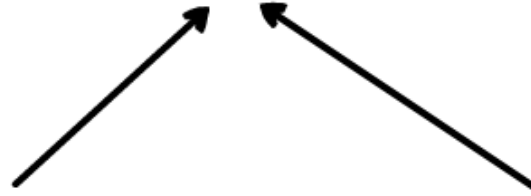


*IEnumerable*



*int[] numbers = new int[] { 1, 2, 3 }*

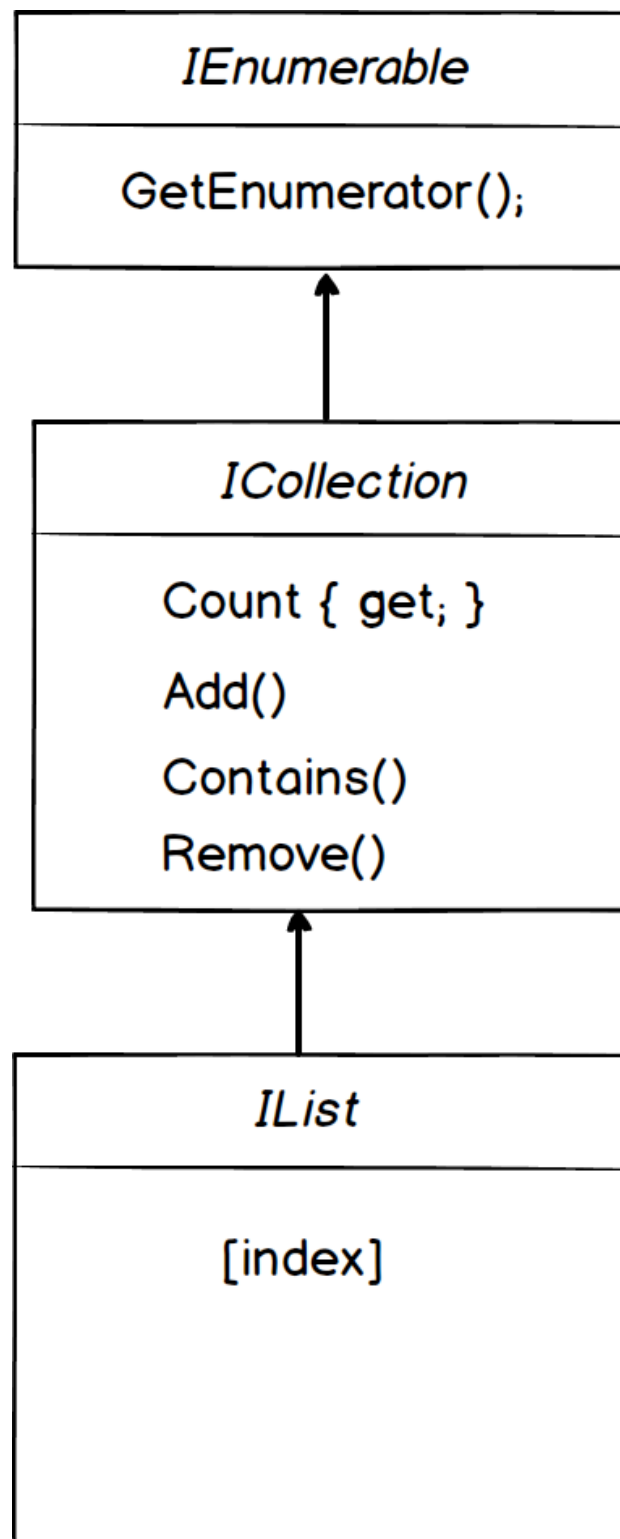
*List<int> numbers = new List<int>();*

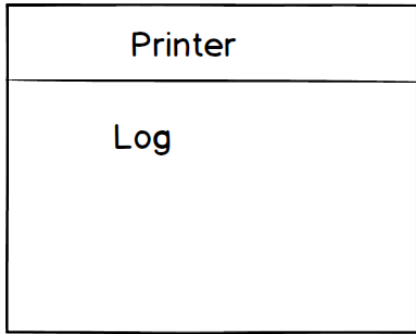
*numbers[1] = 9;*

*numbers[1] = 9;*

*numbers.Add(7);*

*numbers.Remove(7);*

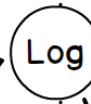




```
Log($"{DateTime.Now} Printing {content}
copy #{copy}");
```

```
decimal? cost =
CalculateCost?.Invoke(copies);
```

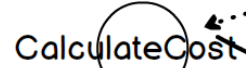
void LogDelegate(string)



Console.WriteLine(message)

File.WriteAllText("log.txt", message);

decimal CalculateCostDelegate(int)



decimal CalculateCost(int copies){ ... return ... }

$$f(x) = x + 2$$

$$g(x) = x^2 \text{ } x \text{ należy do } C$$

$$f(x, y) = x + y \text{ } x \text{ i } y \text{ należy do } C$$

```
int F(int x)
{
    return x + 2;
}
```

```
int F(int x, int y)
{
    return x + y;
}
```

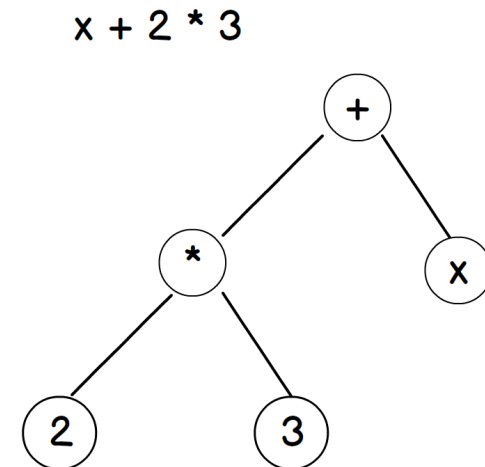
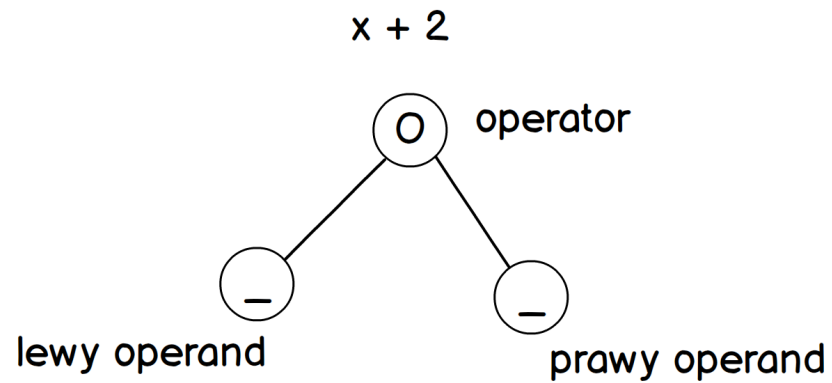
Wyrażenia lambda

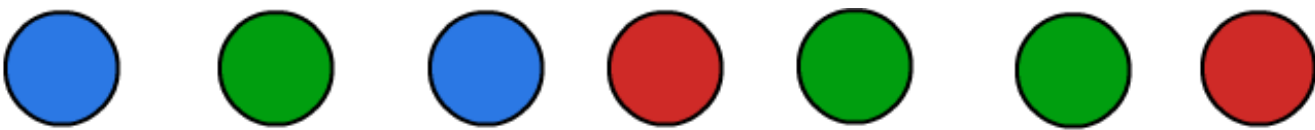
$$x = 1$$

$$x \rightarrow x + 2$$

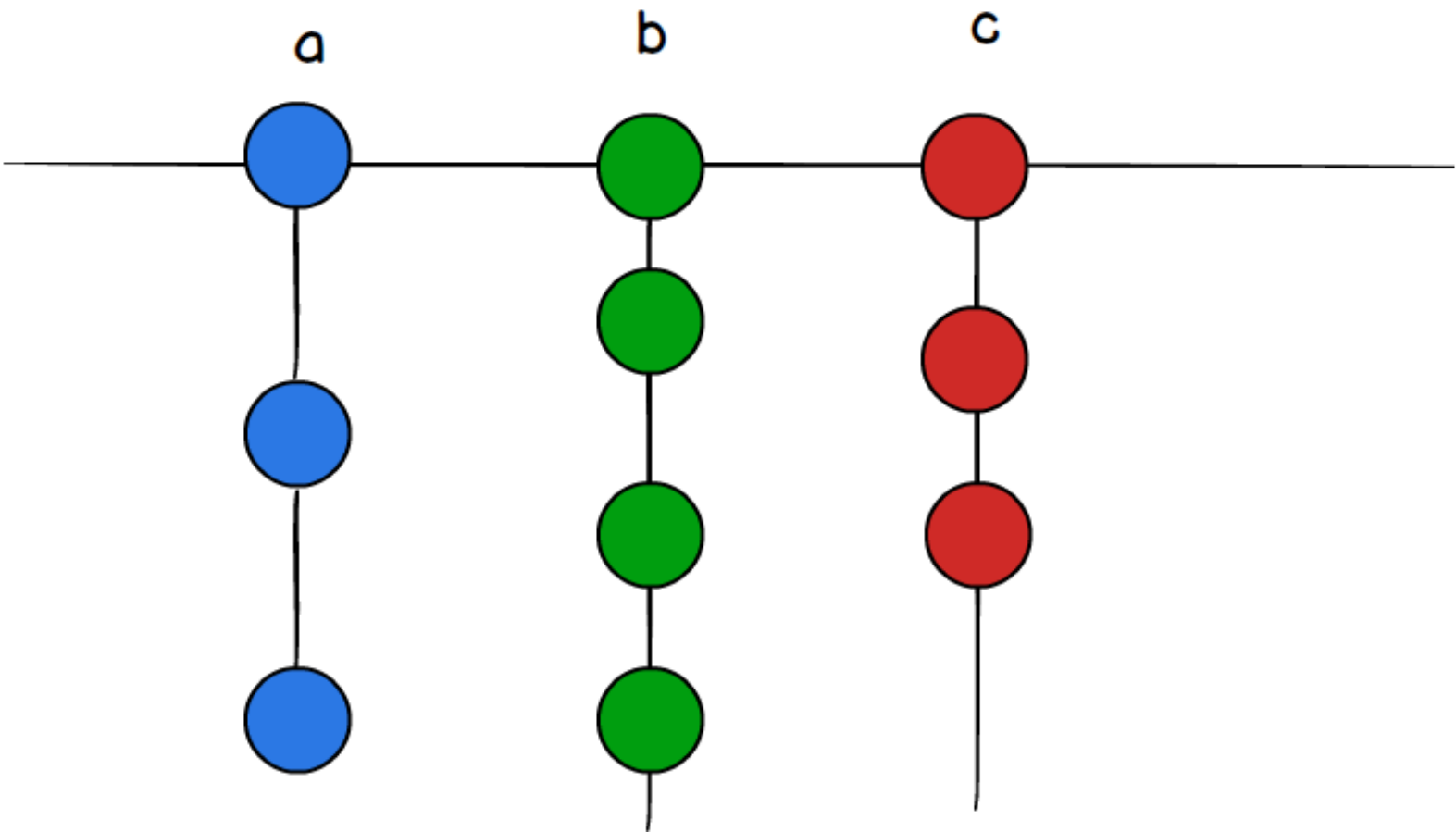
$$x, y \rightarrow x + y$$

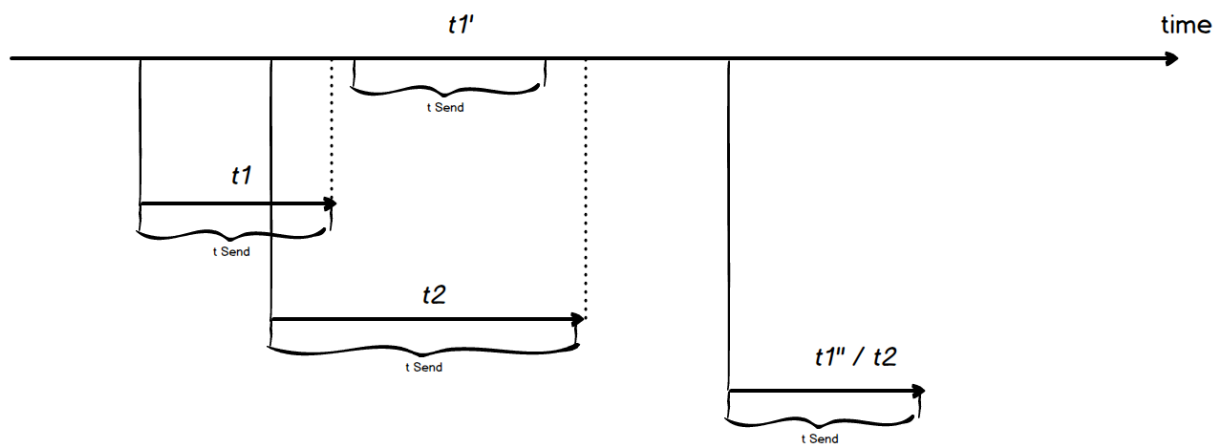
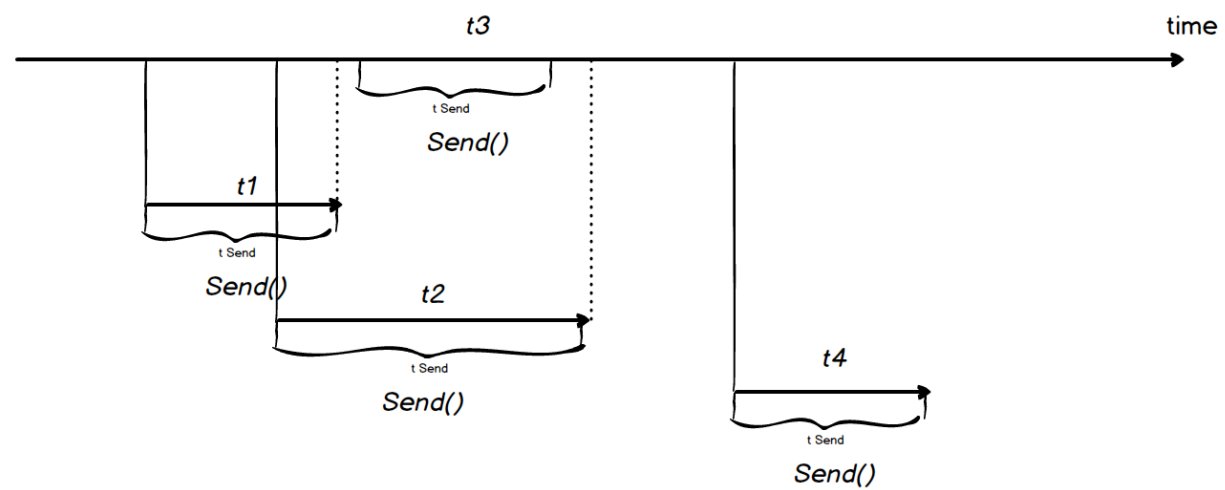
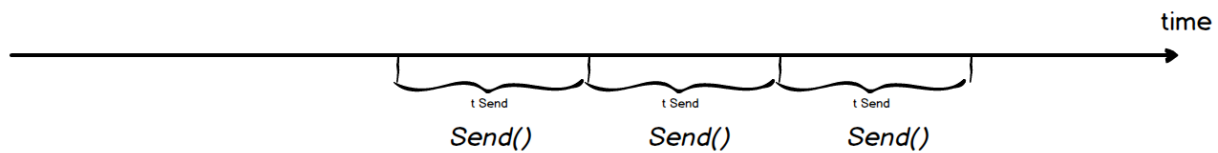
$$x \Rightarrow x + 2$$

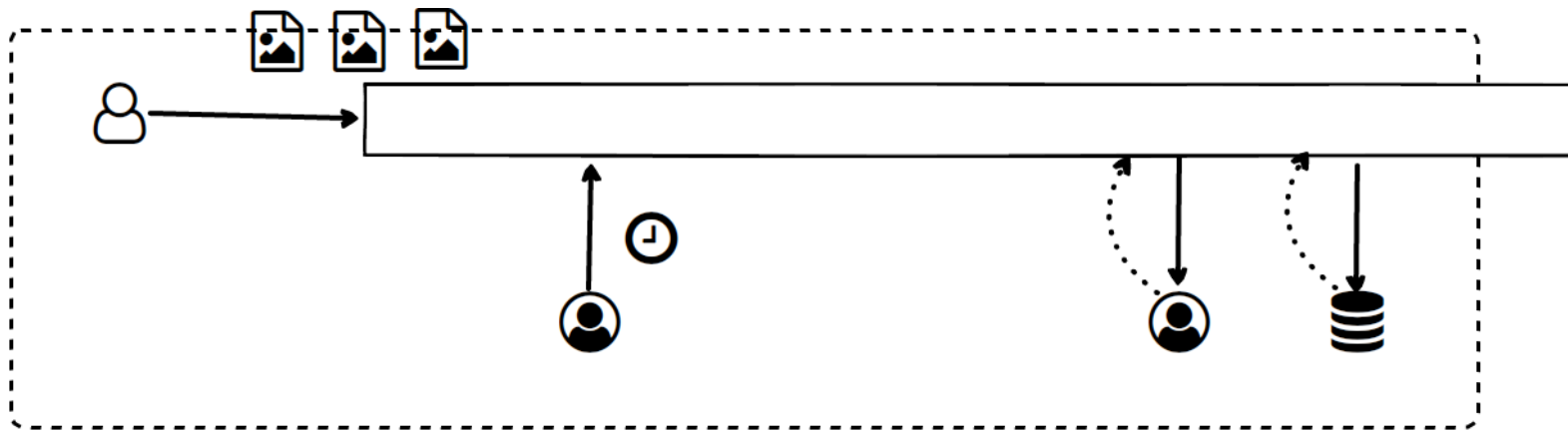




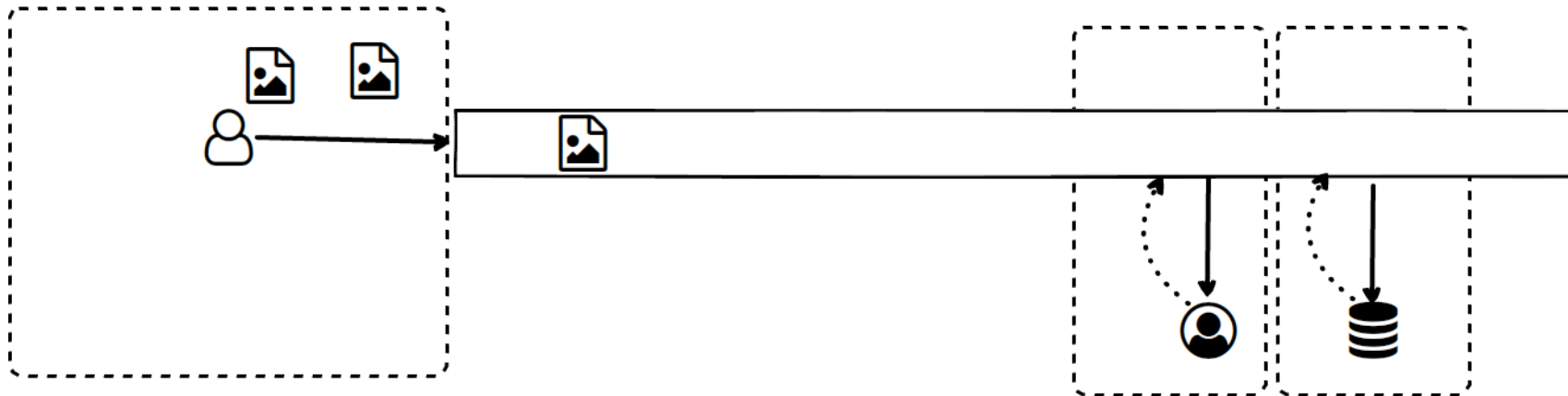
GroupBy







.....



RabbitMQ / Kafka

REDIS