Available at WWW.MATHEMATICSWEB.ORG POWERED BY SCIENCE DIRECT®

ELSEVIER Simulation Modelling Practice and Theory 11 (2003) 251–267

SIMULATION MODELLING PRACTICE AND THEORY

www.elsevier.com/locate/simpat

A distributed system simulation modelling approach

M. Nikolaidou [a,*], D. Anagnostopoulos [b,1]

[a] *Department of Informatics, University of Athens, Panepistimiopolis, 15771 Athens, Greece*
[b] *Department of Geography, Harokopio University of Athens, 70 El. Venizelou Strasse, 17671 Athens, Greece*

Received 12 June 2002; received in revised form 7 April 2003; accepted 16 April 2003

Abstract

The employment of network-based technologies, such as the WWW and middleware platforms, significantly increased the complexity of distributed application, as well as the Quality-of-Service requirements for the underlying network. Distributed application modelling is nowadays far more demanding than network modelling, where numerous solutions are already employed in commercial tools. We introduce a simulation modelling approach for distributed systems, giving emphasis to distributed applications. The proposed scheme enables the in-depth description of application functionality, the accurate estimation of network load and the extension of existing application models to support further customisation. It supports widely employed architectural models, such as the client–server model and its variations, and is based on multi-layer decomposition. Application functionality is described using predefined operations, which can be further decomposed into simpler ones, ultimately resulting into elementary actions corresponding to primitive network operations, such as transfer and processing. Even if realisation of this scheme proves to be time demanding, individual application modelling is performed with consistency and considerably lower overhead. The distributed system simulation environment built to realise the proposed modelling scheme and a case study indicating key features of the overall approach are also presented.

Keywords: Distributed systems; Distributed application modelling; Distributed system simulation; Performance valuation

* Tel.: +30-210-727-5614; fax: +30-210-727-5214.
E-mail addresses: mara@di.uoa.gr (M. Nikolaidou), dimosthe@hua.gr (D. Anagnostopoulos).
[1] Tel.: +30-210-954-9171; fax: +30-210-951-4759.

1569-190X/03/$ - see front matter
doi:10.1016/S1569-190X(03)00050-9

1. Introduction

The outburst in network technology gave rise to different types of applications operating in a network environment. Most of them are based on multi-tiered client–server models [1], and are generally called *distributed applications*. Distributed applications extend to multiple sites and operate on multi-platform networks. Distributed applications and the network infrastructure form a *distributed system* [2]. Most commercial information systems, such as banking and flight control systems, e-mail and WWW applications, distant learning environments and workflow management systems, fall in this category. Development of middleware standards [3], such as CORBA that allows the interaction between heterogeneous, autonomous applications, and of programming languages, such as Java that provides native distributed programming support, have established a well-defined framework for distributed application development.

Simulation modelling has been widely acknowledged as an efficient technique for performance evaluation. Numerous methodological and practical approaches for distributed system simulation have appeared in the literature. In most cases [4–6], application performance exploitation is closely depended on the network infrastructure. Thus, applications running on a network environment are viewed as network traffic generators and application operation mechanisms are not emphasised. Investigation of the Quality-of-Service provided by the network to determine whether application requirements are efficiently supported has also been the objective of simulation studies [7], where applications are usually represented using analytical models. In these cases, distributed application operation is not emphasised due to the significance of networking issues.

When orientation is towards evaluating an aggregate distributed architecture, system components are analytically described and component-specific models are employed. Distributed system modelling is mostly based on the client–server model. However, both client and server functionality is usually represented at an abstract layer and, due to the number and complexity of potential component combinations, behavioural characteristics of individual models are roughly modelled [8,9]. In-depth performance evaluation approaches have also been provided, especially for customised applications, where load generation is modelled at a low layer mostly using mathematical models [10–12], thus not promoting the reusability of simulation models. In [13,14], UML is used to model distributed system functionality, while mathematical modelling, specifically queuing networks, is adopted to estimate application performance. Use of UML sequence diagrams [15] facilitates the description of client–server architectures, process triggering and information exchange. However, the detailed description of process functionality is not facilitated. When examining the operation of distributed applications [6,7,9,10,12,13], object-oriented modelling techniques are usually adopted. Application operation is directly mapped at the elementary action layer as a series of discrete requests for processing, network transfer, etc., using pre-defined, elementary actions. We consider that such approaches lack efficiency and wide applicability, as: