Analysis of Design Alternatives for Reverse Proxy Cache Providers

Bruno Ciciani, Francesco Quaglia, Paolo Romano
Computer Engineering Department, University of Rome "La Sapienza"

Daniel Dias
IBM T.J. Watson Research Center, Yorktown, N.Y.

Main research project: SLA and penalty minimization

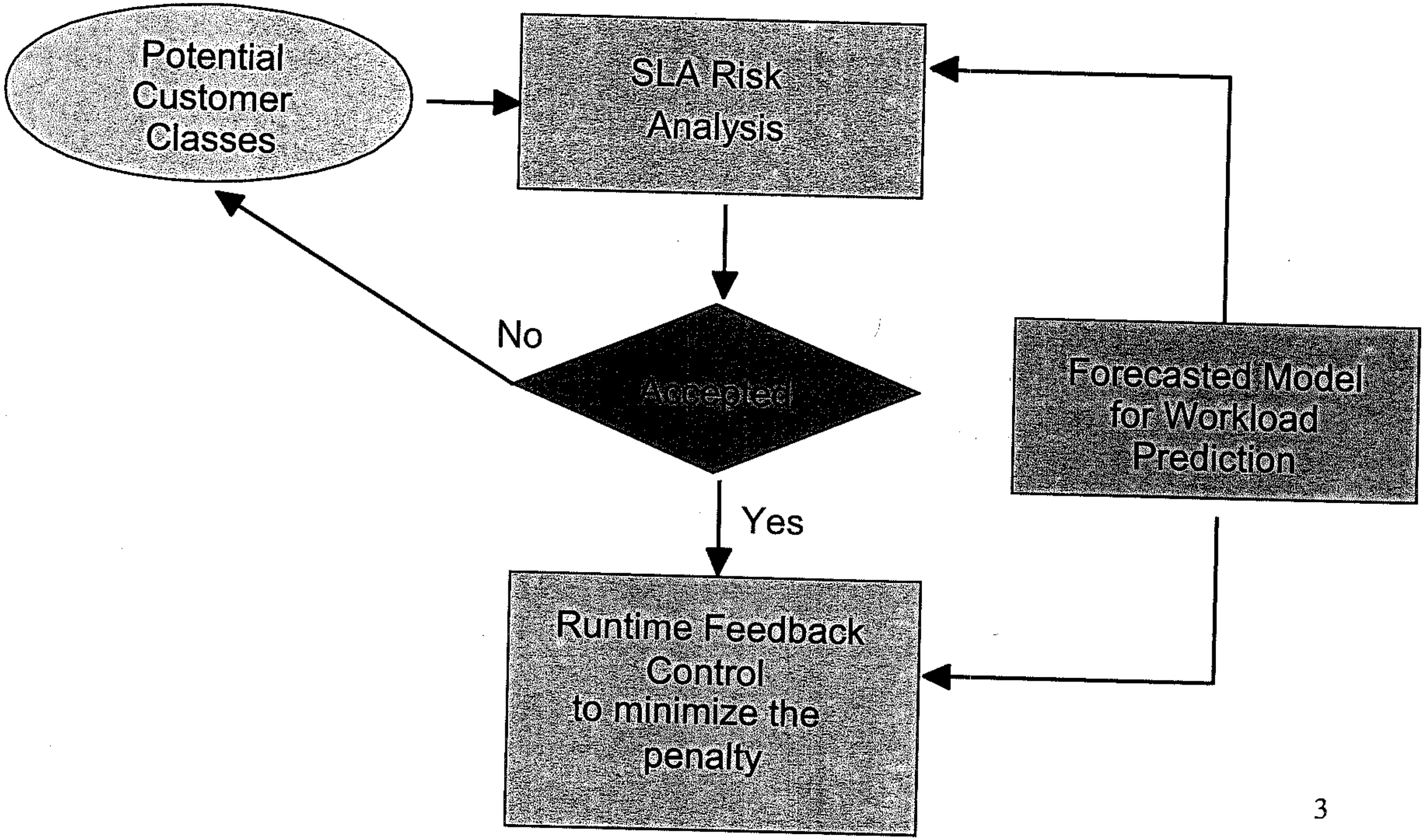
- Service provider economical risk analysis in planning phase

- Run-time minimization penalty control

Reference platform:

- Content hosting
- Reverse proxy cache

Process Flow



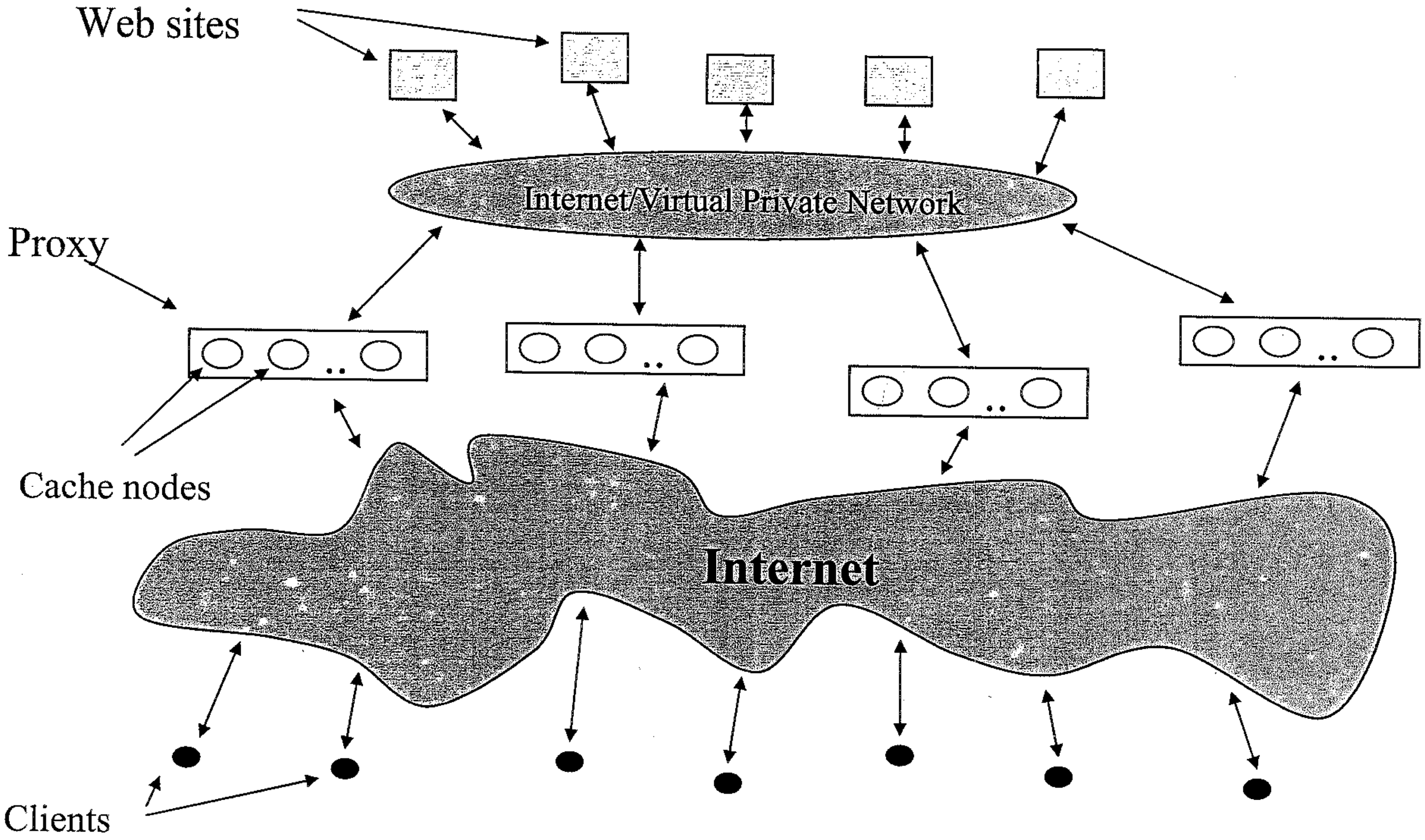
SLA Risk analysis (4 phases)

1. Definition of the parameters involved in the SLA.
2. Worload characterization and service time identification.
3. Platform an resorse allocation policy modeling and evaluation.
4. Economical risk identification.

Modeling and Evaluation of Archicture Alternatives for Reverse Proxy Cache Providers

- Reverse proxy cache geographically distributed, organized in a hierarchical manner.
- Limited number of customers (less than one hundred), that share the resources.
- Proxy servers implemented over cluster of workstation.
- Proxy servers connected through a virtual private networks to the Web Servers.

Architecture



Advantage of Reverse Proxy Cache

- Reduction the load of the Web Servers.
- Improvement of the throughput.
- Reduction of the latency.
- Multiple Web Sites can share the infrastructure.

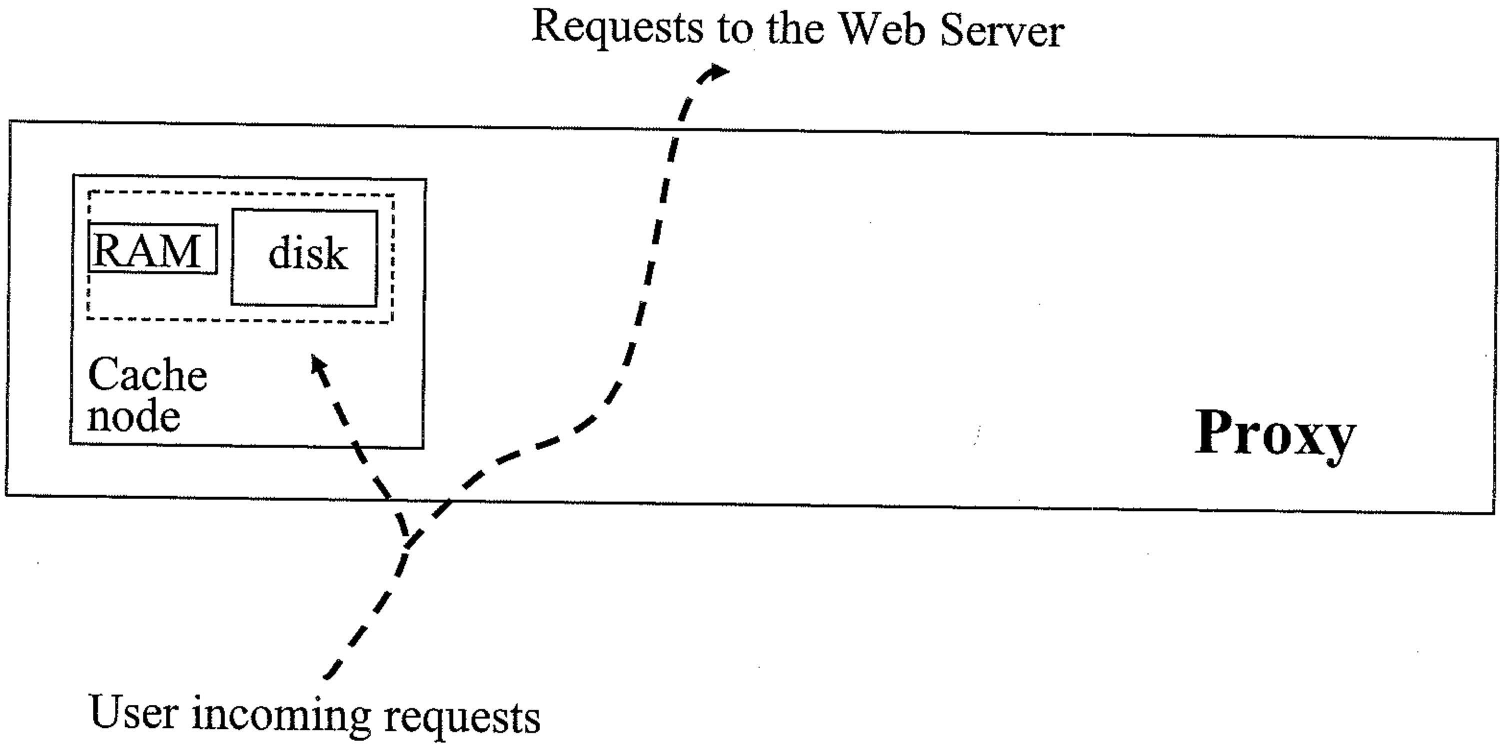
Contribution of the paper

- The proposed model takes care of the real design constraints:
  - » Bounded cache size;
  - » Bounded processing power;
  - » Popularity of the documents;
  - » Update rates of the documents.
- The model permits the identification of the architecture tradeoffs, depending on:
  - » Resource assignment policy;
  - » Workload characteristics.
- The model permits the identification of:
  - » Steady State ans Transient behavior of the architectures.

Analized resource allocation policies

- Exclusive vs Shared Cache Node Assigment.
- Static vs Dynamic RAM Partitioning.
- Statics vs Dynamic Cache Node Assignment.

Request management



- Proxy configuration: no global memory management
- Cache content defined by access pattern (object popularity)

Nomenclature

WS_k	k-th Web site
C_{WS}^k	total RAM capacity of WS_k
λ_k	arrival rate of HTTP requests to WS_k
n_k	total number of cacheable objects associated with WS_k
α_k	parameter of the Zipf-like distribution associated with WS_k
$p_{k,j}$	relative popularity of the j-th cacheable object of WS_k
$\mu_{k,j}$	update rate of the j-th cacheable object of WS_k
λ_k^{CN}	request arrival rate, associated with WS_k, seen by any single cache node
C^{tot}	total cache node RAM capacity
C_k	cache node RAM capacity destined to cacheable objects of WS_k
MR_k	miss ratio within the cache node RAM/disk for requests associated with WS_k
RHR_k	cache node RAM hit ratio for cacheable objects of WS_k
DHR_k	cache node disk hit ratio for cacheable objects of WS_k
N	total number of Web sites hosted by a cache node
NP	total number of Proxy sites
NCN_k	number of cache nodes within a Proxy site that are assigned to WS_k

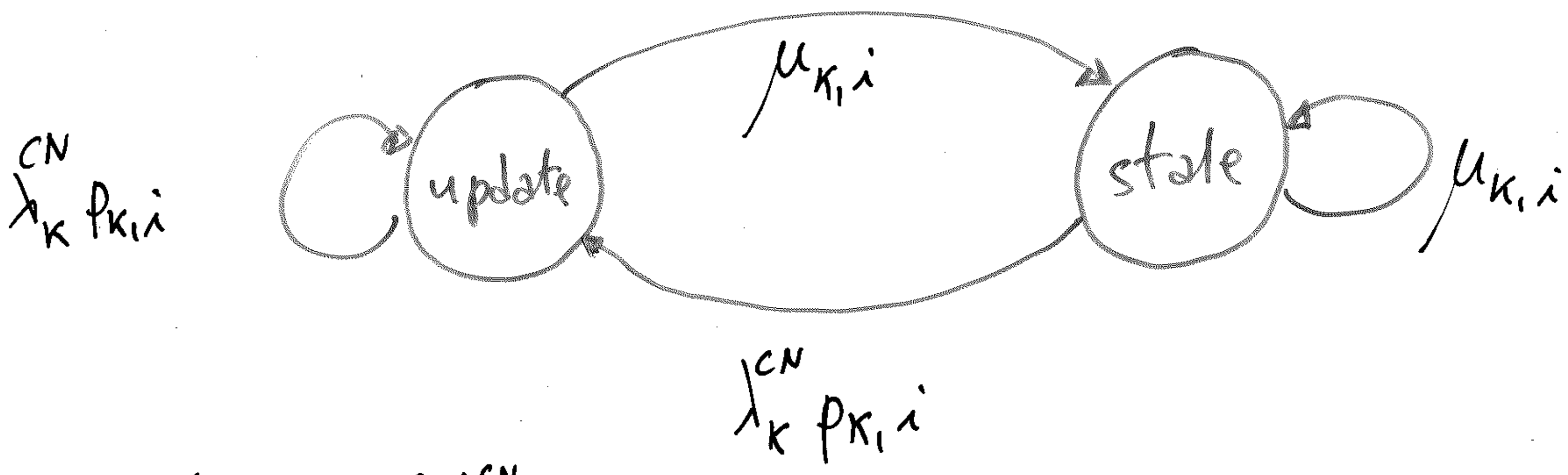
Hypothesis

- Arrival process: Poisson process
- Uniform load for each Proxy site
- LFU replacement policy
- All documents can be memorized in the disk subsystem
- document probability request: Zipf-like distribution

i.e.:

$$p_{k,i} = \frac{\Omega}{i^{\alpha_k}} \qquad \Omega = \sum_{i=1}^{\#doc} p_{k,i}\, i^{\alpha_k}$$

Evaluation of Cache Node Hit/Miss Ratio



$$\begin{cases} p_{up} \cdot \mu_{k,i} = p_{stale}\, \lambda_k^{CN} p_{k,i} \\ p_{up} + p_{stale} = 1 \end{cases} \longrightarrow \quad p_{up} = \frac{\lambda_k^{CN} p_{k,i}}{\mu_{k,i} + \lambda_k^{CN} p_{k,i}}$$

$$p_{stale} = \frac{\mu_{k,i}}{\mu_{k,i} + \lambda_k^{CN} p_{k,i}}$$

Miss Ratio

$$MR_k = \sum_{i=1}^{object\ set} p_{k,i} \frac{\mu_{k,i}}{\lambda_k^{CN} p_{k,i} + \mu_{k,i}}$$

Parameters

- $p_{k,i}$ = document popularity
- $\mu_{k,i}$ = document update rate
- $\lambda_{k,i}^{CN}$ = document cache node access rate

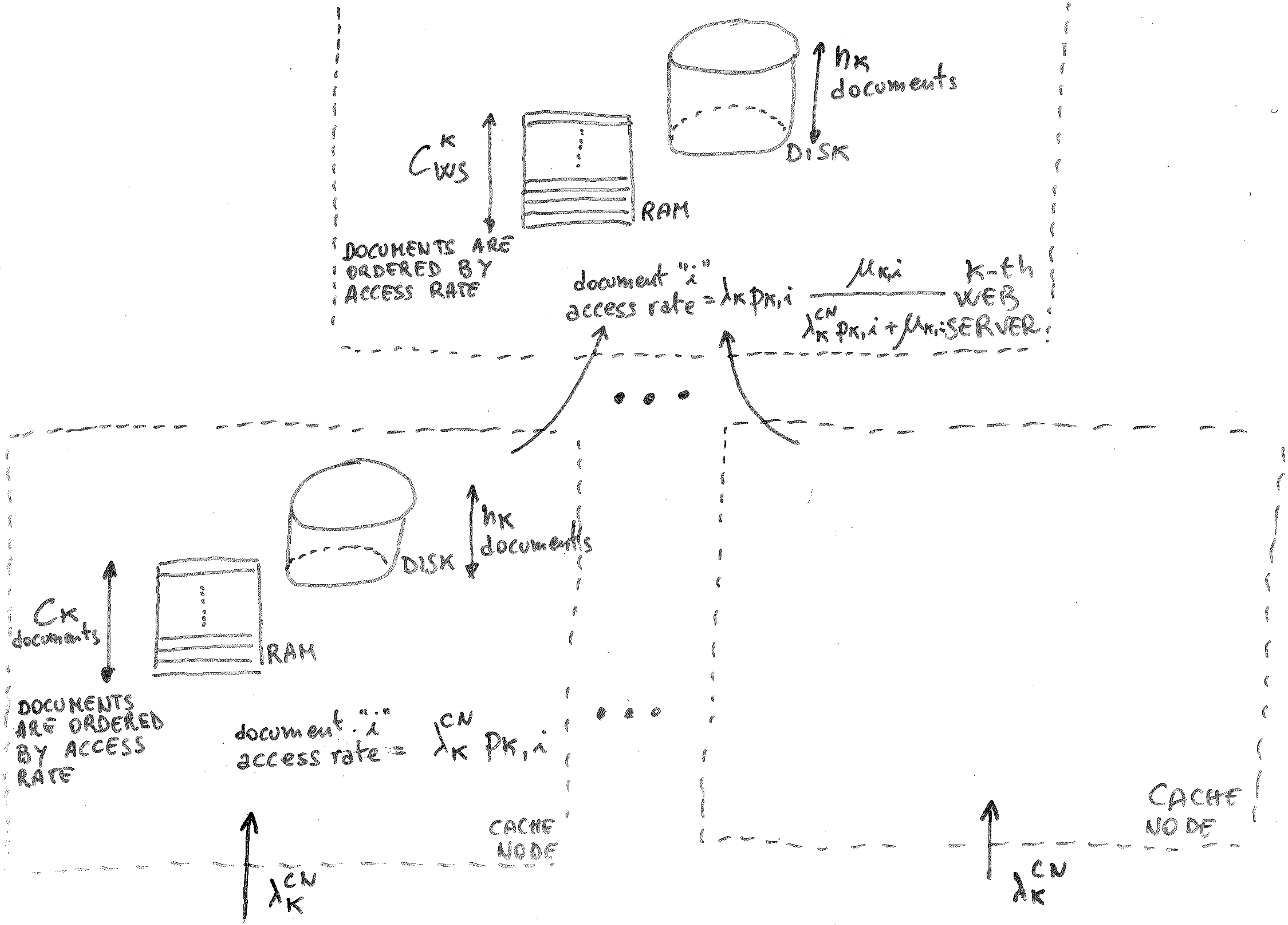
Hit ratio

RAM hit ratio (RAM with finite dimension – capacity for C_k documents)

$$RHR_k = (1 - MR_k) \sum_{i=1}^{min(C_k, n_k)} p_{k,i}$$

DISK hit ratio (storage capacity enough to store all the documents)

$$DHR_k = (1 - MR_k) \sum_{i=min(C_k, n_k)+1}^{n_k} p_{k,i}$$



n_K documents
DISK
C^K_{WS}
RAM
DOCUMENTS ARE ORDERED BY ACCESS RATE
document "i" access rate $= \lambda_K p_{K,i} \dfrac{\mu_{K,i}}{\lambda^{CN}_K p_{K,i} + \mu_{K,i}}$
K-th WEB SERVER
n_K documents
DISK
C_K documents
RAM
DOCUMENTS ARE ORDERED BY ACCESS RATE
document "i" access rate $= \lambda^{CN}_K p_{K,i}$
CACHE NODE
λ^{CN}_K
CACHE NODE
λ^{CN}_K

13'

Exclusive Cache Node Assignment

$$C_k = C^{tot}$$

Processor activities modeled as M/G/1/PS

$$\lambda_k^{CN} = \frac{1}{NCN_k}\frac{\lambda_k}{NP}$$

$$\rho_{CPU} = \lambda_k^{CN}(E[ram\_hit] + DHR_k E[disk\_request] + MR_k E[http])$$

$$\rho_{disk} = \lambda_k^{CN}(DHR_k + MR_k)E[disk]$$

Exclusive Cache Node Assignment (cont.)

$$\rho_{WS\_CPU} = \lambda_k MR_k(E[WS\_http] + \sum_{\forall i:\ I_{k,i} > C^k_{WS}} p_{k,i} E[WS\_disk\_request])$$

$$\rho_{WS\_disk} = \lambda_k MR_k \sum_{\forall i:\ I_{k,i} > C^k_{WS}} p_{k,i} E[disk]$$

$$\begin{aligned} T \;=\; & \frac{E[ram\_hit]}{1-\rho_{CPU}} + DHR_k\left(\frac{E[disk\_request]}{1-\rho_{CPU}} + \frac{E[disk]}{1-\rho_{disk}}\right) + MR_k\left(\frac{E[http]}{1-\rho_{CPU}} + \right. \\ & \left. \frac{E[WS\_http]}{1-\rho_{WS\_CPU}} + \sum_{\forall i:\ I_{k,i} > C^k_{WS}} p_{k,i}\left(\frac{E[WS\_disk\_request]}{1-\rho_{WS\_CPU}} + \frac{E[WS\_disk]}{1-\rho_{WS\_disk}}\right) + \Delta\right) \end{aligned}$$

Shared Cache Node Assignment with Static RAM Partitioning

$$C_k = \frac{C^{tot}}{N}$$

$$\lambda_k^{CN} = \frac{1}{NCN_k} \frac{\lambda_k}{NP}$$

$$\rho_{CPU} = \sum_{k=1}^{N} \lambda_k^{CN} (E[ram\_hit] + DHR_k E[disk\_request] + MR_k E[http])$$

$$\rho_{disk} = \sum_{k=1}^{N} \lambda_k^{CN} (DHR_k + MR_k) E[disk]$$

Shared Cache Node Assignment with Dynamic RAM Partitioning

The document presence is based on the total popularity

$I_{k,j}$: index position of the j-th document of k-th WS

$$I_{k,j} = (\lambda_k / NCN_k\, NP)\, p_{k,j}$$

Memory capacity assigned to the k-th WS

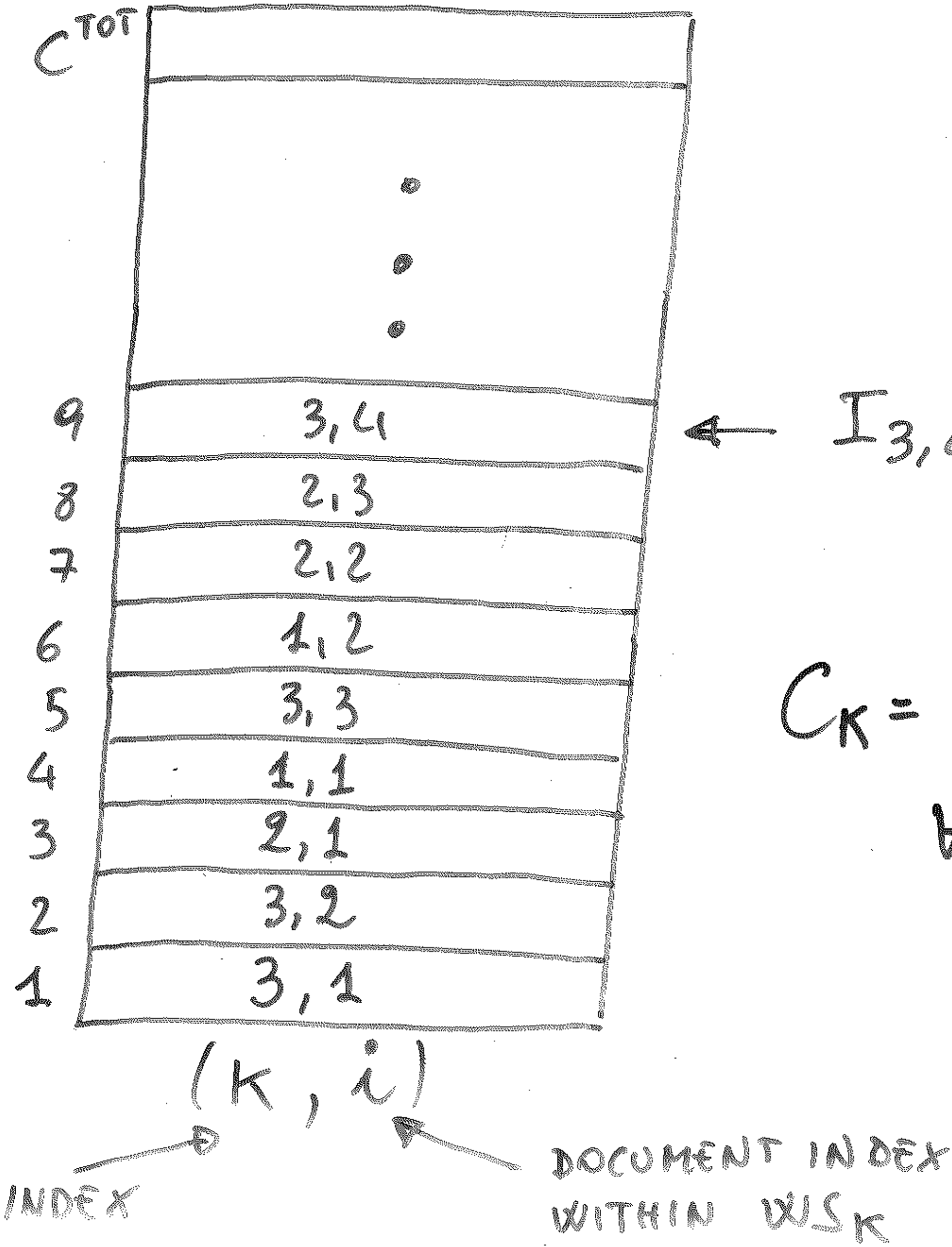
$$C_k = \sum_{\forall I_{k,j} \leq C^{tot}} 1$$

Shared Cache Node Assignment with Dynamic RAM Partitioning

DOCUMENTS ARE ORDERED BY ACCESS RATE

$\lambda_K^{CN} p_{K,i}$

(total popularity)



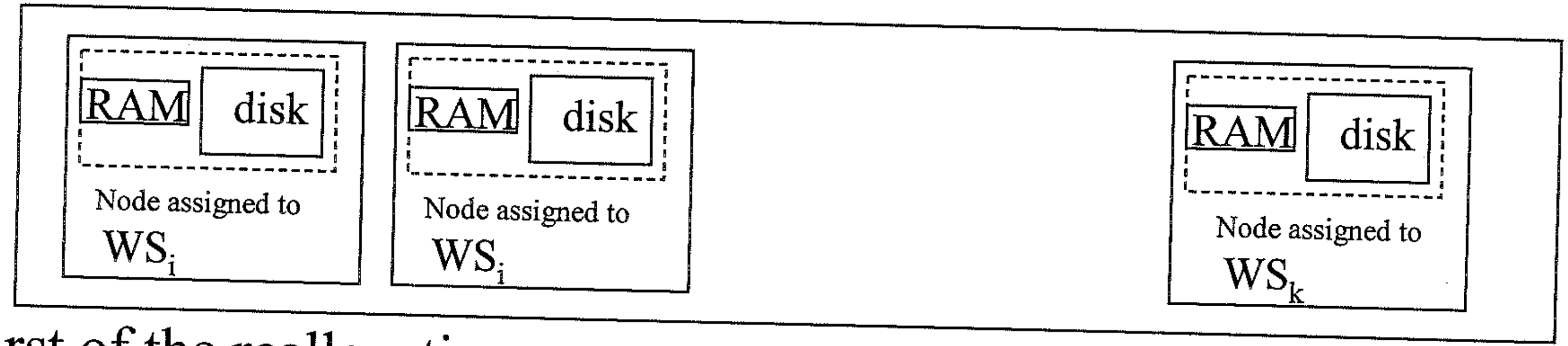
$I_{3,4} = 9$

$$C_K = \sum 1$$

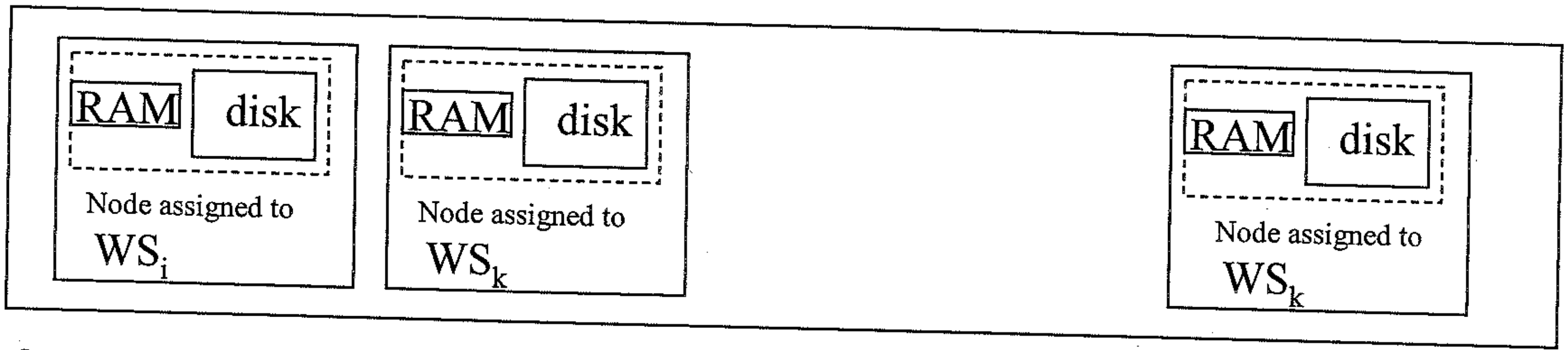
$$\forall I_{K,i} \leq C^{TOT}$$

Transient behavior

(case: node static partition)



First of the reallocation



After the reallocation

Transient behavior

Evaluation of the peak traffic on WS_k due warm-up

Conditional probability no request for the j-th object of WS_k at the newly assigned node, given M request to WS_k have been issued

$$X_{k,j}(M) = (1 - p_{k,j})^M$$

Cache node miss ratio due to warm-up at the M+1 arrival request arrival

$$MRWU_k = \sum_{i=1}^{n_k} p_{k,i} X_{k,i}(M) = \sum_{i=1}^{n_k} p_{k,i} (1 - p_{k,i})^M$$

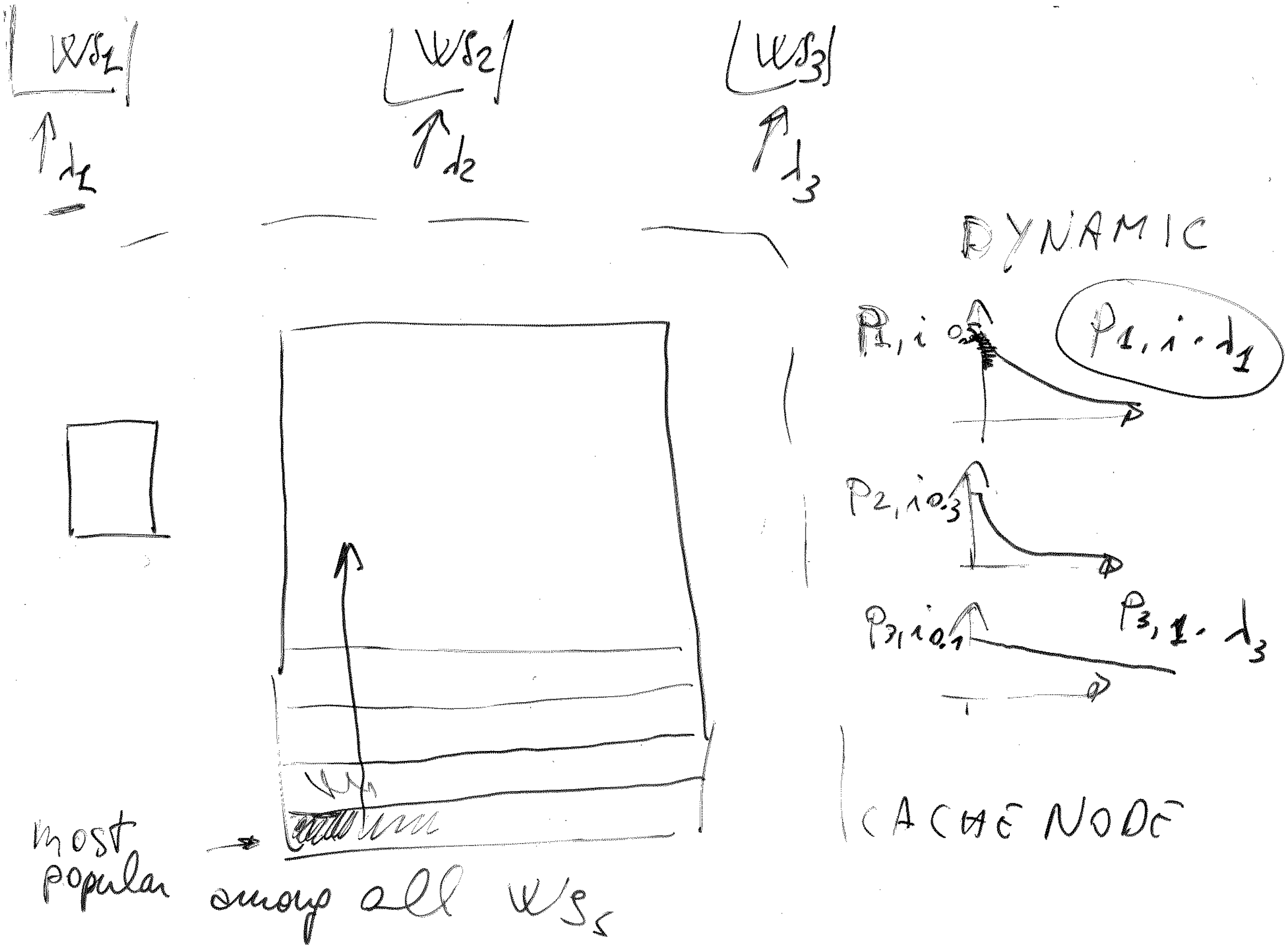
Transient behavior(cont)

Number of request generated in a δt time interval

$$M = \lambda_k^{CN} \delta t$$

Instant arrival rate at WS_k in the warm-up period

$$\lambda_k^{WU} = \lambda_k^{CN} MRWU_k = \lambda_k^{CN} \sum_{i=1}^{n_k} p_{k,i}(1 - p_{k,i})^{\lambda_k^{CN} \delta t}$$



WS1
WS2
WS3
λ1
λ2
λ3
DYNAMIC
p1,i 0.5
p1,i · λ1
p2,i 0.3
p3,i 0.1
p3,1 · λ3
WS1
CACHE NODE
most popular among all WSs

Quantitative comparison

- 50 Web sites (5 homogeneous groups of 10 WS)
- 10 Proxy sites
- 10 Cache nodes per proxy site
- 2 Cache nodes have to manage an homogenous group

WS_0	WS_1	WS_2	WS_3	WS_4	WS_5	WS_6	WS_7	WS_8	WS_9
1/24	1/24	1/24	2/24	2/24	1/24	1/24	1/24	2/24	12/24

Load distribution among the 10 WS of each homogeneous group

System parameters

$E[ram\_hit]$	0.5 msec.
$E[disk\_request]$	0.05 msec.
$E[http]$	1 msec.
$E[disk]$	10 msec.
$E[WS\_http]$	1 msec.
$E[WS\_disk\_request]$	0.05 msec.
$E[WS\_disk]$	10 msec.
Δ	100 msec.

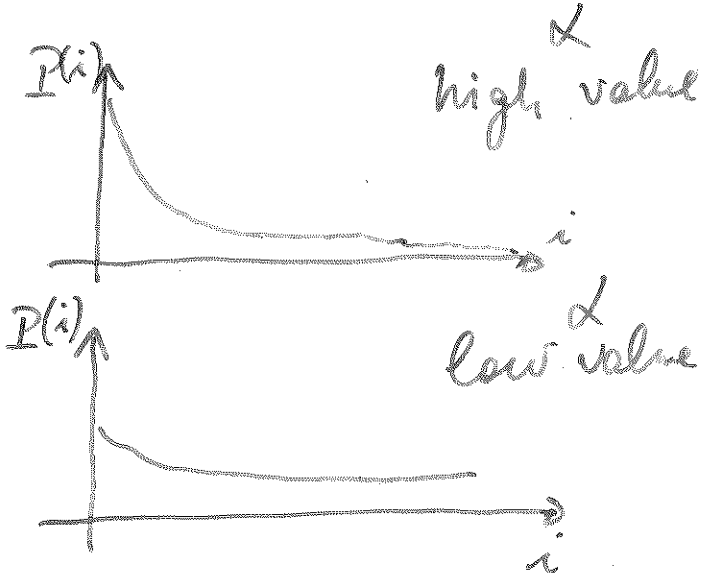
Other parameters

CACHE NODE

- RAM: 1 GB
- CACHEABLE OBJECT: 8K byte

$\Rightarrow$ ~ 130'000 objects in the RAM

\# OF OBJECT x WS:
15'000



$$\alpha = \begin{cases} 0.6 \simeq \text{University traces} \\ 1.4 \simeq \text{World Cup Web Site (2002)} \end{cases}$$

$$\text{Update rate}: \begin{cases} 1/15 \text{ min} \\ 1/24 \text{ hours} \end{cases}$$

Dynamic document: $\simeq$ 20%

Analysed assignment

Configuration 1:	WS_0-WS_4	are assigned to the first node of the couple
	WS_5-WS_9	are assigned to the second node
Configuration 2:	WS_0-WS_8	are assigned to the first node of the couple
	WS_9	is assigned to the second node
Configuration 3:	WS_0-WS_9	are assigned to the both nodes

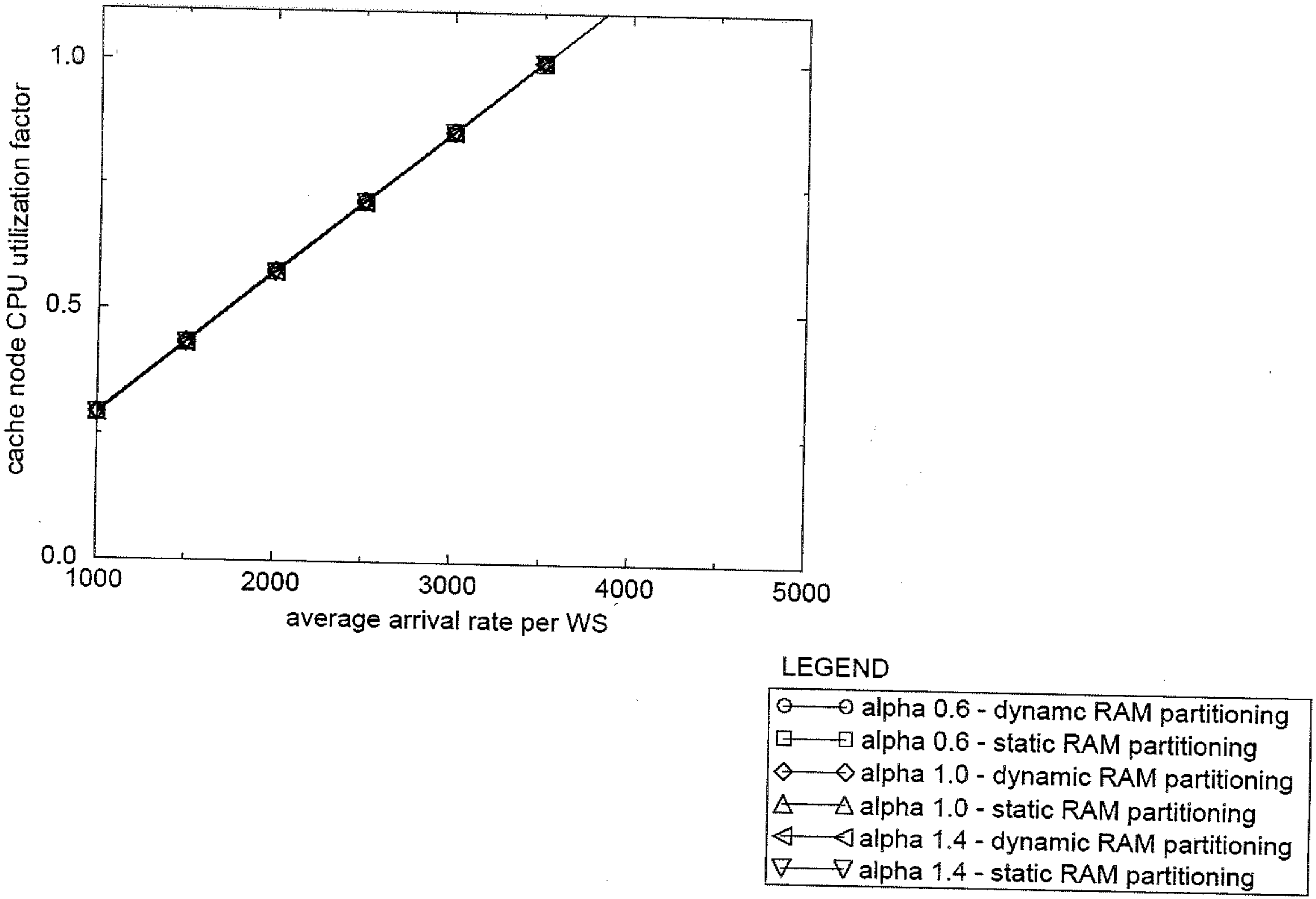
Analysed assignment (cont.)

Configuration 1: better hit ratio and balanced hit between the nodes, unbalanced load.

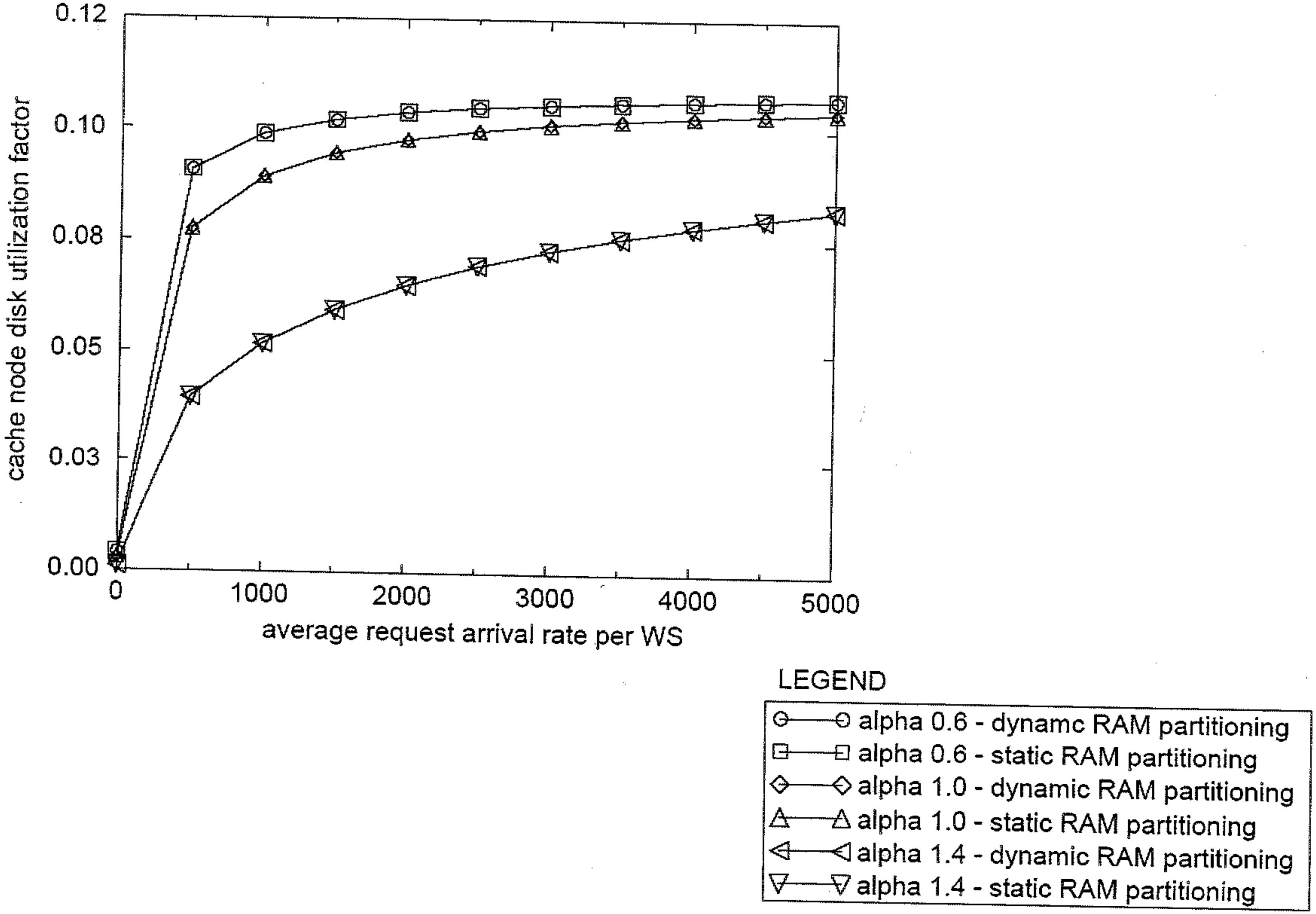
Configuration 2: balanced load between the nodes, bigger miss ratio in the first node.

Configuration 3: balanced load among the nodes, balanced hit ratio between the nodes.

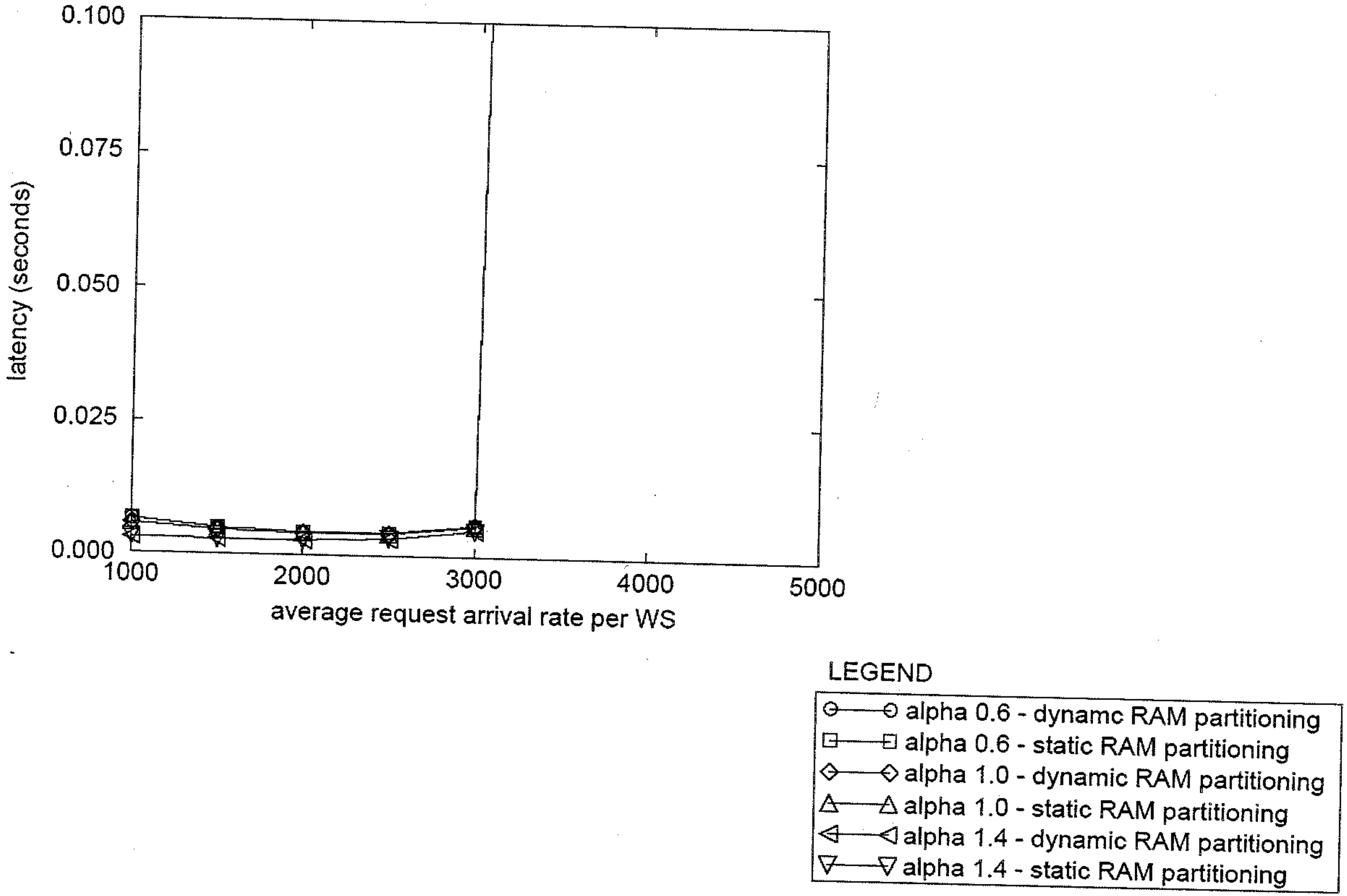
First configuration



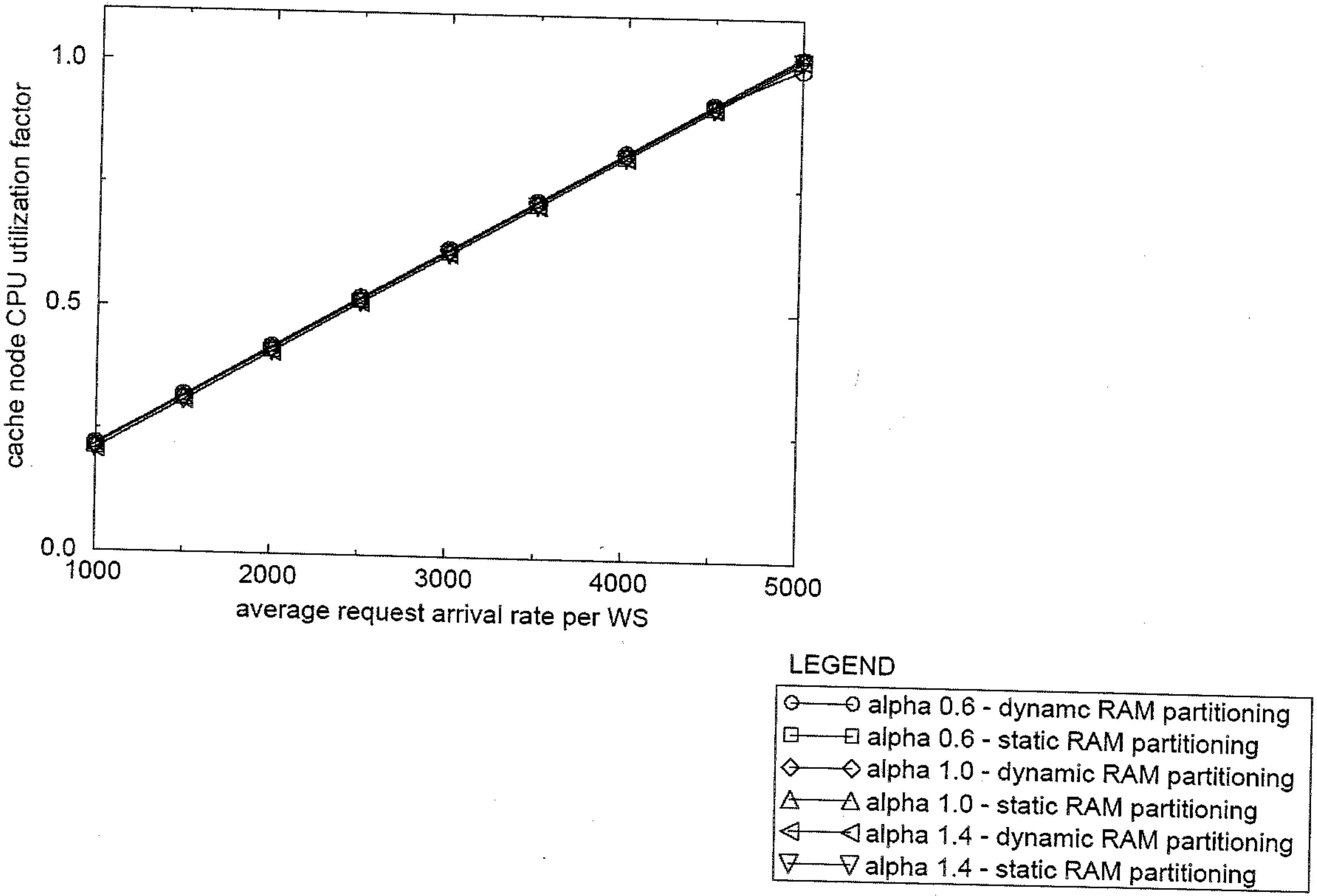
First configuration (cont.)



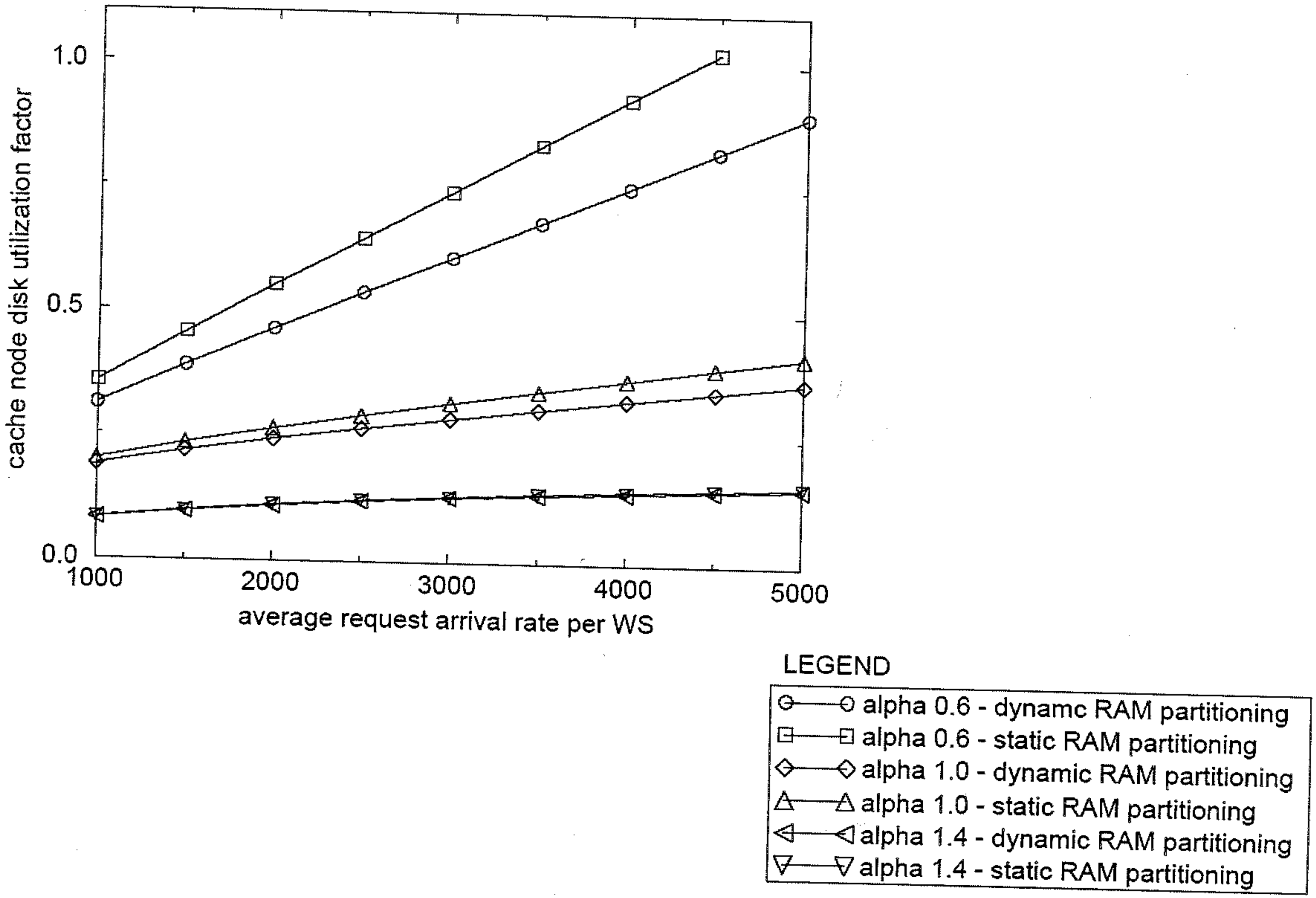
First configuration (cont.2)



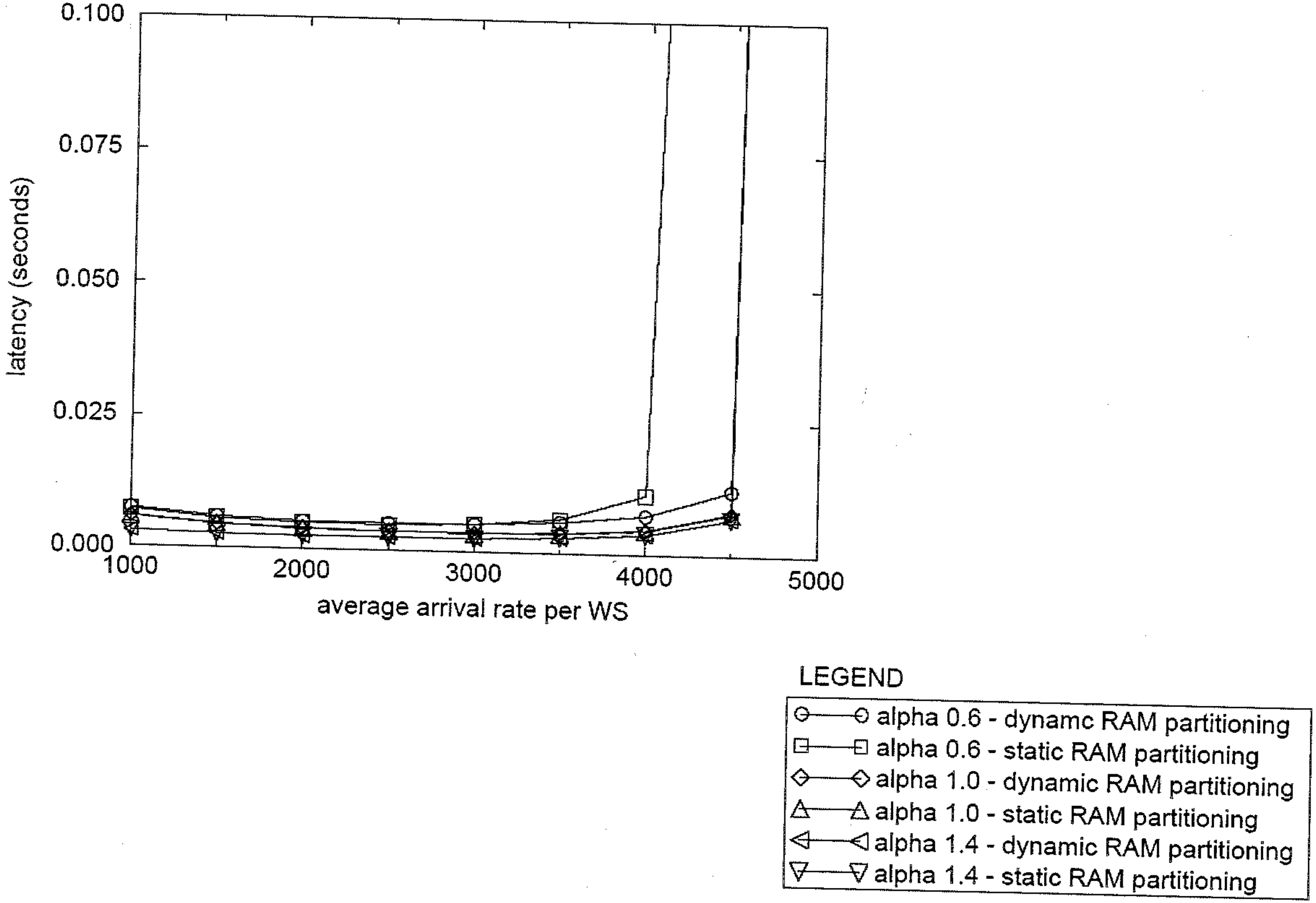
Second configuration



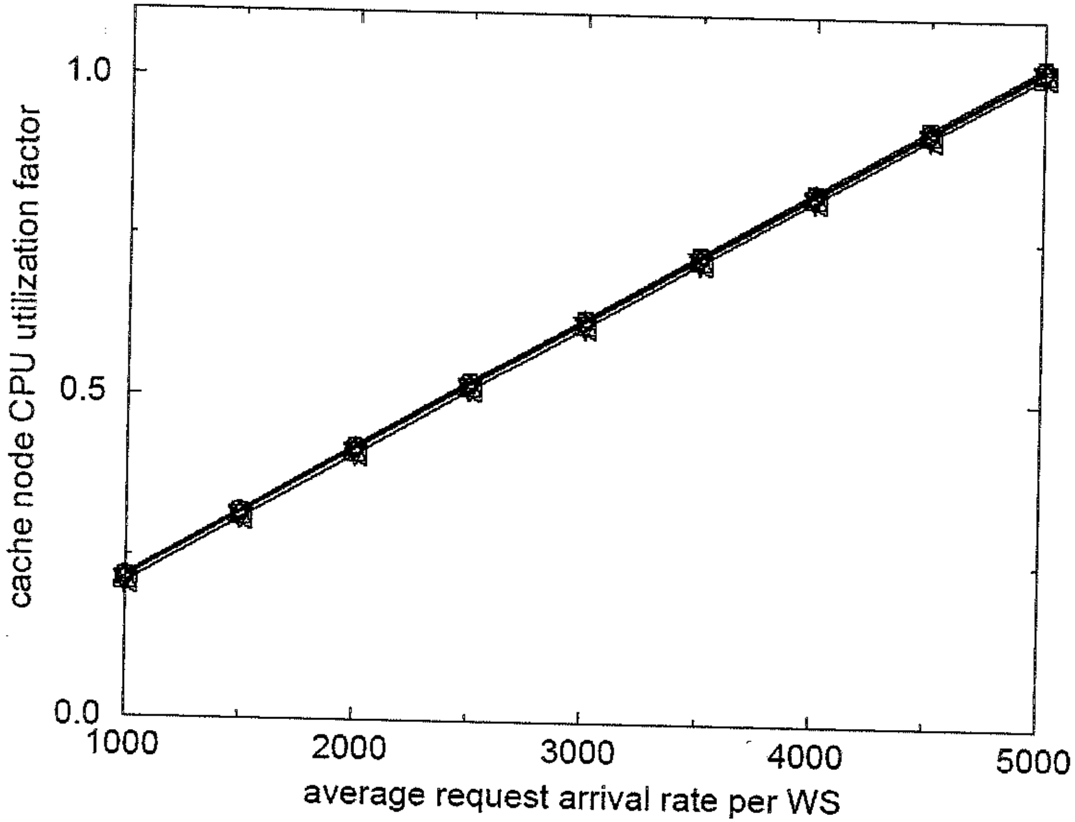
Second configuration (cont.)



Second configuration (cont.2)

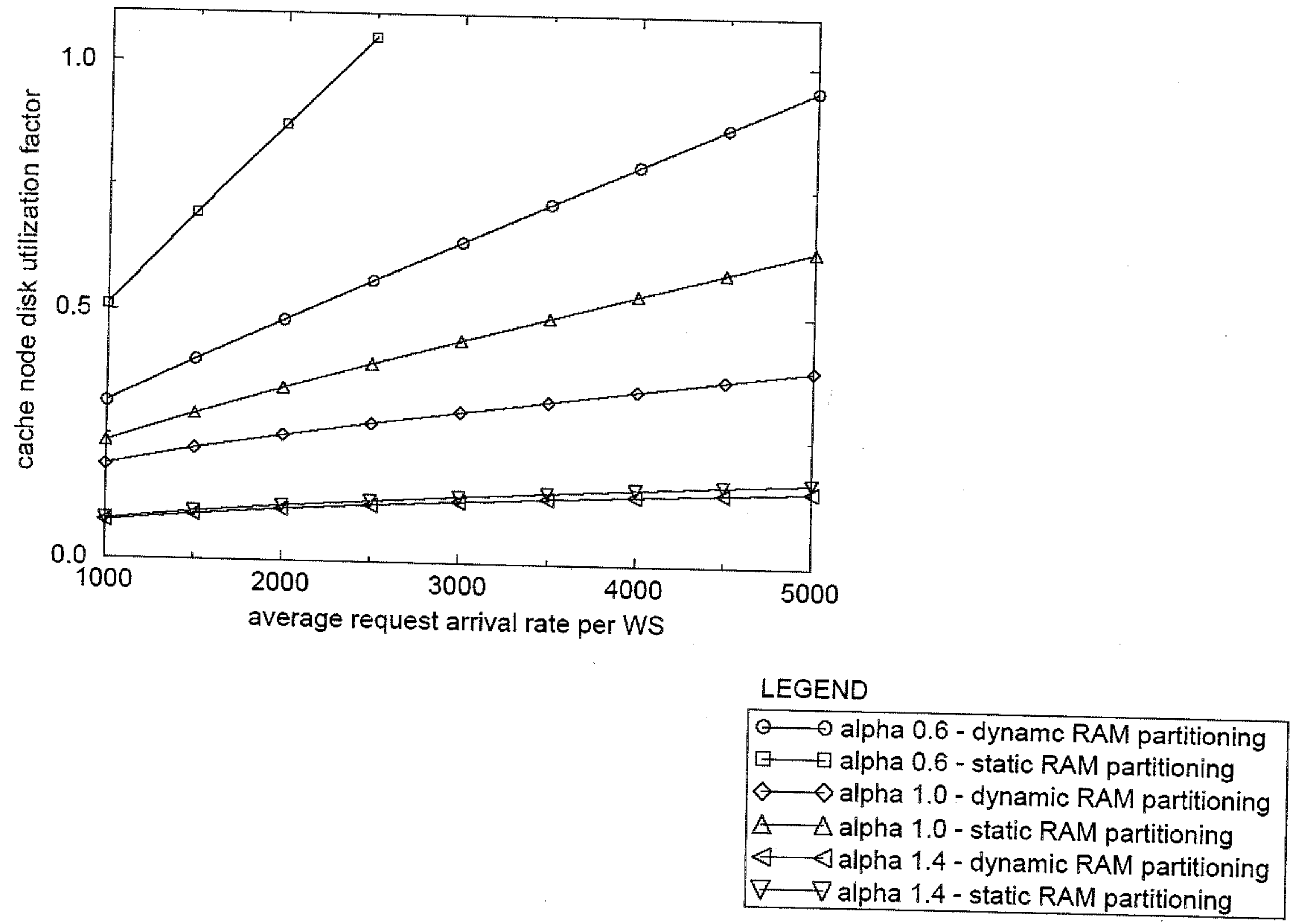


Third configuration

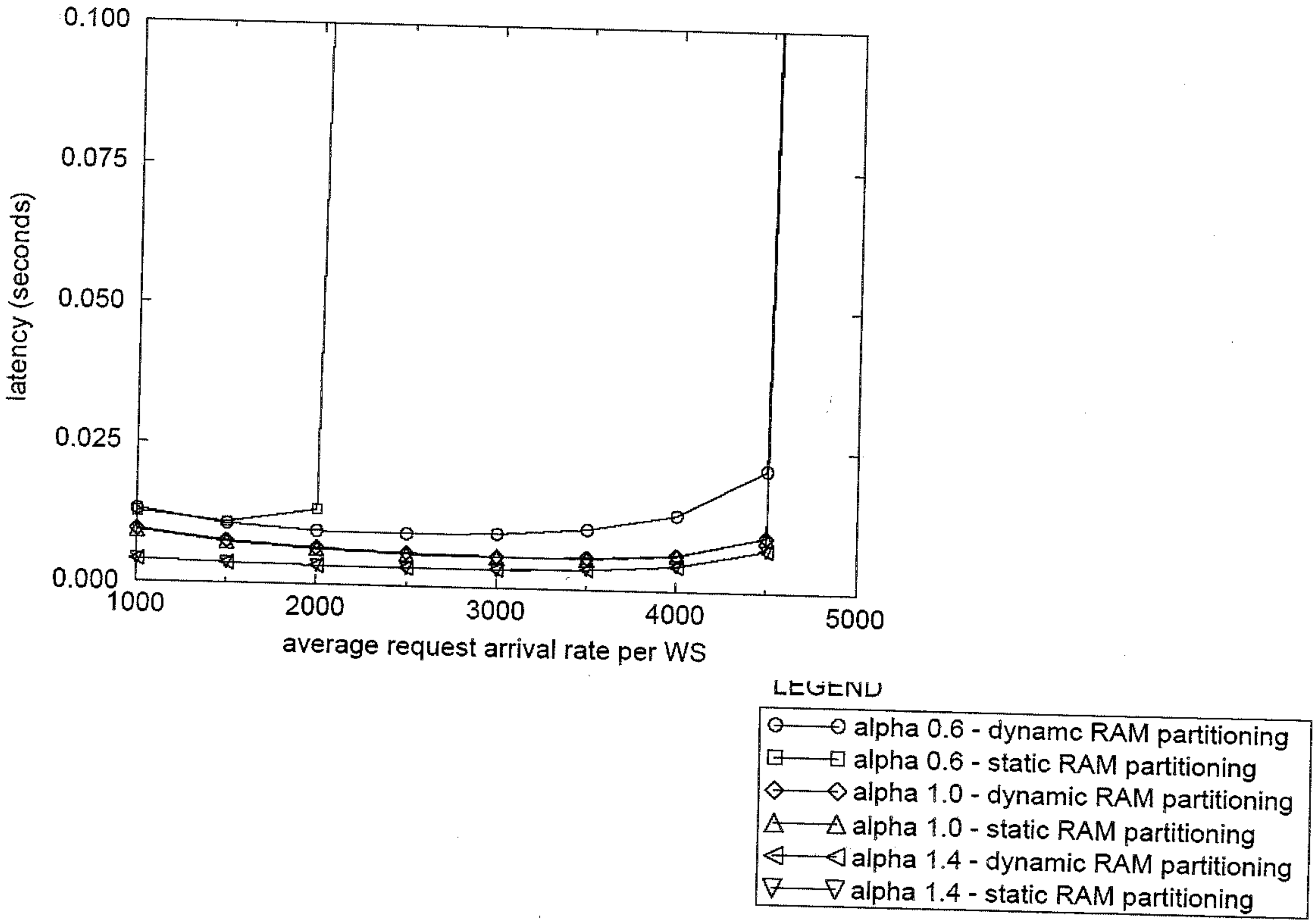


LEGEND

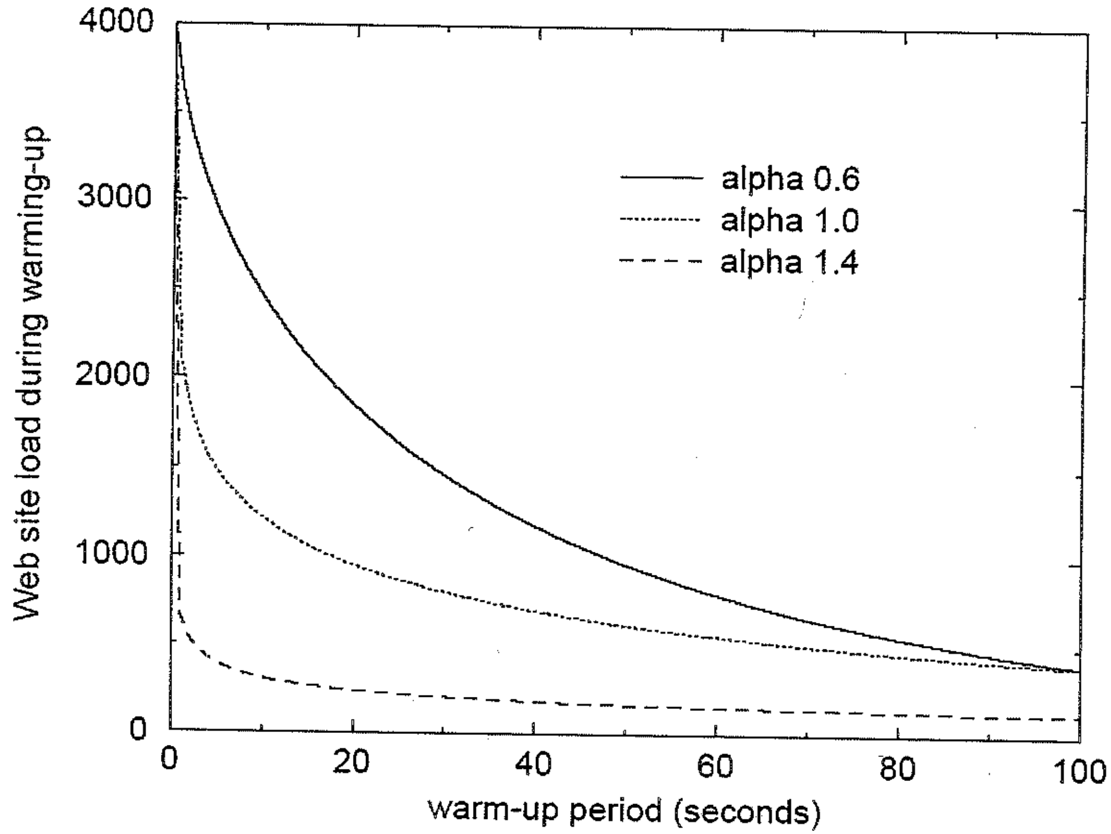
Third configuration (cont.)



Third configuration (cont.2)



Second configuration (warm-up period)



Performance conclusions

- **Configuration 1** has no disk problem but the unbalanced load generates CPU saturation.
- **Configuration 2** presents good steady state performance, but the disk can saturate. Moreover it can generate troubles to the WS in warm-up period.
- **Configuration 3** is a good compromise, but it can generate a high RAM miss ratio.

Performance conclusions (cont)
NO WINNER IN ALL CASES

- The third alternative is the best when hot documents dominate.
- The second is good for both high and moderate skew, but only if the warm-up problem is solved.