

О МЕХАНИЗМЕ СТИМУЛИРОВАНИЯ РАЗРАБОТЧИКОВ ИТ-СИСТЕМ

УДК 004.051

Алексей Сергеевич Добрынин,
зав. лаб. кафедры автоматизации
и информационных систем, старший
преподаватель
Сибирский государственный индустриаль-
ный университет (СибГИУ)
Тел.: (3843) 74-88-06
Эл. почта: serpentfly@mail.ru

Роман Сергеевич Койнов,
ведущий специалист по информатизации,
старший преподаватель
Сибирский государственный индустриаль-
ный университет (СибГИУ)
Тел.: (3843) 74-88-06
Эл. почта: koynov_rs@mail.ru

Александр Викторович Грачев,
старший преподаватель кафедры при-
кладной информатики и программиро-
вания
Сибирский государственный индустриаль-
ный университет (СибГИУ)
Тел.: (3843) 46-35-02
Эл. почта: ansel@zaoproxy.ru

Виталий Валерьевич Митьков,
старший преподаватель
Сибирский государственный индустриаль-
ный университет (СибГИУ)
Тел.: (3843) 74-88-06
Эл. почта: vita-m@ngs.ru

Повышение эффективности деятельности разработчиков аппаратно-программных комплексов и ИТ-систем неразрывно связано с качественным стимулированием. Люди, мотивированные на труд, способны добиваться высоких результатов, ориентируясь на достойное вознаграждение. Непродуманные и нерациональные подходы к оплате труда способны породить социальные проблемы и значительно снизить общую эффективность функционирования организационных систем. Мотивация мобильных команд ИТ-разработчиков и лиц, осуществляющих оперативное руководство выполнением проекта, имеет первостепенное значение. Статья рассматривает вопросы синтеза механизмов стимулирования разработчиков ИТ-систем, исходя из принципа согласованного управления, что позволяет сближать интересы заказчиков и исполнителей. Предполагается, что заказчики и исполнители заинтересованы в создании всех необходимых условий (инвестиционных, социальных, технических) для получения качественного итогового продукта. Заказчик договаривается с разработчиком о примерном количестве проектных итераций, составе каждой итерации, сроках и бюджете, необходимом для получения требуемого результата. Средствами оценки эффективности полученных бизнес-результатов являются метрики, тесты, в том числе функциональные и модульные (unit tests), составленные в непосредственном контакте представителей заказчика и разработчика. Рассматриваемый механизм целесообразно использовать в мобильных командах, в которых разработка осуществляется относительно короткими итерациями, с поэтапным уточнением бизнес-требований.

Ключевые слова: стимулирование, разработка ИТ-систем, принцип согласованного управления, итерация разработки.

1. Условия взаимодействия участников

Договорные отношения, между заказчиками и разработчиками программного обеспечения должны быть гибкими, то есть допускать внесение изменений в проект на протяжении его жизненного цикла. Получение конкурентных преимуществ в условиях рынка вынуждает всех его участников постоянно обновлять продукцию (услуги), искать новые пути повышения эффективности своей деятельности и качества используемых инструментов. Бизнес вынужден функционировать в жестких условиях конкуренции и внешних возмущений, зачастую катастрофических (изменения законодательства, разрушительные действия других игроков и т.д.), которые приводят к необходимости пересмотра требований к ИТ-системе даже на стадиях внедрения и эксплуатации. В контексте эффективного взаимодействия заказчики и разработчики должны придерживаться «правил игры», которые позволяют бизнесу постоянно изменяться. Более того, разработчики должны приветствовать постоянное стремление бизнеса к изменениям, поскольку сами стремятся к оптимизации своей деятельности в рыночных условиях.

Мировоззренческий аспект договорных отношений предусматривает смещение фокуса внимания от некоего процесса (например, процесса разработки), который имеет начало и конец с набором метрик оценивания – к деятельности (взаимодействию), когда каждый участник системы «заказчик-разработчик» изыскивает способы наиболее эффективного взаимодействия для постоянного уточнения общего понимания важнейших проблем и задач. Подобного рода интерпретация целесообразна, так как «классический» процессный подход часто манипулируем, когда каждый участник стремится установить метрики «под себя» и достичь результата кратчайшим способом, игнорируя зачастую требования качества. Получение качественного результата в ИТ-разработке связано с тяжелой и кропотливой ежедневной работой, с постоянной актуализацией моделей и рефакторингом кода, с профилированием. Это в редких случаях находит отражение в метриках, формируемых бизнесом, поскольку бизнес-аналитики предпочитают не вдаваться в технические детали. Например, неудачно спроектированный дизайн может прекрасно работать в краткосрочной перспективе, для одной частной задачи и оказаться совершенно непригодным при небольшом изменении как системного ландшафта, так и отдельных бизнес-процессов (особенно, при государственном вмешательстве). Необходимость повседневного, постоянного рефакторинга моделей и кода становится все более очевидной, по мере роста технической сложности проектов.

Учитывая изложенное необходимо отметить, что механизм взаимодействия заказчиков и разработчиков ИТ-систем должен удовлетворять следующим основным условиям:

а) Разработчики и заказчики всегда должны быть готовы к изменениям на любых стадиях проекта. Разработчики не должны создавать препятствия бизнесу при реализации изменений.

б) ИТ-разработка должна рассматриваться, с одной стороны, как процесс характеризующийся набором сопутствующих метрик, с другой – как *деятельность* по итерационному уточнению бизнес-требований и достижению необходимой функциональности на текущей итерации. При этом отдельная итерация может не достигать желаемых значений всех метрик (или их подмножества) за один раз.

в) *Бюджет, сроки и результаты*, а также *методы их оценивания* (модульные, приемочные тесты, метрики и т.д.) должны оговариваться между представителями заказчика и разработчика для каждой итерации отдельно. Переход на новую итерацию выполняется по мере уточнения (или изменения) требований.

Alexey S. Dobrynin,
Degree-seeking student, senior teacher
Siberian State Industrial University (SibSIU)
Tel.: (3843) 74-88-06
E-mail: serpentfly@mail.ru

Roman S. Koynov,
Degree-seeking student, senior teacher
Siberian State Industrial University (SibSIU)
Tel.: (3843) 74-88-06
E-mail: koynov_rs@mail.ru

Aleksandr V. Grachev,
Degree-seeking student, senior teacher
Siberian State Industrial University (SibSIU)
Tel.: (3843) 46-35-02
E-mail: ansel@zaoproxy.ru

Vitaliy V. Mitkov,
Degree-seeking student, senior teacher
Siberian State Industrial University (SibSIU)
Tel.: (3843) 74-88-06
E-mail: vita-m@ngs.ru

ABOUT THE MECHANISM OF FINANCIAL INCENTIVE FOR THE DEVELOPERS OF IT SYSTEMS

Improving the performance of the developers of hardware and software systems and IT systems it is inextricably linked with quality incentives. People are motivated to work and are able to achieve good results, focusing on decent remuneration. Non-rational approaches to remuneration are able to generate social problems and greatly reduce the overall efficiency of the organizational systems. Motivation mobile teams of IT developers and individuals engaged in the operational management of the project implementation is of paramount importance. The article examines the mechanisms for stimulating the synthesis development of IT systems based on the principle of coordinated control that allows you to bring together the interests of the customers and performers. It is expected that customers and performers interested in creating all the necessary conditions (investment, social, technological) to obtain high-quality final product. The customer agrees with the developer on the approximate number of design iterations, each iteration of the composition, timing and budgets required to achieve the desired result. By means of evaluating the effectiveness of business results are metric tests, including functional and modular (unit tests), created in direct contact representatives of the customer and the developer. This mechanism should be used in mobile teams, in which the development is relatively short iterations with stepwise refinement of business requirements.

Keywords: incentives, the development of IT systems, the principle of coordinated management, development iterations.

в) Достижение всех требуемых значений метрик в процессе разработки на определенном этапе не должно приводить к прекращению деятельности, поскольку разработка нетривиальных систем всегда сопряжена с высокими рисками (риски масштабируемости, отказоустойчивости и т.д.).

г) Условия прекращения деятельности системы «заказчик-разработчик» должны согласовываться отдельно, при этом заказчик испытывает систему в различных условиях работы, опрашивает непосредственных пользователей, формирует списки пожеланий и претензий.

Гибкие технологии разработки программного обеспечения, такие как scum [1], XP[2], уделяют внимание вопросам модульно-итеративной разработки приложений, концентрируясь на бизнес-целях и технических аспектах построения модульных, расширяющихся архитектур. Более подробная информация о предлагаемой авторами неполной итеративной модели разработки программного обеспечения приведена в работе [3].

2. Принцип открытого (согласованного) управления

Теория активных систем [4] и современное ее представление – теория управления организационными системами (ТУОС) [5] рассматривают закон открытого (согласованного) управления, который позволяет обеспечить эффективное неманипулируемое управление деятельностью людей в сложной организационной системе. Основной отличительной стороной этого инструмента является идея согласованного управления, которая позволяет руководящему центру формировать управляющие воздействия с учетом целевых

функций (функций предпочтения) подчиненных ему агентов.

Рассмотрим два основных инструмента согласованного управления в активных системах, а именно принцип (закон) открытого управления и механизм согласованного стимулирования с точки зрения возможности их применения в производственных и экономических задачах [6], в частности, построения систем стимулирования, применительно к ИТ-разработчикам. Сложность таких систем обусловлена в первую очередь трудностями организации (и оценки результатов) трудового процесса людей, задействованных в изготовлении высокотехнологического продукта. Типовая двухуровневая многоэлементная активная (организационная) система, рассматриваемая в теории активных систем, имеет вид, представленный на рисунке 1.

На рисунке показаны целевые функции отдельных агентов (мобильных коллективов разработчиков) обозначаемые как $\varphi_n(x_n, J_n)$, целевая функция управляющего центра (заказчика) $\Phi(\pi, J)$.

В соответствии с принципом согласованного управления [5], используются следующие обозначения: $x_1, \dots, x_i, \dots, x_n$ – устанавливаемые центром планы (из числа допустимых планов X_j) для агентов $AG_1, \dots, AG_i, \dots, AG_n$; $\varphi_1(x_1, J_1), \dots, \varphi_i(x_i, J_i), \dots, \varphi_n(x_n, J_n)$ – функции предпочтения агентов; $\Phi(\pi, J)$ – целевая функция центра; $J = (J_1, \dots, J_i, \dots, J_n) \in \Omega = \prod_{i \in N} \Omega_i, i \in N$ – вектор сведений (сообщений) от агентов, которые необходимы центру для планирования; $N = \{1, 2, \dots, i, \dots, n\}$ – множество агентов; $y_1, \dots, y_i, \dots, y_n$ – действия (по сути, результаты деятельности) агентов $\{AG_i, i = \overline{1, n}\}$. Процедура

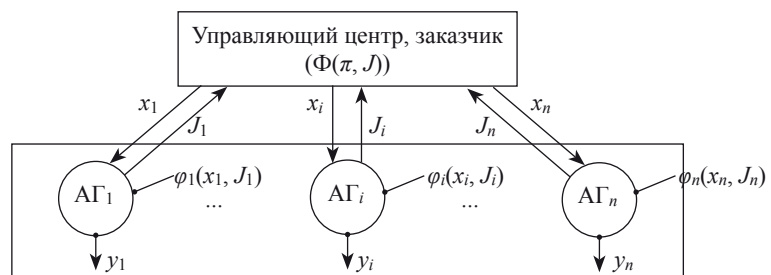


Рис. 1. Структура 2х-уровневой активной (организационной) системы

(механизм) π_i планирования, реализуемая центром, устанавливает соответствие между планом x_i и вектором J сообщений:

$$x_i = \pi_i(J) \in X_i, \text{ где } \pi_i: \Omega \rightarrow X_i, \\ i \in N; \Omega_{i \in N} = \prod \Omega_i \quad (1)$$

Принцип (закон) открытого (согласованного) управления есть такая процедура $\pi_i(J)$ планирования, которая максимизирует целевую функцию $\Phi(\pi, J)$ центра на множестве допустимых X_i планов, которые соответствуют условию совершенного согласования:

$$\varphi_i(\pi_i(y), J_i) = \max \varphi_i(x_i, J_i), i \in N, \\ J \in \Omega, x_i \in X_i(J_i) \quad (2)$$

где $X_i(J_i)$ – множество допустимых планов x_i при заданной обстановке $J_i = (J_1, \dots, J_{i-1}, \dots, J_n)$ в системе для агента $AG_i, i \in N$. Доказано [5], что применение принципа открытого управления делает сообщение каждым агентом AG_i достоверной информации J_i центру главной (основной) стратегией информирования центра. Тем самым достигается взаимная выгода и высокая эффективность взаимодействия центра и агентов.

Механизм согласованного стимулирования агентов $\{AG_i\}$ основан на применении такой функции стимулирования $\sigma(y_i, x_i)$ для каждого из них, которая достигает максимума только при выполнении AG_i запланированного результата x_i , который сформирован на основе принципа согласованного управления.

Опираясь на кратко описанные инструменты открытого (согласованного) управления можно сформулировать следующее заключение. В теории активных систем и современной теории управления организационными системами механизмы согласованного управления представлены в двух взаимосвязанных формах:

– В форме принципа (закона) открытого (согласованного) планирования, обеспечивающего оптимальность планов по критерию максимизации целевой функции $\Phi(\pi, J)$ центра, и неманипулируемость процедуры планирования, то есть достоверность сведений J , получаемых от агентов и используемых

в центре при планировании.

– В форме механизма согласованного стимулирования агентов, побуждающего последних к обязательному достижению запланированных результатов, которые определены центром с использованием сведений J представленных агентами.

Совместное применение закона открытого управления и механизма согласованного стимулирования (при определенных предпосылках) обеспечивает компромиссную оптимизацию целевых функций центра и агентов. Рассмотрим вопросы компромиссной оптимизации и синтеза механизма стимулирования на основе закона открытого управления в следующем разделе.

Механизм стимулирования для отдельной итерации разработки

Рассматриваемые механизмы стимулирования разработаны применительно к неполной модели жизненного цикла программного обеспечения, см. [3]. В рамках этой модели полагаем, что разработка осуществляется итерациями, для каждой из которых согласуются: *бюджет итерации (изменения) B^* , сроки итерации T^*, T^{Max} , оценочные средства* (в т.ч., т.н. метрики) M^* . Заметим, что оценочные средства не всегда могут иметь численное выражение. Следуя общепринятым современным практикам разработки (в частности, разработки с применением тестирования – TDD [2]) оценочные средства могут быть реализованы в виде определенных булевых семантических выражений любой степени сложности (т.н. тестов) со значениями «истина», «ложь». Такие тесты, как правило, служат для оценки реализованной в проекте функциональности в различных информационных «срезах», а также измерения времени исполнения (производительности) отдельных участков кода.

Предлагаемые в статье функции стимулирования отдельных агентов $AG_i \in AG$ и центра синтезированы с учетом требований компромиссной оптимизации и согласованного управления, рассмотренного выше. Организационный механизм совершенного согласования вклю-

чает в себя следующие основные этапы:

1) Руководящий центр (заказчик) в условиях неопределенности формирует свое «видение» ИТ-системы в целом и договаривается с разработчиками об общем (примерном) количестве итераций.

2) Центр (заказчик) в условиях неопределенности формирует бизнес-требования и запрашивает у разработчиков сообщения (сведения) J о бюджете B^* и сроках итерации T^*, T^{Max} . Разработчики, опираясь на собственный опыт, сообщают центру сроки и стоимость итерации.

3) Центр согласовывает с разработчиками оценочные средства (метрики, тесты и т.д.) или назначает их в одностороннем порядке (в условиях абсолютной уверенности в их адекватности текущей ситуации).

4) Центр разрабатывает (или использует готовый) механизм стимулирования одной итерации с позиций компромиссной оптимизации и согласует элементы этого механизма с ИТ-разработчиками (согласует значения отдельных коэффициентов).

В статье авторы исходят из достаточно очевидного предположения, что качество стоит денег и центр не должен «жадничать», инвестируя денежные средства в разработку и стимулирование команды ИТ-специалистов. Суть предлагаемого механизма стимулирования заключается в согласовании доли сэкономленной части бюджета, распределяемой между центром и разработчиками посредством указания коэффициента α . Функция стимулирования отдельного агента $AG_i \in AG$ имеет следующий вид (3):

$$\sigma_i(y_i) = \alpha(t) \cdot (B^* - B) \rightarrow \text{Max} \quad (3)$$

где $\alpha(t)$ – зависящий от времени коэффициент, равный нулю, при $t \geq T^{Max}$. При $t < T^{Max}$, значение $\alpha(t)$ выбирается из диапазона $[0, 1]$, в процессе согласования заказчика и разработчиков ИТ-систем. B^* – согласованный бюджет, B – потраченный бюджет. $\sigma_i(y_i)$ – функция стимулирования i -го коллектива исполнителей, зависящая непосредственно от их действий y .

Функция стимулирования центра представлена выражениями (4), (5):

$$\sigma(\Phi(\pi, J)) = \sum_{i=1}^N (1 - \alpha_i(t)) (B_i^* - B_i) \rightarrow (Max) | Min \quad (4)$$

$$\sigma(\Phi(\pi, J)) = (1 - \alpha(t)) \sum_{i=1}^N (B_i^* - B_i) \rightarrow (Max) | Min \quad (5)$$

по сути, выражения (4), (5) практически идентичны друг другу, за исключением того, что выражение (5) используется в ситуациях, когда для каждого i -го разработчика используется одинаковый коэффициент α .

Заключение

Авторами предложен механизм стимулирования, пригодный для мобильных коллективов ИТ-разработчиков, использующих гибкие (итеративные) методологии разработки программного обеспечения в повседневной деятельности, такие как scrum [1], xp [2]. **Научная новизна** и принципиальная особенность данного механизма заключается в использовании элементов согла-

сованного управления [5] для ком-промиссной оптимизации затрат при реализации проектов высокой сложности и риска.

Литература

1. Майк Кон. Scrum: Succeeding with Agile: Software Development Using Scrum (Addison-Wesley Signature Series). – М.: «Вильямс», 2011. – С. 576. – ISBN 978-5-8459-1731-7.
2. Kent Beck «Test-Driven Development by Example», Addison-Wesley, 2003.
3. Добрынин А.С. Модель неполного жизненного цикла программного обеспечения / А.С. Добрынин, Р.С. Койнов, С.М. Кулаков // Вестник АГТУ. Сер., Управление, вычислительная техника и информатика. – 2015. – № 2. – С. 65–70. – Библиогр.: с. 69 (3 назв.).
4. В.Н. Бурков. Теория активных систем: состояние и перспективы. / Бурков В.Н., Новиков Д.А. – М.: Синтег, 1999. – 128 с. ил.
5. В.Н. Бурков, Н.А. Коргин, Д.А. Новиков. Введение в теорию управления организационными системами / Под ред. чл.-корр. РАН Д.А. Новикова. – М.: Либроком, 2009. – 264 с.
6. Е. Н. Ломкова, А. А. Эпов. Экономико-математические модели

управления производством (теоретические аспекты): Учебное пособие/ ВолгГТУ, Волгоград, 2005. – 67 с.

References

1. Majk Kon. Scrum: Succeeding with Agile: Software Development Using Scrum (Addison-Wesley Signature Series). – M.: «Vil'jams», 2011. – S. 576. – ISBN 978-5-8459-1731-7.
2. Kent Beck «Test-Driven Development by Example», Addison-Wesley, 2003.
3. Dobrynin A.S. Model nepolnogo zhiznennogo cikla programmnogo obespechenija / A.S. Dobrynin, R.S. Koynov, S.M. Kulakov // Vestnik AGTU. Ser., Upravlenie, vychislitel'naja tehnika i informatika. – 2015. – № 2. – S. 65–70. – Bibliogr.: s. 69 (3 nazv.).
4. V.N. Burkov. Teorija aktivnyh sistem: sostojanie i perspektivy. / Burkov V.N., Novikov D.A. – M.: Sinteg, 1999. – 128 s. il.
5. V.N. Burkov, N.A. Korgin, D.A. Novikov. Vvedenie v teoriju upravlenija organizacionnymi sistemami / Pod red. chl.-korr. RAN D.A. Novikova. – M.: Librokom, 2009. – 264 s.
6. E. N. Lomkova, A. A. Jepov. Jekonomiko-matematicheskie modeli upravlenija proizvodstvom (teoreticheskie aspekty): Uchebnoe posobie/ VolgGTU, Volgograd, 2005. – 67 s.