

# TUTORIALS

Sunday, June 25, 2000

**8:30 AM – 12:00 NOON**

## **TUTORIAL 1:**

*Building Dependable Systems: the Power of Negative Thinking*

**Chuck Howell, MITRE Corporation**

There is a natural human tendency to be optimistic and to focus on the positive functional capabilities a new software intensive system will provide. However, for systems that must be trustworthy or dependable, there is much to be gained from "negative thinking": at each stage of development, considering all the ways things could go wrong. This half-day tutorial will use case studies to illustrate the importance of hazard analysis, error-handling design, stress testing, fault injection, and other "negative" tasks. The tutorial is divided into three sections corresponding to stereotyped development stages.

- **Requirements:**

A variety of tools and techniques have evolved to support the identification and management of potential hazards in complex critical systems. Notable examples of these techniques include Hazard and Operability Studies (HAZOP), Failure Modes and Effects Analysis (FMEA), Fault Tree Analysis (FTA), and Deviation Analysis. We will explore a range of requirements issues to be considered from the perspective of what could go wrong and how should the system deal with it.

- **Design and Construction:**

The focus of this section is on an error handling models, graceful degradation, and the use of techniques such as assertions and Design By Contract. Subtle mismatches among error-handling portions of subsystems can be extremely difficult to uncover, and are therefore an important cause of latent software defects.

- **Testing and Validation:**

Testing to provide evidence of robustness differs in focus from general software testing to demonstrate functional behavior. A great deal of emphasis must be placed on demonstrating that, even under stressful conditions, the software does not exhibit fragile behavior. This requires a considerable amount of fault injection, boundary condition and out-of-range testing, and exercising of those portions of the input space that are related to potential failures (e.g., critical operator functions and interactions, deliberate security attacks). It also includes test coverage analysis (both functional and white box) to ensure that the error detection and recovery aspects of the system are well exercised.

*BIOGRAPHY:*

Chuck Howell is Chief Engineer of the Joint and Defense Wide Systems Division at the MITRE Corporation. Previously he was Chief Engineer in the Systems Technology Center of Mitretek Systems, Inc., Director of Consulting Services at Reliable Software Technologies, a Java Technologist at Sun Microsystems, and a

Principal Scientist at the MITRE Corporation focusing on Critical Software Assurance for organizations such as the U.S. Department of Defense, FAA, Nuclear Regulatory Commission, and commercial organizations such as electric utilities and FDA regulated manufacturers. His current interests include techniques to calibrate and reduce residual doubt about the behavior of critical systems, and approaches to making large Networked Information Systems more robust (i.e., less fragile). He is a Senior Member of the IEEE and a Member of the ACM.

**8:30 A.M. – 12:00 NOON**

## **TUTORIAL 2:**

*Requirements,  
Technologies, and  
Architectures for Large  
Web Sites*

**Steven Hunter, IBM Corporation**

As the computing capability continues to track Moore's law and networking bandwidth grows, the division between the network and the computer is beginning to become indistinguishable. One example of this is with the web-hosting environment. The technology required to deploy a web site is very much a mixture of what was once considered network or server specific. This tutorial will provide insight into the workings of a large web site and will offer an understanding of key requirements and technologies that are used as components. Some of the topics covered will include: Dynamic Load Balancing, Quality of Service (QoS), and System Area Networks, such as Infiniband.

### *BIOGRAPHY:*

After completing his bachelor's degree at Auburn University, Steve joined IBM in March

1984 and worked on the development of networking products, such as the 3708 Protocol Converter and 3710 Controller. In 1986, Steve began working on 3745 Communication Controller products where he was involved with several hardware and software projects. In 1988, Steve completed his Master's degree from NC State University, which focused on networking architecture and technology. In 1991, Steve moved into an area to focus on the architecture and design of network routing products. In this position, Steve had responsibilities in both hardware and software with emphasis on hardware architecture, networking protocols and fault tolerance. In 1994, Steve began a 2-year resident study term at Duke University to complete his Ph.D. His primary research interests at Duke were in the areas of high-speed networking, distributed architectures and fault tolerance. Upon returning to IBM in 1996, Steve worked on ISP networking technology, then moved into the Netfinity server organization. As part of Netfinity, Steve is in the group with server architecture and technology responsibility and has had involvement and responsibilities with multiprocessing systems, clustering, server/network attachment, system area networks (i.e., InfiniBand), systems management and RAS. Steve holds several patents and has published papers and presented at a variety of conferences and symposiums.

**1:30 PM – 5:00 PM**

## **TUTORIAL 3:**

*Fault Tolerant CORBA*  
**Shalini Yajnik, Lucent-Bell  
Laboratories**

CORBA (Common Object Request Broker Architecture) is a platform for object-oriented

distributed computing, standardized by the Object Management Group (OMG). It provides a middleware upon which distributed applications can be built very quickly and easily. However, until recently CORBA did not provide any tools for enhancing the reliability of applications. As a result, use of CORBA in building highly reliable distributed systems was limited. OMG realized this problem and issued an RFP for fault tolerant CORBA in April 1998. For the past year, a group of industries has been working to respond to this RFP. In January 2000 the group presented a proposal for Fault Tolerant CORBA to the OMG, which will move towards standardization in the coming months. The proposed standard will provide a standard and efficient way for developers to build highly reliable and available distributed applications.

In this tutorial I will give a brief overview of CORBA and then go on to discuss the proposed Fault Tolerant CORBA specification, which makes use of replication of CORBA objects to provide desired levels of reliability and availability to a system. I will talk about the wide spectrum of fault tolerance covered by the specification and how it can be used to build applications which require widely varying degrees of fault tolerance, e.g. (1) stateless systems like replicated web servers whose clients require simple failover mechanisms, to complex defense and telecom systems that require five nines reliability and availability along with strong data consistency, and (2) systems that do not want to deal with fault tolerance issues and would like the fault tolerance to be fully automated, to systems that require high degree of application-specific control under failure conditions.

#### *BIOGRAPHY:*

Shalini Yajnik is a Member of the Technical Staff in the Distributed Software Research department at Lucent Technologies, Bell Laboratories. She graduated with a Ph.D. degree from Princeton University in 1994. Her research interests are software level fault tolerance and distributed object systems, with the primary focus on studying the impact of failures on distributed object systems and developing fault tolerance solutions for distributed object platforms like CORBA and Java RMI.

**1:30 PM – 5:00 PM**

### **TUTORIAL 4:**

#### *Exception Handling and Software Fault Tolerance*

**Jie Xu, University of Durham  
Brian Randell, University of  
Newcastle upon Tyne**

As the use of computer systems becomes more and more widespread in applications that demand high levels of dependability, these applications themselves are growing in size and complexity at a rapid rate, especially in areas that require concurrent and distributed computing. Such complex systems are very prone to faults and errors from a variety of sources, including the devices and people in the environment of the computer system, the computer and communications hardware, etc. Moreover, given such complexity, no matter how rigorously fault avoidance and fault removal techniques are applied, software design faults often remain in systems when they are delivered to the customers. There is a tremendous need for systematic techniques for building dependable software for such systems.

This half-day tutorial will focus on some basic concepts, state-of-the-art techniques, and state-of-practice approaches to exception handling and software fault tolerance in both sequential programs and complex concurrent systems. The first part of the tutorial will start with an account of both programmed and default exception handling methods in sequential modular programs, and then go on to describe the recovery block approach to software fault tolerance and subsequent extensions to this scheme. The second part of the tutorial will talk about coordinated exception handling and software fault tolerance in concurrent and distributed systems. It will cover the generalized conversation scheme for handling exceptions in process-oriented systems and present the coordinated atomic (CA) action scheme for concurrent object-oriented systems, illustrated with an industrial control application. The CA action approach is in fact based on a very sophisticated exception handling scheme, capable of dealing appropriately even with very complex situations, including multiple concurrent faults.

#### *BIOGRAPHIES:*

Jie Xu is a Lecturer in the Department of Computer Science, University of Durham, UK. He received the Ph.D. degree from University of Newcastle upon Tyne on Advanced Fault-Tolerant Software. From 1990 to 1998, Dr. Xu was with the Computing Laboratory at Newcastle where he was promoted to a Senior Researcher in 1995. He moved to a Lectureship at Durham in 1998 and cofounded the Distributed Systems Engineering group and the DPART laboratory supporting highly dependable enterprise computing. Dr. Xu has published more than 90 academic papers and

refereed reports in areas of system-level fault diagnosis, exception handling, software fault tolerance, and large-scale distributed applications. He has been involved in several research projects on dependable distributed computing systems, including two EC-sponsored ESPRIT BRA projects and one ESPRIT Long Term Research project. He is Principal Investigator of the FTNMS project on fault-tolerant mechanisms for multiprocessors and co-Investigator of the EPSRC Flexx project on highly flexible software.

Brian Randell graduated in Mathematics from Imperial College, London in 1957 and joined the English Electric Company where he led a team which implemented a number of compilers, including the Whetstone KDF9 Algol compiler. From 1964 to 1969 he was with IBM, mainly at the IBM T.J. Watson Research Center in the United States, working on operating systems, the design of ultra-high speed computers and system design methodology. He then became Professor of Computing Science at the University of Newcastle upon Tyne, where in 1971 he set up the project which initiated research into the possibility of software fault tolerance, and introduced the "recovery block" concept. Subsequent major developments included the Newcastle Connection, and the prototype Distributed Secure System. He has been Principal Investigator on a succession of research projects in reliability and security funded by the Science Research Council (now Engineering and Physical Sciences Research Council), the Ministry of Defence, and the European Strategic Programme of Research in Information Technology (ESPRIT). Most recently he has had the role of Project Director of DeVa, the ESPRIT Long Term Research

Project on Design for Validation, and of CaberNet, the ESPRIT Network of Excellence on Distributed Computing Systems Architectures, and is now leading an IST Research Project on Malicious- and Accidental-Fault Tolerance for Internet Applications (MAFTIA). He has published nearly two hundred technical papers and reports, and is coauthor or editor of seven books.

**1:30 PM – 5:00 PM**

## **TUTORIAL 5:**

### *Performance of TCP on Error-Prone Wireless Links*

**Nitin H. Vaidya, Texas A&M University**

This tutorial deals with the impact of wireless transmission errors on the performance of TCP, and techniques for improving performance in presence of such errors. It will provide the attendees with an overview of the state of the art in TCP for error-prone wireless links, and an understanding of the basic principles that guide the design of mechanisms to improve TCP performance in wireless environments.

Topics will include an overview of wireless technologies, an overview of relevant TCP features, the impact of wireless errors on TCP performance, and classification of approaches to improve TCP performance in the presence of transmission errors. A detailed discussion of a few representative approaches will be presented, followed by a summary of other approaches.

Topics to be discussed include impact of link layer retransmission on TCP performance, the

split connection approach, explicit notification schemes, the Snoop protocol, and the delayed dupacks protocol to improve TCP performance in presence of transmission errors.

#### *BIOGRAPHY:*

Nitin Vaidya received the Ph.D. degree from the University of Massachusetts at Amherst. He is presently an Associate Professor of Computer Science at the Texas A&M University. He has held visiting positions at Microsoft Research, Sun Microsystems, and Indian Institute of Technology-Bombay. His research interests include wireless networking, mobile computing, and fault-tolerant computing. He is a speaker for the Distinguished Visitor Program of the IEEE Computer Society, a recipient of a CAREER award from the National Science Foundation, a coauthor of the Best Student Paper Award-winning paper at MOBICOM '98, and of a 1999 IETF PILC working group internet-draft dealing with performance of implications of transmission errors. Nitin served as program cochair for the 1999 International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communication (DIAL M). He has also served on the program committees of several other conferences and workshops, including INFOCOM 2000, 1999 ACM Symposium on Principles of Distributed Computing (PODC), 1999 Workshop on Data Engineering for Wireless and Mobile Access (MobiDE), and 1998 International Workshop on Satellite-based Information Services (WOSBIS). Vaidya is a senior member of the IEEE and member of the ACM.