Эквивалентные выполнения

- Пусть:
  - $R = (e^0, e^1, \ldots e^x, e^{x+1}, \ldots)$ – выполнение
  - $(S^0, S^1, \ldots S^x, S^{x+1}, \ldots)$ – связанная с R последовательность глобальных состояний: $S^{x+1} = e^x(S^x)$ для всех $x \geq 0$
  - $\hat{R} = (\hat{e}^0, \hat{e}^1, \ldots \hat{e}^x, \hat{e}^{x+1}, \ldots)$ – перестановка элементов R
- Тогда:
  - перестановка $\hat{R}$ сохраняет причинно-следственный порядок, если $\hat{e}^x \rightarrow \hat{e}^y \Rightarrow x < y$
  - т.е. ни одно событие $\hat{e}^y$ не появляется в $\hat{R}$ перед событием $\hat{e}^x$, от которого $\hat{e}^y$ зависит

Эквивалентные выполнения

ITIVITI

LET IT TRADE

- Пусть:
  - $\hat{R}$ - перестановка, сохраняющая причинно-следственный порядок
- Тогда:
  - $\hat{R}$ - выполнение распределенной системы с начальным состоянием S^0
  - если R – конечное выполнение из k событий, то $S^k = \hat{S}^k$

УНИВЕРСИТЕТ ИТМО

Эквивалентные выполнения

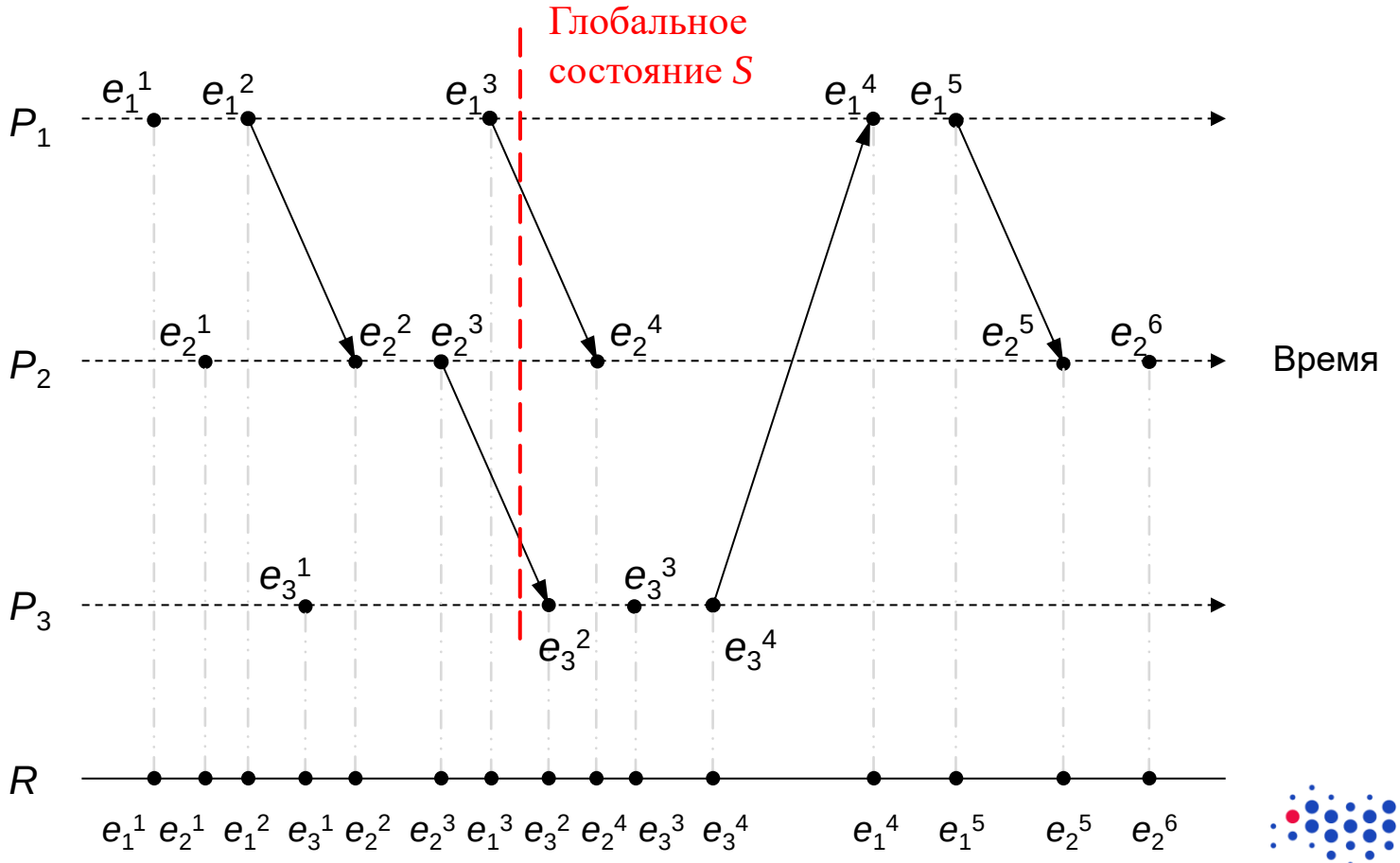
- Доказательство (набросок):
  - порядок событий процесса P_i в перестановке $\hat{R}$ будет таким же, что и в выполнении R, значит каждый процесс будет проходить через те же состояния
  - взаимосвязанные события отправки и получения в $\hat{R}$ расположены в том же порядке, что и в R, значит каждый канал будет проходить через те же состояния
  - состояние P_i в S^k определяется событием, последним для P_i в R; это же событие – последние для P_i в $\hat{R}$
  - в R и $\hat{R}$ содержится одна и та же совокупность событий, значит те же сообщения окажутся в состоянии пересылки и для $\hat{S}^k$ в $\hat{R}$

Эквивалентные выполнения

- В обоих выполнениях R и $\hat{R}$ происходит одна и та же совокупность событий, и причинно-следственный порядок этих событий одинаков для R и $\hat{R}$
- В этом случае выполнения R и $\hat{R}$ называются эквивалентными: $R \sim \hat{R}$
- Порождают разные множества глобальных состояний

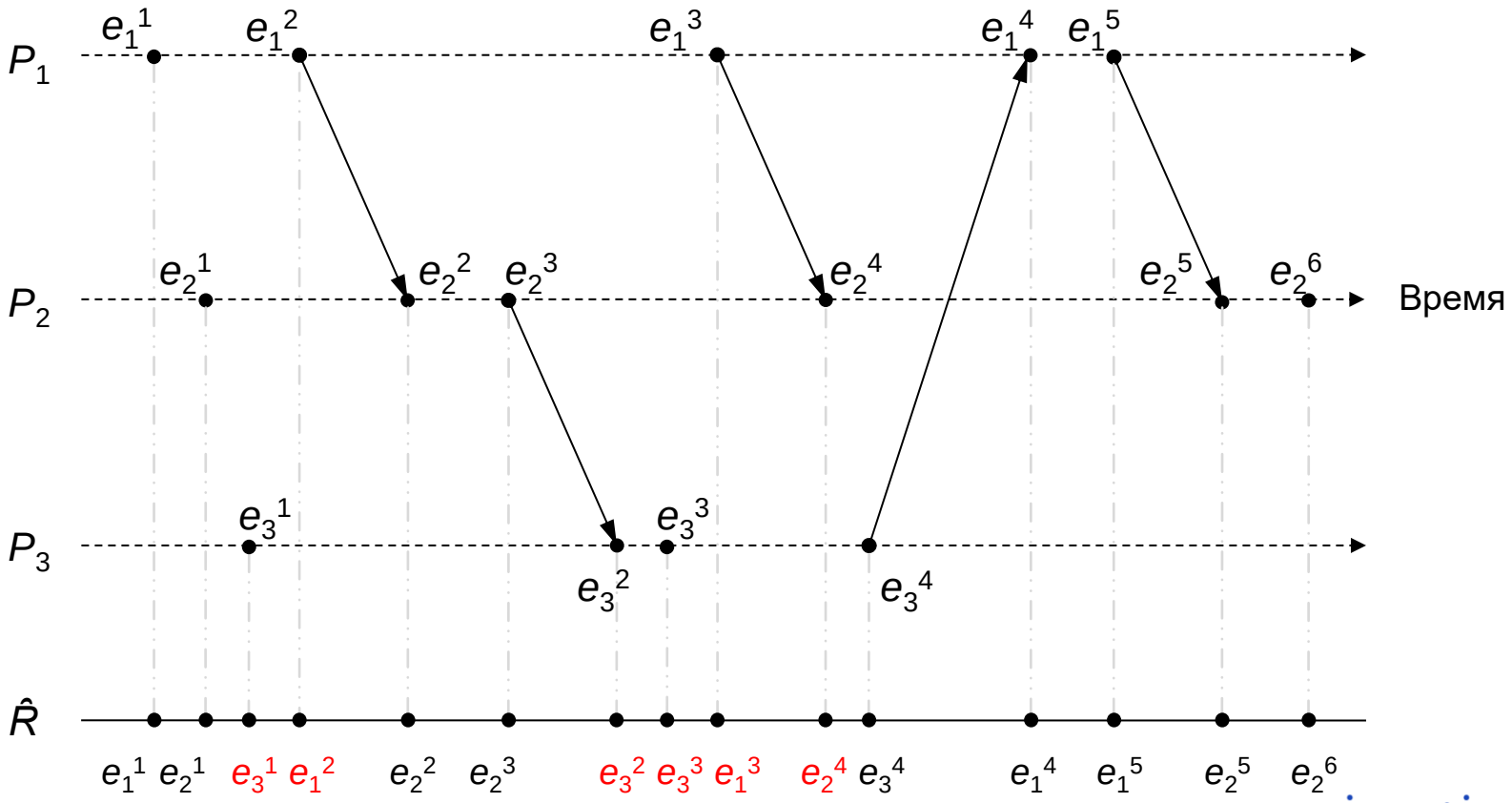
Эквивалентные выполнения

Пространственно-временная диаграмма:



Эквивалентные выполнения

Пространственно-временная диаграмма:



Эквивалентные выполнения

ITIVITI

- Сторонний наблюдатель способен отличить одно эквивалентное выполнение от другого: всякий раз он видит только одно из возможных выполнений
- Процессы не могут отличить два эквивалентных выполнения: с их точки зрения невозможно понять, какое из двух эквивалентных выполнений происходит на самом деле

LET IT TRADE

УНИВЕРСИТЕТ ИТМО

Эквивалентные выполнения: вычисление

ITIVITI

LET IT TRADE

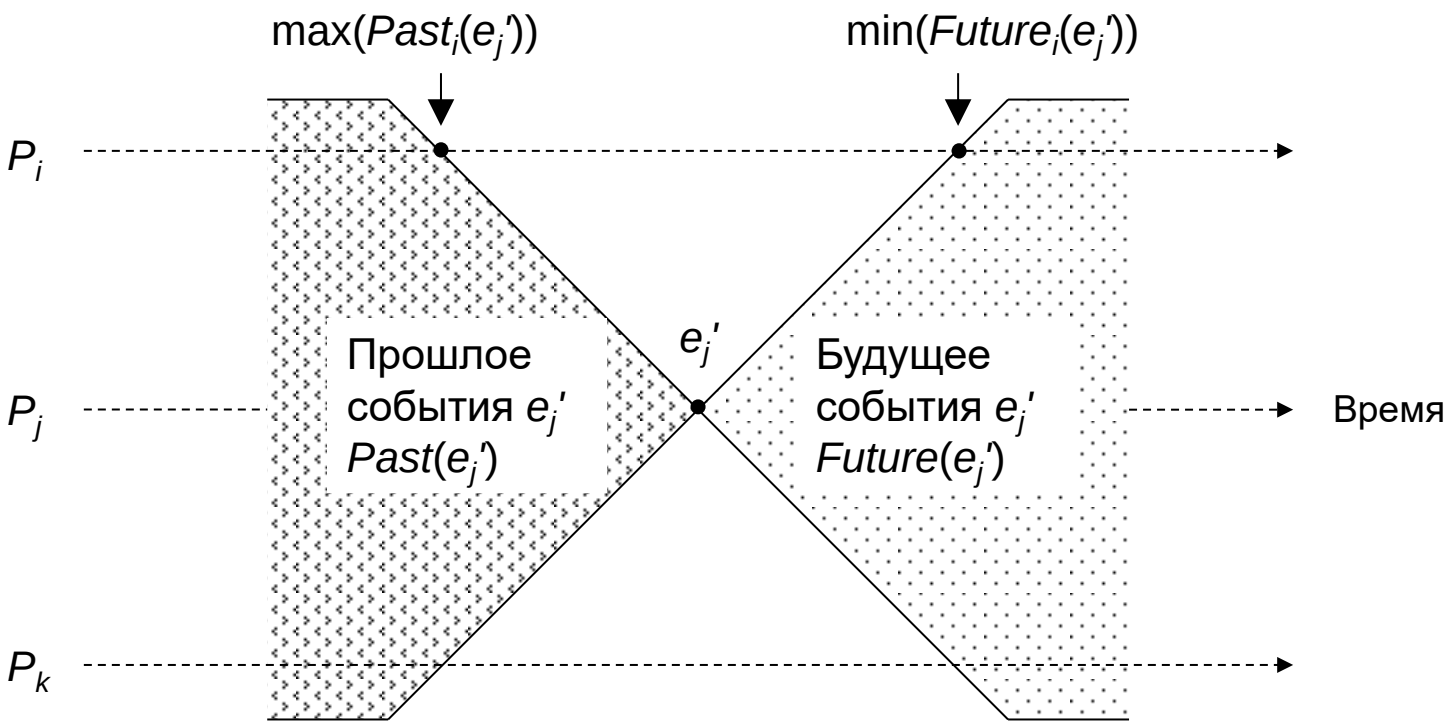
- Класс эквивалентности выполнений по отношению ~
- Однозначно определяется множеством событий и действующим на нем причинно-следственным порядком
- Обозначение: $\boldsymbol{C} = (E, \rightarrow)$

УНИВЕРСИТЕТ ИТМО

Эквивалентные выполнения: вычисление

- Выполнение R – линеаризация (*linear extension*) частичного порядка:
  - введение линейного порядка $<$, сохраняющего частичный причинно-следственный порядок событий $\rightarrow$
  - т.е. для любых двух e_i и e_j': $e_i \rightarrow e_j' \Rightarrow e_i < e_j'$
- Если $e_i \not\rightarrow e_j'$, то существует линеаризация $<$ для которой $e_j' < e_i$
  - если $e_i \parallel e_j'$, то существуют два выполнения R и $\hat{R}$, в одном из которых $e_i < e_j'$, а в другом – наоборот $e_j' < e_i$
- Нет смысла говорить о совокупности <u>глобальных состояний вычисления</u> !!!

Конус прошлого и конус будущего для события



Логический и физический параллелизм

ITIVITI

- Логически параллельные события могут наступать в разные моменты времени
- НО! Если бы скорости выполнения процессов и задержки доставки сообщений были бы другими…
- **Результат вычисления не зависит от того, совпадает ли выполнение логически параллельных событий во времени или нет**
- Отношение логического параллелизма не транзитивно

LET IT TRADE

УНИВЕРСИТЕТ ИТМО

Свойства каналов

- Очередность
- Ёмкость
- Надежность
  - Потеря сообщения
  - Искажение сообщения
  - Дублирование сообщения
  - Спонтанное порождение

ITIVITI

LET IT TRADE

Раздел 3. Логические часы

УНИВЕРСИТЕТ ИТМО

Recap

- Мир непрост
- Отношение «произошло раньше» введено не опираясь на понятие единого времени
- Отношение «произошло раньше» - отношение *частичного порядка*: с точки зрения процессов невозможно понять, какая последовательность событий происходит на самом деле
- Множество эквивалентных выполнений = вычисление: определяется множеством *событий* и действующим на нем причинно-следственным *порядком*

Логические часы

ITIVITI

- Часы vs время?
- Отношение «произошло раньше» введено не опираясь на понятие единого времени
- Логические часы позволяют отслеживать причинно-следственную зависимость между событиями
- Логическое время не течет само по себе
- Логическое время дискретно

LET IT TRADE

УНИВЕРСИТЕТ ИТМО

Общие принципы

ITIVITI

LET IT TRADE

- Что значит «событие произошло в 9:30»?
- Логические часы – функция Θ, отображающая множество событий ***C*** в некоторое упорядоченное множество (T, <)
- Θ_i - часы процесса P_i
- $\Theta(e_i) = \Theta_i(e_i)$ для всех событий e_i процесса P_i

УНИВЕРСИТЕТ ИТМО

Общие принципы

- Непротиворечивые часы:

$$\forall e_i, e_j' \in \boldsymbol{C}: e_i \rightarrow e_j' \Rightarrow \Theta(e_i) < \Theta(e_j')$$

- **Условие 1**: если e_i и e_i' события процесса P_i, и e_i наступает раньше события e_i', то $\Theta(e_i) < \Theta(e_i')$
- **Условие 2**: если e_i и e_j' взаимосвязанные события отправки и получения сообщения из P_i в P_j, то $\Theta(e_i) < \Theta(e_j')$
- Строго непротиворечивые часы:

$$\forall e_i, e_j' \in \boldsymbol{C}: e_i \rightarrow e_j' \Leftrightarrow \Theta(e_i) < \Theta(e_j')$$

Общие принципы: реализация

ITIVITI

LET IT TRADE

- Структура данных в каждом процессе для представления логического времени из *T* и хранения текущих показаний своих часов
- Метод продвижения логического времени для выполнения условия непротиворечивости

УНИВЕРСИТЕТ ИТМО

Общие принципы: реализация

- Составляющие часов процесса:
  - Логические локальные часы: то, что процесс P_i знает сам о своем выполнении
  - Логические глобальные часы: то, что процесс P_i знает о ходе выполнения других процессов
- Правила продвижения:
  - **Правило 1**: как P_i изменяет показания своих локальных часов при наступлении любого события
  - **Правило 2**: как P_i изменяет показания своих глобальных часов для отражения своего представления о ходе выполнения других процессов

Скалярные часы Л. Лэмпорта

- Для линейного упорядочивания событий
- T – множество неотрицательных целых чисел
- Локальное время процесса P_i и его представление о глобальном времени выражается одной скалярной величиной L_i
- $L(e_i)$ – отметка времени события e_i

Скалярные часы: правила работы

- **Правило 1**: перед выполнением любого события P_i увеличивает показания своих часов L_i:

$$L_i = L_i + d, \text{ где } d > 0$$

- Обычно $d = 1$, L_i инициализируется нулем

Скалярные часы: правила работы

- **Правило 2**: С каждым сообщением передается логическое время отправителя L_{msg} на момент отправки этого сообщения

  Когда P_j принимает сообщение с отметкой времени L_{msg}, он выполняет следующие шаги:

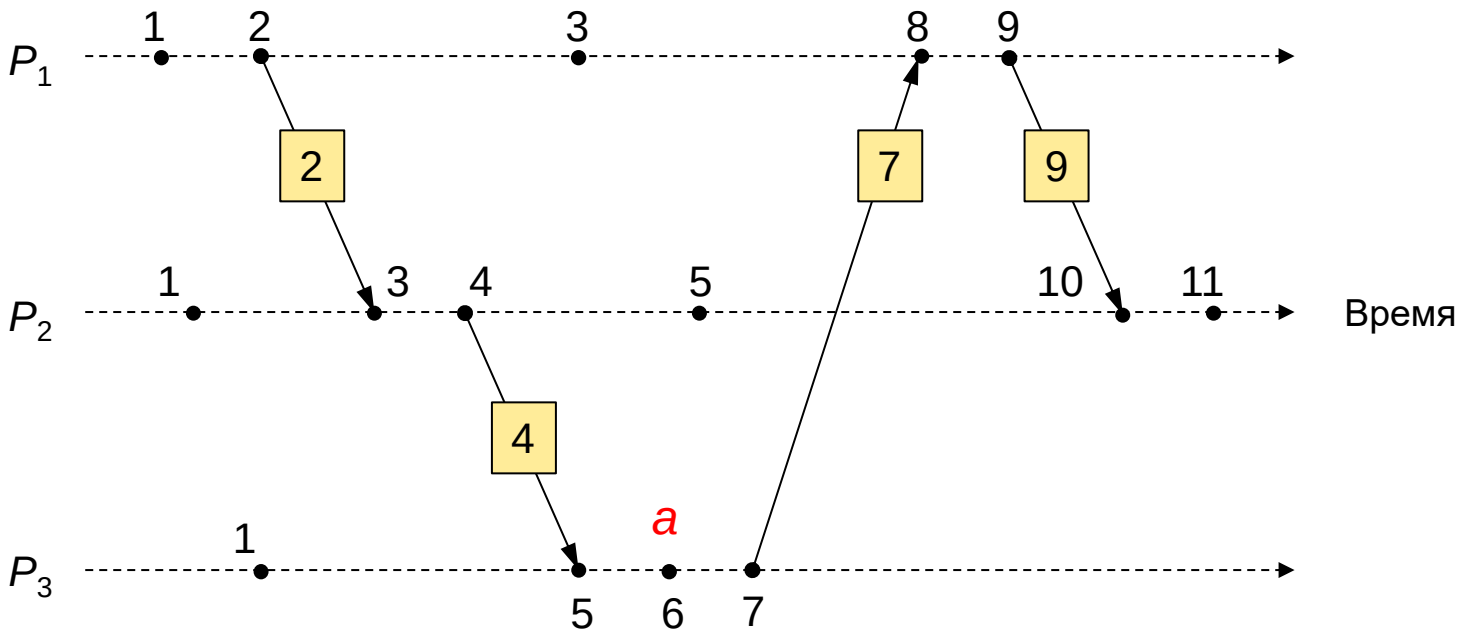
1. $L_j = \max(L_j, L_{msg})$
2. исполняет **Правило1**
3. доставляет сообщение и приступает к его обработке

Скалярные часы

ITIVITI

LET IT TRADE

Пример работы алгоритма скалярных часов:



УНИВЕРСИТЕТ ИТМО

Скалярные часы: линейное упорядочивание событий

ITIVITI

- Можно ввести отношение линейного порядка <, сохраняющее частичный порядок событий → (просто по отметкам времени)
- Основная проблема – два или более событий могут иметь одинаковую отметку времени
- Нужен линейный порядок среди процессов, например, на основе ID

$$e_i < e_j' \Leftrightarrow (L(e_i) < L(e_j')) \vee ((L(e_i) = L(e_j')) \wedge (i < j))$$

- $L(e_i) = L(e_j') \Rightarrow e_i \parallel e_j'$ поэтому < не нарушает →
- Определяет одно из эквивалентных выполнений

LET IT TRADE

УНИВЕРСИТЕТ ИТМО

Скалярные часы: подсчет событий

ITIVITI

LET IT TRADE

- При $d = 1$: $L(e) - 1$ определяет количество событий, которые должны произойти последовательно до наступления e, независимо от процессов, в которых эти события происходят
- $L(e) - 1$ высота (*height*) события e
- См. событие a

УНИВЕРСИТЕТ ИТМО

Скалярные часы: непротиворечивость

ITIVITI

LET IT TRADE

- Непротиворечивость – по построению
- Отсутствие строгой непротиворечивости:

$$L(e_i) < L(e_j') \not\Rightarrow e_i \rightarrow e_j'$$

- Иначе $e_i \parallel e_j' \Rightarrow L(e_i) = L(e_j')$, но может быть ситуация, когда $(e_i \parallel e_j') \wedge (e_j' \parallel e_k'')$, но $e_i \rightarrow e_k''$
- Отсутствие строгой непротиворечивости – основной недостаток скалярных часов

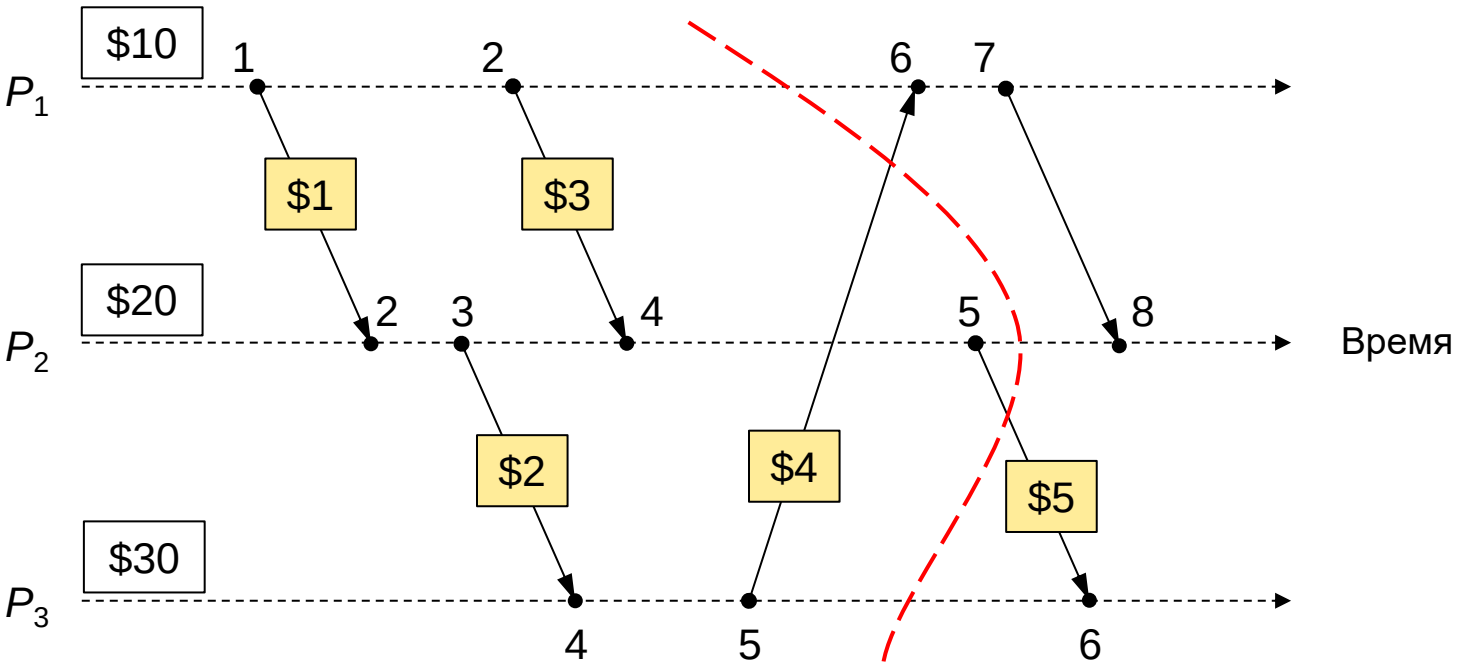
УНИВЕРСИТЕТ ИТМО

Скалярные часы: банковская система

- Каждый процесс отсылает бесконечно много сообщений каждому другому процессу
- Каналы обладают свойством FIFO
- Будем определять сумму денег в момент логического времени $t \in Z_{\geq 0}$
- P_i записывает состояние своего счета перед выполнением первого события e_i: $L(e_i) > t$
- Начиная с этого события P_i записывает все поступающие ему сообщения: $L_{msg} \leq t$

Скалярные часы: банковская система

Пример выполнения ($t = 5$):



Скалярные часы: банковская система

ITIVITI

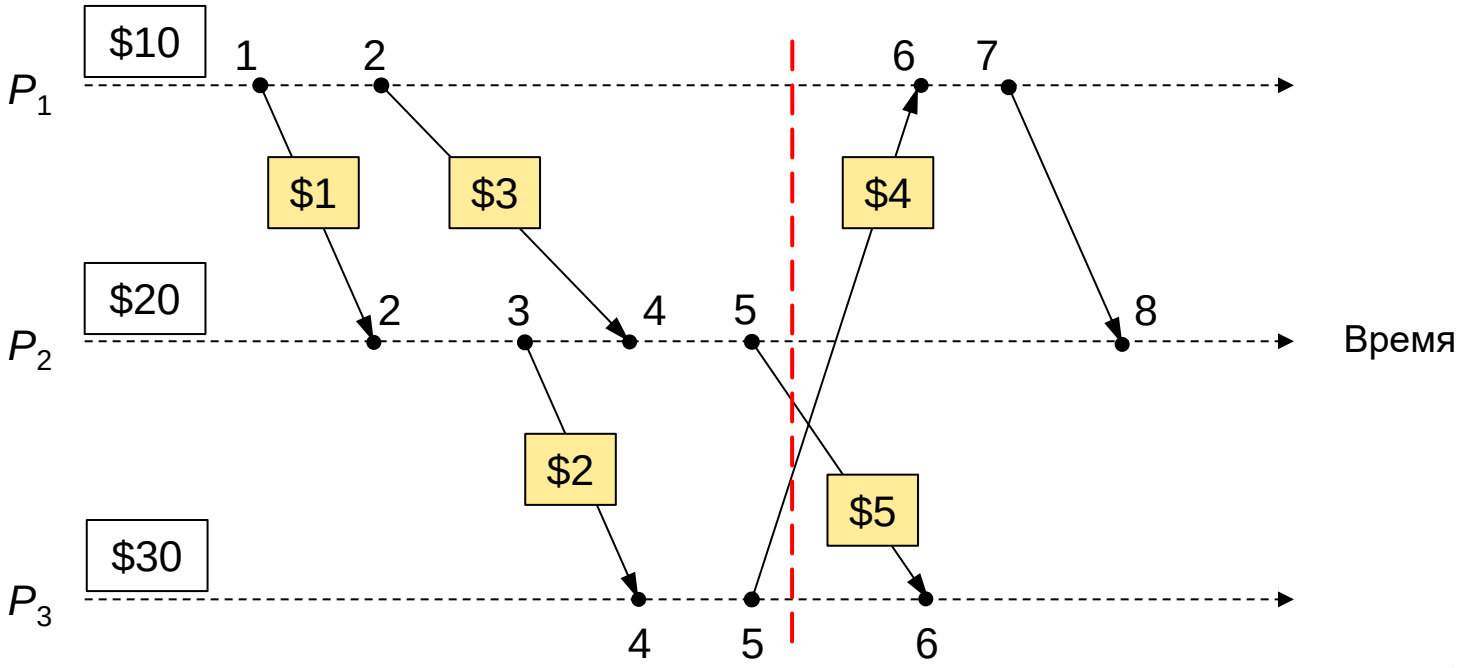
LET IT TRADE

Пример выполнения ($t = 5$):

- P_1: \$10 – \$1 – \$3 = \$6
- P_2: \$20 + \$1 – \$2 + \$3 – \$5 = \$17
- P_3: \$30 + \$2 – \$4 = \$28
- C_{31}: \$4
- C_{23}: \$5
- Итого: \$6 + \$17 + \$28 + \$4 + \$5 = \$60

Скалярные часы: банковская система

Эквивалентное выполнение ($t = 5$):



ITIVITI

LET IT TRADE

Спасибо за внимание!

УНИВЕРСИТЕТ ИТМО