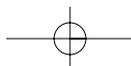
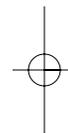
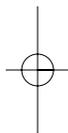
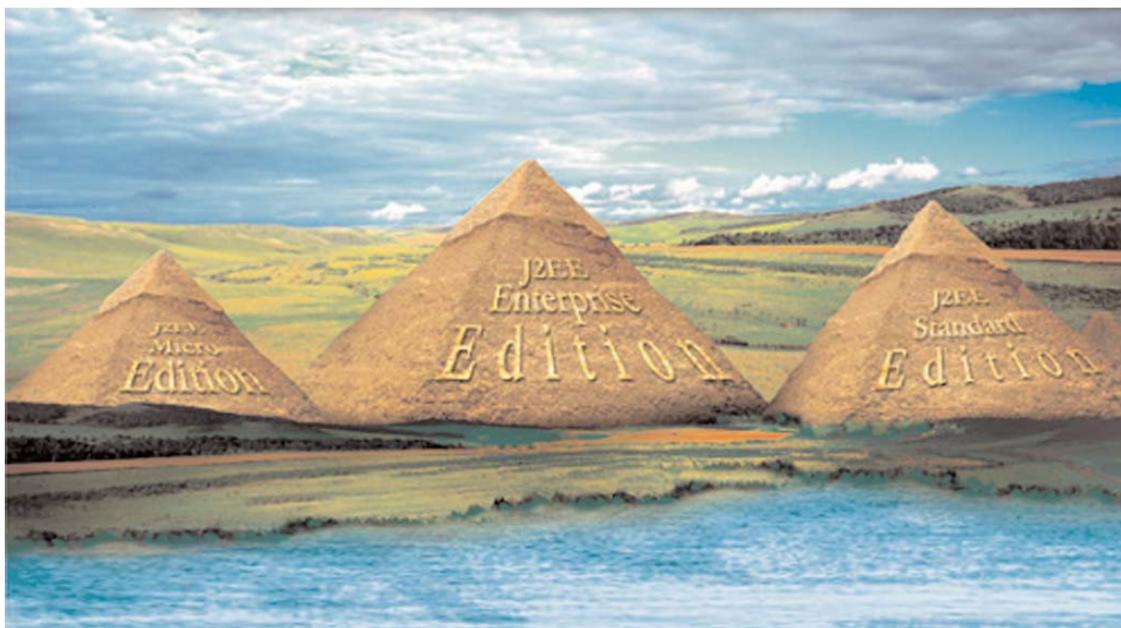
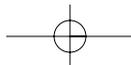


# Part I

---

# Overview



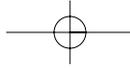


This part is designed to present the essential background information about enterprises and position Java 2 Enterprise Edition (J2EE) as a suitable technology that attempts to solve a significant number of problems faced by these enterprises. The main focus of this part is to elucidate the emergence of J2EE technology on the enterprise scenario. This part also attempts to provide a bird's-eye-view of an enterprise and the various issues and challenges faced by them.

The non-functional quality requirements of an enterprise application and its importance to an enterprise is discussed in this part, and an attempt has been made to highlight the complexities that arise due to the conflicting nature of these enterprise application quality parameters and their relative importance to the organization.

This part also strives to bring out the emergence of java technology in the enterprise arena and provide an overview of J2EE technology. Efforts are made to show how J2EE could address different aspects of the enterprise requirement, particularly those of the non-functional quality parameter aspects of the enterprise.

There are five chapters in this part. Chapter 1 is an introduction to the enterprise, its challenges and issues. Legacy systems and their importance to large enterprises is discussed here. Chapter 2 is on the essentials of the solution to the enterprise. The vastness of the problems faced by the enterprises and the need of an architect who could address the dynamic nature of the problems of an enterprise is presented here. In Chapter 3, the importance of the systemic qualities and the architectural views are discussed. Chapter 4 presents an overview of J2EE technology and explains some of the crucial aspects of this technology as an enterprise platform. In Chapter 5, the suitability of J2EE technology is weighed against other possible proprietary solutions.



# Chapter 1

## Introduction

- ✓ **What is an Enterprise?**
- ✓ **Enterprises and Systems**
- ✓ **Diversity in the Enterprises**
- ✓ **Growth in Enterprises**
- ✓ **Enterprises and Legacy**
- ✓ **Needs of the Growing Enterprise**
- ✓ **Summary**
- ✓ **Review Questions**

## 4 J2EE Architecture

---

### 1.1 WHAT IS AN ENTERPRISE?

An 'Enterprise' may be defined as an organization created for carrying out business ventures. This term is applicable to all those businesses that could encompass corporations, small businesses, non-profit organizations, government bodies, military and other kinds of organizations. However, in practice, this term is generally applied more appropriately to organizations that are larger in size as well as distribution. Enterprises are those organizations whose business activities span over different geographical areas, deal with multiple products and/or services and support a large number of transactions.

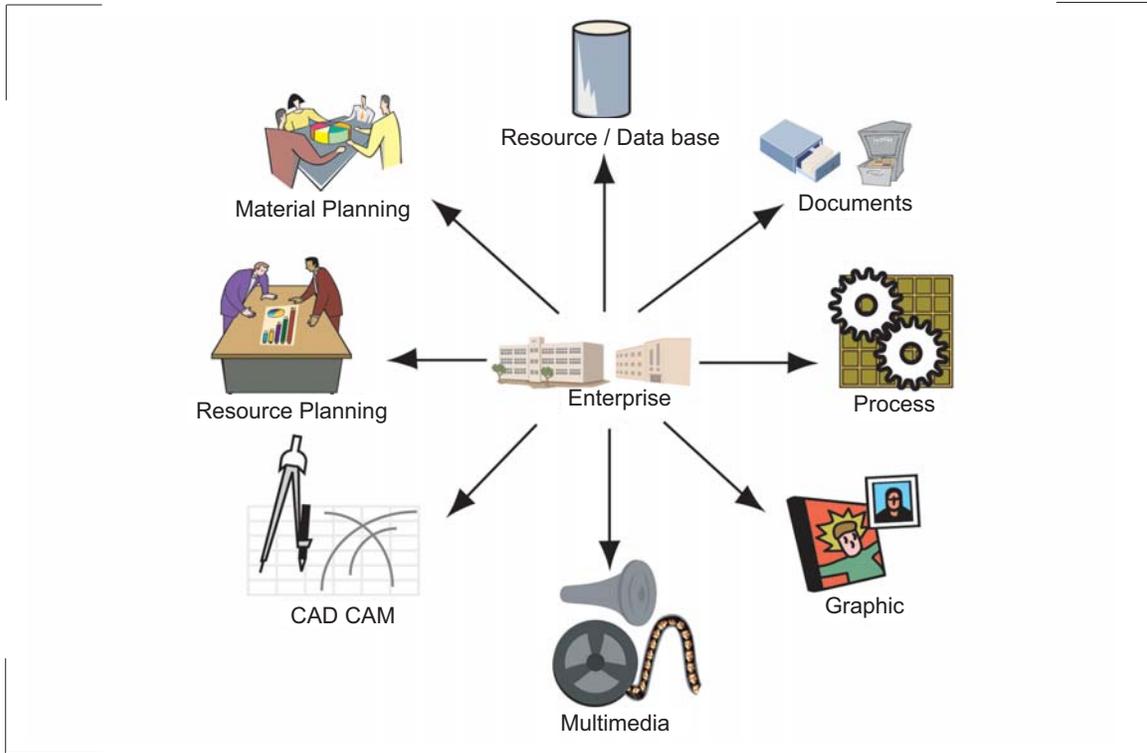
Enterprises are essentially large in size and deal with problems that are potentially complicated in nature and often plagued by issues such as management of businesses and transactions, maintaining records, coping with multiple language and cultural barriers and other business-related issues. In order to manage the business well and be competitive in the market and at the same time be profitable, enterprises often depend on the automation of business activities.

As shown here, **Figure 1-1** highlights some of the complex scenarios that are faced by the enterprises in general.

### 1.2 ENTERPRISES AND SYSTEMS

Invention of computers and business machines were a boon to many of these large organizations. These computer systems and business machines were capable of handling larger volumes of work with comfortable speed and better accuracy. Many large enterprises including business houses, military, governmental departments, research and development establishments and educational institutions could benefit by the introduction of such systems.

Although the first large-scale application was for the Census activity of the United States of America, the first ever computer application in the business was the use of computers as a 'ledger machine' at a small restaurant in the United Kingdom. The adoption of computers by businesses, small or large, has been on the rise since then. Almost no business has gone untouched by



**FIGURE I-1**

**Enterprises and the nature of complexities: The business processes of large enterprises are complex in nature. The enterprises need to be managing, as a part of the process, many items such as MRP, ERP, CAD/CAM, Data and Databases, documents etc.**

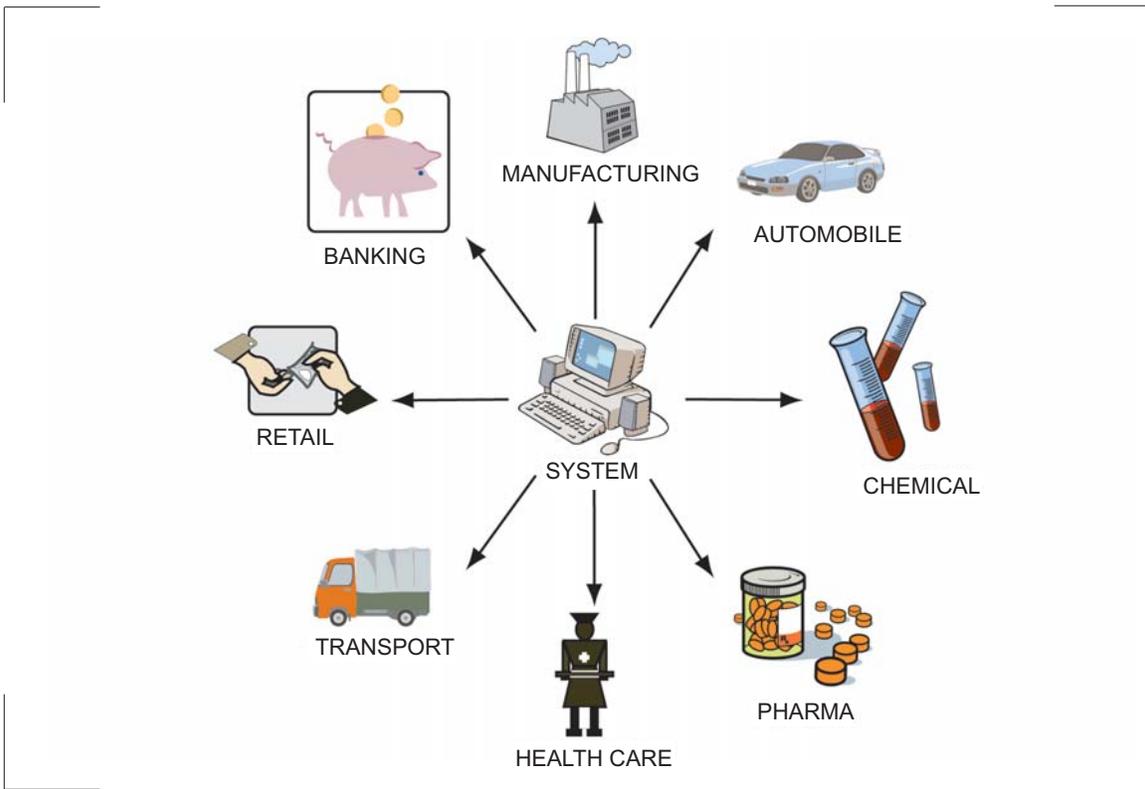
the computers and information technology. Figure 1-2 attempts to display some of the major enterprises affected by the computers and software tools.

As the development and establishment of computers and hardware helped enterprises to tackle many of the complexities of the business process, the need for meeting the growing requirements of business spurred the research and development in the field of computers and related systems, in a recursive

6 *J2EE Architecture*

fashion. Evolution of the technologies such as Client/Server and Distributed Computing technologies and the emergence of Internet and World Wide Web took the business world and organizations by storm.

Exponential advancements in these fields, coupled with the proliferation of systems and technologies affordable to the businesses as well as individuals



**FIGURE I-2**

**Influence of Information Technology over the enterprise businesses: IT has become a pervasive technology. Systems, devices, software and related technologies touch almost all aspects of the enterprise business. Some of the enterprises that immensely depend on IT are manufacturing, Automobile, Financial and Banking are shown in this figure.**

took the enterprises to another dimension all together. While the enterprises could visualize tremendous opportunities for growth and diversification due to such advancements, they also realized the problems, complexities and issues of a different nature.

### 1.3 DIVERSITY IN THE ENTERPRISES

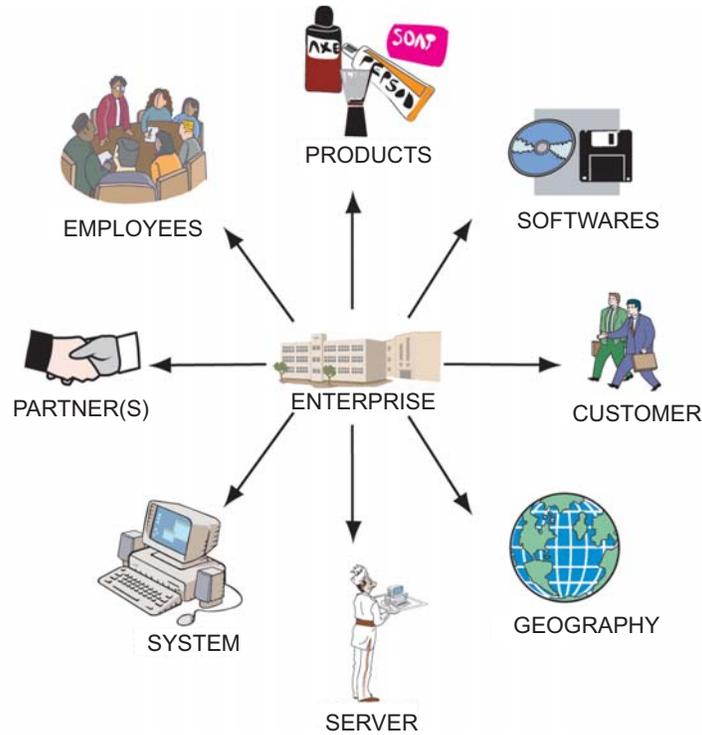
The development in the systems and software solutions over the past few years has been exponential in nature. There has been growth as well as diversification in hardware, software and related technologies, resulting in the creation of an industry by itself, termed Information Technology (IT). Tremendous growth has been seen in the hardware as well as the software industries, resulting in the proliferation of companies specializing in niche areas of enterprise problems. Data and database-related, documentation and document processing, process and workflow, Computer Aided Design (CAD)/ Computer Aided Manufacturing (CAM)/ Computer Aided Engineering (CAE) and analysis, graphics and multi media, and so on are some of the areas that form part of the enterprise activities. Figure 1-3 shows some of the areas of the enterprises that are of significance.

### 1.4 GROWTH IN ENTERPRISES

The growth in an enterprise is usually accompanied by the growth in a number of things such as the number of users and load, geographic expansion including multi-national and multilingual, multiple systems, architecture and solution that could be compatible (or less compatible) with the business interests of the existing enterprise. Furthermore, growth in the enterprise, at times, need not be necessarily organic in nature. Mergers and acquisitions, takeovers—hostile or friendly, diversification in the line of business etc. present unique problems to the enterprises, in terms of managing the business and sustaining the growth. While the purpose of growth is to sustain and profit, the task of managing the enterprises usually falls on to the systems and software.

The systems and software in any enterprise are heterogeneous in nature. One of the major requirements for an enterprise would, therefore, be to integrate these systems so that they cumulatively contribute to the productivity of the

8 *J2EE Architecture*



**FIGURE I-3**

**Areas and themes that are of immense Importance to enterprise businesses: Growth of an enterprise depends on many aspects. Some of the aspects have pronounced effect on the growth. Hardware and application software, products and services management of employees, partners, customers, supply chain etc are shown here.**

enterprise. Integration of all these systems and software in a large enterprise is still considered a challenging task.

The systems and software acquired (or developed in-house) by an enterprise is usually over a period of time. Many of the large organizations, which have a history of early adoption of such systems, take long periods of time to

stabilize the process. However, since the changes transpiring in the technologies of systems and software are far more rapid than that of the stabilization period for these enterprises, most of them face problems in adapting to exploiting the newer technologies/environments.

Shifting to newer systems and changing over to a newer environment demand 'migration'. While migration is technically feasible, most of the large enterprises face a plethora of problems in migrating to the newer environment. Many enterprises have several pressing reasons to maintain certain 'parts'/'portions' of the existing systems and environments and integrate the same to the newer technology. Such existing systems and software are considered as a 'legacy' from the previous era.

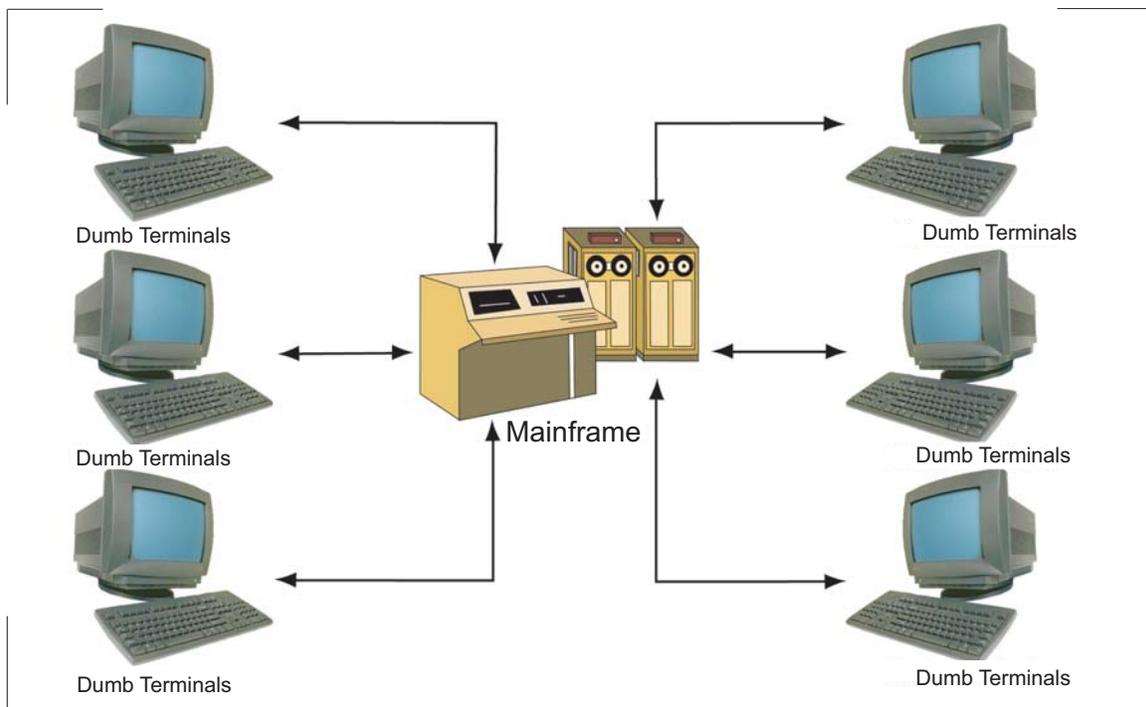
## 1.5 ENTERPRISES AND LEGACY

### 1.5.1 What is Legacy?

Legacy is a commonly encountered term in the field of IT. Legacy, as the term means in the English language, stands for something which has been inherited from the past. In the present context, the term Legacy carries a similar meaning. Legacy represents a collective set of application software, data structures, operational environments and the hardware that an enterprise had adopted earlier. Many of the existing mainframe systems, software and tools that are still in use on such systems considered 'irreplaceable', are referred to as 'legacy'. **Figure 1-4** portrays the legacy systems, such as mainframes, that are of importance to many of the enterprises. Many of the enterprises carry over legacy systems and processes due to historical reasons. Legacy systems and software might be very important for the main line of business of the enterprise, and it might be that they cannot be replaced due to many non-technical reasons. Integration of such systems and processes in the enterprise present additional problems as the enterprises grow. Legacy Integration or Legacy Modernization might involve many things, including use and supporting the existing database, use and supporting the Online Analytical Processing (OLAP) and Online Transaction Processing System (OLTP) systems.

### 1.5.2 Constituents of Legacy

Enterprises could inherit many systems, software and tools. These systems and the corresponding applications running on them could be very important for the organization’s business needs. These applications could be handling business logic that might have been developed over many years and could be crucial for the sustenance of the enterprise itself. The enterprises, therefore, are interested in either maintaining or modernizing these inherited portions of



**FIGURE I-4**

**Legacy Systems in the enterprises providing a ‘Centralized Model’ for enterprise processes: All enterprise applications were located in Mainframe systems. Users needed to connect to them using ‘dumb’ terminals and execute ‘batch’ jobs as a part of business process.**

the system with the current main stream of the business. Some of the reasons for continuing this could be attributed to factors such as:

- Investment and recovery of the same
- Stable, but complicated which the management do not want to disturb
- Code and Maintenance problems
- Valuable, difficult or expensive to translate to newer system.

Some of the parts of the legacy that the enterprises prefer to hold and maintain could be:

- Mainframe and related hardware
- Online Transaction Processing (OLTP) system on the mainframe or other similar hardware
- Online Analytical Processing (OLAP) system on the mainframe or other similar hardware
- Proprietary programs that deal with proprietary file and database management systems (DBMS)
- Proprietary database management systems and Relational Database Management Systems on the mainframe or other similar hardware

Some of these constituents are discussed in brief in the following sections

### 1.5.2.1 Online Transaction Processing

The Online Transaction Processing (OLTP) system may be described as a set of programs that owns the responsibility for ensuring transactions leading to concrete business deals. These deals are normally connecting either business-to-end users (B2C) and/or business-to-business (B2B). Some of the examples that may be cited in this case are applications leading to banking transaction, money/mail order of cash, airline ticket reservation, train/bus reservation, etc. One of the most popular application suites in OLTP are Customer Information Control Systems (CICS) on the IBM mainframes.

## 12 *J2EE Architecture*

---

### 1.5.2.2 Online Analytical Processing

The Online Analytical Processing (OLAP) system is a more comprehensive and complex system that is multi-faceted and multi-dimensional in nature than the OLTP systems. An OLAP system may be defined as a category of applications and technologies for collecting, managing, processing and presenting multi-dimensional data for analysis and management purposes.

The OLAP systems are a bunch of systems and programs that aid the management of large enterprises in many areas. The OLAP systems are capable of capturing and analysing the raw data into meaningful data with speed and efficiency. These systems are capable of fast data-processing and able to compute several multi-dimensional views of the required data that will help the management to quickly and easily comprehend the ongoing scenario of the enterprise process.

There are many applications capable of doing analytical processing, and these applications are available on several systems and hardware. However, to distinguish these systems from the OLAP systems, FASMI, a unique test is suggested. The term FASMI stands for:

- Fast
- Analytical
- Shared
- Multi-dimensional
- Information

The following subsection provides some details on the how and what each of the letter in the term FASMI stands for.

***Fast*** The systems and programs are expected to return the responses in a record time. For example, it should not take longer than 5 seconds for the response. And any response delayed by more than 30 seconds is considered a waste.

***Analytical*** The system is capable of analysing the data based on the business logic of the enterprise, as well as performing a variety of statistical analysis on the current data in question.

*Shared* The system is capable of sharing the data with multiple users and has the capability of read lock/read-write lock the shared data as per the security policies of the enterprise without leading to any corruption of data.

*Multi-dimensional* This is one of the most important features of the OLAP systems. These systems must be capable of providing multiple and multi-dimensional conceptual views of the data, including support to hierarchies and multi-level hierarchies, as required by the enterprises.

*Information* This is the data as well as the metadata of the enterprise. The OLAP systems are expected to collect as well as compute the data that is required for analysis. The amount of data that may be required for any business or statistical processing could be very large (in terms of megabytes or even gigabytes) and the system should be capable of holding and processing this data.

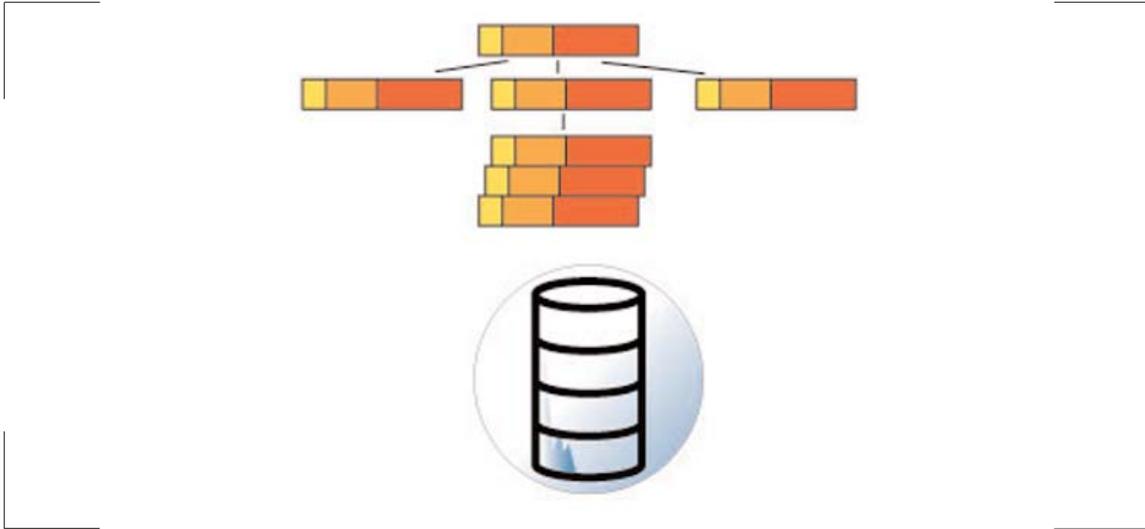
### 1.5.2.3 Databases, Flat files and Proprietary Datastorage systems

Enterprises, earlier, stored the data in a variety of formats. This is particularly true of the time before E. F. Codd proposed the 12 rules of Relational Database Management Systems (RDBMS). The data stored in these enterprises were in different formats. Other than flat and proprietary file systems, the popular way of storing and retrieving the data was in database formats. Long before the advent of RDBMS, the data were stored in two other different formats. They were:

- Hierarchical Databases
- Network Databases

*Hierarchical Databases* These databases, as illustrated in **Figure 1-5**, treat the data in hierarchical fashion. The data stored in such databases need to have a parent-child relation. While these databases were well suited for small and infrequently changing data, this system suffered due to a few important inherent problems. Repetition of the data, inefficient data retrieval, redundancy,

14 *J2EE Architecture*



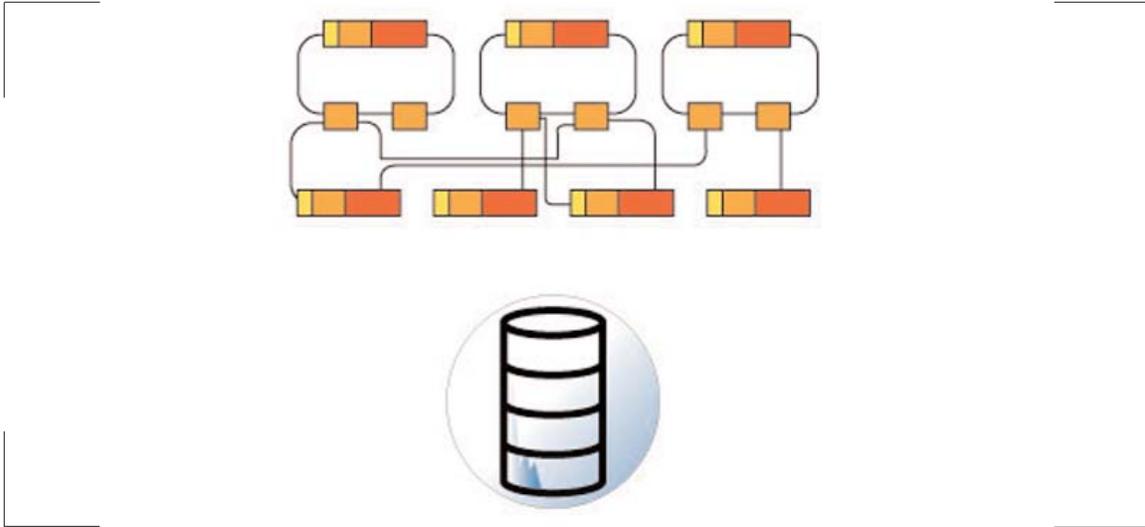
**FIGURE 1-5**

**Hierarchical Databases indicating the relationship between the individual pieces of data: Each of the colour-coded blocks indicates the data. The relationship between different sets of data can be shown in the form of a tree structure, indicating parent—child hierarchy relation.**

strict populace-schema<sup>1</sup> of the data, etc might be sighted as some of the reasons for the unpopularity of these databases.

*Network Databases* The network databases emerged after that of hierarchical databases. This model, as illustrated in **Figure 1-6**, fixed some of the problems of the hierarchical databases, particularly that of the redundancy. However, these databases suffered from issues such as inflexibility, implementation and other problems.

<sup>1</sup> In Hierarchical databases, one cannot populate a child table until it has already been incorporated into the parent table. In enterprise environments, such strict schema populating restrictions are considered a hindrance.



**FIGURE I-6**

**Network Databases indicating the relationship between individual pieces of data: Each of the colour-coded blocks indicates a data. The relationship between the individual blocks shows a ‘networked’ relation.**

### 1.5.3 Legacy and its Importance

As indicated in the above sections, the enterprises often consider these legacy items important and require that the newer systems need to provide appropriate support to these systems. It is also possible that these new and advanced systems may not provide some of the specialized services that have been offered by these systems, such as OLAP, for example. It is therefore imperative to provide integration to such systems, and the enterprises consider that to be an important link in the ongoing business process.

## 1.6 NEEDS OF THE GROWING ENTERPRISE

As the enterprises evolve and grow, the needs of the enterprises change. The enterprise requirements become more stringent and demanding. The

## 16 *J2EE Architecture*

---

requirements of the growing enterprises vary. However, qualitatively, the following are a few of the most important requirements that any of the evolving enterprises look for:

- Maintenance of the existing systems
- Migration of the old system in part/full if not found feasible to maintain the existing system
- Integration of the new system with the existing systems
- The necessity of the new system to scale up, based on load and demand
- Ability of the new system to be available to clients, partners and employees when necessary
- The new system should authenticate and authorize all the users of the enterprise application
- Ability of the new system to operate in a secure environment
- Ability of the system to be flexible to adapt to the changing needs of the enterprise
- The performance of the new system should be acceptable to all users
- Robustness of the new system
- Ability of the new system to operate reliably
- The new system should be available at all times to the users. Some of the enterprises could demand the availability by fixing the availability metrics at 99.999%<sup>2</sup>.

The points highlighted above indicate, at the gross level, the requirements of an enterprise. The degree of the requirements may vary for different organizations and the requirements of an enterprise. Some of the requirements are discussed in some detail in the following subsections.

---

<sup>2</sup> The availability metrics stated here is some times referred to as 'five nines'. Indeed such availability (and the related reliability) requirements demands that the unscheduled downtimes (for whatever reasons) cannot be more that 5 minutes in 365 days, and the systems operating 24×7.

### 1.6.1 Possibility of Migration

Some of the enterprises may opt for a migration path, fully or partly, to newer environment. However, some of the organizations may find migration to an entirely new environment both expensive and prohibitive in terms of culture. Also, in a few cases, migration may be impossible due to reasons such as:

- Technology obsolescence
- Unavailability of resources
- Unavailability of the 'source code'
- Unsupported hardware and/or operating environment

### 1.6.2 Secure Environment

The emergence of Internet and World Wide Web has made worldwide access to an enterprise possible. The enterprise systems can now be accessed from the desktop or even using the mobile devices. This has led to the situation that the 'data is out there'. This poses serious security issues for both the enterprises as well as the stakeholders of the enterprise. It is therefore important for the enterprise to provide a secure environment, to ensure the access of systems and information to the right users, protect the information from unauthorized access, and prevent the information from corruption and misuse.

### 1.6.3 Reliable Transactions

Many of the enterprises could be supporting applications that lead to some kind of transactions. As the business transaction could be taking place 'online' and across geographical boundaries, it is crucial for the enterprise that the transaction transpires in a reliable manner. The enterprise information needs to be consistent and the data integrity needs to be ensured when the transaction takes place. Failure to provide a reliable transaction to enterprises, particularly in the financial and banking sector, could prove costly.

### 1.6.4 System Availability When Accessed

As the businesses and enterprises are sprawling, the number of clients attempting to access the enterprise system could grow large. Depending on the nature of the business of the enterprise, the systems might need to support large numbers of users. When a user attempts to access the enterprise resource, the system should be available for accessing the enterprise resource. Online Stores and Online Stock-broking firms are a few of the examples of enterprises that need to cater to a very large number of users.

### 1.6.5 Scaling to meet Loads

When a large number of users try to access the enterprise system simultaneously, the load on the enterprise systems increases. The loads could be non-uniform in nature. For example, while the load on the system could be very high during the weekends, it could be moderate to manageable during the weekdays. Again, it could enormously increase during holidays and occasions such as New Year or Thanks Giving day<sup>3</sup>. Generally, the enterprise systems are designed for a certain number of users. However, when the number of users increases, the ability of the system to take up the load beyond a specific point is affected. As a result of this, the system may behave in an undesirable manner. The undesirable behaviour in the systems includes slow and sluggish responses, hanging of the server systems and could even lead to the crashing of the server systems in certain cases.

### 1.6.6 Quick Response When Accessed

The users of the enterprise system expect to complete the transaction in a short time. For example, a shareholder (or a stock-broker) intending to trade the stock will be extremely sensitive with respect to the quickness with which he can successfully trade. The response of such systems needs to be very high. It is important that the system ensures a quick and timely response to the user of the enterprise system during the transaction.

---

<sup>3</sup> The examples given here are scheduled calendar holidays. Loads may unexpectedly increase due to situations such as 9/11.

## 1.7 SUMMARY

Enterprises are large-growing organizations whose businesses could be transacted globally and whose needs are dynamically changing. The requirements of these enterprises are dynamic in nature. Moreover, the requirements are growing and the changes in the requirements are accelerating in nature.

Most of the large enterprises have successfully adopted IT systems and applications to meet the challenges and complexities of the business. While the businesses needs are driving the advancement in the IT technology, advances in the IT technology is accelerating the enterprise growth in an exponential manner. This is leading enterprises into newer dimension of problems and issues.

Many enterprises need to carry over systems and applications from the previous era. These legacy systems and applications could be very valuable to the business needs of the organization and need to be integrated with the newer technology. These growing enterprises need to quickly and efficiently get adjusted to the newer environments, efficiently integrate the legacy systems, and meet several crucial systemic qualities for sustenance and growth in the future.

## 1.8 REVIEW QUESTIONS

1. What is an enterprise?
2. What constitutes Legacy Systems?
3. Why are the requirements of the enterprises changing?
4. Why do enterprises consider it important to integrate legacy systems to the newer enterprise applications?
5. Which of the requirements are of importance to your enterprise?

# Index

## A

Abstract Persistence Schema 177  
 Abstract Windowing Toolkit (AWT) 99,102  
 Abstraction 31  
 Access control 145  
 Accessibility 33  
 ACID 263  
 Activating 175  
 addDateHeader() 137  
 addIntHeader() 137  
 Analytical 12  
 Applets 189,289  
 Application Assembler 160  
 Application clients 189  
 Application Deployer 160  
 Application Layer 43  
 Application Lifecycle events 125  
 Application Service 247  
 architect 22,28  
 architectural cube 215  
 Architecture Fundamentals 27  
 Architecture, Multi-tiered 63  
 Architecture, Single-tiered 63  
 Assumptions 29,35,285,298  
 Atomicity 263  
 Attributes 135  
 Auditing 271  
 Authentication 145,272  
 Authentication at the Business Tier 272

Authentication at the Web Tier 272  
 Authentication, Basic 272  
 Authentication, Digest 272  
 Authentication, Form-based 273  
 Authentication, HTTPS Mutual 273  
 Authorization 272,274  
 Authorization, Declarative 275  
 Authorization, Programmatic 275  
 Availability 33,47,114,289,301  
 Application Verification Kit (AVK) 79

## B

Bean Managed Persistence (BMP)  
     224,236,241,242,247  
 Bean, Session 163,224  
 bean, session, stateful 163,168,170,225,227,241,259  
 bean, session, stateless 163,166,168,225,226,241,259  
 Beans, Entity 163,172,175,224  
 Bean Managed Transaction (BMT) 266,267  
 begin 265,266  
 behaviour 28  
 Behavioural Design Pattern 215  
     Command 215  
     Chain of Responsibilities 215  
     Interpreter 215  
     Iterator 215  
     Memento 215  
     Mediator 215  
     Observer 215

**522** *Index*

- State 215
  - Strategy 215
  - Template 215
  - Visitor 215
  - best practices 75,220,240
  - Bidirectional 234
  - blueprint 28,66,75,76
  - BodyTag 159
  - Boundaries 31
  - Brittleness 31
  - Buffers 137
  - Business Components 223
  - Business Containers 223
  - Business-to-Consumer (B2C) 194
  - Business-to-Business (B2B) 11
  - Business-to-Consumer(B2C) 11
- C**
- Capabilities 31
  - Cascading Style Sheets (CSS) 185,190
  - Case Sensitivity 158
  - Census 4
  - Centralized Model 22
  - CGI 103
  - Client/Server 6,22,24,106,207,281
  - Clients, Applet 184,185
  - Clients, HTML 184,185,289
  - Clients, Java Application 184,188,289
  - Clients, MIDlet 184,185
  - Clients, Non-Java Application 184,189
  - Clients, Web Services 184
  - CMT 266,268
  - Comments 149
  - commit 265,266
  - Common Client Interface (CCI) 290
  - Compatibility Test Suite 76
  - components 114
  - Components, Business Tier 82
  - Components, Client Tier 80
  - Components, Presentation Tier 80
  - Composite Entity 247
  - Confidentiality 145
  - Consistency 263
  - Constraints 29,35,36,285,298
  - Consumer-to-Consumer (C2C) 194
  - Container Managed Persistence (CMP)
    - 160,177,224,237,238,241,242
  - Container Managed Relationship (CMR) 238
  - Container Managed Transaction 268
    - Required 268,269
    - Requires New 268,269
    - Supports 268,269
    - Not Supported 268,269
    - Mandatory 269
    - Never 269 271
  - Containers 83,114,126,160,189
  - Context Attributes 133,134
  - Controller 208,209,210,212,240
  - Cookies 135,137,142
  - CORBA 87,105,180,189,265
  - Creation 166,168,172,175
  - Creational Design Pattern 214
    - Abstract Factory 214
    - Factory Method 214
    - Builder 214
    - Prototype 214
    - Singleton 214
  - CTS 76,77
  - Cube, Architect's 41
  - Cube, Architecture 41
  - Custom Tags 203
  - CVM 107
  - CAD 7
  - CAE 7
  - CAM 7
  - CICS 11,281,283,284,288,290,291
  - COBOL 23,281,283,284,288,290,291

**D**

Data Access Objects (DAO) 223,239,243,244,288  
 Data Integrity 145  
 Database 85,255  
 Databases, Hierarchical 13  
 Databases, Network 13,14  
 Delegation model 102  
 Deployment Descriptor 125,126,146  
 Deployment Patterns 213  
 Design Pattern 213,299  
     Front Controller 301  
     Session facade 301  
     Business delegate 301  
     Data Access Objects (DAO) 301  
     Value Objects (VO) 301  
 Design Patterns in Business Tier 245  
     Business Delegate 245  
     Service Locator 245  
     Session Façade 245  
     Application Service 245  
     Composite Entity 245  
     Transfer Object 245, 248  
     Transfer Object Assembler 245,248  
     Value List Handler 245,248  
 destroy() 129,130,131,141  
 Destruction of MDB 176  
 Distributed 6,22,220  
 doAfterBody() 159  
 doEndTag() 159  
 doFilter() 141,142  
 doGet() 201  
 DOM 88  
 doPost() 201  
 doStartTag() 159  
 Durability 263  
 DBMS 11

**E**

EIS 257,258,302  
     Application Systems 252,253  
     Database Systems 252,253  
     Legacy Systems 252,253  
 EIS Access Objects 258  
 EJB Container Provider 161  
 EJB Invocation Model 164  
 EJB QL 177,238,239  
 EJB Roles 159,160  
 EJB Server Provider 161  
 ejbActivate() 170,175  
 ejbCreate() 166,168,169,173,175  
 ejbFind() 173  
 ejbHome() 173  
 ejbLoad() 174,175  
 ejbPassivate() 170,174  
 ejbPostCreate() 173  
 ejbRemove 168,170,175,176  
 ejbStore() 174  
 Enterprise Application Qualities 42,59  
 Enterprise Bean Provider 160  
 Enterprise JavaBeans (EJB)  
 72,73,106,111,122,159,220,288,289  
 EntityContext 173  
 Environment, Browser 190  
 Environment, Java 190  
 Environment, Mobile 190  
 ERP 88,255,282,284,285,286,288,290,302  
 EVAL\_BODY\_AGAIN 159  
 EVAL\_BODY\_INCLUDE 159  
 EVAL\_PAGE 159  
 expression 157  
 Extensibility 47,64,290

**F**

Fast 12  
 Filter 73

**524** *Index*

Filter Lifecycle 140  
 FilterChain 142  
 FilterConfig 141  
 Filtering 125,139  
 Filters 140,195,200,201,289,301  
 Filters, Authentication 139  
 Filters, Encryption 139  
 Filters, Translation 139  
 find 164  
 finder 238  
 Flexibility 33,47,63,216,290  
 flushBuffer 138  
 Friction 31  
 FROM 238  
 functional requirement 28  
 FASMI 12

**G**

GenericServlet 124,125,200  
 GET 198,201  
 getAttribute() 135,144  
 getAttributeNames() 135  
 getBufferSize() 138  
 getCookies() 137  
 getDateHeader() 136  
 getHeader() 136  
 getHeaders() 136  
 getHeadersNames() 136  
 getID() 144  
 getInitParameter() 134  
 getInitParameterNames() 134  
 getIntHeader() 136  
 getMaxInactiveInterval() 144  
 getRemoteUser() 145  
 getResource() 135  
 getResourceAsStream() 135  
 getUserPrincipal() 145  
 guidelines 75,220,240

**H**

Hardware Layer 43  
 Headers 135,137  
 HTTP gateway 186  
 HTTP Sessions 142,144  
 HttpServlet 124,125,200  
 HttpServletRequest 135,136,137,142  
 HttpServletResponse 137,138,139,142  
 HttpSession type 152  
 HttpSessionAttributeListener 145  
 HTML 36,146,147,201,212  
 HTTP 37,86,198,212  
 HTTPS 37,86,272

**I**

IMAP 105  
 Implicit Objects 152  
 include 149  
 include directive 157  
 Information 13  
 init() 129,141  
 Initialization parameters 133  
 Integration Tests 78  
 Interface, Component 162  
 Interface, Home 162  
 Interface, Local 162  
 Interface, Tag 159  
 Interoperability 33  
 invalidate() 144  
 isCommitted 138  
 isNew() 144  
 Isolation 263  
 isUserInRole() 145  
 IterationTag 159  
 IIOP 37,87,194,222  
 IP 24,37,105

**J**

J2EE Connector Architecture  
73,254,256,257,262,288

JAF 73,262,262

Java API for XML Processing (JAXP) 88

Java Community Process JCP 112

Java Database Connectivity (JDBC) 87,102

Java Foundation Classes (JFC) 105

Java Message Service (JMS) 87

Java Naming and Directory Interface (JNDI)  
87,106

Java Network Launching Protocol (JNLP) 188

Java Server Faces (JSF) 240

JavaServer Pages (JSP)  
72,73,110,111,146,195,200,202,  
222,259,288,289,301

Java Specification Requests (JSR) 112,123

Java Transaction API 265

JavaBeans 102,106,153,154,239,288

JavaBeans Activation Framework (JAF) 88

JavaMail 73,87,105,262

JAXR 38

JAX-RPC 38

JDBC 73,102,110,223,254,262,288

JMS 73,106,160,225,239,242,254,255,262

JNDI 73,111,164,223,262

JSP Custom Tag 206

JSP documents 157

JSP engines 120

JSP Standard Action 152,204

jsp.declaration 157

jsp.directive.include 157

jsp.directive.page 157

jsp.expression 157

jsp.scriptlet 157

jsp:fallback 156

jsp:forward 155,156

jsp:include 155

jsp:param 155,156

jsp:plugin 156

jsp.setProperty 154

jspDestroy() 148

jspInit() 148

JSPs 121

JTA 73,87,124,262,265

JTS 73,106,111,124,262,265

JAAS 88,263

JPS 76,111

JVM 99,185,189,190

\_jspService() 148,150

**K**

KVM 107,186

**L**

Layering 31,32

Layers 42

ledger machine 4

Legacy 9

Lifecycle of an Entity Bean 171

Local interfaces 177

Lower Platform 43

LAN 24,52,53,54,55,281

**M**

Mainframe 22

Manageability 33,61,114

Many-to-Many Relationships 233

Materials Requirements Planning (MRP) 255

Message Driven Bean (MDB)  
160,163,164,175,224,225,239,242,267

Message Protecting 271,275

**526** *Index*

Middleware 96  
 MIDlets 180,186,189,190  
 MIME 181  
 Model 208,209,210,211,240  
 Model 1 201,208,210,240  
 Model 2 210,221,286,289,291,299,303  
 Model-View-Controller (MVC)  
 201,206,207,210,286,289,291,299,303  
 Modularity 34  
 MOM 106  
 Multi-threaded Servlet 127  
 Multi-threading 99  
 MVC Architecture 206,212

**N**

Nesting Rules 158

**O**

Object, Application 152  
 Object, Dependent 247  
 Object, Entity 164  
 Object, out 152  
 Object, pageContext 152  
 Object, parent 247  
 Object, request 152  
 Object, response 152  
 Object, session 152,163  
 Object, Transaction Service (OTS)  
 106,111,189,265  
 One-to-Many Relationships 233  
 One-to-One Relationships 233  
 onMessage() 176  
 Open Database Connectivity (ODBC) 106  
 OLAP 9,11,12,15,37,254,281,283,290  
 OLTP 9,11,12,37,254,281,283,290

**P**

page 149  
 page directive 157  
 Passivating 174  
 Pattern, Behavioural Design 214  
 Pattern, Creational Design 214  
 Pattern, Structural Design 214  
 Pattern, Presentation Tier Design 216  
 Pattern, Business Tier Design 216  
 Pattern, Integration Tier Design 216  
 Patterns 213  
 Patterns, Architectural 213  
 Patterns, Deployment 213  
 Patterns, Design 213  
 Performance 33,47,50,216  
 Performance, Communicational 51,52  
 Performance, Computational 51  
 Phase, Execution 147  
 Phase, Translation 147  
 POJO 239,247  
 Poratability 33  
 Presentation Tier Design Patterns  
     Front Controller 216,217  
     View Helper 216,217  
     Composite View 216,217  
     Dispatcher View 216,217  
     Service to Worker 216,217  
 primary key 232

**R**

reference implementation 74  
 Relationships 177,232  
     Cardinality 232  
     Directionality 232  
     Aggregation 232  
     Recursive 232  
     Referential Integrity 232  
 Reliability 33,47,56,114,289,301,303

removeAttribute() 135,144  
 Request 125,198,200,201  
 reset 138  
 resetBuffer 138  
 Resource Adapters 255  
 resource manager 264  
 Resource Manager Drivers 85  
 Resources 133  
 Response 125,137,198,200,201  
 Reference Implementation RI 12 74  
 Risks 29,35,36,285,298  
 rollback 265,266  
 RDBMS 13,59,230,236,237,242,254,265  
 RMI 37,87,194,222  
 RPC 37,87,102,212

## S

SAAJ 38  
 Sagas 264  
 SAX 88  
 Scalability 47,59,114,198,289,290,301,303  
 Scalability Horizontal 60  
 Scalability Vertical 60  
 Scope 150  
 Scope Application 152  
 Scope Page 150  
 Scope Request 151  
 Scope Session 151  
 Scripting Elements 203  
 scriptlet 157  
 Secure Socket Layer (SSL) 272  
 Security 33,47,49,114,125,145,271,301,303  
 Security Declarative 145  
 Security Programmatic 145  
 service() 129,131,135,139  
 Servlet Context 125,132,133,134  
 Servlet Destruction 128,131  
 servlet engines 120  
 Servlet Initiation 128,129  
 Servlet Interface 125

Servlet Services 128,131  
 ServletConfig 129  
 ServletContext 144,146,152  
 ServletContextAttributeListener 145  
 ServletContextListener 144  
 ServletRequest 135  
 ServletResponse 137  
 Servlets 72,73,110,121,195,199,200,222,  
 259,289,301  
 Session Synchronization 271  
 SessionContext 167,169  
 SessionListener 144  
 Sessions 125,142,152  
 setAttribute() 135,144  
 setBufferSize 138  
 setContentype() 139,201  
 setDateHeader() 137  
 setEntityContext 172,173  
 setHeader() 137  
 setIntHeader() 137  
 setMaxInactive 144  
 setMessageDrivenContext() 175  
 setSessionContext 166,168,169  
 Shared 13  
 Single-threaded Servlet 127  
 SingleThreadModel 131  
 SKIP\_BODY 159  
 SKIP\_PAGE 159  
 SMTP 105  
 SOAP 38,189  
 Standard Actions 203  
 statements 255  
 Structure, Empty Tag 158  
 Structure, Standard Tag 158  
 Structured Query Language (SQL) 236,237  
 Structural Design Pattern 214  
   Adapter 214  
   Bridge 214  
   Composite 214  
   Decorator 214  
   Facade 214  
   Flyweight 215

**528** *Index*

Proxy 215  
Struts 240,301  
Supply-Chain 113  
Surface Area 31,32  
System Administrator 161  
system qualities 29

**T**

Tag 159  
Tag Handler 159  
Tag Iteration 159  
Tag Library Descriptor (TLD) 158  
taglib 149,158  
Tags, Declaration 149  
Tags, Directive 149  
Tags, Expression 150  
Tags, Scriptlet 150  
Test, Runtime Verification 79  
Test, Static Verification 79  
Testability 47  
Tests API 78  
Tests, Signature 78  
Tier, Business 44,46  
Tier, Client 44,45,189  
Tier, Integration 46  
Tier, Presentation 44,46  
Tier, Resource 46  
Tiers 42  
Transaction, Chained 264  
Transaction, Manager 265  
Transaction, Nested 264  
Transaction Processing (TP) 255  
Transactional Application 264  
Transactional Resource 264  
TransactionManager 266  
transaction-oriented 220  
two-phase commit 265  
TCP 24,37,105

**U**

UDDI 38  
Unidirectional 233  
unsetEntity 173  
Upper Platform 43  
URL Rewriting 142  
UserTransaction 266  
UML 32,112

**V**

Value Objects (VO) 223,239,245  
View 207,208,209,210,211,212,216  
Virtual Machine 186  
Virtual Platform 43

**W**

WAN 24,52,53,54,55,281  
Web Components 200  
Web Container 199  
web services endpoint 163  
Web Tier 213  
WSDL 38  
WORA 99

**X**

X/Open XA 265  
XAResource 266  
XML 38,125,157,200