CS 672
Workload Characterization

Dr. Daniel A. Menascé

http://www.cs.gmu.edu/faculty/menasce.html

Department of Computer Science

George Mason University

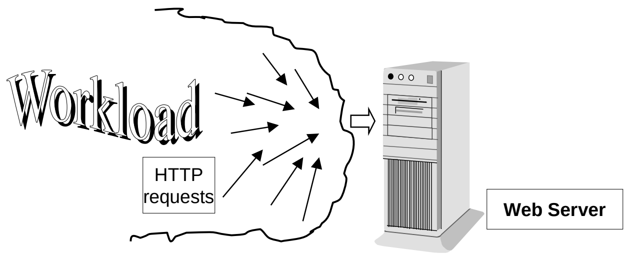
 1

What is Workload Characterization?

 2

Workload

- The workload of a system can be defined as the set of all inputs that the system receives from its environment during any given period of time.



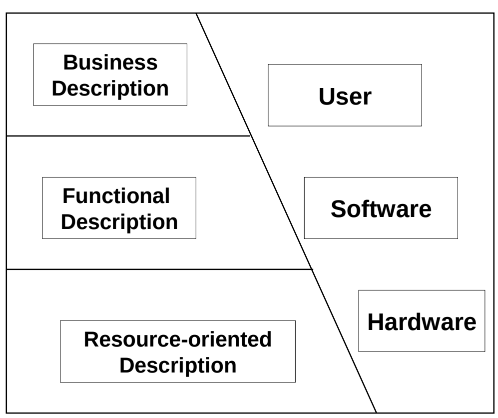
Workload Characterization: concepts and ideas

- <u>Basic component</u> of a workload refers to a generic unit of work that arrives at the system from external sources.
  - Transaction,
  - interactive command,
  - process,
  - HTTP request, and
  - depends on the nature of service provided

Workload Characterization: concepts and ideas

- Workload characterization
  - **workload model is a representation that mimics the workload under study.**
- Workload models can be used for:
  - the selection of systems
  - performance tuning
  - capacity planning

Workload Description



Workload Description

- Business characterization: a user-oriented description that describes the load in terms such as number of employees, invoices per customer, etc.
- Functional characterization: describes programs, commands and requests that make up the workload
- Resource-oriented characterization: describes the consumption of system resources by the workload, such as processor time, disk operations, memory, etc.

 7

A Web Server Example

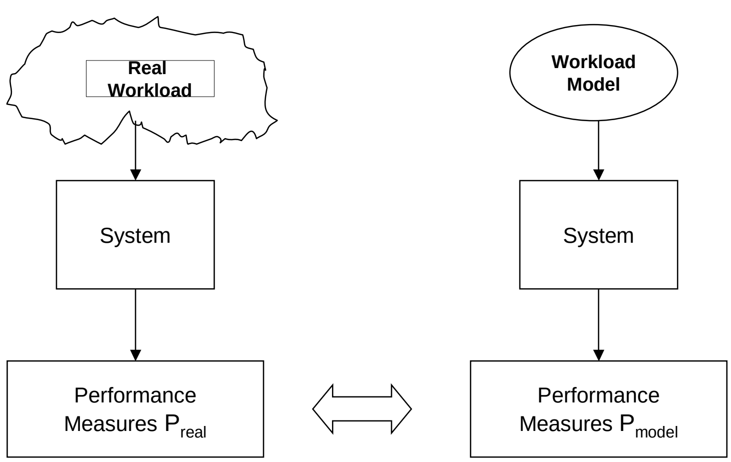
- The pair **(CPU time, I/O time)** characterizes the execution of a request at the server.
- Our basic workload: 10 HTTP requests
- First case: only one document size (15KB)
- 10 executions ---> (0.013 sec, 0.09 sec)
- More realistic workload: documents have different sizes.

 8

Execution of HTTP Requests (sec)

Request No.	CPU time (sec)	I/O time (sec)	Elapsed time (sec)
1	0.0095	0.0400	0.0710
2	0.0130	0.1100	0.1450
3	0.0155	0.1200	0.1560
4	0.0088	0.0400	0.0650
5	0.0111	0.0900	0.1140
6	0.0171	0.1400	0.1630
7	0.2170	1.2000	4.3800
8	0.0129	0.1200	0.1510
9	0.0091	0.0500	0.0630
10	0.0017	0.1400	0.1890
Average	0.03157	0.205	0.5497

Representativeness of a Workload Model



A Refinement in the Workload Model

- The average response time of 0.55 sec does not reflect the behavior of the actual server.
- Due to the heterogeneity of the its components, it is difficult to view the workload as a single collection of requests.
- Three classes
  - small documents
  - medium documents
  - large documents

Execution of HTTP Requests (sec)

Request No.	CPU time (sec)	I/O time (sec)	Elapsed time (sec)
1 small	0.0095	0.0400	0.0710
2 medium	0.0130	0.1100	0.1450
3 medium	0.0155	0.1200	0.1560
4 small	0.0088	0.0400	0.0650
5 medium	0.0111	0.0900	0.1140
6 medium	0.0171	0.1400	0.1630
7 large	0.2170	1.2000	4.3800
8 medium	0.0129	0.1200	0.1510
9 small	0.0091	0.0500	0.0630
10 medium	0.0017	0.1400	0.1890

Three-Class Characterization

Type	CPU time (sec)	I/O time (sec)	No of omponents
Small Docs.	0.0091	0.04	3
Medium Docs.	0.0144	0.12	6
Large Docs.	0.2170	1.20	1
Total	0.331	2.05	10

Workload Models

- A model should be representative and compact.
- Natural models are constructed either using basic components of the real workload or using traces of the execution of real workload.
- Artificial models do not use any basic component of the real workload.
  - Executable models (e.g.: synthetic programs, artificial benchmarks, etc)
  - **Non-executable models, that are described by a set of parameter values that reproduce the same resource usage of the real workload.**

Workload Models

- The basic inputs to analytical models are parameters that describe the service centers (i.e., hardware and software resources) and the customers (e.g. requests and transactions)
  - component (e.g., transactions) interarrival times;
  - service demands
  - execution mix (e.g., levels of multiprogramming)

A Workload Characterization Methodology

- ☐Choice of an analysis standpoint
- ☐Identification of the basic component
- ☐Choice of the characterizing parameters
- ☐Data collection
- ☐Partitioning the workload
- ☐Calculating the class parameters

Selection of characterizing parameters

- Each workload component is characterized by two groups of information:
- Workload intensity
  - arrival rate
  - number of clients and think time
  - number of processes or threads in execution simultaneously
- Service demands (D_{i1}, D_{i2}, ... D_{iK}), where D_{ij} is the service demand of component i at resource j.

Data Collection

- This step assigns values to each component of the model.
  - Identify the time windows that define the measurement sessions.
  - Monitor and measure the system activities during the defined time windows.
  - From the collected data, assign values to each characterizing parameters of every component of the workload.

Partitioning the workload

- Motivation: real workloads can be viewed as a collection of heterogeneous components.
- Partitioning techniques divide the workload into a series of classes such that their populations are composed of quite homogeneous components.
- What attributes can be used for partitioning a workload into classes of similar components?

 19

Partitioning the Workload

- Resource usage
- Applications
- Objects
- Geographical orientation
- Functional
- Organizational units
- Mode

 20

Workload Partitioning: Resource Usage

Transaction Classes	Frequency	Maximum CPU time (msec)	Maximum I/O time (msec)
Trivial	40%	8	120
Light	30%	20	300
Medium	20%	100	700
Heavy	10%	900	1200

Workload Partitioning: Internet Applications

Application Classes	KB Transmitted
WWW	4,216
ftp	378
telnet	97
Mbone	595
Others	63

Workload Partitioning: Document Types

Document Class	Percentage of Access (%)
HTML (html file types)	**30**
Images (e.g., gif or jpeg)	**40**
Sound (e.g., au or wav)	**4.5**
Video (e.g., mpeg, avi or mov)	**7.3**
Dynamic (e.g., cgi or perl)	**12.0**
Formatted (e.g., ps, dvi or doc)	**5.4**
Others	**0.8**

Workload Partitioning: Geographical Orientation

Classes	Percentage of Total Requests
East Coast	**32**
West Coast	**38**
Midwest	**20**
Others	**10**

Calculating the class parameters

- How should one calculate the parameter values that represent a class of components?
  - Averaging: when a class consists of homogeneous components concerning service demands, an average of the parameter values of all components may be used.
  - Clustering of workloads is a process in which a large number of components are grouped into clusters of similar components.

Calculating Class Parameters

- Homogeneous Workload:
  - compute arithmetic mean
  - Workload: {(Di1, Di2, ..., DiK) | i = 1, ..., p}
  - Workload Charaterization:
    - (D1, D2, ..., DK) where
    - $Dj = 1/p \sum_{i=1}^{p} Dij$

Calculating Class Parameters

- Heterogeneous Workload:
  - use clustering analysis to determine groups of "similar" workloads.
  - Use averaging within each group.
  - Clustering analysis algorithms: minimal spanning tree and k-means.

Parameter Transformation

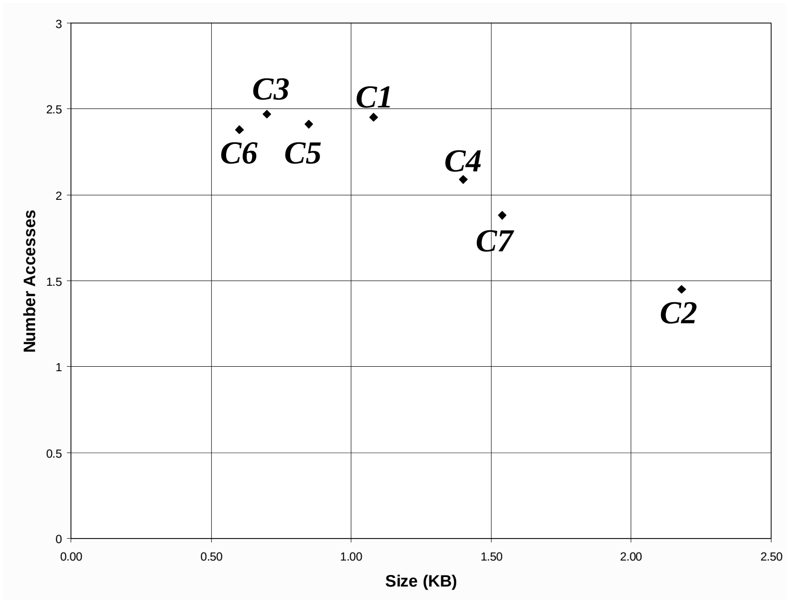
- Preventing extreme values of parameters from distorting distribution use linear transformation:
- Dt = (measured D - minimum{Di}) / (maximum{Di} - minimum{Di})

Workload Sample

Document	Size (KB)	No. Accesses
1	12	281
2	150	28
3	5	293
4	25	123
5	7	259
6	4	241
7	35	75

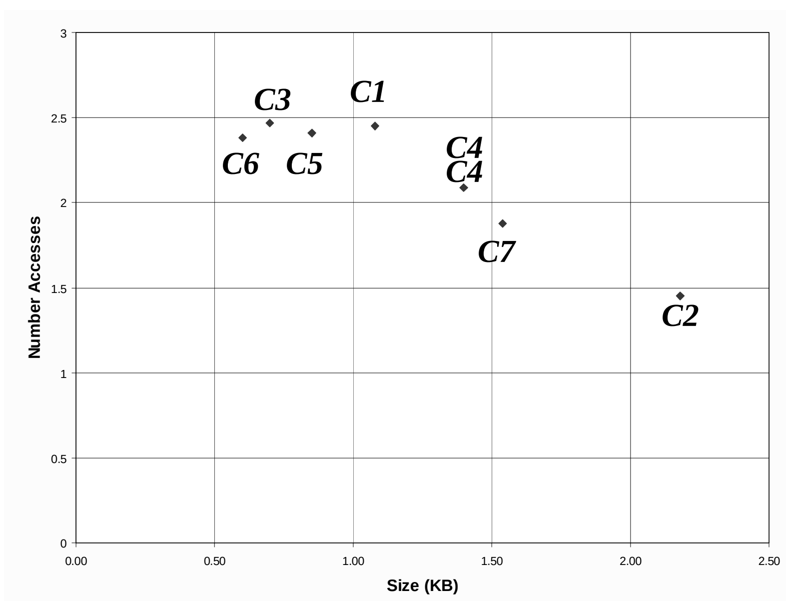
Workload Sample: logarithmic transformation of parameters

Document	Size (KB)	No. Accesses
1	1.08	2.45
2	2.18	1.45
3	0.70	2.47
4	1.40	2.09
5	0.85	2.41
6	0.60	2.38
7	1.54	1.88



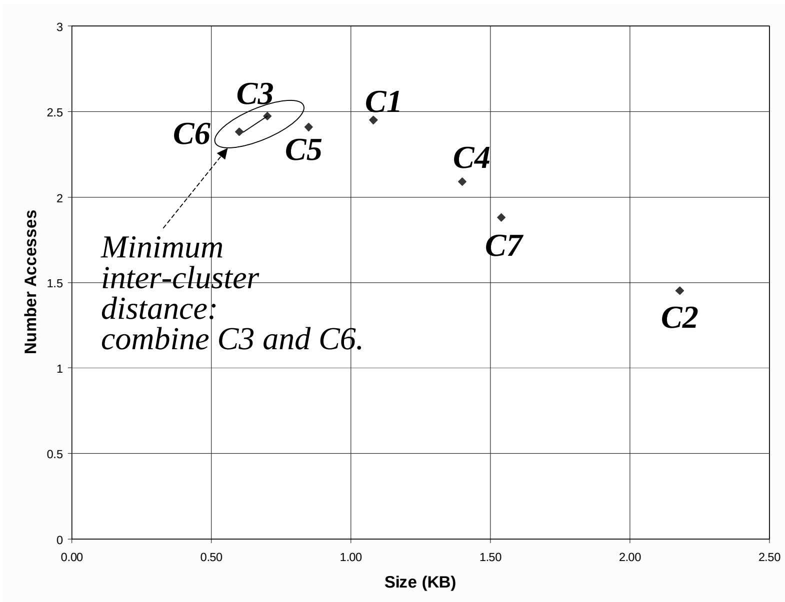
31

Minimal Spanning Tree Example



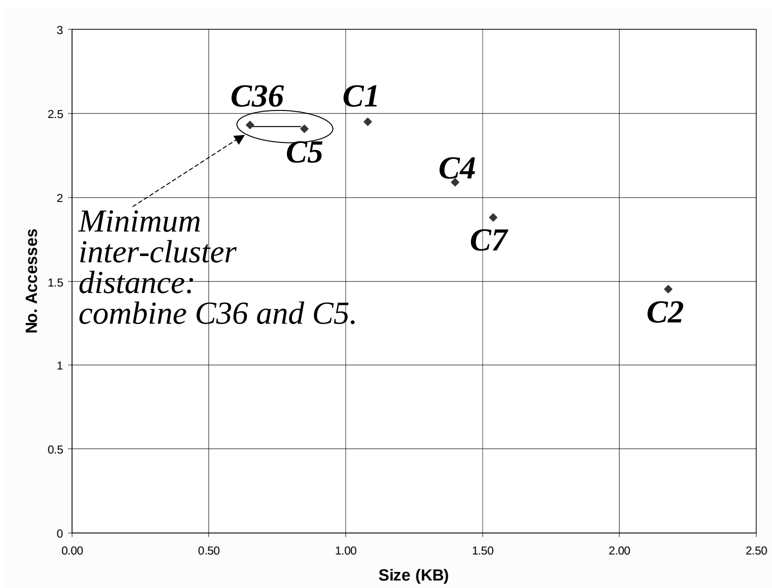
32

Minimal Spanning Tree Example



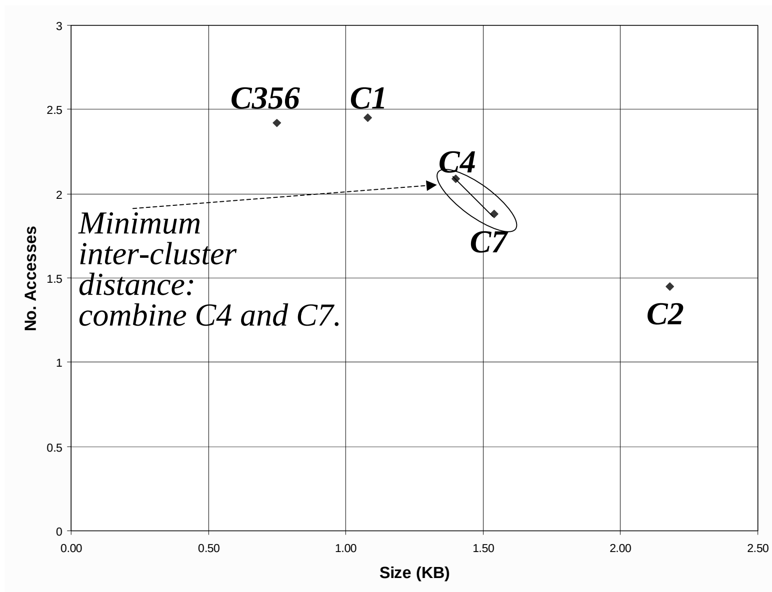
33

Minimal Spanning Tree Example



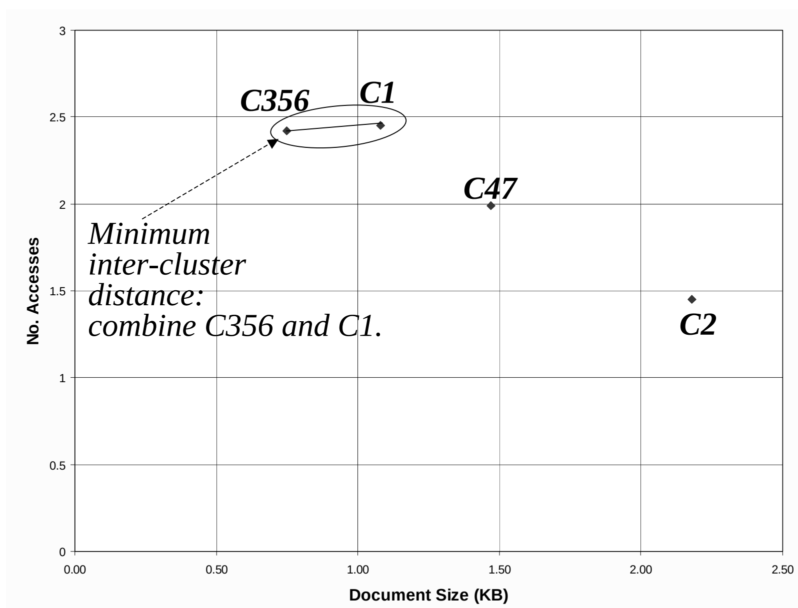
34

Minimal Spanning Tree Example



35

Minimal Spanning Tree Example



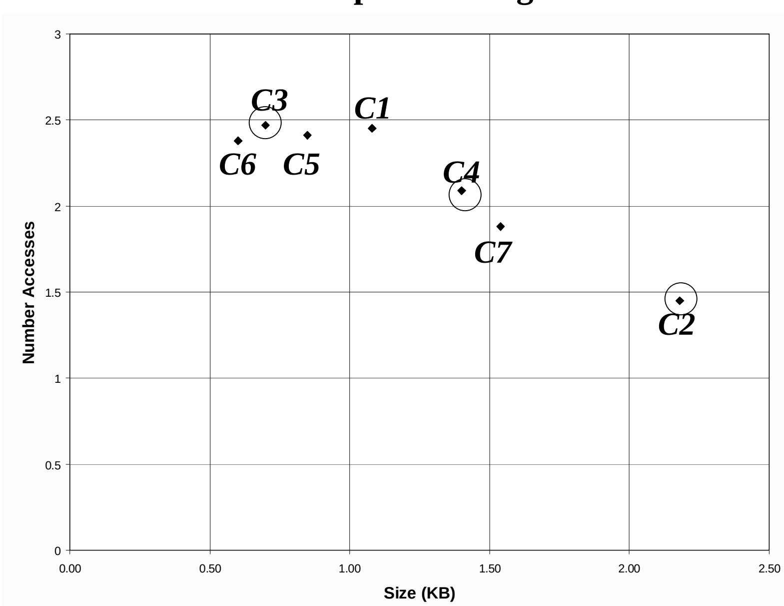
36

Result of Workload Characterization

Type	Class	Size (KB)	No. Accesses	No. Components
Small	C1356	8.19	271.51	4
Medium	C47	29.58	96.05	2
Large	C2	150.00	28.00	1

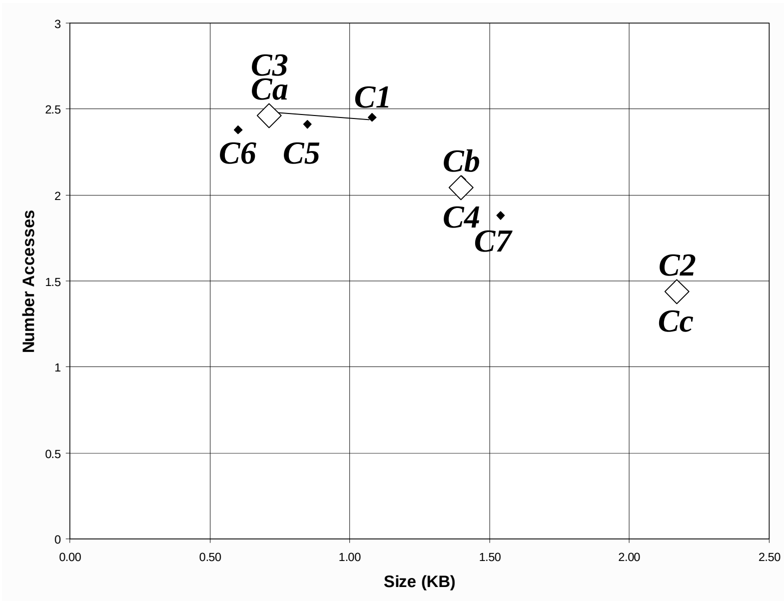
37

K-means Example: starting allocation



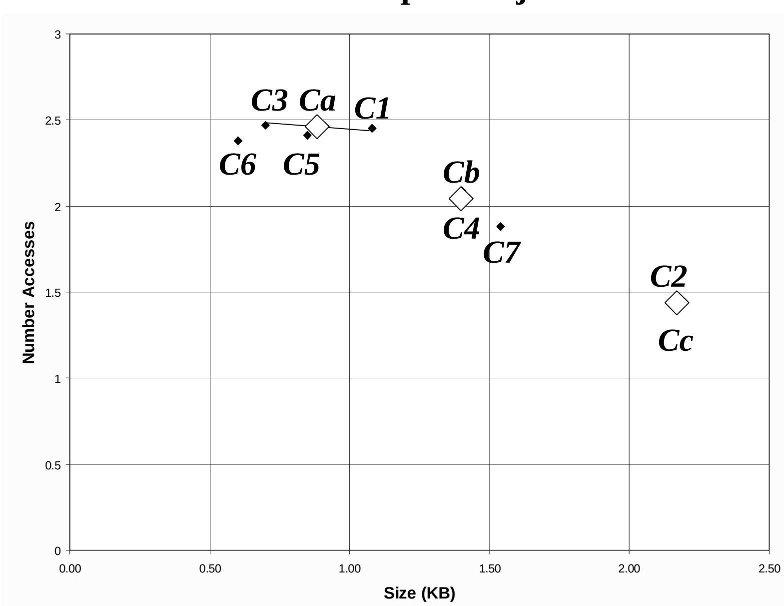
38

K-means Example: starting allocation



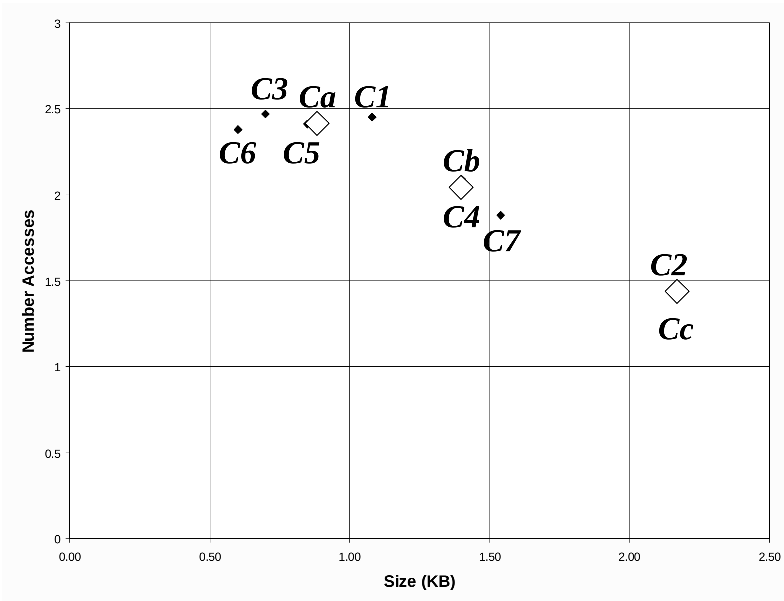
39

K-means Example: C1 joins Ca.

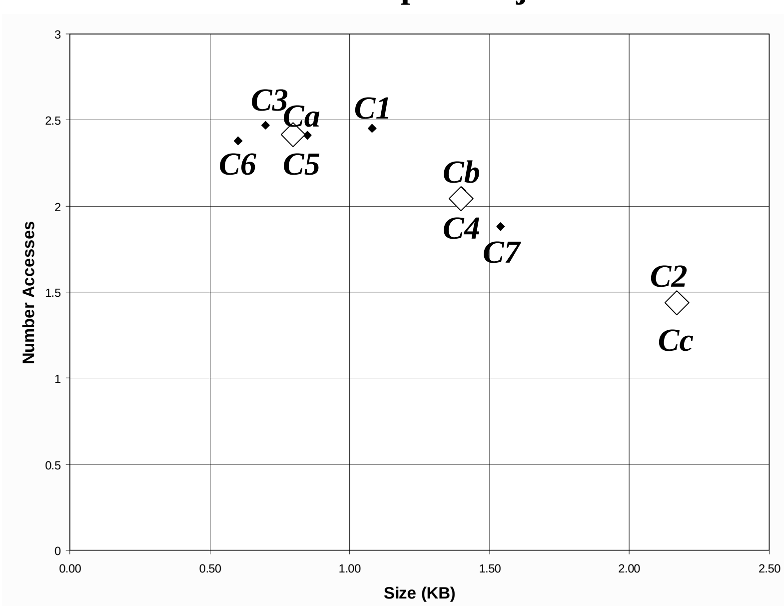


40

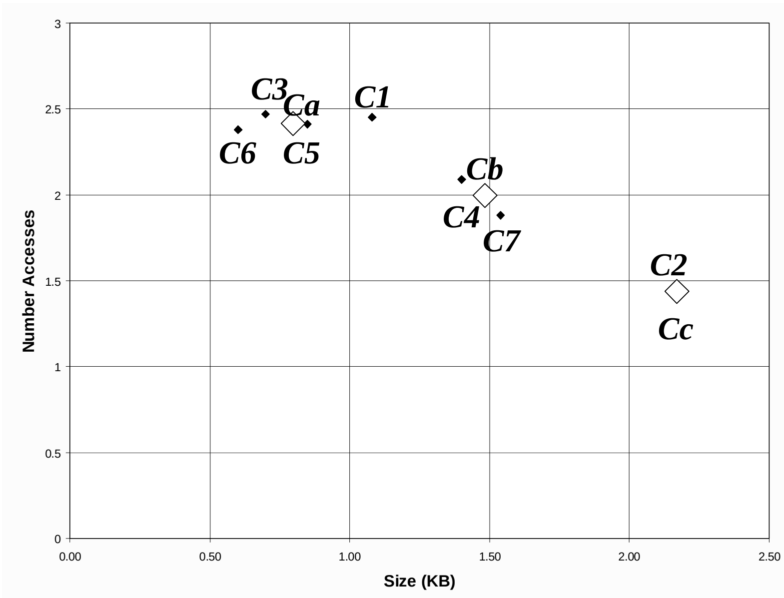
K-means Example: C5 joins Ca.



K-means Example: C6 joins Ca.



K-means Example: C7 joins Cb.



Novel Features in the WWW

- The Web exhibits extreme variability in workload characteristics:
  - Web document sizes vary in the range of 10^3 to 10^6 bytes
  - Access patterns in the Web vary tremendously. Load spikes of 8 to 10 times the average.

Novel Features in the WWW

- The Web exhibits extreme variability in workload characteristics:
  - Web document sizes vary in the range of 10^3 to 10^6 bytes
  - Access patterns in the Web vary tremendously. Load spikes of 8 to 10 times the average.
- Web traffic exhibits a bursty behavior
  - Traffic is bursty across several time scales.
  - It is difficulty to size server capacity and bandwidth to support demand created by load spikes.

 45

Types of Web Requests

- GET of Static HTML requests
- Execution of application at the server:
  - CGI scripts (e.g., to process HTML forms)
    - A new process is started for each request.
    - Stateless application.
  - Server APIs (e.g., NSAPI, ISAPI)
    - Application code is loaded and executed in the same context as the server.
    - Poor security and no isolation.
  - FastCGI
    - Web server and application communicate via light weight TCP or local IPC.
    - Application can be persistent and statefull

 46

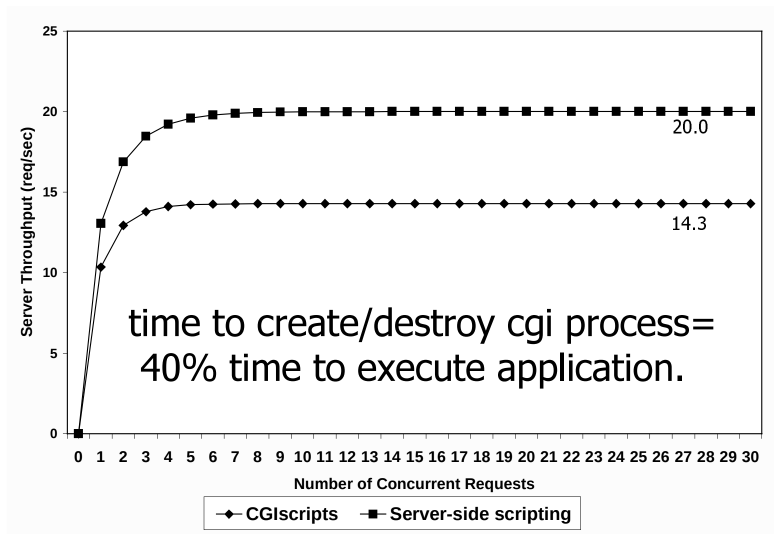
Types of Web Requests (cont'd)

- Execution of application at the server (cont'd):
  - Server-side scripting
    - Server interprets scripts or programs embedded in pages before returning them to the client:
      - MS Active Server Pages (ASP) permits the use of JavaScript and VBScript combined with ActiveX controls written in any programming language.
      - Netscape's LiveWire permits the use of server-side JavaScript.

Types of Web Requests (cont'd)

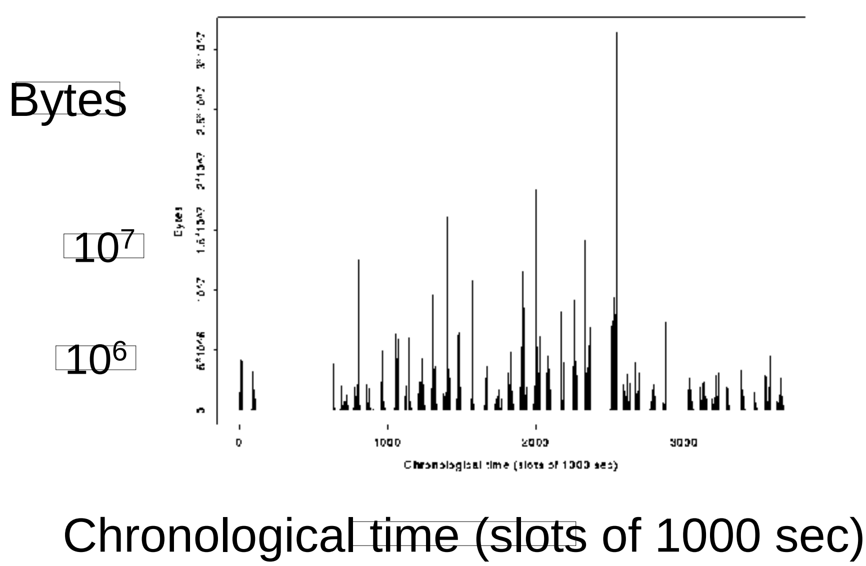
- Execution of application at the client:
  - Client-side scripting (e.g., JavaScript)
  - Download of applications from server for execution at the client:
    - Java
      - Platform independent
      - Limited access to client resources
    - MS ActiveX Controls
      - For MS Windows environments only.
      - Unrestricted access to PC resources.

CGI scripts vs. server-side scripts: performance impact



 49

WWW Traffic Bursts



 50

Burstiness and Throughput

- Burstiness factor (b): fraction of time during which the instantaneous arrival rate exceeds the average arrival rate.

 51

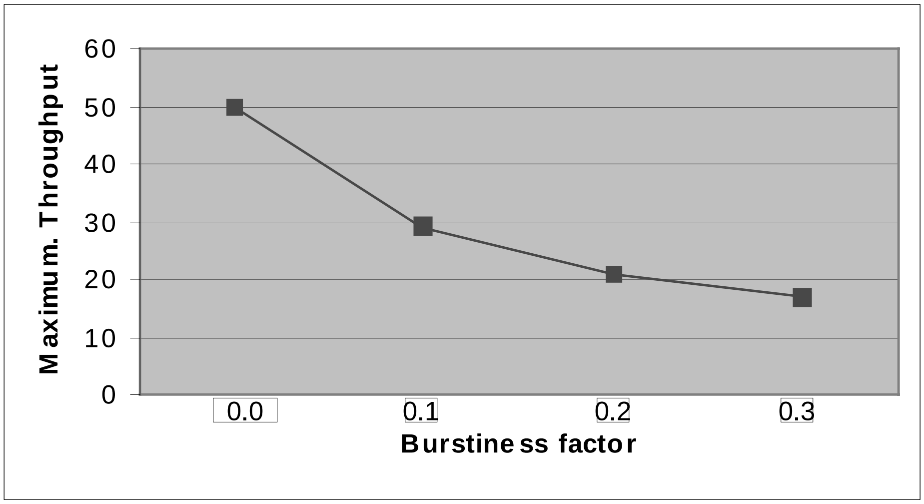
Burstiness and Throughput

- Burstiness factor (b): fraction of time during which the instantaneous arrival rate exceeds the average arrival rate.

- The site throughput decreases with the burstiness of the workload.

 52

The Impact of Burstiness



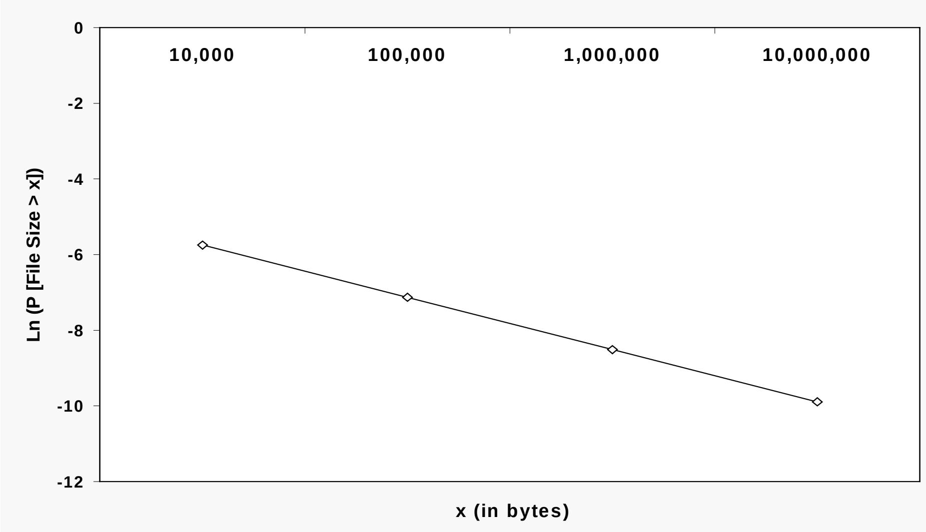
 53

Web Workload Characterization

- The distribution of the size of returned files is heavy tailed:
  - Most files are small but there is a non-negligible probability of returned files being large (e.g., images, video, sound).
  - Prob [returned file size > x] = k / x^{α} for large values of x and $0.4 < \alpha < 0.63$ (Pareto distribution).

 54

Web Workload Characterization: Heavy Tailed Distribution



 55

Incorporating New Phenomena in the Workload Characterization

Accounting for Heavy Tails in the Model

- Due to the large variability of the size of documents, average results for the whole population would have very little statistical meaning.
- Categorizing the requests into a number of classes, defined by ranges of document sizes, improves the accuracy and significance of performance metrics.
- Multiclass queuing network models, with classes associated with requests for docs of different size.

56

Accounting for Heavy Tails: an example

- The HTTP LOG of a Web server was analyzed during 1 hour. A total of 21,600 requests were successfully processed during the interval.
- Let us use a multiclass model to represent the server.
- There are 5 classes in the model, each corresponding to the 5 file size ranges.

57

Accounting for Heavy Tails: an example

- File Size Distributions.

Class	File Size Range (KB)	Percent of Requests
1	Size < 5	25
2	$5 \le$ size ≤ 50	40
3	$50 \le$ size ≤ 100	20
4	$100 \le$ size ≤ 500	10
5	size ≥ 500	5

58

Accounting for Heavy Tails: an example

- The arrival rate for each class r is a fraction of the overall arrival rate λ = 21,600/3,600 = 6 requests/sec.
  - $\lambda_1 = 6 \times 0.25 = 1.5$ req./sec
  - $\lambda_2 = 6 \times 0.40 = 2.4$ req./sec
  - $\lambda_3 = 6 \times 0.20 = 1.2$ req./sec
  - $\lambda_4 = 6 \times 0.10 = 0.6$ req./sec
  - $\lambda_5 = 6 \times 0.05 = 0.3$ req./sec

Web Workload Characterization

- Popularity:
  - Zipf's Law: the number of references, P, to a file tends to be inversely proportional to its rank r:

$$P = k/r$$

- The second most popular file gets half the number of references of the most popular one.

Web Workload Characterization

- Popularity:
  - Zipf's Law: the number of references, P, to a file tends to be inversely proportional to its rank r:

$$P = k/r$$

- The n-th most popular file gets 1/n of the number of references of the most popular one.

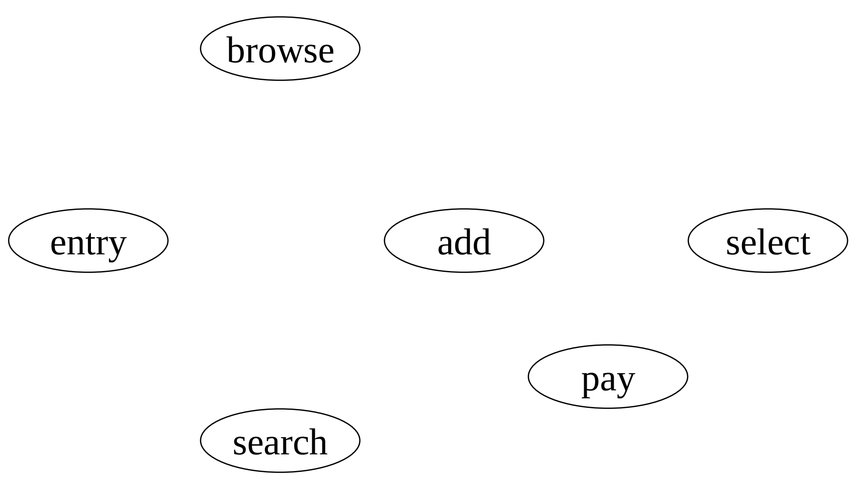
Web Workload Characterization: Zipf Law Example

File	Popularity	% Accesses
A	1	40.8%
B	2	20.4%
C	3	13.6%
D	4	10.2%
E	5	8.2%
F	6	6.8%

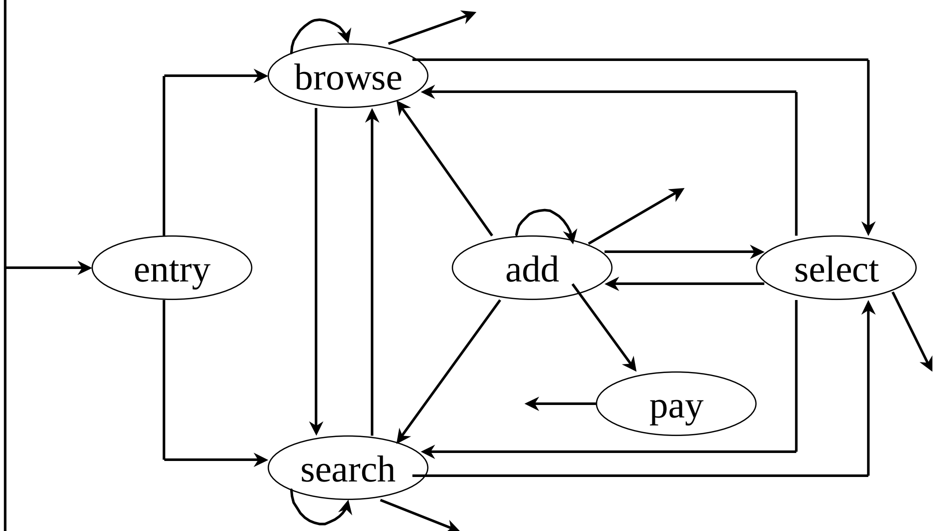
Workload Characterization for E-commerce

- Customers interact with the site through sessions, which are sequences of interrelated requests.
- Sessions can be characterized by a <u>Customer Behavior Model Graph (CBMG)</u>

Customer Behavior Model Graph (CBMG)

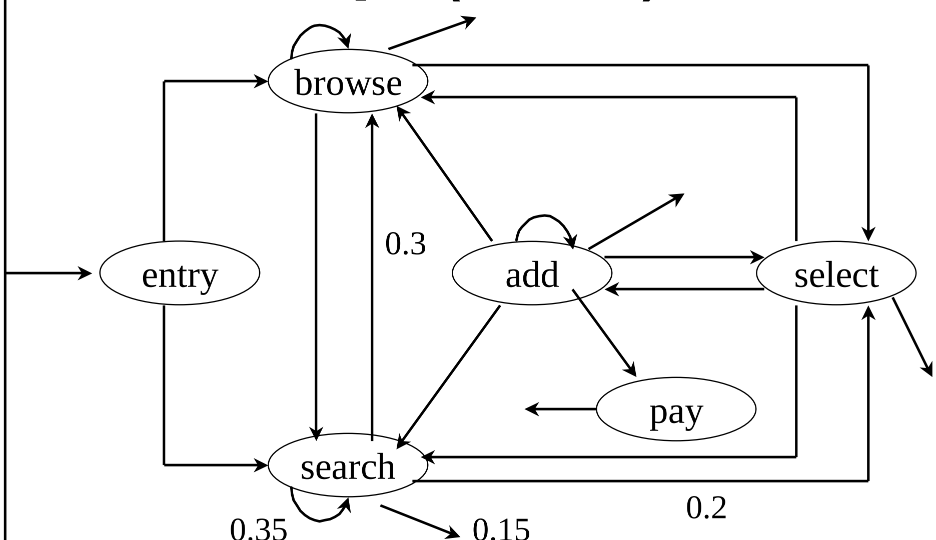


Customer Behavior Model Graph (CBMG)



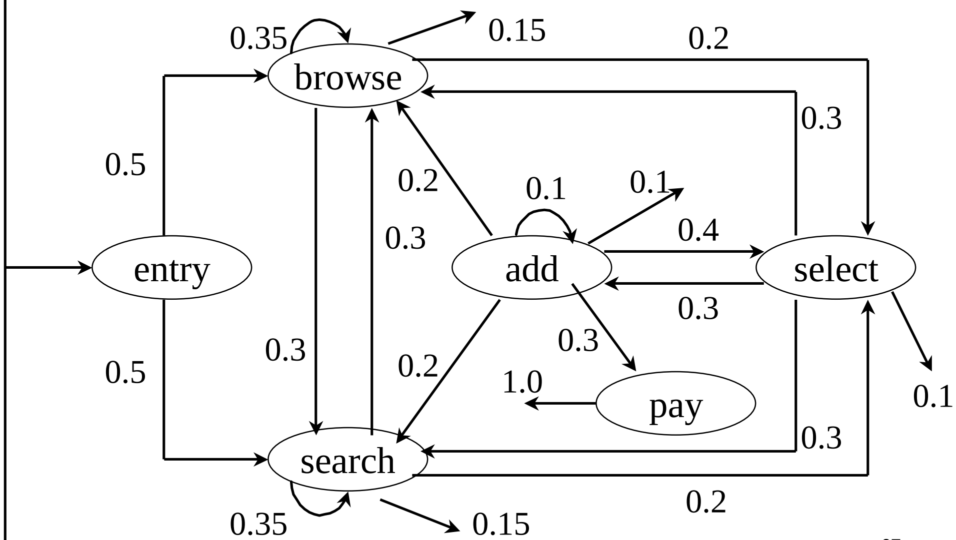
 65

Customer Behavior Model Graph (CBMG)



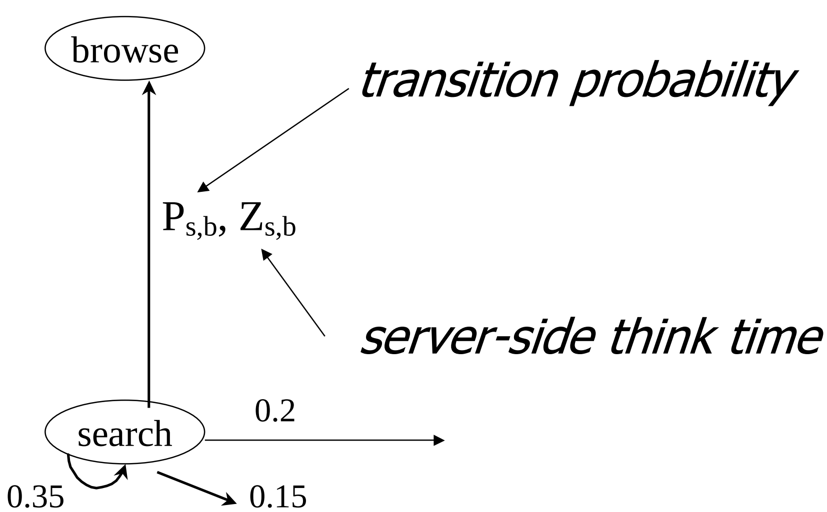
 66

Customer Behavior Model Graph (CBMG)



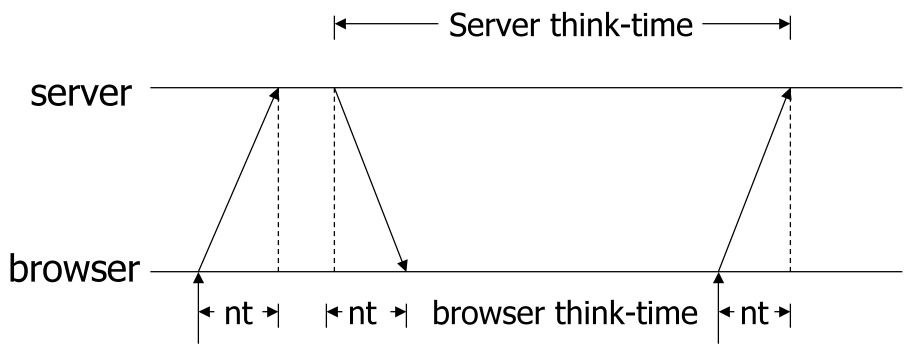
 67

Customer Behavior Model Graph (CBMG)



 68

Think-time



nt = network time

 69

CBMG

CBMG: G= (P, Z)

P = [$p_{i,j}$] *n* x *n transition probability matrix*

Z = [$z_{i,j}$] *n* x *n think-time matrix*

state 1: entry
state n: exit

 70

Metrics Derived from the CBMG

$$V_{entry} = 1$$

$$V_j = \sum_{k=1}^{n-1} V_k \times p_{k,j}$$

 71

Metrics Derived from the CBMG

- Average Number of Visits Per State
  - E.g., average number of searches per visit to the site,
- Average Buy to Visit Ratio: V_{pay}
- Average Session Length Per Visit: $\sum_{k=1}^{n-1} V_k$

 72

Workload Characterization Methodology

❶ Identify the different types of sessions that compose the workload, represented by CBMGs

❷ Compute workload intensity parameters per class:

- session arrival rate λ_r^s
- arrival rate per request type $\lambda_r^j = \lambda_r^s \times V_j^r$
- think-times (matrix Z)

Workload Characterization Methodology

❸ Determine resource usage parameters.

$$D_{i,r} = \sum_{j=1}^{n-1} D_{i,r,j} \times V_j^r$$

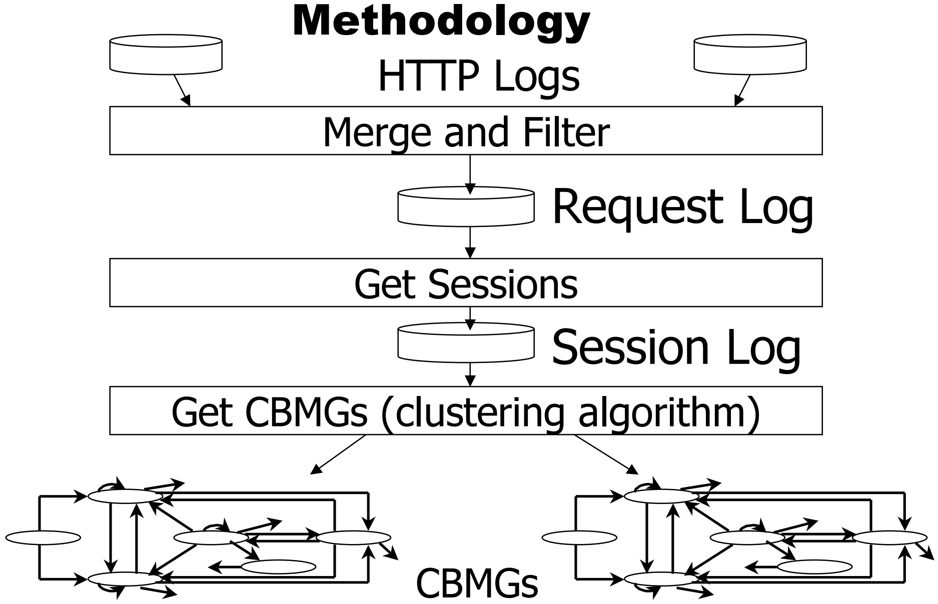
demand at device i by class r

demand at device i by class r for request j

Workload Characterization for E-commerce

- Need to map e-commerce functions to transactions:
  - search: SearchBookByTitle
- Need to map transactions to resource demands:
  - SearchBookByTitle: 3 I/Os in the Index disk, 10 I/Os in the main DB disk, 40 msec CPU, 16 Kbytes transferred over the LAN.

Workload Characterization Methodology



Workload Characterization

- HTTP log entry:
  (user_id, request_type, request_time, execution_time)
- k-th session log entry: (C_k, W_k)

$C_k = [c_{i,j}]$ n x n matrix of transitions counts between i and j

$W_k = [w_{i,j}]$ n x n matrix of accumulated think times between i and j

Procedure GetCBMGs

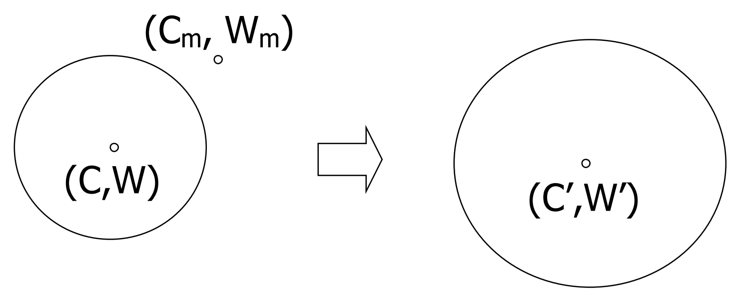
- Session log: $X_m = (C_m, W_m)$, m =1, …, M
- Distance between points X_a and X_b:

$$d_{a,b} = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} \left(C_a[i,j] - C_b[i,j]\right)^2}$$

- k-means clustering algorithm.

Procedure GetCBMGs (cont'd)

- Adding point $X_m = (C_m, W_m)$ to centroid k represented by point (C, W):



 79

Procedure GetCBMGs (cont'd)

- Adding point $X_m = (C_m, W_m)$ to centroid k represented by point (C, W):

$$c'[i,j] = \frac{s(k) \times c[i,j] + c_m[i,j]}{s(k)+1}$$

$$w'[i,j] = \frac{s(k) \times w[i,j] + w_m[i,j]}{s(k)+1}$$

 80

Procedure GetCBMGs (cont'd)

- Obtaining matrices P and Z for each cluster:

$$p[i,j] = \frac{c[i,j]}{\sum_{k=1}^{n} c[i,k]}$$

$$z[i,j] = \frac{w[i,j]}{c[i,j]}$$

Assessing the efficiency of the clustering algorithm

- Intracluster distance for cluster k:

$$\tilde{d}_k = \frac{1}{s(k)} \sum_{x \in C_k} d(x, C_k)$$

- Intercluster distance between clusters i and j:

$$\tilde{D}_{i,j} = d(C_i, C_j)$$

Assessing the efficiency of the clustering algorithm

- Intracluster distance for cluster k: avg, variance and coeff. of variation:

$$\bar{d} = \frac{1}{k} \sum_{j=1}^{k} \tilde{d}_k$$

$$\sigma_{\text{intra}}^2 = \frac{1}{k-1} \sum_{j=1}^{k} (\tilde{d}_k - \bar{d})^2 \qquad k > 1$$

$$C_{\text{intra}} = \sigma_{\text{intra}} / \bar{d}$$

Assessing the efficiency of the clustering algorithm

- Intercluster distance between clusters: avg, variance and coeff. of variation:

$$\bar{D} = \frac{1}{k(k-1)/2} \sum_{j=1}^{k} \sum_{j=i+1}^{k} \tilde{D}_{i,j} \qquad k > 1$$

$$\sigma_{\text{inter}}^2 = \frac{1}{k(k-1)2-1} \sum_{i=1}^{k} \sum_{j=i+1}^{k} (\tilde{D}_{i,j} - \bar{D})^2$$

$$k > 2$$

$$C_{\text{inter}} = \sigma_{\text{inter}} / \bar{D}$$

Assessing the efficiency of the clustering algorithm

- Minimize intracluster variance
- Maximize intercluster variance
- "small" enough number of points to achieve compact and representative workload representation.
- Use ratio between intra and inter cluster variances and coefficients of variation.

Assessing the efficiency of the clustering algorithm

- Ratio between intra and inter cluster variances:

$$\beta_{\text{var}} = \frac{\sigma^2_{\text{intra}}}{\sigma^2_{\text{inter}}}$$

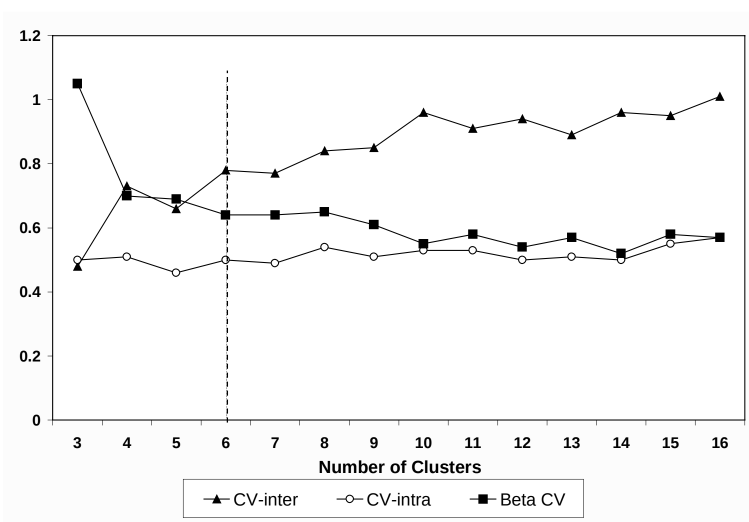
- Ratio between intra and inter cluster coefficients of variation:

$$\beta_{\text{cv}} = \frac{C_{\text{intra}}}{C_{\text{inter}}}$$

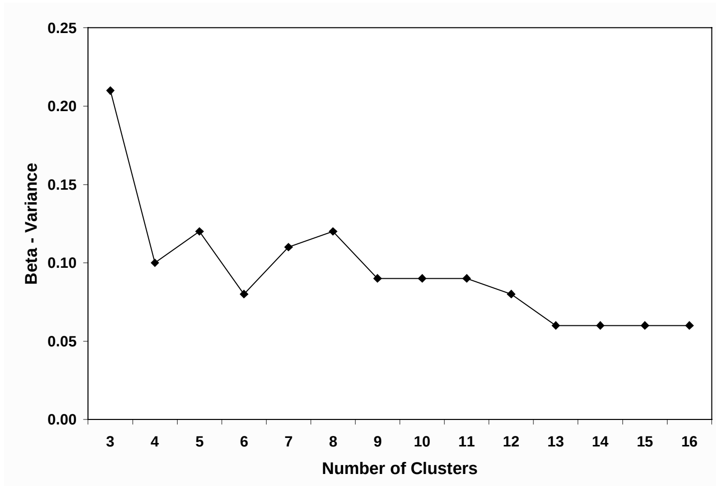
Assessing the efficiency of the clustering algorithm

- Synthetic HTTP log with 340,000 entries.
  - Generated 20,000 sessions
- Real HTTP logs from a retail online store.
  - 628,000 requests after images were eliminated.
  - Robot sessions were detected (very long sessions)
  - identified 34,811 customer sessions with avg. session length of 8.46 requests each.

Intra and Inter cluster CV and β_{cv} vs. number of clusters



β_{var} vs. number of clusters



 89

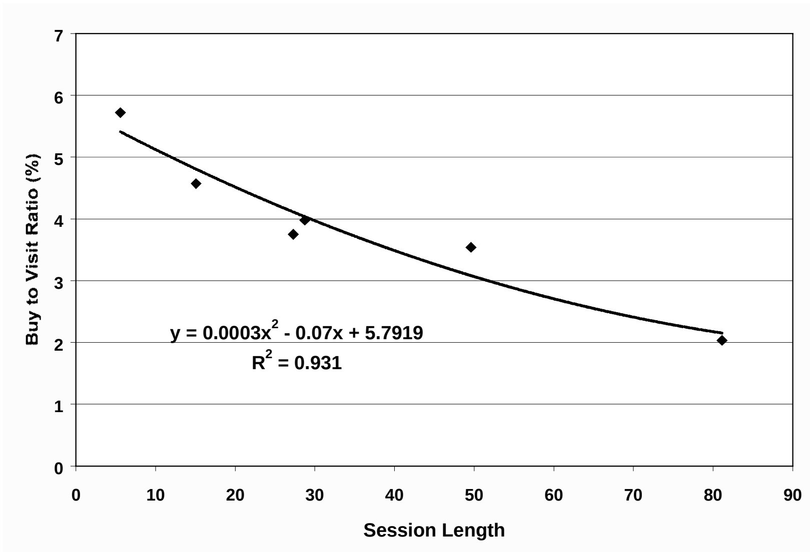
Clusters for Synthetic Logs

Cluster	1	2	3	4	5	6
% of Sessions	44.28	28.00	10.60	9.29	6.20	1.50
BV Ratio (%)	5.70	4.50	3.70	4.00	3.50	2.00
Session Length	5.6	15	27	28	50	81
AV Ratio (%)	11	15	21	20	32	50
Vb+Vs	3.6	11.4	20	23	39	70

- Cluster 1: majority of sessions, short sessions, and highest BV ratio.
- Cluster 6: small fraction of sessions, large sessions, smallest BV ratio.

 90

Buy to Visit Ratio vs. Session Length



Clusters for Real Logs

Cluster	Percent of Points	Avg. Session Length
1	6.5	12.0
2	42.6	6.9
3	20.4	7.2
4	12.7	9.0
5	2.7	14.8
6	8.0	12.0
7	7.2	11.2

Workload Characterization for E-commerce

- Customers interact with the site through sessions, which are sequences of interrelated requests.
- Sessions can be characterized by a CBMG.
- Groups of “similar” customers can be characterized by a CBMG per group.
- CBMGs can provide important metrics such as: buy to visit ratio and average session length.

 93