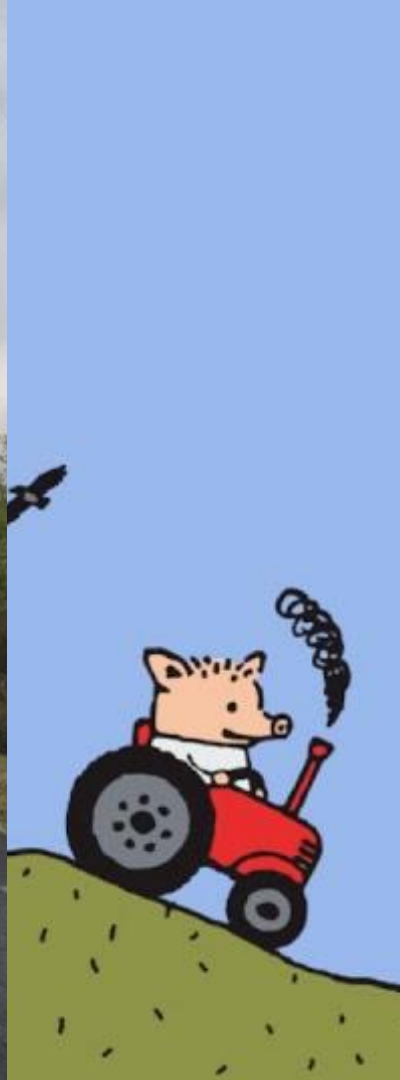

Luxoft
A DXC Technology Company
Aeron.
High performance транспорт
для low latency микросервисов
Иван Землянский
think.
create.
accelerate.


1 рубль
—ТЫ ЖЕЛАНИЕ УСПЕЛ ЗАГАДАТЬ?

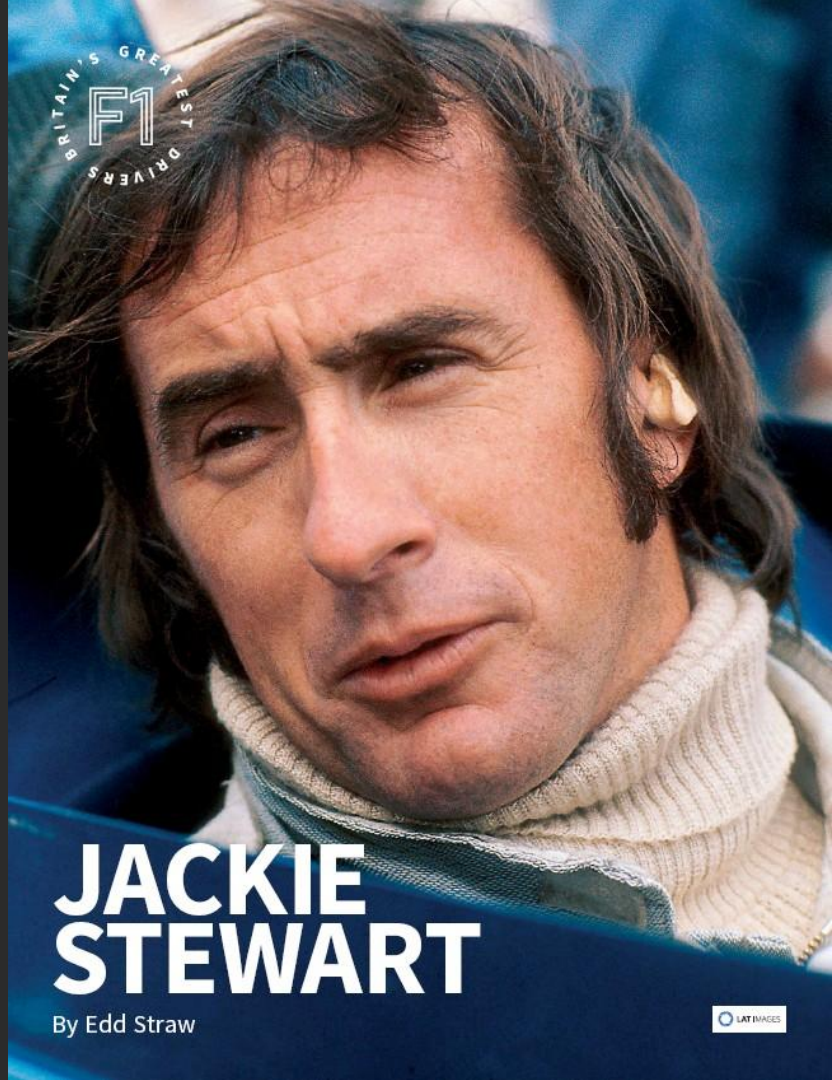

BF-790-GE




POLICE
SEACREST COUNTY
911
755
POLICE


PETRONAS
MISSION
16
5

“You don’t have to be an engineer to be a racing driver, but you do have to have Mechanical Sympathy.”

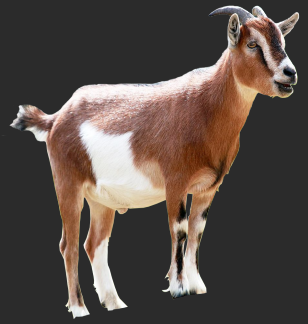


Luxoft
A DXC Technology Company

Роль музыкальных инструментов в жизни домашних парнокопытных

Luxoft
A DXC Technology Company

Роль музыкальных инструментов в жизни домашних парнокопытных


+

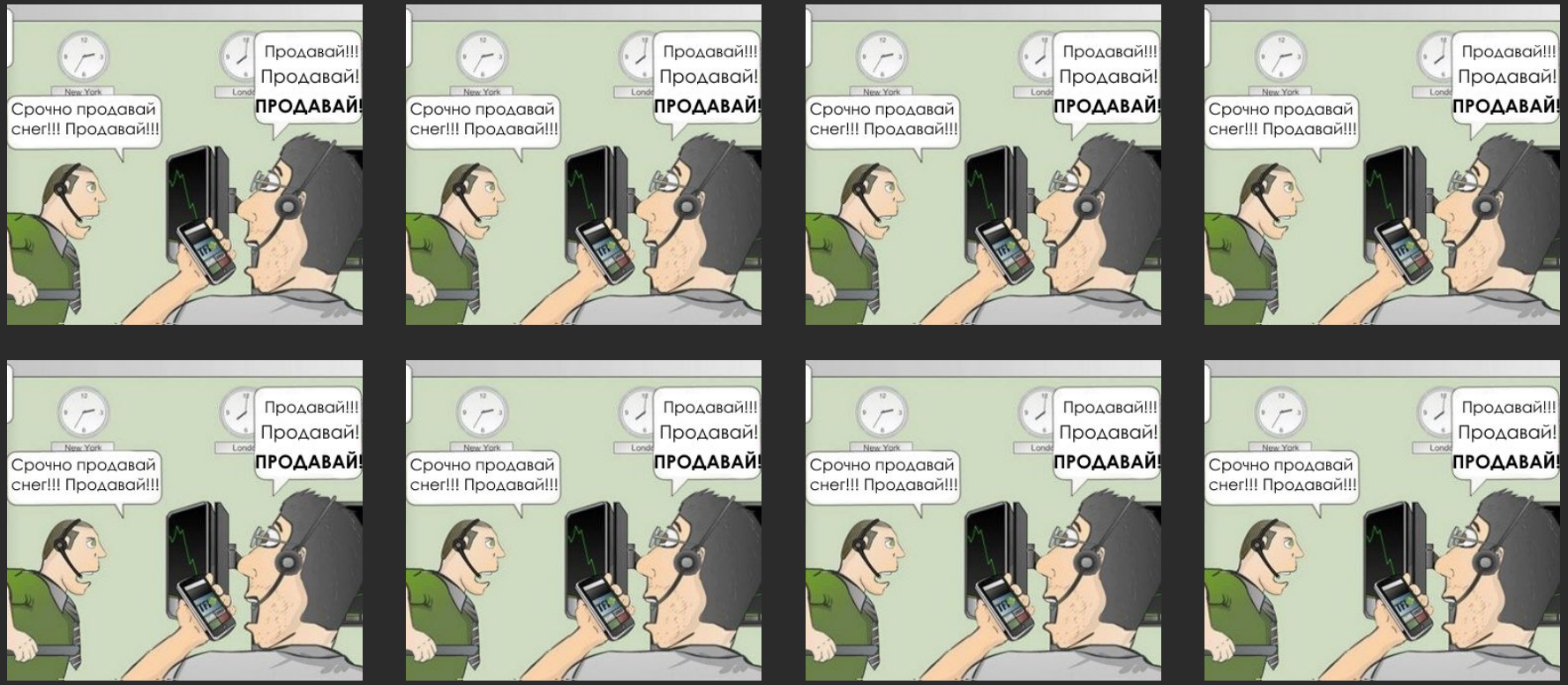
= ?

Luxoft
A DXC Technology Company

Почему важен performance?

Luxoft
A DXC Technology Company


Снег падает...
New York
Срочно продавай снег!!! Продавай!!!
Продавай!!!
Продавай!
ПРОДАВАЙ!

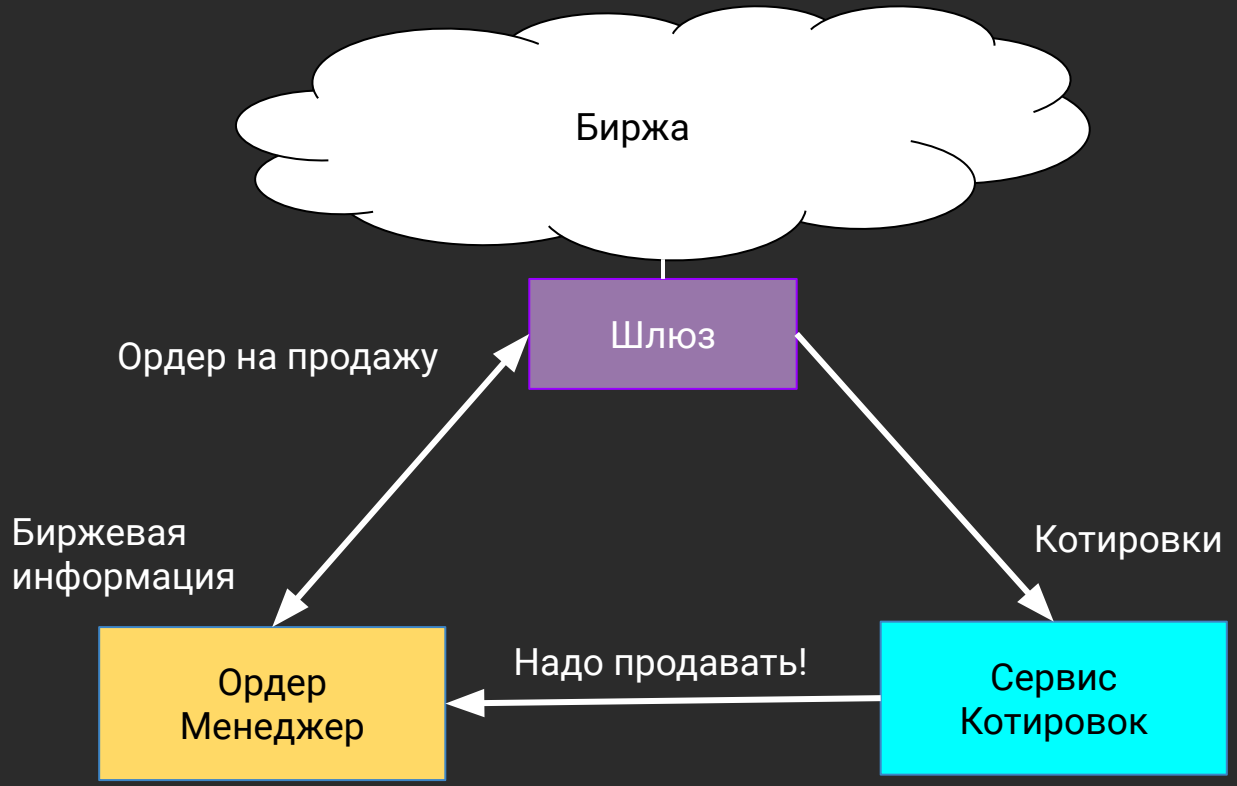

New York
Срочно продавай снег!!! Продавай!!!
Продавай!!!
Продавай!
ПРОДАВАЙ!


290
285
280
275
270
265
260
255
250
245
240
235.27
230
225
220
215
210
205
200
195
190
185


Надо продавать!
Отправили ордер
на продажу
290
285
280
275
270
265
260
255
250
245
240
235.27
230
225
220
215
210
205
200
195
190
185

Требования в цифрах

- Latency
  - 99% = 100 микросекунд
  - Max не более 100 миллисекунд
- ~ 1 000 000 сообщений в день
- Uptime во время работы биржи близкий к 100%



Luxoft
A DXC Technology Company

Spring Boot Http. Ping (микросекунды)

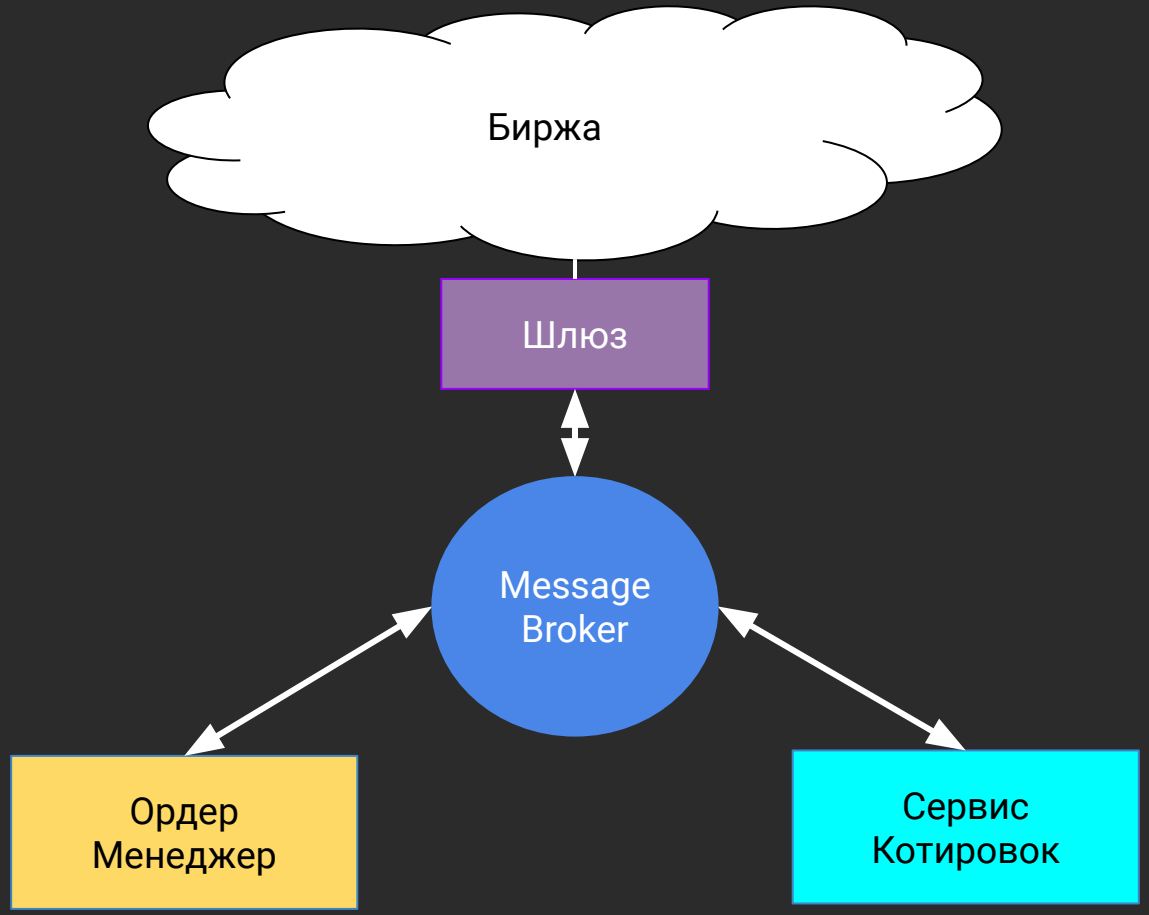
Luxoft
A DXC Technology Company

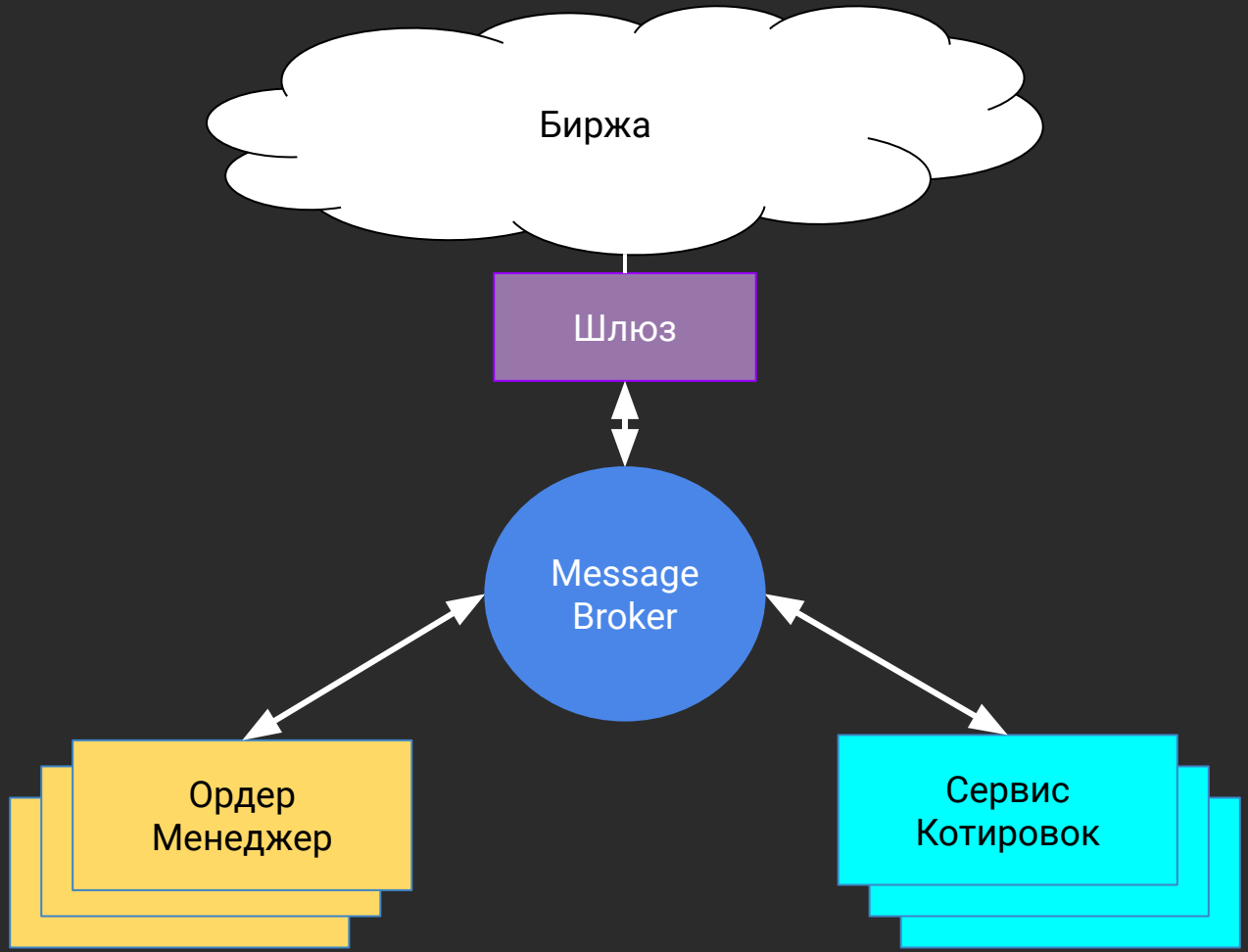
Percentile	50	90	99	max
time	211.711	301.055	587.775	6471.679

* Для localhost
https://github.com/easy-logic/transport-benchmarks


Надо продавать!
Щащаща, конвертнём объект в json
Так, отсылаем по http
Упс,
случился GC
Получили
запрос!
Отправили ордер
на продажу
290
285
280
275
270
265
260
255
250
245
240
235.27
230
225
220
215
210
205
200
195
190
185


БИЗОН
БИЗОН
7



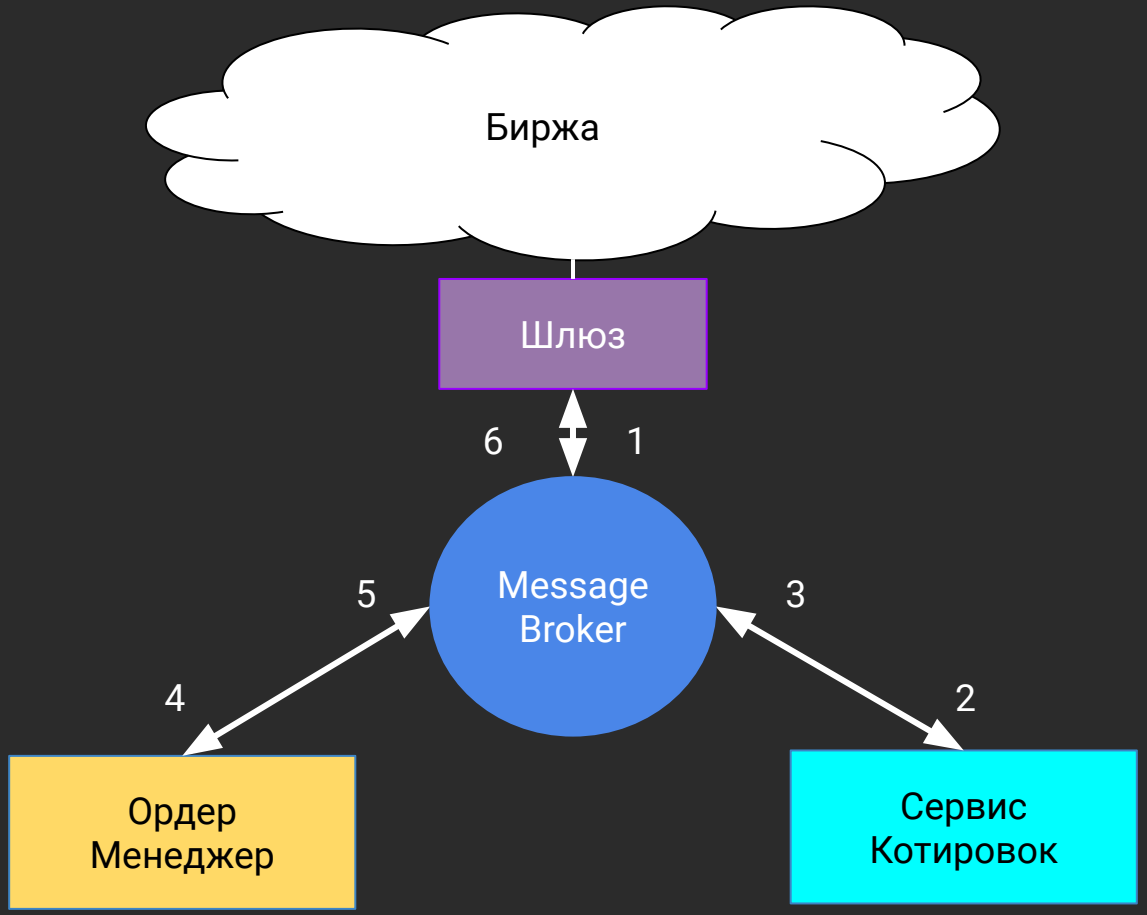


RabbitMQ. Ping (микросекунды)

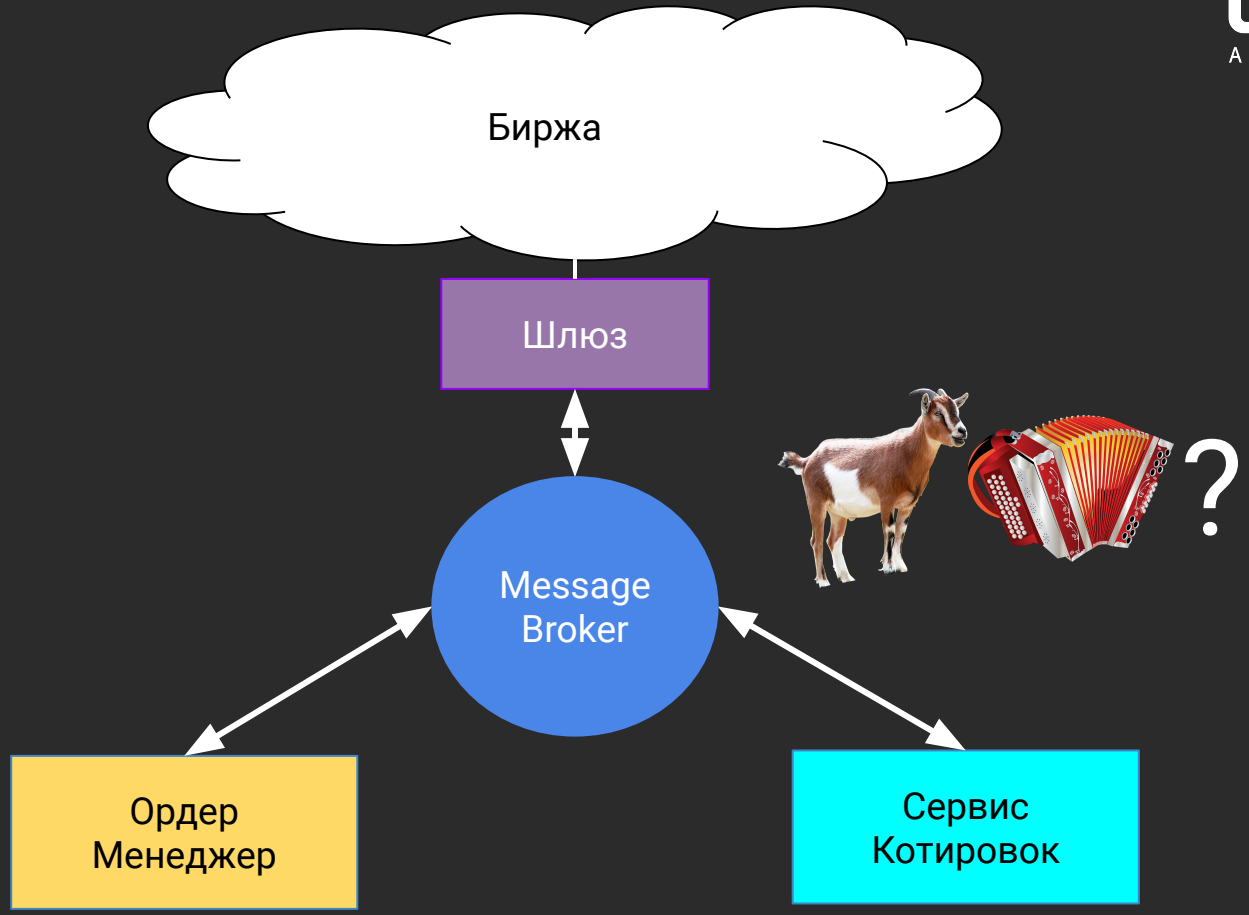
Luxoft
A DXC Technology Company

Percentile	50	90	99	max
time	193.09	211.39	297.60	21749.76

* Для localhost
https://github.com/easy-logic/transport-benchmarks

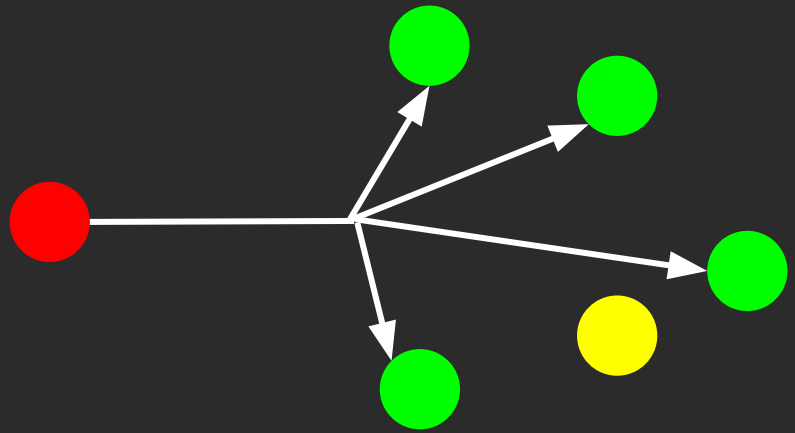

Биржа
Шлюз
6
1
5
Message
Broker
3
4
2
Ордер
Менеджер
Сервис
Котировок

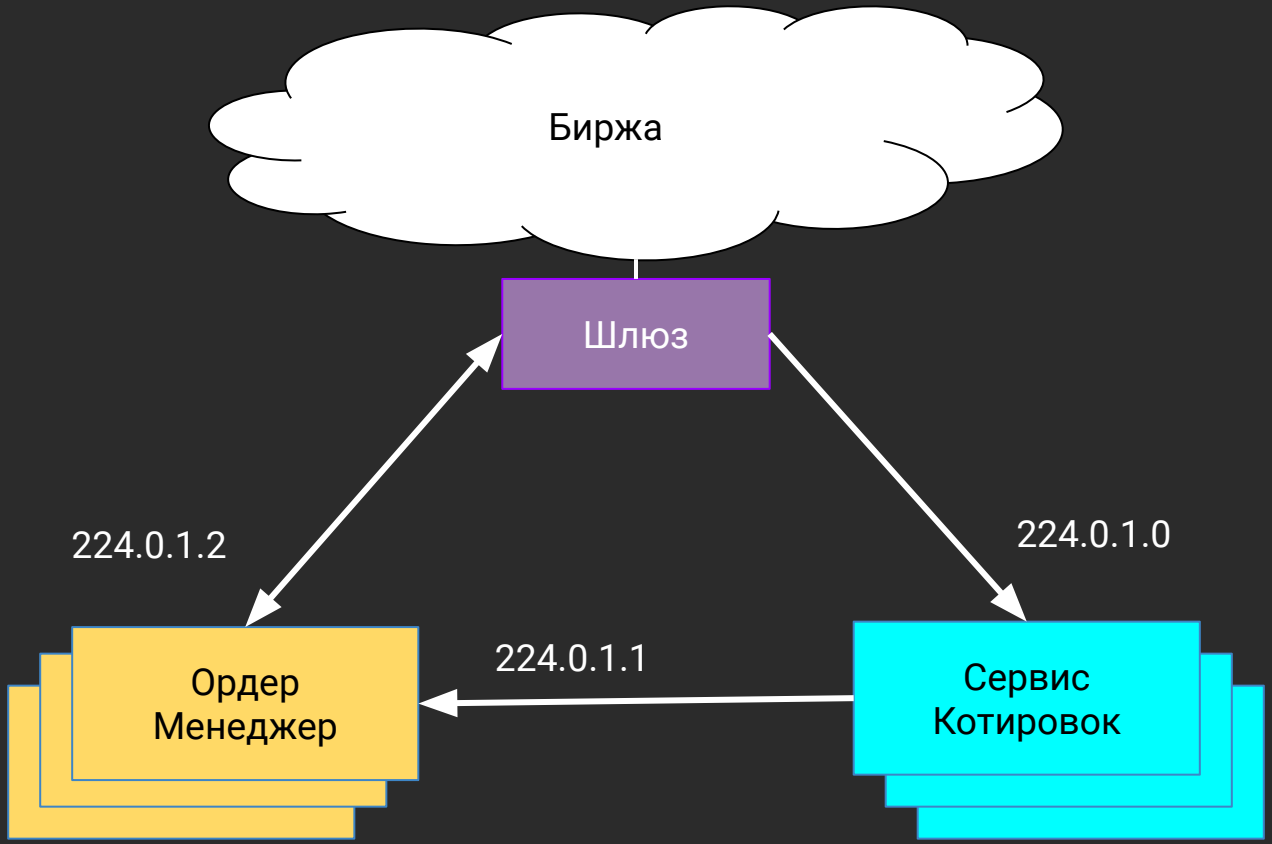
Luxoft
A DXC Technology Company



Luxoft
A DXC Technology Company

UDP Multicast?





Real Logic

Luxoft
A DXC Technology Company

Aeron

Aeron. Ping Pong (микросекунды)

Luxoft
A DXC Technology Company

Percentile	50	90	99	max
time	9.89	12.68	15.615	3395.58

* для мультикаст адреса 224.0.1.1:40457
https://github.com/easy-logic/transport-benchmarks

План

- Узнать что такое Aeron и области его применения
- Посмотреть на API
- Разобраться в архитектуре Aeron’a

Luxoft
A DXC Technology Company

Что такое Aeron?

Что такое Aeron?

Luxoft
A DXC Technology Company

Библиотека для обмена сообщениями с возможностями:

- Reliable UDP unicast
- Reliable UDP multicast
- IPC (Inter Process Communication)

Что такое Aeron?

Ключевые особенности:

- Только фичи которые реально нужны
- Основной упор на performance
- Open-source java, C++ библиотеки
- Встроенный Flow Control
- Multiplexing

Luxoft
A DXC Technology Company

Насколько популярен Aeron?

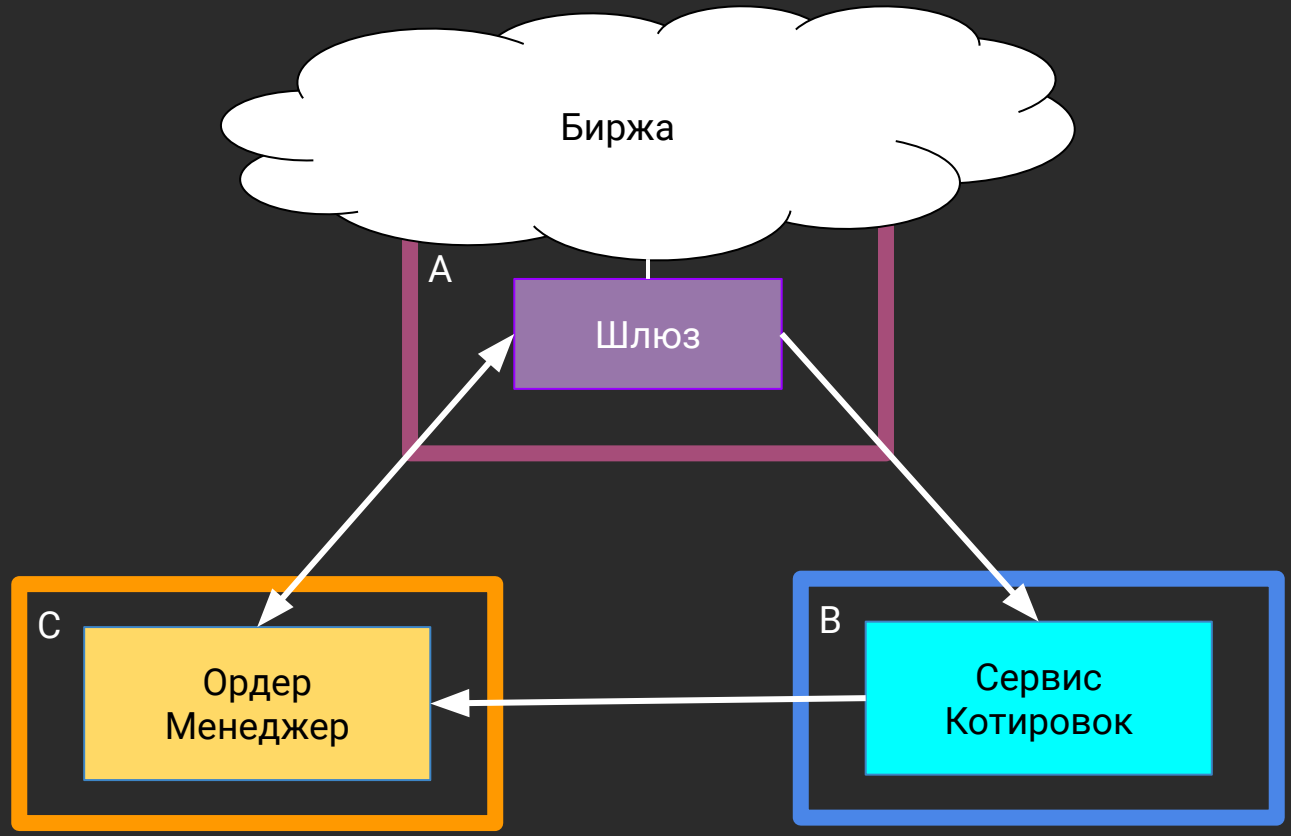
Насколько популярен Aeron?

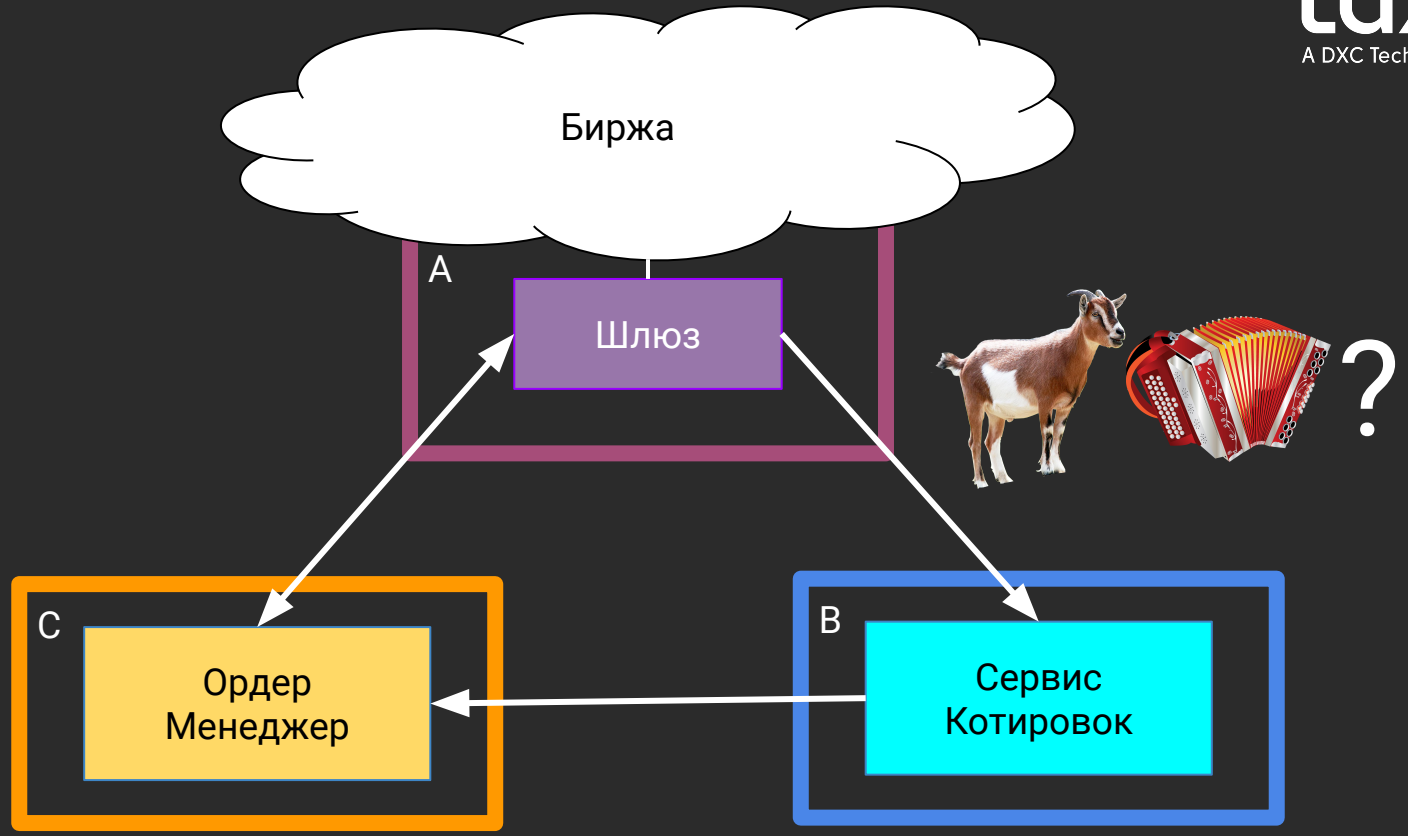
Имеют поддержку:

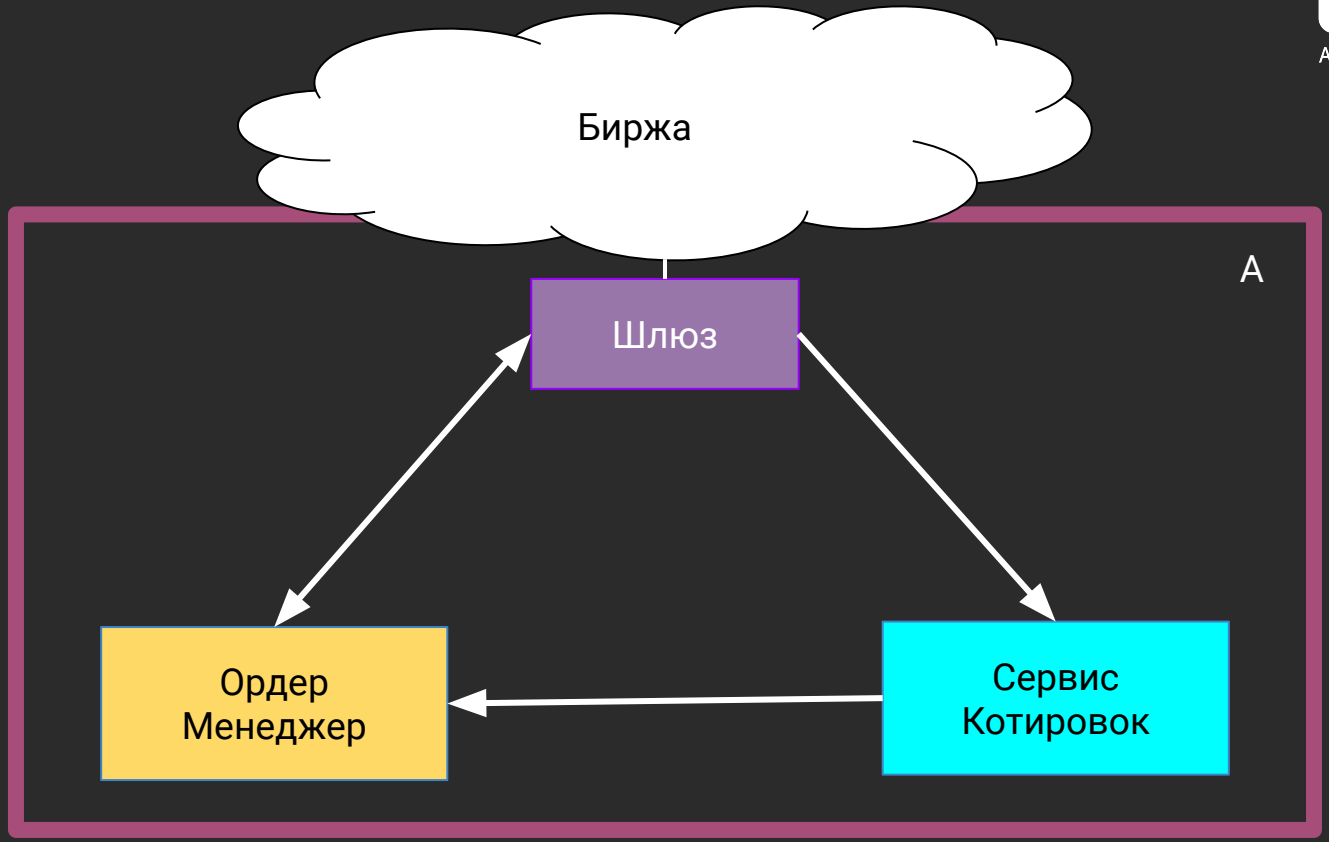
- R-socket
- Akka
- Wireshark

Luxoft
A DXC Technology Company

IPC ?


Биржа
A
Шлюз
C
Ордер
Менеджер
B
Сервис
Котировок


Биржа
A
Шлюз
?
C
Ордер
Менеджер
B
Сервис
Котировок



Aeron IPC. Ping Pong (микросекунды)

Luxoft
A DXC Technology Company

Percentile	50	90	99	max
time	0.229	0.269	0.382	837.631

* для ipc канала
https://github.com/easy-logic/transport-benchmarks

Luxoft
A DXC Technology Company

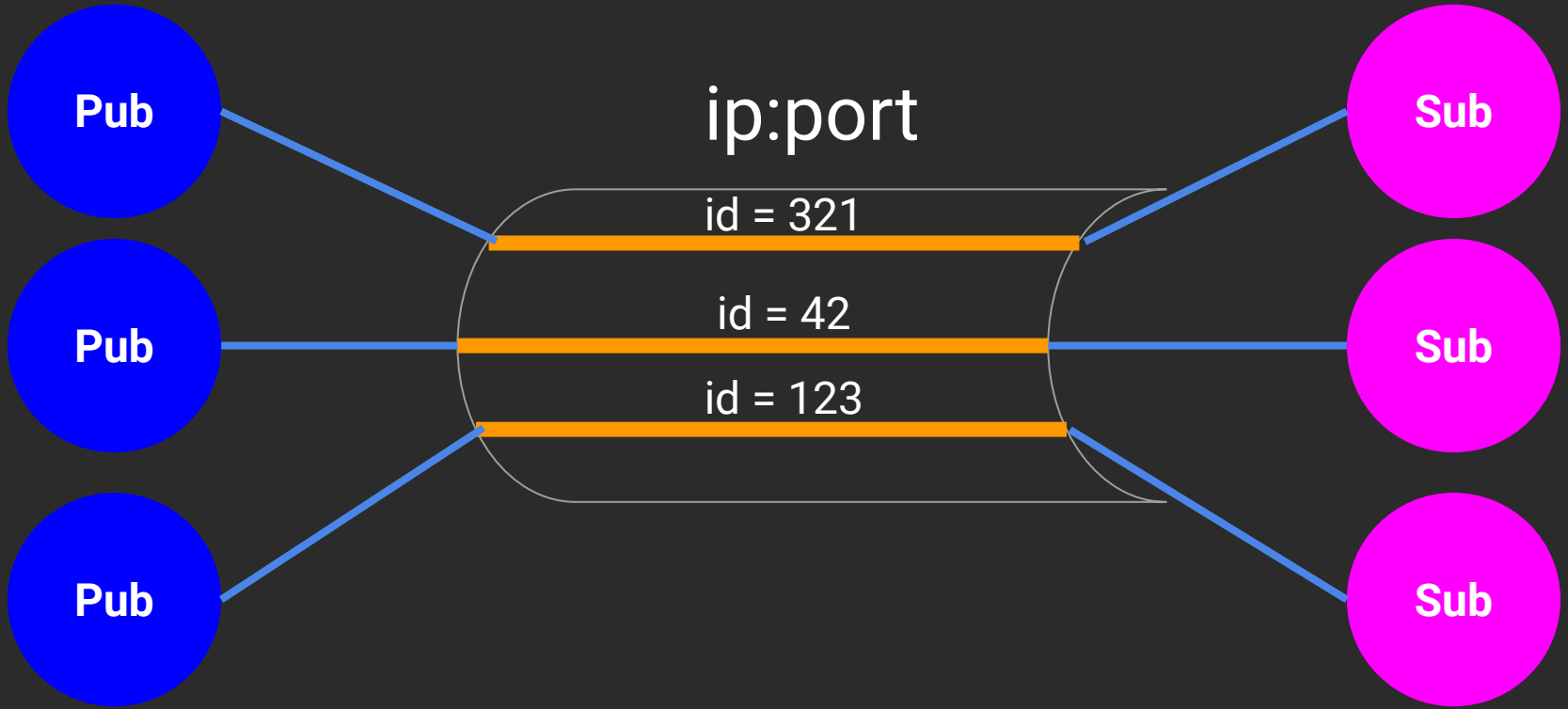
Aeron API

Aeron API. Channels & Streams

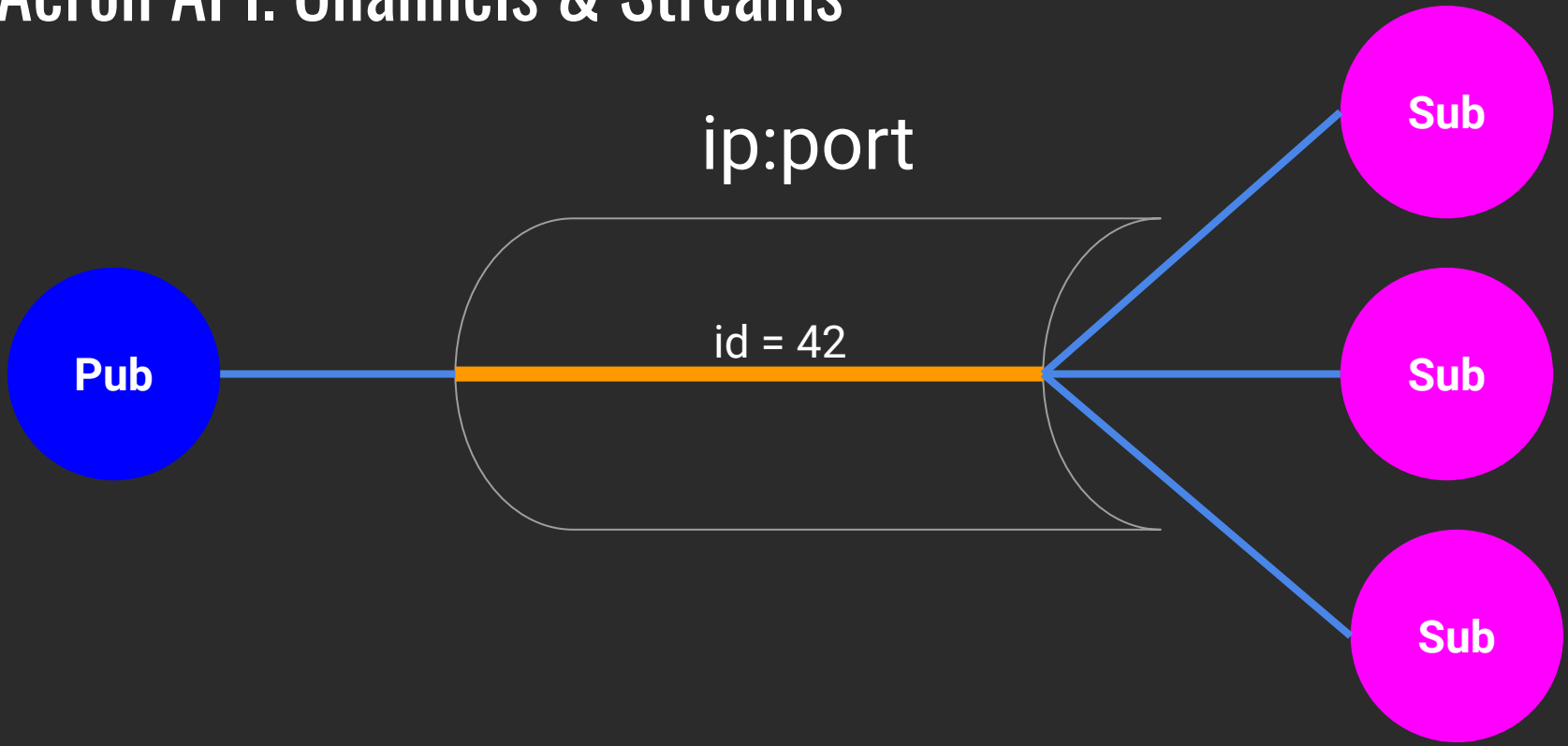
```
Aeron aeron = Aeron.connect();

Publication publication = aeron.addPublication(
    channel:  "aeron:udp?endpoint=localhost:1234",
    streamId: 42);

Subscription subscription = aeron.addSubscription(
    channel:  "aeron:udp?endpoint=localhost:1234",
    streamId: 42);
```

Aeron API. Channels & Streams

Aeron API. Channels & Streams

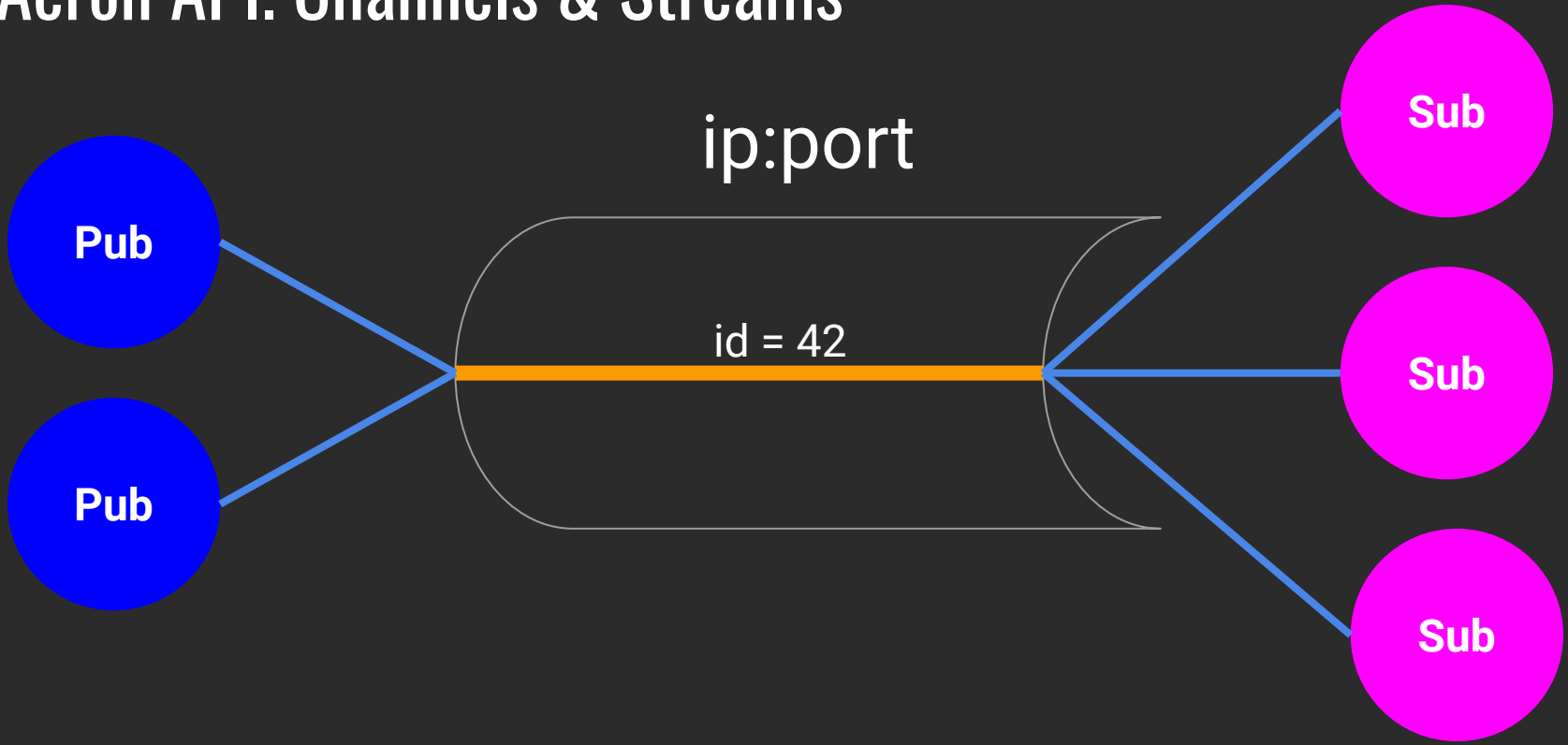


Aeron API. Channels & Streams



Aeron API. Channels & Streams

Aeron API. Channels & Streams



Aeron API. Channels & Streams

```
Aeron aeron = Aeron.connect();

Publication publication = aeron.addPublication(
    channel:  "aeron:udp?endpoint=localhost:1234",
    streamId: 42);

Subscription subscription = aeron.addSubscription(
    channel:  "aeron:udp?endpoint=localhost:1234",
    streamId: 42);
```

Aeron API. Channels & Streams

```
Aeron aeron = Aeron.connect();

Publication publication = aeron.addPublication(
    channel:  "aeron:udp?endpoint=224.0.1.1:1234",
    streamId: 42);

Subscription subscription = aeron.addSubscription(
    channel:  "aeron:udp?endpoint=224.0.1.1:1234",
    streamId: 42);
```

Aeron API. Channels & Streams

```
Aeron aeron = Aeron.connect();

Publication publication = aeron.addPublication(
    channel:  "aeron:ipc",
    streamId: 42);

Subscription subscription = aeron.addSubscription(
    channel:  "aeron:ipc",
    streamId: 42);
```

Aeron API. Publisher

```
/**
 * Non-blocking publish of a partial buffer containing a message
 *
 * @param buffer containing message.
 * @param offset offset in the buffer at which the
 *               encoded message begins.
 * @param length in bytes of the encoded message.
 */
public void offer(DirectBuffer buffer, int offset, int length)
```

Aeron API. Publisher

```
/**
 * Non-blocking publish of a partial buffer containing a message.
 *
 * @param buffer containing message.
 * @param offset offset in the buffer at which the encoded
 *               message begins.
 * @param length in bytes of the encoded message.
 * @return The new stream position,
 *         otherwise a negative error value of
 *         {@link #NOT_CONNECTED}, {@link #BACK_PRESSURED},
 *         {@link #ADMIN_ACTION}, {@link #CLOSED},
 *         or {@link #MAX_POSITION_EXCEEDED}.
 */
public long offer(DirectBuffer buffer, int offset, int length)
```


НЕЛЬЗЯ ПРОСТО ВЗЯТЬ И
ОТПРАВИТЬ СООБЩЕНИЕ В AERON

Aeron API. Publisher

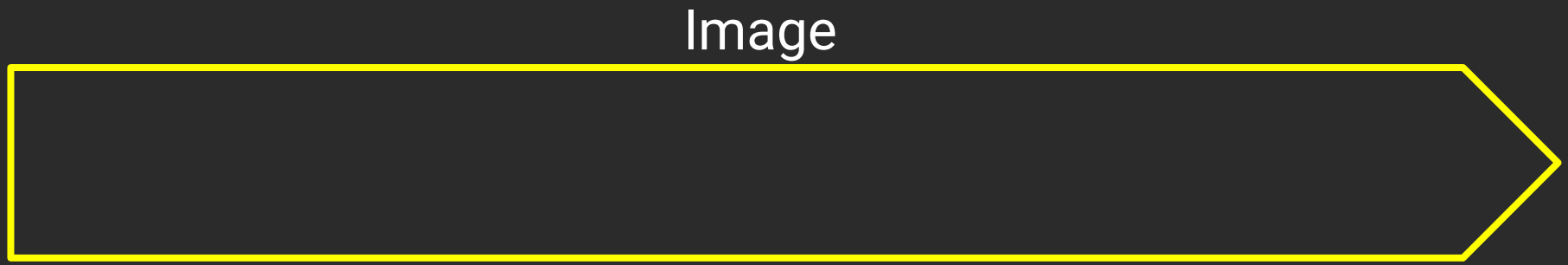
Luxoft
A DXC Technology Company

Что возвращает publisher#offer:

- `long NOT_CONNECTED = -1;`
- `long BACK_PRESSURED = -2;`
- `long ADMIN_ACTION = -3;`
- `long CLOSED = -4;`
- `Position` : следующую **позицию** в стриме данных

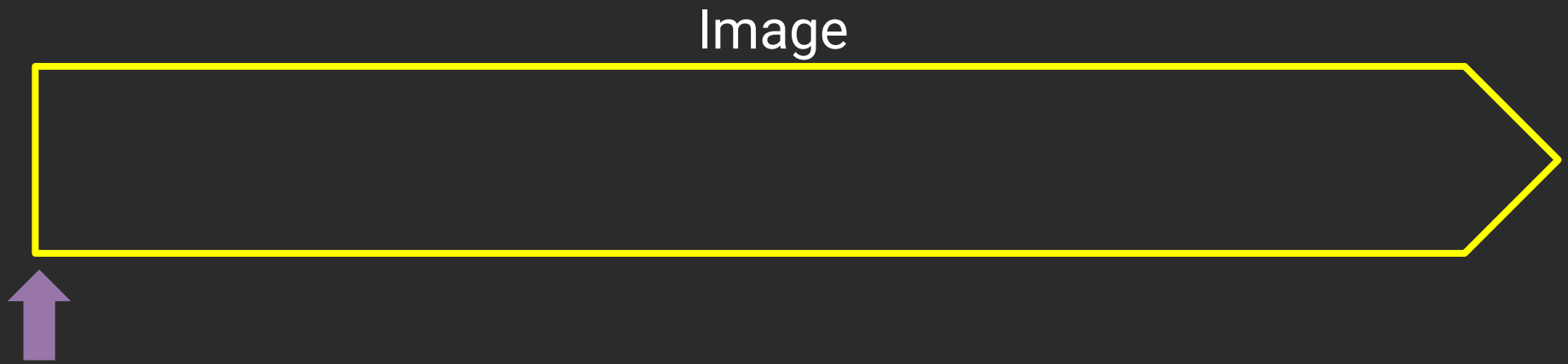
Aeron API. Position & Image

Ключевая сущность в Aeron.



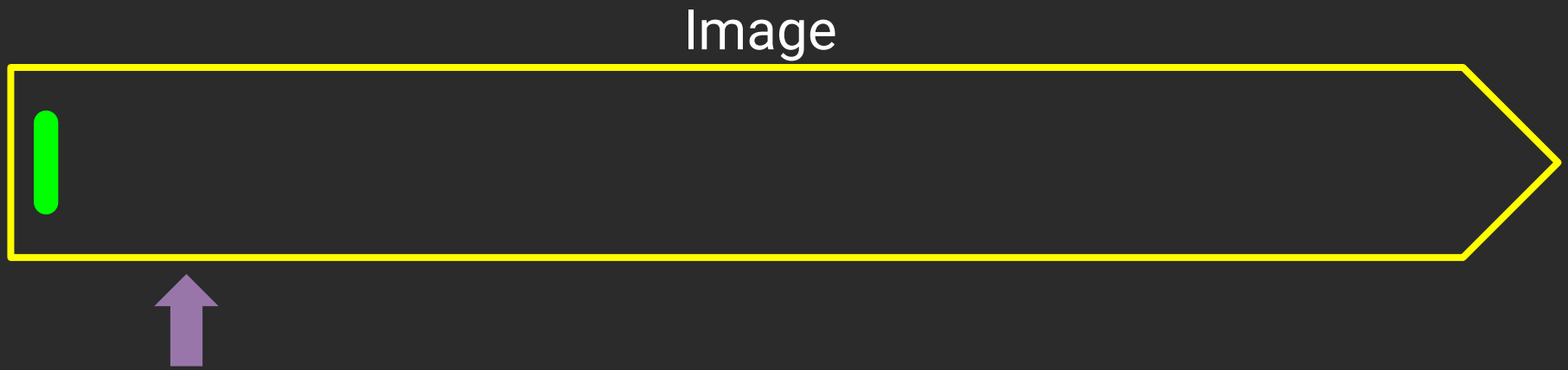
Aeron API. Position & Image

```
result = publication.offer( 100 )
```



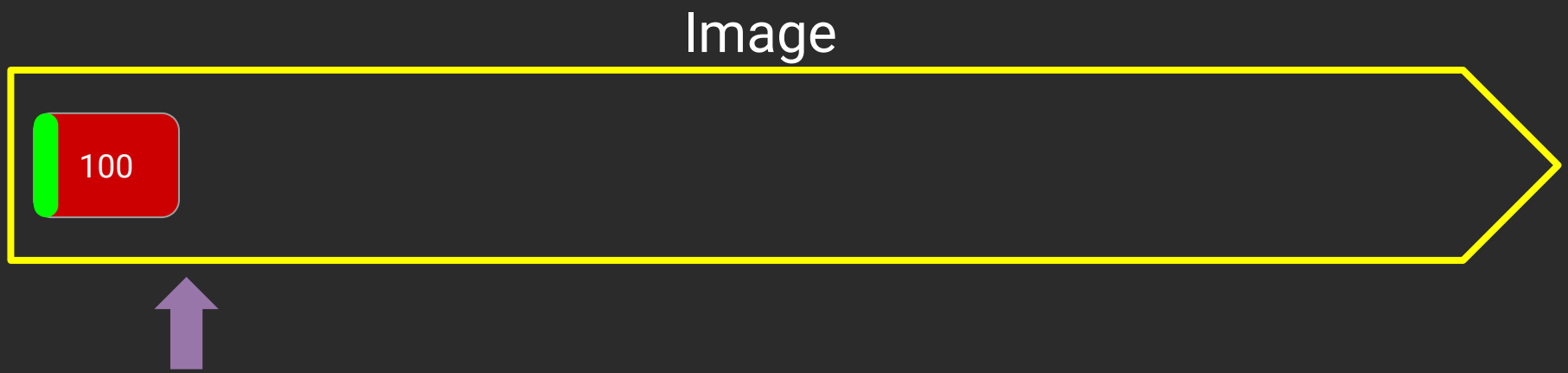
Aeron API. Position & Image

```
result = publication.offer( 100 )
```



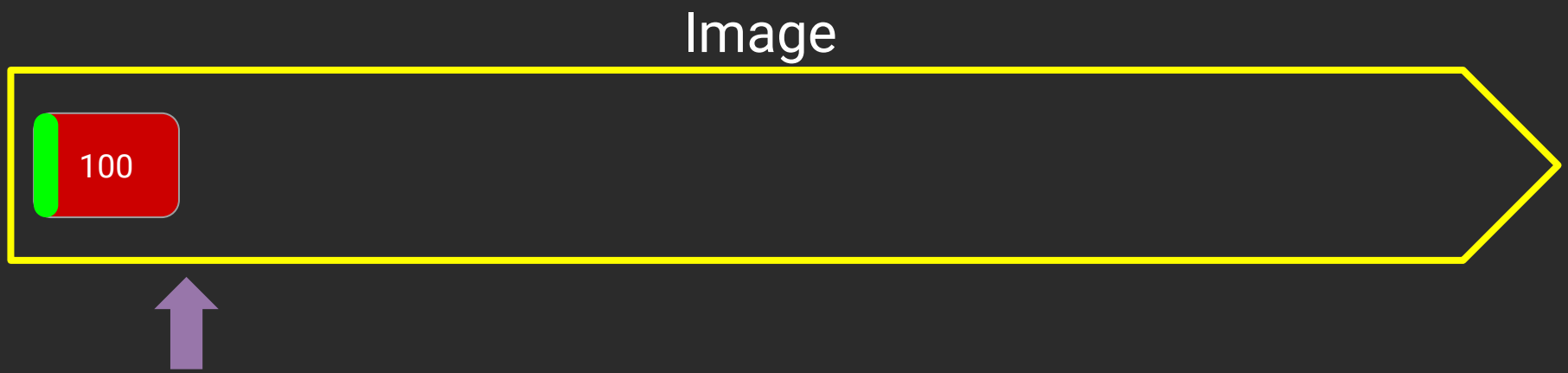
Aeron API. Position & Image

```
result = 132  // (100 + 32)
```



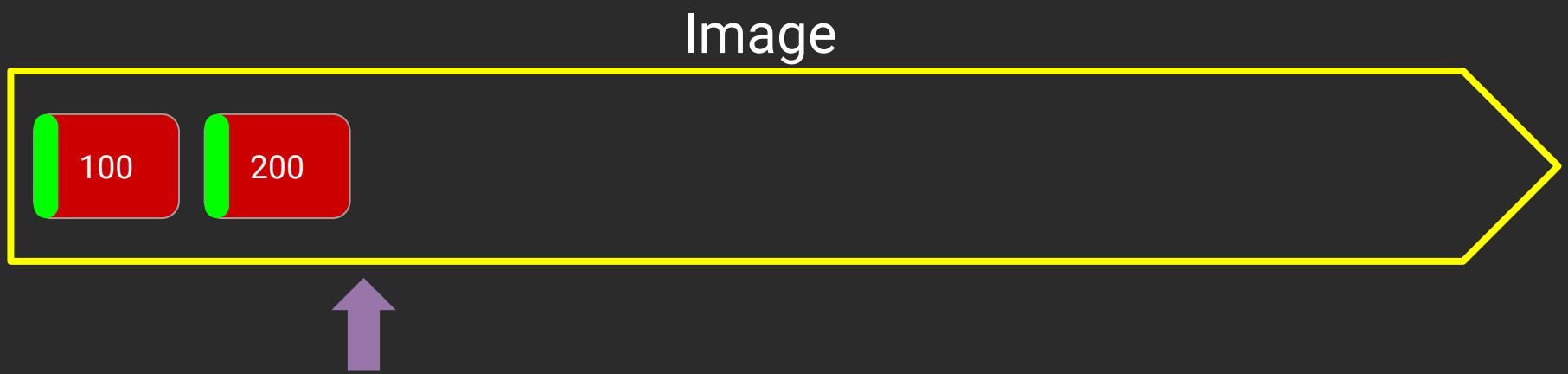
Aeron API. Position & Image

```
result = publication.offer( 200 )
```

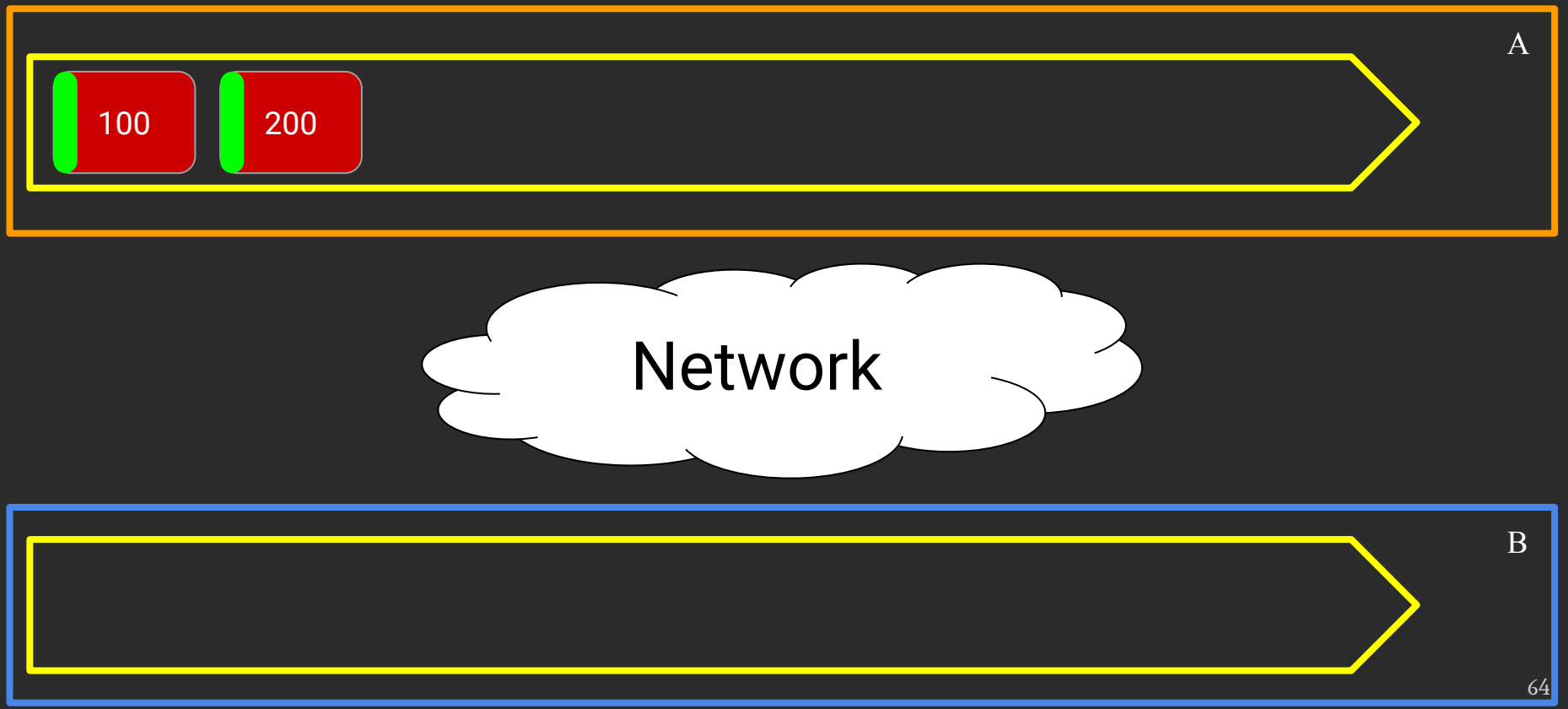


Aeron API. Position & Image

```
result = 364  //132 + 200 + 32
```



Aeron API. Position & Image



Aeron API. Position & Image

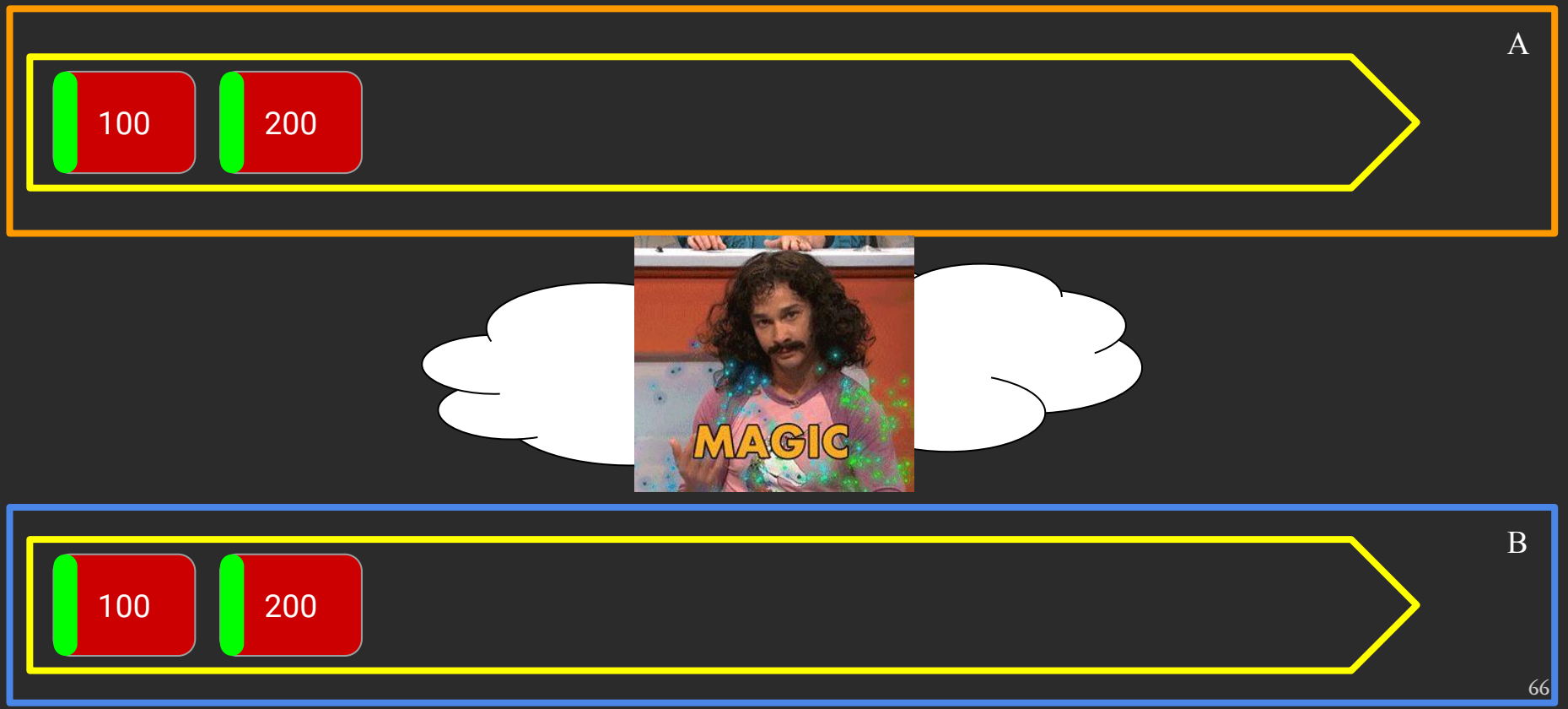
A

100

200

B

Aeron API. Position & Image

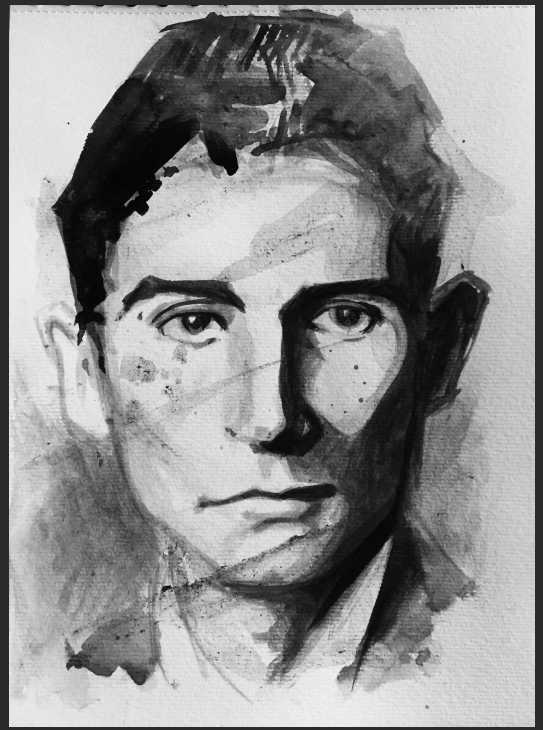


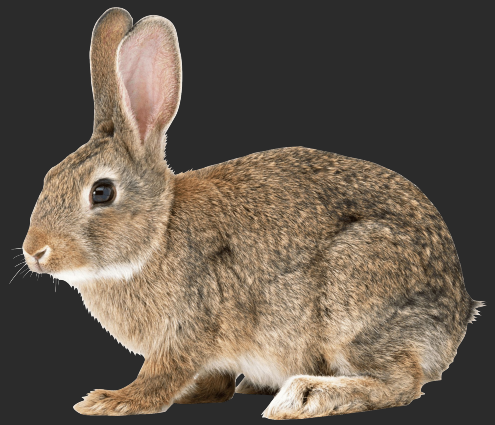
Aeron API. Subscriber

Luxoft
A DXC Technology Company

```
int poll(FragmentHandler fragmentHandler, int fragmentLimit)
{ ... }

subscription.poll(
    fragmentHandler:
    (buffer, offset, length, header) -> {
        consumeString(buffer.getStringUtf8(offset, length));
    },
    fragmentLimit:
    10);
```

Aeron API. Subscriber





Aeron API. Subscriber

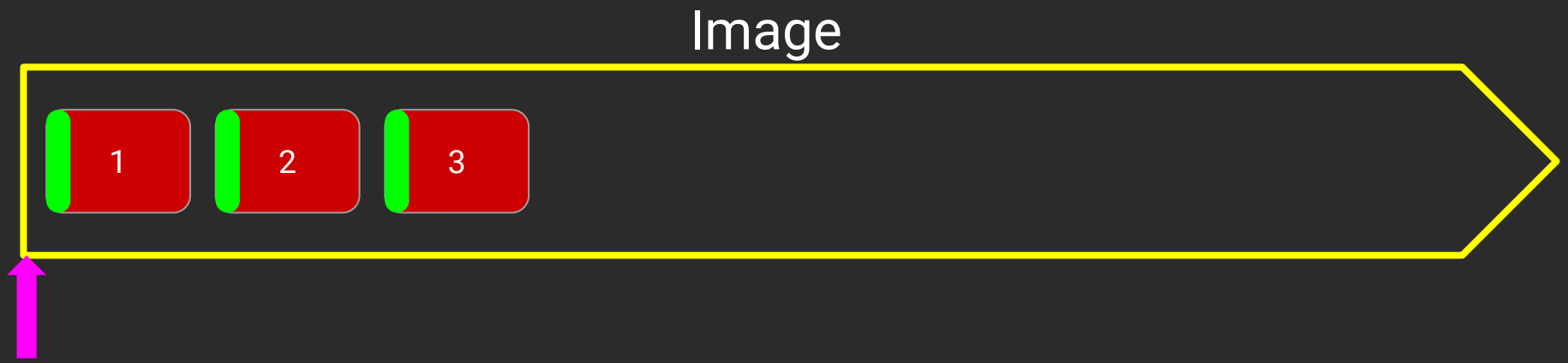
Luxoft
A DXC Technology Company

Spring Integration

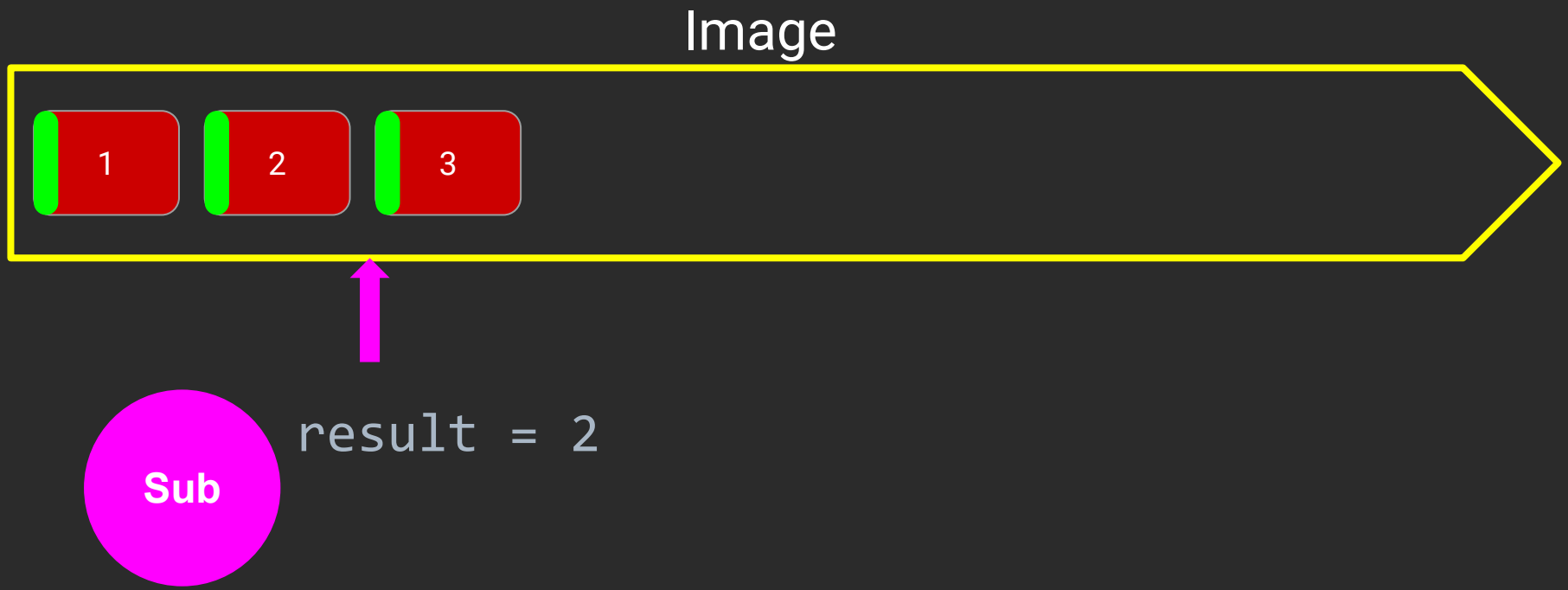
Kafka

Rabbit MQ

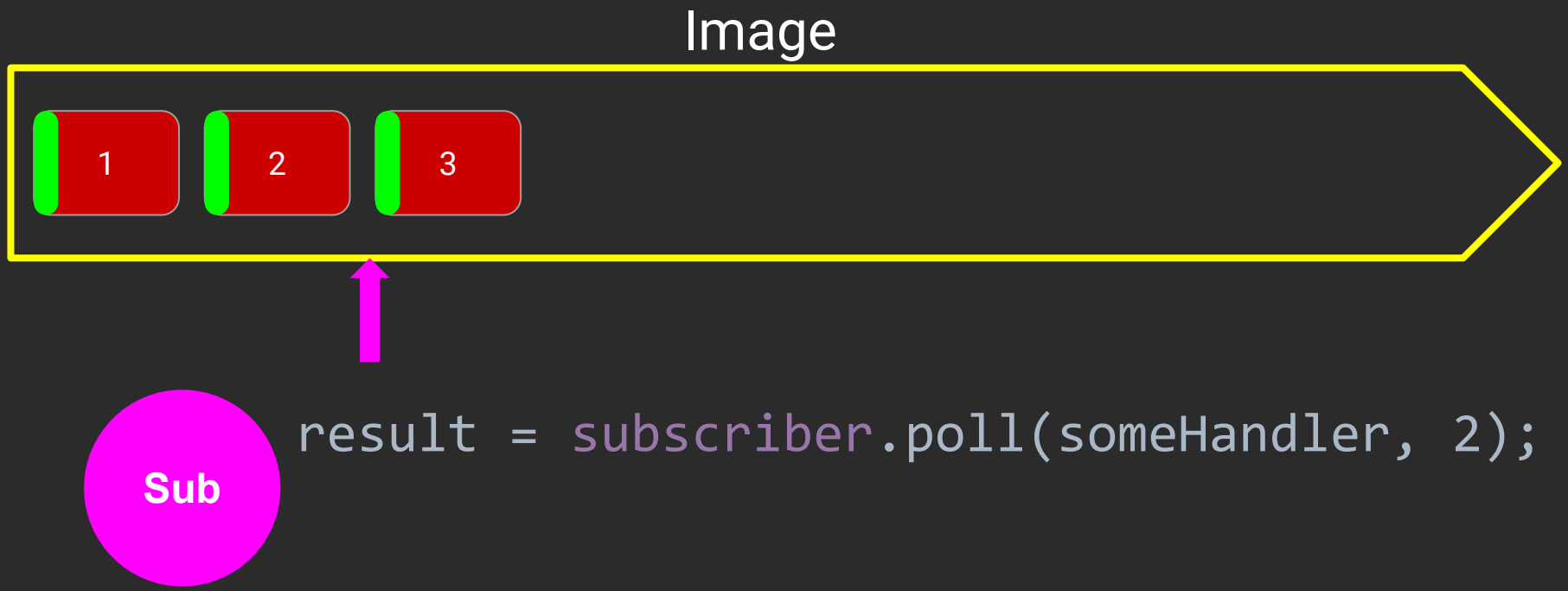
Aeron API. Subscriber. Чтение



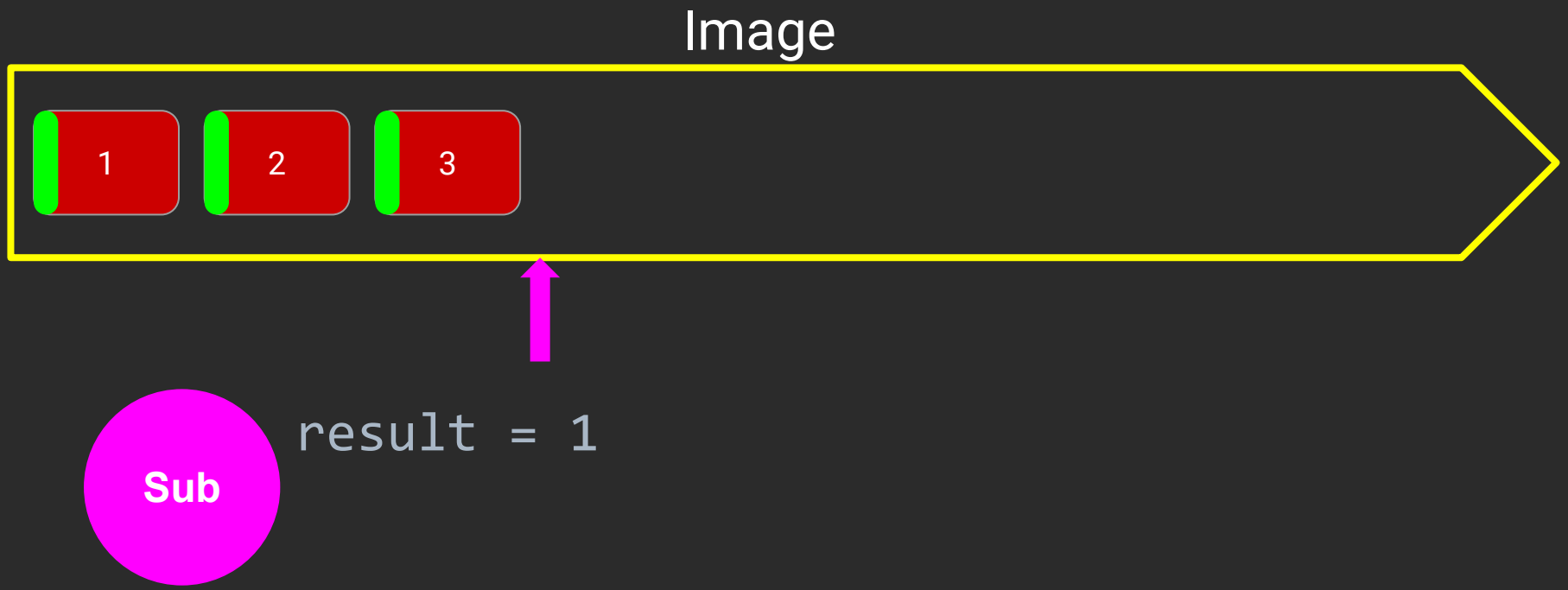
Aeron API. Subscriber. Чтение



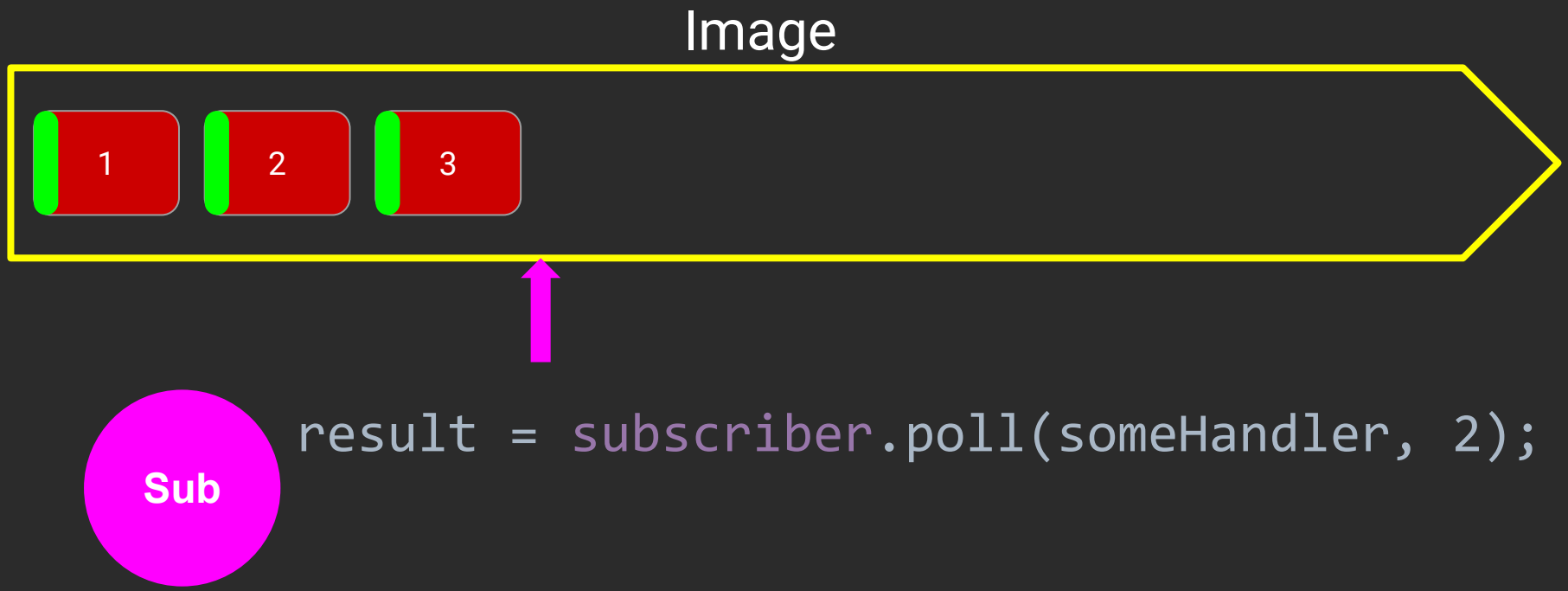
Aeron API. Subscriber. Чтение



Aeron API. Subscriber. Чтение



Aeron API. Subscriber. Чтение



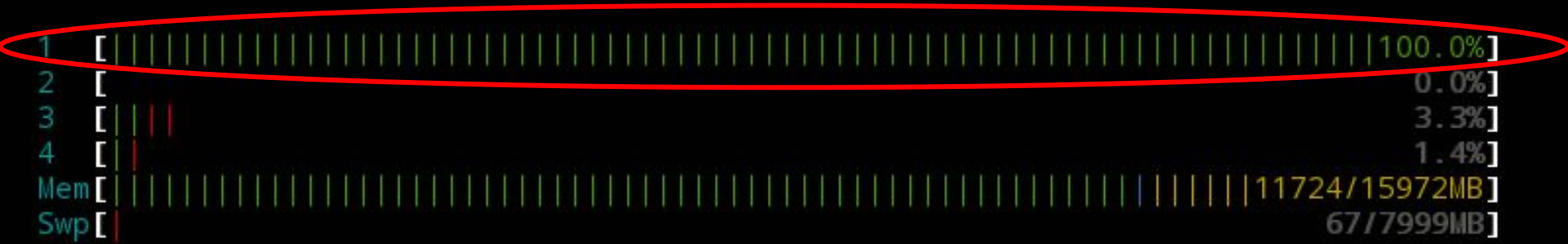
Aeron API. Subscriber. Чтение

Aeron API. Subscriber

poll - неблокирущий вызов, т.е возвращает результат сразу. По этому обычно флоу такой:

```
while (isRunning())
{
    int received = subscription.poll(dataHandler, LIMIT);
}
```

Aeron API. IdleStrategy



PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1	root	20	0	195M	4980	3176	S	0.0	0.0	0:53.92	/usr/lib/systemd/systemd -
29390	ivan	20	0	630M	22492	16488	S	0.0	0.1	0:00.97	├─ /usr/libexec/gvfsd-rece
29392	ivan	20	0	630M	22492	16488	S	0.0	0.1	0:00.02	│ ├─ gdbus
29391	ivan	20	0	630M	22492	16488	S	0.0	0.1	0:00.00	│ └─ gmain
28340	ivan	20	0	800M	43324	29612	S	0.0	0.3	0:00.44	├─ eog /home/ivan/Pictures
28347	ivan	20	0	800M	43324	29612	S	0.0	0.3	0:00.05	│ ├─ EogJobScheduler
28346	ivan	20	0	800M	43324	29612	S	0.0	0.3	0:00.00	│ ├─ gdbus
28345	ivan	20	0	800M	43324	29612	S	0.0	0.3	0:00.00	│ ├─ gmain
28344	ivan	20	0	800M	43324	29612	S	0.0	0.3	0:00.00	│ └─ dconf worker
28265	root	39	19	606M	92712	18044	S	0.0	0.6	0:01.14	├─ /usr/bin/python3 /usr/b

Aeron API. IdleStrategy

```
while (isRunning())
{
    int received = subscription.poll(dataHandler, LIMIT);
    idleStrategy.idle(received);
}
```

Aeron API. IdleStrategy

- `NoOpIdleStrategy`
- `SleepingMillisIdleStrategy` // Thread#*sleep*
- `SleepingIdleStrategy` // LockSupport#*parkNanos*
- `YieldingIdleStrategy` // Thread#*yield*
- `BusySpinIdleStrategy` // Thread#*onSpinWait*
- `BackoffIdleStrategy` // spin + yield + park

Aeron API. IdleStrategy

```
while (isRunning())
{
    int received = subscription.poll(dataHandler, LIMIT);
    idleStrategy.idle(received);
}
```

Aeron API. IdleStrategy

?

```
while (isRunning())
{
    int received = subscription.poll(dataHandler, LIMIT);
    idleStrategy.idle(received);
}
```

Luxoft
A DXC Technology Company

```
select ?
```

Aeron API. BusySpin

```
SELECT * FROM SetVsSelectDemo WHERE GroupNumber = 2
```

132 %

Results | Messages

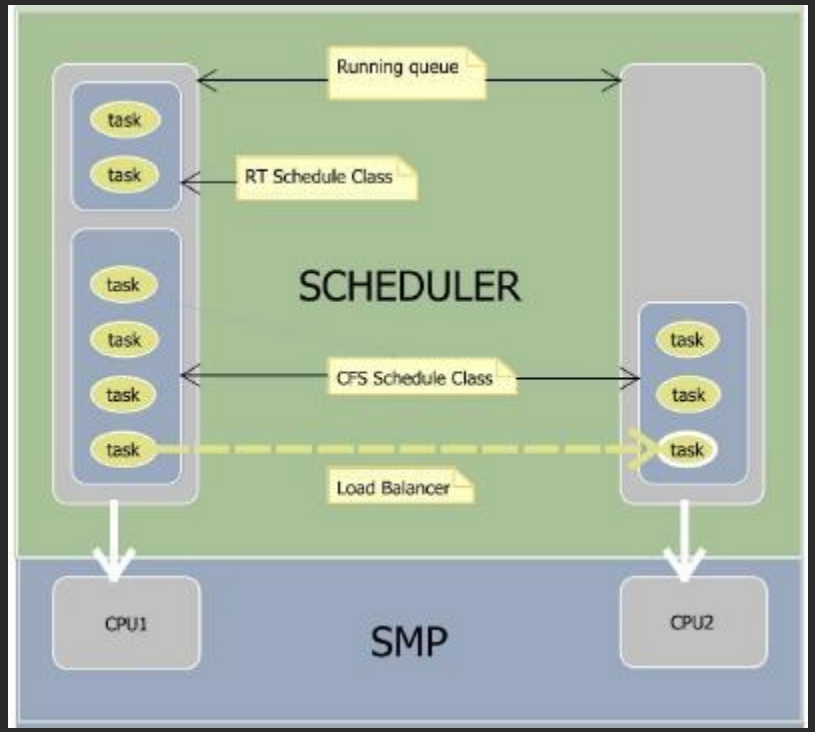
	ID	Name	GroupNumber	Grade
1	3	Huda	2	180
2	4	Zaid	2	170

SELECT здорового человека

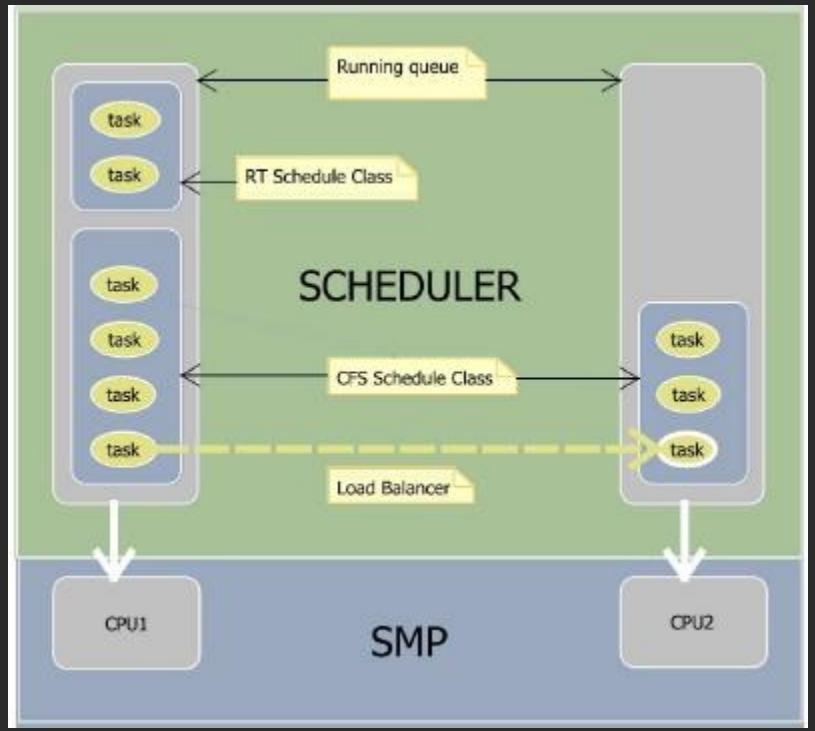
SELECT курильщика

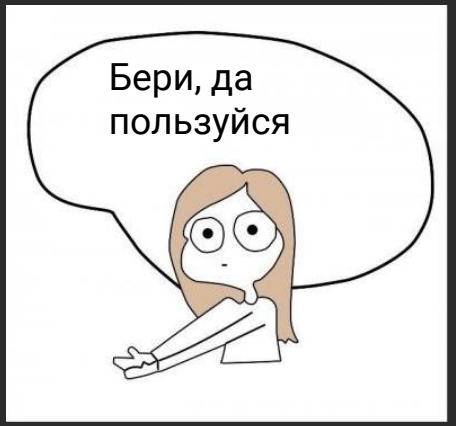
```
SELECT(2)                  Linux Programmer's Manual                  SELECT(2)

NAME        top

       select,  pselect, FD_CLR, FD_ISSET, FD_SET, FD_ZERO - synchronous I/O
       multiplexing

SYNOPSIS        top

       /* According to POSIX.1-2001, POSIX.1-2008 */
       #include <sys/select.h>
```

Aeron API. BusySpin



Aeron API. BusySpin





Aeron API. BusySpin

```
perf sched latency
```

Aeron API. BusySpin

Luxoft
A DXC Technology Company

```
root@asrv241 ~/ilgiz # perf sched record -- sleep 30
[ perf record: Woken up 865 times to write data ]
[ perf record: Captured and wrote 1774.102 MB perf.data (15892533 samples) ]
root@asrv241 ~/ilgiz # perf sched latency

 -----------------------------------------------------------------------------------------------------------------
  Task                |   Runtime ms  | Switches | Average delay ms | Maximum delay ms | Maximum delay at       |
 -----------------------------------------------------------------------------------------------------------------
  ps:(3)              |     63.985 ms |        6 | avg:    0.296 ms | max:    1.716 ms | max at: 17822107.861744 s
  migration/1:16      |      0.000 ms |       15 | avg:    0.137 ms | max:    1.058 ms | max at: 17822095.837621 s
  ksoftirqd/0:3       |      6.997 ms |      388 | avg:    0.131 ms | max:   13.839 ms | max at: 17822081.059072 s
  ksoftirqd/23:150    |      9.790 ms |      352 | avg:    0.122 ms | max:   12.332 ms | max at: 17822100.768687 s
  migration/20:131    |      0.000 ms |        9 | avg:    0.103 ms | max:    0.870 ms | max at: 17822091.981604 s
  migration/7:53      |      0.000 ms |       10 | avg:    0.101 ms | max:    0.979 ms | max at: 17822092.001717 s
  migration/22:143    |      0.000 ms |        6 | avg:    0.098 ms | max:    0.570 ms | max at: 17822081.857756 s
  ksoftirqd/1:17      |     31.848 ms |     1296 | avg:    0.096 ms | max:   15.960 ms | max at: 17822106.052104 s
  check_process_u:(19)|     17.118 ms |       71 | avg:    0.092 ms | max:    5.897 ms | max at: 17822107.830432 s
  apps.plugin:16288   |   1751.462 ms |      125 | avg:    0.086 ms | max:    3.975 ms | max at: 17822080.209278 s
  debian-sa1:17984    |      0.705 ms |        1 | avg:    0.085 ms | max:    0.085 ms | max at: 17822103.727999 s
  kworker/6:1H:464    |      4.593 ms |      101 | avg:    0.081 ms | max:    3.323 ms | max at: 17822100.804354 s
  jbd2/md3-8:587      |      2.704 ms |       47 | avg:    0.080 ms | max:    2.479 ms | max at: 17822086.692834 s
  ksoftirqd/15:102    |     24.939 ms |     1174 | avg:    0.079 ms | max:   15.009 ms | max at: 17822089.859866 s
  ksoftirqd/2:23      |      8.200 ms |      419 | avg:    0.078 ms | max:   15.976 ms | max at: 17822083.041131 s
  kworker/u50:2:29286 |     83.062 ms |       25 | avg:    0.073 ms | max:    0.267 ms | max at: 17822086.014835 s
  ksoftirqd/11:78     |      4.761 ms |      316 | avg:    0.069 ms | max:   15.350 ms | max at: 17822103.903562 s
```

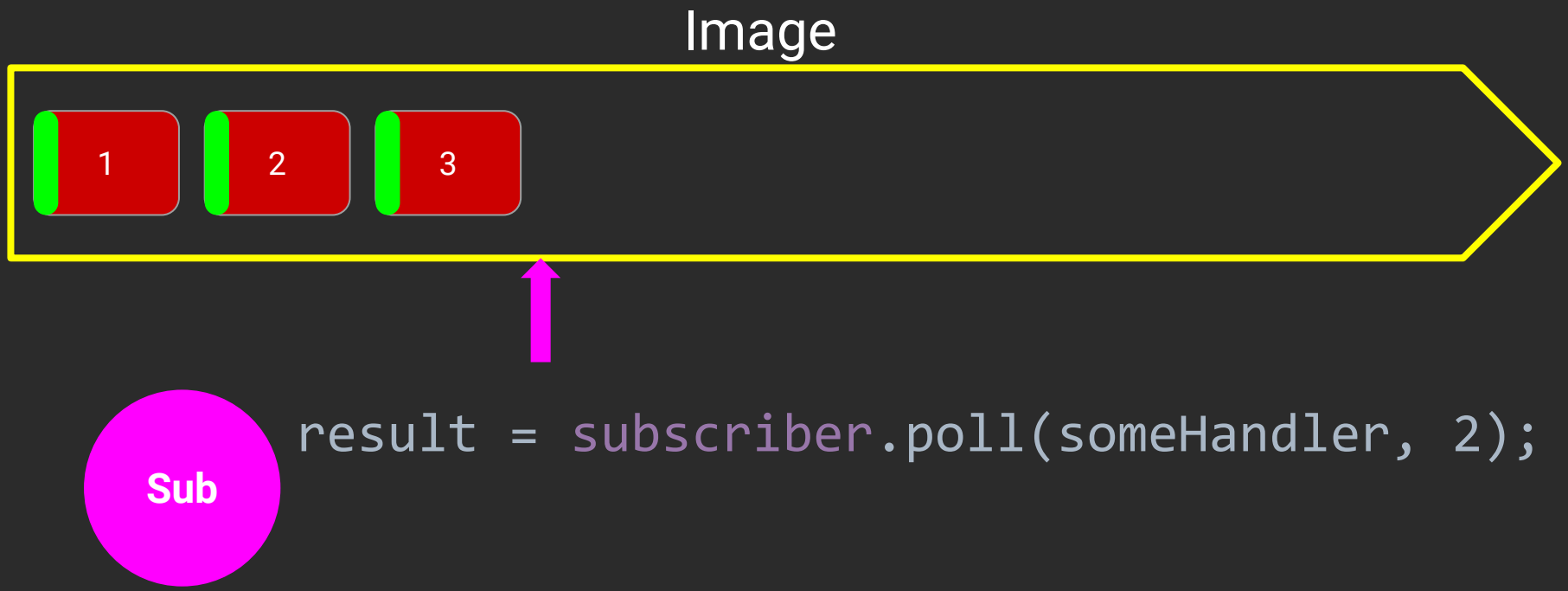
Aeron API. BusySpin

Task	Runtime ms	Switches	Average delay ms	Maximum delay ms
ps:(3)	63.985 ms	6	avg: 0.296 ms	max: 1.716 ms
migration/1:16	0.000 ms	15	avg: 0.137 ms	max: 1.058 ms
ksoftirqd/0:3	6.997 ms	388	avg: 0.131 ms	max: 13.839 ms
ksoftirqd/23:150	9.790 ms	352	avg: 0.122 ms	max: 12.332 ms
migration/20:131	0.000 ms	9	avg: 0.103 ms	max: 0.870 ms
migration/7:53	0.000 ms	10	avg: 0.101 ms	max: 0.979 ms
migration/22:143	0.000 ms	6	avg: 0.098 ms	max: 0.570 ms
ksoftirqd/1:17	31.848 ms	1296	avg: 0.096 ms	max: 15.960 ms
check_process_u:(19)	17.118 ms	71	avg: 0.092 ms	max: 5.897 ms
apps.plugin:16288	1751.462 ms	125	avg: 0.086 ms	max: 3.975 ms
debian-sa1:17984	0.705 ms	1	avg: 0.085 ms	max: 0.085 ms
kworker/6:1H:464	4.593 ms	101	avg: 0.081 ms	max: 3.323 ms
jbd2/md3-8:587	2.704 ms	47	avg: 0.080 ms	max: 2.479 ms
ksoftirqd/15:102	24.939 ms	1174	avg: 0.079 ms	max: 15.009 ms
ksoftirqd/2:23	8.200 ms	419	avg: 0.078 ms	max: 15.976 ms
kworker/u50:2:29286	83.062 ms	25	avg: 0.073 ms	max: 0.267 ms

Aeron API. BusySpin

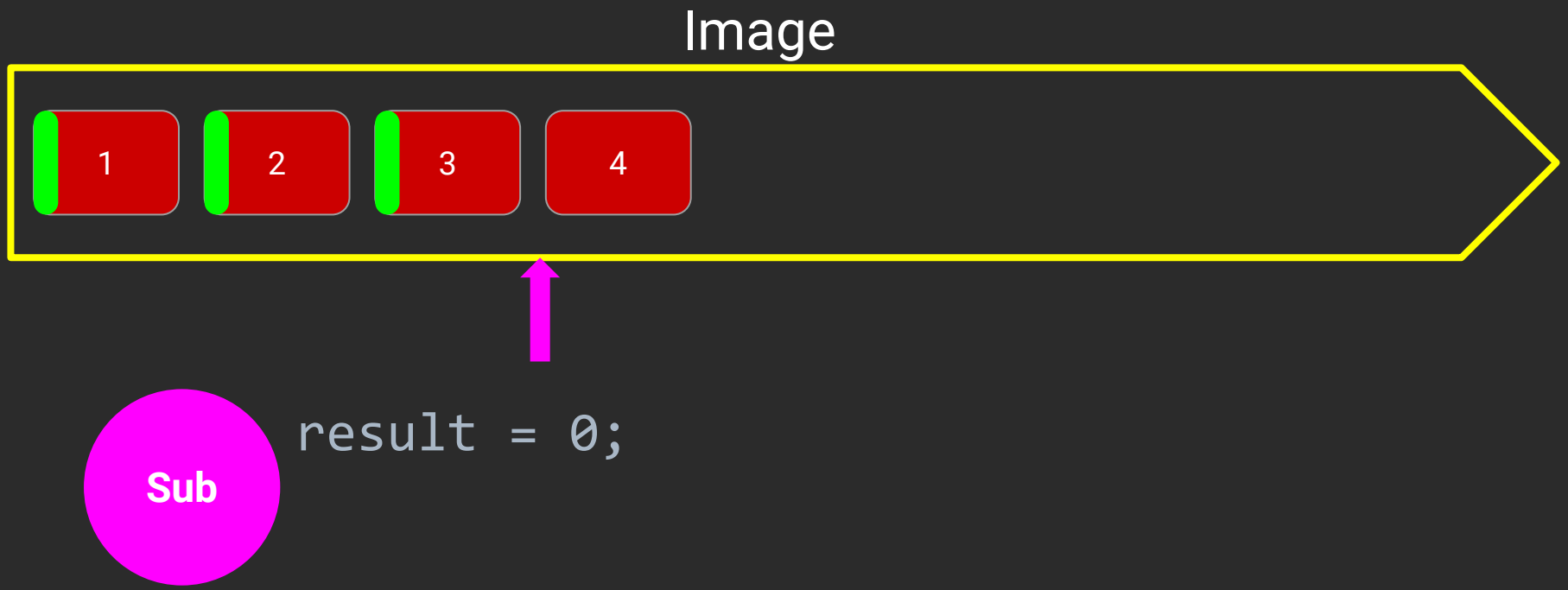
```
while (isRunning())
{
    int received = subscription.poll(dataHandler, LIMIT);
    idleStrategy.idle(received);  // Thread#onSpinWait
}
```

Aeron API. Subscriber. Чтение

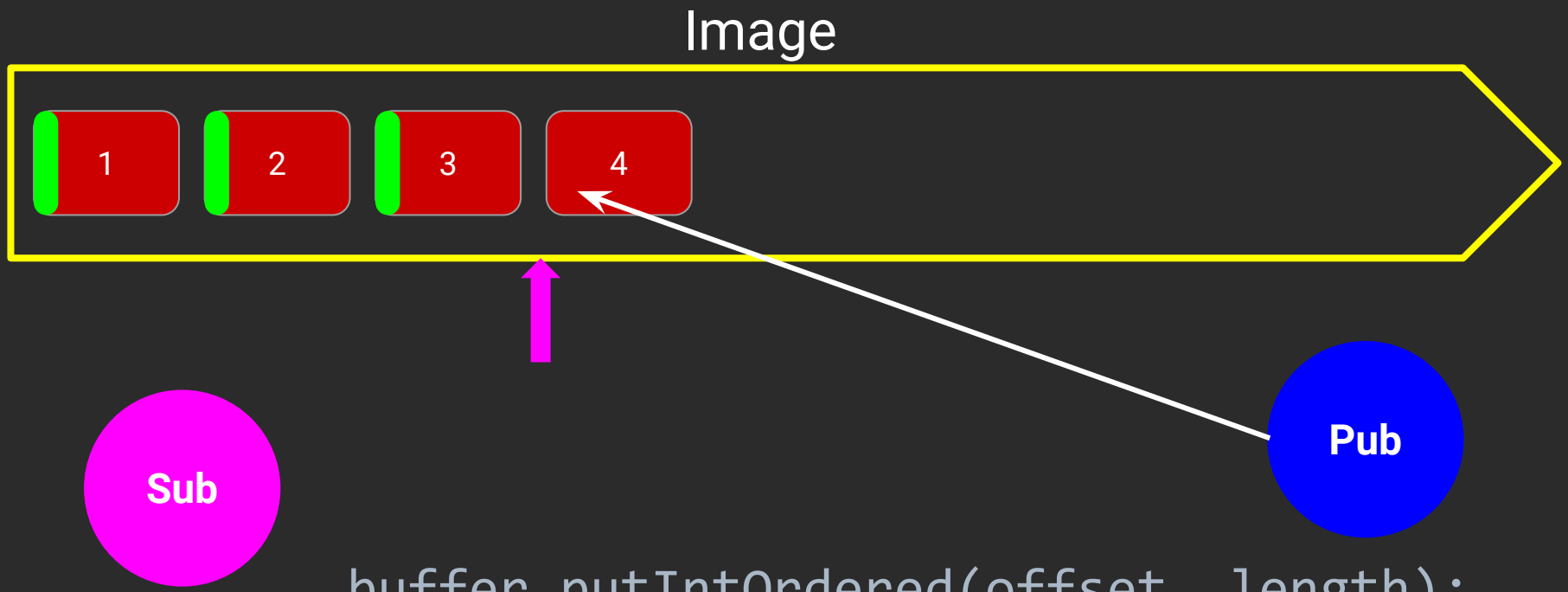


Aeron API. Subscriber. Чтение

Aeron API. Subscriber. Чтение

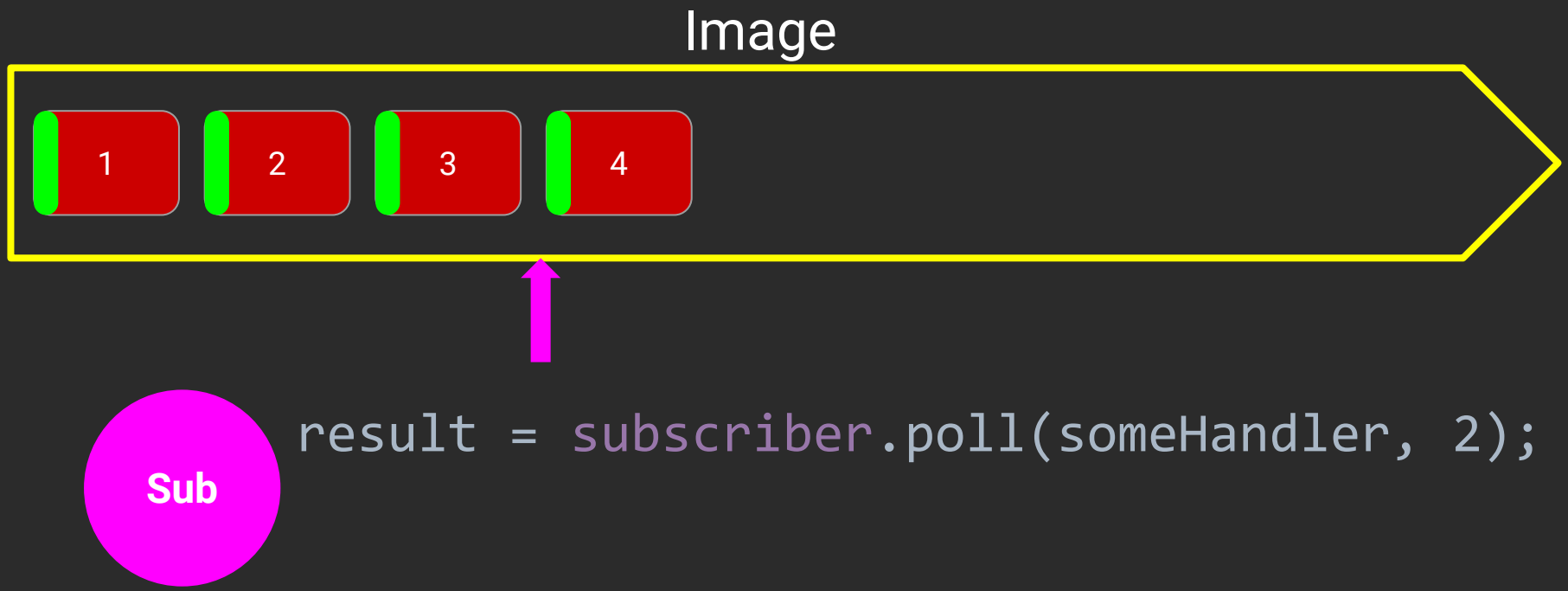


Aeron API. Subscriber. Чтение

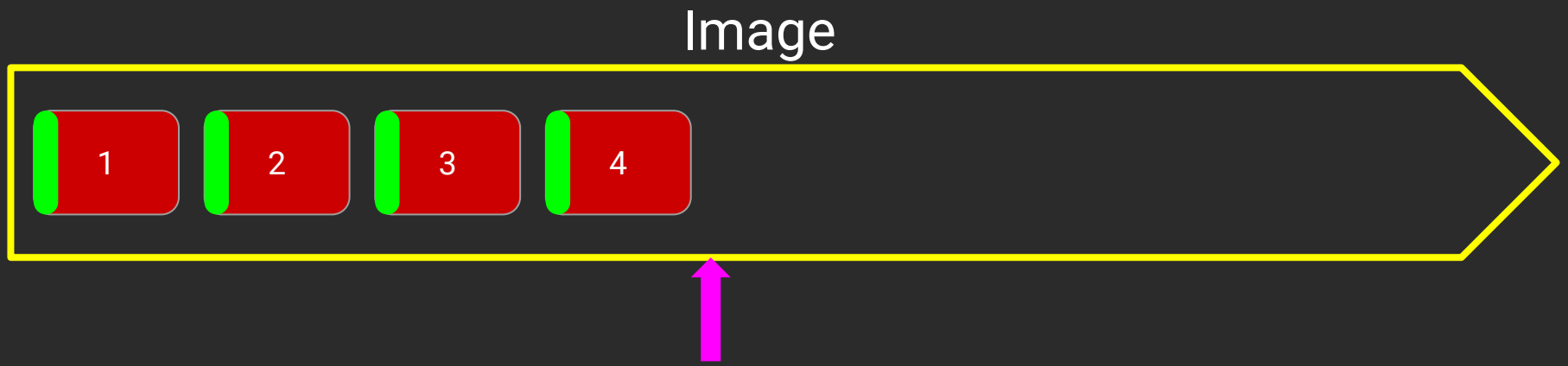



```
buffer.putIntOrdered(offset, length);
```

Aeron API. Subscriber. Чтение



Aeron API. Subscriber. Чтение



Aeron API. Subscriber

```
subscription.poll(
    fragmentHandler:
    (buffer, offset, length, header) -> {
        consumeString(buffer.getStringUtf8(offset, length));
    },
    fragmentLimit:
    10);
```

Fragment == message?

Aeron API. Subscriber

Fragment - пакет влезающий в MTU, если сообщение больше чем MTU, то оно дробится на несколько fragment’ов.

MTU ~= 1500 byte

Чтобы склеивать их нужен `FragmentAssembler`

Aeron API. Subscriber

Declare:

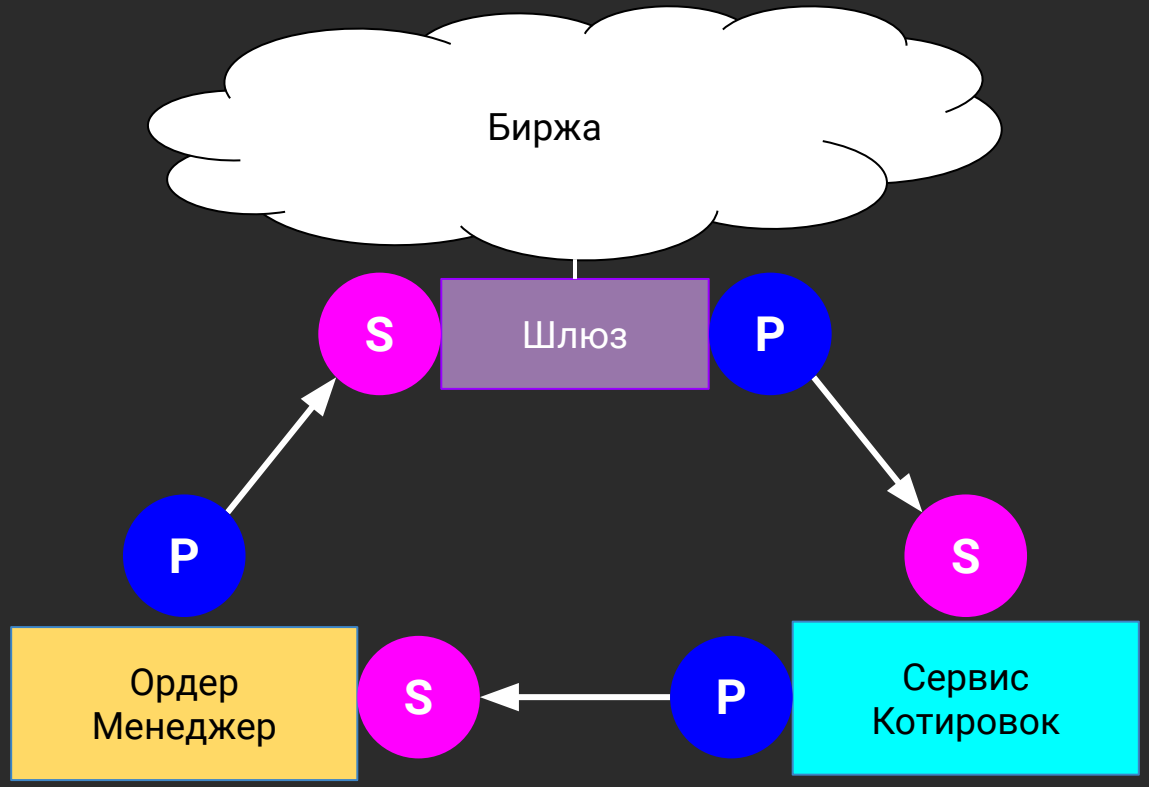
```
FragmentHandler myFragment = new FragmentAssembler(
    (buffer, offset, length, header) -> {
        consumeString(buffer.getStringUtf8(offset, length));
    });
```

Usage:

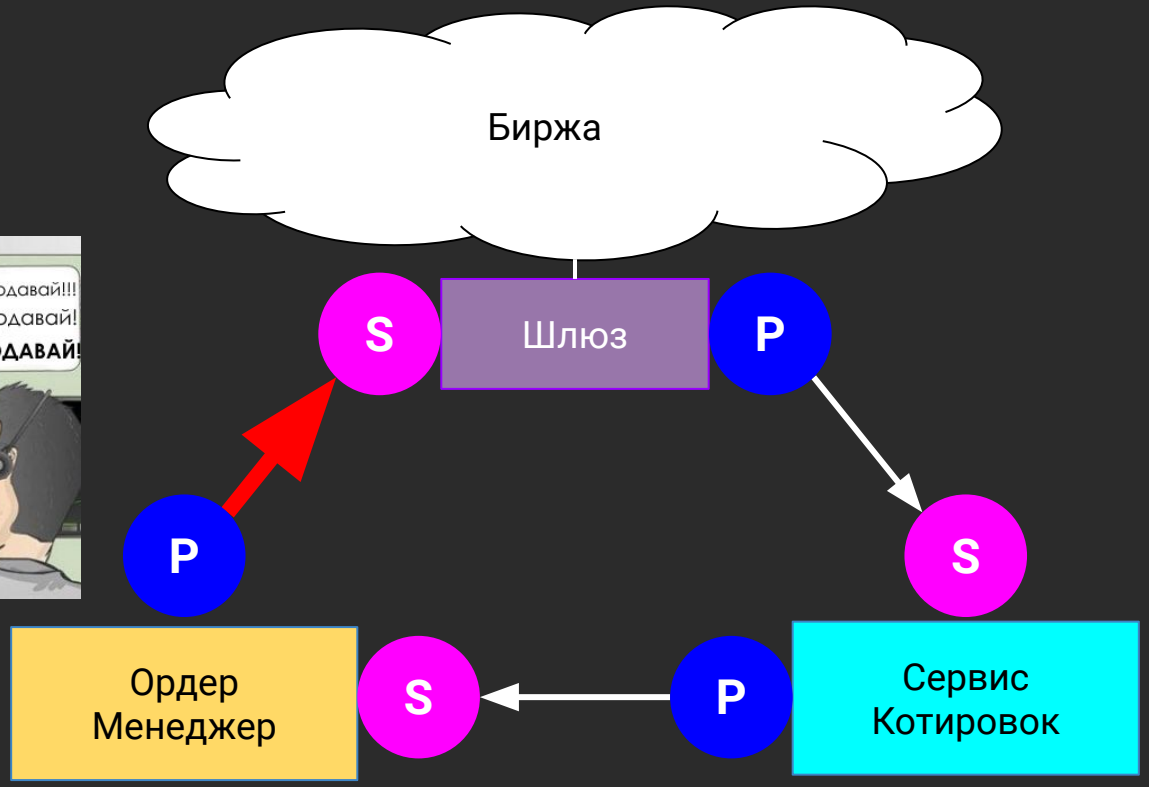
```
subscription.poll(myFragment, 10);
```

Luxoft
A DXC Technology Company

Aeron API. Резюме.




New York
Срочно продавай снег!!! Продавай!!!
Продавай!!!
Продавай!
ПРОДАВАЙ!


Биржа
S
Шлюз
P
P
S
Ордер Менеджер
S
P
Сервис Котировок

Aeron API. Резюме

```
result = publication.offer(
```

```
)
```

 = 30 рублей

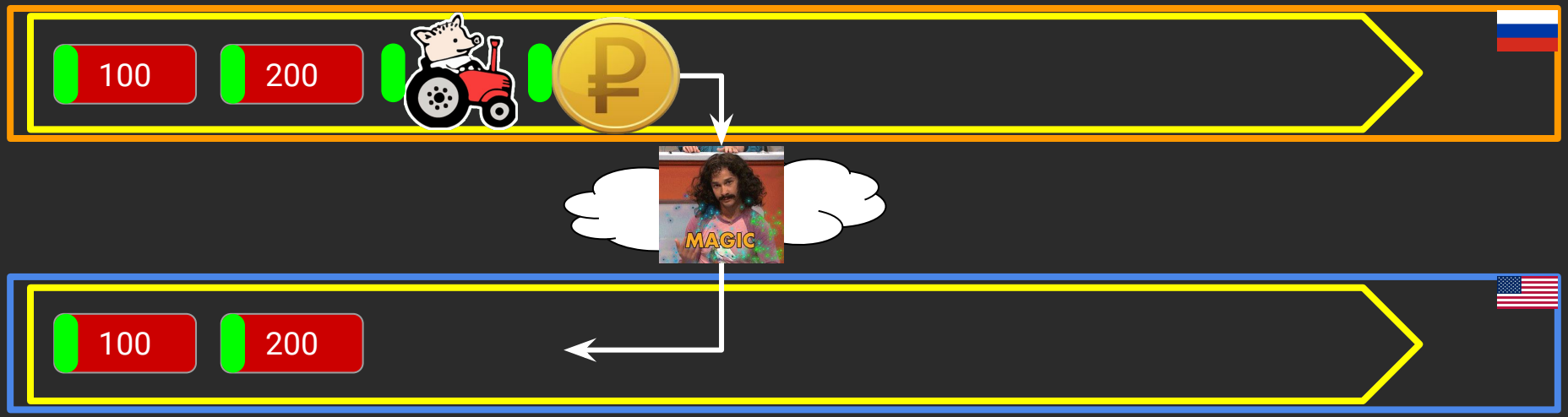
Aeron API. Резюме

```
result = 300
```

 = 30 рублей

Aeron API. Резюме

```
result = 300
```



$ = 30 рублей

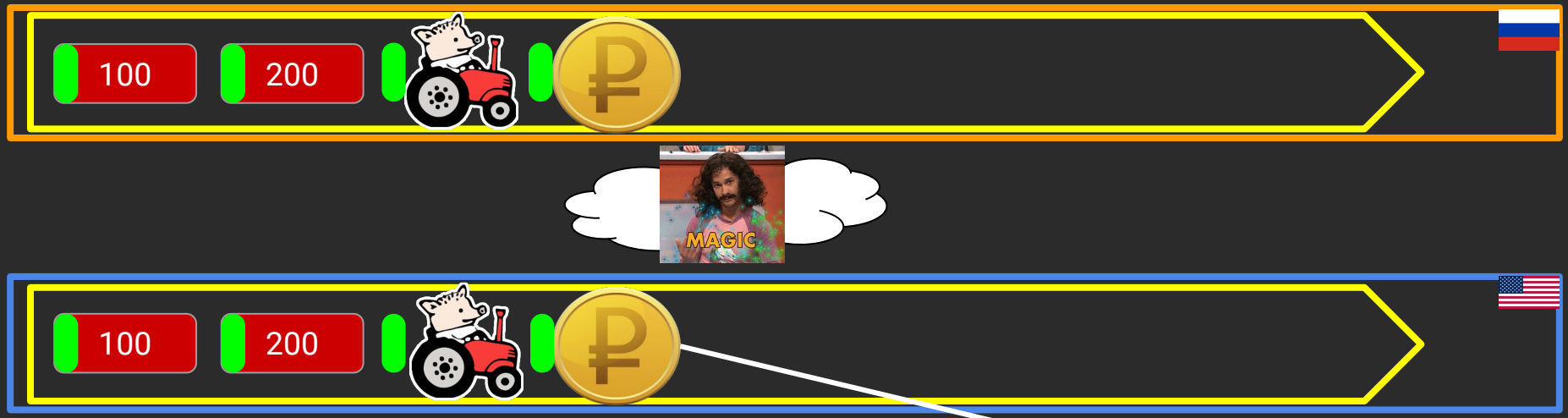
Aeron API. Резюме

```
result = 300
```

$ = 30 рублей

Aeron API. Резюме

result = 300

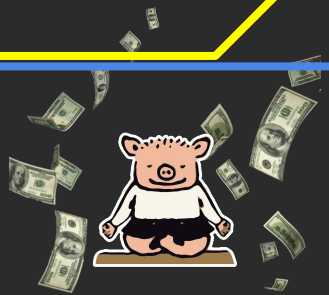



```
FragmentHandler handler = new FragmentAssembler((b, o, l, h) -> { купить });
while (isRunning()) {
   idleStrategy.idle(subscription.poll(handler, LIMIT));
}
```

Aeron API. Резюме

result = 300


```
FragmentHandler handler = new FragmentAssembler((b, o, l, h) -> {                    });
while (isRunning()) {
    idleStrategy.idle(subscription.poll(handler, LIMIT));
}
```



Luxoft
A DXC Technology Company

Архитектура Aeron

Архитектура Aeron. Media Driver

Luxoft
A DXC Technology Company

/dev/shm

Архитектура Aeron. Media Driver

A

100

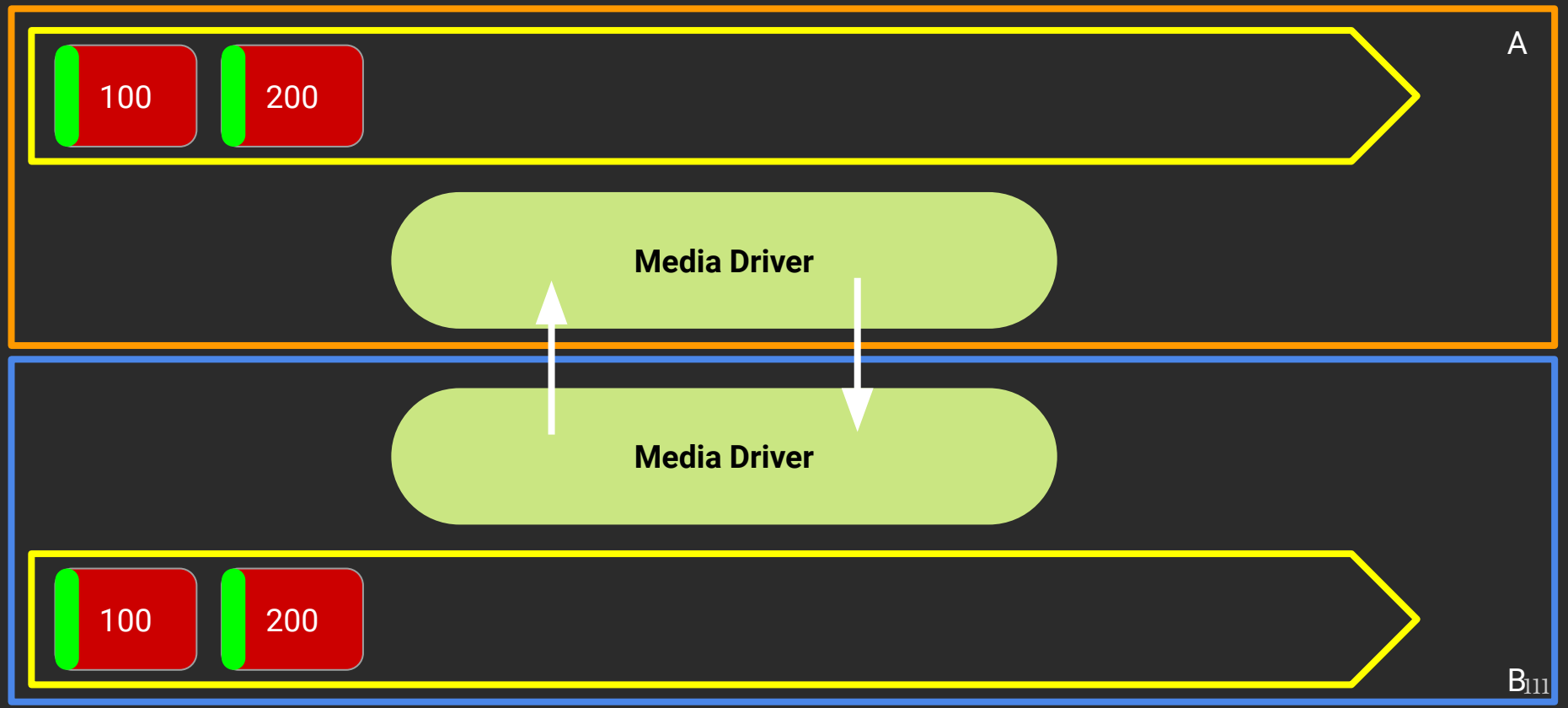
200

B

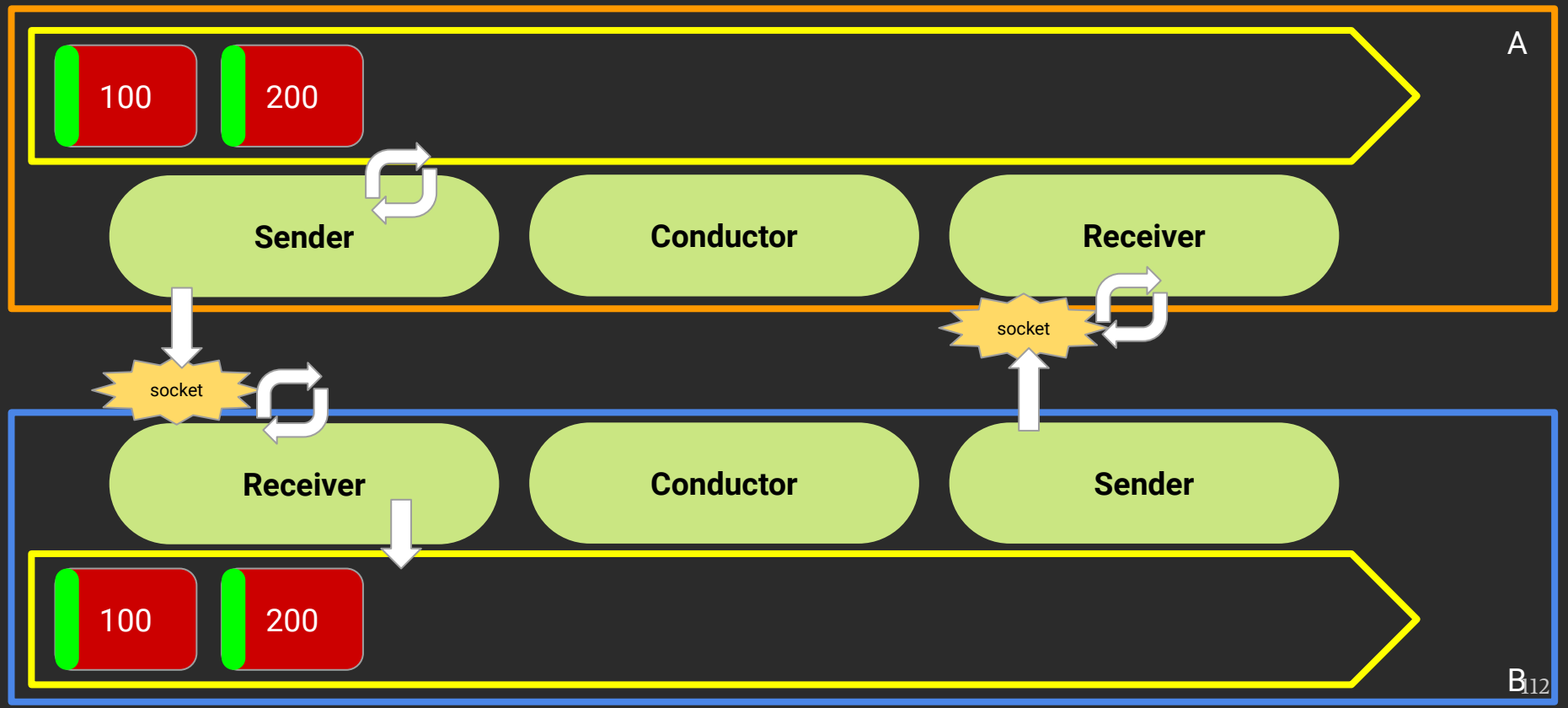
100

200

Архитектура Aeron. Media Driver



Архитектура Aeron. Media Driver



Архитектура Aeron. Media Driver. Conductor

Luxoft
A DXC Technology Company

Что же делает Conductor?

- Посылает Heartbeat
- Принимает команды на создание/закрытие соединений
- Приход/уход Image в subscription
- Проверяет на дырки данные

Архитектура Aeron. Media Driver

Receiver/Sender/Conductor - это всё Agent

```
public interface Agent {
    int doWork() throws Exception
}
```

Основной кейс использования:

```
while (isRunning()) {
   idleStrategy.idle(agent.doWork());
}
```

Архитектура Aeron. Media Driver

Luxoft
A DXC Technology Company

Threading model: DEDICATED

Sender thread	Receiver thread	Conductor thread

`while (isRunning()) {`
 	
`  int work =`
`  sender.doWork();`
`  idle.idle(work);`
 	
`}`
	`while (isRunning()) {`
	
	`  int work =`
	`  receiver.doWork();`
	`  idle.idle(work);`
	
	`}`
		`while (isRunning()) {`
		
		`  int work =`
		`  conductor.doWork();`
		`  idle.idle(work);`
		
		`}`

Архитектура Aeron. Media Driver

Luxoft
A DXC Technology Company

Threading model: SHARED_NETWORK

Network thread	Conductor thread

`while (isRunning()) {`
 	
`  int work = 0;`
`  work += sender.doWork();`
`  work += receiver.doWork();`
`  idle.idle(work);`
 	
`}`
	`while (isRunning()) {`
	
	`  int work =`
	`  conductor.doWork();`
	`  idle.idle(work);`
	
	`}`

Архитектура Aeron. Media Driver

Threading model: SHARED

Aeron thread

```
while (isRunning()) {

    int work = 0;
    work += sender.doWork();
    work += receiver.doWork();
    work += conductor.doWork();
    idle.idle(work);
}
```

Архитектура Aeron. Media Driver

Luxoft
A DXC Technology Company

Threading model: INVOKER

```
public int invoke() {

    int work = 0;
    work += sender.doWork();
    work += receiver.doWork();
    work += conductor.doWork();
    return work;

}
```

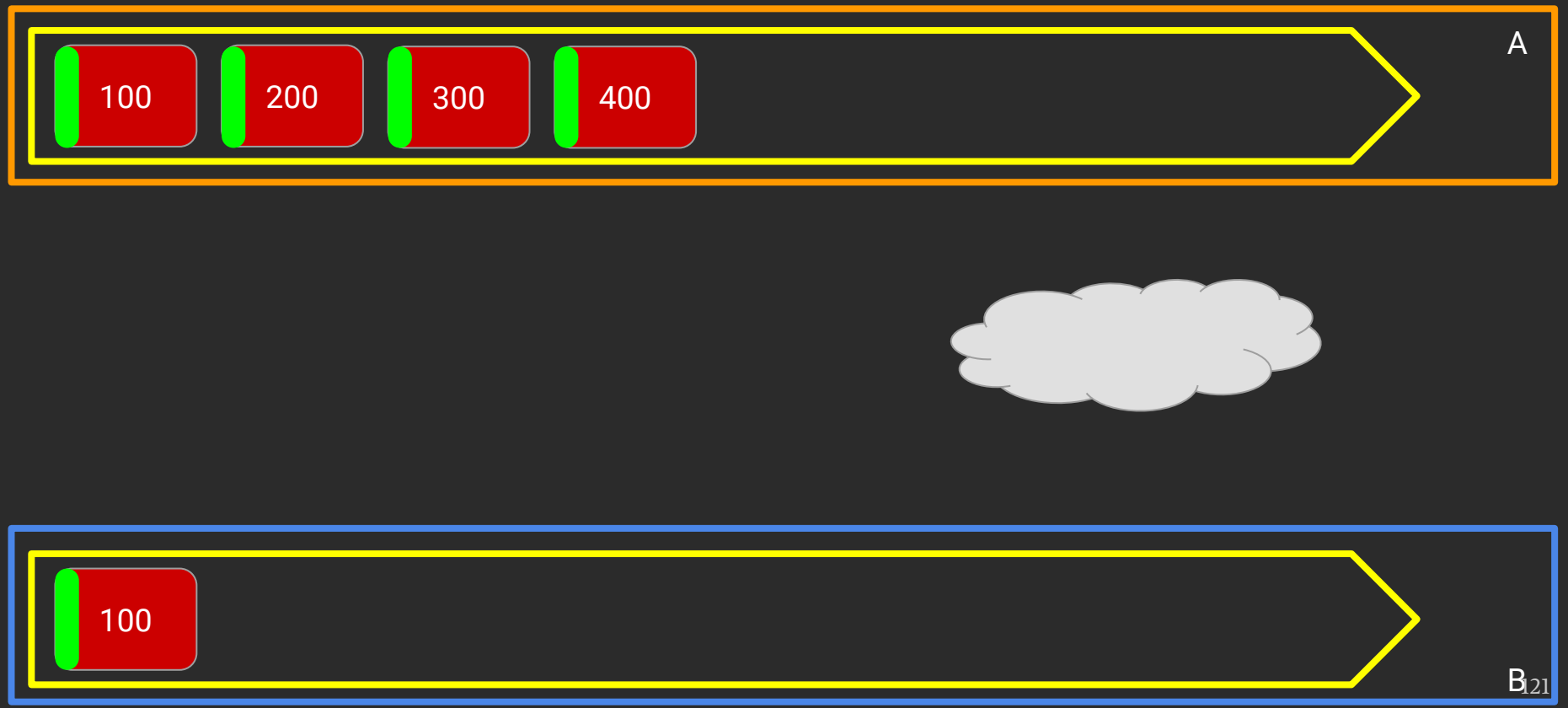
Luxoft
A DXC Technology Company

Протокол Aeron

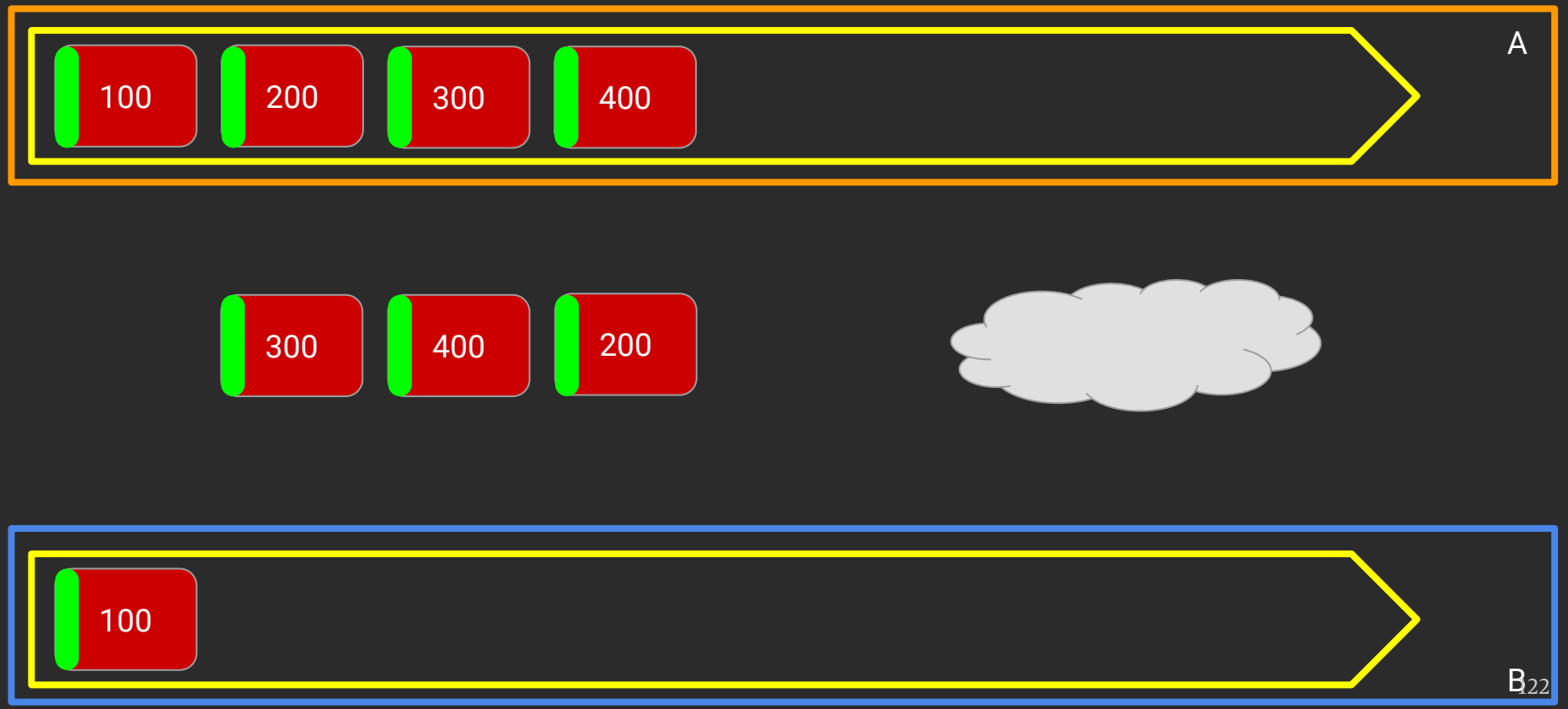
Luxoft
A DXC Technology Company

Reordering

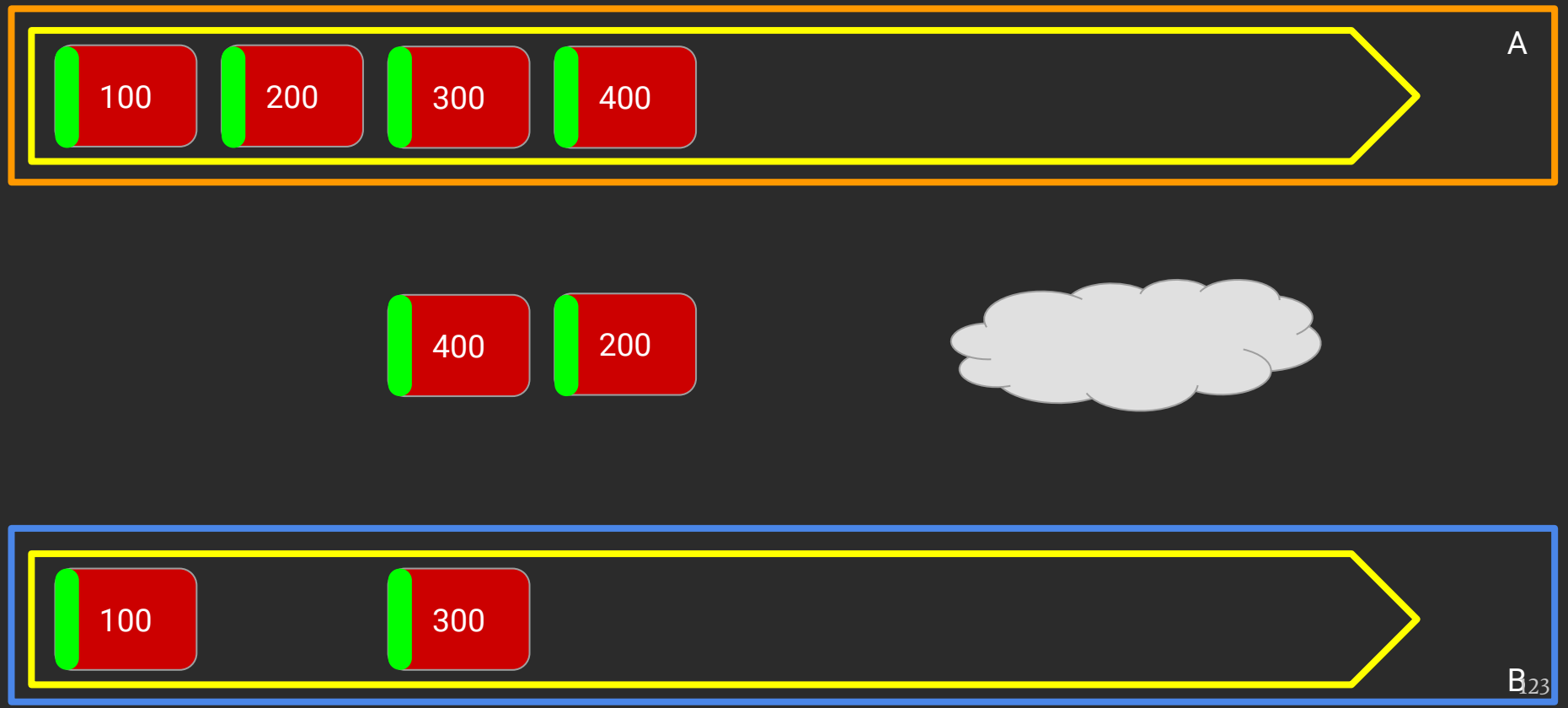
Протокол Aeron. Reordering



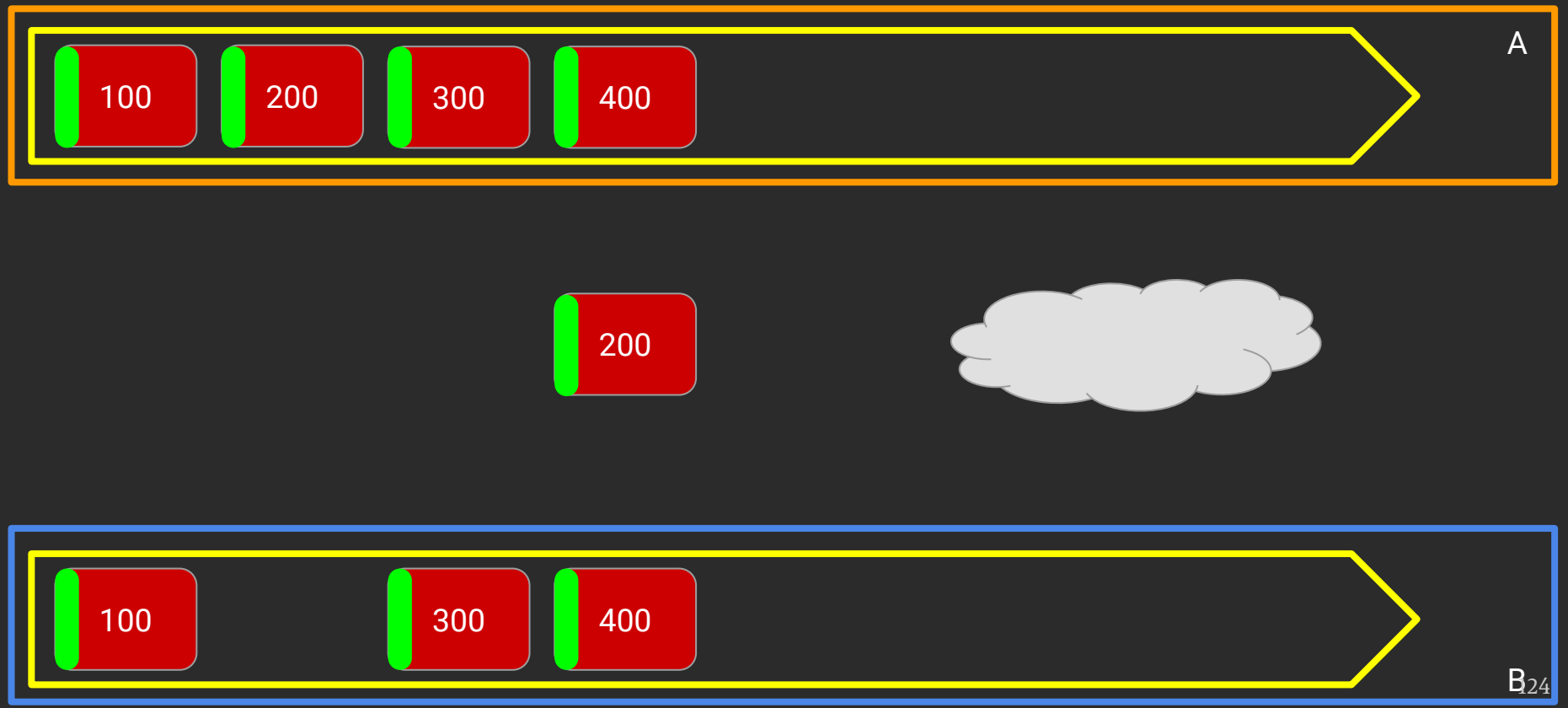
Протокол Aeron. Reordering



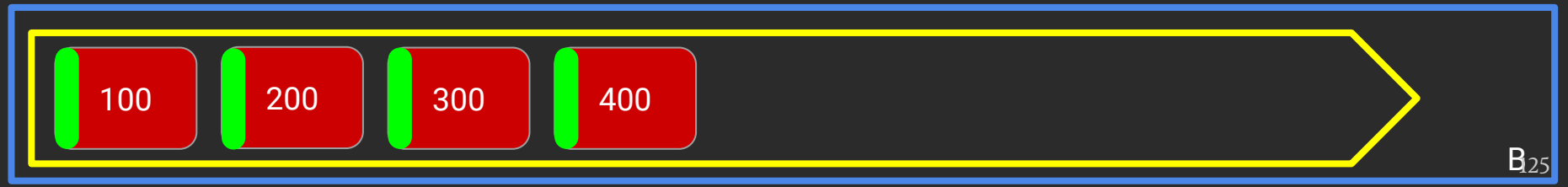
Протокол Aeron. Reordering



Протокол Aeron. Reordering



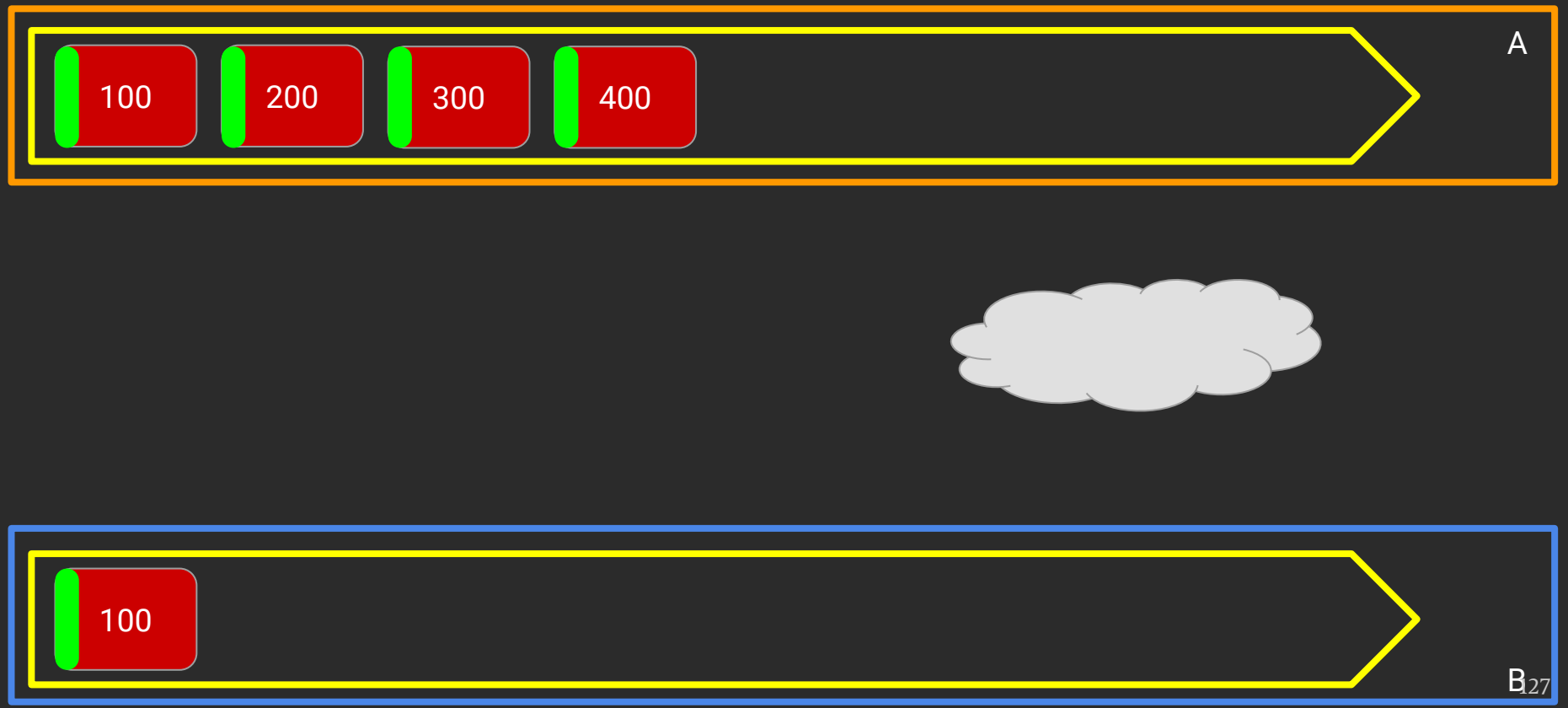
Протокол Aeron. Reordering



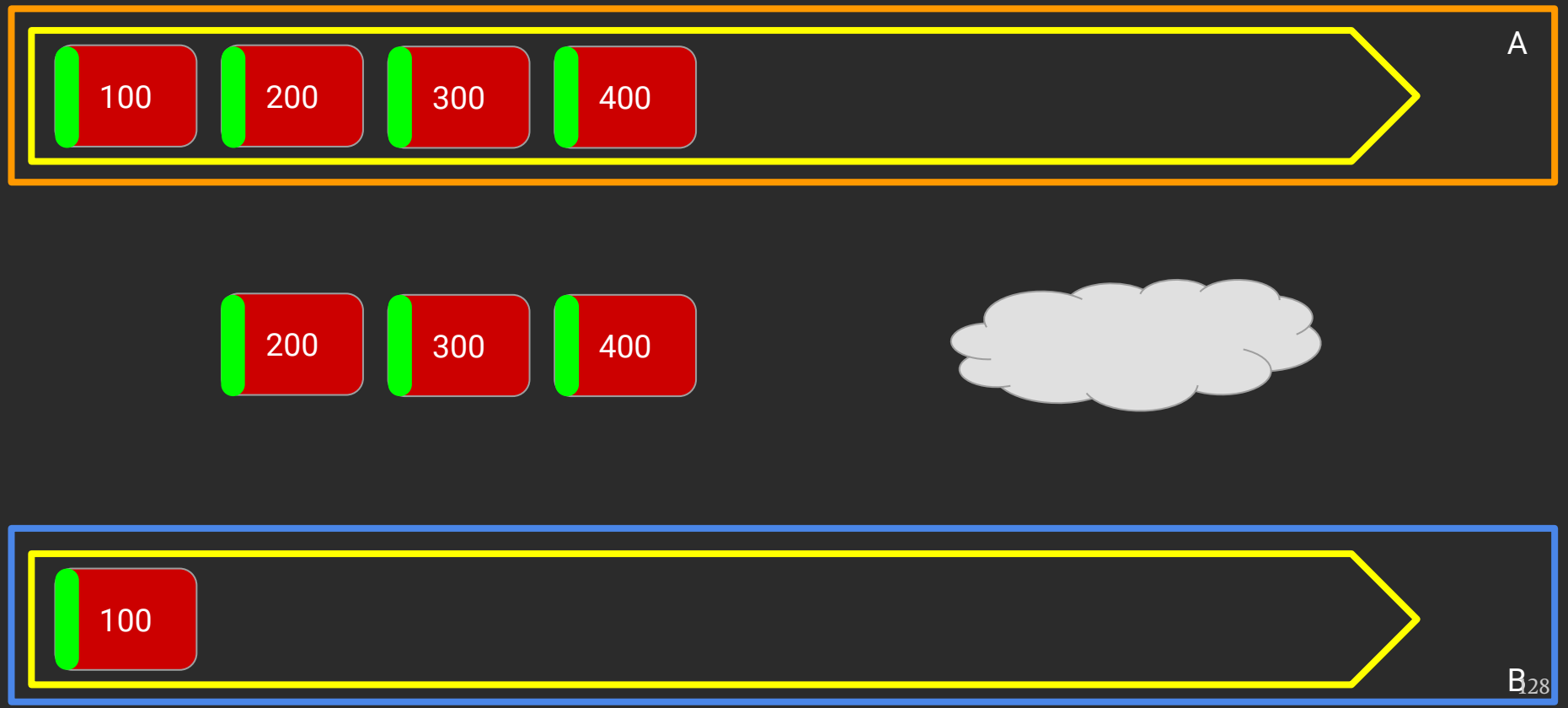
Luxoft
A DXC Technology Company

NAK
(Negative Acknowledgement)

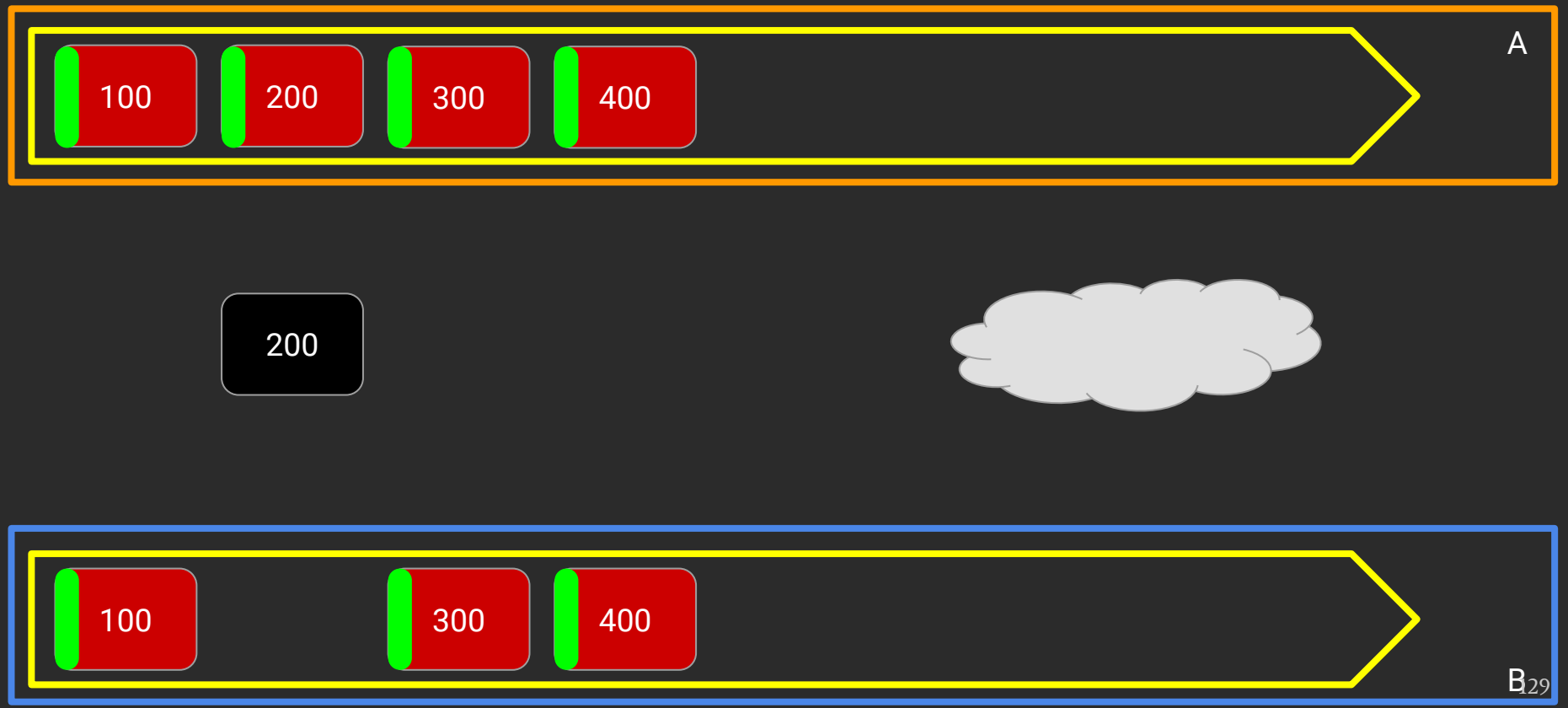
Протокол Aeron. NAK



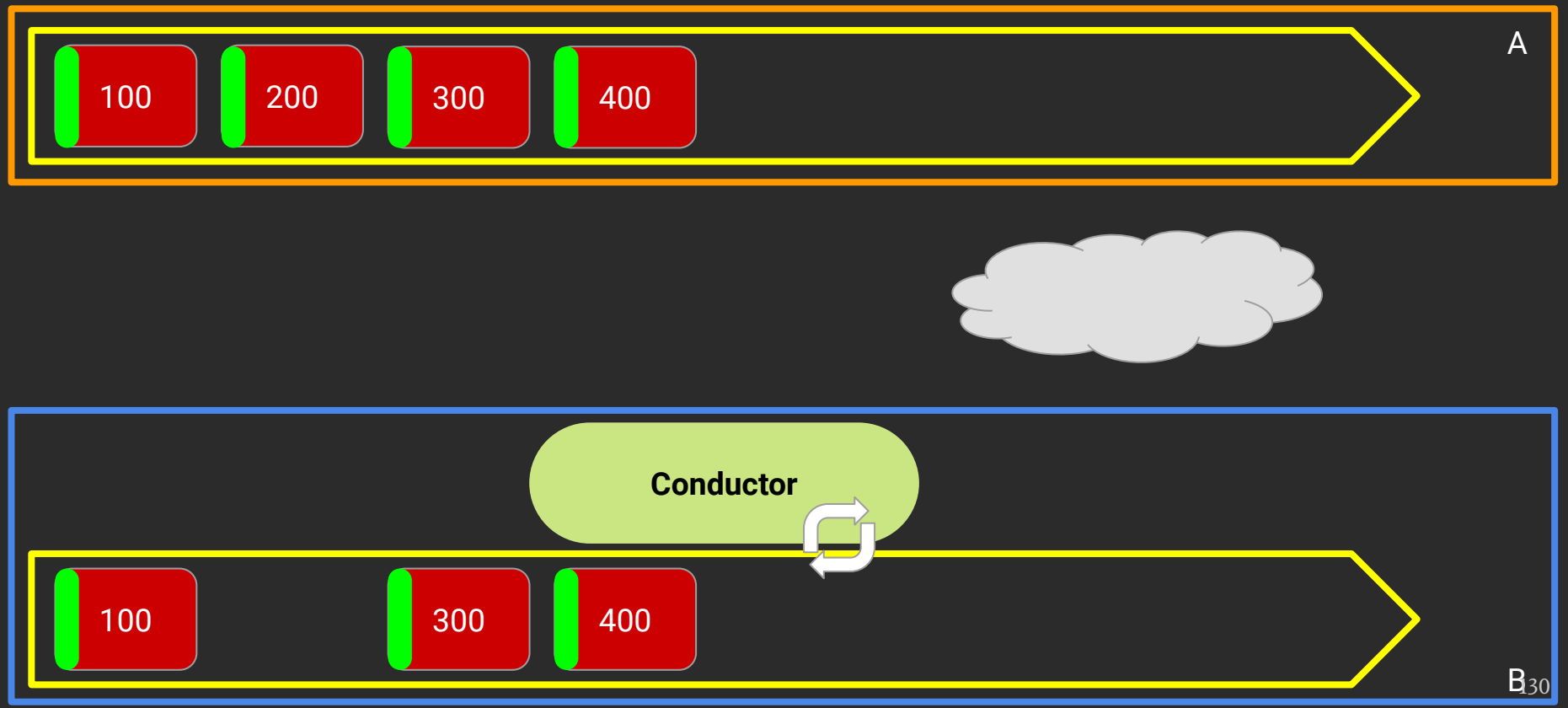
Протокол Aeron. NAK



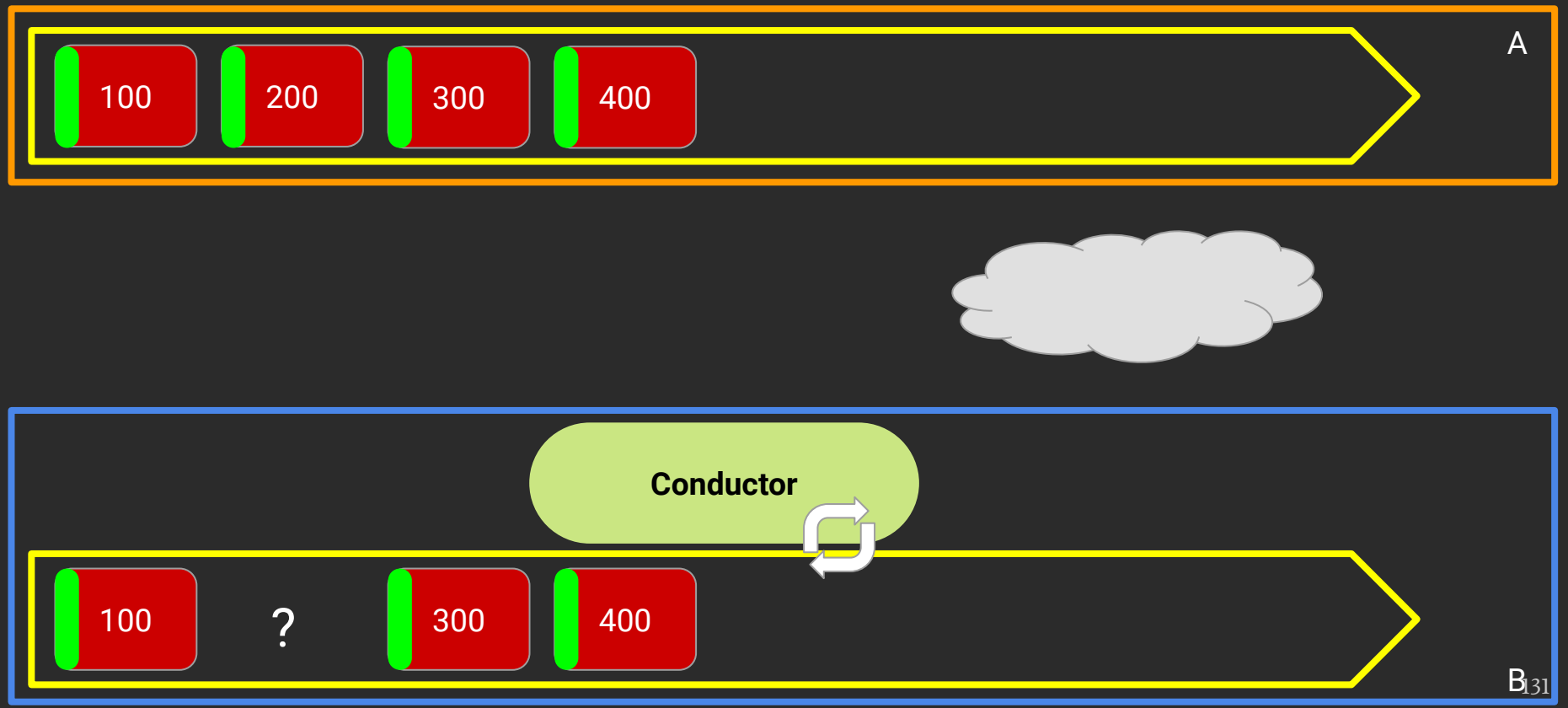
Протокол Aeron. NAK



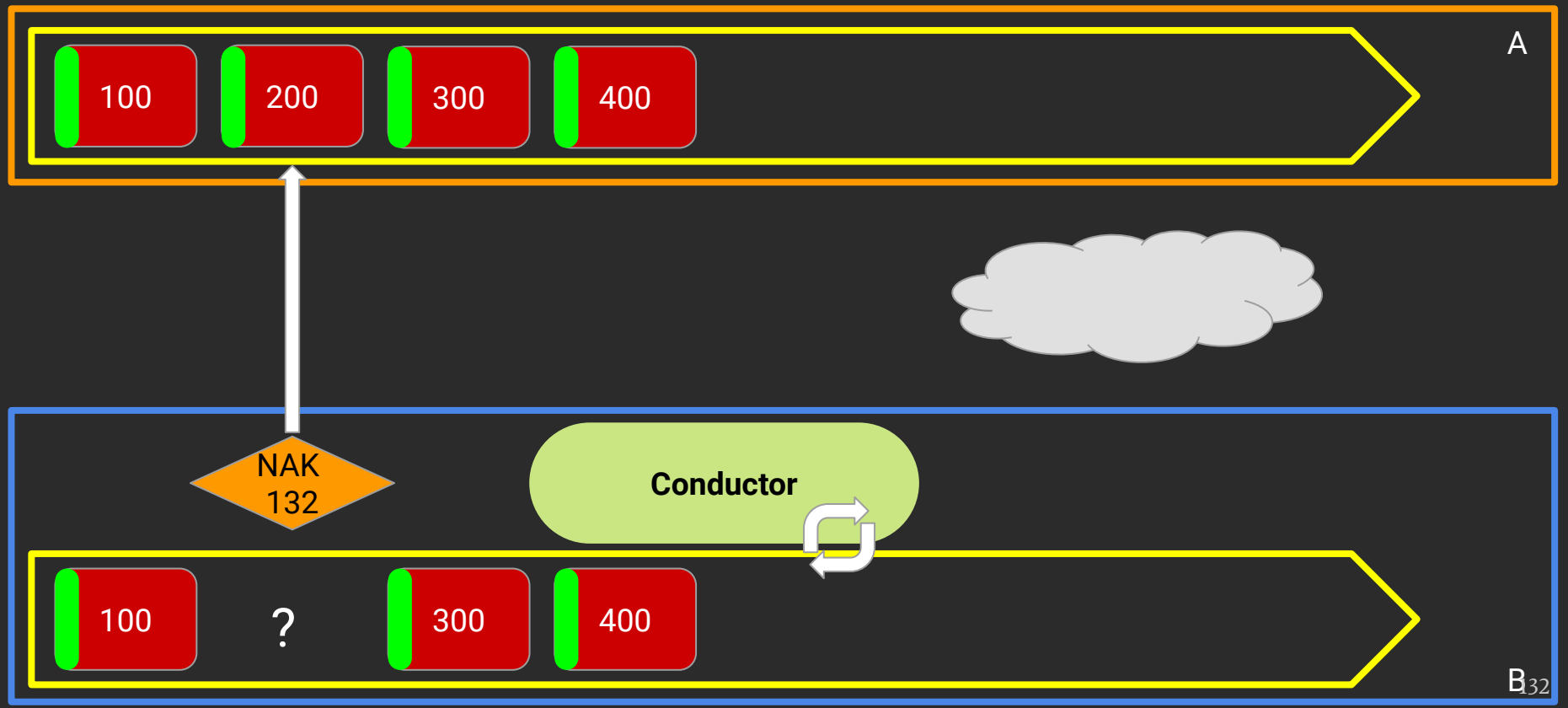
Протокол Aeron. NAK



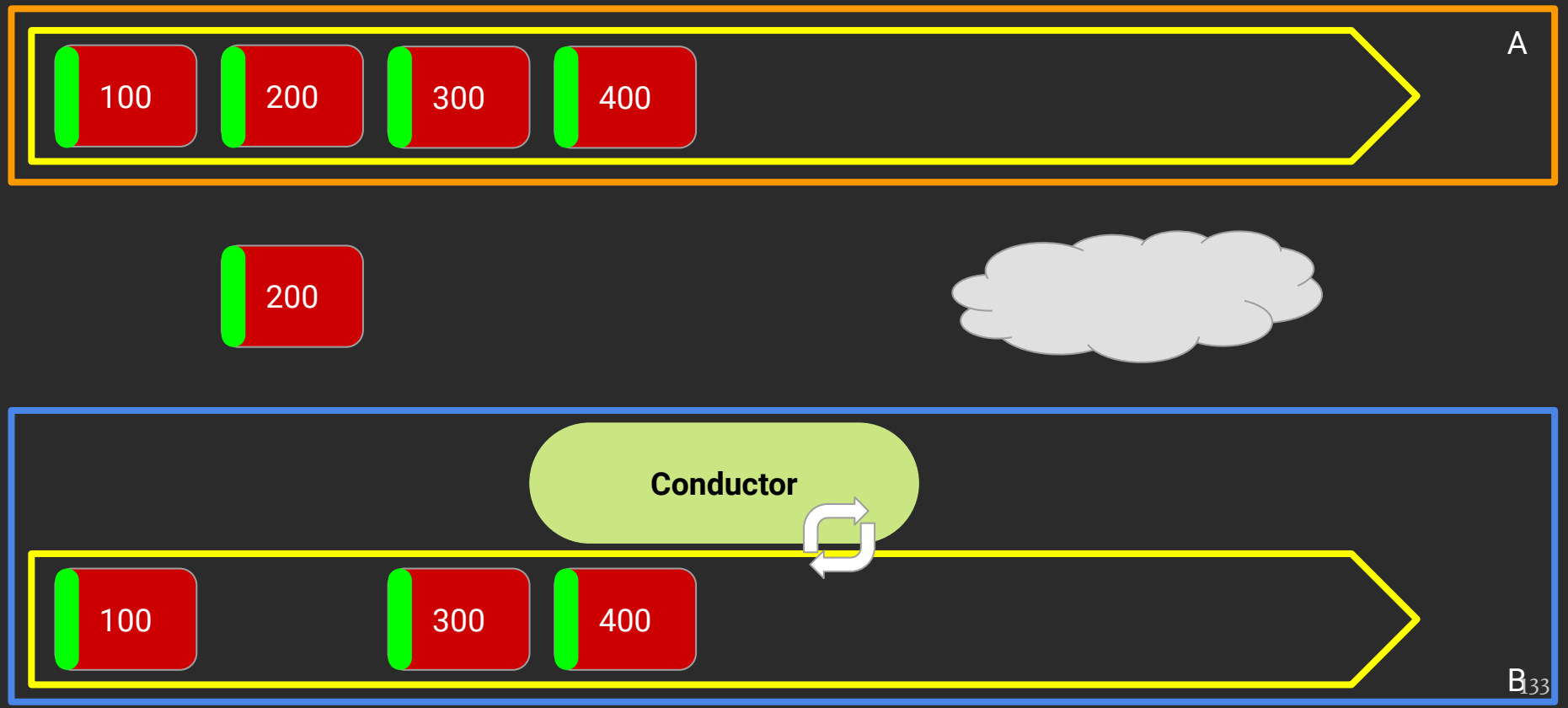
Протокол Aeron. NAK



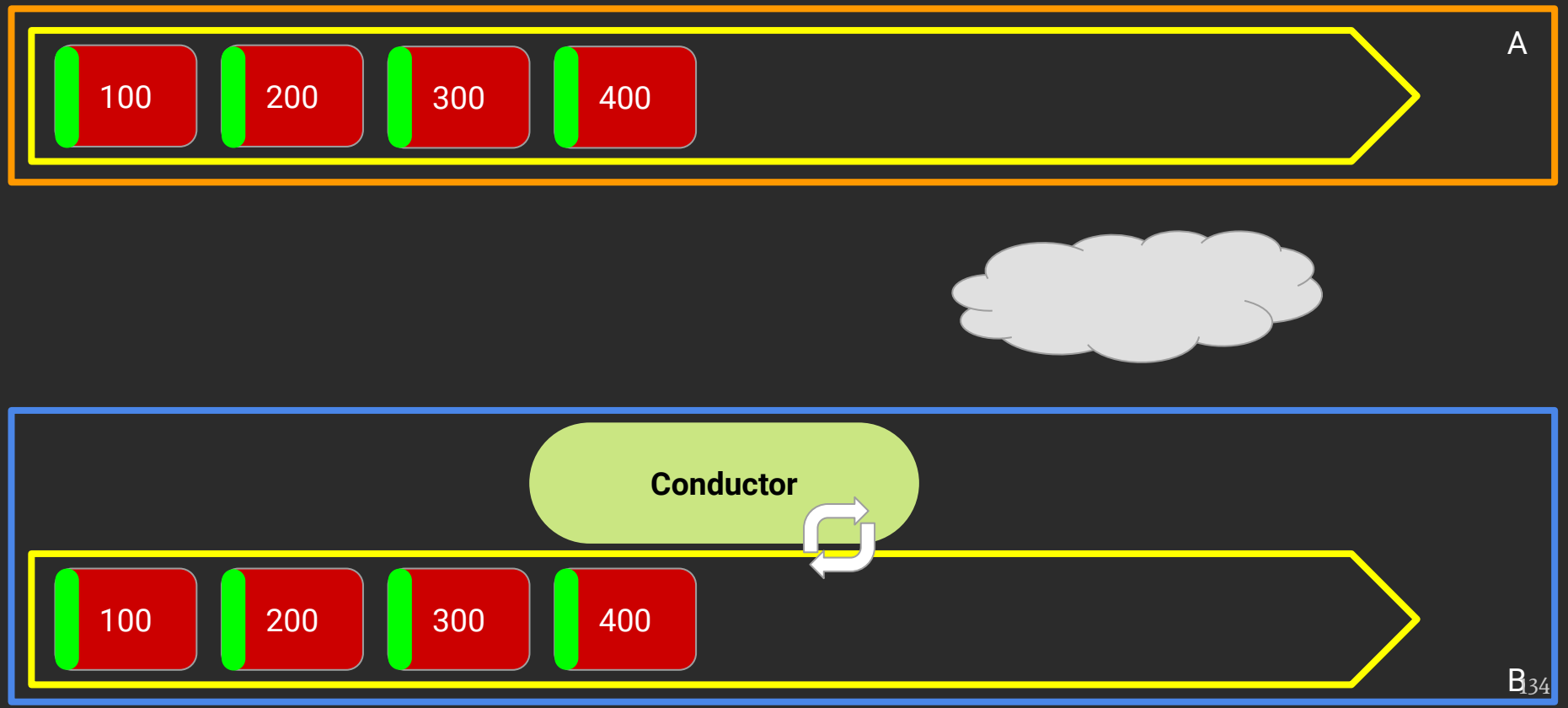
Протокол Aeron. NAK



Протокол Aeron. NAK



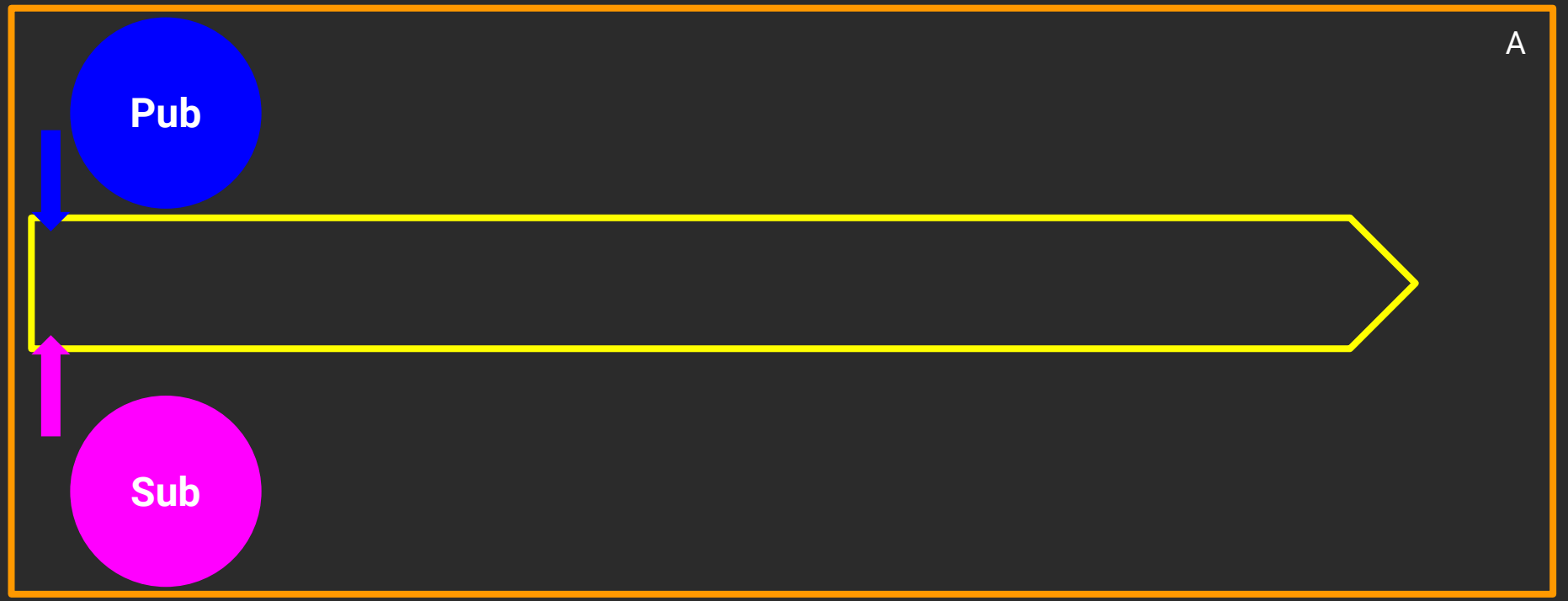
Протокол Aeron. NAK



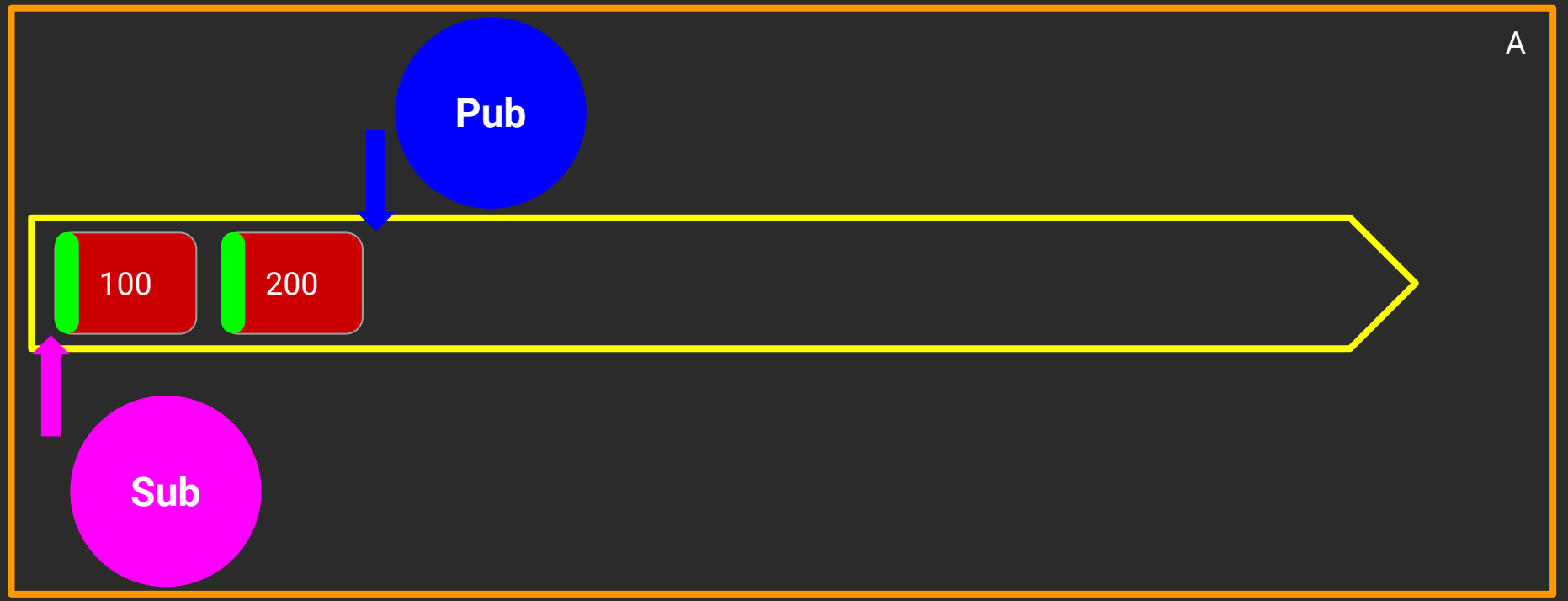
Luxoft
A DXC Technology Company

IPC
(Inter-Process Communication)

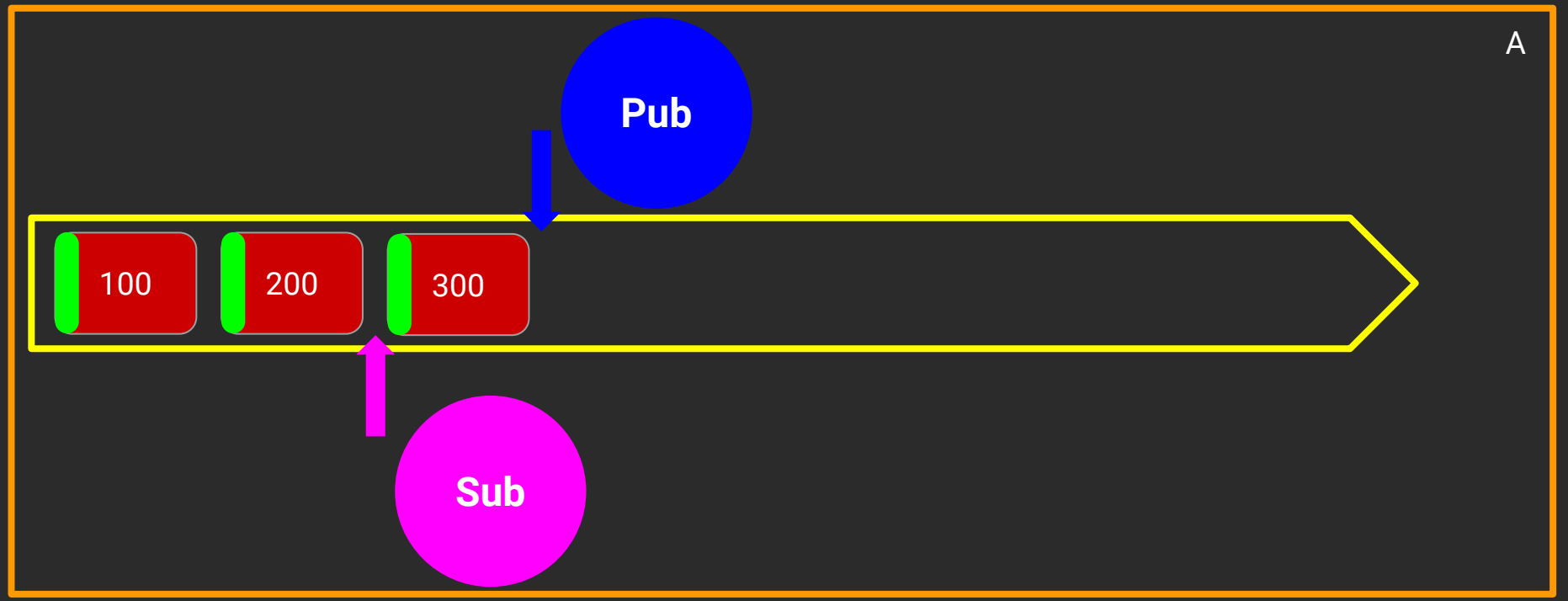
Протокол Aeron. IPC



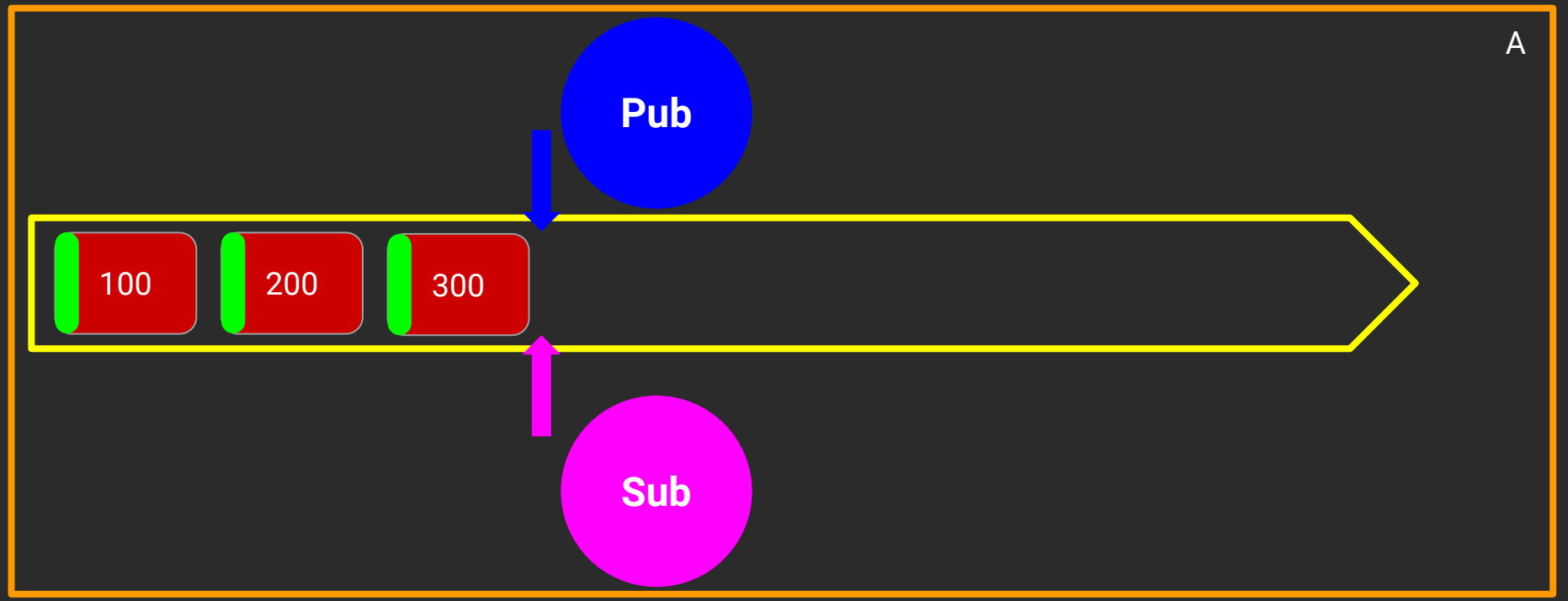
Протокол Aeron. IPC



Протокол Aeron. IPC



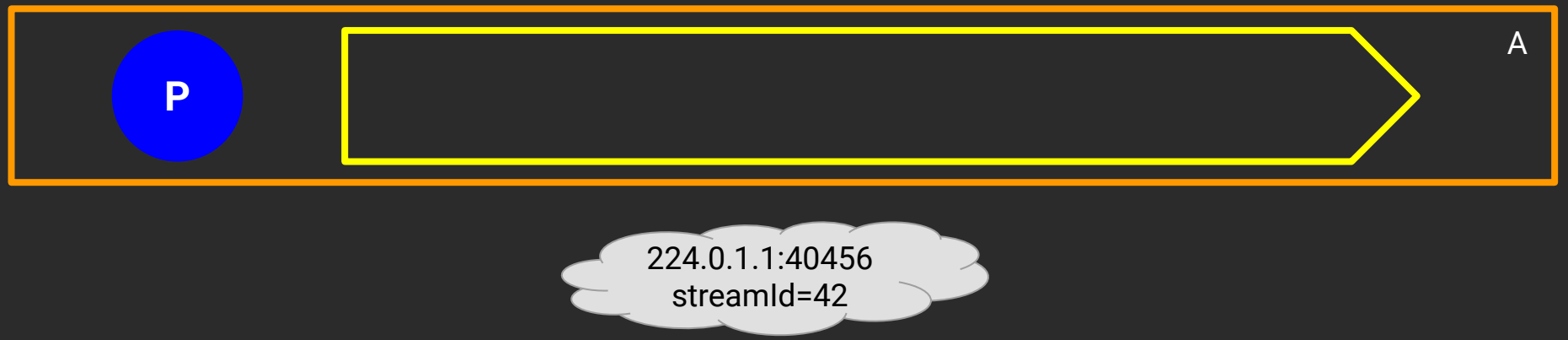
Протокол Aeron. IPC



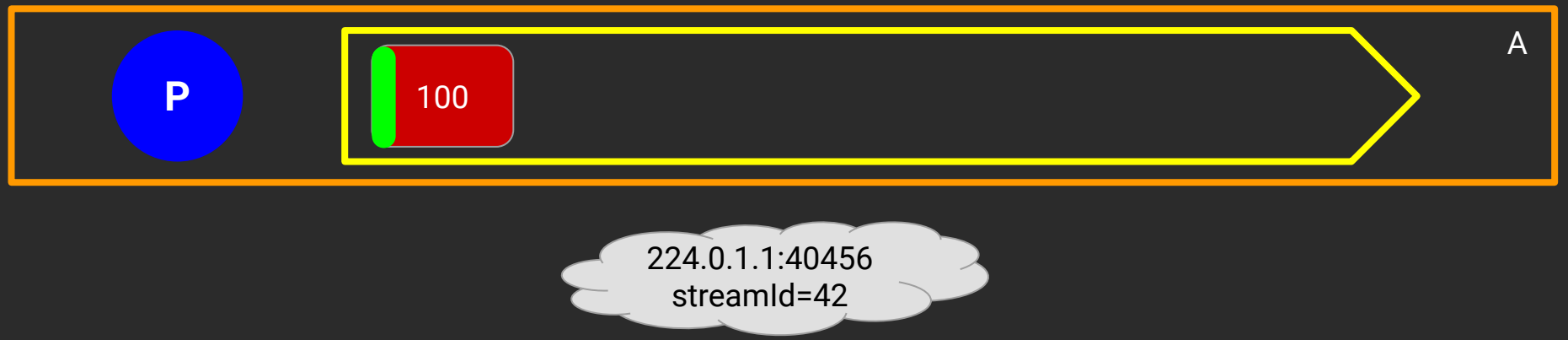
Luxoft
A DXC Technology Company

Multicast

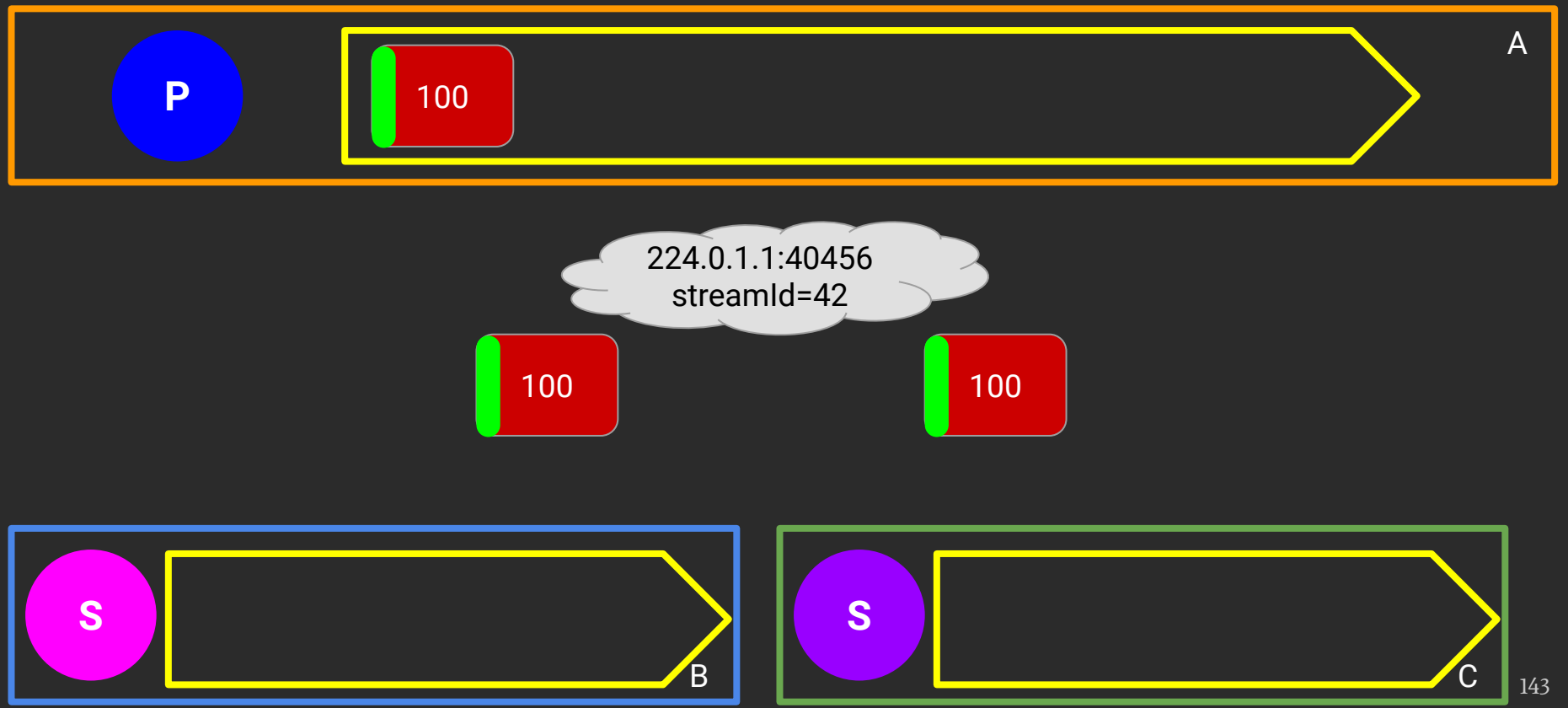
Протокол Aeron. Multicast



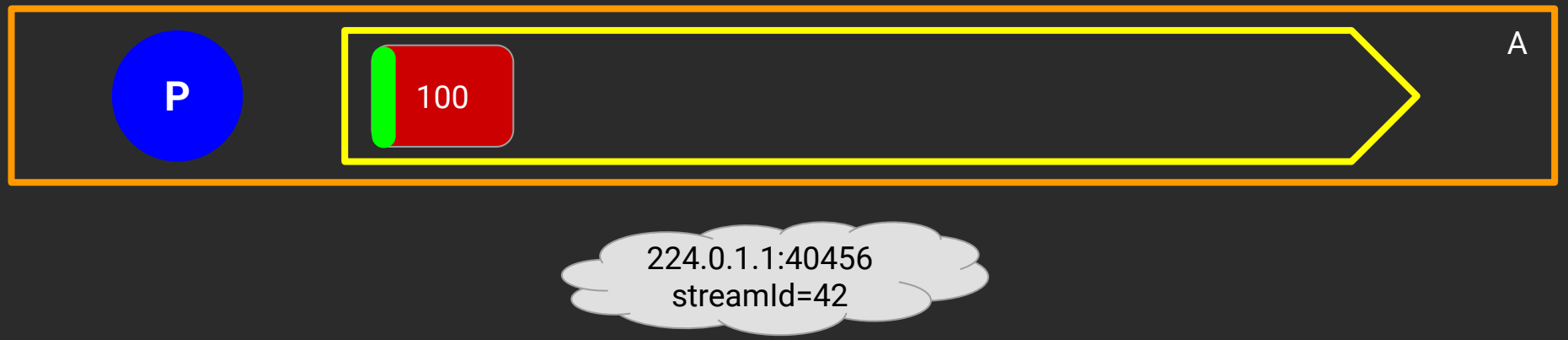
Протокол Aeron. Multicast



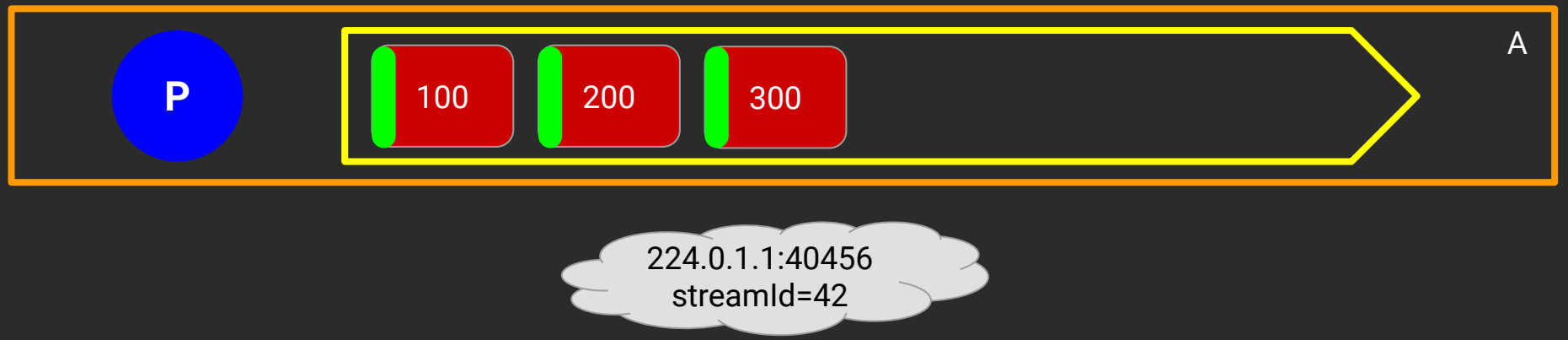
Протокол Aeron. Multicast



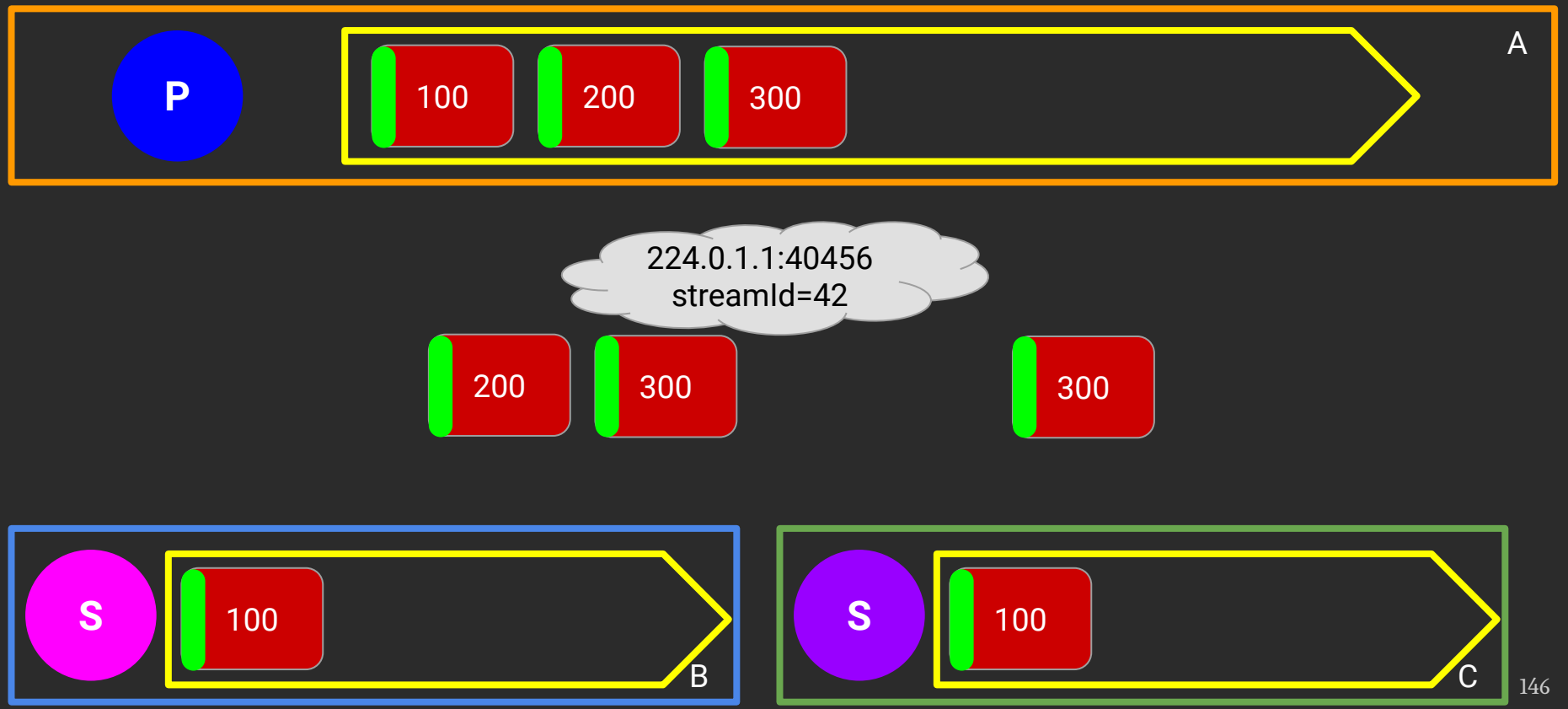
Протокол Aeron. Multicast



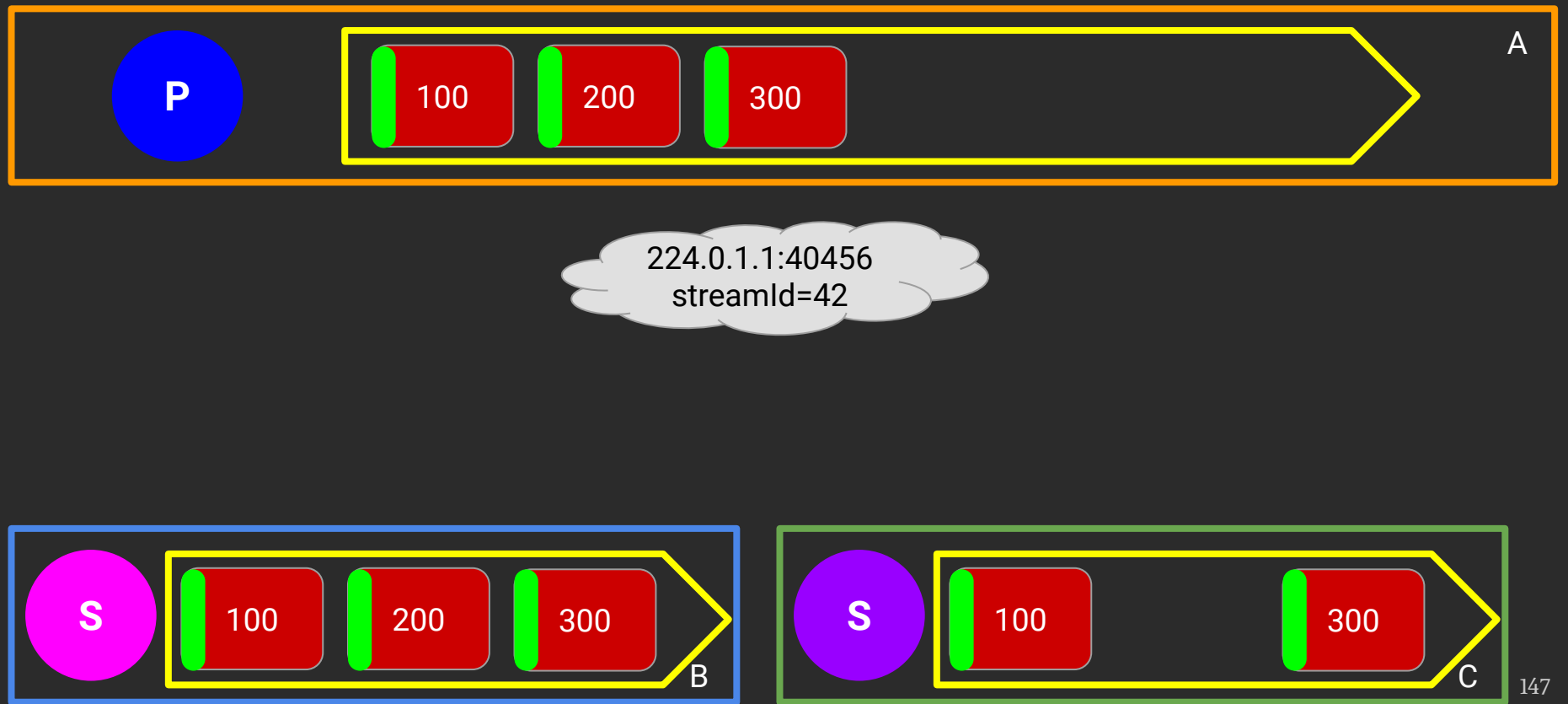
Протокол Aeron. Multicast



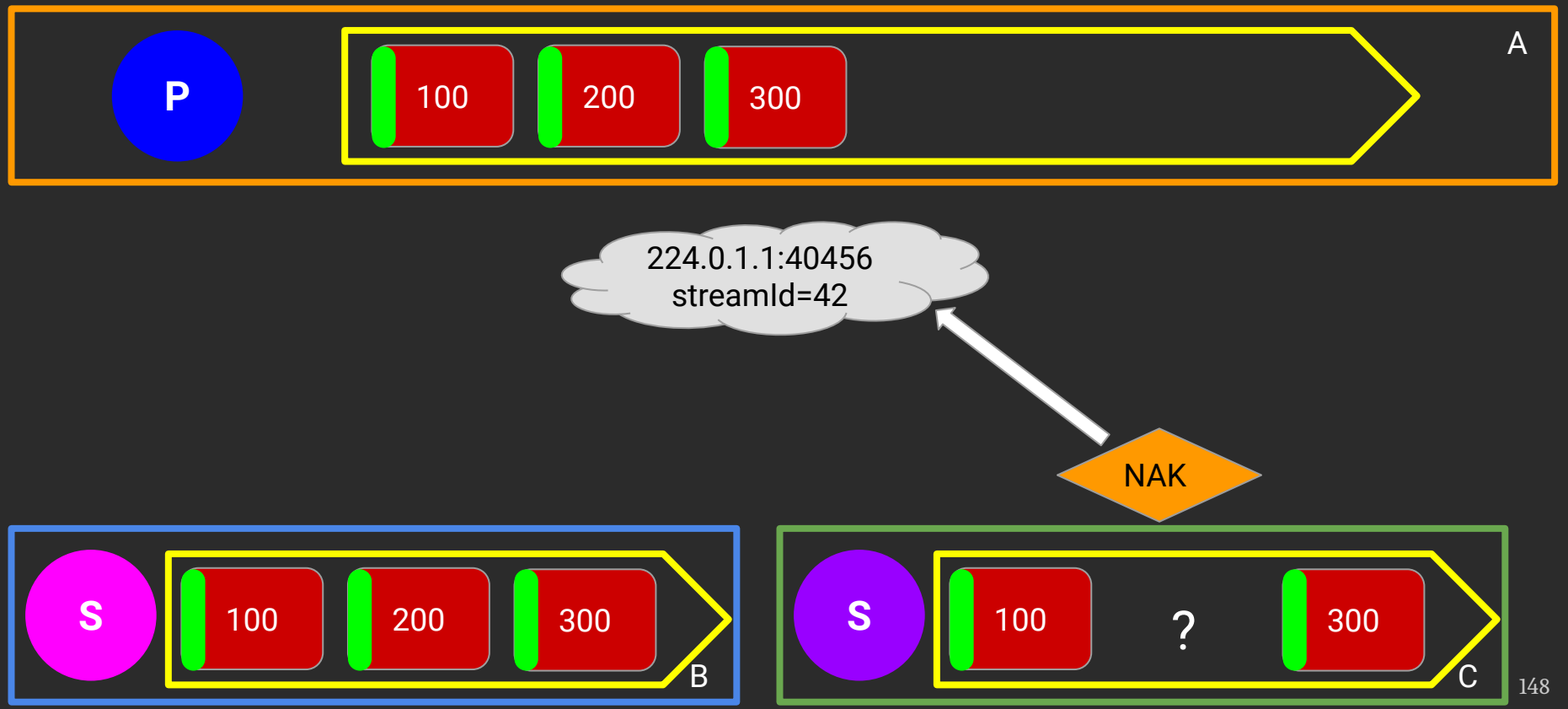
Протокол Aeron. Multicast



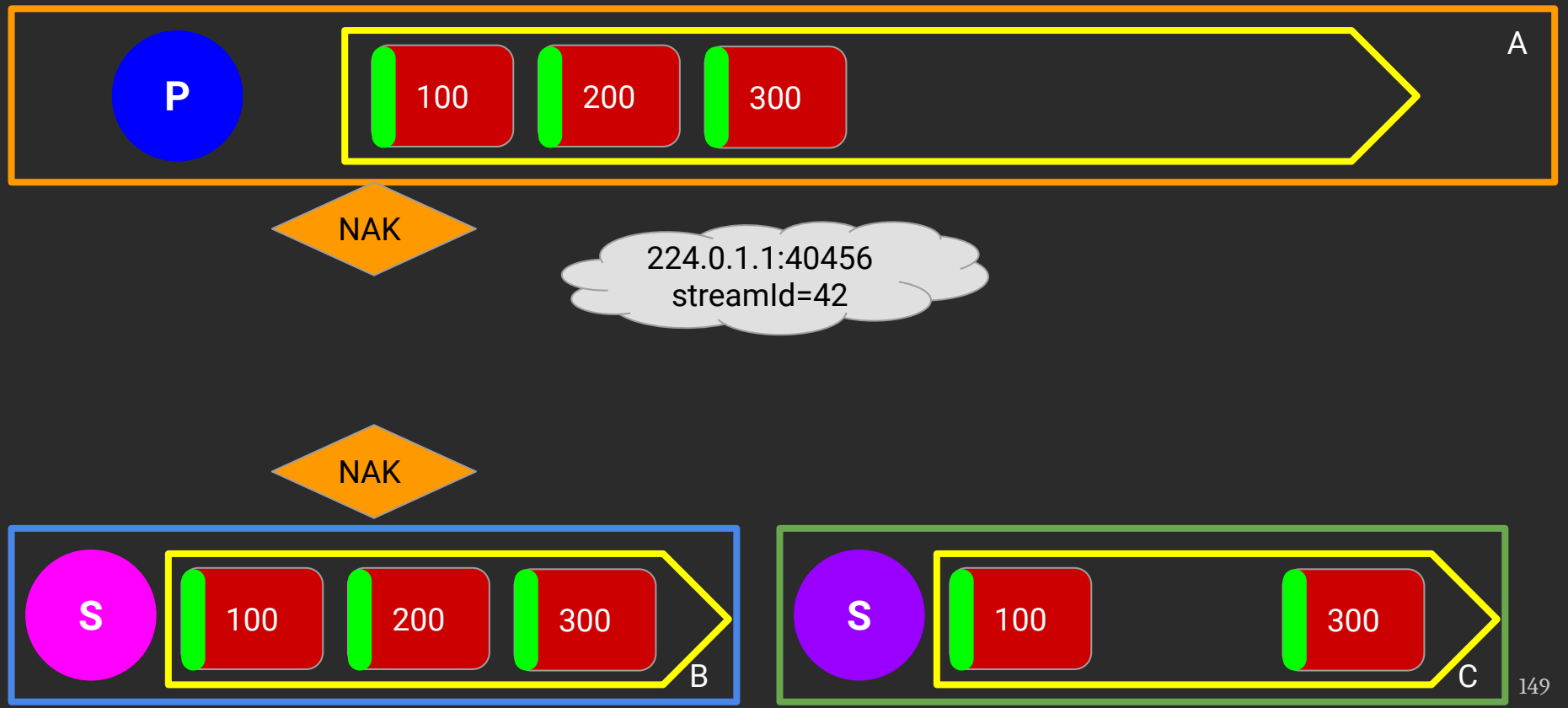
Протокол Aeron. Multicast



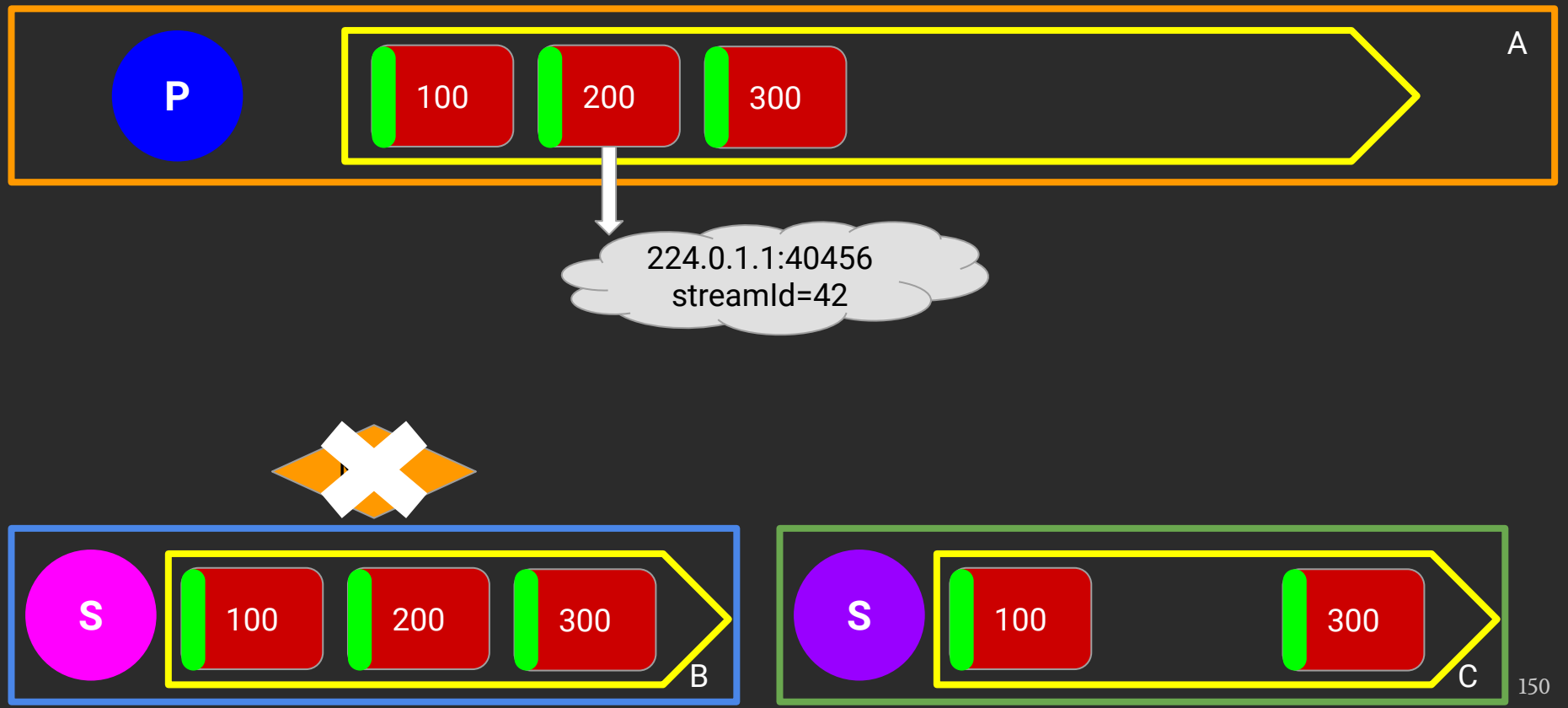
Протокол Aeron. Multicast



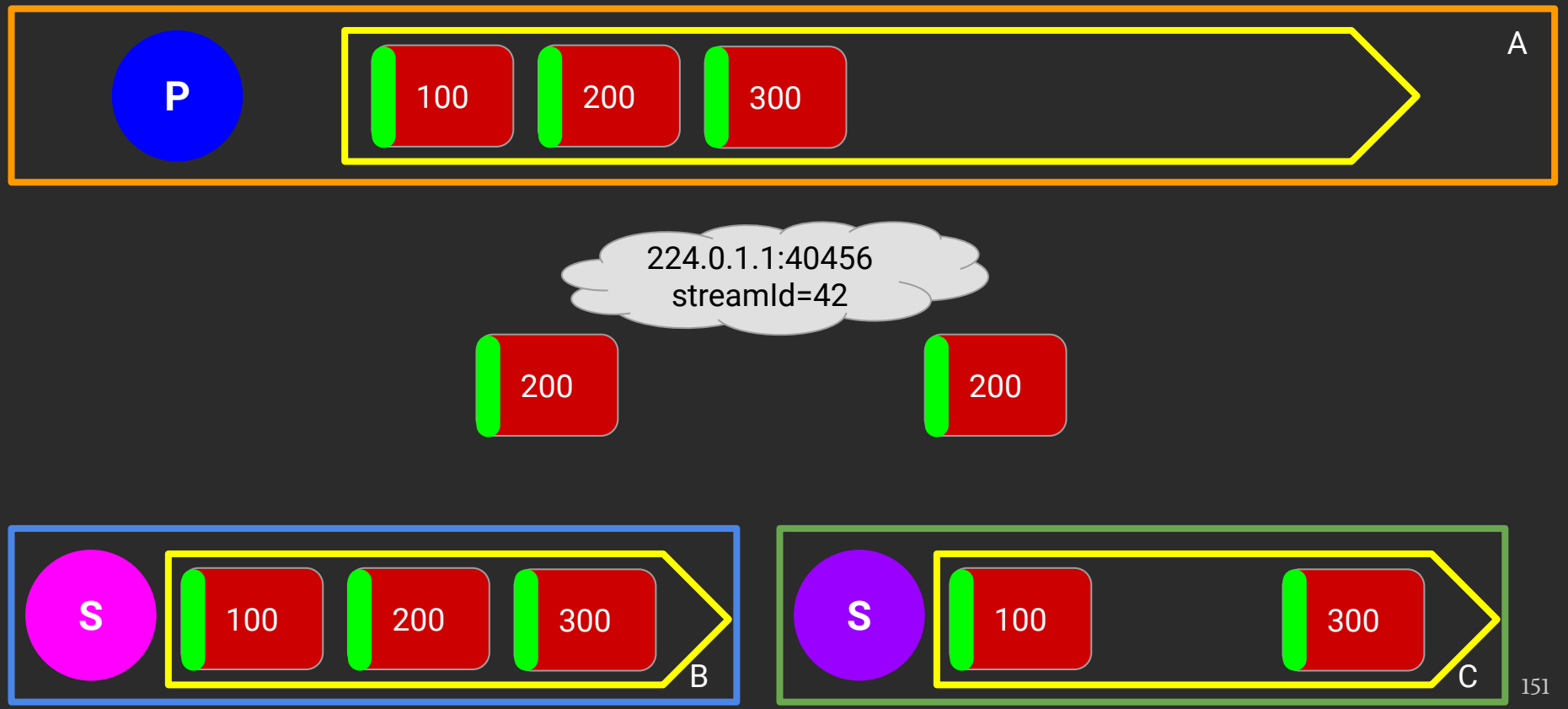
Протокол Aeron. Multicast



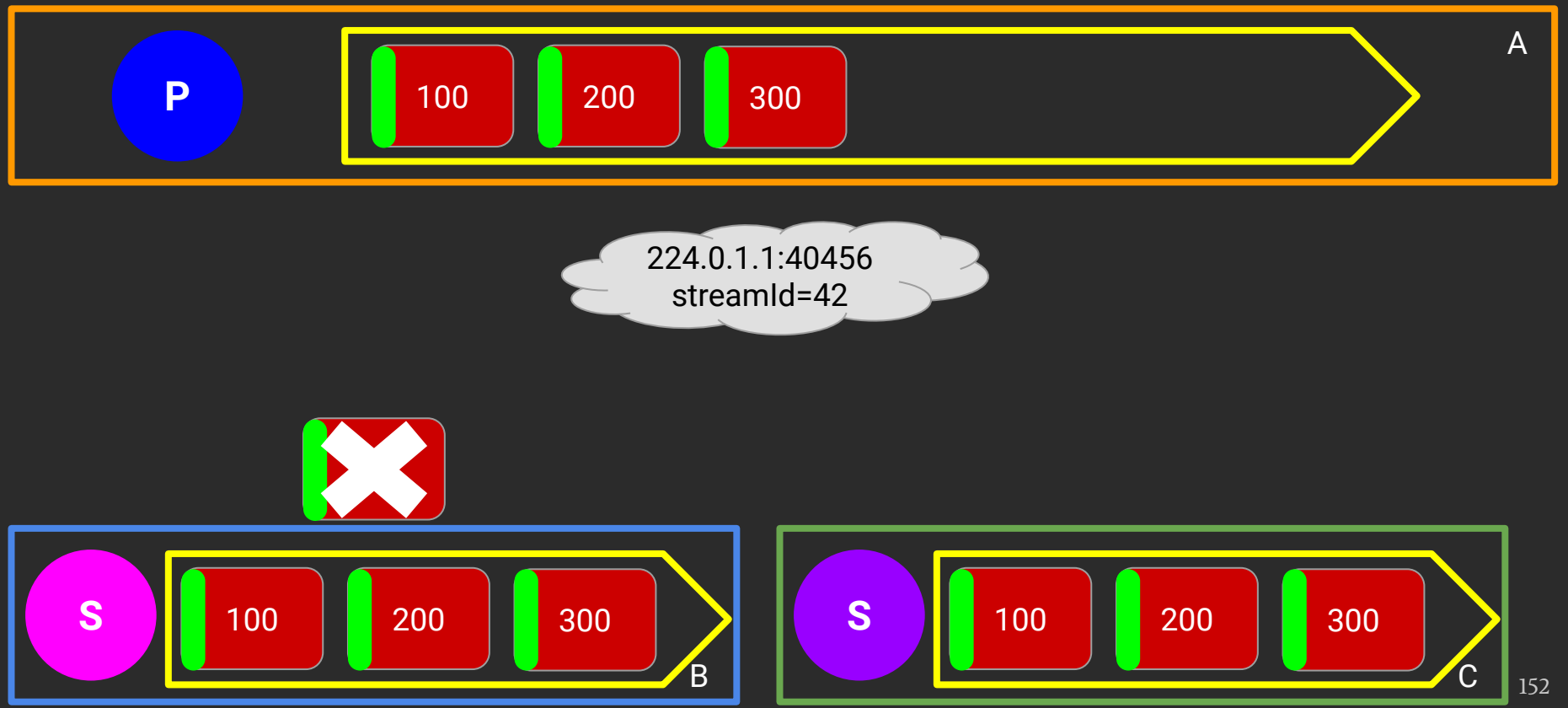
Протокол Aeron. Multicast



Протокол Aeron. Multicast

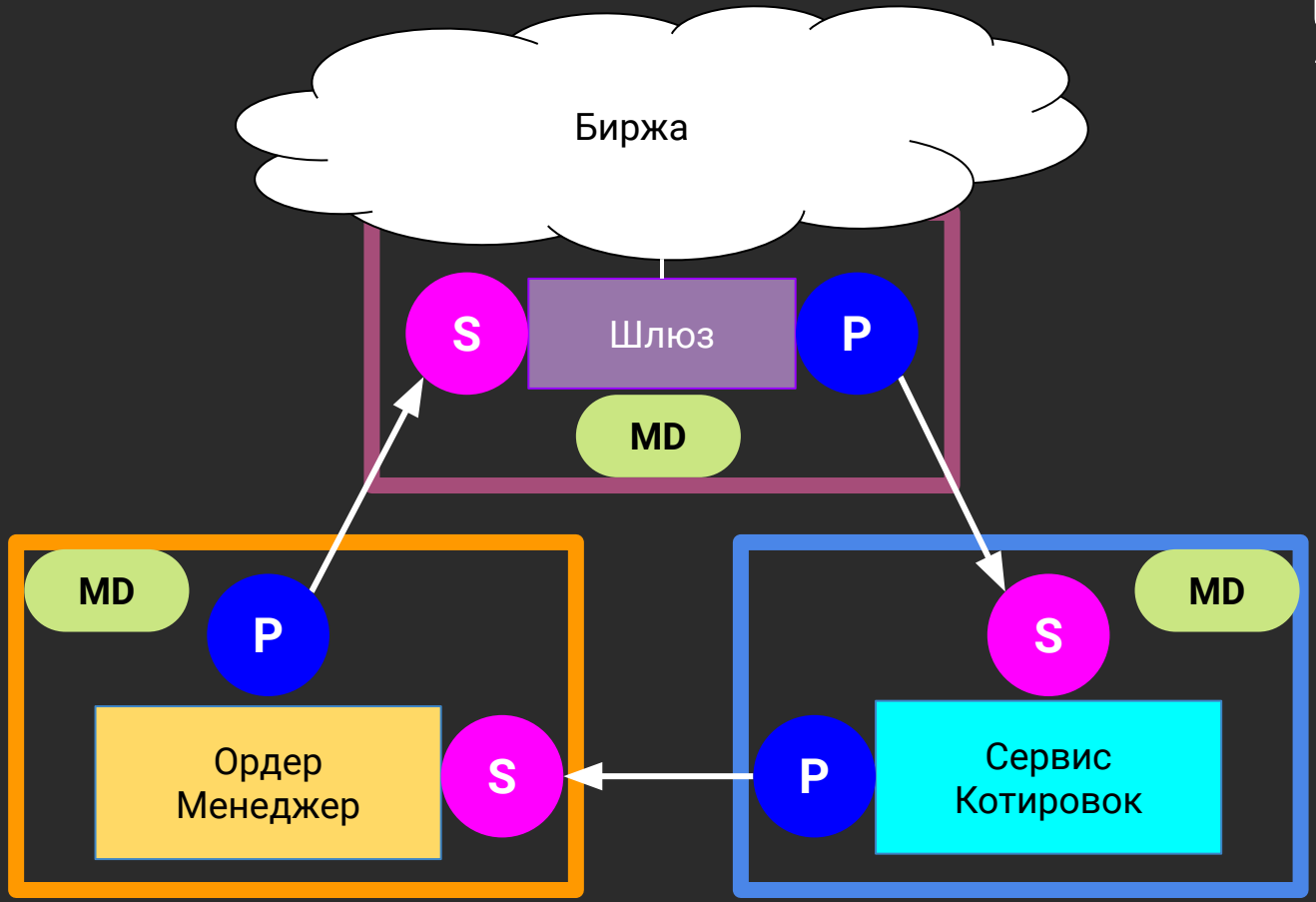


Протокол Aeron. Multicast



Luxoft
A DXC Technology Company

Архитектура. Резюме.




Срочно продавай снег!!! Продавай!!!
Продавай!!!
Продавай!
ПРОДАВАЙ!

Биржа
S
Шлюз
P
MD
MD
P
Ордер Менеджер
S
MD
S
P
Сервис Котировок

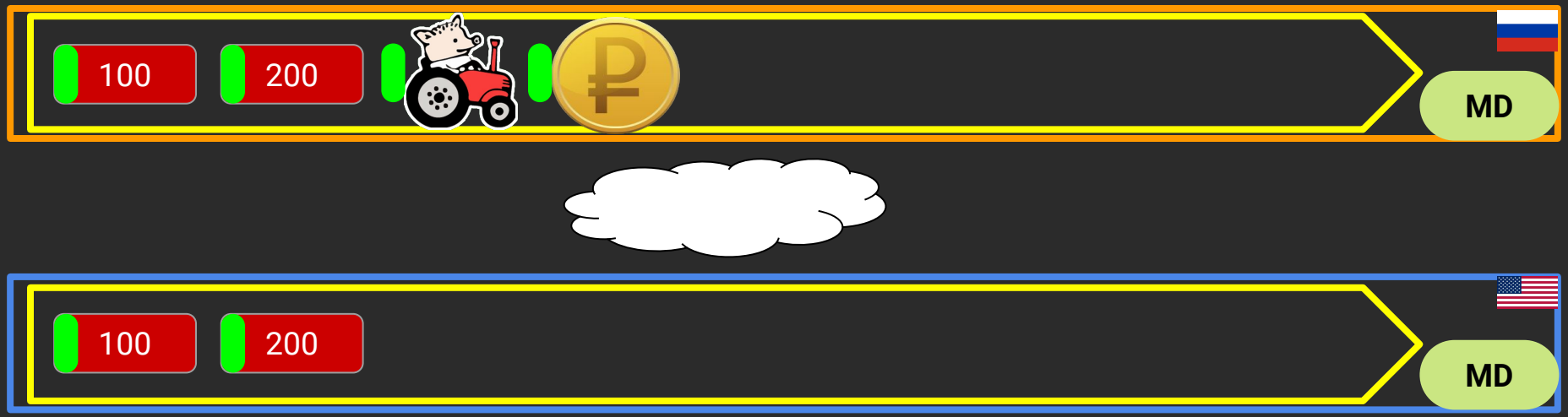
Архитектура. Резюме

```
result = publication.offer(      )
```

 = 30 рублей

Архитектура. Резюме

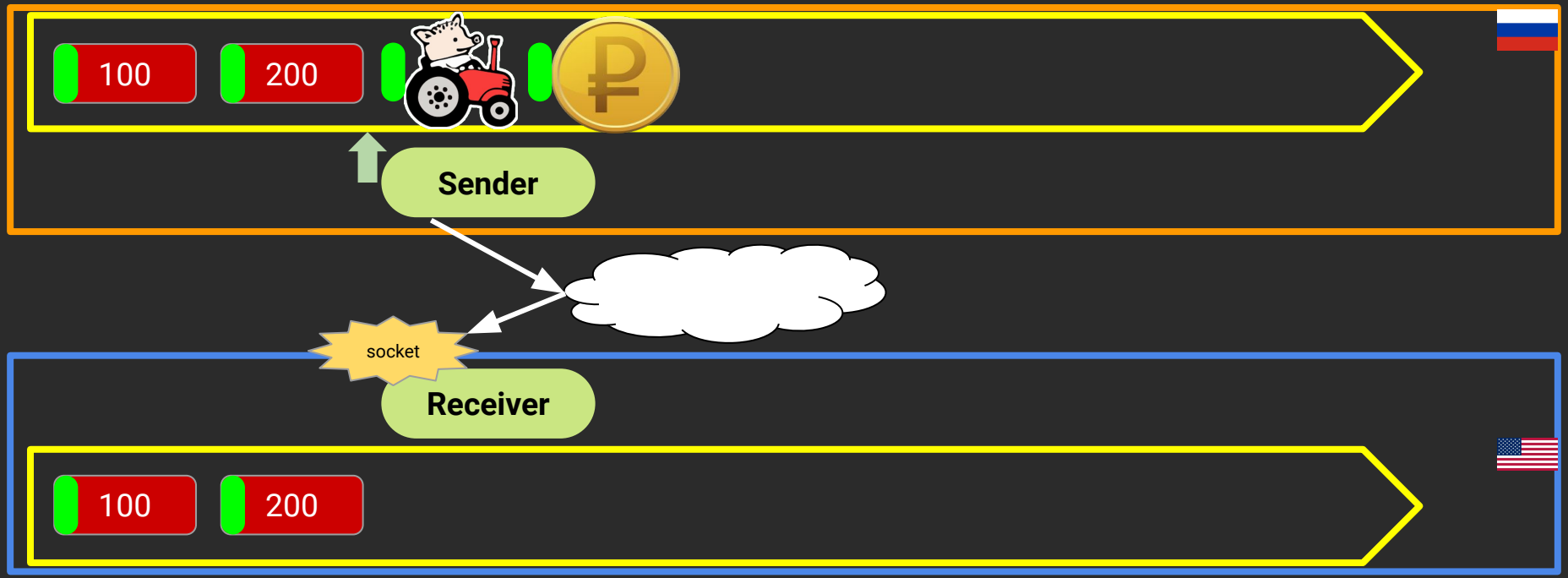
```
result = 300
```



$ = 30 рублей

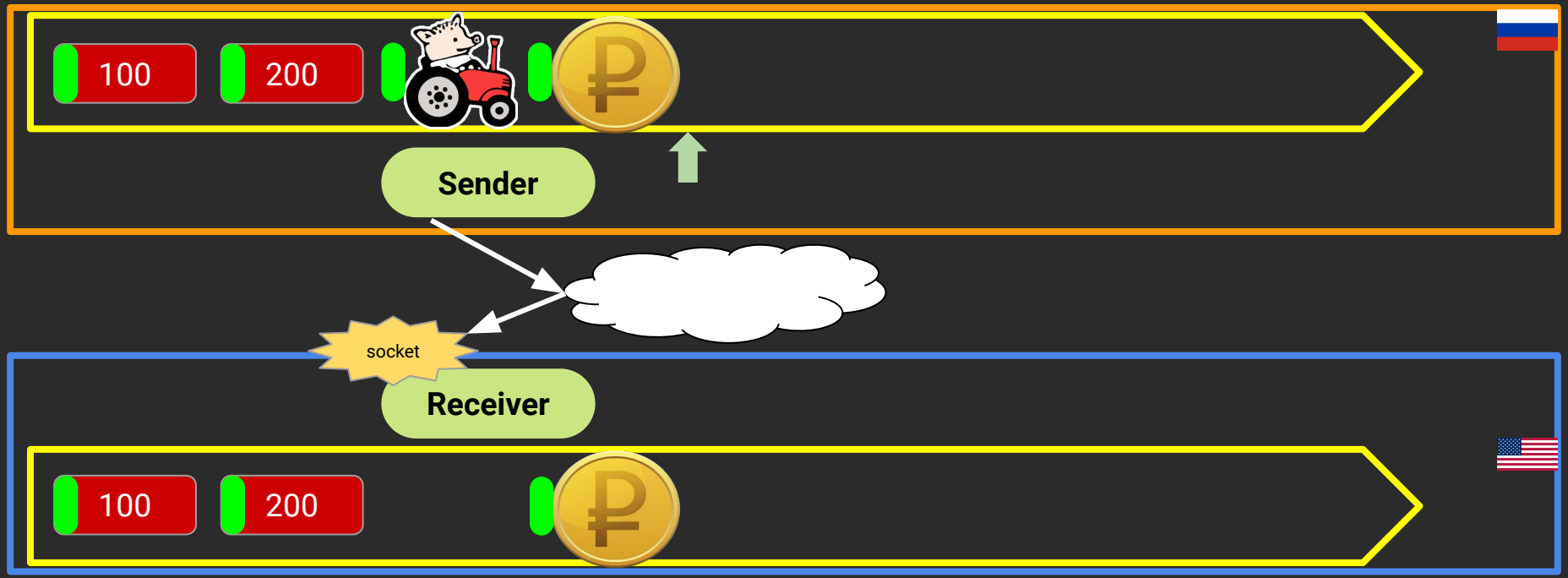
Архитектура. Резюме

```
result = 300
```



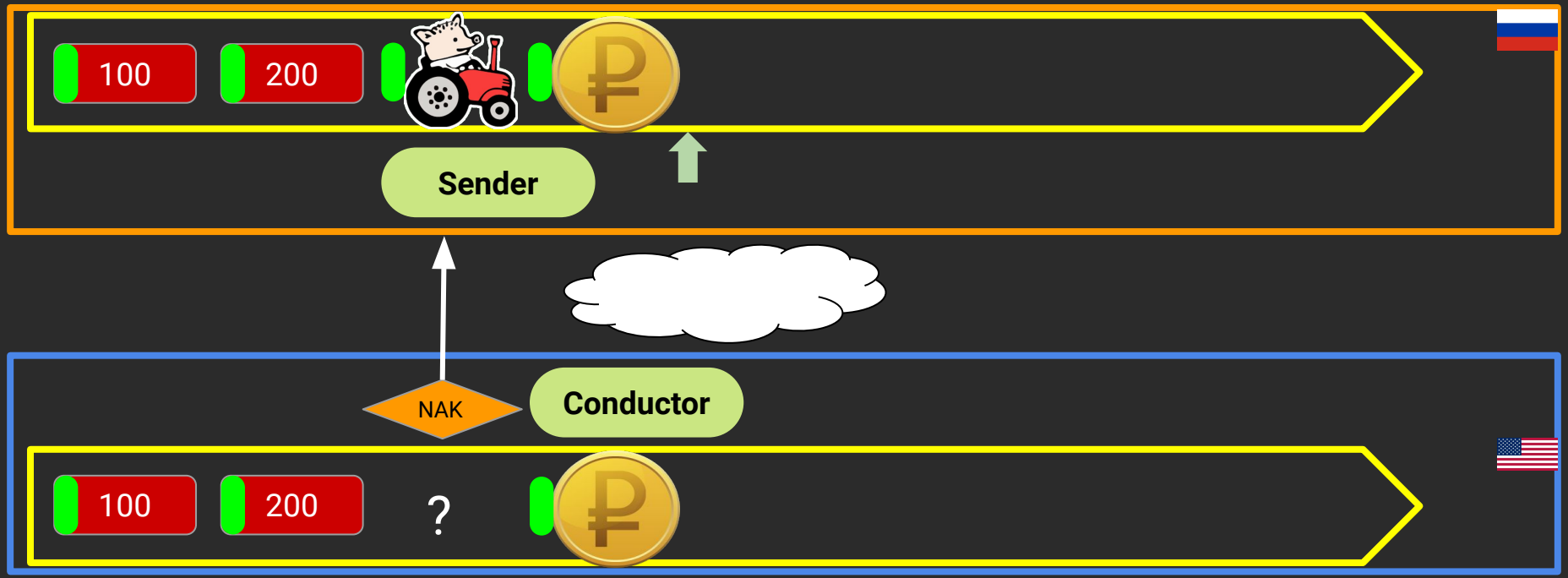
Архитектура. Резюме

```
result = 300
```



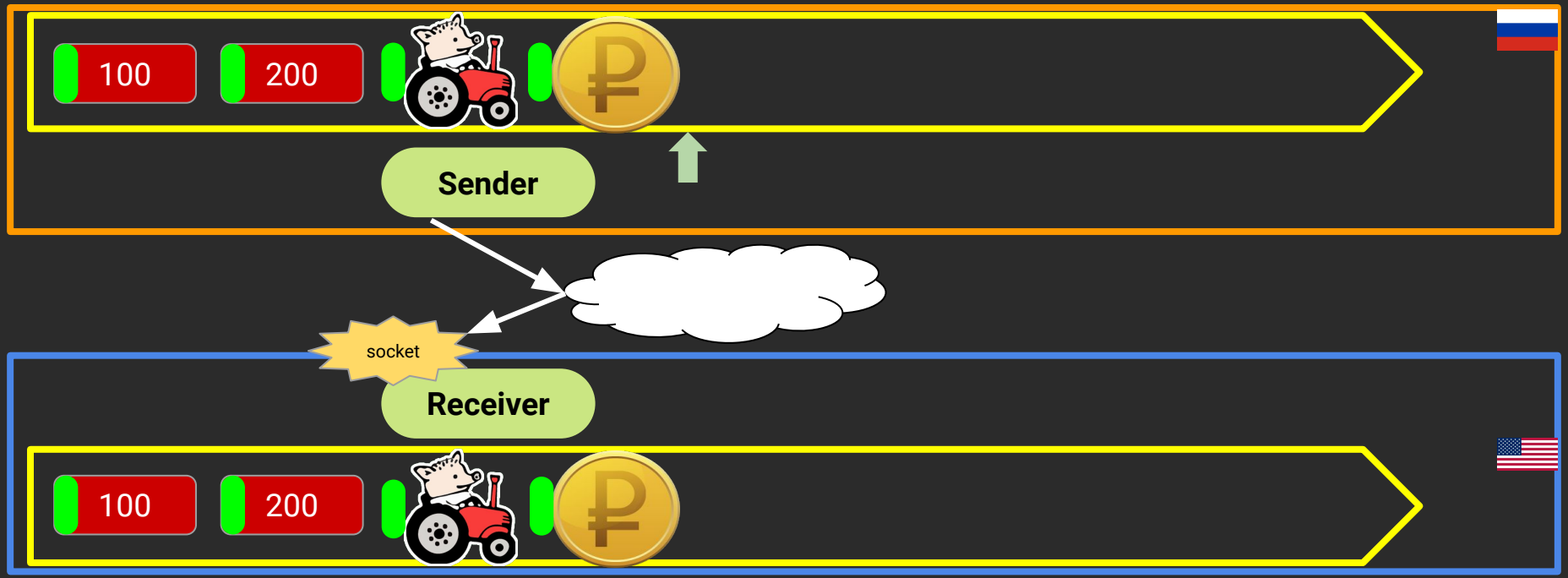
Архитектура. Резюме

```
result = 300
```



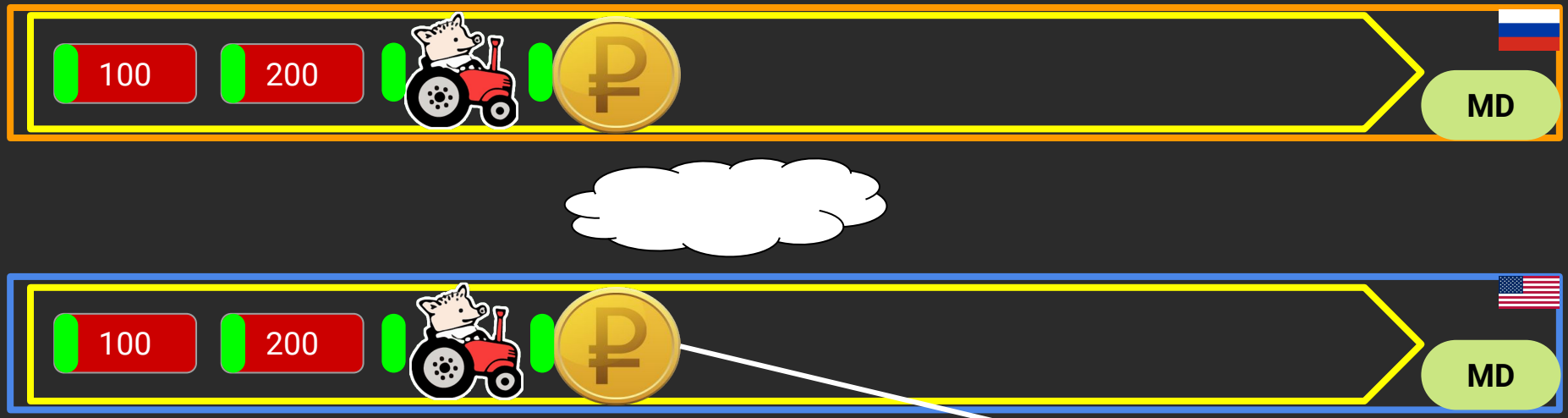
Архитектура. Резюме

```
result = 300
```



Архитектура. Резюме

result = 300




```
FragmentHandler handler = new FragmentAssembler((b, o, l, h) -> { купить });
while (isRunning()) {
    idleStrategy.idle(subscription.poll(handler, LIMIT));
}
```

Архитектура. Резюме

result = 300


```
FragmentHandler handler = new FragmentAssembler((b, o, l, h) -> {                    });
while (isRunning()) {
    idleStrategy.idle(subscription.poll(handler, LIMIT));
}
```

Luxoft
A DXC Technology Company

?

Luxoft
A DXC Technology Company

Имплементация Image
Log buffer

Имплементация Image. Log buffer

Имплементация Image. Log buffer

Luxoft
A DXC Technology Company

Tractorist = 300 byte

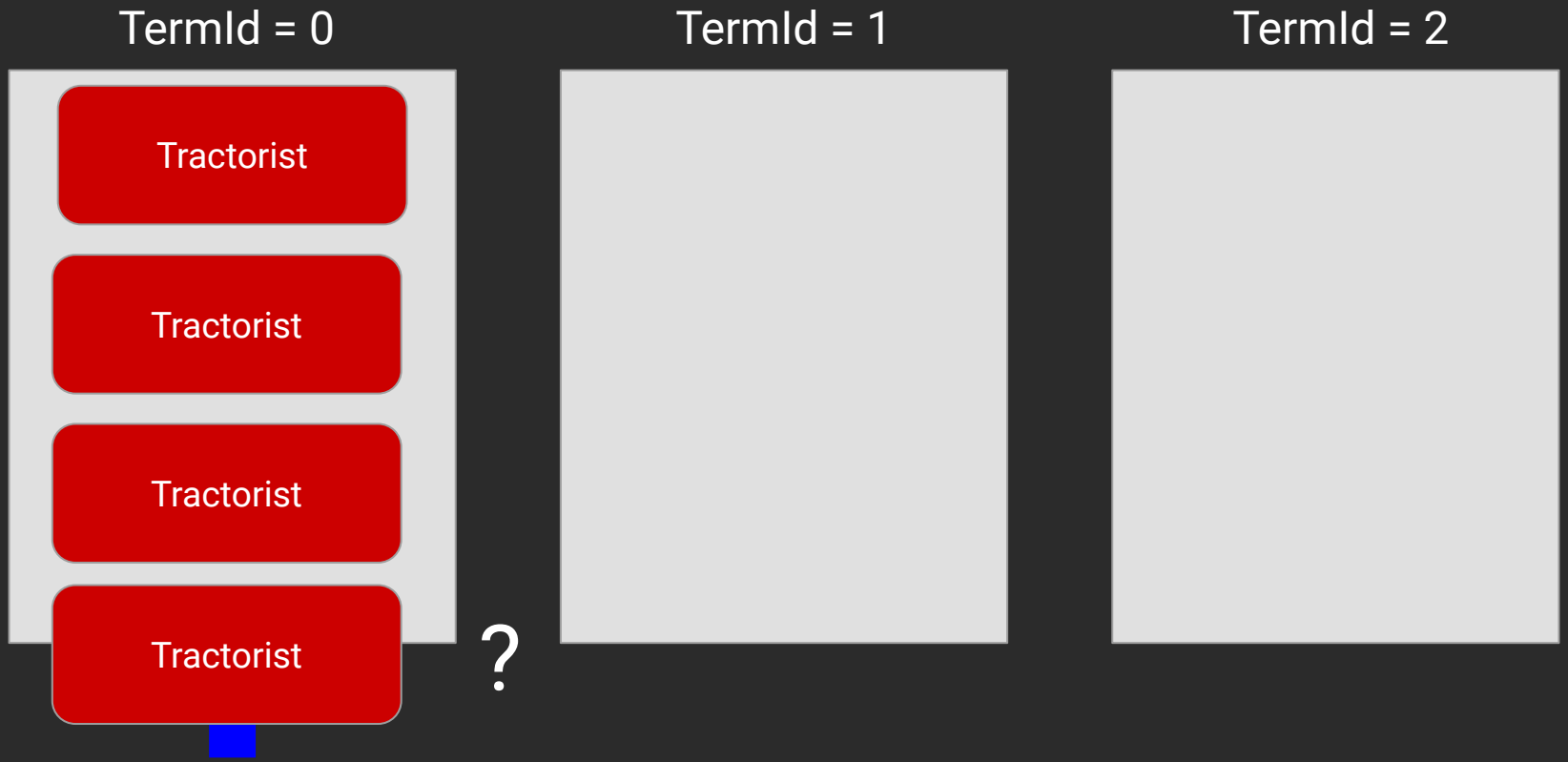
Имплементация Image. Log buffer

Имплементация Image. Log buffer

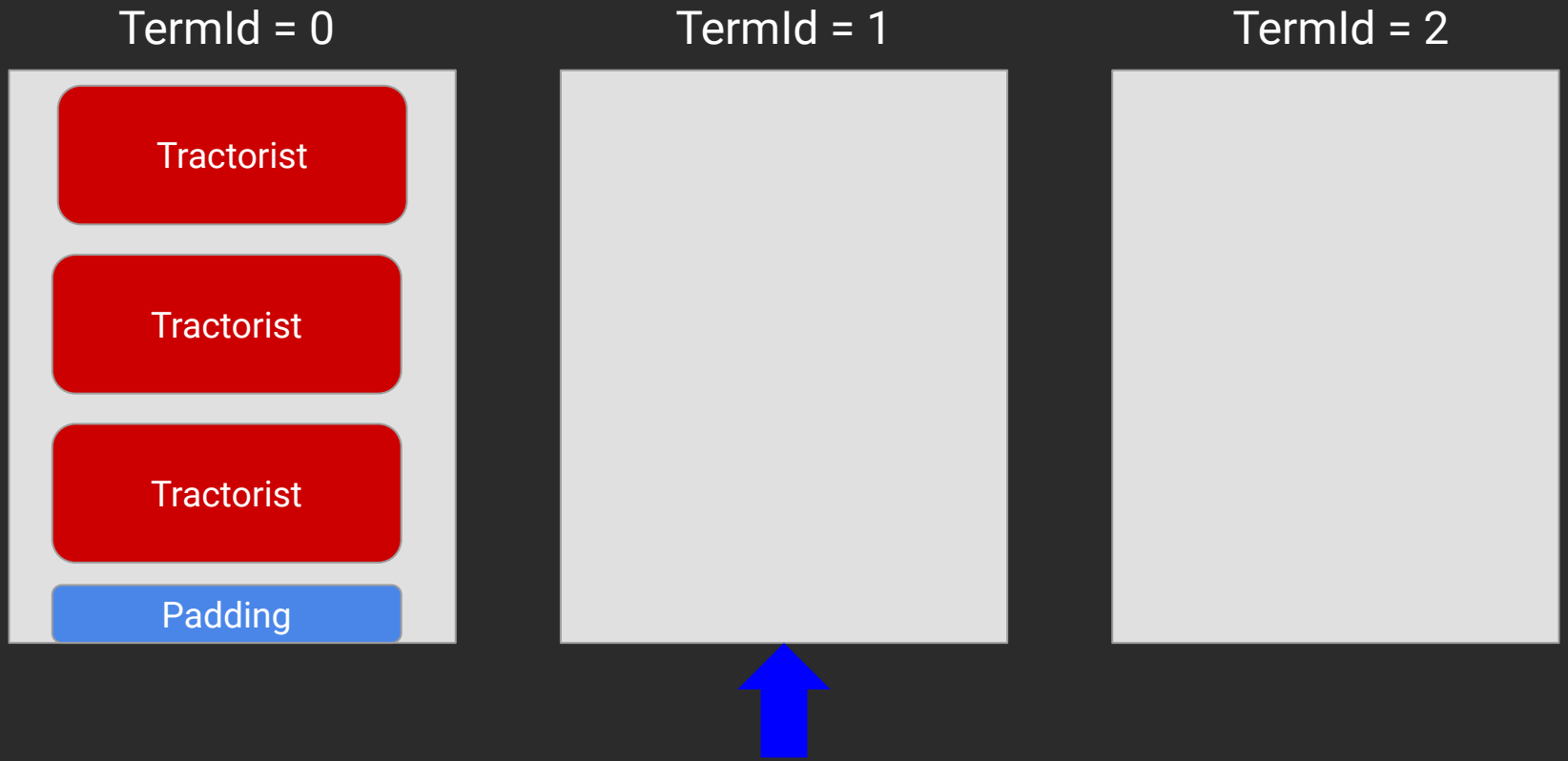
Имплементация Image. Log buffer

Имплементация Image. Log buffer

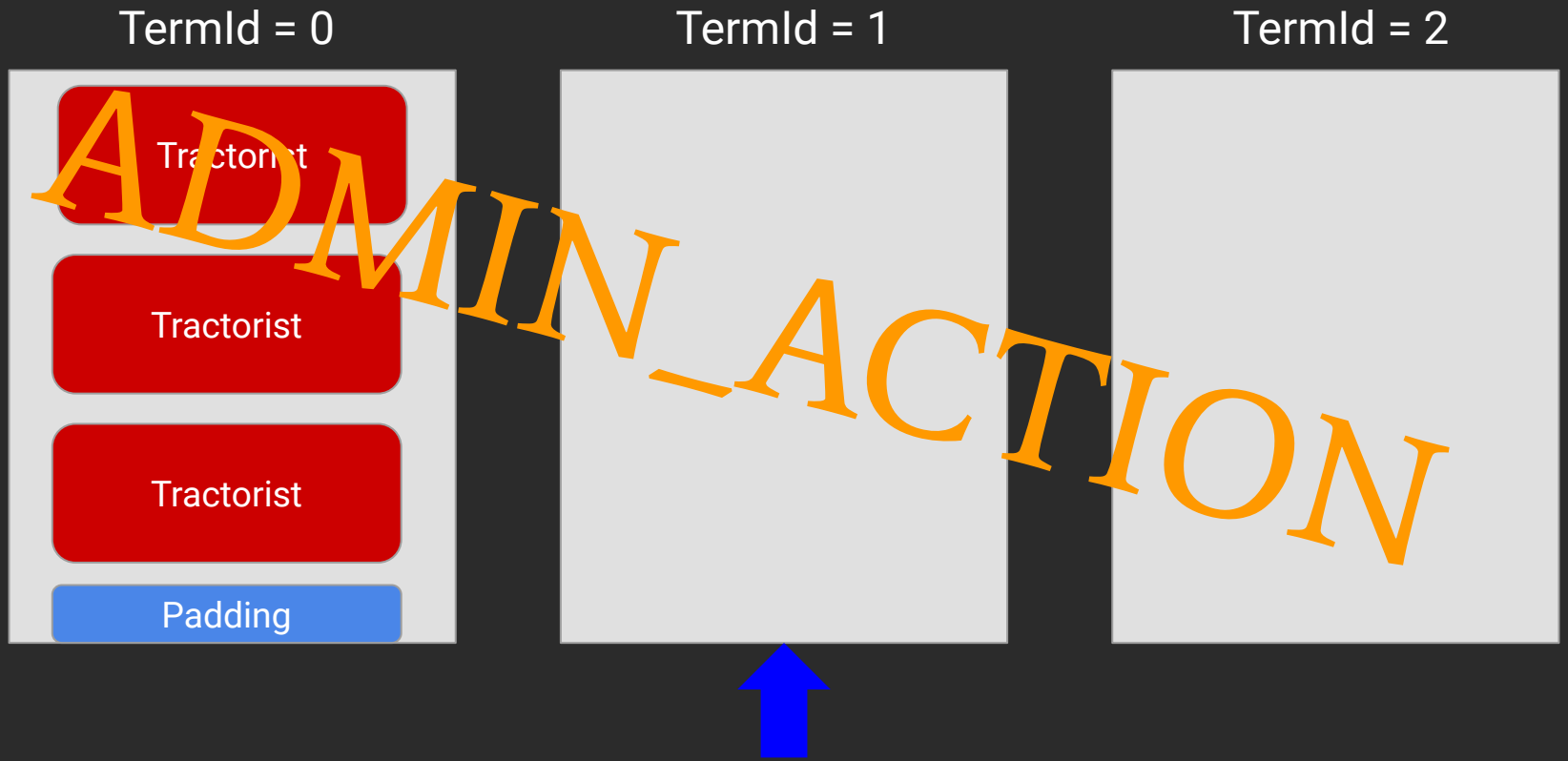
Имплементация Image. Log buffer



Имплементация Image. Log buffer



Имплементация Image. Log buffer

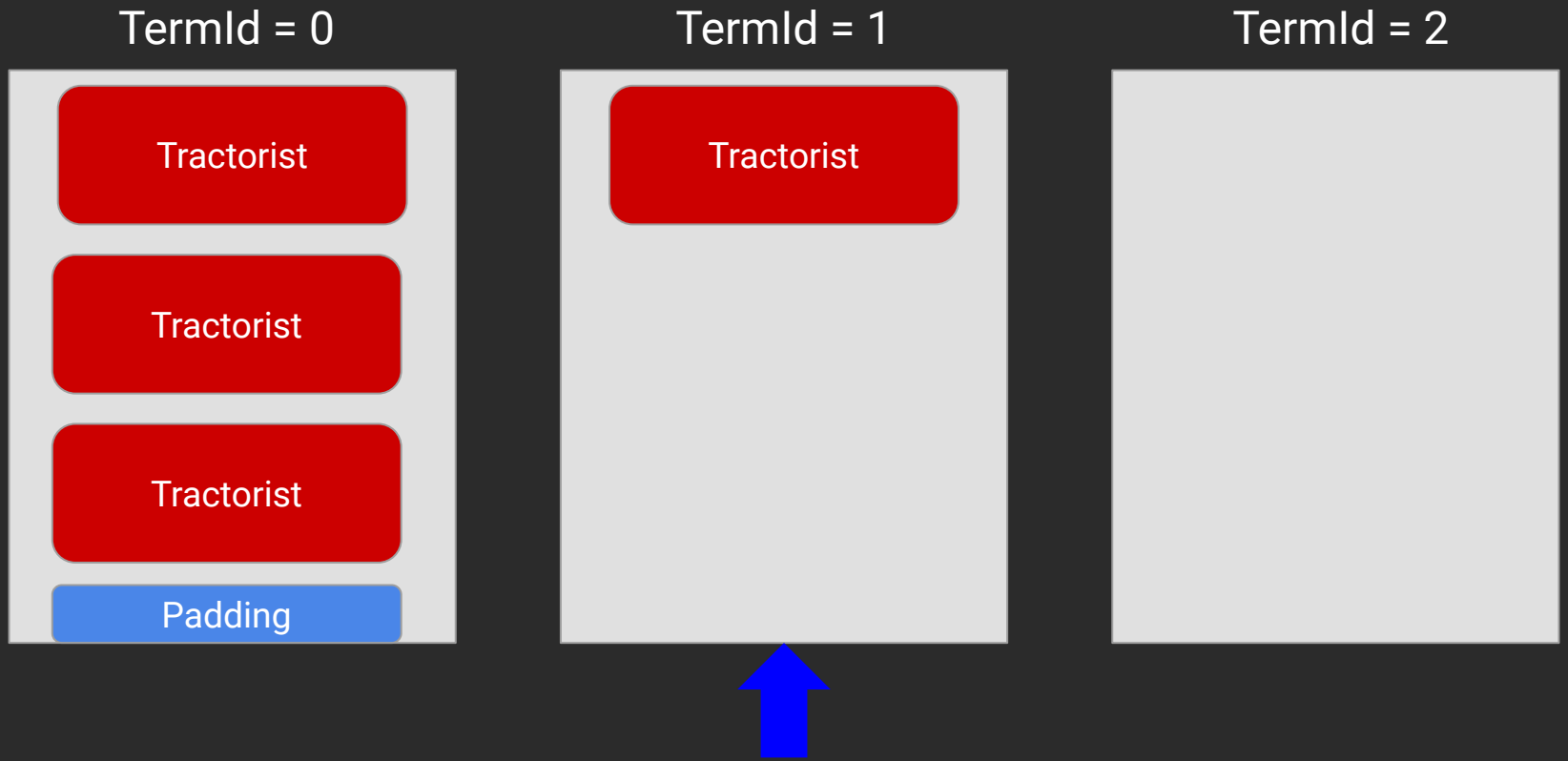


Luxoft
A DXC Technology Company

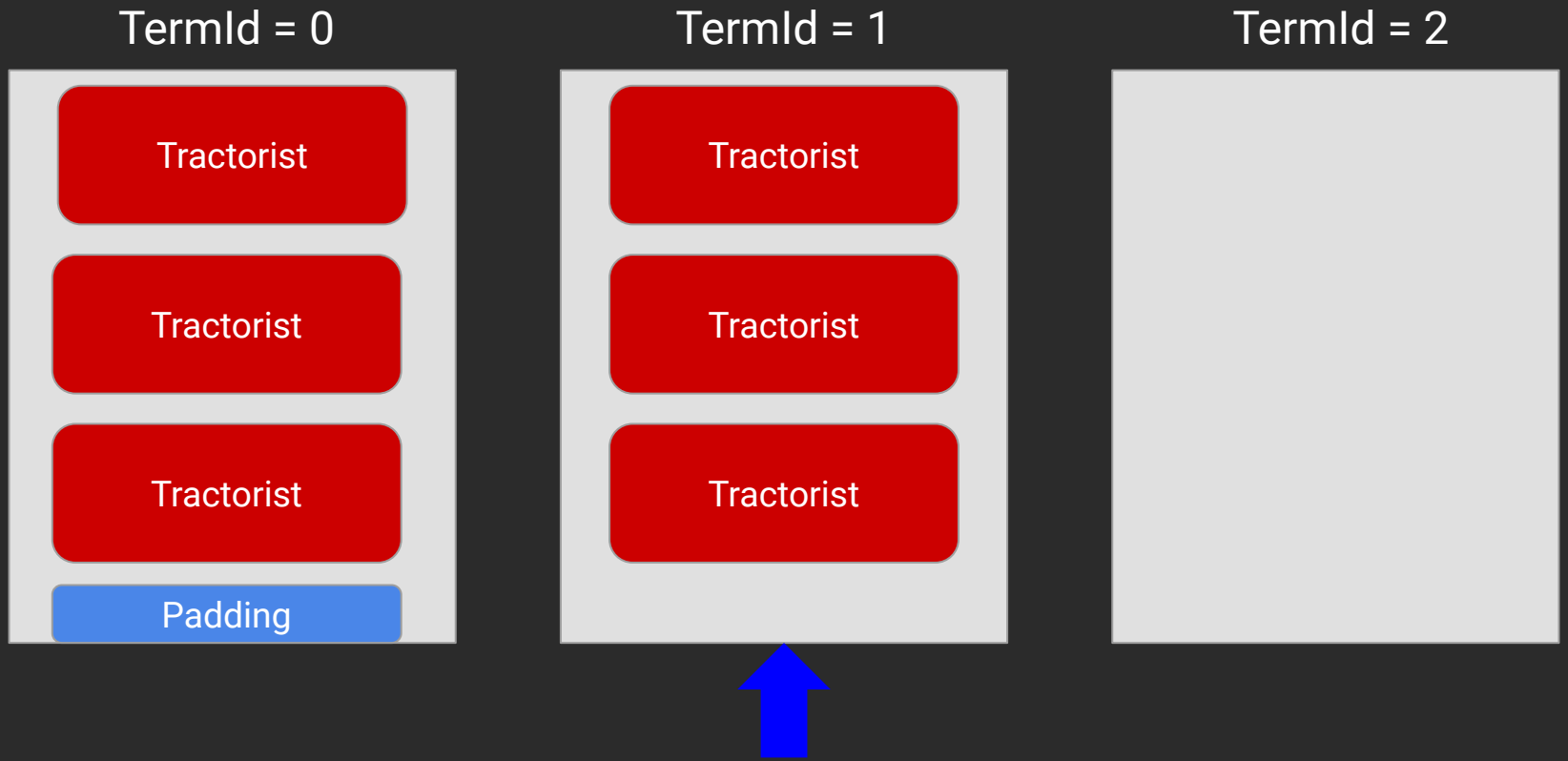
TCP's philosophy is "better late than never".
Many latency-sensitive applications prefer the "better never than late" philosophy.

https://www.informatica.com/downloads/1568_high_perf_messaging_wp/Topics-in-High-Performance-Messaging.htm

Имплементация Image. Log buffer

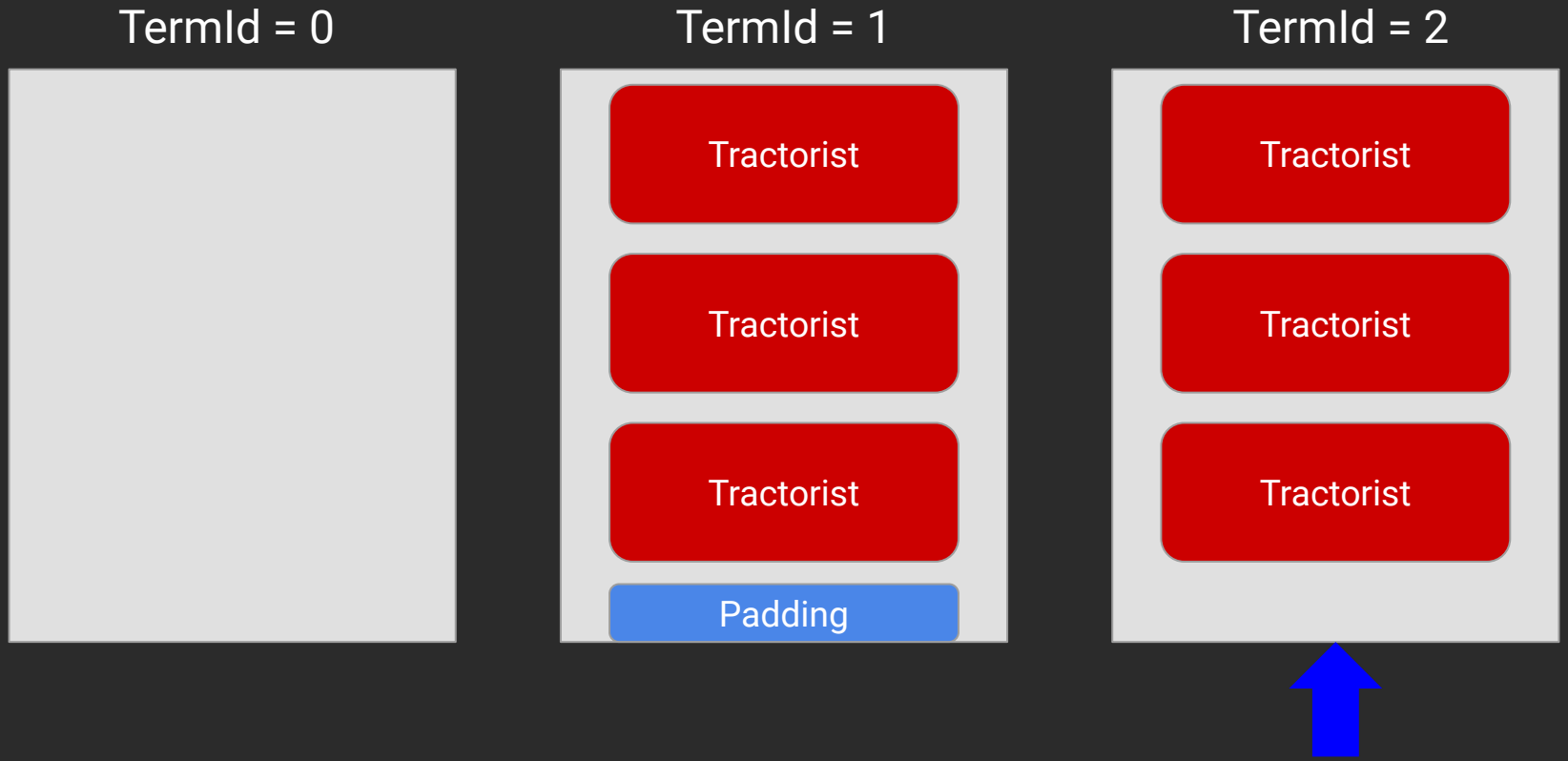


Имплементация Image. Log buffer



Имплементация Image. Log buffer

Имплементация Image. Log buffer



Имплементация Image. Log buffer

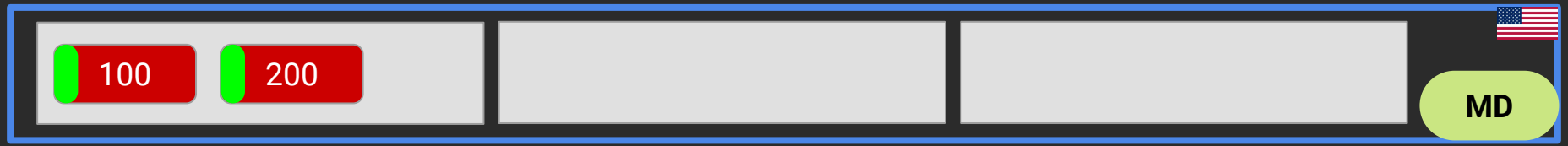
Имплементация Image. Log buffer

Luxoft
A DXC Technology Company

Резюме по Aeron

Архитектура. Резюме

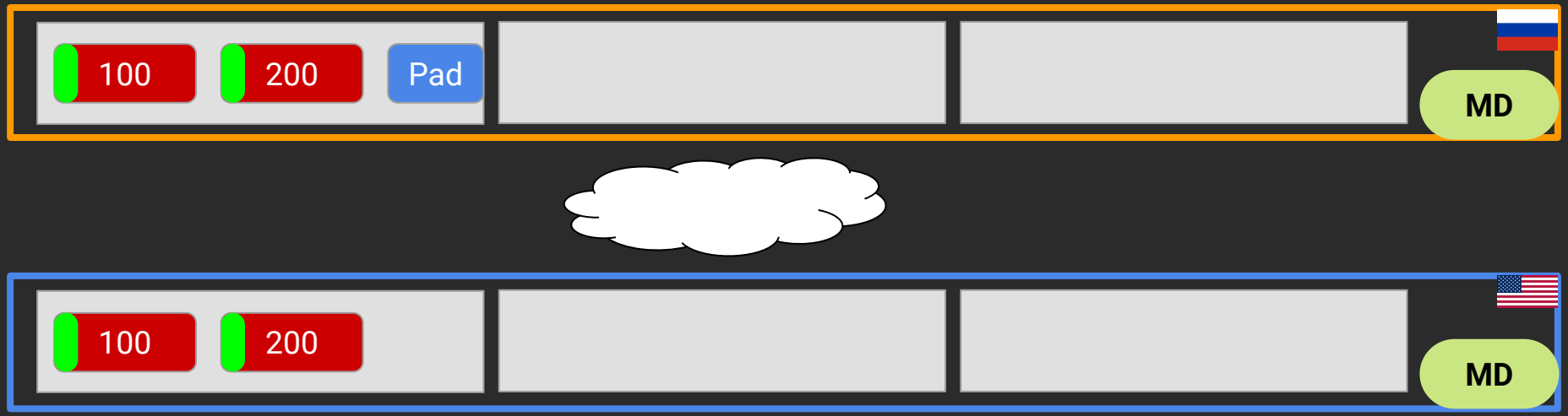
```
result = publication.offer(      )
```



 = 40 рублей

Архитектура. Резюме

result = ADMIN_ACTION



$ = 40 рублей

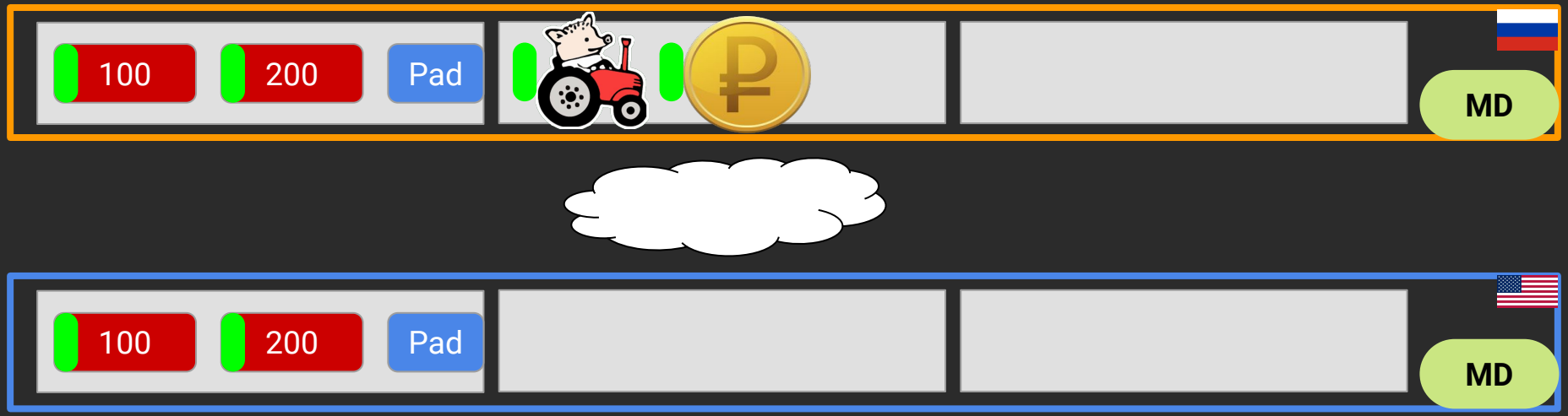
Архитектура. Резюме

```
result = publication.offer(      )
```

 = 30 рублей

Архитектура. Резюме

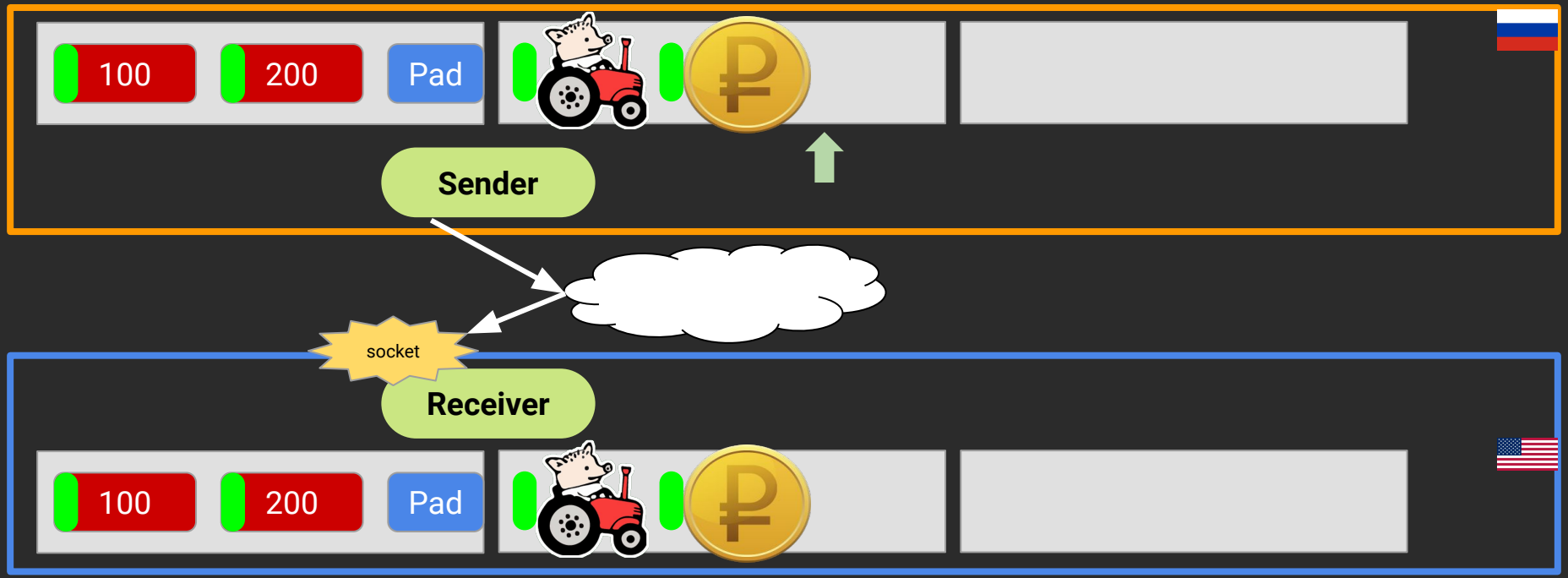
```
result = 300
```



 = 30 рублей

Архитектура. Резюме

```
result = 300
```



Архитектура. Резюме

result = 300


```
FragmentHandler handler = new FragmentAssembler((b, o, l, h) -> { купить });
while (isRunning()) {
    idleStrategy.idle(subscription.poll(handler, LIMIT));
}
```

Архитектура. Резюме

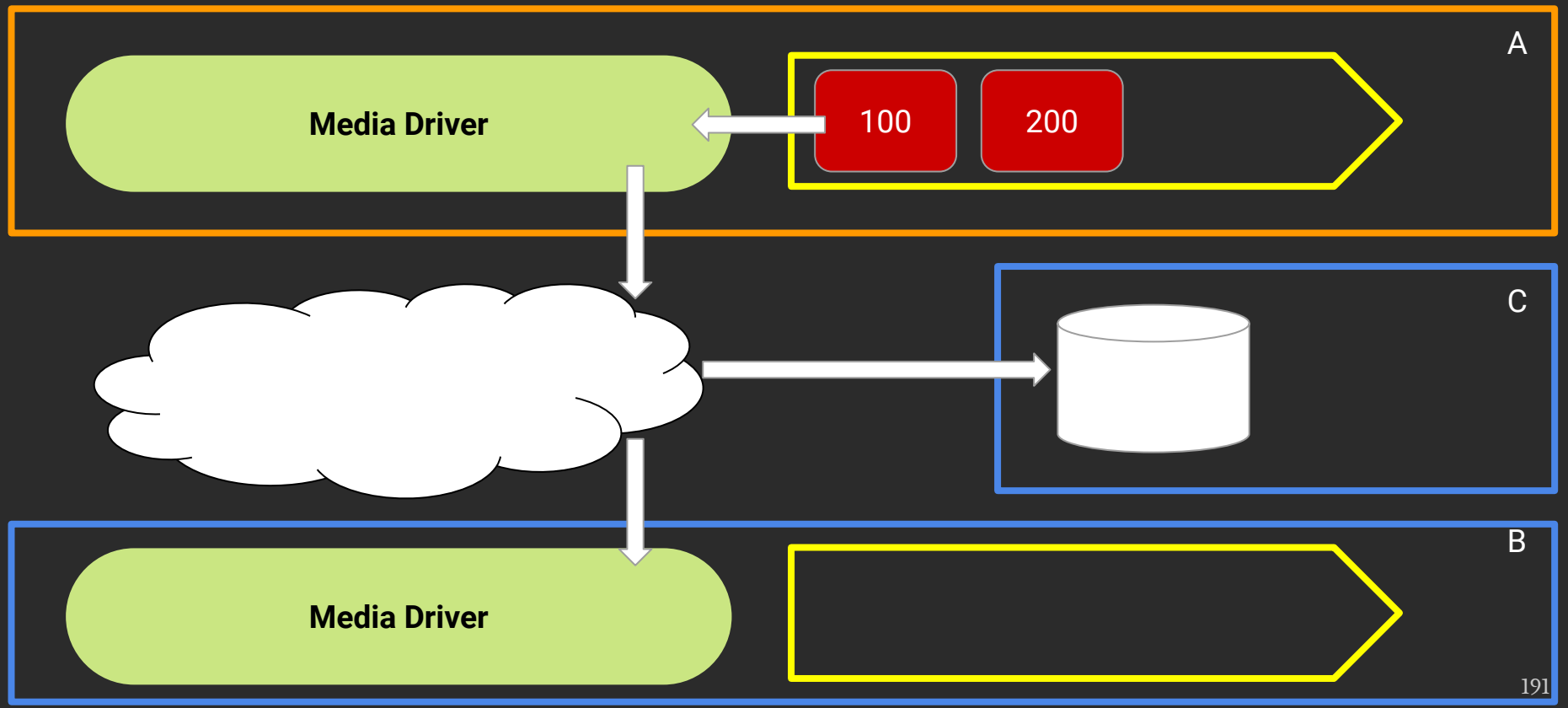
result = 300


```
FragmentHandler handler = new FragmentAssembler((b, o, l, h) -> {                    });
while (isRunning()) {
    idleStrategy.idle(subscription.poll(handler, LIMIT));
}
```

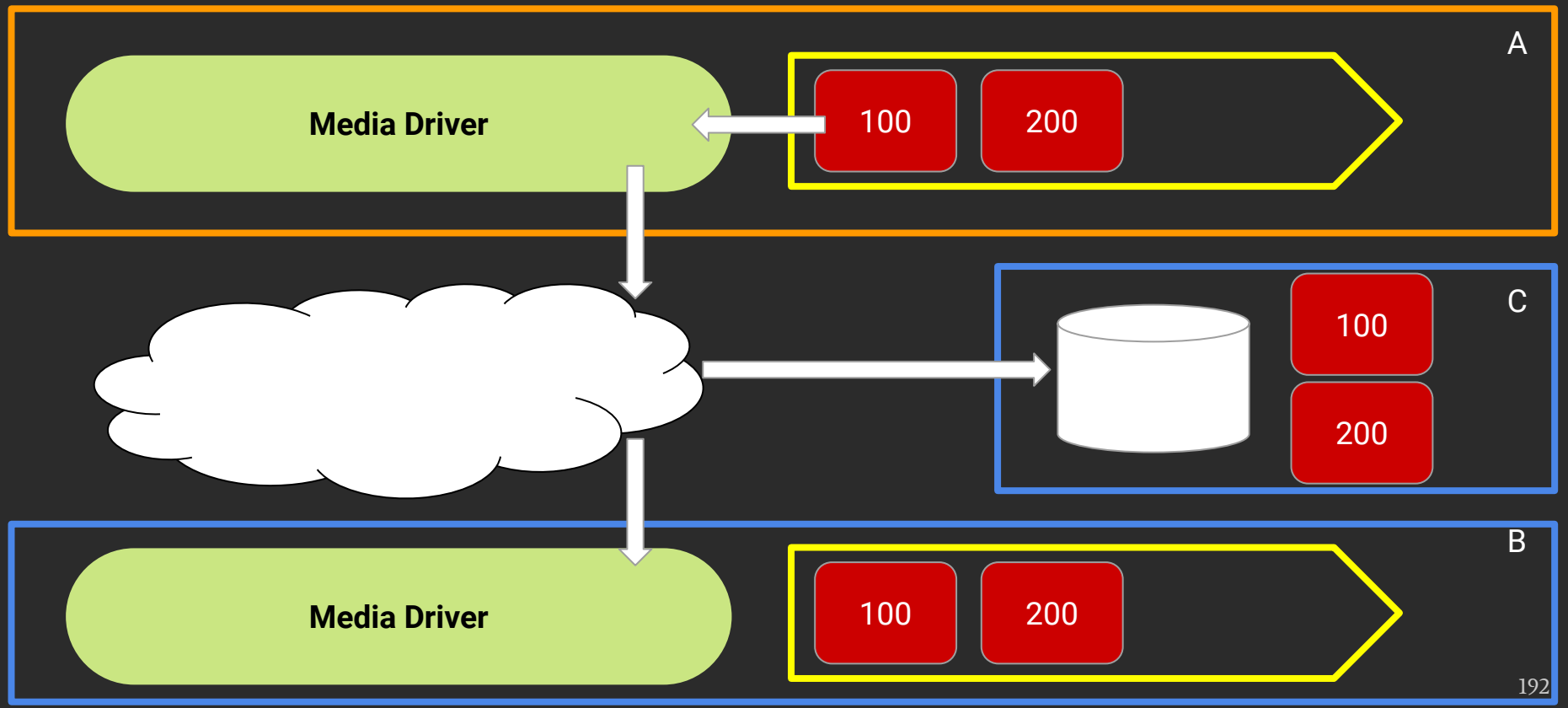
Luxoft
A DXC Technology Company

Что там есть ещё?

Что там есть ещё? Archive



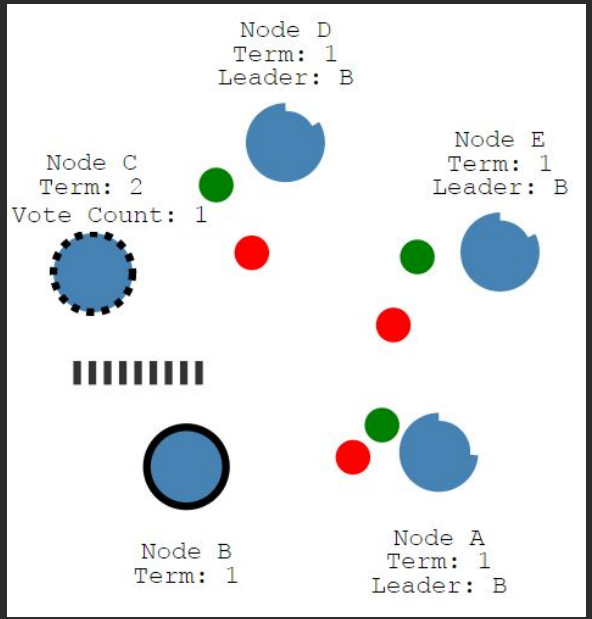
Что там есть ещё? Archive



Что там есть ещё? Cluster

Имплементация RAFT протокола поверх Aeron

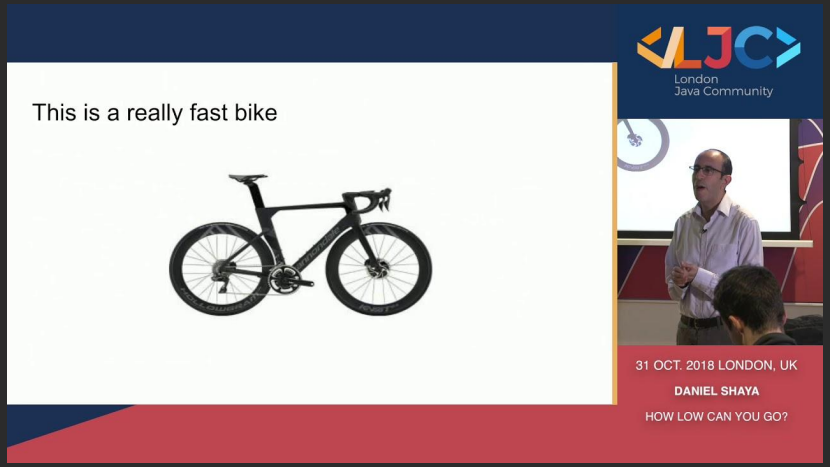
http://thesecretlivesofdata.com/raft/



Стоит посмотреть

How low can you go? By Daniel Shaya

https://www.youtube.com/watch?v=BD9cRbxWQx8



Стоит посмотреть

Сергей Мельников — Профилируем чёрного лебедя

https://www.youtube.com/watch?v=Cn4bw7cE68w

Стоит посмотреть

Роман Елизаров — Миллионы котировок в секунду

https://www.youtube.com/watch?v=Q-7y1u9kZV0

Стоит посмотреть

Интервью с Антоном Батяевым на Хабре

https://habr.com/ru/company/dbtc/blog/449630/

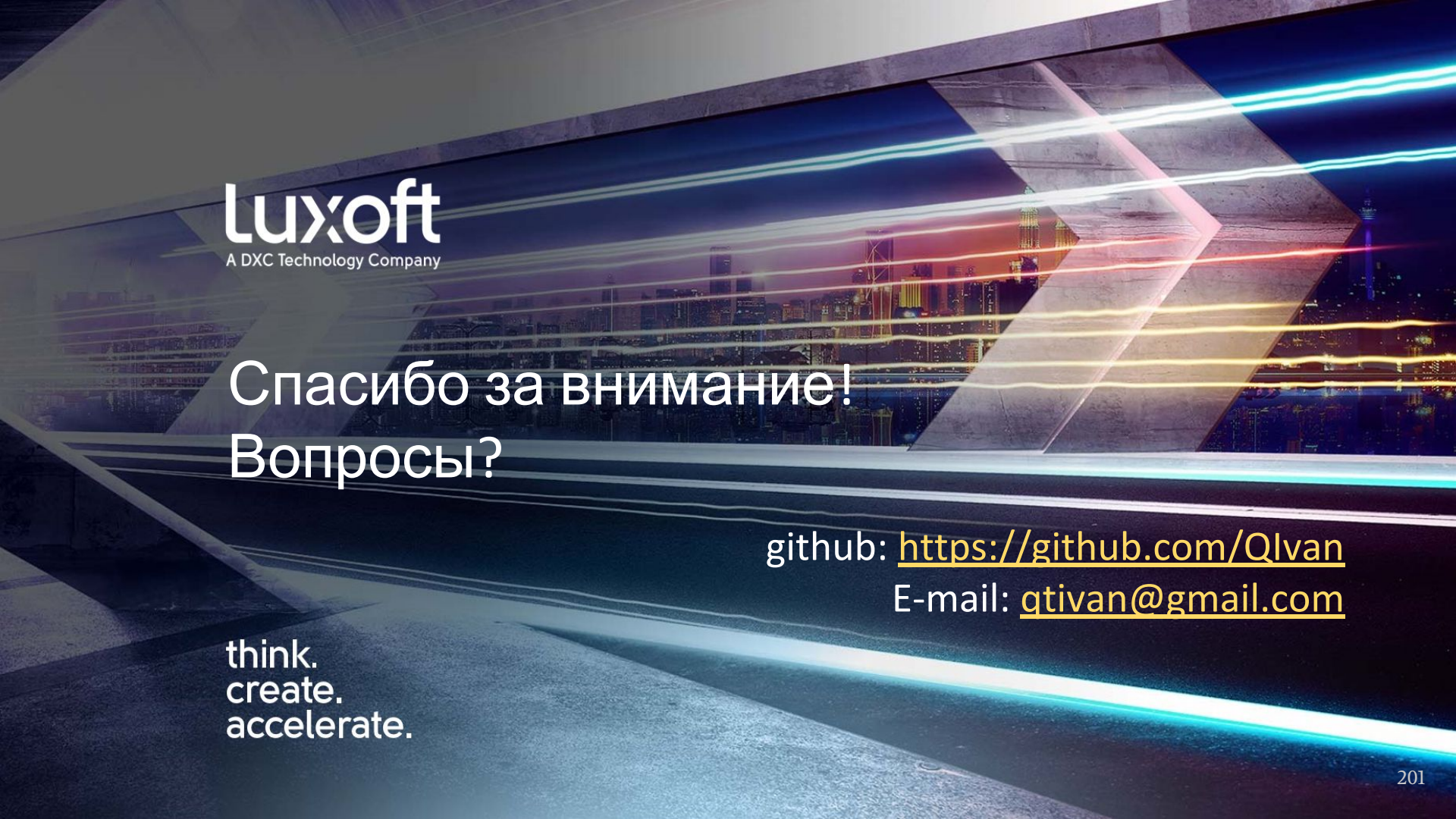
Итоги

- API
  - Publication
  - Subscription
- Media Driver
  - Sender, Receiver, Conductor
- Протокол
  - Reordering, NAK, Multicast
- LogBuffer


Aeron

Luxoft
A DXC Technology Company

Иван Землянский сделал это


luxoft
A DXC Technology Company
Спасибо за внимание!
Вопросы?
github: https://github.com/QIvan
E-mail: qtivan@gmail.com
think.
create.
accelerate.