Modelling Techniques and Tools for Performance Analysis

edited by
D. Potier

MODELLING TECHNIQUES AND TOOLS FOR PERFORMANCE ANALYSIS

Proceedings of the International Conference on
Modelling Techniques and Tools for Performance Analysis
Paris, France, 16–18 May, 1984

organised by the Institut National de Recherche en Informatique et en Automatique (INRIA)

edited by

D. POTIER
INRIA
Domaine de Voluceau
Rocquencourt
Le Chesnay, France

1985

NORTH-HOLLAND
AMSTERDAM · NEW YORK · OXFORD

ISBN: 0 444 87696 0

Published by:
ELSEVIER SCIENCE PUBLISHERS B.V.
P.O. Box 1991
1000 BZ Amsterdam
The Netherlands

Sole distributors for the U.S.A. and Canada:
ELSEVIER SCIENCE PUBLISHING COMPANY, INC.
52 Vanderbilt Avenue
New York, N.Y. 10017
U.S.A.

Library of Congress Cataloging in Publication Data

International Conference on Modelling Techniques and Tools for Performance Analysis (1984 : Paris, France)
Modelling techniques and tools for performance analysis.

1. Digital computer simulation--Congresses. 2. Electronic digital computers--Evaluation--Congresses. I. Potier, D. II. Institut national de recherche en informatique et en automatique (France) III. Title.
QA76.9.C65I55 1984 001.4'34 84-28640
ISBN 0-444-87696-0

PRINTED IN THE NETHERLANDS

FOREWORD

These proceedings contain thirty-five invited or selected papers presented at the International Conference on Modelling Techniques and Tools for Performance Analysis organized by INRIA with the sponsorship of AFCET, CNET and CISI, and held in Paris in May 1984. The aim of the Conference was twofold: we wished to promote the most recent advances in the field of modelling techniques and tools and we wanted to present a spectrum of the specialized modelling products which are now reaching industrial and technical maturity.

The Conference included regular sessions and also an exhibition with technical presentations and alive demonstrations of commercially available performance tools from the following companies:

- AMDAHL Corp.	(USA)
- B.G.S. Systems	(USA)
- BULL SIMULOG	(France)
- C.E.P.	(France)
- DUQUESNE System	(France)
- I.B.M. Corp.	(USA)
- MORINO Associates	(UK)
- NEC Corporation	(Japan)
- SAS Institute	(France)
- SODETEG-TAI	(France)
- SYSECA	(France)
- THOMSON-TITN	(France)

More than 75 full-length papers were submitted and the program committee selected 30 papers for presentation at the Conference. The primary concern was for quality and relevance. Each submitted paper was reviewed by two members of the program committee and two external referees. Besides, the program committee initiated five invited papers on topics or contributions regarded as of major importance with respect to the Conference objectives.

As a whole the Conference and the exhibition stressed the evolution of performance evaluation techniques towards an increase use of analytical approaches through efficient tools. This evolution is illustrated for example by the developments of tools like BEST-1/SN by BGS Systems and by the experience in mainframe system modelling at Amdahl Corp. Also, several general purpose modelling packages presented during the Conference such as RESQ2 (IBM), QNAP2 (Bull, INRIA and SIMULOG), SNAP (University of Stellenbosch), COPE (University of Erlangen, Nixdorf and Siemens), PERFORMS (NEC Corp.), etc. implementing a wide range of analytical techniques offer a cost/efficiency ratio better than standard simulation techniques by several orders of magnitude.

The proceedings are organized into three parts. The first part is devoted to modelling tools with two introductory invited papers. In the second part, case studies sessions on communication systems, performance analysis of multiprocessor architectures and workload characterization are presented. Two invited papers on mainframe system modelling and communication network modelling introduce this section. Methodological aspects are discussed in the third part of the proceedings, with an invited paper on performance and realiability issues and sessions on Petri nets, queueing systems and simulation.

I cannot end this brief introduction without expressing my gratitude to the members of the program committee and to the external referees, and my thanks to the conference secretaries, Sylviane Gosset and Myriam Guilloteau for their invaluable assistance during the course of the organization of the program and of these proceedings

D. POTIER
Conference Chairman

ORGANIZING COMMITTEE

N. ABU EL ATA (CISI, Gif-sur-Yvette, France)
J.F. OMNES (CNET, Lannion, France)
D. POTIER (INRIA, Rocquencourt, France)
G. PUJOLLE (Université Paris VI / INRIA, Rocquencourt, France)

PROGRAM COMMITTEE

N. ABU EL ATA (CISI, Gif-sur-Yvette, France)
F. BACCELLI (INRIA, Rocquencourt, France)
G. BALBO (University of Turin, Italy)
H. BEILNER (Universität Dortmund, FRG)
J.-P. BUZEN (B.G.S., Cambridge, USA)
P. CASEAU (EDF, Paris, France)
P.-J. COURTOIS (Philips Lab., Brussels, Belgium)
P. DENNING (Purdue University, USA)
Ph. DE RIVET (Machines BULL, Paris, France)
D. FERRARI (University of California, Berkeley, USA)
S. GRAHAM (University of Toronto, Canada)
T. HASEGAWA (University of Kyoto, Japan)
G. HEBUTERNE (CNET, Paris, France)
A.E. KRZESINSKI (University of Stellenbosch, South Africa)
S.S. LAVENBERG (IBM, Yorktown Heights, USA)
T. MASUDA (University of Tsukuba, Japan)
D. MITRA (Bell Laboratories, Murray Hill, USA)
J.-F. OMNES (CNET, Lannion, France)
R. PUIGJANER (Faculté de l'Informatique, Barcelona, Spain)
G. PUJOLLE (Université Paris VI / INRIA, Rocquencourt, France)
A. RINNOOY KAN (Erasmus Universiteit, Rotterdam, The Netherlands)

CONTENTS

A.2 Numerical tools

A.3 Simulation and hybrid tools

B. MODELLING STUDIES

Invited papers

B.1 Communication systems

B.2 Multiprocessor architectures

B.3 Workload characterization

C. METHODOLOGY

Invited Paper

C.1 Petri-nets

C.2 Queueing systems

C.3 Simulation

A. MODELLING TOOLS

Invited papers

Modelling Techniques and Tools for Performance Analysis
D. Potier (Editor)
Elsevier Science Publishers B.V. (North Holland)

THE EVOLUTION OF THE RESEARCH QUEUEING PACKAGE

Charles H. Sauer
IBM Entry Systems Division
Austin, Texas 78758

Edward A. MacNair
IBM Thomas J. Watson Research Center
Yorktown Heights, New York 10598

Queueing networks are important as performance models of systems where performance is principally affected by contention for resources. The Research Queueing Package (RESQ) is a well known software package for definition and solution of queueing network models. This paper examines the history of the several versions of RESQ and why RESQ has evolved as it has. Successes, failures and relationships to other research and development efforts are described. The paper assumes the reader has some familiarity with queueing network models and queueing network software.

1. INTRODUCTION

1.1. Queueing Network Models

Queueing models have been used for decades in studying the performance of manufacturing lines, communication networks and similar systems. In the 1960's there were dramatic insights into queueing network models of communication networks and computer systems (particularly those of Kleinrock [24] and Buzen [9]) and product form solutions of queueing networks (particularly those of Jackson [22] and Buzen [9]). In turn, in the 1970's and early 1980's there has been substantial additional progress in the understanding and application of queueing network models. This paper assumes the reader is sufficiently familiar with this work that a review is not necessary. For general discussion of queueing network models, see the recent textbooks in the area (e.g., Sauer and Chandy [48], Lavenberg *et al* [27], Lazowska *et al* [30]) and the queueing network issues of *Computing Surveys* (September 1978) and *Computer* (April 1980).

1.2. Queueing Network Software

For queueing network models to be used effectively, appropriate software is necessary to construct models and to obtain solutions for models. This paper traces the evolution and design issues in one of the earliest and most influential packages of queueing network software, the Research Queueing Package (RESQ). An appendix shows a frequently used example model constructed and solved using the current version of RESQ. The bibliography at the end of the paper includes most of the previous papers and technical reports on RESQ, except for those restricted to IBM internal use. Included in the bibliography are two prior retrospectives [51,54], a monograph [53], user manuals [55,56,57], and a product availability notice [52].

The product version of RESQ is available to IBM customers in the U.S., Puerto Rico, Europe, the Middle East and Africa. Some of the RESQ characteristics described in this paper are considered experimental and are not included in the product version of RESQ. Similarly, this paper speculates about future development of RESQ; this should not be taken as indication of future additions to the product version of RESQ.

There are many other packages of queueing network software. The bibliography includes references for most of the well known packages, but does not attempt to include references for all of the packages that have been developed. A few of these are discussed to the extent they relate to RESQ.

1.3. Significance of RESQ

RESQ success and importance is due to the multiple roles it plays and the synergism between these roles.

- The primary role of RESQ is as a modeling package for practical performance evaluation work. RESQ has been used throughout IBM on a wide variety of modeling problems, including not only computer/communication system modeling, but also manufacturing [35], emergency building evacuation and others. RESQ is currently installed on over 130 IBM internal computing systems.

- RESQ is a flexible and convenient tool for research in performance evaluation methodology. Areas covered in this research include simulation methodology, e.g., output analysis, stopping rules, etc., and approximate solution methods, e.g., development of new methods, empirical validation of methods, etc.

- RESQ is an effective vehicle for making available new results in performance methodology, both those developed using RESQ, e.g., in the areas just cited, and those developed without using RESQ, e.g., new algorithms for exact solution of queueing networks.

- The RESQ concepts themselves are research topics, e.g., the appropriate definition of queueing network extensions for effective modeling of general classes of systems, the development of languages for expressing definitions of queueing networks, etc.

All of these roles will be discussed further in the remainder of the paper, as appropriate. The remainder of the paper is largely chronological, first addressing work prior to the first RESQ prototype, then the prototype, the first widely used version, the current version and future possibilities.

2. PRE-RESQ SOFTWARE AND CONCEPTS

2.1. ASQ, QAL, QSIM, APLOMB (and the Origin of the Passive Queue)

Keller developed the ASQ (Arithmetic Solution of Queues) package [23] based upon Buzen's algorithms for queueing networks with product form solutions [9] and Chandy's generalization of the class of networks with product form solution [11]. ASQ provided an interactive dialogue for definition and solution of networks, making it possible to produce results with much less effort than with conventional simulation languages. ASQ was a great inspiration to Sauer and other students at the University of Texas at Austin, for it both demonstrated the power of a such a tool and pointed out the opportunity to develop more powerful tools, particularly for networks without product form solutions.

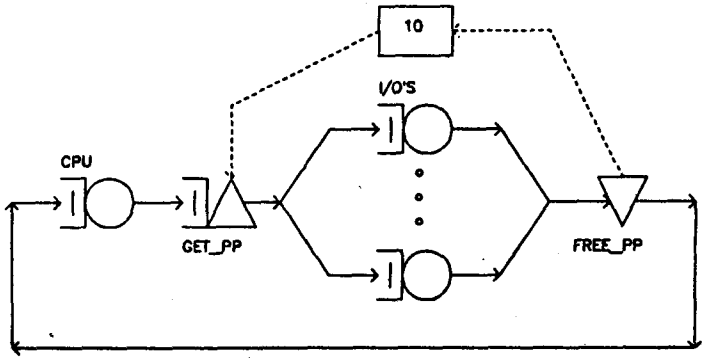

Figure 1. Central Server Model with Peripheral Processors

One of the primary limitations of product form networks was (and is) that a job could only hold one resource at a time. As part of a planned dissertation, F. Palacios-Gomez was seeking the solution of the model shown in Figure 1. The model was intended to represent the CDC 6600 in use at the University of Texas. The model was based on Buzen's central server model [9], but included representation of acquisition of one of the 6600's peripheral processors to perform I/O, and the subsequent release of that processor. (The 6600 CPU cannot perform I/O and is dependent on the peripheral processors for non-computational functions.)

Foster, Sauer and Waggoner generalized the construct used in the Palacios-Gomez model, and termed it a "passive server." (Passive as opposed to the "active" servers of traditional queueing models.) The passive server was a key concept in QAL (Queueing Analysis Language) which they proposed as a general language, analogous to a programming language, for specification of queueing models [19]. Though the definition of passive servers had flaws, the concept would be refined and, with the name "passive queue," become a key aspect of the success of RESQ.

QAL was intended to be implemented with a variety of solution techniques, but the only implementation was the QSIM simulation program of McGehearty [19]. There were three principal problems with QAL. First, there was little direct consideration given to non-simulation implementations and even simulation implementation was difficult. Second, QAL provided little support for representing distinct classes of jobs. Third, because QAL was designed as a language and not an interactive dialogue it required greater understanding on the part of the user.

While still at the University of Texas, Sauer began development of the simulation program APLOMB [42]. This program was initially intended to be a vehicle for empirical validation of approximate solution methods for queueing networks, not an interesting work in its own right. APLOMB provided the regenerative method for estimation of confidence intervals [17] and a sequential stopping rule to determine run lengths [29]. (The name was chosen because the program handled a difficult problem, simulation output analysis, with aplomb.)

After joining IBM, Sauer continued to develop APLOMB, with the intent that it be used for hybrid simulation of SNA networks [61]. New constructs, similar to those of QAL, were added to the program, but the construct definitions were chosen much more carefully than with QAL. Special care was taken to allow efficient implementation, to allow the use of the regenerative method and to provide more unified definitions [43]. For example, the QAL