1986

DYNAMIC FILE MIGRATION IN DISTRIBUTED COMPUTER SYSTEMS

Olivia R. Liu Sheng
University of Arizona

DYNAMIC FILE MIGRATION IN DISTRIBUTED COMPUTER SYSTEMS

Olivia R. Liu Sheng
College of Business and Public Administration
University of Arizona

ABSTRACT

In a distributed computer system files are shared by both local users and remote users for query and update purposes. A user performing data processing activities tends to reference the same file for some time. When the referenced file is stored remotely, large amounts of communication traffic will be generated. For example, when a customer is making a travel plan, an airline reservation database might be accessed repeatedly by a remote operation site. The inquiries will probably all be made within the time of an ordinary telephone conversation.

In many recent developments in distributed computer systems, file migration operations are incorporated into the procedures for processing remote file access requests. Using file migration operations a file may be duplicated or moved to the requesting site in order to reduce communication traffic. As a result, the system is faced with dynamic file placement decisions using a file migration policy. In particular, a file migration policy is expressed as the IF-THEN rules that specify the file migration operations to be implemented at each viable system state. Based on this policy, file migration operations are triggered when the specified conditions are satisfied, and thus dynamically respond to system needs. Because of the dynamic behaviors of systems, the problem of deriving effective file migration policies is extremely complex. An elaborate analysis is required. This paper studies the impact of file migration operations on system performance and develops automatic mechanisms for incorporating file migrations as part of system operations. The mechanisms include optimization models formulated in the form of Markov decision models for deriving optimal file migration policies at system design or redesign points, and heuristic rules to generate adaptive file migration decisions for individual file access requests. The trade-off between these two types of mechanisms is clearly that of performance levels versus implementation complexities. The optimization analysis not only generates the best possible solutions, but provides insight into the problem structure, whereas the rationale for developing heuristics is their simplicity in implementation and acceptable performance levels.

This article is forthcoming in *Communications of the ACM*

The Optimization of File Migration Policies

In the system examined here, file migration operations are incorporated as part of the options for processing file access requests. Specifically, if the file requested is not available locally, the requesting site can either interact with a remote copy, following standard remote processing procedures, or perform a file migration option (e.g., file creation or file transfer) to obtain a local copy so that the transaction can be processed locally. Once cost minimization is chosen as the optimization criterion, any file migration decision should be evaluated in terms of the expected system operating cost generated by file accesses over the remainder of the planning period. This cost is clearly dependent on the current system state and state transitions in the future. With a proper representation of system states and the underlying probabilistic properties of state transitions, the expected system operating cost over the remainder of the planning period can be evaluated through a decomposition up to the next decision point for file migration. The cost incurred between the current and the next decision points is the cost of processing the current file access request. The cost is fixed once a decision on file migration is made. Cost incurred after the next decision point is obtained from the result of evaluating the next file migration decision. As a result, the current file migration decision is recursively related to the next file migration decision. It follows from the principle of optimality that when all components of an optimization criterion are optimally evaluated, the decision derived is optimal as well. Therefore, optimal file migration policies can be generated by Markov decision models which, using the recursive relationship, repeatedly evaluate file migration decisions over the planning period in reverse time order.

Since the trade-offs between file migration and static policies are optimally evaluated by developed Markov decision models, the optimal file migration policies obtained always outperform static file allocations. The degree of the performance improvement generated by file migration increases with the set of dynamic file migration operations considered by the models, and with the degree of locality of file reference.

A Simple Heuristic for Adaptive File Migration Decisions

Based on the results obtained from the numerical analysis of file migration policies, a simple heuristic is proposed for deriving file migration decisions with less computational complexity. In response to changes in the locality of file reference, the proposed heuristic decides about file migration by assuming a static future, i.e., all file copies remain in their new location following the current decision. Under this assumption, a file migration decision involving a comparison of the system operating costs resulting from all feasible file migration options is reduced to the evaluation of the costs of executing a particular option plus the costs of static file management over the remainder of the planning period. Note that a static future is assumed only for the current decision evaluation. In reality, file migration may still take place at later stages and change both file placement and system performance. Numerical results for optimal file migration policies indicate that if future file allocation is modified dynamically, it is likely that the new file allocation will move toward the original one as intensive local file reference decreases. As a result, future system behavior can be treated as if it were static. Performance of the static file allocation provides, therefore, a close approximation for that of the dynamic file allocation.

The proposed heuristic is employed to determine file migrations whenever the system is faced with the file migration decision. Since changes in file access intensities are taken into consideration by the decision-making, the derived decisions are adaptive in nature. As a result, the performance of the heuristic is linked to the accuracy of the forecasts of future file access intensities. Other considerations, such as restricted file migration implementations, anticipated file migrations and prior knowledge of future file access intensities, would also affect the effectiveness of the decisions derived by the heuristic. The results from preliminary experiments with the heuristic show satisfactory performance improvement generated by the derived file migration decisions.

Conclusion

This paper describes research that calls for more attention to the problems of file migration. File migration represents a beneficial means of improving data processing performance in distributed systems. Actual realization of this idea requires a practical and effective control policy. Optimization analysis for file migration policies provides insight into the problem structure and a sound foundation for policy derivation. Practical implementation, however, requires more investigation in simple heuristics for file migration decisions. That should be the emphasis of future research. In addition, such issues as reliability, availability and queuing phenomena in distributed systems should also be taken into consideration in order to exploit fully the potential of such policies.