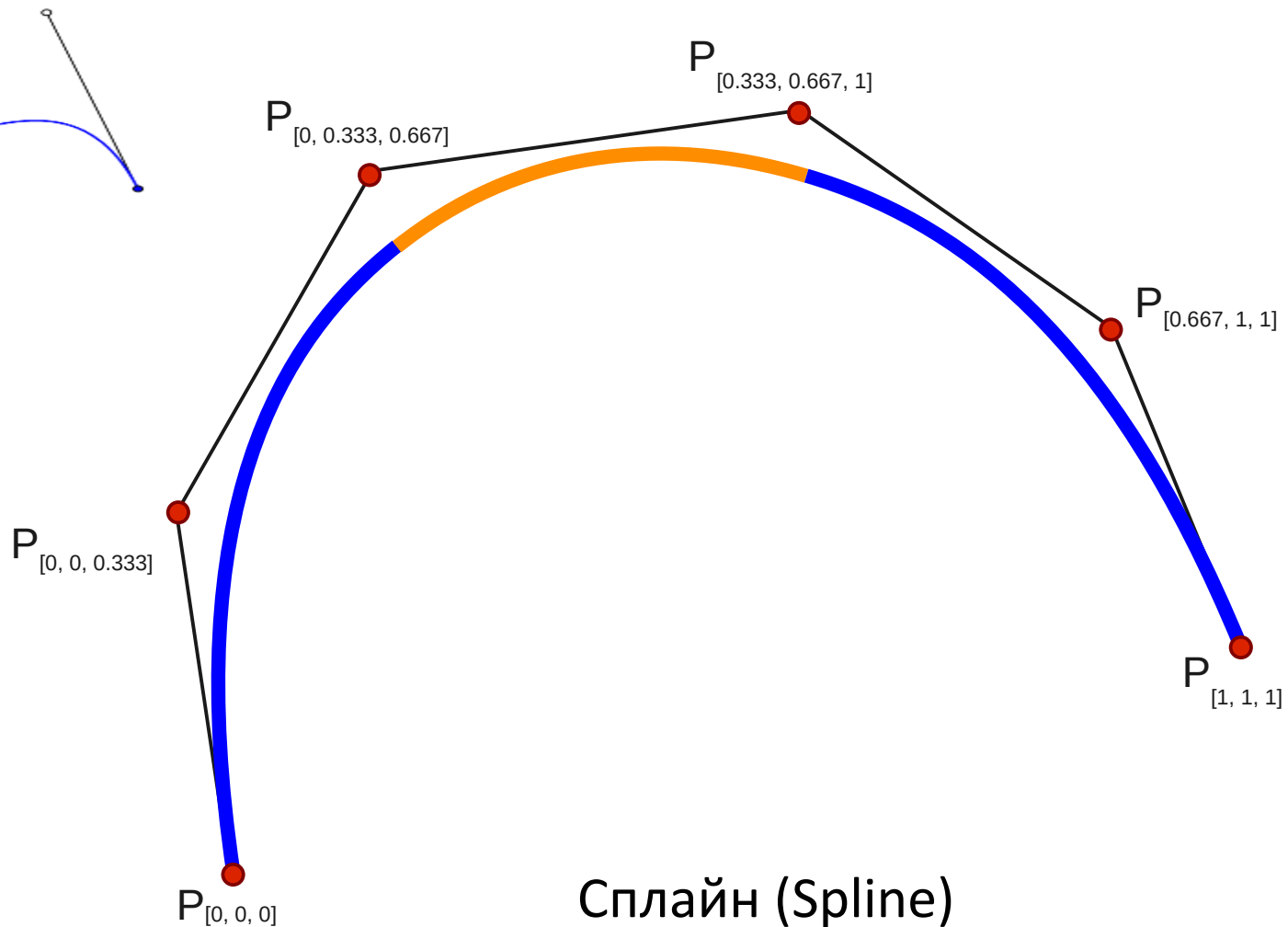
$$F(u, v) \rightarrow (X, Y, Z)$$



Пример

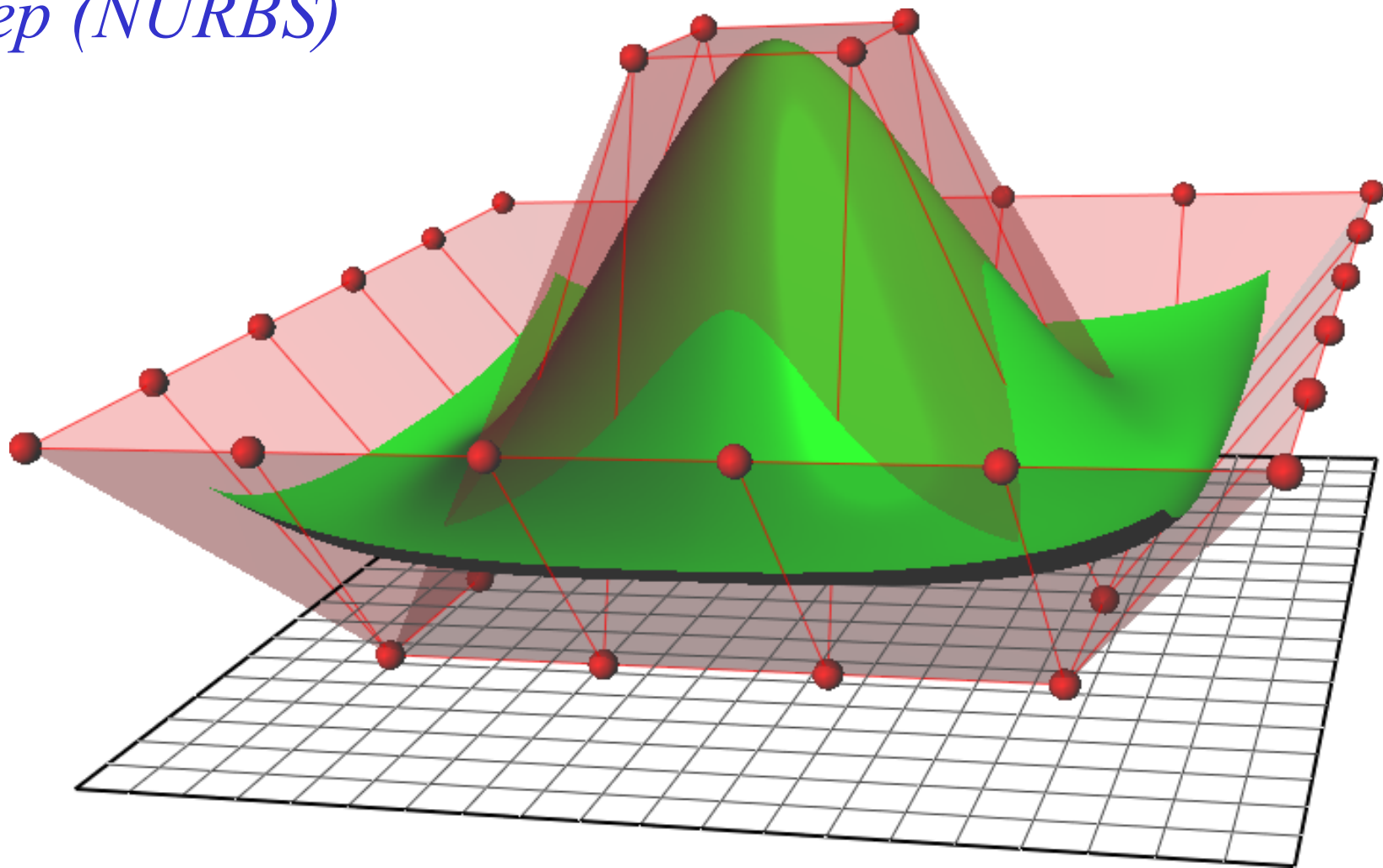


Bezier



Сплайн (Spline)

Пример (NURBS)



Функционално представяне

(Functional Representation, F-Rep)

Функционално представяне

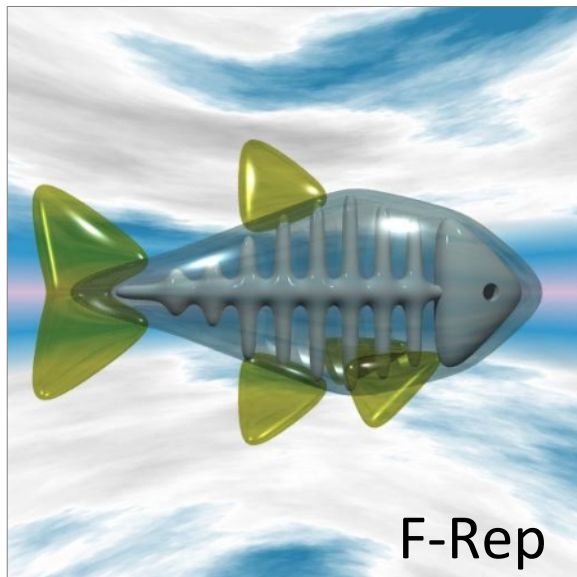
Телата се описват неявно чрез тримерна функция
 $F(x,y,z)$

Точките в пространството принадлежащи на тялото
отговарят на условието $F(x,y,z) < 0$

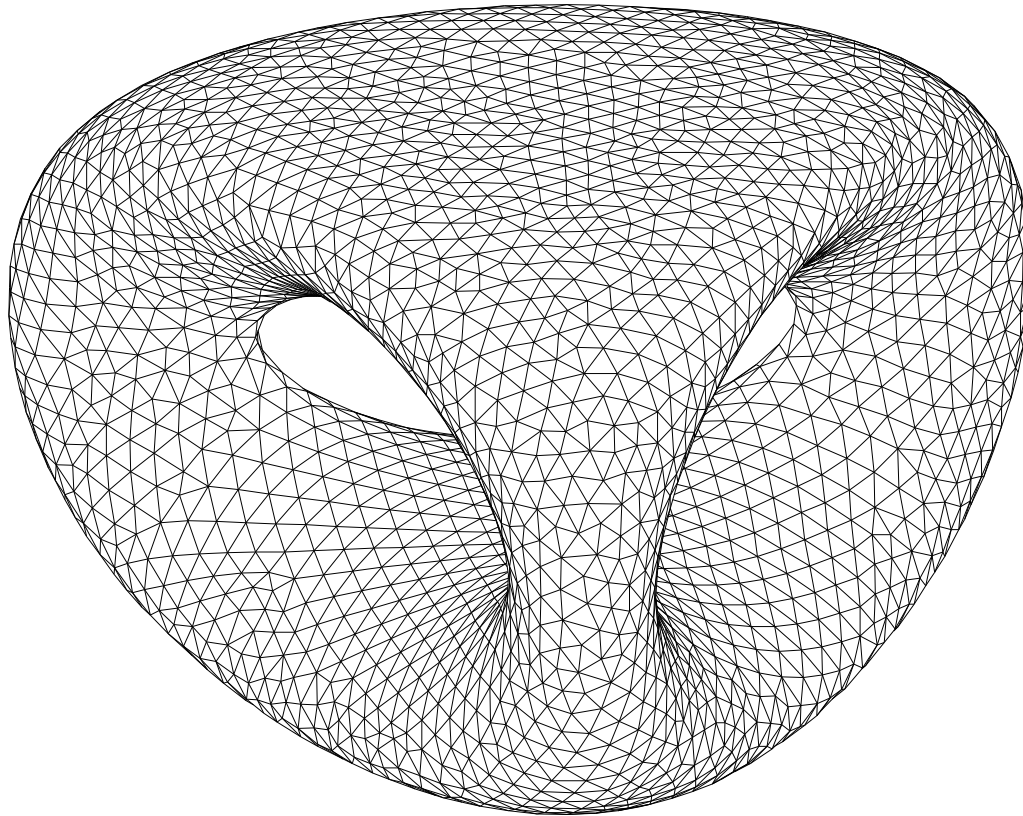
$F(x,y,z) = 0$ е границата на тялото.



Пример (F-Rep)



Пример (Genus 2) – полионизирана повърхност



$$2y(y^2 - 3x^2)(1 - z^2) + (x^2 + y^2)^2 - (9z^2 - 1)(1 - z^2) = 0$$

Запълващи продукции

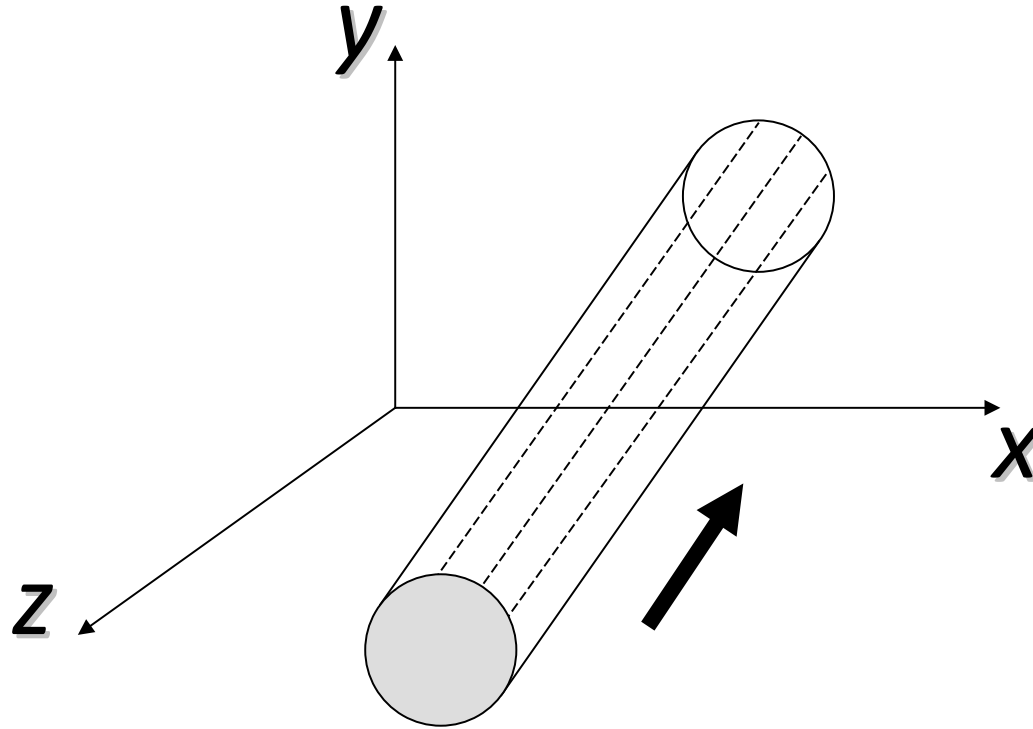
(Sweeping)

Запълващи продукции

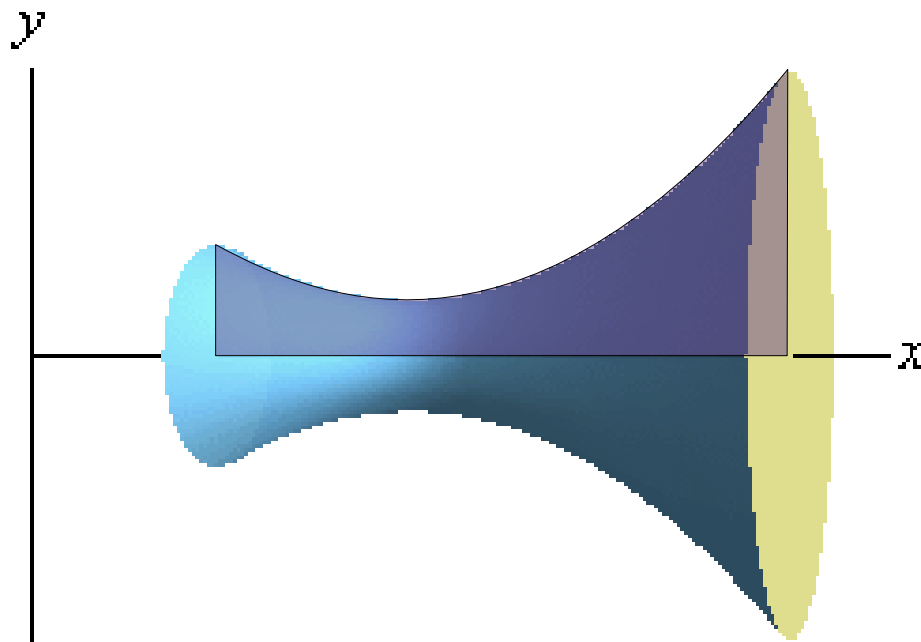
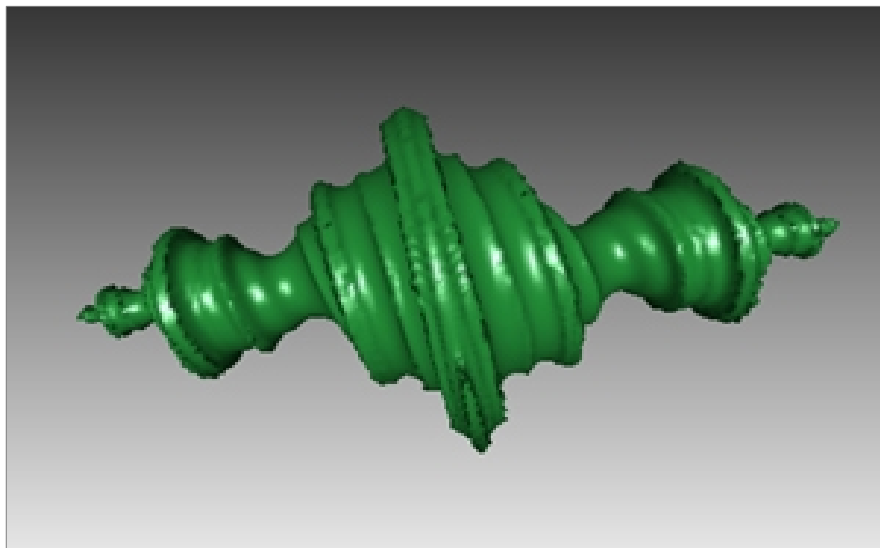
Всяко множество, което се движи през пространството, “замита” тяло, което може да бъде представено чрез представяне на движещото се тяло и траекторията му на движение.



Запълващи продукции



Пример – ротационни тела



Метатопчета

(Metaballs)



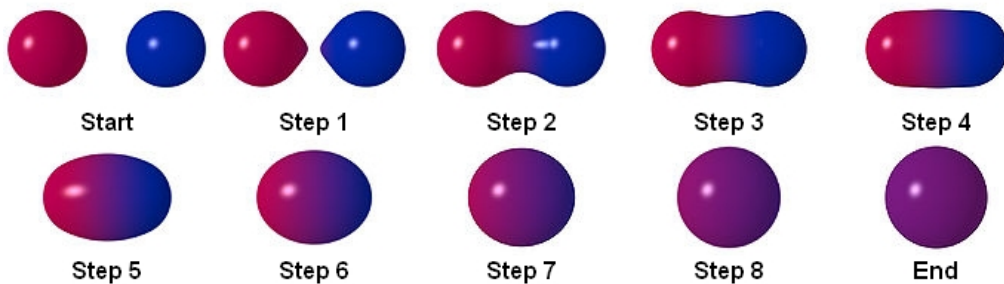
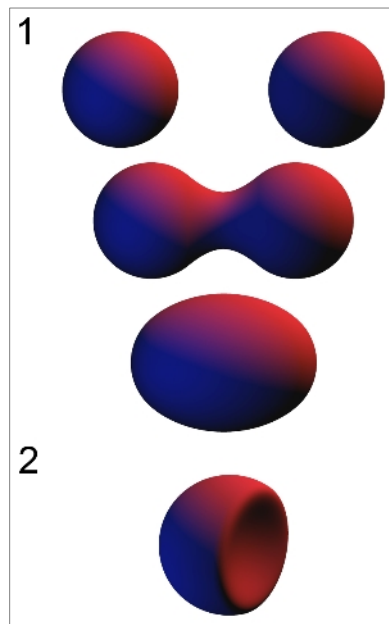
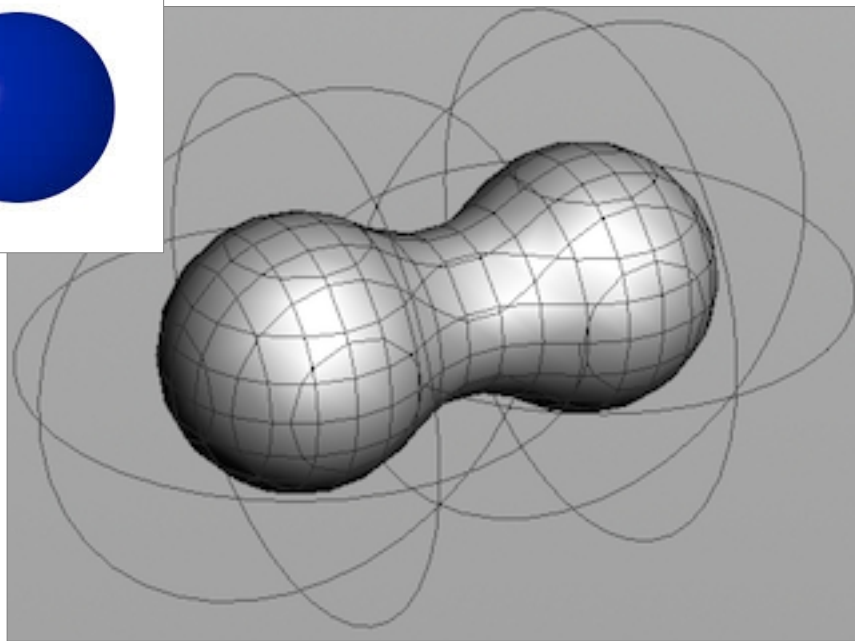
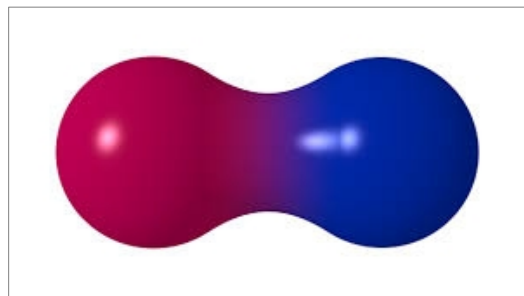
Метатопчета

Всяко „тяло“ се представя като съставено от множество малки сфери, които когато са близо една до друга се сливат гладко помежду си.

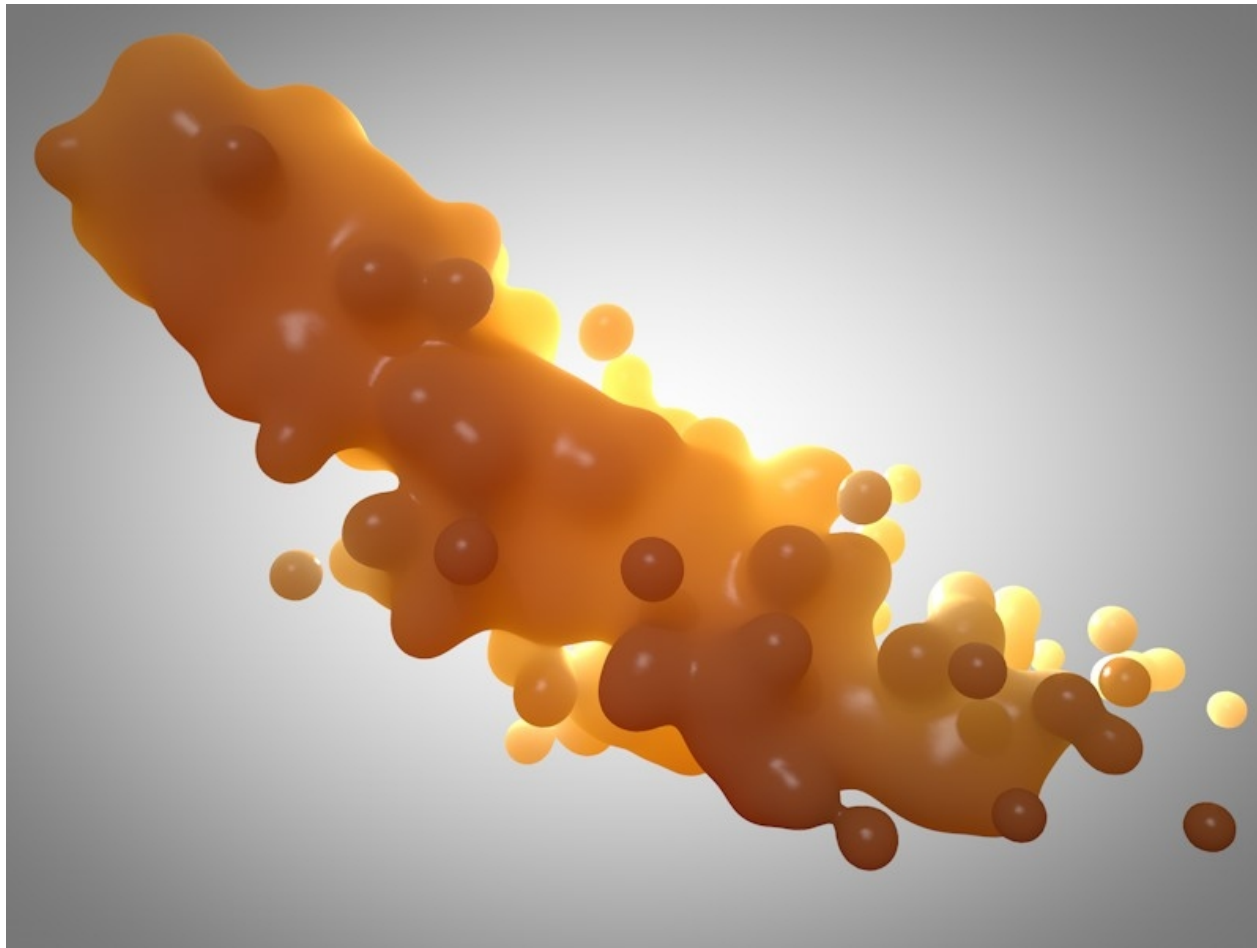
Много често тези схеми се използват за създаване на специални ефекти или „меки“ тела.



Метатопчета



Метатопчета



Системи частици

(Particle systems)



Системи частици

Всяко „тяло“ се представя като съставено от множество малки частици (точки, триъгълници, фасети, ...), които се създават, премахват и/или променят характеристиките си по някакви закони.

Много често тези схеми се използват за създаване на специални ефекти или за моделиране и визуализация на природни стихии, като огън, пушек, сняг, дъжд, торнадо и др.



Системи частици



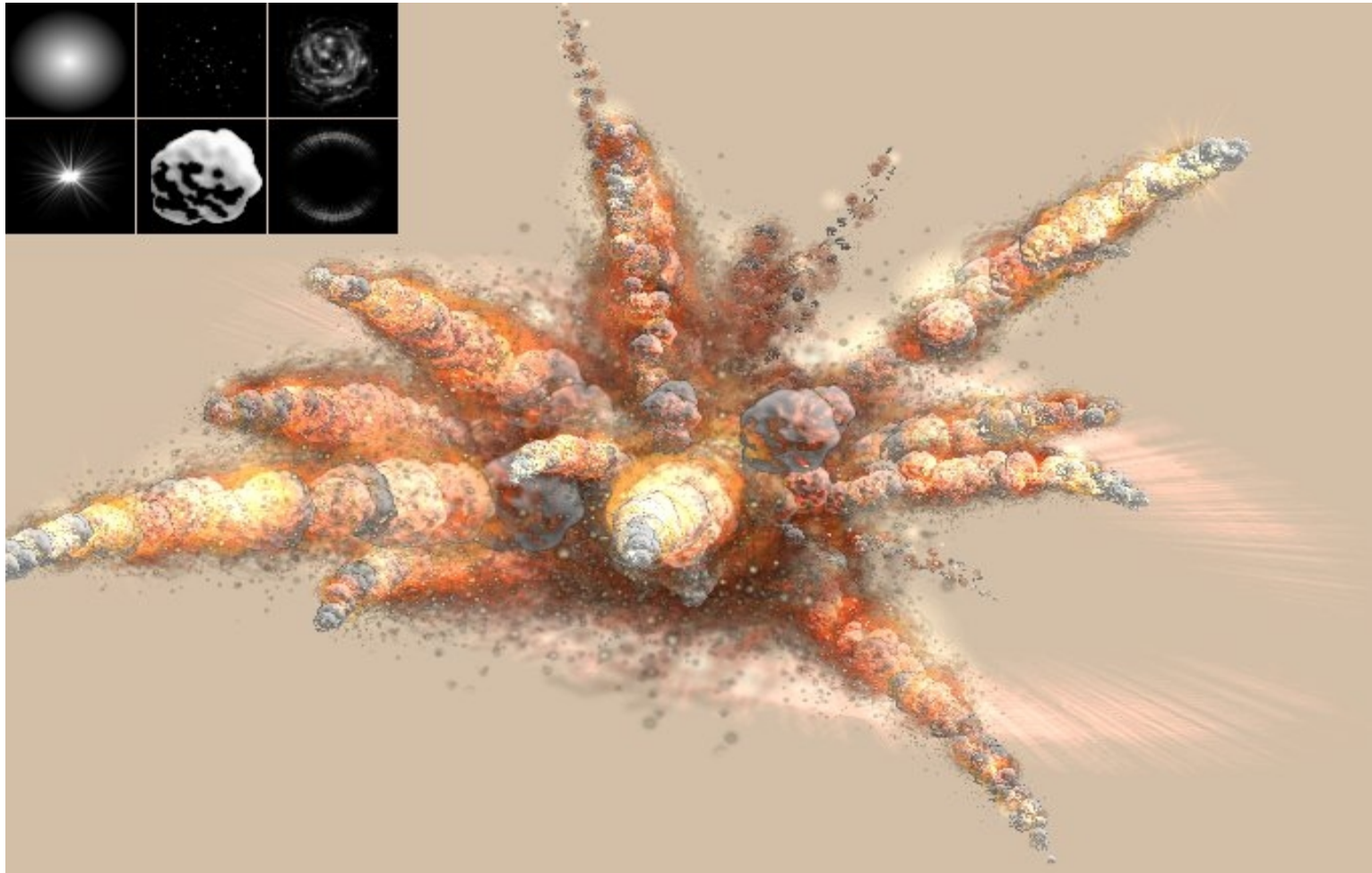
Системи частици



Системи частици



Системи частици



Системи частици (коса)



Рапунцел и разбойникът:

Косата е моделирана със специален много сложен математически модел, който позволява на аниматорите лесно да постигнат естественост на движенията ѝ.



Системи частици (козина)



Невронни мрежи

(Neural representations)



Невронни мрежи

Използват се невронни мрежи за различни начини на (неявно) описание на геометрията на телата.

Геометрията се описва, чрез обучение на невронните мрежи от различен тип, чрез използване на примери – множество двумерни изображения на даден тримерен обект, тримерни облаци от точки и други.

NeRF

NeRF модел получен от 100 входни двумерни изображения и визуализация на модела от други гледни точки.

Входни Изображения



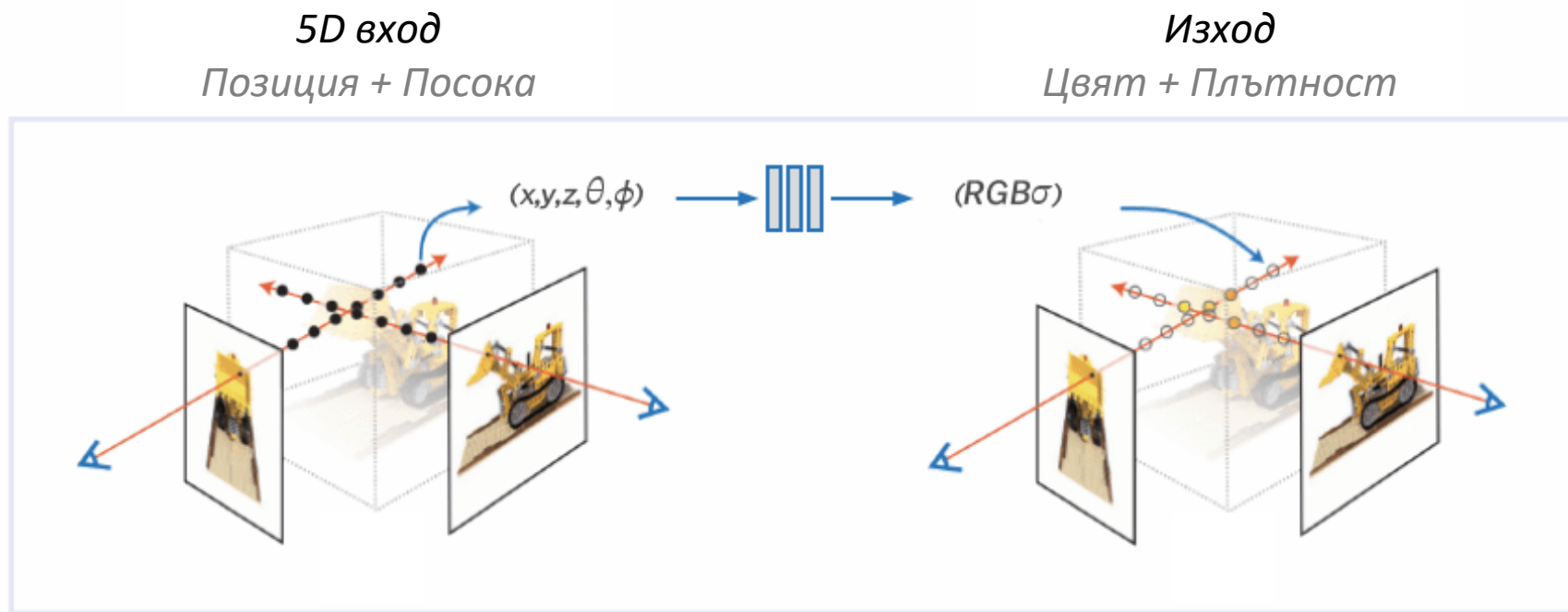
NeRF обучение



Визуализация от нови гледни точки



Neural radiance field scene representation



NeRF



Флуиди и Меки тела

(Fluids, Soft body, ...)

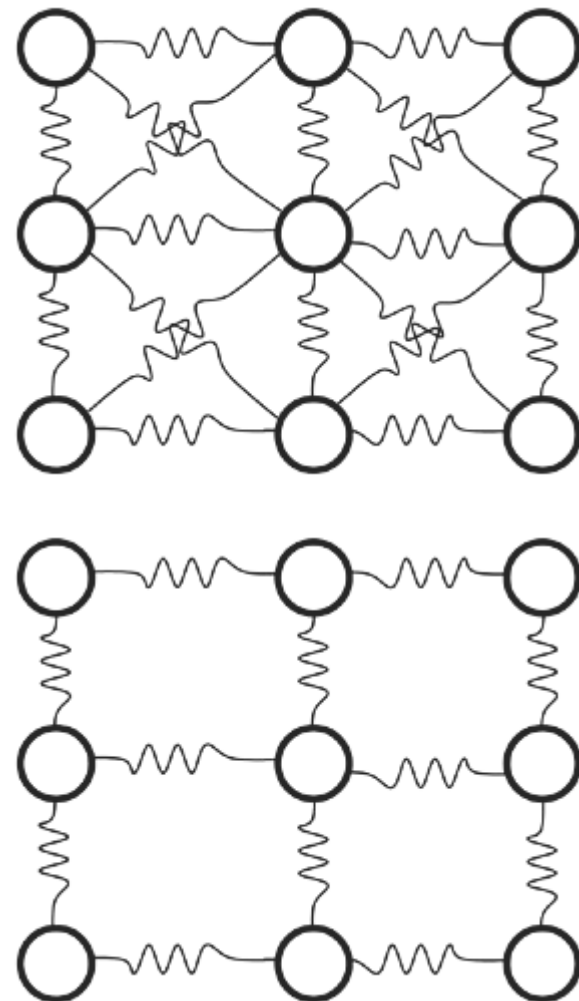


Меки тела

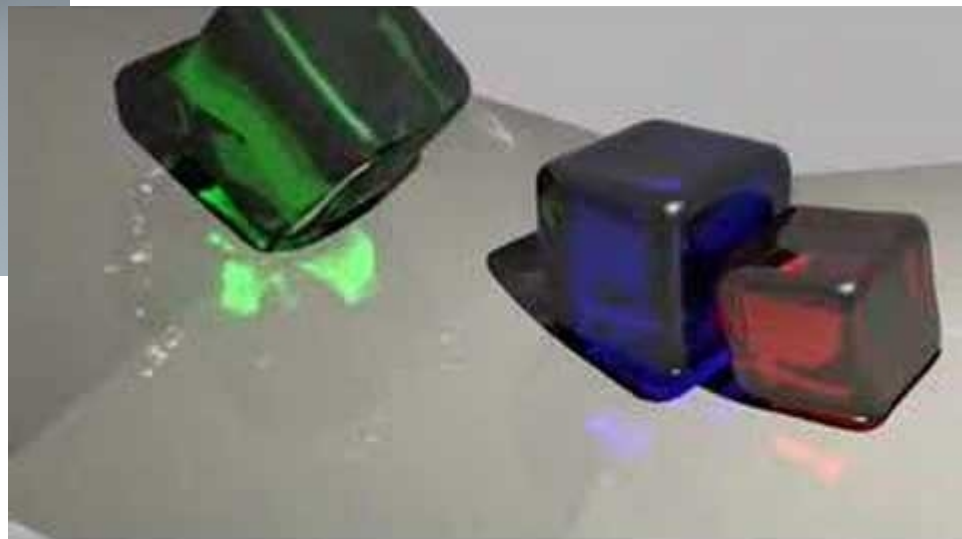
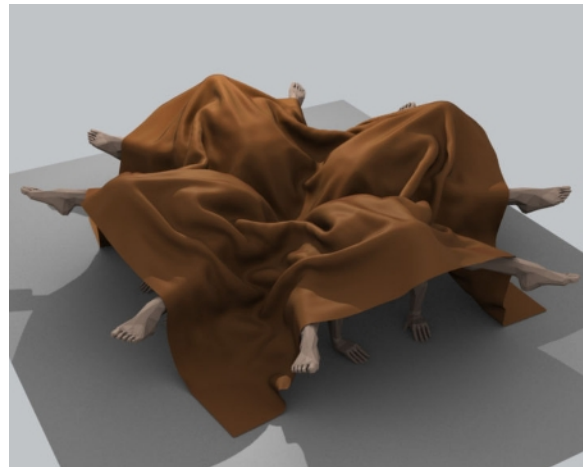
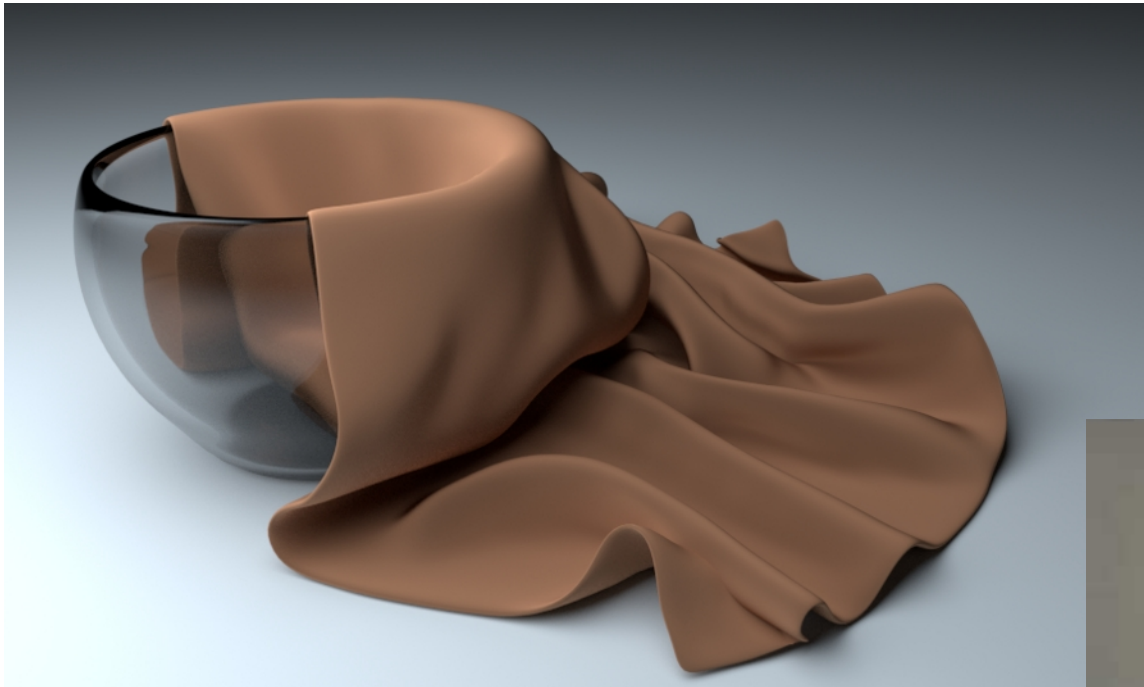
За моделиране на меки тела често се използват класическите структури mesh от триъгълници или четириъгълници, като към структурата са добавени допълнителни физически параметри в модела, позволяващи телата да се деформират реалистично и да могат да се прилагат физически симулации, като гравитация, колизии и други.

Това позволява да се моделират и симулира реалистичното поведение на различни видове плат и други тъкани, еластични тела и др.

Пример – меки тела



Пример – меки тела



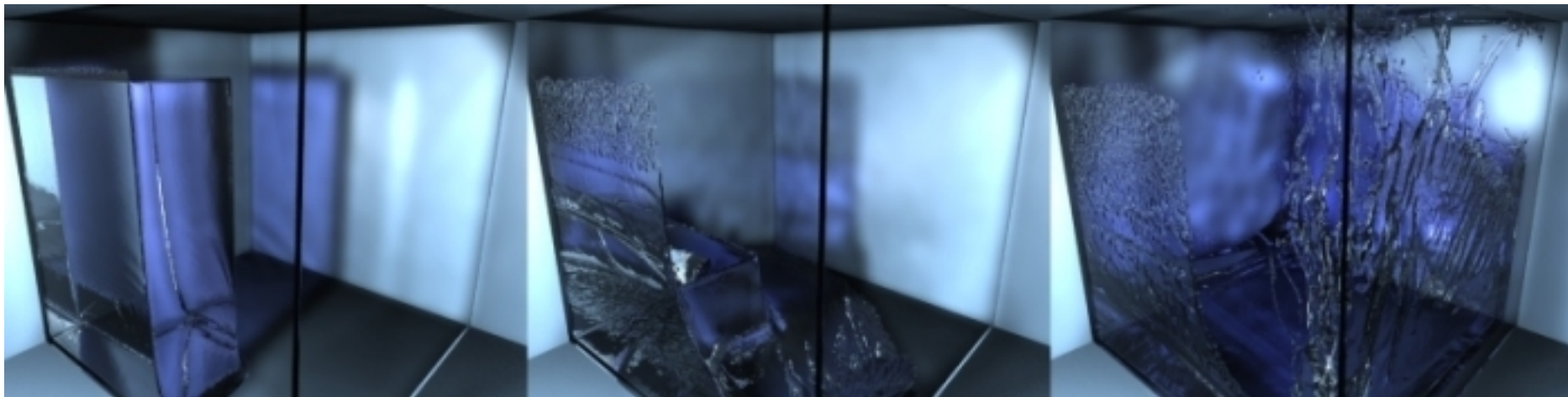
Флуиди

За създаване на модели на флуиди в съвременните системи за моделиране и анимация се използват частни диференциални уравнения, прилагат се законите познати от флуидната динамика и така може да се симулира реалистично поведение на тези модели.

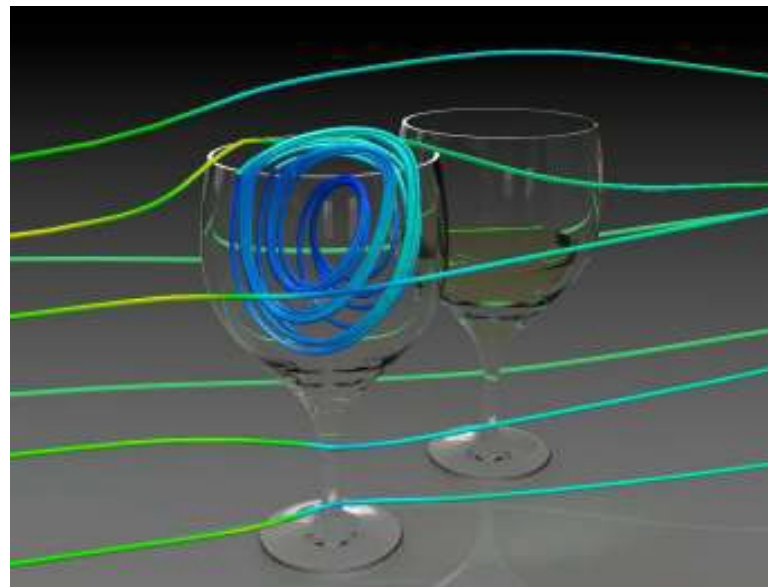
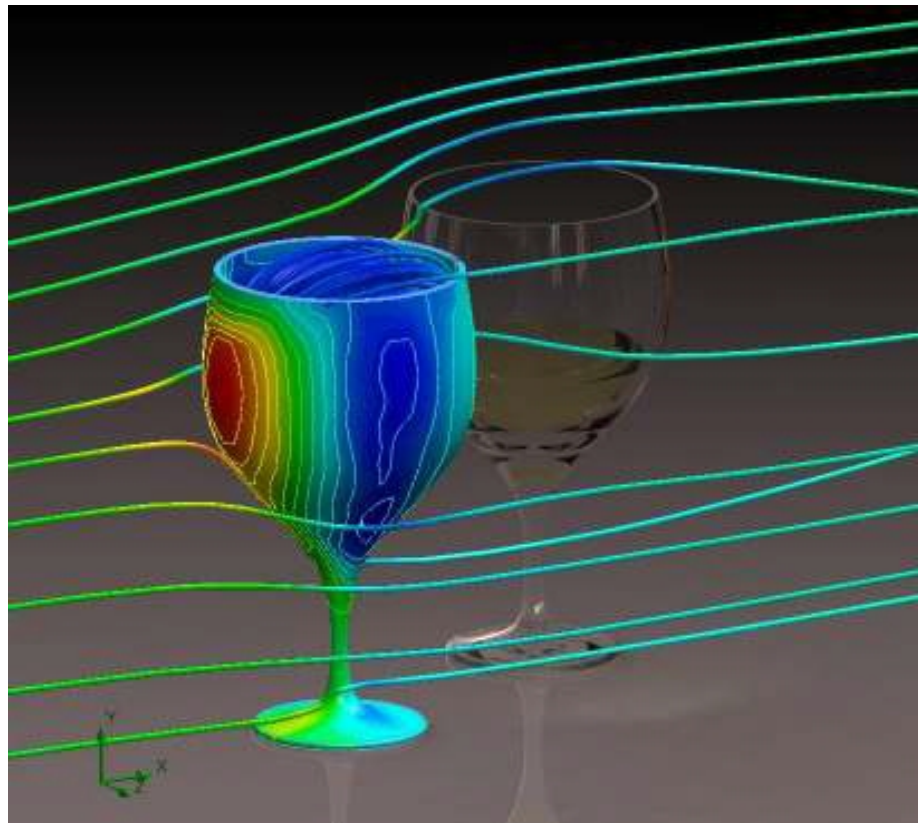
С добавяне на подходящи прозрачни материали към моделите се получават реалистични вълни, плискащи се в съд течности и други.



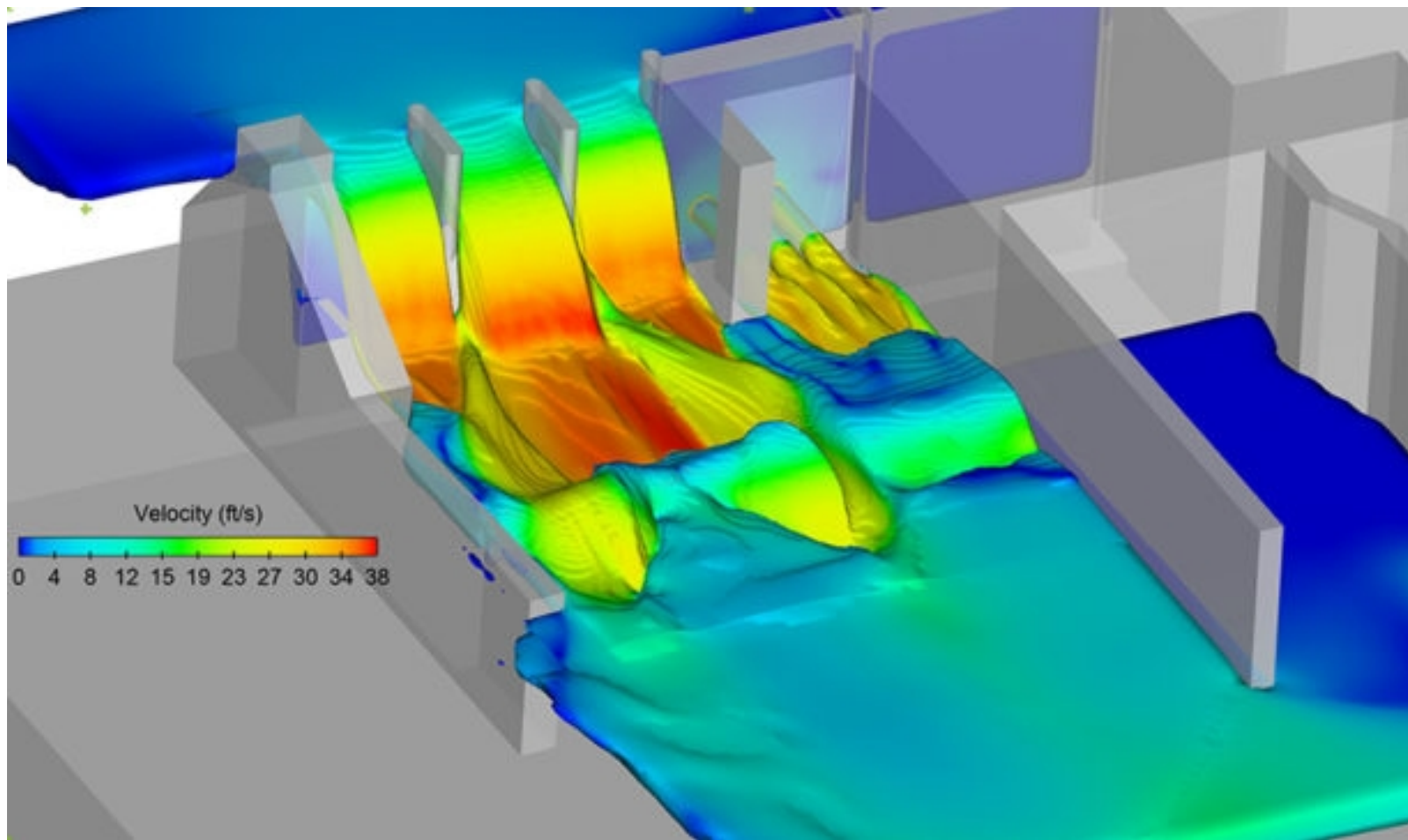
Пример – флуиди



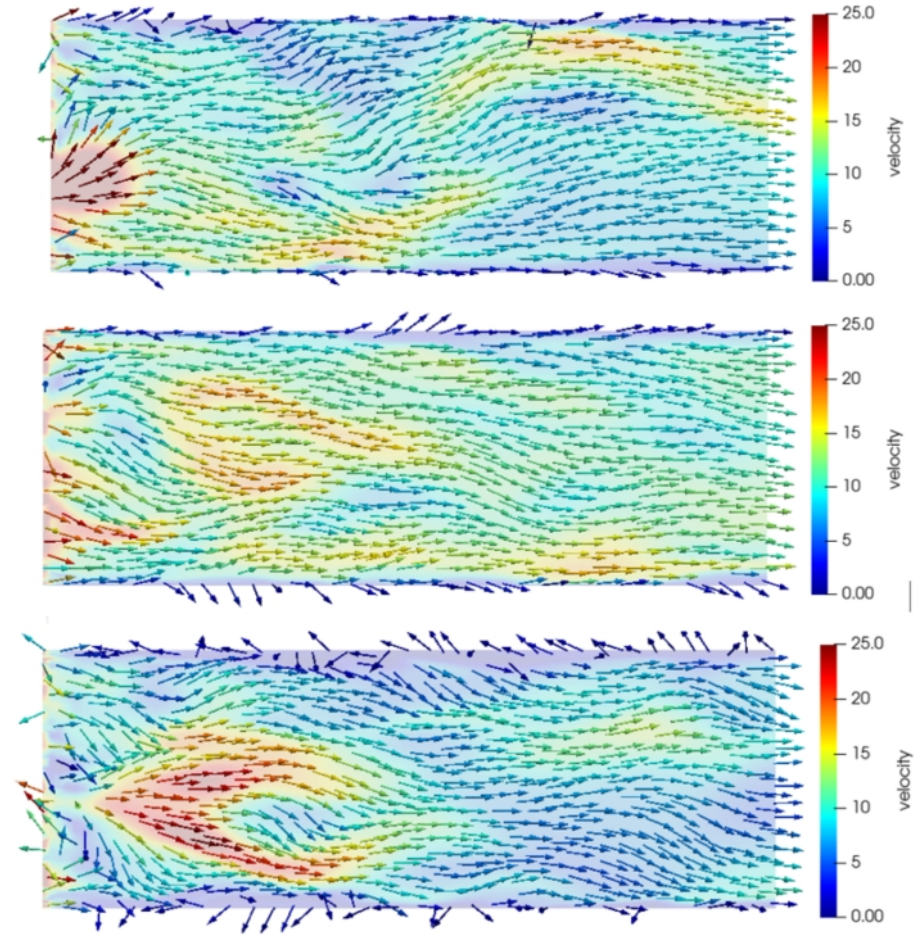
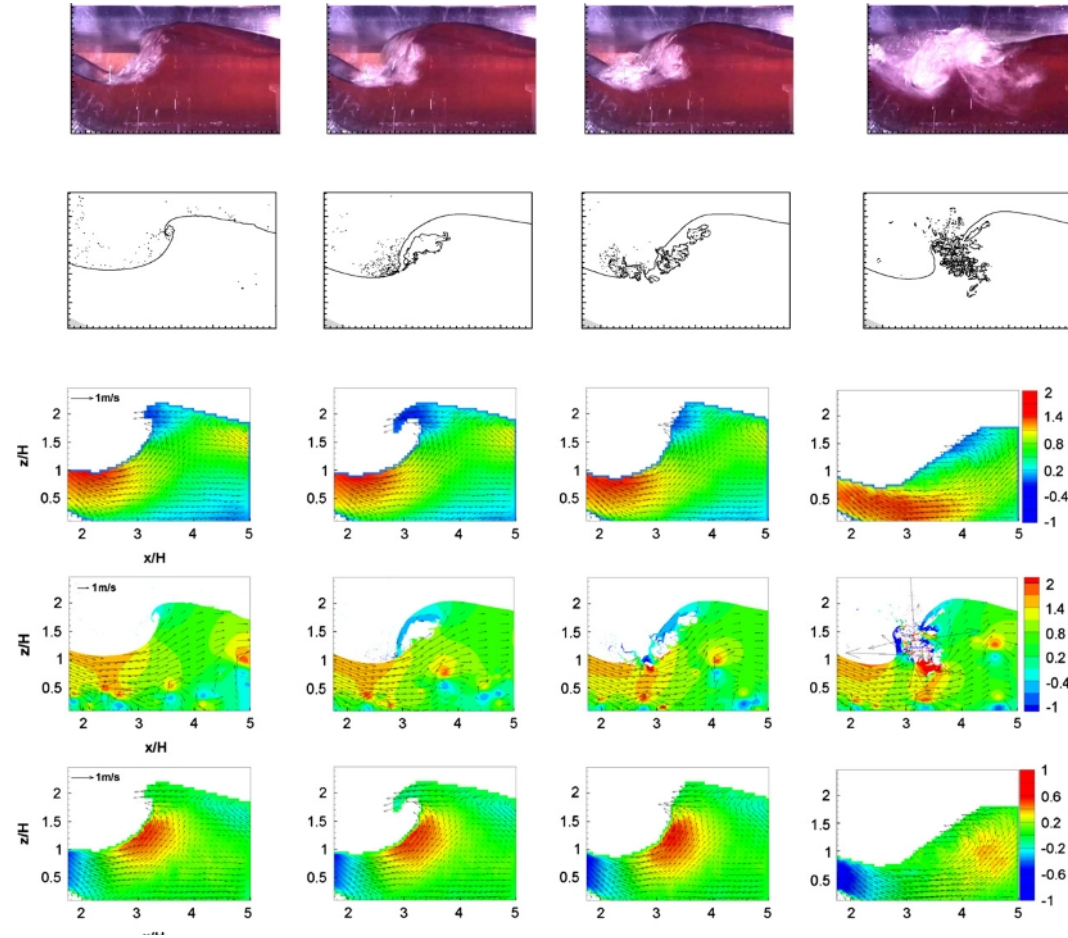
Пример – флуиди



Пример – флуиди



Пример – флуиди



Йерархични

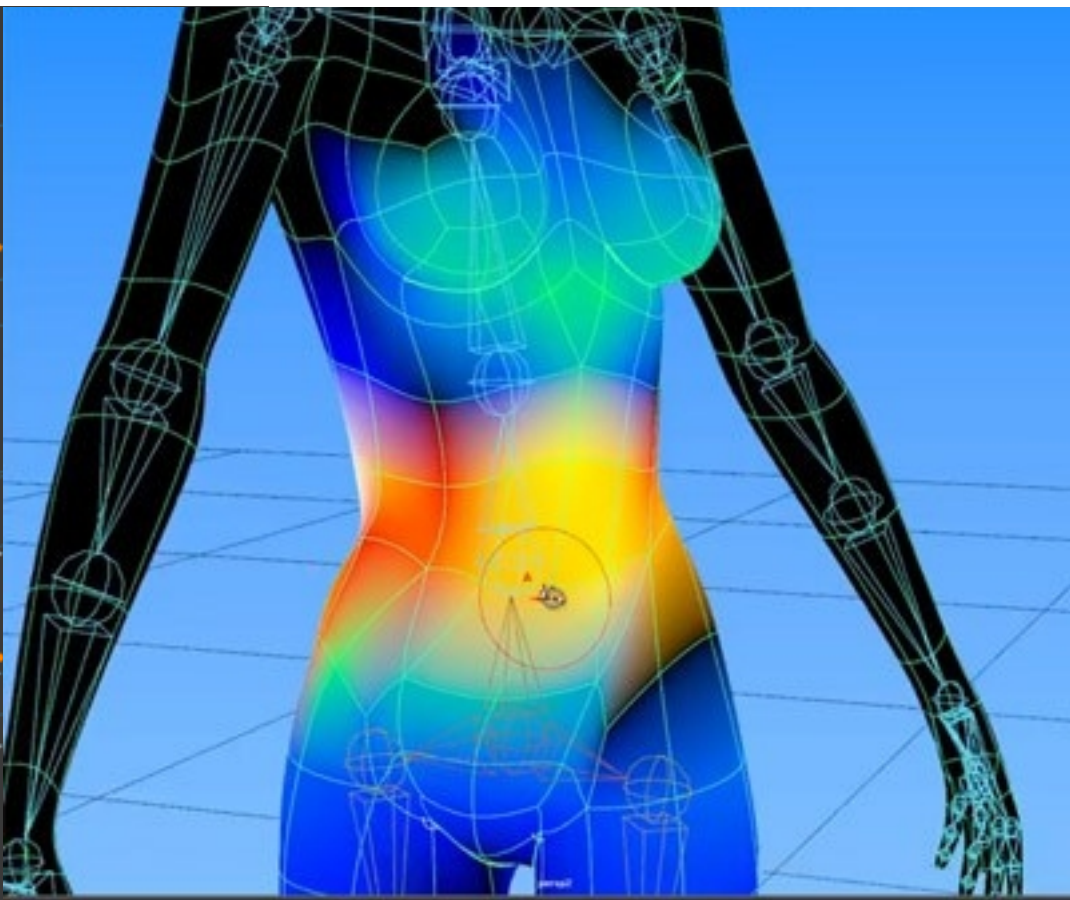
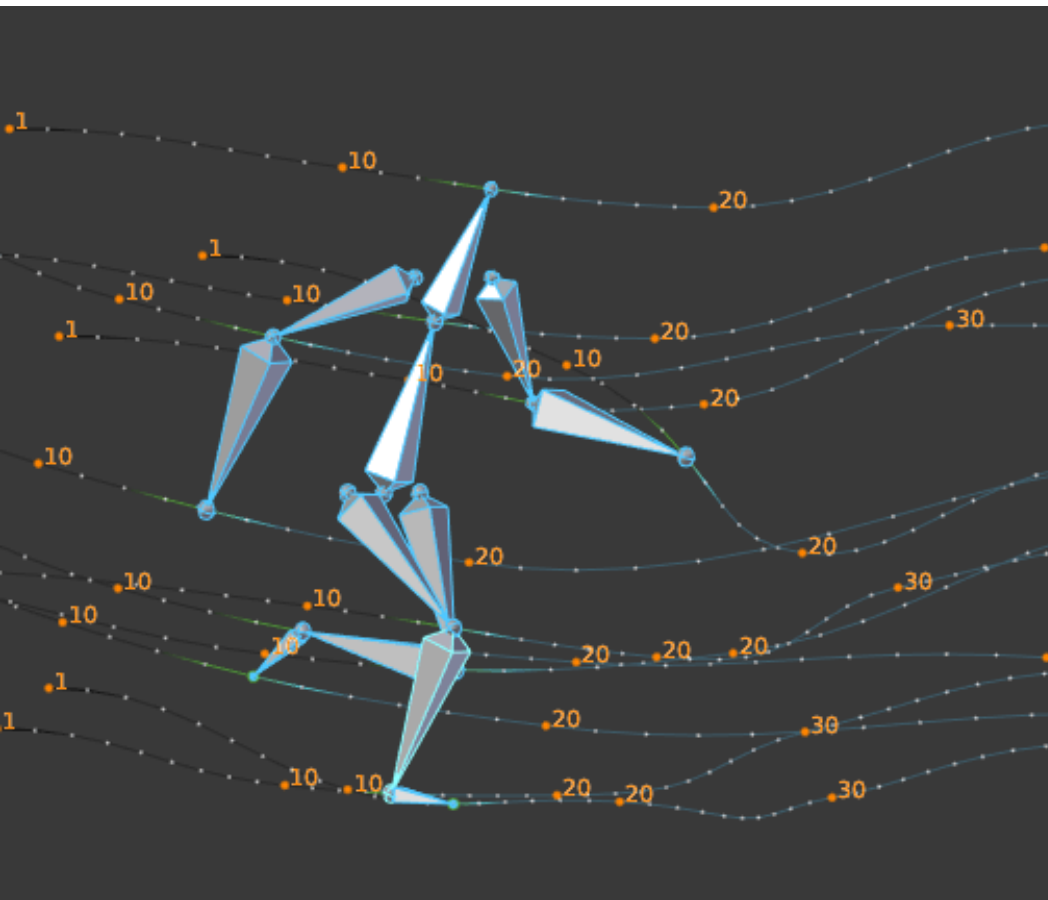
(Armature, Scene graph, etc.)

Йерархични

Използва структури от йерархично свързани елементи, за да опише моделите или техни помощни елементи.

Например при скелетната анимация се създава скелетната структура, изградена от виртуални кости, свързани йерархично помежду си, които позволяват на основния модел да се променя и движи от аниматора. Скелета може да променя позицията си и на базата на техники като *motion capture* за прихващане на движението на реални актьори.

Пример – Скелет (Armature) за скелетна анимация



Въпроси?

arenev@uni-plovdiv.bg

