2009 IEEE 34th Conference on Local Computer Networks (LCN 2009)
Zürich, Switzerland; 20-23 October 2009

A Utility Model for Peering of Multi-Provider Content Delivery Services

Mukaddim Pathan and Rajkumar Buyya
Grid Computing and **D**istributed **S**ystems (GRIDS) Laboratory
Department of Computer Science and Software Engineering
The University of Melbourne, Parkville, Victoria 3010, Australia
{apathan, raj}@csse.unimelb.edu.au

***Abstract*—Peering of Content Delivery Networks (CDNs) allow providers to rapidly scale-out to meet both flash crowds and anticipated increases in demand. Recent trends foster the need for a utility model for content delivery services to provide transparency, high availability, reduced investment cost, and improved content delivery performance. Analysis of prior work reveals only a modest progress in evaluating the utility for peering CDNs. In this paper, we introduce a utility model and measure the content-serving ability of the peering CDNs system. Our model assists in providing a customer view of the system's health for different traffic types. Our model also captures the traffic activities in the system and helps to reveal the true propensities of participating CDNs to cooperate in peering. Through extensive simulations we unveil many interesting observations on how the utility of the peering CDNs system is varied for different system parameters and provide incentives for their exploitation in the system design.**

Keywords-Content Delivery Networks (CDNs), peering, utility.

I. INTRODUCTION

Content Delivery Networks (CDNs) focus on optimizing the delivery of content to Internet end-users from multiple, geographically distributed replica servers located at the *edge* of the network [4][13][14]. Popular Web sites often suffer congestion, bottlenecks, and even lengthy downtime due to large demands made on the resources of the provider hosting the Web content. This phenomenon can manifest itself as instances of unexpected flash crowds [2] resulting from external events of extreme magnitude and interest or sudden increase in visibility after being linked from popular high traffic Websites such as Slashdot[1] or Digg[2]. Interconnection of distinct CDNs, also termed as 'peering CDNs' [15], is one of the possible solutions to handle flash crowds, Web resources over-provisioning, and adverse business impact, by providing coordinated and cooperative content delivery among CDNs. Peering between CDNs can be established for a short or long-period to handle workload variations, thus allowing providers to expand their reach and capacity. In addition, it can achieve economies of scale, in terms of cost effectiveness and performance for both providers and end-users.

The main value proposition for traditional CDN services has shifted over time. Initially, the focus was on improving end-user perceived experience by decreasing response time, especially when the customer Web site experiences unexpected traffic surges. Nowadays, CDN services are treated by content providers as a way to use a shared infrastructure to handle their peak capacity requirements, thus allowing reduced investment cost in their own Web site infrastructure. Moreover, recent trends in CDNs indicate a large paradigm shift towards a utility computing model, which allows customers to exploit advanced content delivery services without having to build a dedicated infrastructure [10][19]. These trends foster the necessity and success of a well-designed content-utility system to provide highly scalable Web content delivery over the Internet.

Utility of content delivery services could be measured using a representative metric which captures the traffic activities in a CDN, expressing the usefulness of its replica servers [18]. In the context of peering CDNs, utility refers to the quantitative measure of the system-specific perceived benefit for content delivery. Customers interact with the peering CDNs system in a limited number of ways and have little experience of the associated complex technologies. The responsibility of ensuring high performance content delivery is largely on the peering CDNs system itself.

In this paper, we exploit a utility-based *privileged provider* model to capture the content-serving ability of peering CDNs via a *utility* measure. This is a monopolistic-natured model, where a CDN that initiates peering has the exclusive authority in the system. The measured utility can be translated into content providers' usage benefits from peering CDNs. It could also be used to assess a provider's propensity to coordinate in peering. With the aid of extensive simulations, we reveal many interesting observations on how different system parameters impact on the utility of peering CDNs. The main contributions of this paper are:

- A model to evaluate the content-serving utility and benefits of the peering CDNs system; and
- Analysis of the impact of system parameters on utility and insights for peering CDNs system design.

The rest of the paper is structured as follows. Section II provides a brief description of peering CDNs. It is followed by the utility model for peering CDNs in Section III. Section IV presents simulation methodology. Next, in Section V, results are discussed. An overview of related work is presented in Section VI. Finally, the paper is concluded in Section VII.

[1] http://www.slashdot.org
[2] http://www.digg.com

II. PEERING CDNS: OVERVIEW

This work considers an interconnection of multi-provider CDNs that interact with one another under negotiation-based relationships for scaling their geographical coverage; these relationships are termed as *peering*. The CDN that initiates the peering is called a *primary*, whereas other CDNs that agree to provide resources are called *peers*. In the peering CDNs system, a provider serves user requests as long as it can handle the load internally. If load exceeds its capacity, the excess requests are offloaded to other least loaded Web server(s) of peers. Each CDN has its own user request stream and a set of Web servers, but delegates only a subset of them, i.e. subCDN, to take part in peering. At any time, a given CDN may be in the roles of either a primary or a peer, i.e. the roles are fluid. The primary CDN directly manages the resources it has acquired, insofar that it determines what content is served and what proportion of the incoming traffic is redirected.

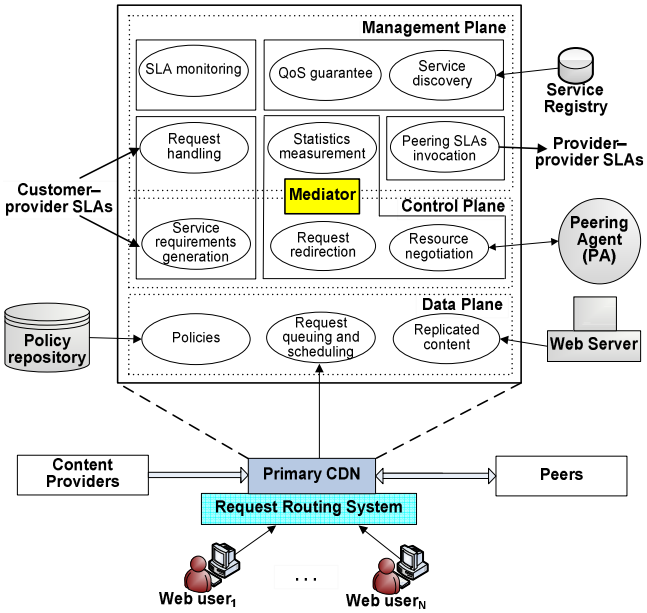

Figure 1. Architectural framework of peering CDNs.

Fig. 1 outlines an architectural framework for peering CDNs, which is proposed in a previous work [15]. For the sake of readability, here we provide a brief description of the architecture. It comprises three planes—*management*, *control* and *data*. The management plane is responsible for (a) interacting with content providers and peers to negotiate contracts; (b) monitoring Service Level Agreements (SLAs) according to the Quality of Service (QoS) requirements; (c) measuring server and network status; and (d) discovering services based on QoS constraints. The operations of the management plane are assisted by the *Service Repository* (*SR*), which encapsulates the status of CDN servers. The control plane covers request-redirection and resource management, i.e. resource negotiation, server load status verification and maintenance. At the heart of the control plane is the *Mediator*, which performs policy-driven authoritative operations on behalf of the primary CDN. It interacts with the *Peering Agent* (*PA*) that performs external resource discovery in the peering CDNs overlay. The management and control planes collectively govern two types of SLAs, namely, *customer-provider* SLAs and *provider-provider* SLAs. As the names suggest, the first one is a service specification contract between the customer (content provider) and the primary CDN, whereas the latter is between the primary and peer(s). Finally, the data plane is responsible for operations, such as content replication, request queuing, scheduling of user content requests according to stored policies, and is configured by the control plane. The data plane incorporates the *Policy Repository* (*PR*), virtualizing all policies within the peering arrangement, and the *Web servers* (*WSs*), actual placeholders of content. The PA, Mediator, SR and PR collectively act as a "conduit" for a given primary CDN, and assist in external resource discovery. User requests for content are made to the Request Routing System (RRS) of the primary. These requests are then forwarded either directly to its server(s), or to a peer.

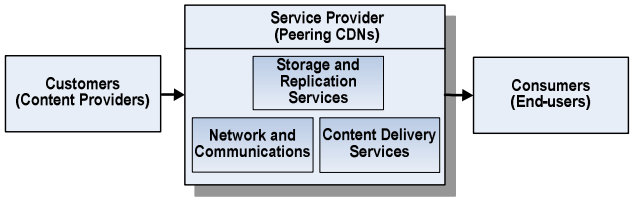

Figure 2. Peering CDNs as a content-utility system.

III. UTILITY MODEL OF PEERING CDNS

The peering CDNs system is a type of content-utility system [19], characterized as offering content delivery services with high availability, transparency, and improved performance without requiring content providers to build or manage complex infrastructure themselves. As shown in Fig. 2, it can be interpreted as a content-utility system comprising three main entities—*customers*, *service providers*, and *consumers*. Under this model, a provider can adjust resource provisioning for its content delivery services dynamically based on user demand.

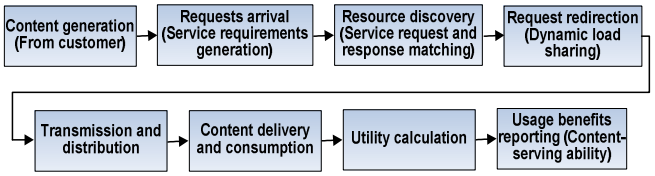

Figure 3. Major operational steps in the content-utility system.

Content providers are responsible for creating content, processing it to conform to certain formats, and developing content metadata. The peering CDNs system as a service provider encapsulates several distinct CDN providers with storage and replication services to provide and manage content storage; network and communication services to enable swift and seamless transfer of content over communications and data networks; and content delivery services to ensure easy and effective content consumption. Finally, consumers interact with the system by specifying the content/service requests through cell phone, smart phone/PDA, laptop and desktop.

Fig. 3 shows the major operational steps in the utility model of peering CDNs. The system first captures/generates content from the customer (content provider). The generated content is then processed, indexed, and stored for online access. Upon

receiving user requests for content, peering resources are discovered using a communication protocol, reminiscent of the public/subscribe paradigm [16]. Under traffic surges, request-redirection is performed from the primary to the least loaded servers of peers. Consequently, the system then delivers the requested content to consumers in a swift and cost-effective manner [17]. Once the system is fully in operation, its utility is disclosed to indicate to what extent it could fit an individual content provider's needs.

An important benefit of this content-utility model is the availability of several services to enhance content usage and enrich end-users' (consumers) experience with minimal burden. It also allows content providers (customers) to seamlessly interact with the peering CDNs system. The benefits for the CDN providers include the opportunity to exploit the availability of powerful, cost-effective services that offer customized content delivery.

A. *Content-Serving Utility*

To quantify the usage benefits of peering CDNs, we develop metrics to measure its content-serving utility. Let us consider a peering CDNs system comprising N replica servers from multiple participants, with the primary CDN having exclusive right to redirect requests for content. We use $R = \{r_j\}, j \in \{1, 2, \ldots, M\}$ to denote the set of user requests, with r_j being the j-th arriving request to the system. A utility metric u_{ij} expresses the value gained by a server i for serving an assigned request r_j according to the specified service requirements, i.e.

$$u_{ij} = \begin{cases} u_i & \text{iff } x_{ij} = 1 \\ 0 & \text{Otherwise} \end{cases}$$

where x_{ij} is the indicator variable to determine whether request r_j is assigned to server i according to the service requirements.

The most useful replicas for the peering CDNs system are those exhibiting the highest utility. In this regard, we have drawn inspiration from Mortazavi and Kesidis [12], which uses the notion of net utility to study the reputation of a node in a Peer-to-Peer (P2P) system. We quantify the utility of a replica server with a value that expresses the relation between the number of serviced content requests against the number of rejected content requests. It is bounded to the range [0, 1] and provides an indication of the traffic activity. Formally, we quantify the utility u_i of a replica i to service the assigned requests in peering CDNs by using the following equation:

$$u_i = (2/\pi) \times \arctan(\zeta) \quad (1)$$

The main idea behind this metric is that a peer's replica is considered to be useful (high utility) if it serves content more than it rejects, and vice-versa. The parameter ζ is the ratio of the serviced requests to the rejected requests,

$$\zeta = \textit{No. of serviced requests} / \textit{No. of rejected requests} \quad (2)$$

The arctan function in (1) assists to obtain scaled resulting utility in the range [0, 1]. The value $u_i = 1$ is achieved if the replica does not reject any request. It can happen when the replica is working well under its capacity (i.e. almost idle) and ready to receive more content requests (yielding $\zeta = \infty$). The value $u_i = 0$ is achieved if the server is down and/or overloaded and cannot serve any request ($\zeta = 0$). In the case of equal number of serviced and rejected requests, the resulting utility value is 0.5. Ideally for the peering CDNs system, the most cooperative replicas are those with high utility values.

By using individual server utility values and assuming that the requested content delivery services have been performed, we can compute the utility of the peering CDNs system by taking the mean value of the yielded replica utilities. We refrain from using weighted average for this purpose as our aim is to reveal the true propensity of a CDN to cooperate in peering. For a peering CDNs system governing N replicas from multiple providers, the content-serving utility U_P is:

$$U_P = \sum_{i=1}^{N} u_i \Big/ N \quad (3)$$

The obtained utility using (3) could be translated into the content-serving ability (*durability*) of the peering CDNs system. A highly durable peering CDNs system exhibits high utility. This quantitative measure can provide an indication of the effectiveness (health) and usage benefits of the peering CDNs system to the content providers.

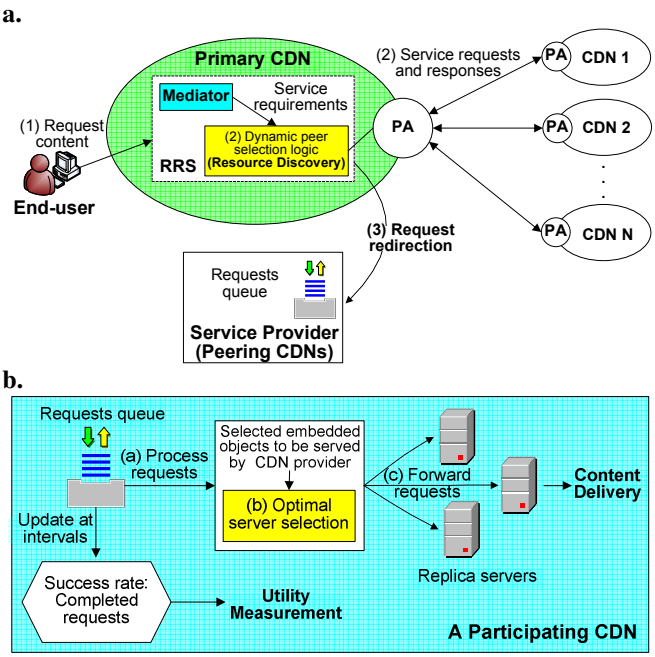

Figure 4. A schematic representation of—(a) Simulation methodology and (b) Request servicing and utility measurement by a provider.

IV. Evaluation Methodology

A schematic representation of the methodology used to evaluate the utility of peering CDNs is provided in Fig. 4. Herein we exploit the resource discovery and request-redirection mechanisms presented in a previous work [16]. The simulation methodology realizes a privileged provider model (Fig. 4(a)), with primary CDN having the authoritative right over the resources it has acquired—which are delegated rights for the peers' physical resources. Content requests from end-users arrive to the Request Routing System (RRS) of the primary CDN. For certain content requests, under peak load or

traffic surges (during a flash crowd event), users are redirected to the least loaded server(s) of peers. Prior to redirection, matching of service requests and responses is performed to find target peers. Upon resource discovery, excess requests are offloaded to peers' server(s) in a cost-effective manner (in terms of traffic load and network proximity). To ensure scalability, redirection is performed in a per-flow manner, i.e. an optimal server is selected to accommodate multiple user flows. Requests for the same content from the same user group are aggregated and routed to the selected server(s). Any load imbalance in the system is alleviated through the repeated use dynamic request-redirections [16].

The request servicing and utility measurement methodology in the peering CDNs system by the selected optimal server of a participating provider is fleshed out in Fig. 4(b). Our approach ensures that requests are serviced by the best responding server under highly skewed load, while shifting traffic away from sites that are unreachable or near capacity. Simulations ensure that the system is autonomic and self-healing in that if some sites are unusable, it moves traffic to others with no manual intervention. Each participating provider publishes aggregated individual success rate for satisfying incoming content requests at a time interval (epoch) during simulations so that replica utilities and consequent CDN utility can be measured. The peering CDNs system in turn uses the individual utility functions to measure its utility and durability under variable incoming traffic. The outcomes can be conveyed to content providers as usage benefits of the peering CDNs system.

TABLE I. PARAMETERS USED FOR EVALUATION

Parameter	Value
Number of CDNs	4
Sets of servers	10
Server capacity	Finite and heterogeneous [17]
Service distribution among servers	Dissimilar [17]. Pareto: $\alpha k^{\alpha} x^{-\alpha-1}$, Log-normal: $\frac{1}{x\sqrt{2\pi\sigma^2}} e^{\frac{-(\ln x-\mu)^2}{2\sigma^2}}$, Hyper-exponential: $\sum_{i=1}^{n} P_i \lambda_i e^{-\lambda_i x}$, and Erlang: $\frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!}$
Traffic (end-user) distribution	$f(x) = \alpha k^{\alpha} x^{-\alpha-1}, \alpha, k > 0, x \geq k$
Traffic types	High to moderate variable, depending on the traffic distribution. Class 1 (α = 1), Class 2 (α = 1.2), Class 3 (α = 1.5), and Class 4 (α = 2)
Agreegated success rate refresh interval	1000s

A. Simulation Environment and Parameters

Measurement-based studies on a real CDN testbed are not always possible due to the proprietary nature of commercial CDNs and their limited availability for research purposes. Therefore, we resort to simulations to evaluate the content-serving utility model of peering CDNs. We have implemented a simulator [17], based on Independent Replication Method, using the CSIM/Java[3] simulation toolkit, to conduct *repeatable* and *controlled* experiments that would otherwise be difficult to perform on a real CDN testbed.

[3] It creates process-oriented discrete-event simulation models. For more information, please check: http://www.mesquite.com

Our simulation model is representative of a peering CDNs environment. It is based on a reference scenario [17], consisting of four CDNs with their sets of Web servers placed at different geographical locations across the Internet. Each CDN has a set of servers and a pool of users to generate its own request stream. Users request content via their own browsers and make use of a proxy server according to the same client-side policy. To take part in peering, each participating CDN defines a subCDN with a subset of its resources. To provide an accurate characterization of the scenario, we simulate the main system entities—Web servers, mediator, distributed service registry, network congestion, and end-users. Table 1 reports the indicative system parameters for the simulation model[4].

CDN servers are implemented as a set of finite and heterogeneous capacity facilities[5], which serve incoming content requests according to different service distributions. They are configured according to the specifications from Fourth Quarter 2006 SPECweb2005 Results[6]. We keep track of the number of active connections at the server side to calculate the aggregated success rate during simulations.

User requests are implemented as CSIM processes[7]. Like the Internet access workloads, these user requests exhibit self-similarity. A self-similar process has observable bursts in all time scales. It exhibits long-range dependence, where values at any instant are typically correlated with all future values. This self-similar nature in user requests can be described by using a heavy-tailed distribution [6]. Therefore, user requests to each CDN Web server follow a Pareto distribution with PDF:

$$f(x) = \alpha k^{\alpha} x^{-\alpha-1}, \alpha, k > 0, x \geq k$$

where α determines the weight of the tail of the distribution.

B. Performance Metrics

We evaluate the utility of the peering CDNs system under four types of request traffic—Class 1 (α = 1), Class 2 (α = 1.2), Class 3 (α = 1.5), and Class 4 (α = 2)—varying the request arrival from high (α = 1) to moderate (α = 2) variability. The reason behind using different traffic classes is due to the observation that subscribing content providers can be highly heterogeneous in terms of their traffic patterns and the type of content they handle [8]. Hence, end-users request for content of varying sizes (ranging from small to large). The processing requirements also vary based on size of the content requested. The use of different traffic types allows us to reflect different user preferences and content request types. Consequently, these traffic types determine the behavior of processing in a given CDN's service capacity and influences the measured utility.

The primary focus of this study is to provide observations on how the peering CDNs system's content-serving ability is varied for different system parameters. The *utility* of peering CDNs is measured according to the model in Section III-A. Using the notion of utility, we express the traffic activity of the

[4] Full listing of system parameters have been reported previously [16][17].
[5] Each facility is a simulated resource with a single server and a queue for waiting requests.
[6] Standard Performance Evaluation Corporation. http://www.spec.org/
[7] CSIM processes are objects, based on Java threads, which make use of simulated resources.

system. High utility value not only indicates the proficient content-serving ability of the system, but also signifies its durability under highly variable traffic activities.

To emphasize the impact of different traffic types on the measured utility, we report the *cumulative frequency* of the *minimum utility* (at a given instant) in the peering CDNs system. We present for each level of utility the probability (or fraction of time) that the system realizes the given minimum utility for a certain traffic class. This metric provides an indication of the relative frequency of satisfying user requests for different traffic types. For example, if the probability of achieving 0.8 utility is 0.75, it implies that there is a probability 0.25 that the system realizes utility below 0.8.

We also measure the mean *response time* of each CDN, which specifies the average serving time of the requests to end-users. Lower values indicate fast serviced content. In addition, we keep track of the number of *completions* to show how a CDN is susceptible to different traffic types. Finally, we use *rejection rate*, which states the number of disruptions due to service unavailability. Table 4 summarizes the performance indices that are used in the experimental evaluation.

TABLE II. LIST OF PERFORMANCE INDICES

Performance Index	Description
Utility	Content-serving ability, ranges in [0, 1]
Minimum utility	Lowest utility at a given instant in the peering CDNs system
Cumulative frequency of minimum utility	The probability that the minimum utility of peering CDNs is below a certain value
Response time	The time experienced by a end-user to get serviced
Completions	Number of completed requests
Rejection rate	Number of dropped requests due to service unavailability

V. EXPERIMENT RESULTS

We run our experiments according to the methodology (Section IV) for a reference model [17], with one provider as primary (CDN 1) and others as peers. Results are averaged over ten simulation runs, where the duration of each simulation run is determined by the run length control algorithm built in CSIM. This approach endeavors to converge to the *true solution* of the simulation model in a finite simulation run. We avoid the adverse effects of both overly short and too long simulation runs, which may respectively cause inaccurate performance statistics, and unnecessary wastage of computing resources and delays in the completion of the simulation study. It is found that each simulation run is for 3 hours of the peering CDNs system activity. During a simulation run there are epochs of 1000s, at which the aggregated success rate of serviced requests is published. For all simulation results, confidence intervals[8] are estimated, and the 95% confidence interval is observed to be within 3% of the mean.

A. Request Traffic vs. Utility

We first present the utility of the participating CDNs for different traffic types (Fig. 5). For a provider, it is a normalized ratio (in [0, 1]) expressing the number of requests serviced against the number of service disruptions. Each bar represents different CDNs, while the bold line represents the utility of the peering CDNs system as a whole. In general, we observe that altering from high to moderate variable request traffic results in lower to higher utility for the participating providers. We discuss more on this issue in the next sections.

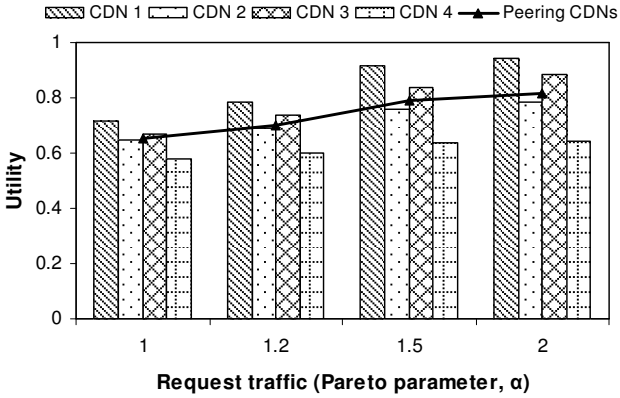

Figure 5. Utility measures for different traffic types.

In the peering CDNs system, a CDN provider's utility implicitly depends on the number of its allocated servers and their associated service capacities. Moreover, optimal server selection using request-redirection [17], taking into account network proximity, congestion and traffic load of a server, also impacts the resulting utility in a CDN. It should also be noted that the use of different redirection policies by the primary to direct requests to the participating peers' server under different scenario may result in different utility values. Fig. 5 shows that CDN 1 (primary) and CDN 3 (a peer) demonstrate higher utility than that of other peers. In this case, they contribute more servers (with higher capacity) to the system than other peers. In addition, server selection results in more requests to be redirected from CDN 1 to the servers of CDN 3, identifying it as a peer with close proximity to the primary.

While CDN 2 does not allocate as many servers as CDN 1 and CDN 3, it still exhibits higher utility than CDN 4. The reason behind this trend lies in the difference of service distribution and capacity of CDN 4. Moreover, this peer contributes the fewest servers to the system. Optimal server selection does not produce as many target servers from CDN 4 as in other peers. Yet it can only gain low utility for the fewer (in comparison to other peers) redirected requests. This low utility value in CDN 4 leads to the logical implication that its contributing server is distant in terms of network proximity, there might be congestion in the network path, and/or it has been suffering high traffic load. We revisit this point with supporting results in Section V-C to justify our reasoning.

From Fig. 5, it can be seen that a low utility value of CDN 4 significantly contributes to the overall utility of peering CDNs. Since the resulting utility of the system is averaged over the individual utilities of participants, without the contribution from CDN 4 the system yields more utility value. Therefore, the primary may decide to either re-negotiate peering or exclude CDN 4 from peering in order to harness better content-serving ability from the system.

[8] A range of values in which the true answer is believed to lie with a high probability.

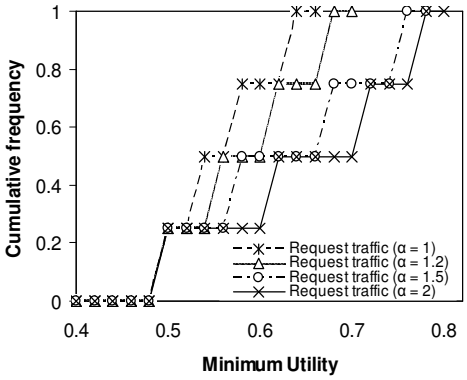

Figure 6. Cumulative frequency of minimum utility in peering CDNs.

B. Content-Serving Ability

In this section, we focus on the content-serving ability of peering CDNs. For that we investigate into detail the impact of different traffic types for the resulting utility in peering CDNs. Fig. 6 summarizes the effect of request distributions of candidate traffic classes on the achieved utility. The main goal is to show to what extent the system can satisfy user requests. For this reason, rather than adopting traditional metrics such as the standard deviation of utilities, we evaluate the primary CDNs performance under different traffic types through the minimum utility observed during simulation epochs. Fig. 6 shows the cumulative frequency of the minimum utility as the major performance criterion. It indicates the probability (or fraction of time) that the system is able to achieve a given utility level. From the figure, it is visible that for moderate traffic ($\alpha = 2$), peering CDNs has a probability of 1.0 that it can realize little higher than 0.8 utility. The system is susceptible to traffic variability and thus exhibits lower utility values for heavy traffic. When the traffic distribution is highly variable ($\alpha = 1$), the finite capacity servers of the participating providers fail to serve many requests. Specifically, under this traffic class, the peering CDNs system can only achieve less than 0.7 utility. It establishes the reasoning that the utility of the peering CDNs system is heavily dependent on the incoming traffic.

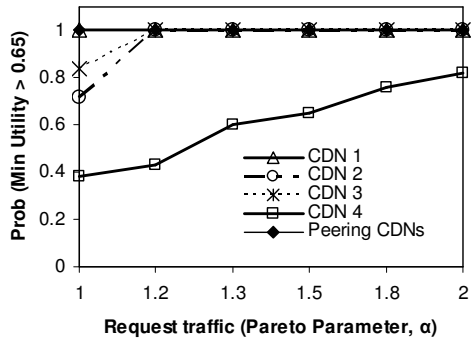

Figure 7. Sensitivity to traffic distribution.

To report the individual content-serving ability of participants, we use the probability that the minimum utility is above 0.65, i.e. Prob (Min Utility > 0.65). Fig. 7 presents the sensitivity to traffic distribution for each participating CDNs. We observe that the primary (CDN 1) realizes invariant performance for any traffic type (moderate to highly variable). It exhibits the maximum utility at all times and does not go below the minimum utility level. On the contrary, its peers (CDN 2 and CDN 3) are prone to the variability of incoming traffic. Specifically, they demonstrate diminishing performance with heavy traffic. Out of all the peers, CDN 4 has the worst performance and shows close to 0.4 probability of gaining utility above the minimum utility level, under heavy traffic demand with high variability. This is due to the fact that this peer drops many requests due to service disruptions. We further elaborate on the service disruption aspect with supporting results in Section V-D.

C. Response Time vs. Utility

Fig. 8 records the utility and mean response time of the participating CDNs. Two scales—Scale 1 and Scale 2—are used to respectively plot the response time and utility of each provider against different traffic types. The mean response time measure indicates the responsiveness of an individual CDN and the user perceived experience when accessing its servers.

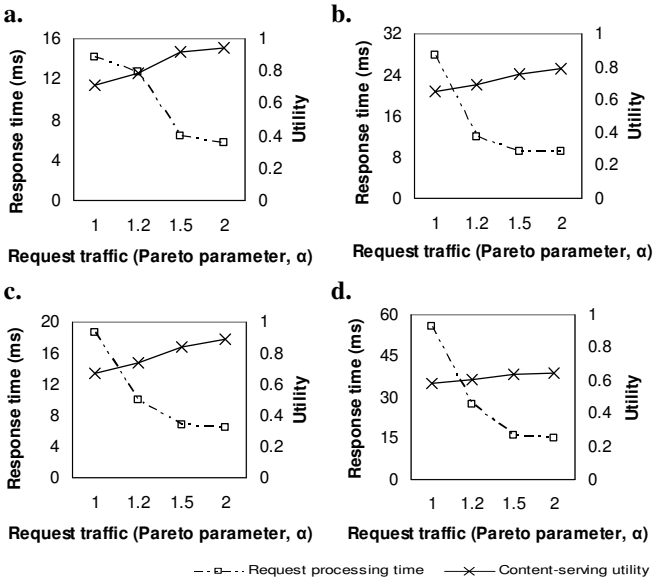

Figure 8. Relation between mean response time and content-serving utility. (a) CDN 1; (b) CDN 2; (c) CDN 3; and (d) CDN 4.

From the figure, it could be noted that response time and utility are inversely related. For a low response time, a high utility value could be achieved. For all the participating providers in the peering CDNs system, we observe a similar trend. When a CDN is more responsive to incoming requests, end-users comprehend low response time, as fewer requests are dropped due to service unavailability. Although it is desirable to achieve low response time and high utility at all times, variability in incoming traffic impacts the response time and thus leads to higher response time and lower resultant utility. In particular, Fig. 8 shows that the mean response time of CDN 2 and CDN 4 are more susceptible to incoming request variability. For highly variable incoming traffic (under traffic surges), the end-user perceived response times from these providers are increased, thus showing the evidence of likely network perturbations and heavy traffic load on their servers.

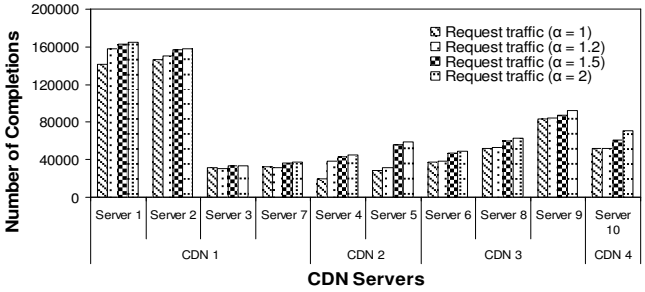

Figure 9. Number of completions in each CDN server.

D. *Service Completions*

Now we investigate how the number of satisfied requests dictates the utility of peering CDNs. For that we first study the number of request completions and rejections due to service unavailability. In our simulation, each server serves requests according to its finite capacity. Fig. 9 presents the average number of completed requests at each server over the simulation runs. It is found that with moderate incoming traffic, CDN servers attempt to serve more requests than that of the presence of highly variable request traffic. A similar trend can be observed from Fig. 10, which shows the total completions in each participating CDN and in the peering CDNs system.

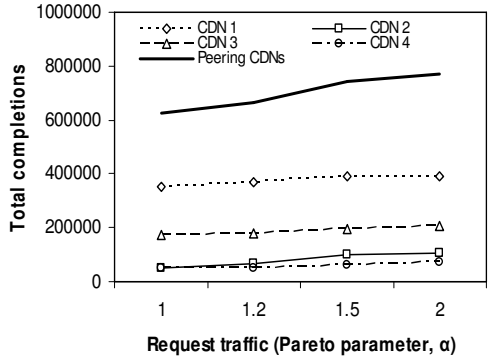

Figure 10. Total completions in each CDN and in the peering CDNs system.

If a server receives more requests than its capacity, unless it is provisioned with enough capacity to serve more concurrent connections, it will end up dropping many requests. Since the servers are configured to operate below their finite capacity, they suffer from service disruptions under traffic surges with highly variable incoming requests. Consequently, many requests can not be served as incoming requests arrive to a CDN server and find that it is operating at its highest capacity. Fig. 11 presents the average percentage of disrupted services for different traffic types at each epoch during the simulation runs. This figure is obtained with a fixed number of 1,000,000 requests. Service disruptions are expressed in terms of the average service rejection rate. To compute this performance metric, we first calculate the rejected service ratio as the number of requests that yielded a negative response (i.e. the system has not found a resource to serve this request), over the number of incoming requests. We then computed the average service rejection ratio as the average value over the number of total requests in the system. From Fig. 11, it is evident that finite server capacities lead to service disruptions, which is more significant for highly variable traffic.

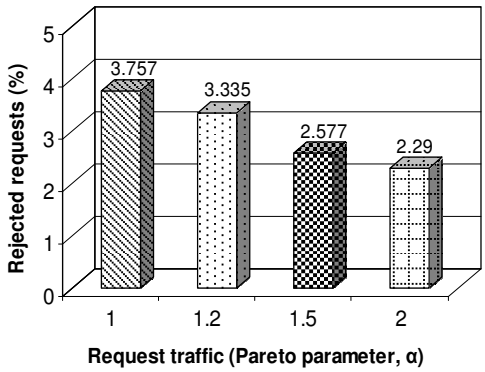

Figure 11. Service rejection rate in peering CDNs for different traffic types.

Finally, we demonstrate the impact of completions on the utility of peering CDNs. As conveyed earlier, the total number of completions and resulting utility lessens with highly variable traffic types. The observed trend is sensible as the number of serviced requests (completions) and associated rejections act as major parameters in our utility model. The supporting results are presented in Fig. 12.

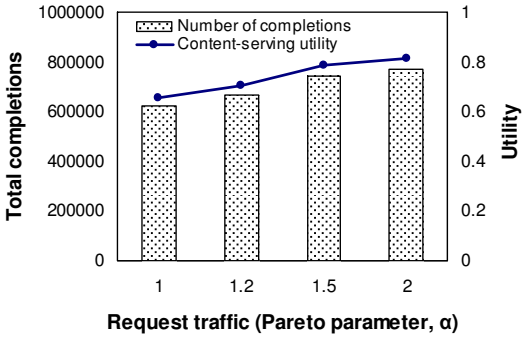

Figure 12. Total completions and utility for peering CDNs.

VI. RELATED WORK

There is a growing interest in interconnecting CDNs according to the Content Distribution Internetworking (CDI) model [7]. The aims are to improve performance for customers, and to achieve pervasive geographical coverage and increased capacity for a provider. While the CDI model lays the foundation for peering, it provides only an abstract view of how peering could be formed. Previous research such as CDN brokering [3] and associated request-routing [9]; multi-provider peering [1]; Synergy overlay internetworking [11] and peer-assisted content delivery [20] explore the benefits of peering/cooperation of CDN providers, P2P networks and/or overlays with main focus on offering increased CDN capacity, intelligent server selection, reduced cost, and better fault tolerance. Our work is complementary as we achieve improved performance, decentralized resource discovery and dynamic load sharing through request-redirection [16][17]. In addition to improving content delivery, we enumerate the content-serving ability of peering CDNs and provide incentives to its exploitation for better system design.

Several recent trends demonstrate the emergence of the utility model for CDNs and general Web applications in the Internet. Prior related work reflecting the utility computing notion includes an architectural framework for Content Serving Utility (CSU) [9], a content-utility model for digital content delivery [19], and a Web-based utility computing model for Internet applications [5]. They respectively provide an overview of the architectural framework, characterize the system features, and describe the approaches and challenges related to the design and development of such a utility computing platform for CDNs. On the contrary, we not only provide an overview of the utility model for peering CDNs, but also quantify the perceived utility.

Our work is in line with the simulation-based evaluation of utility as described in previous work [12][18]. While the first explores a utility-based cumulative reputation system to encourage cooperation in a P2P network, the latter set forth the usefulness of replicas and examine how single CDN utility is affected by various parameters. Our approach differs in that we utilize a privileged provider model to capture the content-serving ability of peering CDNs. We also reveal the impact of system parameters on the utility of peering CDNs.

VII. Conclusions and Future Work

Peering CDNs is a content-utility system that improves site performance and availability without requiring content providers to build or manage complex content delivery infrastructure themselves. In this paper, we use a utility model to measure the availability level and health (content-serving ability) of the peering CDNs system under variable traffic. This measure is crucial as the system wellness greatly affects the delivery and consumption of content. The outcomes can be interpreted as the benefits for a content provider to use peering CDNs. We also show that our utility model can assist to reveal the true propensity of a CDN provider to cooperate in peering.

With the aid of simulation experiments, we analyze the impact of system parameters on perceived system utility. We show that although the peering CDNs system observes high utility in terms of satisfying content requests, its content-serving ability is largely dependent on the participating providers' utilities. These utilities are essentially tailored to individual provider's service capacity, proximity, network conditions, and incoming request traffic. Our observations could be exploited for a better system design to cope with high traffic phenomena such as the flash crowd events.

The experiment results are quite encouraging to spawn a set of possible future work. We are currently implementing the proposed utility model in MetaCDN[9], which resembles a prototype of peering CDNs. We are also devising a utility-based request-redirection policy and defining a pricing policy to measure content provider and system surplus.

Acknowledgment

We would like to thank Marcos Assunção, Christian Vecchiola and Marco Netto from the University of Melbourne for sharing thoughts and for making incisive comments and suggestions on this paper. This work is supported in part by the Australian Research Council (ARC) and Department of Innovation, Industry, Science and Research (DIISR).

[9] An integrated overlay leveraging existing "Storage Clouds" to provide high performance content delivery. Please check: http://www.metacdn.org

References

[1] L. Amini, A. Shaikh, and H. Schulzrinne, "Effective peering for multi-provider content delivery services," In Proc. of the 23rd Annual IEEE Conference on Computer Communicatitons (INFOCOM '04), IEEE CS Press, Los Alamitos, CA, USA, pp. 850-861, 2004.

[2] M. Arlitt, and T. Jin, "Workload characterization of the 1998 world cup web site," IEEE Network, 14(3), pp. 30-37, 2000.

[3] A. Biliris, C. Cranor, F. Douglis, M. Rabinovich, S Sibal, O. Spatscheck, and W. Sturm, "CDN brokering," Computer Communications, 25(4), pp. 393-402, 2002.

[4] R. Buyya, M. Pathan, and A. Vakali, (Eds.) Content Delivery Networks. Springer-Verlag, Germany, 2008.

[5] C. Canali, M. Rabinovich, and Z. Xiao, "Utility computing for Internet applications," In Web Content Delivey, vol. II, X. Tang, J. Xu, and S. T. Chanson, Eds. Springer, 2006, pp. 131-151.

[6] M. E. Crovella, M. Taqqu, and A. Bestavros, "Heavy-tailed probability distributions in the World Wide Web," A Practical Guide To Heavy Tails, Birkhauser Boston Inc., Cambridge, MA, USA, pp. 3-26, 1998.

[7] M. Day, B. Cain, G. Tomlinson, and P. Rzewski, "A model for content internetworking," IETF RFC 3466, Feb. 2003.

[8] M. D. Dikaiakos and A. Stassopoulou, "Content-selection strategies for the periodic prefetching of WWW resources via satellite," Computer Communications, 24(1), pp. 93-104, Jun. 2001.

[9] O. Ercetin and L. Tassiulas, "Request routing in content distribution networks," Technical Report, Sabanci University, Turkey, 2003. http://digital.sabanciuniv.edu/elitfulltext/3011800000049.pdf

[10] P. Gayek, R. Nesbitt, H. Pearthree, A. Shaikh, and B. Snitzer, "A web content serving utility," IBM Systems Journal, 43(1), pp. 43-63, 2004.

[11] M. Kwon and S. Fahmy, "Synergy: An overlay internetworking architecture," In Proc. of the 14th International Conference on Computer Communications and Networks (ICCCN '05), pp. 201-206, Oct. 2005.

[12] B. Mortazavi and G. Kesidis, "Model and simulation study of a peer-to-peer game with a reputation-based incentive mechanism," In Proc. of the Information Theory and Applications (ITA) Workshop, Feb. 2006.

[13] B. Molina, C. E. Palau, and M. Esteve, "Modeling content delivery networks and their performance," Computer Communications, 27(15), pp. 1401-1411, 2004.

[14] G. Pallis, and A. Vakali, "Insight and perspectives for content delivery networks," Communications of the ACM, 49(1), ACM Press, NY, USA, pp. 101-106, Jan. 2006.

[15] M. Pathan, J. Broberg, K. Bubendorfer, K. H. Kim, and R. Buyya, "An architecture for virtual organization (VO)-based effective peering of content delivery networks," UPGRADE-CN'07, In Proc. of the 16th IEEE International Symposium on High Performance Distributed Computing (HPDC '07), ACM Press, NY, USA, Jun. 2007.

[16] M. Pathan and R. Buyya, "Resource discovery and request-redirection for dynamic load sharing in multi-provider peering content delivery networks," Journal of Network and Computer Applications, 32(5), Elsevier, Amsterdam, The Netherlands, pp. 976-990, Sept. 2009.

[17] M. Pathan, C. Vecchiola, and R. Buyya, "Load and proximity aware request-redirection for dynamic load distribution in peering CDNs," In Proc. of the 16th International Conference on Cooperative Information Systems (CoopIS '08), LNCS 5331, pp. 62-81, Nov. 2008.

[18] K. Stamos, G. Pallis, A. Vakali, and M. D. Dikaiakos, "Evaluating the utility of content delivery networks," In Proc. of the 4th UPGRADE-CN Workshop on Content Delivery and Management in Large-Scale Networks, ACM Press, NY, USA, Jun. 2009.

[19] S. R. Subramanya and B. K. Yi, "Utility model for on-demand digital content," IEEE Computer, 38(6), pp. 95-98, 2005.

[20] M. Tran and W. Tavanapong, "Peers-assisted dynamic content distribution networks", In Proc. of the 30th IEEE International Conference on Local Computer Networks (LCN '05), IEEE CS Press, Los Alamitos, CA, USA, pp. 123-131, 2005.