

АНАЛИЗ ПОДХОДОВ К ФОРМАЛЬНОЙ СПЕЦИФИКАЦИИ ПРАВИЛ КОРПОРАТИВНОЙ БЕЗОПАСНОСТИ ИС НА ОСНОВЕ ОНТОЛОГИЙ

О.Р. Козырев,

профессор, директор Нижегородского филиала Государственного университета –
Высшей школы экономики,
e-mail: okozyrev@hse.ru,

Н.А. Климова,

заместитель директора по развитию и управлению Нижегородского филиала
Государственного университета – Высшей школы экономики,
e-mail: nklimova@hse.ru,

М.И. Литвинцева,

директор по организационной работе Государственного университета –
Высшей школы экономики,
e-mail: mlitvintseva@hse.ru.

Адрес: г. Нижний Новгород, ул. Б. Печерская, д. 25/12.

В статье рассматривается метод анализа бизнес-процессов на соответствие корпоративным правилам информационной безопасности с использованием реляционной логики и системы MIT Alloy Analyzer. С целью анализа определяется структура взаимосвязанных онтологий на трех уровнях и соответствующая логическая микротеория. Для иллюстрации предложенного метода используется бизнес-процесс реальной компании.

Практически все современные организации работают сегодня с распределенной ИТ-инфраструктурой, множеством разнородных приложений, реализованных на различных платформах и взаимодействующих между собой посредством набора интерфейсов. Традиционный подход (например, объектно-ориентированная технология CORBA) к интеграции представляет собой создание промежуточного программного слоя, который отве-

чает за объединение и налаживание коммуникации между разнородными приложениями. Однако новые потребности бизнеса диктуют новые условия для интеграции. Динамичность бизнеса требует от ИТ решений гибкости и простоты управления ИТ системами, что тяжело реализуемо в рамках традиционного подхода. Также возникает другая серьезная проблема – избыточность программных компонентов и сложность их многократного использования [1].

Для преодоления перечисленных проблем была предложена новая концепция интеграции бизнес-процессов на основе сервис-ориентированной архитектуры (СОА). Аналитики компании IBM дают следующее определение: «СОА — это прикладная архитектура, в которой все функции определены как независимые сервисы с вызываемыми интерфейсами. Обращение к этим сервисам в определенной последовательности позволяет реализовать тот или иной бизнес-процесс» [2].

К сожалению, практические выгоды от внедрения СОА до сих пор являются неочевидными и вызывают дискуссии в бизнес- и ИТ-сообществах. По результатам опроса, проведенного среди крупных производственных компаний консалтинговой компанией BearingPoint's Wall Street, около 58% респондентов ответили, что внедрение СОА внесло лишь дополнительную сложности в их ИТ ландшафт, нежели снизили ее; 30% отметили превышение затрат на СОА по сравнению с ожидаемым уровнем.

По результатам независимых научных исследований [3, 4, 5] выделяются следующие основные проблемы, возникающие при попытках построения СОА на предприятиях: несовместимость программных продуктов; отсутствие знаний по созданию описания сервисов; узкая направленность архитекторов на решение ИТ задач; неструктурированность, неполнота и ограниченность СОА стандартов; разобщенность бизнеса и ИТ.

Перечисленные выше проблемы в основном относятся к классу проблем управления и интеграции знаний на различных уровнях: как между людьми разных специализаций и уровней, так и между различными методологиями построения сервисов. Разнородность отдельных сервисов и необходимость моделирования различных аспектов всей распределенной системы вносит дополнительные сложности и затраты для внедрения сервис-ориентированных решений. Сейчас для создания информационных систем большинство разработчиков прибегают к моделированию в разных нотациях и стилях (BPMN, UML, EPC), и выполняют ручную или автоматизированную компоновку независимых сервисов, применяя формальное описание сервисов (WSDL) и языки описания потоков работ, таких как, например, WS-BPEL [6].

Одной из актуальных научных задач в области проектирования систем, основанных на СОА, является необходимость учета различных аспектов выполнения бизнес-процессов, не связанных напрямую с выполнением бизнес-функций. Например, одной

из важнейших проблем сервис-ориентированных решений является необходимость согласованности результатов выполнения автоматизируемого процесса с действующими организационными политиками и ограничениями. Одним из примеров таких ограничений являются правила и политики корпоративной информационной безопасности. По определению, данному в Национальном стандарте РФ, Информационная безопасность организации — это все аспекты, связанные с определением, достижением и поддержанием конфиденциальности, целостности, доступности, неотказуемости, подотчетности, аутентичности и достоверности информации или средств ее обработки [9].

В данной работе мы выбрали для изучения возможность моделирования организационных контролей информационной безопасности, обеспечивающих конфиденциальность, при автоматизированном или автоматическом построении последовательности запуска сервисов в составе сложных СОА-решений. Проблема необходимости моделирования организационных контролей была подробно проанализирована в [8]. Для структурированного анализа контролей разграничений полномочий, была введена следующая классификация:

1. Статическое разграничение полномочий. Невозможность принципалом обладать двумя эксклюзивными ролями.
2. Динамическое разграничение полномочий.
 - a. Простое динамическое разграничение полномочий. Невозможность принципалом активизировать две эксклюзивные роли.
 - b. Разграничение полномочий, основанное на объектах. Невозможность принципалом активизировать две эксклюзивные роли по отношению к одному объекту в один момент времени.
 - c. Операционное разграничение полномочий. Невозможность принципалом иметь несколько ролей, которые покрывают некоторый бизнес-процесс.
 - d. Разграничение полномочий, основанное на истории применения. Невозможность принципалом иметь все роли, обрабатывающие один и тот же объект.

Однако помимо правил и политик присвоения ролей одному принципалу и отслеживания уровня риска при выдаче авторизации, в [10] выделены авторизационные политики построения самих ролей.

В данной работе указанные подходы будут использованы для контроля авторизаций на уровне моделирования бизнес-процесса.

ние для целей моделирования организационных контролей — Обязательство (Obligation & ObligationInstance).

Кроме данных объектов вводятся понятия исполнителя бизнес-функции в качестве бизнес-роли (Role) и понятия обрабатываемого документа: Документа (DocumentInstance) и Типа Документа (DocumentType). Роли распределяются среди принципалов (Principal), работающих в рамках описываемого бизнес-процесса, которые могут быть представлены либо Человеком (Person), либо Системой (System). Для целей дальнейшего анализа организационных контролей информационной безопасности введены такие понятия: Политики (PolicyObject), Авторизации (Authorization), Обязательства (Obligation & ObligationInstance) и Критический Набор Авторизаций (Critical_Authorization_Set). Для введенных понятий на языке системы Alloy определяются ограничения (табл. 1).

Таблица 1.

Моделирование организационных контролей на языке Alloy

Тип контроля	Описание на языке Alloy
Shared/Shared	<pre>pred ss (disj r1, r2: Role) { r1->r2 in exclusive => some ((subject.r1 & Authorisation) - (subject.r2 & Authorisation)) && some ((subject.r2 & Authorisation) - (subject.r1 & Authorisation))} assert SS {all disj r1, r2: Role ss[r1,r2]}</pre>
Shared/Disjoint	<pre>pred sd (disj r1, r2: Role) { r1->r2 in exclusive => ss [r1,r2] && no ((subject.r1 & Authorisation).subject - r1 - r2) && no ((subject.r2 & Authorisation).subject - r1 - r2)} assert SD {all disj r1, r2: Role sd[r1,r2]}</pre>
Disjoint/Shared	<pre>pred ds (disj r1, r2: Role){ r1->r2 in exclusive => ss[r1,r2] && // this is required in analysis to avoid empty models no (subject.r1 & Authorisation) &(subject.r2 & Authorisation)} assert DS {all disj r1, r2: Role ds[r1,r2]}</pre>
Disjoint/Disjoint	<pre>pred dd (disj r1, r2: Role) { r1->r2 in exclusive => ds[r1,r2] &&sd[r1,r2]} assert DD {all disj r1, r2: Role dd[r1,r2]}</pre>

Также было введено дополнительное ограничение, проверяющее, что не существует такой роли, которая потенциально смогла бы обработать документ по всему жизненному циклу документа:

```
assert NoRoleCompletingOneDocument { no r:Role |
all di: DocumentInstance| ((contains.di).Operation).
contains.has_type in (Authorisation&subject.r).targets}
```

Все объекты предметной области, кроме тех, которые непосредственно относятся к понятиям бизнес-процесса (Event и Action) являются подтипами общего объекта Object, что позволяет определять общие для всех принципы поведения.

Выбранный подход к моделированию бизнес-процесса в системе Alloy позволяет описывать сложные структуры с ветвлениями и циклами. Для этого в структуру объектов онтологии (сигнатуры в терминах системы Alloy) вводятся дополнительные поля с ключевыми словами, запускающими определенные события. С использованием этих ключевых слов и традиционных логических связей AND, OR становится возможным полностью определить логику сложного бизнес-процесса.

На третьем уровне моделирования определяются спецификации программных сервисов, существующих в ИТ-инфраструктуре предприятия реализующих определенную бизнес-функцию. Поскольку в предложенной методологии мы стремимся обеспечить взаимосвязь всех уровней моделирования, то описание сервисов на третьем уровне взаимосвязано с понятиями модели второго уровня и зависит от конкретного бизнес-процесса. На третьем уровне мы вводим два основных понятия Service и Method. Взаимосвязь этих понятий реализована в системе Alloy через отношение consists_of сигнатуры Service. Каждый метод связан с определенной бизнес-функцией на уровне 2 через отношение corresponds в сигнатуре Method. В дополнение к описанным отношениям вводится ограничение, определяющее вызов метода во время исполнения бизнес-процесса. На языке системы Alloy это ограничение описывается в форме факта:

```
fact Synchronisation {
all s:Service |all m:Method| m in s.consists_of=>m.
(s.current_method_pre)=(m.corresponds).a_pre
all s:Service |all m: Method | m in s.consists_of =>m.
(s.current_method_post) = (m.corresponds).a_post}
```

Построенная иерархия моделей была использована для анализа информационной безопасности тестового бизнес-процесса, соответствующего реально используемому процессу в крупной компании (рис. 2).

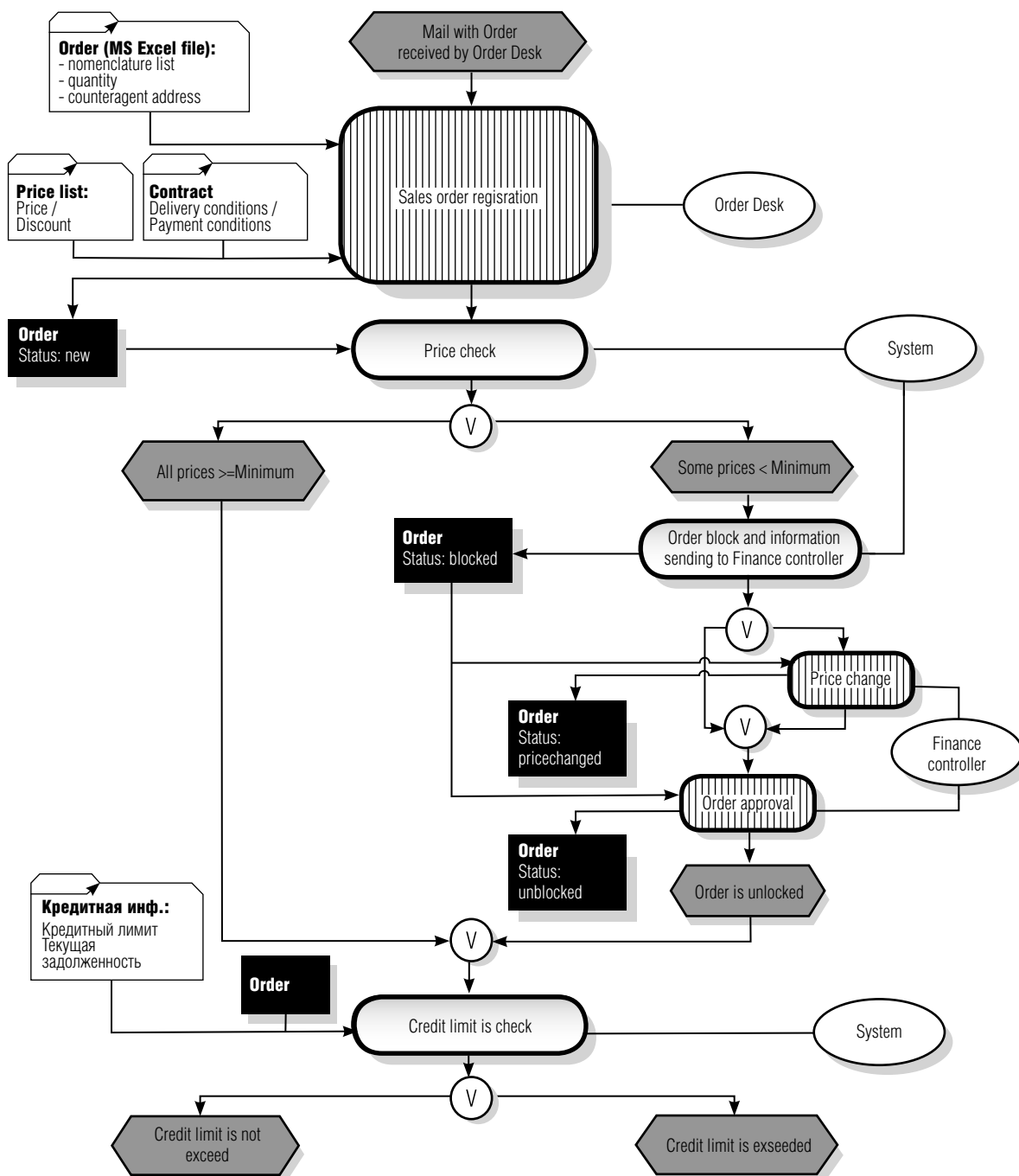


Рис. 2. Бизнес-процесс закупок в нотации EPC.

С целью анализа конкретного бизнес-процесса общая модель была дополнена на втором уровне несколькими понятиями, представляющими конкретные бизнес-функции и роли процесса закупки. С помощью языка системы Alloy были также формально определены переходы между активностями бизнес-процесса и требуемые для использования информационные элементы.

Для проверки непротиворечивости всех возможных сценариев исполнения бизнес-процесса с точки зрения введенных организационных контролей информационной безопасности были использованы средства логического анализа, доступные в системе Alloy.

В результате логического анализа также была получена непротиворечивая последовательность

вызова программных сервисов, соответствующая изучаемому бизнес-процессу. Полученная последовательность может быть использована для автоматической генерации машинно-ориентированной спецификации бизнес-процесса на языке BPEL.

Таким образом, в ходе этой работы было проанализировано перспективное направление построения многоагентных систем на основе сервис-ориентированной архитектуры. Для решения задач автоматизации интеграции и повышения безопасности был предложен метод формальной верификации бизнес-процессов с использованием предметной онтологии и проведен

его анализ на практическом примере.

В качестве основного вывода по проделанной исследовательской работе нужно отметить, что использование специализированного диалекта формальной логики Relation Logic и конкретной реализации Alloy Analyzer для определения корпоративных политик информационной безопасности представляется целесообразным и многообещающим.

Исследование осуществлено в рамках программы фундаментальных исследований ГУ-ВШЭ в 2010 году (проекты ТЗ61.1, ТЗ.29.0). ■

Литература

1. Фейгин Д. «Концепция SOA», http://www.osp.ru/os/2004/06/184447/_p2.html, электронное издательство «Открытые системы», 30.06.2004.
2. Channabasavaiah K., Holley K., Tuggle E.M., Migrating to a service-oriented architecture, IBM, December 2003.
3. Parikh A., Gurajada M., «SOA в реальности», <http://erpnews.ru/doc2610.html>, электронное издание «ERP News», 18.08.2007.
4. «Почему внедрение сервис-ориентированной архитектуры требует много времени», <http://citcity.ru/11420>, электронное издание «CitCity»
5. Бродкин Д., «6 Острых вопросов к SOA», <http://erpnews.ru/doc2584.html>, электронное издание «ERP News», 12.08.2007.
6. Beek M. H., Bucchiarone A., «Formal Methods for Service Composition», Annals of mathematics, computing and teleinformatics, vol. 1, № 5, 2007, PP 1-10.
7. Jackson D., Software Abstractions: Logic, Language, and Analysis, The MIT Press Cambridge, Massachusetts, 2006.
8. Schaad, A., A Framework for Organisational Control Principles, in Department of Computer Science. 2003, University of York,
9. Национальный стандарт РФ, Методы и средства обеспечения безопасности, Часть Концепция и модели менеджмента безопасности информационных и телекоммуникационных технологий, 01.06.2007, http://rfcmd.ru/sphider/docs/InfoSec/GOST-R_ISO_IEC_13335-1-2006.htm.
10. Kuhn, R «Mutual exclusion of roles as a means of implementing separation of duty in role-based access control systems», Proceedings of the second ACM workshop on Role-based access control, 1997, PP 23–30.
11. T. R. Gruber T.R, A translation approach to portable ontologies. Knowledge Acquisition, 5(2):199-220, 1993, <http://tomgruber.org/writing/ontolingua-kaj-1993.htm>.