

А.Н. Афанасьев¹, Н.Н. Войт², С.Ю. Кириллов³

АВТОМАТНЫЙ МЕХАНИЗМ НА ОСНОВЕ ВРЕМЕННОЙ RV-ГРАММАТИКИ ДЛЯ ТРАНСЛЯЦИИ ДИАГРАММАТИЧЕСКИХ МОДЕЛЕЙ ДИНАМИЧЕСКИХ РАСПРЕДЕЛЕННЫХ ПОТОКОВ РАБОТ

Статья посвящена разработке транслирующей грамматики. Основная цель трансляции – это расширение методов семантического анализа диаграмматической модели распределенных потоков работ за счет возможностей целевого языка. В статье приводится описание грамматики, алгоритм её построения, отличия от обычной RV-грамматики и авторские модификации. В конце приведен пример трансляции из диаграммного языка BPMN во временную сеть Петри.

Потоки работ, бизнес процессы, BPMN, временные сети Петри, анализ диаграмм, трансляция диаграмматических моделей.

A.N. Afanasyev, N.N. Voit, S.Yu. Kirillov

AUTOMATIC MECHANISM BASED ON TEMPORAL RV-GRAMMAR FOR TRANSLATION OF DYNAMIC DISTRIBUTED WORKFLOW DIAGRAM MODELS

The article is devoted to the development of translating grammar. The main purpose of translation is to expand the methods of semantic analysis of the diagram model of distributed work flows due to the capabilities of the target language. The article describes grammar, its construction algorithm, differences from the usual RV-grammar and author's modifications. At the end of the example translation of the diagram of the BPMN language in a temporal Petri net.

Automated Systems, Diagrammatic Models, Workflows, Automaton Grammars, Timed petri nets. Diagram Translation

Введение

Диаграмматические модели, разработанные на визуальных языках (UML, IDEF, eEPC, BPMN, SharePoint, ER, DFD и др.) активно используются в практике проектирования сложных автоматизированных систем (ЦС), особенно на концептуальных этапах. Большинство синтаксически

¹ УлГТУ; г. Ульяновск, ул. Сев. Венец, 32; a.afanasev@ulstu.ru; 88422778846; д.т.н.; профессор.

² УлГТУ; г. Ульяновск, ул. Сев. Венец, 32; n.voit@ulstu.ru; 88422778846; к.т.н.; доцент.

³ УлГТУ; г. Ульяновск, ул. Сев. Венец, 32; kirillovsyu@gmail.com; 88422778846; начальник НИО ИДДО УлГТУ.

формальных графических языков не являются семантически формальными. Такие языки гибки и позволяют строить диаграммы, которые могут быть применены в различных предметных областях. Но следствием этого является то, что результирующие диаграммы можно интерпретировать неоднозначно. Машинная обработка графических диаграмм таких языков затруднена. Гибкость языка может привести к возникновению семейства языков, т.е. множества языков, которые концептуально имеют общую базу, но различную интерпретацию, специфичную для предметной области их применения. Большинство существующих подходов рассматривают такие языки изолированно, хотя достаточно определить обобщающую семантику для семейства языков (возможно для некоторых элементов она будет абстрактной) и специализировать семантическую составляющую отдельных элементов языка и(или) диаграммы перед ее интерпретацией.

1. Семантика графических языков

Все существующие графические языки по формальности языка можно разделить на следующие типы:

1. **Формальные.** Синтаксис и семантики таких языков формально определены.

2. **Полуформальные.** Синтаксис языка формален, а семантика может иметь разные интерпретации.

3. **Неформальные.** Синтаксис и семантика языка неформальны [1].

Подавляющее большинство популярных графических языков являются полуформальными. Для них и стоит исследовать методы формализации. Западных исследователей в основном предлагают придавать семантическую формальность языкам следующими методами:

1. Специализировать язык. Отказаться от некоторых возможностей языка, упростить его, наделить новыми возможностями и т.п., что положительно повлияет на формализацию.

2. Определять семантику динамически. Динамическая семантика предполагает трансформацию диаграмм базового графического языка в некий целевой язык.

Второй метод более перспективен, поскольку дает возможность на базе одного универсального инструментального средства, реализовывать диаграммы различных графических языков. Метод развивает идеи библиотек примитивов. В этом случае в библиотеках хранится интерпретация графического образа в терминах целевого графического либо текстового языка. Причем, может быть несколько интерпретаций для одного и того же графического образа, каждой из них назначается уникальное имя во избежание неоднозначного толкования и некорректного использования. В качестве целевого языка выбирают более формальный язык относительно базового [2].

2. RVTITg-грамматика

RVTITg-грамматика является развитием RVTI-грамматики [3], в котором продукции схемы грамматики расширены для хранения в них соответствия в терминах целевого формального описания, а внутренняя память хранит информацию необходимую для процесса трансляции. RVTITg-грамматикой языка L (G) является одиннадцать упорядоченных непустых множеств

$$G = (V, U, \Sigma, \tilde{\Sigma}, M, F, C, E, R, T, r_0),$$

где

- ◆ $V = \{v_e, e = \overline{1, L}\}$ – алфавит операций над внутренней памятью базового языка;
- ◆ $U = \{v_e, e = \overline{1, L}\}$ – алфавит операций над внутренней памятью, используемый целевым языком;
- ◆ $\Sigma = \{a, t = \overline{1, T}\}$ – терминальный алфавит графического языка, являющийся объединением множеств его графических объектов и связей (множество примитивов графического языка);
- ◆ $\tilde{\Sigma} = \{\tilde{a}t, t = \overline{1, T}\}$ – квазитерминальный алфавит, являющийся расширением терминального алфавита. Алфавит включает:
 - квазитермы графических объектов;
 - квазитермы графических объектов имеющих более одного входа;
 - квазитермы связей – меток с определенными для них семантическими различиями.
 - квазитерм, определяющий отсутствие связей-меток.
- ◆ $M = TTUTN$ – объединение алфавитов терминальных (TT) и нетерминальных (TN) символов целевого языка;
- ◆ $F = \{\text{generate_input}(), \text{generate_output}(), \text{select_output}(), \text{stick_connection_points}()\}$ – множество транслирующих функций по работе с элементами множества;
- ◆ C – конечное множество идентификаторов часов;
- ◆ E – множество временных выражений, определенных на C (ограничение часов и сброс часов), ограничено следующими выражениями: с начала $\{c=0\}$ и далее $\{c \sim x\}$, причем c является переменной, а x является константой, $\sim \in \{=, <, >, \geq\}$;
- ◆ $R = \{r_i, i = \overline{0, I}\}$ – схема грамматики G (множество имен комплексов продукций, причем каждый комплекс r_i состоит из подмножества P_j продукций $r_i = \{P_j, j = \overline{1, J}\}$);
- ◆ $T \in \{t_1, t_2, \dots, t_n\}$ – множество темпоральных меток;
- ◆ $r_0 \in R$ – аксиома RVTg – грамматики (имя начального комплекса продукций), $r_k \in R$ – заключительный комплекс продукций.

Графические объекты, содержащие более одного входа или выхода, нагружаются транслирующими функциями из множества F , для обеспечения соответствия входов и выходов таких объектов базового и целевого языков.

Назначение транслирующих функций и выполняемые ими операции:

1) `generate_input()` – формирование набора входных точек соединения, кроме той по которой был достигнут данный графический объект. Выполняется при первичном анализе графических объектов, содержащих более одного входа.

2) `generate_output()` – формирование набора исходящих точек соединения. Выполняется при первичном анализе графических объектов, содержащих более одного выхода, либо, когда единственный выход предполагается использовать как связь – метку, т.е. необходимо изменить направление анализа.

3) `select_output()` – выбор исходящей точки соединения элемента в качестве продолжателя цепочки целевого языка. Событийная функция, выполняемая для графических объектов с динамически изменяемым числом исходящих точек соединения после формирования набора точек соединения для исходящих связей, выбирается она из таких точек. В общем случае алгоритм выбора не регламентируется, т.е. выбор является случайным.

4) `stick_connection_points()` – связывание точек соединения связи и объекта. Событийная функция, выполняемая при вторичном анализе графических объектов, содержащих более одного входа. Производит связывание входящей связи с точкой соединения объекта, информация о которой хранится во внутренней памяти.

Наличие этих функций позволяет сформировать алгоритм построения выходной цепочки, основными операциями в котором являются выбор точки-продолжателя анализа для объектов, содержащих более одного выхода, и компоновка целостной последовательности из уже проанализированных объектов, содержащих более одного входа, и связываемых с ними анализируемых объектов. Порядок применения функций представлен на рис. 2.

Базовый язык, на котором будет проводиться разработка, это хорошо известный BPMN. Спецификация BPMN описывает условные обозначения для отображения бизнес-процессов в виде диаграмм бизнес-процессов [4]. Конечный – временная сеть Петри. Для пары базового BPMN и временной сети Петри операции над внутренней памятью схожи. Возьмем пример абстрактной диаграммы на языке BPMN. Он изображен на рис. 3

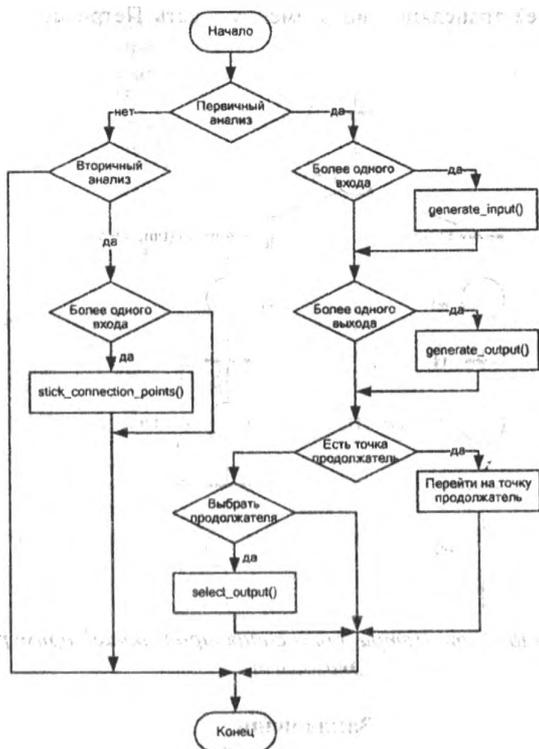


Рис. 2. Порядок применения транслирующих функций при анализе графического объекта

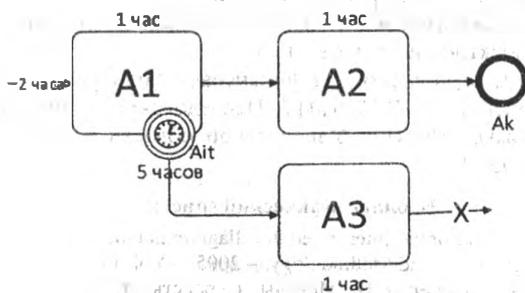


Рис. 3. Пример абстрактной диаграммы на языке BPMN

Результат её трансляции во временную сеть Петри представлен на рис. 4.

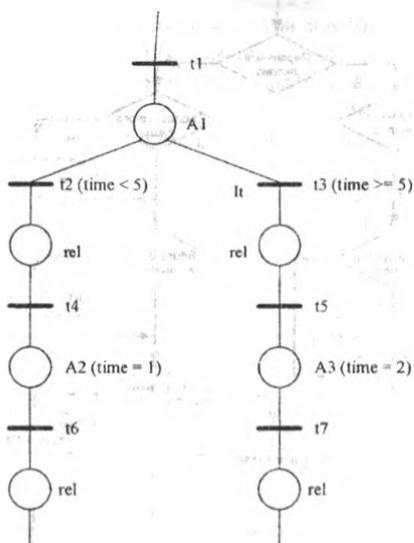


Рис. 4. Временная сеть Петри, полученная трансляцией примера BPMN диаграммы

Заключение

Разработана RVTITg-грамматика, основанная на RVTI-грамматике, которая позволяет учитывать временные характеристики и транслировать диаграмматические модели на разных графических языках. Дальнейшие направления связаны с расширением возможностей семантического анализа диаграмматических моделей в плане согласования текстовой атрибути-ки диаграмм с проектной документацией.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 17-07-01417. Исследование выполнено при финансовой поддержке РФФИ и Ульяновской области в рамках научного проекта № 18-47-730032.

Библиографический список

1. Baresi L., Pezz'e M. Formal interpreters for diagram notations // ACM Transactions on Software Engineering and Methodology. – 2005. – Vol. 14, No. 1. – P. 42-84.
2. Шаров О.Г., Афанасьев А.Н. Методы и средства трансляции графических диаграмм // Программирование. – 2011. – Т. 37, № 3. – С. 65-75.

3. Afanasyev A.N., Voit N.N. and Kirillov S.Y. Development of RYT-grammar for analysis and control dynamic workflows, in *Proc. of International Conference on Computing Networking and Informatics (ICCNi)*, (Lagos, Nigeria, 2017). – Doi: 10.1109/ICCNi.2017.8123797, URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8123797&isnumber=8123766>.
4. <https://ru.wikipedia.org/wiki/BPMN>.

УДК 004, IS&IT'18

М.Е. Уханова¹

РАЗРАБОТКА ОНТОЛОГИЧЕСКОЙ МОДЕЛИ СОГЛАСОВАНИЯ КОНСТРУКТОРСКОЙ ДОКУМЕНТАЦИИ ПРИ ПРОЕКТИРОВАНИИ СЛОЖНЫХ ТЕХНИЧЕСКИХ ИЗДЕЛИЙ

В статье рассматривается процесс конструкторской подготовки производства, включающий в себя разработку и согласование конструкторской документации. Рассматриваются соответствующие потоки работ, и предлагается онтологическая модель процесса создания конструкторской документации, позволяющая исследовать структуру изделий и связанные с ней нормативные и проектные составляющие

Конструкторская подготовка, потоки работ, онтологическая модель.

М.Е. Ukhanova

DEVELOPMENT OF AN ONTOLOGICAL MODEL FOR THE DESIGN DOCUMENTATION COORDINATION FOR THE COMPLEX TECHNICAL PRODUCTS DESIGN

The article deals with the process of design preparation of production, including the development and coordination of design documentation. The corresponding work flows are considered, and the ontological model of the process of creating design documentation is proposed, which allows to study the structure of products and related regulatory and design components

Design preparation of production, work flows, ontological model.

Введение

Задача повышения эффективности процесса конструкторской подготовки производства, которая включает в себя разработку и согласование конструкторской документации, в том числе и ее изменений, является одной из приоритетных направлений перехода к цифровому производству. Наличие электронной модели изделия, электронное согласование доку-

¹ УлГТУ; г. Ульяновск, ул. Сев. Венец, 32; mar-i-u@inbox.ru; 88422778845; аспирант.