

## Инновационная роль стандартов программирования в развитии методологии DevOps

Г. М. Коротенко\*, Л. М. Коротенко

Государственное высшее учебное заведение «Национальный горный университет», г. Днепр, Украина

\*Corresponding author. E-mail: gkorotenko@gmail.com

Paper received 09.12.17; Revised 14.12.17; Accepted for publication 15.12.17.

<https://doi.org/10.31174/SEND-PP2018-153VI63-07>

**Аннотация.** Статья посвящена аспектам реализации образовательной составляющей в системе обучения программированию. Выделены новые компоненты, отвечающие современному состоянию ИТ-отрасли, учет которых необходим при подготовке студентов. Предложена система методов и приемов, способствующих формированию и реализации будущих специалистов. Рассмотрены вопросы реализации методических материалов для процесса обучения студентов младших курсов Университетов в виде стандартов по программированию.

**Ключевые слова:** подготовка студентов, ИТ-отрасль, Третья платформа, DevOps, стандарты программирования.

В настоящее время на плечи Университетов ложится все возрастающая нагрузка по подготовке студентов, отвечающих современному состоянию ИТ-отрасли. Сейчас, одним из основных факторов, существенно влияющих на процесс повышения качества образования, является ведущий технологический тренд, известный под названием «Третья платформа» [1, 2], определяемый как конвергенция и взаимное усиление четырех взаимозависимых тенденций: социального взаимодействия, мобильности, облачных технологий и информации.

Еще в 2013 году одна из ведущих в мире фирм по исследованию трендов (т.н. форсайтов<sup>1</sup>) в ИТ-индустрии IDC ввела в оборот термин «третья платформа» (*third platform*). Данное словообразование, по мысли авторов и ряда других специалистов, должно было описывать конгломерат движущих сил прогресса в развитии современного цифрового бизнеса, опирающегося на: а) мобильные устройства, б) облачные сервисы, в) социальные сети и г) аналитические процедуры, применяемых к «большим данным» [3, 4]. Дальнейшее развитие событий показало устойчивое развитие данного направления, в результате чего появились дополнительные названия возникшего на стыке течений феномена:

– SMAC (social, mobile, analytics and cloud) [5]

– Nexus of Forces<sup>2</sup>: Social, Mobile, Cloud and Information [6].

По последним данным [7], в мире насчитывается 4,7 млрд мобильных пользователей, 3 млрд подключенных к Интернету, 1,7 млрд пользователей только социальной сети Facebook™ и 1 млрд пользователей мобильного кроссплатформенного мессенджера WhatsApp™, не учитывая многие другие доступные практически каждому владельцу мобильных устройств разнообразных программ и ресурсов. По данным appFigures [8], количество приложений в Google Play и App Store в 2016 году достигло 1,5 млн. Сейчас выпуском приложений для Android, размещаемых в Google Play, занимаются около 388 тыс. разра-

ботчиков — против 282 тыс. у Apple [9].

Третья платформа характеризуется стремительно растущим количеством постоянно подключенных к Интернету мобильных устройств в сочетании с широким использованием социальных сетей и развитой облачной инфраструктуры, применяемой для решения комплексных аналитических задач с помощью технологий больших данных [11]. Под терминами «Большие данные», «Big Data» или просто «биг дата» скрывается огромный объем перманентно накапливающейся разнородной информации, настолько значительный, что его обработка стандартными программными и аппаратными средствами представляется крайне сложной. Проблема хранения и обработки взаимопроницающих структурированных и неструктурированных разноформатных компонентов данных существенно создает все новые и новые вызовы. Данная ситуация усложняется расширением и усложнением комплекса стремительно расширяющих свои границы базовых технологий: интернет- и веб-технологий, мультимедийных, сетевых, беспроводных, облачных, мобильных, телекоммуникационных, офисных, корпоративных, защиты информации и данных, геоинформационных, дистанционного зондирования Земли, статистических, управления, технологий искусственного интеллекта, технологий программирования и др.

Постоянно растущая сложность разрабатываемых систем и увеличение числа и размера программных компонентов породили методологию DevOps (объединение сокращений терминов «разработка» (development) и «операции» (operations)) [12]. Впервые данный термин ввели Эндрю Шафер и Патрик Дебоис на конференции Agile в Торонто в 2008 году в процессе обсуждения проблем в рубрике «Agile Infrastructure». В целом — это культура и практика разработки программного обеспечения, целью которой является унификация разработки программного обеспечения (Dev) и его эксплуатации (Ops). Главной характеристикой движения DevOps является решительная поддержка автоматизации и мониторинга на всех этапах разработки программного обеспечения: от интеграции, тестирования, до развертывания и управления инфраструктурой. DevOps нацелен на более короткие циклы разработки, повышенную частоту развертывания, более надежные выпуски, в тесном взаимодействии с бизнес-целями организаций.

<sup>1</sup>Форсайт, от английского Foresight — «взгляд в будущее, видение будущего» — наиболее эффективный инструмент формирования приоритетов в сфере науки и технологий, экономики, государства и общества. По результатам Форсайт-проектов создаются дорожные карты, которые позволяют решить вопрос формирования будущего.

<sup>2</sup> дословно, Сплетение сил.

По определению, предложенному Басом, Вебером и Чжу, является: DevOps – это набор практик, предназначенных для сокращения времени между внесением изменений в систему и изменением в нормальное производство при обеспечении высокого качества.

Феномен DevOps объединил под своими знамена-

ми невероятный конгломерат существующих и постоянно создаваемых организационных структур, платформ, сервисов, инструментов и т.д.

В целом, данный процесс можно представить следующим образом (рис. 1) [13].

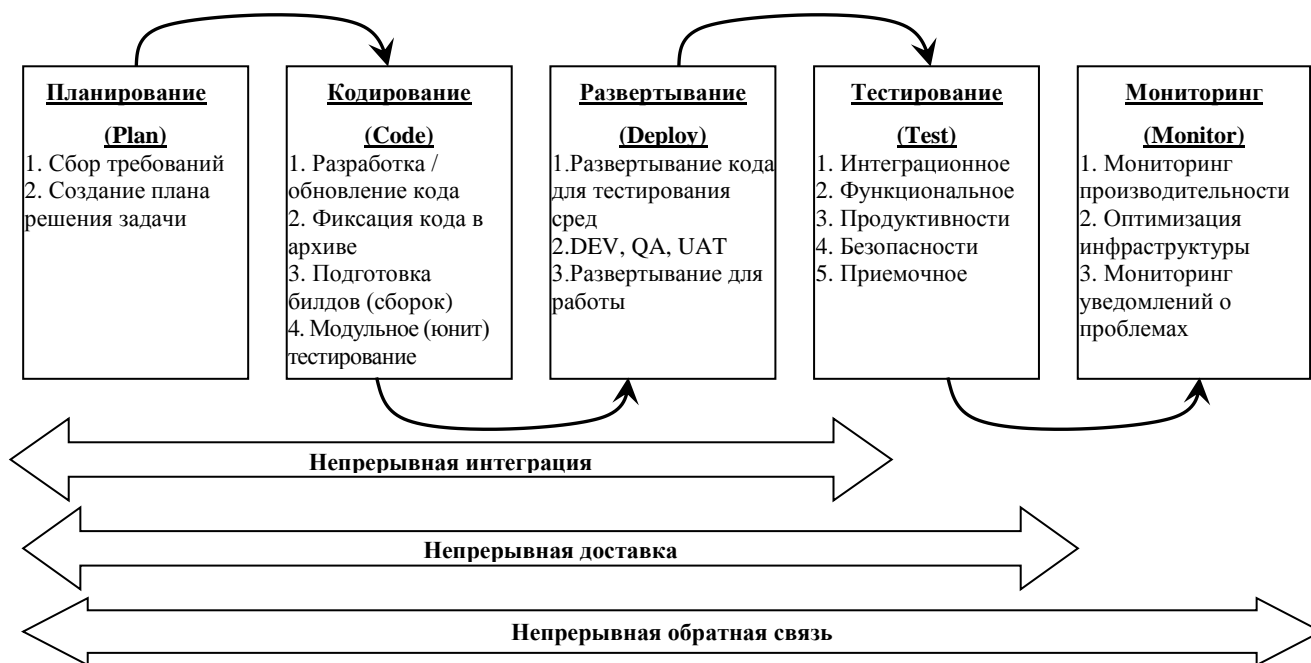


Рис. 1. Структура взаимодействующих процессов в методологии DevOps

Как отмечают многие авторы, несмотря на постоянное развитие подходов, практик, инструментов, следует констатировать, что основу инфраструктуры DevOps составляет собственно код разрабатываемых, совершенствуемых и накапливаемых программных единиц [14, 15].

Поэтому продолжают работы по стандартизации разнообразных характеристик программного кода. Так, в настоящий момент, серия стандартов разрабатывается под эгидой организации Consortium for IT Software Quality (CISQ) [16]. Консорциум по качеству программного обеспечения для ИТ (CISQ™) является лидером в области ИТ-индустрии, в состав которого входят руководители ИТ-подразделений Global 2000, системные интеграторы, поставщики услуг сторонних поставщиков и поставщики программных технологий, приверженные внедрению вычислимых стандартов показателей для измерения качества и размера программного обеспечения. CISQ – это нейтральный открытый форум, на котором клиенты и поставщики программного обеспечения для ИТ-приложений могут разрабатывать общесистемную повестку дня для повышения качества ИТ-приложений для снижения затрат и риска. Проводимые работы рассредоточены по следующим направлениям:

1. Software Sizing (Размер программного обеспечения).
2. Code Quality (Качество кода).
3. Technical Debt (Технический долг).
4. Additional Measures (Дополнительные меры).

В частности, стандартные показатели качества CISQ (или «Показатели CISQ») для обеспечения безопасности, надежности, эффективности и удобства обслуживания – это стандарты, используемые для оценки соответствия восьмидесяти шести (86-ти) хорошо зарекомендовавшим себя правилам разработки программного обеспечения для обеспечения безопасного, надежного, эффективного и простого в обслуживании программного обеспечения. Эти меры предназначены для автоматизации с помощью статического анализа исходного кода, чтобы выявить критические недостатки в программном обеспечении, которые достаточно серьезны, чтобы их можно было устранить. В сочетании с измерением размера может быть получен показатель плотности для каждой характеристики качества.

Пороги могут быть установлены для каждой характеристики для управления качеством разрабатываемого или поставляемого третьим лицом программного обеспечения. Показатели CISQ служат основой для нового стандарта автоматизированного технического долга, который оценивает будущую стоимость корректирующего обслуживания.

Рассматривая любой программный компонент как объект работы различных взаимодействующих групп участников, в первую очередь следует отметить, что непрерываемо существенный вклад в его создание вносит, в первую очередь, разработчики (программисты, кодировщики) (рис. 2).



Рис. 2. Лица, формирующие основные показатели (характеристики) качества программ

Именно поэтому, стандартизация написания исходных текстов (source code) программ, по мнению авторов, должна существенно упростить все дальнейшие этапы взаимодействия с ними. Данные работы предлагается вести уже в Университетах, на базе преподавания курсов по изучению языков программирования. В частности, в ГВУЗ «Национальный горный университет», на кафедре программного обеспечения компьютерных систем разработан Стандарт кафедры по преподаванию языка С++ для процедурного программирования [17], который основан на документах: «Гайд по оформлению кода на С++ от Стэнфордского университета» [18] и «Google C++ Style Guide» [19]. Из стандарта Стэнфордского университета авторами был использован т.н. «ВерблюжийРегистр» (CamelCase) для описания переменных. За основу были приняты именно основные положения стандарта Google. Естественно, что были исключены моменты, связанные с ООП. Опираясь на положения структурной теоремы Бома и Джакопини, основное внимание авторы уделили базовым структурам языка С++: СЛЕДОВАНИЕ, РАЗВИЛКА и ЦИКЛ. При этом учитывалось, что в языке С++ конструкция СЛЕДОВАНИЕ не относится к группе управляющих структур, к которой относятся операторы if, if/else, switch, for, while, do/while. Поэтому, комментирование функционального назначения данных структур выделено в отдельные подразделы. Предлагаемый подход был применен с той целью, что дидактической «клизминкой» для студента является тот факт, что при комментировании введенной структуры, он должен задуматься о ее назначении. Также, особое внимание в процессе обучения уделяется освоению студентами элементов модульного программирования. В упомянутом выше «Стандарте» указывается, что необходимо в вызывающей функции комментировать вызовы

функций, но, главное – писать короткие функции, придерживаться «Стандарта» при выборе имен функций, выдерживать порядок их параметров, комментировать назначение функций и их параметров.

Как отмечал Джек Ривз в журнале «С++ Journal» осенью 1992 года: «Программирование – это проектирование ПО». Поэтому авторы считают, что подготовку будущих профессионалов-программистов необходимо начинать с первого курса Университета. Используя как кубики базовые структуры языка С++ вместе со Стандартом, будущие «профи» в области программирования смогут найти свое место в различных организациях, даже не догадывающихся о внедрении в их деятельность современного ПО. Следующим этапом планируется разработка Стандарта для объектно-ориентированного программирования.

Выводы. На основании анализа современного состояния ИТ-отрасли авторы пришли к выводу, что в целом наблюдается отсутствие начальной фазы – внедрения в обучение на младших курсах Университетов стандартов программирования для дальнейшего использования в процессе разработки программ. Сами стандарты для разных языков программирования должны базироваться на нисходящем проектировании программ методом пошаговой детализации и структурном программировании с использованием основных базовых структур языка, чтобы студенты сосредоточились на проектировании и конструировании программ, а не на особенностях языка программирования. Очевидно, что лучшим вариантом было бы сведение воедино лучших практик для качественной подготовки будущих профессионалов. Для обеспечения процесса дидактической преемственности предлагается разделить стандарты по программированию на Университетские и Профессиональные.

#### ЛИТЕРАТУРА

1. IDC Predictions 2013: Competing on the 3rd Platform. WEB-сайт [Э. ресурс] / Режим доступа: URL: <http://www.idc.com/research/Predictions13/downloadable/238044.pdf>
2. 'Third platform' shift triggers enterprise software evolution | ZDNet. ZDNet. Retrieved 2016-01-01. WEB-сайт [Э. ресурс] / Режим доступа: URL: <http://www.zdnet.com/article/third-platform-shift-triggers-enterprise-software-evolution>
3. EMC World 2013: EMC hails rise of third platform apps. WEB-сайт [Э. ресурс] / Способ доступа: URL: <http://www.itpro.co.uk/apps/19763/emc-world-2013-emc-hails-rise-third-platform-apps>.
4. Third platform. WEB-сайт [Э. ресурс] / Способ доступа: URL: [https://en.wikipedia.org/wiki/Third\\_platform](https://en.wikipedia.org/wiki/Third_platform).
5. Talking SMAC: Revisiting social, mobile, analytics and cloud. WEB-сайт [Э. ресурс] / Способ доступа: URL: <http://www.cio.com/article/2918194/cloud-computing/talking-smac-revisiting-social-mobile-analytics-and-cloud.html>.
6. The Nexus of Forces: Social, Mobile, Cloud and Information. WEB-сайт [Э. ресурс] / Способ доступа: URL: <http://www.gartner.com/technology/research/nexus-of-forces/>
7. Praveen Agrawal. Digital in Utilities – Hype or Hope!!! WEB-сайт [Э. ресурс] / Способ доступа: URL: <http://eaminsights.com/2016/09/07/digital-in-utilities-hype-or-hope/>
8. App Store. WEB-сайт [Э. ресурс] / Способ доступа: URL: [https://ru.wikipedia.org/wiki/App\\_Store](https://ru.wikipedia.org/wiki/App_Store)
9. Android победил iPhone по числу приложений. WEB-сайт [Э. ресурс] / Способ доступа: URL: [http://www.cnews.ru/news/top/android\\_pobedil\\_iphone\\_po\\_chislu\\_prilozheniy](http://www.cnews.ru/news/top/android_pobedil_iphone_po_chislu_prilozheniy)
10. Feng-Sheng. Field Study of Patent Strategies from Patent Map on Big Data: An Empirical Case of Big Data Application Platform in Taiwan. WEB-сайт [Э. ресурс] / Режим доступа: URL: [https://www.researchgate.net/publication/271838610\\_Field\\_Study\\_of\\_Patent\\_Strategies\\_from\\_Patent\\_Map\\_on\\_Big\\_Data\\_An\\_Empirical\\_Case\\_of\\_Big\\_Data\\_Application\\_Platform\\_in\\_Taiwan](https://www.researchgate.net/publication/271838610_Field_Study_of_Patent_Strategies_from_Patent_Map_on_Big_Data_An_Empirical_Case_of_Big_Data_Application_Platform_in_Taiwan).
11. DevOps. WEB-сайт [Э. ресурс] / Режим доступа: URL: <https://en.wikipedia.org/wiki/DevOps>.

12. Roger LeBlanc. DevOps and Middleware, common scenarios. WEB-сайт [Э. ресурс] / Режим доступа: URL: [https://www.ibm.com/developerworks/community/blogs/c914709e-8097-4537-92ef-8982fc416138/entry/DevOps\\_and\\_Middleware\\_common\\_scenarios?lang=en](https://www.ibm.com/developerworks/community/blogs/c914709e-8097-4537-92ef-8982fc416138/entry/DevOps_and_Middleware_common_scenarios?lang=en)
13. Философия DevOps. Искусство управления ИТ / Дэвис Дженнифер, Дэниелс Кэтрин. – СПб.: Питер, 2017. – 416 с.,
14. DevOps for Dummies, 3<sup>rd</sup> IBM Limited Edition. / Sanjeev Sharma, Bernie Coyne. – John Wiley & Sons, Inc., 2017. – 62 p.
15. Consortium for IT Software Quality (CISQ). . WEB-сайт [Э. ресурс] / Режим доступа: URL: <http://it-cisq.org/>
16. Методические рекомендации по разработке программ по курсу «Алгоритмизация и программирование» на языке программирования C++ для бакалавров области знаний 12 «Информационные технологии» / Г.М. Коротенко, Л.М. Коротенко – Д.: ГБУЗ «Национальный горный университет», 2017. – 25 с. (в печати).
17. CS 106B: Programming Abstractions (C++) Summer 2015. WEB-сайт [Э. ресурс] / Режим доступа: URL: <http://stanford.edu/class/archive/cs/cs106b/cs106b.1158/styleguide.shtml>
18. Google C++ Style Guide. WEB-сайт [Э. ресурс] / Режим доступа: URL: <https://google.github.io/styleguide/cppguide.html>

#### REFERENCES

9. Android won the iPhone by the number of applications. [E. source]. – Mode of access: [http://www.cnews.ru/news/top/android\\_pobedil\\_iphone\\_po\\_chislu\\_prilozheniy](http://www.cnews.ru/news/top/android_pobedil_iphone_po_chislu_prilozheniy)
13. Effective DevOps. Building a Culture of Collaboration, Affinity, and Tooling at Scale / Jennifer Davis, Katherine Daniels. – SPb.: Piter, 2017. – 416 p.
16. Methodical recommendations for the development of programs on the course "Algorithmization and programming" in the programming language C ++ for bachelors of the field of knowledge 12 "Information technology" / G.M. Korotenko, L.M. Korotenko – D.: SHEI «National Mining University», 2017. – 25 с. (in the press).

#### The innovative role of programming standards in the development of the DevOps methodology

G. M. Korotenko, L. M. Korotenko

**Abstract.** The article is devoted to aspects of implementation of the educational component in the system of teaching programming. New components have been singled out that correspond to the current state of the IT industry, which are necessary for the preparation of students. A system of methods and techniques that contribute to the formation and implementation of future specialists is proposed. The issues of implementation of methodological materials for the process of teaching students of junior courses of Universities in the form of programming standards are considered.

**Keywords:** preparation of students, IT industry, Third platform, DevOps, programming standards.