Optimal Replication in Multi-Region Peer-to-peer Systems

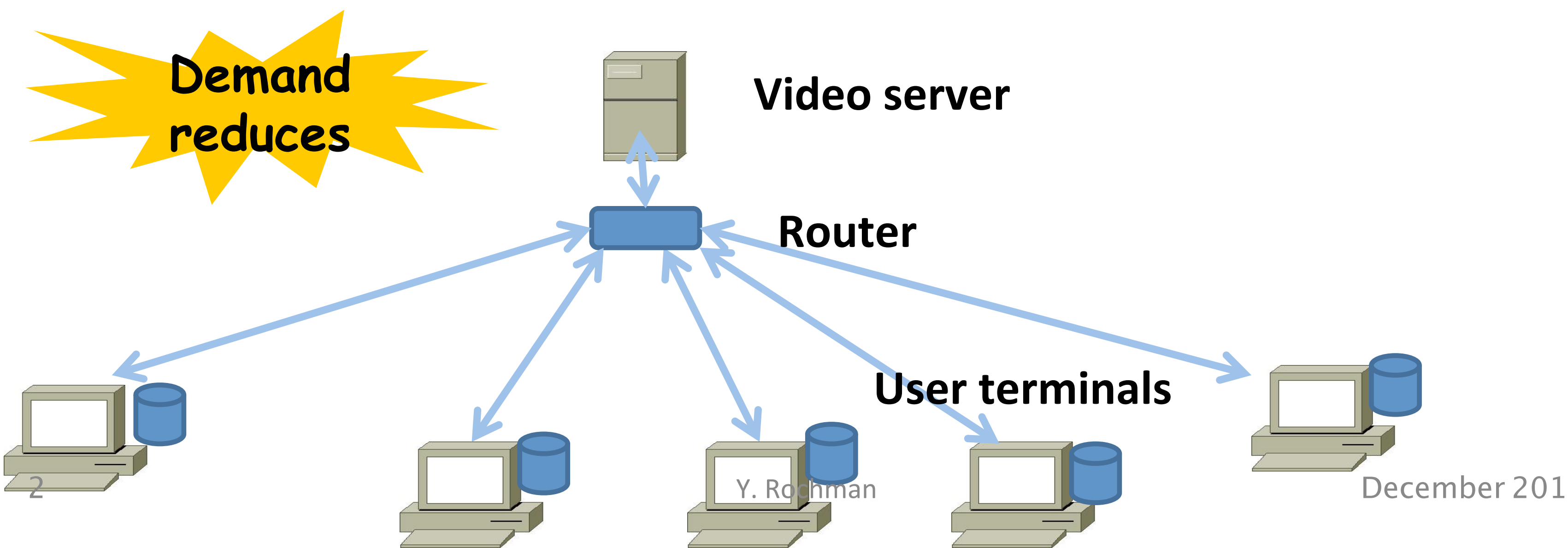
Yuval Rochman
Computer Science
Tel-Aviv Univ.

Hanoch Levy
Computer Science
Tel-Aviv Univ

Eli Brosh
Computer Science
Columbia Univ

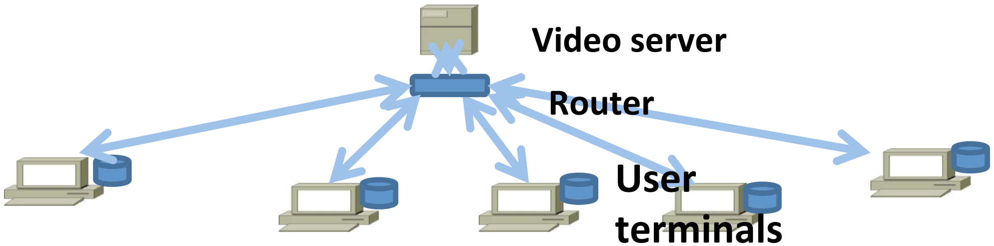
System and Problem

- **<u>Players</u>: End-User Terminals** + a **Central Video Server.**
- **<u>Function:</u> Provide Video-on-Demand (VoD).**
- **Objective: Central server is bottleneck ➔ need to reduce its load.**
- **<u>Solution:</u>** Use user terminals to store movies and upload to other terminals (Peer-to-Peer).
- All controlled by central system (e.g. HOT).

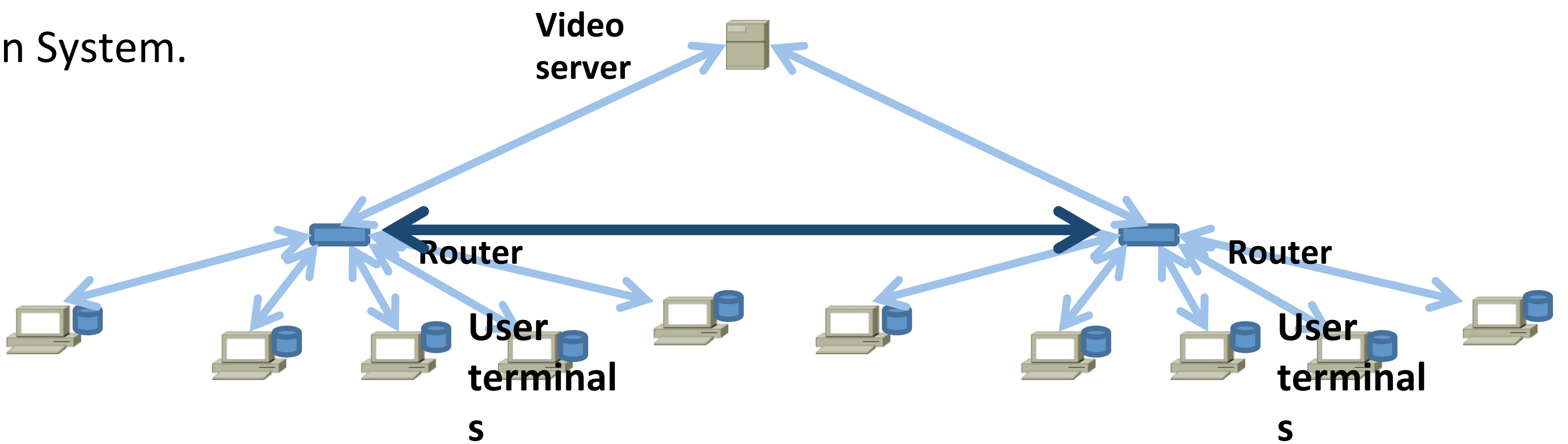


Single- and Multi-Region

Single Region System.

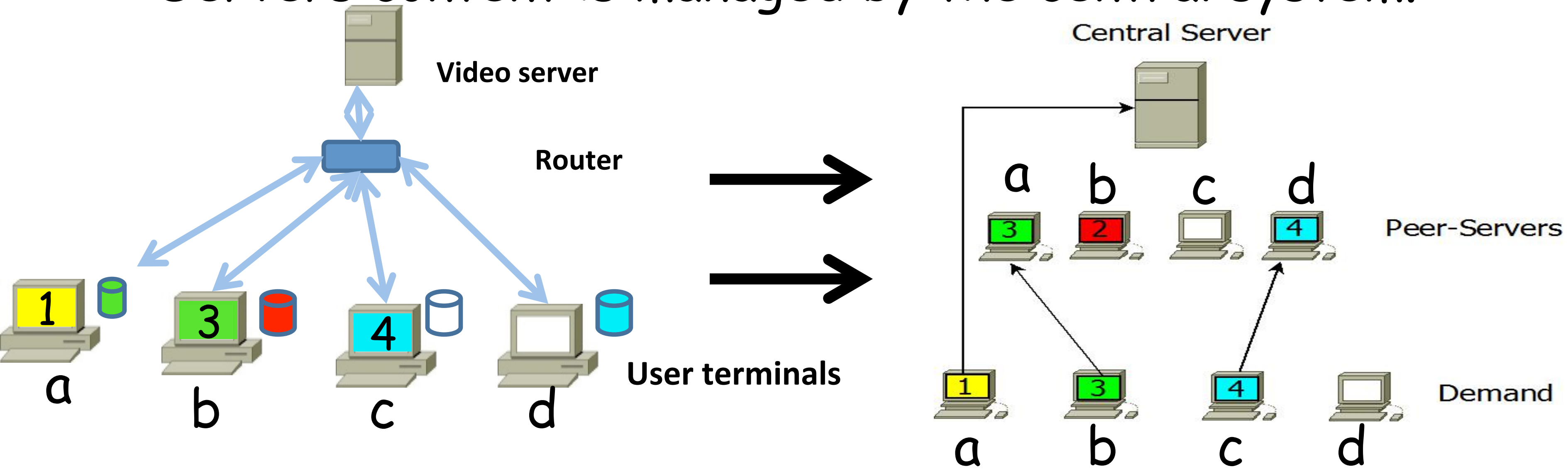


Multi Region System.



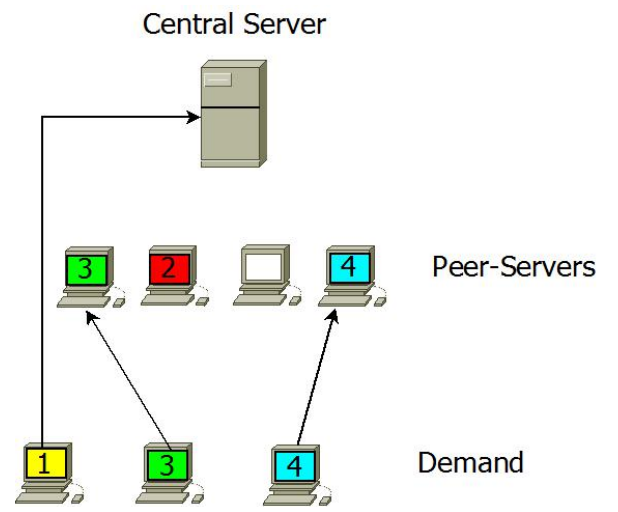
Model and Assumptions

- Decompose terminals to server + client.
- Each server can store <u>one</u> movie.
- Server can upload <u>one</u> movie.
- Client can download one movie.
- Collection of movies.
- Servers content is managed by the central system.



Problem and Profit (single region)

- Given: movie demand expected by the clients.
- Need to:
  - Place movies at the servers .
  - Assign clients requests to the servers.
- Profit: # of requests granted by peer-servers.
  - In figure: profit = 2.
- Objective: maximize the profit.



- Eliminate null-requests

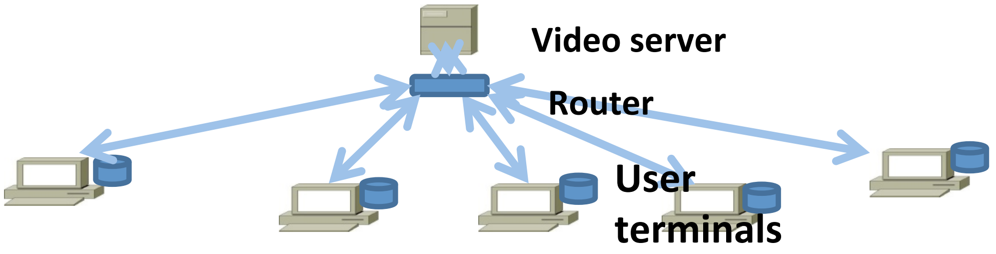
Related Work

- Tewari & Kleinrock [2006]
  Proposed the Proportional Mean Replication.
  - Difference: A peer-server serves unbounded clients.
- Zhou, Fu & Chiu [2011]

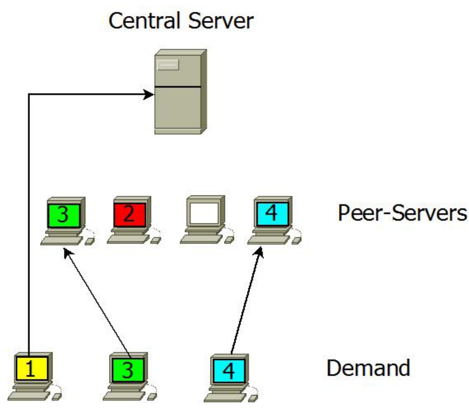
Proposed the RLB Replication.

  - Difference: assume that the number of peers>> number of movies.

Demonstration of the approach:
The Single Region System



Formulation

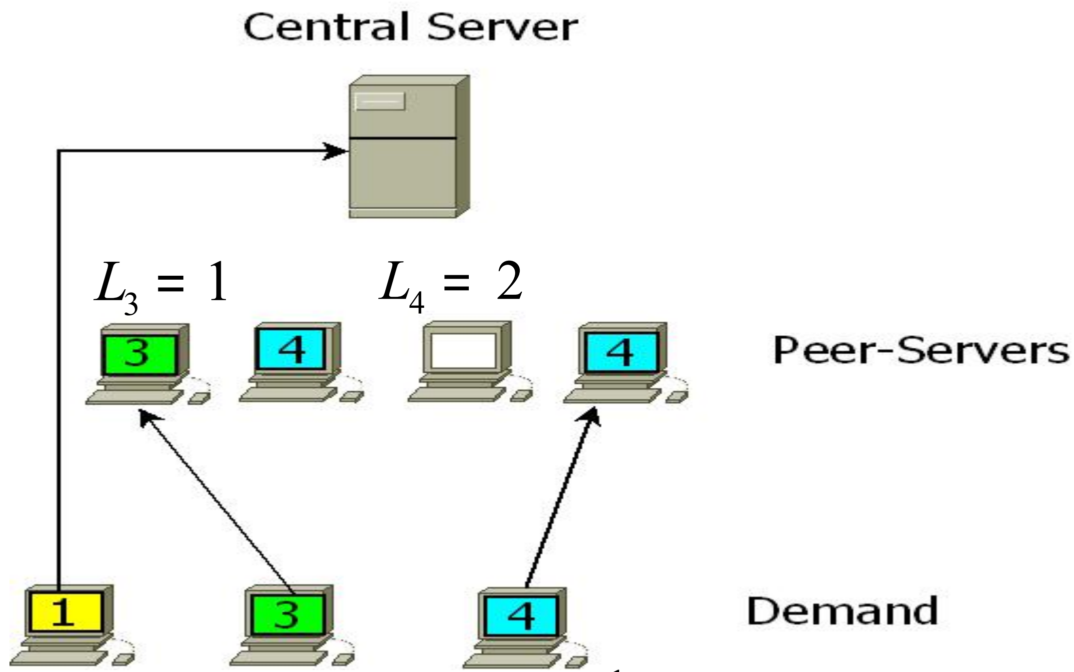


- A collection of movies $\{1,2,\ldots,m\}$
- # of peer-servers: S (constant)
- Demand for movie i: N_i, <u>random variable</u>.
  - Observed value of N_i: n_i .
- **<u>The goal:</u>** Determine the number of movie i replicas, L_i to optimize performance.

Example

$m = 4$

$S = 4$



$n_1 = 1$ $n_3 = 1$ $n_4 = 1$

$$profit = \min(0,1) + \min(1,0) + \min(1,1) + \min(2,1) = 2$$

We have to maximize: $E(profit) = \sum_{i=1}^{m} E_{N_i}(\min(L_i, N_i))$

Under the condition: $\sum_{i=1}^{m} L_i \le S$

Solution(1)

- Tail Formula: For every random non-negative integer variable:

$$E(X) = \sum_{i=1}^{\infty} \Pr(X \geq i)$$

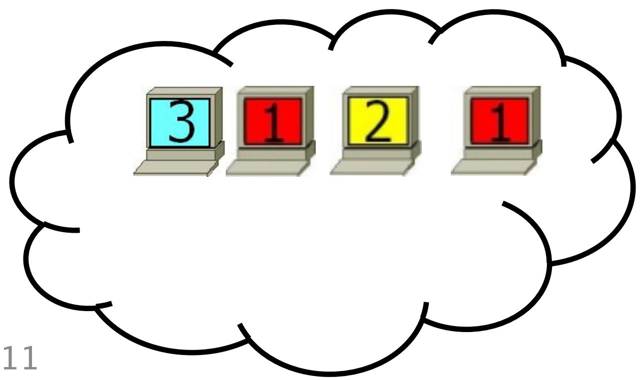
- ➔ $$E(profit) = \sum_{i=1}^{m} E(\min(L_i, N_i)) = \sum_{i=1}^{m} \sum_{j=1}^{L_i} \Pr(N_i \geq j)$$

Solution(2)

Max $E(profit) = \sum_{i=1}^{m} \sum_{j=1}^{L_i} \Pr(N_i \geq j)$

S.t $\sum_{i=1}^{m} L_i \leq S$

- Equivalent to: find S largest elements from $\{d_i\}_{i=1}^{m}$ where $d_i = (\Pr(N_i \geq 1), \Pr(N_i \geq 2), \Pr(N_i \geq 3), \ldots)$
- Solution: Max-percentile algorithm in $O(s \log s + m)$.



Æ

d_1 $\Pr(N_1 \geq 1)$ $\Pr(N_1 \geq 2)$ $\Pr(N_1 \geq 3)$

d_2 $\Pr(N_2 \geq 1)$ $\Pr(N_2 \geq 2)$ $\Pr(N_2 \geq 3)$

d_3 $\Pr(N_3 \geq 1)$ $\Pr(N_3 \geq 2)$ $\Pr(N_3 \geq 3)$

Max-percentile

		$\Pr(N_i \geq 1)$	$\Pr(N_i \geq 2)$	$\Pr(N_i \geq 3)$	$\Pr(N_i \geq 4)$
Movie 1	d_1	1	0.8	0.7	0.1
Movie 2	d_2	0.9	0.6	0.5	0.4
Movie 3	d_3	0.8	0.3	0.2	0.1
Movie 4	d_4	0.35	0.1	0	0

Blue-candidates

Red-selected

Choose the 6 largest elements

Max-percentile

		$Pr(N_i \geq 1)$	$Pr(N_i \geq 2)$	$Pr(N_i \geq 3)$	$Pr(N_i \geq 4)$
Movie 1	d_1	1	0.8	0.7	0.1
Movie 2	d_2	0.9	0.6	0.5	0.4
Movie 3	d_3	0.8	0.3	0.2	0.1
Movie 4	d_4	0.35	0.1	0	0

Blue-candidates

Red-selected

Choose the 6 largest elements

Max-percentile

		$Pr(N_i \geq 1)$	$Pr(N_i \geq 2)$	$Pr(N_i \geq 3)$	$Pr(N_i \geq 4)$
Movie 1	d_1	1	0.8	0.7	0.1
Movie 2	d_2	0.9	0.6	0.5	0.4
Movie 3	d_3	0.8	0.3	0.2	0.1
Movie 4	d_4	0.35	0.1	0	0

Blue-candidates

Red-selected

Choose the 6 largest elements

Max-percentile

		$\Pr(N_i \geq 1)$	$\Pr(N_i \geq 2)$	$\Pr(N_i \geq 3)$	$\Pr(N_i \geq 4)$
Movie 1	d_1	1	0.8	0.7	0.1
Movie 2	d_2	0.9	0.6	0.5	0.4
Movie 3	d_3	0.8	0.3	0.2	0.1
Movie 4	d_4	0.35	0.1	0	0

Blue-candidates

Red-selected

Choose the 6 largest elements

Max-percentile

		$\Pr(N_i \geq 1)$	$\Pr(N_i \geq 2)$	$\Pr(N_i \geq 3)$	$\Pr(N_i \geq 4)$
Movie 1	d_1	1	0.8	0.7	0.1
Movie 2	d_2	0.9	0.6	0.5	0.4
Movie 3	d_3	0.8	0.3	0.2	0.1
Movie 4	d_4	0.35	0.1	0	0

Blue-candidates

Red-selected

Choose the 6 largest elements

Max-percentile

		$\Pr(N_i \geq 1)$	$\Pr(N_i \geq 2)$	$\Pr(N_i \geq 3)$	$\Pr(N_i \geq 4)$
Movie 1	d_1	1	0.8	0.7	0.1
Movie 2	d_2	0.9	0.6	0.5	0.4
Movie 3	d_3	0.8	0.3	0.2	0.1
Movie 4	d_4	0.35	0.1	0	0

Blue-candidates

Red-selected

Choose the 6 largest elements

Max-percentile

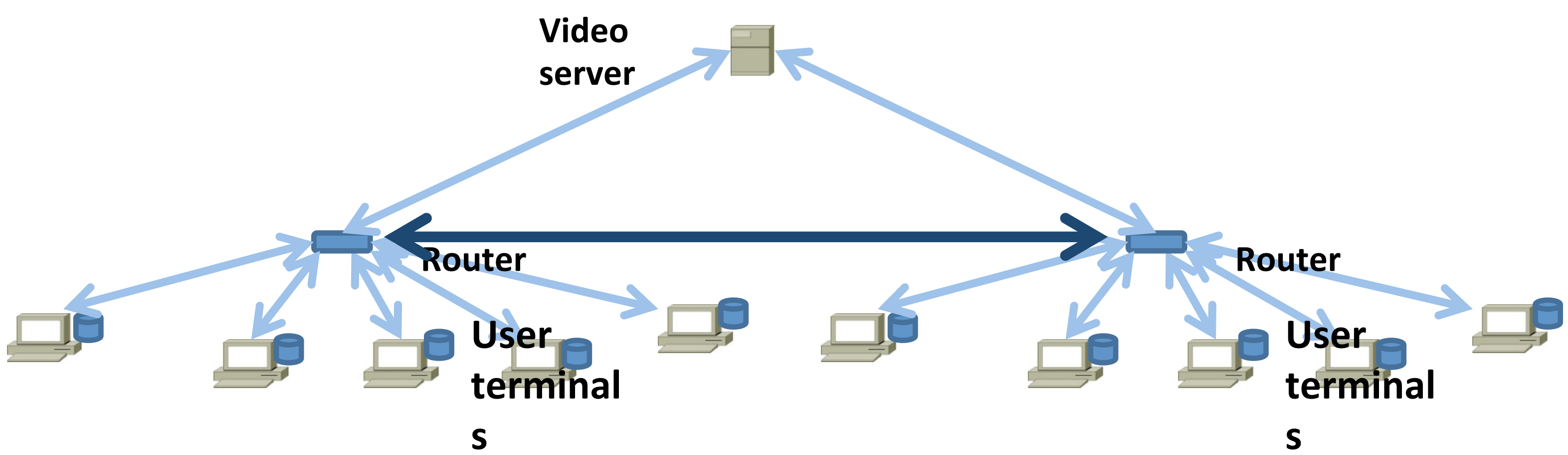
		$\Pr(N_i \geq 1)$	$\Pr(N_i \geq 2)$	$\Pr(N_i \geq 3)$	$\Pr(N_i \geq 4)$
Movie 1	d_1	1	0.8	0.7	0.1
Movie 2	d_2	0.9	0.6	0.5	0.4
Movie 3	d_3	0.8	0.3	0.2	0.1
Movie 4	d_4	0.35	0.1	0	0

Blue-candidates

Red-selected

Choose the 6 largest elements

Multi-region System

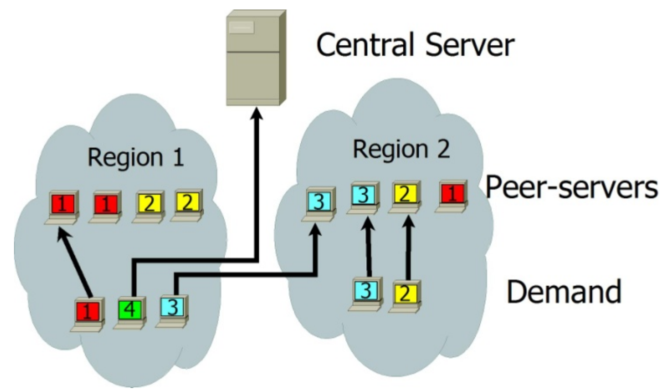


The Multi-Regional problem

$k = 2$

$s = 8$

- Assume: k regions
- A demand is satisfied by:
  - **<u>Local</u>** region
  - **<u>Remote</u>** region
  - **<u>Central server</u>**.
- Added profit per granted demand:
  - Local: R_{loc}
  - Remote: R_{rem}
  - Server: R_{ser}
- Assume: R_{ser} £ R_{rem} £ R_{loc}



- Example:

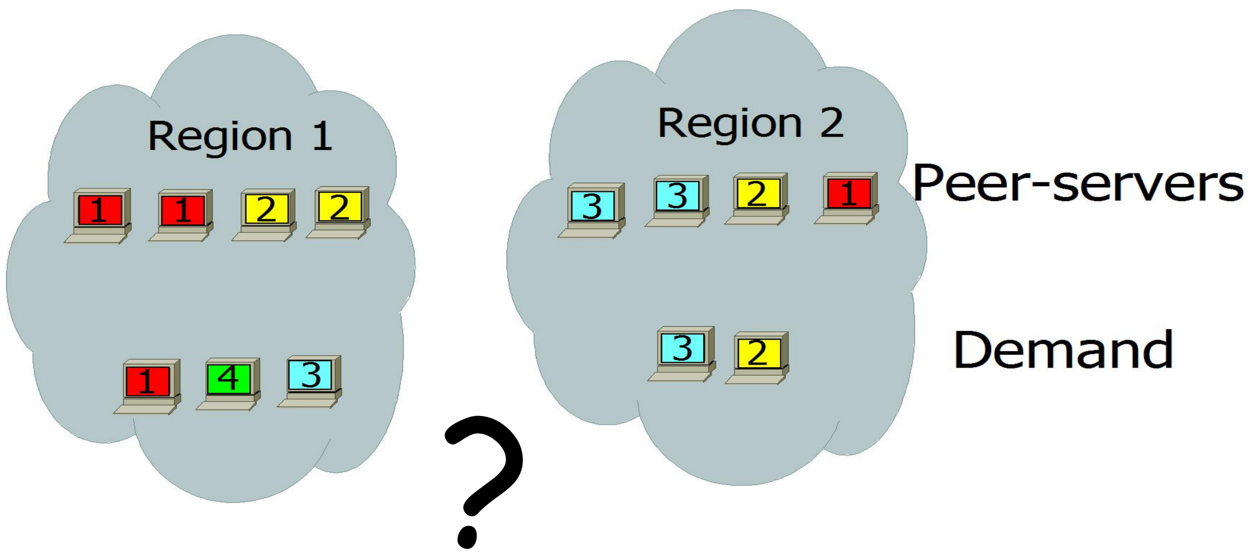
$$1 \lozenge R_{ser} + 1 \lozenge R_{rem} + 3 \lozenge R_{loc}$$

- Profit in general case:

$$R = g_{ser} R_{ser} + g_{rem} R_{rem} + g_{loc} R_{loc}$$

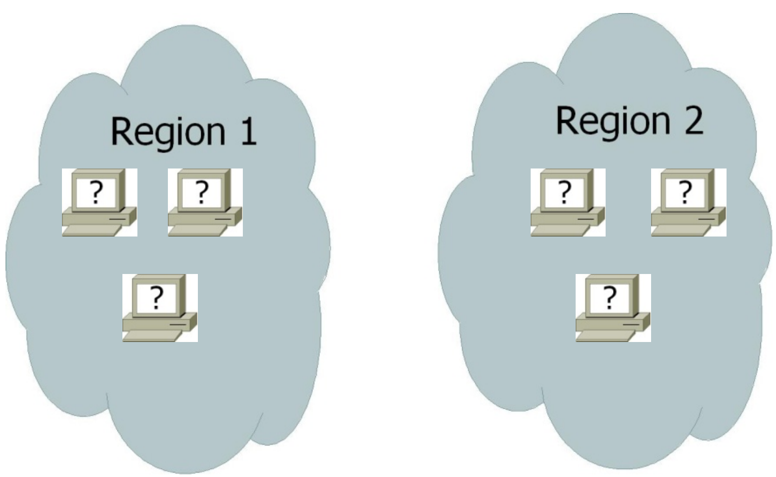
The Multi-Regional problems(1)

- **Matching Problem:**
  - Given deterministic demand, movie replicas.
  - Match between demand and servers
  - Maximize the profit



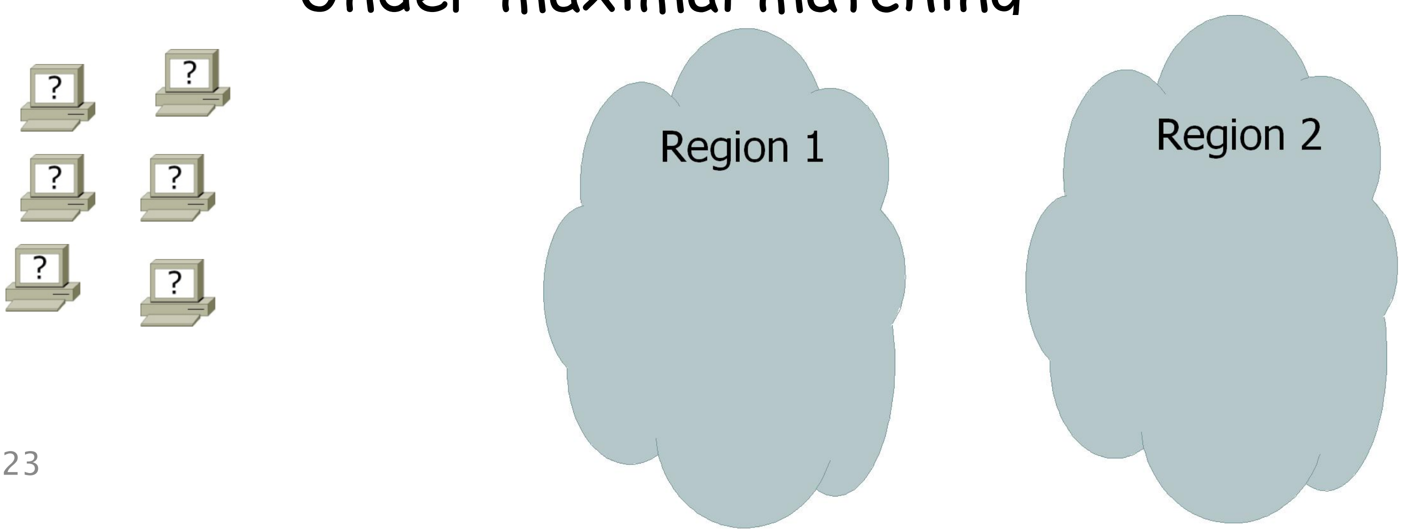
The Multi-Regional problems(2)

- Given arbitrary **<u>stochastic demand.</u>**
- **<u>Symmetrically-Server Allocation (SSA) Problem:</u>**
  – S/k servers in every region.
  – Place movie replicas in servers.
  – Maximize the <u>expected profit</u>
  – Under maximal matching



The Multi-Regional problems(3)

- Given arbitrary **stochastic demand.**
- **Free-Server Allocation(FSA) Problem:**
  - S servers.
  - Free to place servers.
  - Place movies in servers.
  - Maximize the expected profit
  - Under maximal matching

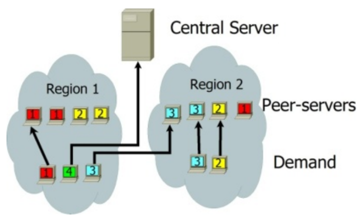


The Multi-Regional problems(3)

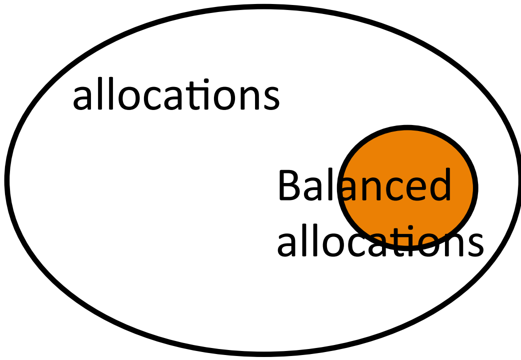
- **<u>Theorem:</u>**
  SSA solution=FSA solution
- MurMap algorithm finds it.

Analysis: Key Principles

- Solve the Matching Problem



- Optimal allocation must be "balanced" ➔ reduce size of candidate set



Questions

?

Thank you