

КОНСТРУКТОР СПЕЦИАЛИЗИРОВАННЫХ КОНСТРУКТОРОВ ВИЗУАЛИЗАЦИИ

М. О. Бахтерев, П. А. Васёв

Институт математики и механики им. Н. Н. Красовского
Уральского отделения Российской академии наук, г. Екатеринбург

При решении задач научной визуализации применяются два вида программ – универсальные системы визуализации и специализированные системы. Универсальные позволяют пользователю настроить конвейер визуализации, составляя визуализацию из готовых блоков. Специализированные системы позволяют сделать работу пользователя эффективной в его конкретной задаче визуализации и показывают именно те особенности, которые важны для данной задачи, обеспечивая корректность и информативность визуализации [1].

Наш опыт показывает, что даже специализированные системы, созданные под решение задач конкретного пользователя или группы пользователей, зачастую требуют их дальнейшего допрограммирования, улучшения под постоянно усложняющиеся новые запросы пользователя. Обычно такое допрограммирование приходится выполнять специалисту – разработчику системы визуализации.

Мы считаем, что существенную часть допрограммирования возможно передать в руки пользователя, если предоставить ему понятный и простой в использовании интерфейс программирования системы визуализации. Тогда пользователь сможет самостоятельно конструировать алгоритм визуализации, быстрее получая требуемый ему результат, без посредников и интерпретаторов в лице специалистов. Этого можно достичь, оснатив универсальную систему визуализации, в которой пользователь обычно самостоятельно и занимается программированием алгоритма визуализации, возможностями специализированных систем. Мы предлагаем следующую конструкцию.

1. Специализированная система визуализации с минимально необходимым набором примитивов.

2. Визуальный язык программирования, построенный на стандартных примитивах GUI, мы называем такой язык «кнопочным». Мы предпочитаем «кнопочные» интерфейсы другим методам, предполагая, что они позволяют оставить пользователя в контексте задачи визуализации, не переключая его мышление в другой контекст, в контекст «программирования». Язык позволяет собирать более сложные визуализации из базовых специализированных примитивов.

3. Этот визуальный язык может иметь разные семантические модели. Мы экспериментируем с моделью, в которой пользователь добавляет «объекты» в «слои». У каждого вида объектов и слоёв своя семантика. Комбинация связанных и настроенных объектов в слоях порождает требуемый визуальный образ.

4. Специализированную систему можно расширить, написав на некотором не «кнопочном» языке программирования для неё «внешние» модули, определяющие новые виды слоёв и объектов для слоёв.

Реализовав эти элементы, мы даём пользователю специализированный конструктор визуализации, который он может запрограммировать на получение требуемого визуального образа в рамках решаемой им задачи визуализации.

Для упрощения создания подобных систем мы создали специальный язык программирования (что в общем случае не является обязательным шагом). Язык позволяет описывать а) трехмерную визуализацию, б) преобразования данных для визуализации, в) пользовательский интерфейс. Язык является текстовым,

использующим идеи из языков Lisp и QML. Реализован интерпретатор этого языка для веб-браузеров. На этом языке мы программируем предложенные элементы **1-4**, получая различные системы визуализации. Подробности о языке и его интерпретаторе публикуются на странице <https://github.com/viewzavr/>.

Мы считаем перспективной возможность управления фильтрующими звеньями в режиме online-визуализации, в том числе когда исходные данные визуализации отправляются на обработку на суперкомпьютеры, а настроенные в интерфейсе программы визуализации алгоритмы фильтрации и расчёты дополнительных данных запускаются на специализированных машинах.

Отметим, что с предложенным подходом хорошо согласуется и прямое управление (direct manipulation), в котором пользователь управляет (программирует) визуализацию не только через двумерный интерфейс, но и через непосредственное взаимодействие с визуальными образами. Объединение возможностей online-визуализации и прямого управления в рамках одной системы открывает путь к реализации виртуального испытательного стенда [2].

Литература.

1. Авербух В.Л., Разработка средств компьютерной визуализации для научных исследований // Труды первой международной конференции «Трёхмерная визуализация научной, технической и социальной реальности», 04-06.02.2009 года, г. Ижевск, Том I, стр. 8-11 [<http://www.cv.imm.uran.ru/e/70912>].
2. V. L. Averbukh, N. V. Averbukh, P. Vasev, I. Gajniyarov and I. Starodubtsev, The Tasks of Designing and Developing Virtual Test Stands // 2020 Global Smart Industry Conference (GloSIC), Chelyabinsk, Russia, 2020, pp. 49-54, doi: 10.1109/GloSIC50886.2020.9267835 [<http://www.cv.imm.uran.ru/e/3241753>].