

## Тенденции развития современных языков программирования высокого уровня

**Г.Н. Исаева**, к.т.н., доцент кафедры Информационных технологий  
и управляющих систем,

**Н.Н. Теодорович**, к.т.н., доцент кафедры Информационных технологий  
и управляющих систем,

**Ю.Ю. Сидоров**, аспирант кафедры Информационных технологий  
и управляющих систем,

Государственное бюджетное образовательное учреждение  
высшего образования Московской области

«Технологический университет», г. Королев, Московская область

*В статье рассмотрены основные направления развития современных языков программирования высокого уровня (ЯП ВУ) и их эволюционное развитие. Приведена статистика по ЯП ВУ, являющимися ключевыми в рамках различных парадигм программирования, опирающаяся на данные современных электронных ресурсов. Показана тенденция к стиранию граней между строгой классификацией ЯП ВУ по принадлежности к различным вычислительным моделям, лежащим в основе разработки языка. Показано, что основная причина популярности скриптовых языков и языков реактивного программирования связана с масштабностью решаемых экономических задач, массовым охватом всех сфер экономики сетевыми структурами, а также желанием разработчиков иметь простой, понятный и универсальный инструмент разработки программного обеспечения (ПО).*

Парадигма программирования, тренд, вычислительная модель.

## Tendencies of development of languages of programming of the high level

**G.N. Isaeva**, Candidate of Technical Sciences, Associate Professor of the Department of Information Technologies and Control System,

**N.N. Teodorovich**, Associate Professor of the Department of Information Technologies and Control System Candidate of Technical Sciences, Assistant professor,

**Y.Y. Sidorov**, Graduate Student of the Department of Information Technologies and Control System,

State Educational Institution of Higher Education

Moscow Region «University of technology», Korolev, Moscow region

*The main directions of the development of modern high-level programming languages (HL PL) and their evolutionary development are considered in the article. The statistics are presented for the HL PL, which are key in the framework of various programming paradigms, based on the data of modern electronic resources. A tendency is shown to obliterate the boundaries between the strict classification of the HL PL, as belonging to different computational models underlying the development of the language. It is shown that the main reason for the popularity of scripting languages and languages of reactive programming is related to the scale of solved economic problems, the mass coverage of all spheres of the economy by network structures, and also the desire of developers to have a simple, understandable and universal software development tool.*

Programming paradigm, trend, computational model.

Языки программирования высокого уровня в эпоху решения любых повседневных задач с помощью средств вычислительной техники (ВТ) имеют важное значение, они являются инструментом многоцелевого назначения с большими функциональными возможностями. На постоянные эволюционные изменения в семействе ЯП ВУ, с одной стороны, оказывает прогресс аппаратного и системного программного обеспечения, с другой стороны, быстро меняющиеся потребности объективного мира. Язык программирования является своего рода интерфейсом между пользователями – прикладными программистами – и средами разработки, находящимися под жестким влиянием вычислительной системы.

С момента зарождения ЯП ВУ выделяют четыре больших направления в классификации традиционных языков программирования по критерию вычислительных моделей, лежащих в основе языка. Это парадигмы: императивного, объектно-ориентированного (ООП), функционального и логического программирования [1, С.111-204].

Языки программирования высокого уровня, относящиеся к указанным направлениям, с течением времени менялись не только в рамках модели, но изменения их носили сближающий характер относительно выделенных парадигм и функциональных возможностей языка в рамках парадигмы. Такая тенденция определилась, прежде всего, желанием разработчиков сделать «некий» универсальный язык своего времени, лучший предшественника в данном классе и ориентированный на широкий пласт современных многоцелевых задач. Язык программирования XXI века должен быть с «изящным» синтаксисом, содержать элементы функционального программирования, быть ориентированным на объектный подход и легко позволять встраиваться программному коду (созданному на нём) на клиентские станции или серверы – поддерживать распределённую обработку данных. Следуя данным тенденциям, было разработано несколько ярких ЯП ВУ: Scala, Elm, Swift, Rust, Dart. Если к традиционным языкам программирования высокого уровня, таким как структурные: Си, Паскаль, Бейсик, как язык гипертекстовой разметки HTML и язык сценариев JavaScript, как объектно-ориентированным Visual Basic, Object Pascal, C++ пользователи и разработчики приложений уже привыкли, то новые языки программирования, которым нет и десятка лет, не всем известны и пока не очень тиражируются, в частности, в высшей школе.

Поэтому, данная статья, опираясь на информационные ресурсы электронных источников и историческую хронологию новых ЯП ВУ, показывает, как изменились требования к инструменту написания кода, какое влияние оказывает на эволюцию языков прикладного программирования характер стремительно меняющейся экономической картины мира. Остановимся на вышеупомянутых ЯП ВУ: Elm, Swift, Scala, Dart, чьё будущее, в плане максимальной востребованности, как инструмента прикладного программирования, ещё впереди.

Например, функциональный язык программирования Elm, представленный американским программистом Эваном Чаплицким в 2012 году в ходе работы над своим дипломным проектом в Гарварде, обладает ключевой особенностью. Системы программирования, поддерживающие этот язык, содержат отладчик реального времени, с помощью которого легко можно изменить программный код в реальном режиме времени и проследить реакцию на внесённые изменения по выбранным параметрам. Подобные действия по изменению параметров записываются, могут быть просмотрены и проанализированы, а также тщательно и многократно оттестированы и проверены.

Этот современный язык программирования относится к парадигме функционального программирования, с поддержкой принципов функционального реактивного программирования (ФРП). Парадигма ФРП ориентирована на параллельную обработку данных, обеспечивает чёткую работу с динамическими потоками данных и позволяет создавать высоко-интерактивные приложения. Концепция, положенная в основу ФРП, была предложена и сформирована в конце прошлого века для упрощения технологии создания пользовательских интерфейсов, анимации, динамических систем. Поэтому программы, написанные на этом языке программирования, легко конвертируются в JavaScript – код, что немало важно для современных условий работы с браузерами всех категорий пользователей. Кроме того, ЯП ВУ Elm удобен и прост в использовании: небольшой по объёму программный код обеспечивает разработчику приложения возможность работать с мышью для воспроизведения любой (даже сложной) фигуры в приложениях или в браузере. Тем не менее, язык изначально – функциональный, о чём говорят его родственные связи с семейством языка Haskell. На рисунке 1 показаны взаимовлияния традиционных языков программирования и новых, ровесников двадцать первого столетия.

Elm, не смотря на ярко выраженную принадлежность к функциональной парадигме, может быть использован для создания различного уровня сложности веб-приложений, практически всегда имеется возможность встроить написанный на этом языке программный код в блок тегов на странице сайта, так как Elm-программа без труда компилируется в коды HTML, CSS JavaScript.



Рисунок 1 – Связи ЯП ВУ

Проведенный анализ популярных языков программирования данной парадигмы, показал, что функциональное программирование и его основные идеи получили в настоящее время новый виток развития. Так, необходимо выделить новый язык с интересными связями, и, являющийся представителем этой вычислительной модели – язык Swift от компании Apple. Данный язык служит отличным полигоном для создания интерактивных приложений для мобильных устройств и в этом отчасти совпадает с предназначением ЯП ВУ Elm [2, С. 166,3, С. 139].

Если рассмотреть генеалогические связи этого языка (рис. 1), то становится понятно, почему Swift такой яркий и его синтаксис вполне лёгкий для освоения современными прикладными программистами. Во-первых, на его появление оказали влияние Си-подобные языки программирования, которые на протяжении десятков лет широко востребованы и освоены разработчиками ПО: это Java, Ruby, C++, Objectiv-C. В таблице 1 указана востребованность у работодателей в нашей стране специалистов со знанием перечисленных языков программирования. И тенденция такова, что основные языки – проверенные временем PHP, Java, C++ – занимают первые строчки таблицы, но современные языки, ровесники века, не уступают им в процентах роста и набирают популярность. Так, если брать цифры 2015 года за 100%-ю отметку, то прирост Swift составил 100%, а стоящий второй по числу заявок Java, прибавил 38% за год [8].

Таблица 1 – Вакансии мест для программистов

N n/n	ЯП ВУ	Временной период	
		2015	2016
1.	PHP	5586	9707
2.	Java	6219	8581
3.	C++	3252	4135
4.	C#	2853	4093
5.	JavaScript	2307	3657
6.	Python	3791	2581
7.	SQL	1449	1955
8.	Oracle	1303	1742
9.	C	917	1338
10.	Ruby	993	1003
11.	Scala	456	362
12.	Perl	165	223
13.	Go	72	188
14.	Objective-C	89	81
15.	Swift	39	78

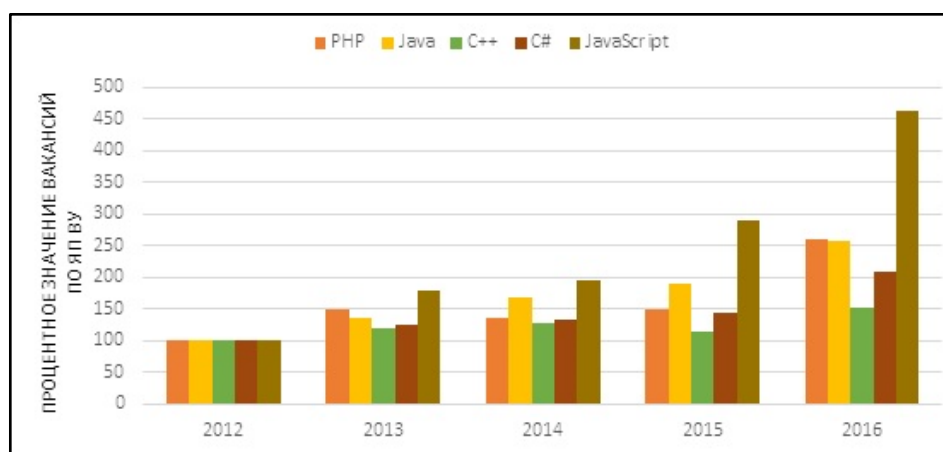
Во-вторых, масштабность решаемых экономических задач, массовый охват всех сфер экономики сетевыми структурами диктует строгие требования к создаваемым приложениям: быстрота, интерактивность, динамичность, инкапсуляция, масштабируемость. Все эти качества присущи тем языкам, о которых упоминалось выше. Анализ динамики вакансий специалистов-программистов за 2012-2016гг в нашей стране (где 2012 год взят за базовую отметку в 100%)[9] (Таблица 2) и построенная по этим данным диаграмма (Рис. 2) показывает наибольшую востребованность программистов, владеющих языками JavaScript, Java, PHP. Значит и

современные языки - потомки этих апробированных временем языков, должны получить со временем признание у массового пользователя – разработчика различного рода приложений. В высшей школе, например, в образовательном стандарте для информационно-технологических направлений обучения за последние пять лет появился даже курс, связанный с веб-технологиями, основу которого составляют знания по применению ЯП ВУ JavaScript, PHP.

**Таблица 2 – Вакансии программистов в России**

Временной интервал	Языки программирования ВУ				
	PHP	Java	C++	C#	JavaScript
2012	100	100	100	100	100
2013	149	135	120	124	178
2014	136	169	127	133	194
2015	149	189	115	144	290
2016	260	257	151	208	462

ЯП ВУ Swift как раз и сочетает в себе понятность Си-подобных языков, опирается на мощную динамическую объектную модель Objective-C, не уступает по быстродействию C++ и Java, обладает доступностью по отношению к интерфейсам прикладного программирования для различных операционных систем [4, С.237-240,6].



**Рисунок 2 – Динамика вакансий программистов**

С помощью этого нового языка программирования пользователи быстрее проверяют свои концепты и в целом быстрее создают свои приложения, так как в реализацию данного языка входит отладчик реального времени, который позволяет экспериментировать с кодом, наблюдать результаты изменений и вычислений мгновенно, без компиляции и запуска приложения. Кроме того, тенденция настоящего времени – это многопарадигмальность (мультипарадигмальность) языков программирования высокого уровня, и Swift – не исключение. Этот язык обладает направленностью и характерными особенностями объектно-ориентированного, функционального, императивного программирования.

Другой представитель мультипарадигменной направленности – язык программирования начала века – Scala, хотя по простоты более десятка лет с момента выхода его в свет, он занимает в рейтинге популярности не самое видное место (таблица 1), но с точки зрения вклада в другие языки, такие как Swift, заслуживает внимания. Scala разрабатывался для простого и быстрого программирования приложений, поэтому в нём органично сочетаются возможности объектно-ориентированного и функционального программирования. На формирование этого языка оказали влияние такие элегантные ЯП ВУ, как Lisp, Haskell, Java, C#, он считается языком хорошей поддержки для компонентного программного обеспечения [7].

Влияние Java на этот язык велико, на Scala можно писать программы лишь немного корректируя синтаксис Java, это тоже является плюсом, так как Java появился на десять лет раньше и следует проверенным и надёжным принципами «чистого» объектно-ориентированного программирования. Например, можно использовать Scala-коллекции из библиотек и неявные преобразования из данной коллекции в Java – коллекции, причём как в одну сторону, так и в другую сторону. Даже инструмент написания встроенных аннотаций совпадает у этих двух языков [6].

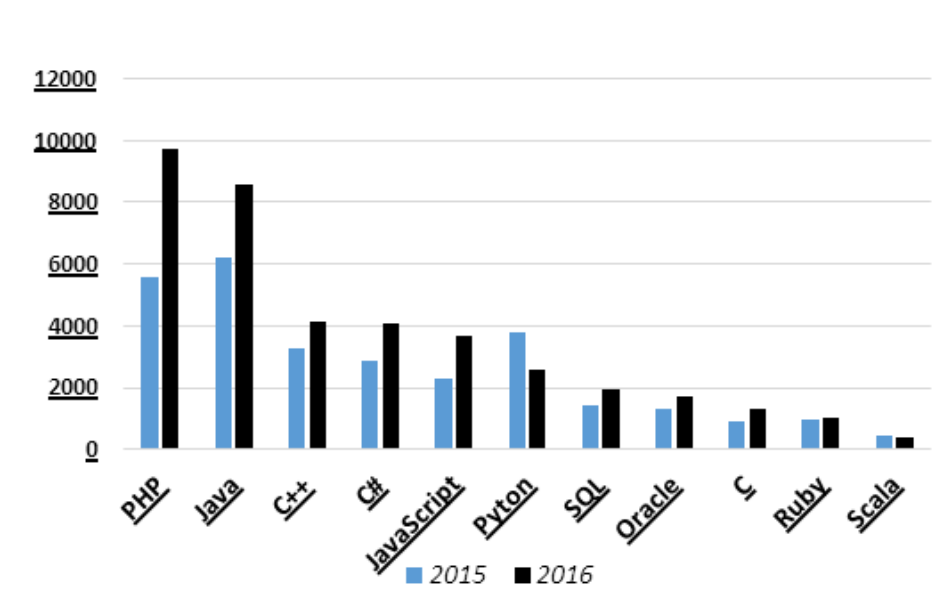


Рисунок 3 – Динамика числа заявок на программистов по ЯП ВУ

Конечно, такие языки, как Scala, молоды и не набрали той популярности, что присуща традиционным ЯП ВУ (что можно проследить на диаграмме, рис. 3), но время не стоит на месте и можно смело утверждать, что будущее этих языков не за горами.

Характерная черта современного прикладного программирования – веб-направленность. Об этом свидетельствует позиция языка PHP на диаграмме востребованности программистов за 2015-2016 годы (Рис. 3). Используя статистические данные, упомянутые выше (таблица 1), можно сказать, на примере PHP, языки объектно-ориентированной парадигмы и скриптовые языки для написания веб-предложений будут востребованы ближайшие годы (рис 4-5).

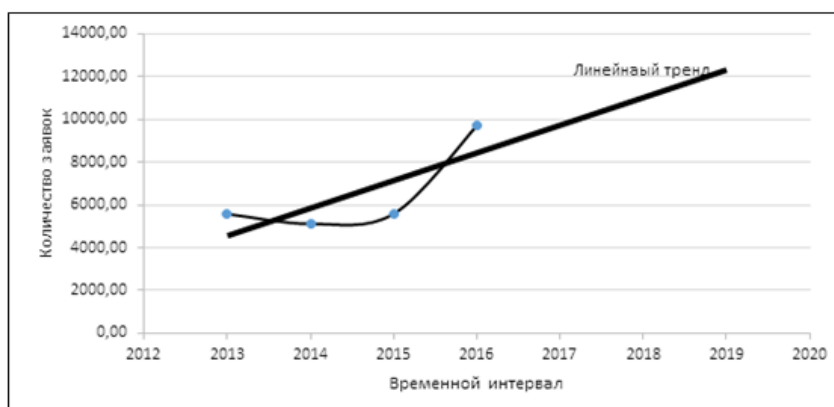
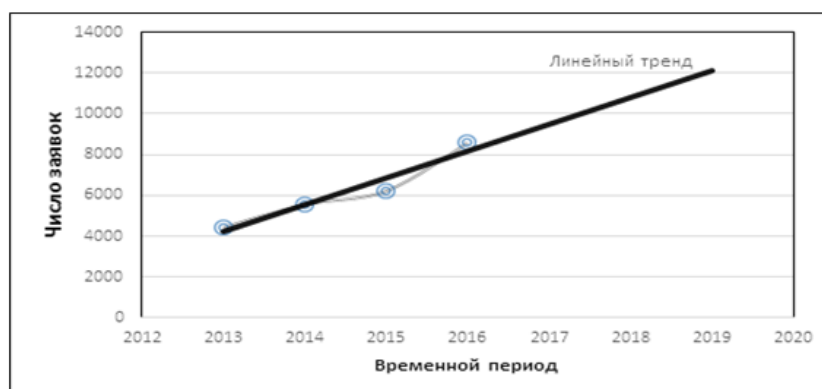


Рисунок 4 – Прогноз востребованности PHP-программистов



**Рисунок 5 – Прогноз востребованности Java-программистов**

Проведенные расчёты в определении прогнозных тенденций, подтверждает и разработка совершенно нового языка веб-программирования Dart, как альтернативы уже широко известному языку программирования JavaScript.

ЯП ВУ Dart – совершенно новый язык, уже относящийся к языкам, появившимся во втором десятилетии двадцать первого века. Так или иначе, но при его создании разработчики попытались избавиться от недостатков, присущих его историческим «родственникам» (рис. 1) [5]. Этот язык относится к парадигме объектно-ориентированного программирования, но это не мешает ему быть хорошим инструментом при написании веб-приложений.

Он позиционируется, как мультипарадигмальный язык программирования, и это уже новое направление в развитии современных ЯП ВУ. Кроме того, Си-подобный синтаксис упрощает его освоение.

Если оценивать эффективность кода, то одним из основных критериев качества разрабатываемых программ, считается быстродействие, что не всегда присуще клиентским программам, написанным на JavaScript. За то небольшое время с момента представления языка (2011 год), программы, созданные на Dart – высокопроизводительны и успешно работают на вычислительных устройствах широкого класса: от смартфонов до серверов. Это является подтверждением качества предлагаемого языка, который в настоящее время проходит апробацию у разработчиков.

По безопасности – также важнейшему критерию современного программирования, ЯП ВУ Dart надёжнее Go и JavaScript. Безопасность веб-приложений, написанных на Dart, уже проверена и одобрена разработчиками. Имеются удобства у пользователей этого языка и в плане поддержки, отладки и модификации кода. Кроме того, 4 июля 2014 вышел первый стандарт языка Dart [5]. Эти аргументы говорят в пользу этого нового ЯП ВУ, что в ближайшее время он должен получить признание и у «массового программиста».

Таким образом, мир прикладного программного обеспечения не стоит на месте и появление новых ЯП ВУ тому подтверждение. В настоящем веке наиболее популярными становятся скриптовые языки и языки реактивного программирования, как правило, относящиеся к мультипарадигмальной концепции и опирающиеся на Си-подобный синтаксис. Данные тенденции обусловлены новыми разноплановыми экономическими задачами, сетевыми всемирными структурами, желанием программистов иметь простой, понятный и универсальный инструмент разработки современных приложений.

#### *Литература*

1. Голицына О.Л., Попов И.И. Программирование на языках высокого уровня // М.: ФОРУМ. 2011. 496 с.
2. Исаева Г.Н, Пахомов Д.А., Возможности современных языков программирования высокого уровня. В сборнике: СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ Сборник трудов по материалам межвузовской научно-технической конференции. МГОТУ под науч. ред. В.М. Артюшенко. 2015. С. 163-167.
3. Исаева Г.Н, Клешнев И.Б., Функциональное программирование: эффективный инструмент решения современных задач. Современные информационные технологии / сборник трудов по материалам II-й международной научно-технической конференции 14 сентября 2016г. г. Королёв, «МГОТУ» Под общей науч. ред. док. техн. наук проф В.М. Артюшенко // М:

- Издательство «Научный консультант». 2016. С. 136-142.
4. Фридман А.Л. Построение Интернет-приложений на языке Java // М.: Горячая линия–Телеком. 2012. 336 с.
  5. Сайт о языке программирования Dart – руководства, инструкции, новости, статьи и многое другое: [Электронный ресурс]. // dartdot.ru. URL: <http://dartdot.ru/> (дата обращения 15.10.2017).
  6. Язык программирования Swift. Русская версия: [Электронный ресурс]. // Хабрахабр. URL: <https://habrahabr.ru/post/225841/>(дата обращения 15.10.2017).
  7. Scala. Язык программирования [Электронный ресурс]. URL: // <http://progopedia.ru/language/scala/> (дата обращения 24.01.2017).
  8. Very Long Term History [Электронный ресурс]. URL: // <http://www.tiobe.com/tiobe-index/> (дата обращения 24.01.2017).
  9. Самые востребованные языки программирования 2016 [Электронный ресурс]. URL: // <https://habrahabr.ru/company/hh/blog/318450/>(<http://dartdot.ru/> (дата обращения 24.01.2017)).