

АВТОМАТИЗАЦИЯ ВЫДЕЛЕНИЯ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ ИЗ SADT-ДИАГРАММ

О.А. ЗМЕЕВ, А.Ю. МАЛИНОВСКИЙ

Введение

В настоящее время развитие программного обеспечения идет в сторону усложнения и укрупнения разрабатываемых информационных систем. Эта тенденция связана, с одной стороны, с тем, что современные компьютеры становятся все мощнее, и это позволяет их пользователям предъявлять повышенные требования к кругу задач, которые решаются с их помощью. В качестве другой причины можно указать все возрастающее использование технологий Интернет, которые за чрезвычайно короткий промежуток времени прошли путь от простого форматированного текста к полнофункциональным мультимедийным приложениям. С другой стороны, существует другая тенденция, согласно которой время, отпущенное на разработку программного обеспечения (ПО), существенно сокращается. В этих условиях одной из центральных проблем индустрии ПО становится проблема организации процесса разработки ПО.

В последнее время в индустрии разработки ПО часто используются процессы, основанные на методологии Унифицированного процесса разработки программного обеспечения (UP). К числу таких процессов относится RUP (Rational Unified Process) [1], появившийся в 1995 году в результате слияния процессов Rational Approach и Objectory Process 3.8, процесс ICONIX [2, 3], разработанный Д. Розенбергом, к числу таких процессов достаточно большое число авторов относит и процесс XP [4]. Полностью теоретические концепции, лежащие в основе UP, изложены в [5], в [6] приведен краткий обзор основных идей, на которых базируется процесс.

Одной из ключевых концепций UP является понятие *варианта использования* (*use case*). Обычно UP характеризуется как управляемый вариантами использования, это означает, что команда разработчиков применяет варианты использования для выполнения полного комплекса работ по проекту, начиная с начального сбора информации и заканчивая тестированием готовой программной системы. К сожалению, вопросы, связанные с формированием модели вариантов использования для построения программных систем, на сегодня являются наиболее сложной и наименее формализованной частью UP.

К тому же достаточно часто проекты, связанные с разработкой и внедрением различного рода информационных систем, происходят в рамках процедуры реинжиниринга бизнес-процессов организации клиента. Ключевыми артефактами этой процедуры являются так называемые модели «IT IS» и «TO BE», для получения которых бизнес-аналитики чаще всего используют методологию SADT [7, 8].

В настоящей работе предлагается подход, позволяющий использовать для построения первого приближения модели вариантов использования диаграммы, полученные в результате SADT-моделирования. Если методология SADT применялась для разработки функциональной модели целевой организации в целом, то предлагаемый подход позволяет построить первое приближение для модели деловых (бизнес) вариантов использования. Если соответствующая SADT-модель декомпозирует функциональность целевой системы, то в качестве результата может быть получено первое приближение модели вариантов использования в обычном смысле. Такой подход, на наш взгляд, с одной стороны, увеличивает степень повторного использования артефактов процесса реинжиниринга, а с другой – позволяет получить первое приближение модели вариантов использования более высокого качества и за более короткие сроки.

Особенности и проблемы построения модели вариантов использования в рамках UP

Как уже упоминалось выше, методология UP строится вокруг вариантов использования, причем одной из ключевых задач этого артефакта процесса разработки является формальное описание множества функциональных требований к будущей программной системе. В рамках UML для описания модели вариантов использования применяются диаграммы вариантов использования, которые определяют структурную организацию модели требований и отношения между ними, сами же сценарии взаимодействия актера с программной системой предлагается моделировать с использованием различного вида диаграмм взаимодействия.

Такой подход не нашел широкой поддержки у специалистов по написанию вариантов использования, которые, справедливо полагая, что диаграммы взаимодействия не наглядны и не имеют всей функциональности текстового описания варианта использования, предпочитают описывать сценарии вариантов использования в текстовом виде [9]. Выделение вариантов использования проводится с помощью опросов заинтересованных лиц, мозговых штурмов и т.д., то есть, создавая модель вариантов использования, аналитик одновременно создает и высокоуровневую модель будущей или уже существующей системы.

Для анализа предметной области разрабатываемой системы в рамках UP предлагается технологический процесс бизнес-моделирования, который основан на той же концепции вариантов использования, но в качестве системы в этом случае выступает не разрабатываемая программная система, а сама целевая организация. И хотя процесс перехода от бизнес-вариантов использования к вариантам использования в обычном смысле достаточно формализован [1], разработка подобного рода бизнес-моделей сопряжена с дополнительными трудностями, основной причиной которых является нестандартный для большинства бизнес-аналитиков аппарат.

Существует метод моделирования бизнес-процессов, использующий процессный подход, аналогичный SADT, – метод Ericsson – Penker [10, 11]. Авторы соз-

дали свой профиль UML для моделирования бизнес-процессов. Основной диаграммой, используемой в этом подходе, является диаграмма деятельности (activity diagram), которую авторы расширили с помощью механизма стереотипов. В результате получился процесс, альтернативный RUP, в котором не применяются варианты использования. Это решение непопулярно в среде разработчиков, поскольку механизм вариантов использования чрезвычайно удобен и хорошо себя зарекомендовал за годы существования.

Выделение вариантов использования из модели SADT

В качестве модели бизнес-процессов системы также может быть использована модель, построенная с использованием методологии SADT, которая хорошо известна бизнес-аналитикам и пользуется среди них большой популярностью. Существует ряд инструментов, которые автоматизируют построение моделей SADT, обычно для этого используется нотация стандарта IDEF0.

Процедура выделения вариантов использования из SADT-модели бизнес-процессов позволит использовать эту методологию в начальной стадии цикла разработки программного обеспечения для создания высокоуровневой модели бизнес-процессов и первоначального варианта модели вариантов использования. Эта высокоуровневая модель впоследствии будет углубляться путем детализации модели вариантов использования. Попытаемся выделить связи между моделью бизнес-процессов, построенной в нотации SADT, и моделью вариантов использования.

Проанализируем основные элементы обеих моделей, а затем попытаемся сформулировать некоторый набор правил, позволяющих установить соответствие между этими элементами.

Характерными элементами SADT-диаграмм являются: работа – функция или активная часть системы; вход – ресурсы, которые требуются системе; выход – ресурсы, которые были созданы в результате преобразования или потребления входных ресурсов; управление – стандарты и правила, руководящие работой; механизм – действующие силы и средства, выполняющие работу.

Для модели вариантов использования основными элементами являются: действующее лицо – участник, который обращается к системе за одной из ее услуг; цель – то, чего пытается добиться от системы действующее лицо; предусловия и гарантии – условия, которые должны быть истинными на момент активации варианта использования и после его окончания; основной сценарий – наиболее типичный набор последовательных шагов, позволяющий действующему лицу получить заданную цель; расширения – отклонения от основного сценария варианта использования.

Отображение элемента «работа»

Работа в нотации SADT чаще всего, будет соответствовать самому варианту использования. При этом отношение вида «декомпозируется» между работами превращается в отношение вида «включает» между соответствующими вариантами использования.

Уровень в иерархии диаграмм SADT определяет уровень цели варианта использования. Работы, которые не декомпозируются на других диаграммах, являются элементарными вариантами использования или, если говорить терминами, применяемыми разработчиками вариантов использования, имеют уровень элементарных подфункций (шагов). Работа, расположенная на диаграмме самого верхнего уровня, отображается на корневой вариант использования, имеющий самый высокий уровень. Остальные уровни могут быть определены примерно и впоследствии уточнены автором вариантов использования, осуществляющим преобразование.

Отображение элементов «вход»

Наличие ресурсов для выполнения варианта использования является условием, которое должно быть истинным. Соответственно входы в нотации SADT отображаются на предусловия в варианте использования.

Отображение элементов «выход»

Целью варианта использования является получение одного или нескольких ресурсов из числа описанных в выходах работы. То есть часть выходов в SADT-

диаграммах отображается на цели вариантов использования. Оставшиеся выходы представляют собой отдельные элементарные варианты использования, которые должны быть включены в вариант использования, соответствующий работе.

Отображение элементов «управление»

Вариант использования, как и работа, должен соответствовать определенным ограничениям и подчиняться определенным правилам, которые описываются в управлении работы. Условие соответствия этим правилам и ограничениям должно быть истинным на протяжении всего времени исполнения варианта использования. Соответственно управление в нотации SADT отображается на элемент «гарантии» в варианте использования.

Отображение элементов «механизм»

Механизмы в нотации SADT делятся на две части: исполнители работы и вспомогательные инструменты. В частности, к вспомогательным инструментам относится разрабатываемая система или ее часть. По той части системы, которая служит механизмом для работы, вариант использования, соответствующий этой работе, может быть отнесен к определенному пакету вариантов использования. То есть от механизмов, выполняющих данную работу, зависит то, какой группе разработчиков будет передан соответствующий вариант использования. Механизмы, относящиеся к исполнителям данной работы, должны быть отображены на действующих лиц варианта использования.

Основной сценарий и расширения варианта использования

Поскольку в нотации SADT не определен порядок выполнения работ, то построить пошаговый сценарий возможно только в том случае, если автор, осуществляющий преобразование, явно укажет порядок выполнения работ в системе. После этого можно создать основной сценарий в первом приближении, состоящий из включенных вариантов использования, которые соответствуют работам на декомпозирующей диаграмме.

Поскольку диаграмма в методологии SADT строится без учета возможных сбоев в системе, то в общем случае расширения на SADT-диаграмме отсутствуют.

Однако автор, выполняющий преобразование, может явно указать работы, а значит, и соответствующие им варианты использования, которые будут входить в расширение родительского варианта использования.

Пример преобразования SADT-диаграммы

В качестве примера выделения вариантов использования из SADT-диаграммы возьмем диаграмму, которая используется как образец в [7] (см. рис.1). Диаграмма является декомпозицией работы «Изготовить нестандартную деталь».

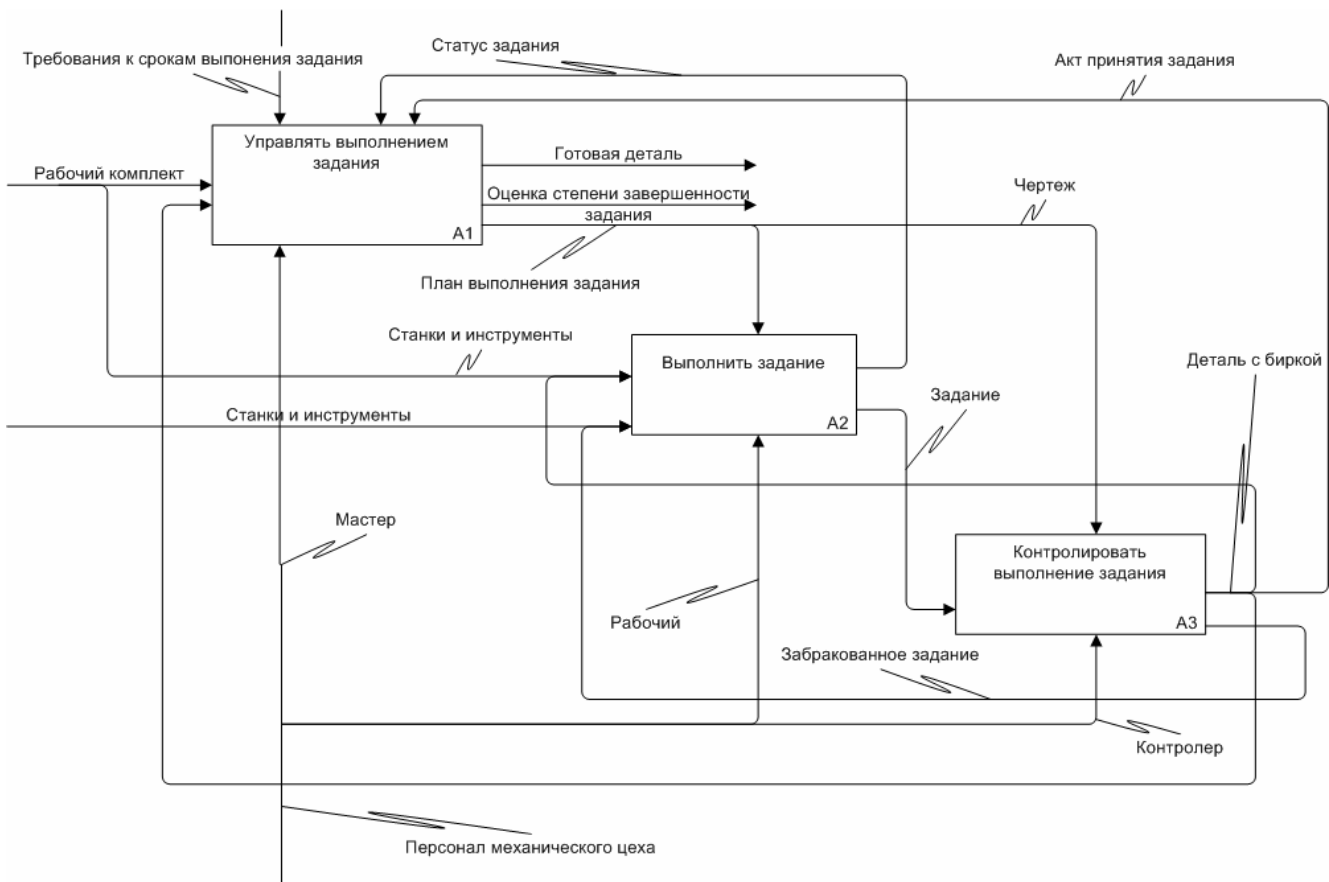


Рис. 1. Диаграмма, декомпозирующая работу «Изготовить нестандартную деталь»

Из этой диаграммы с помощью предложенной процедуры возможно выделить три сложных варианта использования, в дальнейшем они могут быть также декомпозированы, и четыре простых варианта использования. Сложные варианты использования: «Управлять выполнением задания», «Выполнить задание», «Контролировать выполнение задания»; простые варианты использования: «Оценить степень завершенности задания», «Разработать план выполнения задания»,

«Снабдить деталь биркой», «Отправить забракованное задание на повторное выполнение». Действующими лицами варианта использования «Изготовить нестандартную деталь» являются мастер, рабочий, контролер. Цель этого варианта использования – изготовление детали. В ходе варианта использования гарантируется то, что срок выполнения задания будет отвечать требованиям к срокам выполнения задания. Автор, выполняющий преобразование, может также указать варианты использования, относящиеся к основному потоку и к расширениям.

Заключение

При рассмотрении SADT-диаграмм и вариантов использования выявлены и сформулированы правила выделения вариантов использования из модели бизнес процессов, построенной в нотации SADT. Для того чтобы SADT-модель стала частью процесса разработки программного обеспечения, построенного вокруг вариантов использования, необходимо создать инструмент, позволяющий автоматизировать подобное выделение. Исходя из правил преобразования, нетрудно определить требования к инструменту автоматического преобразования:

- Возможность преобразования работы в варианты использования.
- Возможность задания уровня работы.
- Возможность преобразования входов в предусловия варианта использования.
- Возможность преобразования выходов в цели или во включенные варианты использования в зависимости от выбора.
- Возможность преобразования управлений в гарантии вариантов использования.
- Возможность преобразования выбранных механизмов в действующих лиц вариантов использования
- Возможность классификации вариантов по используемым механизмам.
- Возможность явного задания порядка выполнения работ и создания основного сценария.

– Возможность явного задания работ, входящих в расширение варианта использования.

Помимо этого, инструмент должен быть способен использовать готовые SADT-диаграммы, построенные в различных системах моделирования, также должен представлять собой полноценный редактор SADT-диаграмм.

ЛИТЕРАТУРА

1. Кратчет Ф. Введение в Rational Unified Process. –2-е изд. / Пер. с англ. –М.: Издательский дом «Вильямс», 2002. –240 с.
2. Rosenberg D., Scott K. Use Case Driven Object Modeling with UML: A Practical Approach. –Addison-Wesley, 1999.
3. Розенберг Д., Скотт К. Применение объектного моделирования с использованием UML и анализ прецедентов на примере разработки книжного Internet-магазина. –М.: ДМК Пресс, 2002.
4. Beck K. Extreme Programming Explained: Embrace Change. –Addison-Wesley, 2000.
5. Якобсон А., Буч Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения. –СПб.: Питер, 2002.
6. Скотт К. Унифицированный процесс. Основные концепции. –М.: Издательский дом «Вильямс», 2002. –150 с.
7. Марка Д.А., МакГоуэн К. Методология структурного анализа и проектирования SADT. – М.: 1993.
8. Маклаков С. Инструментальная поддержка разработки и внедрения корпоративных информационных систем // Компьютер-Пресс. –2001. –№9. С. 46-49.
9. Коберн А. Современные методы описания функциональных требований к системам : Пер. с англ. – М.: Лори, 2002. – 263 с.
10. Eriksson H. Business modeling with UML. –New York: John Wiley & Sons, 1998.
11. Norman O. Business modeling UML vs. IDEF. –Griffith: Griffith University, 2002.