Computers and Electrical Engineering 38 (2012) 1409–1432

Contents lists available at SciVerse ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

The xDotGrid native, cross-platform, high-performance xDFS file transfer framework ☆

Alireza Poshtkohi, M.B. Ghaznavi-Ghoushchi *

Department of Electrical Engineering, Shahed University, Persian Gulf Highway, Tehran 3319118651, Iran

ARTICLE INFO

Article history:
Received 16 May 2011
Received in revised form 13 April 2012
Accepted 13 April 2012
Available online 1 June 2012

ABSTRACT

In this paper we introduce and describe the highly concurrent xDFS file transfer protocol and examine its cross-platform and cross-language implementation in native code for both Linux and Windows in 32 or 64-bit multi-core processor architectures. The implemented xDFS protocol based on xDotGrid.NET framework is fully compared with the Globus GridFTP protocol. We finally propose the xDFS protocol as a new paradigm of distributed systems for Internet services, and data-intensive Grid and Cloud applications. Also, we incrementally consider different developmental methods of the optimum file transfer systems, and their advantages and disadvantages. The vision of this paper tries as possible to minimize the overhead concerned with the file transfer protocol itself and to examine optimal software design patterns of that protocol. In all disk-to-disk tests for transferring a 2 GB file with or without parallelism, the xDFS throughput at minimum 30% and at most 53% was superior to the GridFTP.

1. Introduction

As a fundamental result of information age, communication mechanisms and data transmission protocols provide an infrastructural foundation for the emergence and evolution of enormous computing paradigms, and integrating data access on a world-wide scale. The two open standards protocols of HTTP [1] and FTP [2] have provided basic file transfer functionalities. To overcome the problems concerned with these two protocols that are mainly due to the overheads of the TCP protocol in its window-based congestion control mechanisms used, the GridFTP protocol [3–5] has been proposed. In [6–12], we introduced a hybrid concurrent file transfer protocol, called as DotDFS, integrated with a set of event-driven and threaded-based models. DotDFS was the first file transfer protocol that, in addition to propose a new computing paradigm in the field of data transmission protocols, unveiled many architectural problems regarding the FTP and GridFTP protocols.

The TCP protocol has been used as a transport-level communication protocol on the Internet over the years. However, TCP is a rather old communication protocol designed in the 1970s. Many problems regarding the TCP have been reported such as its debility to support the increasing speeds of modern networks. One commonly-used way to reduce the overheads posed by TCP is to simultaneously choose an optimum number of TCP connections and the TCP socket buffer size, which are discussed fully in [6–9]. This paper tries as possible to minimize the overheads concerned with the file transfer protocol itself and to examine optimal software design patterns of that protocol. This goal plays a key role to reduce the problems associated with TCP overheads which decrease the throughput of the file transfer system. It increases dramatically the entire system efficiency and reliefs exposed drawbacks. However, it is necessary noting that the XDSI structure (refer to the Sections 3.1 and 3.2) allows xDFS protocol to operate over more optimum transport protocols (e.g., SCTP [21]) than TCP. XDSI and the

☆ Reviews processed and approved for publication by Editor-in-Chief Dr. Manu Malek.

* Corresponding author.
E-mail addresses: alireza.poshtkohi@gmail.com (A. Poshtkohi), ghaznavi@shahed.ac.ir (M.B. Ghaznavi-Ghoushchi).

0045-7906/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compeleceng.2012.04.007