

32. Ishida Y., Mizessyn F. Learning algorithms on an immune network model: application to sensor diagnosis // In: Proc. of international joint conference on neural networks, China, November 3-6, 1992. V. 1. – P. 33-38.

33. Ishiguro A, Watanabe Y., Uchikawa Y. Fault diagnosis of plant systems using immune networks // In: Proc. of the 1994 IEEE international conference on multisensor fusion and integration for intelligent systems (MFI'94), Las Vegas, October 2-5, 1994. – P. 34-42.

34. Нестерук Ф.Г., Баранюк Т.Н., Нестерук Л.Г., Марченко А.А., Нестерук Г.Ф. Комплементарное кодирование информации в нейросетевых средствах систем защиты информации // Вопросы защиты информации. – 2007. – № 4. – С. 53-57.

35. Carpenter G.A., Milenova B.L. Distributed ARTMAP // Proc. of the International Joint Conference on Neural Networks. 1999.

36. Kohonen T. The self-organizing map // Proceedings of the IEEE, 78, 1990. – P. 1464-1480.

37. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. – М.: Мир, 1992.

38. Zhou Z., Chen S., Chen Z. FANNC: A Fast Adaptive Neural Network Classifier // Knowledge and Information Systems. – 2000. – P. 115-129.

Завьялов А.А.

### **Методология создания корпоративных сервисов и приложений под управлением бизнеса**

*(СПбГУЭФ, Санкт-Петербург)*

#### **Введение**

Компаниям в современных условиях приходится мириться с темпом изменения конъюнктуры рынка, изменяя свой бизнес под изменения конкурентного рыночного спроса. Примерно 80 процентов ИТ бюджета компаний тратится на настройку и на расширение существующих приложений. Эти приложения не создаются с задумкой быть гибкими и поэтому, пока бизнес переключается с некоторой задержкой к новым или расширенным процессам, база ИТ не способна справиться с основными изменениями. Традиционные приложения и архитектуры не способны быть на одной волне с бизнес инновациями в основном потому, что процессы не адаптированы к нуждам бизнеса. Требования бизнеса часто трансформируют в ИТ проекты, которые не могут работать вместе, возможность переиспользования артефактов, созданных для разных проектов зачастую очень низка. Создание приложений, которые были бы достаточно гибки для реагирования на неопределенность, требуют более систематического подхода к разработке самого приложения. Пока бизнес не сможет создать необходимый функционал для ИТ, который бы смог адекватно реагировать на неопределенность, традиционно сложно будет определить требования к размеру бюджета, необходимого для создания гиб-

кого приложения. Традиционная негибкость архитектур приложений делает даже малые улучшения на столько дорогими, что уже на уровне прогнозирования невозможно предположить точную сумму и сроки.

Механизм, который должен быть продуман в плане, какие силы и средства ИТ будут завязаны на бизнес-стратегию и требования, в то время как исполняющий стандартизованный фреймворк должен быть легко понимаем и может исполняться повторно и, что самое главное, успешно. Компания может достичь гибкости своего бизнеса через ИТ за счет моделирования бизнес-процессов, которые фактически коллективно определяют способ его исполнения. Первым необходимым этапом является моделирование бизнес процесса через поэтапное моделирование компонент процесса. При помощи измерений бизнес процесса или ключевого прецедента через подсчеты возврата от инвестиций (ROI), ключевых показателей эффективности (KPI), или других метрик, компания может использовать эти модели бизнес процессов (BPM) как основной механизм для коммуникации между требованиями бизнеса и сферой ИТ. Бизнес и ИТ могут построить жизненно важный мост через пропасть коммуникации при помощи хорошо раздерижированных моделей бизнес-процессов, которые создают связь между требованиями бизнеса и тем что ИТ исполняет и предоставляет.

В то время как первым этапом для Разработки Управляемой Бизнесом или Business Driven Development (BDD) является создание Моделей Бизнес Процессов (BPM), структура ИТ решения также нуждается в адаптации к использованию Моделей Бизнес-Процессов как входных артефактов для проектирования и разработки фаз жизненного цикла разработки программного продукта. ИТ архитектура должна быть готова к проектированию и исполнению деятельности процесса как компонент программного продукта или сервиса.

При использовании Разработки Управляемой Бизнесом (BDD), компания моделирует и предоставляет новые бизнес-процессы (когда они написаны на концептуальном уровне) в ИТ подразделение. Анализ нового процесса может показать что уже существует необходимый ИТ сервис для необходимого требования и нужно будет только соединить уже существующий сервис для соответствия с новым бизнес-процессом или может показать, что компании необходимо разработать новые ИТ сервисы и добавить их в портфолио ИТ сервисов. Схожим образом, если изменения применяются к существующему процессу, Модель Бизнес-Процессов поправляется для отражения изменений и отправляется в ИТ подразделение для последующей технической ревизии, основанной на тех сервисах, которые необходимо расширить или модернизировать.

Подход Разработки Управляемой Бизнесом (BDD) помогает увеличить быстродействие бизнеса и позволяет расставить приоритеты и выровнять инициативы ИТ с распоряжениями бизнеса. Также это косвенно помогает упростить процесс определения затрат на ИТ бюджет внутри компании.

### Подход к разработке через создание исполняемых моделей

Как было упомянуто в предыдущей части, ИТ компании должны стремиться заполнить разрыв между бизнес требованиями и ИТ решениями и также должны быть быстры и ответственны в создании ИТ решений. Эта необходимость привела к развитию Сервис Ориентированной Архитектуры (SOA), которая предоставляет ИТ Фреймворк с наряду с набором принципов и директив для создания ИТ решений как набора переиспользуемых, компонентных и конфигурируемых сервисов, которые не зависят от приложений и от исполняемых платформ. Переход компаний к COA (SOA) подразумевает подход Разработки Управляемой Бизнесом (BDD), который использует бизнес цели и требования для управления последующим проектированием, разработкой и тестированием. Эти перспективы создания композитных бизнес приложений при помощи переиспользования существующих или вновь созданных сервисов, позволят создать адаптируемые и гибкие бизнес-решения. Так же это приносит так необходимую гибкость в ИТ компании и помогает выровнять ИТ решения с нуждами бизнеса.

Этот раздел предоставляет обзор множества этапов, включаемых в типичный проект, который соблюдает методологию Разработки Управляемой Бизнесом (BDD). В следующем разделе опишем этапы более детально. На рисунке 1 показан цикл деятельности, который определяет высокий уровень этапов по Разработке Управляемой Бизнесом (BDD).

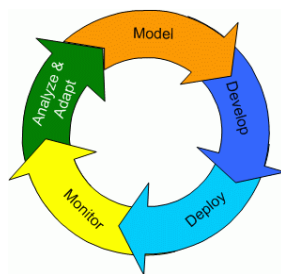


Рис. 1. Этапы разработки, управляемой бизнесом

Первый этап есть моделирование бизнес-процесса, который требует применение ИТ. Рекомендовано начинать с моделирования ключевых бизнес процессов и, используя выходы деятельности по моделированию, соединить бизнес-требования в предметную область ИТ. Обязанностью бизнес-пользователей и заинтересованных лиц является ассоциирование процессов и их значимых компонент с Возвратом от Инвестиций (ROI), ключевыми показателями эффективности (KPI) и другими уместными метриками. На более поздних этапах жизненного цикла ИТ эти меры помогут показать, что ИТ преподнесла то, что было нужно бизнесу.

Как только процессы смоделированы, выходы моделей могут быть использованы как входы для фазы сбора требований инициативы. Деятельно-

сти или этапы процесса, которые создают предоставленную модель бизнес-процессов, могут быть анализированы для формирования базы для моделирования диаграмм прецедентов (Use Case Diagrams). Создание диаграмм прецедентов является важным этапом в фазе сбора требований проекта. Основанная на диаграммах прецедентов, архитектура приложения является структурированной, и сервисы компании идентифицированы, спроектированы, разработаны и в последующем соединены вместе как сервисная композиция, которая реализует бизнес-процессы. После разработки, проект переходит в фазу внедрения, в ходе которого разработанные компоненты показываются как публикуемые, доступно расположенные, видимые сервисы. Эти программные сервисы разворачиваются в исполняющих средах, таких как Сервер Приложений (Application Server).

После процесса внедрения проект входит в фазу мониторинга и управления. Как только сервисы подняты и работают, бизнес-процессы могут мониториться на производительность и сбор данных в условиях реального времени, отчетность и анализ. Для того, чтобы это произошло, этапы в бизнес процессе (как моделировалось на первой фазе) должны иметь привязку к различным бизнес-метрикам (таким как ROI и KPI), относительно которых производительность в условиях реального времени, наличие задержек и другие факторы могут быть измерены. Измерение является неотъемлемой частью для определения, соответствует ли ИТ решение требованиям бизнеса, описанным в Соглашении об уровне сервиса или SLA (Service Level Agreement).

Данные, собранные с мониторинга в режиме реального времени, анализируются в сравнении с ожиданиями в SLA или другими мерами производительности, метрик или критериев. Собранная информация представляется архитекторам, проектантам и разработчикам, которые в свою очередь исследуют данные и находят инновационные пути оптимизации или улучшения процесса через повышение и подстройку производительности исполняющего кода. Иногда изменения также могут быть выполнены бизнес-пользователями путем изменения бизнес-правил, используя внешние интерфейсы, которые не предполагают изменений кода. Если анализ предлагает внести изменения в бизнес-процесс, соответствующие модели процесса могут быть модифицированы по тем же этапам (Разработка – Внедрение – Мониторинг), повторяющимся для улучшения реализации. Это завершает круг исполнения анализом и приемами адаптации процесса, возвращаясь к этапу моделирования. Этот механизм помогает и бизнесу и ИТ адаптироваться к изменяющимся требованиям бизнеса с малым временем оборота. Это позволяет определить увеличения затрат на существующую функциональность системы при помощи прямой ассоциации нужд бизнеса и оправдать ИТ инициативу.

### Роли, необходимые для выполнения проекта

Как было сказано ранее, BDD проект подразумевает вклад различных людей, с различным набором уникальных навыков. В этом разделе представлены различные роли, необходимые для успешного выполнения ИТ проекта, управляемого бизнес целями, видением и требованиями. Рис. 2 показывает несколько важных ролей, необходимых для реализации различных фаз инициативы по BDD.

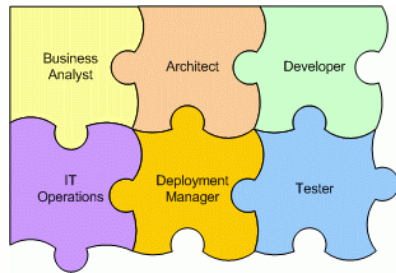


Рис. 2. Необходимые роли

Описание каждой роли:

- **Бизнес аналитик.** Роль высокого уровня, ответственная за бизнес анализ и рабочие деятельности BPM. Бизнес аналитик идентифицирует прецеденты и создает функциональные спецификации для каждого прецедента. Роль высокого уровня предполагает, что аналитик должен заниматься своей специализированной задачей, а именно анализ бизнес-процессов.

- **Архитектор.** Роль высокого уровня, ответственная за всю работу по созданию архитектуры и проекта системы. Более специализированные роли, такие как архитектор компании, архитектор приложения, архитектор SOA, ведущий архитектор (на уровнях микропроектирования, например моделирование классов, диаграммы последовательностей и прочее) и архитектор инфраструктуры отвечают за множество работ по проектированию архитектуры на разных уровнях в этой фазе проекта.

- **Разработчик.** Роль высокого уровня, ответственная за реализацию решения. Более специализированные актеры в роли могут выполнять работу, такую как программист баз данных, Java разработчик, Web разработчик, и хореограф бизнес-процессов. Каждый разработчик работает на специфических слоях множества приложений, и каждая работа предполагает наличие необходимых навыков у актеров.

- **Тестер.** Это роль, ответственная за исполнение необходимой деятельности по тестированию приложения, до того как оно попадет в среду предприятия. Тестер создает скрипты тестирования прямо из функциональных требований, которые проиллюстрированы в прецеденте. Эти тестовые скрипты исполняются с различными входными условиями, для того чтобы получить плановые выходы. От того, насколько полны тестовые прецеденты и их исполнения, тем сильнее приложение. Также минимизируются шансы появления проблем во время работы приложения в среде предприятия.

- **Менеджер внедрения.** Ответственен за внедрение кода приложения в инфраструктуру в различные среды (например, тестирования, производственную и т.д.) и все работы, необходимые для плавного внедрения, пригодного для всех сред (например, разработка инсталляционных скриптов, управление конфигурациями и т.д.).

- **Менеджер ИТ операций.** Ответственен за поддержку деятельности, неотъемлемых для непрерывной поддержки приложения, когда оно уже поднато и работает, особенно в среде предприятия. Эта роль также может быть ответственна за сбор данных режима реальной работы приложения и анализе результатов относительно требований в SLA.

Не обязательно, что только вышеперечисленные роли должны присутствовать в проекте. Здесь описаны основные роли, которые привлекаются в типичный проект, и они естественно могут быть подобраны под организационную структуру организации, требования проекта и его сложность.

### Практическая реализация

Примером применения данной методологии может служить создание сервисов, доступных для граждан страны через портал государственных услуг. Заказчиком или бизнес-стороной является само государство, и ввиду того, что оно осуществляет процесс обслуживания граждан, который имеет только вербальное описание и задокументирован на уровне различных инструкций и требований к нему, возникает идея обработать этот процесс с различными целями. Во-первых, формирование не только визуального представления последовательности действий в процессе с необходимыми ролями, но по возможности провести его анализ при помощи имитационного моделирования, с целью выявления узких мест. Во-вторых, определить ключевые показатели производительности подпроцессов, действий, самого процесса. И, в-третьих, сделать из этого процесса, нарисованного в графической нотации управления бизнес-процессами (BPMN), понятный Процесс серверу алгоритм выполнения данного процесса выраженный в коде на языках исполнения бизнес-процессов (BPEL) и описания веб-сервисов (WSDL) с целью дальнейшего контроля при помощи расставленных показателей. Так же этот подход будет удобен для дальнейших изменений процесса, которые в динамике будут отражены на порталном решении.

Итак, приведу пример такого решения на продуктах компании IBM и их линейки WebSphere. У IBM есть целый спектр инструментов, который представляет свое видение интегрированных инструментов, которые поддерживают все аспекты BDD. Далее описаны программные продукты IBM и каким образом они участвуют в различных фазах BDD:

- **WebSphere Business Modeler Advanced (WBMA)** – отображение текущего процесса (AS IS) с последующим созданием имитации, с использованием KPI, и создание нового, оптимизированного бизнес процесса (to be), создание модели на языке BPEL для WebSphere Integration Developer.
- **WebSphere Integration Developer** – сборка процесса на языке BPEL и внедрение JAVA кода (Enterprise Java Bins).
- **WebSphere Process Server** – Запуск процесса для отображения его в Business Monitor.
- **Business Monitor** – отображение бизнес измерений (требований) и готового продукта. Просмотр соответствия Готового продукта заданным требованиям.

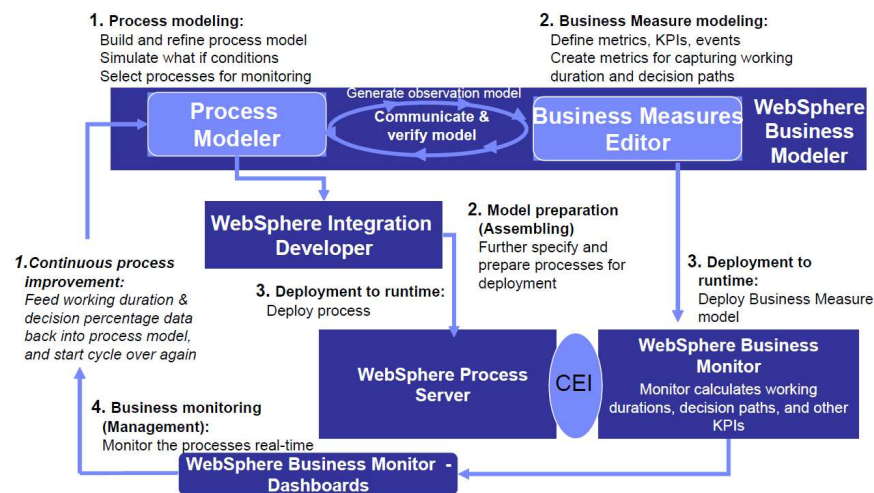


Рис. 3

На схеме (рис. 3) видно, какова последовательность действий по реализации нашего веб-сервиса. Первый шаг – происходит моделирование процессов, проводятся симуляции (имитационное моделирование), выбираются процессы, которые необходимо будет отправить на дальнейшее отслеживание или мониторинг (подготовить для экспорта Business Monitor). Второй шаг – необходимо провести либо подготовку модели в WebSphere Integration Developer, то есть подготовить процессы к разворачиванию их на процесс-сервере, либо определить измерения модели, то есть определить метрики, которые бы охватывали всю длительность процесса и путей принятия решения. Третий шаг заключается в разворачивании нашего процесса на процесс-сервере прямо из Integration Developer и разворачивания моделей измерений в Business Monitor. Business Monitor подсчитывает рабочую длительность, пути принятия решения и другие KPI. Основная Инфраструктура Событий (CEI) разрешает сервисным компонентам выпускать события, которые может использовать Business Monitor для мониторинга бизнес процессов в режиме реально времени.

Как раз четвертым шагом является мониторинг бизнес-процесса в режиме реального времени через дэшборды Business Monitor, это фактически и является нашим сервисом. Дальше все возвращается к моделированию, так как в модель вводятся новые длительности работ, проценты принятия соответствующих решений участниками процесса и т.д. Таким образом, наша последовательность действий становится циклом, так как каждая итерация завершается получением данных, которые помогают улучшить процесс, например, путем отказа от каких-либо вилок принятия решения, если пользователи, например, просто перестали пользоваться каким-либо путем решения или, наоборот, изменить процесс под новые решения проблемы.