

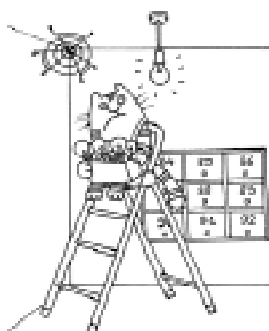
*Колесов Юрий Борисович,  
Макарова Наталья Владимировна,  
Сениченков Юрий Борисович,  
Титова Юлияна Францевна*

## ШКОЛА МОДЕЛИРОВАНИЯ 2003.

### ЗАНЯТИЕ 6.

## ВИЗУАЛИЗАЦИЯ МОДЕЛЕЙ

В руководстве для начинающих изучать Model Vision Studium, созданном Федосовым Борисом Трофимовичем (Рудненский индустриальный институт, Казахстан), в качестве примера рассматривается модель электрического фонарика и его двумерный и трехмерные образы.

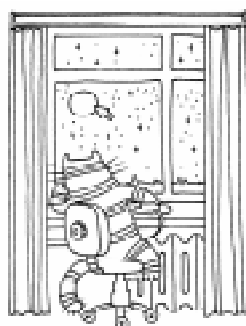


Пример с простейшим двумерным образом мы поместили в папку «Примеры к статье» под именем «Лампочка\_Федосова». Этот простейший фонарик можно зажигать от выключателя (рисунок 1). (Внимание! Для того чтобы лампочка в примере загорелась, не забудьте запустить приложение – команда «Пуск» основного меню испытательного стенда.) Именно его мы и выбрали в качестве примера, с которого мы начнем разговор о возможностях визуализации в Model Vision Studium.

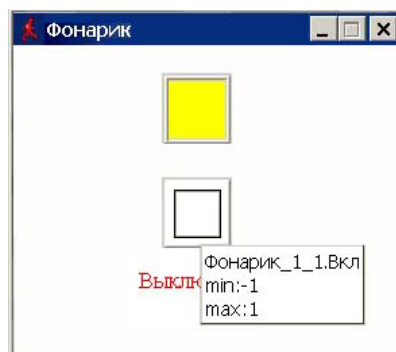
В Model Vision Studium предусмотрена стандартная «математическая» графика, а именно, окно «Временная диаграмма», перестраиваемое при необходимости в окно «Фазовая диаграмма», двумерная и трехмерная графика. Окно «Временная диаграмма» мы уже неоднократно использовали, поэтому перейдем сразу же к двумерной анимации.

Для реализации двумерных образов существует окно «2D-анимация» и специальная панель двумерных активных элементов (рисунок 2).

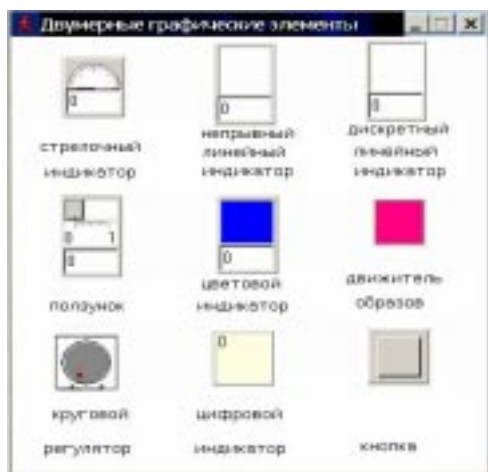
Двумерные активные элементы используются в основном для проектирования пультов управления. Как и на настоящем пульте управления, на созданном виртуальном пульте можно разместить измерительные приборы (стрелочный индикатор), индикаторы (линейный индикатор, примером которого может служить обыкновенный градусник), управляющие элементы (та же кнопка или движок). Особое место занимают диаграммы и движитель образов. Диаграммами мы уже пользоваться научились, поэтому поговорим о движителе образов.



Представим себе картинку, например, звездное небо, и объект – луну, которую мы хотим перемещать по заданному закону. Очевидно, что можно взять картинку (фон), изображение луны и



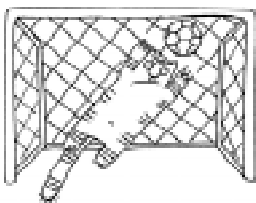
**Рисунок 1.** Двумерное изображение фонарика и выключателя.



**Рисунок 2.**  
Компоненты двумерной анимации.

перемещать это изображение относительно картинки. Следовательно, для работы движителя образов необходимо подобрать фон (файл с расширением .bmp), перемещаемый образ и задать закон изменения координат ( $x$ ,  $y$ ) образа относительно фона.

Попробуйте создать образ движущейся Луны самостоятельно, изучив пример «Прыгающий мячик», который вы найдете в папке ProgramFiles/mv30/examples/ball при стандартной установке среды Model Vision Studium. В этом проекте строится модель движения мяча, брошенного с начальной скоростью  $V_0$  под углом  $\alpha$  к горизонту. После броска мячик взлетает вверх, достигает



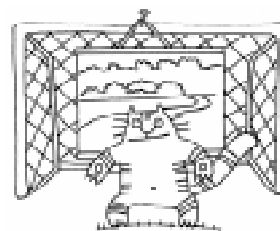
**Рисунок 3.** Использование компонентов «движитель образов», «стрелочный прибор» и «линейный индикатор».

некоторой наивысшей точки подъема и затем падает вниз. Достигнув земли на некотором удалении от точки броска, мячик ударяется о нее, отскакивает и снова летит вверх. Предполагаем, что на мяч не действует сопротивление воздуха и отскок является абсолютно упругим, то есть численное значение скорости до и после отскока не меняется, а угол падения равен углу отражения.

Не будем останавливаться на описании математической модели. Она известна из школьного курса физики. Восстановить математическую модель вы можете, исследовав окна проекта Класс, Система уравнений и др.

Цель нашей сегодняшней работы – визуализация модели.

Откройте в среде MVS проект «Прыгающий мячик» и запустите его. Откроется окно «Испытательный стенд» модели. Одновременно с виртуальным стендом открывается созданная заранее двумерная визуальная модель (рисунок 3). В окне расположен рисунок фона (дорога, убегающая вдаль) и на нем движущийся объект «мяч», а



также двумерные графические элементы: два стрелочных индикатора, отображающих текущую высоту мячика (Ball height) и вертикальную скорость (Vertical velocity), и движитель образов (Kinetic Energy – кинетическая энергия). Щелкните по кнопке Пуск на панели испытательного стенда и наблюдайте за движением мячика и показаниями стрелочных индикаторов. Остановите проект.

Щелкнув правой кнопкой мыши на объект, просмотрите настройки этих графических примитивов: с какими переменными проекта они связаны, как задаются?

Попробуйте построить самостоятельно аналогичную двумерную анимацию.

Разработка подобных окон осуществляется по очень простым правилам.

1. Создаем новое двумерное окно (команда Окна/Новая 2D анимация). Создавая

новое окно и размещая его на виртуальном стенде, среда MVS предупреждает, что обновлять изображение в окне надо через равные промежутки времени, чтобы образы двигались плавно, без рывков, и спрашивает у вас на это разрешение.

2. Открываем панель стандартных двумерных компонентов и мышкой перемещаем нужные нам образы компонентов в поле окна.

3. Для каждого компонента щелчком правой кнопки мыши открываем диалоговое окно или контекстное меню и настраиваем компонент так, как этого требует задача. А именно, указываем в каком виде должен предстать перед нами компонент и какие переменные он должен отображать (изменять).

Меню компонента «движитель образов» показано на рисунке 4. Первый его пункт «Variables...» открывает окно, которое носит информационный характер и показывает, какая переменная модели связана с осью абсцисс (нижняя горизонтальная сторона заштрихованного прямоугольника), а какая с осью ординат (левая вертикальная сторона). Как вы догадались, заштрихованный прямоугольник и есть часть плоскости, по которой будет двигаться образ. Чтобы увидеть этот прямоугольник, надо выделить в контекстном меню пункт «Track rectangle» (он будет отмечен «галочкой»), а затем мышью расположить его внутри окна в нуж-

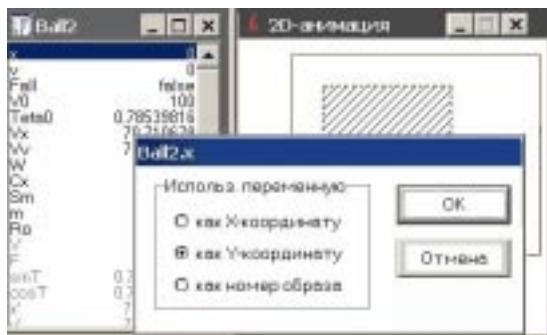


Рисунок 5.

Связывание переменных с осями.

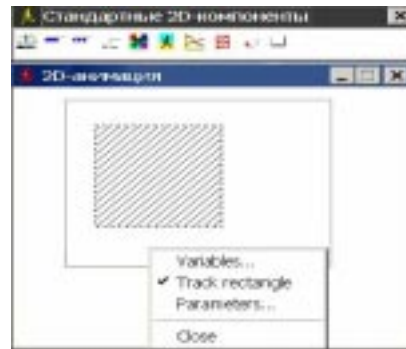


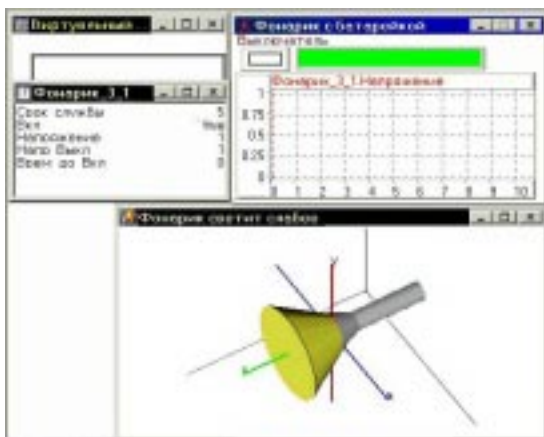
Рисунок 4. Панель инструментов, новое двумерное окно и диалог «движителя образов».

ном месте и придать ему нужные размеры. Переменные связываются с осями обычным приемом – открываем окно «переменные», выбираем нужную переменную и мышью перетаскиваем ее в поле окна «движитель образов». Открывшийся диалог поможет вам связать переменную с конкретной осью (рисунок 5).

И, наконец, необходимо указать, где найти фон и сам образ (третья строка меню – «Parameters», рисунок 6). В верхней таблице вы можете только указать предельные значения переменных, третье поле – «Value» – служит для отображения значения координат во время движения объекта. Если вы приостановите выполнение модели и откроете этот диалог, то в этих полях вы увидите текущие координаты образа. В поле «BackGround» вы выбираете имя файла, который будет служить фоном, в окне Color задаете при необходимости цвет фона,



Рисунок 6. Выбор фона и образов.



**Рисунок 7.** Трехмерное изображение светящегося фонарика.

а в окне «Images» – имя образа. (Файлы с изображением фона и образов необходимо предварительно расположить в папке, где находится ваша модель, и тогда их имена вы увидите в окне «Images»). В нашем случае фоном является файл с изображением дороги, образом – мяч, а в результате он будет прыгать поперек дороги на ближнем плане.

«Движитель образов» наиболее сложный двумерный «пассивный» компонент – остальные много проще. Например, для работы стрелочного прибора достаточно лишь указать имя отображаемой переменной и ее предельные значения. Ничуть не сложнее активные элементы, например – кнопка. С ее помощью вы можете менять текущее значение выбранной вами переменной булевского типа, то есть принимающей всего два



**Рисунок 8.** Графические примитивы трехмерной анимации.

значения – ИСТИНА и ЛОЖЬ (Да и Нет, Включено и Выключено и т. д.).

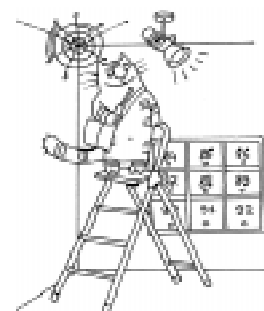


Еще одним примером использования двумерных компонентов может служить двумерная модель бассейна (папка Program Files/mv30/Examples/Pool).

В руководстве Федосова существует также трехмерное изображение фонарика (рисунок 7).

Руководство и пример можно найти на сайте [www.exponenta.ru](http://www.exponenta.ru) в разделе, посвященном отечественным разработкам («Другие пакеты. MVS»). Давайте и мы создадим свой трехмерный образ лампочки.

Откройте проект «3D\_Анимация\_Федосова» (папка «Примеры к статье») и запустите модель. На испытательном стенде вы найдете окно двумерной анимации «Фонарик с батареейкой». На нем расположена кнопка включения фонарика и графический элемент «линейный индикатор», моделирующий заряд батарейки, а также окно диаграммы, на которой при работе модели будет отображаться график уменьшения заряда батарейки с течением времени. Здесь же расположено окно трехмерной анимации (3D-анимации) с именем «Фонарик светит слабее», моделирующее собственно само изображение фонарика.



Нажмите на кнопку «Пуск», а затем нажимайте на кнопку выключателя, включая и выключая фонарь. Следите за изображениями в окнах двумерной и трехмерной анимации. Остановите проект.

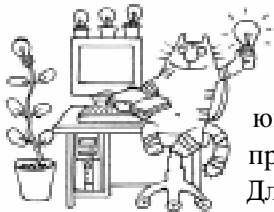
Попробуйте создать самостоятельно окно 3D-анимации.

Редактор трехмерных изображений в Model Vision Studium представляет собой конструктор, который позволяет собирать нужный образ из графических примитивов.

В списке графических примитивов вы найдете (рисунок 8):



- линию,
- сферу,
- конус,
- тор,
- пружину,
- брус,
- четырехугольник,
- стрелку,
- поверхность,
- текст.



Попробуем создать трехмерный образ лампочки из имеющихся графических примитивов лампочки. Для изображения лампочки выберем три графических примитива – две сферы и пружину (рисунок 9).

Откройте новое 3D графическое окно (Окна/Новая 3D-анимация). Появится трехмерная сцена с осями координат. Щелкните правой кнопкой мыши в поле окна, откроется диалоговое окно. Первая часть диалога «свойства» предназначена для создания картинок из примитивов. Вторая часть – справочная система.

Открыв «свойства», вы обнаружите две закладки. Первая – «Общие установки» – касается характеристик сцены и осей, вторая – «Объекты» – собственно рисунка. На рисунке 8 вы видите графический редактор: слева расположены пиктограммы примитивов, а справа характеристики каждого из выбранных примитивов. При щелчке на примитиве его имя появляется в левой части окна, а свойства этого объекта – в правой. Вы можете изменить эти свойства. Подсвеченное синим цветом имя объекта «line1» (линия1) говорит о том, что мы работаем с этим примитивом и видим справа его характеристики. Очевидно, что положение отрезка полностью задается двумя точками  $(x_1, y_1, z_1)$  и  $(x_2, y_2, z_2)$ . Написанные в поле значений константы заданы по умолчанию. Если теперь в поле переменных поместить конкретные переменные, описанные в проекте, то соответствующая координата будет меняться по нужному вам закону. У от-

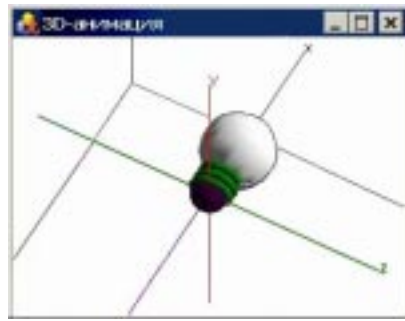
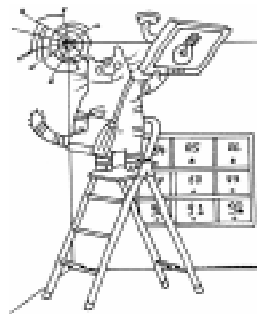


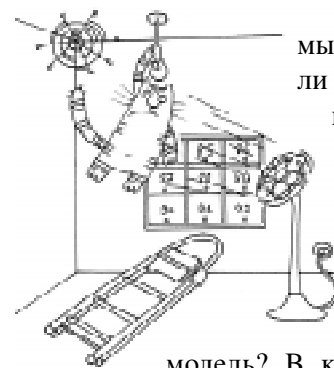
Рисунок 9. Трехмерный образ лампочки.

резка вы можете менять толщину линии и ее цвет, у других объектов будут другие характеристики.



Создаем образ лампочки – одну сферу (оголовок цоколя лампочки) помещаем в начало координат, далее вдоль оси X располагаем пружину (винтовая часть цоколя) и затем еще одну сферу большего диаметра (колба лампы) помещаем так, чтобы ее центр оказался смещенным на заданное расстояние вдоль оси X (см. пример «Лампочка» из папки «Примеры к статье»).

Образ лампочки мы создали, а можно ли повесить эту «лампочку» к потолку на шнуре и раскачивать ее небольшими порывами ветра? Можете ли вы самостоятельно создать такую модель? В качестве подсказки предложим за основу взять математическую модель маятника, движущегося под воздействием синусоидального возмущения, и выбрать систему координат так, чтобы основание маятника было прикреплено к «потолку». Точка подвеса маятника будет фиксирована, а координаты другого конца стержня вам и надо рассчитать. Их надо будет связать с созданными образами, и они начнут перемещаться так, как вы хотите. Нач-



модель? В качестве подсказки предложим за основу взять математическую модель маятника, движущегося под воздействием синусоидального возмущения, и выбрать систему координат так, чтобы основание маятника было прикреплено к «потолку». Точка подвеса маятника будет фиксирована, а координаты другого конца стержня вам и надо рассчитать. Их надо будет связать с созданными образами, и они начнут перемещаться так, как вы хотите. Нач-

ните с графических элементов – «линия» и «шар», а затем замените «линию» на «пружину» (витая морская веревка), а «шар» – на созданный вами образ лампочки.

Успехов вам!

В завершение этой статьи, приведем две трехмерные анимации (рисунки 10 и 11), созданные Г.Н. Петровым, доцентом кафедры теории машин и механизмов Санкт-Петербургского политехнического университета. (Эти и другие примеры вы найдете в папке ProgramFiles/mv30/examples). Приведенные картинки иллюстрируют работу реальных механизмов и уже не столь просты, как приведенные в нашей статье. Несомненно, они требуют больше усилий, чем потратили мы, но и не чрезмерно много, чтобы их не создавать вообще.

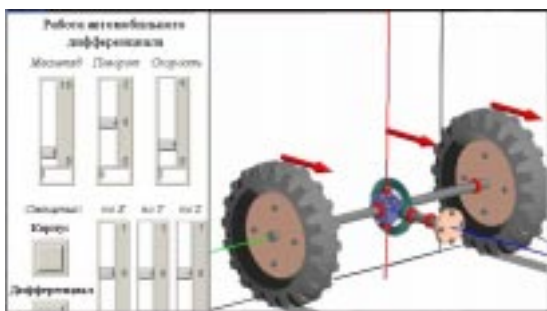


Рисунок 10. Дифференциал автомобиля.

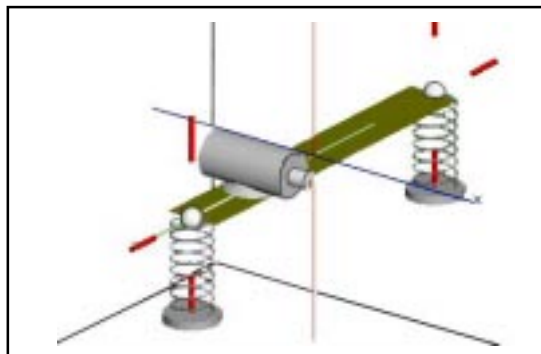


Рисунок 11. Мотор на платформе.

**Колесов Юрий Борисович,**  
канд. техн. наук,  
доцент кафедры Распределенных  
вычислений и компьютерных сетей  
Санкт-Петербургского  
Политехнического Университета,  
**Макарова Наталья Владимировна,**  
доктор пед. наук, канд. техн. наук,  
проректор МБИ (СПб), зав. каф.  
информационных систем и технологий,  
**Сениченков Юрий Борисович,**  
канд. физ.-мат. наук,  
доцент кафедры Распределенных  
вычислений и компьютерных сетей  
Санкт-Петербургского  
Политехнического Университета,  
**Титова Юлия Францевна,**  
канд. пед. наук, доцент кафедры  
информационных систем и  
технологий МБИ (СПб).

© Наши авторы, 2003.  
Our authors, 2003.