Diss. ETH No. 14950

Zero Copy Strategies for Distributed CORBA Objects in Clusters of PCs

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH
(ETH ZÜRICH)

for the degree of
Doctor of Technical Sciences

presented by
Christian A. Kurmann
Dipl. Informatik-Ing. ETH
born December 5, 1970
citizen of Gossau (SG) and Hohenrain (LU), Switzerland

accepted on the recommendation of
Prof. Dr. Thomas M. Stricker, examiner
Prof. Dr. Burkhard Stiller, co-examiner

2002

Abstract

Clusters of Personal Computers (CoPs) offer the best compute performance at the lowest price. Workstations with 'Gigabit networking to the Desktop' can enable a new game of multimedia applications that benefit from higher communication bandwidth and lower latency. In order to reach the full Gigabit/s speed on normal PCs with their typically weak memory subsystems it requires either additional hardware for protocol processing or alternatively, a highly efficient software system that circumvents data copies.

In this dissertation we successfully introduced speculation techniques into system software design and managed to implement a clean zero-copy solution entirely in software that runs with commodity network interface cards (NICs) like the ubiquitous and cheap Gigabit Ethernet adapters, using the standard TCP/IP protocol and the socket API. The implementation techniques are similar to the ones that are already widely used in the hardware design of pipelined microprocessors and should be considered to be used in software as well. Measurements and statistical studies show a huge potential for such techniques to achieve better software efficiency, that means to provide in software what the hardware promises to be able to deliver.

Distributed and parallel computing is one of the major trends in the computer industry. As systems become more distributed, they also become more complex and have to deal with new kinds of problems. To answer the growing demand in distributed software, several middleware environments have emerged during the last few years. The Component Object Request Broker Architecture (CORBA) is an example of a middleware that shows the concepts used also in many of the competing standards. These environments however typically are not implemented for being used in high speed communication settings and therefore cannot deliver the performance up to the application. Furthermore these environments often lack support for "one-to-many" communication primitives; such primitives greatly simplify the development of several types of applications that have requirements for parallel processing, high availability, fault tolerance, or collaborative work.

Since the zero-copy principle is applicable and must be rigidly used to all levels of software, we extend the design from low level drivers and protocol stack implementations to middleware packages like CORBA that ease the implementation of distributed applications. We demonstrate a study of this topic using a data and compute intensive application, a real-time distributed DVD-to-MPEG4-Transcoder, that is properly modeled by parallel objects in CORBA and still strictly adheres to the zero-copy paradigm of a highly efficient software implementation running on a Cluster of commodity PCs.

Kurzfassung

Clusters of Personal Computers (CoPs) liefern exzellente Rechenleistung zu tiefem Preis. Auch Computernetzwerke basierend auf modernen, allgemein verfügbaren Technologien bieten immer höhere Bandbreiten und kleinere Latenzzeiten. Um aber tatsächlich Bandbreiten von einem Gigabit/Sekunde in realen Anwendungen zu erreichen, benötigen solche Cluster umfangreiche Hardwareunterstützung oder alternativ hochoptimierte Softwaresysteme, welche Kopien im limitierenden Memorysystem der Maschinen vermeiden.

Wir verwenden spekulative Methoden, die auch in der Prozessoroptimierung gebräuchlich sind. Damit gelingt es uns, auch die bis anhin unumgängliche letzte Datenkopie in einem TCP/IP-Stack zu eliminieren und dadurch mit existierenden, einfachen Gigabit Ethernet Adaptern effiziente Zero-Copy-Kommunikation zu realisieren. Messungen und statistische Auswertungen zeigen, dass solche spekulativen Techniken auch in Software ihre Berechtigung haben und ein riesiges Potential zur Effizienzsteigerung ausspielen können. D.h. sie erlauben Implementationen, welche die von der Hardware versprochenen Leistungen auch in Software der Applikation zur Verfügung stellen können.

Verteilte und parallele Systeme sind mitunter einer der Haupttrends in der aktuellen Computer Industrie. Aber während die Systeme immer mehr verteilt ablaufen, werden sie gleichzeitig auch immer komplexer und kämpfen mit neuartigen Problemen. Entsprechend der grossen Nachfrage nach Unterstützung beim Verteilen von Prozessen wurden deshalb in den letzten Jahren einige sogenannte Middleware Umgebungen entwickelt. Die Component Object Request Broker Architecture (CORBA) ist ein Beispiel, welches viele ähnliche Konzepte, die auch deren Konkurrenzprodukte benutzen, vereint. Solche Umgebungen sind aber leider oft nicht für Hochgeschwindigkeitsnetze konzipiert und können daher die exzellenten Kommunikationsleistungen der Hardware nicht nutzen. Im weiteren fehlt es oft an “one-to-many”-Primitiven. Diese sind aber gerade die Voraussetzung zur Unterstützung des Entwicklungsprozesses von Applikationen, welche parallel laufende Prozesse, hohe Verfügbarkeit und Fehlertoleranz benötigen.

Da das Zero-Copy-Konzept auch auf andere Software-Schichten als das Betriebssystem anwendbar ist, zeigen wir, wie die Zero-Copy-Fähigkeit auch in Middlewareplattformen wie CORBA realisiert werden kann. Wir demonstrieren die Konzepte und Resultate dieser Thematik mit einer daten- und rechenintensiven, verteilten MPEG2-zu-MPEG4 Transkodierungs-Applikation. Diese wurde mittels CORBA und parallelen Objekten modelliert und profitiert trotzdem von der hoch optimierten Cluster-Plattform.