

УДК 378.091.33:004.4'236

## СОВРЕМЕННЫЕ ТЕНДЕНЦИИ ВИЗУАЛЬНОГО ПРОГРАММИРОВАНИЯ



*Осадчая Екатерина Петровна*  
кандидат педагогических наук, доцент

*Марчук Николай Сергеевич*  
аспирант

*Мелитопольский государственный педагогический университет  
имени Богдана Хмельницкого*

## CURRENT TRENDS VISUAL PROGRAMMING

*Osadchaya Ekaterina Petrovna*

*Candidate of Pedagogical Science, associate professor,  
Melitopol State Pedagogical University named after B. Khmelniyskiy*

*Marchuk Nickolay Sergeevich*

*Melitopol State Pedagogical University named after B. Khmelniyskiy*



## АННОТАЦИЯ

В статье проанализированы основные понятия визуального программирования, на основе чего выделено три семантические группы значений: среды визуальной разработки, способ создания и проектирования программ и систем путем манипулирования графическими объектами; программы для визуальных языков программирования. Предлагается следующее определение визуального программирования – процесс представления программных структур (алгоритмов и данных) с помощью многомерных элементов (значков, иконок, форм, блоков) с использованием визуального языка программирования.

**Ключевые слова:** визуальное программирование, язык визуального программирования, среды визуального программирования

## ABSTRACT

The article analyzes the basic concepts of visual programming, which is allocated on the basis of three groups of semantic values: visual development environment to create and design programs and systems by manipulating graphic objects; program for visual programming languages. The following definition of visual programming - the process of submission of program structures (algorithms and data) using the multi-dimensional elements (icons, icons, shapes, blocks) using a visual programming language.

**Keywords:** visual programming language visual programming, visual programming environment

Устоявшийся алгоритмический подход к созданию программ сегодня не отвечает в полной мере современным запросам общества и экономики на эффективную и быструю разработку программных продуктов. Поэтому сегодня актуальной становится потребность в обучении будущих программистов визуальной технологии программирования, результатом которой должна быть разработка программ с функционалом и интерфейсом, соответствующие требованиям современных операционных систем.

Технологии визуального программирования возникли уже в конце прошлого века, но на сегодня нет единого мнения относительно терминологии и понимания парадигмы визуального программирования как в научно-методических работах да и в специальной профессиональной литературе. Поэтому мы поставили за цель проанализировать семантическое ядро ВП и дать определение сущности понятий «визуальное программирование», «визуальная среда», «визуальный язык программирования». Теоретические и практические аспекты визуального

программирования отражены в современных работах педагогов и ИТ-специалистов, а именно: Ахметзянова И.М., Завадского И.А., Заболотного Р.И., Кашеева Л.Б., Коваленко С.В., Коваленко С.Н., Пунтамбекера А.А., Хайдаров К.А., и др.

Одним из первых попытку определения понятий визуального программирования сделал Майерс Б.А., который отмечал, что визуальное программирование (ВП) относится к любой системе, которая позволяет пользователю написать программу в двух (или более) измерениях [1]. Автор различал понятия визуального программирования и программ визуализации, которые понимались им как класс программных средств, помогающих сделать программы более понятными благодаря использованию графики, и обычно используются для обучения студентов программированию.

Сейчас к сфере ВП относят как визуальные языки программирования и визуальные среды разработки программ, так и средства программирования 2D и 3D-графики, визуализации данных (графики, диаграммы) и построения GUI

(Graphical user interface – графический пользовательский интерфейс).

По мнению Камаева В.А. и Костерина В.В. ВП заключается в автоматизированной разработке программ с использованием особой диалоговой оболочки, чаще всего базируется на объектно-ориентированном программировании и является его логическим продолжением. Наиболее часто ВП используется для создания интерфейса программ и систем управления базами данных [8, с. 319]. Такого же мнения придерживаются Ахметзянов И.М. и Ракова И.К. [4], Кашеев Л.Б. и Коваленко С.В. [9] предлагая освоение основ ВП будущими программистами на базе создания приложений с использованием среды визуальной разработки Borland Delphi, а также Завадский И.А. и Заболотный Р.И. [6], предлагая методику обучения программированию на примерах разработки программ с графическим интерфейсом в визуальной среде Visual Basic.NET и т.д..

По нашему мнению, такой подход иллюстрирует лишь часть технологии ВП: использование визуальных средств программирования, которые позволяют облегчить процесс разработки интерфейса программы, но базируются на текстовых языках программирования, которые можно назвать языками программирования для визуальной среды (ObjectPascal, Visual Basic). Визуальные средства разработки оперируют преимущественно стандартными объектами – окнами, списками, текстами, которые легко можно связать с данными из базы данных и отобразить на экране монитора, а также элементами управления – кнопками, переключателями, флажками, меню и т.п..

К технологией ВП относят работу со средами, которые используют блоки команд, перетаскивая которые с помощью манипулятора мыши и выставляя в нужном порядке, можно создавать различные программы (Scratch, App Inventor, Google Blockly и др.). Использование таких сред иллюстрирует понимание ВП как способа создания программ с помощью блоков, сопоставляя которые можно составить набор инструкций. Для этого используется как традиционный (цикл, условные операторы), так и дополнительный набор управляющих конструкций (движение, вращение, рисование, проигрывание звука и т.д.), процедуры и функции с параметрами, полноценный набор логических выражений, а также возможности работы с цветом и графикой. Для таких сред присущий сильный акцент на визуальной составляющей, что для профессиональной работы в некоторых из них компенсируется возможностью компиляции на традиционный язык программирования на выбор (например, в Google Blockly на Python, JavaScript, Dart, PHP, XML).

Кроме блоков как графические объекты для ВП могут использоваться такие структуры как блок-схемы. То есть в отличие от языков низкого, среднего, высокого уровня, где программирование и язык носят существенно линейный, последовательный характер, в визуальных средах

мы имеем дело с существенно разветвленными «пространственными» структурами типа блок-схем. При этом «кирпичиками» этих блок-схем являются заранее разработанные подпрограммы и функции с унифицированным автоматическим «интеллектуальным» интерфейсом. При соединении таких «кирпичиков» их взаимное интерфейс настраивается без участия программиста [13].

В качестве примеров таких программ можно назвать Keysight VEE, HiAsm, Marten, Lab View. Например, HiAsm является системой визуального проектирования и разработки приложений, не требующей от пользователя знания языков программирования и особенностей функционирования операционной системы и позволяет быстро и легко писать (рисовать) небольшие программы. Весь процесс проектирования состоит в размещении элементов на рабочем столе приложения и их связывании между собой, что в первом приближении можно сравнить с построением обычного алгоритма на основе стандартизированных функциональных блоков (циклов, условных блоков, переходов, операторов и прочего). Возможность вставки скриптового кода на языках Java и Visual Basic или непосредственно кода того языка, на котором генерируется проект, позволяет использовать отсутствующие в стандартном наборе компонентов функции или выполнять сложные математические расчеты [11].

Эти и подобные системы разработки программ, дают представление о процессе программирования как о процессе проектирования или моделирования с использованием визуальных средств (элементов блок-схем) и средств языков программирования (в части генерации или дописки кода). Большая часть разработки может быть выполнена путем переноса объектов «мышкой» по принципу «drag-and-drop». По нашему мнению, такие манипуляции нельзя назвать визуальным программированием. Ведь, если программирование является процессом создания программ с помощью языков программирования и язык программирования является формальной знаковой системой, предназначенной для записи компьютерных программ, то элементы блок-схем не являются элементами знаковой системы – это графические объект. Следовательно, такой процесс целесообразно назвать визуальным проектированием или моделированием программ.

В свободном онлайн словаре вычислительной техники [3] подается такая трактовка понятия «визуальный язык программирования» (ВЯП): любой язык программирования, который позволяет пользователю разработать программу в двух (или более) измерениях. Обычные текстовые языка не работают как двумерные, так как компилятор или интерпретатор обрабатывает их как одномерные потоки символов. ВЯП позволяет программировать из визуальных элементов, размещая в пространстве текстовые и графические символы. Среда визуального программирования предоставляет графические или изобразительные элементы, которые могут управляться пользователем в

интерактивном режиме в соответствии с какой-то конкретной пространственной грамматикой для построения программ. Как некоторые примеры ВЯП можно назвать DRAGON, Prograph, Pict, Tinkertoy, Fabrik, CODE 2.0 и Hyperpascal.

Языки Delphi, C++ Builder, C#, а также семейство Microsoft Visual, являются текстовыми языками, которые используют графический GUI Builder для того, чтобы сделать программирование интерфейсов проще. Поэтому мы не считаем их визуальными языками программирования. Ведь визуальные языки имеют неотъемлемое визуальное выражение, для которого нет никакого очевидного текстуального эквивалента [3].

По нашему мнению, целесообразной является классификация ВЯП, представленная Гасько Р.Т. [5], в которой автор разделяет ВЯП по уровню сложности на:

1) языки высокого уровня функциональности и сложности, как LabView или DRAGON, для моделирования электронных комплексов и систем управления беспилотным космическим кораблем Буран соответственно;

2) визуальные языки для программирования роботов, например, Microsoft Robotics Studio со встроенным специализированным визуальным языком программирования Visual Programming Language (VPL) или Lego Mindstorm, основанная на выше упомянутой системе LabView;

3) специализированные учебные языки для обучения основам программирования как, например, LOGO, Scratch, Squeak, RoboMind и ряд других.

Гасько Р.Т. отмечает, что в грамматике ВЯП применяются изображения, соединительные линии и другие графические элементы языка как механизм для определения синтаксиса. Они позволяют описать спецификацию языка как логическую структуру. Само описание собственно программного кода на этом языке характеризуется двумерным представлением в отличие от традиционного одномерного (сверху вниз) текстового кода [5].

Главное преимущество ВЯП заключается в том, что они позволяют наглядно представить программные структуры – алгоритмы и данные. Это отличает их от традиционных текстовых языков программирования, где такие многомерные структуры закодированы в одномерные строки с помощью достаточно сложного синтаксиса. Однако можно указать и на минусы визуальных средств – графических нотаций. Они пригодны для представления небольших объектов. В условиях увеличения количества визуальных элементов их видимость и удобство ссылок катастрофически падают. Поэтому технологию ВП следует с

осторожностью использовать для разработки крупных проектов.

Таким образом, на основе анализа понятия «визуальное программирование» в литературных источниках мы выделили несколько семантических групп: программирование с помощью сред визуальной разработки; способ создания и проектирования программ и систем путем манипулирования графическими объектами (блоками или элементами блок-схем); использования в процессе разработки программ визуальных языков программирования. По нашему мнению, к парадигме собственно визуального программирования целесообразно отнести понимание процесса разработки программ с помощью специальных визуальных языков, к которым относятся (LabView, Microsoft Visual Programming Language, DRAGON, Prograph, Pict, Tinkertoy, Fabrik, CODE 2.0, Hyperpascal). Ведь при таком понимании парадигмы визуального программирования не искажается понимание сути самого визуального программирования, которое мы предлагаем понимать как процесс представления программных структур (алгоритмов и данных) с помощью многомерных элементов (значков (иконок), форм, блоков) с использованием визуального языка программирования.

#### СПИСОК ЛИТЕРАТУРЫ:

1. Myers B.A. Taxonomies of Visual Programming and Program Visualization (1989). [Электронный ресурс]. URL: <http://www.cs.cmu.edu/~bam/papers/vltax2.pdf> (дата обращения: 27.09.2016)
2. Keysight VEE [Электронный ресурс]. URL: <http://www.keysight.com/ru/pc-1000003078%3Aeapsg%3Aapgr/agilent-vee?nid=-32809.00&cc=RU&lc=rus> (дата обращения: 27.09.2016)
3. The Free On-line Dictionary of Computing. [Электронный ресурс]. URL: <http://www.dictionary.com> (дата обращения: 27.09.2016)
4. Ахметзянов И.М., Ракова И.К. Визуальное программирование: лабораторный практикум. – СПб., 2008. – 121 с.
5. Гасько Р.Т. Візуальна навчальна веб-орієнтована мова програмування [Электронный ресурс]. URL: <http://ena.lp.edu.ua:8080/bitstream/ntb/15885/1/152-Hasko-319-320.pdf> (дата обращения: 27.09.2016)
6. Завадський І.О., Заболотний Р.І. Основи візуального програмування. – К.: Вид. група ВНУ, 2007. – 272 с.
7. Избачков Ю.С. Информационные системы: Учеб. для вузов. – СПб.: Питер, 2011. – 544 с.
8. Камаев В.А., Костерин В.В. Технологии программирования. – М.: Высш. шк., 2006. – 454 с.
9. Кашцев Л.Б., Коваленко С.В., Коваленко С.М. Информатика. Основи візуального програмування: навч. посіб.. – Харків: Ранок: Веста, 2011. – 191 с.
10. Хайдаров К.А. 4GL-Технологии. Основы визуального программирования [Электронный ресурс]. URL: <http://bourabai.kz/einf/4gl.htm> (дата обращения: 27.09.2016)
11. Что такое HiAsm? [Электронный ресурс]. URL: <http://www.hiasm.com/sdk.html> (дата обращения: 27.09.2016)