

19. Gladkov L.A., Kureychik V.V., Kureychik V.M. Geneticheskie algoritmy [Kureychik V.M. Genetic algorithms]. Moscow: Fizmatlit, 2010.
20. Emel'yanov V.V., Kureychik V.V., Kureychik V.M. Teoriya i praktika evolyutsionnogo modelirovaniya [Theory and practice of evolutionary modeling]. Moscow: Fizmatlit, 2003.
21. Gladkov L.A., Gladkova N.V., Gromov S.A. Gibrnidnyy algoritm resheniya zadach operativnogo planirovaniya proizvodstvennogo protsessa [Hybrid algorithm for solving problems of operational planning of the production process], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2017, No. 9 (194), pp. 112-123.
22. Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions, *J. Soft Computing*, 2003, pp. 545-562.
23. Michael A., Takagi H. Dynamic control of genetic algorithms using fuzzy logic techniques, *Proceedings of the Fifth International Conference on Genetic Algorithms*. Morgan Kaufmann, 1993, pp. 76-83
24. Lee M.A., Takagi H. Integrating design stages of fuzzy systems using genetic algorithms, *Proceedings of the 2nd IEEE International Conference on Fuzzy System*, 1993, pp. 612-617.

Статью рекомендовала к опубликованию д.т.н., профессор Л.С. Лисицына.

Саак Андрей Эрнестович – Южный федеральный университет; e-mail: aesaak@sfedu.ru; г. Таганрог, Россия; кафедра государственного и муниципального управления; зав. кафедрой.

Гладков Леонид Анатольевич – e-mail: lagladkov@sfedu.ru; тел.: 88634371625; кафедра САПР; к.т.н.; доцент.

Гладкова Надежда Викторовна – e-mail: nvgladkova@sfedu.ru; тел.: 88634393260; кафедра САПР; старший преподаватель.

Saak Andrey Ernestovich – Southern Federal University; e-mail: aesaak@sfedu.ru; Taganrog, Russia; the department of state and municipal administration; head of department.

Gladkov Leonid Anatol'evich – e-mail: lagladkov@sfedu.ru; phone: +78634371625; the department of CAD; cand. of eng. sc.; associate professor.

Gladkova Nadezda Viktorovna – e-mail: nvgladkova@sfedu.ru; phone: +788634393260; the department of CAD; senior teacher.

УДК 004

DOI 10.18522/2311-3103-2022-5-128-141

А.Н. Беликов, С.А. Кучеров, В.С. Лапшин, Ю.Ю. Липко, А.С. Свиридов

АНАЛИЗ ИСПОЛЬЗОВАНИЯ МОДЕЛЕЙ КОММУНИКАЦИИ В ЖИЗНЕННОМ ЦИКЛЕ РАЗРАБОТКИ ИНФОРМАЦИОННЫХ СИСТЕМ

Проводится анализ существующих моделей коммуникации и их применимости к задачам организации эффективной коммуникации в процессе разработки информационных систем. Рассматривается использование моделей коммуникации в сочетании с различными, наиболее часто применяющимися моделями жизненного цикла информационных систем – водопадной, спиральной, SCRUM. Анализируются достоинства и недостатки таких сочетаний и их причины на примере модели Т. Ньюкомба. Основным элементом коммуникации является обеспечение понимания между двумя участниками относительно объекта, который возникает в проекте на очередном шаге его выполнения. Представлена систематизация знаний и информации с учетом субъектов и форм ее представления. Проведенный анализ показывает, что модели, реализуя заложенные в них идеи, в различной степени соответствуют разным моделям жизненного цикла, однако ни одна из моделей коммуникации в полной мере не позволяет решить проблемы коммуникации при разработке информационных систем. Выполненные исследования показывают, что ни одно из сочетаний моделей не позволяет эффективно организовать коммуникацию между участниками в процессе разработки. Необходимо кардинально пересматривать модели и механизмы таким обра-

зом, чтобы субъекты могли осуществлять коммуникацию на этапе подготовки действий по созданию артефактов разработки программного обеспечения, а не по результатам их осуществления. На устранение таких проблем ориентируются подходы, предполагающие разработку и использование инструментов коммуникации – LOW/No code платформ, или иного CASE-инструментария, позволяющего пользователю и разработчику коммуницировать с сохранением контекста передаваемых знаний. Изучение особенностей и структуры таких инструментов будет являться предметом дальнейшего исследования.

Модели коммуникации; разработка информационных систем; жизненный цикл; участники коммуникации; CASE-инструменты; модель коммуникации Ньюкомба; Scram.

A.N. Belikov, S.A. Kucherov, V.S. Lapshin, Yu.Yu. Lipko, A.S. Sviridov

ANALYSIS OF THE COMMUNICATION MODELS USAGE IN THE LIFE CYCLE OF INFORMATION SYSTEMS DEVELOPMENT

The article analyzes the existing communication models and their applicability to the tasks of organizing effective communication in the process of developing information systems. The use of communication models in combination with various, most commonly used models of the life cycle of information systems – waterfall, spiral, SCRUM - is considered. The advantages and disadvantages of such combinations and their causes are analyzed on the example of T. Newcomb's model. The main element of communication is to ensure understanding between the two participants regarding the object that arises in the project at the next step of its implementation. The systematization of knowledge and information is presented, taking into account the subjects and forms of its presentation. The analysis shows that the models, realizing the ideas embedded in them, correspond to different life cycle models to varying degrees, but none of the communication models fully solves communication problems in the development of information systems. The performed studies show that none of the combinations of models allows to effectively organize communication between participants in the development process. It is necessary to radically revise models and mechanisms in such a way that subjects can communicate at the stage of preparing actions to create software development artifacts, and not based on the results of their implementation. Approaches involving the development and use of communication tools – LOW/No code platforms, or other CASE-tools that allow the user and the developer to communicate while preserving the context of the transmitted knowledge are focused on eliminating such problems. The study of the features and structure of such tools will be the subject of further research.

Communication models; information systems development; life cycle; communication participants; CASE-tools; Newcomb communication model; Scram.

Введение. Разработка информационных систем является сложным комплексом процессов и задач, которые выполняются различными группами участников проекта. Одним из ключевых факторов, влияющих на успешность выполнения проекта, является эффективный обмен информацией между его участниками. При этом такая потребность возникает при любой модели жизненного цикла. Статистика выполнения проектов [1] показывает, что такую коммуникацию организовать удастся не всегда, что приводит к нарушению параметров проекта или его срыву.

В данной статье проводится анализ возможности применения существующих моделей коммуникации для решения проблем взаимодействия участников проекта в основных моделях жизненного цикла – каскадной, спиральной, SCRUM и т.д. Целью проведения анализа является выявление основных идей, заложенных в существующие модели коммуникации и их недостатков для применения в процессе разработки информационных систем.

Понятие коммуникации. Особую актуальность коммуникация приобретает в связи с развитием междисциплинарных исследований, так как такого рода исследования требуют организации эффективного взаимодействия между группами участников (субъектов) из различных областей знаний.

Сущность коммуникации изучается еще с первой половины XX века, и за это время накоплено достаточно большое количество определений, отражающих те или иные взгляды на данное явление [2–5]:

- ◆ коммуникация есть информационная связь субъекта с тем или иным объектом - человеком, животным, машиной (М. Каган);
- ◆ коммуникация – обмен информацией между сложными динамическими системами и их частями, которые в состоянии принимать информацию, накапливать ее, преобразовывать (А. Урсул);
- ◆ коммуникация – специфический обмен информацией, процесс передачи эмоционального и интеллектуального содержания (А.Б. Зверинцев, А.П. Панфилова);
- ◆ коммуникация – это «процессы перекодировки вербальной в невербальную и невербальной в вербальную сферы (Почепцов Г.Г.);
- ◆ коммуникация – это тип активного взаимодействия между объектами любой природы, предполагающий информационный обмен (С.А. Азаренко, Г.И. Рузавин, А.Я. Флиер, В.Л. Бернштейн, П.С. Александров);
- ◆ коммуникация – это процесс информационного обмена (М.А. Василик).

Эти и другие определения можно проанализировать с точки зрения основной идеи, заложенной в него, а также соответствующую этому определению модель.

Из существующих определений термина «коммуникация», можно судить от том, что этот процесс основывается на взаимодействии знаний субъектов, то есть на том, что коммуникация – это взаимодействие субъектов, при котором происходит обмен знаний или их передача. Подобное тождество суть процесса коммуникации [4]. Обеспечение должного качества взаимодействия наиболее критично при разработке программных продуктов (ПО), поскольку от него зависит степень соответствия ПО пользовательским требованиям и успех IT-проекта в целом. Однако, как показывает статистика агентства The Standish Group International, доля успешных IT-проектов в среднем составляет 30%, а доля неуспешных и проблемных – 70% [6], что свидетельствует о наличии проблем при взаимодействии субъектов процесса разработки ПО.

Проблемы взаимодействия характеризуются прежде всего «семантическими разрывами», возникающими в процессе обмена знаниями по средствам общения друг с другом. В суждениях и рассуждениях о предметной области субъектами используются термины, которые не имеют однозначного определения для каждого субъекта из группы взаимодействия. В процессе диалога одни субъекты пытаются определить контекст терминов, употребляемых другими субъектами и преобразовать его в аксиоматически определенные понятия [7]. Однако, как показывает практика разработки ПО, эффективность такого преобразования составляет порядка 20–30% правильно интерпретируемых понятий от общего числа используемых субъектами терминов.

Проблемы коммуникации, возникающие при разработке ПО усугубляются наличием разнородных групп участников (субъектов) этого процесса, использующих базовые наборы терминологии, методов и подходов к реализации своих задач в общем процессе разработки.

Основываясь на этом, актуальной становится задача определения такого вида коммуникации, которая будет адекватна процессу разработки ПО, и позволит устранить указанные ошибки взаимодействия субъектов.

Коммуникация как объект исследования предполагает построение ее модели [5]. Из текущего состояния исследований, ориентированных на построение и анализ моделей коммуникаций, представленных в работах [2–5], можно выделить следующие идеи, заложенные в модели коммуникации:

- ◆ организовать безошибочную передачу данных (модели коммуникации – (Шенона-Вивера, Шрама, Берло, ДеФлера) [8–11];
- ◆ достичь понимания у всех субъектов коммуникации о структурных и поведенческих характеристиках разрабатываемой системы (модели коммуникации – Ньюкомба, Гербнера, Якобсона, Лотмана) [12–15];
- ◆ повлиять на некоторых субъектов коммуникации (модели коммуникации Уэстли-Маклейн, Малецке, Луман) ориентируя их на подготовку к выполнению целевых действий [16–18].

С точки зрения процесса разработки информационных систем, коммуникацию можно определить как единство процессов формирования группы субъектов и организации их взаимодействия для достижения понимания о структурных и поведенческих характеристиках разрабатываемой системы. Исходя из данного определения, далее рассмотрим реализацию одной из существующих моделей коммуникации разработанных на основе идеи о достижении понимания у субъектов – модели коммуникации Ньюкомба [12]. Выбор такой модели обусловлен наличием в ее структуре образа предмета коммуникации, относительно которого достигается понимание. В рамках процесса проектирования систем образа предмета коммуникации является образом разрабатываемой системы. Модель коммуникации будет применена в рамках наиболее популярных процессов организации жизненного цикла разработки систем: водопад, спираль, Scram [19].

Применение модели коммуникации Ньюкомба в жизненном цикле SCRUM. Рассмотрим проблему коммуникации и её негативное влияние на результаты процесса разработки программного обеспечения на примере модели Ньюкомба [12] и метода организации процесса разработки программного обеспечения SCRUM [20].

Основной идеей коммуникации по Ньюкомбу является достижение симметрии в треугольнике Субъект-Объект-Субъект. Симметрия должна выражаться в едином восприятии объекта двумя субъектами и едином восприятии субъектами друг друга. Применительно к процессу разработки программного обеспечения это означает, что субъект, испытывающий потребность в некоторой системе, субъект, имеющий возможность реализовать данную систему и сам образ системы находятся в согласованном виде.

SCRUM, как один из методов организации разработки программного обеспечения преследует отчасти аналогичные цели – заказчик (субъект, испытывающий потребность в системе) и разработчик (субъект, имеющий возможность реализовать систему) должны в конечном итоге прийти к согласованной точке зрения относительно системы (её образа, представленного моделями, прототипами и пр.), что также должно означать что заказчик с разработчиком понимают друг друга. Ключевыми особенностями здесь являются цикличность процесса, поскольку SCRUM – итеративный метод, и наличие обратной связи, так как каждая итерация подразумевает оценку степени соответствия результатов ожиданиям. Изложенное выше показывает, что модель коммуникации Ньюкомба не противоречит принципам, лежащим в основе SCRUM.

В целях выявления проблем коммуникации приведем исходную А-В-Х модель к терминологии SCRUM. Согласно Т. Ньюкомбу, простейший коммуникативный акт состоит в том, что субъект (А) передает информацию о каком-либо объекте или явлении (Х) субъекту (В). В качестве субъекта А в SCRUM участвует владелец SCRUM продукта (Владелец, SCRUM Product Owner), в качестве субъекта В – команда разработчиков (Разработчики, SCRUM Development Team), в качестве объекта Х рассматривается Продукт.

Таким образом, модель единичного коммуникативного акта по Т. Ньюкомбу в терминологиях SCRUM будет выглядеть следующим образом (рис. 1).

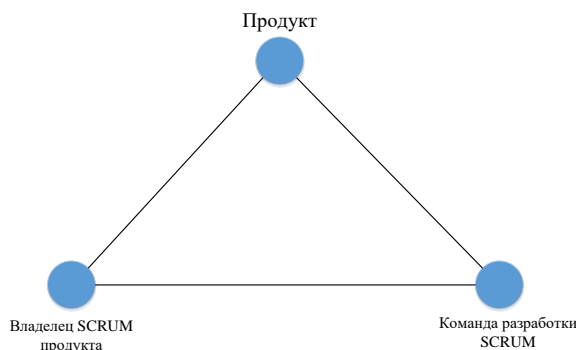


Рис. 1. Модель единичного коммуникативного по Т. Ньюкомбу на примере SCRUM

Метод SCRUM носит итеративный характер и направлен на создание Разработчиками продукта, соответствующего ожиданиям Владельца, а по Ньюкомбу коммуникативные акты осуществляются для достижения симметрии, выражающейся, по сути, в одинаковом понимании объекта. Следовательно, в итеративном процессе происходит некоторое «уточнение» представления, причём понятие представления применимо не только к объекту, но и к обоим субъектам.

Так, владелец SCRUM продукта имеет собственное представление о системе, которое уточняется, расширяется и корректируется в процессе коммуникации с командой разработки. Это обуславливается техническими и технологическими ограничениями (не все желаемые функции могут быть реализованы программно), компетенциями команды (не все желаемые функции и возможности могут быть реализованы разработчиками), ресурсами и т.д. Аналогичным образом команда разработчиков SCRUM имеет своё представление о системе, что с учетом специфики данного метода означает наличие знаний и опыта по реализации типовой системы и одновременное отсутствие знаний по специфике её применения владельцем SCRUM продукта. Данное представление уточняется, расширяется и корректируется в процессе коммуникации с Владельцем. Поскольку основная цель SCRUM – создание продукта, а содержание SCRUM – работы по доведению его прототипов и вариантов реализации до уровня, удовлетворяющего целям Владельца, то и продукт (система) также имеет свой образ, который базируется на образе типового решения (например, интернет-магазин или CRM-система) и уточняется в процессе коммуникации Владельца с Командой.

Сказанное выше отражает первый недостаток модели коммуникации по Т. Ньюкомбу – она не содержит элементов, позволяющих описать и формализовать процесс схождения представлений в точку симметрии.

Следующий аспект заключается в том, чем обмениваются субъекты в процессе коммуникации. Очевидно, что каждый из них обладает своим багажом знаний, а знания, в свою очередь, могут быть классифицированы на следующие виды:

- ◆ Знания, необходимые для осуществления коммуникации. Сюда входят знания терминологии предметной области, технических терминов, знания о моделях и нотациях моделирования, используемых для представления образа продукта и пр.

- ◆ Знания, необходимые для осуществления схождения представления в точку симметрии. Сюда входят знания о процессе создания объекта, о процессе его использования, о степени соответствия результатов ожиданиям и пр.

В табл. 1 представлена систематизация знаний и информации с учетом субъектов и форм представления.

Таблица 1

Знания, используемые в SCRUM-коммуникации

Субъект	Категория знания	Знание	Информация	Способ представления
Владелец SCRUM продукта	Знания для осуществления коммуникации	Терминология	–	–
		Модели и нотации	–	–
		Процесс создания объекта		
	Знания для осуществления схождения представления в точку симметрии	Формы воплощения объекта	Информация о форме воплощения объекта	Естественный язык, с использованием терминологии
		Представление о будущем процессе эксплуатации объекта	Информация о будущем процессе эксплуатации объекта	Естественный язык, с использованием терминологии и привязкой к процессу создания объекта
		Степень понимания задачи командой разработчиков SCRUM	–	–
Команда разработчиков SCRUM	Знания для осуществления коммуникации	Терминология	–	–
		Модели и нотации	–	–
		Процесс создания объекта	–	–
	Знания для осуществления схождения представления в точку симметрии	Формы воплощения объекта	Информация о форме воплощения объекта	Конкретное понятие из онтологии объектов
		О будущем объекте (Функции, структура)	Информация об особенностях эксплуатации объекта заданного типа	Модели, диаграммы, текстовые описания с использованием терминологии
		Степень соответствия продукта ожиданиям заказчика	–	–

Из таблицы видно, что не все знания могут быть представлены информацией. Часть из них, например, терминология, выполняют роль средства при формировании другой информации. Другая часть знаний, например, степень соответствия продукта ожиданиям заказчика, используется в качестве триггера или критерия при принятии решения о готовности продукта. Стоит также учитывать, что каждый субъект, помимо знаний, характеризуется навыками передачи информации, которые также вносят вклад в конечный результат коммуникации.

В процессе передачи знания для осуществления схождения представления в точку симметрии превращаются с помощью знаний, необходимых для осуществления коммуникации, в информацию, релевантную объекту и передаваемую меж-

ду субъектами. Каждая информация, помимо направленности и содержащихся в ней знаний имеет свою форму представления. По аналогии с отсутствием элементов, позволяющих описать и формализовать процесс схождения представлений в точку симметрии, модель Т. Ньюкомба также не содержит аспектов, характеризующих знания, информацию и способы её передачи. Недостатки модели коммуникации по Т. Ньюкомбу применительно к методу организации процесса разработки программного обеспечения SCRUM показаны на рис. 2.



Рис. 2. Недостатки модели коммуникации по Т. Ньюкомбу в SCRUM

Рассмотренный в примере анализ коммуникации может быть спроецирован на иные известные подходы к организации процесса разработки, как классические водопад и спираль.

Применение модели коммуникации Ньюкомба в жизненном цикле «Водопад» (каскад). Водопадная модель (или модель Waterfall) жизненного цикла подразумевает под собой строгую последовательность этапов [21]. На каждом этапе жизненного цикла порождается определенный набор технических решений и документов, при этом для каждого этапа исходными являются документы и решения, принятые на предыдущем этапе.

Весь процесс разработки ИС можно разделить на несколько этапов [22]:

Деление может производиться по различным принципам. Проведем условное разделение по содержанию работ, выполняемых на каждом из этапов:

1. Обследование предметной области (сбор исходных данных, выявление потоков документов, функций и т.д.);
2. Бизнес-моделирование (разработка моделей и требований к системе);
3. Проектирование (проектирование архитектуры системы, проектирование БД и т.д.);
4. Реализация системы (программирование, тестирование);
5. Ввод системы в эксплуатацию.

Однако такой последовательности этапов присущи недостатки. В частности, если какая-либо недоработка закралась на начальном этапе разработки и осталась незамеченной, устранять ее зачастую приходится уже после внедрения. Подобных недоработок всегда бывает предостаточно, и потери от их устранения и преобразования всей системы выражаются значительными суммами.

Рассматривая необходимость коммуникаций между участниками проекта по модели Ньюкомба можно выделить ключевые точки и объекты над которыми производится коммуникация (рис. 3).



Рис. 3. Схема проецирования модели Ньюкомба на водопадную модель ЖЦ

Как видно из схемы, образуется несколько объектов (бизнес, проект системы, система) вокруг которых достигается понимание между различными субъектами (заказчик, системный аналитик, проектировщик системы, команда разработки).

При этом у каждого участника коммуникации в рамках водопадной модели свой набор знаний, представленный в табл. 2.

Таблица 2

Знания, используемые в водопадной-коммуникации

Знания заказчика	Знания системного аналитика	Знания проектировщика	Знания команды разработки
- Знания о предметной области (профессиональной деятельности)	- Знания о предметной области - Знания о возможности ИТ Знания терминологии ИТ	- Знания о возможности ИТ - Знания терминологии ИТ - Знания о типах объектов ИТ - Знания о форме воплощения объектов	- Знания о форме воплощения объектов ИТ - Знание о процессе создания объектов ИТ

Рассмотрим более подробно проецирование модели Ньюкомба для одного объекта в рамках водопадной модели ЖЦ (в качестве примера рассмотрим коммуникацию между заказчиком и системным аналитиком).

Итак, в рамках коммуникации между заказчиком и системным аналитиком происходит передача информации (знаний) между субъектами о объекте-бизнесе. При такой передаче заказчик оперирует знаниями о предметной области, а системный аналитик должен при этом обладать как знаниями о предметной области, так и знаниями о возможности ИТ. Передача знаний от заказчика к разработчику происходит в форме естественного языка, а в обратном направлении в форме моделей. Результатом такой коммуникации должны служить артефакты (модели, техническое задание и т.д.), которые будут исходными данными для коммуникации между системным аналитиком и проектировщиком системы вокруг объекта-проекта системы.

Анализируя применение модели, можно сделать вывод, о том, что при существующей коммуникации в рамках водопадной модели возникает проблема, заключающаяся в передаче полной информации от этапа к этапу (субъекта к субъекту) и невозможности отследить её отследить вовремя. Другой проблемой в рамках водопадной модели является разобщенность между знаниями субъектов и согласования знаний в рамках различных объектов (представлений объекта).

Применение модели коммуникации Ньюкомба в спиральной модели жизненного цикла. Спиральная модель жизненного цикла [23], как и каскадная модель разделяет разработку на несколько фаз, а формирование результата проекта происходит несколькими итерациями. Каждый виток спирали соответствует созданию части продукта, уточняются параметры проекта, определяется его качество и планируются работы следующей итерации.

Спиральная модель хорошо подходит для проектов с высоким уровнем неопределенности требований к конечному продукту и позволяет уточнять требования при выполнении дальнейших итераций. Как и в каскадной при использовании спиральной модели существует несколько фаз [23], на которых задействованы разные участники.

Рассматривая необходимость коммуникаций между участниками проекта по модели Ньюкомба, можно выделить ключевые точки, в которых проходит коммуникация – на фазе определения целей между заказчиком и аналитиком, на фазе проектирования – между аналитиком и проектировщиком, на фазе разработки – между проектировщиком и командой разработки, на фазе планирования – между тестирующим и заказчиком.

При это можно заметить, что на разных фазах один и тот же участник проекта является участником коммуникации в разной роли – как потребитель и как источник информации, необходимой для выполнения проекта.

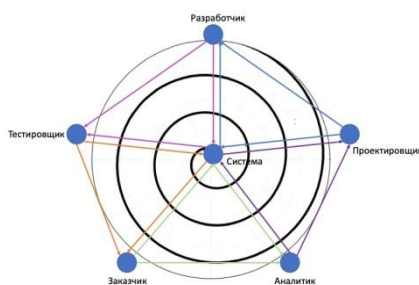


Рис. 4. Коммуникация по модели Ньюкомба для спиральной модели жизненного цикла

При выполнении проекта необходимо обеспечить передачу знаний при осуществлении коммуникации участников. Содержание таких знаний аналогично рассмотренным в табл. 1. и оформляется в виде требований и моделей, чаще всего построенных на языке UML.

Анализ структуры коммуникации и передаваемых знаний показывает, что знания о терминологии выполняют роль средства при формировании другой информации. Другая часть знаний, например, степень соответствия продукта ожиданиям заказчика, используется в качестве триггера или критерия при принятии решения о готовности продукта. Стоит также учитывать, что каждый субъект, помимо знаний, характеризуется навыками передачи информации, которые также вносят вклад в конечный результат коммуникации.

Анализ полученных результатов. Также были проанализированы модели коммуникации Д. Берло, ДеФлера, Гербнера, Уэсли Маклейна [5] применительно к моделям жизненного цикла разработки программного обеспечения – спиральной, водопадной, Scrum. При рассмотрении проводился анализ, насколько заложенные в модель коммуникации идея и средства (элементы модели) могут решить проблемы, возникающие при взаимодействии участников проекта.

Проведенный анализ показывает, что модели, реализуя заложенные в них идеи, в различной степени соответствуют разным моделям жизненного цикла, однако ни одна из моделей коммуникации в полной мере не позволяет решить проблемы коммуникации при разработке информационных систем.

Заложенная в модели коммуникации Шенона-Вивера, Шрама, Берло, ДеФлера [8–11] идея об обеспечении безошибочной передачи данных может применяться при подходе к разработке с ориентацией на фиксацию проектной информации в виде спецификаций и требований, однако общей проблемой моделей данного типа является недостаток взаимопонимания. Этот недостаток пытаются преодолеть за счет унификации кодирования информации – представления их в виде, фиксируемом методологиями, моделями. Однако содержательное их понимание при этом не обеспечивается.

Модели коммуникации Ньюкомба, Гербнера, Якобсона, Лотмана [12–15], ставят перед собой задачу обеспечения понимания между субъектами и в большей степени подходят для жизненного цикла, ориентированного на обеспечение единообразного представления всех участников проекта относительно его результатов. Однако при этом сохраняются недостатки, связанные с возможностью различных интерпретаций этого представления и его преобразования в действия по созданию целевого объекта.

Попытки совмещения моделей ориентированных на влияние на субъектов коммуникации (модели коммуникации Уэстли-Маклейн, Малецке, Луман [16–18]) с целью их на подготовки к выполнению действий также имеют проблемы, которые заключаются в их ориентацией на канал передачи данных, представляемый техническими средствами массовых коммуникаций. Интерпретация такой формы канала на процессы разработки системы не представляется возможной вследствие упрощения параметров канала С, отвечающего за описание обратной связи. Параметры канала СМК, позволяют формировать внешнее действие на основе передачи «ощущения» других субъектов, однако моделирование таких свойств канала С в рамках разработки информационных систем не представляется возможным.

Попытка адаптировать известные модели коммуникации к потребностям процессов разработки программного обеспечения приводит лишь к частичному устранению недостатков. Обратимся для примера к возможной модификации модели коммуникации по Т. Ньюкомбу, чтобы она удовлетворяла потребностям метода организации процесса разработки программного обеспечения SCRUM. Для этого попытаемся устранить недостатки, изложенные при рассмотрении этой модели (рис. 5).

В представленной на рисунке выше модели коммуникации отражены характеристики информации и знаний. Сочетание видов и форм представления знаний, используемых в методах разработки программного обеспечения, а также способов обмена ими, заложенными в моделях коммуникаций так или иначе приводят к одному – коммуницирующие субъекты вынуждены обмениваться информацией по факту уже совершенных действий. Коммуницирующие субъекты способны влиять только на оценку некоторого созданного объекта или свершившегося события, вызывая тем самым к повторному исполнению процессов их создания. Необходимо кардинально пересматривать модели и механизмы таким образом, чтобы субъекты могли осуществлять коммуникацию на этапе подготовки действий по созданию артефактов разработки программного обеспечения, а не по результатам их осуществления.

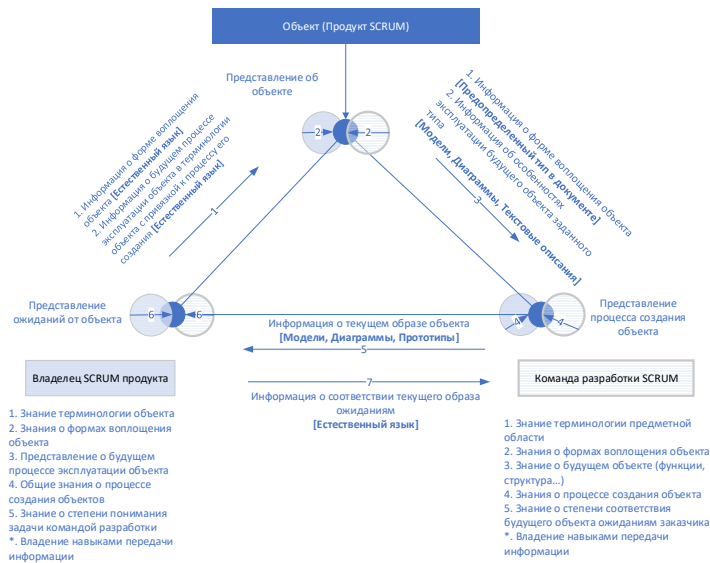


Рис. 5. Модифицированная модель коммуникации Т. Ньюкомба для обеспечения потребностей метода Scrum

Заключение. Таким образом, проецирование существующих моделей коммуникации на жизненный цикл процесса разработки характеризуется рядом проблем, связанных со спецификой той или иной модели коммуникации. Также были получены выводы, свидетельствующие о том, что модели коммуникации, ориентированные на достижения понимания у коммуницирующих субъектов, в наибольшей мере соответствуют специфике процесса разработки ПО и методологии проектной деятельности. Однако их применение также характеризуется рядом проблем, связанных с использованием разных видов и форм представления знаний, а также отсутствием способов формального представления контекста передаваемых знаний. На устранение таких проблем ориентируются подходы, предполагающие разработку и использование инструментов коммуникации – LOW/No code платформ, или иного CASE-инструментария, позволяющего пользователю и разработчику коммуницировать с сохранением контекста передаваемых знаний. Изучение особенностей и структуры таких инструментов будет являться предметом дальнейшего исследования.

Благодарности. Исследование выполнено за счет гранта Российского научного фонда № 22-19-00723, <https://rscf.ru/project/22-19-00723/>.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Davis. Why software development projects fail. – <https://www.3pillarglobal.com>. – URL: <https://www.3pillarglobal.com/insights/why-software-development-projects-fail/> (дата обращения: 06.11.2022).
2. Василик М.А. Основы теории коммуникации / под ред. М.А. Василика. Гардерика. – М., 2005.
3. Почепцов Г.Г. Теория коммуникации. – 2001.
4. Шарков Ф. Коммуникология: основы теории коммуникации. – Litres, 2020.
5. Тухватулина Л.Р. Принципы классификации моделей коммуникации // Вестник Томского государственного педагогического университета. – 2006. – №. 7.
6. Standish Group 2022 Chaos Report – Beyond Infinity Overview. – Режим доступа: <https://www.standishgroup.com/news/49> (дата доступа: 06.11.2022).

7. *Финн В.К.* Философские проблемы логики интеллектуальных систем // Новости искусственного интеллекта. – 1999. – № 1. – С. 36-51.
8. *Дорофеева И.В.* Модель Шеннона-Уивера и ее значение для развития теории коммуникации // Языковой дискурс в социальной практике. – 2013. – С. 49-53.
9. *Schramm W.* How communication works // The process and effects of mass communication. – 1954. – Vol. 3. – P. 26.
10. *Berlo D.K.* Communication as process: Review and commentary // Annals of the International Communication Association. – 1977. – Vol. 1, No. 1. – P. 11-27.
11. *De Fleur M.L.* A mass communication model of stimulus response relationships: An experiment in leaflet message diffusion // Sociometry. – 1956. – Vol. 19, No. 1. – P. 12-25.
12. *Newcomb H.M.* On the dialogic aspects of mass communication // Critical Studies in Media Communication. – 1984. – Vol. 1, No. 1. – P. 34-50.
13. *Gerbner G.* Toward a general model of communication // Audio visual communication review. – 1956. – P. 171-199.
14. *Якобсон Р.* Лингвистика и поэтика. – 1975.
15. *Лотман Ю.М.* Автокоммуникация: «Я» и «Другой» как адресаты // Внутри мыслящих миров. Человек–текст–семиосфера–история. – 1996. – С. 36.
16. *Westley B.H., MacLean Jr M.S.* A conceptual model for communications research // Journalism Quarterly. – 1957. – Vol. 34, No. 1. – P. 31-38.
17. *Maletzke G.* Psychologie der Massenkommunikation: theorie und Systematik. – 1963.
18. *Луман Н., Озирченко Д.* Что такое коммуникация? // Социологический журнал. – 1995. – № 3. – С. 114-124.
19. *Ruparelia N.B.* Software development lifecycle models // ACM SIGSOFT Software Engineering Notes. – 2010. – Vol. 35, No. 3. – P. 8-13.
20. *Джефф Сазерленд.* SCRUM. Революционный метод управления проектами = SCRUM. The art of doing twice the work in half the time. – Манн, Иванов и Фербер, 2016. – 288 с. – ISBN 978-5-00057-722-6.
21. *Royce Winston.* Managing the Development of Large Software Systems // Technical Papers of Western Electronic Show and Convention. – August 1970. – P. 1-9.
22. *Шлапак Вера Сергеевна, Теодорович Наталья Николаевна.* Совершенствование жизненного цикла информационных систем малого и среднего бизнеса // Сервис +. – 2015. – № 3. – URL: <http://cyberleninka.ru/article/n/sovershenstvovanie-zhiznennogo-tsikla-informatsionnyh-sistem-malogo-i-srednego-biznesa>.
23. *Boehm B.W.* A Spiral Model of Software Development and Enhancement // Computer. – 1988. – Vol. 21 (5). – P. 61-72.

REFERENCES

1. *Davis.* Why software development projects fail. Available at: <https://www.3pillarglobal.com/insights/why-software-development-projects-fail/> (accessed 06 November 2022).
2. *Vasilik M.A.* Osnovy teorii kommunikatsii [Fundamentals of the theory of communication], ed. by M.A. Vasilika. Garderika. Moscow, 2005.
3. *Pochepstov G.G.* Teoriya kommunikatsii [Theory of communication], 2001.
4. *Sharkov F.* Kommunikologiya: osnovy teorii kommunikatsii [Communicology: fundamentals of the theory of communication]. Litres, 2020.
5. *Tukhvatulina L.R.* Printsipy klassifikatsii modeley kommunikatsii [Principles of classification of communication models], *Vestnik Tomskogo gosudarstvennogo pedagogicheskogo universiteta* [Bulletin of Tomsk State Pedagogical University], 2006, No. 7.
6. Standish Group 2022 Chaos Report – Beyond Infinity Overview. Available at: <https://www.standishgroup.com/news/49> (accessed 06 November 2022).
7. *Finn V.K.* Filozofskie problemy logiki intellektual'nykh sistem [Philosophical problems of logic of intelligent systems], *Novosti iskusstvennogo intellekta* [Artificial intelligence News], 1999, No. 1, pp. 36-51.
8. *Dorofeeva I.V.* Model' SHennona-Uivera i ee znachenie dlya razvitiya teorii kommunikatsii [The Shannon-Weaver model and its significance for the development of communication theory], *Yazykovoy diskurs v sotsial'noy praktike* [Language discourse in social practice], 2013, pp. 49-53.

9. Schramm W. How communication works, *The process and effects of mass communication*, 1954, Vol. 3, pp. 26.
10. Berlo D.K. Communication as process: Review and commentary, *Annals of the International Communication Association*, 1977, Vol. 1, No. 1, pp. 11-27.
11. De Fleur M.L. A mass communication model of stimulus response relationships: An experiment in leaflet message diffusion, *Sociometry*, 1956, Vol. 19, No. 1, pp. 12-25.
12. Newcomb H.M. On the dialogic aspects of mass communication, *Critical Studies in Media Communication*, 1984, Vol. 1, No. 1, pp. 34-50.
13. Gerbner G. Toward a general model of communication, *Audio visual communication review*, 1956, pp. 171-199.
14. Yakobson R. Lingvistika i poetika [Linguistics and poetics], 1975.
15. Lotman Yu.M. Avtokommunikatsiya: «Ya» i «Drugoy» kak adresaty [Autocommunication: "I" and "The Other" as addressees], *Vnutri myslyashchikh mirov. Chelovek–tekst–semiosfera–istoriya* [Inside thinking worlds. Man–text–semiosphere–history], 1996, pp. 36.
16. Westley B.H., MacLean Jr M.S. A conceptual model for communications research, *Journalism Quarterly*, 1957, Vol. 34, No. 1, pp. 31-38.
17. Maletzke G. Psychologie der Massenkommunikation: theorie und Systematik. 1963.
18. Luman N., Ozirchenko D. Chto takoe kommunikatsiya? [What is communication?], *Sotsiologicheskii zhurnal* [Sociological Journal.], 1995, No. 3, pp. 114-124.
19. Ruparelia N.B. Software development lifecycle models, *ACM SIGSOFT Software Engineering Notes*, 2010, Vol. 35, No. 3, pp. 8-13.
20. Dzheff Sazerlend. SCRUM. Revolyutsionnyy metod upravleniya proektami = SCRUM [SCRUM. A revolutionary method of project management = SCRUM]. The art of doing twice the work in half the time. Mann, Ivanov i Ferber, 2016, 288 p. ISBN 978-5-00057-722-6.
21. Royce Winston. Managing the Development of Large Software Systems, *Technical Papers of Western Electronic Show and Convention*, August 1970, pp. 1-9.
22. Shlapak Vera Sergeevna, Teodorovich Nataliya Nikolaevna. Sovershenstvovanie zhiznennogo tsikla informatsionnykh sistem malogo i srednego biznesa [Improving the life cycle of small and medium-sized business information systems.], *Servis +* [Service +], 2015, No. 3. Available at: <http://cyberleninka.ru/article/n/sovershenstvovanie-zhiznennogo-tsikla-informatsionnyh-sistem-malogo-i-srednego-biznesa>.
23. Boehm B.W. A Spiral Model of Software Development and Enhancement, *Computer*, 1988, Vol. 21 (5), pp. 61-72.

Статью рекомендовал к опубликованию д.т.н. Ю.А. Кравченко.

Беликов Александр Николаевич – Южный федеральный университет; e-mail: anbelikov@sfedu.ru; г. Таганрог, Россия; тел.: +79185945508; кафедра системного анализа и телекоммуникаций; старший преподаватель.

Кучеров Сергей Александрович – e-mail: skuchеров@sfedu.ru; тел.: +79612966919; кафедра системного анализа и телекоммуникаций; к.т.н.; доцент.

Лапшин Вячеслав Сергеевич – e-mail: lapshin@sfedu.ru; тел.: +79525690073; . кафедра системного анализа и телекоммуникаций; к.т.н.; старший преподаватель.

Липко Юлия Юрьевна – e-mail: yลิปко@sfedu.ru; тел.: +79185566893; кафедра системного анализа и телекоммуникаций; к.т.н.; доцент.

Свиридов Александр Славьевич – e-mail: asviridov@sfedu.ru; тел.: +79185567529; кафедра системного анализа и телекоммуникаций; к.т.н.; доцент.

Belikov Alexsandr Nikolaevich – Southern Federal University; e-mail: anbelikov@sfedu.ru; Taganrog, Russia; phone: +79185945508; the department of system analysis and telecommunications; senior lecturer.

Kuchеров Sergey Alexandrovich – e-mail: skuchеров@sfedu.ru; phone: +79612966919; the department of system analysis and telecommunications; cand. of eng. sc.; associate professor.

Lapshin Vyacheslav Sergeevich – e-mail: lapshin@sfedu.ru; phone: +79525690073; the department of system analysis and telecommunications; cand. of eng. sc.; senior lecturer.

Lipko Yuliya Yur'evna – e-mail: ylipko@sfedu.ru; phone: +79185566893; the department of system analysis and telecommunications; cand. of eng. sc.; associate professor.

Sviridov Alexandr Slav'evich – e-mail: asviridov@sfedu.ru; phone: +79185567529; the department of system analysis and telecommunications; cand. of eng. sc.; associate professor.

УДК 004.032

DOI 10.18522/2311-3103-2022-5-141-151

С.М. Гушанский, В.С. Потапов

ИССЛЕДОВАНИЕ КВАНТОВОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ И РЕАЛИЗАЦИЯ КВАНТОВОГО ЯДРА НА ПЛИС

Метод квантового ядра является одним из важнейших методов в квантовом машинном обучении. Однако количество признаков, используемых для квантовых ядер, ограничено несколькими десятками признаков. Используется структура состояния блочного продукта в качестве карты квантовых признаков и демонстрируется реализация программируемых вентиляционных матриц. Актуальность данных исследований заключается в математическом и программном моделировании и реализации квантовой вычислительной системы в рамках разработки реализации квантового ядра на ПЛИС для решения классов задач классического характера. Научная новизна данного направления исследований заключается в разработке гибридного симулятора квантового ядра центрального процессорного устройства (ЦПУ) и программируемой логической интегральной схемы (ПЛИС) на несколько порядков быстрее, чем обычный симулятор квантовых вычислений. Эта совместная разработка реализованного квантового ядра и его эффективная реализация ПЛИС позволили выполнить численное моделирование квантового ядра на основе вентилях с точки зрения входных признаков, вплоть до 780-мерных признаков с использованием 4000 выборок. Применили квантовое ядро к задачам классификации изображений с использованием набора данных Fashion-MNIST и показали, что квантовое ядро сравнимо с ядрами Гаусса с оптимизированной пропускной способностью. Проведенный анализ работ данной области показал, что в настоящее время достигнут новый качественный уровень, открывающий перспективные возможности по реализации многокубитовых квантовых вычислений. Перспективы реализации и развития связаны не только с технологическими возможностями, но и с решением вопросов построения эффективных квантовых систем решения актуальных математических задач, задач криптографии и задач управления (оптимизации).

Моделирование; квантовый алгоритм; кубит; модель квантового вычислителя; запутывание; суперпозиция; квантовый оператор.

S.M. Gushanskiy, V.S. Potapov

STUDY OF A QUANTUM COMPUTING SYSTEM AND IMPLEMENTATION OF A QUANTUM CORE ON FPGA

The quantum core method is one of the most important methods in quantum machine learning. However, the number of features used for quantum nuclei is limited to a few dozen features. The block product state structure is used as a quantum feature map and the implementation of programmable gate matrices is demonstrated. The relevance of these studies lies in the mathematical and software modeling and implementation of a quantum computing system as part of the development of the implementation of a quantum core on FPGA for solving classes of problems of a classical nature. The scientific novelty of this research area is the development of a hybrid simulator of the quantum cores of a central processing unit (CPU) and a programmable logic integrated circuit (FPGA) several orders of magnitude faster than a conventional quantum computing simulator. This joint development of the implemented quantum core and its efficient FPGA imple-