

**«Балтийский государственный технический университет «ВОЕНМЕХ»
им. Д.Ф. Устинова»**

(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова)

Факультет	И	Информационные и управляющие системы
	шифр	наименование
Кафедра	И9	Систем управления и компьютерных технологий
	шифр	наименование

**ОТЧЁТ О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ
ПО ТЕМЕ**

**Исследование особенностей эволюции экосистем
программного обеспечения**

Выполнил студент группы И9М33

Крылов К.А.

Фамилия И.О.

РУКОВОДИТЕЛЬ

Гущин А.Н.

Фамилия И.О.

Подпись

Оценка _____

«_____» _____ 2018 г.

САНКТ-ПЕТЕРБУРГ
2018 г.

РЕФЕРАТ

Отчёт о НИР 16 с., 1 рис., 2 табл., 13 источников.

ЭКОСИСТЕМЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, ЭВОЛЮЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, ПОДДЕРЖКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.

Цель работы – выявление особенностей эволюции экосистем программного обеспечения, происходящих на этапе поддержки программного обеспечения.

В ходе выполнения работы был произведен обзор литературы и введены понятия, связанные с понятиями экосистем программного обеспечения, были приведены основные положения теории эволюции программного обеспечения, существующие на данный момент. Также, было произведено сравнение естественной эволюции с эволюцией программного обеспечения. В результате работы были описаны основные особенности эволюции экосистем программного обеспечения, которые позволят глубже взглянуть на этап поддержки программного обеспечения.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 ЭКОСИСТЕМЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	5
2 ЭВОЛЮЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	7
2.1 Естественная эволюция	7
2.2 Эволюция программного обеспечения и её связь с естественной эволюцией	9
2.3 Законы эволюции программного обеспечения	11
ЗАКЛЮЧЕНИЕ	15
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	16

ВВЕДЕНИЕ

Ввиду стремительного развития индустрии разработки программного обеспечения, а также усложнения протекающих в ней процессов, существует потребность моделях, способных предоставить разработчикам возможность принимать более эффективные решения на этапах проектирования, разработки, поддержки и анализа состояния программного обеспечения. Одной из таких моделей является экосистема программного обеспечения.

Обычно под этим термином подразумевается множество организаций, связанных между собой использованием общей технологической платформы, и взаимодействующих внутри общего рынка, для получения и использования программного обеспечения и сервисов [1]. Использование такой модели при разработке программного обеспечения принимает во внимание не только факторы, связанные с областью программной инженерии, но также и факторы, относящиеся к областям менеджмента и взаимодействия между различными организациями.

Вследствие нестабильности рынка, появления новых технологий, перемены потребностей пользователей и прочих факторов, некоторые экосистемы программного обеспечения перестают существовать, в то время как другие справляются с возникающими трудностями, и продолжают своё развитие на протяжении десятилетий. Это происходит вследствие того, что в процессе разработки программного обеспечения разработчики принимают решения, которые приводят к позитивным или негативным последствиям для программного обеспечения с точки зрения его положения в экосистеме. Процесс принятия влияющих на развитие программного обеспечения решений можно рассмотреть как эволюционный процесс развития системы.

В данной работе будет рассмотрен процесс эволюции экосистем программного обеспечения, а также произведён обзор исследований из областей, так или иначе, затрагивающих данный предмет.

1 ЭКОСИСТЕМЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В работах, посвящённых исследованию экосистем программного обеспечения на текущий момент нет общепринятого определения данного предмета. Отчасти, его поиск усложнён ввиду того, что используемое определение зависит от рассматриваемых аспектов экосистемы программного обеспечения, которые могут касаться как вопросов программной инженерии, так и вопросов менеджмента или взаимодействия её участников.

Так, например, в самой ранней работе [2] затрагивающей данный предмет, экосистема программного обеспечения определяется как «множество программных продуктов, в определённой степени участвующих в симбиотических отношениях». Из более поздних работ можно, например, выделить следующее определение – «экосистема программного обеспечения это множество организаций, связанных между собой использованием общей технологической платформы, и взаимодействующих внутри общего рынка, для получения и использования программного обеспечения и сервисов» [3].

В большинстве источников под экосистемами программного обеспечения принимаются крупные компании, раскрывшие свою технологическую платформу и предоставившие открытый доступ к ней. Таким образом, доступ к технологической платформе получают независимые разработчики, которые могут вносить непосредственный вклад в её развитие, добавляя в её состав новые компоненты и расширяя доступную функциональность.

Однако на данный предмет можно взглянуть иначе. Как отмечают в работе [4] – в настоящее время разработчики программного обеспечения больше не могут независимо создавать программные продукты, так как они, в той или иной степени, зависят от других разработчиков, используя предоставляемые ими сервисы, библиотеки, магазины приложений, технологические платформы и операционные системы. Таким образом, чтобы разработать программный продукт, разработчик должен устанавливать и

поддерживать внешние связи, участвуя, тем самым, в формировании экосистемы программного обеспечения. При использовании такого подхода экосистему программного обеспечения можно рассматривать как более полную модель для описания процессов, протекающих в организациях занимающихся разработкой программного обеспечения.

В процессе функционирования экосистемы программного обеспечения на её состояние могут негативно влиять факторы, которые могут быть как внешними, проявляющимися в таких видах, как нестабильность рынка, конкуренция, появление новых технологий, перемена потребностей пользователей, так и внутренними, такими как растущая сложность внесения изменений, архитектурные ограничения на введение необходимой функциональности, потеря основных участников. Некоторые экосистемы программного обеспечения перестают существовать, не справляясь с возникающими трудностями, в то время как другие, испытывая схожие трудности, продолжают своё развитие на протяжении десятилетий.

Для определения факторов, которые могут влиять на продолжительность жизни экосистем программного обеспечения, в данной работе будет рассмотрено направление исследования, занимающееся эволюцией программного обеспечения.

2 ЭВОЛЮЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Можно выделить два основных подхода к изменению систем: эволюционный и революционный. Обычно, эволюционные изменения происходят постепенно, в течение продолжительного времени, изменяя лишь некоторое подмножество её свойств. Революционные изменения, напротив, являются фундаментальными, охватывают большую часть свойств системы и происходят в краткие сроки. Также, важной особенностью является то, что результат эволюционных изменений является более устойчивым и предсказуемым, нежели результат революционных изменений.

Для получения более широкого взгляда на явление эволюции программного обеспечения, можно рассмотреть естественную эволюцию, как наиболее изученный процесс эволюционного изменения, и постараться найти возможные сходства и различия между ними.

2.1 Естественная эволюция

Биологическая эволюция это происходящий в череде поколений процесс приспособления биологических систем к условиям окружающей среды. Отличительным свойством биологической эволюции является адаптивность, позволяющая биологическим системам приспосабливаться под условия существования.

Для того чтобы приступить к описанию сути эволюционного процесса, следует ввести ряд ключевых определений.

Ген – последовательность ДНК или РНК, которая несёт информацию о строении молекулы, выполняющей какую-либо функцию. **Генотип** – совокупность генов одной особи. **Экспрессия гена** – это процесс, в ходе которого наследственная информация от гена преобразуется в функциональный продукт — РНК или белок. **Фенотип** – совокупность характеристик, присущих особи на определённой стадии развития. Фенотип является результатом экспрессии генотипа и влияния окружающей среды, а

также совокупности этих двух факторов. **Генофонд** – совокупность всех генных вариаций определённой популяции. **Популяция** – совокупность особей одного вида, обладающая общим генофондом, способная к более-менее устойчивому самовоспроизводству [5].

В 1960 году академик И.И. Шмальгаузен сформулировал представление об эволюции, как об авторегулируемом процессе, основанном на обратной связи. В любой авторегулируемой системе выделяются два блока: управляющий блок – регулятор и регулируемый блок – объект управления. Рассматривая процесс эволюции, в качестве регулируемого блока можно выделить популяцию, а в качестве регулятора – среду в которой обитает популяция, и к условиям которой она должна приспосабливаться. Средой, в которой происходят эволюционные преобразования популяции, является экосистема. Данная система представлена на рисунке 1.

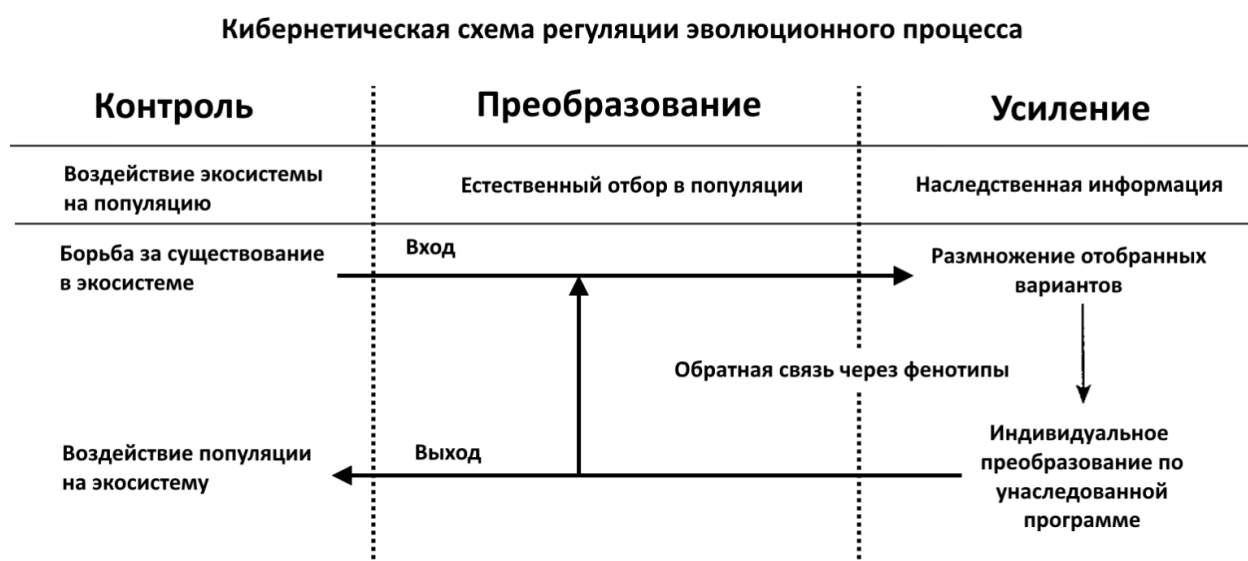


Рисунок 1 – эволюция, как авторегуляторный процесс [5]

Популяция меняется с каждым поколением благодаря генетической изменчивости, подавляющий вклад в которую приносят случайные мутации. В данной модели генетическая изменчивость является каналом прямой связи. Канал обратной связи действует на фенотипическом уровне – на основе генетической информации экосистема «узнаёт» насколько популяция соответствует условиям регулятора.

Таким образом, в эволюционном процессе происходит два этапа перекодировки информации: первый – превращение генетической информации в фенотипическую, и второй – размножение особей, прошедших естественный отбор [5].

2.2 Эволюция программного обеспечения и её связь с естественной эволюцией

Основываясь на описании особенностей процесса естественной эволюции, представим эволюционный подход к изменениям в отношении процесса разработки программного обеспечения.

Так, например, как и в случае с изменяющейся естественной средой – изначальные требования пользователей, в соответствии с которыми создаётся программное обеспечение, в течение времени будут изменяться вслед за изменениями, происходящими в реальном мире. Регулятором, в случае с экосистемой программного обеспечения, является множество участников экосистемы, состоящее из владельцев программного обеспечения, его разработчиков и пользователей. Также вклад могут вносить такие факторы как текущее состояние той области рынка, в которой программное обеспечение конкурирует со своими аналогами, совпадение с запросами и ожиданиями пользователей, актуальность используемой технологической платформы и инструментов и многое другое [6, 7, 8].

Производя сравнение процессов естественной эволюции и эволюции программного обеспечения, можно выделить некоторое множество схожих элементов. Общие элементы, выявленные в процессе соотнесения процессов естественной эволюции и эволюции программного обеспечения, представлены в таблице 1.

Таблица 1 – соотнесение элементов процессов естественной эволюции и эволюции программного обеспечения

Элемент системы	Естественная эволюция	Эволюция программного обеспечения
-----------------	-----------------------	-----------------------------------

Элементарная единица эволюции	Поколение популяции	Версия программного обеспечения
Способ изменения	Случайные изменения генома	Осознанные изменения, которые разработчики принимают, на основании внешних факторов, с учётом неполноты информации
Прямая связь (вход)	Генофонд популяции	Исходный код
Регулируемый блок	Фенотип популяции	Скомпилированный исходный код, который выполняется на компьютере пользователя
Регулятор	Естественная экосистема (живые организмы, окружение, их взаимодействие)	Экосистема программного обеспечения (владельцы, разработчики, пользователи, их взаимодействие)
Обратная связь (выход)	Успех (захват жизненно-важных ресурсов, размножение) или неуспех (вымирание) популяции	Успех (рост популярности, увеличение продаж) или неуспех (потеря аудитории, падение продаж) программного обеспечения

Основываясь на результатах соотнесения процессов естественной эволюции и эволюции программного обеспечения, можно выделить основное различие, а именно — способ внесения изменений в эволюционирующую систему. В случае естественной эволюции все происходящие изменения имеют случайную природу, следовательно результат такого процесса непредсказуем — он может равновероятно привести как к негативным, так и позитивным последствиям. В случае эволюции программного обеспечения, подавляющее большинство производимых изменений является результатом осознанных изменений производимых разработчиками. Однако следует учесть, что хоть разработчики и опираются при принятии решений на рациональные предпосылки, которые, возможно, подкреплены некоторой статистикой, невозможно полностью нивелировать фактор случайности.

Исходя из вышесказанного, можно сделать вывод, что чем ближе вносимое разработчиками изменение сдвигает набор характеристик программного обеспечения к условному «эталонному» набору характеристик,

который установлен регулятором на текущий момент, тем оно эффективнее. Иначе говоря, если усовершенствовать процесс разработки программного обеспечения таким образом, чтобы уменьшилось влияние случайности на процесс разработки, точнее оценивать текущие критерии регулятора, а также предугадывать динамику их изменения, то это в разы сократило временные и финансовые затраты на итерации разработки, а также повысило качество программного обеспечения с точки зрения пользователя.

2.3 Законы эволюции программного обеспечения

Данный подход к рассмотрению процессов связанных с разработкой программного обеспечения не является новым – ещё в 1972 году, в работе Лемана и Беладди, был впервые использован термин «эволюция программного обеспечения». В соответствии с этой работой, эволюция программного обеспечения это ряд последовательных изменений, производимых в процессах разработки и поддержки, которые происходят на всём протяжении жизненного цикла программного обеспечения [9].

Изначально, в работах рассматривающих особенности эволюции программного обеспечения, не уделялось особого внимания классам программ, которые можно было рассматривать как эволюционирующие – упоминалось лишь то, что такие программы должны разрабатываться большими группами разработчиков и использоваться большой аудиторией пользователей. Впоследствии, возникла необходимость уточнения этого аспекта, так как особенности разработки различного программного обеспечения, обладающего схожими масштабами, может значительно отличаться.

В связи с этим, произошли изменения, в ходе которых применимость теории эволюции программного обеспечения стала зависеть не от масштаба разработки, а от принадлежности программного обеспечения к определённому классу в системе SPE. Эта схема делит программное обеспечение на три типа:

1. Тип S (специальная) – программа, разработанная в соответствии с формальной спецификацией, работоспособность которой можно однозначно доказать.

2. Тип P (проблемно-ориентированная) – программа, разработанная для решения формально описанной проблемы, однако работоспособность такой программы невозможно однозначно доказать, так как способ решения проблемы основывается на эвристических методах или приближённых вычислениях.

3. Тип E (эволюционирующая) – программа, разработанная для поддержки происходящих в реальном мире процессов, которые предполагают участие в них людей.

Таким образом, эволюция программного обеспечения стала рассматриваться только в контексте систем E-типа. Это объясняется тем, что реальный мир постоянно изменяется и данный тип программного обеспечения должен развиваться в соответствии с ним. Также, сам факт создания программного обеспечения для решения какой-либо задачи, повлечёт изменение изначальных потребностей пользователей, что в свою очередь создаст необходимость дальнейшего изменения программного обеспечения. Эта особенность приводит к возникновению обратной связи между пользователями и разработчиками, являющейся ключевым механизмом в контексте эволюции программного обеспечения [10].

Одним из важнейших результатов работы Лемана по исследованию особенностей эволюции программного обеспечения E-типа стало формирование законов, по которым происходит эволюция таких систем. Первая работа, в которой были впервые представлены законы эволюции программного обеспечения, была выпущена в 1974 году и содержала только три закона [7].

Впоследствии, на протяжении 20 лет, законы не раз пересматривались, уточняя важные аспекты и упущенные детали, основываясь при этом на работах по оценке эволюции существующего программного обеспечения. В результате, последняя редакция законов эволюции программного обеспечения была выпущена в 1996 году [11]. Законы эволюции программного обеспечения в последней редакции представлены в таблице 2.

Таблица 2 - законы эволюции программного обеспечения в последней редакции

Название закона	Описание
1. Закон непрерывного изменения (Law of continuing change)	Система Е-типа должна постоянно адаптироваться, иначе она будет становиться всё менее удовлетворительной для использования
2. Закон растущей сложности (Law of increasing complexity)	Во время изменения системы Е-типа её сложность возрастает и поддержка затрудняется, если не производить работы по снижению её сложности
3. Закон саморегуляции (Law of self-regulation)	Эволюция крупных систем Е-типа регулируется по принципу обратной связи
4. Закон сохранения организационной стабильности (Law of conservation of organizational stability)	Общий объём выполняемой организацией работы, направленной на эволюцию системы Е-типа, за время её жизни или отдельные этапы развития в среднем постоянен
5. Закон сохранения общности (Law of conservation of familiarity)	Темпы роста системы Е-типа ограничены необходимостью в поддержке общности
6. Закон непрерывного роста (Law of continuing growth)	Функциональные возможности системы Е-типа должны постоянно расширяться, чтобы удовлетворять потребности пользователей
7. Закон падения качества (Law of declining quality)	Если система Е-типа не будет адаптироваться вслед за изменениями среды, в которой она функционирует, её качество будет падать
8. Закон системы с обратной связью (Law of feedback system)	Процесс эволюции системы Е-типа это многоуровневая нелинейная многоагентная система с обратной связью

За прошедшие годы научное сообщество произвело большое количество работ по оценке применимости законов эволюции программного обеспечения. Так, в работе [12], закон системы с обратной связью выделяется как наиболее значимый, являющийся основой к пониманию остальных

законов. Это объясняется тем, что как только программное обеспечение выпускается, на него начинают воздействовать внешние факторы, которые необходимо учитывать при принятии решений, формируя обратную связь между разработчиками и средой, в которой исполняется программное обеспечение.

В работе [13] рассматривается полная история развития идеи эволюции законов программного обеспечения. В ней приводятся ссылки на эмпирические исследования, которые подтвердили, что законы 1 и 4 выполняются во всех рассмотренных случаях. Также, в этой работе отдельно выделяется случай эволюции свободного программного обеспечения, так как для него не удалось подтвердить закон 2.

ЗАКЛЮЧЕНИЕ

В ходе выполнения работы был произведен обзор литературы и введены понятия, связанные с понятиями экосистем программного обеспечения, были приведены основные положения теории эволюции программного обеспечения, существующие на данный момент. Также, было произведено сравнение естественной эволюции с эволюцией программного обеспечения. В результате работы были описаны основные особенности эволюции экосистем программного обеспечения, которые позволят глубже взглянуть на этап поддержки программного обеспечения.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Manikas K., Hansen K. M. Software ecosystems—A systematic literature review //Journal of Systems and Software. – 2013. – Vol. 86, №. 5. – P. 1294-1306.
2. Messerschmitt D. G. et al. Software ecosystem: understanding an indispensable technology and industry //MIT Press Books. – 2005. – V. 1. – P. 432.
3. Jansen S., Brinkkemper S., Finkelstein A. Business Network Management as a Survival Strategy: A Tale of Two Software Ecosystems //IWSECO@ ICSR. – 2009. – P. 34-48.
4. dos Santos R. P., Werner C. M. L. A Proposal for Software Ecosystems Engineering //IWSECO@ ICSOB. – 2011. – P. 40-51.
5. Северцов А. С. Теория эволюции: учебник для студентов вузов, обучающихся по направлению 510600 «Биология»/Алексей Сергеевич Северцов //М.: Гуманитар. изд. центр ВЛАДОС. – 2005. – 380 с.
6. Kutschera U., Niklas K. J. The modern theory of biological evolution: an expanded synthesis //Naturwissenschaften. – 2004. – V. 91. – №. 6. – P. 255-276.
7. Hall B. K., Hallgrímsson B. Strickberger's evolution. The integration of genes, organisms, and populations 4th edition. – 2008. – P. 760.
8. Godfrey M. W., German D. M. The past, present, and future of software evolution //Frontiers of Software Maintenance, 2008. FoSM 2008. – IEEE, 2008. – P. 129-138.
9. Belady L. A., Lehman M. M. An introduction to growth dynamics //Statistical Computer Performance Evaluation. – 1972. – P. 503-511.
10. Ali S. M. et al. Developing an agent-based simulation model of software evolution //Information and Software Technology. – 2018. – V. 96. – P. 126-140.
11. Lehman M. M. Laws of software evolution revisited //European Workshop on Software Process Technology. – Springer, Berlin, Heidelberg, 1996. – P. 108-124.
12. Herraiz I. et al. The evolution of the laws of software evolution: A discussion based on a systematic literature review //ACM Computing Surveys (CSUR). – 2013. – V. 46. – №. 2. – P. 28.
13. Godfrey M. W., German D. M. On the evolution of Lehman's Laws //Journal of Software: Evolution and Process. – 2014. – V. 26. – №. 7. – P. 613-619.