Трёхмерная графика

§ 66. Введение
§ 67. Работа с объектами
§ 68. Сеточные модели
§ 69. Модификаторы
§ 70. Кривые
§ 71. Материалы и текстуры
§ 72. Рендеринг
§ 73. Анимация
§ 74. Язык § 74. Язык VRML

© К.Ю. Поляков, Е.А. Ерёмин, 2013 http://kpolyakov.spb.ru

Трёхмерная графика

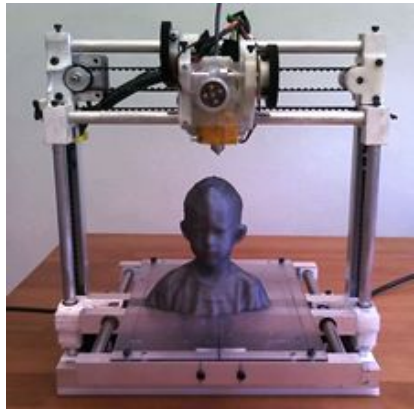
§ 66. Введение

Что такое 3D-графика?

Трёхмерная графика (3D-графика) – это раздел компьютерной графики, который занимается созданием моделей и изображений *трёхмерных* объектов.

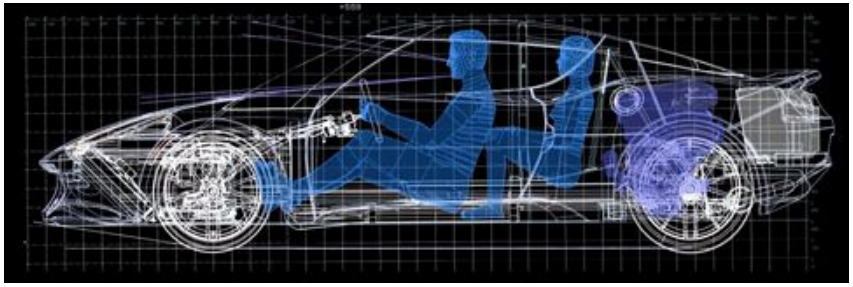
3D-модели: каждая точка имеет 3 координаты

3D-модели
- проекции (на плоскость)
- анимация
- расчёты (на прочность и т.п.)
- 3D-печать



Области применения

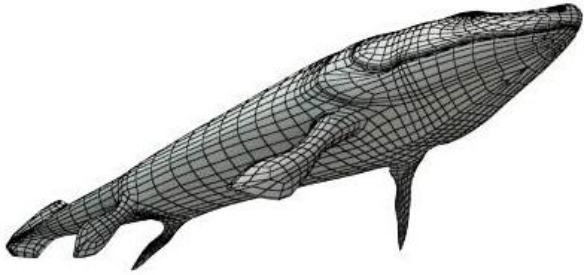
- САПР – системы автоматизированного проектирования



- компьютерные тренажеры и обучающие программы

Области применения

- построение трёхмерных моделей в науке, промышленности, медицине



- дизайн зданий и интерьера

- компьютерные эффекты в кино и телевидении
- телевизионная реклама
- интерактивные игры

Этапы работы с 3D-моделями

- **моделирование** — создание трёхмерных моделей объектов
- **текстурирование** (раскраска) — наложение на модели рисунков (текстур), имитирующих материал
- **освещение** — установка и настройка источников света
- **анимация** — описание изменения объектов во времени (изменение положения, углов поворота, свойств)
- **съемка** — установка камер (выбор точек съёмки), перемещение камер по сцене
- **рендеринг** (визуализация) — построение фотореалистичного изображения или анимации

© К.Ю. Поляков, Е.А. Ерёмин, 2013

Программы для 3D-моделирования

3ds MAX (www.autodesk.com)

Maya (www.autodesk.com)

AutoCAD (www.autodesk.com)

Cinema4D (www.maxon.net)

Blender (www.blender.org) – свободное ПО, версии для *Windows*, *Linux*, *Mac OS* и др.

Требования к компьютеру:

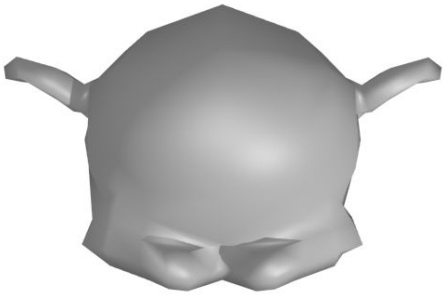
- мощный процессор
- оперативная память
- дисковая память

для версии **2.69**

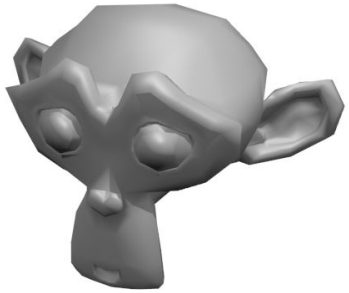
Проекции

Проекция – плоское (двухмерное) изображение трёхмерного объекта.

вид сверху:

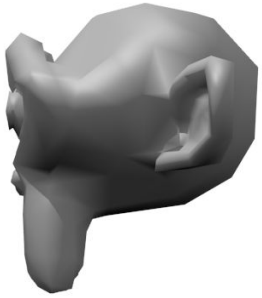


произвольная проекция:



вид спереди:

вид справа:

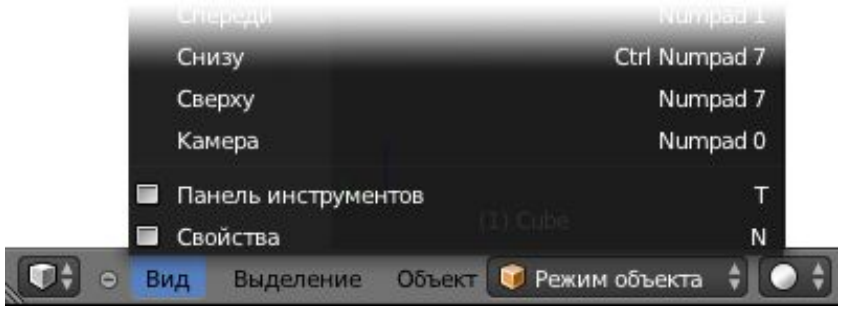


Сюзанна

Ctrl+Alt+Q: 4 проекции/только проекция пользователя

Управление проекциями

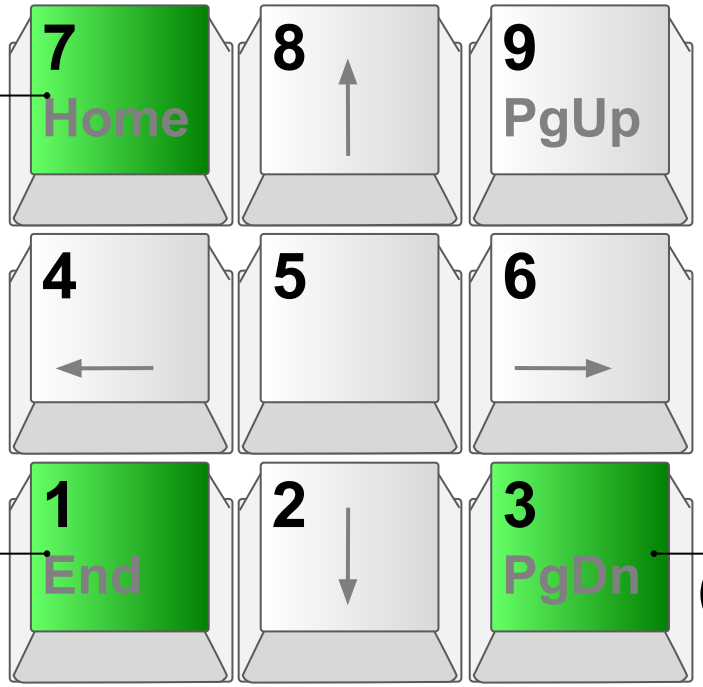
меню **Вид** (View)



или…

Num7

вид сверху
(+*Ctrl*: вид снизу)



вид спереди
(+*Ctrl*: вид сзади)

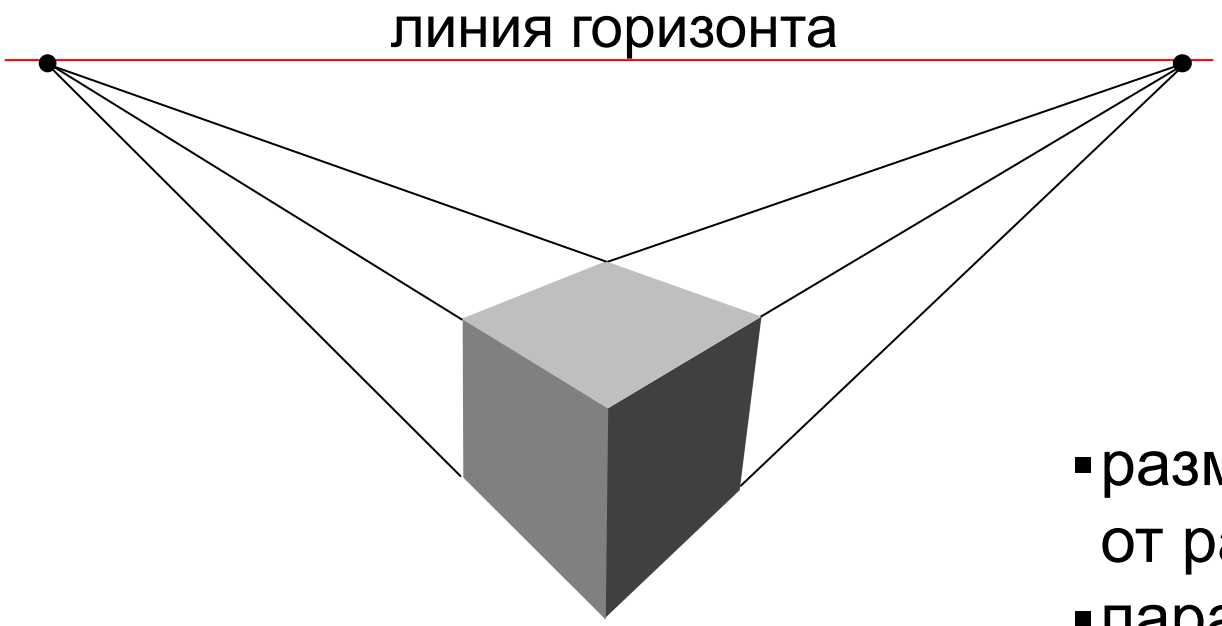
Num1

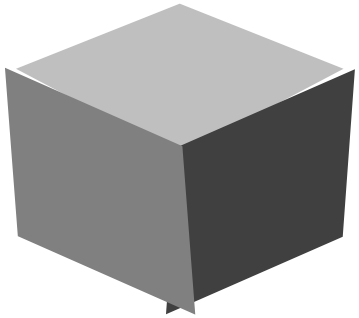
вид справа
(+*Ctrl*: вид слева)

Num3

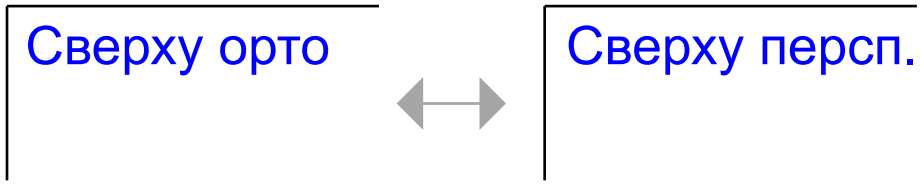
Ортогональные и перспективные проекции

перспективная **ортогональная**





- размеры не зависят от расстояния
- параллельные грани остаются параллельными



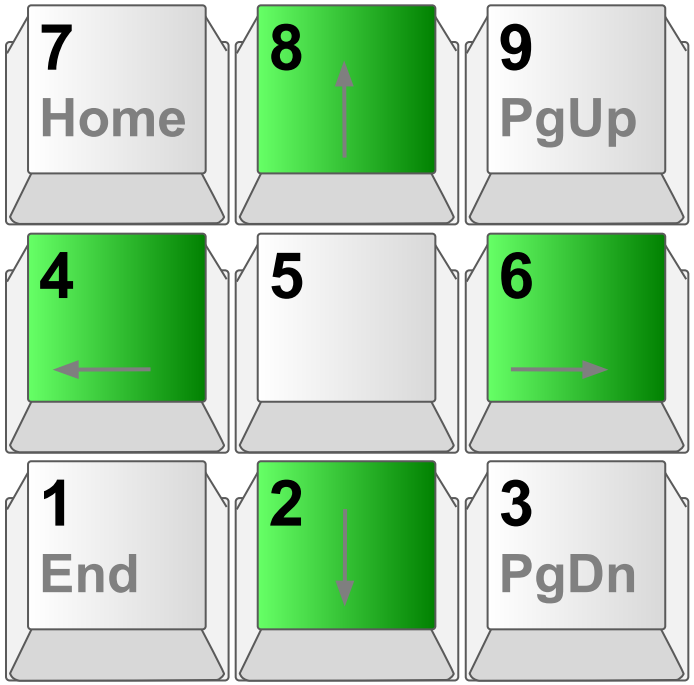
Num5 – ортогональная/перспективная

Управление видами

Колёсико мыши: изменить масштаб

При нажатом колёсике мыши: вращение проекции

+Shift: перемещение без вращения



Num2, Num4, Num6, Num8: вращение проекции

+Ctrl: перемещение

Трёхмерная графика

§ 67. Работа с объектами

© К.Ю. Поляков, Е.А. Ерёмин, 2013

Выделение объектов

ПКМ: выделить объект!

Shift+ПКМ: выделить несколько объектов

A: выделить всё/снять выделение со всех

B: выделить нужные объекты прямоугольной рамкой

ПКМ, затем **Ctrl+ЛКМ**: произвольным контуром

C: выделить нужные объекты круглой кистью
колесико регулирует размер кисти
ЛКМ – завершить выделение
нажать на колесико – отменить выделение объекта

Num. (*Вид – Показать выделенное*) – приблизить выделенные объекты

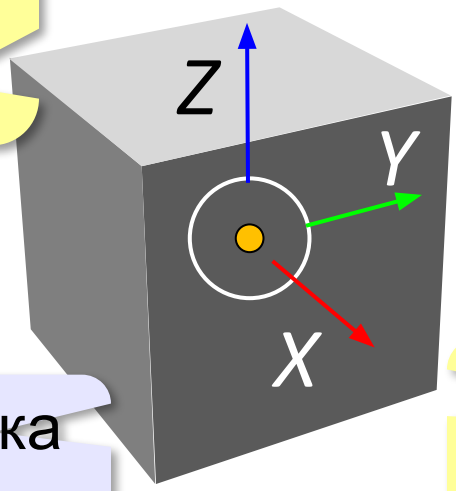
Num/ (*Вид – Глобальный/Локальный вид*) – скрыть все, кроме выделенных объектов

Преобразования

Перемещение:

свободное перемещение в плоскости проекции

включение манипулятора



опорная точка *(origin)*

перемещение вдоль одной оси

G: перемещение в плоскости проекции

ЛКМ: фиксация

G, **X**: перемещение вдоль оси X

G, **Y**: перемещение вдоль оси Y

G, **Z**: перемещение вдоль оси Z

Преобразования

Вращение:

R *(rotate)*: вращение в плоскости проекции

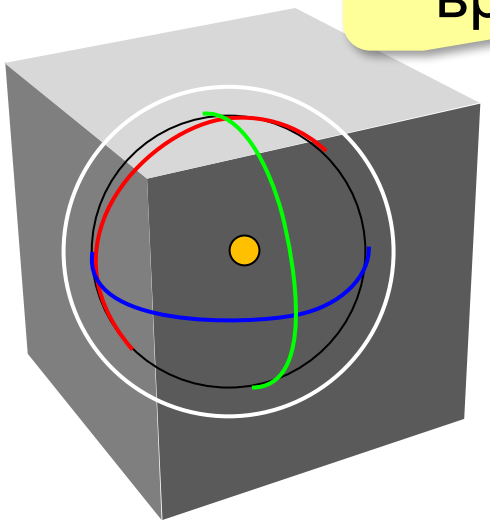
R, **X**: вращение вокруг оси X

R, **Y**: вращение вокруг оси Y

R, **Z**: вращение вокруг оси Z

ЛКМ: закончить

манипулятор вращения



Преобразования

Размеры объекта:

S *(scale)*: все размеры одновременно

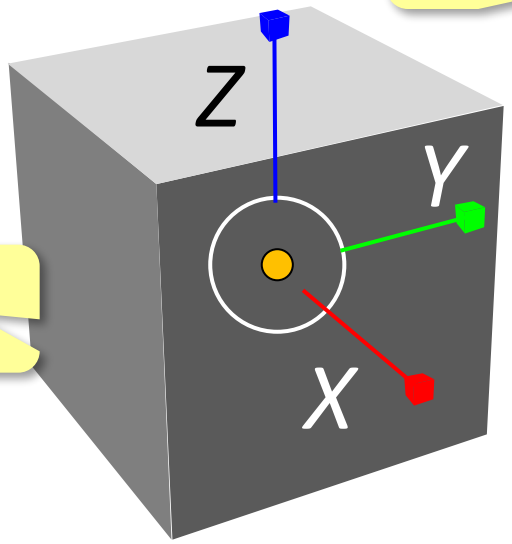
S, **X**: только по оси X

S, **Y**: только по оси Y

S, **Z**: только по оси Z

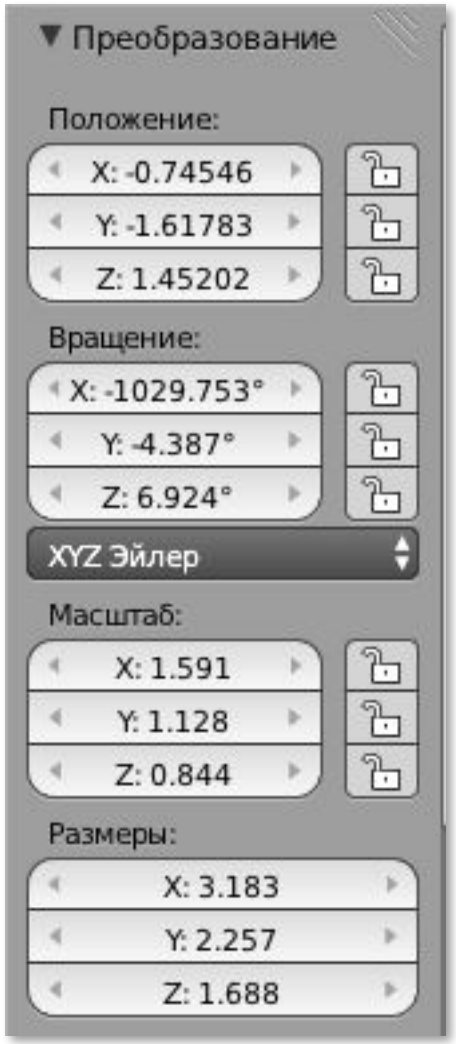
ЛКМ: закончить

по одной оси



все сразу

Преобразования



N: показать/скрыть панель преобразований

Shift+D: создать копию объекта

ЛКМ: зафиксировать

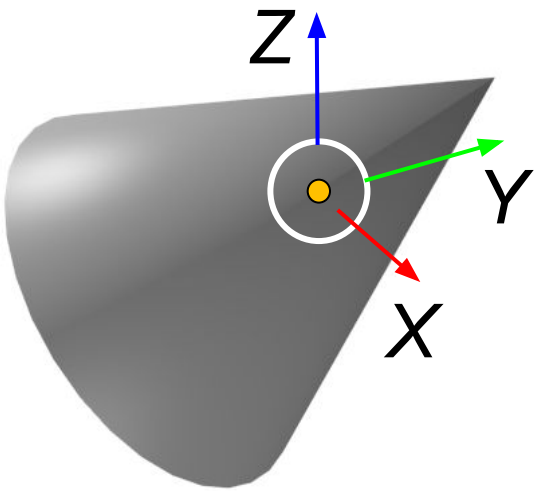
X: перемещать только вдоль оси X

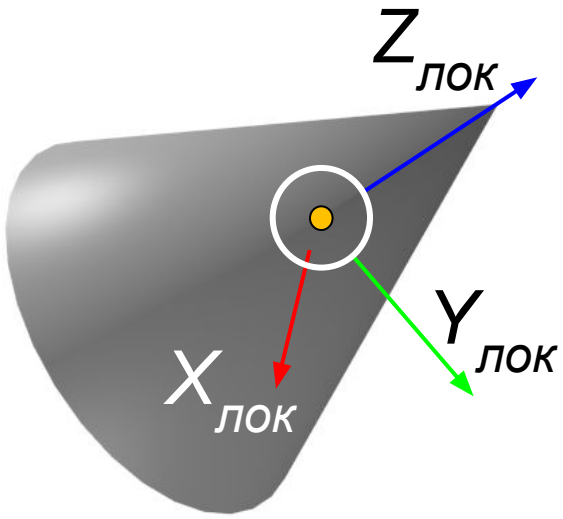
Y: перемещать только вдоль оси Y

Z: перемещать только вдоль оси Z

Del: удалить выделенные объекты

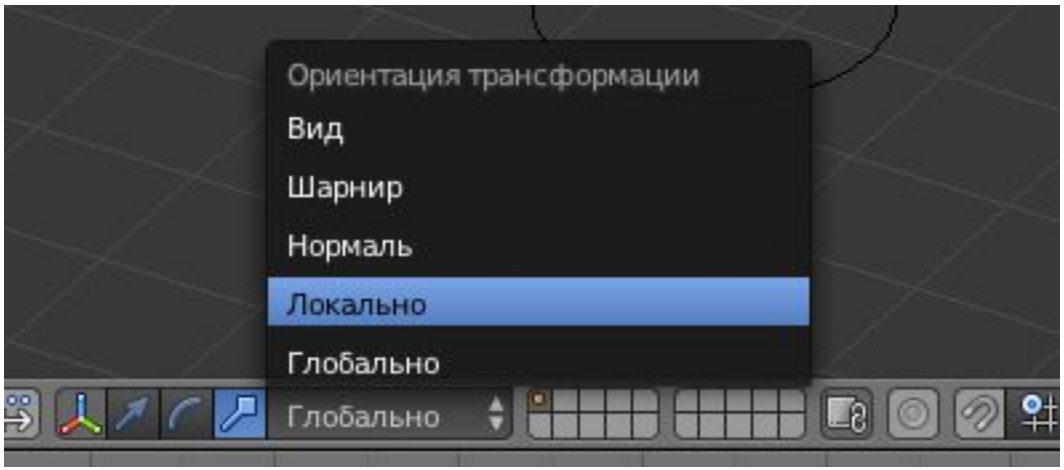
Системы координат





глобальная («мировая»)
не зависит от положения объекта

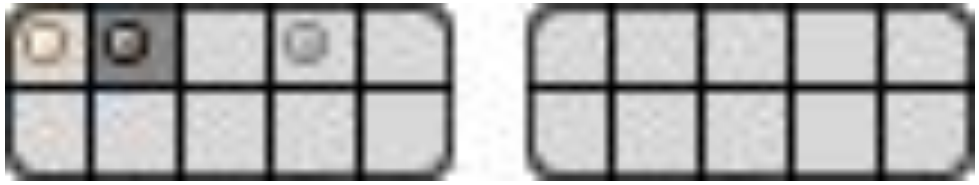
локальная
связана с объектом



Слои

активный слой

есть объекты



Shift+ЛКМ: выделить несколько слоёв

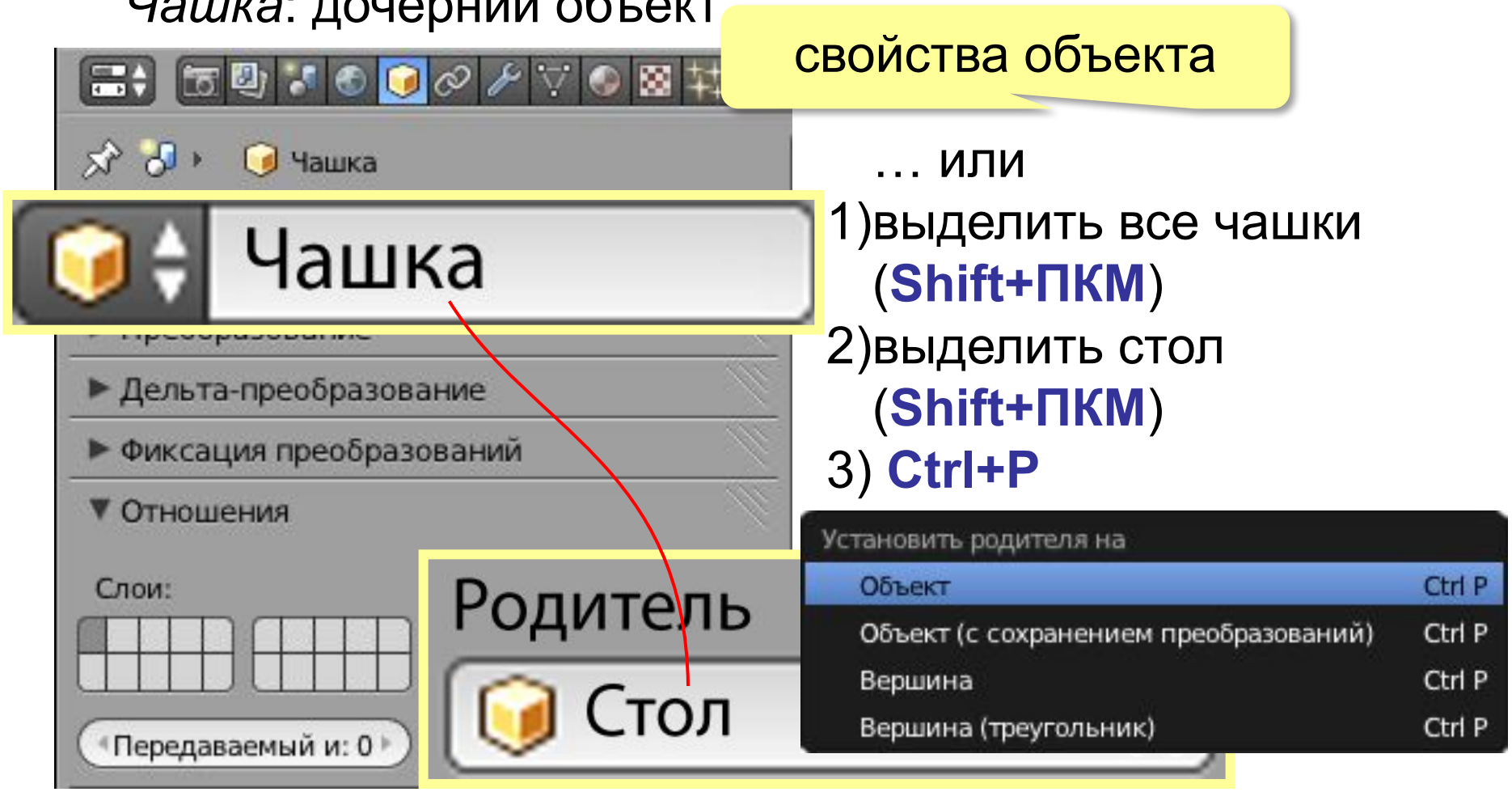
M: перевести объект на другой слой

Связывание объектов

Задача: привязать к столу стоящие на нём предметы.

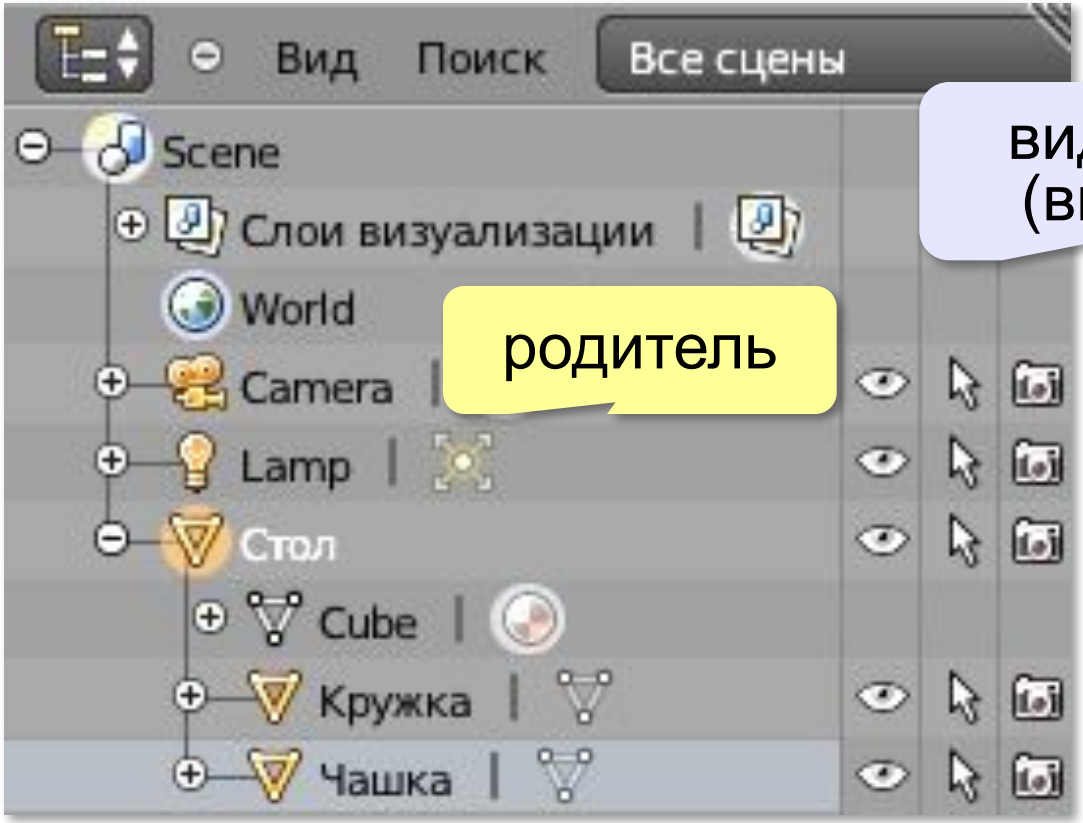
Стол: **родительский** объект

Чашка: дочерний объект



Структура проекта

Окно ***Структура проекта***:



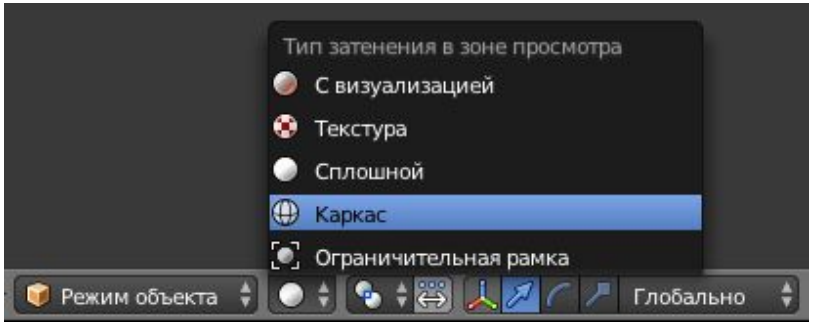
Трёхмерная графика

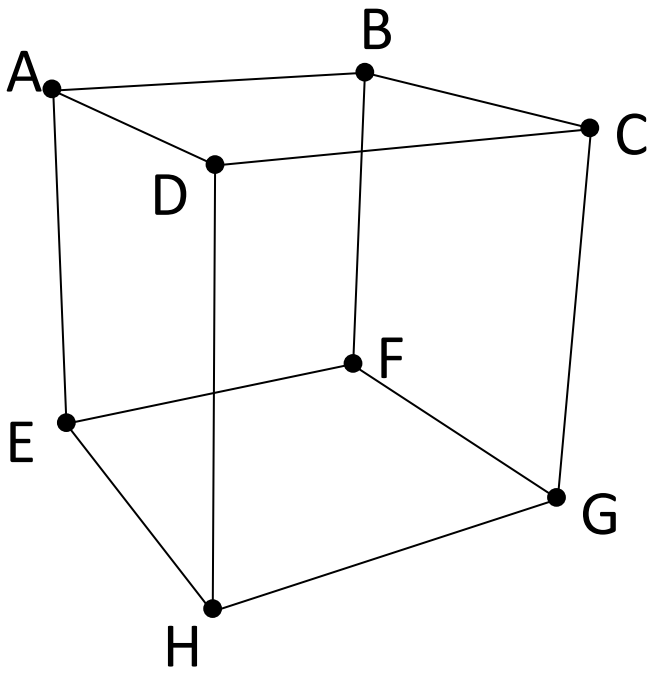
§ 68. Сеточные модели

© К.Ю. Поляков, Е.А. Ерёмин, 2013

Что такое сеточная модель?

Режим ***Каркас*** (*Wireframe*):





- 8 **вершин** (*vertex*) – A, B, C, D, E, F, G и H;
- 12 **рёбер** *(edge)* – AB, AD, BC, CD, EF, EH, FG, GH, AE, BF, CG и DH
- 6 **граней** (*face*) – ABCD, EFGH, ABFE, CDHG, ADHE и BCGF

Что такое сеточная модель?

Сеточная модель (*mesh*) – это поверхность, которая строится на сетке из рёбер.

Грани:

- треугольники
- четырехугольники
- многоугольники (полигоны)

полигональные модели

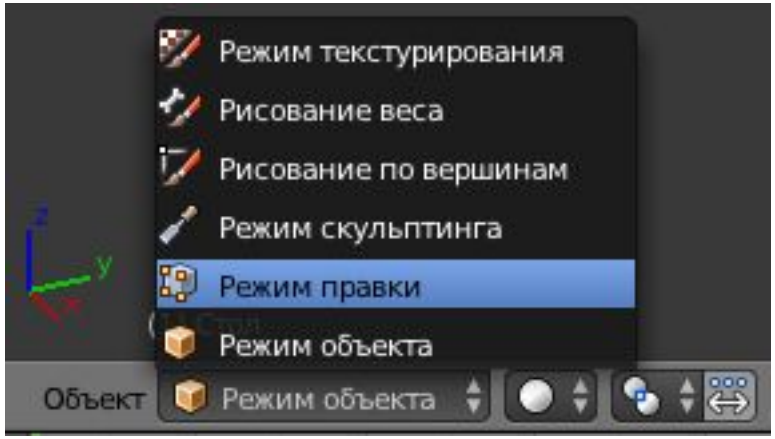
Все тела (в т.ч. сферы!) состоят из граней!

Векторный формат: хранятся координаты вершин.

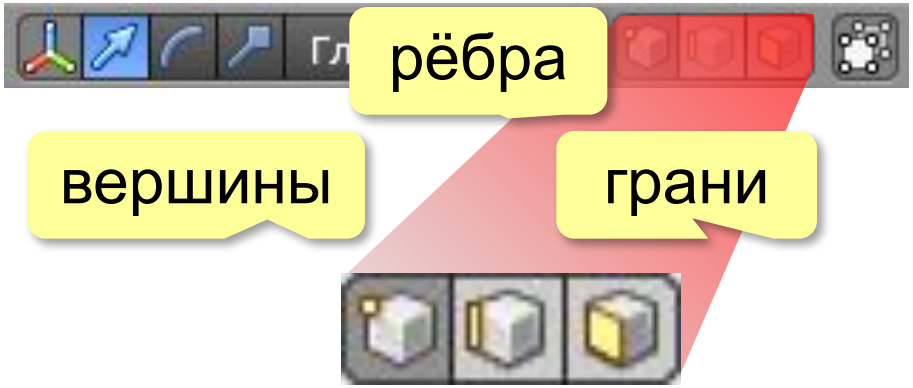
Чем больше граней, тем сложнее расчёты!

Редактирование сетки

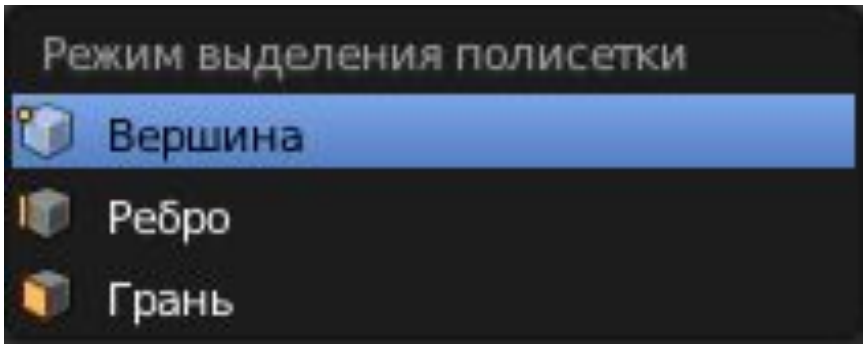
Tab – перейти в режим редактирования сетки (*Edit mode*)



Выбор объектов:



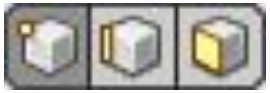
или **Ctrl+Tab**:



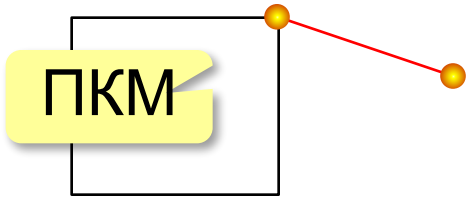
+Shift – несколько

выделять только видимые

Редактирование сеточной модели

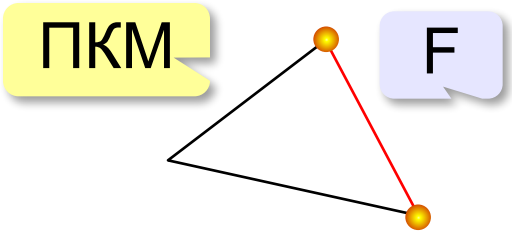
в режиме работы с вершинами: 

Ctrl+ЛКМ – новая **вершина**, соединённая с выделенной



Ctrl+V – меню для вершин

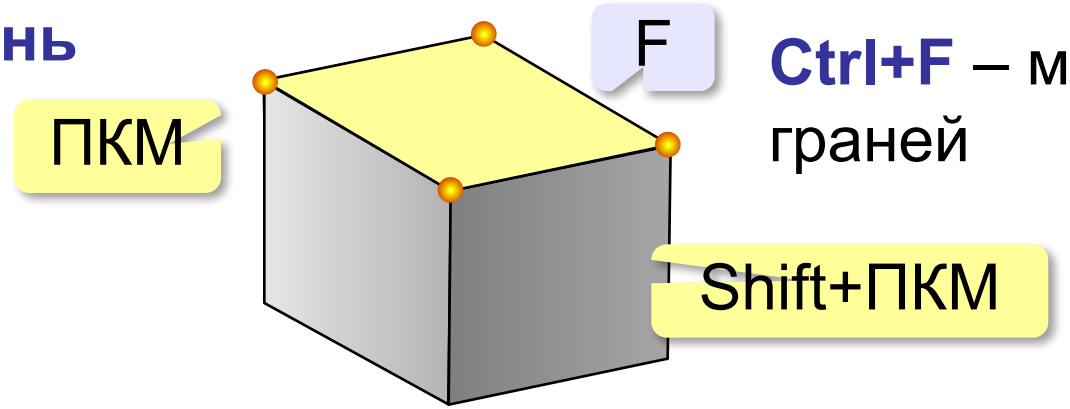
новое **ребро**



Ctrl+E – меню для рёбер

Shift+ПКМ

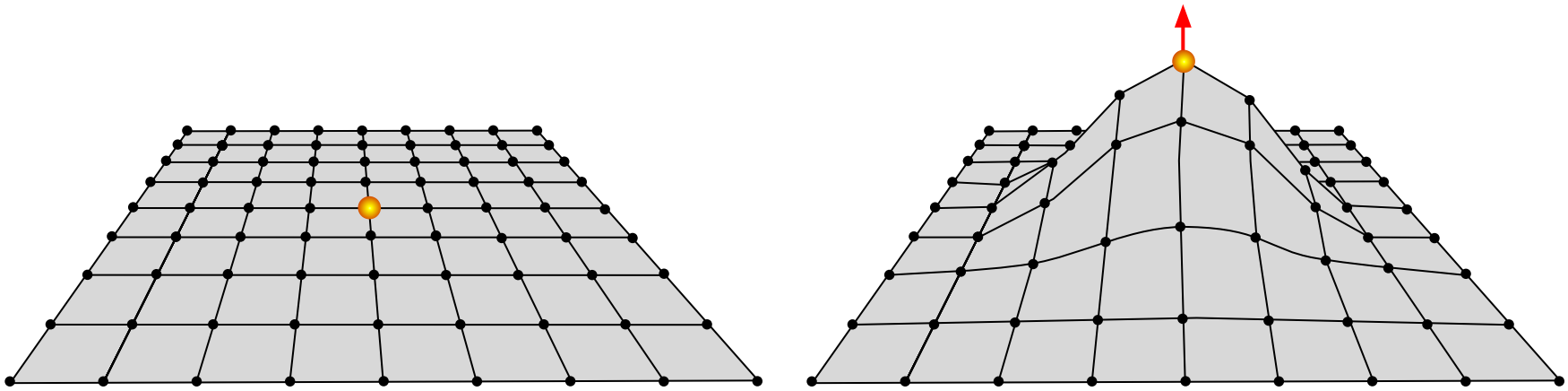
новая **грань**



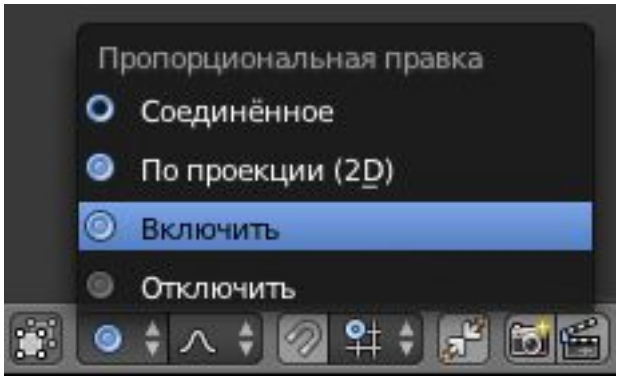
Ctrl+F – меню для граней

Редактирование сеточной модели

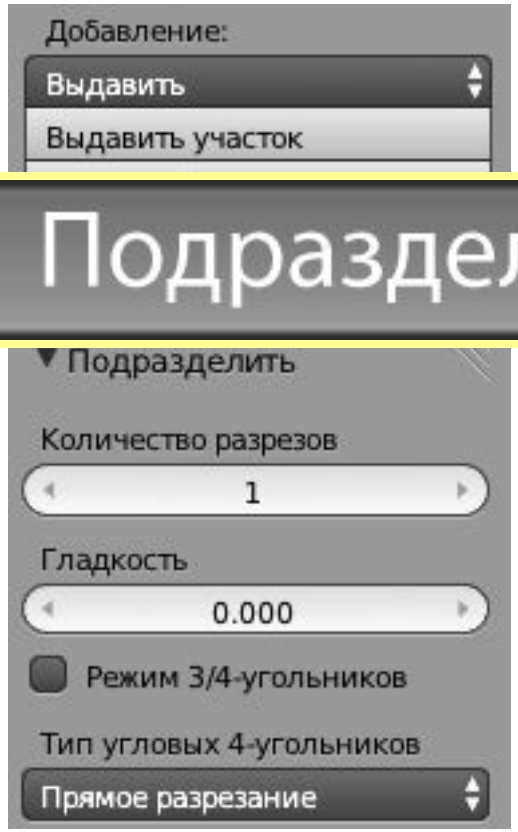
Пропорциональное редактирование:



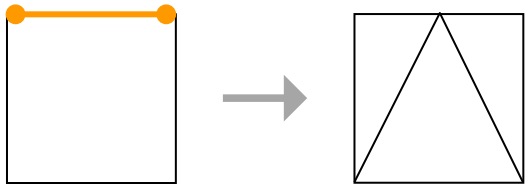
включение:



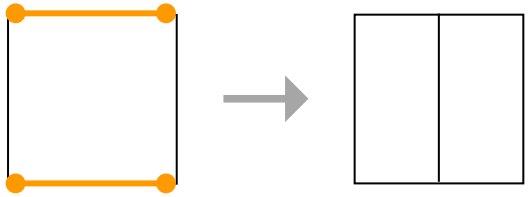
Деление рёбер и граней



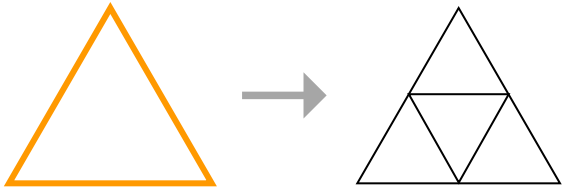
одно ребро:



два ребра:



треугольная грань:

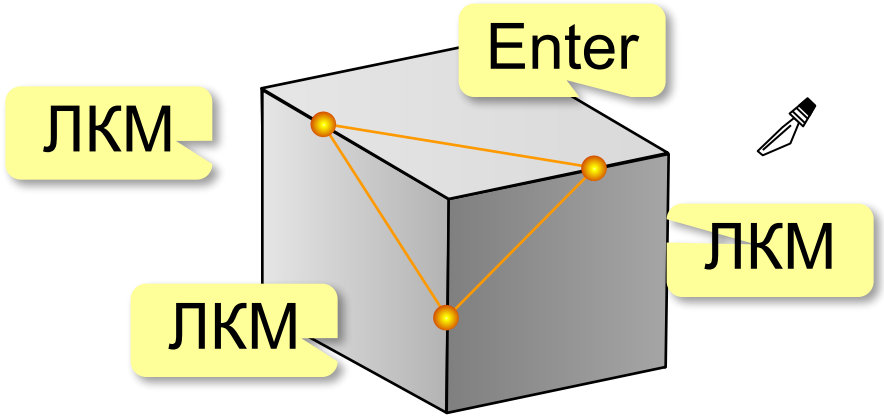


четырёхугольная грань:

Деление рёбер и граней: Нож

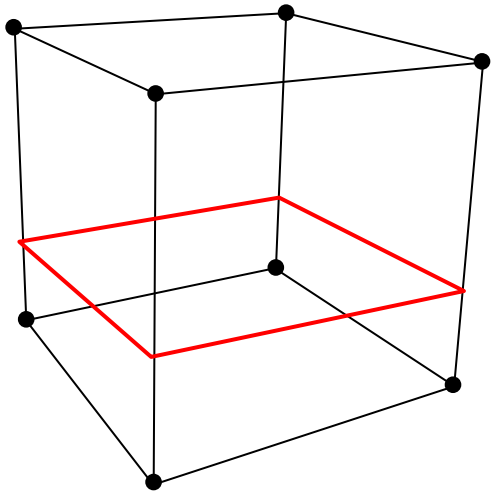
Нож (*Knife*):

при нажатой **К** резать **ЛКМ**, *Enter* – завершить

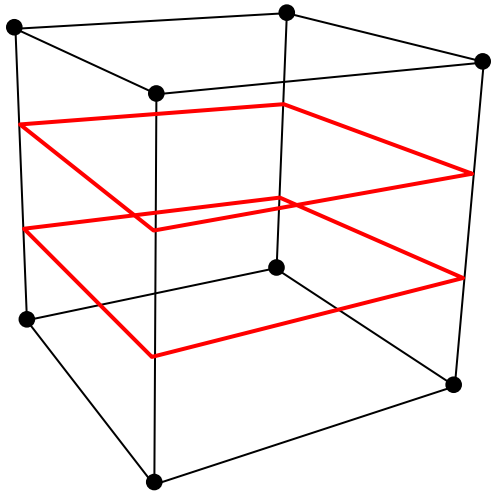


Деление рёбер и граней

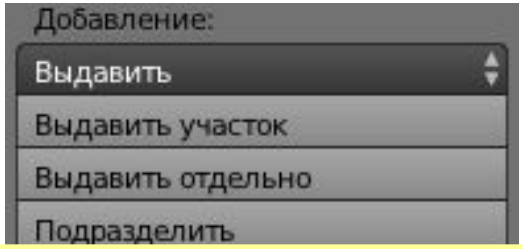
одиночное сечение



множественное сечение



Разрезать петлей со сдвигом (*Loop Cut and Slide*):



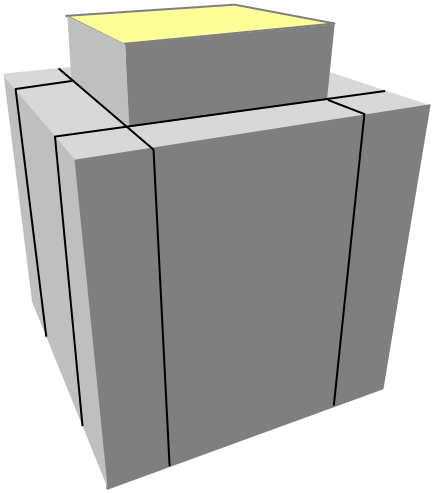
Колёсико мыши – менять число сечений

ЛКМ – зафиксировать число сечений

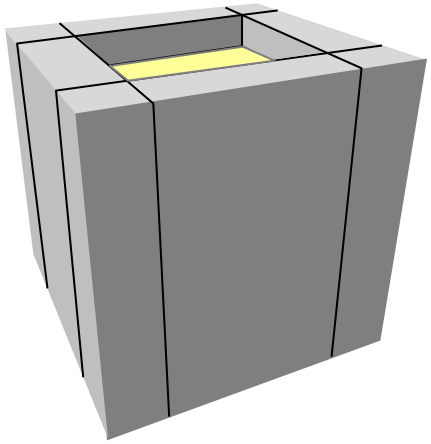
ЛКМ – зафиксировать сдвиг

Выдавливание (*Extrude*)

выдавливание наружу



выдавливание вовнутрь



E (*extrude*): выдавить выделенную грань по нормали

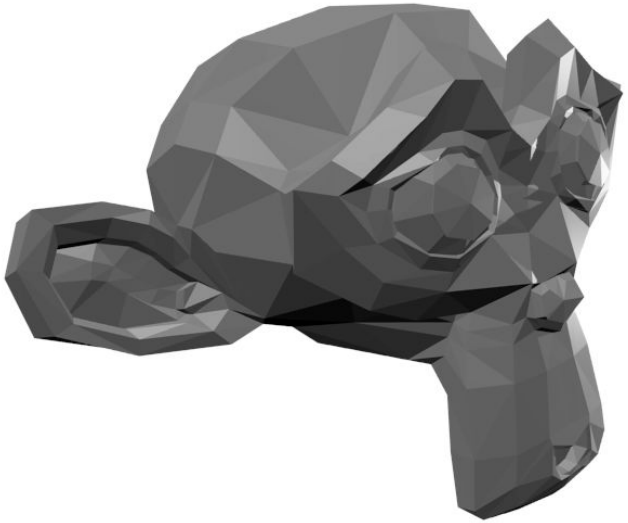
E, **X**: выдавить вдоль оси X

E, **Y**: выдавить вдоль оси Y

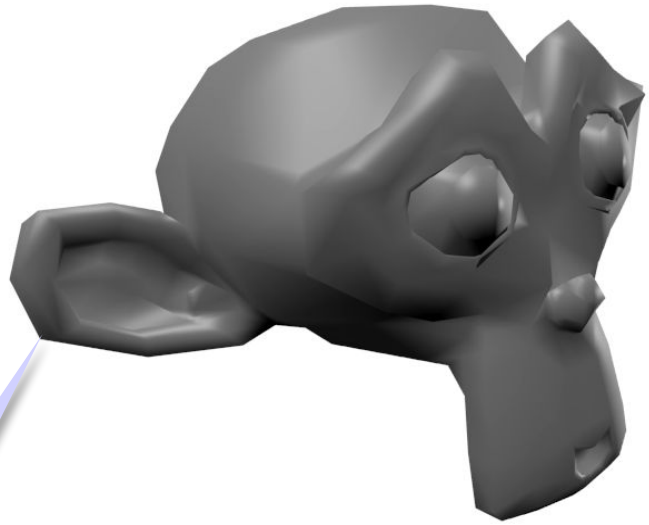
E, **Z**: выдавить вдоль оси Z

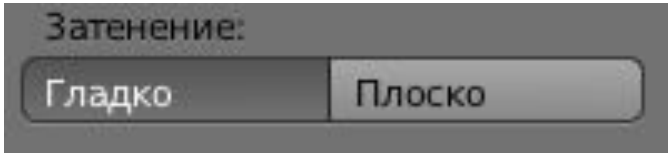
Сглаживание (*Smooth*)

без сглаживания



со сглаживанием





! Сглаживание (затенение) только при выводе (число граней не меняется)!

Трёхмерная графика

§ 69. Модификаторы

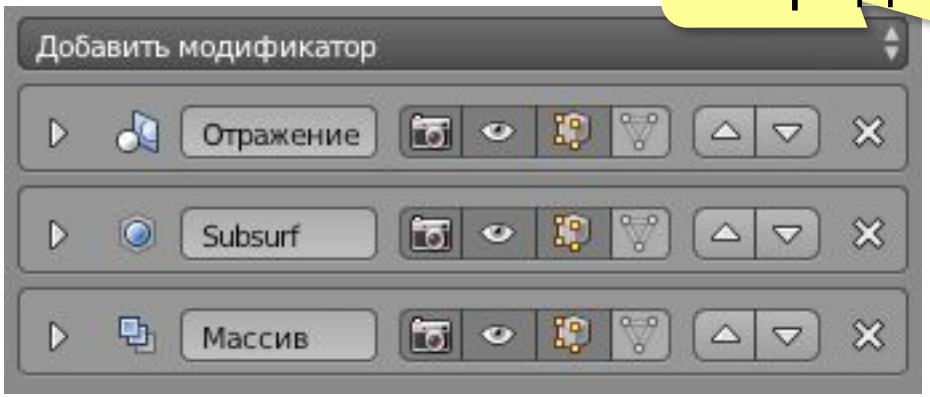
Что такое модификатор?

Модификатор – это преобразование объекта, которое выполняется автоматически при выводе проекции на экран или построении готового изображения (*рендеринге*).

! Это неразрушающая операция!

Стек модификаторов:

изменить порядок



применяется первым

применяется последним

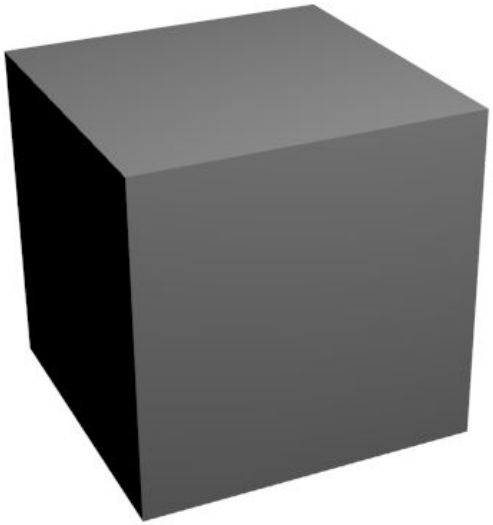
Применить модификатор = изменить сетку!

Сглаживание

Подразделение поверхности

(*Subsurf* = *Subdivision surface*):

без сглаживания



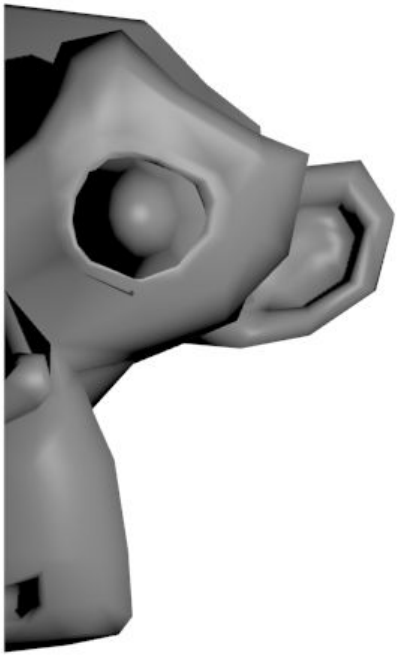
уровень 1
(24 грани)

уровень 4
(1536 граней)

Симметрия

Зеркало (*Mirror*):

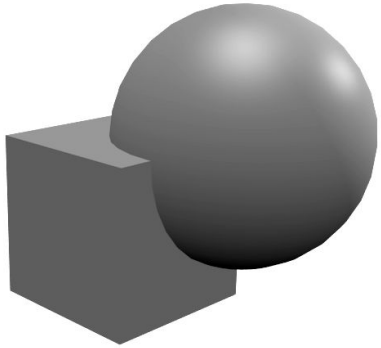
половина головы



с модификатором *Зеркало*

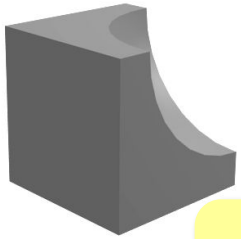
Логические операции

объединение (*Union*):

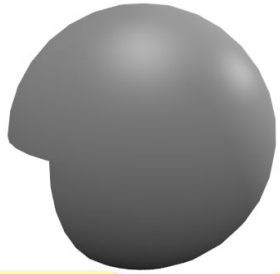


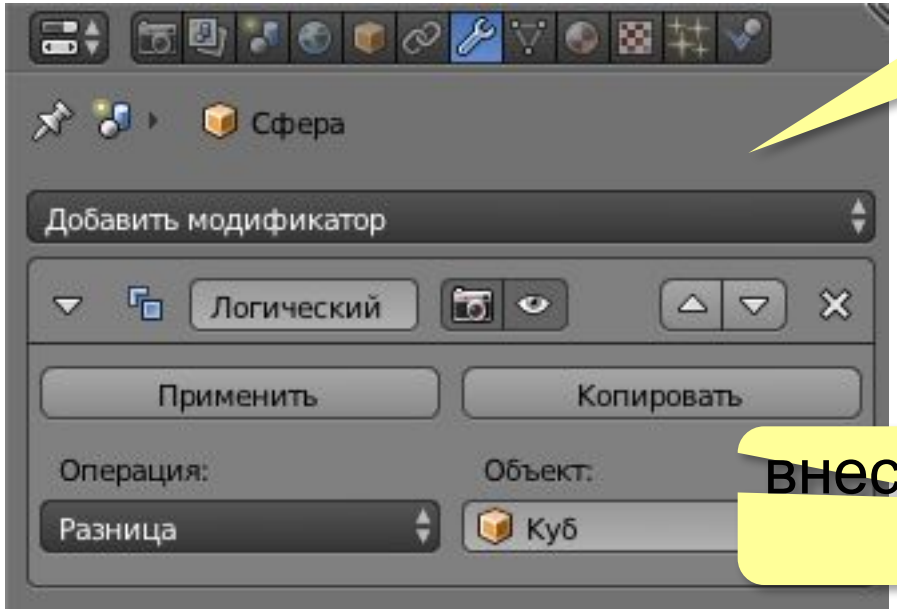
пересечение (*Intersection*):

«куб минус сфера» (*Difference*):



«сфера минус куб» (*Difference*):

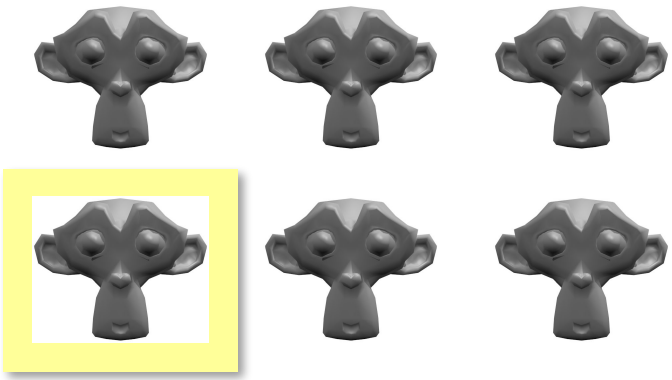




! При сдвиге куба результат меняется!

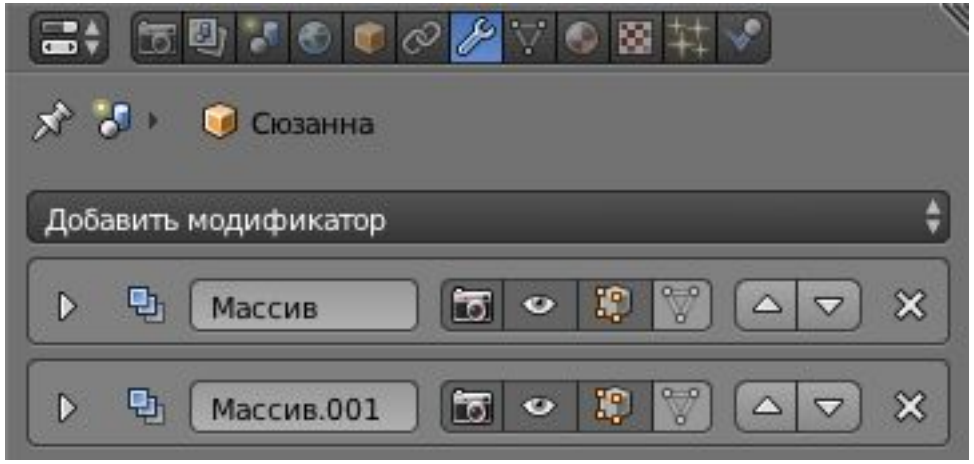
внести изменения в сетку

Массив



оригинал

копии

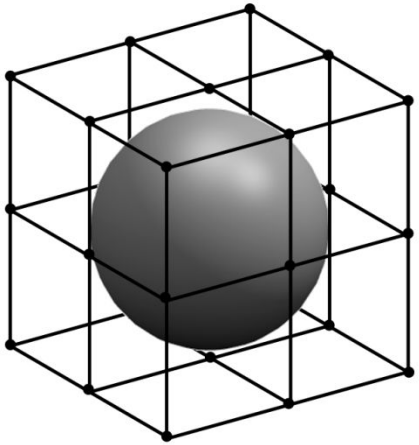


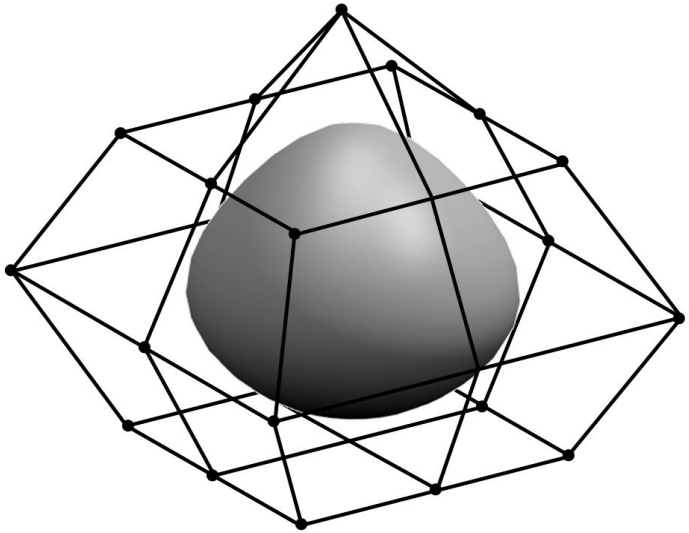
! Копии сохраняют связь с оригиналом!

= разорвать связь

Деформация

Решётка (*Lattice*):



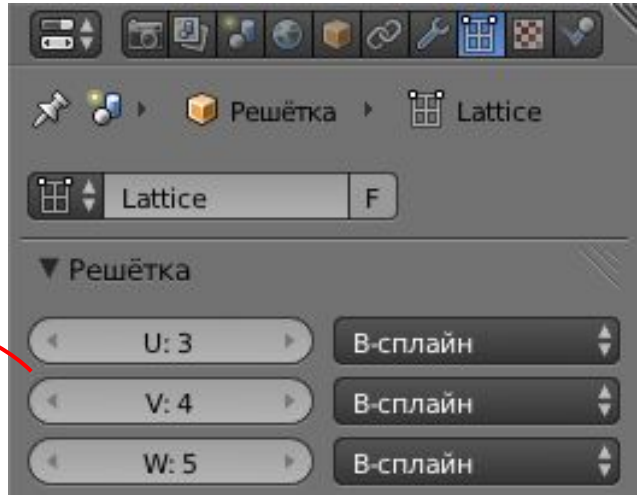


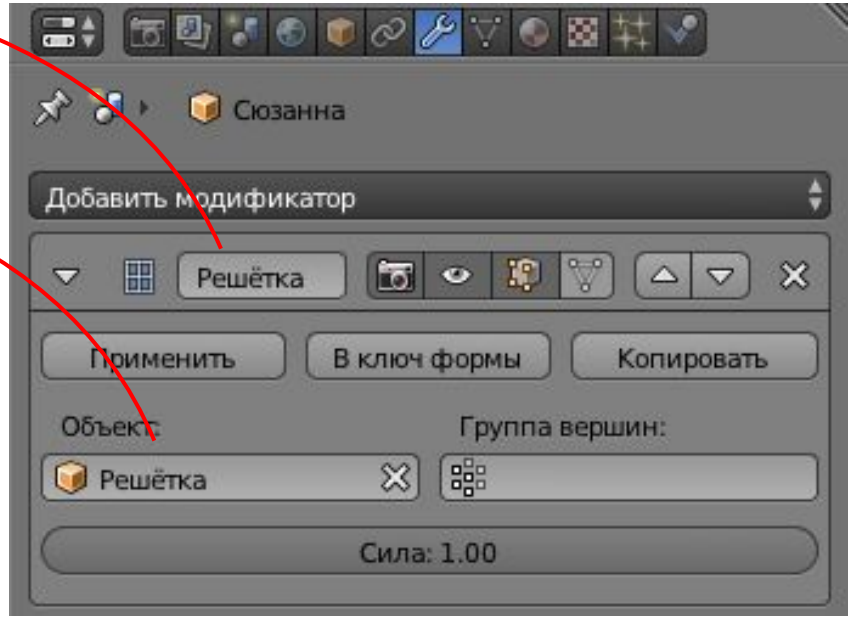
! Полезно при анимации!

Деформация

Решётка (*Lattice*):

1) *Добавить – Решётка*
2) выбрать число разбиений сторон решетки
3) Заключить объект в решётку
4) Применить модификатор **Решётка**
5) выбрать нужный объект-решётку
6) перемещать узлы решётки – будет изменяться форма объекта





Трёхмерная графика

§ 70. Кривые

Зачем нужны кривые?

Кривые – вспомогательные векторные объекты.

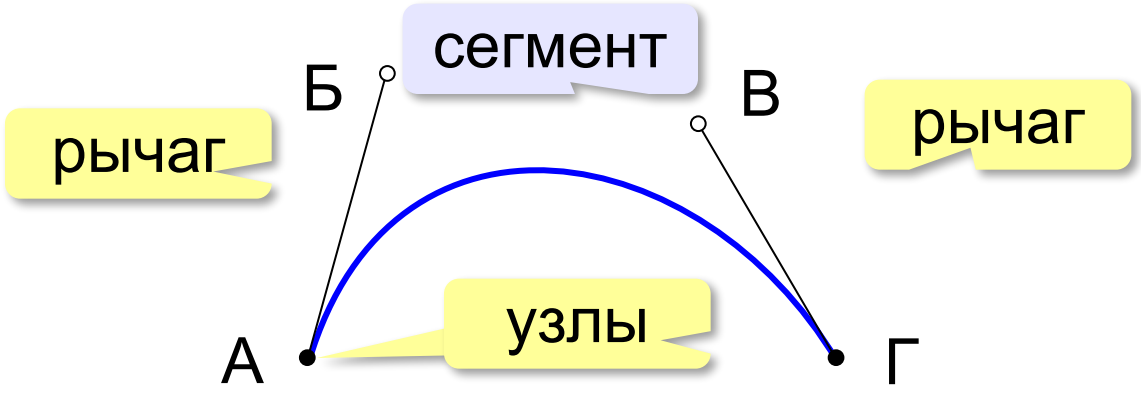
- форма тела вращения
- поперечное сечение трубы
- траектории движения объектов при анимации
- искривление текстовых строк
- моделирование проводов и нитей

Blender – два типа кривых:

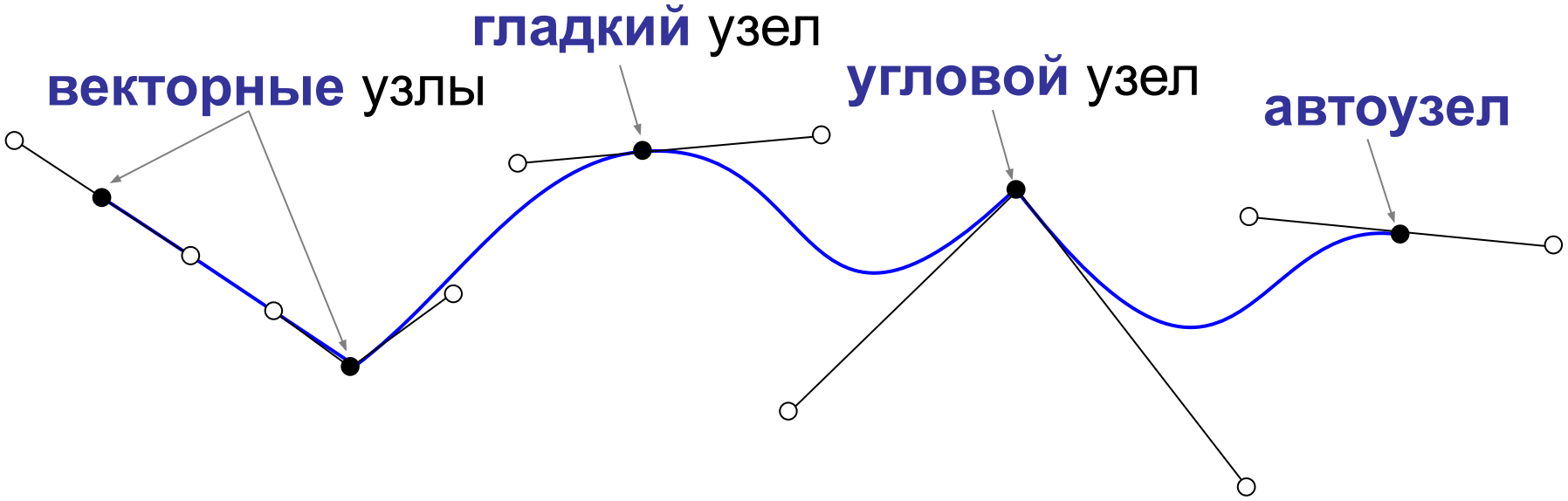
- кривые Безье (*Bezier Curves*)
- кривые NURBS (*Non Uniform Rational B-Splines* – неравномерные рациональные B-сплайны)

Кривые Безье

Сегмент задается четырьмя точками (кривая 3-го порядка)

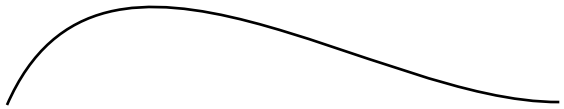


Типы узлов:



Редактирование кривых

Добавить – Кривая – Безье



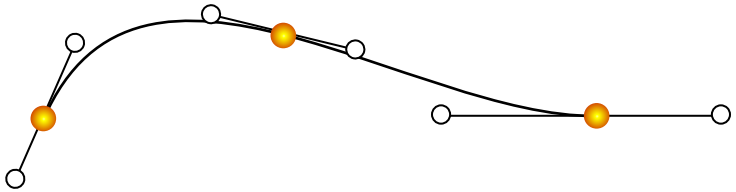
перемещение, вращение, изменение размеров – как обычно

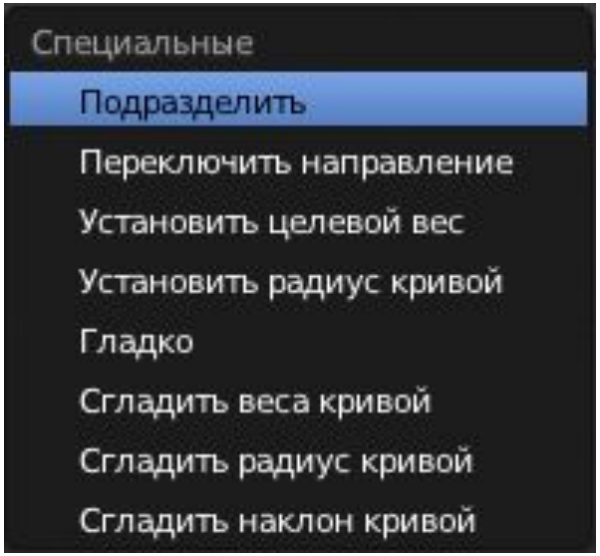
Tab – перейти в режим редактирования узлов

V – меню для выбора типа выделенного узла

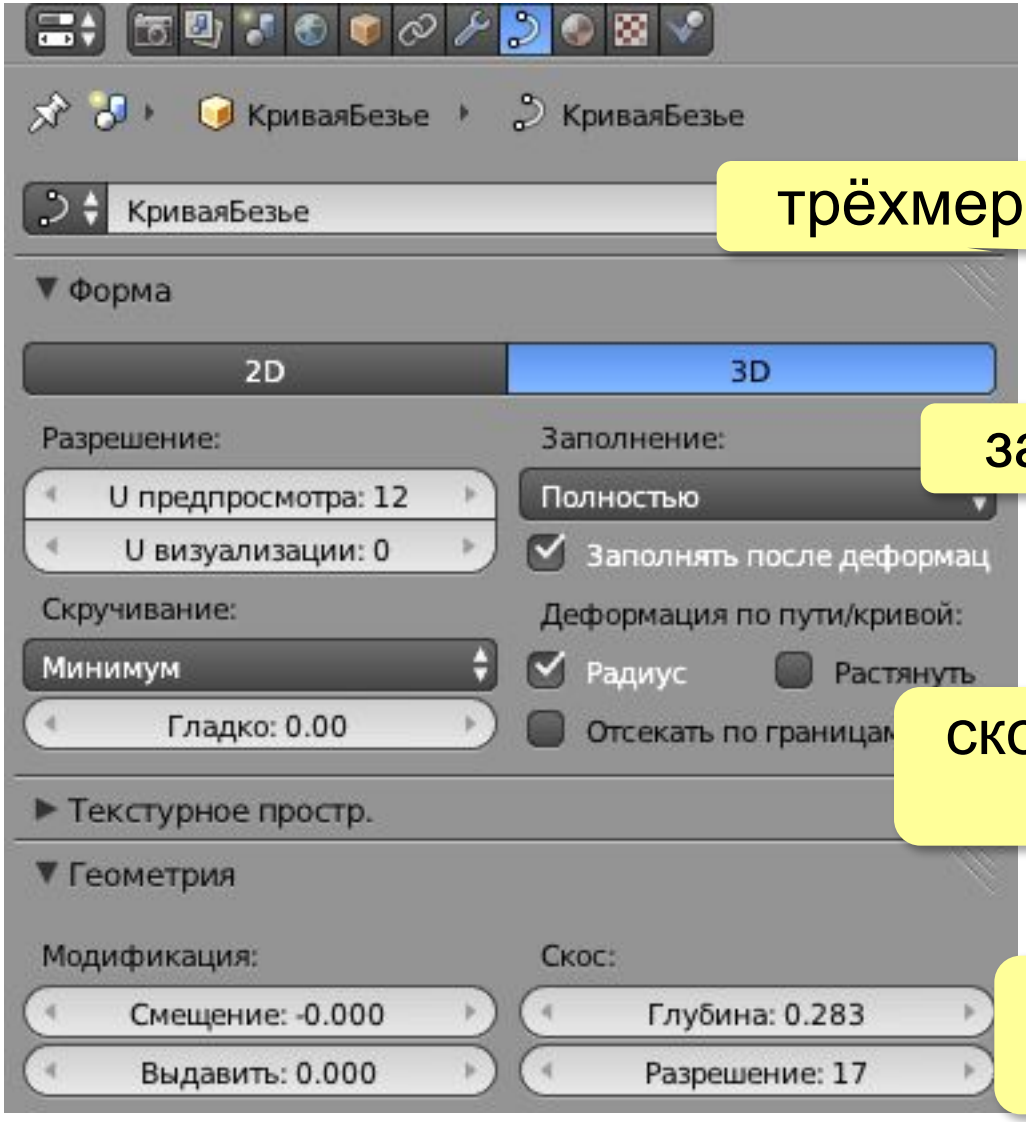
Добавление узла :

1) выделить два соседних узла
2) **W** – меню специальных команд (*Specials*)
3) выбрать *Подразделить*





Моделирование трубы (провода)



трёхмерная

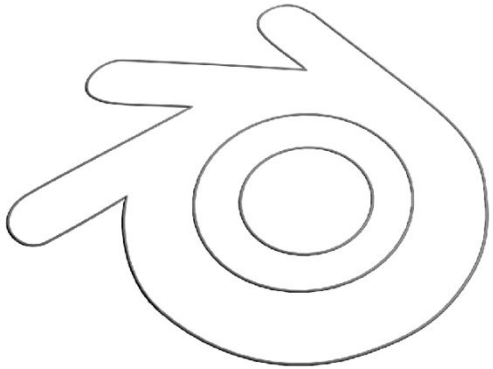
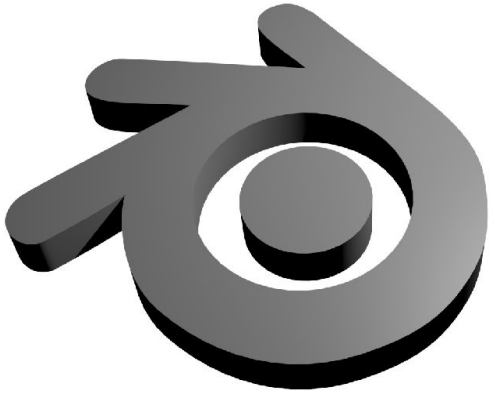
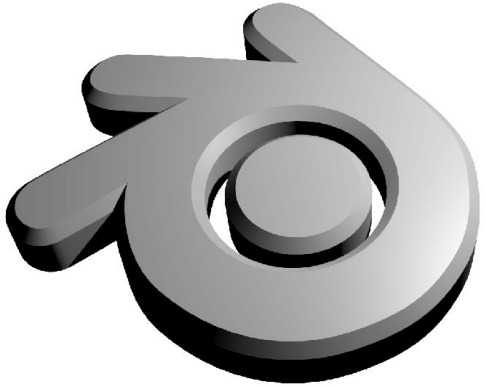
заполнять полностью

скос определяет диаметр

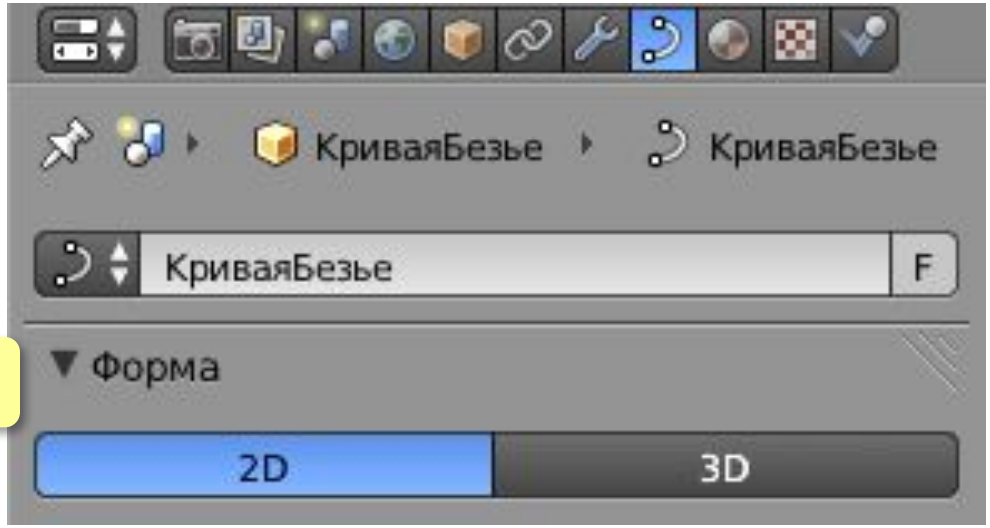
разрешение определяет гладкость

Пластины

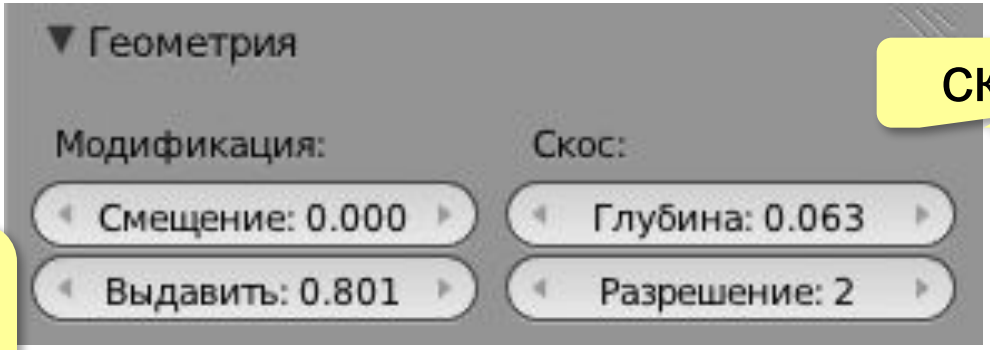
Alt+C– замкнуть кривую

контур	выдавливание	выдавливание+скос
		

Пластины



двухмерная

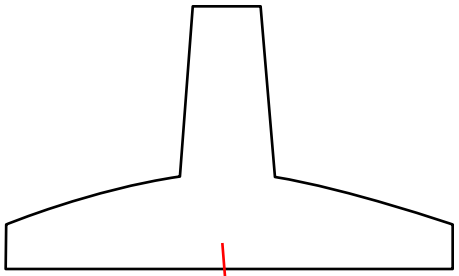


скос

выдавливание (толщина пластины)

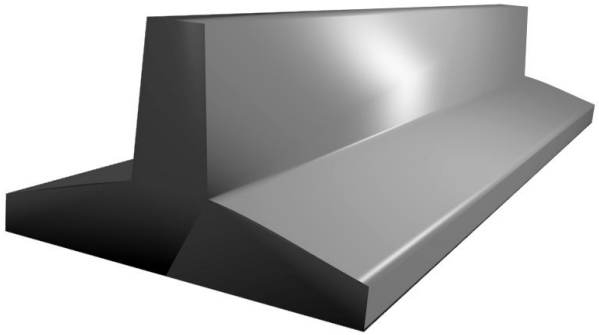
Профили («лофтинг»)

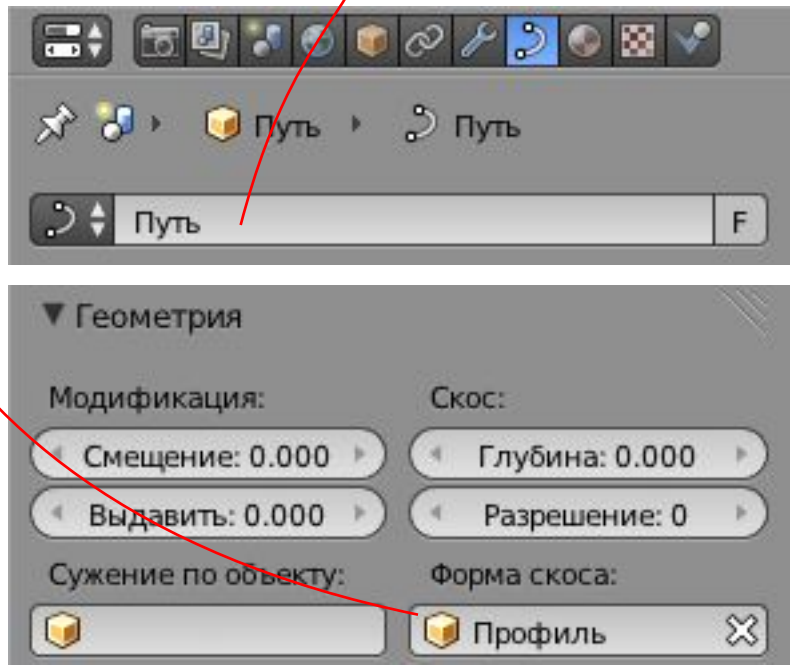
сечение



путь

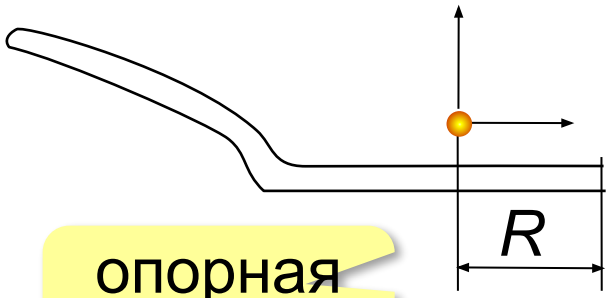
результат





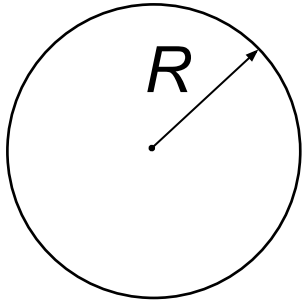
Тела вращения

сечение



опорная точка

путь



результат

Перемещение опорной точки:

Объект – Преобразование – Опорную точку к 3D-курсору
(*Object – Transform – Origin to 3D Cursor*)

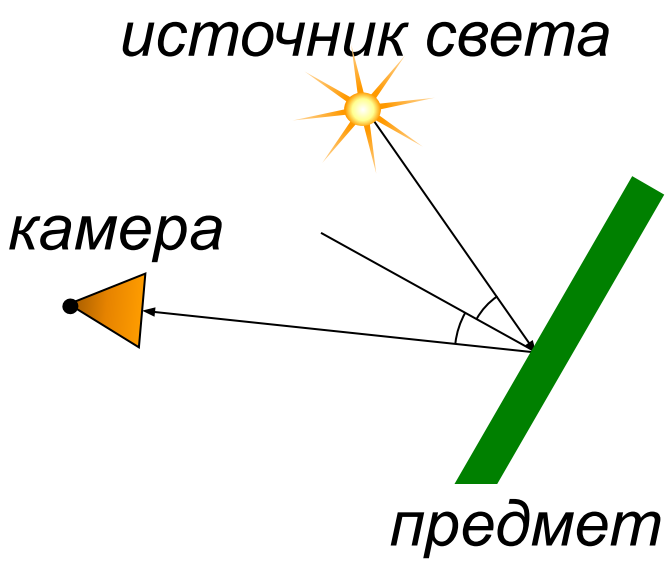
Трёхмерная графика

§ 71. Материалы и текстуры

Как мы видим цвет предметов?

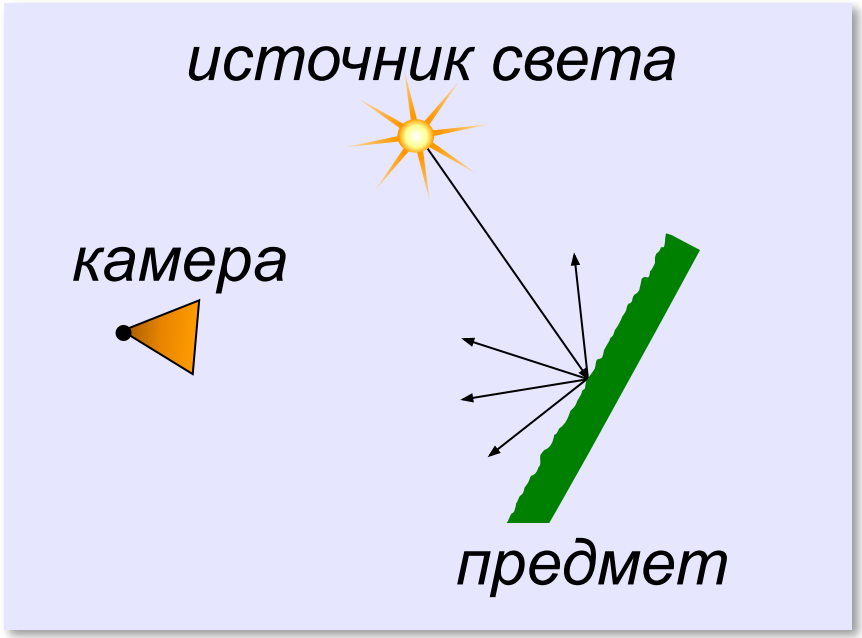
Глаз видит лучи, отражённые от предметов!

зеркальное отражение



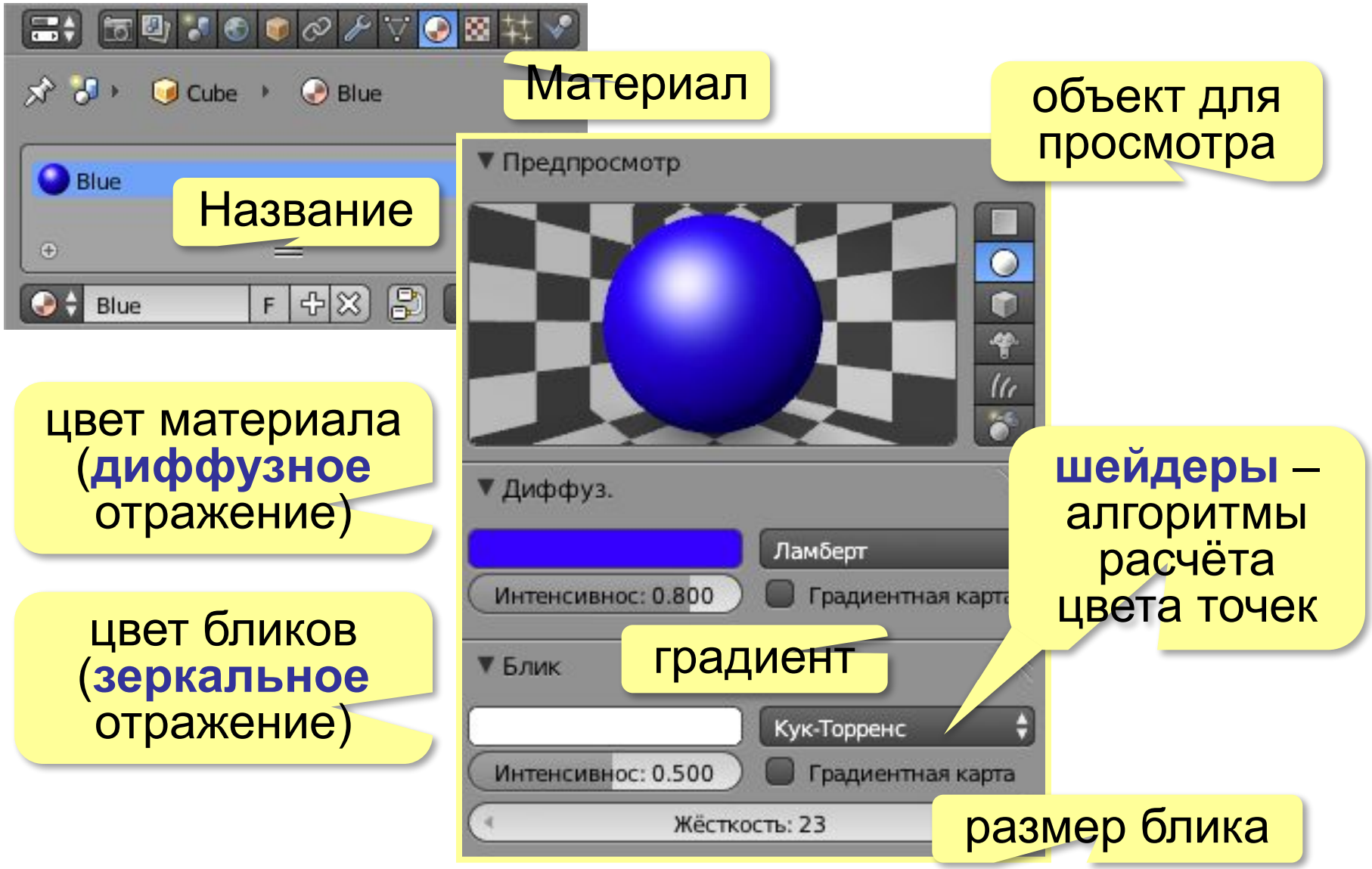
- поверхность идеально ровная
- угол отражения равен углу падения

диффузное отражение

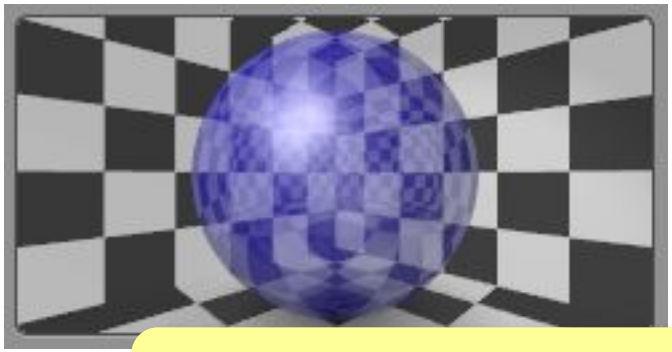


- размер неровностей больше длины волны
- отражение во всех направлениях

Простые материалы



Простые материалы

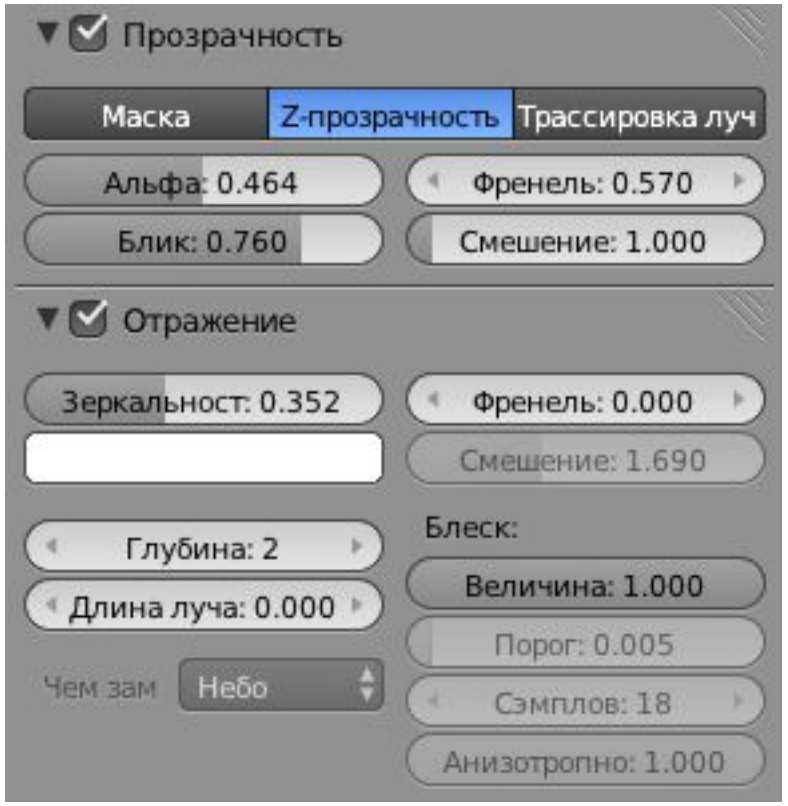


непрозрачность материала

непрозрачность блика

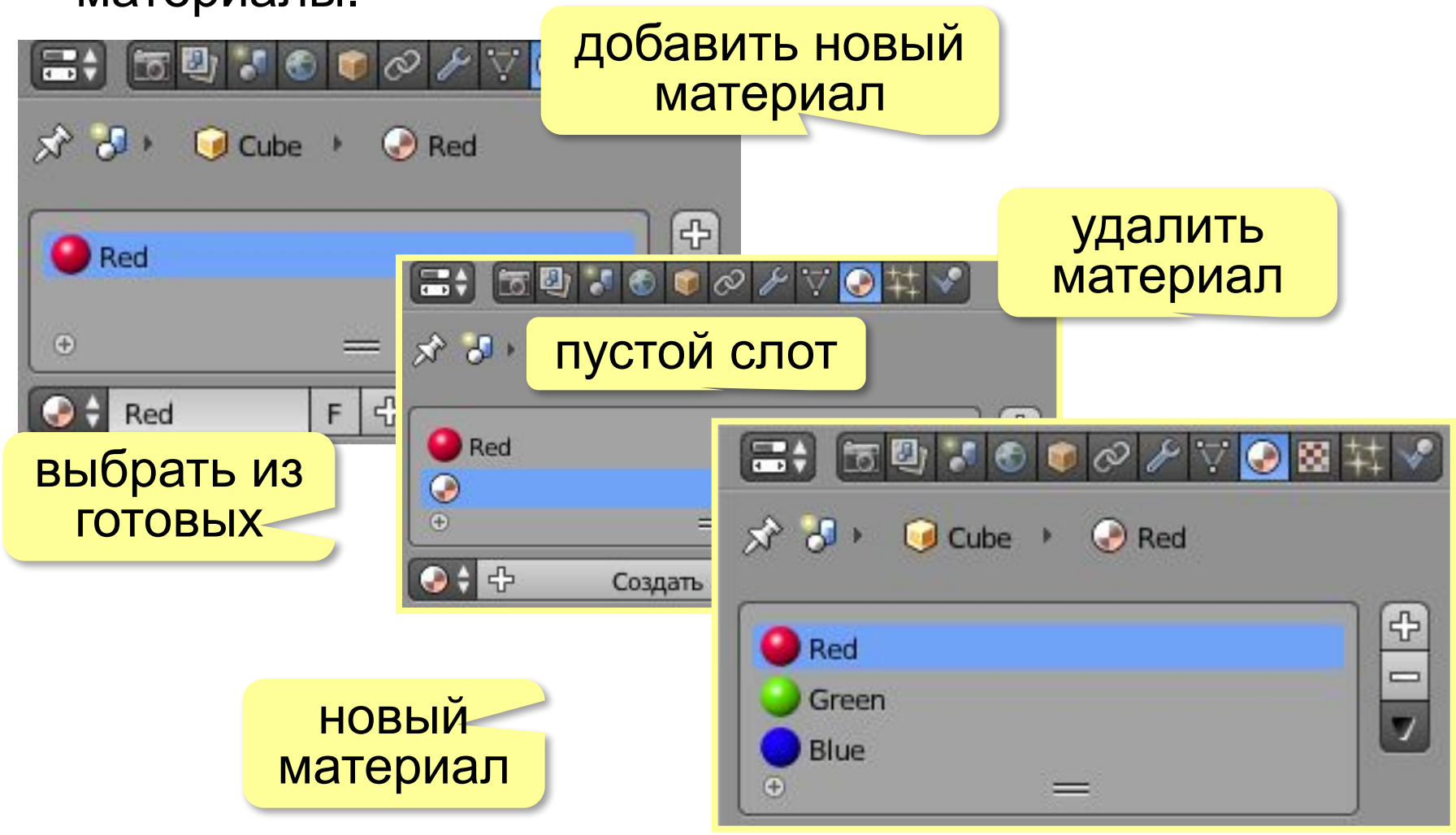
отражение окружающих прежметов

! Прозрачность и отражение!



Многокомпонентные материалы

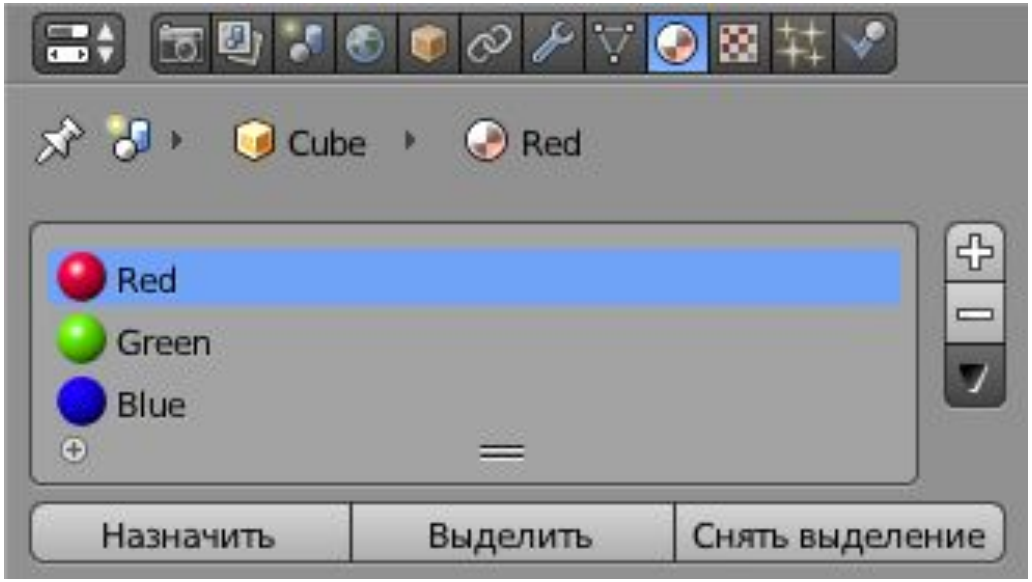
Задача: для разных граней объекта использовать разные материалы.



Многокомпонентные материалы

Назначение материала граням:

Tab – перейти в режим правки *(Edit mode)*



назначить материал выделенным граням

выделить грани с этим материалом

снять выделение граней с этим материалом

Текстуры

Текстура — точечное (растровое) изображение, которое накладывается на поверхность для изменения окраски или имитации рельефа.

рисунок на сфере

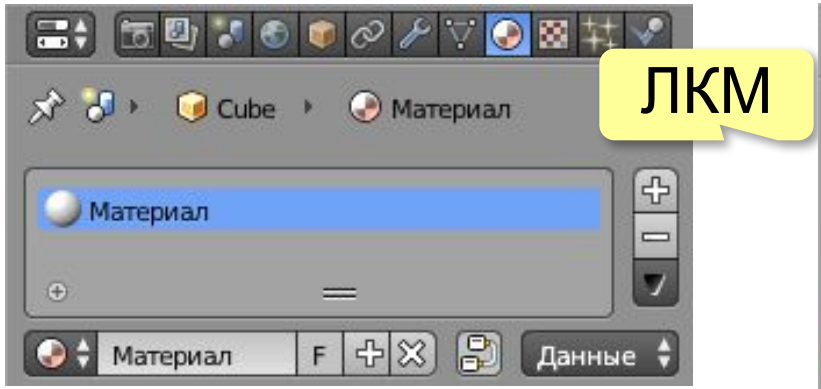
имитация рельефа

Типы текстур:

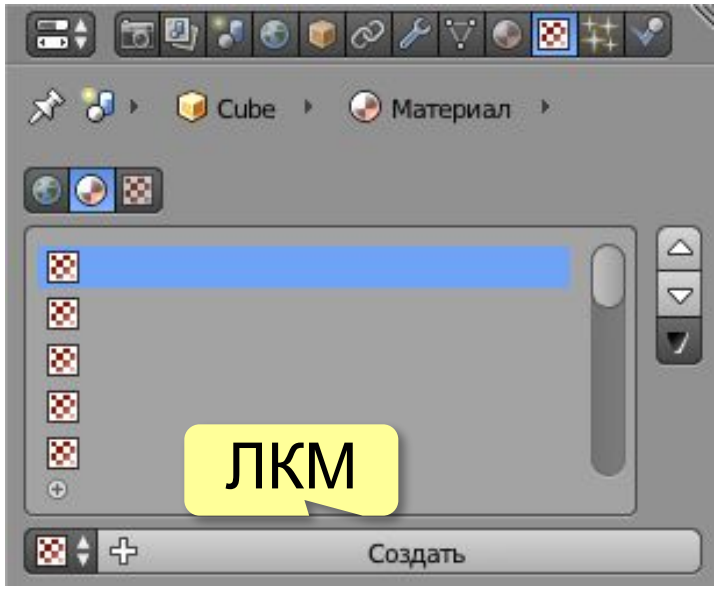
- готовые изображения
- *процедурные* текстуры (строятся по некоторому алгоритму)

Текстуры

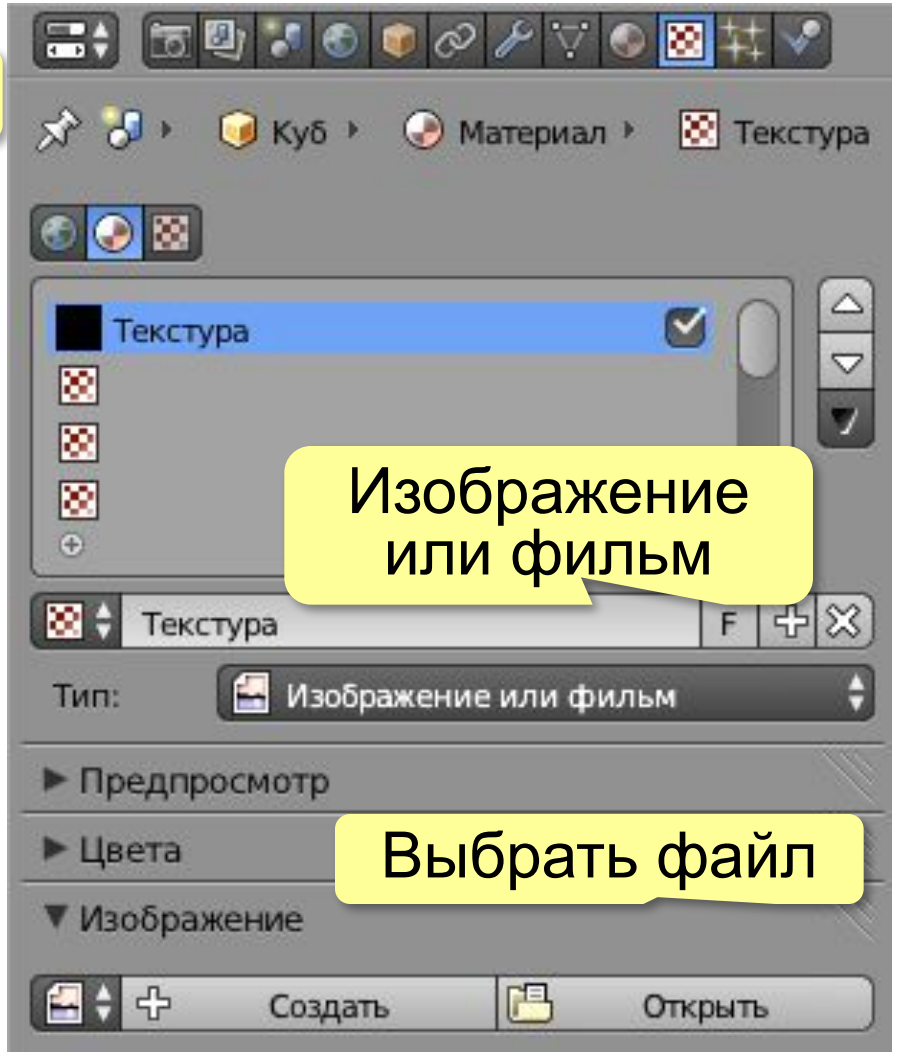
1) Создать новый материал



2) Создать новую текстуру

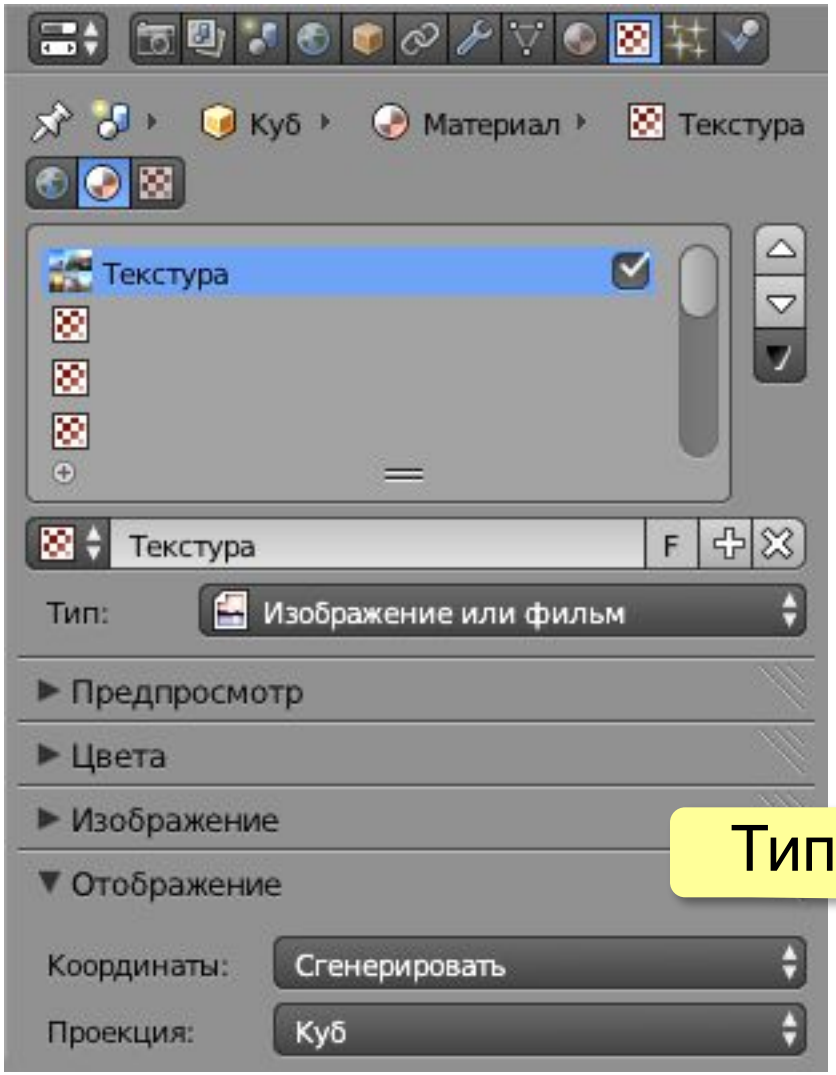


3) Выбрать файл с рисунком



Текстуры

4) Выбрать тип объекта

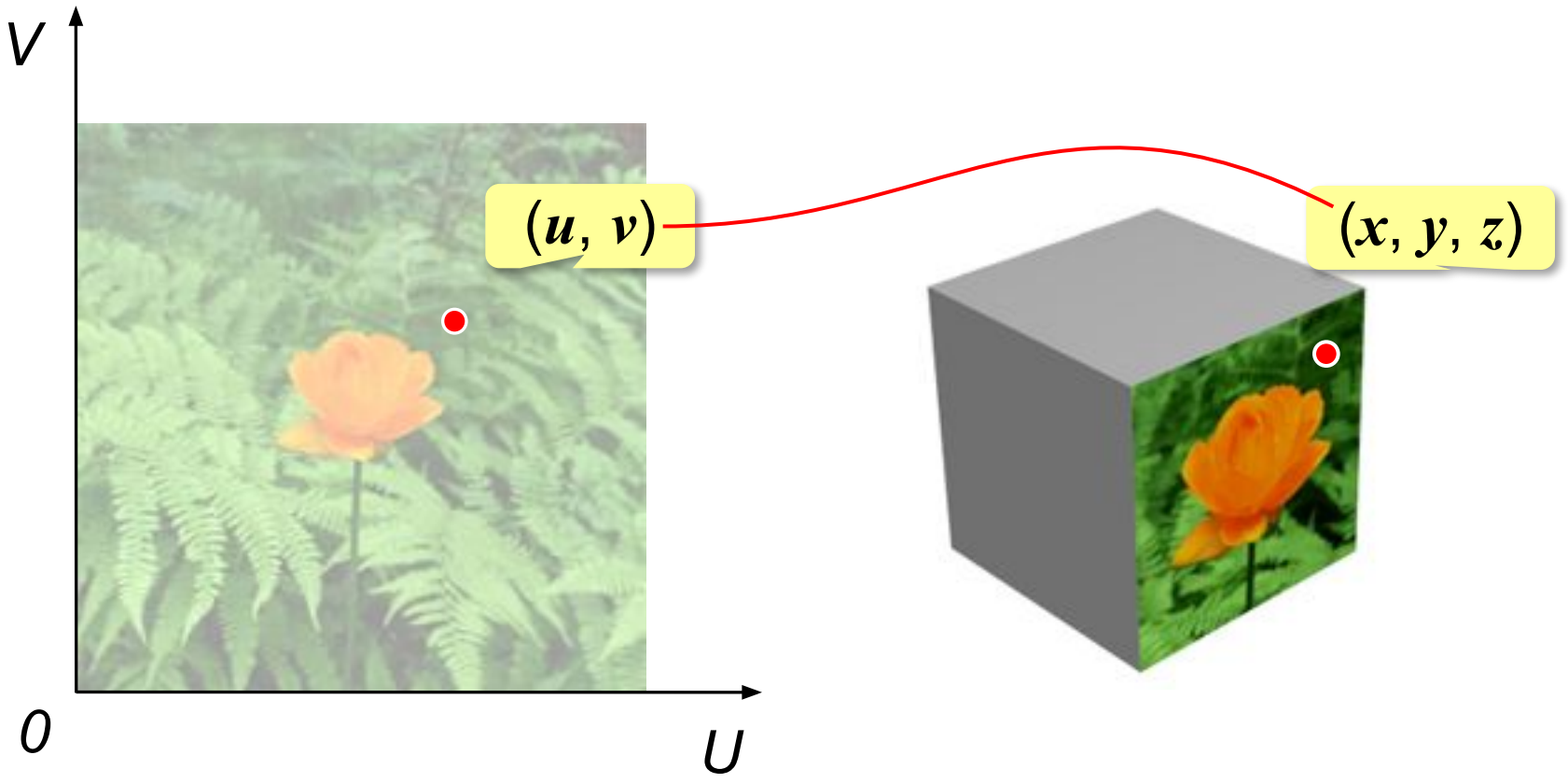


Тип объекта

F12 – построить изображение с текстурой (выполнить рендеринг)!

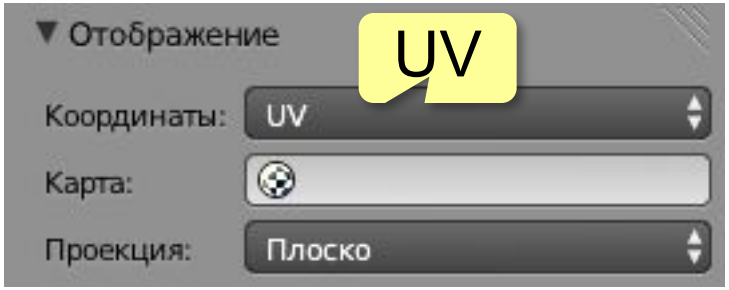
Текстуры: UV-развертка

UV-развёртка (*UV-mapping*): $(x, y, z) \rightarrow (u, v)$



Текстуры: UV-развёртка

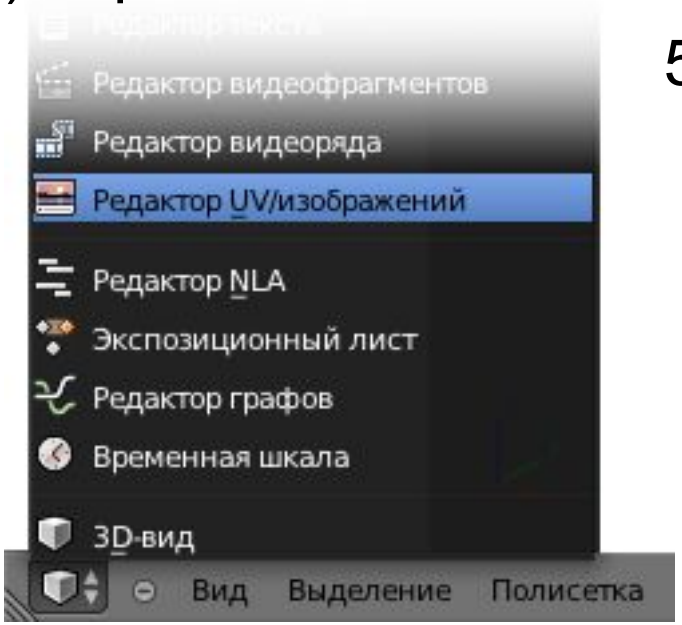
1) в свойствах текстуры включить UV-координаты



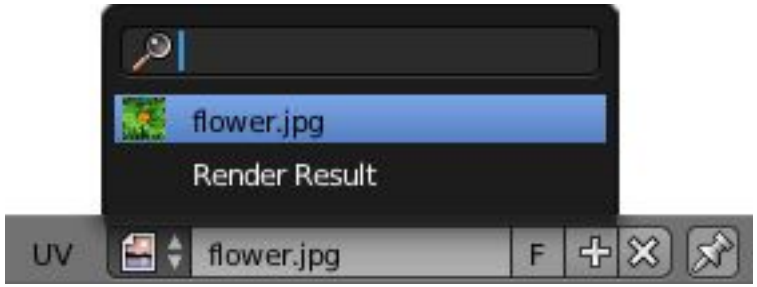
2) **Tab** – перейти в режим правки (*Edit mode*)

3) выделить нужную грань

4) перейти в окно *Редактор UV / изображений*



5) выбрать рисунок-текстуру



Текстуры: UV-развёртка

6) установить вершины рамки в нужное положение

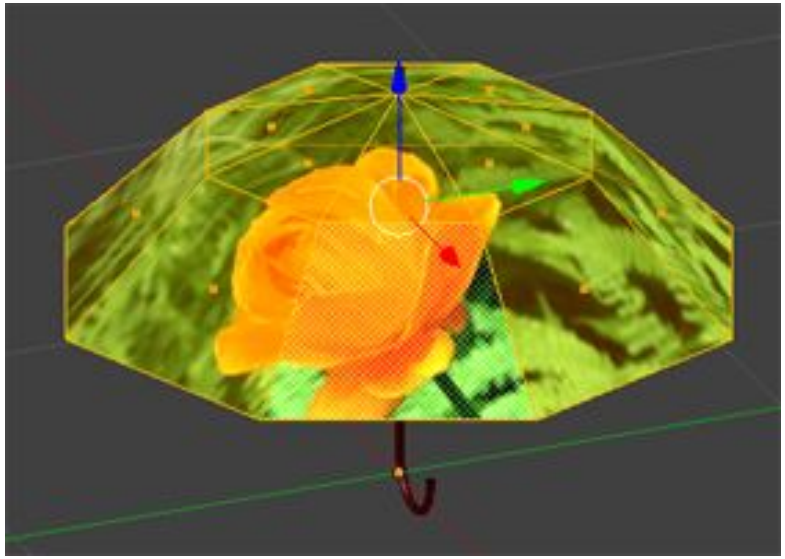
ПКМ – выделение вершин
G – перемещение
R – вращение
S – изменение размера
ЛКМ – зафиксировать

F12 – рендеринг!

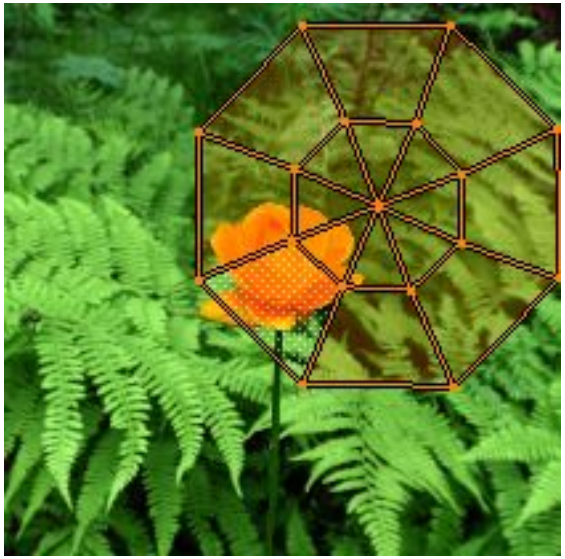
Текстуры: UV-развёртка

Сетка сразу для нескольких вершин:

Выделить все нужные вершины



Редактор UV / изображений



ПКМ – выделение вершин
G – перемещение
R – вращение
S – изменение размера
ЛКМ – зафиксировать

Трёхмерная графика

§ 72. Рендеринг

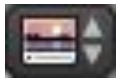
Что такое рендеринг?

Рендеринг – это построение двухмерного изображения: проекции трёхмерной сцены на плоскость с учетом материалов, текстур, освещенности, свойств внешней среды и т.п.

Что нужно сделать:

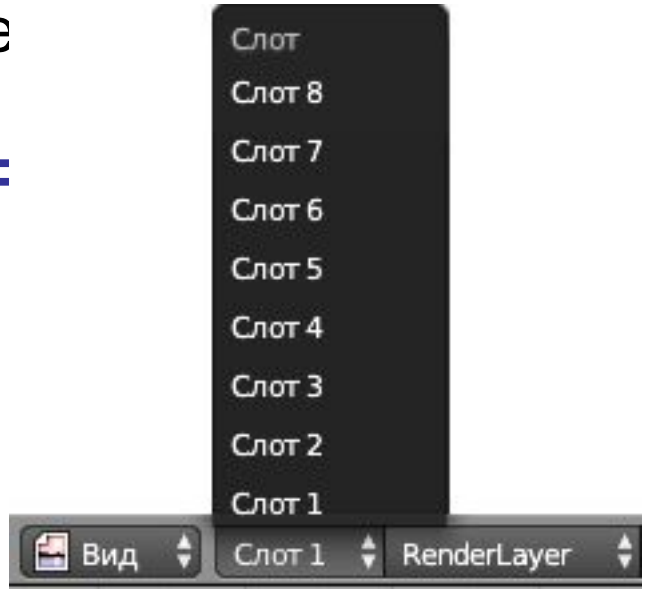
- установить источники света
- установить камеру
- определить свойства внешней сре
- выполнить рендеринг (**F12**)
- сохранить готовое изображение (**F**

Результат рендеринга:



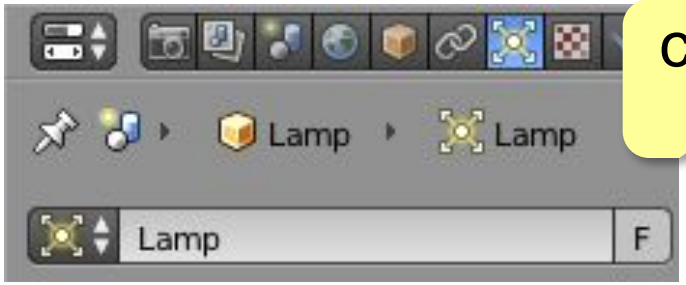
Редактор UV/изображений *(UV/Image Editor)*

слоты для изображений:



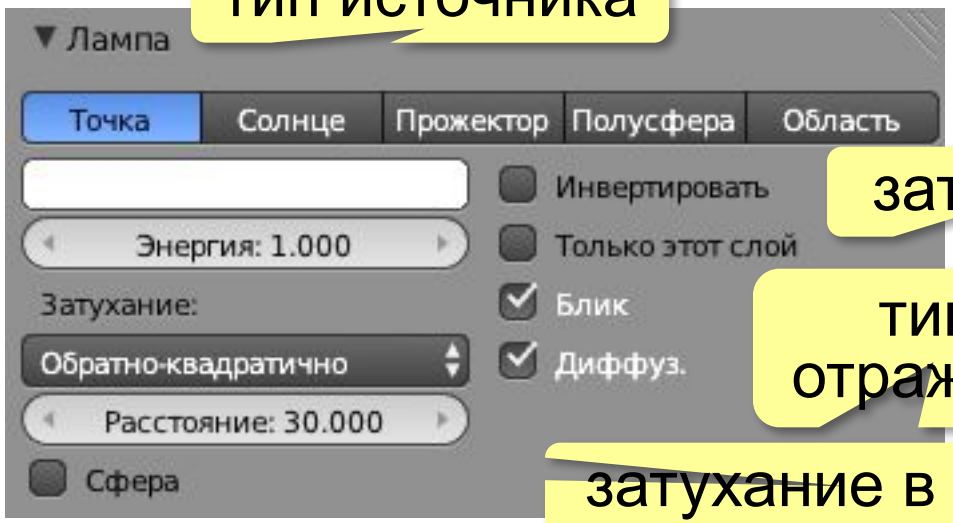
Источники света: Точка *(Point)*

лучи из центра, освещенность пропорциональна $1/R^2$



свойства лампы

тип источника



цвет

энергия

закон затухания

затенение

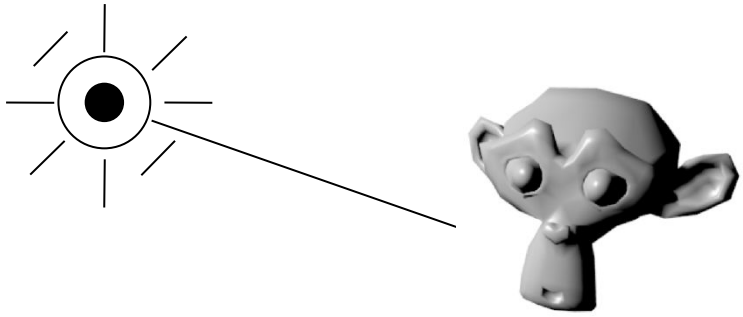
типы отражения

затухание в 2 раза

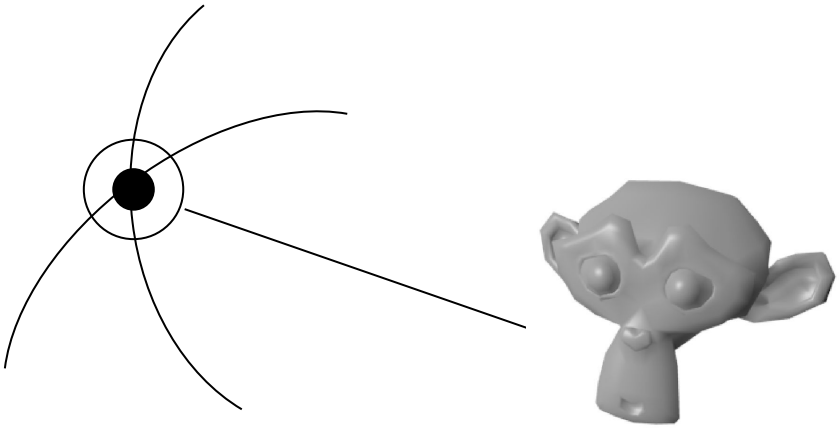
только в пределах сферы

Источники света

Солнце (*Sun*): лучи параллельны, освещенность зависит только от направления лучей



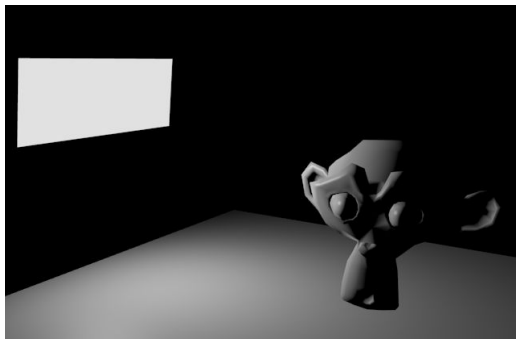
Полусфера (*Hemi*): лучи от накрывающей полусферы, освещенность зависит только от направления лучей, мягкое освещение, нет падающих теней



! Подсветка теневых частей!

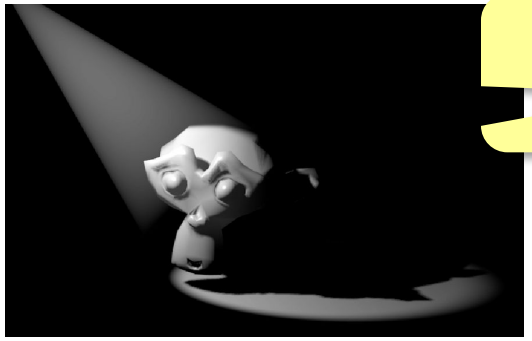
Источники света

Область (*Area*): направленный свет от площадки; освещенность зависит от направления лучей и расстояния



Подсветка от экрана телевизора!

Прожектор (*Spot*): свет в пределах конуса; освещенность зависит от направления лучей и расстояния; площадка круглая или прямоугольная

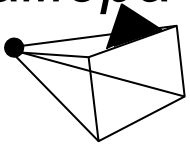


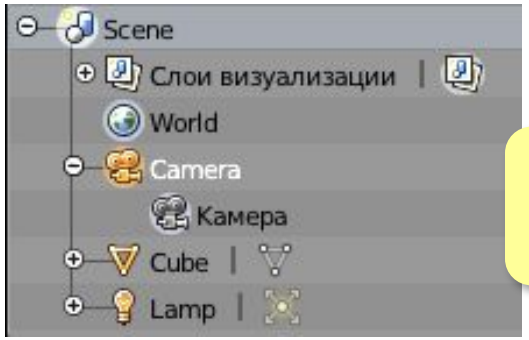
эффект «гало» *(halo)*

Камеры

Камеры — это специальные объекты, которые позволяют посмотреть на сцену с разных точек, как через видоискатель фотоаппарата или видеокамеры.

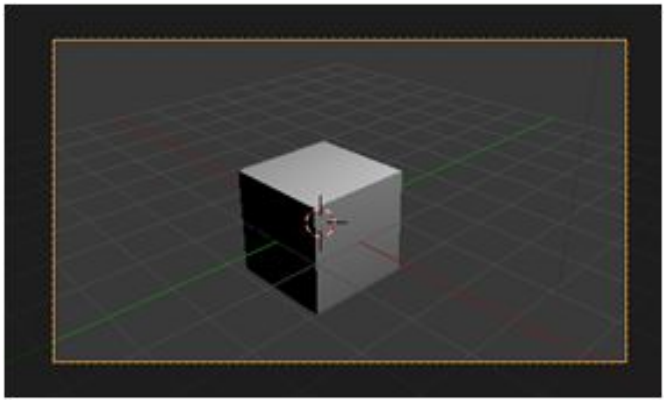
камера





по умолчанию одна камера

G – перемещение объекта
R – вращение объекта
Num0 – вид с камеры
Ctrl+Alt+Num0 – камеру на этот вид



Shift+F – режим полёта
колёсико – приближение
движение мыши – перемещение, **ЛКМ** – фиксировать

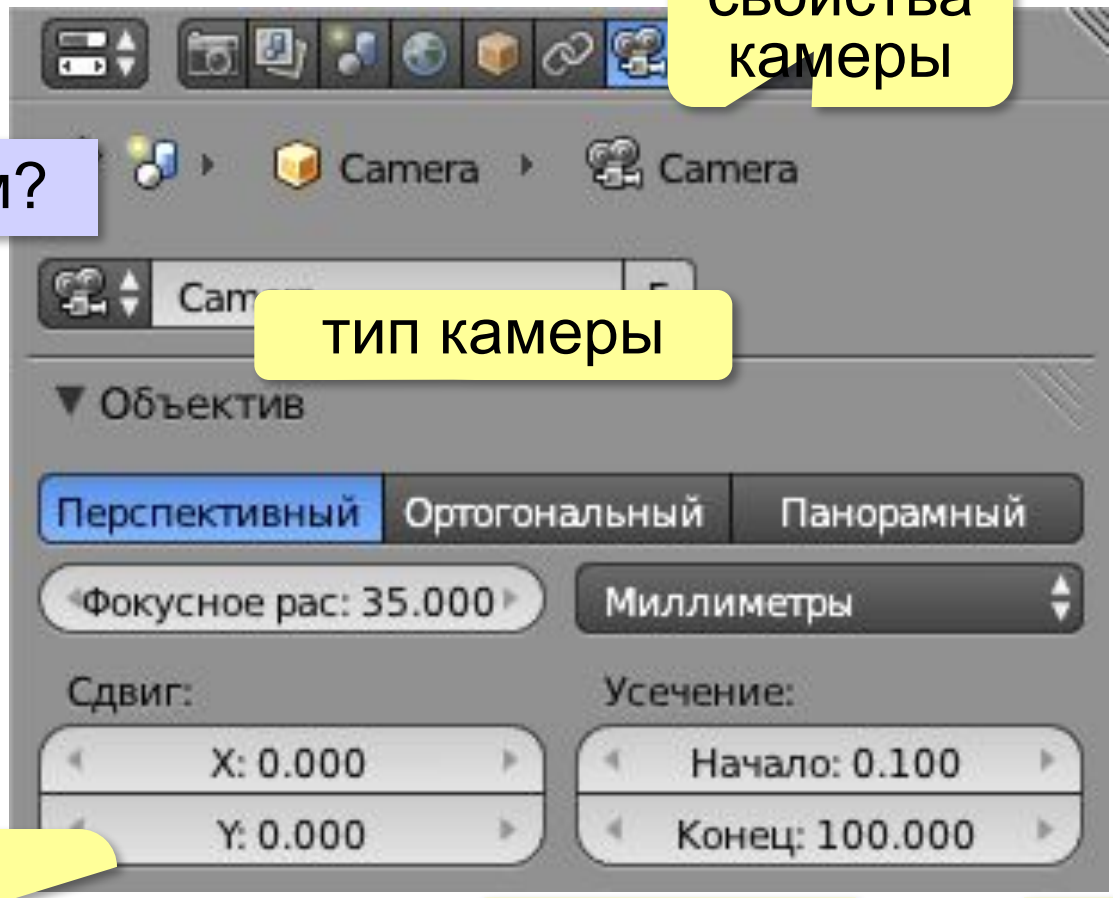
Свойства камеры

свойства камеры

Почему 35 мм?

фокусное расстояние объектива

35 мм – поле зрения человека



сдвинуть поле зрения, не меняя положения камеры

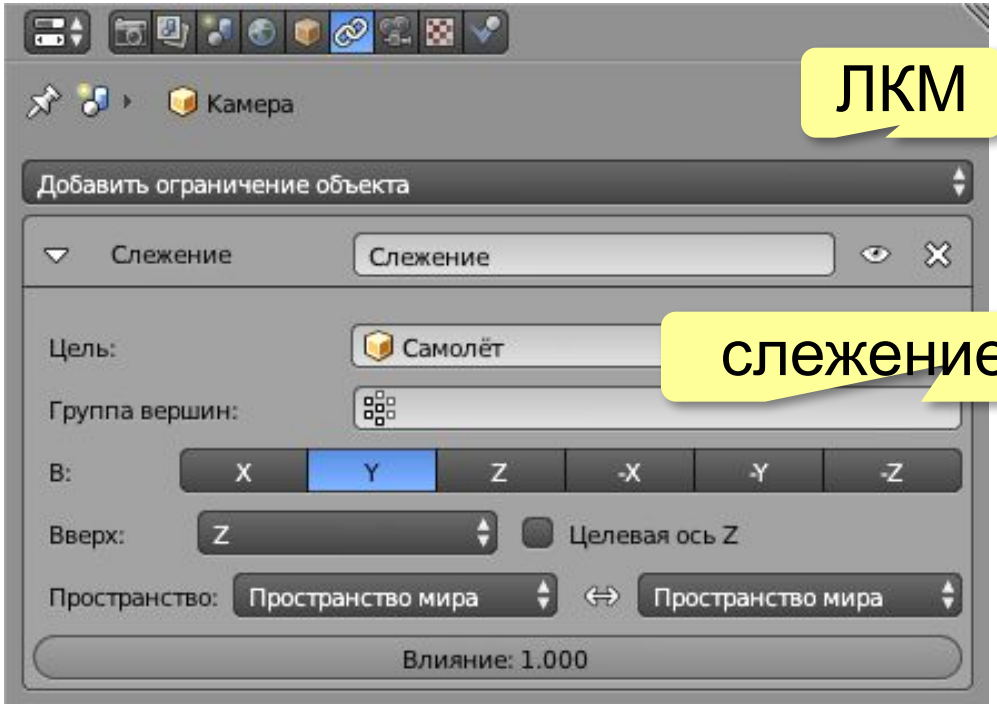
не «видит» за этими границами

Камеры

Добавить – Камера

Ctrl+Num0 – сделать выделенную камеру активной (с неё выполняется рендеринг!)

Связь с объектом-целью *(Track To, Look At):*



ЛКМ

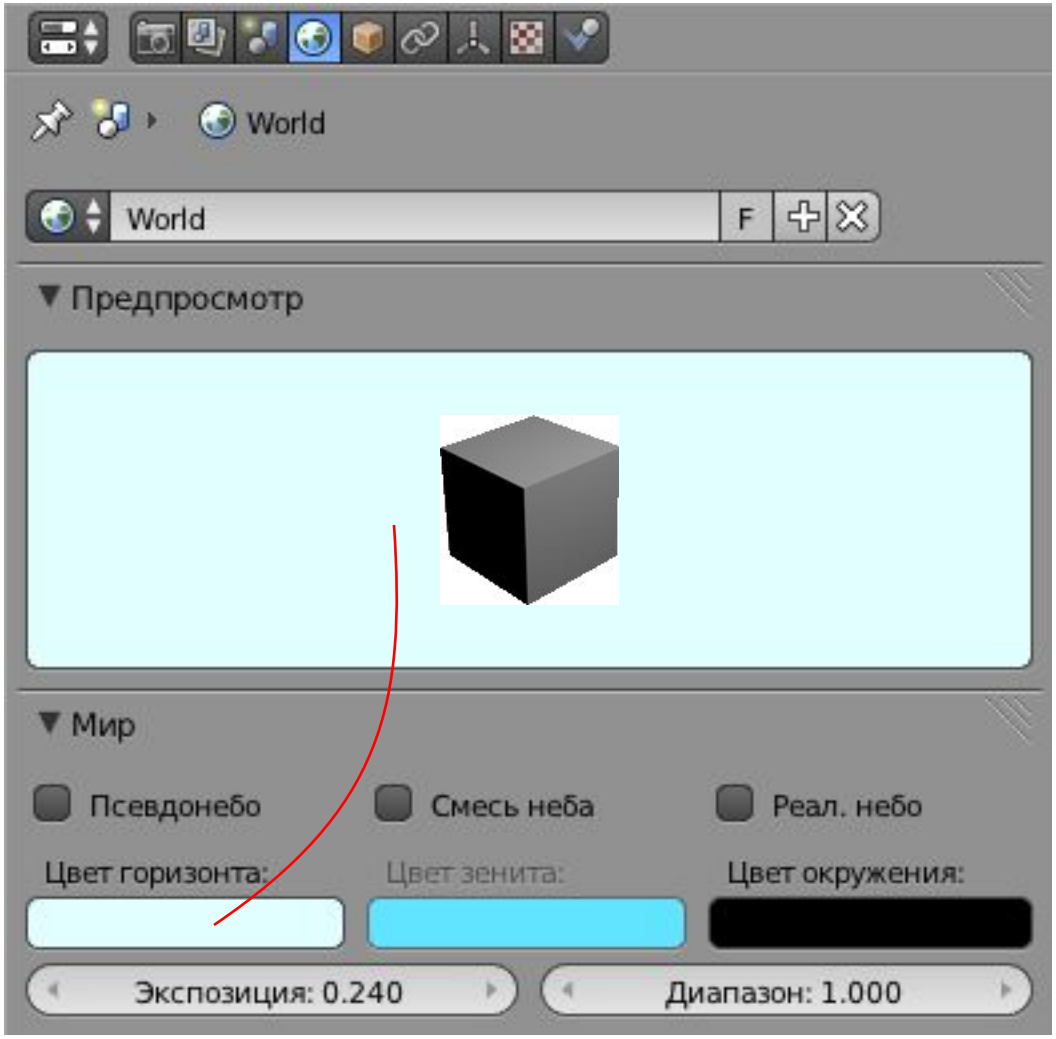
слежение за ним

? Если нет объекта?

Добавить – Пустышка

Внешняя среда *(World)*

Цвет фона

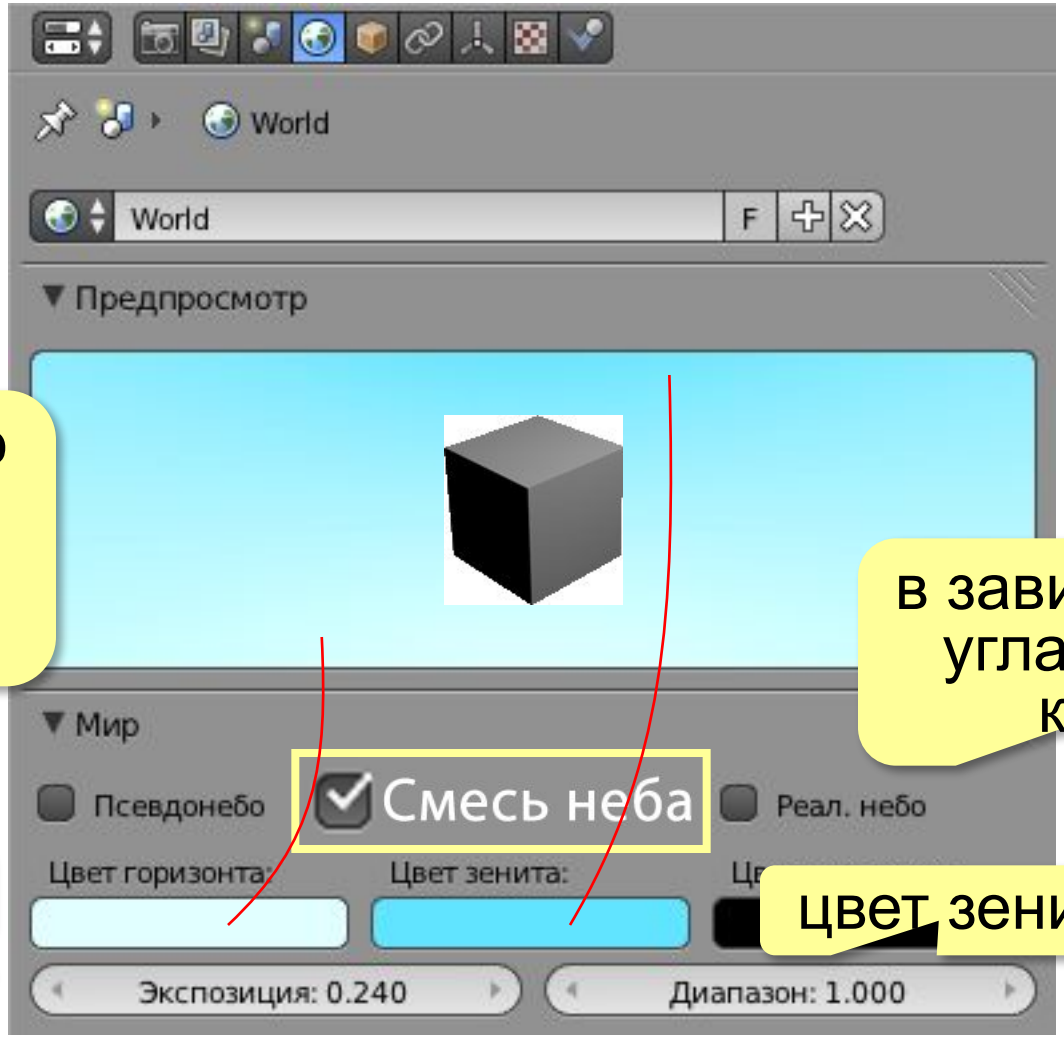


цвет фона

цвет теней

Внешняя среда *(World)*

Градиентный фон



независимо от угла поворота камеры

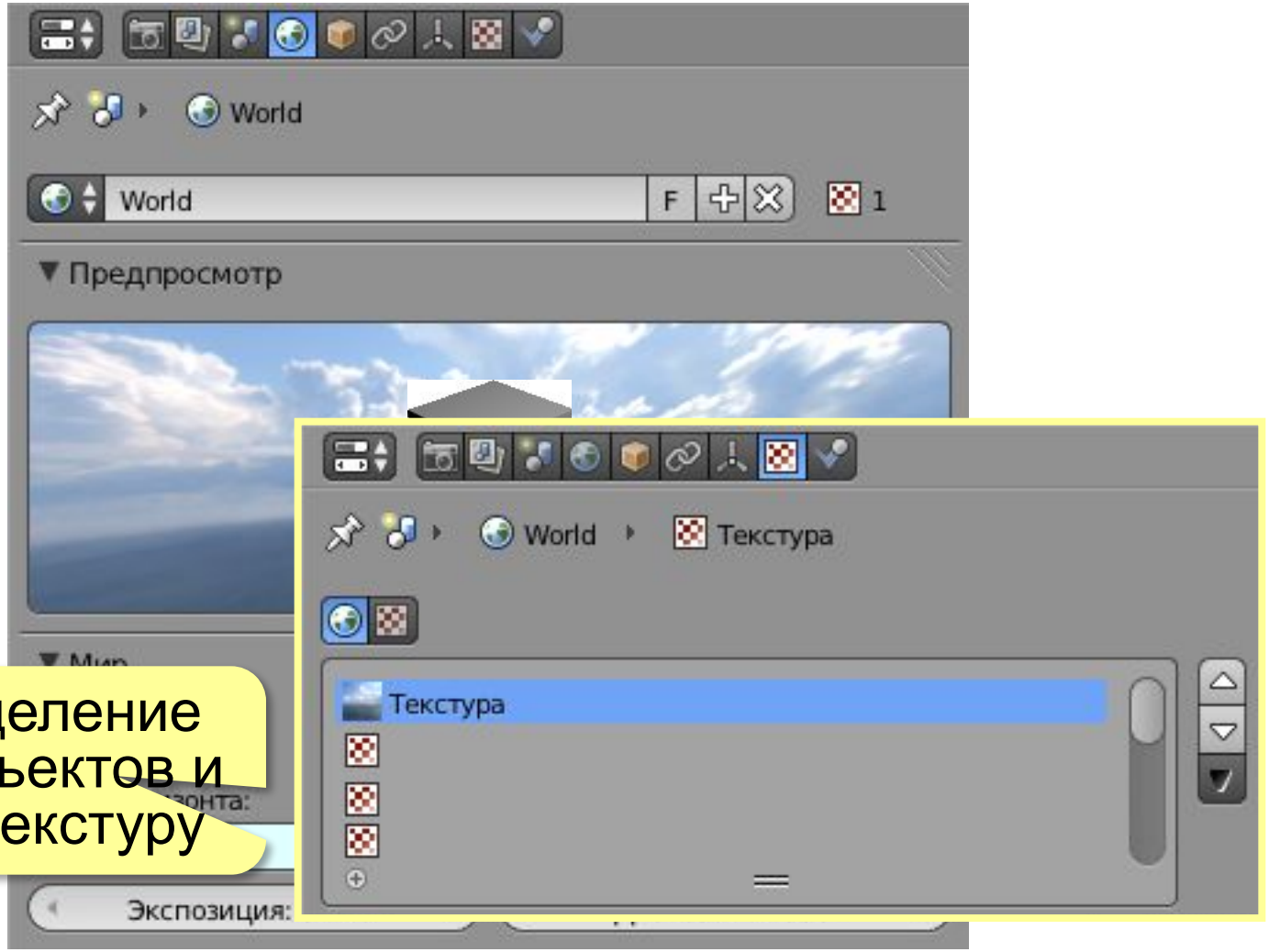
в зависимости от угла поворота камеры

цвет горизонта

цвет зенита

Внешняя среда *(World)*

Фон-текстура



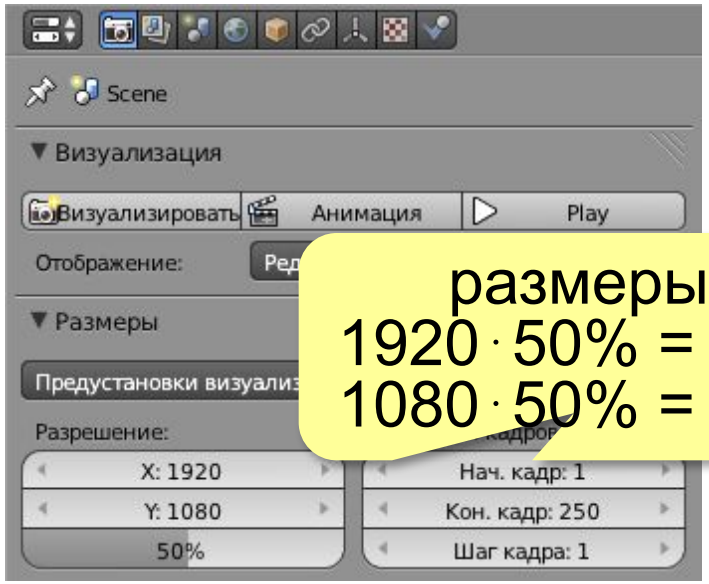
снять выделение со всех объектов и выбрать текстуру

Параметры рендеринга

Разрешение (*Resolution*)

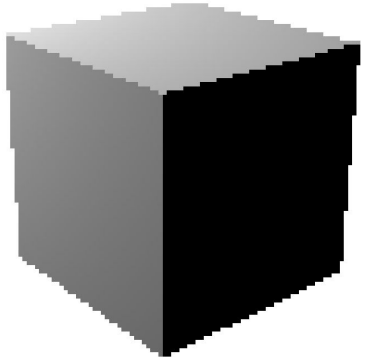
размеры картинки

масштаб

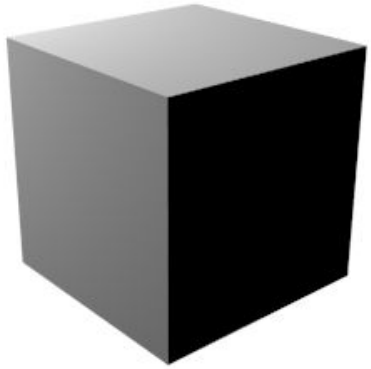


размеры
1920 · 50% = 960
1080 · 50% = 540

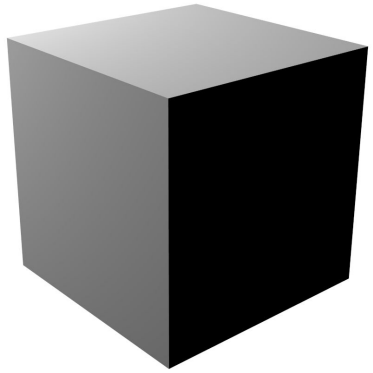
масштаб 10%



масштаб 25%

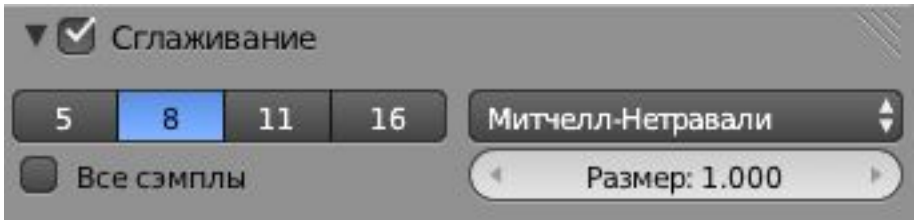


масштаб 100%

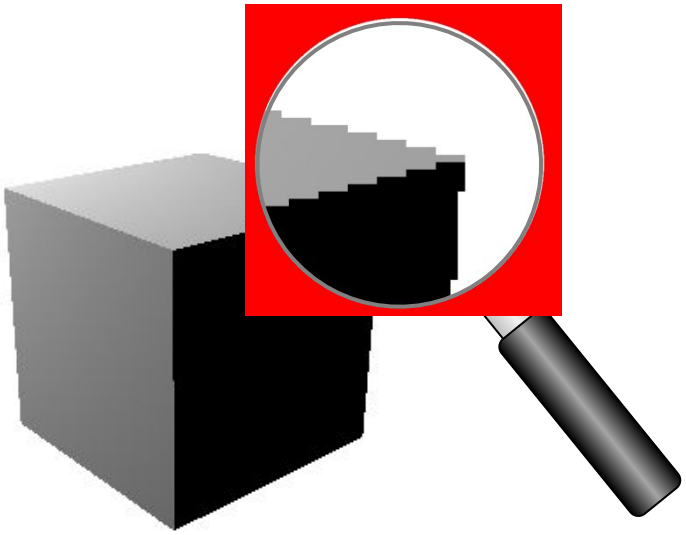


Параметры рендеринга

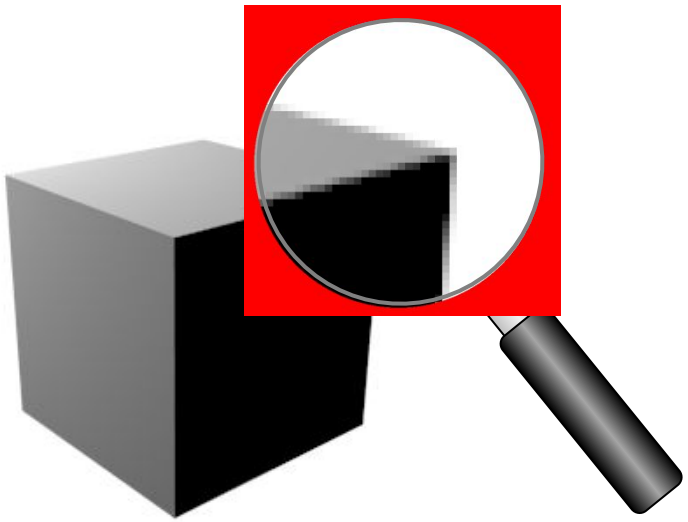
Сглаживание (*Ant-aliasing*)



без сглаживания

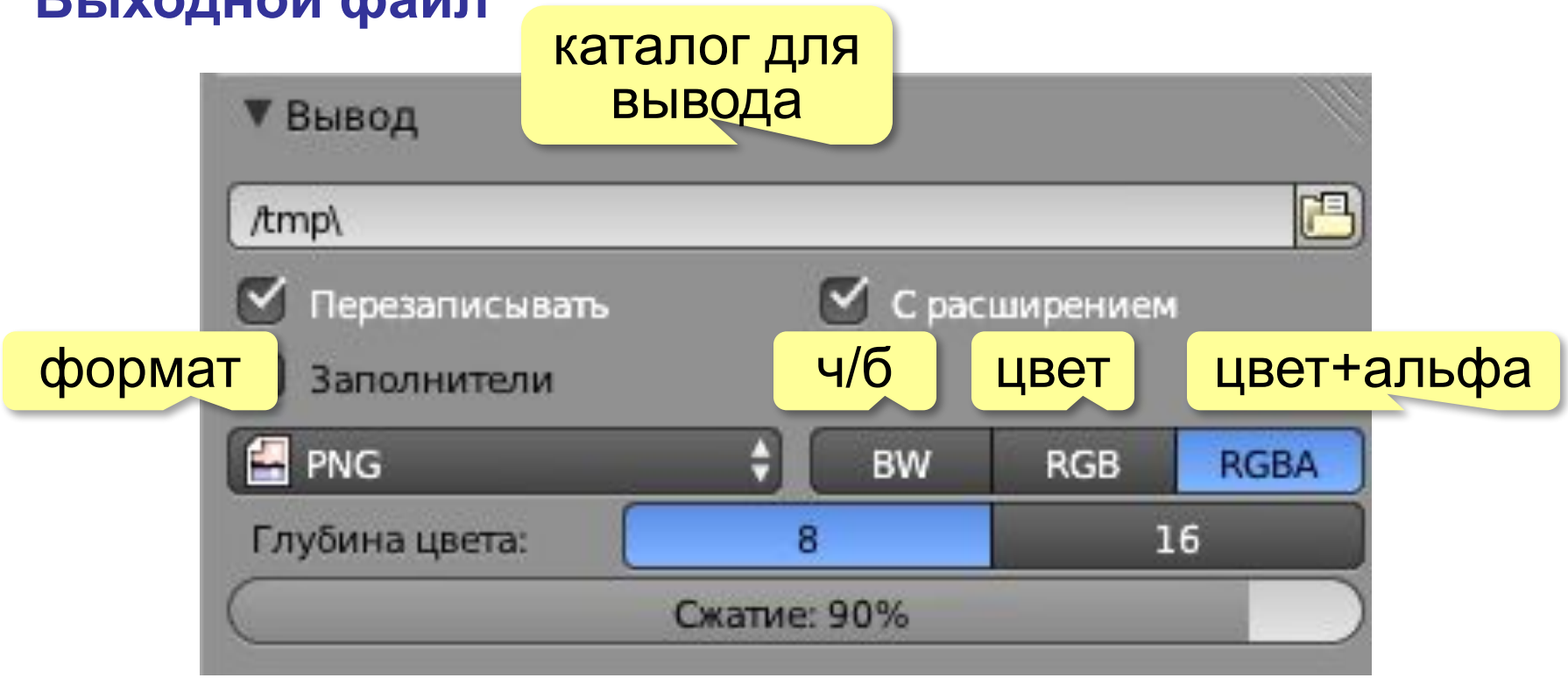


сглаживание



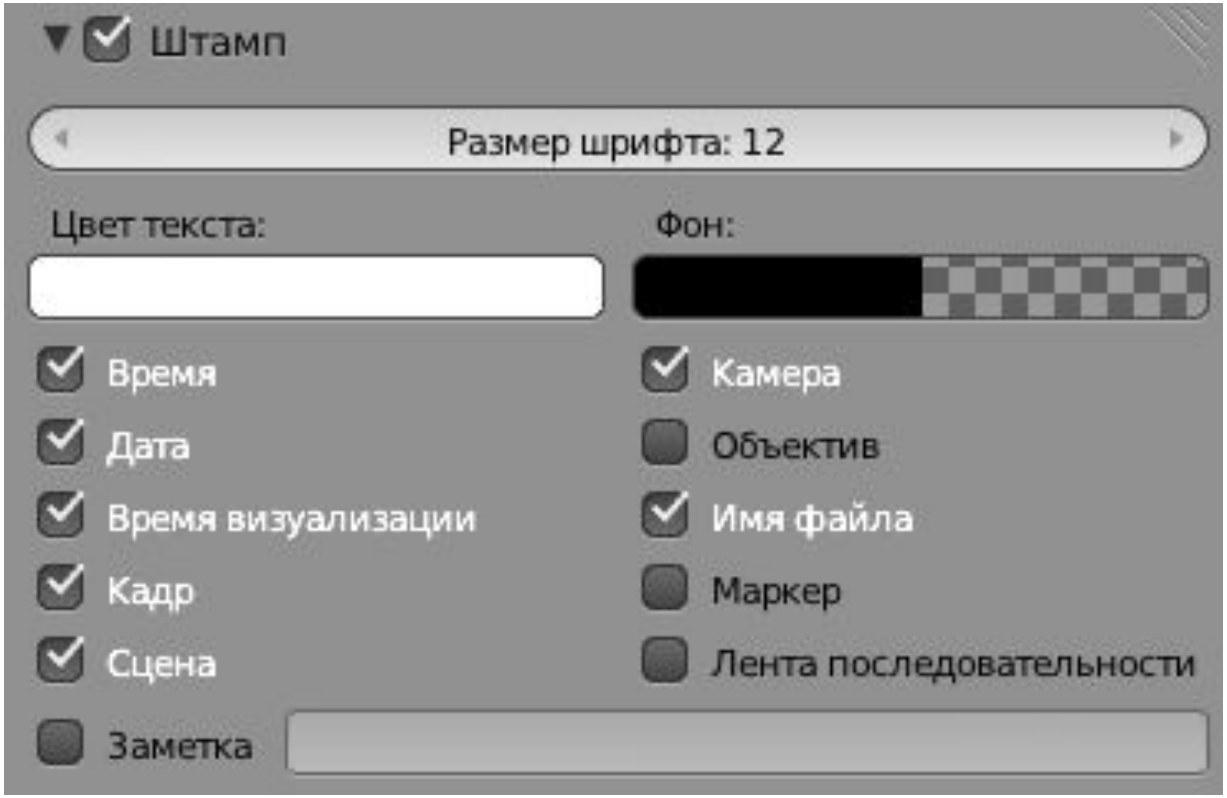
Параметры рендеринга

Выходной файл



Параметры рендеринга

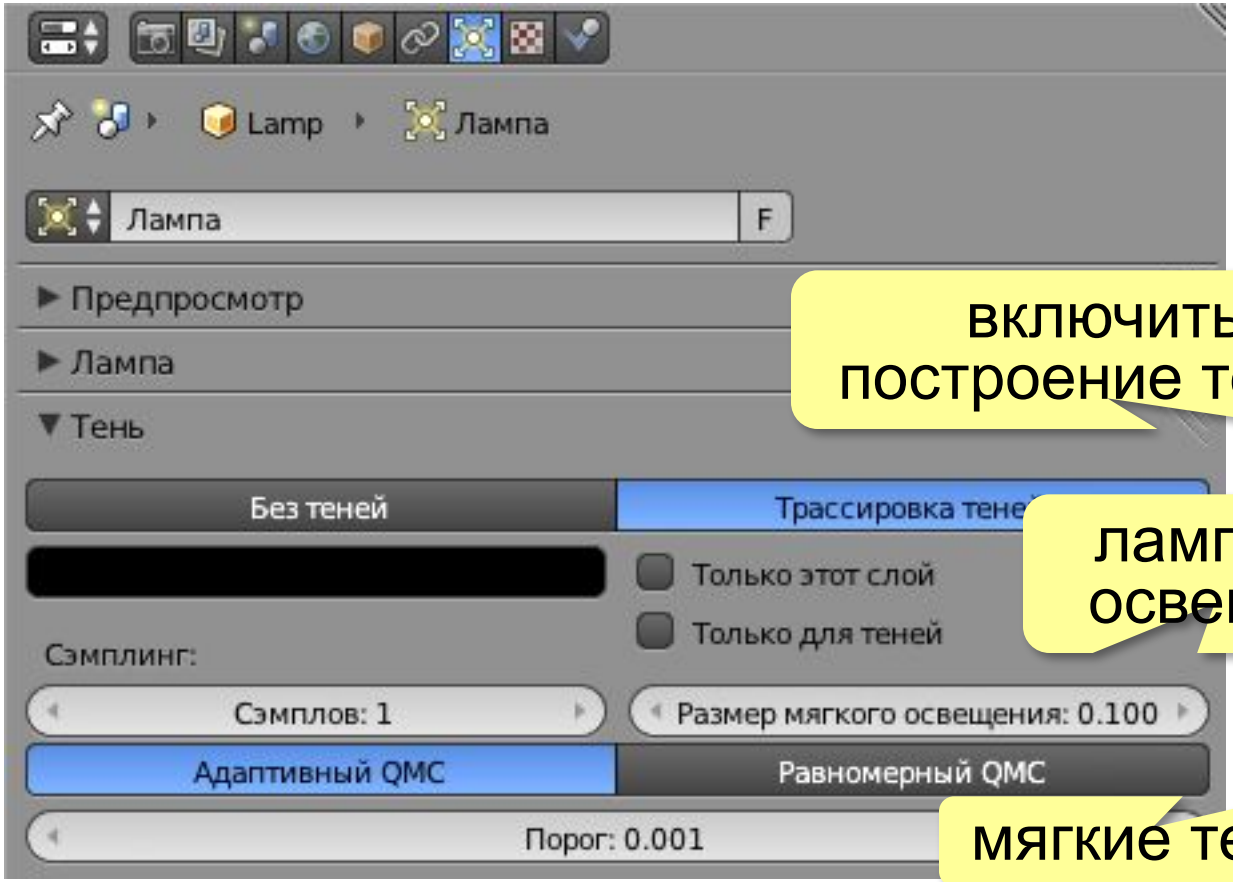
Штамп (*Stamp*) – встраиваемая информация



Тени

Трассировка лучей – учёт эффекта от прохождения большого количества лучей от источника.

Свойства источника



включить построение теней

цвет теней

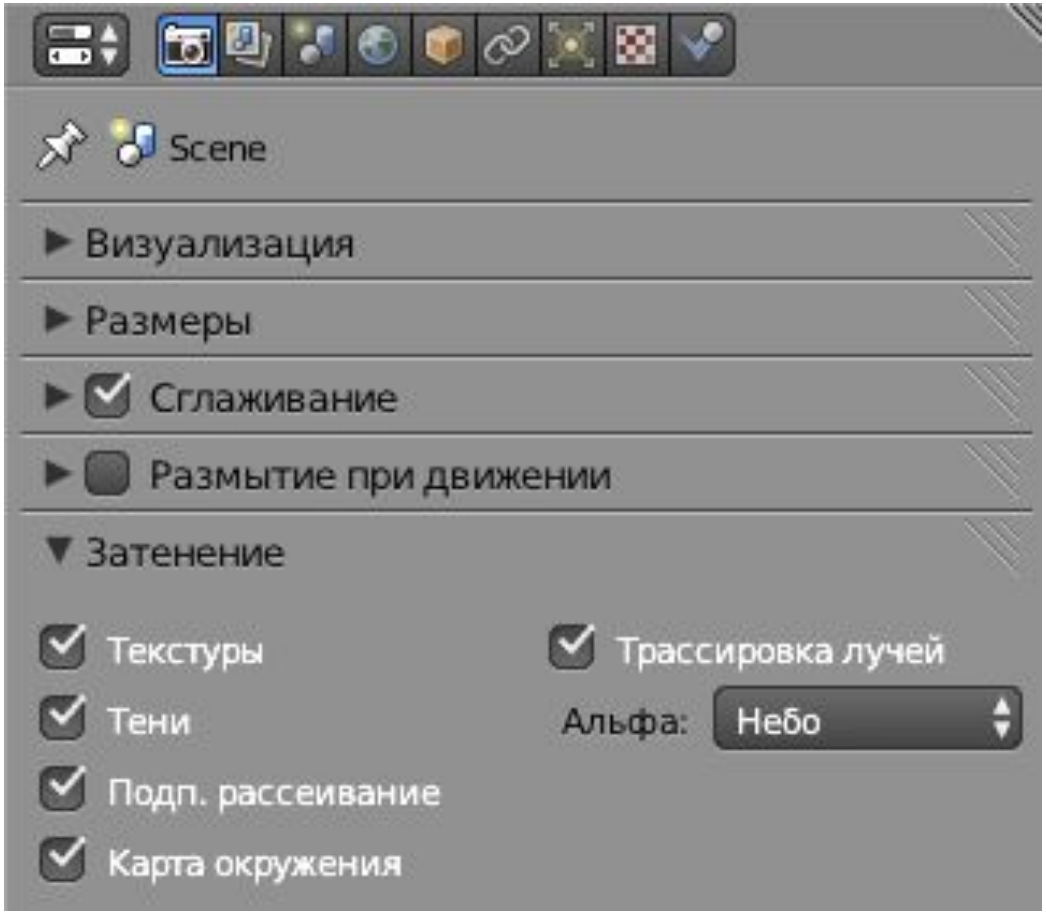
лампа не освещает

качество (↑ время расчётов)

мягкие тени

Тени

Параметры рендеринга



Трёхмерная графика

§ 73. Анимация

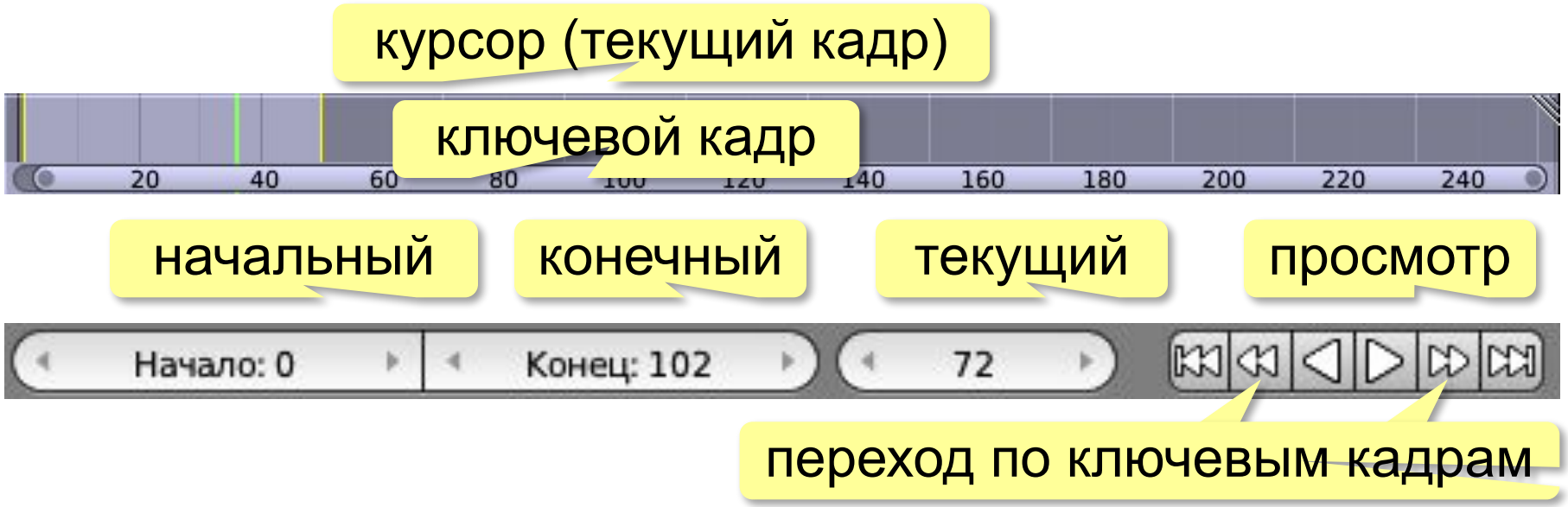
Что такое анимация?

Анимация – это быстрая смена изображений-*кадров* (*frames*), ≥ 16-18 кадров в секунду.

Как строится анимация?

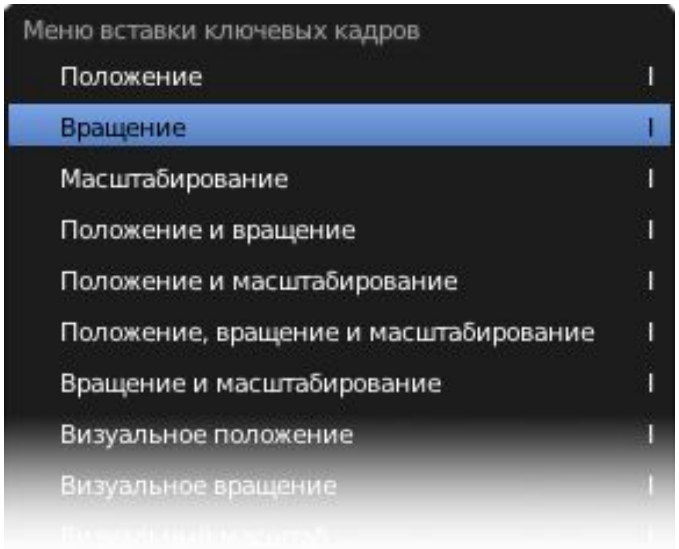
1) задаётся положение объектов в ключевых кадрах
2) автоматически достраиваются переходы

Временна́я шкала (*Timeline*)



Ключевые кадры

1) установить курсор на временной шкале на выбранный кадр
2) задать свойства объекта для этого кадра
3) вставить ключевой кадр, нажав клавишу **I**



! Курсор над полем!

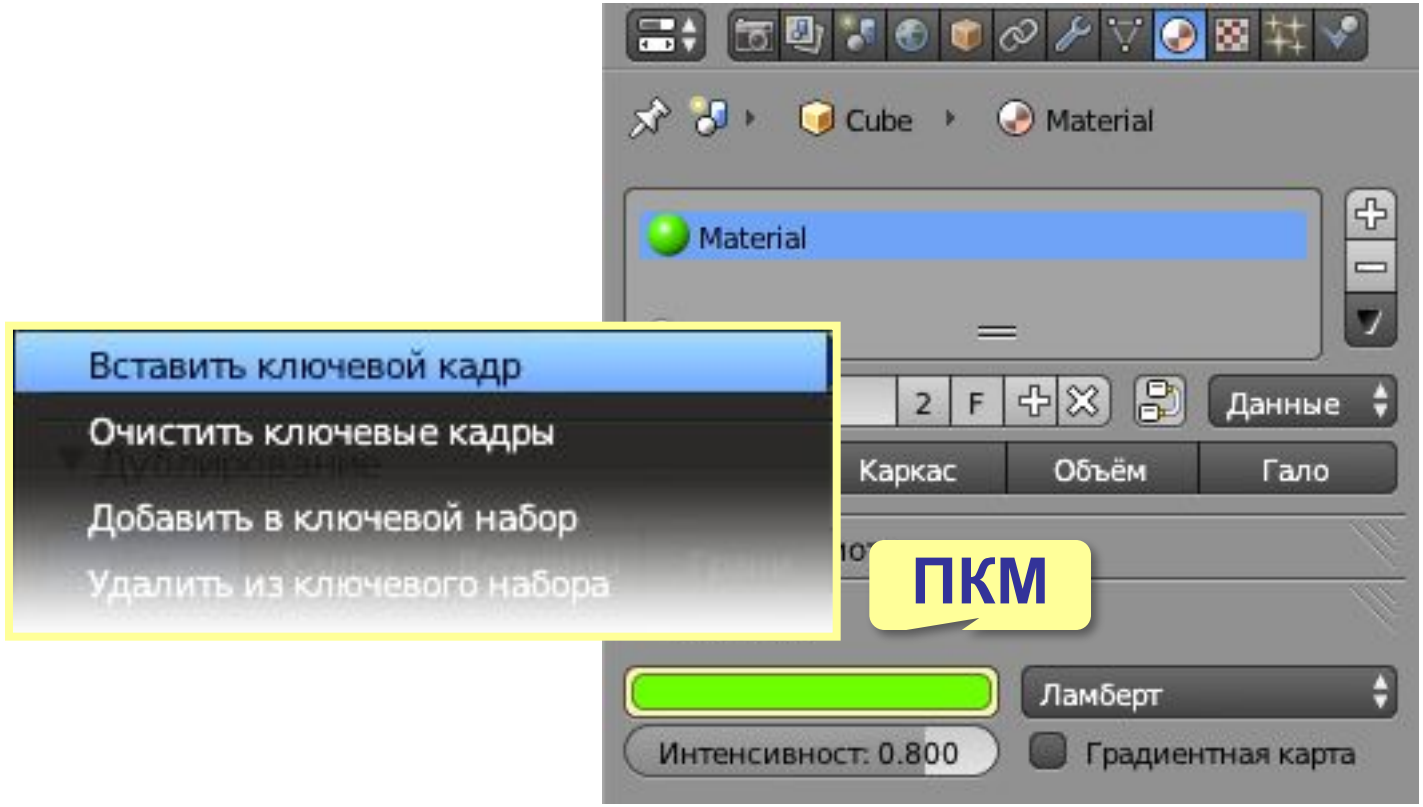
для каждого свойства свои ключевые кадры

4) повторить для всех ключевых кадров

Удаление ключевого кадра: **Alt+I**

Анимация

Для любого свойства



Анимация

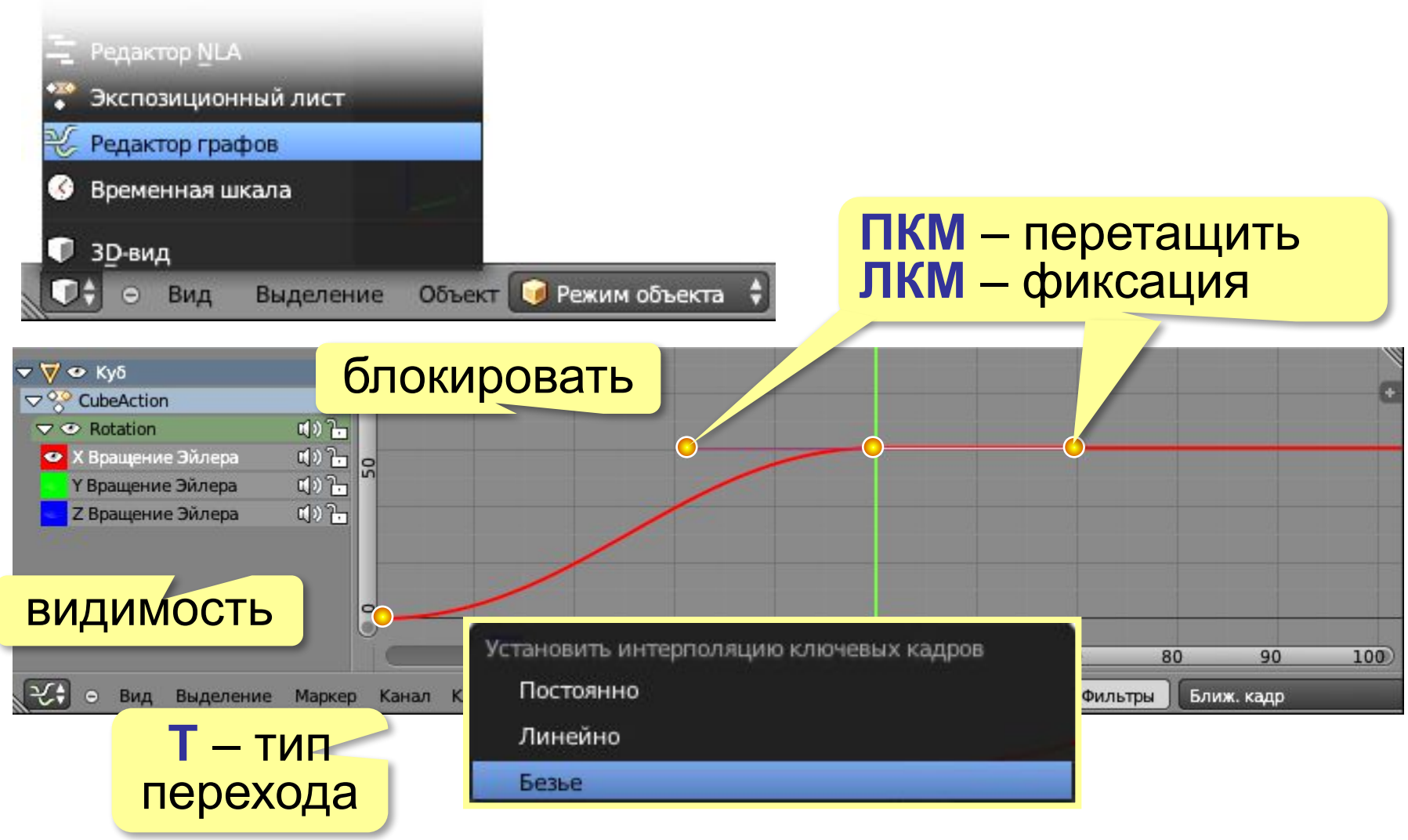
автоматическая вставка ключевых кадров

Запуск анимации «вперёд»: **Alt+A**

Запуск анимации «назад»: **Shift+Alt+A**

Останов и на первый кадр: **Esc**

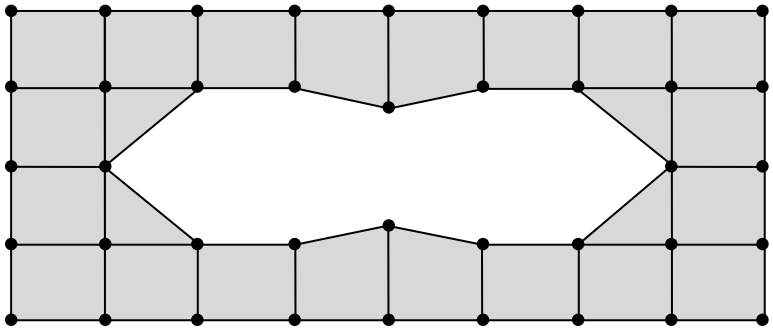
Редактор кривых (*Graph Editor*)



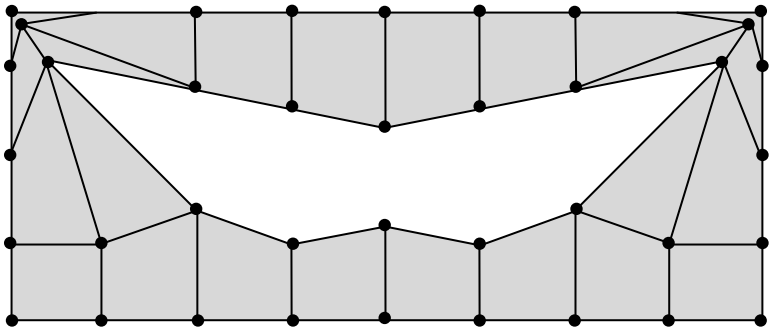
Анимация сеточных моделей

Ключи формы (ключевые формы, *shape keys*) –заранее заданные положения сеточной модели, между которыми выполняется переход.

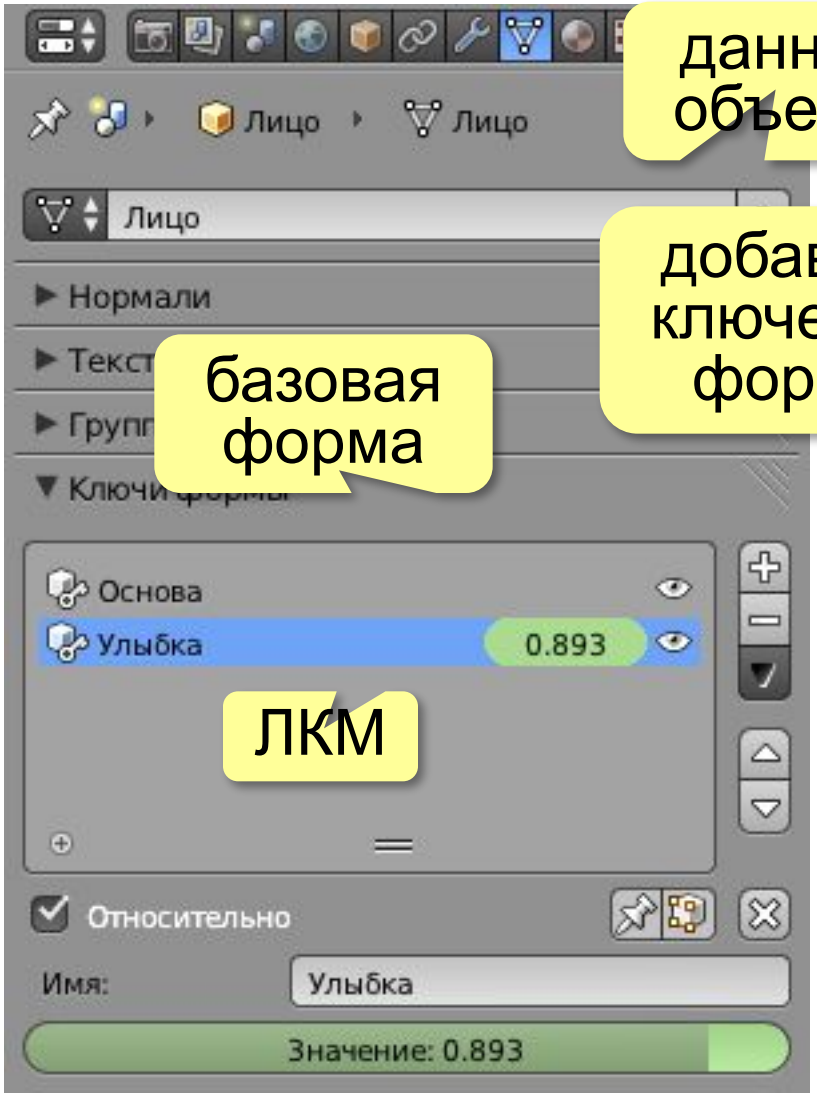
базовая форма (*Основа*)



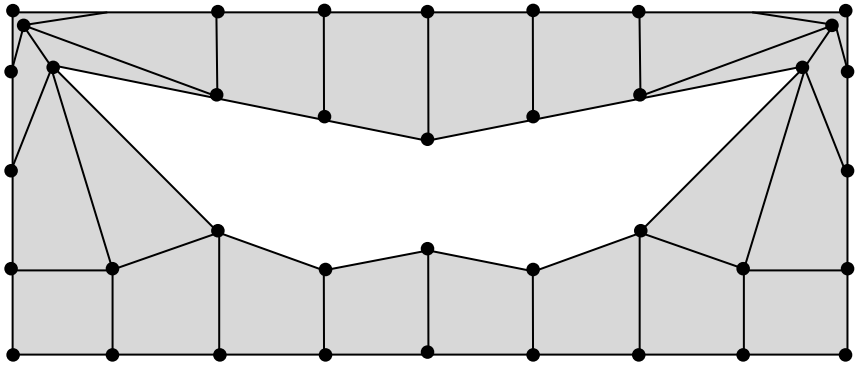
форма *Улыбка*



Ключи формы

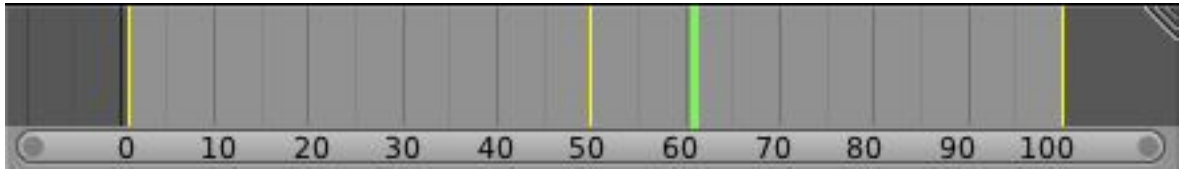


переместить вершины

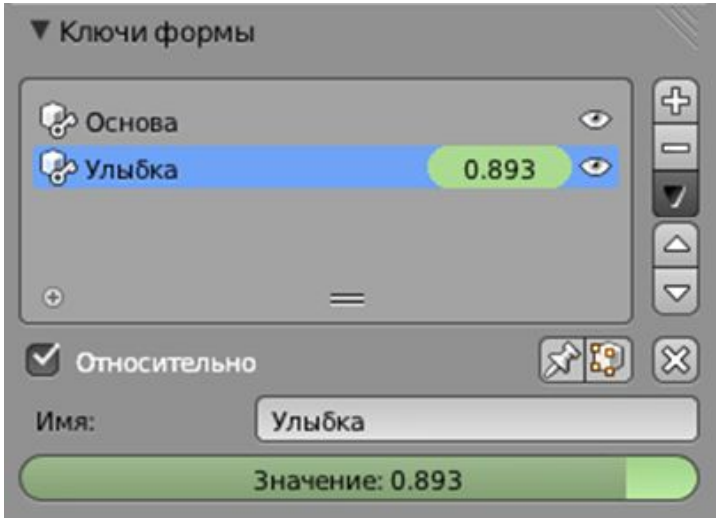


Ключи формы

установить курсор в место ключевого кадра



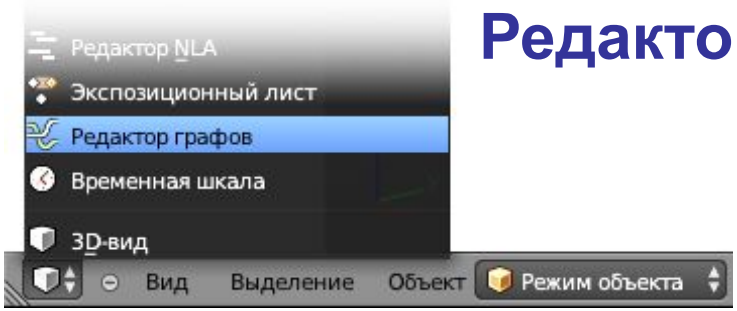
изменить долю ключевой формы



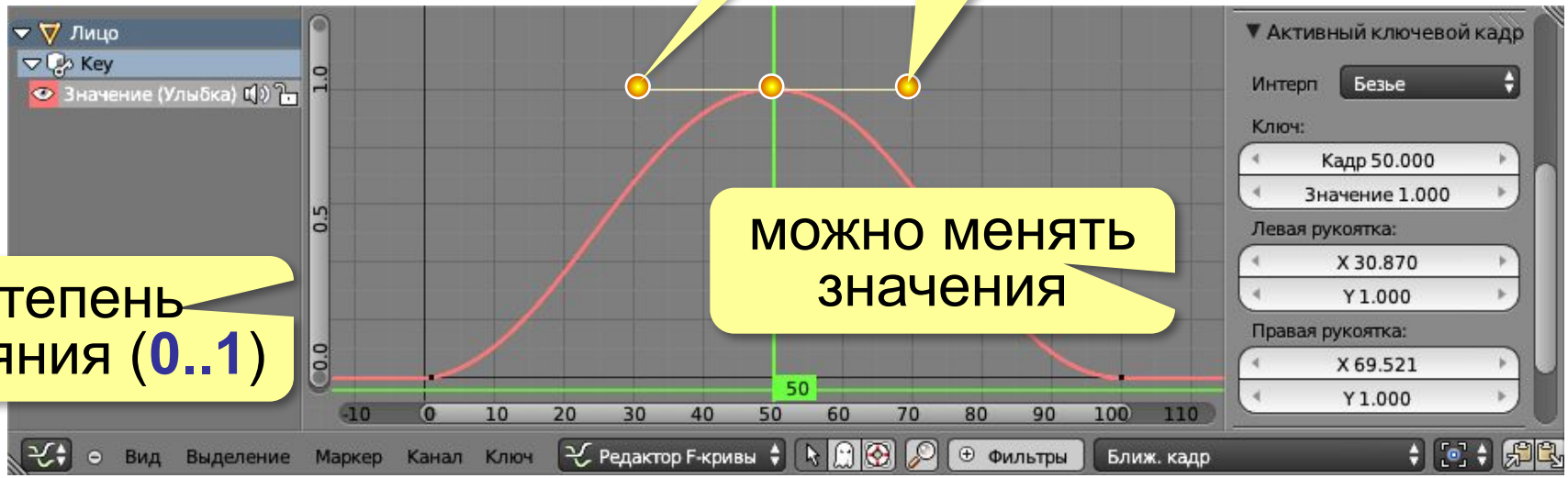
ПКМ – Вставить ключевой кадр

Настройка переходов

Редактор кривых



ПКМ – перетащить
ЛКМ – фиксация



можно менять значения

степень влияния (**0..1**)

N – показать свойства

Арматура (скелет)

Задача: передвинуть сразу много вершин

арматура (скелет)

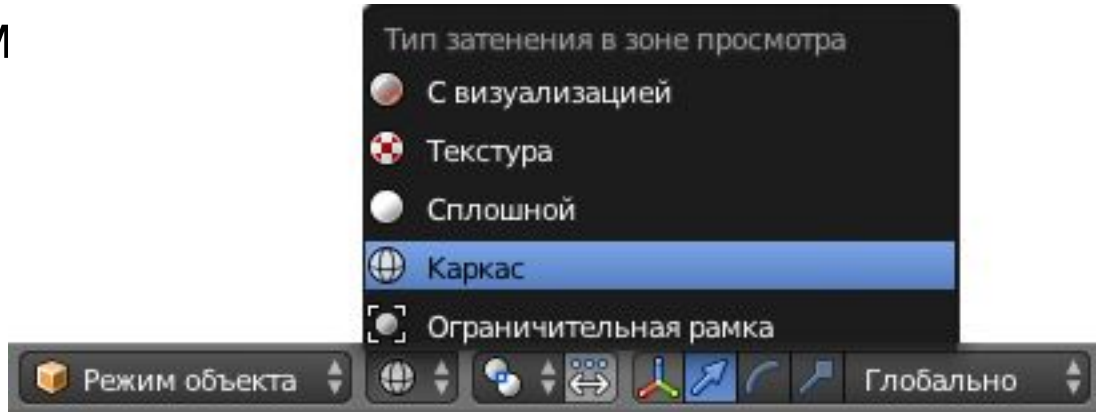
Кости при рендеринге не видны!

Этапы:

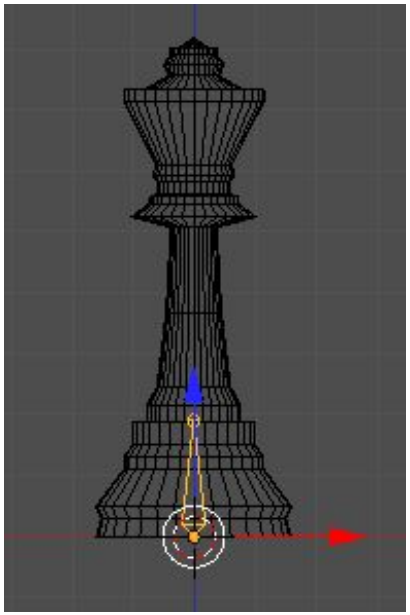
- создание скелета из костей
- привязка вершин сеточной модели к костям
- придание персонажу нужной позы (установка положения костей)

Арматура (скелет)

режим



Добавить – Скелет – Одна кость

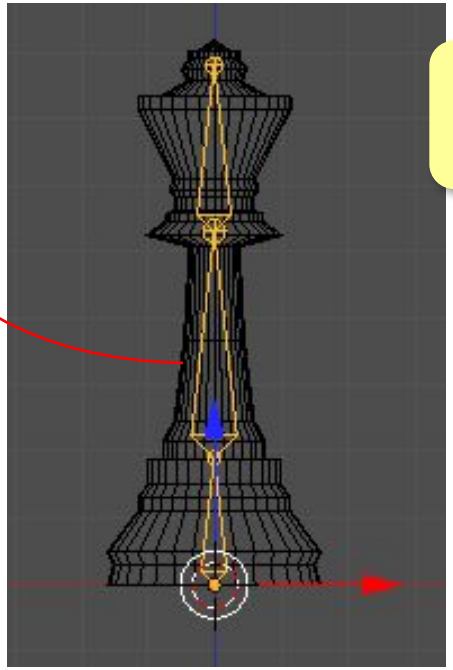


Арматура (скелет)

Tab – режим правки

выделить верхнюю вершину

E, затем **Z** – выдавливание вдоль оси Z



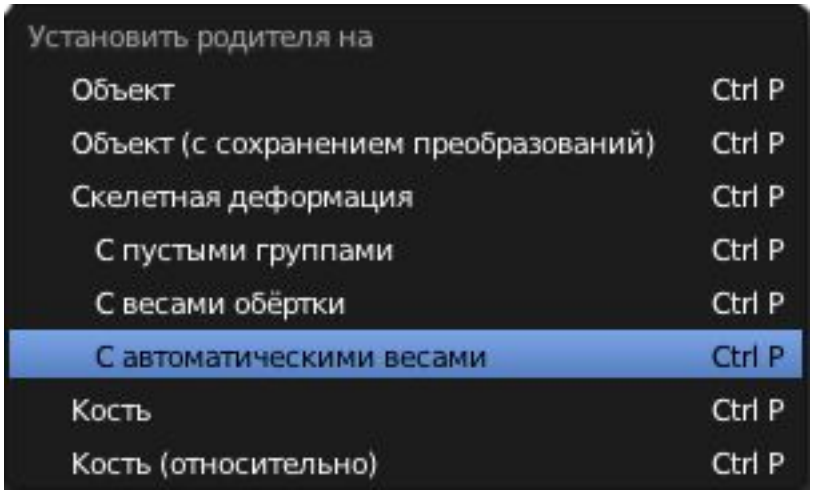
аналогично 3-я кость

Арматура (скелет)

Tab – режим работы с объектами

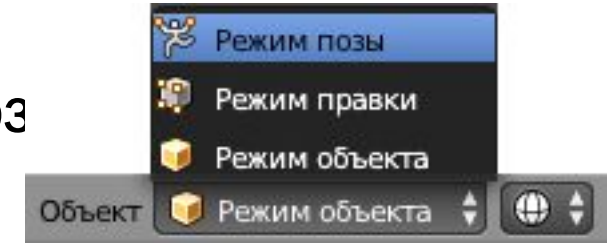
выделить фигуру короля, затем арматуру

Ctrl+P – установить родительский объект

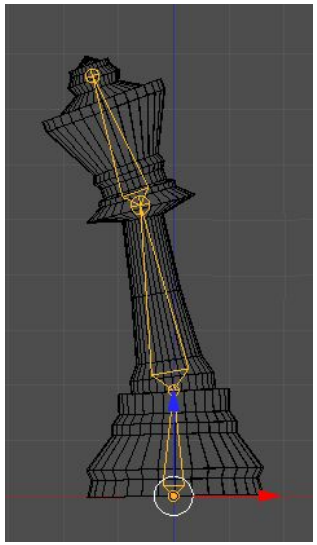
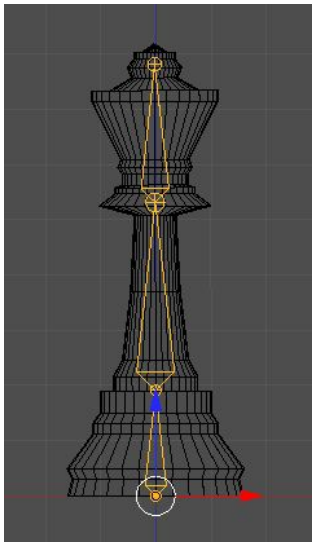
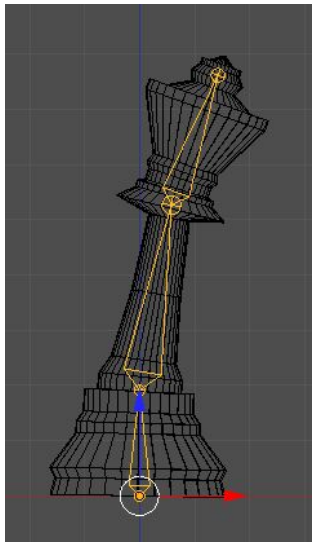


Арматура (скелет)

выделить арматуру
перейти в режим поз



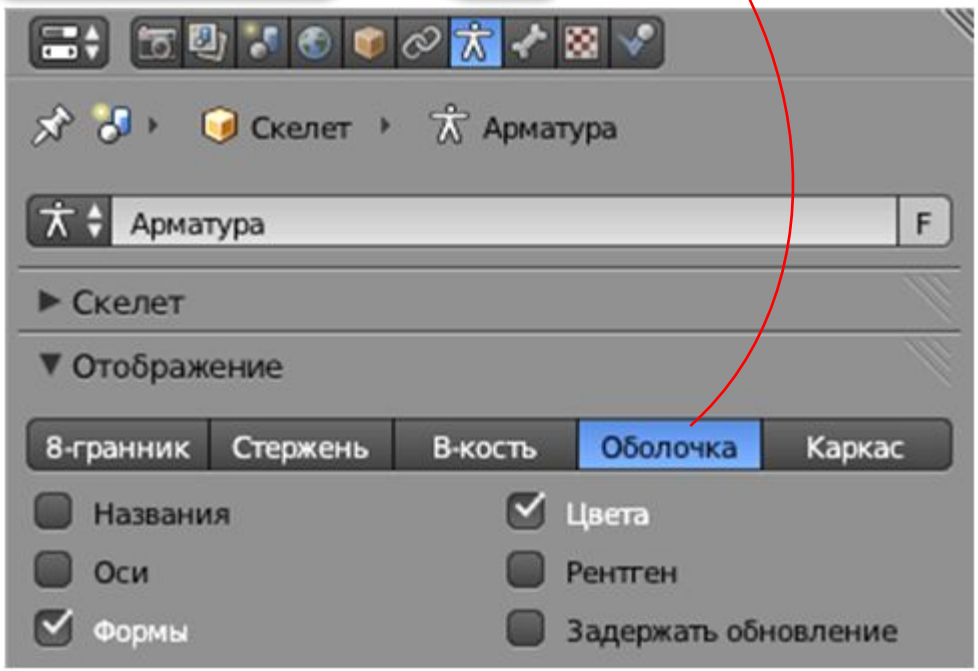
в ключевых кадрах:
изменить положение костей (**R** – вращение)

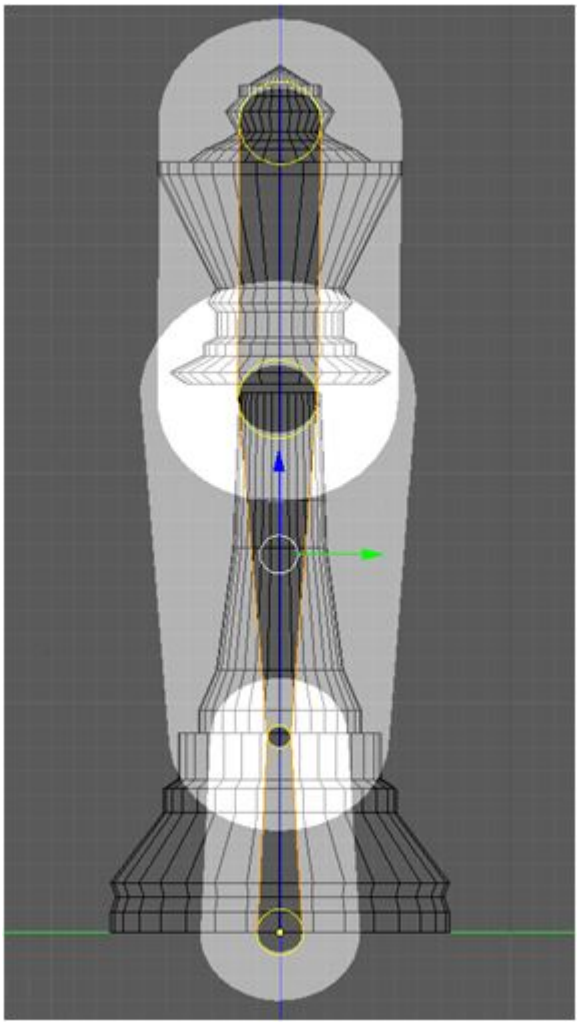
Арматура (скелет)

Оболочка *(Envelope)*

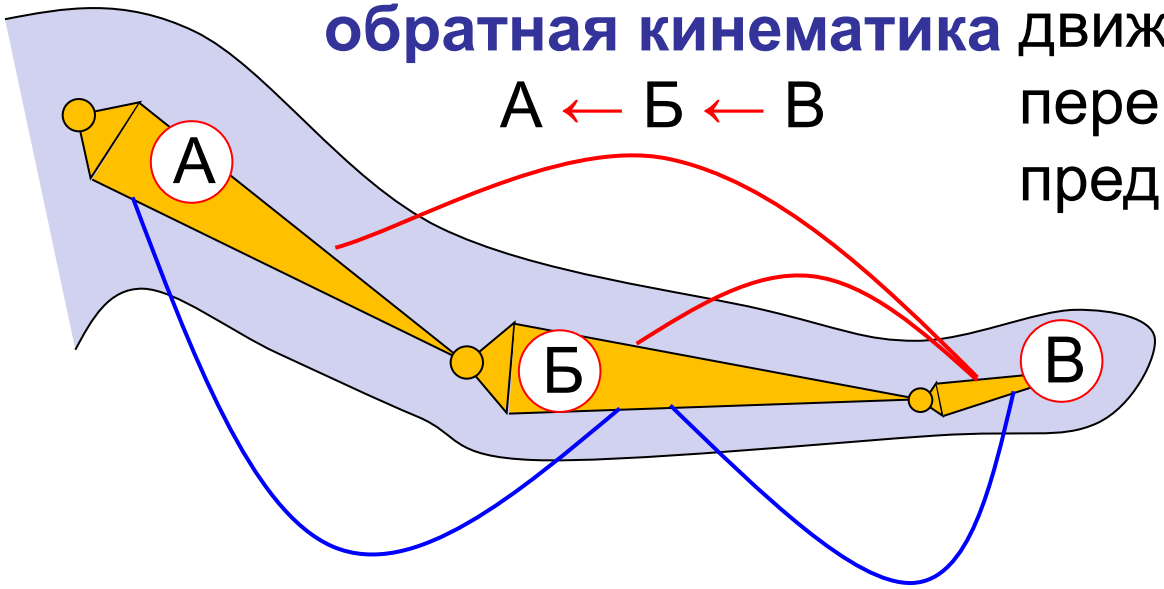
Данные объекта



выделить узел, **S** (изменить радиус влияния)



Прямая и обратная кинематика



обратная кинематика движение кисти В перемещает предплечье Б и плечо А

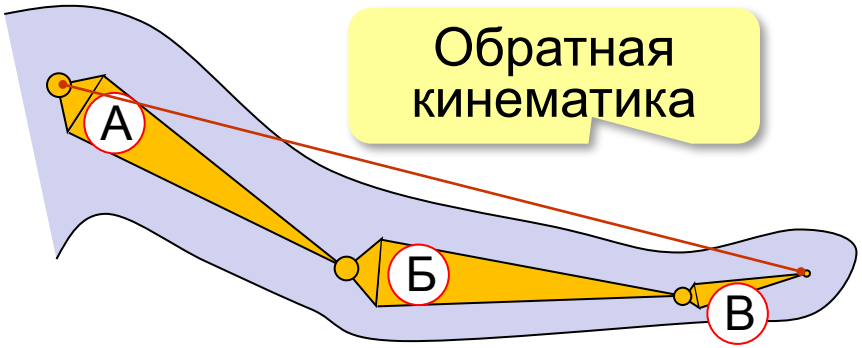
А ← Б ← В

прямая кинематика

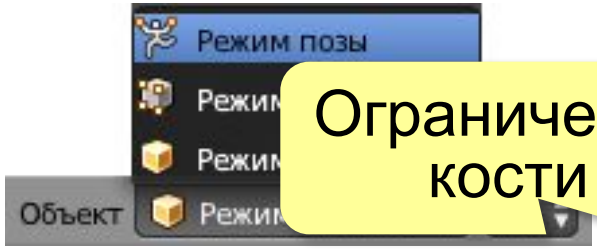
А → Б → В

движение плеча А перемещает предплечье Б и кисть В

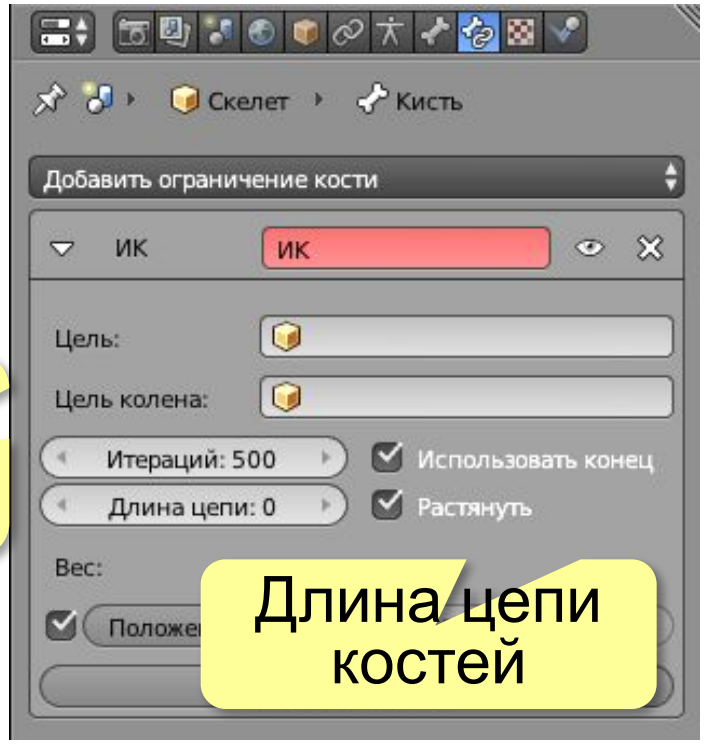
Обратная кинематика



1) выделить арматуру
2) перейти в режим позы



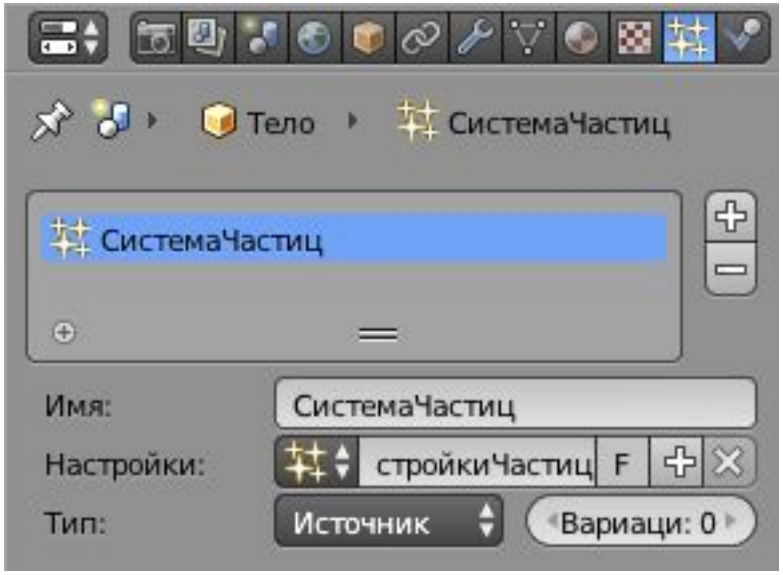
3) выделить последнюю кость В
4) установить ограничение кости «Инверсная кинематика»
5) выбрать длину цепочки (0 – до конца скелета)



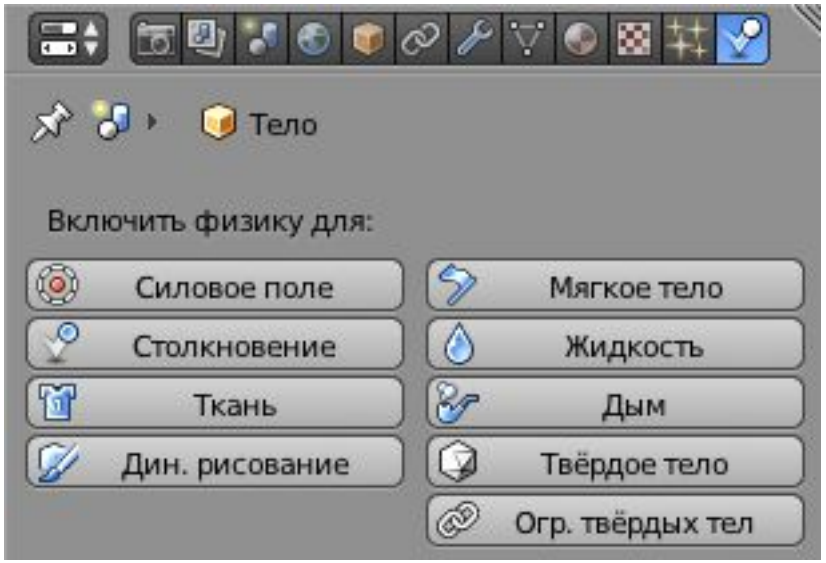
Физические явления

- системы частиц (дым, огонь, волосы, траву)
- жидкости
- столкновения тел
- силовые поля (ветер и магнитное поле)
- ткань

Частицы (*Particles*)



Физика (*Physics*)



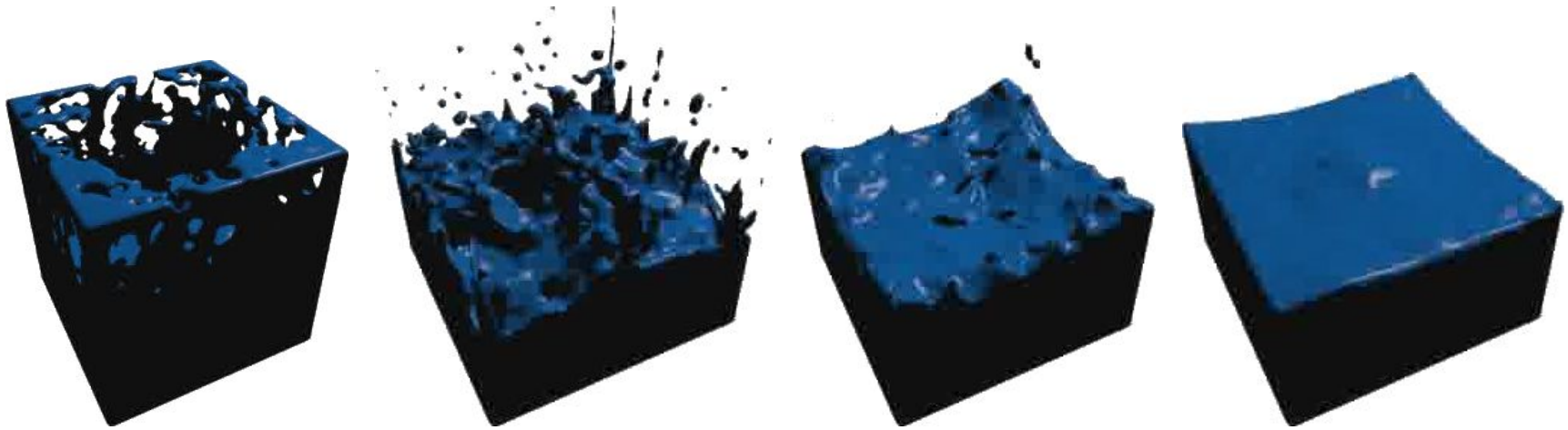
Система частиц

- пар, огонь, дым, дождь, снег, волосы, …
- невозможно построить сеточную модель
- моделируется как поток частиц
- учитываются свойства среды (гравитация, ветер)

Волосы:

Жидкость

- поверхность с большим число граней
- анимация – расчет следующего положения всех вершин с учетом физических свойств (вязкости) – «выпечка»
- сеточные модели для каждого кадра сохраняются на диске во временном каталоге (*кэширование*)
- при изменении параметров всё просчитывается заново

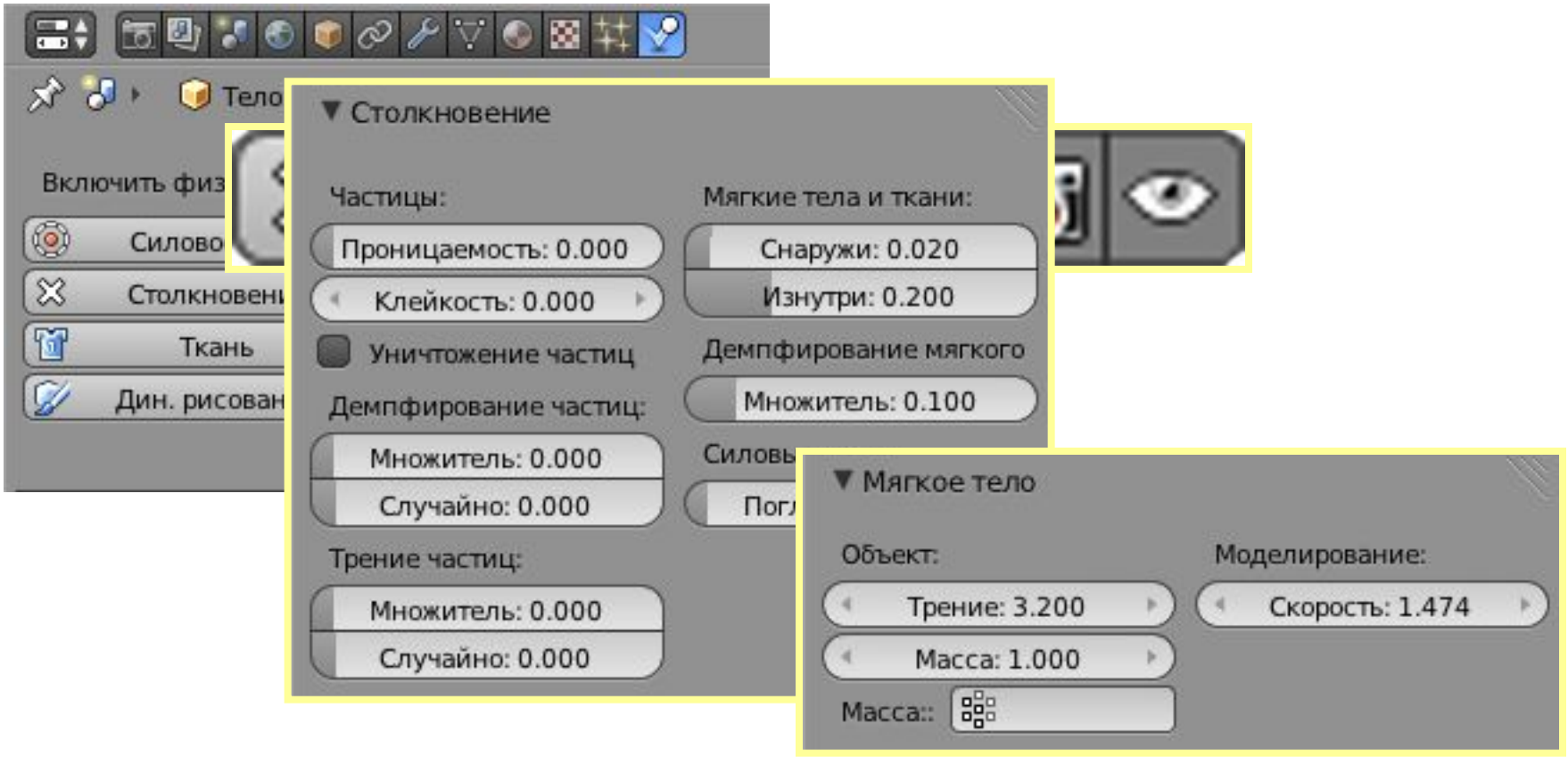


Ткань

- разбивается на грани вручную (плоскость, инструмент *Подразделить*)
- чем мельче грани, тем точнее моделирование, но больше время расчётов
- учитывается тип ткани (хлопок, шёлк, кожа, …)
- учитываются силовые поля
- для объектов, которые задерживают ткань, устанавливается свойство *Столкновение* (страница свойств *Физика*)

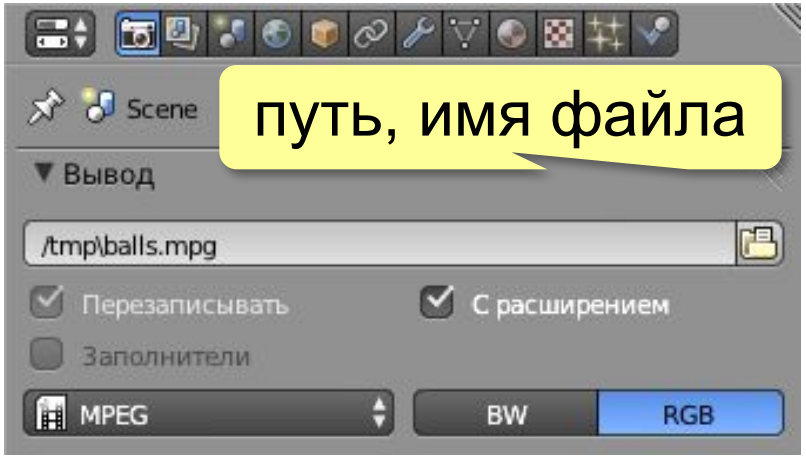
Мягкие тела

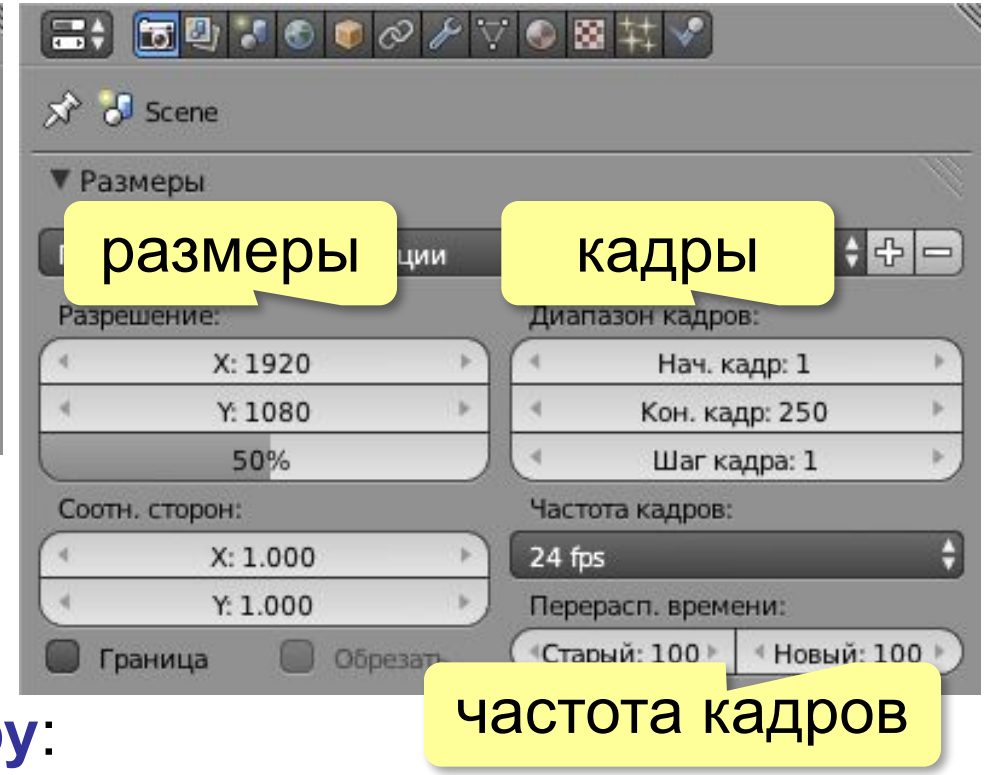
- моделирование тел, обладающих упругостью
- грани приобретают свойства пружинок
- учёт законов физики
- столкновения, деформации, отталкивание



Рендеринг видео

Форматы: **AVI**, **H.264**, **MPEG**, **Xvid**, **Ogg Theora**





Требования к компьютеру:

- быстродействующий процессор
- большой объём оперативной памяти

Рендер-фермы: farmerjoe.infoРендер-фермы: farmerjoe.info, renderfarm.fi

Трёхмерная графика

§ 74. Язык VRML

Что такое VRML?

Трёхмерные сцены – векторные объекты!

- нужен универсальный текстовый формат для обмена данными
- просмотр в браузерах (с помощью плагина – *plugin*)

VRML (*Virtual Reality Modeling Language* = язык моделирования виртуальной реальности)

Примеры использования:

- виртуальная экскурсия по Мамаеву кургану wwwwww.www.volgogradwww.volgograd.www.volgograd.ruwww.volgograd.ru/www.volgograd.ru/mamayevwww.volgograd.ru/mamayev-www.volgograd.ru/mamayev-kurgan
- *3D*-модели исторических событий

X3D: современный стандарт для работы с 3D-моделями на основе VRML и XML:

wwwwww.www.parallelgraphicswww.parallelgraphics.www.parallelgraphics.comwww.parallelgraphics.com/w

Программное обеспечение

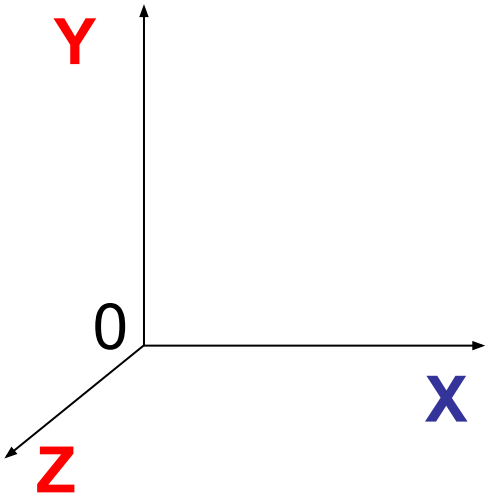
Файлы `*.wrl` – простой текст!

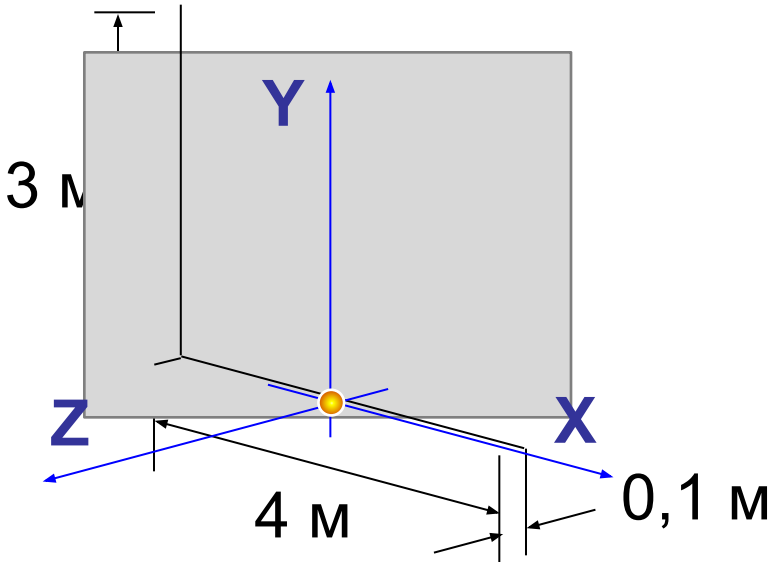
world

форма объектов, цвет, текстура, прозрачность, освещение, анимация, звуки, …

- текстовые редакторы
- VRML-редакторы,
  WhiteDune:
  vrmlvrml.vrml.cipvrml.cip.vrml.cip.icavrml.cip.ica.vrml.cip.ica.univrml.cip.ica.uni-vrml.cip.ica.uni-stuttgartvrml.cip.ica.uni-stuttgart.vrml.cip.ica.uni-stuttgart.devrml.cip.ica.uni-stuttgart.de/vrml.cip.ica.uni-stuttgart.de/dune
- плагины для браузеров
  Cortona3D Viewer: www.cortona3d.com/cortona
- программы для просмотра 3D-сцен
  view3dscene:

Стена





Стена в плоскости XOY:

форма

свойство

кодировка

«коробка»

X Y Z

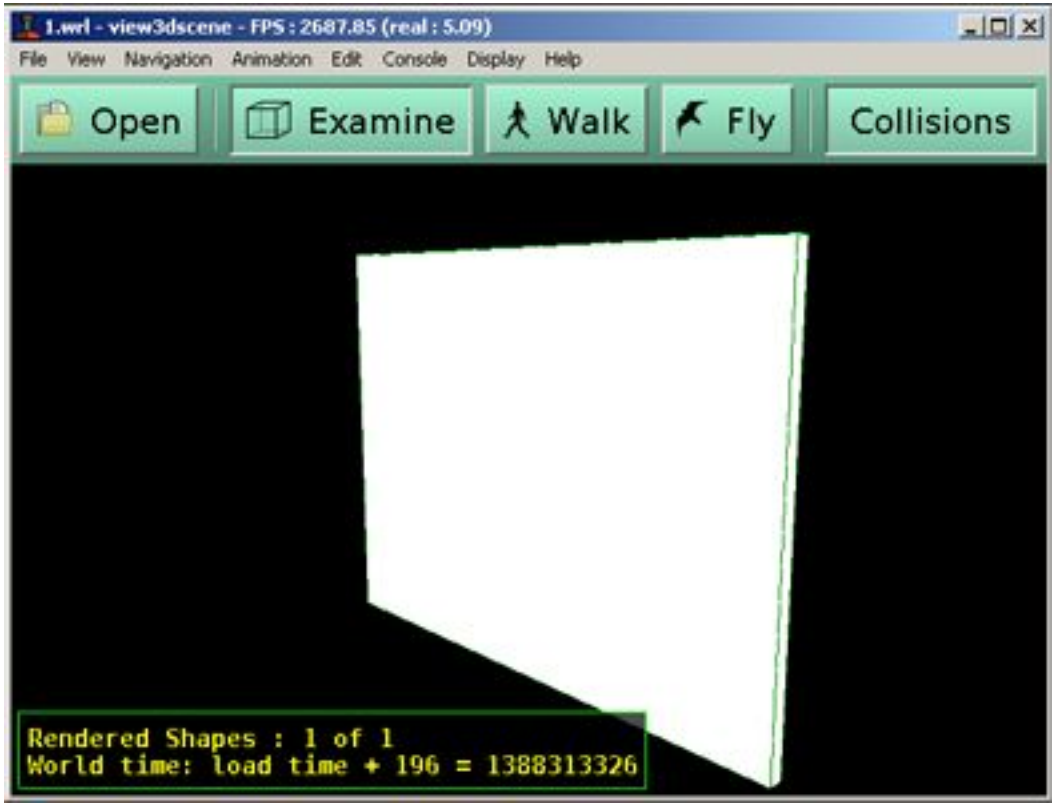
размеры

```
#VRML V2.0 utf8
Shape
   {
   geometry Box { size 4 3 0.1 }
   }
```

Просмотр модели

view3dscene:

просмотр | прогулка | полёт | столкновения



по умолчанию:
- центр в начале координат
- цвет – белый

Точка наблюдения

```
#VRML V2.0 utf8
Shape
   {
   geometry Box { size 4 3 0.1 }
   }
Viewpoint
   {
   position 0 0 5
   }
```

начальная позиция наблюдателя (X, Y, Z)

Материал: цвет

```
#VRML V2.0 utf8
Shape
 {
 geometry Box { size 4 3 0.1 }
 appearance Appearance
   {
   material Material{ diffuseColor 0.7 0.7 0.7 }
   }
 }
```

внешний вид

материал

цвет диффузного отражения

R G B

от 0 до 1

Материал: текстура

```
#VRML V2.0 utf8
Shape
  {
  geometry Box { size 4 3 0.1 }

  appearance Appearance
    {
    texture ImageTexture { url["texture.png"] }
    }
  }
```

текстура

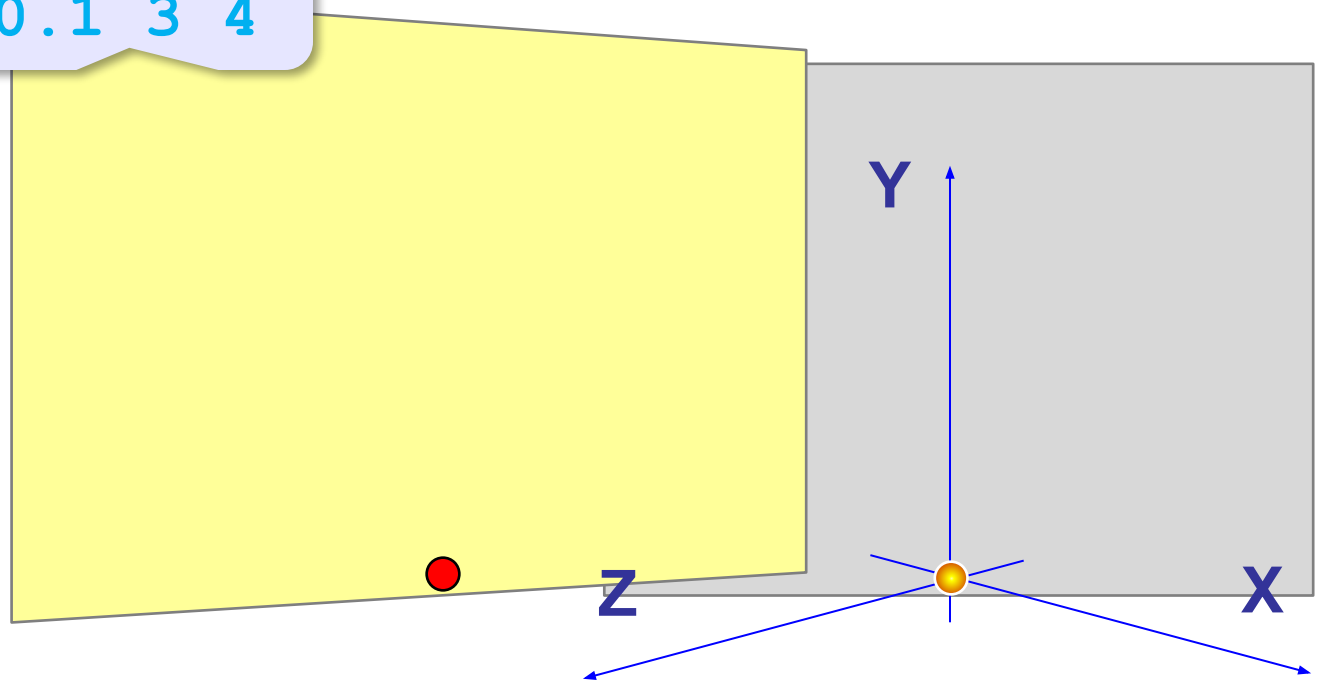
рисунок

адрес файла

Рисунок растягивается на всю поверхность!

«Левая стена»

размеры по осям
```
size 0.1 3 4
```



смещение опорной точки по осям
```
translation -2.05 0 1.95
```

Преобразования (*Transform*)

преобразования

rotation – поворот
scale – изменение размеров

```
Transform
   {
   translation -2.05 0 1.95
   children
      [
      Shape
         {
         geometry Box { size 0.1 3 4 }
         }
      ]
   }
```

перенос опорной точки

дочерние объекты

Конец фильма

ПОЛЯКОВ Константин Юрьевич
д.т.н., учитель информатики
ГБОУ СОШ № 163, г. Санкт-Петербург
kpolyakov@mail.ru

ЕРЕМИН Евгений Александрович
к.ф.-м.н., доцент кафедры мультимедийной дидактики и ИТО ПГГПУ, г. Пермь
eremin@pspu.ac.ru

Источники иллюстраций

1. www.fallingpixel.com
2. 3darchitect.by
3. oldchisinau.com
4. aneyeforscience.tumblr.com
5. sketchup.google.com
6. games.imhonet.ru
7. www.flightgear.org
8. www.solidworks.ru
9. www.carbodydesign.com
10. ru.wikipedia.org
11. авторские материалы