

МЕТОД СРАВНЕНИЯ СТРУКТУРЫ БИЗНЕС-ПРОЦЕССОВ ПУТЕМ НАХОЖДЕНИЯ ЛОГИКО-ТЕРМАЛЬНОЙ ЭКВИВАЛЕНТНОСТИ ИХ СХЕМ

А. А. Рындин*, С. В. Сапегин*, С. И. Хабаров**

**Воронежский государственный университет*

***Воронежский государственный технический университет*

Поступила в редакцию 14.11.2011 г.

Аннотация. Рассматриваются особенности бизнес-моделирования, как этапа разработки корпоративных ИС. Исследуются проблемы удешевления автоматизации уникальных бизнес-процессов заказчика на основе использования типовых программных решений.

Ключевые слова: бизнес-процессы, Internet-приложения, схемы программ, логико-термальная эквивалентность

Annotation. In article the features of business-modelling inside the process of CIS development are considered. Problems of obtaining a well-fit CIS for a specific business process without business reengineering or designing a new custom CIS are considered

Key words: business-process corporative IS, program schemes equivalence.

Одним из наиболее перспективных и динамично развивающихся рынков в сфере разработки программного обеспечения является рынок Internet-приложений. Современные Internet-приложения, построенные на базе концепции RIA, позволяют обеспечить пользователю функциональность, сравнимую с традиционными Desktop-приложениями, а простота развертывания и масштабирования таких систем обеспечивает их неоспоримые конкурентные преимущества.

Работа в области разработки Web-решений для корпоративных клиентов обычно связана со следующими специфическими особенностями:

1. Высокая конкуренция среди разработчиков, обусловленная популярностью в этом сегменте механизмов удаленной работы.

2. Широкие возможности интеграции отдельных компонентов Web-решений в единую систему, как на уровне баз данных и программных интерфейсов, так и на уровне WWW/HTTP.

3. Большой выбор компонентов и модулей сторонних производителей, включая свободно распространяемое ПО с открытым кодом.

Все эти особенности обуславливают необходимость повышения производительности программиста при разработке Web-решений за счет увеличения процента использования в них уже

разработанных компонентов, модулей и даже отдельных сервисов.

Первым шагом автоматизации предприятия за счет использования Web-решений является исследование и реинжиниринг его бизнес-процессов, ориентированных на использование Internet-технологий. При этом, используемые для оптимизации бизнес-процессов подходы базируются на постулируемом утверждении о безусловной первичности бизнес-процессов перед разрабатываемыми автоматизированными решениями. Исходя из этого, построение оптимальных бизнес-процессов ведется с минимальным привлечением информации о существующих технологиях, а полученная модель бизнес-процессов передается разработчикам для автоматизации.

Вместе с тем, как показывает практика, использование ранее разработанных компонентов позволяет в некоторых случаях в разы сократить материальные и трудовые затраты при автоматизации бизнес-процессов. Этим фактом часто пользуются предприятия, уже имеющие готовый набор решений и предлагающие его внедрить в похожих случаях, апеллируя к величине требуемых издержек. При этом возможны случаи, когда изменение оригинального бизнес-процесса с целью использования для его автоматизации готового решения может серьезно отразиться на эффективности работы предприятия, его конкурентос-

пособности. Поэтому, успешное внедрение и использование в работе предприятий автоматизированных систем невозможно без тщательного исследования организационной структуры предприятия, выявления характера и состава бизнес-процессов с учетом реальной деятельности предприятия.

Одним из наиболее важных вопросов, возникающих в процессе исследования предприятия с целью его дальнейшей автоматизации является вопрос о принципиальном сходстве (взаимозаменяемости) двух произвольных бизнес-процессов. Если рассматривать бизнес-процессы предприятия в виде определенным образом связанных друг с другом технологических операций (т.е. в виде алгоритмов обработки тех или иных событий), то для исследования структуры бизнес-процессов возможно использовать математический аппарат теории схем программ, разрабатывавшейся для изучения свойств алгоритмов. Тогда, для исследуемых бизнес-процессов, как и для любых алгоритмов, можно построить схему программ, допускающую анализ и сравнение бизнес-процесса с себе подобными. При этом, схема программ — это математическая модель алгоритма, представляющая собой описание на некоем формальном языке. В этом описании все индивидуальные операторы и функции заменены абстрактными символами переменных-операций и переменных-предикатов. Схема не задает алгоритма, с ее помощью нельзя вычислять что-либо. Она описывает строение программы, или, более точно, строение множества программ, получающихся из схемы в результате задания различных интерпретаций. Интерпретируя по-разному переменные, функциональные и предикатные символы, можно получить совершенно разные реализации схемы, которые, тем не менее, являются принципиально сходными. В случае исследования бизнес-процессов при составлении их схем можно игнорировать подробности преобразования информации и ресурсов внутри элементарных технологических операций.

Для сравнения схем программ обычно используется понятие эквивалентности, основанное на совпадении множеств историй всех конечных цепочек. Наиболее удобно в этом случае использовать определение логико-термальной эквивалентности, основанной на понятии логико-термальной истории. Определим термальное

значение переменной x для конечного пути w схемы S как терм $t(w, x)$, который строится следующим образом.

1. Если путь w содержит только один оператор A , то

$$t(w, x) = \left\{ \begin{array}{l} \tau, \text{ если } A - \text{ оператор присваивания } \chi := \tau \\ \chi, \text{ в остальных случаях} \end{array} \right\}$$

2. Если $w = w' Ae$, где A — оператор, e — выходящая из него дуга, w' — непустой путь, ведущий к A , а x_1, \dots, x_n — все переменные термина $t(Ae, x)$, то $t(w, x) = t(Ae, x) [t(w', x_1)/x_1, \dots, t(w', x_n)/x_n]$.

3. Понятие термального значения распространим на произвольные термы τ : если x_1, \dots, x_n — все переменные термина, то положим

$$t(w, \tau) = [t(w, x_1)/x_1, \dots, t(w, x_n)/x_n].$$

Для пути w в некоторой стандартной схеме S определим её логико-термальную историю $lt(S, w)$ как слово, которое строится следующим образом:

1. Если путь w не содержит распознавателей и заключительной вершины, то $lt(S, w)$ — пустое слово.

2. Если $w = w' ve$, где v — распознаватель с тестом $p(\tau_1, \dots, \tau_k)$, а e — выходящая из него дуга, $\Delta \in \{0, 1\}$, то

$$lt(S, w) = lt(S, w') p \Delta (t(w', \tau_1), \dots, t(w', \tau_k))$$

3. Если $w = w' v \vee$ — заключительная вершина с оператором стоп (τ_1, \dots, τ_k) , то $lt(S, w) = lt(S, w') (t(w', \tau_1), \dots, t(w', \tau_k))$

Детерминантом ($det(S)$) стандартной схемы S называют множество логико-термальных историй всех её цепочек, завершающихся заключительным оператором. Схемы S_1 и S_2 называются логико-термально эквивалентными (сокращённо *лт-эквивалентными*), обозначение $S_1 S_2$, если их детерминанты совпадают.

ЛТ-эквивалентность имеет существенное значение с практической точки зрения по следующим причинам:

1) ЛТ-эквивалентность корректна. Это означает, что если мы ограничимся преобразованиями, сохраняющими ЛТ-эквивалентность, то такие преобразования не будут нарушать и функциональную эквивалентность схем

2) ЛТ-эквивалентность, в отличие от функциональной эквивалентности, является разрешимой. Более того, существует относительно несложный алгоритм распознавания ЛТ-эквивалентности стандартных схем

3) Класс схем, ЛТ-эквивалентных заданной схеме, является достаточно «широким», чтобы допускать такие преобразования, как перестановку преобразователей на линейных участках (т.е. фрагментах, не содержащих распознавателей), расщепление операторов, вставку и удаление неиспользуемых преобразователей.

4) ЛТ-эквивалентность не допускает существенной перестройки логической структуры схемы. В применении к бизнес-процессам это означает, что бизнес-процессы, схемы которых ЛТ-эквивалентны, настолько похожи, что возможен перенос практически любых управленческих и технических решений с одного процесса на другой.

Существует набор преобразований, сохраняющих ЛТ-эквивалентность схемы. К ним относятся: удаление фрагментов без входов; приравнивание фрагмента без выходов оператору «петля»; копирование или склеивание копий преобразователей; замена незацепленных переменных; удаление неиспользуемых преобразователей; замена переменных; удаление неиспользуемого результата входа. Существует также алгоритм распознавания ЛТ-эквивалентности схем программ.

Рассмотрим на конкретном примере практическую применимость математического аппарата схем программ в рамках автоматизации бизнес-процессов. Исходной задачей примера является автоматизация работы агентства по созданию сайтов Б для повышения эффективности его работы. Отличительная особенность данного агентства — ориентация на юзабилити-тестирование своих работ и тщательный подход к проектированию пользовательских интерфейсов. Сроки и бюджет разработки КИС исключают создание полностью оригинального решения. В портфолио компании-разработчика есть два похожих проекта — автоматизация небольшой веб-студии А и филиала крупной компании В, занимающейся производством программного обеспечения. Рассмотрим бизнес-процессы, происходящие в организациях А, Б и В в терминах схем программ.

Построим приведенные схемы для всех трех бизнес-процессов. Приведенная схема не должна содержать преобразователей, к которым ведет больше одной дуги. Это достигается путем использования ЛТ-эквивалентного преобразования. Теперь становится очевидно, что схемы Б и

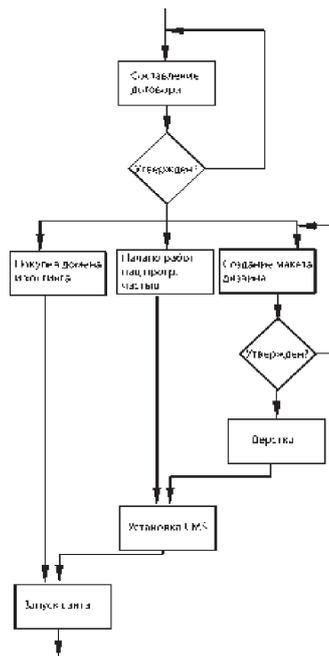


Рис. 1. Бизнес-процесс создания сайта в студии А

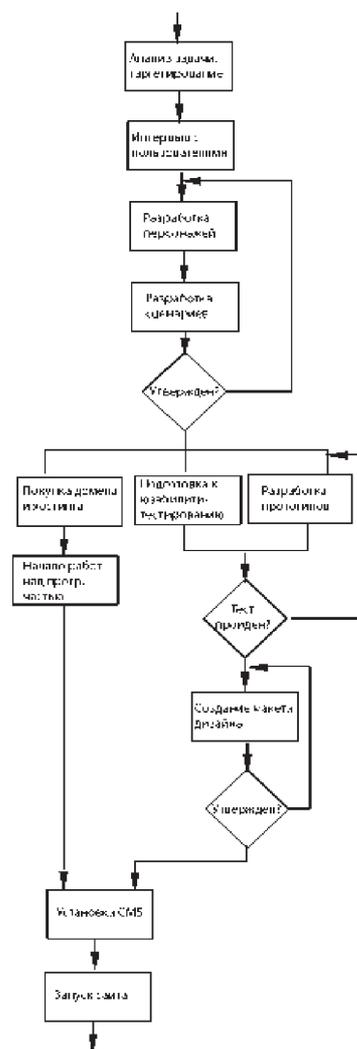


Рис. 2. Бизнес-процесс создания сайта в студии Б

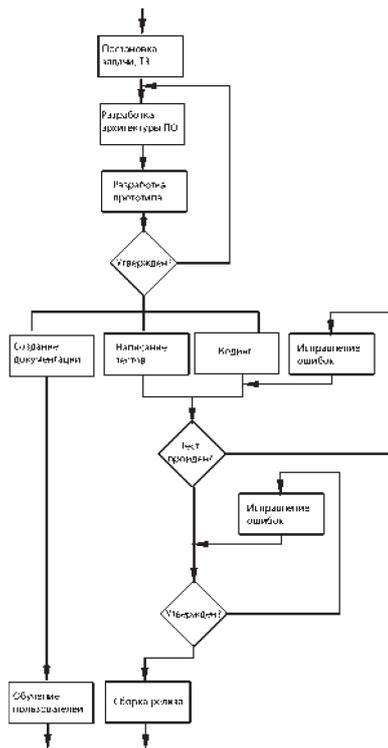


Рис. 3. Бизнес-процесс разработки программного продукта в фирме В

В ЛТ-эквивалентны, так как образуются друг из друга путем ЛТ-эквивалентных преобразований. Проверим схемы А и Б. Для этого построим их информационные графы и проверим их подобие. Поскольку они не являются подобными, следовательно, эти схемы не являются ЛТ-эквивалентными.

На основании проведенного исследования можно сделать вывод о том, что бизнес-процесс создания программного продукта В ближе к интересующему нас бизнес-процессу Б, чем процесс разработки шаблонных сайтов А. Проверим, насколько это утверждение совпадает с действительностью. Рассмотрим основные особенности бизнес-процессов А, Б и В.

Процесс А (разработка типовых сайтов) ориентирован на сокращение стоимости разработки при сохранении степени удовлетворенности клиента. Основным (и единственным) показателем качества продукта является степень удовлетворенности клиента. В этом бизнес-процессе только клиент может дать команду полностью или частично переделать задание. Поэтому число версий каждого проекта редко достигает десяти. При этом самих проектов очень много. Сотрудники переходят из проекта в проект ежедневно, иногда несколько раз в

день. Поэтому, основной проблемой для А является тайм-менеджмент для сотрудников и оптимальное распределение ресурсов между проектами с целью обеспечить их равномерное, бесперебойное выполнение.

Процесс Б в значительной мере ориентирован на выпуск дорогой, трудоемкой, качественной продукции. Поэтому здесь большое значение уделяют проверке сделанной работы. Проверка ведется несколькими службами (арт-директор, отдел usability-тестирования и т.д.), каждая из которых выдвигает свои требования. Это способствует появлению большого количества мелких требований, работа над которыми резко увеличивает число версий каждого файла. Рабочая группа для каждого проекта формируется в его начале и работает над ним довольно продолжительное время. Основной проблемой этого типа организации является управление требованиями, а также организацией тестирования продукта несколькими группами.

Процесс В – это масштабный международный проект с продолжительностью в несколько лет. Проблему контроля версий и учета ошибок здесь успешно решена (Clearcase+TRAC), а основной проблемой стала передача знаний новым сотрудникам от уходящих. Частично она была решена размещением всей документации во внутрикорпоративной сети, а также созданием внутренней wiki.

Исходя из анализа используемых программных решений, можно заключить, что проблема управления файлами проекта, хранения истории их изменения, а также точного учета всех пожеланий заказчика и арт-директора и негативных результатов тестов, столь актуальная для бизнес-процесса Б, не может быть устранена с использованием решений для бизнес-процесса А. С другой стороны, в бизнес-процессе В именно эта задача была успешно решена введением системы контроля версий (Clearcase) и системы учета ошибок (TRAC). Система контроля версий (с учетом масштаба и бюджета Clearcase был заменен на бесплатный svn) действительно решила проблему с организацией единого хранилища относящихся к проекту материалов в бизнес-процессе Б. Использование багтрекера TRAC позволило команде, проводящей тестирование, оперативно уведомлять всех участников проекта о найденных в ходе теста недостатках. Раньше эта часть работы не была автоматизирована вообще.

Сравнительные характеристики бизнес-процессов А, Б, В

	А	Б	В
Типовая задача бизнеса	«Поточное» производство недорогих сайтов	Создание сложных сайтов с индивидуальным подходом к клиенту	Один очень большой распределенный проект
Число сотрудников	12 + 1 вне штата	17 + 8 вне штата	1 группа — 25, 2 группа — 9, 3 группа — 5
Затраты времени на проект (ч/д)	8-16	50-200	Более 20000
Решенные проблемы	Централизованное хранение материалов, учет рабочего времени, быстрый обмен информацией	?	Контроль версий, устранение коллизий с одновременной правкой одного и того же файла, учет исправления ошибок, учет рабочего времени, быстрый обмен информацией
Нерешенные проблемы	Точная оценка сроков, оптимальное распределение ресурсов между проектами	Контроль версий, учет пожеланий и результатов тестов	Передача информации о проекте от уходящих сотрудников новым
Внедренные IT-решения	1С, Мегаплан, Я.online, файл-сервер с общими данными, общий почтовый ящик	?	SAP, Clearcase, Trac, Wiki, Rational Rose, MS Project, MS Outlook, Jabber
Основные подпроцессы	Оценка ТЗ, разработка макета, верстка, установка CMS	Анализ требований, исследование предметной области, проект интерфейса, тестирование, разработка макета, верстка, установка и доработка CMS	Анализ требований, изучение предметной области, разработка архитектуры ПО, кодирование, тестирование, устранение ошибок, сборка релиза, создание документации.

Заимствование коснулось не только используемого ПО. Некоторые управленческие решения из практики В тоже оказались полезны. В частности, концепция «релиза» и «предрелизной подготовки», как финальной стадии развития проекта, при которой запрещены любые правки, кроме исправления ранее обнаруженных ошибок, теперь активно практикуется и в Б.

Таким образом, использование схем-бизнес-процессов для построения моделей функционирования предприятий, а также применение аппарата логико-термальной эквивалентности для анализа сходства бизнес-процессов позволяет существенно упорядочить процесс заимс-

тования программных и управленческих решений в сходных условиях, что позволяет повысить эффективность построения автоматизированных решений.

СПИСОК ЛИТЕРАТУРЫ

1. Воронин А.А., Мишин С.П. Оптимальные иерархические структуры. – М.: ИПУ РАН, 2003. – 214 с.
2. Заложнев А.Ю. Прикладные модели и методы внутрифирменного управления. – М.: ИПУ РАН, 2003. – 167 с.
3. Котов В.Е., Сабельфельд В.К. Теория схем программ. – М.: Наука, 1991. – 248 с.
4. Иткин В.Э. Комбинированная эквивалентность схем программ. – Кишинев, Изд-во Киш. Гос. Университета, 1974. – 255 с.

Рындин Александр Алексеевич – ВГТУ, д.т.н., проф., Воронежский государственный технический университет, Тел. (4732)77-45-24, E-mail:ar@sb.vrn.ru

Ryndin A.A. – Doctor of Technical Sciences, Professor, Voronezh State Technical University. Tel. (4732)77-45-24, E-mail:ar@sb.vrn.ru.

Сапегин Сергей Владимирович – к.т.н., кафедра Программирования и информационных технологий, Воронежский государственный университет, Тел.+7-920-402-76-55, E-mail:svsapegin@mail.ru

Хабаров Сергей Игоревич – ВГТУ, аспирант, Тел.+7-951-562-0649, E-mail: habagreya@mail.ru

Sapegin S.V. – Candidate of Technical Sciences, Associate Professor, Department of Programming and Information Technologies, Voronezh State University . Tel. (4732)208-470. E-mail: svsaepgin@mail.ru

Khabarov S.I. – Post-graduate student, Voronezh State Technical University. Tel.+7-951-562-0649, E-mail: habagreya@mail.ru