

УДК 681.3.06

ПОСТРОЕНИЕ ИМИТАЦИОННЫХ МОДЕЛЕЙ БИЗНЕС-ПРОЦЕССОВ ДЛЯ СБОРА ТРЕБОВАНИЙ КАЧЕСТВА

А.Г. Долгарев, В.А. Шеховцов

Национальный технический университет “Харьковский политехнический институт”,
61002, Харьков, ул.Фрунзе, 21.

Тел.: (057) 707 6474, a.dolgarev@gmail.com, shekvl@yahoo.com

Описаны общие принципы подхода, позволяющего определять требования качества к программной системе на основе оценок, данных заинтересованными лицами в ходе их интерактивного взаимодействия с имитационной моделью или прототипом такой системы. Данное взаимодействие осуществляется в рамках сценариев, определенных имитационными моделями бизнес-процессов, в которых предполагается использовать эту систему. В данной работе предлагается один из возможных вариантов реализации имитационных моделей, необходимых для решения этой задачи, на основе существующих промышленных стандартов и платформ.

We present general principles of an approach allowing capturing software system quality requirements through assessments made by stakeholders during their interactions with a simulated model of this system or its prototype. These interactions are guided by scenarios defined by simulations of the business processes describing the usage of the system. We propose one of the possible implementations of the necessary simulation models based on existing industry standards and platforms.

Введение

Сбор требований для новых компьютерных приложений обычно основывается на опросе заинтересованных лиц, но этот процесс, как правило, затрудняется тем, что предполагаемые пользователи далеко не всегда ориентируются в вычислительных средствах и в приемлемых значениях параметров качества. Корректный сбор таких требований очень важен, так как получение неверных требований при разработке программного продукта повышает вероятность провала проекта. Ранее в [1, 2] было предложено преодоление этой проблемы посредством создания и использования интерактивной имитационной среды (ИИС), в которой пользователи могут взаимодействовать с имитационной моделью системы в рамках бизнес-процессов ее использования, тоже заданных имитационными моделями. В ходе этой работы они могут оценивать качественные характеристики, заданные моделью, определяя тем самым (или уточняя для себя) качественные требования к исходной системе. В работах [1, 2] определяются требования к такой среде и делаются наброски ее архитектуры.

В рамках реализации ИИС возникает задача имитационного моделирования бизнес-процессов. В данной работе рассматриваются возможные варианты решения такой задачи на основе существующих промышленных стандартов и платформ. Мы рассмотрим какие программные средства управления бизнес процессами (business process management, BPM [3]) могут использоваться для реализации такого имитационного моделирования, и как система ИИС может быть интегрирована с доступными на данный момент стандартами.

Дальнейшее изложение организовано следующим образом. Раздел 1 посвящен описанию основных принципов, положенных в основу интерактивной имитационной среды. В разделе 2 рассмотрим стандарты описания бизнес-процессов и приведем варианты реализации средства управления бизнес-процессами (business process engine), а также простейший пример моделирования бизнес-процесса. В разделе 3 будут рассмотрены возможные подходы к реализации языка описания имитационной модели бизнес-процесса.

1. Принципы организации интерактивной имитационной среды

В соответствии с [1, 2] рассматриваем организацию как систему, в которой выполняется определенный набор бизнес-процессов. Как следствие, для создания имитационной модели организации нам необходимо смоделировать ее структуру, ресурсы (в том числе людские), которые она использует, ее бизнес-процессы, процессы поддержки и процессы управления. Если такие модели созданы, доступна информация по нагрузке, т.е. информация по распределению вероятностей активизации процессов, требуемое время обработки для этих процессов, информация по утилизации ресурсов и т.д., в принципе можно построить имитационную модель всей организации. Основная цель разработки ИИС – дать возможность потенциальным пользователям разрабатываемой системы (РС) получить опыт взаимодействия с имитационной моделью, которая опирается на ключевые параметры использования РС. Мы намереваемся обеспечить взаимодействие серверных компонентов имитационной модели и моделей программных компонентов, через которые пользователи задают свои запросы. Таким образом, мы можем предоставить пользователям возможность получить впечатления от испытания функциональности РС, приближенные к реальному опыту, с опорой на статистический анализ, заложенный в имитационной модели.

Мы предполагаем, что ИИС может использоваться в трех различных режимах: 1) в режиме адаптации, в котором происходит адаптация среды имитационного моделирования для конкретной задачи; 2) в режиме эксперимента, предназначенном для интерактивного сбора требований в ходе взаимодействия заинтересованных

лиц с имитационными моделями, для которых определены различные значения ключевых параметров; 3) в режиме анализа, который используется для анализа результатов экспериментов и формулирования требований, собранных в ходе этих экспериментов.

Режим адаптации. Одной из наиболее важных задач, которую должны решить пользователи ИИС, является начальное определение области предполагаемого моделирования, т.е. определение того, что считать моделируемой организацией. Эта задача требует понимания бизнес-процессов, для участия в которых предполагается использовать РС. В режиме адаптации, следовательно, необходимо определить и формализовать бизнес-процессы данной организации.

Пользователи ИИС предоставляют все начальные данные, которые необходимы для проведения имитационных экспериментов. В частности, в этом режиме необходимо задать структуру организации, включая роли персонала, а также информацию о нагрузке. Данный процесс называется параметризацией ИИС, он охватывает типичные данные о нагрузке, такие как частотные характеристики, распределения появления событий, уровень использования инструментов, доступность персонала и других ресурсов, вероятность возникновения различных катастроф, степень серьезности этих катастроф и т.д.

Режим эксперимента и режим анализа. В режиме эксперимента производится имитационное моделирование процессов, определенных в режиме адаптации. По выбору пользователей, модели компонентов РС, зарегистрированных в ИИС, могут сопровождаться графической анимацией или выполняться как реальные компоненты. Возможно переопределение значений параметров, заданных при параметризации ИИС. Кроме того, пользователи ИИС могут оставлять комментарии по поводу их впечатлений от характеристик качества системы. Для этого предлагается дать им возможность формально оценивать зарегистрированные компоненты РС (дав оценку версии компонента РС по десятибалльной шкале или сопоставив с каждой парой версий РС предикаты сравнения “намного лучше”, “лучше”, “немного лучше”, и “эквивалентны”).

В режиме анализа пользователи анализируют данные, полученные в режиме эксперимента. Для таких пользователей (как правило, это – инженеры по сбору требований), ИИС предоставляет ряд отчетов с заранее проанализированными экспериментальными данными. Каждый такой отчет включает номер версии модели, информацию о процессах, характеристиках нагрузки, компонентах РС, структуре организации и т.д. Возможная семантическая модель для описания требований качества описана в [4, 5].

2. Средства описания бизнес-процессов

Дальнейшее изложение будет посвящено вопросам реализации имитационного моделирования бизнес-процессов. Данная реализация необходима ИИС для поддержки моделей разрабатываемой системы и бизнес-процессов ее использования.

В настоящее время стандартной графической нотацией описания бизнес-процессов является Business Process Modeling Notation (BPMN) [3]. К сожалению, BPMN не содержит информации, необходимой для выполнения бизнес-процессов, кроме того, моделирование других аспектов организации (организационной структуры, модели данных) лежит вне области ее применимости. Поэтому для реализации сбора требований на основе имитационного моделирования бизнес-процессов данная нотация подходит недостаточно хорошо. В качестве альтернативы желательно рассматривать нотацию, позволяющую представлять информацию о том, как должны быть реализованы различные этапы данного процесса (деятельности, activities), и как моделируемый бизнес-процесс взаимодействует с внешними системами.

С целью решить данную задачу был предложен стандарт языка моделирования Business Process Modeling Language (BPML). Впоследствии (в 2004 году) данный стандарт был замещен более простым в реализации (хотя и несколько ограниченным по функциональности) языком Business Process Execution Language (BPEL), который и является де-факто стандартом на сегодняшний день (спецификации BPEL4WS и WS-BPEL). Так как BPEL не имеет некоторых необходимых семантических конструкций, он, в отличие от BPML, не является полным языком, в том смысле, что на языке BPEL нельзя смоделировать и выполнить любой бизнес-процесс. Вследствие этого BPEL часто используется вместе с одним из языков программирования, например, Java (в частности, для этого используется спецификация BPELJ).

Язык BPEL представляет собой модель и грамматику описания поведения бизнес-процессов, основанного на взаимодействии процесса (координатора) и его партнерами. Взаимодействие с каждым партнером осуществляется посредством веб-сервисов. BPEL состоит из следующих компонентов (в скобках указан используемый стандарт):

- отношения между партнерами (WSDL, раздел «типы портов»);
- представление данных (XML Schema);
- обработка данных (XPath);
- обмен сообщениями (WSDL, раздел «сообщения»).

В настоящее время реализацию BPEL поддерживают следующие продукты:

- WebLogic Integration 8.1 компании BEA (реализация BPELJ). Кроме этого, компания BEA представила на рассмотрение комитета по стандартизации Java спецификацию JSR 207, описывающую разработку бизнес-процессов в Java и их развертывание на J2EE-контейнере;
- WebSphere Business Integration Server компании IBM;
- BizTalk 2004 компании Microsoft (включает возможность импорта и экспорта BPEL);

- Jboss (предоставляет реализацию WS-BPEL на своей платформе jBPM - jbpmp-bpel);
- ActiveBPEL компании Active Endpoints (предоставляет реализацию BPEL4WS и WS-BPEL с открытым исходным кодом).

В данной работе мы не будем рассматривать преимущества и недостатки различных реализаций, отметим только то, что все они позволяют разворачивать и выполнять бизнес-процесс, представленный в соответствии со стандартом BPEL4WS 1.1 и/или WS-BPEL 2.0, при этом для «партнеров» бизнес-процесс будет доступен как некоторый веб-сервис.

3. Пример моделирования бизнес-процесса

В качестве примера рассмотрим моделирование бизнес-процесса на платформе jBPM компании Jboss. Клиент реализуем в виде приложения на Java. Для этого нам понадобится Jboss Application Server версии 4.2 и пакет jbpmp-bpel, который кроме библиотек самой платформы jBPM, содержит библиотеки, необходимые для работы с BPEL. Для написания клиентского приложения и создания описания бизнес-процесса был использован Eclipse 3.3 с плагинами BPEL, Eclipse Modeling Framework 2.3, библиотеки WTP и GEF.

Для иллюстрации покажем моделирование простейшего процесса, который получает некоторое имя и возвращает ответ с приветствием (рис. 1).

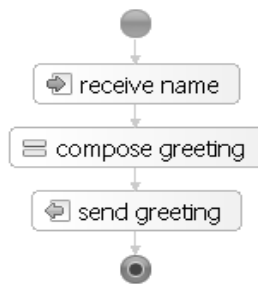


Рис. 1. Представление простейшего бизнес-процесса

Первым шагом является создание документа, описывающего процесс (hello.bpel). При этом мы создали партнерскую связь (partner link) для связи с клиентским приложением.

```

<process name="HelloWorld" targetNamespace="http://jbpm.org/examples/hello"
  xmlns:tns="http://jbpm.org/examples/hello"
  xmlns:bpel="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
  xmlns="http://schemas.xmlsoap.org/ws/2003/03/business-process/">
  <partnerLinks>
    <!-- устанавливает связь с клиентским приложением -->
    <partnerLink name="caller" partnerLinkType="tns:Greeter-Caller"
      myRole="Greeter" />
  </partnerLinks>
  <variables>
    <!-- содержит входящее сообщение -->
    <variable name="request" messageType="tns:nameMessage" />
    <!-- содержит исходящее сообщение -->
    <variable name="response" messageType="tns:greetingMessage" />
  </variables>
  <sequence name="MainSeq">
    <!-- содержит имя -->
    <receive name="ReceiveName" operation="sayHello" partnerLink="caller"
      portType="tns:Greeter" variable="request" createInstance="yes" />
    <!-- составляет приветственную фразу -->
    <assign name="ComposeGreeting">
      <copy>
        <from expression=
          "concat('Hello, ', bpel:getVariableData('request', 'name'), '!')" />
        <to variable="response" part="greeting" />
      </copy>
    </assign>
    <!-- управляет приветствием вызывающему агенту -->
    <reply name="SendGreeting" operation="sayHello" partnerLink="caller"
      portType="tns:Greeter" variable="response" />
  </sequence>
</process>
  
```

Вторым шагом является создание WSDL-описания интерфейса бизнес-процесса (hello.wsdl), который будет доступен внешним системам («партнерам»). Для указания, какие функции бизнес-процесса доступны каким клиентам, используем механизм ролей (важно учитывать необходимость использования пространства имен <http://schemas.xmlsoap.org/ws/2003/05/partner-link/>).

```
<definitions targetNamespace="http://jbpm.org/examples/hello"
  xmlns:tns="http://jbpm.org/examples/hello"
  xmlns:plt="http://schemas.xmlsoap.org/ws/2003/05/partner-link/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
  <!-- содержит имя -->
  <message name="nameMessage">
    <part name="name" type="xsd:string" />
  </message>
  <!-- содержит приветственную фразу -->
  <message name="greetingMessage">
    <part name="greeting" type="xsd:string" />
  </message>
  <!-- описывает интерфейс, предоставляемый вызывающим агентам -->
  <portType name="Greeter">
    <operation name="sayHello">
      <input message="tns:nameMessage" />
      <output message="tns:greetingMessage" />
    </operation>
  </portType>
  <!-- характеризует связь между сторонами -->
  <plt:partnerLinkType name="Greeter-Caller">
    <plt:role name="Greeter">
      <plt:portType name="tns:Greeter" />
    </plt:role>
  </plt:partnerLinkType>
</definitions>
```

На следующем шаге необходимо загрузить описание бизнес-процесса на контейнер Jboss (а именно, в базе данных jBPM). С этого момента бизнес-процесс становится доступен как веб-сервис для внешних клиентов.

После этого создадим клиента с использованием JbossWS (здесь мы не будем подробно описывать создание клиентского приложения с помощью утилит пакета JbossWS, скажем только, что он позволяет по полному WSDL-описанию сгенерировать заглушки (stubs) для веб-сервиса).

```
public void testSayHello_proxy() throws Exception {
  // получить динамический прокси для порта веб-сервиса
  Greeter proxy = service.getGreeterPort();
  // использовать прокси как локальный java-объект
  String greeting = proxy.sayHello("World");
  // проверить, корректно ли выводится строка
  assertEquals("Hello, World!", greeting);
}
```

Выполнение данного кода приводит к посылке SOAP-запроса на сервер Jboss AS, запуску нового экземпляра бизнес-процесса, выполнению деятельности «Compose greeting» и отправке SOAP-ответа.

4. Средства имитационного моделирования бизнес-процессов

В рамках POSE необходимо осуществить имитационное моделирование заданного бизнес-процесса, для чего надо расширить имеющееся описание этого процесса в стандарте BPEL недостающей информацией. Рассмотрим характер этой информации и сделаем краткий обзор решений, предлагаемых на данный момент.

Исследования в области моделирования бизнес-процессов проводились немецкой компанией IDS-Scheer, основной специализацией которой являлась разработка утилит для моделирования процессов. Основной проблемой являлся сбор статистических данных, необходимых для создания модели процесса, такой сбор требовал больших трудозатрат. Современный подход к решению этой задачи сводится к интеграции среды имитационного моделирования бизнес-процессов со средством управления бизнес-процессами (business-process engine), при этом можно повторно использовать описание процесса (например, на языке BPEL) и собранные статистические данные. В качестве примера программного обеспечения, реализующего данный подход, можно привести ARIS компании IDS-Scheer. К сожалению, высокая стоимость данного продукта делает неприемлемым его использование в исследовательских целях.

Хорошую возможность построить имитационное окружение предоставляет компания Jboss и ее продукт с открытым исходным кодом jBPM. В частности, [6] предлагает открытый язык имитационного моделирования

бизнес-процессов, расширяющий язык jPDL, принятый в jBPM, средствами задания информации, относящейся к имитационному моделированию. В частности, определены следующие элементы языка:

- distribution — распределение вероятности (нормальное, экспоненциальное и др.); задается тип распределения и его числовые параметры;
- start-distribution — распределение, которое используется для создания экземпляра процесса, если это распределение не задано, то экземпляры процесса не создаются имитационной средой;
- time-distribution — используется для определения времени выполнения задачи или времени нахождения в определенном состоянии;
- resource-needed — указывает на то, что задача или состояние требует ресурсов в определенном количестве;
- task и swimlane — задача автоматически требует ресурса из пула, представленного swimlane («плавательной дорожкой» в терминах диаграмм деятельности UML), в рамках которой она выполняется;
- resource-pool — пул доступных ресурсов (swimlane может выступать в качестве пула ресурсов, если задан атрибут pool-size).

Приведем пример описания модели в данной системе. На рис. 2 показан простой бизнес-процесс.

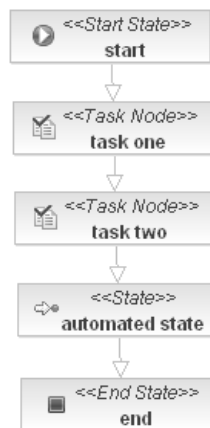


Рис.2. Бизнес-процесс с двумя узлами и промежуточным состоянием

Представим данный процесс на языке моделирования, описанном в [6]:

```

<process-definition name='FirstSteps'
  start-distribution='start new process instances of test'>
  <distribution name='start new process instances of test' sample-type='real'
    type='constant' value='20' />
  <distribution name='time required for task one' sample-type='real'
    type='normal' mean='25' standardDeviation='10' />
  <distribution name='time required for task two' sample-type='real'
    type='normal' mean='6' standardDeviation='1' />
  <distribution name='time required for automated state' sample-type='real'
    type='normal' mean='6' standardDeviation='1' />
  <resource-pool name='big machine' pool-size='3' />
  <swimlane name='tester' pool-size='2' />
  <start-state name='start'>
    <transition to='task one' />
  </start-state>
  <task-node name='task one'>
    <task swimlane='tester' time-distribution='time required for task one' />
    <transition to='task two' />
  </task-node>
  <task-node name='task two'>
    <task swimlane='tester' time-distribution='time required for task two' />
    <transition to='automated state' />
  </task-node>
  <state name='automated state' time-distribution='time required for automated
state'>
    <resource-needed pool='big machine' amount='2' />
    <transition to='end' />
  </state>
  <end-state name='end' />
</process-definition>
  
```

Рассмотрим организацию стохастических переходов между состояниями. Для этого в систему была добавлена возможность задавать вероятность того или иного перехода из текущего состояния, после выполнения некоторой задачи или после осуществления некоторого перехода (атрибут `probability`):

```
<task-node name="task">
  <task name='task' time-distribution='no work' />
  <transition to="invalid end-state" name="invalid" probability='0.0' />
  <transition to="state" name="to state" probability='1.0' />
</task-node>
<state name="state" time-distribution='no work'>
  <transition to="invalid end-state" name="invalid" />
  <transition to="unaffected decision" name="possibility two" probability='1.0' />
  <transition to="unaffected decision" name="possibility one" probability='1.0' />
</state>
<decision name="unaffected decision" expression='valid'>
  <transition to="invalid end-state" name="invalid" />
  <transition to="simulated decision" name="valid" />
</decision>
<decision name="simulated decision" expression='invalid'>
  <transition to="invalid end-state" name="invalid" />
  <transition to="end-state1" name="possibility one" probability='0.2' />
  <transition to="end-state2" name="possibility two" probability='0.7' />
</decision>
```

Предложенный язык в общих чертах показывает, какая информация необходима для осуществления моделирования и должна быть добавлена к описанию бизнес-процесса в BPEL. Он реализует базовую функциональность, необходимую для организации имитационного моделирования бизнес-процессов.

Данная реализация оставляет некоторые открытые вопросы:

- 1) не определено, как задавать эксперименты для нескольких процессов;
- 2) неясно, как задавать сценарии (т.е. поведение внешних систем);
- 3) остается не определенным задание политики управления очередями (в том числе избежание взаимных блокировок);
- 4) необходимо реализовать задание задач, выполняемых в определенное время (по таймеру);

Областью наших дальнейших исследований будет разработка расширения BPEL, свободного от перечисленных выше ограничений.

Некоторые возможности, требуемые для реализации ИИС, не могут быть реализованы в рамках описанных средств. Например, вопросы для оценок показателей качества должны задаваться в произвольные моменты функционирования системы, при этом средства задания этих вопросов должны быть отделены от основной функциональности системы. Для реализации данной функциональности целесообразно использовать аспектно-ориентированные расширения BPEL, например, AO4BPEL [7]. Некоторые задачи будут требовать использования вставок кода на Java в рамках интерфейсов JBPM.

Помимо языка, задающего имитационную модель, необходимо окружение, которое выполнит эту модель. Рынок таких окружений довольно широк, для языка Java распространен DESMOJ (<http://www.desmoj.de>). Планируется использовать данное средство для практической реализации ИИС.

Выводы

В ходе исследования выяснилось, что реализация имитационного моделирования бизнес-процессов в рамках интерактивной имитационной среды может быть осуществлена на основе программного обеспечения с открытым исходным кодом в рамках платформы Jboss jBPM-BPEL и продукта DESMOJ. При этом необходима разработка языка, связывающего данные программные средства в единую систему.

1. Shekhovtsov V., Kaschek R., Zlatkin S. Constructing POSE: a Tool for Eliciting Quality Requirements // Proc. ISTA'2007, LNI P-107, GI-Edition, 2007. – P. 187–199.
2. Шеховцов В.А., Кашек Р., Златкин С.А., Долгарев А.Г. О сборе требований качества с использованием имитационного моделирования. // Вестник НТУ "ХПИ", 2007. – № 7. – С. 17–31.
3. Havey M. Essential Business Process Modeling. O'Reilly, 2005.
4. Shekhovtsov V.; Kostanyan A. Aspectual Predesign. In: Proceedings of ISTA'2005. LNI P-63. – 2005. – P. 216–226.
5. Shekhovtsov V.; Kostanyan A., Grytskov E., Lytvynenko Yu. Tool Supported Aspectual Predesign. In: Proceedings of ISTA'2006. LNI P-84. – 2006. – P.153–164.
6. URL: http://www.camunda.com/jbpm_simulation/introduction.html
7. Charfi A. Mezini M. AO4BPEL: An Aspect-oriented Extension to BPEL. – 2007. – World Wide Web 10, 3. – P. 309 – 344.