Adaptive Content Networking

Sven Buchholz
Dresden University of Technology, Germany
buchholz@rn.inf.tu-dresden.de

Thomas Buchholz
University of Munich, Germany
buchholz@informatik.uni-muenchen.de

Abstract. *Content adaptation is an essential concept to meet the heterogeneous requirements of clients in Pervasive Computing environments. However, content adaptation interferes with replication applied in content networks to improve the performance of information access. Leveraging the advantages of replication in the world of Pervasive Computing is the subject of this paper.*
We introduce the notion of Cost-Quality Optimized Content Networking and formalize the inherent optimization problem. Additionally, web-caching and Content Distribution Networks (CDN) are compared with respect to their potential to accommodate cost-quality optimized content adaptation showing web-caching to be inappropriate due to intolerable response delays. The feasibility in the CDN scenario is illustrated by outlining a CDN system architecture.

1 Introduction

In the upcoming world of Pervasive Computing, users access information in the Internet by a huge variety of devices featuring heterogeneous capabilities (with respect to display, data input, computing capacity, etc.). The devices are attached to the Internet by various communication systems featuring heterogeneous transmission characteristics. The key to cope with the heterogeneity is the modification of the representation of content in order to meet the media handling capabilities of the devices and the transmission restrictions imposed by the networks. This is referred to as content adaptation. In previous works, a lot of effort has been spent in the field of content adaptation that may be classified into three basic approaches: server-side adaptation, proxy-based adaptation, and adaptation paths (cf. [1]).

These approaches have been designed without taking the requisites of content networking into account, leveraging replication[1] to improve the performance by avoiding redundant data transfers. However, content adaptation interferes with the effectiveness of content replication as (1) replication of adapted contents reduces the sharing of replicas and (2) replication of generic representations results in full adaptation costs with every request and may undermine cache memory efficiency. We have examined these issues in detail in [1].

Leveraging the advantages of replication in the context of content adaptation is subject of our research. This is achieved by coordinating the replication with the composition of adaptation paths in order to store those representations in caches or surrogate servers that are adequately generic to fulfill subsequent requests from heterogeneous clients and sufficiently adapted to be efficient by means of cache memory consumption and adaptation costs. Those representations might be generic, partially adapted, or fully adapted.

The remainder of the paper is organized as follows: The next section refines the objectives of our research. Section 3 discusses the feasibility of the approach in the scenario of web-caching and Content Distribution Networks. Related work is surveyed in section 4. Finally, concluding remarks and future directions of our work are given in section 5.

---

[1] By the term replication we refer to all techniques maintaining multiple distributed physical instances of a data object. Thus, content networking comprises both web-caching by proxy servers and content replication by Content Distribution Networks.

2 Cost-Quality Optimized Content Networking

The objective of our work is efficient content delivery to a heterogeneous client population by exploiting content networking technologies. The term content networking covers all technologies applying replication to improve the delivery of content. Within content networks objects are replicated at nodes of the content network, so-called surrogates, and delivered from those nodes upon request from a client [2].

The heterogeneity of the client population is addressed by content adaptation. By content adaptation, multiple representations r of each object k can be generated by performing adaptation operations on an origin representation r_0 of k as provided by the origin server or on derived representations previously generated from r_0. We assume adaptation operations to be performed within the content network either by the surrogates themselves or by separate adaptation servers that are invoked by the surrogates (cf. section 3). By this means, objects may be converted from one representation into another one within the content network.

In general, content adaptation comprises all operations targeted at adapting the representation of information to the application context. Besides operations converting a single object from one representation into another one, an additional important class of adaptation operations are structural adaptations. By structural adaptations, objects are composed from multiple objects (e.g. composition of layered encoded streams [3]) or decomposed into sets of objects (e.g. document fragmentation [4]). For the sake of simplicity, we restrict our considerations to conversion operations, transforming a single object from one representation into another single object representation. The class of conversion operation is very comprehensive, comprising powerful operations such as: format transcoding, scaling, lossy compression, media conversion, semantic conversion.

Moreover, we exclude user model based adaptations [5], such as personalized information filtering, as maintaining user models is beyond the scope of content networks. Also, user model based adaptations is often application specific whereas the content network is meant to be generic. We assume user model based adaptations to be performed on the server resulting in personalized documents. One observes that a growing number of services in the Web produces personalized hypertext documents, being improper for replication in content networks[2]. Hence, we exclude hypertext documents from our considerations. On the other hand, multimedia content portions (e.g. images, audio, video) are typically not personalized; only the selection of those objects by the referring hypertext document is adapted to a user model. Thus multimedia objects are perfect candidates to be adapted within content networks.

A representation r is defined by a m-tuple $(m_1,\cdots,m_m)$ of media features as specified by the IETF Network Working Group as Media Feature Collections [6]. Hence, an adaptation operation is a mapping $o: R \rightarrow R$ (where R is the set of possible representations) from a m-tuple $(m_1,\cdots,m_m)$ of media features onto another m-tuple $(m_1',\cdots,m_m')$. Accordingly, a client c specifies its capabilities to handle objects in certain representations, its media handling capabilities, by a predicate over the media features $(m_1,\cdots,m_m)$ as specified by the IETF Media Feature Sets [6]. Formally, the media handling capabilities of a client c define a set $MHC_c \subseteq R$ of representations that can be handled by the client. A client c must receive an object k in response to the i^{th} request for k from c only in a representation $r_{cki} \in MHC_c$.

Among the representations in MHC_c the optimal one in terms of quality and cost as outlined in the following has to be selected by the content network and delivered to the client. From the clients' perspective content should be delivered as prompt as possible in the representation that best matches the media handling capabilities of the device. Thus, the optimality criterion

[2] Distributed delivery of dynamically generated, personalized hypertext documents from surrogates demands for mirrored application server logic that is beyond the scope of content networks as considered in this paper.

from the clients' perspective is the quality of service in terms of the deviation of the received representation from the representation leveraging the capabilities of the device optimally and the promptness of the response.

The quality of the representation r_{cki} of an object k delivered to a client c is captured by the quality function $qual_{ck}(r_{cki})$ mapping the representation r_{cki} to a quality value within the closed interval [0,1]. A quality value of 1 refers to the maximum achievable quality or more precisely a quality indistinguishable from maximum quality[3]. Zero quality ($qual_{ck}(r_{cki}) = 0$) means, the object is useless to the client. An object should never be delivered with zero quality. For instance, an image with a resolution that does not allow the user to recognize any information in the image has zero quality. Similarly, a threshold for unacceptably long response delays may be assigned zero quality.

Obviously, the notion of quality of a representation comprises multiple dimensions, e.g. spatial resolution, color resolution, peak signal to noise ratio (PSNR), and response delay. Defining a multidimensional quality function, mapping the characteristics of the multiple dimensions onto a scalar quality value, is non-trivial. For this reason, we employ the concept of dimensional quality functions, mapping the characteristics in one quality dimension d onto a dimensional quality value $qual_{ckd}(r_{cki})$, as proposed by [7]. The dimensional quality values are eventually weighted (by the dimensional weights w_d) and combined to an overall quality value $qual_{ck}(r_{cki})$ by geometric mean calculation:

$$qual_{ck}(r_{cki}) = \prod_d qual_{ckd}(r_{cki})^{w_d} \quad \text{where} \quad \sum_d w_d = 1$$

The quality $qual_{ck}(o(r))$ of the representation $o(r)$ of object k to client c resulting from the application of an adaptation operation o to a source representation r may and typically will differ from the quality of the source representation $qual_{ck}(r)$. For the sake of tractability, we assume quality to be monotonically decreasing with the invocation of each adaptation operation o, i.e.

$$\forall o. \forall c. \forall k.\ qual_{ck}(o(r)) \le qual_{ck}(r).$$

Admittedly, there are operations that may result in an increase of user perceived quality, e.g. the computational reduction of JPEG compression artefacts [8]. Nonetheless, the above assumption holds for almost all relevant operations.

From the provider's perspective the service imposes costs by resource consumption, e.g. paid bandwidth at its gateways to the public Internet or processing costs for adaptation operations[4]. Costs for providing an object k to a client c in response to the i^{th} request for k from c is captured by the cost value $cost_{cki}$, where $0 \le cost_{cki}$. This might be both, the monetary cost or some abstract cost measure. The cost $cost_{cki}$ is the sum of costs for all individual adaptation operations and data transfers necessary to deliver the object k to client c in response to the i^{th} request for k from c. As the costs for individual adaptation operations and data transfers are non-negative, costs increase monotonically with the invocations of adaptation operations or data transfers. The overall costs

$$cost_{all} = \sum_c \sum_k \sum_i cost_{cki}$$

arising from servicing all requests i for all objects k by all clients c are to minimize.

---

[3] If lossy compression techniques are applied, the compression results in a loss of entropy in the object. Nevertheless, as long as the loss of entropy cannot be perceived by a human observer the compressed image is assigned a quality value of 1.

[4] For the sake of tractability, we consider adaptation operations to be executed by uncapacitated adaptation services imposing costs and delay for the operation regardless of the load. Likewise, overall bandwidth is uncapacitated but subject to communication costs whereas the bandwidth for a single object transfer is restricted resulting in communication latency.

The storage capacity SC_n at each surrogate n in the content network is restricted. The objective is effective utilization of this resource. In the problem definition, the storage capacity at each surrogate n accounts for the constraint

$$\sum_{(k,n,r)\in P} size_{kr} \leq SC_n ,$$

where $size_{kr}$ is the size of object k in representation r and the set P describes the placement of replicas as $(k,n,r)\in P$ if and only if k is stored at n in representation r.

Finally, the quality of all responses for the entire client population should be maximized to ensure customer satisfaction. This goal conflicts with the objective of minimized overall costs. The tradeoff motivates a cost-quality optimized approach for content delivery. In order to make the problem tractable, we project quality values into the cost domain. This is accomplished by assigning revenue to quality. The revenue value rev_{cki} corresponds to the revenue that can be yielded by delivering object k in representation r_{cki} with quality $qual_{ck}(r_{cki})$. The overall revenue

$$rev_{all} = \sum_c \sum_k \sum_i rev_{cki}$$

yielded by responding to all requests i for all objects k requested by all clients c, is to be maximize. Let

$$rev_{cki} = p_k \cdot qual_{ck}(r_{cki})$$

where p_k is the nominal price of object k. The nominal price p_k may differ with different objects, e.g. a video stream may have a certain nominal price per minute resulting in distinct p_k with different stream lengths. However, p_k is defined to be constant for all representations of an object. Likewise, p_k is assumed to be independent of the client. Nonetheless, price discrimination, i.e. client dependent nominal prices, is conceivable resulting in different service classes.

Based on the above definitions we define the optimization problem as follows:

$$\begin{aligned} \text{minimize} \quad & cost_{all} - rev_{all} \\ \text{subject to} \quad & \forall n. \sum_{(k,n,r)\in P} size_{kr} \leq SC_n \\ & \forall c.\forall k.\forall i.(r_{cki} \in MHC_c) \wedge (qual_{ck}(r_{cki}) > 0) \end{aligned}$$

This problem is NP-hard. One can simply find a specialization of the problem that is homomorphic to the well-known knapsack problem (proof omitted due to space constraints). Hence, a scalable realization depends on heuristics to approximate the optimal solution.

3 Web-Caching vs. Content Distribution Networks

Both adaptation-aware web-caching and adaptive Content Distribution Networks (CDN) are promising technologies for improving the performance of information access in Pervasive Computing environments. In previous work [1], we outlined an architecture of independent caching proxies for adaptation-aware web-caching. From this approach, we learned that the requirements of Cost-Quality Optimized Content Networking violate the scalability of web-caching approaches. This shall be illustrated in the following.

In conformity with section 2, a request may be satisfied by multiple representations $r_{cki} \in MHC_c$. Furthermore, each of the possible representations may be generated from different representations by applying one or successively multiple adaptation operations.

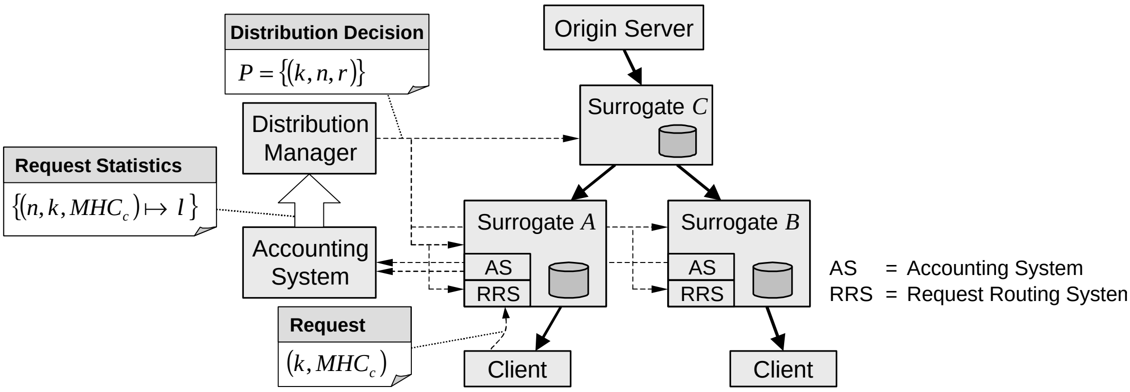

Figure 1. Adaptive Content Distribution Network: System Architecture

Hence, there are potentially various different representations that might be cached in order to fulfill the request. It is up to the proxy to decide which representation to cache. Therefore the proxy has to determine a rating for each representation *r* according to the quality experienced by the client and the costs for the response in case *r* is cached. Likewise, the proxy must calculate a rating for the case in which the object is not cached at all. With web-caching the decision making is necessary with every single request resulting in intolerable response delays. Furthermore, the proxies must hold a table of ratings for all possible representations *r* of all objects *k* that might be cached wasting storage capacity for meta information in all proxies. Hence, we consider the web-caching approach as infeasible especially if the request rate and the number of potentially cacheable representations is high.

As opposed to web-caching, the placement of objects in the surrogates of a CDN is stable for a certain period of time, so-called epoch. Thus, only the best adaptation paths with respect to a predetermined placement has to be calculated with every request. This problem is considerably less complex. The cache placement is determined only once per epoch. Hence, the runtime of the placement algorithm is not critical for the response time.

An appropriate algorithm for determining the optimal adaptation path with respect to a predetermined placement can be constructed based on the concepts presented by Choi et al. [9]. The basic idea is mapping the different possible representations *r* of an object *k* on the different surrogates *n* onto virtual nodes (*n*,*r*) in a virtual graph. By this means, the adaptation paths search is simply a multiple-sources, single-destination shortest paths search problem in the virtual graph.

As the complexity of such an algorithm depends quadratically on the potential number of nodes in the adaptation path, we want to restrict the potential number of nodes by imposing a hierarchical structure on the surrogates in the content network (cf. fig. 1). Hence, the number of physical nodes traversed by an object is bounded to the depth of the hierarchy. Likewise, the hierarchy structure reduces the number of potential surrogates to retrieve a replica from. Thus, global knowledge about the entire content network is unnecessary for calculating the optimal adaptation path. We delegate the responsibility of calculating the adaptation path for current requests to the base-level surrogates, holding placement information about the surrogates in the physical path to the root. They receive all requests from their local client population, thereby realizing an application-layer request routing with the local base-level surrogate acting as an in-path element (cf. [10]). The request redirection to the base-level surrogates must be performed by transparent interception of the request (as described e.g. by [11]) to enable the mobile clients to contact the nearest base-level surrogate.

The global algorithm determining the placement of objects is assumed to be performed by a component of the CDN's distribution system[5], the distribution manager (cf. fig. 1). It receives aggregated request statistics by the accounting system of the CDN. The statistics are used to estimate the request pattern in the next epoch. Based on the estimates, the distribution manager determines the optimal placement by a replica placement algorithm run once per epoch. Currently, we are about to evaluate replica placement algorithms with respect to their applicability in this scenario. The placement decision is eventually distributed to the surrogates in the CDN and deployed with the beginning of the next epoch. Since in accordance with the request routing mechanisms described above, each base-level surrogate needs information about the occupation of its ancestor surrogates, the distribution of the placement decision in the CDN must be well considered to deal efficiently with the network bandwidth. We envision to send the placement decision for the base-level surrogates individually from the distribution manager to the surrogates. As opposed to that, the placement decision for the higher-level surrogates is distributed by reliable multicast mechanisms using the hierarchy of surrogates for application-level multicast.

The actual implementation of the new placement, i.e. uploading and storing the objects in the proper representation according to the placement decision, may be done by push or pull replication. Push replication means that objects are uploaded to the surrogates immediately at epoch changeovers, generating high network and adaptation load at epoch changeovers. For this reason, it is applicable only if epochs start in periods of low client load. In contrast to that, uploading is deferred to the first request for a replica within an epoch using pull replication. Thus, the load is distributed throughout the epoch. The penalty for distributing the load is an additional delay with every first request within an epoch.

4 Related Research

Related research to our work spans multiple research areas, mainly (1) content adaptation, (2) replica placement, and (3) quality-adaptive multimedia caching. Content adaptation research is a very broad research area. The most interesting work with respect to our approach deals with the composition of adaptation paths. The notion of distributed adaptation paths has been introduced by Ninja Paths [12]; but path optimization has not been considered. An algorithm composing optimal adaptation paths in collaborative multimedia systems is introduced by [13]. Another solution has been presented in [9], mapping path optimization onto a shortest path problem in layered graphs. As opposed to [13], [9] has not considered multiple replicas nor cost-quality optimized path composition. However, an extension of the approach by [9] leads to a similar solution as the one presented by [13] but with a complexity that is polynomially bounded with respect to the potential number of representations and nodes in the adaptation path. We are about to utilize this method to determine optimal adaptation paths with respect to a predefined placement.

Replica placement deals with the allocation of object replicas to distributed sites. The problem is proven to be NP-hard [14]. That is why several heuristic approaches has been evaluated (see [15] for a survey). We are currently evaluating replica placement algorithms with respect to their applicability in the context of cost-quality optimized content networks.

Caching of quality adaptive multimedia streams [3, 16] is related to our approach as streams may be cached in different representations (qualities). As opposed to our ideas, considerations are restricted to adaptations by dropping layers or frames. Multidimensional adaptations are out of scope. In addition, this work is focused on single caching proxies. Cooperation between different surrogates within a content network is not considered.

[5] For a description of reference system components of a CDN refer to [17].

5 Conclusion

This paper has discussed the application of content networking in the domain of Pervasive Computing where content adaptation is needed to adjust the representation of contents to heterogeneous clients. We have pointed out that content adaptation interferes with replication in content networks, demanding for novel schemes coordinating replication and adaptation.

The first part of the paper has introduced the notion of Cost-Quality Optimized Content Networking. We have defined an optimization objective based on cost and quality values allowing for both, taking the providers cost for running the service and the clients' demand for qualitative service into account. Based thereon, we have formulated a problem definition for Cost-Quality Optimized Content Networking. The second part has discussed the feasibility of the approach in the scenarios of web-caching and CDNs. We have illustrated that an algorithm to determine the cost-quality optimized placement of replicas causes intolerable response delays in the web-caching scenario. Hence, we have favored the CDN approach where response delays are not impacted by the runtime of the placement algorithm. The feasibility of this approach has been further illustrated by outlining a CDN system architecture.

The presented approach is still subject to ongoing research. Currently we are evaluating replica placement algorithms with respect to their applicability in the context of cost-quality optimized content networks. Experimental results will be published in the future.

References

[1] Buchholz, S., Schill, A., "Adaptation-Aware Web Caching: Caching in the Future Pervasive Web", *Kommunikation in Verteilten Systemen (KiVS)*, Leipzig, Germany, 2003.

[2] Day, M., Cain, B., Tomlinson, G., Rzewski, P., "A Model for Content Internetworking (CDI)", *RFC 4466*, IETF Network Working Group, 2003.

[3] Kangasharju, J., Hartanto, F., Reisslein, M., Ross, K., "Distributing Layered Encoded Video through Caches", *IEEE Infocom*, 2001.

[4] Göbel, S., Buchholz, S., Ziegert, T., Schill, A., "Device Independent Representation of Web-based Dialogs and Contents", *IEEE YUFORIC'01*, Valencia, Spain 2001.

[5] Brusilovsky, P., Maybury, M., "From Adaptive Hypermedia to the Adaptive Web", *Communications of the ACM 45 (5)*, 2002.

[6] Klyne, G., "A Syntax for Describing Media Feature Sets", *RFC 2533*, IETF Network Working Group, 1999.

[7] Walpole, J., Krasic, C., Liu, L., Maier, D., Pu, C., McNamee, D., Steere, D., "Quality of Service Semantics for Multimedia Database Systems", *IFIP TC-2 Working Conference on Data Semantics*, 1999.

[8] Lazzaro, J., Wawrzynek, J., "JPEG Quality Transcoding using Neural Networks Trained with a Perceptual Error Measure", *Neural Computation 11(1)*, 1999.

[9] Choi, S., Turner, J., Wolf, T., "Configuring Sessions in Programmable Networks", *IEEE Infocom*, 2001.

[10] Barbir, B., Cain, B., Nair, R., Spatscheck, O., "Known CN Request-Routing Mechanisms", *Internet-Draft*, IETF Network Working Group, 2003.

[11] Cohen, A., Rangarajan, S., Singh, N., "Supporting Transparent Caching with Standard Proxy Caches", *4th International Web Caching Workshop*, San Diego, CA 1999.

[12] Gribble, S., Welsh, M., von Behren, R., Brewer, E., Culler, D., Borisov, N., Czerwinski, S., Gummadi, R., Hill, J., Joseph, A., Katz, R., Mao, Z., Ross, S., Zhao, B., "The Ninja Architecture for Robust Internet-Scale Systems and Services", *Computer Networks 35(4)*, 2001.

[13] Candan, K. S., Subrahmanian, V. S., Rangan, P. V, "Collaborative Multimedia Systems: Synthesis of Media Objects", *IEEE Transactions on Knowledge and Data Engineering 10(3)*, 1998.

[14] Kangasharju, J., Roberts, J., Ross, K. W., "Object Replication Strategies in Content Distribution Networks", *Web Caching and Content Distribution Workshop (WCW'01)*, Boston, MA 2001.

[15] Karlsson, M., Karamanolis, C., Mahalingam M., "A Framework for Evaluating Replica Placement Algorithms", HPL Technical Report HPL-2002-219, Hewlett Packard Laboratories, Palo Alto, CA 2003.

[16] Sehgal, A., Chou, P., "Cost-Distortion Optimized Caching Of Streaming Media", *IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, Orlando, FL 2002.

[17] Peng, G., "CDN: Content Distribution Network", *Technical Report TR-125*, Experimental Computer Systems Lab, Department of Computer Science, State University of New York, Stony Brook, NY 2003.