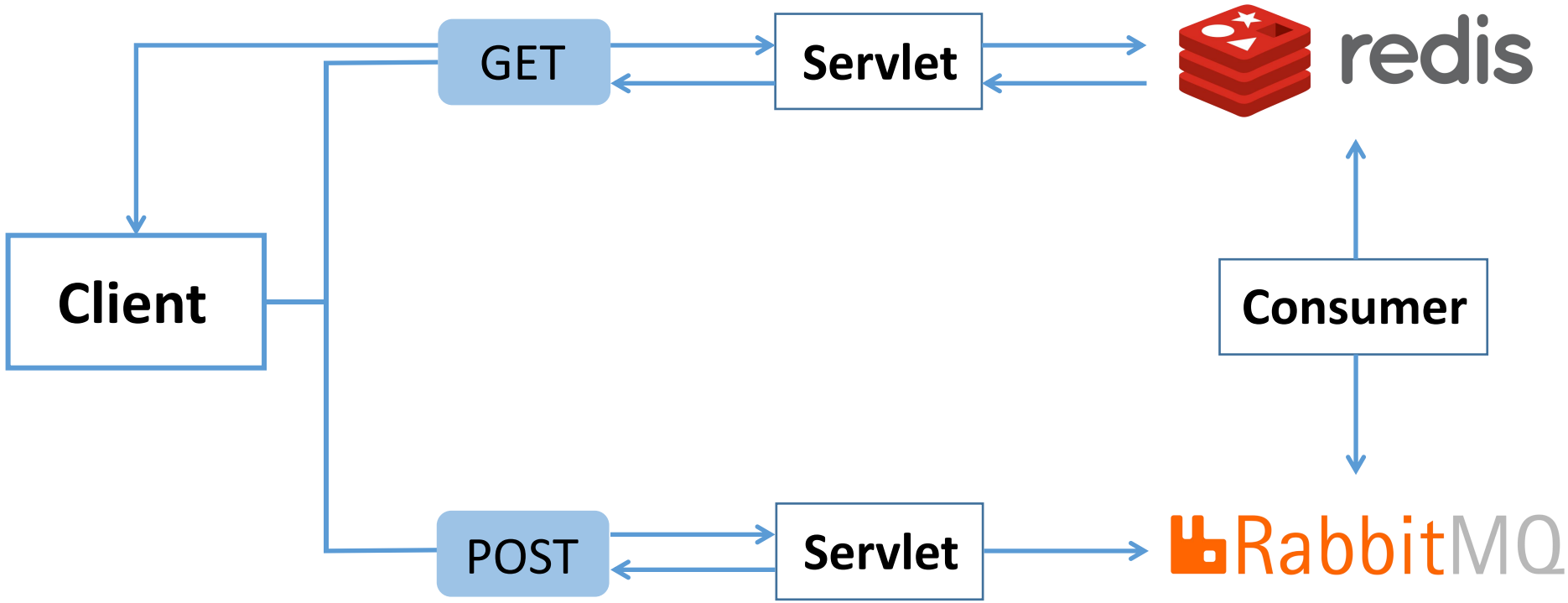
Assignment 4

Jiayue Wu
Donghui Zhang
Jianxun Wang
Sheng Chen

Architecture



Deployment

Name	Instance ID	Instance state	Instance type
6650_servlet_0	i-08d0eee8f74d2816c	Stopped	t2.micro
RMQ	i-0d85ddc98b1b7b60a	Stopped	t2.micro
redis	i-03dc53e082ff2ad9a	Stopped	t2.micro
consumer	i-0b248055a318216b5	Stopped	t2.micro

Client on local machine: sends 600k POST requests and 6400*3 GET requests to servlets.

6650_servlet_0 on AWS Linux EC2 instance: receives POST and GET requests from client.
Only deploys one servlet due to budget.

RMQ on Ubuntu instance: the remote queue using RabbitMQ to store messages.
Messages are posted from servlets and consumed by consumers.

consumer on AWS Linux EC2 instance: consumers that consume messages and write data into Redis database.

Redis on AWS Linux EC2 instance: Redis database

Data Model

```
public void writeEntry(String skierId, String resortId, String seasonId, String dayId, Integer liftId) {
    String skierStatKey = String.join( delimiter: "/", resortId, seasonId, dayId);
    connection.sadd(skierStatKey, skierId);

    String verticalStatKey = String.join( delimiter: "/", skierId, resortId, seasonId, dayId);
    connection.incrby(verticalStatKey, liftId);

    String skierVerticalStatValue = String.join( delimiter: "/", resortId, seasonId, liftId.toString());
    connection.rpush(skierId, skierVerticalStatValue);
}
```

API	Key Type	Key Content	Value Type	Value Content
GET: /resorts/{resortID}/seasons/{seasonID}/day/{dayID}/skiers	String	resortId/seasonId/dayId	Set<String>	set of unique skierId
GET: /skiers/{resortID}/seasons/{seasonID}/days/{dayID}/skiers/{skierID}	String	skierId/resortId/seasonId/dayId	Integer	sum of liftId
GET: /skiers/{skierID}/vertical	String	skierId	List<String>	resortId/seasonId/liftId

JMeter Tests

APDEX (Application Performance Index)

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
1.000	**500 ms**	**1 sec 500 ms**	**Total**
1.000	500 ms	1 sec 500 ms	Skier Total Vertical GET
1.000	500 ms	1 sec 500 ms	Skier Day Vertical GET
1.000	500 ms	1 sec 500 ms	Resort GET

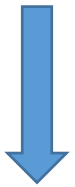
Requests Summary

Statistics

Requests	Executions			Response Times (ms)							Throughput	Network (KB/sec)	
Label	#Samples	FAIL	Error %	Average	Min	Max	Median	90th pct	95th pct	99th pct	Transactions/s	Received	Sent
Total	**19200**	**0**	**0.00%**	**21.23**	**9**	**120**	**21.00**	**28.00**	**31.00**	**35.00**	**1541.80**	**451.91**	**264.49**
Resort GET	6400	0	0.00%	19.22	9	120	19.00	25.00	27.00	32.00	514.88	111.33	86.48
Skier Day Vertical GET	6400	0	0.00%	22.04	9	46	22.00	29.00	31.00	36.00	516.17	95.77	89.22
Skier Total Vertical GET	6400	0	0.00%	22.43	9	83	23.00	29.00	31.00	35.00	516.17	246.49	89.72

JMeter Tests

single servlet



Average Response Time: 21.23ms

P99 Response Time: 35.00ms

Uses **only one servlet** due to budget, which is the temporary bottleneck of this system.

Could highly increase throughput if adding more servlets with load balancers.

Improvement

Servlet

single servlet --> more servlets with load balancers

Data Model

- fast read, but slow write
- data redundancy
- lack extensibility


NORTHEASTERN UNIVERSITY
LVX
VERITAS
VIRTVS
FOUNDED AD 1898
BOSTON, MASSACHUSETTS

Thank you!