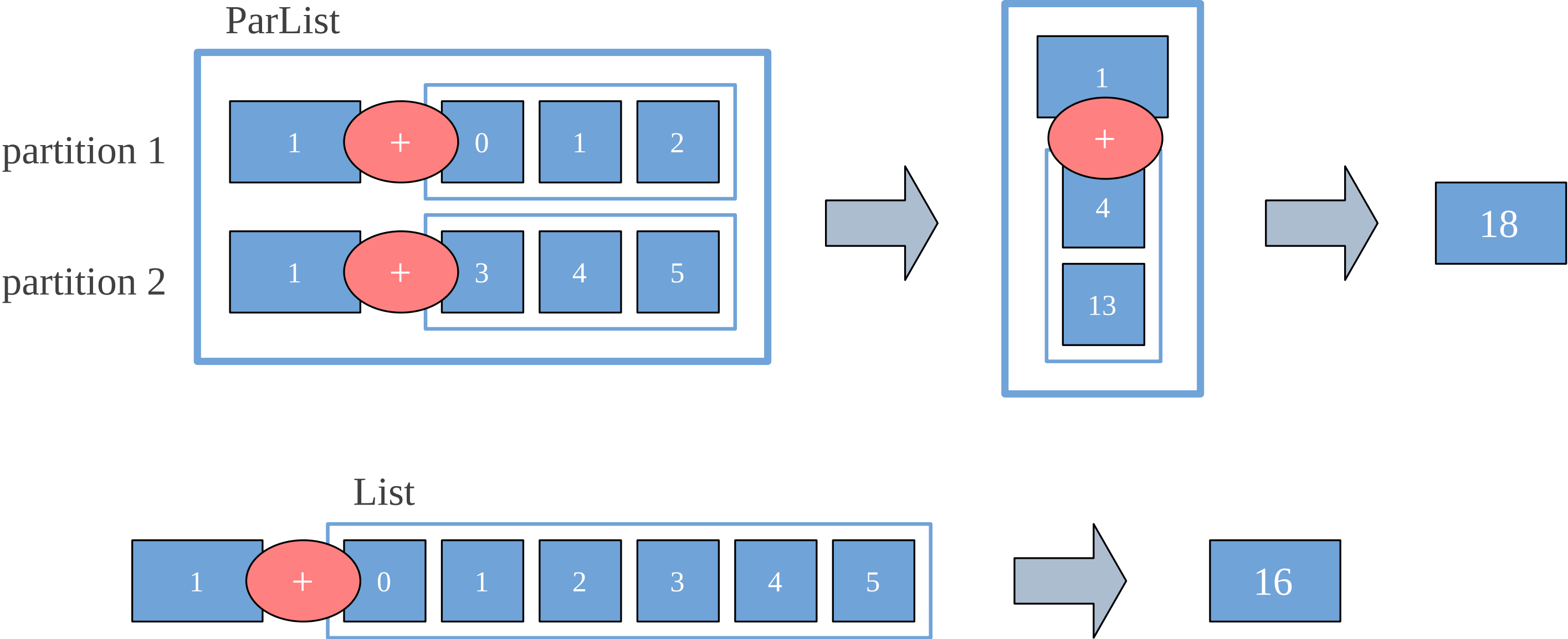


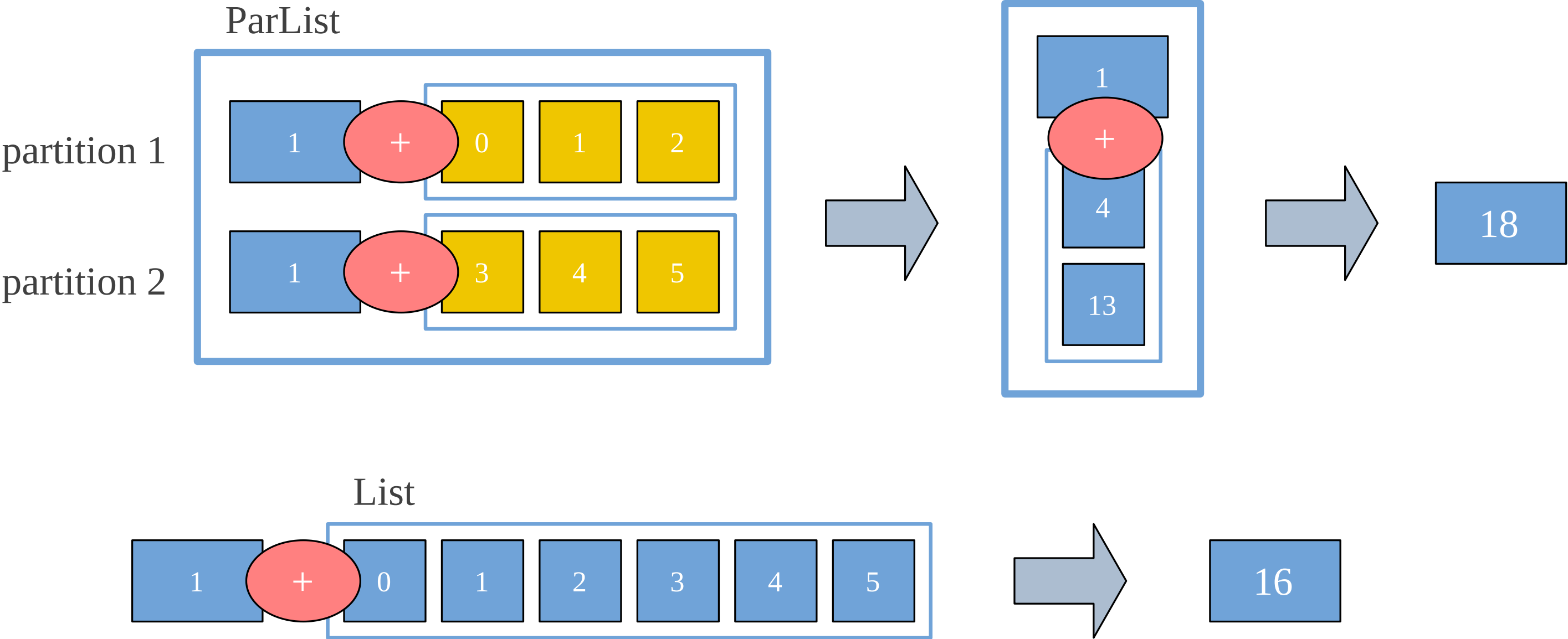
The background features a complex network of blue dots of varying sizes connected by thin, light blue lines. The dots are scattered across the frame, with some forming dense clusters and others standing alone. The lines create a web-like pattern that fills the entire background, giving it a technical or digital feel.

# **ANALYSIS OF GLOBAL TEMPERATURE PART 2**

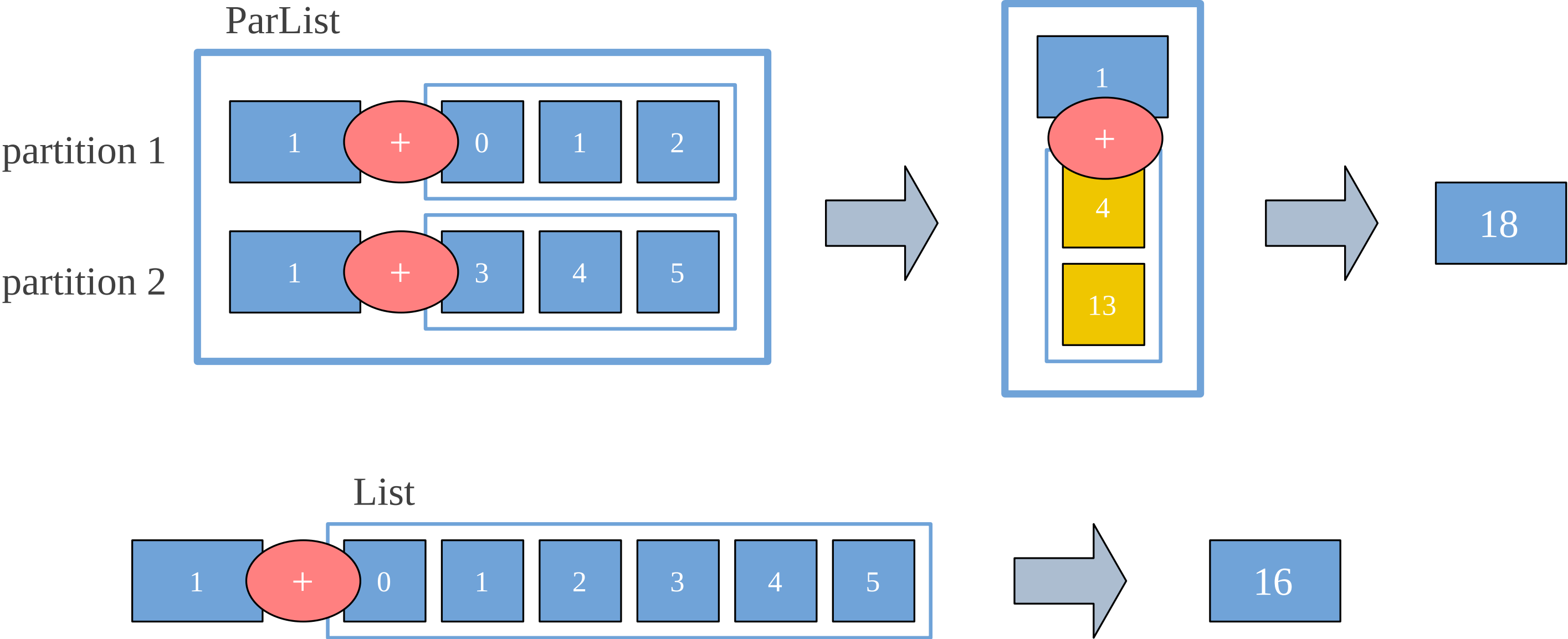
# monoFoldLeft vs List foldLeft



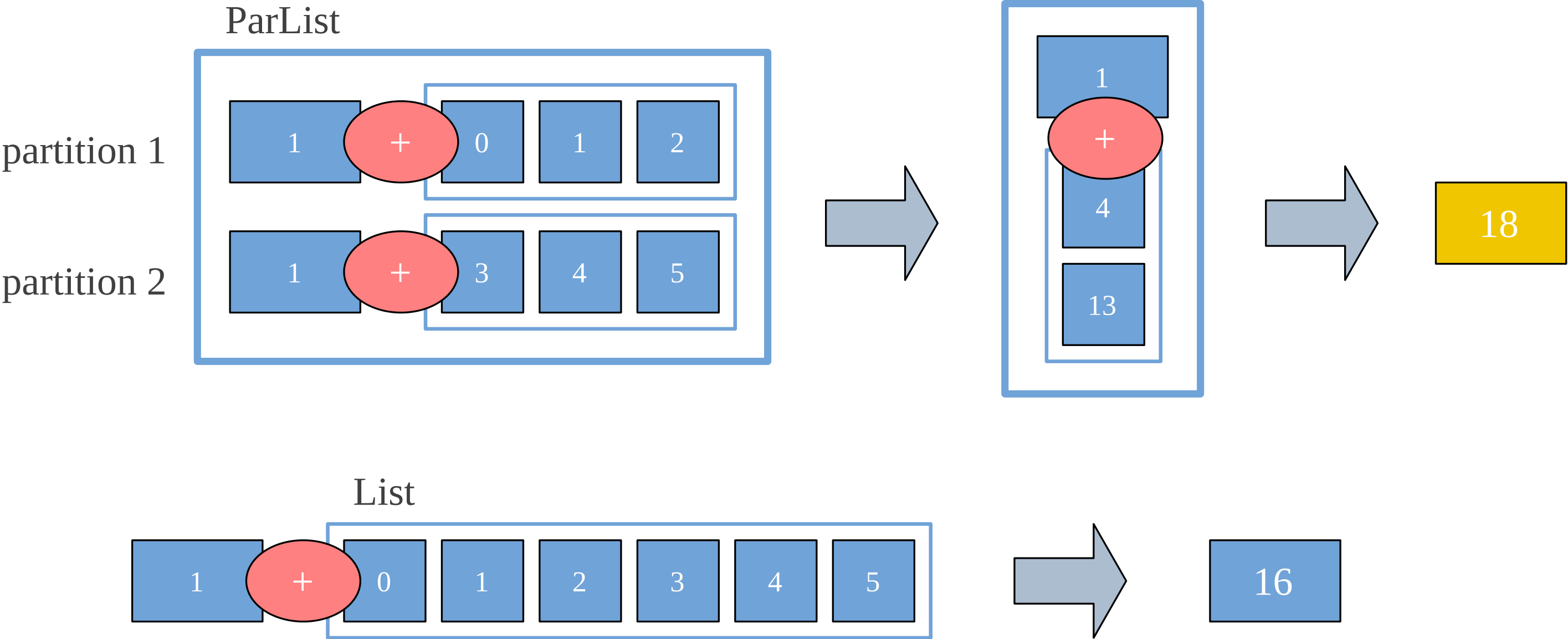
# monoFoldLeft vs List foldLeft



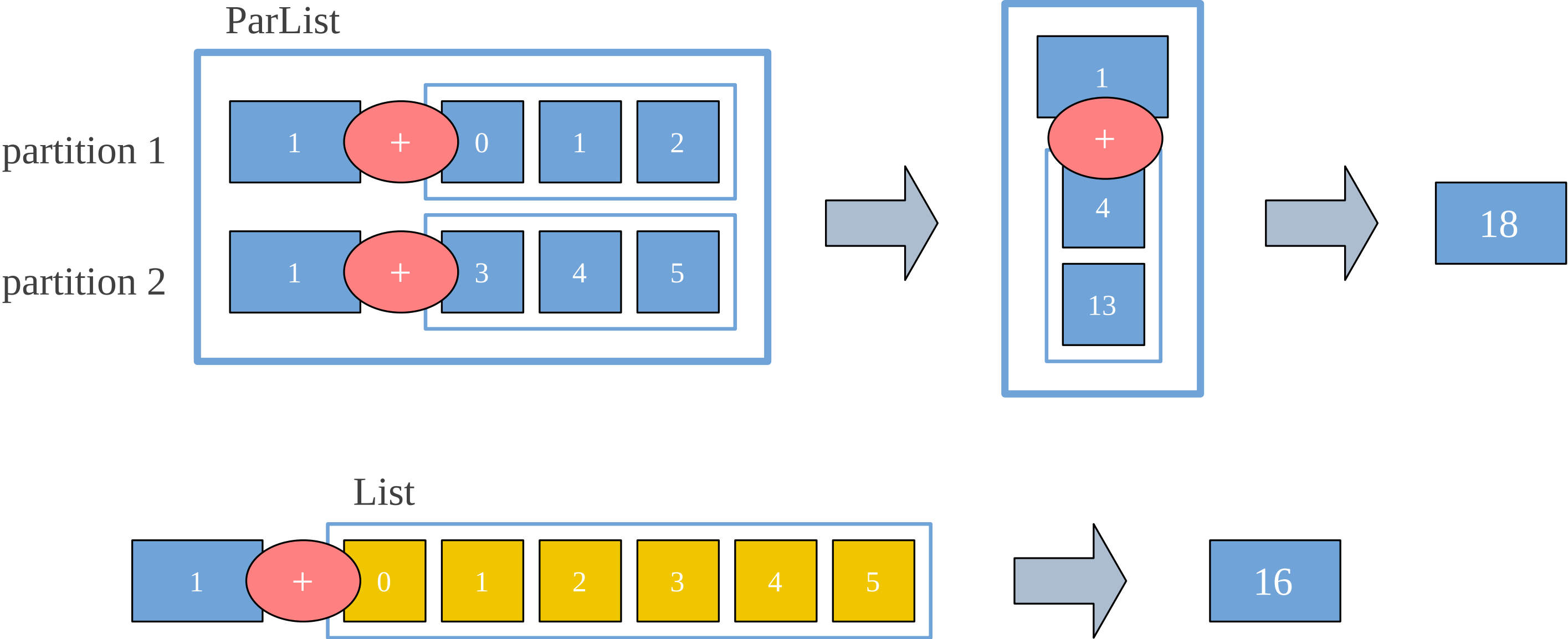
# monoFoldLeft vs List foldLeft



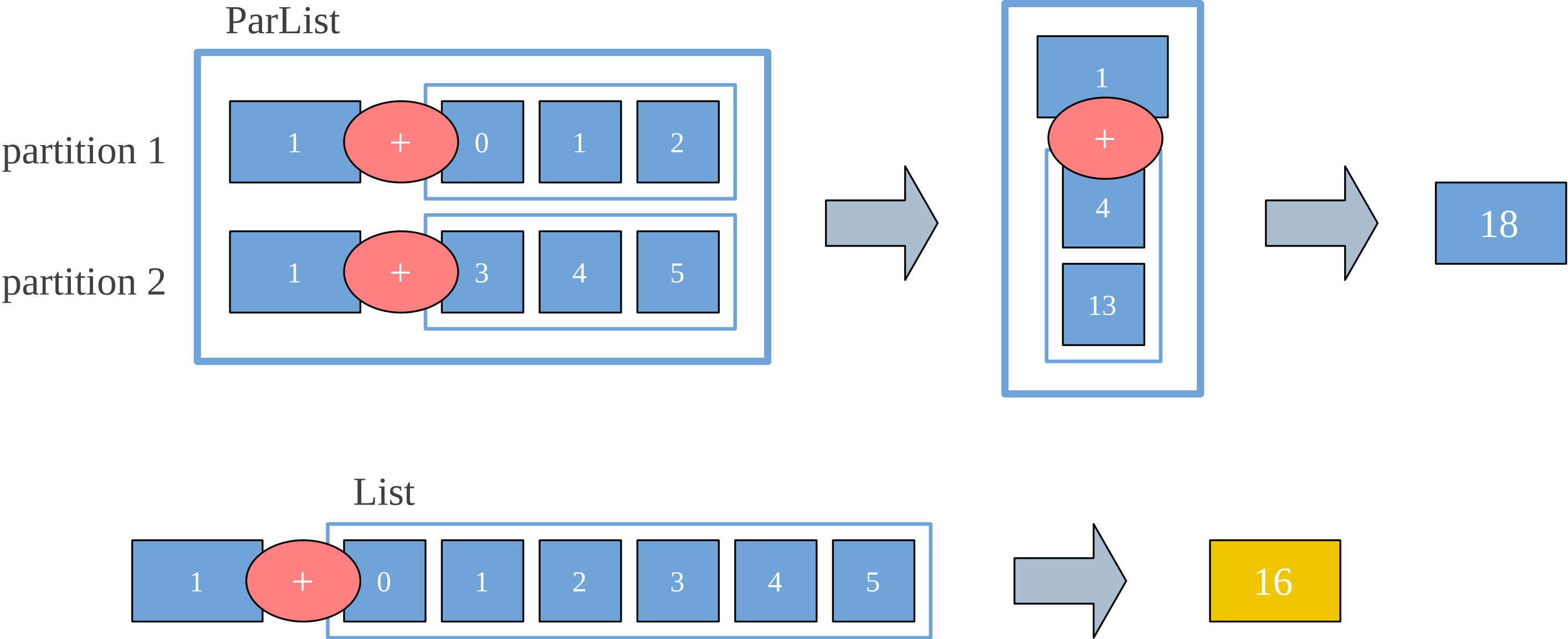
# monoFoldLeft vs List foldLeft



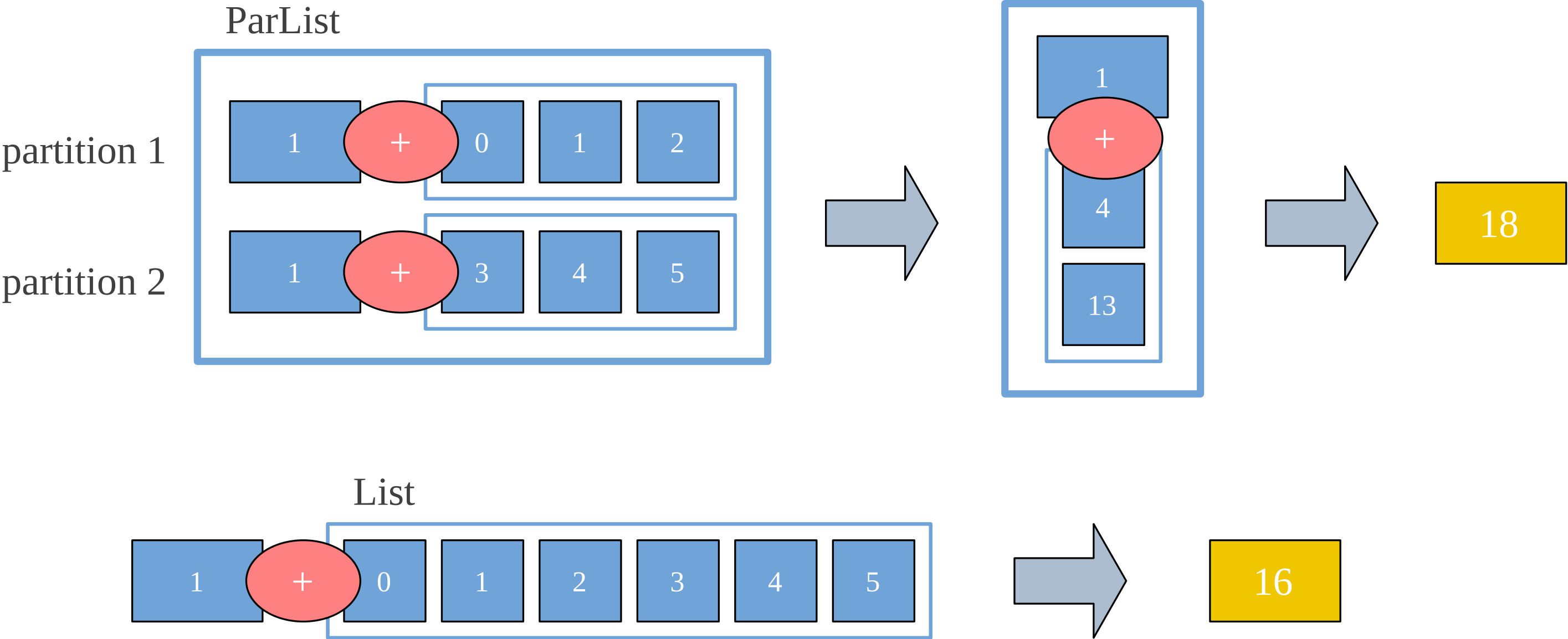
# monoFoldLeft vs List foldLeft



# monoFoldLeft vs List foldLeft

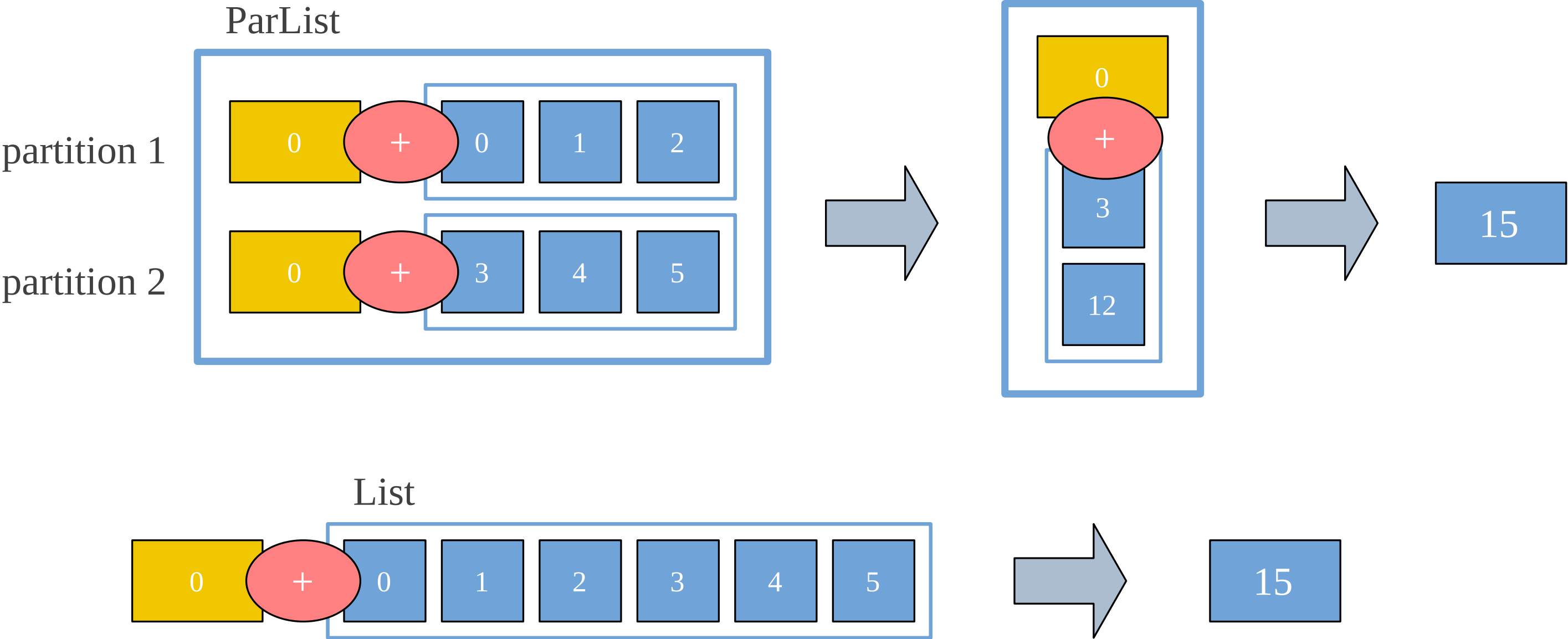


# monoFoldLeft vs List foldLeft

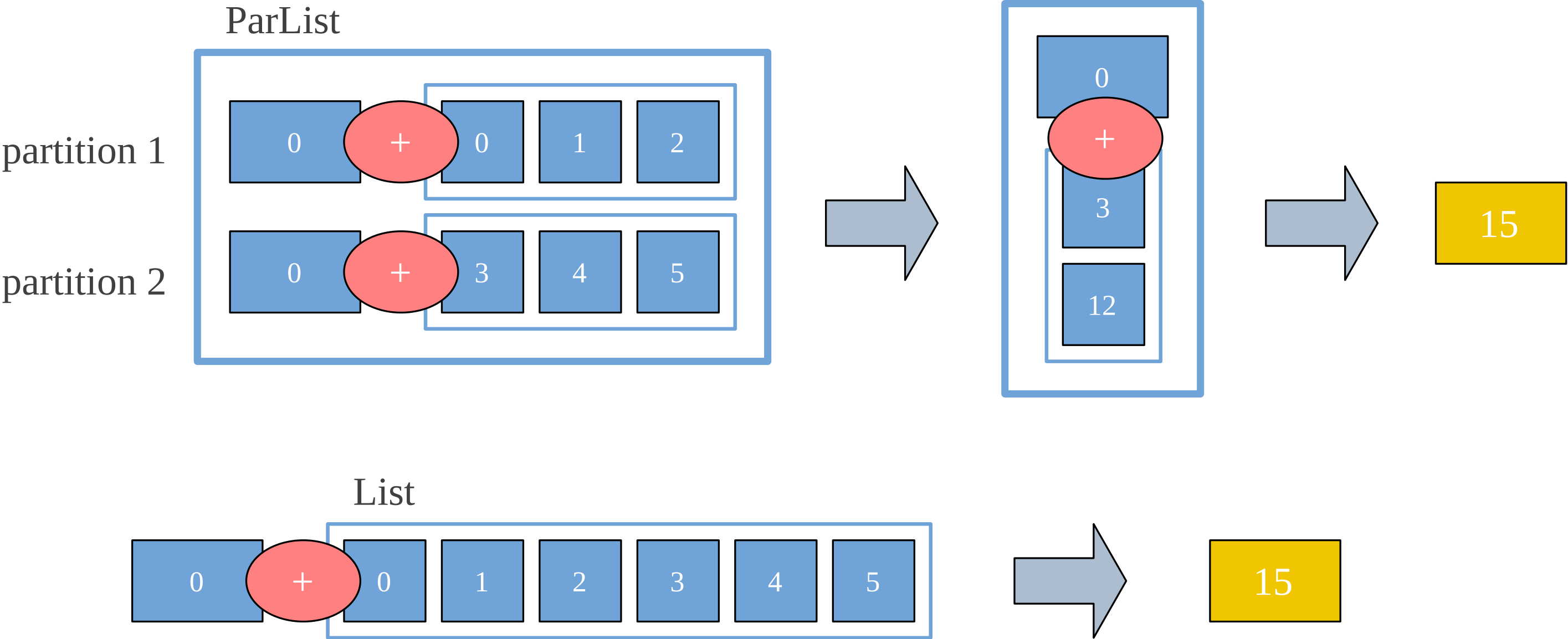




# monoFoldLeft vs List foldLeft



# monoFoldLeft vs List foldLeft



$$0 + X == X$$

$$X + 0 == X$$

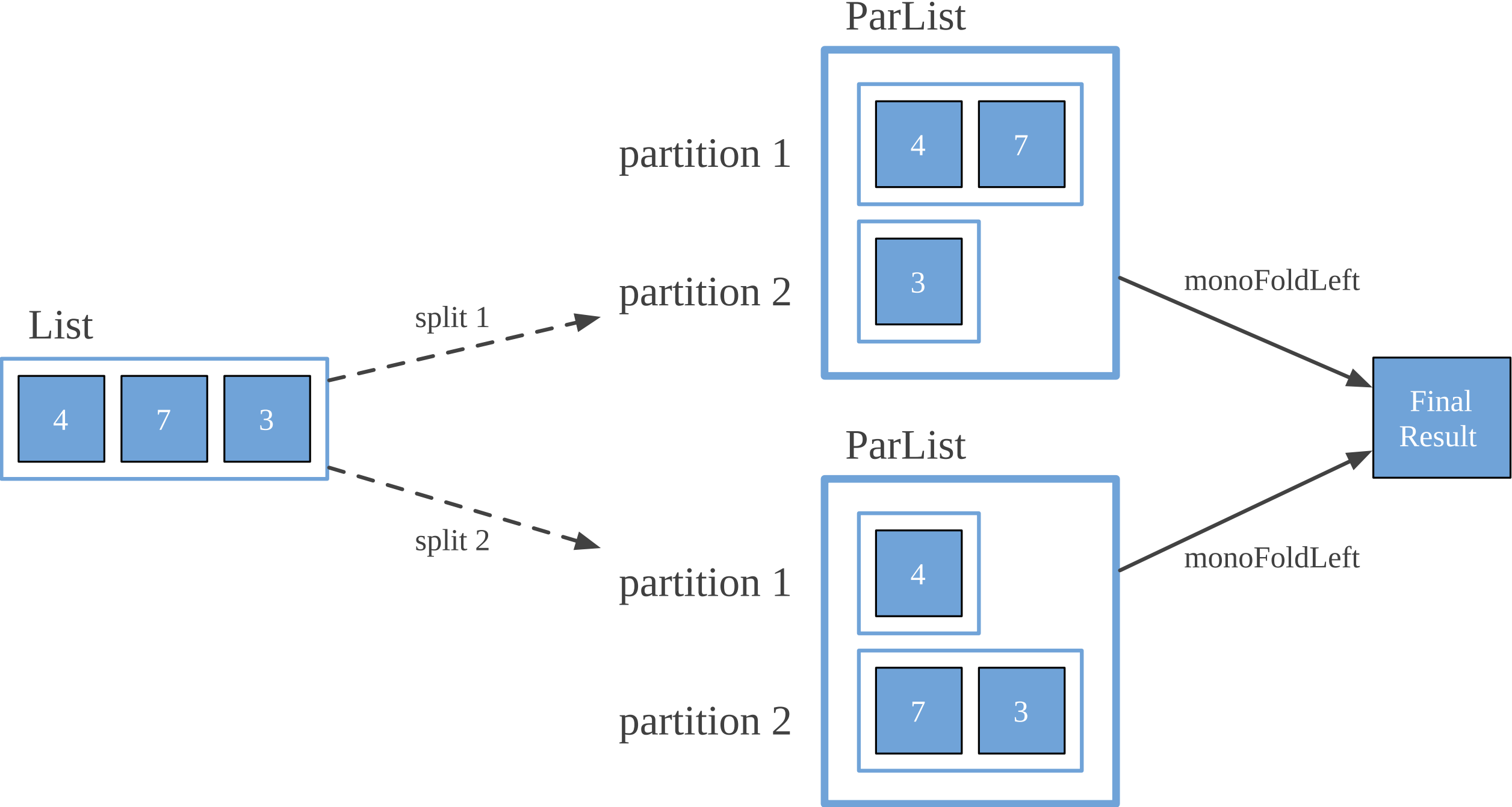
$0 + x == x$

$x + 0 == x$

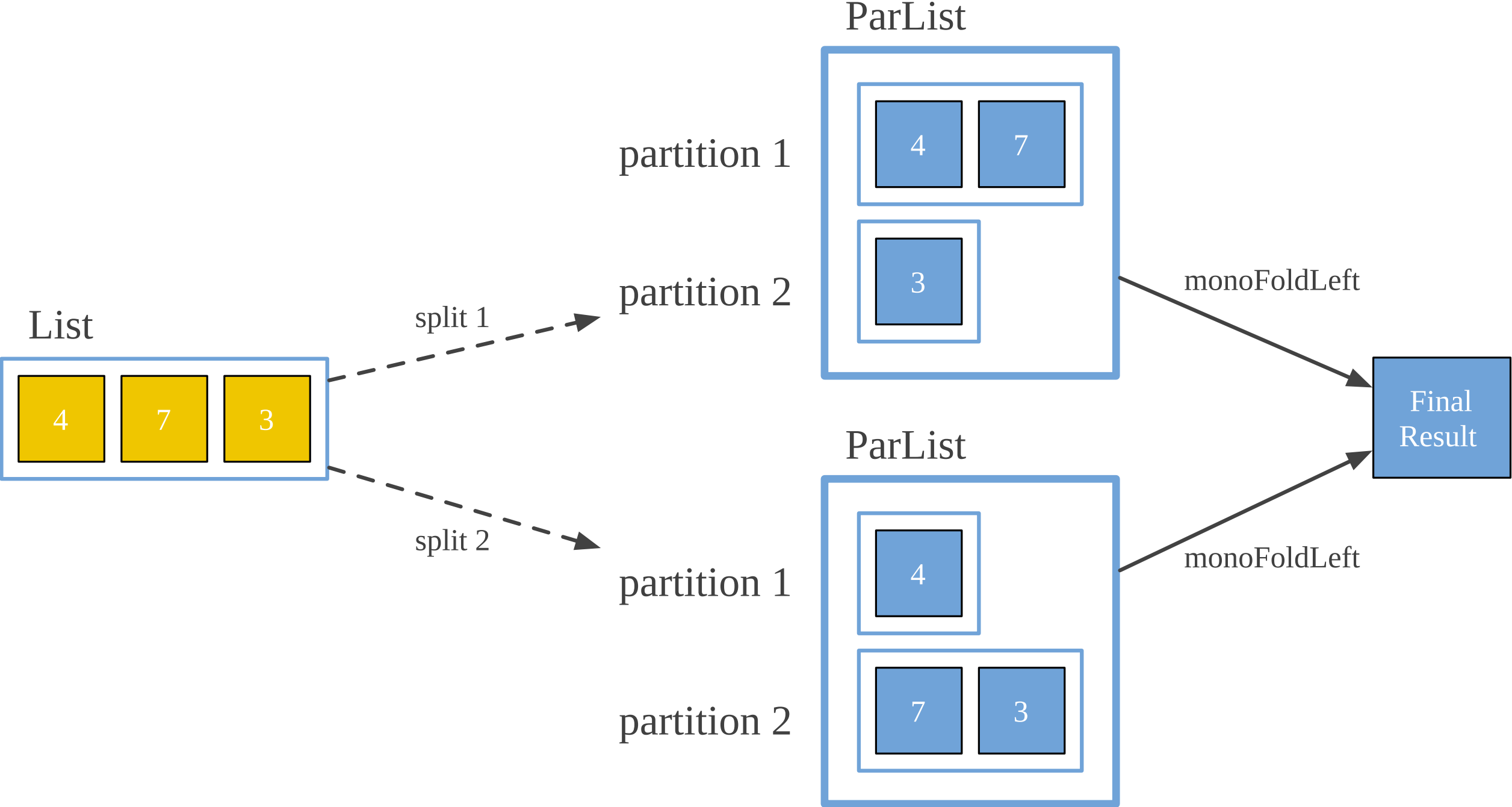
`combine(default, x) == x`

`combine(x, default) == x`

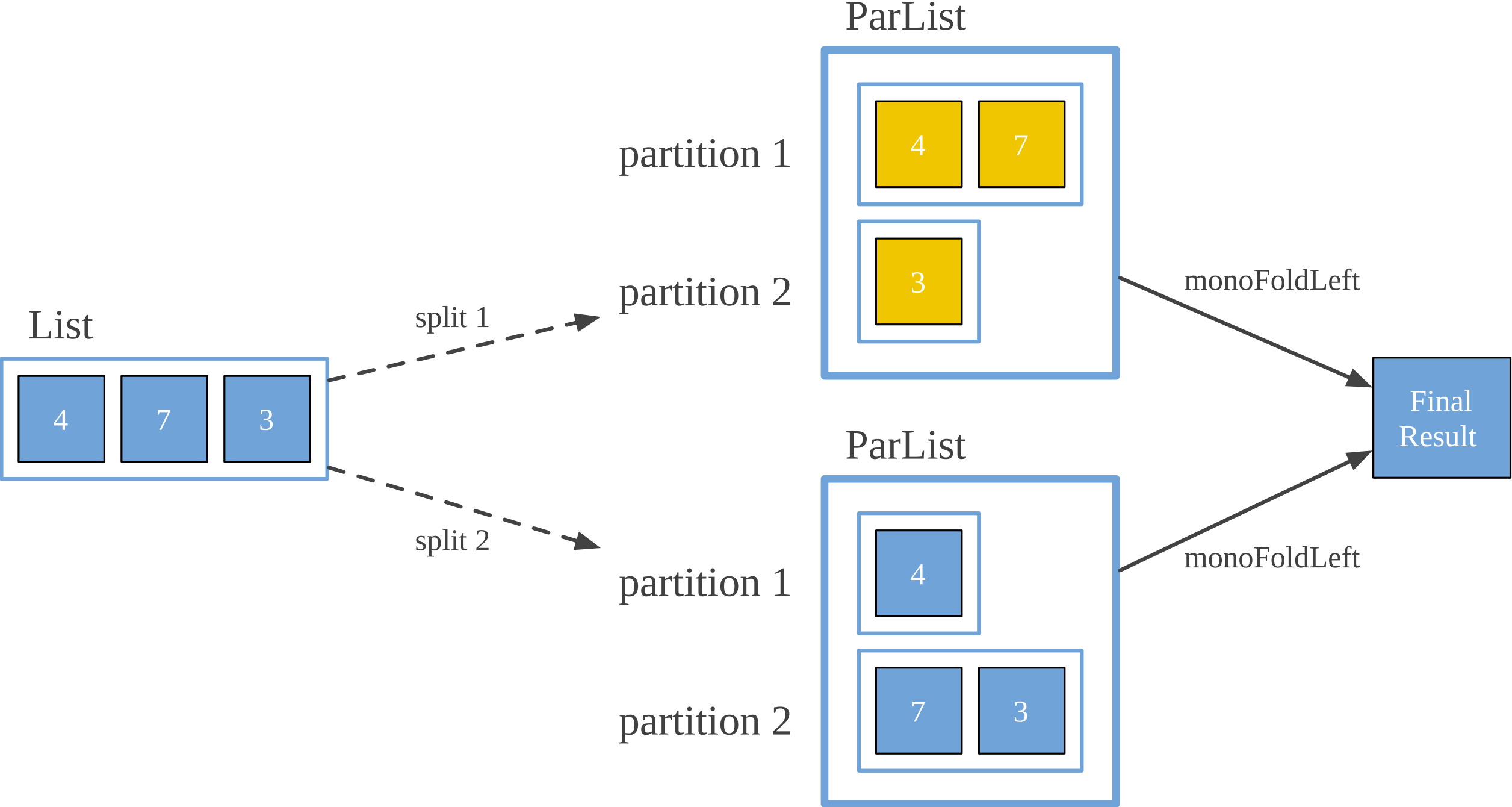
# monoFoldLeft is stable



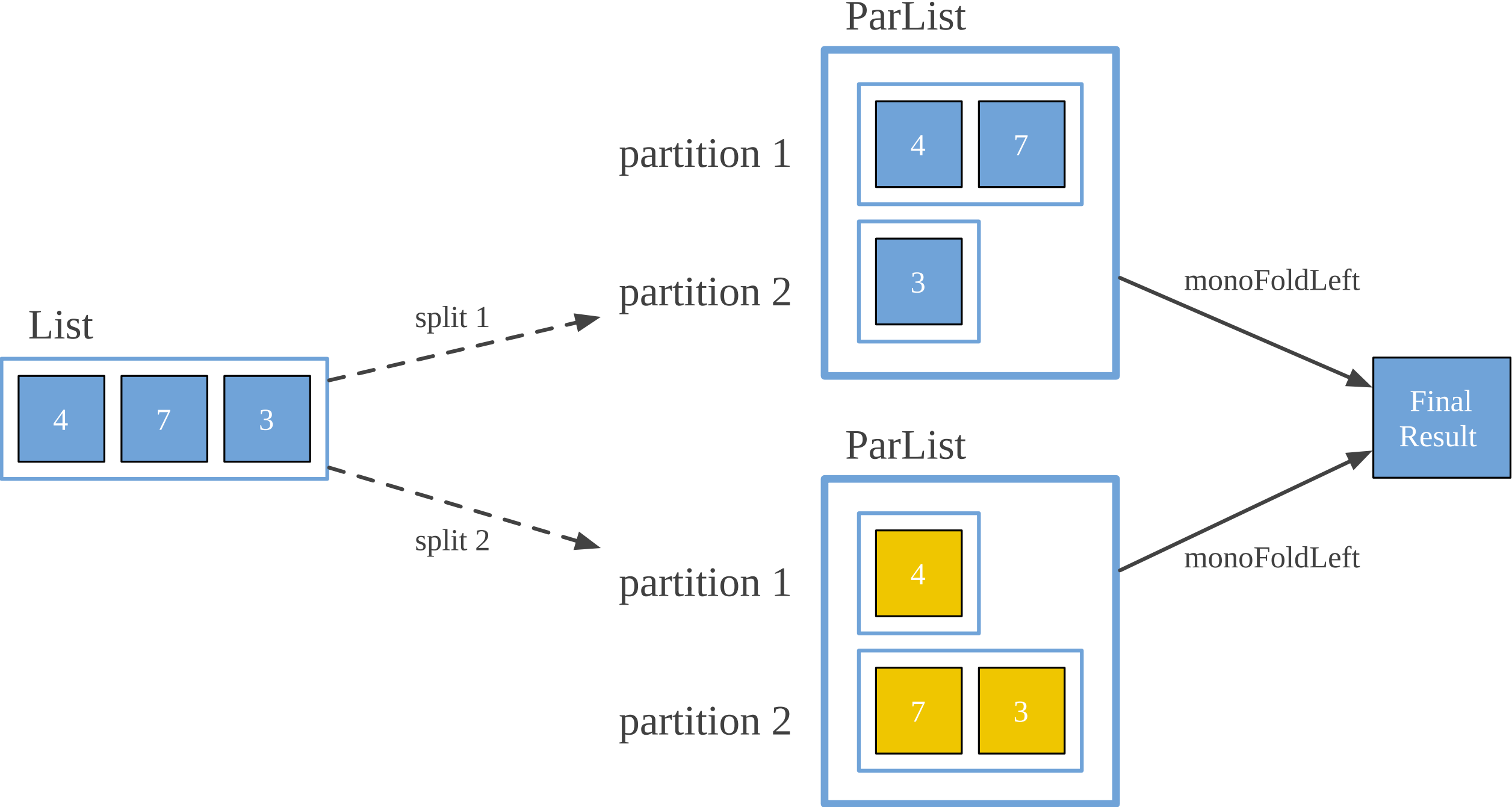
# monoFoldLeft is stable



# monoFoldLeft is stable

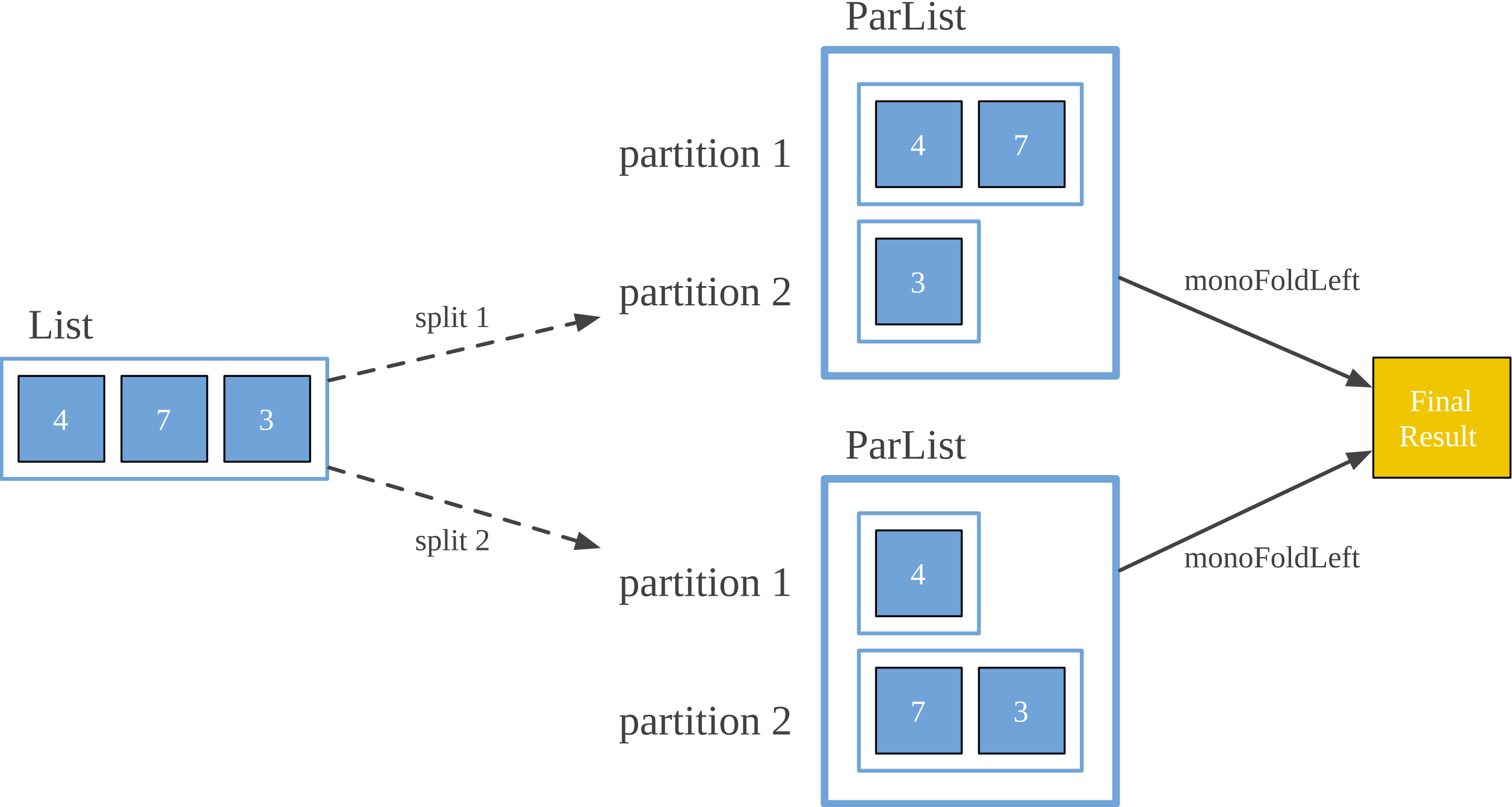


# monoFoldLeft is stable

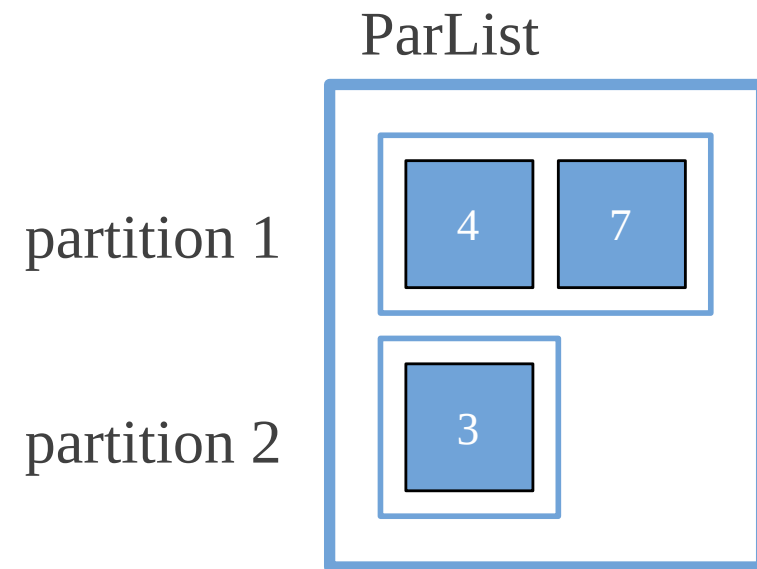




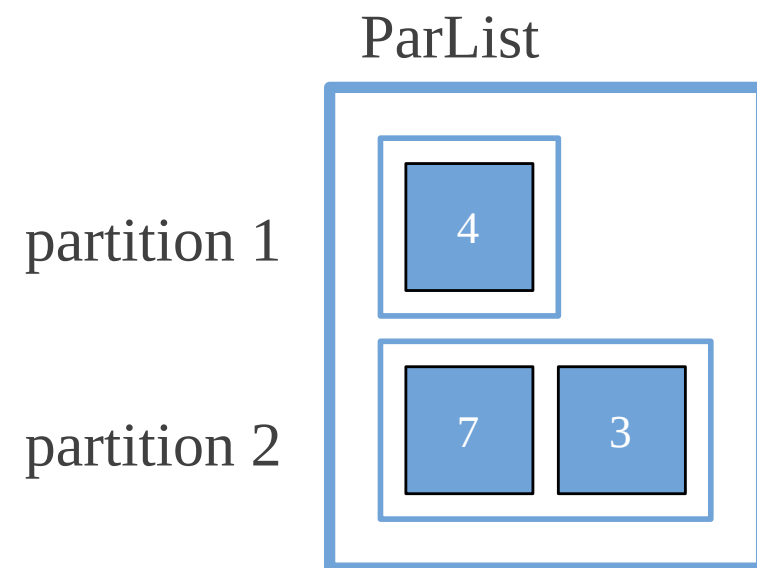
# monoFoldLeft is stable



# monoFoldLeft is stable

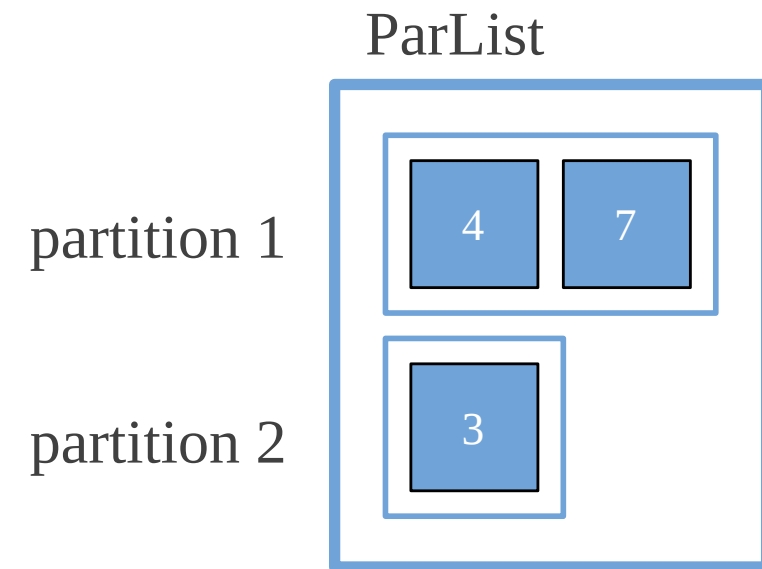


```
val partition1 = combine(combine(default, 4), 7)
val partition2 = combine(default, 3)
val finalResult = combine(combine(default, partition1), partition2)
```

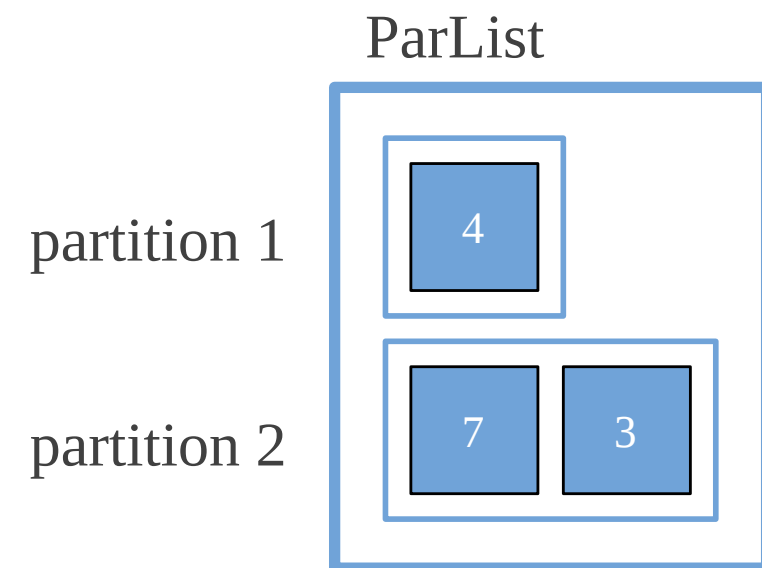


```
val partition1 = combine(default, 4)
val partition2 = combine(combine(default, 7), 3)
val finalResult = combine(combine(default, partition1), partition2)
```

# monoFoldLeft is stable

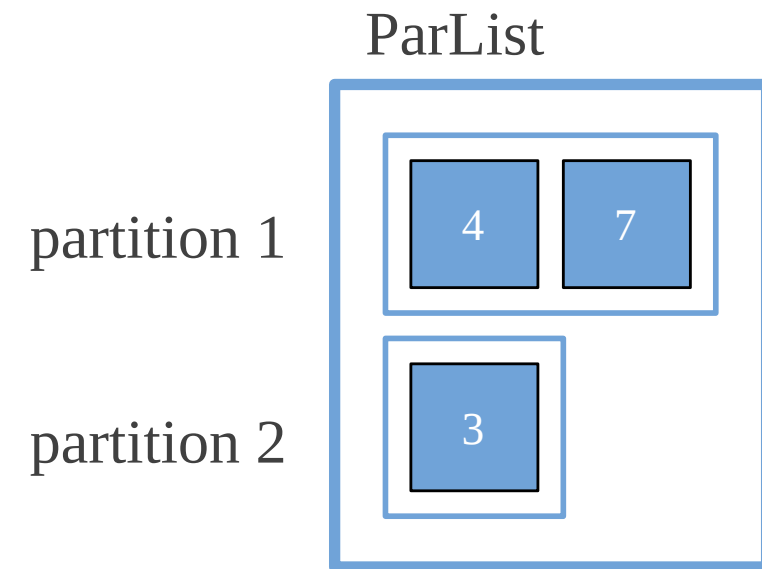


```
val partition1 = combine(combine(default, 4), 7)
val partition2 = combine(default, 3)
val finalResult = combine(combine(default, partition1), partition2)
```

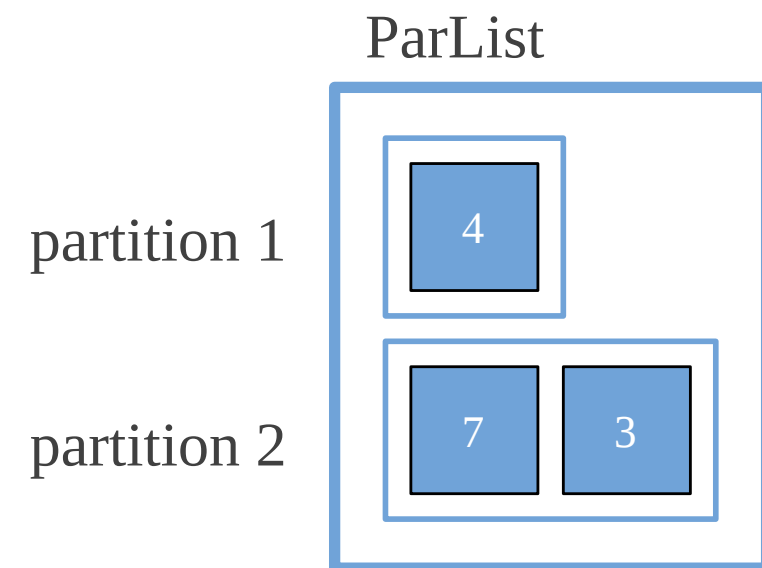


```
val partition1 = combine(default, 4)
val partition2 = combine(combine(default, 7), 3)
val finalResult = combine(combine(default, partition1), partition2)
```

# monoFoldLeft is stable

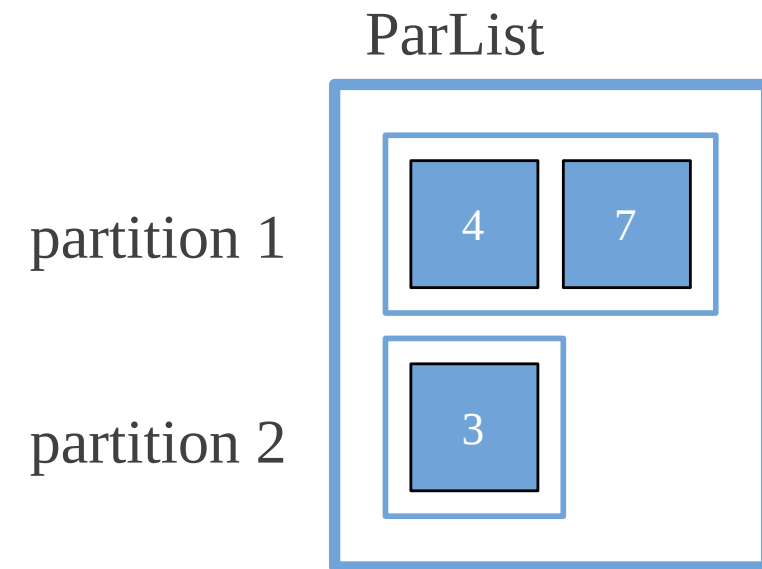


```
val partition1 = combine(combine(default, 4), 7)
val partition2 = combine(default, 3)
val finalResult = combine(combine(default, partition1), partition2)
```

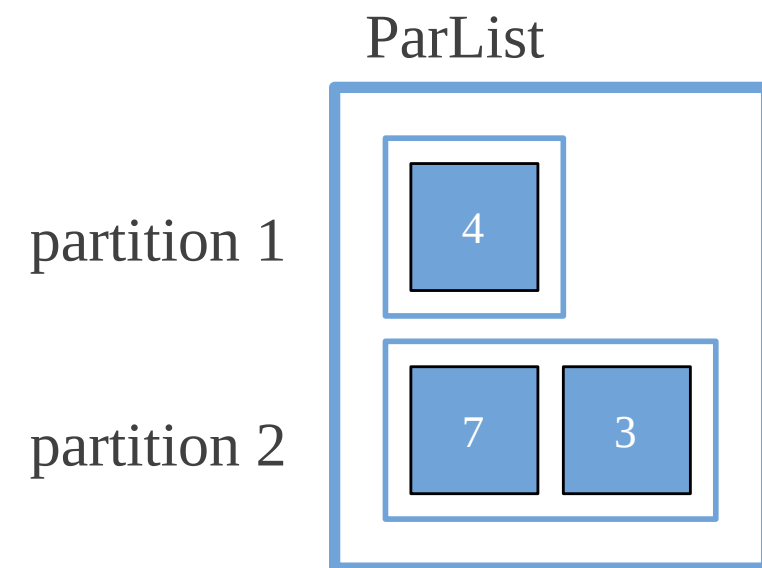


```
val partition1 = combine(default, 4)
val partition2 = combine(combine(default, 7), 3)
val finalResult = combine(combine(default, partition1), partition2)
```

# monoFoldLeft is stable

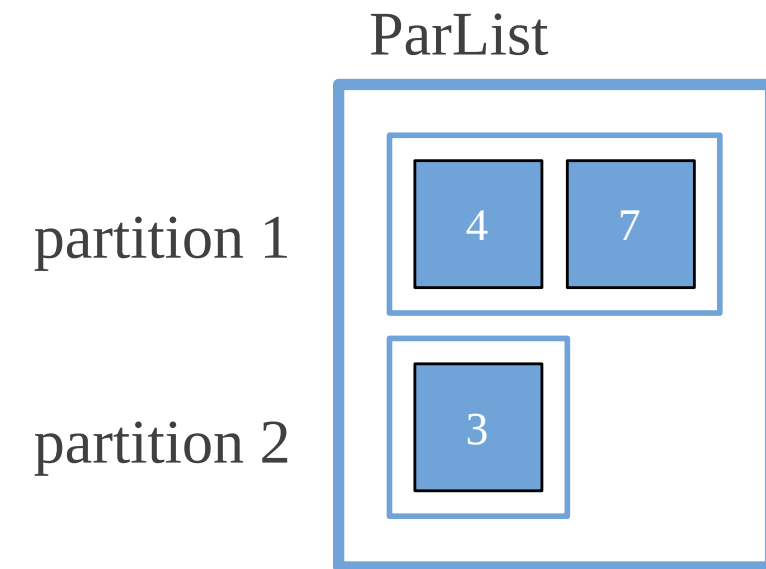


```
val partition1 = combine(combine(default, 4), 7)
val partition2 = combine(default, 3)
val finalResult = combine(combine(default, partition1), partition2)
```

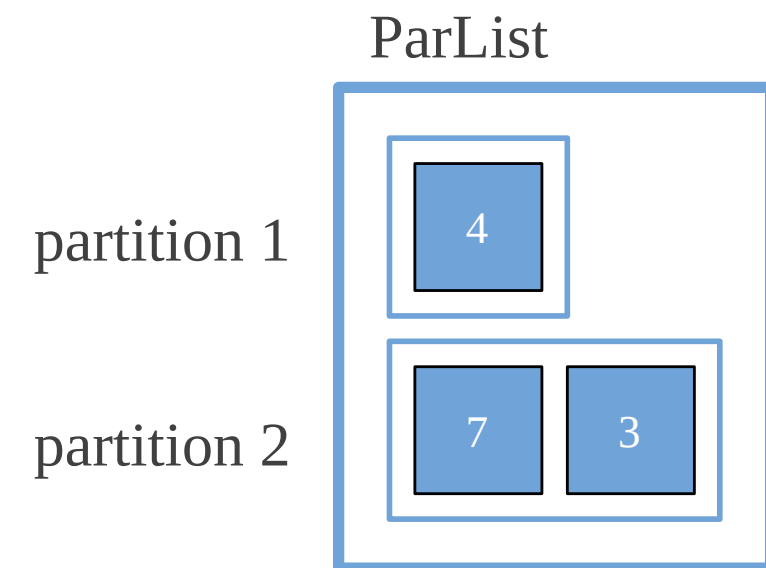


```
val partition1 = combine(default, 4)
val partition2 = combine(combine(default, 7), 3)
val finalResult = combine(combine(default, partition1), partition2)
```

# monoFoldLeft is stable

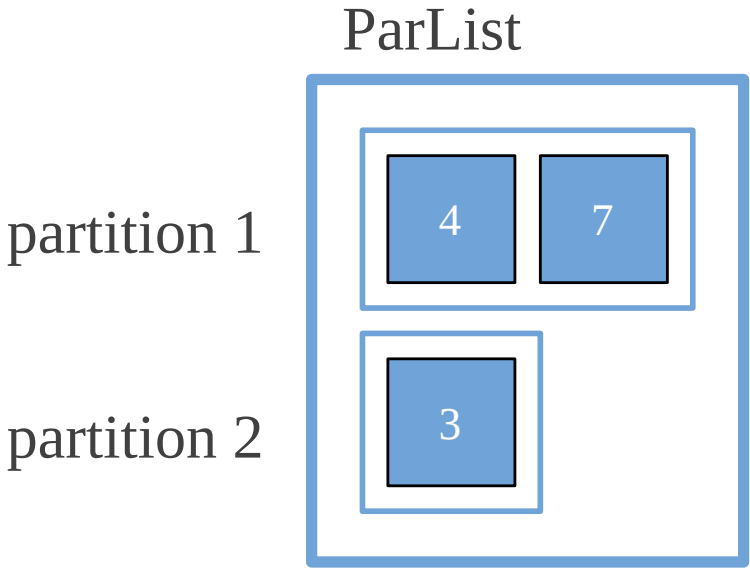


```
val partition1 = combine(4, 7)
val partition2 = 3
val finalResult = combine(partition1, partition2)
```

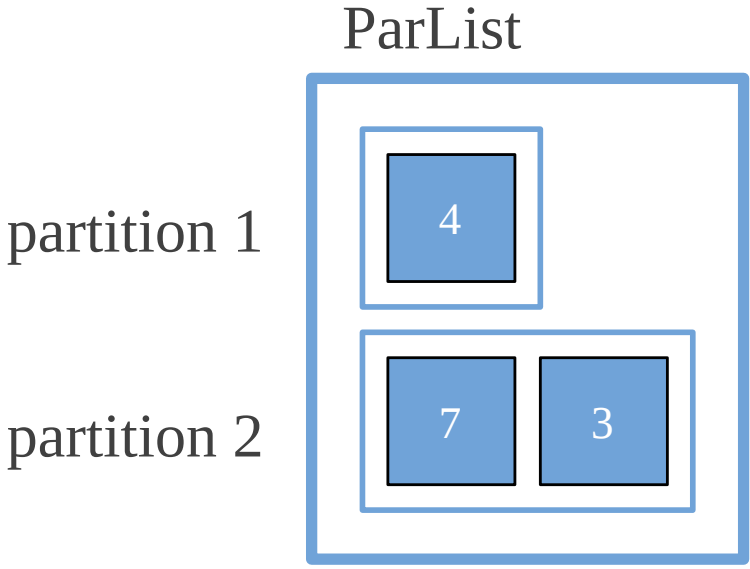


```
val partition1 = 4
val partition2 = combine(7, 3)
val finalResult = combine(partition1, partition2)
```

# monoFoldLeft is stable



```
combine(combine(4, 7), 3)
```



```
combine(4, combine(7, 3))
```

# Associative functions

```
(1 + (2 + 3)) == ((1 + 2) + 3)  
// res0: Boolean = true
```

```
(1 min (2 min 3)) == ((1 min 2) min 3)  
// res1: Boolean = true
```