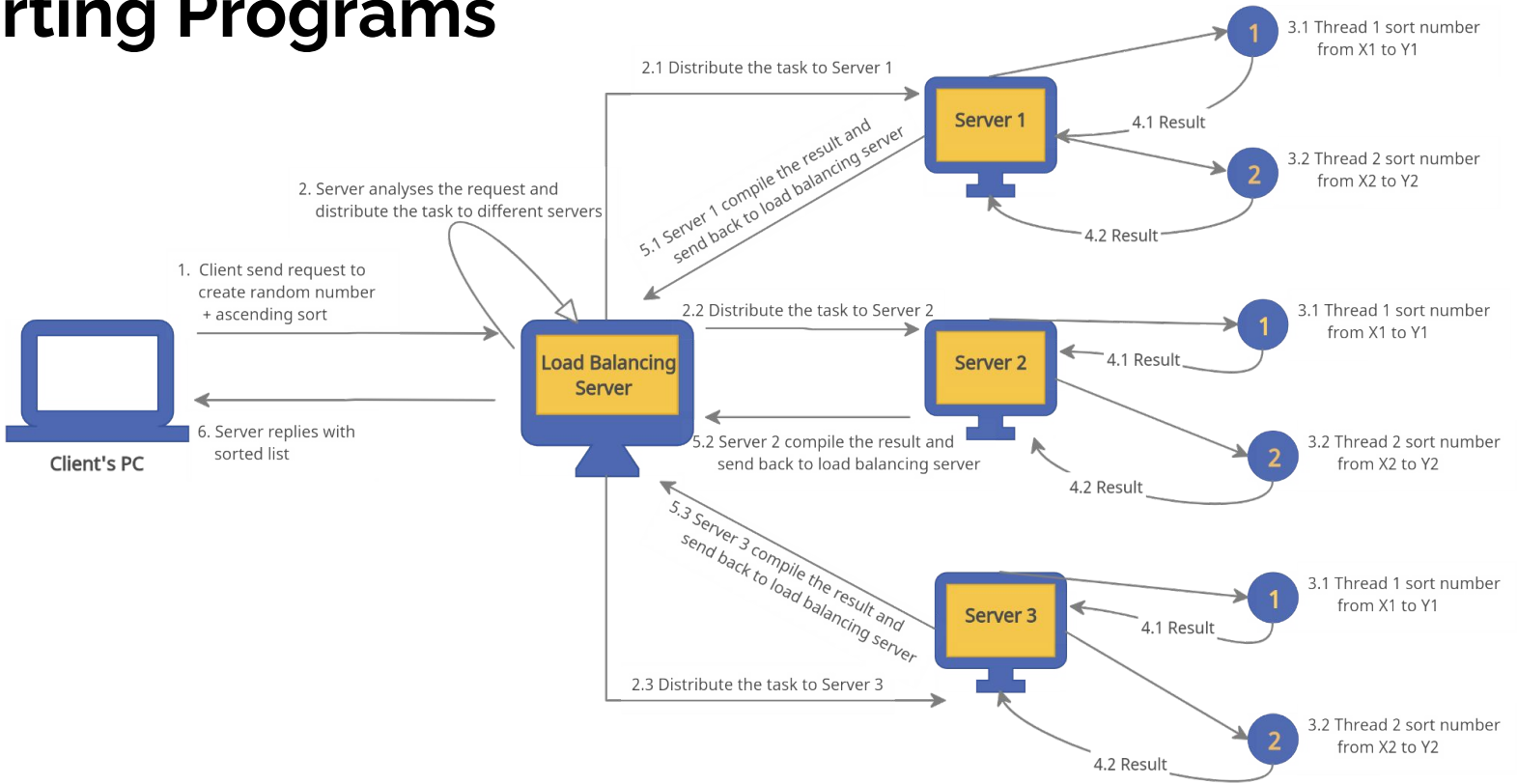
Sorting random numbers in distributed systems

Presenters:
Ibrahim Yunuslu
Kanan Jafarli

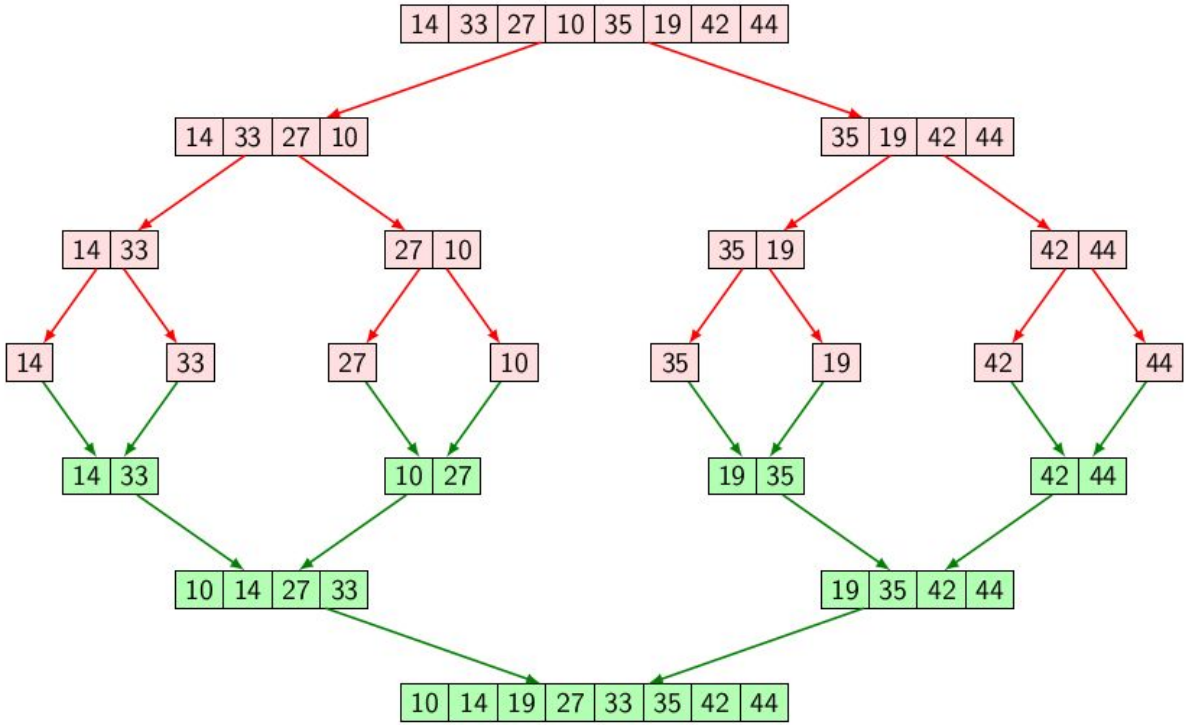
Application of Distributed Systems In Speeding Up Sorting Programs



Merge Sort

```
MergeSort(arr[], l,  r)
If r > l
     1. Find the middle point to divide the array into two halves:
             middle m = l+ (r-l)/2
     2. Call mergeSort for first half:
             Call mergeSort(arr, l, m)
     3. Call mergeSort for second half:
             Call mergeSort(arr, m+1, r)
     4. Merge the two halves sorted in step 2 and 3:
              Call merge(arr, l, m, r)
```

How Merge Sort works?



A list of the main functions

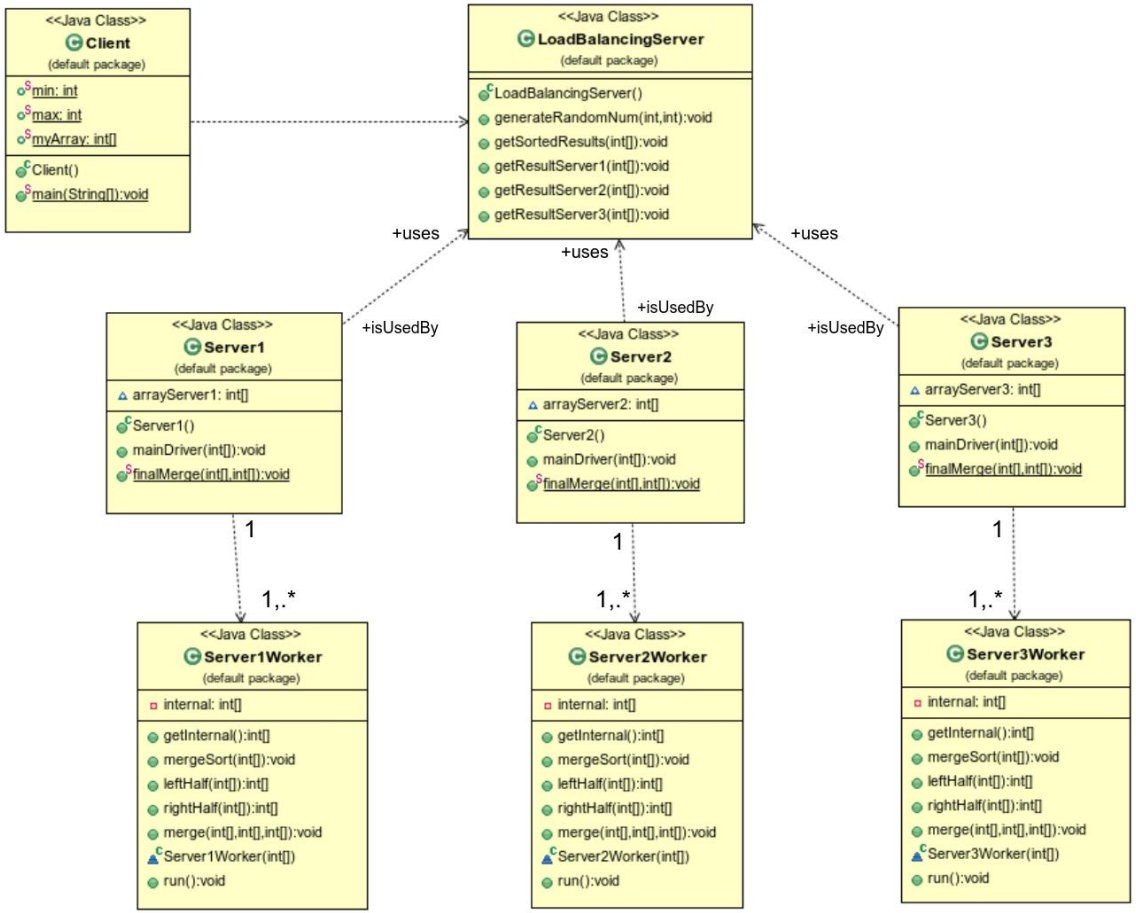
- generateRandomNum
- getSortedResults
- mainDriver
- finalMerge
- mergeSort

Error cases

- definable range format : InputMismatchException error
- resend request format : InputMismatchException error

Architecture

UML Class Diagram



How to run

```
// to compile programs
javac Client.java LoadBalancingServer.java Server1.java Server2.java Server3.java

// to execute the generated main class file
java Client
```

Thanks for your attention