Развитие программных средств научной визуализации

В.Л. Авербух[1], П.А. Васёв[1], М.О. Бахтерев[1], Я.Д. Кузнецов[2]

ИММ УрО РАН[1], ОАО ОКБ Новатор[2]

Визуализация играет важную роль при проведении научных исследований. В связи с этим в арсенале визуализации создано множество программных средств. Но что делать, если исследуемое явление настолько новое, что нет готовых программ, визуализирующих его?

Можно все же попытаться выразить визуальные сущности в терминах готовых систем визуализации. Наиболее прогрессивные из систем даже предлагают скриптовые возможности для своего программирования. Однако такое выражение не всегда осуществимо или неудобно.

Можно создать программу для визуализации с нуля. Этот подход обеспечивает максимальную гибкость, но сопряжен с большими затратами.

В качестве альтернативного способа описания требуемой визуализации авторами предлагается следующий подход.

Визуальная сцена описывается с помощью специального языка. Задача этого описания - рассказать, какие данные и как надо визуализировать.

Предлагаемый язык описания должен быть достаточно выразительным, чтобы обеспечить: работу с массивами, чтение файлов, указание произвольных атрибутов для данных, запуск программ обработки данных, работу с параметризованными данными, и т.д.

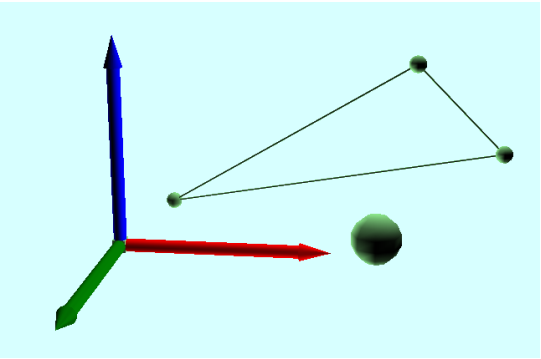
Сформированное описание сцены считывается программой визуализации, которая интерпретирует его и сообразно своим возможностям отображает сцену.

Для облегчения чтения описания существует отдельная программа. Она разбирается во всех тонкостях языка и способна служить сервером данных, предоставляя простой протокол.

Принципиально важно, что язык описания не фиксирует типы визуальных сущностей. Язык относится к описанию как к набору неких данных с именами. А семантическое значение этих имен определяется на уровне отдельных соглашений.

Рассмотрим пример, изображенный в таблице 1.

Таблица 1. Пример

Описание сцены	Полученное изображение
set scene.model1.spheres = array 3 3 1 1 1 2 5 2 2 4 3 set scene.model1.lines = array 3 6 1 1 1 2 5 2 1 1 1 2 4 3 2 4 3 2 5 2 set scene.model2.spheres.radius = 0.3 set scene.model2.spheres = array 1 3 2.2 3.6 1	

Система визуализации делает запрос – «дайте мне все имена вида *scene.**», и анализирует полученные имена. В нашем примере система среди прочего научена реагировать на суффикс «*spheres*». Встречая его, она ожидает, что данные с этим именем есть массив троек чисел, означающих центры сфер. Система запрашивает этот массив и отображает сферы.

Кроме того, система дополнительно пытается считать данные с припиской «*.radius*», и если такие данные указаны – интерпретирует их как радиус сфер.

Предложенный подход визуализации имеет ряд положительных качеств, которые в настоящее время исследуются авторами. Результаты публикуются по адресу www.view.lact.ru.