

# **МОДЕЛИ, МЕТОДЫ И АЛГОРИТМЫ ОБРАБОТКИ ПОТОКОВ ДАННЫХ В ТУМАННЫХ ВЫЧИСЛИТЕЛЬНЫХ СРЕДАХ**

05.13.11 - математическое и программное обеспечение  
вычислительных машин, комплексов и компьютерных сетей

Диссертация на соискание ученой степени кандидата физико-математических наук

**Амир Басим Абдуламир Алаасам**

Научный руководитель:  
РАДЧЕНКО Глеб Игоревич,  
кандидат физ.-мат. наук, доцент

Исследование выполнено при финансовой поддержке РФФИ в рамках научных проектов № 19-37-90073 и № 18-07-01224 и при финансовой поддержке Министерства науки и высшего образования РФ (FENU-2020-0022)

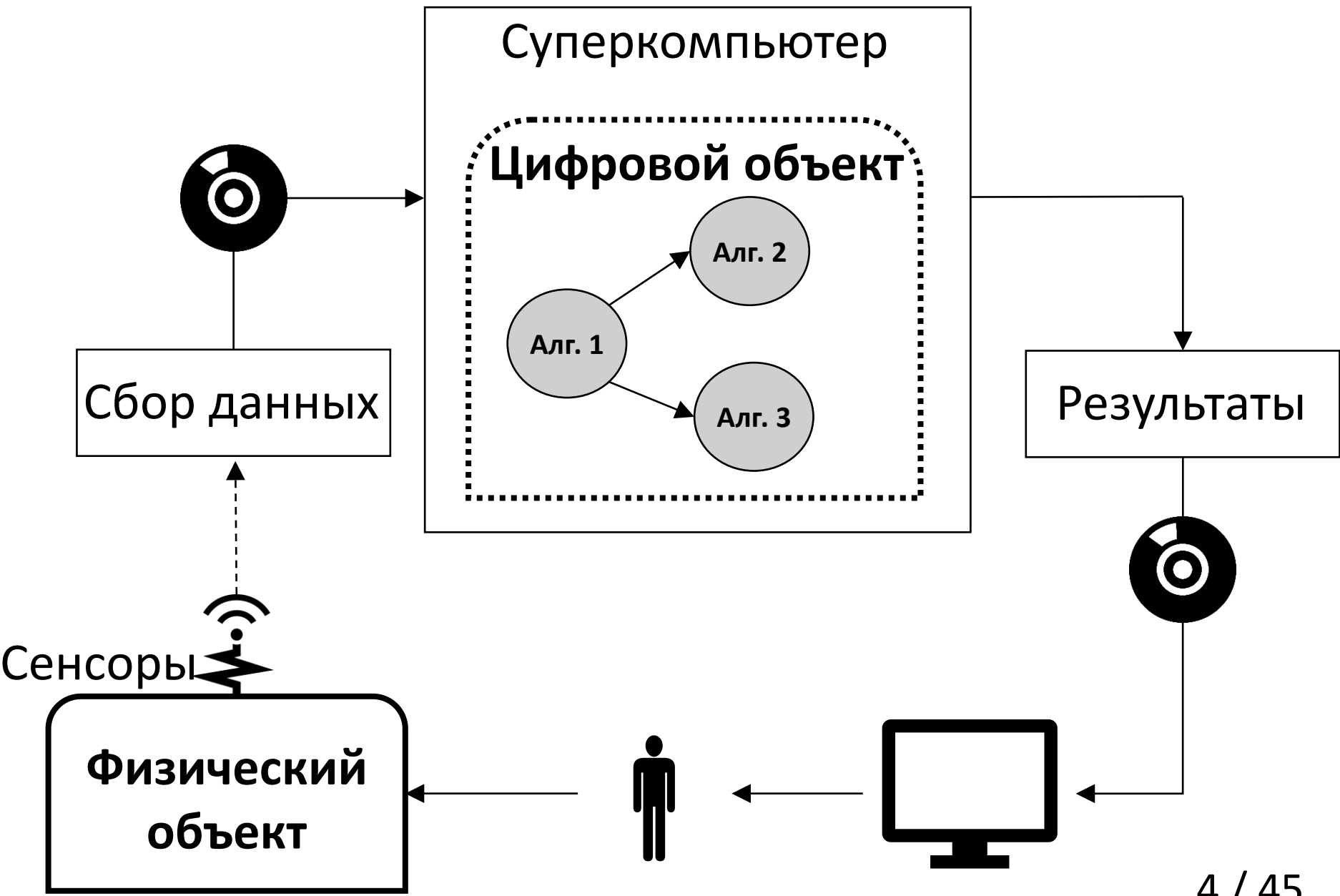
# Цель диссертационной работы

Разработка новой концепции организации потоков работ, включая математическую модель, методы и алгоритмы, позволяющей организовать эффективную обработку потоков данных в облачных и туманных вычислительных средах

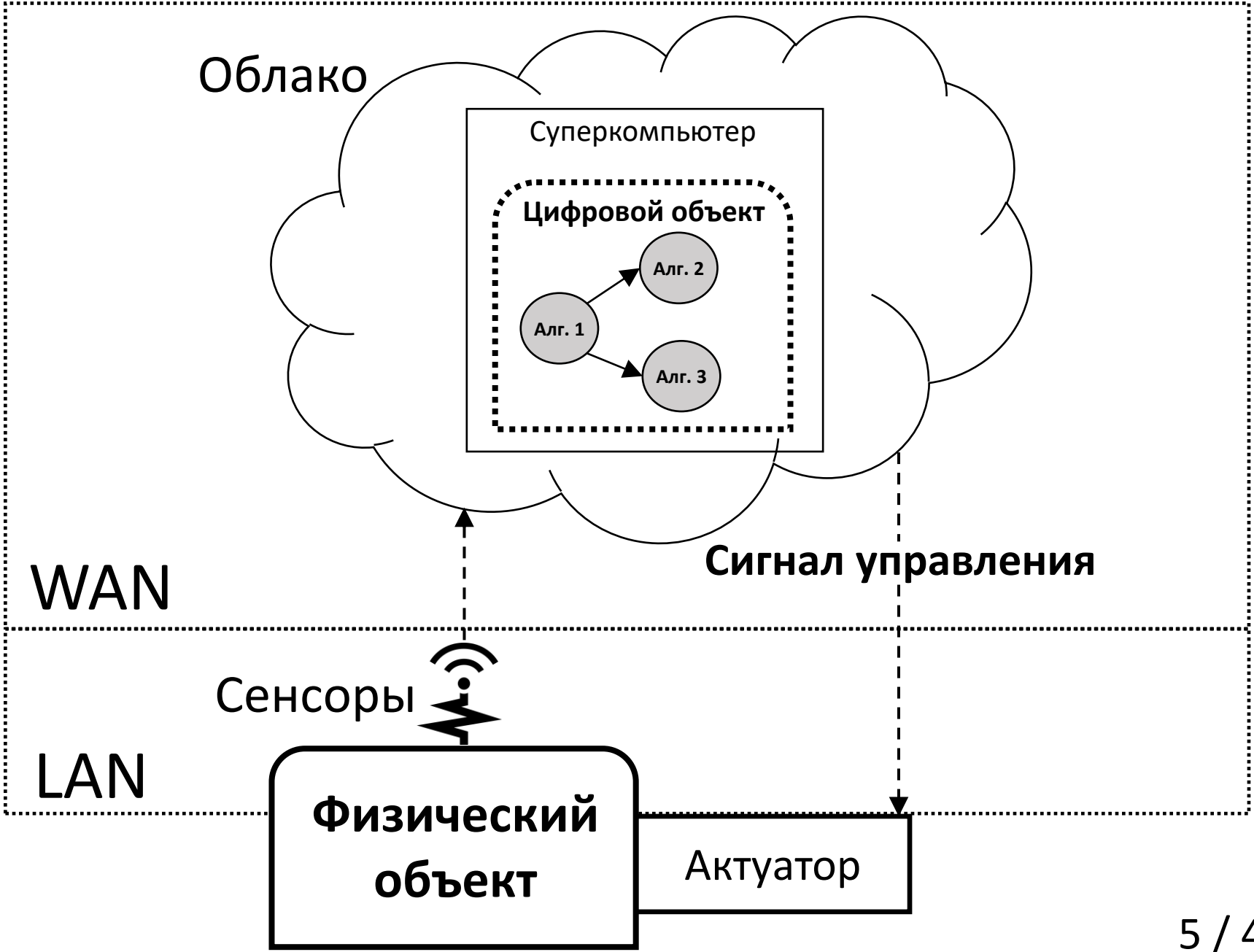
# Ключевые понятия диссертации на примере индустриального цифрового двойника

**Индустриальный цифровой двойник** - цифровая копия физического индустриального объекта или процесса

# Цифровой двойник до появления сети Интернет



# Цифровой двойник с использованием облачных вычислений



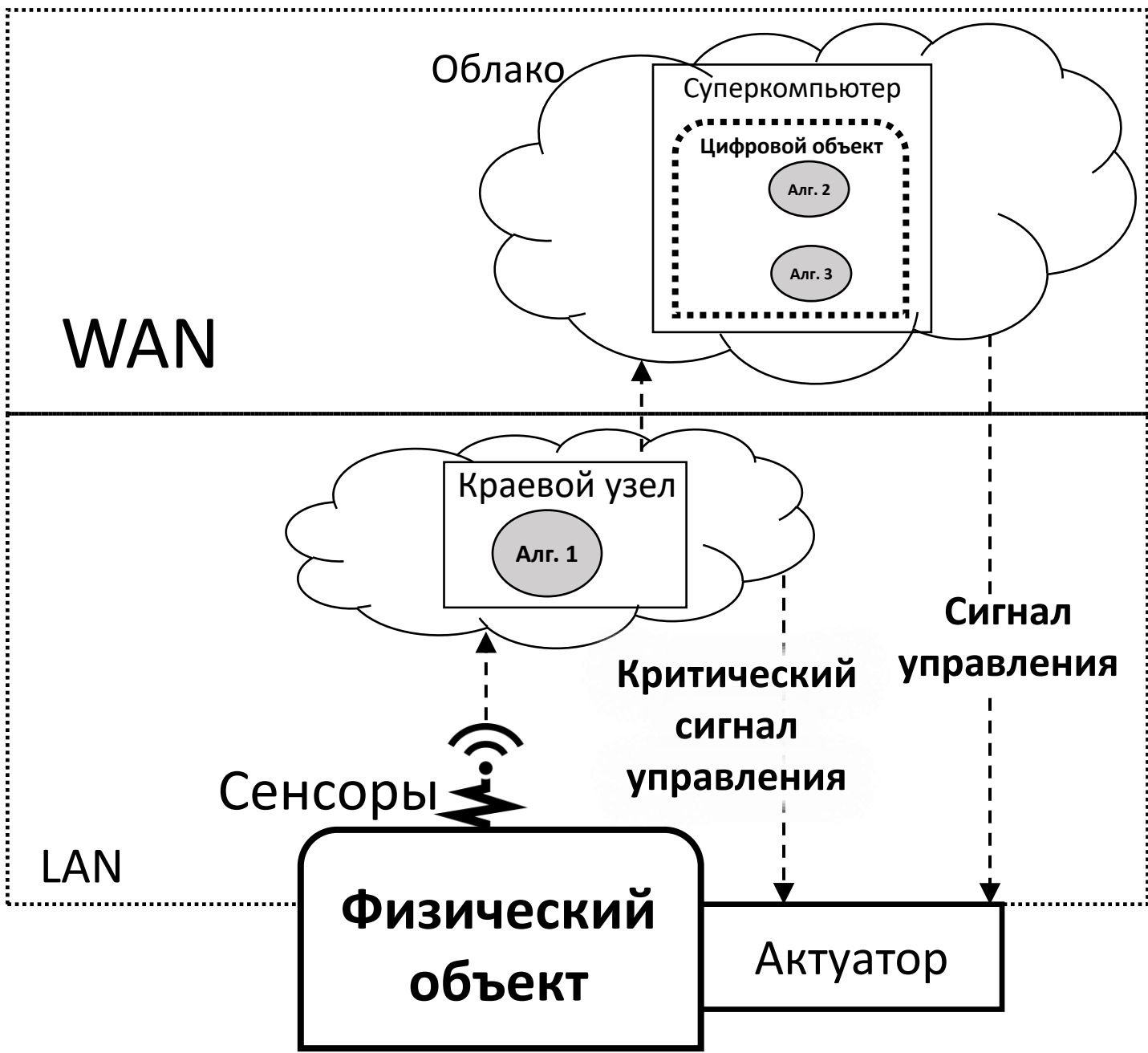
# Определение облачных вычислений

**Облака** представляют собой большой пул глобально-доступных и удобных в использовании виртуализированных ресурсов.

- Эти ресурсы могут быть динамически реконфигурированы для адаптации к переменной нагрузке
- Этот пул ресурсов обычно использует модель оплаты по мере использования

Vaquero L.M. et al. A break in the clouds: towards a cloud definition // ACM SIGCOMM Computer Communication Review. 2008. Vol. 39, no. 1. P. 50–55.  
DOI:10.1145/1496091.1496100

# Цифровой двойник с использованием туманных вычислений



# Определение туманных вычислений

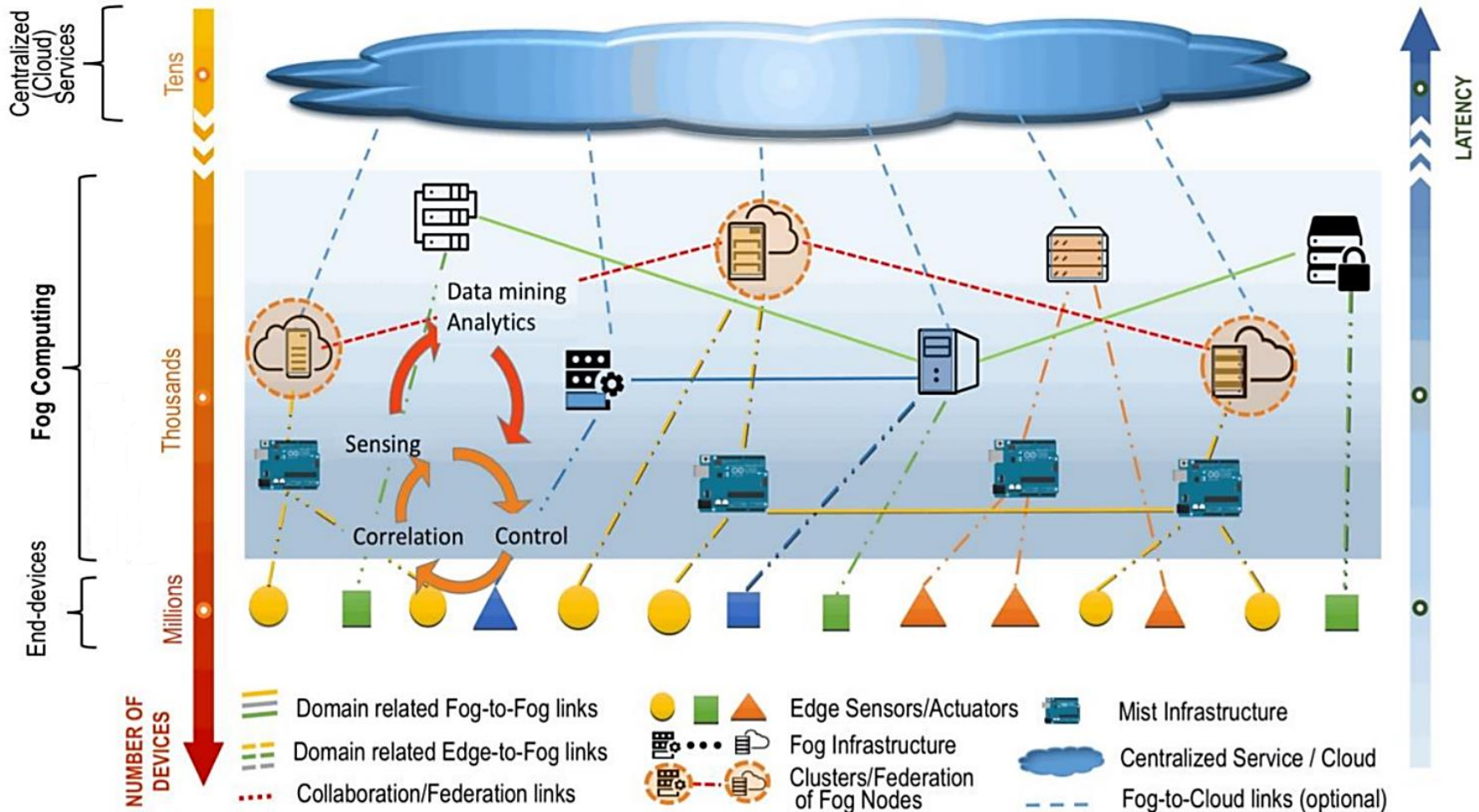
**Туманные вычисления** — это многоуровневая модель, поддерживающая развертывание распределенных приложений и сервисов с учетом латентности

- Туманная вычислительная среда состоит из туманных (краевых) узлов, расположенных между конечными устройствами и централизованными (облачными) сервисами

Iorga M. et al. Fog computing conceptual model. Gaithersburg, MD, 2018.  
DOI:10.6028/NIST.SP.500-325



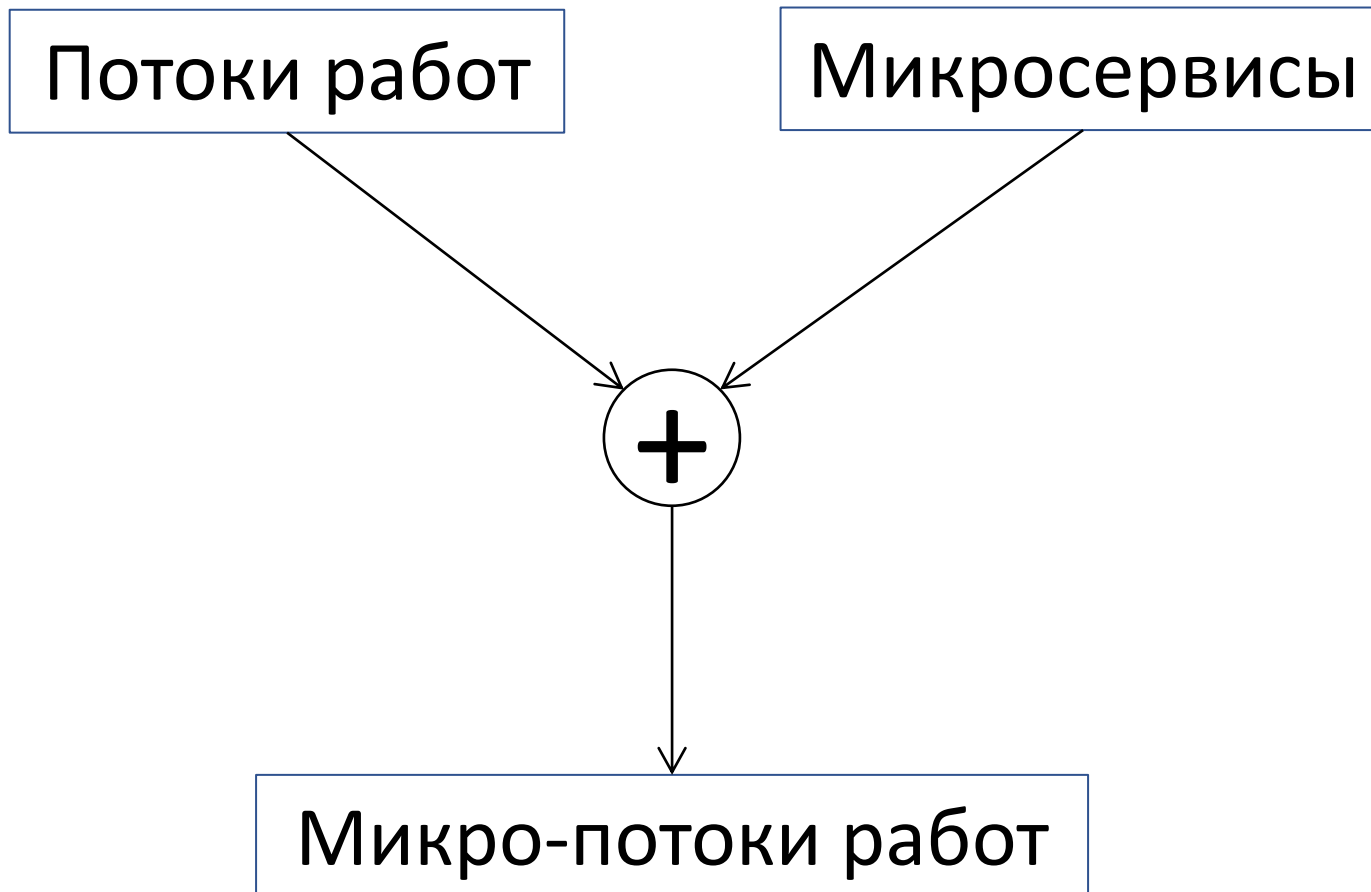
# Туманные вычисления уменьшают латентность и увеличивают количество задействованных устройств



Iorga M. et al. Fog computing conceptual model. Gaithersburg, MD, 2018.

DOI:10.6028/NIST.SP.500-325

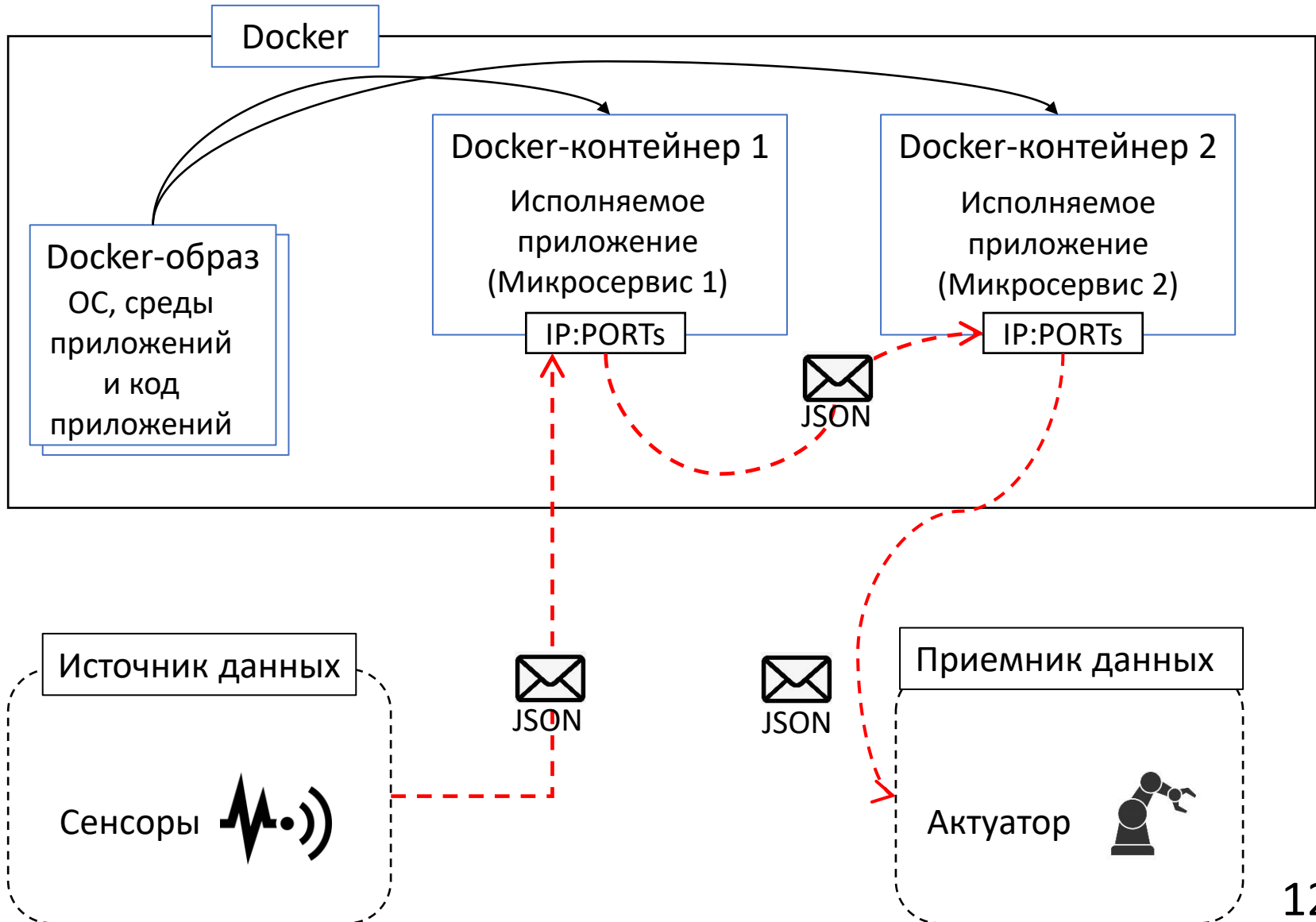
# Концепция микро-потоков работ, предложенная в диссертации



# Микросервисная архитектура

**Микросервисная архитектура** — вариант сервис-ориентированной архитектуры программного обеспечения, направленный на взаимодействие небольших, слабо связанных и легко изменяемых модулей — микросервисов, каждый из которых отвечает за выполнение определенной функции

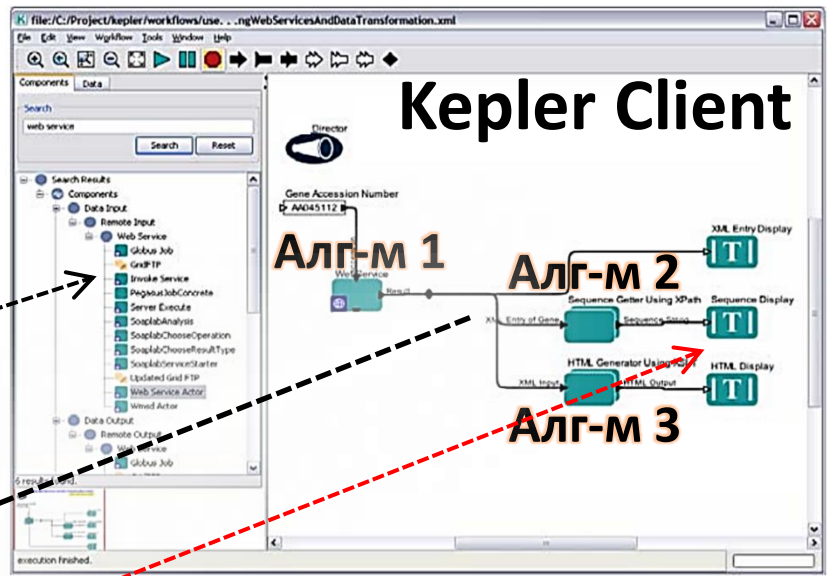
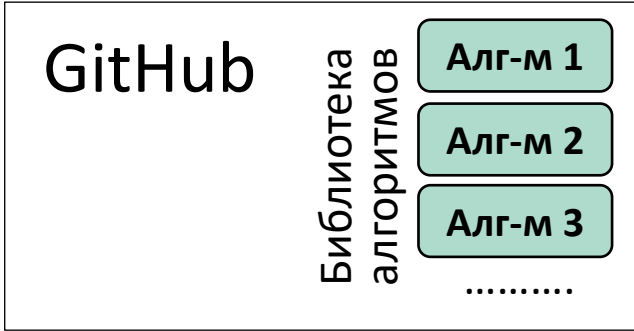
# Реализация микросервисов на примере PaaS «Docker»



# Система управления вычислительными потоками работ

- Система управления вычислительными потоками работ - это программная система, обеспечивающая представление задачи в виде ориентированного графа, в узлах которого находятся алгоритмы, экспортируемые из библиотек программ, и автоматическое выполнение этого представления в распределенной вычислительной среде
- Системы управления потоков работ не поддерживают микросервисы

# Поток работ (на примере системы Кеплер)



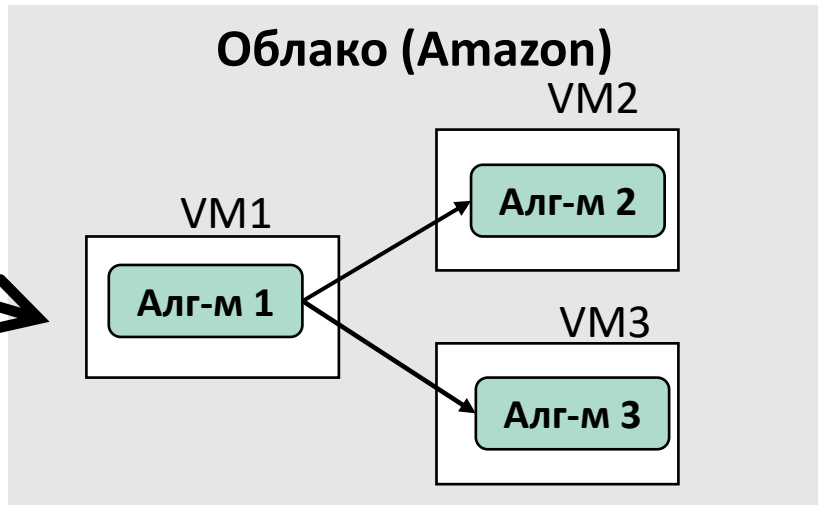
Алгоритмы

Абстрактный поток работ

Результаты

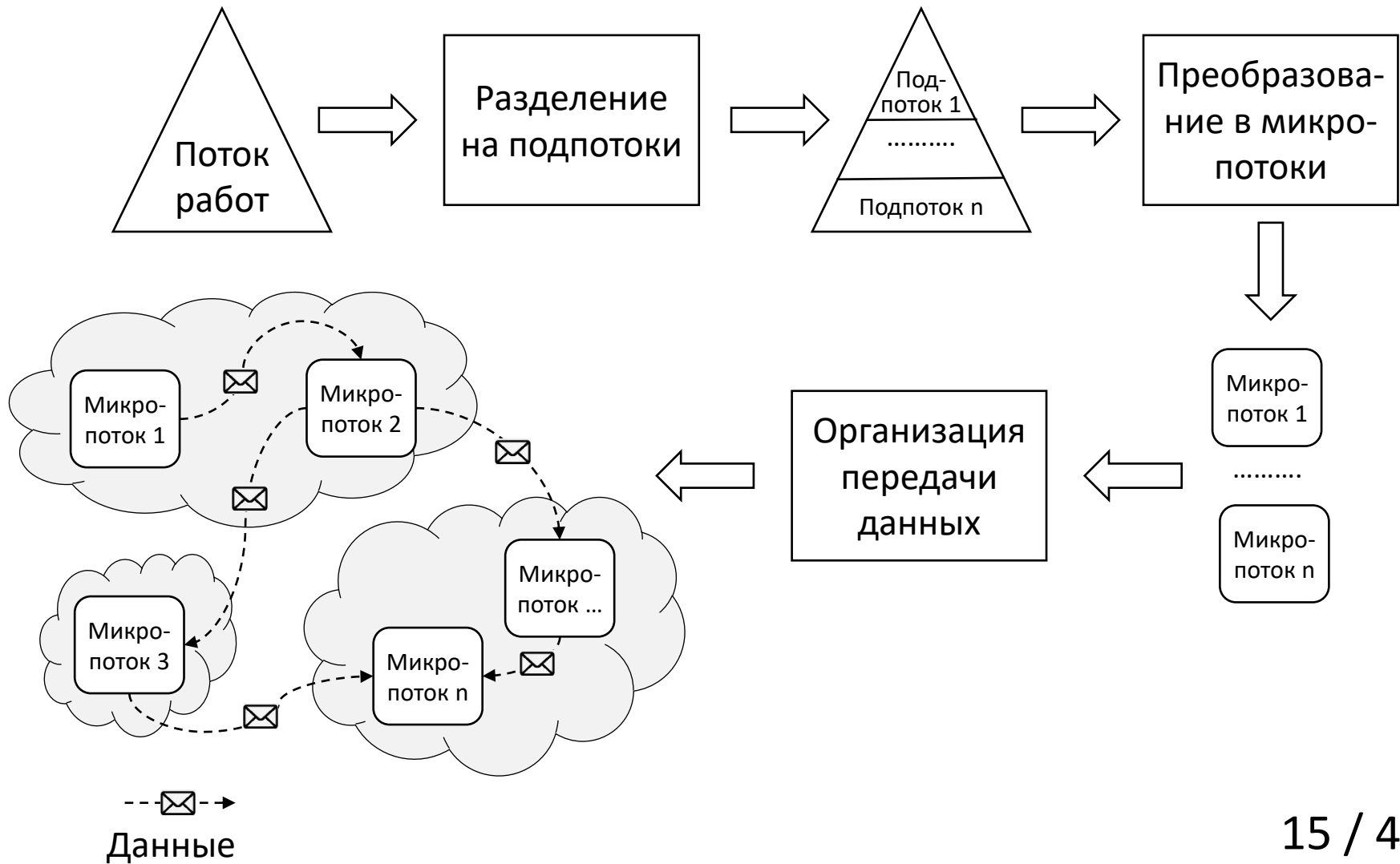
Результаты

Исполняемый поток работ

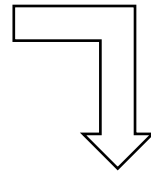
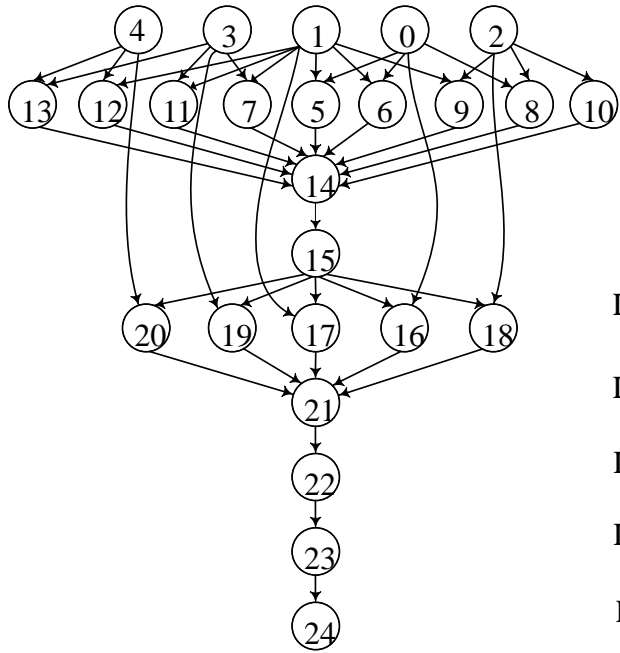


Wang J., Altintas I. Early cloud experiences with the Kepler scientific workflow system // Procedia Computer Science. Elsevier B.V., Vol. 9. P. 1630–1634. DOI:10.1016/j.procs.2012.04.179.

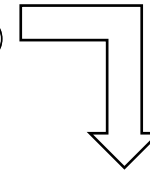
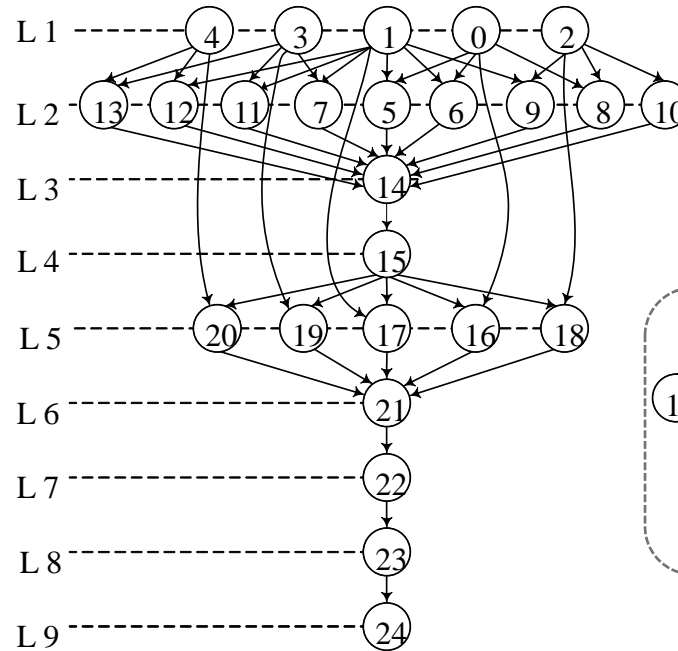
# Организации системы микро-потоков работ, предложенная в диссертации



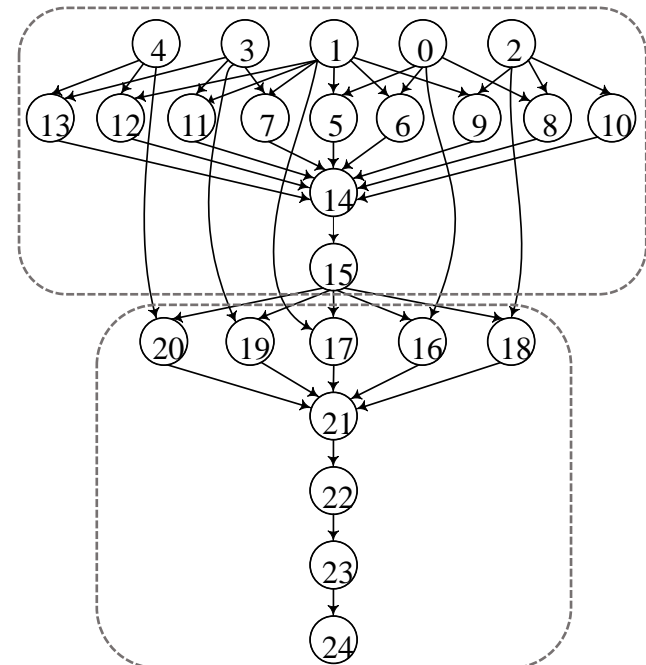
# Разделение на подпотоки



Разделение на уровни  
(Level-based partitioning)



Разделение на подпотоки

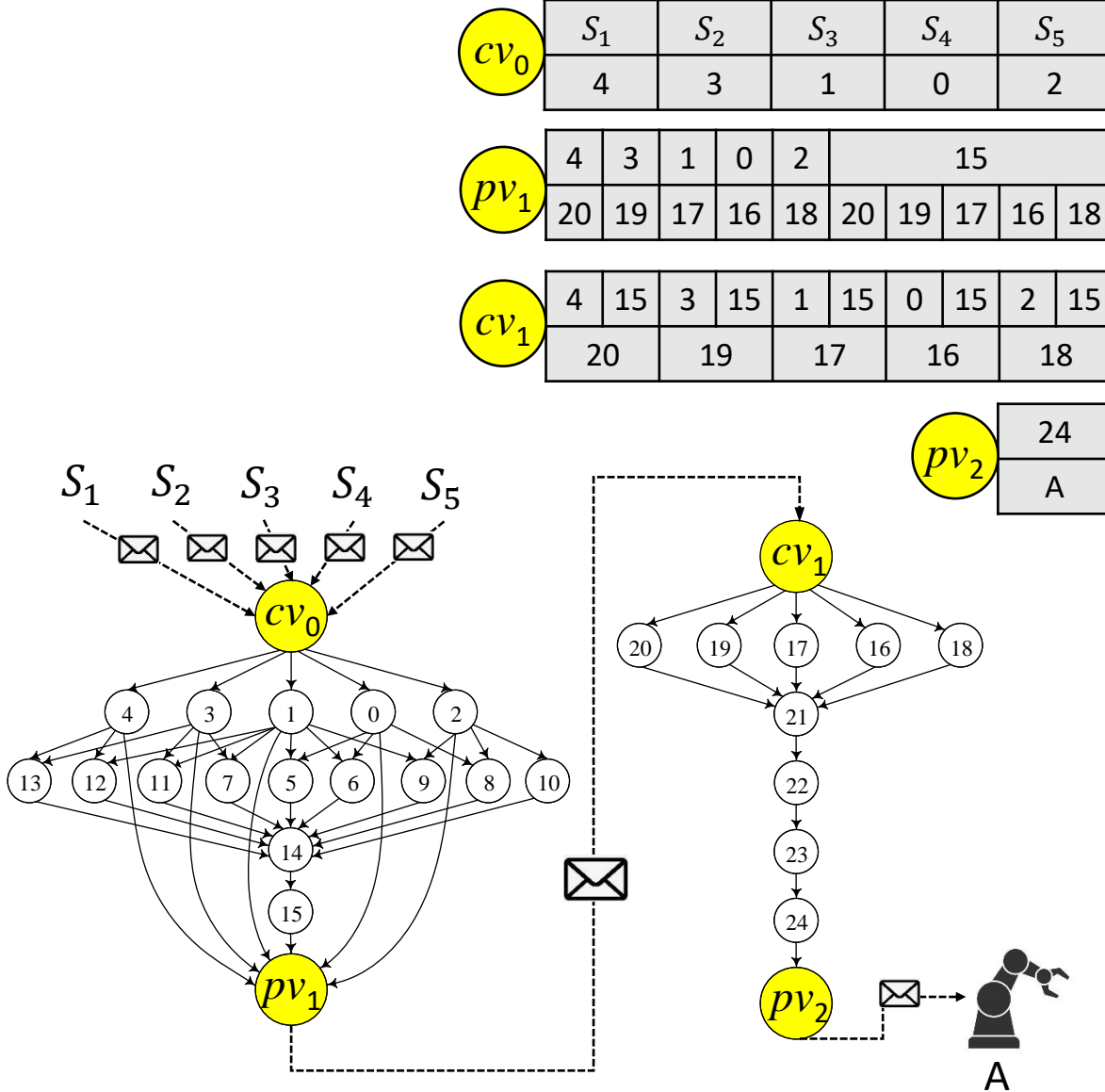
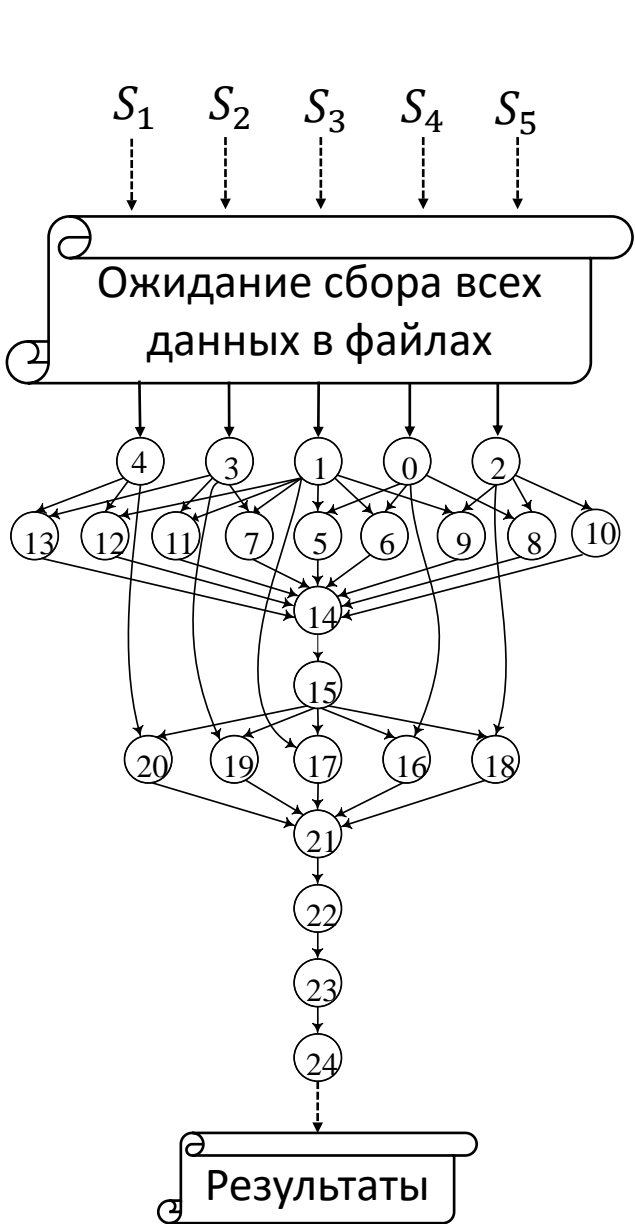




# Представление подпотока в виде микросервиса (микро-потока работ)

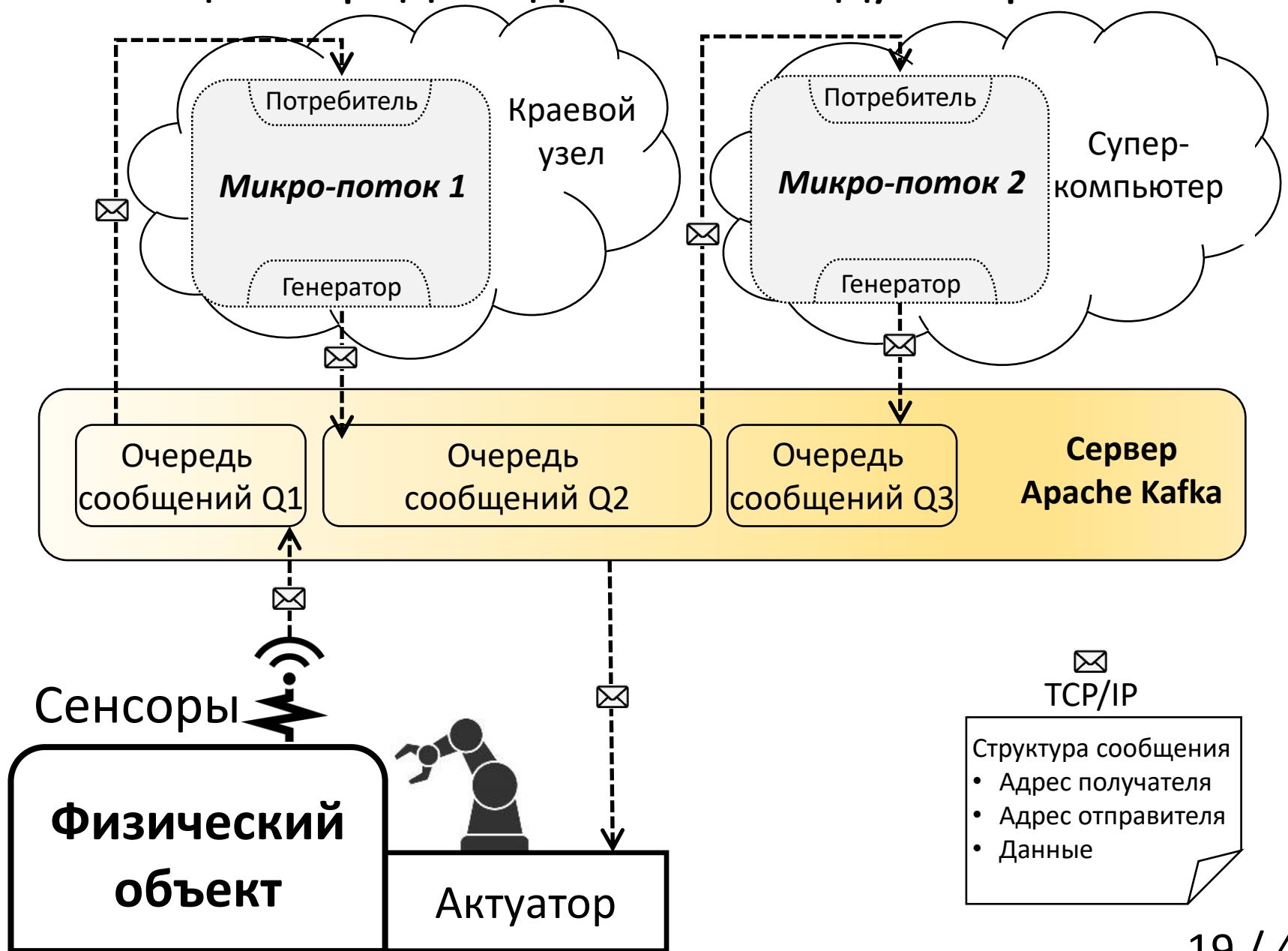
- Контейнер:
  - **Среда выполнения потока работ (например, Kepler):**
    - 1) Выделенный под-поток работ:**
      - Вершины под-потока работ, выделенные из основного потока работ, со связями между ними
    - 2) Интерфейс-потребитель (Consumer) - структура данных потребителя:**
      - Вершины источника данных -> Вершины моего подпотока
    - 3) Интерфейс-генератор (Producer) - структура данных генератора:**
      - Вершины моего подпотока -> Вершины потребителя данных

# Пример разбиения потока на два микро-потока



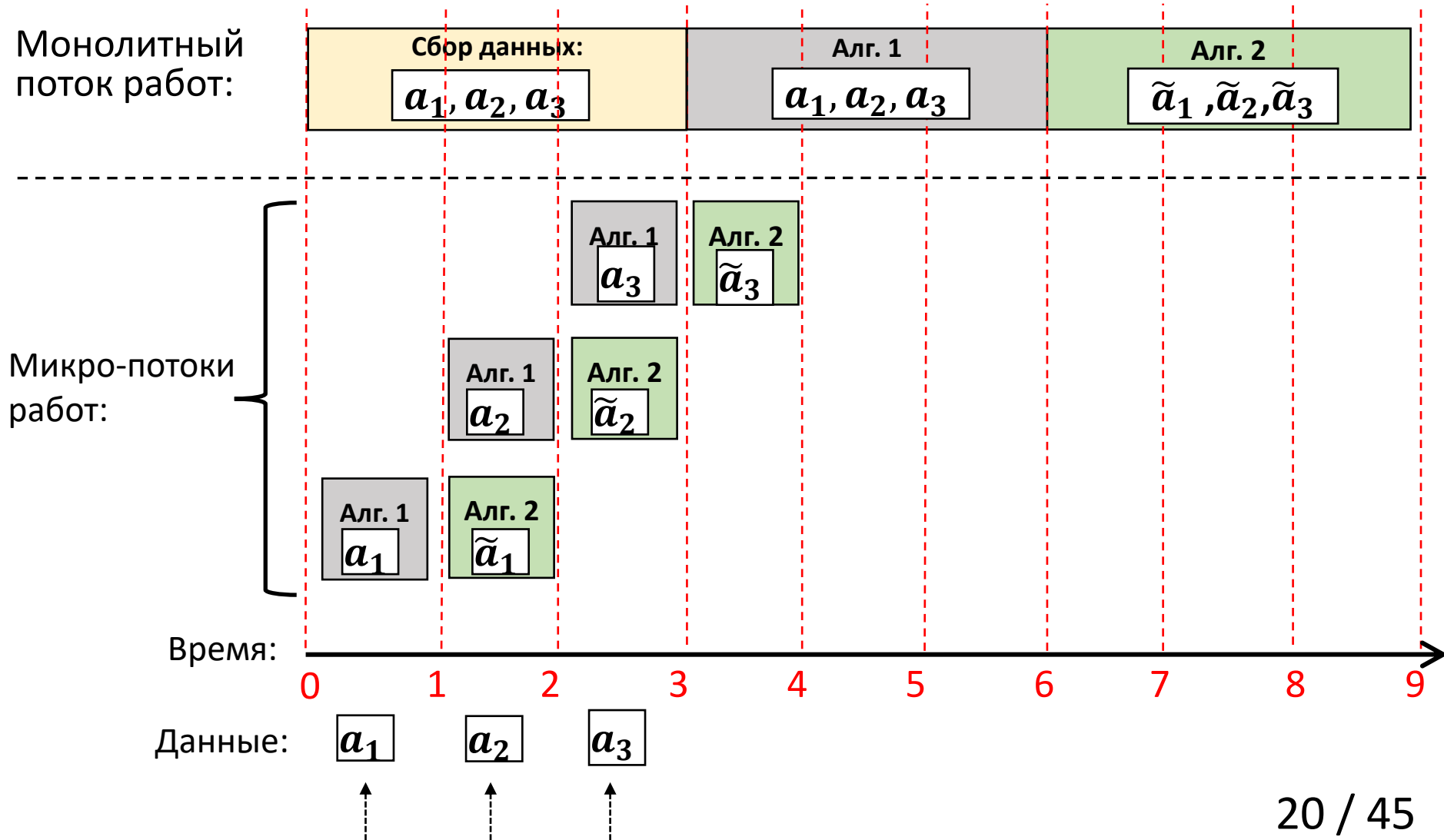
$cv_0$	$S_1$	$S_2$	$S_3$	$S_4$	$S_5$					
	4	3	1	0	2					
$pv_1$	4	3	1	0	2	15				
	20	19	17	16	18	20	19	17	16	18
$cv_1$	4	15	3	15	1	15	0	15	2	15
	20		19		17		16		18	
$pv_2$										24
										A

# Организация передачи данных между микро-потоками



# Микро-потоки работ против монолитного потока работ

Допустим, что данные должны обрабатываться сначала в **Алг. 1**, а затем в **Алг. 2**:



# Задача "Химический завод"

Имеется

- 8 сенсоров
- Сигналы поступают с частотой 10 мс

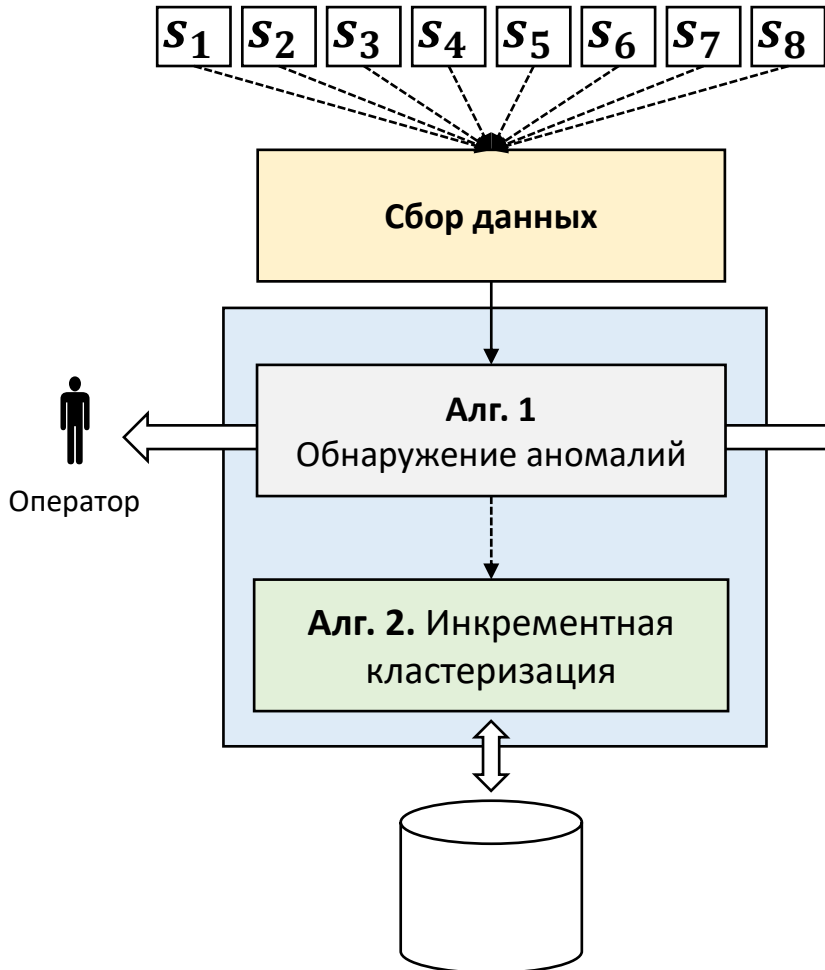
Задача

**Алг.1:** Обнаружить в потоке данных каждого сенсора аномалию и дать сигнал оператору

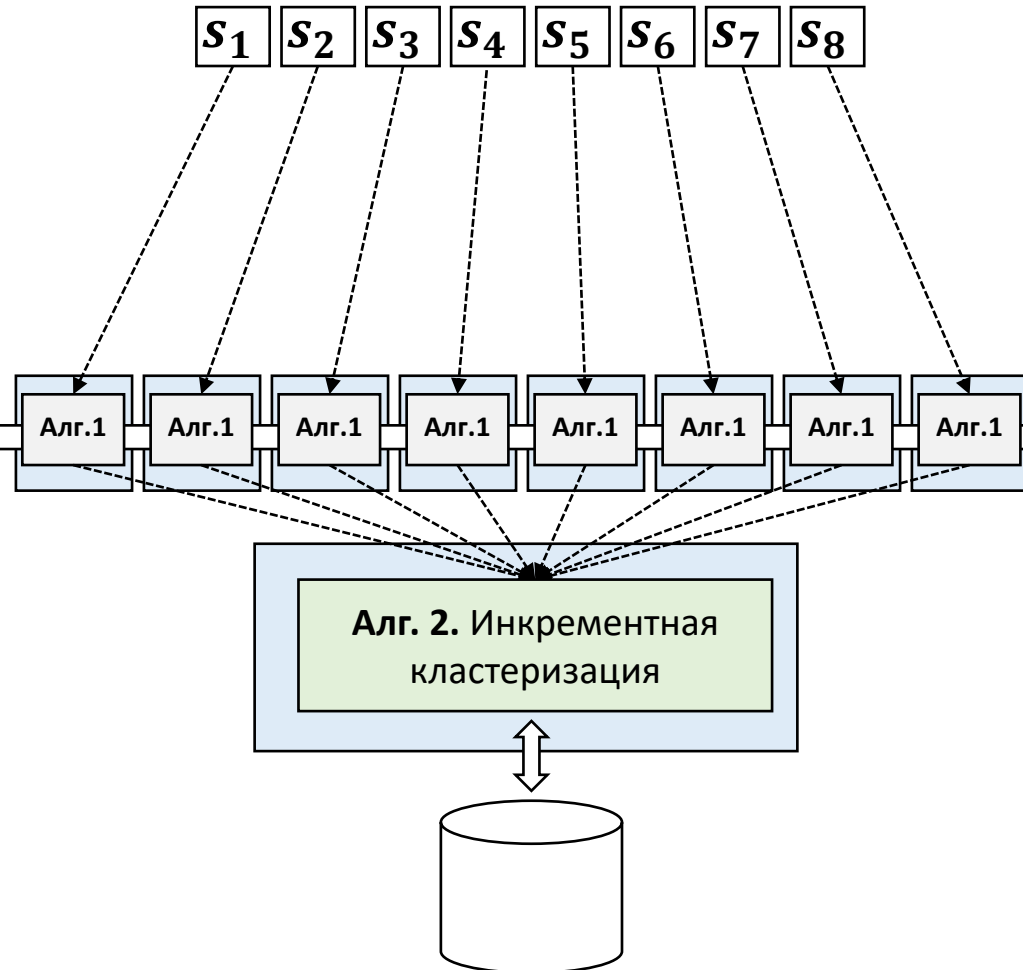
**Алг.2:** Выполнить инкрементную кластеризацию аномалий

# Реализация

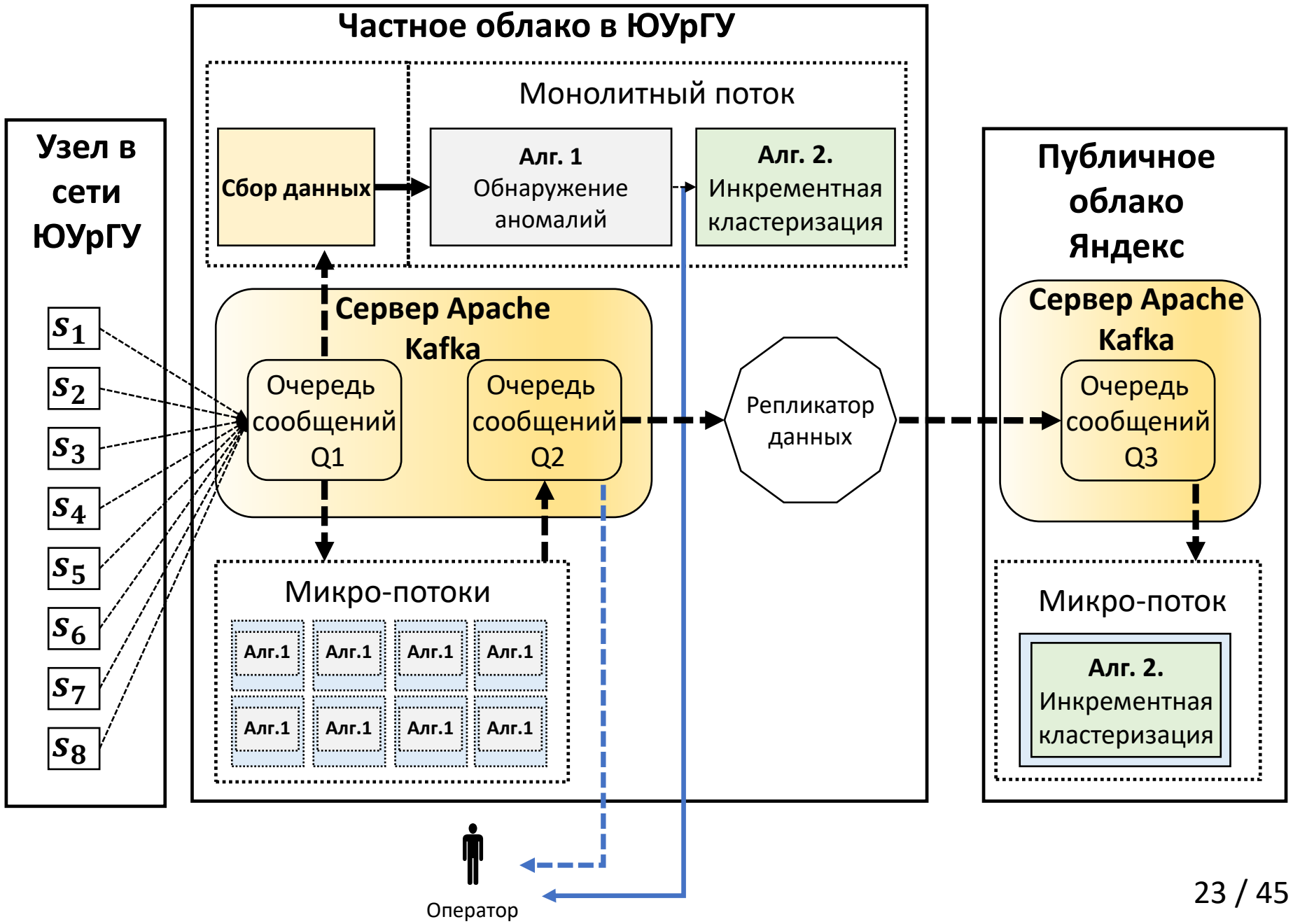
## Монолитный поток



## Микро-потоки



# Реализация эксперимента



# Результаты эксперимента

	Средняя задержка в подаче сигнала оператору
Монолитный поток работ	<b>670 732 мс</b>
<b>Микро-поток работ</b>	<b>1.4 мс</b>



# Математическая модель микро-потоков работ

# Монолитный поток работ

Поток работ может быть представлен в виде ориентированного ациклического графа (Directed Acyclic Graph - DAG)

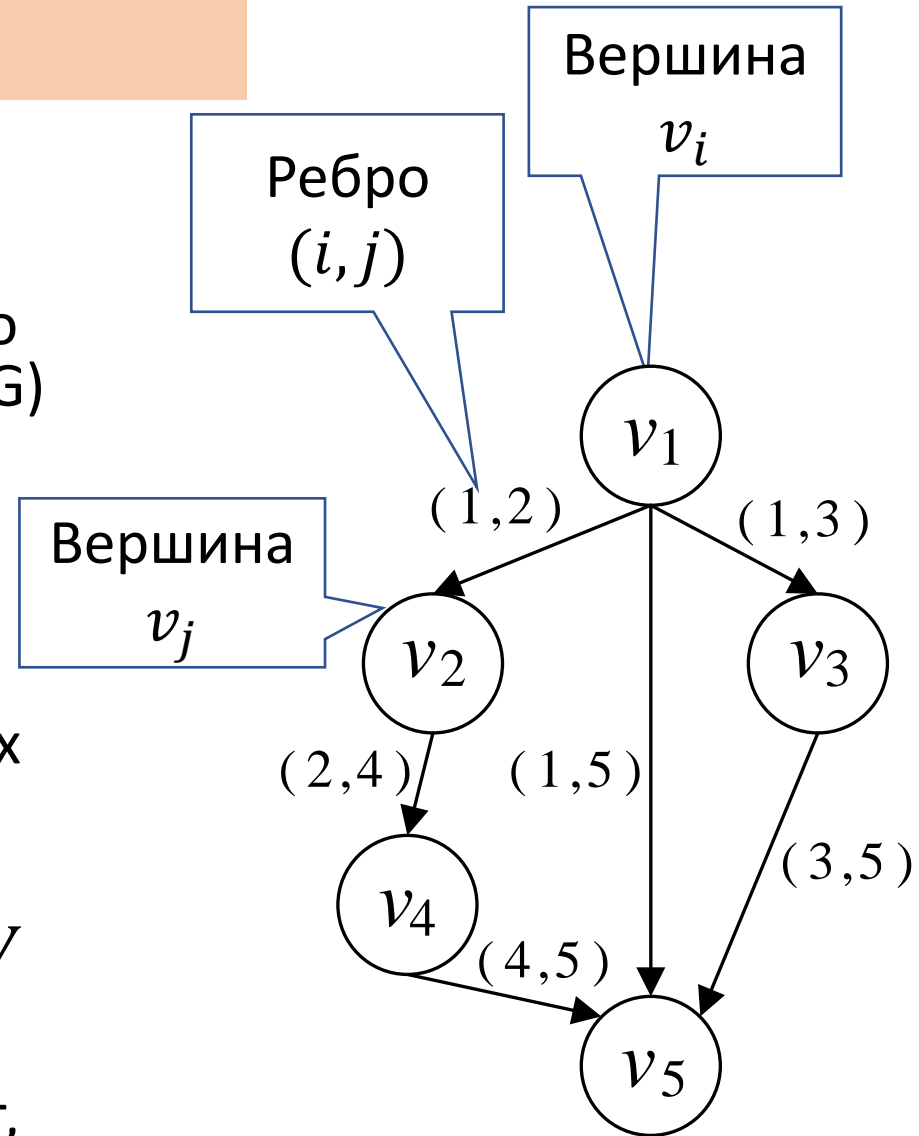
$$W = (V, E)$$

$V$  – набор вершин, представляющих собой вычислительные задачи.

$E$  – набор ориентированных ребер, соединяющий вершины.

$n$  – общее число вершин в  $V$

Каждое ребро  $(i, j) \in E$  означает, что есть данные, которые идут от источника  $v_i$  к цели  $v_j$

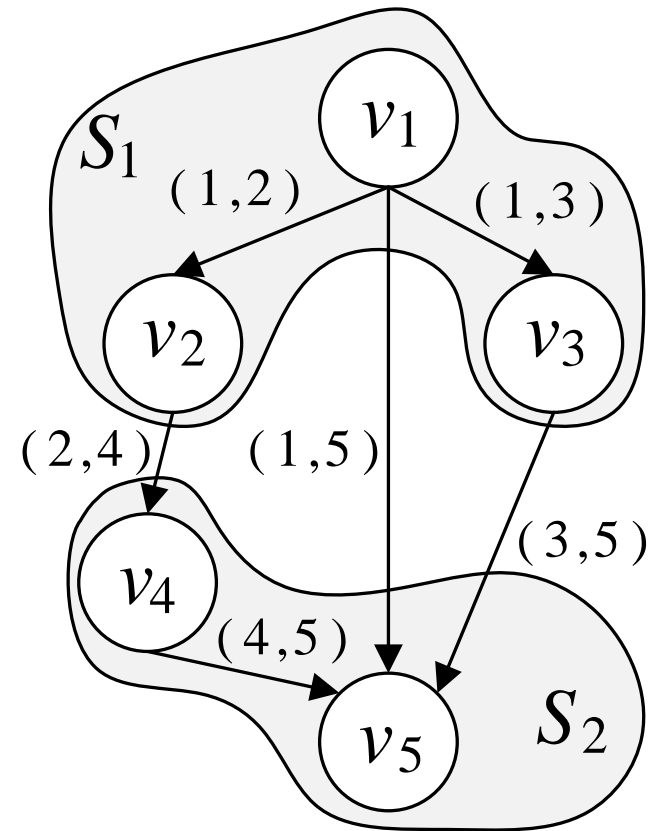


$W$  подразделяется на множество подпотоков работ  $S$ , где:

$$S = (S_1, \dots, S_k): \forall S_i = (V_i, E_i)$$

Когда:

- $\forall i \in 1..k: V_i \subset V, E_i \subset E;$
- $\forall v \in V, \exists i \in 1..k: v \in V_i;$
- $E_i = \{(v_k, v_l) \in E: v_k, v_l \in V_i\};$
- $\forall i, j: i \neq j \Rightarrow V_i \cap V_j = \emptyset.$



# Классификация ребер и вершин

**$EI_i$** : набор входящих ребер с начальной вершиной вне подпотока работ  $S_i$  и конечной вершиной внутри подпотока работ  $S_i$

$$EI_i = \{(v_k, v_l) \in E : v_k \notin S_i, v_l \in S_i\};$$

**$EO_i$** : набор исходящих ребер с начальной вершиной внутри подпотока работ  $S_i$  и конечной вершиной вне подпотока работ  $S_i$

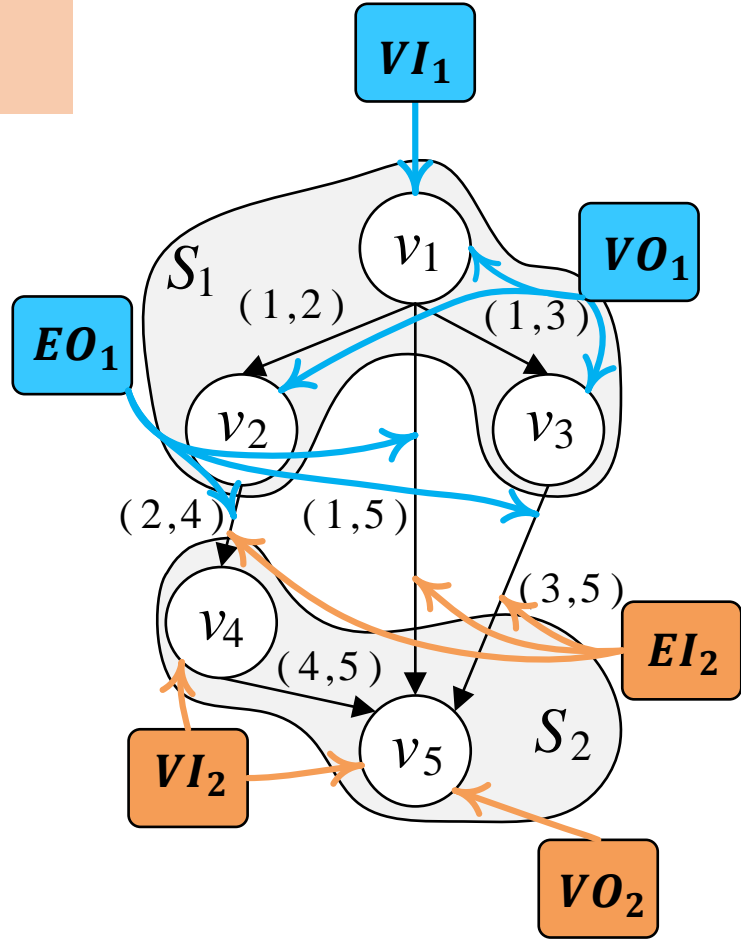
$$EO_i = \{(v_k, v_l) \in E : v_k \in S_i, v_l \notin S_i\};$$

**$VI_i$** : набор вершин в  $S_i$ , расположенных на головной части ребер  $EI_i$ , а также вершин, не имеющих входящих ребер

$$VI_i = \{v_l \in S_i : (v_k, v_l) \in EI_i\} \cup \{v \in S_i : deg^-(v) = 0\};$$

**$VO_i$** : набор вершин в  $S_i$ , расположенных на концах ребер  $EO_i$ , а также вершин, не имеющих исходящих ребер

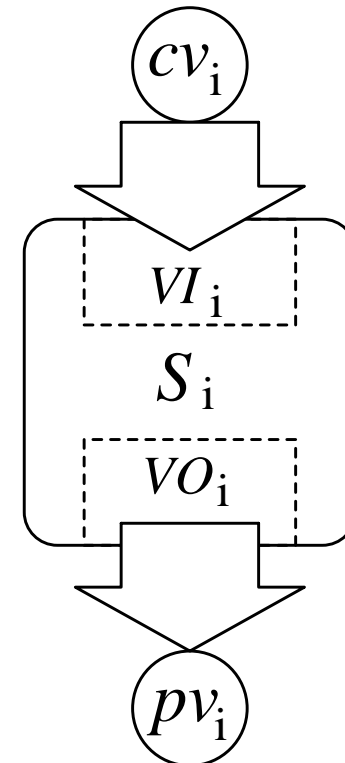
$$VO_i = \{v_k \in S_i : (v_k, v_l) \in EO_i\} \cup \{v \in S_i : deg^+(v) = 0\}.$$



## Формирование микро-потока работ

Для формирования микро-потока работ из под-потока  $S_i$  необходимо перенести все вершины  $S_i$  из  $W$  и обеспечить коммуникационные механизмы, связывающие  $MWF_i$  с платформой потоковой передачи событий через  $cv_i$  и  $pv_i$

- **Вершина-потребитель (consumer vertex,  $cv_i$ )** обеспечивает потребление потока входных данных от платформы потоковой передачи данных и распределяет его между вершинами  $VI_i$
- **Вершина-генератор (producer vertex,  $pv_i$ )** действует как сток, который собирает выходные данные из вершин, составляющих множество  $VO_i$  и передает их в виде сообщений на платформу потоковой передачи данных



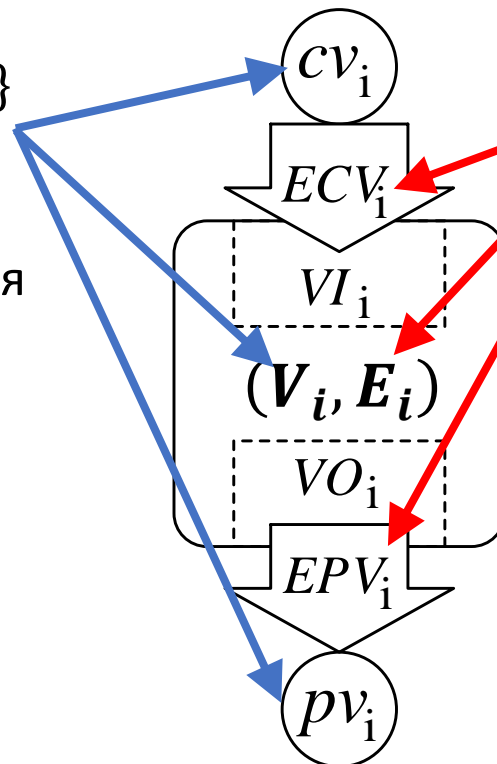
# Определение микро-потока работ

$ECV_i = \{(cv_i, v) : v \in VI_i\}$ , набор ребер, идущих от  $cv_i$  к  $VI_i$ ;

$EPV_i = \{(v, cp_i) : v \in VO_i\}$ , набор ребер, идущих от  $VO_i$  к  $cp_i$ .

$$MV_i = V_i \cup \{cv_i, pv_i\}$$

множество всех  
вершин, находящихся  
внутри  $S_i$ , в  
дополнение к  $cv_i$  и  
 $pv_i$ ;



$$ME_i = E_i \cup ECV_i \cup EPV_i$$

множество всех ребер,  
расположенных внутри  $S_i$ , в  
дополнение ко всем ребрам,  
которые идут из  $cv_i$  к  
вершинам в  $VI_i$ , а также всем  
ребрам, идущих от вершин в  
 $VO_i$  к  $cp_i$

$$MWF_i = (MV_i, ME_i)$$

# Алгоритм рефакторинга потока работ на микро-потоки работ

---

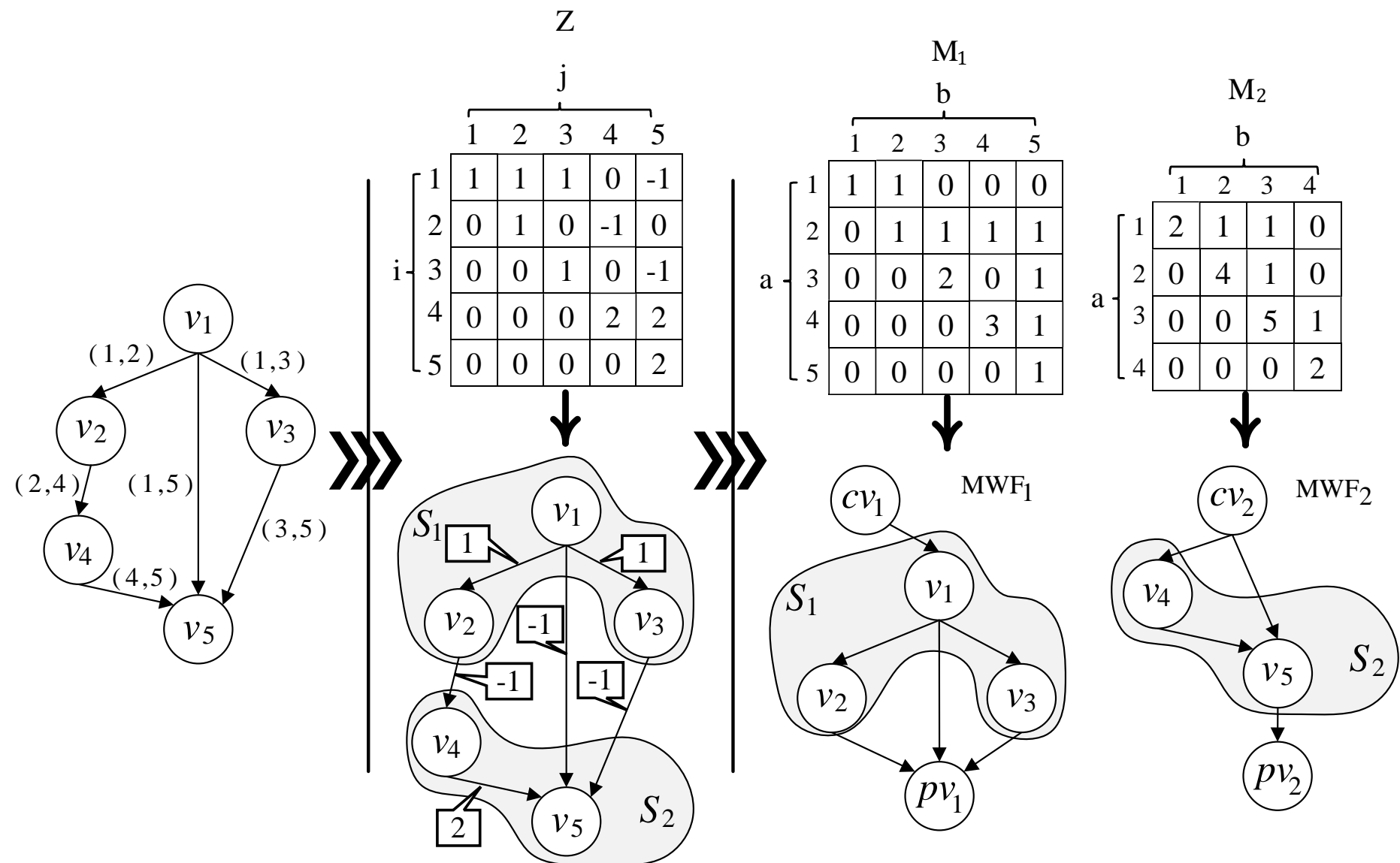
## Алгоритм рефакторинга потока работ на микро-потоки работ

---

```
1: procedure Refactor
2:  $wf, subWfs \leftarrow \text{ReadWorkflow}(PARAMS.SOURCE)$ 
3:  $Z \leftarrow \text{InitializeZ}(wf, subWfs)$ 
4:  $Z \leftarrow \text{IdentifyInternalExternalEdges}(Z)$ 
5: for  $i$  from 1 to  $subWfs.Length()$ :
6:    $MWF[i] \leftarrow \text{InitializeM}(subWfs[i], Z)$ 
7:    $MWF[i] \leftarrow \text{InternalEdges}(MWF[i], Z)$ 
8:    $MWF[i] \leftarrow \text{AddCv}(MWF[i], Z)$ 
9:    $MWF[i] \leftarrow \text{AddPv}(MWF[i], Z)$ 
10:  $\text{WriteMWFs}(MWF)$ 
11: end procedure
```

---

# Работа алгоритма рефакторинга

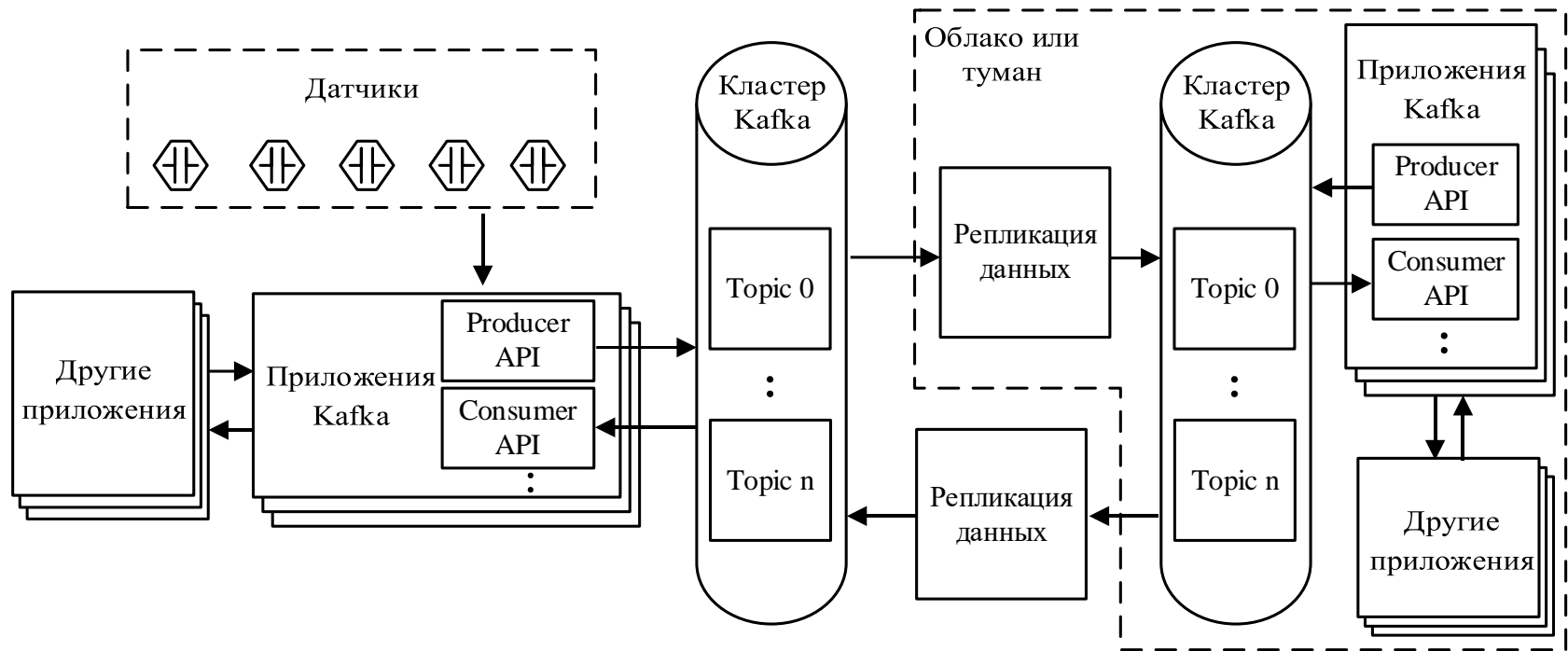




# Программная поддержка микро-потоков работ

# Используемые для реализации технологии и платформы

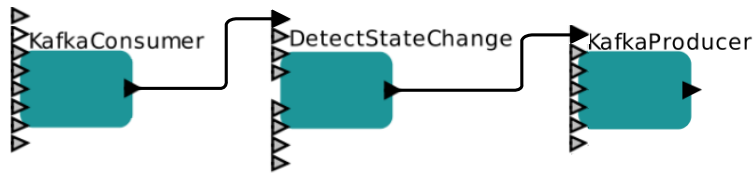
1. Код системы написан на языке Java
2. В качестве технологии контейнеризации используется платформа Docker
3. В качестве системы управления потоками работ используется система Kepler
4. В качестве потокового промежуточного программного обеспечения используется Apache Kafka



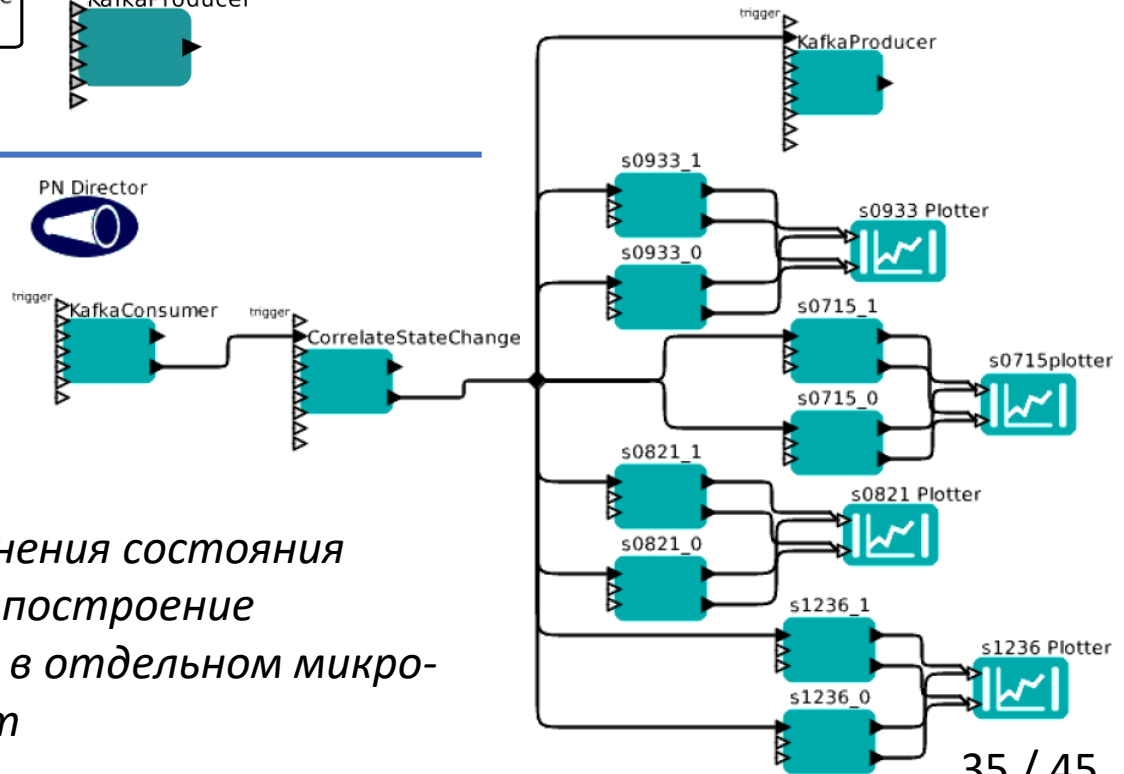
# Программная поддержка модели микро-потоков работ



1- Этап изменения состояния в отдельном микро-потоке работ

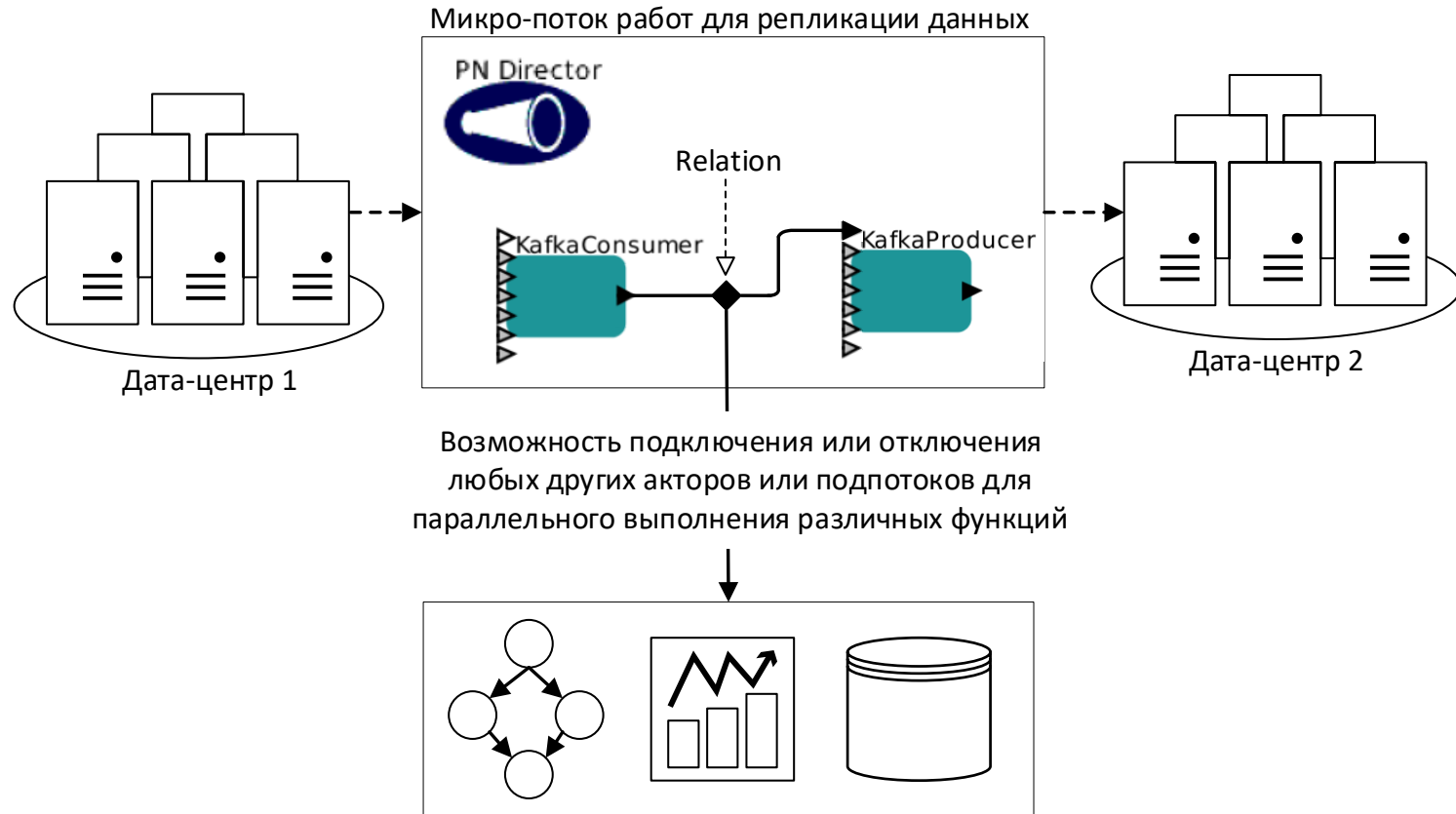


2- Этап изменения состояния корреляции и построение результатов в отдельном микро-потоке работ



# Программная поддержка модели микро-потоков работ

## Репликатор данных

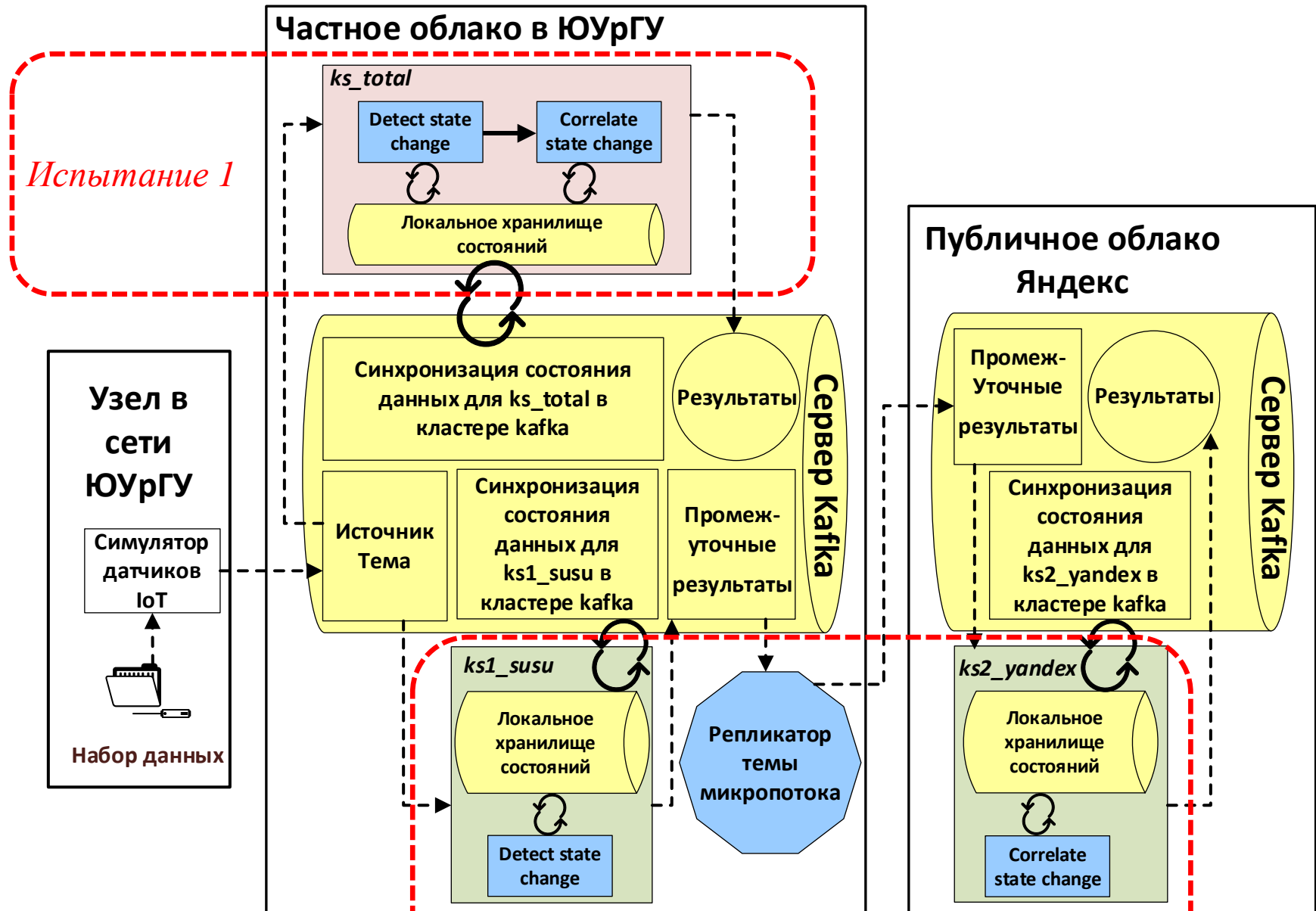


# Эксперимент по распределению вычислительной нагрузки в туманной вычислительной среде

Цель эксперимента:

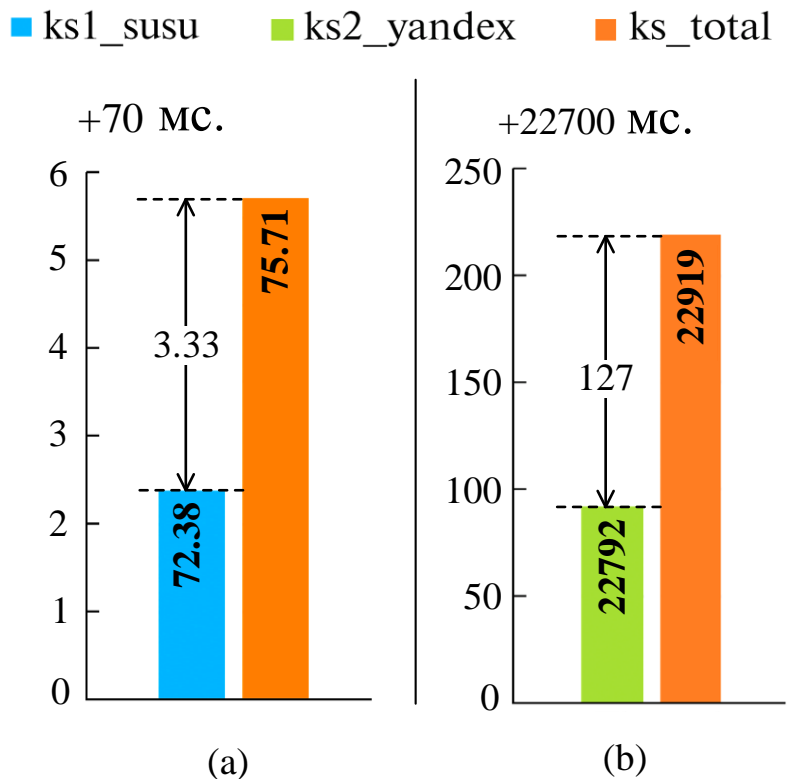
Исследовать характеристику развертывания  
микротока с сохранением состояния в модели  
туманных вычислений.

# Эксперимент по распределению вычислительной нагрузки в туманной вычислительной среде



*Испытание 2*

# Эксперимент по распределению нагрузки в туманной вычислительной среде



Среднее время предоставления результата: (a) ks\_susu и его соответствующей части в ks\_total, (b) ks\_yandex и его соответствующей части в ks\_total

	Вычислительные этапы в одном большом микропоточе ks_total	Модель микропотока работы ks1_susu, ks2_yandex
Публичная отправка сообщений при использовании публичного облака	1 638 104 сообщений	1 616 сообщений (уменьшение объема в 1 014 раз)

# Эксперимент по живой миграции микро-потока работ

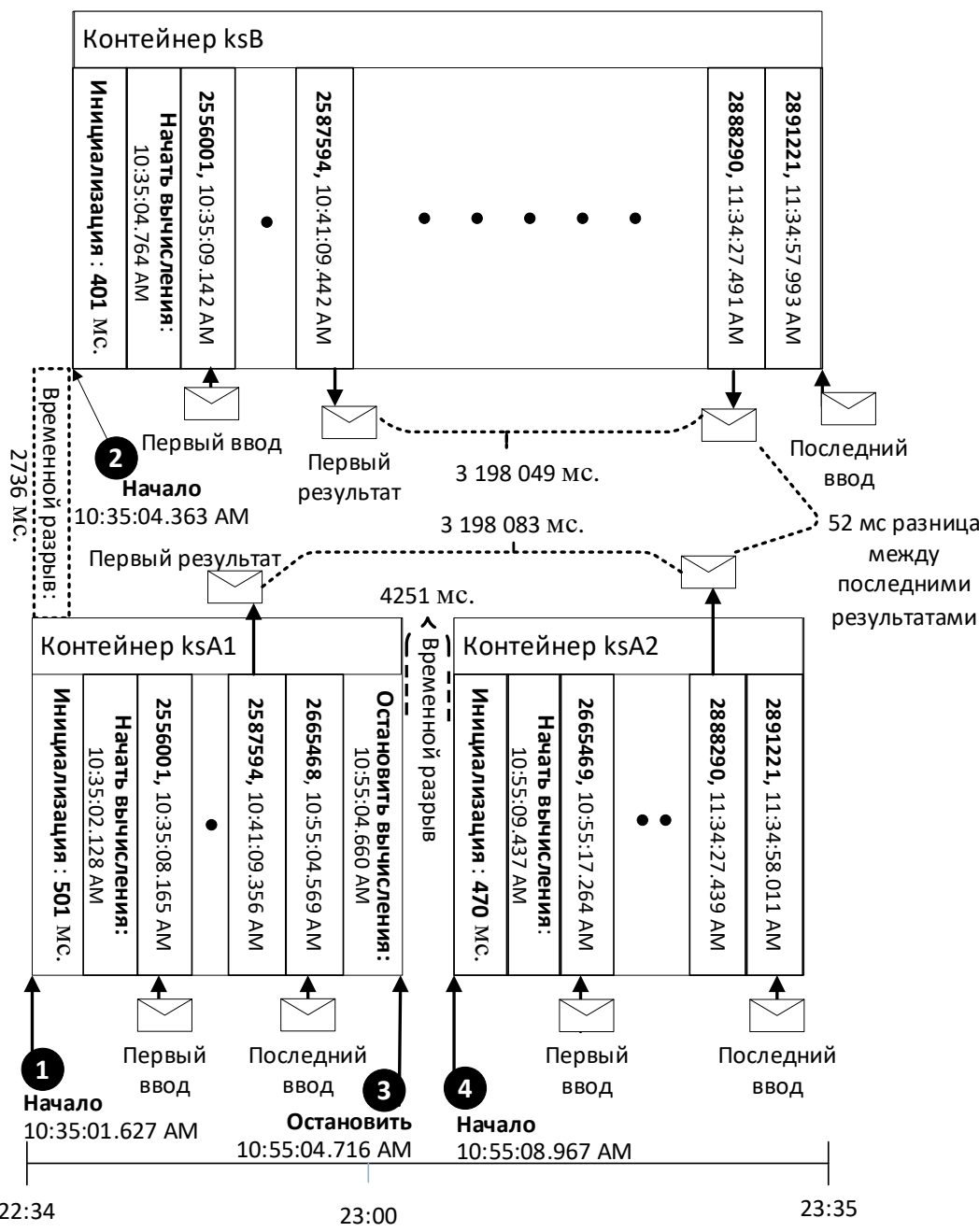
Цель эксперимента:

Обеспечить живую миграцию вычислительной задачи из контейнера в другой контейнер, обеспечивая при этом непрерывность результатов, полученных в предыдущей точке остановки.

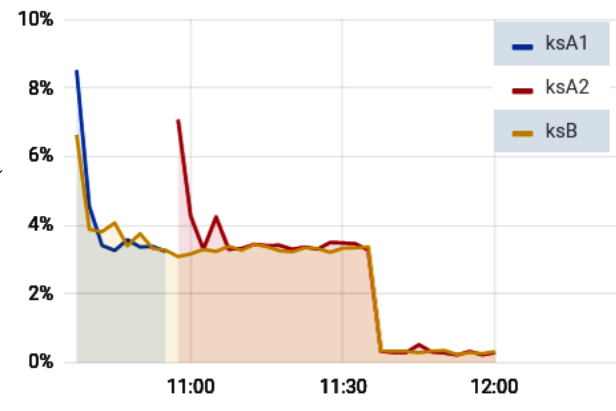
\* Живой миграцией называют возможность бесперебойной миграции сервиса между физическими машинами без воздействия на клиентские процессы или приложения.



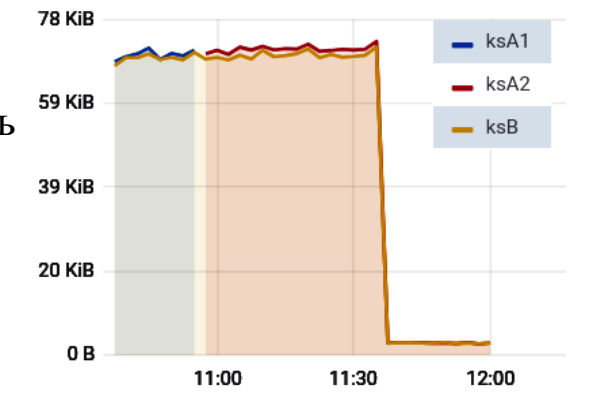
# Эксперимент по живой миграции микро-потока работ



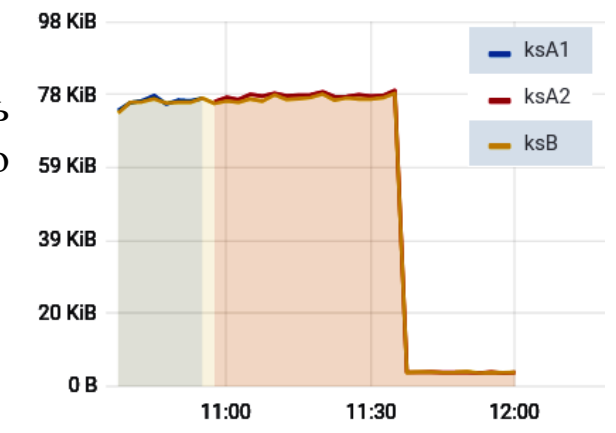
## 1. Загрузка CPU



## 2. Скорость входящего сетевого трафика



## 3. Скорость исходящего сетевого трафика



# Положения, выносимые на защиту

1. Разработана новая концепция микро-потоков работ, ориентированная на организацию обработки данных в туманных вычислительных средах, позволяющая значительно уменьшить время задержки получения результата при обработке потоков данных
2. Разработан алгоритм преобразование монолитных потоков работ в микро-потоки работ
3. Разработана программная система для автоматического преобразования монолитных потоков работ в микро-потоки работ
4. С использованием разработанной программной системы созданы микро-потоки работ, на базе которых проведены вычислительные эксперименты, подтверждающие эффективность предложенных подходов

## Публикации в журналах

1	<b>Alaasam, A.B.A.</b> The Challenges and Prerequisites of Data Stream Processing in Fog Environment for Digital Twin in Smart Industry // <b>International Journal of Interactive Mobile Technologies</b> , 15 (15), pp. 126-139. 2021 DOI: <a href="https://doi.org/10.3991/ijim.v15i15.24181">https://doi.org/10.3991/ijim.v15i15.24181</a>	-	<b>Scopus</b>	-
2	<b>Alaasam A. B. A.</b> , Radchenko G., Tchernykh A., and González Compeán J. L., “Analytic Study of Containerizing Stateful Stream Processing as Microservice to Support Digital Twins in Fog Computing,” <b>Programming and Computer Software</b> , vol. 46, no. 8, pp. 511–525, Dec. 2020, doi: <a href="https://doi.org/10.1134/S0361768820080083">https://doi.org/10.1134/S0361768820080083</a>	<b>WoS</b>	<b>Scopus</b>	<b>BAK</b>
3	<b>Alaasam, A.B.A.</b> , Radchenko, G., Tchernykh, A. Refactoring the Monolith Workflow into Independent Micro-Workflows to Support Stream Processing// <b>Programming and Computer Software</b> , –2021.Vol. 47, No. 8. –P. 591-600. DOI: <a href="https://doi.org/10.1134/S0361768821080077">https://doi.org/10.1134/S0361768821080077</a>	<b>WoS</b>	<b>Scopus</b>	<b>BAK</b>
4	<b>Alaasam A. B. A.</b> , Radchenko G., and Tchernykh A., “Micro-Workflows Data Stream Processing Model for Industrial Internet of Things,” <b>Supercomputing Frontiers and Innovations</b> , vol. 8, no. 1, pp. 82–98, Mar. 2021, doi: <a href="https://doi.org/10.14529/jsfi210106">https://doi.org/10.14529/jsfi210106</a>	-	<b>Scopus</b>	<b>BAK</b>
5	Radchenko G., <b>Alaasam A. B. A.</b> , and Tchernykh A., “Comparative Analysis of Virtualization Methods in Big Data Processing,” <b>Supercomputing Frontiers and Innovations</b> , vol. 6, no. 1, pp. 48–79, Mar. 2019, doi: <a href="https://doi.org/10.14529/jsfi190107">https://doi.org/10.14529/jsfi190107</a>	-	<b>Scopus</b>	<b>BAK</b>
6	<b>Алаасам А. Б. А.</b> , Радченко Г. И., Черных А. Н., and Гонсалес-Компеан Х. Л., “Цифровые двойники в туманных вычислениях: организация обработки данных с сохранением состояния на базе микропотоков работ,” <b>Труды Института системного программирования РАН</b> , vol. 33, no. 1, pp. 65–80, Dec. 2021, doi: <a href="https://doi.org/10.15514/ISPRAS-2021-33(1)-5">https://doi.org/10.15514/ISPRAS-2021-33(1)-5</a>	-	-	<b>BAK</b>
7	<b>Алаасам А. Б. А.</b> , Радченко Г. И., and Черных А. Н., “Микро-потоки работ: сочетание потоков работ и потоковой обработки данных для поддержки цифровых двойников технологических процессов,” <b>Вестник ЮУрГУ. Серия: Вычислительная математика и информатика</b> , vol. 8, no. 4, pp. 100–116, 2019, doi: <a href="https://doi.org/10.14529/cmse190387">https://doi.org/10.14529/cmse190387</a>	-	-	<b>BAK</b>

## В трудах конференций

1	Radchenko G., <b>Alaasam A. B. A.</b> , and Tchernykh A., “Micro-Workflows: Kafka and Kepler Fusion to Support Digital Twins of Industrial Processes,” in <b>2018 IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC Companion)</b> , Dec. 2018, no. 18, pp. 83–88, doi: <a href="https://doi.org/10.1109/UCC-Companion.2018.00039">https://doi.org/10.1109/UCC-Companion.2018.00039</a>	WoS	Scopus	РИНЦ
2	<b>Alaasam A. B. A.</b> , Radchenko G., and Tchernykh A., “Stateful Stream Processing for Digital Twins: Microservice-Based Kafka Stream DSL,” in <b>2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)</b> , Oct. 2019, pp. 0804–0809, doi: <a href="https://doi.org/10.1109/SIBIRCON48586.2019.8958367">https://doi.org/10.1109/SIBIRCON48586.2019.8958367</a>	-	Scopus	РИНЦ
3	<b>Alaasam A. B. A.</b> , Radchenko G., Tchernykh A., Borodulin K., and Podkorytov A., “Scientific Micro-Workflows : Where Event-Driven Approach Meets Workflows to Support Digital Twins,” in <b>Proceedings of the international conference RuSCDays’18 - Russian Supercomputing Days (September 24-25, 2018, Moscow, Russia)</b> , MSU, 2018, vol. 1, pp. 489–495. <a href="https://elibrary.ru/item.asp?id=36548900">https://elibrary.ru/item.asp?id=36548900</a>	-	-	РИНЦ

## Свидетельства о регистрации программ для ЭВМ

Алаасам, А.Б.А., Радченко Г.И. Свидетельство Роспатента о государственной регистрации программы для ЭВМ "Комплекс акторов для поддержки концепции Micro-Workflow на платформе Kepler" № 2021661464 от 12.07.2021

Спасибо за внимание!

Вопросы?