

10. Борщев А. В. Практическое агентное моделирование и его место в арсенале аналитика // Exponenta PRO. 2004. № 3–4 (7–8). С. 38–47.

11. Девятков В. В. Методология и технология имитационных исследований сложных систем: современное состояние и перспективы развития. М.: Вузовский учебник; ИНФРА-М, 2014. 448 с.

12. Шрайбер Т. Дж. Моделирование на GPSS / пер. с англ.; под ред. М. А. Файнберг. М.: Машиностроение, 1980. 592 с.

13. Борщев А. В. Применение имитационного моделирования в России – состояние на 2007 г. // сб. докл. III Всерос. науч.-практ. конф. «Имитационное моделирование. Теория и практика» (ИММОД-2007), СПб., 17–19 окт. 2007. Т. I. С. 11–16.

S. A. Dadenkov, E. L. Kon
Perm national research polytechnical university

COMPARISON OF AGENT-BASED AND DISCRETE-EVENT SIMULATION MODELS AND METHODS

In this paper agent-based and discrete-event modeling approach are compared. Through analysis of basic steps in simulation model development process, such as designing, testing and completion, the pros and cons of aforementioned methods are revealed.

Simulation modeling, discrete-event simulation, agent-based simulation, analysis methods simulation

УДК 004.43

А. В. Малов
Motorola Solutions

С. В. Родионов
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

К вопросу классификации современных парадигм программирования

Рассматриваются основные парадигмы и подходы к программированию, недостатки некоторых из существующих классификаций парадигм программирования. Подчеркивается, что множество современных подходов к программированию определяется не характеристиками языков программирования, а способами проектирования и организации программ и программных комплексов. Предлагается подход к классификации современных парадигм программирования.

Парадигмы программирования, языки программирования, классификация парадигм программирования

В настоящее время развитие информационных технологий идет очень динамично. Множество существующих языков программирования постоянно пополняется новыми языками [1], обновляются стандарты для существующих языков программирования. В основе каждого языка лежит одна либо несколько парадигм программирования. Следует отметить, что современные языки программирования, базирующиеся на какой-либо парадигме, часто обладают свойствами и механизма-

ми управления, относящимися к другим парадигмам программирования. Как правило, целью сочетания нескольких парадигм в одном языке являются удобство и упрощение решения требуемого множества задач с использованием данных языков. В настоящей статье представлен обзор существующих парадигм и подходов к программированию, рассматривается возможный подход к их классификации и построению иерархии.

Парадигму программирования можно определить как подход к разработке программ, который базируется на логически цельном множестве принципов. Каждая парадигма обладает свойствами, которые определяют ее оптимальность для решения определенных классов задач. В статье приведено возможное краткое описание понятия парадигмы программирования. Более детально с данным понятием можно ознакомиться, например, в [2], [3].

Существует достаточно большое число работ, посвященных современным парадигмам и языкам программирования. Тем не менее, в данных работах вопрос классификации не рассматривается подробно. Однако классификация парадигм программирования должна способствовать в процессе изучения современных языков программирования обобщению их свойств, сравнению базовых приемов программирования и формированию группы языков для изучения в рамках предметной области их применения.

В [1], [4], [5] авторы предлагают разделять языки и парадигмы программирования на 4 основных типа:

- императивные;
- функциональные;
- логические;
- объектно-ориентированные.

В [5], кроме перечисленных выше парадигм, выделяют дополнительные парадигмы программирования, к которым предлагается отнести:

- визуальное программирование;
- параллельное программирование;
- программирование ограничениями.

В [2] детально рассматриваются механизмы управления и сущности, на которых строятся современные языки программирования. В качестве основных групп языков программирования и парадигм авторы выделяют следующие:

- императивные;
- объектно-ориентированные;
- параллельное программирование;
- функциональные;
- логические;
- скриптовые.

Следует отметить, что предложенные в перечисленных работах подходы к классификации являются достаточно общими, в них не рассматриваются многие современные подходы к программированию.

В [3] достаточно подробно рассматривается большинство современных парадигм и подходов к программированию, но основной акцент дела-

ется на детальной структуризации, ориентированной больше не на парадигмы и подходы к программированию, а на возможности конкретных языков программирования.

Таким образом, рассмотренные ранее подходы к классификации можно дополнить новым подходом, учитывающим современные парадигмы программирования.

Множество современных подходов к программированию определяют не характеристики языков программирования, а способы проектирования и организации программ и программных комплексов. Таким образом, логично выделить следующие парадигмы программирования верхнего уровня:

- парадигмы программирования уровня архитектуры программного обеспечения;
- парадигмы программирования уровня языка программирования.

Парадигмы программирования уровня архитектуры программного обеспечения можно охарактеризовать как парадигмы, определяющие способ построения компьютерной программы. Реализовать данный способ построения можно посредством различных технологий, базирующихся на парадигмах программирования уровня языка программирования.

Парадигмы программирования уровня языка программирования можно разделить на две большие группы [6]:

- *императивную* (от англ. imperative – повеление; приказ), называемую также *директивной* (directive), к которой относят автоматное, структурное и объектно-ориентированное программирование;
- *декларативную* (declarative), к которой относят функциональное и логическое программирование.

На рис. 1 приведена иерархия парадигм программирования верхнего уровня.

В состав парадигм программирования уровня языка программирования дополнительно к уже двум известным парадигмам входят прочие парадигмы программирования. К *прочим парадигмам* программирования следует отнести парадигмы, которые вносят дополнительные идеи и подходы, сочетающиеся с видами программирования из обеих предыдущих парадигм, но не входящие ни в одну из них. В качестве примеров таких парадигм могут служить следующие:

- метапрограммирование;
- визуальное программирование.

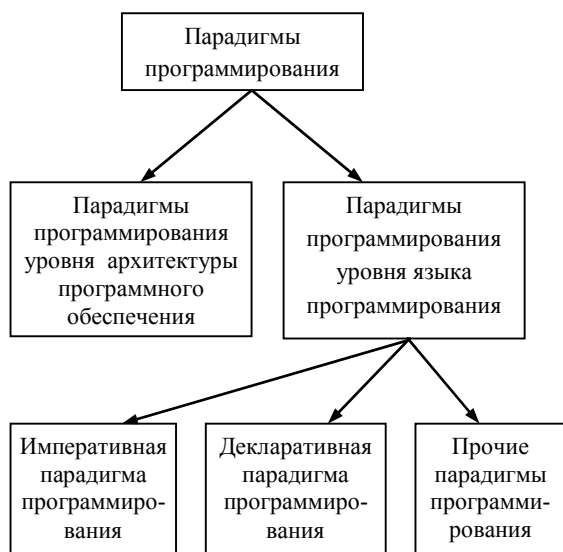


Рис. 1

В случае программирования в соответствии с императивной парадигмой программист пошагово описывает алгоритм решения задачи.

При использовании декларативной парадигмы программист описывает логику решения задачи. Из декларативной парадигмы программирования можно выделить виды программирования, представленные на рис. 2.

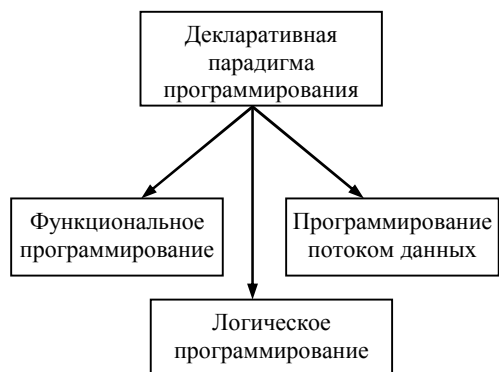


Рис. 2

Безусловно, существуют и другие типы декларативного программирования. Перечисленные типы решено выделить как наиболее значимые, по мнению авторов.

Как уже отмечалось, императивное программирование представляет собой парадигму программирования, в рамках которой программа описывает шаги, необходимые для решения задачи. При этом шаги изменяют состояние программы в процессе решения задачи. Из императивной пара-

дигмы программирования можно выделить виды программирования, представленные на рис. 3.

Подвидами структурного программирования являются:

- процедурное программирование;
- объектно-ориентированное программирование.

Из парадигм программирования уровня архитектуры программного обеспечения можно выделить в качестве основных виды программирования, представленные на рис. 4.

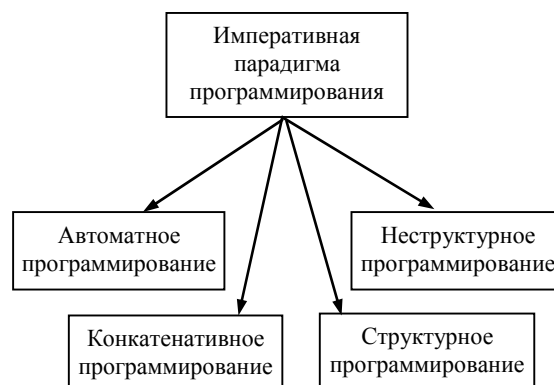


Рис. 3

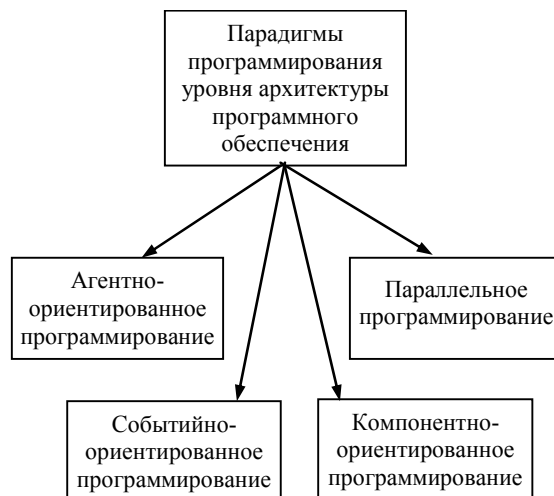


Рис. 4

Таким образом, в статье рассмотрен подход к классификации современных парадигм и подходов к программированию, формированию их иерархии. Представленный в статье подход дополняет существующие подходы и может как отдельно, так и совместно с существующими использоваться как студентами, так и специалистами для облегчения восприятия и формирования целостной картины современного состояния данной области технических знаний.

СПИСОК ЛИТЕРАТУРЫ

1. Парадигмы и языки в обучении информатике и программированию / Е. И. Большакова, Н. В. Баева, Н. В. Груздева, И. В. Горячая // Материалы Междунар. науч.-практ. конф. «Перспективные инновации в науке, образовании, производстве и транспорте», Одесса, 2012. Т. 4, вып. 2. С. 77–82.

2. Watt D., Findlay W. Programming language design concepts. Great Britain: John Wiley & Sons Ltd, 2004. 473 p.

3. Van Roy P. Programming Paradigms for Dummies: What Every Programmer Should Know. 2009. URL: <http://www.info.ucl.ac.be/~pvr/paradigms.html>.

4. Себеста Р. У. Основные концепции языков программирования. М.: Вильямс, 2001. 672 с.

5. Normark K. Overview of the four main programming paradigms / Aalborg University. 2013. URL: <http://people.cs.aau.dk/~normark/prog3-03/html/notes/theme-index.html>.

6. Пантелеев М. Г., Родионов С. В. Модели и средства построения экспертных систем: учеб. пособие. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2003. 68 с.

A. V. Malov

Motorola Solutions

S. V. Rodionov

Saint-Petersburg state electrotechnical university «LETI»

TO THE QUESTION OF THE CLASSIFICATION OF MODERN PROGRAMMING PARADIGMS

The basic paradigms and approaches to programming, the shortcomings of some of the existing classifications of programming paradigms are reviewed. It is emphasized that many modern approaches to programming are determined not in the characteristics of programming languages, but in ways of designing and organizing programs and software systems. The approach to classification of modern programming paradigms is proposed.

Programming paradigms, programming languages, the classification of programming paradigms

УДК 159.9 + 303.732

Е. В. Андреевский, Е. А. Бурков, Н. А. Назаренко, П. И. Падерно
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Формализация описания и оценки процессов профессионального отбора (подбора)

Рассмотрен пример использования обобщенного структурного метода (разработанного А. И. Губинским и В. Г. Евграфовым с использованием типовых элементов (типовых функциональных единиц)) для описания и оценки процессов профессионального отбора (подбора). Приведены способы оценки процессов профессионального отбора и профессионального подбора.

Профессиональный психологический отбор, обобщенный структурный метод, рабочая операция, контрольная операция, метрика

Общая постановка задач профессионального отбора и подбора. Задачи профессионального отбора (профотбора) и профессионального подбора (профподбора) довольно схожи и имеют одинаковую структуру, описанную в [1]. Профотбор и профподбор отличаются только конечной целью:

первый проводится с целью отбора субъекта на конкретную должность (специальность), а второй – с целью подбора подходящей специальности (должности) для субъекта. Исходя из этого, при формализации описания и оценки данные процессы можно рассматривать как аналоги друг друга.