

УДК [004.41+ 519.711.7]:658

DOI 10.34822/1999-7604-2020-1-53-60

## АНОМАЛИИ ДАННЫХ В МОДЕЛЯХ БИЗНЕС-ПРОЦЕССОВ

**А. Г. Ивашко, Е. О. Кабардинский, И. Г. Семихина <sup>✉</sup>, Д. В. Семихин**

*Тюменский государственный университет, Тюмень, Россия*

<sup>✉</sup> E-mail: i.g.semikhina@utmn.ru

Статья посвящена проблеме валидации бизнес-процессов при разработке решений на основе платформ их моделирования. Приведена классификация известных на сегодняшний день моделей валидации потоков данных в бизнес-процессах, а также рассмотрены виды моделей бизнес-процессов: сети Петри, мета-графы, модели на основе действий, комбинированные модели, матрица потоков данных. Рассмотрены преимущества и недостатки каждой модели, а также представлено краткое описание основных компонентов модели. Описаны виды аномалий данных и алгоритмы, используемые для валидации потока данных бизнес-процесса. Особое внимание уделяется методам валидации, использующим матрицу потоков данных в качестве основного инструмента анализа потоков данных. Приводится алгоритм проверки аномалий отсутствия и избыточности данных с использованием матрицы потоков данных. Проанализирован пример модели бизнес-процесса, в котором бизнес-процесс работает с несколькими доменными моделями данных. Сделан вывод о ложном срабатывании алгоритмов, построенных на основе матрицы потоков данных, при работе с несколькими доменными моделями данных. Предлагается выяснить возможность применения онтологической модели концептуальных объектов для идентификации семантических конфликтов между наборами данных из множества доменных моделей.

*Ключевые слова:* бизнес-процесс, потоки данных, аномалии данных, матрица потоков данных.

## DATA ANOMALIES IN BUSINESS PROCESS MODELS

**A. G. Ivashko, E. O. Kabardinsky, I. G. Semikhina <sup>✉</sup>, D. V. Semikhin**

*University of Tyumen, Tyumen, Russia*

<sup>✉</sup> E-mail: i.g.semikhina@utmn.ru

The article is devoted to the problem of validating business processes when developing solutions based on business process modeling platforms. The article provides a classification of the currently known models of validating data flows in business processes and also considers the types of business processes models: Petri nets, metagraphs, activity-based models, combined models, data flow matrix. The advantages and disadvantages of each model are analyzed, as well as a brief description of the main components of the models. The types of data anomalies and the algorithms used to validate the data flow of a business process are presented. Particular attention is paid to validation methods that use the data flow matrix as the main tool for analyzing data flows. An algorithm for checking anomalies in the absence and redundancy of data using a data flow matrix is given. An example of a model in which a business process works with several domain data models is analyzed. The analyzed example leads to the conclusion that the algorithms built upon the data flow matrix are false when working with several domain data models. The article proposes to find out the possibility of using an ontological model of conceptual objects to identify semantic conflicts between data sets from a variety of domain models.

*Keywords:* business process, data flows, data anomalies, data flow matrix.

**Введение.** Моделирование рабочего процесса или бизнес-процесса можно разделить на несколько аспектов, включая структурные, информационные, временные, транзакционные

и функциональные [1, 2]. Традиционно моделирование рабочих процессов фокусировалось на структурных аспектах, главным образом указывая порядок выполнения действий компонента в процессе с помощью различных языков спецификаций процесса/процессов. Существует большое количество предложений для спецификации процесса, каждое из которых имеет свои достоинства и недостатки, а также различные уровни выразимости и грамматики.

Важным этапом создания модели процесса является синтаксическая и структурная проверка в терминах базового языка. Основываясь на ней, можно было бы, по крайней мере, утверждать, что модель процесса может быть выполнима в данной системе управления процессами. Проверка может выявить как структурные ошибки (вызывающие тупики или взаимоблокировки), так и действия, вызывающие множественные исполнения. При этом используемый язык спецификации процесса и критерии корректности будут определять тип выявленных ошибок. Реализация структурной проверки там, где проверка синтаксической корректности была достаточно простой задачей, может потребовать использования комплексных алгоритмов, но независимо от бизнес-контекста является общей для всех моделей, построенных на определенном языке [3–5].

### **Модели бизнес-процессов**

На сегодняшний день известно несколько возможных моделей бизнес-процессов:

1. Сети Петри.
2. Мета-графы.
3. Модели на основе действий.
4. Комбинированные модели [6, 7].
5. Матрица потоков данных.

Все модели условно можно разделить на два класса:

1. Моделирующие структурные аспекты:
  - сети Петри;
  - мета-графы;
  - модели на основе действий;
  - комбинированные модели.
2. Моделирующие поток данных:
  - матрица потоков данных.

Разделение моделей на классы тесно связано с набором аномалий, которые могут выявлены с помощью моделей:

1. Моделирующих структурные аспекты:
  - несоответствие данных;
  - потерю данных;
  - противоречие данных;
  - неверную адресацию данных;
  - недостаточность данных.
2. Моделирующих поток данных:
  - избыточность данных;
  - отсутствие данных.

Как правило, для валидации бизнес-процесса используют несколько моделей, что позволяет выявлять аномалии, связанные со структурными аспектами, и аномалии потоков данных.

Так как матрица потока данных является единственной моделью, которая моделирует поток данных в бизнес-процессе, все алгоритмы валидации используют ее для выявления аномалий избыточности и отсутствия данных.

**Сети Петри.** Сетью Петри называется двудольный ориентированный граф  $N = \langle P, T, * \rangle$ , где  $P = \{p_i\}$ ,  $T = \{t_i\}$  – конечные непустые множества вершин, называемые соответственно позициями (места) и переходами;  $*$  – отношение между вершинами, соответствующее дугам графа. Позиции или переходы, являющиеся вершинами этого графа, соединяются дугами. Вершины одного типа соединяться друг с другом не могут. Метки (маркеры, фишки, токены)

ны), размещенные в позициях, могут перемещаться по сети (рис. 1). Распределение меток по позициям определяет состояние сети. Дуги графа строятся для каждого активированного перехода и помечаются соответствующим символом. Построение прекращается, когда сеть Петри содержит маркировку, в которой не активирован ни один переход, либо позиция содержится в графе [8].

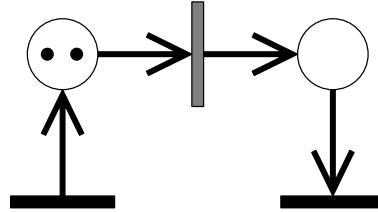


Рис. 1. Переход метки из  $p_1$  в  $p_2$  в сети Петри [9]

Разновидностью сетей Петри являются сети потоков работ (так называемые Workflow). Их формализованное описание было сделано Wil van der Aalst для моделирования потоков работ в workflow-системах.

Сеть Петри можно отнести к Workflow, если выполняются следующие условия:

- существует только одна исходная позиция  $i$ , в которой отсутствуют входящие переходы;
- существует только одна конечная позиция  $o$ , в которой отсутствуют выходящие переходы;
- каждый узел данной сети расположен на пути от  $i$  к  $o$  [9].

Для преобразования бизнес-процессов к Workflow сетям Петри используют как проприетарные алгоритмы, так и открытые.

Алгоритмы валидации бизнес-процесса, построенные на основе сетей Петри, используют матрицу потоков данных, позволяющую дополнить структурные аспекты процесса потоками данных [10]. Основным недостатком заключается в методе моделирования входного/выходного набора данных действия – структура данных должна быть развернута в плоскую структуру, что делает невозможным определение аномалии несоответствия данных, так как структура данных, содержащая коллекции связанных наборов данных, не может быть обработана с помощью матрицы потоков данных [6].

**Метаграф.** Метаграф представляет собой обобщение множеств и ориентированных графов. Вершины графа принято называть метавершинами, а ребра метаграфа – метаребрами. Метаребра отражают следующие ориентированные связи (рис. 2):

1. Связь вершины и вершины.
2. Связь вершины и множества вершин.
3. Связь множества вершин и множества вершин.

Отражение бизнес-процесса в метаграф происходит следующим образом:

1. Задачи бизнес-процесса представляются в виде метавершин.
2. Управляющие элементы отображаются в виде метаребер.

3. Задачи бизнес-процесса, объединенные управляющими элементами, например, условным координатором, объединяются в множества.

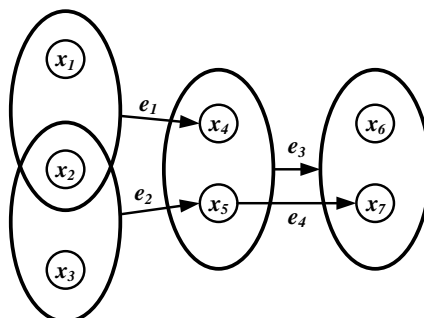


Рис. 2. Пример метаграфа для вершин  $x_1$ –  $x_7$  [11]

Основная отличительная особенность метаграфов состоит в том, что они могут представлять не только отношения между отдельными элементами, но и отношения между наборами элементов. Еще одна важная особенность метаграфов заключается в том, что они поддерживают формальный анализ рабочего процесса с точки зрения информации и отвечают на вопросы не только о задачах, но и о данных и ресурсах. Однако ни один из этих методов моделирования не фокусируется на моделировании потока данных, и поэтому модели на основе метаграфа не могут поддерживать проверку потока данных для определения аномалий данных [6].

**Модели на основе действий.** Модель на основе действий представляет собой граф, построенный на основе модели бизнес-процесса, где узлы графа представляют собой задачи бизнес-процесса, а также управляющие элементы. Узлы графа, обозначающие управляющие элементы в оригинальном бизнес-процессе, принято называть координаторами (рис. 3).

Узлы-координаторы позволяют строить поток управления переходов от задач к задачам в соответствии требованиям бизнес-процесса. Выделяют следующие типы координаторов [5]:

- координатор последовательности;
- условный координатор;
- координатор слияния;
- координатор разбиения;
- координатор синхронизации.

Узел задач также содержит описание входного и выходного набора данных, необходимого для обработки атомарного действия.

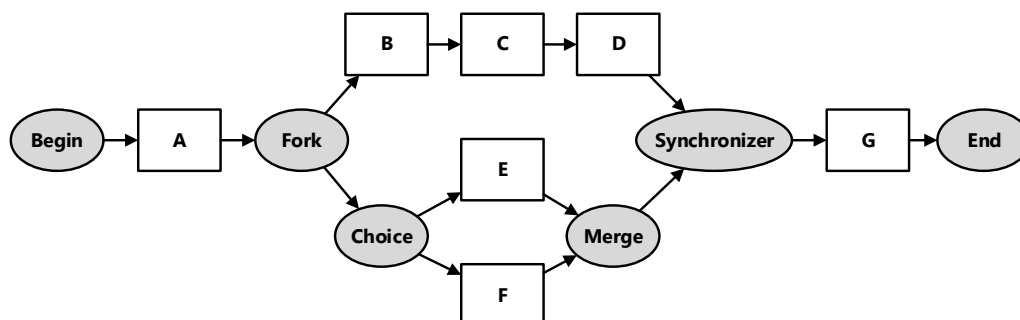


Рис. 3. Пример графического представления модели на основе действий [5]

В моделировании рабочих процессов на основе действий нет простого способа сбора потока данных. Хотя метаграфы могут помочь в анализе потока данных, метаграфы не предоставляют ни инструментов для проверки синтаксической и семантической корректности модели, ни методов для прямого отображения элементов данных, моделируемых в метаграфах, в конкретные элементы данных, хранящиеся в базах данных. Таким образом, модели на основе действий не ориентированы на моделирование и проверку потока данных, следовательно, не могут быть использованы для определения аномалий потока данных.

**Комбинированные модели.** Комбинированные модели, как правило, используют для интеграции одной модели с другой в целях устранения недостатков друг друга. Примером интеграции может служить комбинация моделей: сети Петри, матрицы потоков данных и модели на основе действий. Данная модель была описана в статье «Detection of Dataflow Anomalies in Business Process» [7].

**Матрица потоков данных.** Матрица потоков данных представляет собой матрицу размерности  $|D| \times |V|$ , где:

- $D$  – множество атомарных элементов данных, используемых в задачах бизнес-процесса;
- $V$  – множество задач в бизнес-процессе.

В каждой ячейке матрицы потоков данных содержится значение  $r$ ,  $w$ , пустое значение, а также  $r$  и  $w$  одновременно, где:

- $r$  – чтение элемента данных;
- $w$  – запись элемента данных.

Алгоритм проверки аномалий достаточно прост: для каждой задачи в процессе проверить каждый считанный/записанный атомарный элемент данных:

1. В случае считывания – проверить все предшествующие задачи на предмет записи атомарного элемента данных. Если элемент не был записан ранее, отобразить аномалию отсутствия данных пользователю в задаче  $V_i$  для элемента данных  $D_j$ .

2. В случае записи – проверить все последующие задачи на предмет чтения атомарного элемента данных. Если элемент не был считан последующими задачами, отобразить аномалию избыточности данных пользователю в задаче  $V_i$  для элемента данных  $D_j$ .

**Аномалии данных.** Аномалии данных представляют собой ошибки бизнес-процесса в потоке данных, связанные с входными/выходными параметрами задач (действий). Выделяют 7 типов аномалий [5]:

- несоответствия данных;
- избыточности данных;
- потери данных;
- отсутствия данных;
- противоречия данных;
- неверной адресации данных;
- недостаточности данных.

Аномалия несоответствия данных возникает в том случае, когда структура данных, созданная источником, несовместима со структурой, требуемой операцией.

Аномалия избыточности данных возникает в том случае, когда структура данных, созданная источником, содержит поля, не используемые приемником данных.

Аномалия потери данных возникает в том случае, когда один и тот же элемент выходных данных является результатом нескольких задач в процессе. При параллельном выполнении таких задач данные могут быть обновлены любой из задач, в результате чего итог работы других задач будет утерян. В классическом программировании такой эффект принято называть «Условие гонки».

Аномалия отсутствия данных возникает в том случае, когда элемент выходных данных был указан во входной схеме для действия, но источник элемента данных неизвестен/не указан.

Аномалия противоречия данных возникает в том случае, когда действие извлекает элемент данных из приложения или участника-человека и сохраняет этот элемент и его значение в хранилище данных процесса. Этот элемент данных и его значение могут быть использованы позже в рабочем процессе другим действием. Проблема заключается в том, что в течение этого простоя (после того как элемент был записан в хранилище и до момента прочтения) элемент данных может быть обновлен внешне для рабочего процесса, что может вызвать несогласованное представление данных.

Аномалия неверной адресации данных возникает в том случае, когда поток данных бизнес-процесса построен таким образом, что элемент данных может быть считан до того, как был записан. Основная причина такой аномалии – проблема согласованности потока данных и потока управления в бизнес-процессе.

Аномалия недостаточности данных является критической и в то же время достаточно трудной для обнаружения аномалией. В некоторых случаях, когда указанных данных достаточно для успешного завершения процесса, может потребоваться дополнительный анализ целого ряда контекстных и семантических проблем в отношении процесса, которые зависят от истории процессов, исключений и информации о внешних системах, а также от опыта дизайнеров.

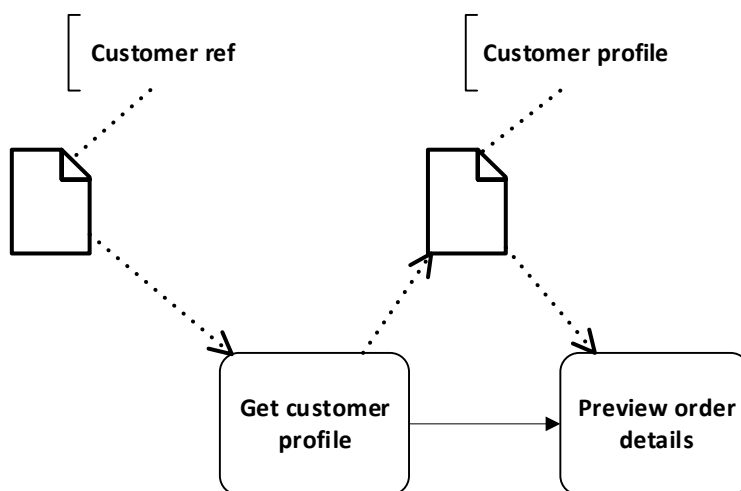
**Аномалии данных при работе с внешними системами.** Рассмотрим часть бизнес-процесса, в котором происходит взаимодействие с внешними системами (рис. 4):

- MDM-система управления данными заказчиков;
- Payment Gateway система оплаты заказов.

Бизнес-процесс описывает последовательность действий (задач) для получения информации о предпросмотре информации о заказе для пользователя, где:

- Customer ref – уникальный идентификатор заказчика;

- Customer profile – профиль заказчика;
- Get customer profile API – сервис получения профиля заказчика по уникальному идентификатору;
- Preview order details API – сервис получения информации для предпросмотра деталей заказа. Сервис требует предоставить в запросе некоторую информацию о заказчике, которая может быть извлечена из профиля заказчика.



**Рис. 4. Пример бизнес-процесса просмотра деталей заказа для заказчика**  
*Примечание: составлено авторами.*

В данном случае задачи «Get customer profile» и «Preview order detail» выполняют API-вызовы во внешних системах для получения необходимой информации:

- профиль заказчика (MDM-система);
- детали заказа (Payment Gateway система).

Исходя из того, что оба API реализованы в различных системах, не связанных друг с другом, следует вывод, что системы используют различные доменные модели. Опишем примеры ответа «Get customer profile API» и запроса «Preview order details API».

Пример Json-ответа от API «Get customer profile»:

```
{
  "customer": {
    "address": {
      "line1": "Perekopskaya",
      "line2": "15a",
      "state": "Tyumen",
      "city": "Tyumen",
      "country": "Russia"
    },
    "fullname": "Kabardinsky Evgeny"
  }
}
```

Пример Json-запроса в API «Preview order detail»:

```
{
  "line1": "Perekopskaya",
  "line2": "15a",
  "state": "Tyumen",
  "city": "Tyumen",
  "country": "Russia",
  "fullname": "Kabardinsky Evgeny"
}
```

Рассмотрим матрицу потоков данных для данного бизнес-процесса:

	V1 ("Get customer profile")	V2 ("Preview order detail")
d1 (customer.address.line1)	w	
d2 (customer.address.line2)	w	
d3 (customer.address.state)	w	
d4 (customer.address.city)	w	
d5 (customer.address.country)	w	
d6 (customer.fullname)	w	
d7 (line1)		r
d8 (line2)		r
d9 (state)		r
d10 (city)		r
d11 (country)		r
d12 (fullname)		r

Как мы видим из данной матрицы, задача V2 должна считать элементы данных, которые не были заполнены. В данном случае алгоритм, основанный на матрице потоков, будет ложно сигнализировать об аномалиях:

- отсутствия данных;
- избыточности данных.

Данный пример иллюстрирует то, что алгоритм, основанный на проверке матрицы потоков данных, не способен выявлять аномалии отсутствия и избыточности данных для процессов, которые работают с множеством доменных моделей. На практике достаточно часто появляется необходимость интеграции бизнес-процесса с внешними системами, что предполагает работу бизнес-процесса с различными доменными моделями. В таких случаях алгоритм валидации аномалий данных является неэффективным.

Проблема семантической валидации интегрируемых данных также была рассмотрена в работе [12], где было предложено ее решение с помощью онтологической модели концептуальных объектов.

Данная модель предоставляет правила описания концептуальных схем «сущность – связь» в виде аксиом и утверждений дескрипционной логики [12]:

- сущности представляются через набор терминологических аксиом;
- тип связи с отображением 1:n и 1:1 представляется через набор терминологических аксиом;
- атрибут представляется через набор терминологических аксиом.

Также в статье иллюстрируется эксперимент, показывающий достаточно высокую точность в определении несоответствия интегрируемых данных.

Онтологическая модель концептуальных объектов может быть расширена и дополнена для определения соответствия между доменными моделями различных систем.

**Заключение.** В данной статье были рассмотрены модели для моделирования бизнес-процессов: сети Петри, модели на основе действий, комбинированные модели, а также матри-

ца потоков данных. Проанализировав модели бизнес-процессов, можно сделать вывод, что на данный момент для определения аномалий отсутствия и избыточности данных используется матрица потоков данных, с помощью которой решается проблема формализации работы с входными/выходными наборами данных для задач в процессе. Основной проблемой данного подхода является невозможность работы матрицы потока со множеством доменных моделей, что делает методы поиска аномалий отсутствия и избыточности данных неэффективными.

Для решения данной проблемы предлагается выяснить возможность применения онтологической модели концептуальных объектов для идентификации семантических конфликтов между наборами данных из множества доменных моделей.

### **Литература**

1. Zo H., Nazareth D. L., Jain H. K. Service-oriented Application Composition with Evolutionary Heuristics and Multiple Criteria // Association for Computing Machinery (ACM) in ACM Transactions on Management Information Systems. 2019. Vol. 10, P. 1–28.
2. Aghabagherey R., Golpayegani A. H., Esmaeili L. A New Method for Organizational Process Model Discovery Through the Analysis of Workflows and Data Exchange Networks // Social Network Analysis and Mining. 2020. Vol. 10, No. 12. P. 1–15.
3. Rostami K., Heinrich R., Busch A., Reussner R. Architecture-Based Change Impact Analysis in Information Systems and Business Processes // IEEE International Conference on Software Architecture (ICSA), Gothenburg, 2017. P. 179–188.
4. Li X., Liu Y. A Formal Modeling Method for Workflow Based on Selection Logic // 18th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES), Wuhan, China, 2019. P. 116–119.
5. Sadiq S., Orlowska M. E., Sadiq W., Foulger C. Data Flow and Validation in Workflow Modelling // Conferences in Research and Practice in Information Technology. 2004. P. 1–8. URL: <https://crpit.scem.westernsydney.edu.au/confpapers/CRPITV27Sadiq.pdf> (дата обращения: 07.02.2020).
6. Sun S., Zhao L., Sheng O. Data Flow Modeling and Verification in Business Process Management // AMCIS 2004 Proceedings. P. 4064–4073.
7. Chadli N., Kabbaj M. I., Bakkoury Z. Detection of Dataflow Anomalies in Business Process An Overview of Modeling Approaches. // EasyChair Preprint No. 226. 2018. P. 1–6. URL: [https://easychair.org/publications/preprint\\_open/bVXK](https://easychair.org/publications/preprint_open/bVXK) (дата обращения: 07.03.2020).
8. Petri C. Communication with Automata. DTIC Research Report AD0630125, Defense Technical Information Centre. 1966. 97 p.
9. Сети Петри – математический аппарат для моделирования. URL: [http://bourabai.kz/cm/petri\\_nets.htm/](http://bourabai.kz/cm/petri_nets.htm/) (дата обращения: 07.03.2020).
10. Stackelberg S., Putze S., Mülle J., Böhm K. Detecting Data-Flow Errors in BPMN 2.0 // Open Journal of Information Systems (OJIS). Vol. 1. P. 1–19.
11. Астанин С. В., Драгныш Н. В., Жуковская Н. К. Вложенные метаграфы как модели сложных объектов // Инженер. вест. Дона. 2012. № 4 (часть 2). URL: <http://ivdon.ru/magazine/archive/n4p2y2012/1434> (дата обращения: 07.03.2020).
12. Кропотин А. А., Бидуля Ю. В., Ивашко А. Г., Самойлов М. Ю. Онтологический метод проверки семантической несогласованности реляционных баз данных и официальных документов // Вестн. Тюмен. гос. ун-та. Физико-математическое моделирование. Нефть, газ, энергетика. 2018. Т. 4, № 3. С. 120–131.