Модел. и анализ информ. систем. Т. **17**, № **2** (2010) 5–16

УДК 681.3.06

Алгоритм преобразования моделей бизнес-процессов в одноцветные сети Петри

Доррер М. Г.

Сибирский государственный технологический университет

e-mail: mdorrer@mail.ru

получена 22 ноября 2009

Ключевые слова: бизнес-процесс, сеть Петри, инвариант

Предлагается алгоритм, позволяющий однозначно преобразовывать событийную модель бизнес-процесса в формате ARIS EPC в одноцветную сеть Петри. Полученное в результате преобразования матричное представление сети Петри дает возможность исследовать топологию бизнес-процесса, находить циклы и тупики. Кроме того, матричное описание сети Петри позволяет находить инварианты сети и тем самым решать задачу нахождения всех вариантов развития бизнес-процесса.

1. Введение

В современной практике организационного управления широкое распространение получили графические модели бизнес-процессов. Этот класс методов организационной науки изначально был предназначен для решения проблем избыточной сложности описания и исследования систем. Однако исследование графической модели, даже выполненной в соответствии с правилами структурного подхода (ограниченный контекст, ограничение числа элементов на каждом уровне декомпозиции и т.п.), представляет значительную сложность. Данная проблема осознана, попытки ее решения предпринимаются разработчиками структурных методологий и соответствующих программных средств (следует упомянуть средства семантического и синтаксического анализа моделей в ARIS ToolSet [7]). Вместе с тем, существует давно известный и обладающий мощными аналитическими возможностями при исследовании дискретных систем аппарат сетей Петри (см. [2, 4, 5]). Поскольку организационные системы, описанные при помощи событийных моделей типа IDEF3 и ARIS EPC, см. [6, 7, 9, 10]), относятся именно к этому классу, их исследование при помощи сетей Петри обещает достаточно интересные результаты.

Эти результаты можно ожидать в двух направлениях:

1. Семантический и синтаксический анализ моделей. Например, исследование инвариантов сети Петри, соответствующей бизнес-модели, позволяет получить все возможные сценарии выполнения бизнес-процесса, а эта задача в практике бизнес-консультантов достаточно востребована при подготовке процедур интеграционного тестирования корпоративной информационной системы.

2. Имитационное моделирование бизнес-систем, описанных графическими моделями. Например, исследуя переходы сети Петри на "живость", можно определить "мертвые" (никогда не выполняющиеся) функции бизнес-системы. Используя информацию о типах листьев дерева разметок сети Петри, можно определить тупиковые состояния системы, что особенно важно при анализе циклических процессов.

Цель данной статьи – предложить единый подход к получению сети Петри, моделирующей микроэкономическую систему (предприятие) на основе применяемых для бизнес-моделирования графических методологий. Подобную задачу ставит перед собой и работа ван дер Аалста и ван Хея [3], однако ее авторы изначально предлагают не только анализировать, но и моделировать потоки работ в формате сетей Петри. В практике консалтинговой работы автора заказчики, готовые согласовать модель бизнес-процессов, описанную в виде сети Петри, не встречались, поэтому практическая значимость предлагаемого решения имеет несомненную актуальность. Задача создания "очевидного эквивалентного определения диаграммы в терминах сетей Петри" в работе ван дер Аалста и ван Хея поставлена, но ее решение вынесено за рамки работы, "в качестве упражнения для заинтересованного читателя" ([3], с. 262). Вместе с тем, алгоритмы и методы анализа бизнес-процессов, описанные в [3], вполне могут быть применены и к сетям, полученным в результате предлагаемого в данной статье преобразования.

В данной работе предполагается рассматривать методы анализа систем, не ограниченных ресурсами, а в этом случае, как будет показано ниже, достаточно обычной ("одноцветной") сети Петри. Однако небольшая модернизация формул позволит исследовать и аспекты доступности ресурсов, и временных параметров процессов. В этом случае анализ минимального и максимального количества фишек в позициях сети Петри, моделирующих хранилища ресурсов, позволяет определить минимальное и максимальное наличие ресурсов в системе за анализируемый период.

2. Объект исследования

Для описания бизнес-процессов выбрана методология ARIS EPC [6, 8], которая, впрочем, безболезненно может быть заменена на IDEF3 или UML State-Chart Diagram, словом – на любую нотацию, предназначенную для описания сценариев выполнения бизнес-процессов. Примеры разработаны при помощи программного средства ARIS ToolSet [6].

В соответствии с нотацией ARIS EPC основные типы элементов моделей ARIS EPC, используемые в статье, изображаются так, как это показано на Рисунке 1.

Рассмотрим назначение этих элементов.

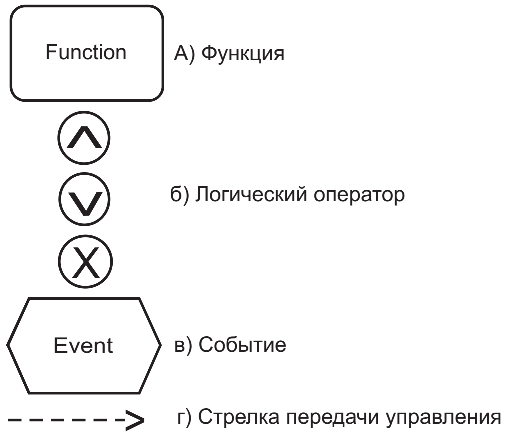

Рис. 1. Основные типы элементов моделей ARIS EPC

Объект “Функция” (рис. 1, а). Служит для описания функций (процедур, работ), выполняемых в бизнес-системе.

Логические операторы, позволяющие изобразить ветвление процесса (рис. 1, б). Соответственно, “И” – выполнение всех ветвей процесса, “Или” – выполнение хотя бы одной из ветвей процесса и “Исключающее или” – выполнение точно одной из ветвей процесса.

Событие (рис. 1, в). В ARIS EPC служит для изображения изменений состояния моделируемой системы.

Стрелки, связи (рис. 1, г). В ARIS вид связи семантически зависим от вида элементов, между которыми она установлена – например, между событием и функцией – Activate (вызывает, активирует), между функцией и документом – “Is input for” или “Has output” (является входом или имеет на выходе).

3. Сеть Петри – термины и определения

Для того, чтобы применить моделирующие возможности сетей Петри к анализу событийных моделей, предварительно необходимо решить задачу преобразования графической модели в соответствующую ей сеть.

Известно, что сеть Петри может быть представлена в трех видах: графическом, матричном и алгебраическом, причем графический вид понятен только человеку и для работы над сетями наиболее прост.

Введем сначала определения сетей Петри, которыми будем пользоваться в данной работе [2, 8]. Далее в статье будут приведены примеры, разработанные с помощью программного средства CPNTools [1].

Рассмотрим модель некоторой системы $\sum_0$, рабочий режим которой состоит в выполнении поступающих заявок. Модель реализует событийный подход, когда функционирование системы представляется в виде последовательности событий, происходящих при наличии соответствующих условий. Произошедшее событие изменяет условия, что дает возможность произойти новым событиям, и так далее. В этом случае система $\sum_0$ может быть описана в виде сети Петри

$$\sum_0 = \{\theta, P_0, T_0, F_0, M_0(0)\}, \tag{1}$$

где θ – дискретное время, $\theta = 0, 1, 2, ...$;

$P_0 = \{p_1, ..., p_n\}$ – множество узлов, называемых позициями;

$T_0 = \{t_1, ..., t_m\}$ – множество узлов, называемых переходами;

$F_0 = F_0^p \cup F_0^t$ – функция инцидентности;

$F_0^p = \left\| f_{ij}^p \right\|$ – $n \times m$ матрица, $F_0^t = \left\| f_{ji}^t \right\| - m \times n$ матрица, $f_{ij}^p \geq 0$ – кратность дуги от p_i к t_j, $f_{ji}^t \geq 0$ – кратность дуги от t_j к p_i, $i = 1, ..., n$, $j = 1, ..., m$.

Каждая позиция p_i в момент времени θ может содержать некоторый целочисленный ресурс $m_i(\theta) \geq 0$. Совокупность этих ресурсов образует вектор маркировки сети $\sum_0$:

$$M_0(\theta) = [m_1(\theta), m_2(\theta), ..., m_n(\theta)].$$

Начальная маркировка $M_0(0) = [m_{10}(0), m_{20}(0), ..., m_{n0}(0)]$ определяет наличие ресурсов в позициях p_i, $i = 1, ..., n$ в момент старта системы.

Функционирование сети Петри заключается в изменении маркировки $M(\theta)$ путем срабатывания переходов. Переход t_j может сработать в момент θ, если

$$m_i(\theta) \geq f_{ij}^p, \quad i = 1, ..., n. \tag{2}$$

Переход, удовлетворяющий условию (2), называется разрешенным в момент θ. При срабатывании разрешенного перехода t_j происходит изменение маркировок позиций по правилу:

$$m_i(\theta + 1) = m_i(\theta) - f_{ij}^p + f_{ji}^t, \quad i = 1, ..., n. \tag{3}$$

Пусть t_j – разрешенный переход в момент времени θ , $t_j \in T_0$. Введем m-вектор $\tau_j(\theta) = [0, 0, ..., 1, ..., 0]$, в котором на j-м месте стоит единица, а все остальные элементы равны нулю. Тогда формула (3) может быть представлена в векторном виде

$$M_0(\theta + 1) = M_0(\theta) + \tau_i(\theta)\Phi_0, \quad \theta = 0, 1, 2, ..., \tag{4}$$

где

$$\Phi_0 = \left\| f_{ji}^t - f_{ij}^p \right\| = F_0^t - (F_0^p)^T \tag{5}$$

– $m \times n$ матрица приращений ресурсов в позициях системы $\sum_0$.

Таким образом, функционирование системы $\sum_0$ заключается в последовательном выполнении двух действий: а) проверка возможности срабатывания перехода по формуле (2) и б) при выполнении этого условия пересчет маркировок по формуле (4). Результаты функционирования системы $\sum_0$ могут быть представлены либо в виде дерева маркировок $M(\theta)$, получаемого при последовательном срабатывании всех разрешенных переходов, либо в виде свободного языка сети Петри, в котором алфавит состоит из имен переходов t_j, $j = 1, ..., m$, а слова представляют собой последовательность имен переходов, сработавших к моменту θ [1].

Отметим особенности сети Петри, моделирующей рассматриваемый в данной работе класс систем $\sum_0$.

Во-первых, мы будем иметь дело с так называемыми безопасными сетями Петри, у которых ресурс в каждой позиции $m_i(\theta)$ может принимать значения 0 или 1. При этом $m_i(\theta) = 1$ свидетельствует о том, что имеется условие для выполнения определенной функции системы. Соответственно при $m_i(\theta) = 0$ такое условие не выполнено.

Во-вторых, кратности всех дуг рассматриваемой сети f_{ij}^p и f_{ji}^t, $i = 1, ..., n$, $j = 1, ..., m$ могут принимать только значения 0 или 1, что говорит либо об отсутствии связи между узлами, либо о передаче (или потреблении) единичного ресурса от узла к узлу. Благодаря такому характеру связей обеспечивается свойство безопасности сети.

Таким образом, система описывает топологию бизнес-модели и правила передачи управления между бизнес-функциями. На физическом уровне эти ограничения соответствуют исследованию системы без учета движения ресурсов в системе.

4. Алгоритм преобразования – общие положения

Основные предположения, на которых строится процедура исследования, заключаются в следующем:

Диаграмма в нотации ARIS eEPC в упрощенном виде может быть описана графом D.

$$D = (S_0, L_0), \tag{6}$$

где

$S_0 = \{s_1, ..., s_q\}$ – множество символов модели *EPC* (процессы, события, перекрестки), q – количество символов;

$L_0 = \{l_1, ..., l_r\}$ – множество связей модели *EPC,* r – количество связей.

Тогда D может быть задано в виде функций инцидентности, подобно тому, как это делается при описании сетей Петри.

$F_0^L = \left\|f_{ij}^L\right\|$ – $r \times q$ матрица, $F_0^S = \left\|f_{ji}^S\right\|$ – $q \times r$ матрица, $f_{ij}^L \subset [0, 1]$ – инцидентность l_i с s_j , $f_{ji}^S \subset [0, 1]$ – инцидентность s_j с l_i , $i = 1, ..., r$, $j = 1, ..., q$..

Для трансляции модели *EPC* в соответствующую безопасную сеть Петри необходимо выполнить следующие действия.

Сопоставим каждому символу модели EPC переход сети Петри, а каждой связи – позицию сети Петри:

Для каждого l_i создается соответствующий ему p_i, тем самым на этом этапе $n = r$.

Для каждого s_j создается соответствующий ему t_j, на этом этапе $m = q$.

После этих сопоставлений первоначальный вариант функции инцидентности сети Петри может быть задан простым приравниванием $F_0^P = F_0^L$ и $F_0^T = F_0^S$. Заметим при этом, что в каждой строке F_0^P и в каждом столбце F_0^T может быть только один – не более и не менее – элемент, равный 1. Это обусловлено тем, что каждая связь модели *ARIS* имеет только одно начало и одно завершение, соответственно позиция, моделирующая связь, обязательно должна быть инцидентна ровно с одним переходом, моделирующим символ модели на входе, и с одним – на выходе.

5. Обеспечение связей на вход и на выход

Для обеспечения моделирующих возможностей сети Петри необходимо ввести дополнительные позиции сети Петри, соответствующие началу и завершению цепочки процесса. Для этого добавляются дополнительные позиции для каждого перехода, не имеющего инцидентной позиции на входе или на выходе.

Обнаружив такую позицию, в матрицу F_0^P необходимо ввести дополнительную $(n{+}1)$ строку, а в матрицу F_0^T – соответственно дополнительный $(n{+}1)$ столбец. Связи между соответствующими переходами j и вновь введенными позициями задаются указанием значения 1 в соответствующих позициях $f^p_{n+1,j}$ и $f^t_{j,n+1}$, остальные ячейки в этих строках и столбцах имеют значение 0.

Затем значение n увеличивается на единицу, и процесс повторяется для следующего перехода, соответствующего условиям, описанным выше, до тех пор, пока все они не будут перебраны.

6. Моделирование оператора, описывающего слияние ветвей процесса

Для обеспечения адекватного моделирования развития процесса на логических операторах необходимо заменить переходы, сопоставленные с каждым из логических операторов, на набор из переходов по правилу таблиц истинности соответствующего оператора, а также соответствующим образом дополнить матрицы инцидентности F_0^P и F_0^T. Следует отметить, что предлагаемый в данной статье подход преобразования логических операторов событийной модели исчерпывающе формализует набор “типичных конструкций” сети Петри, предложенный ван дер Ааалстом и ван Хеем ([3], с. 58) в виде априорно заданных конфигураций, причем без разъяснения логики формирования развилок и для числа ветвей не более 3. Предлагаемая матричная формула исчерпывающе описывает построение фрагментов сетей Петри, эквивалентных любым мыслимым логическим операторам для общего случая N ветвей процесса.

После первоначального сопоставления модели EPC, включающей оператор (перекресток) с сетью Петри, соответствующий фрагмент сети выглядит так, как показано на рисунке 2. Каждая связь соответствует позиции, оператору соответствует переход.

Далее один переход заменяется на набор переходов по следующему правилу. Пусть переход соответствует перекрестку бизнес-модели, на котором сходится r ветвей процесса. В сеть Петри вместо одного перехода, сопоставленного с перекрестком, вводятся от 1 до 2^r переходов. Каждый из вводимых переходов соответствует “единице” таблицы истинности r-арной функции алгебры логики, которой обозначен перекресток бизнес-модели.

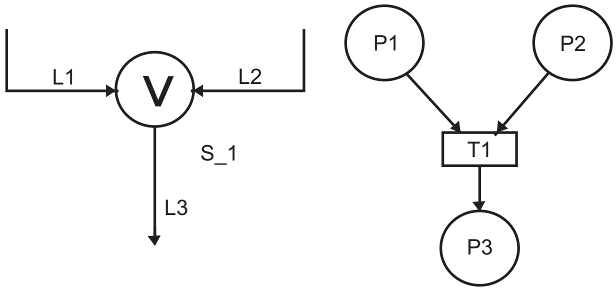

Рис. 2. Первый этап преобразования в сеть Петри фрагмента модели EPC с оператором, моделирующим слияние ветвей процесса

Обозначим матрицу инцидентности между позициями, соответствующими сходящимся ветвям модели бизнес-процесса, и переходами, соответствующими "единицам" таблицы истинности, T_M^P. Ее размерность – $n_T \times m_T$, где n_T – количество столбцов матрицы, соответствующее числу "единиц" таблицы истинности R-арной функции алгебры логики, которой обозначен перекресток бизнес-модели, $m_T = r$ – количество строк матрицы, соответствующее числу сходящихся на перекресток ветвей процесса. Матрица инцидентности состоит из столбцов (транспонированных строк), соответствующих наборам аргументов, доставляющих "единицу" функции.

Так, в приведенном на рисунке 2 примере это будет функция "или" с двумя аргументами. Ее таблица истинности:

X_1	X_2	"или"
1	1	1
1	0	1
0	1	1
0	0	0

Таким образом, для функции "или" и двух аргументов (двух сходящихся ветвей процесса) вводится три перехода вместо исходного одного, а $T_M^P = \begin{vmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{vmatrix}$.

Матрица F_0^T дополняется T_M^T – матрицей инцидентности переходов, вводимых взамен исходного. У каждого из этих переходов одна выходная позиция, соответствующая выходу с перекрестка в модели EPC. Ее размерность – $1 \times n_T$, где n_T – количество столбцов матрицы, соответствующее числу "единиц" таблицы истинности r-арной функции алгебры логики, которой обозначен перекресток бизнес-модели. Все значения T_M^T равны "1", поскольку с позицией сети Петри, соответствующей выходной связи, инцидентны все вновь введенные переходы. В приведенном примере $T_M^T = \begin{vmatrix} 1 \\ 1 \\ 1 \end{vmatrix}$.

Рассмотрим замену i-го перехода, соответствующего перекрестку на схождение процесса, из r ветвей, которые полагаем сгруппированными подряд, начиная с j-й. После прохождения i-го перекрестка процесс развивается по ветви, соответствующей k-й позиции сети Петри, исходные матрицы инцидентности преобразуются следующим образом:

$$F_0^P = \left[\begin{array}{ccc} \left| \begin{array}{ccc} f_{11}^p & \cdots & f_{1,j-1}^p \\ \cdots & \cdots & \cdots \\ f_{i-1,1}^p & \cdots & f_{i-1,j-1}^p \end{array} \right| & |0| & \left| \begin{array}{ccc} f_{1,j+r+1}^p & \cdots & f_{1,m}^p \\ \cdots & \cdots & \cdots \\ f_{i-1,j+r+1}^p & \cdots & f_{i-1,m}^p \end{array} \right| \\ |0| & T_M^P & |0| \\ \left| \begin{array}{ccc} f_{i+1,1}^p & \cdots & f_{i+1,j-1}^p \\ \cdots & \cdots & \cdots \\ f_{1,n}^p & \cdots & f_{n,j-1}^p \end{array} \right| & |0| & \left| \begin{array}{ccc} f_{i+1,j+r+1}^p & \cdots & f_{i+1,m}^p \\ \cdots & \cdots & \cdots \\ f_{n,j+r+1}^p & \cdots & f_{n,m}^p \end{array} \right| \end{array} \right] \qquad (7)$$

$$F_0^T = \left[\begin{array}{ccc} \left| \begin{array}{ccc} f_{11}^t & \cdots & f_{1,i-1}^t \\ \cdots & \cdots & \cdots \\ f_{k-1,1}^t & \cdots & f_{k-1,i-1}^t \end{array} \right| & |0| & \left| \begin{array}{ccc} f_{1,i+1}^t & \cdots & f_{1,n}^t \\ \cdots & \cdots & \cdots \\ f_{k-1,i+1}^t & \cdots & f_{k-1,n}^t \end{array} \right| \\ |0| & T_M^T & |0| \\ \left| \begin{array}{ccc} f_{k+1,1}^t & \cdots & f_{k+1,i-1}^t \\ \cdots & \cdots & \cdots \\ f_{m,1}^t & \cdots & f_{m,i-1}^t \end{array} \right| & |0| & \left| \begin{array}{ccc} f_{k+1,i+1}^t & \cdots & f_{k+1,n}^t \\ \cdots & \cdots & \cdots \\ f_{m,i+1}^t & \cdots & f_{m,n}^t \end{array} \right| \end{array} \right] \qquad (8)$$

Здесь и далее в формулах $|0|$ – нулевая матрица соответствующей размерности.

Графически фрагмент сети Петри в приведенном примере после преобразования выглядит так, как это показано на рисунке 3.

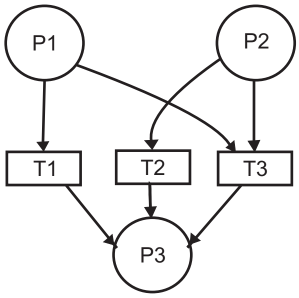

Рис. 3. Второй этап преобразования в сеть Петри фрагмента модели EPC с оператором, моделирующим слияние ветвей процесса

При анализе полученной сети Петри можно видеть, что начальные разметки, допустимые исходя из физической сути моделируемого явления, соответствуют возможным комбинациям передачи управления перекрестку по связям. Фишка – процессный маркер может появиться в позициях $P1$, $P2$ или в них обеих одновременно. При появлении фишки в позиции $P1$ сработает переход $T1$, в $P2$ – $T2$, и, наконец, появление фишек в позициях $P1$ и $P2$ одновременно будет отработано либо срабатыванием $T3$, либо последовательным срабатыванием $T1$ – $T2$ или $T2$ – $T1$. Таким образом, сеть воспроизведет все возможные комбинации прохождения перекрестка, причем наличие фишек в обеих входных позициях будет проанализировано как для синхронного прохождения процесса по обеим входным ветвям (срабатывание перехода $T3$), так и для асинхронного (последовательности $T1$ – $T2$ и $T2$ – $T1$).

7. Моделирование оператора, описывающего разветвление хода процесса

После первоначального сопоставления модели EPC, включающей оператор (перекресток) с сетью Петри, соответствующий фрагмент сети выглядит так, как показано на рисунке 4. Каждая связь соответствует позиции, оператору соответствует переход.

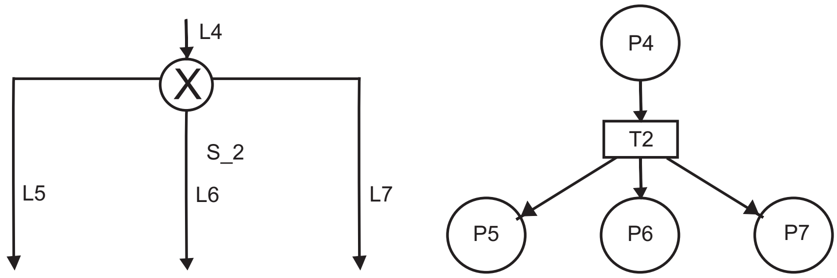

Рис. 4. Первый этап преобразования в сеть Петри фрагмента модели EPC с оператором, моделирующим разветвление хода процесса по нескольким ветвям

Далее один переход заменяется на набор переходов по правилам, подобным тем, которые применяются при моделировании развилок на слияние. В сеть Петри вместо одного перехода, сопоставленного с перекрестком, вводятся от 1 до 2^r переходов. Каждый из вводимых переходов соответствует "единице" таблицы истинности r-арной функции алгебры логики, которой обозначен перекресток бизнес-модели.

Введем матрицы T_S^P и T_S^T, подобные тем, что применялись для описания слияния ветвей процесса. Однако при решении задачи на разветвление более важной смысловой нагрузкой будет обладать матрица T_S^T, описывающая связи между вновь введенными переходами и позициями, моделирующими разветвляющиеся связи процесса. Ее размерность – $m_T \times n_T$, где n_T – количество строк матрицы, соответствующее числу "единиц" таблицы истинности r-арной функции алгебры логики, которой обозначен перекресток бизнес-модели, m_T – количество столбцов матрицы, соответствующее числу расходящихся из перекрестка ветвей процесса.

Матрица инцидентности, таким образом, состоит из строк, соответствующих наборам аргументов, доставляющих "единицу" функции. Вообще, при равном количестве ветвей процесса и одинаковой функции в перекрестке $T_M^P = \left(T_S^T\right)^T$, то есть матрица, дополняющая исходную матрицу инцидентности по переходам (F_0^T) при моделировании схождения ветвей процесса равна транспонированной матрице, дополняющей исходную матрицу инцидентности по позициям (F_0^P) при моделировании расхождения ветвей процесса. В приведенном на рисунке 3 примере это будет функция "исключающее или" с тремя аргументами. Ее таблица истинности:

Вводится три перехода вместо исходного одного, а $T_S^T = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$.

Матрица F_0^P дополняется T_S^P – матрицей инцидентности переходов, вводимых взамен исходного. У каждого из этих переходов одна входная позиция, соответству-

X_1	X_2	X_3	“исключающее или”
1	1	1	0
1	1	0	0
1	0	1	0
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	1
0	0	0	0

ющая входу на перекресток в модели EPC. Ее размерность – $n_T \times 1$, где n_T – количество столбцов матрицы, соответствующее числу “единиц” таблицы истинности r-арной функции алгебры логики, которой обозначен перекресток бизнес-модели. Все значения T_S^P равны 1, поскольку с позицией сети Петри, соответствующей выходной связи, инцидентны все вновь введенные переходы. В приведенном примере $T_S^P = \begin{vmatrix} 1 & 1 & 1 \end{vmatrix}$.

Рассмотрим замену i-го перехода, соответствующего перекрестку на расхождение процесса на r ветвей, которые моделируются соответствующим числом позиций сети Петри. Полагаем позиции сгруппированными подряд, начиная с j-й. Процесс приходит на i-й перекресток по ветви, соответствующей k-й позиции сети Петри. При этих условиях исходные матрицы инцидентности преобразуются следующим образом:

$$F_0^P = \begin{bmatrix} \begin{vmatrix} f_{11}^p & \cdots & f_{1,k-1}^p \\ \cdots & \cdots & \cdots \\ f_{i-1,1}^p & \cdots & f_{i-1,k-1}^p \end{vmatrix} & |0| & \begin{vmatrix} f_{1,k+1}^p & \cdots & f_{1,m}^p \\ \cdots & \cdots & \cdots \\ f_{i-1,k+1}^p & \cdots & f_{i-1,m}^p \end{vmatrix} \\ |0| & T_S^P & |0| \\ \begin{vmatrix} f_{i+1,1}^p & \cdots & f_{i+1,k-1}^p \\ \cdots & \cdots & \cdots \\ f_{1,n}^p & \cdots & f_{n,k-1}^p \end{vmatrix} & |0| & \begin{vmatrix} f_{i+1,k+1}^p & \cdots & f_{i+1,m}^p \\ \cdots & \cdots & \cdots \\ f_{n,k+1}^p & \cdots & f_{n,m}^p \end{vmatrix} \end{bmatrix} \tag{9}$$

$$F_0^T = \begin{bmatrix} \begin{vmatrix} f_{11}^t & \cdots & f_{1,i-1}^t \\ \cdots & \cdots & \cdots \\ f_{j-1,1}^t & \cdots & f_{j-1,i-1}^t \end{vmatrix} & |0| & \begin{vmatrix} f_{1,i+1}^t & \cdots & f_{1,n}^t \\ \cdots & \cdots & \cdots \\ f_{j-1,i+1}^t & \cdots & f_{j-1,n}^t \end{vmatrix} \\ |0| & T_S^T & |0| \\ \begin{vmatrix} f_{j+r+1,1}^t & \cdots & f_{j+r+1,i-1}^t \\ \cdots & \cdots & \cdots \\ f_{m,1}^t & \cdots & f_{m,i-1}^t \end{vmatrix} & |0| & \begin{vmatrix} f_{j+r+1,i+1}^t & \cdots & f_{j+r+1,n}^t \\ \cdots & \cdots & \cdots \\ f_{m,i+1}^t & \cdots & f_{m,n}^t \end{vmatrix} \end{bmatrix} \tag{10}$$

Очевидно, что после таких преобразований – причем это касается преобразований как на схождение, так и на разветвление процесса, количество переходов m увеличивается на n_T.

Графически фрагмент сети Петри в приведенном примере после преобразования выглядит следующим образом:

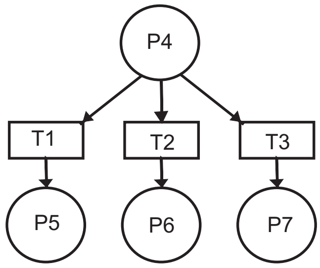

Рис. 5. Второй этап преобразования в сеть Петри фрагмента модели EPC с оператором, моделирующим разветвление хода процесса по нескольким ветвям.

Очевидно, что стартовая разметка сети только одна – фишка в позиции, соответствующей входной на перекресток связи.

В этой ситуации переходы $T1$, $T2$ и $T3$ конкурируют за эту фишку, так что дерево разметок будет включать отдельные ветви, соответствующие срабатыванию каждого из этих переходов. На содержательном уровне это будет соответствовать моделированию срабатывания точно одной из ветвей процесса, что и предполагается семантикой оператора ветвления "исключающее или".

8. Заключение

Таким образом, предложен набор формул, позволяющих однозначно конвертировать матричную форму описания бизнес-модели в формате ARIS EPC в матричную форму описания одноцветной сети Петри. Полученное описание сети Петри позволяет исследовать топологию бизнес-процесса, находить циклы и тупики. Кроме того, матричное описание сети Петри позволяет находить инварианты сети, что позволяет решать задачу нахождения всех вариантов развития бизнес-процесса.

В настоящее время ведется работа по реализации программного обеспечения, позволяющего строить сценарии интеграционного тестирования для моделей бизнес-процессов, описанных при помощи ARIS ToolSet. Результаты этой работы будут рассмотрены в отдельной статье.

Список литературы

1. http://wiki.daimi.au.dk/cpntools/cpntools.wiki

2. Jensen K. Coloured Petri nets: Basic concepts, analysis methods and practical use. Vol. 1. Basic concepts. Berlin a. o.: Springer-Verlag, 1996.

3. Аалст Вил ван дер, Хей Кейс ван. Управление потоками работ: модели, методы и системы. М.: Физматлит, 2007.

4. Доррер Г.А. Исследование и проектирование региональных информационно-телекоммуникационных систем на основе имитационного моделирования // Вестник СибГТУ. 1999, №1. С. 146-159.

5. Доррер М.Г., Курохтин В.В. Решение задачи прямого и обратного преобразования между цветной сетью Петри и моделью бизнес-процессов // Вестник Сибирского Государственного Аэрокосмического Университета. Красноярск: СибГАУ, 2006. Выпуск 5. С. 83-87.

6. Калянов Г.Н. Теория и практика реорганизации бизнес-процессов. М.: СИНТЕГ, 2000.

7. Каменнова М., Громов А., Ферапонтов М., Шматалюк А. Моделирование бизнеса. Методология ARIS: Практическое руководство. М.: Весть-Метатехнология, 2001.

8. Котов В.Е. Сети Петри. М.: Наука, 1984.

9. Шеер А.В. Бизнес-процессы. Основные понятия. Теория. Методы: Пер. с англ. 2-е изд., испр. и доп. М.: Просветитель, 1999.

10. Шеер А.В.. Моделирование бизнес-процессов: Пер. с англ. 2-е изд., испр. и доп. М.: Серебряные нити, 2000.

An algorithm of converting business-process models into monochrome Petri nets on the basis of matrix formulas

Dorrer M.G.

Keywords: business-process, Petri net, invariant

A mechanism allowing the unique conversion of an event-based business process model of the ARIS EPC format into a monochrome Petri net is suggested. The Petri net matrix representation resulting from this conversion allows investigating a business process topology and finding cycles and dead-ends. Moreover the Petri net matrix representation allows finding net invariants thus solving a task of finding all the business process development variants.

Сведения об авторе:
Доррер Михаил Георгиевич,
Сибирский государственный технологический университет,
канд. техн. наук, доцент;
ведущий консультант компании "Партнер Бизнес Консалтинг",
г. Красноярск.