

ReMDoR 2.0:
REMOTE MULTIMEDIA DOCUMENT RETRIEVAL
OVER PARTIALLY-ORDERED, PARTIALLY-RELIABLE
TRANSPORT PROTOCOLS

by

Armando L. Caro Jr.

A thesis submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer and Information Sciences with Distinction.

Spring 1988

Copyright 1998 Armando L. Caro Jr.
All Rights Reserved

ReMDoR 2.0:
REMOTE MULTIMEDIA DOCUMENT RETRIEVAL
OVER PARTIALLY-ORDERED, PARTIALLY-RELIABLE
TRANSPORT PROTOCOLS

by

Armando L. Caro Jr.

Approved: _____
Paul D. Amer, Ph.D.
Professor in charge of the thesis on behalf of the Advisory Committee

Approved: _____
Chandra Kambhamettu, Ph.D.
Committee member from the Department of Computer and Information
Sciences

Approved: _____
Michael Keefe, Ph.D.
Committee member from the Board of Senior Thesis Readers

Approved: _____
Kevin Kerrane, Ph.D.
Chair of the University Committee on Student and Faculty Honors

ACKNOWLEDGMENTS

I am grateful to my advisor, Dr. Paul D. Amer. Without him, my research would have been non-existent. He has not only given me the opportunity to do my undergraduate research, but he has also made it worthwhile. He continually challenged and encouraged me to learn more and to participate in conferences and other research activities that were not required of me.

I also thank Dr. Chandra Kambhamettu and Dr. Michael Keefe for serving in my senior thesis committee.

I appreciate the useful input and contribution provided by my Protocol Engineering Lab graduate student colleagues: Phill Conrad, Ed Golden, Sami Iren, Gul Sezen, and Mason Taube. I give a special thanks to Phill, who has served as a mentor for me throughout my research. He went out of his way to guide, advise, and assist me in many areas where I was inexperienced. Without his help, I would have had a greater struggle making my research a success.

I thank the Undergraduate Research Program for making research available to me at the undergraduate level. The incorporation of research into my undergraduate education has made it more fulfilling and more rewarding. The experience has also helped me realize that graduate school and a career in research should be my focus. I especially thank Sue Serra for her consistent and helpful guidance along the way

I am extremely thankful for my family's support. I particularly thank my parents, who have consistently challenged me to perform my best throughout my

undergraduate career. They have also served as a source of guidance and advice whenever I needed it; I appreciate their enthusiasm and devotion to helping me in every way possible.

Finally, I thank my girlfriend, Michelle. She has been very supportive and understanding, even through the periods when there was little time to spare for her.

This research has been financially supported, in part, by the Advanced Telecommunication and Information Distribution Research Program (ATIRP) Consortium sponsored by the U. S. Army Research Laboratory under the Federated Laboratory under the Federated Laboratory Program, Agreement DAAL01-96-2-0002.

TABLE OF CONTENTS

LIST OF FIGURES	vi
LIST OF TABLES	vii
ABSTRACT.....	viii
 Chapter	
1 INTRODUCTION AND MOTIVATION	1
1.1 Transport Layer	1
1.2 Traditional Protocols.....	2
1.3 Internet Application Developer's Dilemma.....	2
1.4 POCv2 as a Solution.....	3
1.5 ReMDoR: POCv2's Test System.....	3
2 ReMDoR	6
2.1 Architecture	6
2.2 Temporal Documents.....	8
2.3 Authoring ReMDoR Documents	8
2.3.1 Prototype Multimedia Specification Language (PMSL).....	9
2.3.2 PMSL To PMFF Compiler (PTPC)	12
2.3.3 Prototype Multimedia File Format (PMFF)	14
2.4 Server	15
2.5 Browser (Client)	16
3 CONCLUSION AND FUTURE WORK	23
 APPENDIX	
A PMSL 2.0 SYNTAX	26
B PMFF FILE STRUCTURE	30
REFERENCES	33

LIST OF FIGURES

Figure 1	Spectrum of PO/PR Services	4
Figure 2	Architecture of ReMDoR 2.0	7
Figure 3	Example of PO Graph for a Multimedia Presentation	10
Figure 4	(A) Original PO Graph (B) Transitively Reduced PO Graph.....	15
Figure 5	Changes in Browser's User Interface	18
Figure 6a	Snapshot 1 of Front Queue Processing.....	20
Figure 6b	Snapshot 2 of Front Queue Processing.....	20
Figure 6c	Snapshot 3 of Front Queue Processing.....	21
Figure 6d	Snapshot 4 of Front Queue Processing.....	21
Figure 6e	Snapshot 5 of Front Queue Processing.....	22
Figure 6f	Snapshot 6 of Front Queue Processing.....	22
Figure 6g	Snapshot 7 of Front Queue Processing.....	23
Figure 6h	Snapshot 8 of Front Queue Processing.....	23
Figure 7	Testing Using the Lossy Router	26

LIST OF TABLES

Table 1	Division of Work.....	24
---------	-----------------------	----

ABSTRACT

Today's Internet provides a choice between two transport protocols: Unit Datagram Protocol (UDP) and Transmission Control Protocol (TCP). These protocols present extremes in terms of quality of service (QoS). UDP's service is unordered, unreliable, maybe-duplicates, and not flow-controlled; TCP's service is ordered, reliable, no-duplicates, and flow-controlled. The QoS needed by different applications varies greatly, creating a dilemma due to the limited nature of the available transport protocols. When an application has to choose either TCP or UDP when neither is appropriate, negative consequences result. Ideally, a flexible transport service would exist that provides varying degrees of quality of service, so that each application could get the quality of service it requires.

To address this need, the Protocol Engineering Lab at University of Delaware has developed and implemented a new transport protocol, Partial Order Connection version 2 (POCv2), which provides a spectrum of partially-ordered, partially-reliable services. POCv2 is a transport service ideal for applications, such as multimedia applications, that need flexible control over the ordering and reliability of individual objects. In addition, POCv2 offers an extra service that neither TCP nor UDP provide: a mechanism that facilitates synchronization of multimedia objects.

To test the benefits of POCv2, a system named ReMDoR (Remote Multimedia Document Retrieval) was developed with POCv2 features in mind. ReMDoR is a client/server system with interactions similar to those used between World Wide Web browsers and HTTP servers. ReMDoR multimedia documents have

order and reliability requirements that are incorporated directly into the documents during the authoring process. ReMDoR allows for comparative testing between POCv2 and traditional protocols such as TCP and UDP.

The presented work has contributed to the development of the current version of ReMDoR, which is in essence the second generation of the system. ReMDoR 2.0's client/server system is being rebuilt to improve it from both a user perspective and a research perspective. From a user perspective, improvements to the system include: (1) a more user-friendly browser, (2) a more efficient server for faster document retrieval, (3) a browser message field showing connection progress, and (4) better document presentation because of progressive image display. From a research perspective, improvements include: (1) faster software development due to modularization of code and better debugging tools, (2) easier gathering of experimental data with a scripting ability to automate experiments, (3) easier integration of new image compression algorithms, and (4) better plug-in capabilities for new transport protocols.