

*И. Г. Фёдоров, канд. техн. наук, доцент,
Российский экономический университет им. Плеханова, г. Москва, Igor.Fiodorov@mail.ru*

Принципы декомпозиции модели процесса

Проблема декомпозиции модели бизнес-процесса является ключевой при разработке ИТ архитектуры предприятия. Пока она не будет решена, работа аналитика останется ремеслом, тогда как требуется инженерный подход. Цель статьи — анализ и теоретическое обоснование в контексте онтологической модели Бунге–Ванда–Вебера свойств такой декомпозиции, которую можно считать «правильной», разработке принципов, позволяющих получить правильную декомпозицию.

Ключевые слова: ИТ архитектура предприятия, декомпозиция модели процесса, онтология Бунге–Ванда–Вебера, сцепление, связность.

Введение

Одна из актуальных проблем прикладной информатики — правильная декомпозиция модели бизнес-процесса. Практически любой прикладной проект в области ИТ включает моделирование бизнес-процессов, так что его успех существенно зависит от качества моделей, используемых для анализа, реинжиниринга и автоматизации [1]. Модель, содержащая много мелких деталей, сложна для понимания [2], однако, если детали опущены, она окажется непригодной для автоматизации [3]. Чтобы обеспечить одновременно полноту и точность модели, сохранив при этом ее понимание, используют декомпозицию [4; 5], выполняемую с целью замены рассмотрения сложной проблемы решением нескольких задач меньшей сложности [6; 7]. Преимущества декомпозиции неоспоримы, однако принципы ее реализации до сих пор рождают споры. Обзор существующих подходов

к проблеме декомпозиции [8] показывает, что общепринятых методик структуризации модели процесса не существует, результат зависит от личного мастерства аналитика [11], так что модели, созданные разными аналитиками, могут различаться и с высокой вероятностью содержать ошибки. Проблема декомпозиции модели бизнес-процесса — ключевая для разработки архитектуры процессов предприятия, без ее решения невозможно снизить число неудач проектов по внедрению новых ИТ.

Так как успех разработки и использования информационных технологий зависит от качества декомпозиции модели процесса на образующие ее подпроцессы, поставим цель — теоретически обосновать свойства «правильной» декомпозиции, разработать принципы, позволяющие исключить субъективизм аналитика при структуризации модели процесса, поскольку в основе многих работ в области теории моделирования бизнес-процессов лежит концептуализация, предложенная Я. Вандом и Р. Вебером [9] на базе онтологии, разработанной М. Бунге [12; 13].

¹ Работа выполнена при поддержке Минобрнауки России, в рамках базовой части государственного задания проект 2966 (шифр 2014/162).

Особенность декомпозиции системы

Будем различать состав системы, определяющий номенклатуру ее компонентов, и структуру, трактуемую как совокупность устойчивых связей между этими элементами [12]. Неверно понимать декомпозицию как выявление состава системы, следует обязательно учитывать связи. Если они имеют сложный характер, их анализ может свести на нет преимущество декомпозиции — сильно связанные компоненты нельзя рассматривать по отдельности, но только совместно. Таким образом, система не определяется ее частями, не может быть познана и объяснена на основе одного лишь знания о ее составе [14].

Чтобы понять, что мы имеем дело с системой, нужно найти эмерджентное, системообразующее свойство [13]. Если в результате декомпозиции свойство исчезнет, это будет свидетельствовать о том, что разрушенная связь между элементами системы сильна. В качестве системообразующего свойства выберем детерминированность поведения. Поскольку нас изначально интересуют только детерминированные процессы, нарушение детерминации в результате декомпозиции будем считать недопустимым.

М. Бунге настоятельно рекомендует различать агрегат, образованный совокупностью независимых элементов, — его состояние в любой момент времени равно сумме состояний образующих его компонентов, — и систему, состояние которой не аддитивно [12]. Можно сделать вывод, что декомпозиция агрегата тривиальна, поскольку его части не взаимодействуют, а декомпозиция системы не неординарна, поскольку взаимодействующие подсистемы нельзя рассматривать изолированно друг от друга, но только во взаимосвязи.

Содержимое понятия «процесс»

Исследуем содержимое понятия «процесс», воспользуемся моделью Бунге–Ванда–

Вебера [9]. М. Бунге называет процессом историю состояний некоторого объекта, который изменяется под действием трансформаций, инициируемых событиями [13]. Объект может быть простым — неделимым, или комплексным — иметь внутреннюю структуру, в последнем случае он может быть разделен на образующие его элементы. Трансформация понимается как работа, изменяющая объект, она может быть простой или сложной, состоящей из нескольких простых.

Событие отражает факт и момент времени, когда происходит изменение объекта [15]. Событие предполагается дискретным, его длительностью принято пренебрегать. Различают внутренние события, связанные с наблюдаемым объектом, и внешние — связанные с объектом окружения. Внутреннее событие отмечает момент времени, когда закончилась предыдущая трансформация, — объект готов к исполнению очередной операции. Однако наступления внутреннего события недостаточно для начала исполнения очередной трансформации. Внешнее событие отражает факт изменения состояния объекта окружения системы, оно инициирует исполнение операции, фиксирует момент времени, когда началась трансформация. Оно не только инициирует выполнение операции процесса, но может прекратить ее, например, если возникла нестандартная ситуация, требующая специальной обработки. Можно говорить, что внешнее событие «управляет» исполнением операции процесса. Внешнее событие можно назвать генерирующим, оно связано с некоторой операцией процесса, которую будем называть обработчиком.

М. Бунге называет два объекта связанными, если изменение одного приводит к изменению другого. Он отмечает, что природа связи у объектов разной физической природы различна, так что не существует универсального подхода для исследования этой связи [13]. Это побуждает более внимательно исследовать возможное взаимодействие между элементами бизнес-процесса. Мы исключим самопроизволь-

ное изменение информационного объекта, будем считать, что он не подвержен «старению», один информационный объект не может влиять на другой объект, так что трансформация является единственной причиной изменения состояния информационного объекта.

Особенности декомпозиции модели процесса

Необходимо учитывать следующие общесистемные принципы декомпозиции. Разделение системы на подсистемы должно выполняться по определенному основанию. В методологии SADT выделяют следующие стратегии декомпозиции: 1) функциональную, 2) структурную, 3) по этапам жизненного цикла, 4) по физическому процессу [16]. Две последние относятся к темпоральной моде, описывают историю некоторого объекта: одна рассматривает значительные временные интервалы и фиксирует существенные изменения, происходящие в объекте, другая — короткие промежутки и фиксирует любые изменения. Выбор стратегии предопределяет результат, так что функциональная декомпозиция приводит к функциональным моделям, а темпоральная — к процессным [17]. Следует использовать одно основание деления для всех уровней декомпозиции — избрав определенную моду, необходимо придерживаться ее до тех пор, пока исходный объект не будет полностью раскрыт [19]. Использование одновременно нескольких оснований деления недопустимо, так как может привести к пересечению объема понятий или к потере элементов в составе системы. Необходимо обеспечить следующее: а) непрерывность деления — при разбиении делимого необходимо последовательно переходить от уровня декомпозиции раскрытого последним к следующему, не перескакивая к уровням, относящимся к другому порядку; б) бездефектность — каждый нижележащий уровень должен раскрывать предыдущий полностью, не упустив ни одного элемента; в) безызбы-

точность — недопустимо добавлять в ходе деления то, чего в оригинале нет [18].

Вначале рассмотрим процесс, образованный простым неделимым объектом, который изменяется под действием операций, исполняемых в определенном порядке. С процессом целиком и с каждой его операцией связаны внешние события, управляющие исполнением операций. Ограничимся рассмотрением детерминированного процесса, так что состояние объекта в любой момент времени определяется его исходным состоянием и теми операциями, которые его изменяют. В этом случае операции могут исполняться только последовательно, в противном случае мы не сможем указать, результатом какой операции является текущее состояние объекта.

Теперь предположим, что объект является комплексным, может быть разделен на элементы меньшего размера. В этом случае исходный процесс распадается на соответствующее число подпроцессов, каждый описывает историю отдельного субобъекта, при этом операции процесса разделяются так, что каждая связана с соответствующим субобъектом. Аналогично разделяется внешний объект окружения. Таким образом, декомпозиция процесса предполагает разделение информационного объекта на образующие его компоненты, разложение работы на простые операции, разложение событий на элементарные. Рассмотрим эти три декомпозиции по отдельности, обращая внимание на случаи, когда в результате будет нарушена детерминация.

Декомпозиция информационного объекта процесса

Декомпозиция информационного объекта выполняется по функциональному принципу. Если придерживаться общесистемных критериев декомпозиции, следует так разделить исходный объект, чтобы субобъекты были независимы друг от друга — не имели общих (разделяемых) свойств (элементов данных) [20]. Однако при моделировании процессов это условие часто не выполняется,

так что два или более субобъекта имеют общие данные. Это означает, что, изменяя один из них, мы автоматически меняем второй (в части общих для них элементов данных). В результате декомпозиции оба информационных объекта окажутся в разных подпроцессах, в каждом из которых операции исполняются асинхронно, так что возникнет коллизия, когда два подпроцесса одновременно изменяют разделяемый фрагмент данных, так что невозможно определить, результатом какой из операций является состояние информационного объекта в данный момент времени (1А). Будем считать связь, возникающую между двумя подпроцессами, которые обращаются к разделяемому информационному объекту, сильной, поскольку происходит нарушение детерминации.

Чтобы исключить взаимовлияние через разделяемые объекты, подпроцессы следует синхронизировать [21]. Можно предложить не исполнять оба подпроцесса одновременно, а только поочередно. Именно так работают так называемые последовательные процессы [21]. В этом случае детерминация не нарушается, связь между последовательными процессами остается слабой.

Декомпозиция работ, образующих процесс

Рассмотрим различные стратегии декомпозиции работ процесса. Если используется функциональная стратегия, то ее результатом

является функциональная декомпозиция работ (*work breakdown structure*), другое название — библиотека функций [22]. Результат такой декомпозиции — бездефектная и безызыбыточная иерархическая модель, в которой работы на одном уровне иерархии независимы друг от друга. Функциональная декомпозиция — мощное средство для выявления пропущенных, избыточных, дублирующих или ненужных работ процесса [23], однако она приводит к функциональным моделям, тогда как нас интересуют процессные.

При декомпозиции работ используется темпоральная мода декомпозиции, при этом вначале выбирается объект наблюдения, затем рассматривается последовательность операций, изменяющих этот объект. Если каждая работа процесса связана только с одним объектом, причем последний не имеет общих, разделяемых элементов данных, то результирующие подпроцессы окажутся полностью взаимонезависимыми, так что они образуют агрегат, а не систему (рис. 1, Б). Однако часто работы разных процессов могут «пересекаться». Рассмотрим два сценария. В первом случае один объект связан с двумя работами, которые в результате декомпозиции оказались в разных подпроцессах, при этом обе асинхронно изменяют этот объект (рис. 1, В). Поскольку мы не знаем, в каком порядке выполняются эти операции, детерминация нарушается. Например, поток управления процесса разветвляется на логической опера-

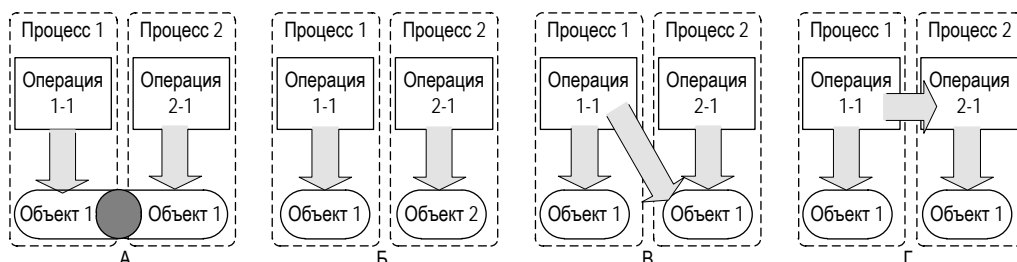


Рис. 1. Связь между подпроцессами: а — через разделяемый информационный объект; б — отсутствует; в — перекрестное влияние; г — взаимовлияние работ

Fig. 1. The interaction of subprocesses: а — via shared information object; б — absent; в — cross link; г — superposition of operations

ции «И», при этом информационный поток не делится, так что две параллельные операции изменяют один информационный объект. Возникает перекрестное влияние двух операций разных подпроцессов на один информационный объект, происходит нарушение детерминации, связь можно квалифицировать как сильную.

Во втором случае операция одного процесса оказывает воздействие на операцию другого процесса — одна работа изменяет другую (рис. 1, Г). Например, если одна программа выходит за пределы отведенного ей адресного пространства, она может изменить код другой программы. Именно так ведут себя «зловредные» программы, когда пытаются нарушить безопасность ИТ системы. Вследствие взаимовлияния работ происходит потеря детерминации; поскольку операции выполняются асинхронно, мы не знаем, является ли текущее состояние объекта результатом оригинальной или модифицированной трансформации, поэтому будем считать связь между подпроцессами сильной.

Декомпозиция событий

Будем трактовать событие как операцию одного процесса, которая оказывает воздействие на операцию другого процесса, но не изменяет ее, а инициирует или приостанавливает исполнение. Будем говорить, что операция одного процесса (генератор) управляет операцией другого (обработчик). Рассмотрим два случая декомпозиции события «генератор». Генерирующее событие ранее сопоставлено

с объектом окружения, который достигает некоторого предопределенного состояния. Если предположить, что этот объект комплексный, то его можно разложить на субобъекты меньшего размера (рис. 2). Теперь важно решить, является внешний объект агрегатом или системой. Состояние агрегата равно сумме состояний образующих его элементов, при этом простое генерирующее событие превращается в комплексное. Поскольку состояние системы не аддитивно, выбранное предопределенное состояние не может быть сведено к сумме состояний каждого из субобъектов в отдельности. В результате нельзя понять, достигается ли требуемое состояние, происходит нарушение детерминации — можно говорить, что связь сильная.

Если генерирующее событие управляет отдельной операцией, то наступление этого события оказывает влияние только на эту операцию, а если оно управляет процессом, то оказывает влияние на все операции этого процесса. Например, событие, заключающееся в том, что клиент отозвал свой заказ, так что дальнейшее исполнение процесса нецелесообразно, прерывает исполнение процесса, на какой бы операции он не находился. Обработчики, которые первоначально находились в рамках одного процесса, в результате декомпозиции могут оказаться в разных подпроцессах, при этом связь генерирующего события с обрабатывающим должна сохраниться (рис. 3). Поскольку детерминация не нарушается, будем считать эту связь слабой.

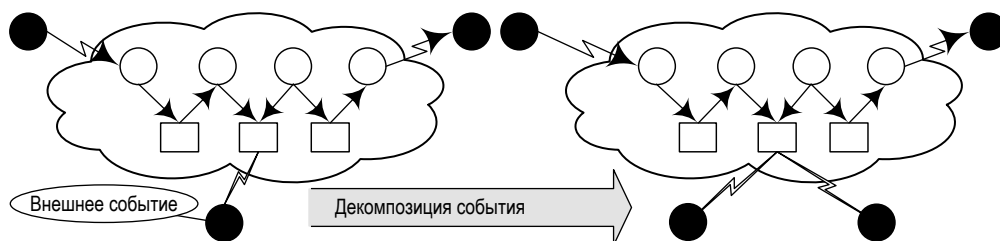


Рис. 2. Декомпозиция внешнего события

Fig. 2. The decomposition of an external event

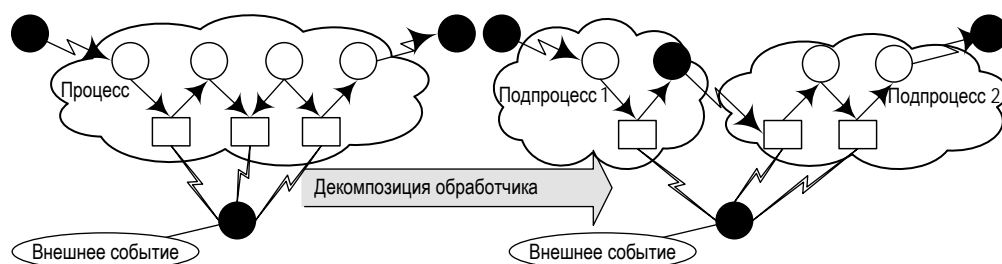


Рис. 3. Декомпозиция обрабатывающего события

Fig. 3. The decomposition of a cathing event

Рассмотрим декомпозицию обработчика события. Предположим, что обработчик — процесс, образованный некоторым комплексным объектом (рис. 4, А). Внешнее стартовое событие инициирует последовательность трансформаций этого объекта, с каждым связано внутреннее событие. Исполнение процесса прекратится, когда произойдет завершающее событие, оно является внутренним для данного процесса и внешним для следующего в цепочке. Теперь представим себе, что процесс декомпозировали на два подпроцесса (рис. 3, Б); в результате событие, которое первоначально было внутренним, превращается во внешнее для второго подпроцесса. Поскольку детерминация не нарушилась, можно считать связь слабой.

Проведенный анализ позволяет сделать следующие выводы. Во-первых, причиной нарушения детерминации процесса является пересечение субобъектов и подпроцессов, образованных в результате декомпозиции. Сильная связь между процессами воз-

никает в следующих случаях: а) когда объекты имеют разделяемые элементы данных; б) вследствие перекрестного влияния, когда один объект изменяется операциями, относящимися к разным процессам; в) в случае взаимовлияния операций разных процессов, когда одна операция изменяет работу другого процесса; г) когда генерирующее событие декомпозируется без учета связей внутри информационного объекта окружения. Во-вторых, природа связи внутри процесса, называемая связностью, и между подпроцессами, называемая сцеплением [24], одна и та же и обусловлена пересечением объемов данных и работ разных подпроцессов, в результате чего нарушается детерминация. В-третьих, сильно связанные подпроцессы нельзя рассматривать по отдельности, но только совместно, накладывая дополнительные условия на их синхронизацию. Например, если два объекта имеют общее поле данных, то трансформации, которые их изменяют, нельзя выполнять одновременно, но только поочередно.

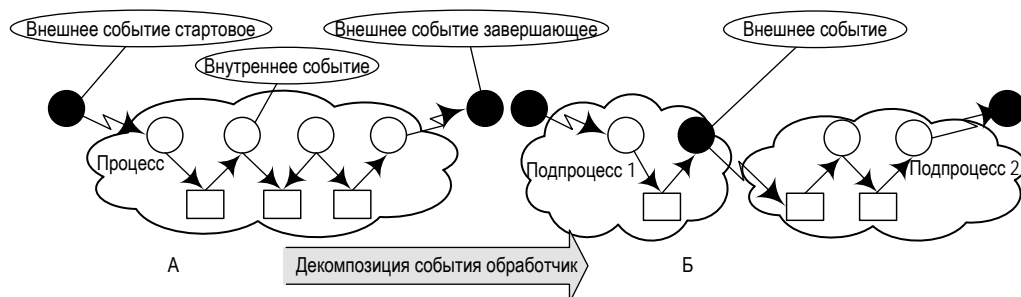


Рис. 4. Внутреннее событие в результате декомпозиции превращается во внешнее

Fig. 4. The transformation of internal event into external as a result of decomposition

Критерии «хорошей» декомпозиции Я. Ванда и Р. Вебера

В рамках онтологии моделирования информационных систем Я. Ванд и Р. Вебер предложили критерии «хорошей» декомпозиции для модели процесса [9]. К сожалению, смысл этих критериев не всегда ясен, трактовки неоднозначны [10]. Рассмотрим существующие трактовки, дадим им новую интерпретацию в свете полученных результатов.

Минимальность — в любой подсистеме на любом уровне декомпозиции отсутствуют избыточные переменные состояния [25]. В других трактовках критерий означает, что всякая подсистема на любом уровне декомпозиции использует минимальное число переменных состояния [25], у подсистемы наименьший возможный размер переменной состояния; в переменной состояния отсутствуют избыточные данные, для которых не предусмотрены соответствующие методы; переменная должна измениться хотя бы раз под действием работ процесса [26]. Дадим новое толкование — каждый подпроцесс на любом уровне имеет свою переменную состояния, минимально зависящую от других подпроцессов.

Отсутствие потерь и дублирования — расчленение не должно привести к утере существующих свойств системы или появлению новых. Это требование трактуют как необходимость декомпонировать данные процесса так, чтобы не потерять существующие переменные состояния и не создать новых, поэтому его часто относят только к данным процесса [10]. Мы будем считать, что безызыточность означает также отсутствие дублирования работ процесса. Учтем, что отдельный подпроцесс может быть повторно используемым и входить несколько раз в процесс более высокого уровня. Обратим внимание, что стратегия декомпозиции по функциям исключает дублирование, тогда как декомпозиция по порядку исполнения допускает повторно

используемые модули [17]. Рекомендуем аналитикам проводить не только процессную декомпозицию, но строить также дерево функций — это облегчит им поиск дублирующих функций [23].

Детерминизм событий — предопределенность поведения системы. Я. Ванд и Р. Вебер трактуют детерминацию исключительно применительно к событиям процесса, так что любое внешнее событие для элемента декомпозиции на любом уровне иерархии должно либо «быть хорошо определенным» внутренним для другого элемента декомпозиции, либо внешним для всей системы. Мы трактуем детерминацию шире, применительно ко всем элементам модели процесса. Потеря детерминации, по нашему мнению, является признаком «плохой» декомпозиции процесса.

Минимальная связность и максимальное сцепление. Я. Ванд и Р. Вебер считают, что низкая связность обусловлена тем, что мощность множества входов системы меньше или равна мощности суммы входов подсистем, образующих нижний уровень [25]. В модели процесса входы рассматриваются как элементы данных, поэтому связность трактуется как количество связей между элементами данных, образующих входы разных процессов. Они называют процесс максимально сцепленным, если набор его выходов включает все выходные переменные, затронутые входными переменными. Добавление любого другого выхода в набор не расширяет набор входов, от которых зависят существующие выходы, — нет других выходов, которые зависят от входа. Каждый входной объект данных этого подпроцесса не является входным для любого другого подпроцесса на том же уровне модели [27]. Последнее определение подозрительно напоминает связность. Мы установили, что сцепление и связность имеют общий механизм возникновения, обусловленный пересечением объемов данных и работ разных подпроцессов, в результате чего на-

рушается детерминация, так что рассматривать связанные подпроцессы по отдельности нельзя — только совместно. В отдельных случаях детерминацию можно сохранить, если ограничиться последовательным исполнением подпроцессов.

Отметим, что предложенные Я. Вандом и Р. Вебером критерии, с учетом сделанных выше замечаний, — необходимое, но не достаточное условие «хорошей» декомпозиции. Их правильнее трактовать как набор эмпирических правил, помогающих получить требуемый результат. В качестве достаточного условия «хорошей» декомпозиции процесса в данной работе предлагается исследовать вопрос об эквивалентности исходной модели процесса и полученной в результате декомпозиции. Модель потоков работ можно рассматривать как алгоритм, который, согласно определению А. П. Ершова, есть «точное предписание исполнителю осуществить последовательность действий, направленных на достижение указанной цели или на решение предлагаемой задачи». Для оценки эквивалентности моделей можно применить предложенные им критерии [28]:

- функциональная (трансформационная) — операции, реализуемые процессами, имеющими одинаковые множества аргументов, совпадают;
- операционно-логическая (структурная, трассовая) — в обоих процессах совпадают последовательности пройденных операций и логических операторов;
- логико-термальная (поведенческая) — для любой трассы одного процесса существует трасса в другом, такая, что логические условия проверяются в одной и той же последовательности для одних и тех же наборов значений переменных.

Если цель моделирования — реинжиниринг, то процессы «как есть» и «как должно быть» функционально эквивалентны, оба выполняют одинаковое преобразование, но по-разному. К сожалению, доказательство функциональной эквивалентности алгорит-

мов в общем случае невозможно [28]. В тех случаях, когда стоит задача раскрыть и смоделировать неизвестный процесс (*process mining*), ставится задача достичь трассовой эквивалентности. Однако этот тип эквивалентности не гарантирует достижения процессом требуемого результата. Для моделей, используемых для автоматизации, необходимо добиться поведенческой эквивалентности таким образом, чтобы модель процесса в точности повторяла реальную работу в организации.

При декомпозиции мы исключили те случаи, когда сильно связанные компоненты оказываются в разных компонентах, исполняемых асинхронно. Таким образом, можно утверждать, что либо в декомпозированном процессе отсутствуют разделяемые данные, взаимовлияние работ, неопределенные внешние события, либо он является последовательным процессом. Таким образом, можно утверждать, что существует логико-термальная эквивалентность исходного и декомпозированного процессов.

Принципы «правильной» декомпозиции модели процесса

Проведенное исследование позволяет сформулировать принципы декомпозиции, следование которым позволит разделить модель на подпроцессы «правильным» образом:

- 1) результатом декомпозиции детерминированного процесса должно стать конечное множество детерминированных подпроцессов;
- 2) критерием «правильной» декомпозиции является поведенческая эквивалентность исходной и полученной в результате декомпозиции моделей;
- 3) декомпозиция работ и данных процесса выполняется согласованно;
- 4) декомпозиция начинается с деления данных, выделения в них важных в аспекте результата сущностей;

5) недопустимо пересечение информационных объектов, находящихся на одном уровне декомпозиции (если пересечения избежать не удастся, не допускать одновременного исполнения соответствующих сцепленных подпроцессов);

6) декомпозировать работы процесса следует таким образом, чтобы в один подпроцесс попадали операции, изменяющие один объект;

7) следует избегать перекрестного влияния операций разных подпроцессов на один информационный объект (если перекрестного влияния избежать не удастся, исполнять эти подпроцессы только поочередно);

8) нельзя допускать взаимного влияния операций разных подпроцессов друг на друга (если взаимовлияния избежать не удастся, то такие подпроцессы нельзя рассматривать изолированно, но только совместно, исполнять поочередно);

9) смена объекта, образующего процесс, сигнализирует аналитику о необходимости разделить сквозной процесс на подпроцессы;

10) в результате декомпозиции событие-генератор не должно потерять связь с обработчиком;

11) следует принимать во внимание, что декомпозиция объекта может оказать опосредованное влияние на другой процесс, для которого данный информационный объект является генератором события.

Выводы

Главный вывод исследования — работы и данные бизнес-процесса следует декомпозировать согласованно, а не раздельно, как это делалось ранее. Неверно понимать декомпозицию как выявление состава системы. Следует обязательно учитывать связи между модулями — если они имеют сложный характер, их анализ может свести на нет преимущество декомпозиции. Сильно связанные компоненты нельзя рассматривать по отдельности, но только совместно.

Причиной сильной связи между подпроцессами является пересечение субобъектов и подпроцессов, образованных в результате декомпозиции. Она возникает в следующих случаях:

а) когда объекты имеют разделяемые элементы данных;

б) вследствие перекрестного влияния, когда один объект изменяется операциями, относящимися к разным процессам;

в) в случае взаимовлияния операций разных процессов, когда одна операция изменяет работу другого процесса;

г) когда генерирующее событие декомпозируется без учета внутренних связей.

Предложены принципы, выполнение которых является необходимым условием «правильной» декомпозиции. В качестве достаточного условия качества декомпозиции предлагается использовать поведенческую (логико-термальную) эквивалентность исходной и полученной в результате декомпозиции моделей процессов.

Достоверность результатов исследования подтверждается использованием согласованной онтологии моделирования бизнес-процессов и системного подхода, основанного на теоретических работах М. Бунге. Принципиально важным является его замечание о различии агрегата и системы, а также о необходимости учитывать внутренние связи между объектами системы и внешние связи между системой и ее окружением. Новизна исследования заключается в рассмотрении природы связи между элементами декомпозированного подпроцесса. Для оценки силы этой связи предложено использовать свойство детерминации, причем потеря этого признака означает, что компоненты нельзя рассматривать по отдельности — только совместно.

Практическая ценность результата для прикладной информатики заключается в том, что предложенные принципы декомпозиции помогают превратить работу аналитика в инженерную деятельность, улучшить качество

моделей, способствуют успеху применения современных ИТ. Модульная структура процесса окажется удобной при автоматизации бизнеса, облегчит разработку ИТ архитектуры предприятия. Предложенный в работе подход лег в основу метода проектирования архитектуры модели бизнес-процессов, который на практике демонстрирует хорошие результаты [29]. Полученные структурированные модели оказываются полезными и удобными для анализа разными категориями пользователей. Для бизнеса станет ясна суть процесса при изучении диаграмм верхнего уровня, эксперты предметной области смогут разобраться в подробностях, используя детализацию, разработчики найдут важные подробности на диаграммах нижнего уровня. Повышение качества моделей процесса позволяет уверенно добиваться успеха при выполнении проектов по внедрению новых ИТ.

Список литературы

1. *Becker J., Thome I., Weiß B., Winkelmann A.* Constructing a semantic business process modeling language for the banking sector // *EnterpriseModelling and Information Systems Architectures*. 2010. Vol. 5. No. 1. P. 4–25.
2. *Mendling J., Reijers H., van der Aalst W.* Seven process modeling guidelines // *Information and Software Technology*. 2010. Vol. 52. No. 2. P. 127–136.
3. *Bobrik R., Reichert M., Bauer T.* View-Based Process Visualization // *Lecture Notes in Computer Science*. 2007. No. 4714. P. 88–95.
4. *Smirnov S., Reijers H., Weske M.* A semantic approach for business process model abstraction // *Lecture Notes in Computer Science*. 2011. No. 6741. P. 497–511.
5. *Reijers H., Mendling J., Dijkman R.* Human and automatic modularizations of process models to enhance their comprehension // *Information Systems*. 2011. Vol. 36. No. 5. P. 881–897.
6. *Langlois R.* Modularity in technology and organization // *Journal of Economic Behavior & Organization*. 2002. Vol. 49. P. 19–37.
7. *Parnas D.* On the Criteria To Be Used in Decomposing Systems into Modules // *Communications of the ACM*. 1972. December. Vol. 15. No. 12. P. 1053–1058.
8. *Johannsen F., Leist S.* Reflecting modeling languages regarding Wand and Weber's Decomposition Model // *Proceedings Modellierung*. Bamberg. 2012. P. 27–42.
9. *Wand Y., Weber R.* An Ontological Model of an Information System // *IEEE Transactions on Software Engineering*. 1990. November. Vol. 16. No. 11. P. 1282–1292.
10. *Burton-Jones A., Meso P.* Conceptualizing Systems for Understanding: An Empirical Test of Decomposition Principles in Object-Oriented Analysis // *Information Systems Research*. 2006. No. 17. P. 38–60.
11. *Reijers H., Mendling J., Dijkman R.* On the Usefulness of Subprocesses in Business Process Models, BPM center report, BPMcenter.org, Eindhoven, BPM-10-03, 2010.
12. *Bunge M.* Treatise on Basic Philosophy. Ontology II: A the World of Systems. Dordrecht: D. Reidel Publishing Company. — 379 p.
13. *Bunge M.* Treatise on Basic Philosophy. Ontology I: The Furniture of the World. Vol 3. Boston, MA: D. Reidel Publishing Company, 1977. — 369 p.
14. *Доброхотов А. Л.* Новая философская энциклопедия. 2 изд. М.: Мысль, 2010. Т. 4
15. *Фёдоров И. Г.* Адаптация онтологии Бунге-Ванда-Вебера к описанию исполняемых моделей бизнес-процессов // *Прикладная информатика*. 2015. Т. 58. №. 4. С. 82–92.
16. *Марка Д., МакГоуэн К.* Методология структурного анализа и проектирования SADT. М.: Метатехнология, 1993. — 240 с.
17. *Тельнов Ю. Ф., Фёдоров И. Г.* Функциональные и процессные модели бизнес-процессов // *Экономика. Статистика. Информатика*. 2012. № 2. С. 193–199.
18. *Шадрин Д. А.* Логика: конспект лекций. М.: Эксмо, 2008. — 190 с.
19. *Хорошев А. Н.* Введение в управление проектированием механических систем. URL http://www.cfin.ru/management/controlling/sys_project.shtml ed. Белгород. 1990. — 372 с.
20. *Moody D.* A decomposition Method for Enterprise Relationship Models: A System Theoretic Approach // *Proceedings of the Twenty-First International Conference on Information Systems, ICIS 2000*. Brisbane, Australia. 2000 December 10–13. P. 462–469.
21. *Дейкстра Э.* Взаимодействие последовательных процессов // *Сб. Языки программирования*. М.: Мир, 1972. — 86 с.
22. *Owen J.* Business Function Modelling eBook // *John Owen's Integrated Modelling Method*. 2009. URL: <http://integrated-modeling-method.com/imm-bpm-business-process-modeling-store/business-function-modeling-ebook/> (дата обращения: 15.01.2014).
23. *Фёдоров И. Г.* Системный подход к выявлению бизнес-процессов методом «сверху вниз» // *Прикладная информатика*. 2012. Т. 41. №. 5. С. 5–13.

24. Калянов Г. Н. Теория и практика реорганизации бизнес-процессов. Синтер, 2000. — 203 с.
25. Wand Y., Weber R. An Ontological Model of an Information System // *IEEE Transactions on software engineering*. 1999. Vol. 16. No. 11. P. 1282–1292.
26. Burton-Jones A., Meso P. How Good Are These UML Diagrams? An Empirical Test of the Wand and Weber Good Decomposition Model // *Twenty-Third International Conference on Information Systems*. 2002. P. 101–114.
27. Wand Y., Weber R. Toward a theory of the deep structure Of information systems // *Journal of Information Systems*. 1995. Vol. 5. No. 3. P. 203–223.
28. Еришов А. П. Основы информатики и вычислительной техники: Пробное учебное пособие для средних учебных заведений. В 2 ч. / под редакцией А. П. Еришова и В. М. Монахова. М.: Просвещение, 1985. — 96 с.
29. Фёдоров И. Г. Проектирование модели бизнес-процессов // *Открытые системы, СУБД*. 2013. №. 5. P. 46–49.
30. Fiedorow I. G. The adaptation of a Bunge-Wand-Weber ontolgy or a business process modeling. *Prikladnaya Informatika — Journal of Applied Informatics*, 2015, vol. 58, no. 4, pp. 82–92 (in Russian).
31. Fiedorow I. G. The adaptation of a Bunge-Wand-Weber ontolgy or a business process modeling. *Prikladnaya Informatika — Journal of Applied Informatics*, 2015, vol. 58, no. 4, pp. 82–92 (in Russian).
32. Marca D., McGowan C. SADT: structured analysis and design technique. McGraw-Hill Book Co., Inc.: New York, NY, 1988. 240 p.
33. Telnov Y. F., Fiodorov I. G. A functional and a process models of a buiness process // *Economics, Statistics, Informatics*, 2012, no. 2, pp. 193–199 (in Russian).
34. Shadrin D. A. Logics: a lecturer notes. Moscow, Exsmo Publ., 2008. 190 p. (in Russian).
35. Khoroshev A. N. An introduction into a nechanical engineering. URL http://www.cfin.ru/management/controlling/sys_project.shtml ed. Belgorod, 1990. 372 p. (in Russian).
36. Moody D. A decomposition Method for Enterprise Relationship Models: A System Theoretic Approach. // *Proceedings of the Twenty-First International Conference on Information Systems, ICIS 2000*. Brisbane, Australia. 2000 December 10–13, pp. 462–469.
37. Dijkstra E. Cooperating Sequential Processes. Reprinted in F. Genuys (ed.), *Programming Languages*, Academic Press, Orlando, Florida, 1968, pp. 43–112.
38. Owen J. Business Function Modelling eBook. John Owen's Integrated Modelling Method. 2009. Available at: <http://integrated-modeling-method.com/imm-bpm-business-process-modeling-store/business-function-modeling-ebook/> (date of access: 15.01.2014).
39. Fiodorov I. G. A system approach to a process dicoverly from top to bottom. *Prikladnaya Informatika — Journal of Applied Informatics*, 2012, vol. 41, no. 5. pp. 5–13 (in Russian).

References

24. Kalyanov G. N. *A theory and practics of a business process reorganization*. Sinteg, 2000. 203 p. (in Russian).
25. Wand Y., Weber R. An Ontological Model of an Information System. *IEEE Transactions on software engineering*, 1999, vol. 16, no. 11, pp. 1282–1292.
26. Burton-Jones A., Meso P. How Good Are These UML Diagrams? An Empirical Test of the Wand and Weber Good Decomposition Model. *Twenty-Third International Conference on Information Systems*, 2002, pp. 101–114.
27. Wand Y., Weber R. Toward a theory of the deep structure Of information systems. *Journal of Information Systems*, 1995, vol. 5, no. 3, pp. 203–223.
28. Ershov A. P. *A foundation of informatics and computer engineering*. In 2 parts, eds A. P. Ershov and V. M. Monakhova. 1985. Prosveshchenie Publ. 96 p. (in Russian).
29. Fiodorov I. G. A design of business process models. *Open systems*, SUBD, 2013, no. 5, pp. 46–49 (in Russian).

I. Fiodorov, *Plekhanov Russian University of Economics, Moscow, Russia, Igor.Fiodorov@mail.ru*

A principles of a process model decomposition

One of the actual problems of applied informatics is the issue of a good decomposition of a business process model. Almost any applied project in the field of IT development includes modeling of a business processes, so that their success depends significantly on the quality of the models used for the analysis, re-engineering and automation. The decomposition of business process model is a key problem in the development of IT enterprise architecture. While this problem is not yet solved, the work of an analyst remain a craft, while engineering approach is required. The purpose of this article is to analyze the theoretical basis and in the context of the Bunge-Wanda Weber ontology, develop the principles of a decomposition, which can be considered as a «good» one, suggest criterion which will help to achieve a proper model decomposition. It is wrong to understand decomposition as an identification of components of a system. One should take into consideration the links between its modules — if the connections are complex, their analysis may negate the advantage of decomposition. Strongly connected components cannot be considered separately, but only jointly. The reason for the strong connection is the intersection of an information objects and sub-processes, resulting decomposition. The peculiarity of this research is that we consider a coordinated decomposition of data objects, process operations and external events.

Keywords: process model decomposition, enterprise IT architecture, Bunge–Wand–Weber ontology, coupling, cohesion.

About author:

I. Fiodorov, *PhD in Computer Science*

For citation:

Fiodorov I. A principles of a process model decomposition. *Prikladnaya Informatika — Journal of Applied Informatics*, 2016, vol. 11, no. 5 (65), pp. 19–30 (in Russian).