

## OVERVIEW OF OPEN SYSTEMS AND OBJECT ORIENTED TECHNOLOGY

### General:

This course provides an overview of the concepts and terminologies associated with both open systems and object oriented technologies. It provides the students with insight into the general philosophy and capabilities provided by the UNIX® operating system. It also describes the hardware UNIX has been ported to, as well as, the advantages and disadvantages of using UNIX. Object oriented technology is described with emphasis placed on the risks and benefits of using an OO approach to solving a business problem.

### Objectives:

Upon successful completion of this course, the student will be able to:

- Describe the history, features, and philosophy of UNIX and open systems.
- Describe the UNIX File System structure, features, and advantages.
- Describe the features provided by the Shell interpreter.
- List and describe text editors and programming languages commonly available under the UNIX operating system.
- Describe Open Systems and the current UNIX standardization efforts and organizations.
- List the types of micro-chips and hardware that will support the UNIX operating system.
- Describe the differences between real UNIX and *UNIX-like* systems.
- List the advantages and disadvantages of the UNIX system.
- State the risks and benefits of using object oriented technology.
- Describe the object oriented concepts of encapsulation, inheritance, and polymorphism.
- Describe object oriented standardization efforts.

### Audience:

Managers, Technical Users, Programmers and Sales Personnel who desire a broad understanding both open systems and object oriented technologies.

### Prerequisites:

None.

### Duration:

One (1) day.

**OVERVIEW OF OPEN SYSTEMS  
AND OBJECT ORIENTED TECHNOLOGY  
COURSE OUTLINE**

**I. INTRODUCTION**

- A. What are Open Systems
  - 1. Portability
  - 2. Scalability
  - 3. Interoperability
- B. What is UNIX
- C. Evolution of the UNIX System
- D. UNIX System Structure
  - 1. Kernel
  - 2. Shell
  - 3. Utilities
- E. UNIX System Features
- F. UNIX Philosophy
- G. Pipelines and Tools

**E. Shell Programming Language Features**

- 1. Shell Procedures
- 2. Shell Variables
- 3. Program Control Constructs
- 4. Structured Programming

**IV. TEXT EDITORS AND PROGRAMMING LANGUAGES**

- A. Line Oriented Editors
- B. Screen Oriented Editors
- C. Standard UNIX Programming Languages
- D. Available Miscellaneous Languages

**II. UNIX FILE SYSTEM**

- A. What is a UNIX File
- B. Types of Files
  - 1. Regular Files
  - 2. Directory Files
  - 3. Special Files
  - 4. FIFO Files
- C. File System Structure
  - 1. Hierarchical Inverted Tree
  - 2. Advantages of File System
  - 3. Maneuvering within the File System
- D. File System Features
- E. Standard System Files
- F. Mountable File Systems

**V. STANDARD UNIX SYSTEM TOOLS**

- A. Software Development Tools
  - 1. Software Generation
  - 2. Software Debugging
  - 3. Software Maintenance
- B. System Communications Tools
  - 1. UNIX Intra System
  - 2. UNIX Inter System
- C. Text Processing Tools
- D. General Purpose UNIX Tools
  - 1. File Print Tools
  - 2. File Comparison Tools
  - 3. File Interrogation Tools
  - 4. Miscellaneous Tools

**III. SHELL INTERPRETER**

- A. Individual Shells
- B. Shell Environment
- C. Standard Shell User Interface
- D. Shell Command Language Features
  - 1. Shell Command Line Types
  - 2. File Name Generation
  - 3. Pipes and Filters
  - 4. I/O Redirection
  - 5. Asynchronous Command Lines

**VI. MISCELLANEOUS UNIX INFORMATION**

- A. Licensing Information
- B. Current Standardization Efforts
  - 1. Open Systems
  - 2. POSIX
  - 3. OSF
  - 4. X/Open
- C. UNIX Information Sources

**VII. UNIX HARDWARE AND SOFTWARE VARIATIONS**

- A. Micro Chips with UNIX Support
- B. Hardware Supporting UNIX
  - 1. Super Micros
  - 2. Mini Computers
  - 3. Mainframes
  - 4. Alternate Architectures
- C. UNIX Software Variations
  - 1. UNIX Like Systems
  - 2. UNIX Look Alikes
  - 3. UNIX Act Alikes

**VIII. ADVANTAGES AND DISADVANTAGES OF THE UNIX SYSTEM**

- A. Advantages
- B. Disadvantages

**IX. OBJECT PARADIGM**

- A. Problems with Software Development
- B. Using Objects to Solve Problems (Object Model)
- C. Risks, Benefits, and Costs
- D. Features of Object Oriented Languages
- E. Procedural vs. Object Oriented
- F. Client/Server Evolution
- G. Effects of OO Approach

**X. OBJECT ORIENTED CONCEPTS**

- A. Data Abstraction
- B. Classes
- C. Objects
- D. Messages
- E. Encapsulation
- F. Inheritance
- G. Polymorphism
- H. Basic OO Design (CRC Cards)

**XI. OBJECT ORIENTED STANDARDS**

- A. Distributed Object Computing (DOC)
- B. Object Management Group (OMG)
- C. Object Management Architecture
- D. OMA Reference Model
- E. Common Object Request Broker Architecture (CORBA)
  - 1. What is CORBA
  - 2. CORBA Implementations
- F. Interface Definition Language (IDL)