

## Impact Factor:

ISRA (India) = 3.117  
ISI (Dubai, UAE) = 0.829  
GIF (Australia) = 0.564  
JIF = 1.500

SIS (USA) = 0.912  
PIHII (Russia) = 0.156  
ESJI (KZ) = 8.716  
SJIF (Morocco) = 5.667

ICV (Poland) = 6.630  
PIF (India) = 1.940  
IBI (India) = 4.260  
OAJI (USA) = 0.350

SOI: [1.1/TAS](#) DOI: [10.15863/TAS](#)

## International Scientific Journal Theoretical & Applied Science

p-ISSN: 2308-4944 (print) e-ISSN: 2409-0085 (online)

Year: 2019 Issue: 05 Volume: 73

Published: 27.05.2019 <http://T-Science.org>

QR – Issue



QR – Article



Oleg Yurievich Sabinin

Candidate of Engineering Sciences, Associate Professor  
Peter the Great St. Petersburg Polytechnic University  
[olegsabinin@mail.ru](mailto:olegsabinin@mail.ru)

Mikhail Vladimirovich Toporov  
student

Peter the Great St. Petersburg Polytechnic University  
[tmv1995@gmail.com](mailto:tmv1995@gmail.com)

SECTION 4. Computer science, computer  
engineering and automation.  
UDC 004.

## IDENTIFICATION AND VALIDATION OF BUSINESS PROCESSES USING PROCESS MINING ALGORITHMS AND A DISTRIBUTED LEDGER TECHNOLOGY ALGORITHMS

**Abstract:** This article discusses the possibility of validating the results of the work of the process mining algorithms using consensus algorithms of distributed ledger technology.

**Key words:** Process Mining, Consensus Algorithms, Alpha.

**Language:** Russian

**Citation:** Sabinin, O. Y., & Toporov, M. V. (2019). Identification and validation of business processes using process mining algorithms and a distributed ledger technology algorithms. *ISJ Theoretical & Applied Science*, 05 (73), 344-349.

**Soi:** <http://s-o-i.org/1.1/TAS-05-73-50> **Doi:**  <https://dx.doi.org/10.15863/TAS.2019.05.73.50>

### ИДЕНТИФИКАЦИЯ И ВАЛИДАЦИЯ БИЗНЕС-ПРОЦЕССОВ ПРИ ПОМОЩИ АЛГОРИТМОВ PROCESS MINING И РАСПРЕДЕЛЕННОГО РЕЕСТРА

**Аннотация:** В данной статье рассматривается возможность удостоверения результатов работы алгоритмов process mining благодаря использованию алгоритмов консенсуса технологии распределенного реестра.

**Ключевые слова:** Process Mining, Алгоритмы консенсуса, Alpha.

#### 1 Introduction

Большая часть программных продуктов, представленных на рынке на данный момент, не удовлетворяют потребностям крупных ритейлеров, вследствие чего каждая компания имеет свой IT отдел, занимающийся разработкой программного обеспечения (ПО) для собственных нужд.

С ростом компании растет и потребность в автоматизации многих процессов. Чтобы оставаться конкурентно способными на рынке, IT отделы сильно разрастаются, и из ранга интернациональных переходят уже в межнациональные, а все задачи начинают разделять между отдельными командами разработчиков.

В виду данного разделения труда при проектировании сложных систем становится трудно удерживать все знания в актуальном состоянии. Как правило, это приводит к отсутствию понимания того, как на самом деле начинают в системе происходить бизнес-процессы.

Восстановлением и актуализацией знаний по бизнес-процессам занимаются бизнес-аналитики, которые тратят на это достаточно много времени, не гарантируя при этом достоверного результата, что существенно затрудняет работу компании.

В данной статье будет описан практический опыт применения одного из алгоритмов process mining- подхода по получению знаний о бизнес-процессах из машинных данных, который из системных журналов каждой подсистемы,

## Impact Factor:

ISRA (India) = 3.117	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	ПИИЦ (Russia) = 0.156	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 8.716	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 5.667	OAJI (USA) = 0.350

собранных в единой базе данных, извлекает бизнес-процесс по согласованному изменению цены в магазинах внутри крупного ритейлера.

Данная категория алгоритмов отлично подходит для решения вышеупомянутой проблемы, поскольку с их помощью выстраиваются автоматизированные системы, способные существенно облегчить работу аналитикам. [1]

Однако алгоритмы process mining не могут гарантировать стопроцентного результата. Результат их работы сильно зависит от входных данных, выбора алгоритма, а также фильтрации конечных результатов. Поэтому помимо алгоритмов process mining также требуется и алгоритм для валидации полученных результатов, пример которого также будет описан в этой статье.

## 2 Identification of business-process with process mining algorithm

Предположим, что есть единое хранилище всех журналов абсолютно всех подсистем. Как

правило, в крупных компаниях агрегированным сбором и первичной аналитикой занимаются продукты наподобие Splunk (система автоматизированного сбора машинных данных[2]), однако в данной статье будут использоваться измененные данные действующей продуктивной системы, поэтому для их хранения и последующего тестирования было принято решение видоизменить часть данных и сохранить их в базе данных Oracle.

Если представлять себе бизнес-процесс по согласованному изменению цен в магазинах, то при условии использования решений продуктов Oracle Retail (Приложения от компании Oracle для крупных магазинов[3]) в упрощенном виде его можно изобразить следующей диаграммой, представленной на рисунке 1, где в блоках описывается производимое действие, а рядом с блоком то, что считывается из машинных данных:

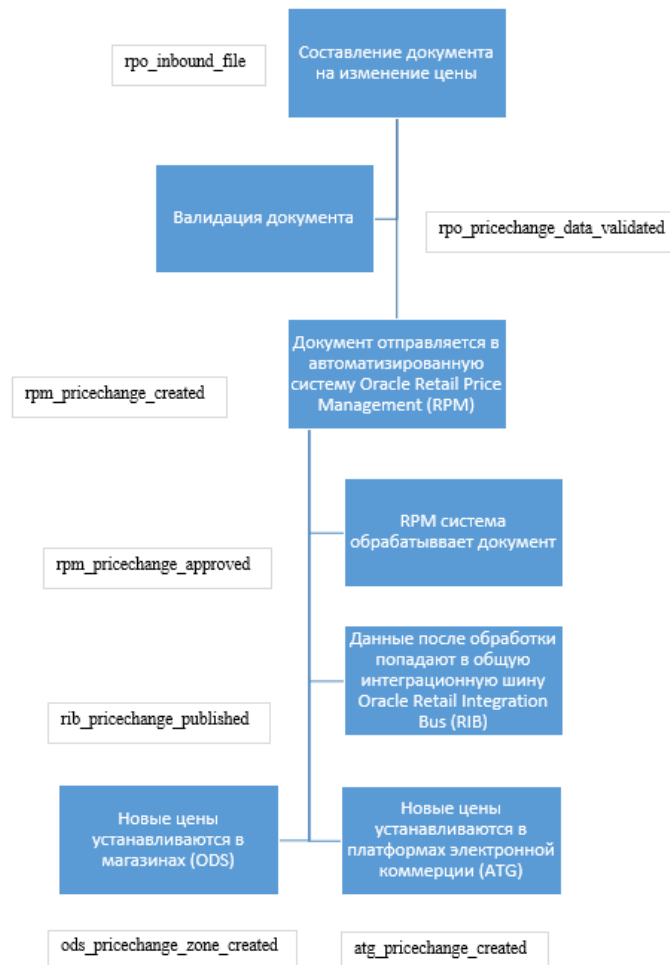


Рисунок 1 - Упрощенная схема бизнес-процесса изменения цен в магазинах

## Impact Factor:

ISRA (India) = 3.117  
ISI (Dubai, UAE) = 0.829  
GIF (Australia) = 0.564  
JIF = 1.500

SIS (USA) = 0.912  
ПИИЦ (Russia) = 0.156  
ESJI (KZ) = 8.716  
SJIF (Morocco) = 5.667

ICV (Poland) = 6.630  
PIF (India) = 1.940  
IBI (India) = 4.260  
OAJI (USA) = 0.350

Как уже говорилось ранее, первая часть задачи – найти бизнес- процесс среди записей в базе данных. Каждая запись содержит структурированные машинные данные и имеет следующий вид:

```
{
  "id": "841fd6ae-e7e6-5c0d-e055-b530523b1a",
  "agentId": "NBSBatchHostFA",
  "trackingPointId": "rpo_inbound_file",
  "createdAt": "2019-03-15T09:41:10.353Z",
  "objectType": "PriceChangeEvent",
  "correlationIds": [
    "004542584-1-23/03/2019",
    "15-03-2019"
  ],
  "payload": {
    "item": "004542584",
    "active_date": "23/03/2019",
    "channel": "COMMON",
    "change_amount": 42.54,
    "user_id": "alain.frecon"
  }
}
```

Для нахождения бизнес-процесса был выбран алгоритм «alpha»[4], суть которого можно описать следующим образом. На вход алгоритму подаются данные в формате кейс-событие. Представим, что каждая из подсистем имеет свой уникальный идентификатор от 1 до 7, тогда последовательность подсистем, обозначающая экземпляр бизнес-процесса, будет иметь следующий вид: 1, 2, 3, 4, 5, (6 | 7). Последняя пара чисел была взята в скобки ввиду того, что в точке 5 бизнес-процесс разветвляется. Подобная последовательность чисел называется кейсом, а каждый элемент этого кейса – определенное событие, произошедшее в определенный момент времени. [5,6]

Таким образом, извлекая кейсы из журналов подсистем можно получить примерно следующий результат экземпляров работы бизнес-процесса:

1 2 3 4 5 6 7 – процесс отработал и завершился успешно;

1 2 3 4 – процесс остановился из-за ошибки в системе;

4 5 6 – данные мониторинга были потеряны.

Как можно видеть, машинные данные, хоть и структурированные, достаточно сложны даже в зрительной интерпретации. Более того, нет никакой обработки на ветвление бизнес-процессов, поэтому эту задачу также приходится решать алгоритму «alpha».

Как только алгоритм получает на вход данные, первым делом идет составление словаря связанных точек по всем кейсам. В данный словарь попадают уникальные пары точек, которые идут друг за другом как минимум в одном из кейсов. Далее алгоритм составляет еще 3 дополнительных словаря:

1. Словарь причинных связей – в него попадают те пары точек из словаря связанных, которые имеют прямую следственную связь, например, это пара 1 -> 2. Если бы в словаре связанных точек имелись пары 1 -> 2 и 2 -> 1, то ни одна из них в словарь причинных связей не попадает.

2. Словарь несвязных точек – как следует из названия, сюда попадают те точки, которые не имеют пары в словаре причинных связей. Как правило точки сюда попадают по причине использования фильтрации.

3. Словарь циклов – это словарь, в который попадают пары взаимосвязанных точек, как, например, 1 -> 2 и 2 -> 1.

Как только все точки были распределены по словарям, начинается следующий этап – выстраивание пути. Для этого составляются списки начальных и конечных точек, а затем строится словарь всех возможных последовательностей из всех значений точек, которые были найдены в кейсах. Далее алгоритм выбирает две последовательности из этого словаря и для каждой точек из каждой последовательности проверяет, не находятся ли точки в словаре несвязных точек. Если точки последовательности находятся в этом словаре, то это означает, что все точки этой последовательности никак не связаны между собой. Как только две последовательности проверены, выполняется финальная проверка – из обеих последовательностей берутся всевозможные пары всех точек и проверяются на наличие в словаре причинных связей, если найдена хоть одна пара, которая не имеет связи между собой, то две последовательности считаются несвязными между собой, в противном случае обе последовательности помечаются как возможные подпоследовательности искомого бизнес-процесса.

Финальный этап работы алгоритма заключается в обработке результата, полученного на предпоследнем этапе. Для этого берется пара последовательностей, которые были помечены как возможные подпоследовательности искомого бизнес-процесса, и проверяются на то, не является ли одна из них подпоследовательностью другой. Если данный факт был выявлен, то та последовательность, которая является подпоследовательностью для другой, исключается из результата. После этой проверки все оставшиеся последовательности собираются в одну, что и является искомым бизнес-процессом.

Результат работы описанного алгоритма, которому на вход подавались реальные данные, можно видеть на рисунке 2 (расшифровку названий подсистем см. на рисунке 1):

## Impact Factor:

ISRA (India) = 3.117	SIS (USA) = 0.912	ICV (Poland) = 6.630
ISI (Dubai, UAE) = 0.829	PIHИ (Russia) = 0.156	PIF (India) = 1.940
GIF (Australia) = 0.564	ESJI (KZ) = 8.716	IBI (India) = 4.260
JIF = 1.500	SJIF (Morocco) = 5.667	OAJI (USA) = 0.350

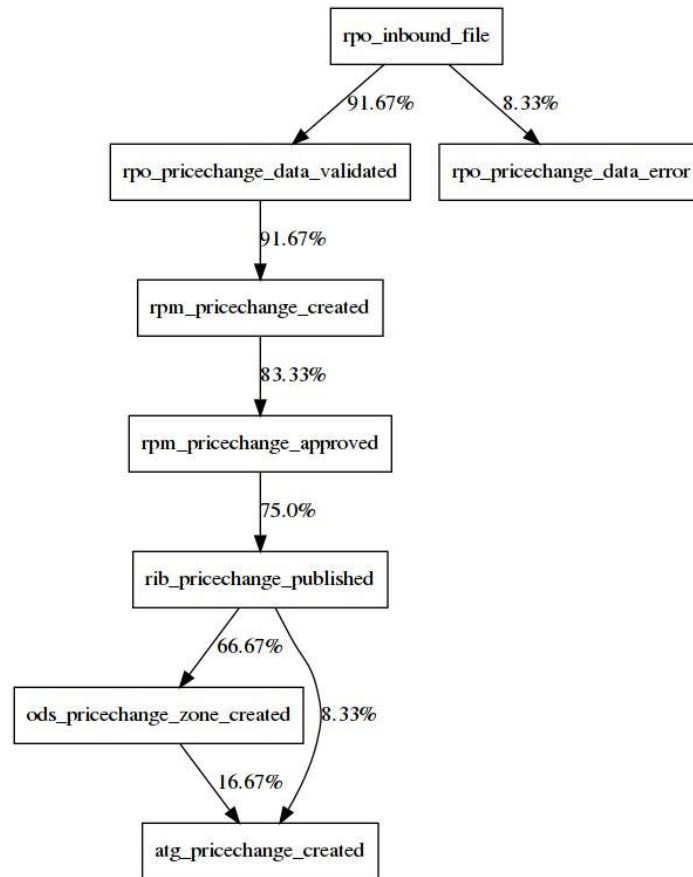


Рисунок 2 - Результат работы реализованного алгоритма

Проценты на стрелках означают то, с какой частотой связь двух соседних точек встречалась в момент обработки всех кейсов. Как можно видеть, алгоритм не только справился с задачей, но также выявил и упущенный процесс – `rpo_pricechange_data_error`, который является не задокументированной модификацией системы, и который также существенно влияет на то, будет ли новая цена применена в магазинах.

### 3 Validation of the found process

Как уже говорилось ранее, результат работы бизнес-аналитиков очень далек от гарантированного. Иными словами, результат достоверен лишь с некоторой статистической вероятностью, а значит, не может быть использован бизнесом для принятия каких-либо решений в виду своей ненадежности. Таким образом поднимается вопрос о том, как валидировать результат работы алгоритмов `process mining`, чтобы затем использовать для принятия важных стратегических решений? Ведь результат автономной работы алгоритма также подлежит сомнению, поскольку на него сильно влияют входные данные, выбор алгоритма и способы фильтрации. [4]

В данном направлении также активно ведутся исследования, но ни одно из них не

использует в своей основе алгоритмы консенсуса. Поиск информации на предмет подобных имплементаций не принес результатов, однако же очевидно, что если каждая из точек, через которую предположительно прошел бизнес-процесс, подтвердит, что факт его прохождения через эту подсистему, то возможно будет получить достоверную картину работы бизнес-процесса путем запуска голосования среди точек, через которые он предположительно прошел. В тоже время, каждая точка имеет все необходимые данные для того, чтобы однозначно утверждать, проходил ли через нее бизнес-процесс. Итогом голосования в данном случае станет достижение консенсуса относительно того, какие точки участвуют в бизнес-процессе. [7,8]

Технических решения подобного голосования в распределенной системе достаточно много, и все они объединены в одну группу – алгоритмы для решения задачи консенсуса. Данные алгоритмы можно встретить как в блокчейн, так и в распределенном реестре. В этой статье будет рассмотрен алгоритм распределенного реестра `Federated Byzantine Agreement`. [9]

В данном алгоритме ключевая идея состоит в том, чтобы организовать голосование таким образом, что участники голосуют внутри

## Impact Factor:

ISRA (India) = 3.117  
ISI (Dubai, UAE) = 0.829  
GIF (Australia) = 0.564  
JIF = 1.500

SIS (USA) = 0.912  
ПИИЦ (Russia) = 0.156  
ESJI (KZ) = 8.716  
SJIF (Morocco) = 5.667

ICV (Poland) = 6.630  
PIF (India) = 1.940  
IBI (India) = 4.260  
OAJI (USA) = 0.350

доверенных групп, и только затем уже группы голосуют между собой. В случае Federated Byzantine Agreement каждый участник может решить, кому доверять, и стать частью доверенной группы принятия решений, которую можно назвать кворумным срезом.

Общее соглашение достигается в данном алгоритме за счет того, что данные кворумные срезы, или же доверенные группы, специально пересекаются друг с другом. Таким образом, при определенном соотношении кворумных срезов к общему числу участников алгоритм гарантирует, что решение будет принято всегда. [9,10]

Стоит отметить, что поскольку каждая точка может однозначно сказать, проходил ли через нее бизнес-процесс, то используя данный алгоритм не сложно будет запустить голосование для валидации работы алгоритма process mining из предыдущей части статьи. Однако, подтверждения того, что процесс прошел через точку недостаточно. Поэтому следует объединять подпоследовательности, которые мы находили в предыдущей части статьи, в кворумные срезы. Таким образом, для каждого участника голосования будет ставиться достаточно комплексный вопрос: От кого пришел бизнес-процесс, и куда он был отправлен? Ответ на данный вопрос не только однозначно определяет, проходил ли через точку бизнес-процесс, но также и определяет соседей этой точки: вышеследующую и нижеследующую подсистему.

В результате такого голосования можно получить достоверную картину того, как именно точки, участвовавшие в бизнес-процессе, связаны между собой. Иными словами, в результате опроса мы получим результирующую последовательность всех подсистем, через которые прошел бизнес-процесс и которую можно сопоставить с результатом работы process mining.

Данный алгоритм не может быть использован для самостоятельного поиска бизнес-процесса, поскольку такая имплементация

практически невозможна и потребует переработки всех подсистем. Однако же, использовать агентов для голосования в каждой из подсистем достаточно просто и не требует вмешательства в инфраструктуру компании, и в тоже время сам агент не будет иметь какой-либо сложной логики или конфигурации внутри себя, а просто будет отсылать статистику по мониторингу за настроенными интерфейсами. Более того, подобные агенты уже фактически присутствуют в каждой из компаний – это те агенты мониторинговых систем, как, например, Splunk, которые занимаются сбором и отсылкой машинных данных.

Таким образом, заранее зная, кого нужно опрашивать, будет очень просто собрать информацию в системе мониторинга лишь по тем подсистемам, которые участвуют в бизнес-процессе. Если подходить к вопросу с этой стороны, то не требуется сложная модификация на стороне каждой из подсистем. Достаточно лишь организовать несложную логику внутри системы мониторинга, которая, принимая на вход последовательность идентификаторов подсистем, будет устанавливать взаимосвязь между ними ранее описанным способом.

## 4 Conclusion

Результат работы алгоритмов process mining зачастую не идеален и содержит неточности. В первую очередь это связано с тем, что на результат очень сильно влияют входные данные и способы их фильтрации. Однако же, если подходить к вопросу о нахождении бизнес-процесса среди машинных данных более комплексно, то при использовании сторонних методов можно достичь практически стопроцентного результата. Как описывалось в данной статье, решения могут быть весьма специфическими, однако же полученные результаты говорят о том, что они имеют право на существование.

## References:

1. R'bigui, H., & Cho, C. (2017). The state-of-the-art of business process mining challenges // *Int. J. Bus. Process Integr. Manag.* Vol. 8, № 4, p. 285.
2. (n.d.). *Splunk* [Electronic resource]. Retrieved 2019, from <https://www.splunk.com/>
3. (n.d.). *Oracle Retail Products* [Electronic resource]. Retrieved 2019, from <https://www.oracle.com/industries/retail/>
4. Wang, J., et al. (n.d.). *Efficient Selection of Process Mining Algorithms*.
5. Medeiros, et al. (n.d.). *Process Mining: Extending the  $\alpha$ -algorithm to Mine Short Loops*

<b>Impact Factor:</b>	<b>ISRA (India) = 3.117</b>	<b>SIS (USA) = 0.912</b>	<b>ICV (Poland) = 6.630</b>
	<b>ISI (Dubai, UAE) = 0.829</b>	<b>PIHHI (Russia) = 0.156</b>	<b>PIF (India) = 1.940</b>
	<b>GIF (Australia) = 0.564</b>	<b>ESJI (KZ) = 8.716</b>	<b>IBI (India) = 4.260</b>
	<b>JIF = 1.500</b>	<b>SJIF (Morocco) = 5.667</b>	<b>OAJI (USA) = 0.350</b>

---

- // Dep. Technol. Manag. Eindhoven Univ. Technol.
6. Porouhan, P., Jongsawat, N., & Wichian, P. (2014). *Process and deviation exploration through Alpha-algorithm and Heuristic miner techniques* // 2014 Twelfth Int. Conf. ICT Knowl. Eng.
  7. Reed, B., & Junqueira, F. P. (2008). *A simple totally ordered broadcast protocol* // Proceedings of the 2nd Workshop on Large-Scale Distributed Systems and Middleware - LADIS '08, p. 1.
  8. Lamport, L., et al. (2004). *In Search of an Understandable Consensus Algorithm* // Proc. 2014 USENIX Annu. Tech. Conf.
  9. Mazieres, D. (2016). *Stellar Consensus Protocol*. p.32.
  10. Garcia-Perez, A., & Gotsman, A. (2016). *Federated Byzantine Quorum Systems* // IMDEA Softw. Institute, Madrid, Spain.