

УДК 334.012

***ОСОБЕННОСТИ ОРГАНИЗАЦИИ ДЕЯТЕЛЬНОСТИ ПРЕДПРИЯТИЯ В
СФЕРЕ РАЗРАБОТКИ И ВНЕДРЕНИЯ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ, РИСКИ, ХАРАКТЕРНЫЕ ДЛЯ ДАННОЙ СФЕРЫ***

Ежов Д.С.

ФГБОУ ВО "НИ МГУ им. Н. П. Огарёва",

Саранск, Россия

Аннотация

Компании сферы разработки и внедрения программного обеспечения на сегодняшний день образуют фундамент процесса информатизации отечественной и мировой экономики. В связи с этим значимость изучения компаний отрасли существенно возрастает, поскольку эффективность таковых во многих аспектах может существенно способствовать эффективности в других отраслях народного хозяйства. Необходимость перевода экономики на инновационные рельсы, ставшая символом отечественной экономики во втором десятилетии двадцать первого века, лишь подтверждает факт того, что в ближайшей перспективе отрасль будет развиваться масштабными темпами.

Статья посвящена изучению организации деятельности предприятий в области разработки и внедрения программного обеспечения. Рассмотрены основные модели и риски характерные для данной сферы.

Ключевые слова: Разработка и внедрение программного обеспечения, программная инженерия, модели производственного процесса, экономические риски.

***FEATURES OF THE ORGANIZATION OF THE ENTERPRISE'S ACTIVITIES
IN THE FIELD OF DEVELOPMENT AND
IMPLEMENTATION OF THE SOFTWARE, RISKS CHARACTERISTIC FOR
THIS SPHERE***

Ezhov D.S.

National Research Mordovia State University,

Saransk, Russia

Annotation

Companies in the field of software development and implementation today form the foundation of the process of informatization of the domestic and global economies. In this regard, the importance of studying companies in the industry is significantly increasing, since the effectiveness of such in many aspects can significantly contribute to efficiency in other sectors of the economy. The need to transfer the economy to innovative tracks, which has become a symbol of the domestic economy in the second decade of the twenty-first century, only confirms the fact that in the near future the industry will develop at a large pace.

The article is devoted to the study of the organization of enterprises in the field of software development and implementation. The main models and risks specific to this area are considered.

Keywords: Software development and implementation, software engineering, production process models, economic risks.

В рамках развития практического знания в текущем столетии обозначились инновационные отрасли, призванные обеспечить переход экономической системы на новый этап развития. Так называемый новый технологический уклад возымел особенно сильное влияние по окончании мирового финансового кризиса 2008-2010 гг. Компьютерные технологии не просто плотно вошли в повседневную рутину не только производственных институтов, но в жизнь каждого отдельно взятого индивида. Вспоминая Н. Д. Кондратьева и его теорию больших циклов в увязке с последним глобальным финансовым кризисом [4] можно поспорить с данным суждением на основе того, что развитие компьютерных технологий в период Вектор экономики | www.vectoreconomy.ru | СМИ Эл № ФС 77-66790, ISSN 2500-3666

посткризисного развития как раз только должно было стать фундаментально новым и формирующимся направлением. Однако сегодня все более очевидно, что период первоначального развития и становления предприятий, специализирующихся на разработке программного обеспечения, пришелся не на последний технологический цикл (2010-н.в.), но на предыдущий, завершившийся во время упомянутой выше рецессии [4]. В связи с этим можно утверждать, что на данный момент деятельность по разработке и внедрению программного обеспечения (далее – ПО) становится фундаментом уже нового уклада, что значительно увеличивает ее значимость для современной высокотехнологичной экономике. Однако необходимо отметить, что данная тенденция характерна в большей мере для стран с развивающимися и развитыми экономическими система, поскольку только лишь они дают требуемый базис для поддержания инновационных трендов.

Согласно общероссийскому классификатору видов экономической деятельности (ОКВЭД), разработка компьютерного программного обеспечения, консультационные услуги в данной области и другие сопутствующие услуги объединены под кодом 62. Данный код и входящие в него виды деятельности в лучшей степени детализирует отрасль. Необходимо отметить несколько подходов к понятию «разработка и внедрение ПО» (далее – software engineering). Представим основные из них в таблице 1.

Таблица 1 – Ключевые подходы к сущности понятия «разработка и внедрение программного обеспечения» (software engineering)

Автор	Определение	Подход
Иэн Соммервилл	Создание спецификации требований, разработка, модификация и сопровождение систем ПО [3, с. 456]	Процессный подход
Ассоциация института электрической и электронной инженерии (ISO/IEC TR 19759:2005)	Систематическое применение инженерных подходов для создания ПО [7, с. 120]	Инженерный подход
Объединенная европейская инициатива по ПО и услугам (NESSI)	Применение принципов области инженерии для дизайна, развития, тестирования, внедрения и менеджмента ПО [6, с. 4]	Инженерный подход

Стив Макконнелл	Использование принципов инженерной науки для обеспечения создания и сопровождения ПО [2, с.25]	Инженерный подход
-----------------	--	-------------------

Исходя из наиболее емкостных определений процесса создания и внедрения ПО, целесообразно выделить два ключевых подхода авторов к данному понятию. При описании непосредственно операций, сопровождающих исследуемое понятие, необходимо говорить о процессном подходе. Если же речь в определении идет о методическом описательном сегменте, то имеет место быть инженерный подход. Многие авторы отмечают, что инженерный подход куда более целостный с описательной точки зрения, поскольку он преподносит software engineering с позиции научного знания. Так, Стив Макконнелл в своем труде «Профессиональная разработка программного обеспечения» говорит о том, что с позиции научного знания инженерный подход более перспективен, так как описывает углубленные методические характеристики, а не поверхностные операции [2, с. 145]. Однако при этом, вспоминая концепцию научной парадигмы Т. Куна, необходимо отметить, что на настоящем этапе теоретические знания в сфере разработки и внедрения ПО имеют признаки неполного методического обеспечения, поскольку наука оперирует имеющимся набором решенных задач вместо устоявшейся инженерной концепции. Однако и сам Кун подчеркивает, что научная парадигма может состоять из набора решенных задач в рамках незначительно «возраста» научного знания [2, с. 120]. В связи с этим остановимся на факте, что на данный момент, как уже отмечалось выше, науки инженерии ПО находится на ранних стадиях развития и будет существенно дополнена с точки зрения качественной составляющей научной парадигмы в ближайшие годы.

Разобравшись со спецификой деятельности предприятий в сфере разработки и внедрения ПО, необходимо перейти к оценке рисков, которые характерны для компаний данного типа. Данный шаг необходим с точки зрения понимания

особенностей производственного процесса и взаимоотношений фирмы с внешним ее окружением. Представим виды рисков, характерных для отрасли в таблице 2.

Таблица 2 – Классификация экономических рисков, характерных для предприятий в сфере разработки и внедрения ПО [1, с. 65]

Внешние риски	Внутренние риски
Риски макроэкономической конъюнктуры: - инфляционный риск; - валютный риск; - системный риск; - процентный риск и т.д.	Коммерческие риски: - ценовые риски; - риски планирования и т.д.
Клиентские риски: - риск банкротства; - договорной риск; - риск системных требований; - титериски; - дебиторские риски и т.д.	Финансовые риски: - риски бюджетирования; - риск финансовой несостоятельности фирмы; - риск ликвидности; - кредитный риск и т.д.
Правовые риски: - риск ужесточения законодательного контроля; - монопольные риски; - риск изменения законодательства и т.д.	Организационные риски - риск текучести сотрудников; - риск управленческих изменений; - риск недооценки и т.д.
Кибер-риски: - риск «кражи» программного продукта; - прочие риски, связанные с несанкционированным вмешательством	Процессные риски: - риск неэффективности Case-инструментов; - риск задержек в разработке; - риск неготовности аппаратных средств; - технологические риски и т.д.
Конкурентные риски: - риск появления аналогичных программных продуктов; - риски целевого хэдхантинга и т.д.	

Более подробно остановимся на ряде наиболее часто возникающих и проблемных рисков для предприятий в отрасли разработки и внедрения ПО.

Среди таких необходимо в первую очередь отметить клиентские риски, кибер-риски, конкурентные риски, а также финансовые, организационные и процессные риски. Во многом эта выборка объясняется контролируемым характером данных рисков (за исключением коммерчески рисков, которые возникают в результате «чистых ошибок»). Рассмотрим подробнее возможные сценарные условия при катализации данных рисков и степень их угрозы для операционной деятельности фирмы, приводимые А. М. Гудовым, С. Ю. Завозкиным и С. Н. Трофимовым [1, с. 56] (рис. 1).

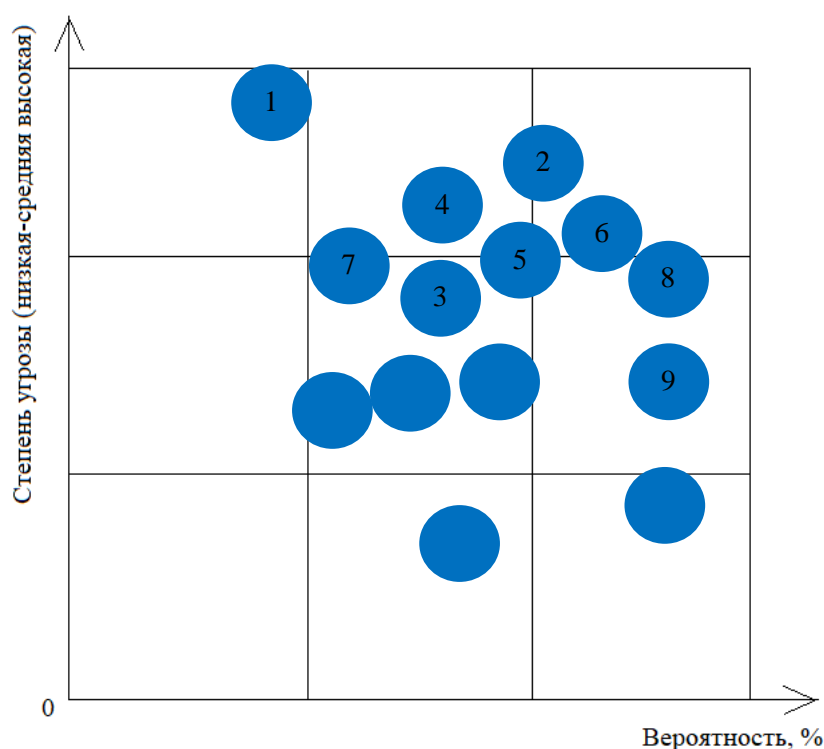
Исходя из данных таблицы 2 и рисунка 1, можно констатировать, что наиболее чувствительными для предприятий рассматриваемой отрасли сегментами становятся:

А. Взаимоотношения с клиентами и планирование процесса реализации проекта;

Б. Менеджмент трудового капитала;

В. Поддержание высокого производственного потенциала;

Стоит обратить особое внимание на сегмент А. Он имеет особое значение с точки зрения организации производственного процесса (в рамках основного вида деятельности) для предприятий в сфере software engineering.



1 – Финансовые трудности и уменьшение бюджета проекта
 2 – Отсутствие работников требуемого уровня
 3 – Отсутствие ключевого сотрудника (болезнь, смена фирмы)
 4 – Дефекты используемых программных компонентов
 5 – Изменение требований заказчика
 6 – Смещение проектных приоритетов вследствие реорганизации в компании
 7 – Нехватка функционала ключевой для ПО базы данных

8 – Недооценка времени выполнения проекта
 9 – Неинтегрируемость Case-средств
 10 – Нечеткая пользовательская формулировка производственного задания
 11 – Невозможность обучения персонала
 12 – Низкая скорость выявления дефектов
 13 – Превышенные размеры системы относительно запланированных
 14 – Неэффективность программного кода

Рис. 1 – Матрица наиболее характерных рисков для компании в сфере разработки и внедрения ПО [1, с. 56]

Необходимо отметить, что сегодня наиболее часто как на отечественных, так и на зарубежных предприятиях данного типа используются механизмы проектного менеджмента, когда весь кадровый состав фирмы, непосредственно занятый производством продукта, подразделяется на проектные команды, каждая из которых сконцентрирована на отдельных программных продуктах. В состав такой группы, как правило, входит руководитель (проектный менеджер) и специалисты узкого и широкого профиля (знание отдельных программных кодов, опыт использования case-средств и т.д.) [1, с. 87]. При этом среди ключевых этапов непосредственного создания и внедрения ПО выделяют:

- 1) подготовку спецификации;
- 2) дизайн, создание проекта и его реализация;
- 3) тестирование;
- 4) качественные усовершенствования;
- 5) дальнейшая эволюция [1, с. 89].

Наиболее часто при организации производственного процесса на предприятиях данного типа на сегодняшний день применяются следующие модели:

1. Каскадная модель (рис. 2).

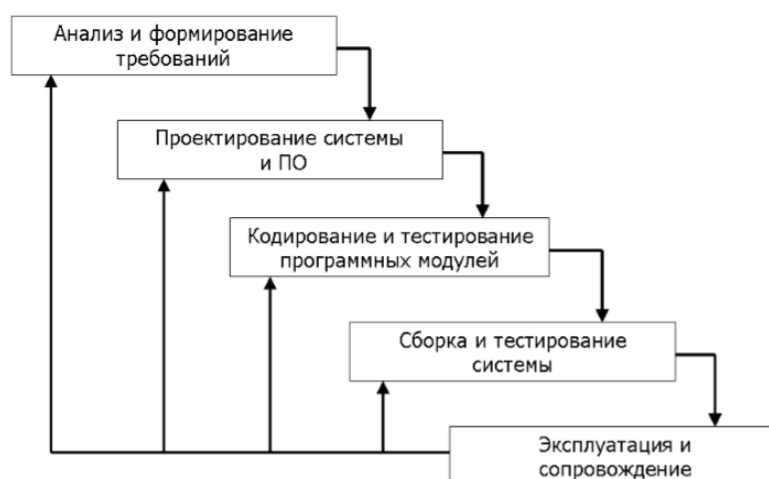


Рис. 2 – Каскадная модель организации software engineering [1, с. 92]

Среди достоинств данной модели можно выделить документирование каждого этапа, что упрощает «формальное» общение с заказчиком и снижает клиентские риски. Однако разбиение процесса на этапы негибкое и не дает свободы исполнителям проекта [1, с. 92]. Такая модель эффективна для небольших подсистем крупных проектов.

2. Эволюционная модель (рис. 3).



Рис. 3 – Эволюционная модель организации software engineering [1, с. 93]

К достоинствам эволюционной модели следует отнести поступательный характер разработки спецификации – по требованию и желанию заказчика. Это уменьшает количество претензий и дефектов конечного продукта, однако существенно увеличивает временные затраты на проект и негативно влияет на структуру ПО, поскольку ее приходится корректировать «на ходу». Такая модель также более приемлема для систем небольшого размера.

3. Модель формальной разработки или формальных преобразований (рис. 4).

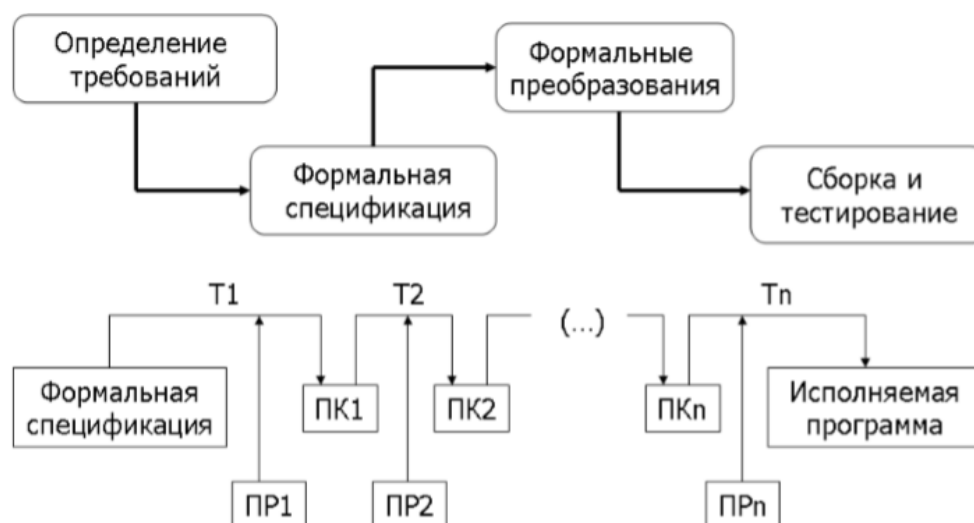


Рис. 4 – Модель формальной разработки [1, с. 94]

Такая система подразумевает целостное тестирование, когда на конечном этапе все подсистемы «соединяются» в единый продукт. Однако такая система сложна с точки зрения описания на конечном этапе и требует серьезного опыта среди сотрудников предприятия-программистов. Помимо этого возможна несочетаемость case-средств на конечном этапе, что приводит к дефектам. Модель формальной разработки применяется лишь для проектирования сложных и многоэтапных систем [1, с. 95].

4. Модель пошаговой разработки (рис. 5).

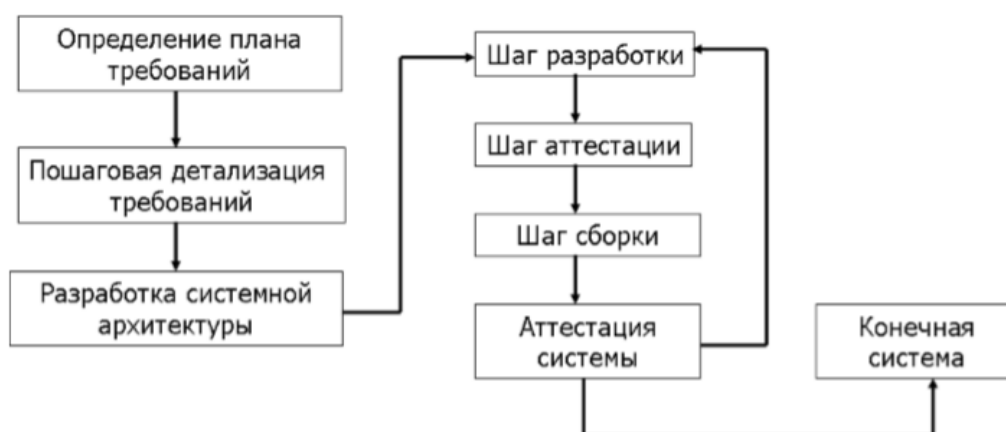


Рис. 5 – Пошаговая модель организации software engineering [1, с. 96]

Преимуществом такой системы выступает наличие возможности развития подсистем без ожидания разработки подсистем более низкого порядка (наличие

системной архитектуры). Однако необходимо отметить, что это зачастую приводят к цепным дефектам, которые необходимо исправлять во всех созданных подсистемах при выявлении ошибки на определенном этапе. При отсутствии согласованности между персоналом предприятия может сложиться ситуация полного несоответствия подсистем, что ведет к повышающейся угрозе time-рисков [1, с. 96]. Поэтому зачастую в тестировании подсистем принимает активное участие персонал заказчика с целью заблаговременного обнаружения дефектов и их оперативного исправления во всех подсистемах. Кроме того, зачастую в рамках данной модели используется метод дробления проектной команды, когда каждый ее член занимается отдельной подсистемой (или, как вариант, определены зоны ответственности программистов).

5. Спиральная модель (рис. 6).

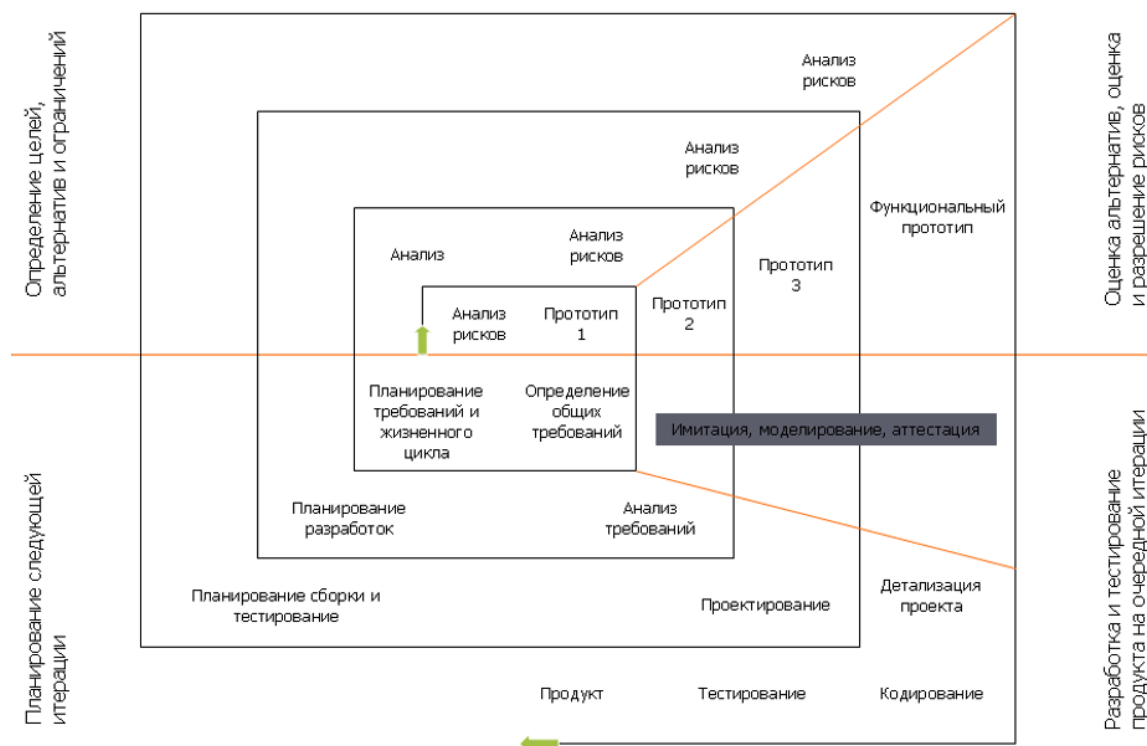


Рис. 6 – Спиральная модель [1, с. 98]

Данная модель признана особенно эффективной в рамках крупных проектов, однако может быть использована и для малых подсистем. Структура

аналогична устоявшейся концепции менеджмента «планирование-организация-мотивация-контроль» и несет черты рационального и прагматического подхода. Несмотря на сложность автоматизации процесса разработки, такая модель может подразумевать адаптацию любых других моделей на отдельных витках спирали. Здесь значительная роль отводится управляющему проектом, роль которого – максимально снизить клиентские и чистые процессные риски за счет организации взаимодействия и грамотному планированию процесса.

Таким образом, нужно подчеркнуть, что сложность и фундаментальность отрасли разработки и внедрения ПО для инновационной экономики предъявляет особо высокие требования к организации производственных процессов на предприятии. Многочисленные риски внешнего и внутреннего характера повышают роль управляющих менеджеров в таких компаниях (проектных менеджеров и руководства), что особенно ярко находит отражение в наиболее эффективной и применяемой сегодня спиральной модель производственного процесса в исследуемой отрасли. Важной проблемой предприятий отрасли также является не проработанность методического содержания научного знания в отрасли *software engineering*, что повышает значимость эмпирических экспериментов и практического знания.

Библиографический список:

1 Гудов А. М. Технология разработки программного обеспечения: учебное пособие / А. М. Гудов, С. Ю. Завозкин, С. Н. Трофимов – Кемерово: Издательство КГУ. – 2009. – 138 с.

2 Макконнелл С. Профессиональная разработка программного обеспечения: учебник / Макконнелл С. – М. : Профессионально. – 2007. – 236 с.

3 Соммервилл И. Инженерия программного обеспечения, 6-е издание / И. Соммервилл – М. : Издательский дом «Вильямс», 2002. – 624 с.

4 Beecham S. Making Software Engineering Research Relevant / S. Beecham, P. O'Leary, S. Baker, I. Richardson, J. Noll // IEEE Computer. – 2014. – 4 (47). – P. 80-83.

5 Freeman R. Labour productivity indicators: comparison of two OECD databases productivity differentials & the Balassa-Samuelson effect / R. Freeman. – Paris: OECD Statistics Directorate, 2016. – 76 p.

6 Software engineering: Key Enabler for Innovation / Bellavista P., Butler J., Franz M. et al.; ed. by P. Bellavista. – New York: NESSI, 2001. – 23 p.

7 Software Engineering – Guide to the Software Engineering Body of Knowledge: ISO/IEC TR 19759:2005. – Int. 2005–09–01. М: ИПК ИЗДАТЕЛЬСТВО СТАНДАРТОВ, 2005 – 187 с.

8 Jumpstarting Scrum with Design Thinking / H. Plattner, A. Back, W. Brenner, R. Jung, H. Österle, R. Winter; ed. by H. Plattner. – St. Gallen: University of St. Gallen, 2015. – 56 p.

Оригинальность 96%