Integrated document caching and prefetching in storage hierarchies based on Markov-chain predictions

Achim Kraiss Gerhard Weikum
University of Saarland (Germany)
The VLDB Journal (1998) 7

Outline

- **Introduction**
- **Architecture**
- **Stochastic Model**
- **Integrated Migration Policy**
- **Implementation**
- **Experiment**
- **Conclusion**

Introduction

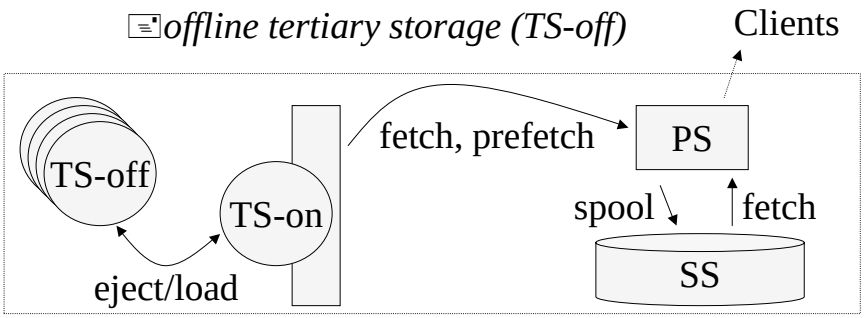
- **Background**
  - **multimedia document archives (volumes)**
  - **high latency of volume exchanges**
- **Motivations**
  - **document popularity/access pattern**
  - **good cache replacement policies**
- **Issues**
  - **cache hit rate**
    - *quantitatively assess the cache-worthiness*
    - *throttle the overly aggressive prefetching*

Introduction

- **Issues (Continued)**
  - **resource contention at the tertiary storage**
    - *reduce queuing delays of pending requests*
    - *minimize volume exchanges*
  - **resource contention at the secondary storage**
    - *write (migration) vs. read (cache hit)*
    - *the primary storage: faster cache space*
- **Goal**
  - **a unified approach to cache replacement and speculative prefetching based on a stochastic model for predicting document accesses**

Architecture

- **Storage hierarchy**
  - **primary storage (PS): cache/transfer buffer**
  - **secondary storage (SS): cache disk**
  - **tertiary storage (TS): optical-disk jukebox**
    - *online tertiary storage (TS-on)*
    - *offline tertiary storage (TS-off)*



Architecture

- **Upward data migration**
  - **fetch from TS-on into PS**
  - **fetch from SS into PS**
  - **prefetch from TS-on into PS**
  - **spool from PS onto SS**
  - **load from TS-off to TS-on**
- **Downward data migration**
  - **eject an online volume from TS drive**
- **Scenario**
  - **docs(PS) is not always a subset of docs(SS)**

Stochastic Model

- **Basics**
  - **S: set of active user sessions, D: document set**
  - **Markov chain model**
    - *interaction times between success requests of the same user are exponentially distributed (confirmed by Web server traces)*
  - **a continuous-time Markov chain (CTMC)**
    - *the probability of entering a state depends only on the current state*
    - *state residence time must be an exponentially distributed random variable*

Stochastic Model

- **CTMC**
  - **p_{ij}: transition probability from state i to j**
  - **H_i: mean residence time of state i**
  - **v_i (=1/H_i): state departure rate**
  - **v_{ij} (=p_{ij}*v_i): transition rate from state i to j**
    - *$p_{ij}(t)$: the probability that a session will be in state j at time t from now (given state i)*
- **Transformed CTMC**
  - **CTMC with uniform mean residence times**
    - *generated by a Poisson process with rate v (v=max(v_i) of the original CTMC)*

Stochastic Model

- **Mathematical foundations**
  - $\overline{p}_{ij} = \begin{cases} \frac{v_i}{v} * p_{ij}, & j \neq i \\ 1 - \frac{v_i}{v}, & j = i \end{cases}$, where $v = \{v_i | i = 1..N\}$
  - $\overline{p}_{ij}^{(m)} = \sum_{k=1}^{N} \overline{p}_{ik}^{(m-1)} p_{kj}$, with $\overline{p}_{ij}^{(0)} = \begin{cases} 1, \text{if } i = j \\ 0, \text{otherwise} \end{cases}$
  - $p_{ij}(t) = \sum_{m=0}^{\infty} e^{-vt} \frac{(vt)^m}{m!} * \overline{p}_{ij}^{(m)}$, $\forall$i, j and t > 0
  - $E_{ij}(t) = \frac{1}{v} * \sum_{n=1}^{\infty} e^{-vt} \frac{(vt)^n}{n!} * \sum_{m=0}^{n-1} \overline{p}_{ij}^{(m)}$

Stochastic Model

- **Single session**
  - **E[number of accesses to d_k in time t]**

    $= \sum_{j=1}^{N} v * E_{ij}(t) * \overline{p}_{jk}$
- **Multiple sessions**
  - **expected number of speculative requests**
  - **E[total number of accesses to d_k in time t]**

    $= N_{spec}(d_k, t) = \sum_{s \in S} \sum_{j=1}^{N} v * E_{d(s),j}(t) * \overline{p}_{jk}$
  - **approximation**
    - *dynamic arrival and termination*

Integrated Migration Policy

- **Metrics**
  - **near-term heat: $NH(d,t)=N_{spec}(d,t)$**
  - **near-term temperature (normalized)**
    - *$NT(d,t)=NH(d,t)/S(d)$*
  - **replacement cost: RC(d)**
  - **weight(d,t)=(NH(d,t)/S(d))*RC(d)**
    - *maintain a sorted list L of interesting documents containing the top m documents*
    - *prefetch d from L into PS or SS if and only if d is not cached and its weight exceeds the maximum weight among replacement victims*

Integrated Migration Policy

- **Scenario**
  - **overly aggressive prefetching may lead to contention at the TS drives**
- **Two more metrics**
  - **benefit**
    - *the aggregated savings in the response time*
  - **penalty**
    - *the aggregated delays of pending requests*
  - **the prefetching request is initiated if and only if its benefit exceeds its penalty**
- **Example**

Integrated Migration Policy

- **Decision process**
  - **rank documents by their weights**
    - *identify speculative prefetching candidates and insert them into the queues of the TS*
  - **prefetch document d from TS-on**
    - *select the lowest ranked documents of the target level as replacement victims RV*
    - *d/RV weight comparison*
    - *benefit/penalty comparison*
  - **write d from PS to SS**
    - *repeat stage 2*

Implementation

- **Bookkeeping data**
  - **keep moving-average statistics for all state transition probabilities of CTMC and all state residence times**
  - **monitor the state-changes of any session**
    - *incrementally update parameters* $E_{ij}(t)$
    - *terminate the chain traversal by a threshold*
  - **ranked lists of data <id, size, weight>**
    - l_{DOC}*: all documents with nonzero weights*
    - l_{PS}*: all documents cached in PS*
    - l_{SS}*: all documents cached in SS*

Experiment

- **Markov chain migration policies**
  - **McMin**
    - *weight(d,t)=NH(d,t), RC(d)=1*
    - *eagerly prefetching based only on weights*
  - **McMin+**
    - *document-specific replacement costs*
    - *benefit/penalty comparison*
  - **McMin-**
    - *just a CTMC-based cache replacement*
- **Temperature-based migration policies**

Experiment

- **Simulation**
  - **four synthetic workloads which differ in their session arrival rates and the distribution of mean residence times**
    - *LOW_SLOW, LOW_FAST, ...*
- **Results**
  - **MRT: McMin < TEMP+ (LOW_SLOW)**
  - **MRT: McMin+ < McMin (large caches)**
  - **miss rate: McMin < TEMP+($HR_{TS\text{-}off}$)**
  - **space overhead: 10MB/23GB**
  - **CPU consumption: 200ms per session step**

Conclusion

- **Contributions**
  - **incorporate document-specific client interaction time between successive requests**
  - **reconcile the induced access patterns of all active client sessions into a global prediction**
- **Features**
  - **better than the stationary-probability model**
  - **much more bookkeeping overheads**
- **Applications**
  - **prefetching and caching for Web servers**
  - **data hoarding in mobile system**