

М.Д. ГОДЛЕВСКИЙ, д-р техн. наук, НТУ "ХПИ",
А.А. ПОЛЯКОВ, НТУ "ХПИ"

ПОДХОД К ПОСТРОЕНИЮ ИНТЕГРИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ВЫСОТЕХНОЛОГИЧНЫМИ ПРОЕКТАМИ

Висвітлена необхідність побудови інтегрованої системи управління проектами з метою забезпечення процесу прийняття рішення з ретроспективними даними. Запропоновано архітектуру до побудови такої системи та впроваджена реалізація системи, що підтверджують теоретичні результати.

The paper shortly describes and explains purpose of creation new integration system for project management. The main goal of system is provide retrospective data about project tasks. The theoretical result of research was respected by implementation of referenced architecture for information system.

Постановка проблемы. Успех организации-производителя программного обеспечения (ПО) зависит от ее способности адаптироваться в условиях изменчивой внешней среды, в умении распоряжаться и обрабатывать стремительно увеличивающиеся информационные потоки для нахождения конкурентного преимущества, которое позволит достичь организации временного успеха. Временность обусловлена тем, что техническое преимущество, достигнутое сегодня, завтра будет у всех производителей. Поэтому процесс поиска способов дифференциации своих продуктов среди прочих является непрерывным процессом, что приводит к необходимости постоянных изменений существующих продуктов, создания новых, а также совершенствования внутренних управленческих процессов.

Рассматриваемые проекты по разработке программного обеспечения, будем классифицировать как технические отдельные (моно) инновационные, краткосрочные (продолжительностью до 3-х лет) проекты.

Такие проекты характеризуются: четким определением главной цели проекта, но отдельные цели могут уточняться по мере достижения частных результатов; срок завершения и продолжительность проекта определяются заранее, желательно их точное соблюдение, однако они могут корректироваться в зависимости от полученных промежуточных результатов и общего продвижения проекта. Основные ограничения связаны с лимитированной возможностью использования ресурсов (оборудования и специалистов), и, как правило, имеющиеся ресурсы и определяют расходы на проект и срок его готовности.

Согласно исследованиям, проведенным исследовательско-консультативной фирмой "The Standish Group International" в области проектов в сфере информационных технологий, осуществляемых для частных и государственных компаний, около 31% проектов не доходят до завершения, 58% выполнены с перерасходом средств в 158% от их первоначальной

стоимости, и только 16% проектов выполнены в срок без превышения сметной стоимости [1]. Подобная статистика свидетельствует о том, что методы и средства управления проектами в проектах по разработке программного обеспечения используются недостаточно эффективно.

Анализ литературы. Анализируя [2, 3] причины неудач проектов можно выделить основные проблемные области в управлении программными проектами: внутренние изъяны календарного планирования, неуправляемый поток требований, нарушение спецификаций, низкая производительность и текучесть кадров. Две последние проблемы тесно связаны с вопросами стимулирования персонала, координации интересов отдельных участников проекта, и являются предметом таких дисциплин, как теория активных систем [4], управление персоналом [5], социальной психологии [6, 7], поэтому эти вопросы мы исключим из рассмотрения. Таким образом, в поле нашего рассмотрения остаются вопросы – календарного планирования, управления требованиями и контроля выполнения хода проекта.

Для описания, анализа и оптимизации календарного плана расписания проектов наиболее часто в программном обеспечении для управления проектами (ПОУП) [8] используются сетевые модели с работами в узлах, метод критического пути (Critical Path Method), временные диаграммы и т.д. Выбор этого типа сетевой модели позволяет исключить избыточность по сравнению с сетевой моделью, основанной на дугах, за счет отсутствия необходимости введения фиктивных работ для синхронизации одновременных и параллельных событий [9]. Основой для построения структуры таких сетевых моделей является иерархическая декомпозиция работ (WBS – Work Breakdown Structure) [3, 10], построенная на этапе разработки концепции проекта [11]. Для оценки временных параметров работ на практике используются экспертные оценки [7, 12], что приводит к значительной погрешности для получаемых оценок. Возможным подходом к решению этой проблемы, является использование аналитических и комбинированных моделей оценки стоимости ПО [10], где используются исторические данные для коррекции различных коэффициентов, которые определяются в рамках отдельных компаний, команд, типов проектов с использованием специальных репозиторий. В открытом доступе таких репозиторий не существует [13], что существенно ограничивает применимость таких методов.

В процессе жизни проекта все составляющие предметной области проекта могут претерпевать изменения. Управления требованиями или управления предметной областью заключаются в управлении изменениями на протяжении жизненного цикла проекта. Цели, результаты, работы и их характеристики могут изменяться или уточняться как в процессе разработки проекта, так и по мере достижения промежуточных результатов. Основными характеристиками требований является разнородность их представления – не структурированные текстовые описания, графические изображения [14] и, конечно, быстрая изменчивость.

В методологиях разработки программного обеспечения этому вопросу уделяется особое внимание, но при этом не существует единого подхода к решению этого вопроса. Двумя крайними подходами являются: строгое документирование всех требований в различного рода документах определенного формата [1], второй же подход декларирует, что вопросу управления требованиями не стоит уделять достаточного внимания, поскольку они слишком изменчивы, и достаточно иметь набор липких листов бумаги для записей [15]. Каждый из методов имеет существенные недостатки, для первого характерна с течением времени рассогласованность документов, описывающих требования, с реальными потребностями заказчика, в результате разработанная система не будет отвечать требованиям заказчика. Второй метод, лишен этого недостатка, но при увеличении количества требований, неприменимость этого метода становится очевидной. В последнее время получили развитие гибкие (agile) технологии разработки программного обеспечения, основой которых является синтез и адаптация различных подходов управления, методики разработки конкретных проектов в зависимости от размера и участников проекта [16]. В результате эта методология явилась основой для разработки программных продуктов управления требованиями, представляющими требования в виде, аналогичном табличному представлению требований модели КСРМ [17].

При рассмотрении вопроса о контроле выполнения проекта, может быть применена методика освоенного объема [18], основой которой является использование различных показателей для сравнения текущего состояния с плановым. Применительно к проектам по разработке ПО, сложность применения методики состоит в том, что сложно идентифицировать текущее состояние системы в силу изменчивости требований.

Исходя из проведенного анализа, обозначим основные проблемные области:

- отсутствие доступных репозитивов, представляющих ретроспективные данные реализованных проектов, что не позволяет использовать большинство методов оценки стоимости программного обеспечения [13];
- проблема мониторинга текущего состояния проекта, которая возникает вследствие того, что при планировании проекта рассматривается только набор требований, определенных на фазе разработки концепции;
- отсутствие статистических данных для эффективного анализа завершенных проектов.

Цель статьи. Предложить подход для построения интегрированной платформы на основе существующих программных систем для управления небольшими краткосрочными проектами по разработке программного обеспечения, позволяющей эффективно управлять требованиями, сохранять ретроспективную информацию о проекте, оценивать состояние проекта на различных стадиях его жизненного цикла, а также использовать собранную статистику о выполнении проектов при оценке новых проектов.

Построение интегрированной системы управления программными проектами. Основываясь на проведенном анализе проблемных областей в управлении программными проектами, выделим основные функциональные требования, которым должна отвечать интегрированная система управления проектами (ИСУП):

- управление проектом (создание WBS, планирование проекта на основе доступных ресурсов, оценка стоимости проекта, возможность мониторинга процесса реализации проекта);
- управление требованиями (спецификация требований в текстовом и графическом виде; возможность систематизации требований);
- сбор информации о проекте в процессе всего жизненного цикла для его разностороннего анализа.

На основе требований сформируем общую архитектуру системы ИСУП (см. рис.).

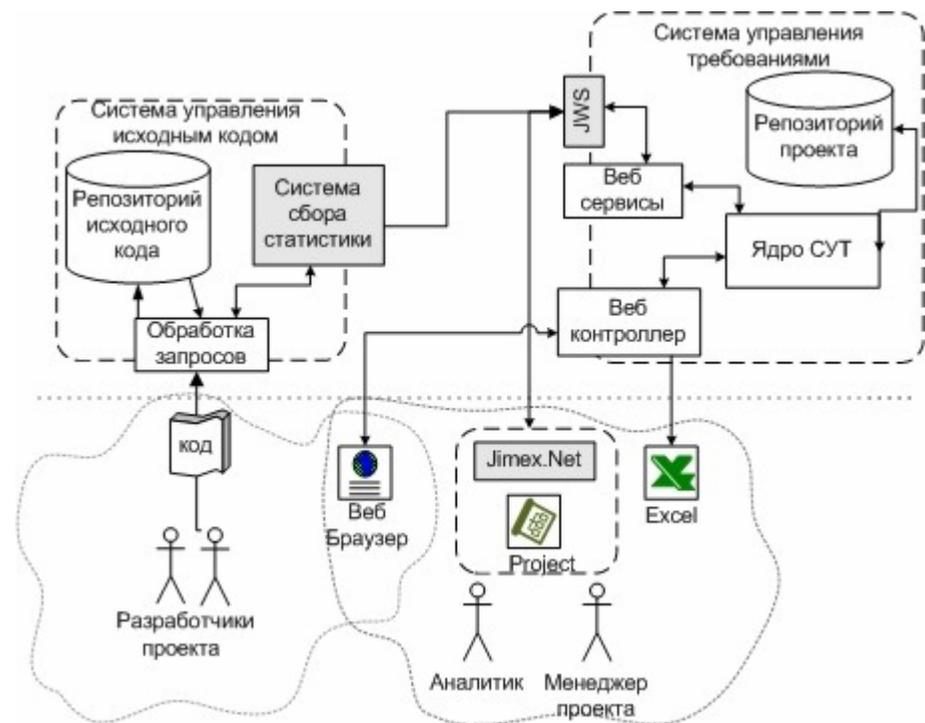


Рис. Общая архитектура ИСУП

Поскольку проекты по разработке программного обеспечения предполагают разработку исходного кода с использованием определенного

языка программирования, то в ИСУП может быть включена система управления исходным кодом (УИК), как дополнительный источник информации о ходе выполнения отдельных задач проекта. Основой системы УИК является репозиторий исходного кода, который хранит различные версии исходного кода. Используя такой репозиторий, можно получать информацию, например, какое количество строк кода потребовалось для реализации той или иной работы, что в дальнейшем может быть основой при использовании аналитических методов стоимостной оценки схожих работ в новых проектах.

При построении ИСУП будем использовать, по возможности, готовые решения, доступные на рынке программного обеспечения. На основе исследования доступных программных систем управления проектами (СУП) [13] выберем в качестве СУП продукт Microsoft Project Professional [8], он полностью удовлетворяет сформулированным требованиям к СУП.

Рассматривая системы управления требованиями, можно прийти к выводу, что их вполне можно использовать в качестве основы для построения репозитория ретроспективных данных. Поэтому основным требованием к системе управления требованиями будет возможность ее расширения – как программных компонент, так и репозитория, на котором она построена. Среди таких систем можно выделить две, отвечающие такому требованию – бесплатная система управления требованиями Mantis и коммерческая система с широкими возможностями Atlassian JIRA. Для построения системы ИСУП выберем второй продукт, т.к. он реализует функциональность не только системы управления требованиями, но и некоторые функции системы управления заданиями, позволяет назначать задачам приоритеты, ресурсы, распределять требования и задачи по проектам, версиям проектов, собирать статистику о задачах и т.д.

Каждый из выбранных компонентов реализуют один из аспектов требований ИСУП, но простое совместное их использование не позволяет достичь требований, поставленных перед системой ИСУП. Поэтому для каждого из компонентов ИСУП были разработаны программные компоненты (на рис. они выделены серым цветом), которые позволяют использовать разнородные программные продукты различных производителей как однородную тесно интегрированную систему.

Компонент "система сбора статистики" в системе УИК, предоставляет различную информацию о выполняемых работах или задачах проекта. Практически все результаты (документы, исходный код) задач проекта могут быть сохранены в системе УИК. Такой информацией могут быть данные о затраченном времени на выполнение задач, которые извлекаются из сопроводительного сообщения при сохранении результатов, и после через интерфейс другого разработанного интеграционного компонента "JWS" (Jimex Web Service) [19], сохраняются в репозитории проекта, т.о. происходит сбор статистической информации о ходе работ проекта. Эта информация и информация о новых требованиях, состоянии текущих требований может быть

получена в СУП с помощью компонента Jimex.Net, расширяющего СУП для работы с СУТ.

Выводы. В результате проведенного исследования предложен новый подход к созданию автоматизированных интегрированных систем управления проектами. Подход получил свою частичную реализацию и внедрение при выполнении краткосрочных проектов, обеспечивая менеджеров проектов дополнительной информацией в ходе принятия решений о количественных характеристиках работ и в процессе всего жизненного цикла высокотехнологичных проектов. С использованием информации, предоставляемой ИСУП о проекте, может быть построен алгоритм оперативного управления высокотехнологичными проектами, что позволит выявлять отклонения проекта от намеченного плана и корректировать его для успешного достижения целей проекта.

Список литературы: 1. *Роис У.* Управление проектами по созданию программного обеспечения – М.: Лори, 2002. – 448 с. 2. *Годлевский М.Д., Поляков А.А.* Управление рисками в высокотехнологичных проектах: состояние и подходы управления // Проблемы программирования. Институт программных систем НАН Украины. – 2006. – № 2–3. – С. 311–319. 3. *ДеМарко Том, Листер Тимоти.* Вальсируя с медведями: управление рисками в проектах по разработке программного обеспечения. – М.: р.т. Office, 2005. – 190 с. 4. *Бурков В.Н., Новиков Д.А.* Как управлять проектами. – М.: СИНТЕГ ГЕО, 1997. – 195 с. 5. *Базарова Т.Ю., Еремина Б.Л.* Управление персоналом. – М.: ЮНИТИ, 2002. – 560 с. 6. *Гуленко В.В.* Менеджмент слаженной команды. Соционика для руководителей. – М.: АСТ, 2004. – 288 с. 7. *ДеМарко Том, Листер Тимоти.* Человеческий фактор, успешные проекты и команды. – СПб.-М.: Символ, 2005. – 237 с. 8. Microsoft Project, URL: <http://www.microsoft.com/project>. 9. *Ефремов В.С.* Проектное управление: модели и методы принятия решений, URL: <http://www.cfin.ru/press/management/1998-6/11.shtml>. 10. *Баркалов С.А., Воропаев В.И. и др.* Математические основы управления проектами / Под ред. В.Н. Буркова. – М.: ВШ, 2005. – 423 с. 11. *Товб А.С., Цунес Г.Л.* Управление проектами: стандарты, методы, опыт. – М.: Олимп-Бизнес, 2003. – 239 с. 12. *Сидоров Н.А., Баценко Д.В.* Модели, методы и средства оценки стоимости программного обеспечения // Проблемы программирования. Институт программных систем НАН Украины. – 2006. – № 2–3. – С. 311–319. 13. *Михайловский Н.Э.* Сравнение методов оценки стоимости проектов по разработке информационных систем. URL: <http://www.cfin.ru/management/practice/supremum2002/15.shtml>. 14. *Карл И. Вигерс,* Разработка требований к программному обеспечению. – М.: Русская Редакция, 2004. – 576 с. 15. *Бек Кент, Фаулер Мартин.* Экстремальное программирование: планирование. – СПб.: Питер, 2003. – 144 с. 16. *Амблер Скотт.* Гибкие технологии: Экстремальное программирование и унифицированный процесс разработки. Библиотека программиста. – СПб.: Питер, 2005. – 412 с. 17. *Kop Ch., Mayr H.C.* Mapping Functional Requirements: From Natural Language to Conceptual Schemata. In: Proc. International Conference SEA 2002, Cambridge, USA, Nov. 4-6, 2002. – P. 82–87. 18. *Колосова Е.В., Новиков Д.А., Цветков А.В.* Методика освоения объема в оперативном управлении проектами. – М.: ООО "НИЦ"Апостроф", 2000 – 156 с. 19. *Polyakov A.* Confluence Jimex Project Description. URL: <http://confluence.atlassian.com/display/JIRAEXT/Jimex+Plugin>.

Поступила в редакцию 27.10.2006