

УДК 004.896

А.Н. Афанасьев, Н.Н. Войт

ИНТЕЛЛЕКТУАЛЬНАЯ АГЕНТНАЯ СИСТЕМА АНАЛИЗА МОДЕЛЕЙ ПОТОКОВ ПРОЕКТНЫХ РАБОТ¹

Афанасьев Александр Николаевич, доктор технических наук, профессор Ульяновского государственного технического университета. Имеет более 200 статей в области САПР. Область научных интересов: автоматизированные системы обучения, организация вычислительных процессов и структур ЭВМ, проектирование интеллектуальных систем, САПР, управление сложными потоками работ, диаграмматика графических языков. [e-mail: a.afanasev@ulstu.ru].

Войт Николай Николаевич, кандидат технических наук, доцент кафедры «Вычислительная техника» УлГТУ. Имеет более 138 научных статей в области интеллектуальных САПР, Case-, Cals-технологий. Область научных интересов: интеллектуальные системы разработки сложных автоматизированных систем, автоматизированные среды обучения, графические языки и грамматики, распределенные системы. [e-mail: n.voit@ulstu.ru].

Аннотация

Бизнес-процессы и потоки работ проектной организации могут быть представлены диаграммами, содержащими допущенные проектировщиком при разработке ошибки и описанными на языках нотаций Business Process Model Language (BPML), Unified Model Language (UML), Architecture of Integrated Information Systems (ARIS), Integration Definition Methodology (IDEF0 и IDEF3). Чтобы исправить такие ошибки в диаграммах, авторами разработана новая грамматика – RVM-грамматика, которая позволила увеличить качество управления бизнес-процессами, потоками проектных работ на 20% по сравнению с применением конкурирующих сохраняющих, позиционных и реляционных грамматик. Также авторами добавлено к стандартным типам диаграмматических ошибок новых 4 типа. Чтобы учитывать распределенный динамический характер бизнес-процессов, авторами предложена новая интеллектуальная агентная система, основанная на RVM-грамматике и анализирующая модели бизнес-процессов с линейной закономерностью затрат времени, повышающая точность определения синтаксических, семантических ошибок в моделях потоков проектных работ.

Ключевые слова: агентные системы, графические грамматики, бизнес-процессы.

INTELLIGENT AGENT SYSTEM FOR ANALYZING DESIGN WORKFLOW MODELS

Aleksandr Nikolaevich Afanasiev, Doctor of Engineering, Professor at the Department of Computer Engineering of Ulyanovsk State Technical University; an author of more than 200 articles in the field of CAD; interested in the field of automated training systems, computational process and computer structure organization, intelligent system design, CAD, composite workflow control, graphical language diagrammatics. e-mail: a.afanasev@ulstu.ru.

¹ Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 15-07-08268 а.

Nikolai Nikolaevich Voyt, Candidate of Engineering, Associate Professor at the Department of Computer Engineering of Ulyanovsk State Technical University; an author of more than 138 articles in the field of intelligent CAD systems, CASE and CALS technologies; interested in the field of intelligent systems for complex computer-aided system design, automated training environments, graphical languages and the grammar of graphics, distributed systems. e-mail: n.voyt@ulstu.ru.

Abstract

Business process and workflows of project organization represent diagrams as Business Process Model Language (BPML), Unified Model Language (UML), Architecture of Integrated Information Systems (ARIS), Integration Definition Methodology (IDEF0 и IDEF3), which may contain errors made by a designer. In order to check that errors we developed a new grammar (called RVM-grammar) that allows us to increase quality of business process management and workflow systems by 20% oppositely with a reserved graph grammar, a positional grammar, and a relational grammar. Thus, we propose an intelligent agent system based on RVM-grammar to analyze business-process models for manufacturing. The system has linear time in order to facilitate the automatically identification of a variety of syntax and semantical errors in workflow models, which enables the analysis of four additional types of errors. In order to consider the distributed dynamic character of business processes, the authors proposed a new intelligent agent system based on the RVM-grammar and analyzing models of business processes with line timetable conformity that extends evaluation precision of syntax and semantic errors in the models of design work flows.

Key words: agent systems, the grammar of graphics, business processes.

ВВЕДЕНИЕ

Начиная с 1980 года, сектор производства и трансконтинентальные компании активно интересуются прикладными областями научных исследований распределенных систем, в это же время ученые вводят новый термин – распределенный искусственный интеллект (РИИ). Появились первые книги Nuhns [1], работа Parunak [2], Fischer и др. [3], свидетельства коммерческого интереса, например, компаний Whitestein Technologies и Magenta Technologies, описывающих производство как главную прикладную область науки. Приведенные ссылки в работе принадлежат большому периоду исследований, из которых видно, что производство интересует планирование проектной деятельности и логистика поставок продукции. Следовательно, такие аспекты, как виртуальные предприятия, менеджмент бизнес-процессов, системы обслуживания потоков работ, а также информационные системы и агенты играют важную роль в управлении.

Whitestein Technologies и Magenta Technologies отмечают, что первое поколение статических систем управления жизненным циклом изделия и потоками проектных работ больше не удовлетворяют потребности многих компаний, подход и автоматизированные средства первого поколения стандартизации процессов потоков проектных работ исчерпали свой ресурс, как следствие, появляются плохо формализованные (неподходящие) процессы, стимулирующие рост затрат на их развитие и улучшение [4].

Промышленные агентно-ориентированные системы имеют задачи распределенного (активного динамического взаимодействия между всеми участниками проектного процесса) управления бизнес-процессами и потоками работ проектной организации. Авторы реализовали агентную архитектуру, названную в работе [5] архитектурой распределенного управления (APУ), заменив в ней компоненты на следующие агенты: агент RVM-грамматики, агент моде-

ли диаграммы, агент планирования, агент исполнительного контроля и агент диагностики. Агент RVM-грамматики создает грамматику, чтобы описать диаграмму. Агент модели диаграммы содержит XML-описание диаграммы. Агент планирования является главным агентом, управляющим интеллектуальной агентной системой. Агент исполнительного контроля выполняет действия, полученные от агента планирования. Агент диагностики проверяет диаграмму на ошибки. Авторы использовали стандарт Foundation for Intelligent Physical Agents (FIPA), который содержит протоколы Объявления, Опроса, Соглашения Задач/Действий, хорошо описанные в работе [6], с целью разработки стандартной имплементации принципов открытости, гетерогенности и итероперабельности агентов.

Существует современная теория графических языков для представления диаграмм, содержащая синтаксические модели в пространственном и логическом видах, включающая атрибуты графических объектов (например, прямоугольник или круг) и типы связи.

Пространственная модель имеет относительные или абсолютные координаты графических объектов. Применение пространственной модели является сложным, чтобы контролировать, анализировать структуру или топологию (синтаксис) и атрибуты диаграмм.

Как правило, логическая модель используется, чтобы описать синтаксис диаграмм на основе графической грамматики.

Изучение зарубежной литературы выявило ряд важных ученых в области графических грамматик. John L. Pfaltz и Azreil Rosenfeld предложили веб-грамматику [7]. Zhang и Costagliola [8] разработали позиционную графическую грамматику, относящуюся к контекстно-свободной грамматике. Wittenberg и Weitzman [9] разработали реляционную графическую грамматику. Zhang и Orgun [10, 11] описали сохраняющую графическую грамматику в своих работах.

Однако упомянутые графические грамматики имеют следующие недостатки.

Большинство инструментов, поддерживающих системы автоматизации проектирования (IBM Rational Unified Process [12], ARIS [13]), имеют прямой метод анализа, который требует много итераций, чтобы проверить диаграмму на ошибки. Следовательно, такие инструменты могут не зафиксировать ошибки, названные разрывом контекста, связанные с использованием логических связей, таких как 'AND', 'OR', 'XOR'. Контекстно-зависимые ошибки, которые не будут найдены с помощью упомянутых инструментов, могут быть в тексте (нотациях) любой диаграммы. Они становятся «дорогими» ошибками в проектировании.

Бизнес-процессы и потоки проектных работ могут быть смоделированы в IBM Rational Software Architecture [12], MS Visio [14], Dia [15], ARIS с помощью BPMN [4], UML [12], представлены диаграммами и включать ошибки, допущенные проектировщиком. Таким образом, чтобы распределенно проверить диаграммы на ошибки, авторы разработали интеллектуальную агентную систему, содержащую упомянутые выше агенты.

В разделе «Грамматические модели» характеризуются грамматики и дается определение новой RVM-грамматике [16]. Для примеров авторы взяли UML-диаграммы, чтобы представить бизнес-процессы, хотя RVM-грамматика может проверять и исправлять ошибки в BPMN, IDEF0 [17], IDEF3, сетях Петри. Раздел «Проверка диаграмм» содержит авторский метод анализа диаграмм на ошибки. Интеллектуальная агентная система предложена авторами в разделе «Интеллектуальная агентная система». В «Результатах» авторы оценили возможный эффект от применения интеллектуальной агентной системы в проектной организации [18].

МОДЕЛИ ГРАММАТИК

В современной теории графических и визуальных языков для представления диаграмматических моделей, в том числе и бизнес-процессов, рассматриваются две основные синтаксические модели, в основу которых положены атрибуты графических объектов и типы их связывания: пространственная и логическая. Пространственная модель содержит относительные или абсолютные координаты графических объектов. Использование такой модели для контроля и анализа структурных, т. е. топологических (синтаксических), особенностей диаграмм крайне затруднительно. Поэтому для описания синтаксиса графических языков диаграмматических моделей используется логическая модель. Однако существующие логические модели грамматики [7–11] имеют следующие недостатки:

1. Позиционные грамматики, развиваясь на базе плекс-структур, не предполагают использование областей соединения и не могут применяться для графических языков, объекты которых имеют динамически изменяемое количество входов/выходов.

2. Авторы реляционных грамматики оговариваются о несовершенстве механизма нейтрализации в плане неполноты формируемого списка ошибок.

3. Отсутствует семантический контроль текстовой атрибутики диаграмматических моделей графических языков.

4. Общими недостатками вышеописанных грамматик являются: увеличение числа продукций при построении грамматики для неструктурированных графических языков (при неизменном количестве примитивов графического языка для описания всех вариантов неструктурированности происходит значительное увеличение количества продукций), сложность построения грамматики, большие временные затраты анализа (анализаторы, построенные на базе рассмотренных грамматик, обладают полиномиальным или экспоненциальным временем анализа диаграмм графических языков).

Российскими учеными исследования в области диаграмматики и обработки визуальных языков практически не ведутся. Известна работа проф. Жоголева Е.А. [19], которая носит обзорный характер. В ней подчеркивается важность направления, связанного с компьютерной графикой. Однако проблематика, связанная с анализом и контролем диаграмматических моделей, не рассматривается. В работе Степанова П.А. [20] подобные вопросы также не затрагиваются.

МНОГОУРОВНЕВАЯ ГРАММАТИКА

Базовый математический аппарат анализа и контроля UML-диаграмм основан на применении семейства RV-грамматик [21–23]. При коллективном проектировании диаграмматические модели имеют сложную иерархическую структуру (комплексные диаграммы), количество термов в них увеличивается многократно. Классическая графическая RV-грамматика становится громоздкой, ее разработка усложняется, не обеспечивается контроль взаимосвязанных вершин и диаграмм комплексных моделей. Для устранения этих недостатков предлагается аппарат многоуровневых грамматик.

Рассмотрим многоуровневую систему RV-грамматик, представленную в виде кортежа из четырех элементов:

$$RVM = \langle n, \Sigma^n, RV^n, r_0 \rangle,$$

где n – индекс грамматики;

Σ^n – алфавит n -й грамматики;

RV^n – множество продукций n -й грамматики;

r_0 – аксиома грамматики верхнего уровня.

Грамматика RV^i содержит в качестве одного из состояний грамматику RV^j . Грамматика RV^j также может быть составной.

RV^n – грамматикой языка $L(G)$ является упорядоченная пятерка непустых множеств $G = (V, \Sigma, \tilde{\Sigma}, R, r_0)$, где $V = \{v_l, l = \overline{1, L}\}$ – алфавит операций над внутренней памятью;

$\Sigma = \{a_t, t = \overline{1, T}\}$ – терминальный алфавит графического языка (множество примитивов графического языка);

$\tilde{\Sigma} = \{\tilde{a}_t, t = \overline{1, \tilde{T}}\}$ – квазитерминальный алфавит, являющийся расширением терминального алфавита.

Алфавит включает:

- квазитермы графических объектов, не являющихся продолжателями анализа,
- квазитермы графических объектов, имеющих более одного входа,
- квазитермы связей-меток с определенными для них семантическими различиями,
- квазитерм для завершения анализа;

$R = \{r_i, i = \overline{1, I}\}$ – схема грамматики G (множество имен комплексов продукций, причем каждый комплекс r_i состоит из подмножества P_{ij} продукций $r_i = \{P_{ij}, j = \overline{1, J}\}$);

$r_0 \in R$ – аксиома RV-грамматики (имя начального комплекса продукций), $r_k \in R$ – заключительный комплекс продукций.

Продукция $P_{ij} \in r_i$ имеет вид:

$$\tilde{a}_i \xrightarrow{W_\gamma(\gamma_1, \dots, \gamma_n)} r_m,$$

где $W_\gamma(\gamma_1, \dots, \gamma_n)$ – n -арное отношение, определяющее вид операции над внутренней памятью в зависимости от $\gamma \in \{0, 1, 2, 3\}$;

$r_m \in R$ – имя комплекса продукции-преемника.

В структуру внутренней памяти входят стеки для обработки графических объектов, имеющих более одного выхода, и эластичные ленты для обработки графических объектов, имеющих более одного входа.

Система RVM получает с входной ленты символы терминального алфавита и передает их на соответствующий уровень. Элементы, которые переводят грамматику на другой уровень, назовем «сабтермами». Тогда описание RV^n -грамматики примет вид:

$$G = (V, \Sigma, \tilde{\Sigma}, \bar{\Sigma}, R, r_0),$$

где $\bar{\Sigma}$ – множество сабтермов, т. е. элементов грамматики, переводящих автомат на следующий более низкий уровень.

Продукция, содержащая сабтерм, имеет вид:

$$\bar{a}_i \xrightarrow[r_0^{n+1}]{{W_\gamma(\gamma_1, \dots, \gamma_n)}} r_m^n,$$

где r_0^{n+1} – комплекс-преемник – начальный комплекс грамматики следующего уровня,

r_m^n – комплекс-преемник, к которому производится переход при достижении r_k^{n+1} .

В качестве примера в таблице 1 приведена табличная форма грамматики диаграммы активности UML.

ПРОВЕРКА ДИАГРАММ

При коллективном проектировании важным является контроль онтологической согласованности комплекса про-

ектируемых диаграмм. Ошибки такого типа являются семантическими. В этом плане для анализа семантической корректности предложено использовать многоуровневую грамматику. Верхним уровнем многоуровневой грамматики является грамматика диаграмм вариантов использования, так как разработка автоматизированных систем, согласно методологии IBM Rational Unified Process, начинается именно с этой диаграммы. В процессе анализа накапливается семантическая информация о предметной области в виде графа связей между семантическими понятиями (текстовой информацией). Каждая новая диаграмма анализируется на условия непротиворечивого расширения понятий предметной области.

При построении первых диаграмм проверяется только семантическая непротиворечивость внутри диаграммы: возможность использования понятий в семантической паре. При добавлении новых диаграмм проверяется согласованность диаграммы обособленно и на согласованность комплексной модели проектируемой АС.

Для проверки комплексной модели необходимо построить граф семантических связей между элементами автоматизированной системы. Для решения данной задачи выбран адаптированный метод лексико-синтаксических шаблонов.

Предложенный метод позволяет диагностировать следующие семантические ошибки: большая синонимия, антонимия объектов, конверсивность отношений. Один из классов ошибок не контролируется предложенным методом – несовместимость объектов.

Разработанные методы анализа и контроля позволяют диагностировать ошибки, разделенные контекстом, а

Таблица 1

Табличная форма грамматики диаграмм активности UML

№	Комплекс	Квазитерм	Комплекс-преемник	RV-отношение
1	r_0	label	r_3	\emptyset
2	r_1	label	r_3	\emptyset
3	r_2	labelP	r_3	$W_2(b^{1m})$
4		labelW	r_3	$W_2(b^{2m})$
5		labelR	r_3	$W_2(b^{3m})$
6		labelL	r_3	$W_2(b^{4m}) / W_3(m^{(2)} = k^{(3)})$
7	r_3	A	r_1	\emptyset
8		P	r_1	$W_1(t^{1m})$
9		W	r_2	$W_1(1^{(1)}, t^{2m}) / W_2(e^{t(1)})$
10		W	r_2	$W_1(2^{(1)}) / W_2(1^{(1)})$
11		R	r_1	$W_1(t^{3m(k-1)}) / W_3(k > 1)$
12		L	r_2	$W_1(1^{(2)}, k^{(3)}, t^{4m}) / W_2(e^{t(2)})$
13		L	r_2	$W_1(inc(m^{(2)})), / W_3(m^{(2)} < k^{(3)})$
14		AK	r_k	\emptyset

также семантические ошибки, которые не определяются в большинстве современных редакторов.

В процессе исследования были выявлены следующие классы ошибок, встречающиеся в диаграммах UML (табл. 2).

В таблице использованы следующие сокращения: ДИ – диаграмма использования, ДА – диаграмма активности, ДП – диаграмма последовательности, ДК – диаграмма классов, ДР – диаграмма развертывания.

Исследование современных систем создания диаграмматических нотаций UML показало, что они позволяют обнаруживать первые 16 из перечисленных типов ошибок. Авторский аппарат RVM-грамматик позволяет обнаруживать 4 дополнительных типа ошибок: множественная связь, кольцевые связи, синхронный вызов до получения ответа, ошибка удаленного контекста.

Ошибкой удаленного контекста называется ошибка, возникающая в парных элементах, например, условное ветвление и слияние условных ветвей, и характеризующаяся возможным присутствием неограниченного количества блоков и связей между ними. Примером такой ошибки может служить условное ветвление на диаграмме активности, которое предполагает исполнение только одной из возможных ветвей, и парное ему слияние параллельных ветвей, которое передает управление дальше только при достижении всех входных связей. Такое сочетание элементов никогда не позволит достигнуть конца диаграммы.

Разработаны программные плагины анализа и контроля [24]: анализатор диаграмматических моделей потоков бизнес-процессов вопросно-ответной системы моделирования сложных автоматизированных систем, синтаксически-ориентированный анализатор UML-диаграмм для MS Visio (см. рис. 1), сетевая система анализа и контроля диаграмматических моделей, предлагающая полный набор функциональности для анализа и контроля синтаксических и семантических ошибок.

Инструментарий последней системы позволяет выполнять следующие основные функции для пользователя:

- создание диаграмм моделей бизнес-процессов с использованием различных графических языков;
- добавление новых нотаций в графические языки;
- анализ построенных диаграмм, с использованием аппарата RVM-грамматик, по предварительно загруженным в систему описаниям языка и правилам анализа;
- добавление новых алгоритмов анализа с помощью плагинов;

Классы ошибок UML-диаграмм

№	Тип ошибки	ДИ	ДА	ДП	ДК	ДР
1	Отсутствие связи	да	да	да	да	да
2	Ошибка передачи управления					
3	Ошибка кратности входов		да			да
4	Ошибка кратности выходов		да			
5	Недопустимая связь	да	да	да	да	да
6	Ошибка связи	да	да			да
7	Ошибка уровня доступа				да	
8	Ошибка передачи сообщения		да	да		
9	Ошибка делегирования управления				да	
10	Количественная ошибка элементов диаграммы		да			да
11	Исключающие связи неверного типа				да	
12	Вызов, направленный в линию жизни			да		
13	Несвязанная связь	да	да	да	да	да
14	Нарушение кратности зависимостей	да	да			
15	Взаимоисключающие связи	да				
16	Множественная связь	да				
17	Бесконечный цикл	да				
18	Кольцевые связи	да				
19	Синхронный вызов до получения ответа			да		
20	Ошибка удаленного контекста		да			

- добавление синтаксических и семантических правил графических языков для использования в RVM-анализаторе;
- создание взаимосвязи между построенными проектировщиками диаграммами;
- одновременный доступ нескольких проектировщиков к базе данных построенных диаграмм.

ИНТЕЛЛЕКТУАЛЬНАЯ АГЕНТНАЯ СИСТЕМА

В предыдущем разделе авторы описали математическое обеспечение RVM-грамматики, являющейся ядром интеллектуальной агентной системы для исправления ошибок в диаграммах. В данном разделе авторы рассмотрят работы по развертыванию промышленных мультиагентных систем (МАС) [5], чтобы разработать собственную интеллектуальную агентную систему. МАС применяется обычно в следующих секторах: промышленность, логистика, воздушное движение, исследование космоса, обучение [21], автотранспорт, система поставок и организация вычислительной сети. Авторы сфокусировали работу на промышленном секторе, содержащем модели бизнес-процессов и потоки проектных работ, а также рассмотрели концепции построения МАС: убеждение-желания-намерение (Belief-Desire-Intention – BDI), переговоры (negotiation), моде-

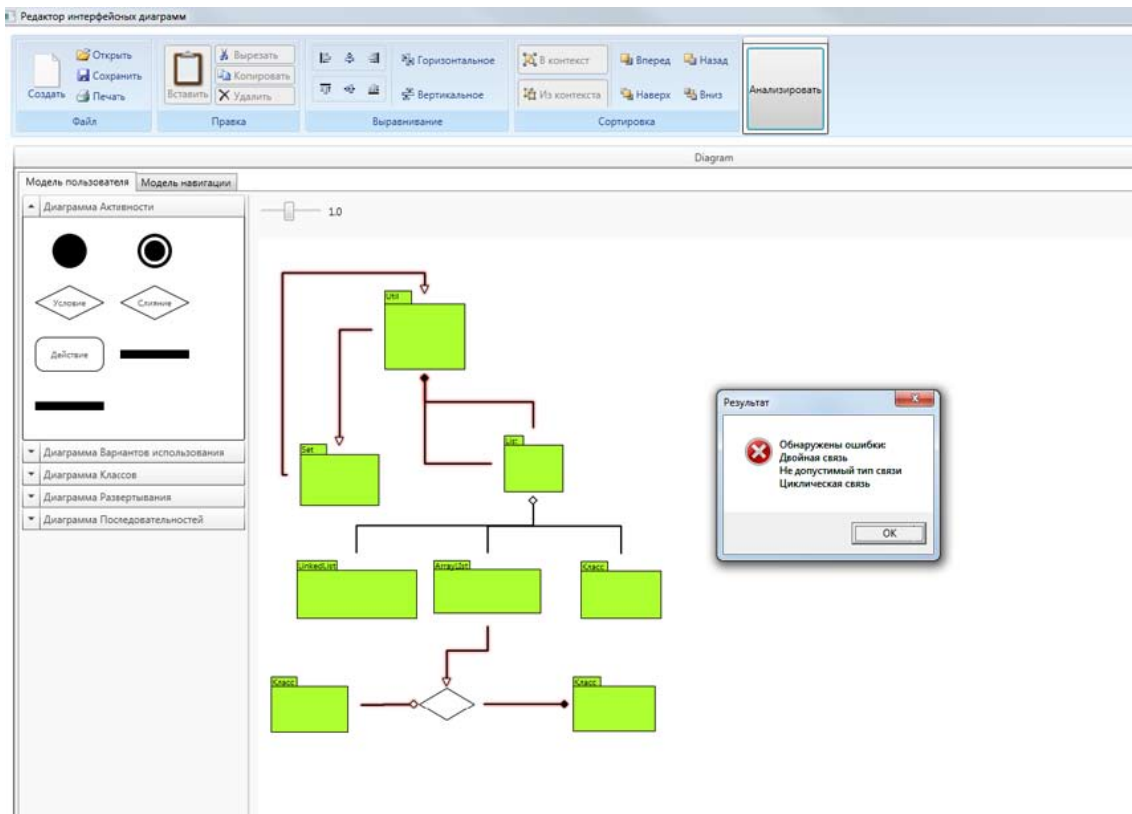


Рис. 1. Плагин интеллектуальной агентной системы для MS Visio

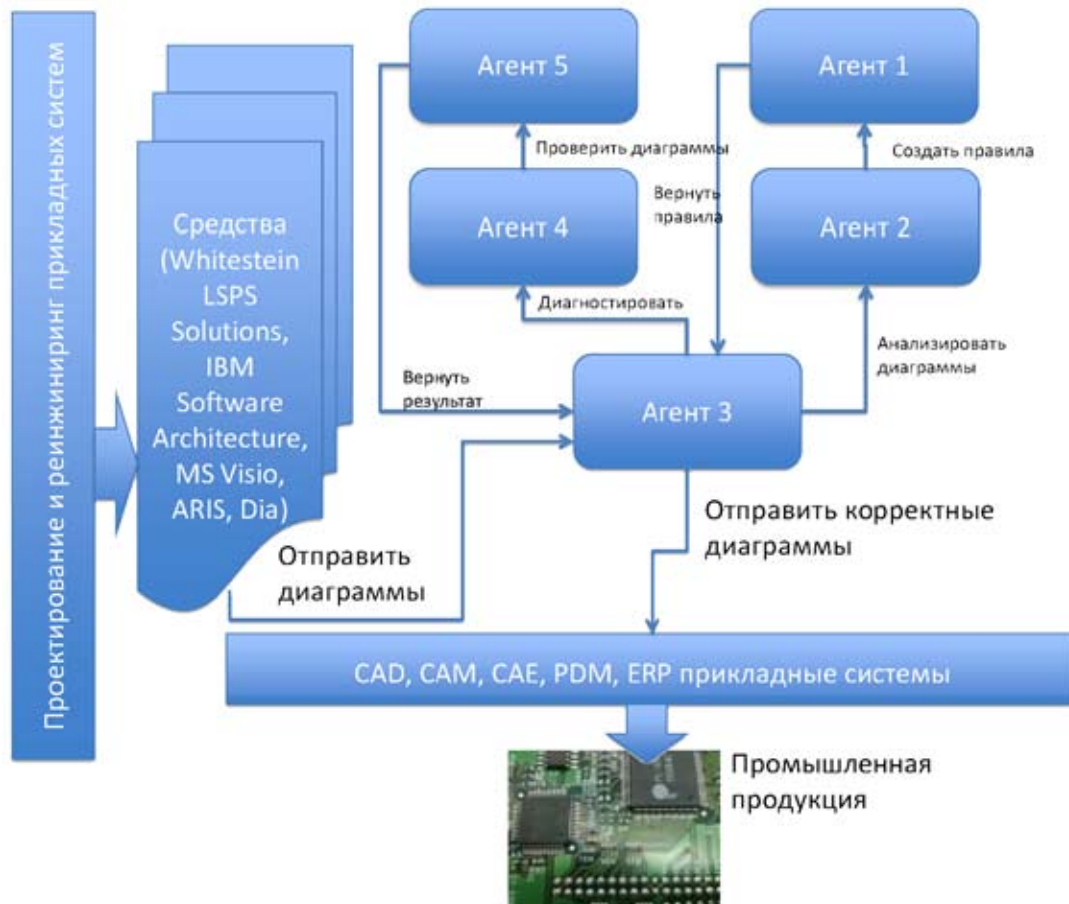


Рис. 2. Архитектура распределенного управления (APU) интеллектуальной агентной системы

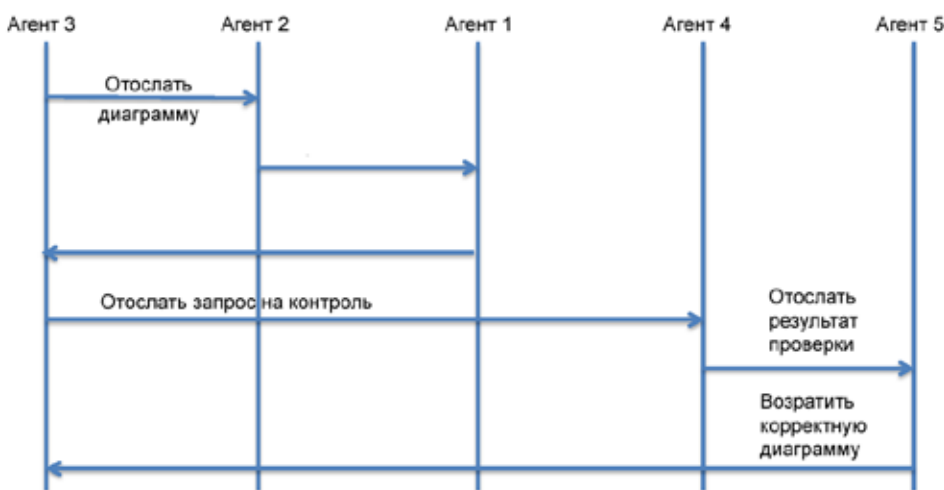


Рис. 3. Диаграмма последовательности взаимодействия агентов

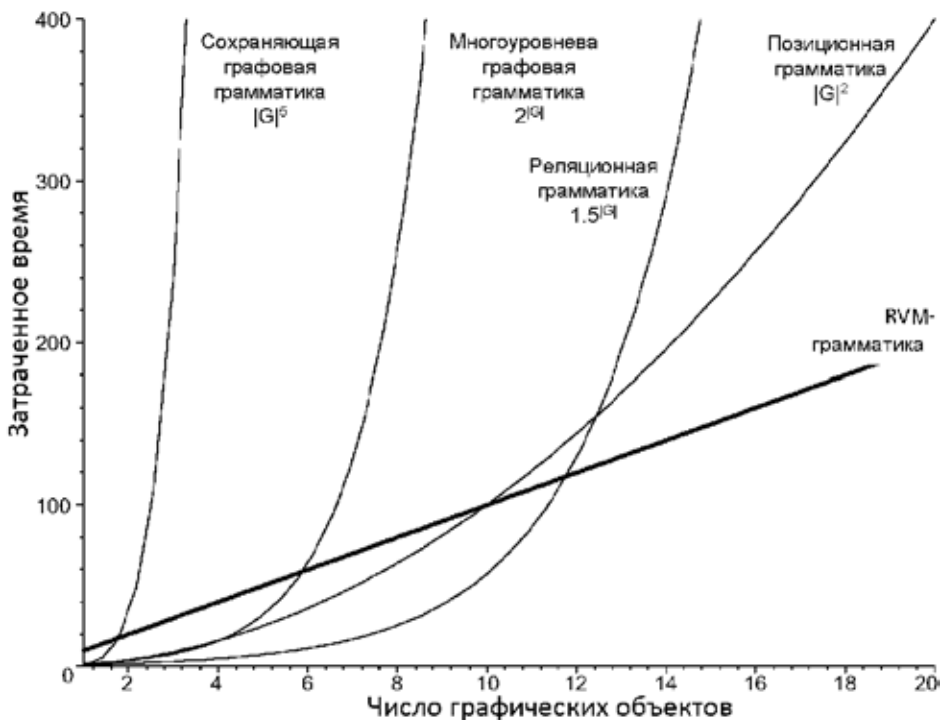


Рис. 4. Сравнительная оценка эффективности применения RVM-грамматики для устранения ошибок в диаграммах

лирование (simulation), совместимость (interoperability), координация (coordination), распределен контроль и диагностика (distributed control and diagnostics) [3]. В результате авторы выбрали концепцию распределенного контроля и диагностики. Производственная интеллектуальная агентная система содержит следующую функциональность: контроль (проверка) и диагностика (исправление) [3]. В работе [5] представлена АРУ-архитектура, мы создали АРУ-архитектуру для контроля и диагностики диаграмм, представленную на рисунке 2.

Каждый блок на рисунке 2 выполняет следующие функции.

Проектирование и реинжиниринг прикладных систем: разработать модель бизнес-процесса и потока проектных работ, чтобы описать процесс изготовления промышленного оборудования.

MS Visio, IBM Rational Software Architecture, Dia, ARIS: помочь представить модель бизнес-процесса и потока проектных работ с помощью диаграмм.

Агент модели диаграммы (1): содержит описание диаграммы в виде таблицы правил.

Агент RVM-грамматики (2): создает грамматику, чтобы описать диаграмму бизнес-процесса.

Агент планирования (3): менеджер агентов.

Агент исполнительного контроля (4): исполняет команды, полученные от агента планирования.

Агент диагностики (5): проверяет и исправляет ошибки в диаграмме.

Прикладные системы, такие как CAD/CAE/CAM/PDM/ERP: используют диаграммы при планировании и разработке производственного оборудования.

Любой из агентов может проактивно начать планировать деятельность. Процесс планирования имеет три фазы: Создание, Требование (Запрос) и Исполнение. Во время Создания агент распознает настоящую ситуацию и создает шаблон плана, используя протоколы Объявления и Опроса, с

остальными агентами, чтобы сформировать общий план. В фазе Требования агент запрашивает ресурсы для выполнения задачи. В последней фазе Исполнения план выполняется всеми участниками-агентами, которые используют протоколы Соглашения Задач/Действий.

Протокол Объявления обслуживает следующие взаимодействия агентов: агент с задачей рассылает объявление о задаче в виде спецификации в закодированном виде, описывающем ее и свои ограничения.

Протокол Опроса обслуживает следующие взаимодействия агентов: агенты, получившие объявление о задаче, решают выполнить задачу.

Протокол Соглашения Задач/Действий обслуживает следующие взаимодействия агентов: агенты исполняют план.

Агенты взаимодействуют друг с другом с помощью языка описания работ (Job Description Language – JDL) [14] как содержательного языка в FIPA-ACL. Интеллектуальная агентная система поддерживает как Директивного Помощника (Directory Facilitator – DF), обеспечивающего нахождение агентов по функционалу, так и Сервисы Управления Агентом (Agent Management Services – AMS), обеспечивающие нахождение агентов по адресам. Коммуникация агентов между собой представлена на рисунке 3.

Авторы применили WADE-платформу [23], чтобы разработать интеллектуальную агентную систему, плагины которой могут быть интегрированы в MS Visio, Dia.

РЕЗУЛЬТАТЫ

В статье рассмотрена промышленная отрасль, в которой есть интерес к прикладным средствам контроля и диагностики бизнес-процессов и потоков проектных работ в виде диаграмм. Авторы сформулировали 20 типов ошибок, возникающих в BPML, UML, IDEF0, IDEF3, сети Петри. Современные средства могут проверять только 16 типов ошибок, авторская разработка RVM-грамматики фиксирует и исправляет все 20 типов ошибок, имеет линейную зависимость затрат времени на анализ и исправление по сравнению с остальными видами грамматик, имеющих экспоненциальный и полиномиальный виды зависимостей затрат времени на анализ. На рисунке 4 представлен сравнительный график эффективности.

В статье было рассмотрено много работ и средств, авторы выявили, что среди них нет инструмента для проверки и исправления семантических ошибок в диаграммах потоков проектных работ и бизнес-процессов (особенно распределенных). Исследована и разработана интеллектуальная агентная система, поддерживающая APU, FIPA-протоколы и использующая WADE-платформу, являющуюся распределенной, что помогает контролировать и диагностировать диаграммы потоков проектных работ в распределенных промышленных корпорациях (транскомпаниях). Предложенные авторами модели, методы и средства позволяют устранить много ошибок на ранних этапах проектной деятельности, содержащихся в семантике проектной документации, полученной на разработку промышленной продукции.

ЗАКЛЮЧЕНИЕ

Авторы исследовали задачу управления бизнес-процессами и потоками проектных работ производственных организаций с целью разработки новой грамматики и интеллектуальной агентной системы, обеспечивающих повышение качества диаграмм бизнес-процессов и потоков проектных работ, сокращение издержек проектной организации и сроков разработки [24]. Бизнес-процессы могут быть представлены в виде диаграмм. При устранении ошибок диаграмм качество диаграмм увеличивается, причем любая диаграмма описывается грамматикой, которая

может быть применена для ее проверки на ошибки. Таким образом, была разработана RVM-грамматика для анализа и контроля диаграмм, обеспечивающая проверку и исправление 20 типов ошибок, включая 4 новых авторских типа ошибки. Следовательно, RVM-грамматика фиксирует и исправляет на 20% больше ошибок, чем конкурирующие грамматики: сохраняющая, позиционная и реляционная.

Многие проектные организации желают иметь распределенные средства контроля и диагностики потока проектных работ в виде диаграмм. RVM-грамматика может быть реализована с помощью MAC, чтобы создать распределенную систему для проектных организаций, способную эффективно проверять и исправлять ошибки, улучшая качество бизнес-процессов и потоков проектных работ в промышленности. Поэтому авторами была разработана для проектной организации интеллектуальная агентная система, основанная на RVM-грамматике, содержащая APU, поддерживающая FIPA-протоколы, 5 новых агентов, коммуникационные протоколы и WADE-платформу.

СПИСОК ЛИТЕРАТУРЫ

1. Huhns M.N. Distributed Artificial Intelligence. Pitman, Morgan Kaufmann, San Mateo, CA, 1987.
2. Parunak H.V.D. Manufacturing experience with the contract net. In M.N. Huhns, editor // Distributed Artificial Intelligence. – Morgan Kaufmann Publishers. – San Mateo, CA. – 1987. – pp. 285–310.
3. Jörg P. Müller and Klaus Fischer. Application Impact of Multiagent Systems and Technologies: A Survey, 2014. – URL: <http://www.researchgate.net>.
4. A global Swiss company offering advanced intelligent application software for multiple business sectors. – URL: <http://whitestein.com/>.
5. Michal Pechoucek and Vladimir Marik. Review of Industrial Deployment of Multi-Agent Systems, 2008. – URL: <http://citeseerx.ist.psu.edu/>.
6. Multi-Agent Systems – Modeling, Control, Programming, Simulations and Applications. / Edited by Faisal Alkhateeb, Eslam Al Maghayreh and Iyad Abu Doush, 2011.
7. Fu K. Structural methods of pattern recognition. – Moscow: Mir, 1977. – P. 319.
8. Costagliola G., Lucia A.D., Orece S., Tortora G. A parsing methodology for the implementation of visual systems. – URL: <http://www.dmi.unisa.it/people/costagliola/www/home/papers/method.ps.gz>.
9. Wittenburg K., Weitzman L. Relational grammars: Theory and practice in a visual language interface for process modeling (1996). – URL: <http://citeseer.ist.psu.edu/wittenburg96relational.html>.
10. Zhang D.Q., Zhang K. Reserved graph grammar: A specification tool for diagrammatic VPLs // Visual Languages. Proceedings. 1997 IEEE Symposium on. – IEEE, 1997. – pp. 284–291.
11. Zhang K.B., Zhang K., Orgun M.A. Using Graph Grammar to Implement Global Layout for a Visual Programming Language Generation System (2002). – URL: http://www.researchgate/publication/221565032_Using_Graph_

Grammer_to_Implement_Global_Layout_for_a_Visual_Programming_Language_Generation_System.

12. Aho A.V. and S. C. Johnson. LR Parsing. (1974). – URL: <http://courses.engr.illinois.edu/cs421/sp2012/project/p99-aho.pdf>.

13. Жоголев Е.А. Графические редакторы и графические грамматики // Программирование. – 2001. – № 3. – С. 30–42.

14. Степанов П.А., Охтилев М.Ю. Вычислительная модель визуального языка // Изв. вузов. Приборостроение. – 2006. – № 11. – С. 28–32.

15. Шаров О.Г., Афанасьев А.Н. Синтаксически ориентированная реализация графических языков на основе автоматных графических грамматик // Программирование. – 2005. – № 6. – С. 56–66.

16. Шаров О.Г., Афанасьев А.Н. Нейтрализация синтаксических ошибок в графических языках // Программирование. – 2008. – № 1. – С. 61–66.

17. Шаров О.Г., Афанасьев А.Н. Методы и средства трансляции графических диаграмм // Программирование. – 2011. – № 3. – С. 65–76.

18. Афанасьев А.Н., Гайнуллин Р.Ф., Шаров О.Г. Программная система анализа диаграммных языков // Программные продукты и системы. – 2012. – № 3. – С. 138–141.

19. Michal Pechoucek and Vladimir Marik. Industrial deployment of multi-agent technologies: review and selected case studies. // *Autonomous Agents and Multi-Agent Systems*. – 2008. – Vol. 17, – № 3. – pp. 397–431.

20. S. Seiler et al. Current Trends in Remote and Virtual Lab Engineering. Where are we in 2013? // *International Journal of Online Engineering*. – 2013. – Vol. 9, – № 6. – pp. 12–16.

21. Войт Н.Н., Афанасьев А.Н. Разработка алгоритмического, методического и информационного обеспечения АОС для САПР КОМПАС-3D // *Вестник УлГТУ*. – 2005. – № 3. – С. 50–56.

22. Tich P., Lechta P., Maturana F., and S. Balasubramanian (2002): *Industrial MAS for Planning and Control*. // LNAI. Heidelberg: Springer Verlag. – № 2322, pp. 280–295.

23. A software platform based on JADE that provides support for the execution of tasks defined according to the workflow metaphor. – URL: <http://jade.tilab.com/wadeproject/>.

24. Belecheanu R.A., Munroe S., Luck M., Miller T., McBurney P., and M. Pechoucek (2006): *Commercial Applications of Agents: Lessons, Experiences and Challenges*. // Proc. of AAMAS-06– Industry Track. ACM Press. – pp. 1549–1555.

REFERENCES

1. Huhns M.N. *Distributed Artificial Intelligence*. Pitman, Morgan Kaufmann, San Mateo, CA, 1987.

2. Parunak H.V.D. Manufacturing Experience with the Contract Net. In M.N. Huhns, editor. *Distributed Artificial Intelligence*. – San Mateo, CA, Morgan Kaufmann Publishers, 1987, pp. 285–310.

3. Jörg P. Müller and Klaus Fischer. *Application Impact of Multiagent Systems and Technologies: A Survey*. 2014. Available at: <http://www.researchgate.net>.

4. *A global Swiss Company Offering Advanced Intelligent Application Software for Multiple Business Sectors*. Available at: <http://whitestein.com/>.

5. Michal Pechoucek and Vladimir Marik. *Review of Industrial Deployment of Multi-Agent Systems*. 2008. Available at: <http://citeseerx.ist.psu.edu/>.

6. *Multi-Agent Systems – Modeling, Control, Programming, Simulations and Applications*. Edited by Faisal Alkhateeb, Eslam Al Maghayreh and Iyad Abu Doush. 2011.

7. Fu K. *Structural Methods of Pattern Recognition*. Moscow, Mir Publ., 1977. 319 p.

8. Costagliola G., Lucia A.D., Orece S., Tortora G. *A Parsing Methodology for the Implementation of Visual Systems*. Available at: <http://www.dmi.unisa.it/people/costagliola/www/home/papers/method.ps.gz>.

9. Wittenburg K., Weitzman L. *Relational Grammars: Theory and Practice in a Visual Language Interface for Process Modeling (1996)*. Available at: <http://citeseer.ist.psu.edu/wittenburg96relational.html>.

10. Zhang D.Q., Zhang K. Reserved Graph Grammar: A specification Tool for Diagrammatic VPLs. *Visual Languages. Proceedings. 1997 IEEE Symposium on. IEEE, 1997*, pp. 284–291.

11. Zhang K.B., Zhang K., Orgun M.A. *Using Graph Grammar to Implement Global Layout for a Visual Programming Language Generation System* (2002). Available at: http://www.researchgate/publication/221565032_Using_Graph_Grammer_to_Implement_Global_Layout_for_a_Visual_Programming_Language_Generation_System.

12. A.V. Aho and S.C. Johnson. *LR Parsing*. 1974. Available at: <http://courses.engr.illinois.edu/cs421/sp2012/project/p99-aho.pdf>.

13. Zhogolev E.A. Graficheskie redaktory i graficheskie grammatiki [Graphics Editors and Grammar of Graphics]. *Programmirovaniye* [Programming], 2001, no. 3, pp. 30–42.

14. Stepanov P.A., Okhtilev M.Yu. Vychislitelnaia model vizualnogo iazyka [Computational Model of Visual Language]. *Izv. vuzov. Priborostroenie* [Proc. of Higher Educational Institutions. Instrument Engineering], 2006, no. 11, pp. 28–32.

15. Sharov O.G., Afanasev A.N. Sintaksicheski-orientirovannaia realizatsiia graficheskikh iazykov na osnove avtomatnykh graficheskikh grammatik [Syntaxes-Oriented Implementation of Graphic Languages on the basis of Automation Grammar of Graphics]. *Programmirovaniye* [Programming], 2005, no. 6, pp. 56–66.

16. Sharov O.G., Afanasev A.N. Neitralizatsiia sintaksicheskikh oshibok v graficheskikh iazykakh [Syntax Error Recovery in Graphic Languages]. *Programmirovaniye* [Programming], 2008, no. 1, pp. 61–66.

17. Sharov O.G., Afanasev A.N. Metody i sredstva translitsii graficheskikh diagramm [Methods and Tools for Graphic Charts Translation]. *Programmirovaniye* [Programming], 2011, no. 3, pp. 65–76.

18. Afanasev A.N., Gainullin R.F., Sharov O.G. Programmaia sistema analiza diagrammykh iazykov [Program System for Visual Language Analysis]. *Programmnye produkty i sistemy* [Software and Systems], 2012, no.3, pp. 138–141.

19. Michal Pechoucek and Vladimir Marik. Industrial Deployment of Multi-Agent Technologies: Review and Selected Case Studies. *Autonomous Agents and Multi-Agent Systems*, 2008, vol. 17, no. 3, pp. 397–431.

20. S. Seiler et al. Current Trends in Remote and Virtual Lab Engineering. Where are we in 2013? *International Journal of Online Engineering*, 2013, vol. 9, no. 6, pp. 12– 16.

21. Voyt N.N., Afanasev A.N. Razrabotka Algoritmicheskogo, Metodicheskogo i informatsionnogo obespecheniia AOS dlia SAPR KOMPAS-ZD [Development of

Knoware, Methodical Ware, and Dataware]. *Vestnik UIGTU* [Bulletin of Ulyanovsk State Technical University], 2005, no. 3, pp. 50–56.

22. Tich P., Lechta P., Maturana F., and S. Balasubramanian (2002). Industrial MAS for Planning and Control. In: *LNA*. Heidelberg, Springer Verlag, no. 2322, pp. 280–295.

23. *A software Platform based on JADE that Provides Support for the Execution of Tasks Defined According to the Workflow Metaphor*. Available at: <http://jade.tilab.com/wadeproject/>.

24. Belecheanu R.A., Munroe S., Luck M., Miller T., McBurney P., and M. Pechoucek. Commercial Applications of Agents: Lessons, Experiences and Challenges. *Proc. of AAMAS-06 – Industry Track*. ACM Press. pp. 1549–1555.