

УДК 651.01+ 005.8 + 330.341

РАЗВИТИЕ МОДЕЛЕЙ ПРОЕКТНОГО УПРАВЛЕНИЯ В УСЛОВИЯХ РОСТА ТУРБУЛЕНТНОСТИ ЭКОНОМИКИ

А.Н. Короленко

Проведена классификация проектов с точки зрения сложности систем, описаны ключевые принципы *agile* и особенности гибкого проектного управления. Предложены возможные варианты взаимодействия заказчика и подрядчика при переходе к гибкому проектному управлению, а также варианты выбора проектной методологии.

Ключевые слова: проект, управление проектами, *agile*, принципы *agile*, элементы *agile*.

Современные экономические условия характеризуются существенными изменениями в поведении потребителей как на *B2B*-рынках, так и на рынках *B2C*. Усиление конкуренции приводит к тому, что значительно увели-

чивается динамика как продуктовых инноваций, так и технологических (скорость и количество), сокращаются жизненные циклы продукции, заказчик требует не только быстрой реакции, но и прослеживается тенденция «кастомизации» (рост доли персональных запросов, индивидуальных требований) спроса [4]. Очевидно, что для достижения указанных результатов необходимо развитие методов и инструментов управления. Не является исключением в этом ряду и проектное управление.

Развитие методологии проектного управления обусловлено изменением характеристик самих проектов и условий их существования в связи с приведенными выше факторами. Согласно теории запутанности (*complexity theory*) и *Synefin framework* мы имеем дело с разными системами: упорядоченные простые, упорядоченные сложные, запутанные системы и хаотичные системы. Вопрос хаотичных систем рассматриваться не будет вследствие сложности формализации (а порой и невозможности на данном этапе развития) процессов управления.

Упорядоченные простые системы – системы, где некое действие X точно ведет к результату Y . То есть цели в таких системах ясны, причинно-следственные связи очевидны, очень высока степень определенности всех факторов. Если ваш проект соответствует таким параметрам, то хорошо работает директивное управление, например, модель «водопада». В этом случае необходимо собрать все требования, сформировать техническое задание, разработать планы высокой степени детализации на весь проект и следовать этим планам.

В упорядоченных сложных системах действие X также ведет к результату Y , но между воздействием и результатом может существовать целая цепочка процессов и событий, которые в некоторой степени определяются ситуацией (в зависимости от того, какой сценарий сработает). С точки зрения проектного управления это может означать, что цели скорее всего ясны, причинно-следственные связи в целом понятны (однако корреляция не жесткая), но зависят от контекста. Здесь хорошо работают «лучшие практики», такие как *PMBOK* (свод знаний по управлению проектами), стандарты *IPMA* (Международной ассоциации управления проектами) и другие. Управление уже представляется нелинейным процессом, хотя можно четко выделять фазы подготовки/инициации, планирования/ структурирования, реализации, контроля и завершения. Другой вопрос, что чем выше степень неопределенности, тем с большей вероятностью эти фазы будут накладываться друг на друга. Однако на этот счет проработаны рекомендации по управлению изменениями, рисками и другими ключевыми процессами.

Наибольший же интерес сегодня представляют «запутанные системы». Эта ситуация, когда мы знаем, что существует связь между параметрами X и Y . Но взаимосвязь для нас не совсем понятна, более того, если вы делаете

X , получая Y , то необходимо учитывать, что полученный параметр (например, результат) Y оказывает влияние на параметр X . То есть с одной стороны имеет место в некоторой степени причинная неопределенность, с другой – взаимовлияние факторов. По сути, все живые организмы, организации относятся к этим системам. С позиции же проектного менеджмента, это означает, что цели до конца не ясны. Не ясно полностью, и как построена система. Например, заказчик понимает, что для продвижения на рынке ему нужен новый веб-ресурс (сайт), но при этом определить точно, какое влияние это окажет на результат, четко сформулировать требования к ожидаемому продукту он не может (как вариант, по причине сложности интеграции всех процессов организации в некое целое). Здесь не работают стандартные методы выявления ожиданий заинтересованных сторон, так как факторов принятия решения большое множество, и они будут зависеть (и меняться) от промежуточных результатов. Отчасти ответом на такие требования и явился *Agile*-подход как гибкий подход к проектному управлению.

Определим ключевые элементы *Agile*-подхода [1]. Здесь сразу нужно обозначить, что сам по себе термин *Agile* часто употребляется некорректно. *Agile* – это все же изначально система ценностей (в 2001 году был разработан *agile*-манифест), а не готовый рабочий инструмент. На основе *Agile*-манифеста развивается комплекс подходов (рис. 1).

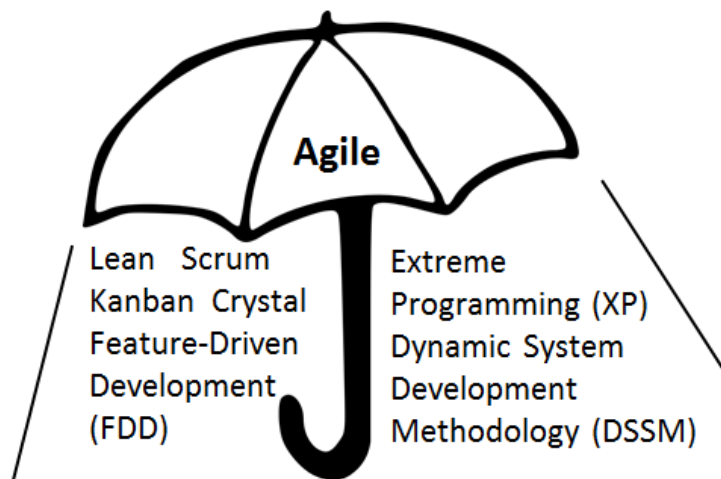


Рис. 1. Инструменты управления на основе *Agile*-манифеста

Так или иначе *Agile*-принципы направлены на:

- снижение объема документации (снижение бюрократии);
- развитие обмена знаниями;
- повышение «прозрачности» проекта;
- снижение давления и стресса участников;
- эффективную координацию;

– повышение производительности.

Многие ключевые инструменты в рамках *agile*-принципов вышли из *IT*-сферы. Тем не менее отдельные практики успешно интегрируются не только в *IT*, но и в другие проекты, и даже в операционную деятельность [3]. *Agile* – это возможность создавать изменения и реагировать на них. Для рассмотрения способов повышения гибкости используем в качестве базиса один из самых распространенных подходов в рамках *Agile* – *SCRUM*-методологию («фреймворк») [5].

Сформулируем основные особенности гибкого подхода:

1. Первое, с чем сталкиваются субъекты принятия решений в «запутанных» системах, – неполное понимание целей и параметров требуемого продукта. Формализация этих требований в «классических» подходах предполагает их составление и фиксацию на весь срок проекта. При гибком же подходе формируется список требований (ключевых заинтересованных лиц) – бэклог продукта, который может меняться в ходе проекта.

2. Сам процесс планирования делится на итерации определенной фиксированной длины (обычно от недели до четырех), что позволяет после каждой итерации вносить коррективы. Очевидно, что и работа по проекту делится в этом случае на «маленькие» компоненты, которые оцениваются и приоритизируются.

3. В конце каждой итерации промежуточный результат (рабочий продукт) демонстрируется заказчику, по результатам чего владелец продукта (человек, отвечающий за формирование бэклога продукта) может менять список требований и приоритеты с учетом полученных результатов.

4. При планировании каждой итерации отбираются задачи, которые будут выполнены. В случае невыполнения задач итерация не продлевается. Невыполненные задачи возвращаются в бэклог продукта, при этом не всегда они могут попасть в следующую итерацию.

5. Повышение прозрачности достигается за счет использования средств визуализации, например, это могут быть доски задач (для каждой итерации), содержащие элементы: «необходимо сделать», «в работе», «выполнено». Задачи на итерацию не могут добавляться в ходе итерации.

6. Команды формируются из людей с широким разнообразием навыков (кросс-функциональные команды). Поскольку команды отвечают за взятые на себя обязательства в рамках итерации, то им дается автономия, возможность самоорганизации и самоуправления (в рамках итерации). Ежедневно проводятся короткие собрания для оценки прогресса выполнения работ. В конце каждой итерации организуются ретроспективные встречи для рефлексии и анализа того, что можно улучшить на следующих этапах. Метрикой для команд является производительность [2].

7. Заказчик вовлечен в проект и принимает в нем активное участие (либо непосредственно с «владельцем продукта», либо через него).

8. По сравнению с «классическим» управлением для упорядоченных систем предполагаются введение новых ролей (например, для SCRUM – это владелец продукта, скрам-мастер).

Указанные особенности выводят на первый план как минимум два аспекта – это готовность компании к гибким подходам (поскольку требуется переосмысление многих принципов работы, да и в целом изменение менталитета сотрудников) и готовность (а иногда и возможность) к изменению принципов взаимодействия заказчика с подрядчиком.

Что касается доверия со стороны заказчика, то можно предложить постепенный переход в моделях ценообразования:

- фиксированная цена и объем работ (нет доверия);
- фиксированная цена на первый этап, потом по принципу *Time & Material* (оплата идет не за объем работы, а за человеко-часы команды подрядчика);
- фиксированный бюджет (окончательный объем работ определяется в ходе проекта);
- принцип *Time & Material* (высокое доверие).

Говоря о готовности организации к гибким подходам в управлении, необходимо отметить, что возможно частичное применение *agile*-принципов, например, в рамках пакетов задач (когда строится укрупненный план проекта традиционными методами, а внутри задач применяются итерационные гибкий принципы). В итоге можно предложить схему выбора подходов к управлению (рис. 2).

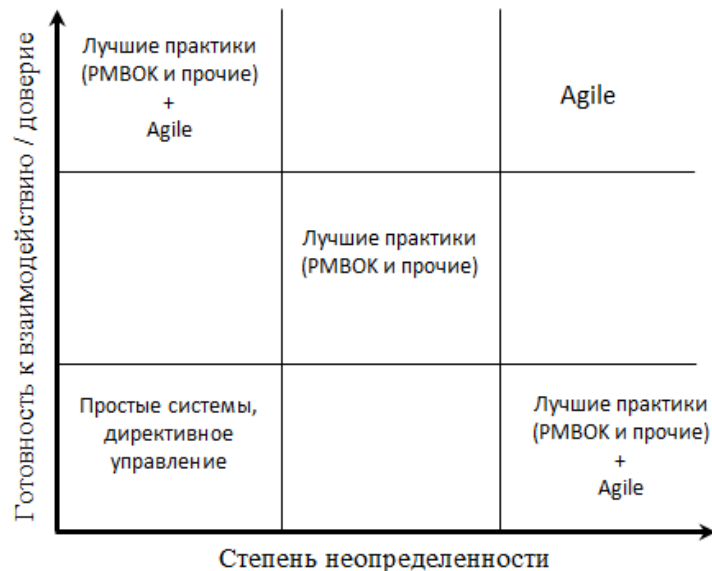


Рис. 2. Выбор подходов к управлению проектами

Таким образом, для достижения лучших результатов необходимо иметь набор методик и инструментов, использование которых будет определяться спецификой проектов и уровнем развития компании и персонала.

Библиографический список

1. Agile-манифест [Электронный ресурс]. – URL: <http://agilemanifesto.org/iso/ru/>.
2. Breivold, P.H. What Does Research Say About Agile and Architecture? / P.H. Breivold, D. Sundmark, P. Wallin, S. Larsson. – Fifth International Conference on Software Engineering Advances, 2010.
3. Демиденко, М.А. Модель управления инновационным проектом в контексте общей системы управления предприятием / М.А. Демиденко // Известия Ростовского государственного строительного университета. – 2012. – № 16. – С. 153–154.
4. Короленко, А.Н. Управление изменениями на инновационно-активном предприятии / А.Н. Короленко // Вестник ЮУрГУ. Серия: «Экономика и менеджмент». – 2013. – Т. 7. – № 4. – С. 194–196.
5. Schwaber, K. Agile Project Management with Scrum / K. Schwaber. – Redmond, Washington: Microsoft Press, 2004.

[К содержанию](#)