An MPI Implementation on the Top of the Virtual Interface Architecture*

Massimo Bertozzi, Franco Boselli, Gianni Conte, and Monica Reggiani

Dipartimento di Ingegneria dell’Informazione
Universitá di Parma, I-43100 Parma, Italy
{bertozzi,boselli,conte,reggiani}@ce.unipr.it

Abstract. This paper describes an implementation of the LAM MPI suite on the top of the Virtual Interface Architecture. The Virtual Interface Architecture (VIA) is an emerging standard designed by Intel, Microsoft, and Compaq aimed at the reduction of communication latency for cluster of workstations or system area networks. Thanks to M-VIA, a Linux software module that emulates VIA and provides programmers of VIA APIs, it has been possible to develop an MPI implementation even in absence of a hardware VIA interface. Nonetheless, M-VIA module high optimization as well as VIA protocol simplicity permitted a reduction of latency time with respect to the use of the TCP/IP protocol on Ethernet network interfaces.

1 Introduction

Thanks to the widespread of NOW prototypes [1], message passing is going to be the most used communication paradigm for parallel machines. Standard programming environment and libraries have been designed to enable the programmers to write fully portable parallel code. Amongst others, the Message Passing Interface (MPI) has rapidly received large acceptance because it has been carefully designed to maximize performance on a large variety of systems [8].

Unfortunately, the use of MPI on cluster of workstations generally relies on OS services such as the TCP/IP protocol. This has an obvious disadvantage, mainly the high latency associated to the requirements that TCP enforces in order to work in almost all conditions across highly heterogeneous platforms and WANs. The main advantage is that the programmer is automatically endowed of a truly multi user, reliable programming environment. In addition, most NOW prototypes rely on off-the-shelf network solutions like Ethernet (generally Fast– or, even, Giga–Ethernet), switched Ethernet, ATM, or FDDI. While these solutions provide a good price/performance ratio from the point of view of the bandwidth, they require the operating system intervention for dispatching messages from and to the applications, therefore introducing another latency in the link between application and network.

* The work described in this paper has been carried out under the financial support of the Italian Ministero dell’Università e della Ricerca Scientifica e Tecnologica (MURST) in the framework of the MOSAICO (Design Methodologies and Tools of High Performance Systems for Distributed Applications) Project.
Many solutions to these problems have been proposed, ranging from the development of hardware components dedicated to speedup communications [2, 6, 3] to the implementation of light communication protocols [11, 15, 10]. Nevertheless, no one of the proposed solutions has gained a wide acceptance.

Starting from previous experiences [16], a consortium composed by Intel, Compaq, and Microsoft introduced a new standard for cluster of workstation: the Virtual Interface Architecture (VIA), that seems the most promising solution for reducing the communication latency and speeding up the application–network link. The VIA standard, in fact, endorses a direct interface between user applications and network layer without requiring participation of the operating system. The first products based on the VIA standard are now appearing and the proposing consortium seems strong enough to guarantee a wide acceptance of this network standard.

This paper presents an implementation of the LAM MPI suite on the top of the Virtual Interface Architecture. Thanks to M-VIA, a Linux software module that emulates VIA and provides programmers of VIA APIs, it has been possible to develop an MPI implementation even in absence of a hardware VIA implementation.

This paper is organized as follows. Section 2 discusses communication issues and introduces VIA. Section 3 presents the implementation, while section 4 discusses the results and performance of the final product. Section 5 ends the paper with some final remarks.

2 The Communication Architecture

An ideal communication interface/protocol for cluster of workstations or system area networks should feature the following characteristics: low communication latency, high bandwidth, small overhead, and small CPU intervention.

A number of solutions have been proposed to overcome these problems: U-Net [16], Active Messages [17], or Fast Messages [14]. These solutions share the underlying idea of giving to user processes direct access to network resources. Unfortunately, none of the above solutions has reached a wide acceptance. More recently, starting from these experiences, a consortium composed by Microsoft, Compaq and Intel authored a new standard, the Virtual Interface Architecture specifically designed for fast interprocess communication in a cluster or System Area Network environment.

2.1 The Virtual Interface Architecture

The basic concepts that have been used for the VIA design are the following:

- the process has to execute a small number of operations for starting, ending, or managing a communication session;
- these operations have to be as simple as possible;
- I/O operations must not require the use of interrupts;
- context switches as well as copies of data have to be avoided whenever possible;
- concurrency for the communication interface must be managed without operating system intervention.