CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BCA - 404

BLOCK 1:
BASICS OF CLIENT SERVER ARCHITECTURE

Dr. Babasaheb Ambedkar Open University
Ahmedabad

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Er. Nishit Mathur

Language Editor
Prof. Jaipal Gaikwad

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Ms. Pinaz Driver
Ms. Tejashree Bhosale
Mr. Kiran Shinde
Mr. Prashant Tikone
Mr. Akshay Mirajkar

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Contents

BLOCK 3: CLIENT SERVER DEVELOPMENT

UNIT 1 CLIENT SERVER SYSTEM DEVELOPMENT SOFTWARE

Factors driving demand for Applications Software Development, Client/Server system Development methodology, Project management, Architecture Definition, System Development environment, Productivity Measures, CASE, Client/Server System Development –Hardware, Hardware/Network acquisition, PC level processing units, Unix, Unix workstation server Hardware, Data storage, Network interface Cards, Power protection Device.

UNIT 2 CLIENT/SERVER SYSTEM DEVELOPMENT

Service and Supports, System Administration, Availability, Serviceability, Software Distribution, performance network management issues, Case studies.

BLOCK 4: INTRODUCTION TO C#

UNIT 1 INTRODUCTION TO .NET

The .NET Framework: an Overview, Framework Components, Framework Versions, Types of Applications which can be developed using MS.NET, MS.NET Base Class Library, MS.NET Namespaces, MSIL/ Metadata and PE files., The Common Language Runtime (CLR), Managed Code, MS.NET Memory Management / Garbage Collection, Common Type System (CTS), Common Language Specification (CLS), Types of JIT Compilers, Security Manager

UNIT 2 ASP.NET AND C#P

Working with Standard Controls, Navigation Controls, Validation Controls, Login Controls, Introduction to ASP.NET Objects, Building the StyleSheet, Creating the Content Master, Adding Elements, Building the Site Navigation, Adding Authentication, Adding Content Pages, Working with Data, Using ASP.NET Web Services and WCF, Creating a simple ASP.NET Web Service

Dr. Babasaheb
Ambedkar
Open University

BCA - 404

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BLOCK 1: BASICS OF CLIENT SERVER ARCHITECTURE

BLOCK 1: BASICS OF CLIENT SERVER ARCHITECTURE

Block Introduction

Client/server architecture is producer consumer computing arrangement where server will work as producer and client will work as consumer. Such type of arrangement will work when client computer sends resource or process request to server over network connection, which is further carries out and delivered to clients. In this block, we will detail about client/server environment with technology and importance with aspect of synchronization of application processing. The block will focus on the study and concept of Client/server architecture and explains the concepts related to functions and features of client and server modelling. You will be given an idea on various types of serves available with their usability and characteristics.

In this block, you will make to learn and understand about the basic of Object Request Broker with interference with CORBA. The concept related to Remote Procedure Call and detailed related client-server applications are explained to you. You will be demonstrated practically about the various arrangement of Client Server.

Block Objective

After learning this block, you will be able to understand:

- About Client Server Model
- Basic of Object Request Broker
- Features of CORBA
- Concept of synchronization Client server applications
- Detailed about different types of Servers

Block Structure

Unit 1: Introduction to Client/Server

Unit 2: Client/Server Architecture and Servers

UNIT 1: INTRODUCTION TO CLIENT/SERVER

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- About Client and Server
- About advantages of Client Server

1.1 Introduction

Client/server is a computing technology which is a good source for certain tools which allow employees with certain authority and responsibility. This computing technology is wide series in computer industry which covers all the aspects of computer. It requires mixture of skills which can be applied for development of certain client/server applications that will include:

- Database Design
- Transaction Processing

- Communication Skills
- Graphical User Interface Design And Development

Advanced applications require expertise of distributed objects and component infrastructures.

Client: A client is single user workstation which involves in presentation services, database services and connects interface for user communication to obtain business requirements.

Server: A server is multi user processors having high shared memory capacity that will provide connectivity and various database services with interfaces that can be applied in certain business processes.

1.2 Characteristics of Client and Server

Client/Server is a computing methodology that shows environment with technology in order to upgrade business processes with correct synchronization of related application processing which exists among client and server as shown in fig 1.1.

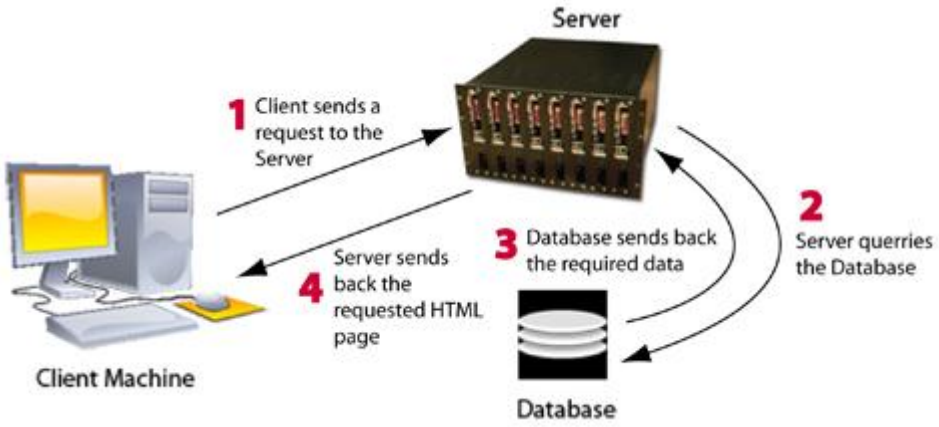


Fig 1.1 Client Server Arrangement

Client/Server model is a computing platform which is stand alone and moves towards cooperating of process or peer-to-peer modeling. It shows opportunity for users to work on business functionality which further will open in certain risky situations as it is crystal clear with technology and with the user.

Characteristics of Client/Server

There are certain characteristics features of clients and servers which have made them to work together on a network to do work.

Service:

It is seen that client/server exists as relationship among processes which are running on distributed devices. This methodology shows separation of functionalities which is subject to services that are offered.

Sharing of Resource:

A server is that part of computing which can take care of clients simultaneously thereby handling service access for resources.

Asymmetrical protocols:

It is noted that client/server acts as many-to-one relationship which is set off by clients by way of service request while server inertly wait for. Many times, client pass with reference to call back object on requesting for service.

Location Clarity:

It is found that process with the server lives inside a client or in machine all through the network. Further the Client/Server software arranges server location by redirecting service calls which makes the program as client/server or both.

Inter Communication:

Communication between clients and servers results by way of messaging. In this, both will interact through messages to deliver service requests and responses.

Encapsulation of services:

It is found that server has certain specialised features which can able to satisfy client requests varyingly and in due course of time will manage to upgrade them without affecting environment.

Scalability:

The Client/Server systems can be scaled horizontally or vertically as they can add or remove client workstations with certain performance effect. It can move with other efficient servers and can share work load with various servers in case of vertical scaling.

Integrity:

As, both server data and codes can be handled centrally, so it uses less maintenance cost with shared data consistency and not depending on clients.

Check your progress 1

1. What is server?
   a. Single user system
   b. Multi user processors
   c. Both of these
   d. None of these
2. Which of the following is the characteristics of client server architecture?
   a. Scalability
   b. Integrity
   c. Resource Sharing
   d. All of these

1.3 Merits and Demerits of the Client Server

Merits

Centralized control:

It is noted that access, resources and data integrity is handled by particular server in order to save the system from unauthorized client. Such type of centralized facilitates will update data or resources.

Scalability:

Another features of clients/servers is ability to increase clients and servers capacity separately. In this, every element can increase at any time or can add fresh nodes to network.

Easy maintenance:

In client server modelling, by distributing work and responsibilities to various standalone computers will help in changing, maintenance, upgradation and can shift server without affecting the customers.

Demerits

Congestion of Traffic:

In client/server, there is a problem of traffic congestion which results when large number of concurrent clients sends requests to similar server which can cause certain related problems.

Network robustness:

It is found that client/server theory has no robustness in case of P2P network. It is found that in case of server failure, customer will not be able to receive answer for their requests. It is found that P2P networks resources are distributed across many nodes of network.

Not Cost effective:

Software and hardware of server is an important concept which cannot be easily handled by regular computer hardware staff. For this, the server requires specific software and hardware especially on server side to work which will increase in cost.

Availability of resources:

It is found that client does not carry resources which are present on server. In case of web application, it is hard to write the data or information on storage devices attached with the client nor it is easy to take the print of the document unless seeing the preview version in the browser.

Check your progress 2

1. The main disadvantage of using client server architecture is_____.
   a. Robustness
   b. Ease of maintenance
   c. Centralized control
   d. Scalability
2. What is meant by congestion?
   a. A software
   b. Client side program
   c. When network is carrying much more requests than it can handle, network congestion occurs.
   d. None of these

1.4 Let Us Sum Up

In this unit we have learnt that client/server acts as computing technology that serves as good source for certain tools that allow employees with certain authority and responsibility.

It is noted that client/server shows environment with technology so as to upgrade business processes with correct synchronization of related application processing which exists among client and server

1.5 Answers for Check Your Progress

Check your progress 1

Answers: (1-b), (2-d)

Check your progress 2

Answers: (1-a), (2-c)

1.6 Glossary

1. **API -** The interface by which an application program accesses operating system and other services.
2. **Client/Server -** The model of interaction in a distributed system in which a program at one side sends a request to a program at another site and awaits a response
3. **CORBA -** It defines IDL and APIs that enable client/server object interaction within a specific implementation of an ORB.
4. **Graphical User Interface -** The use of pictures rather than just words to represent the input and output of a program.

1.7 Assignment

Explain the concept of Client/Server?

1.8 Activities

Study about the characteristics of Client/Server applications?

1.9 Case Study

Study the various types of client server application in daily life?

1.10 Further Readings

1. Orfali, Robert, Dan Harkey. Client/server programming with OS/2, Van Nostrand Reinhold, 1992.
2. Dewire, Dawna Travis, Client/server computing, McGraw-Hill, 1993.
3. Renaud, Paul E., 1957, Introduction to Client/Server Systems, 2nd Edition, Wiley, c1996.

UNIT 2: CLIENT/SERVER ARCHITECTURE AND SERVERS

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- About Client Server Architectures
- About Stored Procedure
- About Remote Procedure Call

2.1 Introduction

Client/server Architecture is a technology which differentiates computers and application software in two categories:

- Clients
- Servers

Such type of arrangement is present in order to employ better availability of computing resources along with shared data processing loads. In this, client computer gives user interaction facility along with some application processing, while server gives high volume storage capacity with heavy data crunching with good quality resolution graphics. Normally it is visualised that many client computers gets connected by network to server that works in case of big PC, minicomputer or mainframes.

Fig 2.1 Network

Client/server architecture is a producer-consumer computing architecture where the server acts as the producer and the client as a consumer. The server houses and provides high-end, computing-intensive services to the client on demand. These services can include applications access, storage, file sharing, printer access and/or direct access to the server's raw computing power.

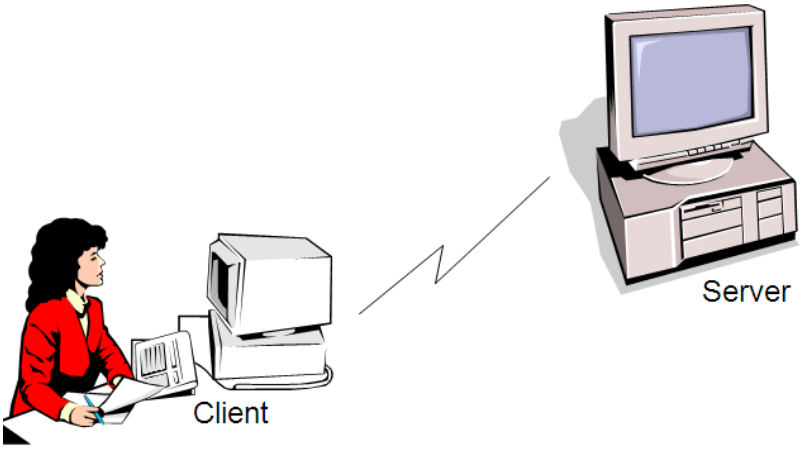


Fig 2.2 Client server

Client/server architecture works when the client computer sends a resource or process request to the server over the network connection, which is then processed and delivered to the client. A server computer can manage several clients simultaneously, whereas one client can be connected to several servers at a time, each providing a different set of services. In its simplest form, the Internet is also based on client/server architecture where the Web server serves many simultaneous users with Web page and or website data.

2.2 Types of Servers

Different servers do different jobs, from serving email and video to protecting internal networks and hosting Web sites. Servers are often dedicated, meaning that they perform no other tasks besides their server tasks. There are many types of servers available:

- Application Servers
- Audio / Video Servers
- Chat Servers
- Fax Servers
- FTP Servers
- Groupware Servers
- IRC Servers
- List Servers
- Mail Servers
- News Servers
- Proxy Servers
- Telnet Servers
- Web Servers
- Z39.50 Servers

The explanation of some of the Servers is shown below:

Proxy Server: It is a type of server which is placed in between client program and the external server which performs filter requests, increased performance and sharing of connections.

Mail Server: It is as important as Web servers that is responsible for movement and storing of mails either on LAN or WAN across Internet.

Server Platforms: It is a form of platform for hardware or software and works as an engine in order to drive the server.

Web Server: It is a fixed content server which serves to Web browser by way of uploading a file from computer disk and made available for users across the network.

Application Server: It is a middleware server which carry large amount of computing among database and end user servers.

Real-Time Communication Server: It is commonly called as chat servers or IRC Servers. It is also a instant messaging (IM) server which allow ample of users to exchange their information at the same time.

FTP Server: It is called as File Transfer Protocol which moves one or more files safely among computers along with file security and secure transfer control.

Collaboration Server: It is a groupware server that shows actual power of the Web. Such type of server software allows users to collaborate immaterial of their location with the use of Internet or intranet and helps them to work together virtual atmosphere.

List Server: It is a good way to handle mailing lists with interactive discussions which is public or single list which transfers announcements, newsletters or advertising.

Telnet Server: It allow users to log on to host computer where they can do their working on remote computer itself.

Open Source Server: It is a form of original open source server software which will help in getting work done. It is an important part of several IT infrastructures.

Check your progress 1

1. What is the use of FTP server?

   a. It is open source server software to do the work

   b. It is used to handle mailing lists

   c. It is used to transfer files safely among computers securely

   d. None of these

2. Which server is used for sending and receiving requests from web server?
   a. Collaboration server
   b. Web server
   c. FTP server
   d. None of these

2.3 ORB

The Object Request Broker (ORB) is middleware that uses the CORBA specification. The Object Request Broker (ORB) manages interaction between clients and servers.

The Object Request Broker or ORB takes care of all of the details involved in routing a request from client to object, and routing the response to its destination. The ORB is also the custodian of the Interface Repository, an OMG-standardized distributed database containing OMG IDL interface definitions. This includes the distributed computing responsibilities of location, referencing and 'marshalling' of parameters and results.

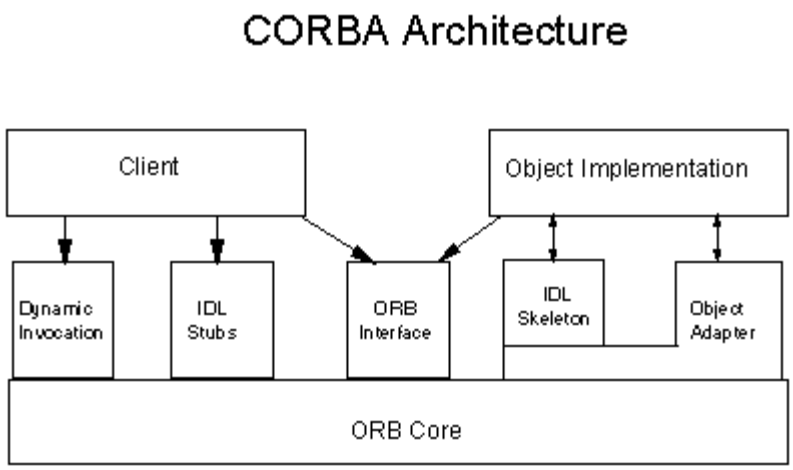


Fig 2.1 CORBA Architecture

The CORBA specification defines an architecture of interfaces and services that must be provided by the ORB, no implementation details. These are modular components so different implementations could be used, satisfying the needs of different platforms. The ORB manages the interactions between clients and object implementations. Clients issue requests and invoke methods of object implementations.

On the client side, then, the ORB provides interface definitions from the IFR, and constructs invocations for use with the Dynamic Invocation Interface (DII). It also converts Object References between session and stringified format, and converts URL-format corbaloc and corbaname object references to session references.

On the server side, the ORB de-activates inactive objects, and re-activates them whenever a request comes in. CORBA supports a number of activation patterns, so that different object or component types can activate and de-activate in the way that uses resources best.

Check your progress 2

1. CORBA stands for_____.
   a. Common object request broker architecture
   b. Computing objects request broker architecture
   c. Common object response broker architecture
   d. None of these
2. Which of the following serve as the glue between the client and server applications respectively, and object request Broker(ORB) ?
   a. ORB and ORB Interface
   b. CORBA IDL stubs and skeletons
   c. Client and server
   d. None of these

2.4 Client Server Architectures

Client-server architecture is an arrangement of computer network where clients request and receive service from the host server as shown in fig 2.2.

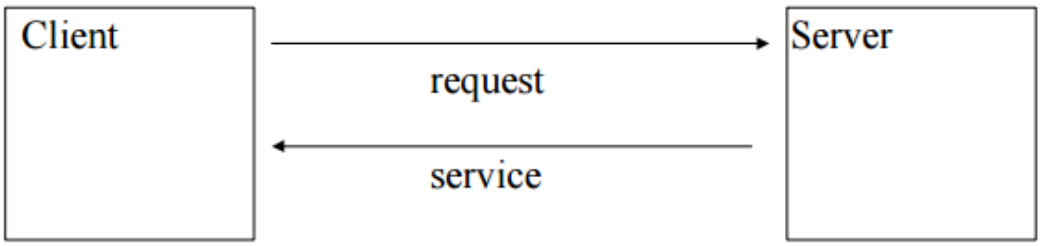


Fig 2.2 Client server Interaction

In this, we see that client computers shows interface which allow user to request for services of server whose results gets displayed. In this, the server will wait for requests to arrive from clients and then further responds it back. Also a server shows standardized clear interface to clients which clients might not be aware of ware-about of system. Clients are located at workstations or on personal computers, while servers are there anywhere on the network. Such type of model is particularly successful when clients and server each have different tasks which they used to do every day.

It is noted that several clients will able to access server's information at the same time when client computer do other tasks. Since client and server computers are intelligent devices, so we can analyse that such model is different from mainframe model where common mainframe computer do all tasks for dumb terminals.

Fig 2.3 shows component diagram where server component performs certain operations that are precised in Services interface and also the client component which rely on certain such services.

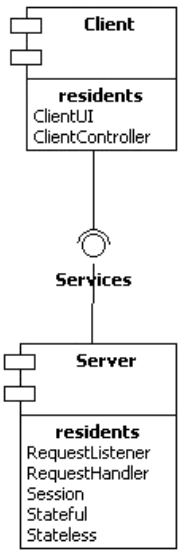


Fig 2.3 component diagram

We see that, internally client component has ClientUI which directs user requests to client controller. In this, the controller will forward the request which occurs across a process or machine to RequestListener, which is located inside the server. Here the listener, that works as master which forms RequestHandler slave and forwards the request to it:

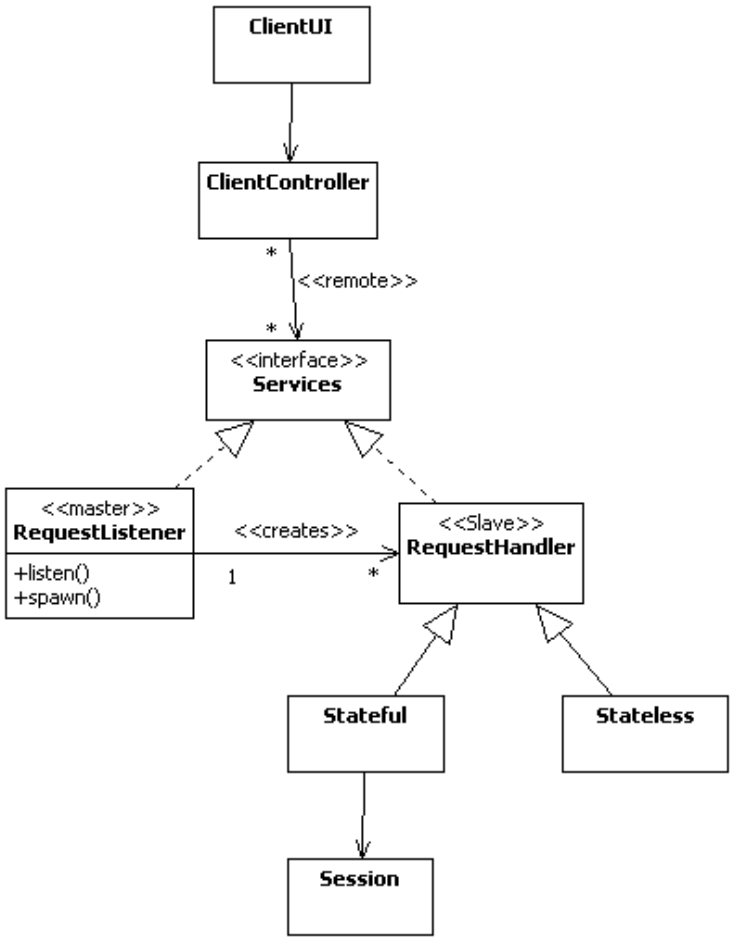


Fig 2.4 ClientUI

We see that in fig 2.5, there appears certain sequence which shows typical client-server interaction:

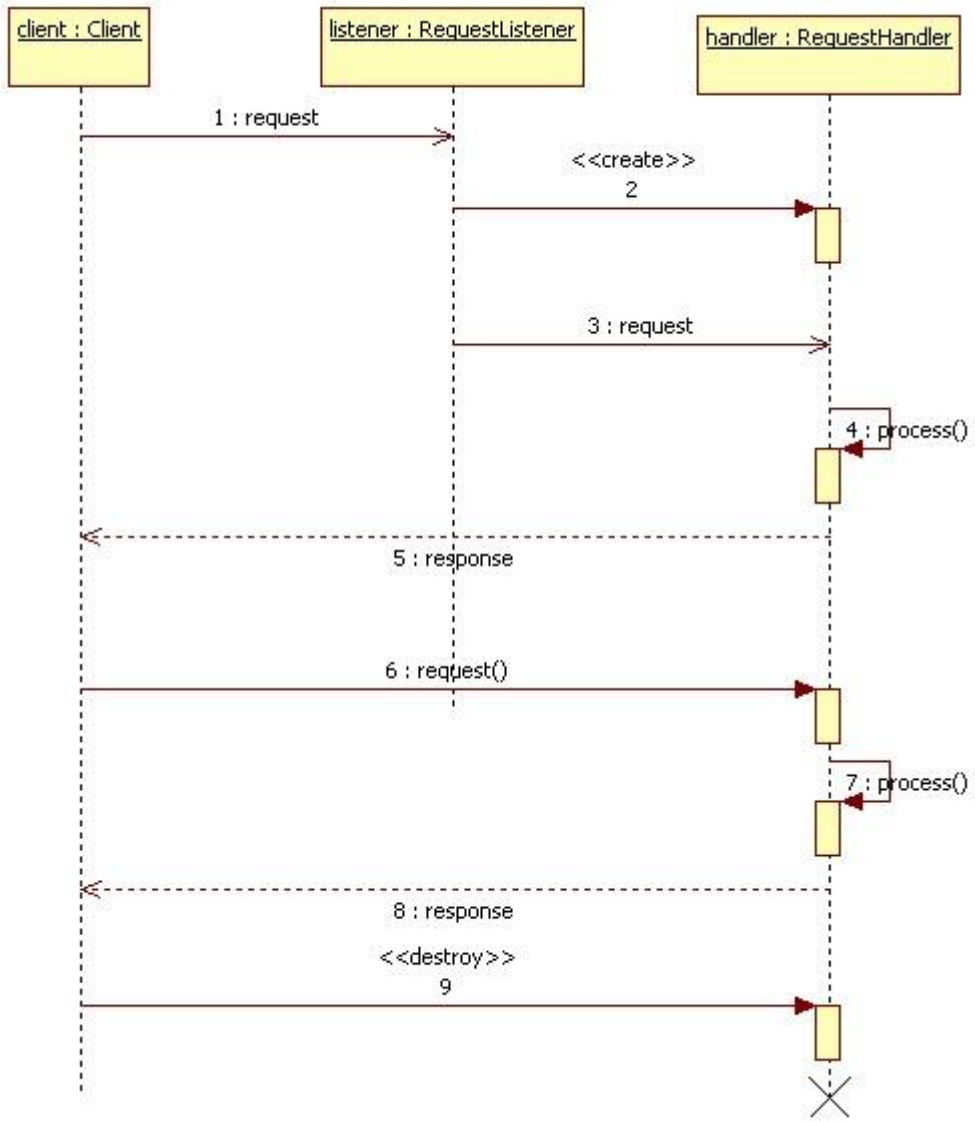


Fig 2.5 client-server interaction

Check your progress 3

1. The Client can_____.
   a. Send request to server
   b. Send response to server
   c. Receive request from server
   d. None of these

2.5 Stored Procedure

In Client server, stored procedure is a process of factorizing of code which ease the development and maintenance of applications. It is found that when a method is applied to access data change, in such case, the stored programs gets modified on the server without doing any sort of alteration in deployed applications which uses particular procedure. As noted, stored procedure also is applied to restrict the number of back and forth processes which occurs among client computer and server that finally helps in increasing the speed of processes.

Features of Stored Procedures:

- They result in improved performance since the database will able to optimize data access plan which are used by procedure and cache for frequent applications.
- They can be safely kept in database where a client allows to work on stored procedure without any request.
- They can be safely and simply maintain as it quickly modify and can alter hard code statements inside a component.
- They add extra abstraction from required data schema.
- They lowers network traffic since statements can be worked out in batches instead of sending multiple requests from client.

Check your progress 4

1. What are the features of stored procedure?
   a. Client can work on stored procedure without any request.
   b. It can be quickly modify
   c. Database can be able to optimize data access plan which are used by procedure
   d. All of these

2.6 Remote Procedure Call (RPC)

RPC or Remote Procedure Call is a powerful technique for constructing distributed, client-server based applications. It is based on extension of conventional notion or local procedure calling where called procedure is not present in same address space as calling procedure. It is seen that here the two processes will not on similar system or can be on different systems with network connection. With the use of RPC, programmers of distributed applications tries to avoid details about network interface. Here, transport independence of RPC gets isolated with application which result from physical and logical elements of data communications and makes the application to be applied on different transports.

Remote Procedure Call shows different concept for accessing network services. As a replacement remote access which carried out by sending and receiving of messages where client call upon services acts in making local procedure call. The local procedure hides the details of the network communication.

While making remote procedure call:

- The calling environment gets wiped off with transferring of procedure parameters throughout the network in an environment where the procedure is carried out.
- On completion of procedure and generation of results, the results gets transferred back to calling environment where execution start again as if returning from standard procedure call.

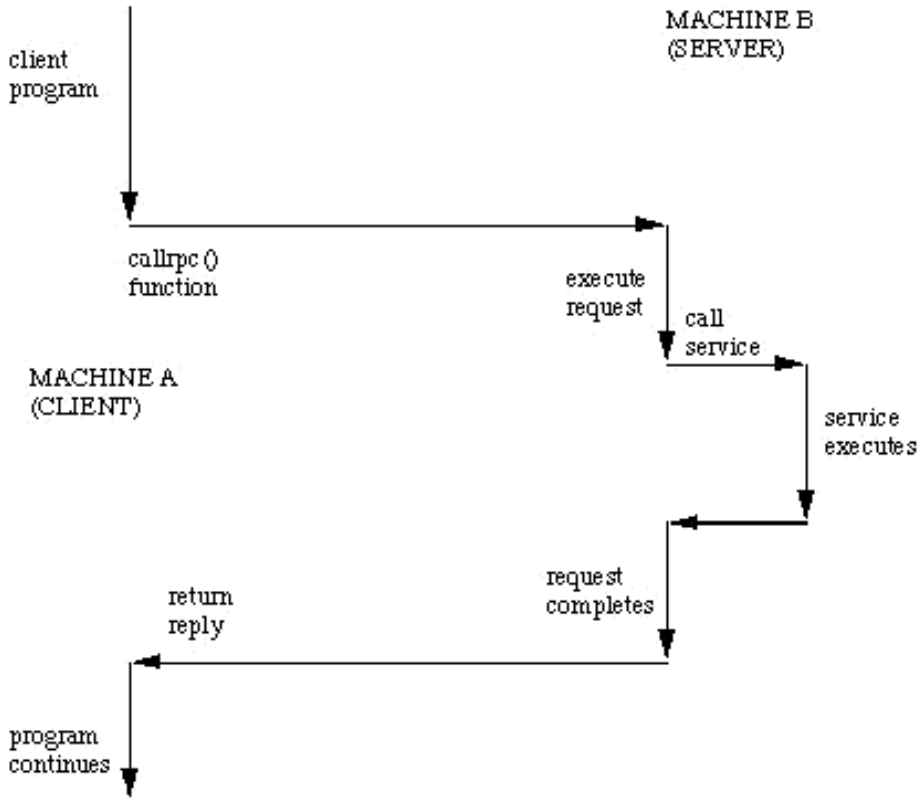


Fig 2.6 Remote Procedure Calling Mechanism

Goals of RPC:

The main idea behind RPC is to cover the presence of network from program. While doing this, the RPC doesn't fully fit in OSI model as:

- Message passing of network communication gets hidden from user where user fails to initially open up the connection, read and write data, and further close it.
- It often leaves out various protocol layers so as to upgrade the performance. It is found that the small performance improvement is useful as a program can remove RPCs.

It is noted that the RPC is well fit for client-server interaction where flow of control interchanges among the caller and callee. Logically, client and server will not work on both at same time, rather thread of execution jumps from caller to callee and then back again. The following steps take place during an RPC:

- A client call upon client stub procedure and pass the parameter in usual way that placed inside the client's own address space.

- The client stub will convert parameters into messages. It changes the representation of parameters into standard format and copy all parameters in message.
- The client stub will allow message to be there on transport layer, that sends further to remote server machine.
- On server, transport layer will move the message to server stub which will cancel all parameters and further will call required server routine by regular procedure call concept.
- On completion of server procedure, it comes back to server stub, which converts return values in message. Further the server stub will transfer the message to transport layer.
- Transport layer will allow result message back to client transport layer and transfers message back to client stub.

It is noted that every RPC will take place in terms of thread which is a sequential flow of control from single execution point at any time. A thread created and managed by application code is an application thread.

The applications of RPC make use of application threads in order to issue RPCs and RPC run-time calls. The RPC client will have one or more client application threads that performs one or more RPCs.

To work with remote procedures, RPC server uses one or many call threads which is given by RPC run-time system. On starting, server application thread will show maximum number of simultaneous calls it carries out. It is found that single threaded applications carry out maximum of single call thread. The maximum number of call threads in case of multi threaded applications will depend on layout of an application and working of RPC policies. It is noted that RPC run-time system will form the call threads during server execution.

On extension of RPC across client and server execution works out when client application thread will call upon remote procedure that is part of logical thread as RPC thread. The structure of RPC thread is logical which shows various phases of RPC since it extends across actual threads execution and network. After making an RPC, calling client application thread becomes part of RPC thread.

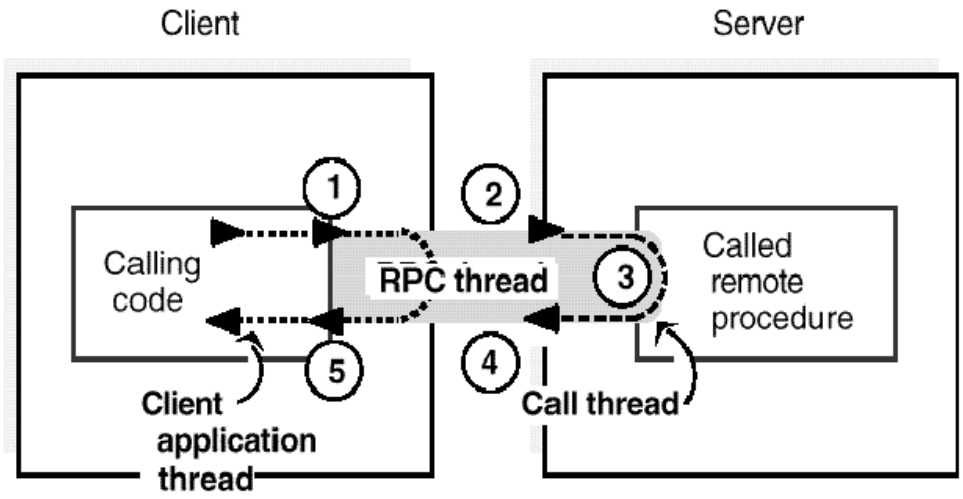


Fig 2.7 Execution Phases of an RPC Thread

Check your progress 5

1. A Remote Procedure Call is initiated by the _____.
   a. Server
   b. Client
   c. Both of these
   d. None of these
2. Remote procedure calls is _____.
   a. Inter-process communication
   b. A single process
   c. Both of these
   d. None of these
3. Why RPC is used to
   a. To establish a server on remote machine that can respond to queries
   b. To retrieve information by calling a query
   c. Both A and B
   d. None of these

2.7 Let Us Sum Up

While studying this unit, we learnt that client/server architecture is a producer consumer computing architecture where server acts as producer and client acts as a consumer. In the Client/server architecture, the working of system starts when client computer sends a resource or process request to server over network connection, which is processed and delivered to client.

It is studied that there are different servers which does different jobs right from serving email and video to protecting of internal networks and hosting of Web pages and sites. The Object Request Broker is a middleware which uses CORBA specification and handles interaction which exists among clients and servers.

The Remote Procedure Call is a powerful technique for constructing distributed, client-server based applications

2.8 Answers for Check Your Progress

Check your progress 1

Answers: (1 –c), (2-b)

Check your progress 2

Answers: (1 -a), (2-b)

Check your progress 3

Answers: (1 –a)

Check your progress 4

Answers: (1 -d)

Check your progress 5

Answers: (1 –b), (2-a), (3-c)

2.9 Glossary

1. **API -** The interface by which an application program accesses operating system and other services.
2. **Client/Server -** The model of interaction in a distributed system in which a program at one side sends a request to a program at another site and awaits a response
3. **CORBA -** It defines IDL and APIs that enable client/server object interaction within a specific implementation of an ORB.
4. **Graphical User Interface -** The use of pictures rather than just words to represent the input and output of a program.

2.10 Assignment

Write short note on features of stored Procedure in Client server Model?

2.11 Activities

Collect some information on Client server Architecture arrangement.

2.12 Case Study

Generalised the basic necessity of RPC in Client server Modelling?

2.13 Further Readings

1. Orfali, Robert, Dan Harkey. Client/server programming with OS/2, Van Nostrand Reinhold, 1992.
2. Dewire, Dawna Travis, Client/server computing, McGraw-Hill, 1993.
3. Renaud, Paul E., 1957, Introduction to Client/Server Systems, 2nd Edition, Wiley, 1996.

Block Summary

In this block, you have learnt and understand about the basic of client/server computing technology and its related features. The block gives an idea on study and concept of Object Request Broker interaction with client and server. You have explained on the concepts of various servers with their uses.

The block detailed about the basic of Remote Procedure Call technique in terms of client-server applications. The concept related to CORBA specification are also well explained to you. You will be demonstrated practically about Client server specifications used in modelling.

Block Assignment

Short Answer Questions

1. What are the merits of Remote Procedure Call?
2. Explain the function of producer-consumer computing architecture.
3. List the type of servers used nowadays.
4. Write short note CORBA.

Long Answer Questions

1. Write short notes on Client/server architecture.
2. Write short note on Remote Procedure Call.
3. Write note on Object Request Broker.

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

……

Education is something which ought to be brought within the reach of every one.

- Dr. B. R. Ambedkar

Dr. Babasaheb Ambedkar Open University
'Jyotirmay Parisar', Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,
Ahmedabad-382 481.

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BCA - 404

BLOCK 2:
CLIENT SIDE AND SERVER SIDE SERVICES

स्वाध्यायः परमं तपः

Dr. Babasaheb Ambedkar Open University
Ahmedabad

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Er. Nishit Mathur

Language Editor
Prof. Jaipal Gaikwad

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Ms. Pinaz Driver
Ms. Tejashree Bhosale
Mr. Kiran Shinde
Mr. Prashant Tikone
Mr. Akshay Mirajkar

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Contents

BLOCK 3: CLIENT SERVER DEVELOPMENT

UNIT 1 CLIENT SERVER SYSTEM DEVELOPMENT SOFTWARE

Factors driving demand for Applications Software Development, Client/Server system Development methodology, Project management, Architecture Definition, System Development environment, Productivity Measures, CASE, Client/Server System Development –Hardware, Hardware/Network acquisition, PC level processing units, Unix, Unix workstation server Hardware, Data storage, Network interface Cards, Power protection Device.

UNIT 2 CLIENT/SERVER SYSTEM DEVELOPMENT

Service and Supports, System Administration, Availability, Serviceability, Software Distribution, performance network management issues, Case studies.

BLOCK 4: INTRODUCTION TO C#

UNIT 1 INTRODUCTION TO .NET

The .NET Framework: an Overview, Framework Components, Framework Versions, Types of Applications which can be developed using MS.NET, MS.NET Base Class Library, MS.NET Namespaces, MSIL/ Metadata and PE files., The Common Language Runtime (CLR), Managed Code, MS.NET Memory Management / Garbage Collection, Common Type System (CTS), Common Language Specification (CLS), Types of JIT Compilers, Security Manager

UNIT 2 ASP.NET AND C#P

Working with Standard Controls, Navigation Controls, Validation Controls, Login Controls, Introduction to ASP.NET Objects, Building the StyleSheet, Creating the Content Master, Adding Elements, Building the Site Navigation, Adding Authentication, Adding Content Pages, Working with Data, Using ASP.NET Web Services and WCF, Creating a simple ASP.NET Web Service

Dr. Babasaheb
Ambedkar
Open University

BCA - 404

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BLOCK 2: CLIENT SIDE AND SERVER SIDE SERVICES

BLOCK 2: CLIENT SIDE AND SERVER SIDE SERVICES

Block Introduction

Services are requires in order to call and communicate among another. In certain application, services call upon another by way of language method or procedure calls. In a standard distributed system, services will run at particular locations which easily calls upon another with the help of HTTP/REST or with RPC mechanism. It is seen that latest micro service based application normally runs on virtual environments where number of instances of service and locations changes dynamically.

In this block, we will detail about the basic of Client side and server side applications with services. The block will focus on the study and concept of Client side services such as Services Print Services, Remote Services, Utility Services, Message Services, Network Services, Application Services and Database Services. You will give an idea on Object Linking and Embedding.

In this block, you will make to learn and understand about server side applications such as Request Processing, Print Services, Database Services, Security Services, File Services and Communication Services. The concept related to Dynamic Data Exchange (DDE) is explained to you.

Block Objective

After learning this block, you will be able to understand:

- About client/server
- Features of Client Services
- Characteristics of Dynamic Data Exchange
- Basic about Object Linking and Embedding
- Idea about several Client Tools
- Concept of GUI and Non GUI Clients
- Idea about Object Oriented User Interface Clients
- Features of Server Functionality

Client Side and Server Side Services

- Basic of Server Request and Print Services

Block Structure

Unit 1: Client Side Services

Unit 2: Server Side Services

UNIT 1: CLIENT SIDE SERVICES

Unit Structure

1.0 Learning Objectives

After learning this Unit, you will be able to understand:

- About different server services
- About Dynamic Data Exchange
- About Object Linking
- About Client Tools

1.1 Introduction

In client/server applications, there are many functions which are performed by combination of resources which can be done by client workstation processor as well as server workstation processor. The database server provides data in response to request which is given by client application where local processing by client involves finding of invoice amount and formatting of response to the workstation screen.

1.2 Services

It is noted that, client workstation will make business functions with the use of combination of personal productivity products which are mixed together with custom application. The services will have ability to cut and paste input from different sources. The client is mainly the consumer of services which is given by one or more server processors and provides services related to presentation. The user input and final output are presentation at client workstation. There are many more services which are performed on the client side.

1.2.1 Print Services

In Client side, the network operating system will allow client to generate the print requests inspite of the working printer is engaged. It is a form of software which redirect the request and manages the printing request. The client has facility to view the status of the print queue at any time and client is also notified when the print is completed. Similarly there are also fax services.

1.2.2 Remote Services

In remote service, the client appeals to application which is performed on remote server which includes all backup services. Here, applications related to downloading of data from host or checking list of stock process also calls upon that will workout remotely. Network Operating System will help in running applications on client workstation which begins such remote applications. The services related to remote application will cover working from home that allows people to work from home which makes them to communicate with office network.

1.2.3 Utility Services

In client, the operating system ease some local functions that are applied in order to do actions such as edit, copy, move, compare and help that are easily carried out at client side.

1.2.4 Message Services

In clients, the messages will be easily send or receive from network synchronously. Here the message services will deliver buffering, scheduling and settlement of certain services in order to handle such function.

1.2.5 Network Services

It is found that the client workstation will tend to communicate with network by use of certain protocol, which are packet of service and API's which forms, send and receive the required formatted network messages. These protocols that supports will normally include IPX, TCP/IP, NETBIOS, Ethernet, Token Ring and FDDI.

1.2.6 Application Services

Apart from remote execution of certain services which network system provide, there are certain other services like custom applications where they have their own API's which is fixed in RPC which call upon particular services form remote server.

1.2.7 Database Services

There are certain database service requests which are made in client side using particular syntax. With the use of certain standard language having standard form, certain similar application gets back on multiple platforms which are syntactical different.

Check your progress 1

1. Which of the following services can be performed at client side?
   a. Application Services
   b. Database Services
   c. Utility Services
   d. All of these
2. What is network service ?
   a. The client can generate print request using OS
   b. The client can generate database service request
   c. The client can communicate with network using TCP/IP, IPX,Token ring etc. protocols
   d. None of these

1.3 Dynamic Data Exchange (DDE)

DDE is Dynamic Data Exchange protocol which is built in Microsoft Windows where users can share their data among applications which can be on local machine or on Network. It is an application for Windows where communication is done through sharing of information in dynamic manner. Normally it is seen that DDE is applied to share data among two applications and helps in transferring commands to other application.

It is noted that when an application starts with DDE process, then such application is called as client since it’s ask information from other application which is the server as it gives services to DDE client. It is highlighted that a client and server communicating by DDE will never communicate directly among each other. They tends to transfer messages to Windows, which further passes

information to required destination. This shows that with wide scenario of DDE, almost all networks is not restricted to particular workstation. Through DDE, the Windows application can send data to application which is running on other workstation, provided that they are on same network.

Protocol

It is noted that with several communications, there appears to be rules which applies to find how and when data is transferred. Such rules will determine the protocol applied to manage transferring and exchanging of information. It is found that DDE needs to have its own protocol in order to see communication which exists among applications. Such type of protocol will describe the process applied to start and end DDE session by:

- Transferring data to other application
- Receiving of data from another application
- Instructing other application to work with macro or command

Apart from these there are relatively less parameters which constitute certain protocol that makes DDE communications elementary. Further, DDE will handle firm links which is created in order to notify to the client about certain server application about changing of data with request made by client to receive it. It is an attractive information when volume changes to server data where client needs not to be bothered about repetitive processing of information. On finding the blockage of server link at particular point, the firm link will work instead of hot link which ensures about presence of last data alteration. It is found that the request link be framed for direct copy and paste operations which happens among server and client without the requirement of intermediate clipboard.

Check your progress 2

1. What is Dynamic Data Exchange?
   a. Client side method
   b. It is a application which helps in sharing information in dynamic manner
   c. It is a Windows application
   d. Both B and C

1.4 Object Linking and Embedding (OLE)

OLE which is also known as object linking and embedding permit users to put the data together from various applications. In this, object link permits the users to distribute individual source of data for required object. The document carries particular name of file having data with pictures. As soon as the source will get updated, every documents using data gets updated automatically.

By means of object embedding, single application will show data or image which carries inside the document of other application. In this the destination application having data or images which unable to understand and can able to alter will show prints and plays embedded contents. To amend or bring up to date, such type of embedded object gets opened in basis application which made it. This can be done directly as soon as you make a double click on an item or selecting correct edit command at the time when object is highlighted.

OLE continues a distinguished way to conduct information from one application in another application, which can be beneficial for illustrations and more. To conduct OLE, you expect both source as well as destination applications that acknowledge OLE. Information from one document can be embedded into another document by matching arranging or implanting the information. Either linked as well as implanted OLE objects can be edited from within the destination application. Furthermore, linking as well as implanting deposit information discretely so it is critical to assure that you are utilizing the exact option for the arrangement. The connection between embedding as well as lining up is analogous to that between inserting a block as well as creating an external reference.

While embedding doesn't authorize users to have an individual source of data, it does develop it easier to encompass applications. An embedded object contains the actual data for the object, the name of the application that created it, and a picture of the data. The following Works objects can be embedded in a Word Processor document or a Database form:

- Works Spreadsheet
- Works Chart
- Microsoft Draw
- Clip art from the Microsoft ClipArt Gallary
- Microsoft Note – It

- Microsoft WordArt

Embedding OLE Objects

An implanted OLE object is clearly a representation of information from another account for instance, a copy of an Excel spreadsheet in AutoCAD Architecture. When you implant objects, several changes brought about to the source document are not analyzed in destination documents on account of there continues no link to the source document. You should implant objects only if you expect to be accomplished to conduct the application that commenced them for editing furthermore you do not assume the OLE object to be updated when you edit information in the source document.

Linking OLE Objects

A linked OLE object is simply a reference to information that is located in another document. Link objects when you want to use the same information in more than one document. If you change the original information, you only need to update the links in order to update the document that contains the OLE objects. Links can be set to update automatically. It is important to note that when you link a drawing you need to maintain access to both the source application and the linked document. If you rename or move either of them, you may need to re-establish the link.

Check your progress 3

1. The _____is a simple object model that use OLE DB, and is frequently used for database applications.
   a. ADO
   b. XML
   c. ODBC
   d. None of these
2. Which of the following is considered to be one of the foundations of data access in the Microsoft world?
   a. ADO
   b. ODBC
   c. OLE DB
   d. None of these

1.5 Client Tools

The Java client sharing has utility programs which was written by Java client. Such utility programs are present inside rabbitmq-client-tests.jar file. This jar file has many small programs which performs different functions of RabbitMQ server. The program with source code is in test/src folder which is inside source distribution.

The script runjava.{sh,bat} will run Java with class path that is configured correctly for certain examples, e.g. runjava.sh com.rabbitmq.examples.TestMain runs the TestMain functional tests.

PerfTest

It is earlier called as MulticastMain which serves as performance testing tool that begins with 0 or more producers along with consumers furthermore reports the rate at which messages are sent as well as received along with latency. This tool support many command line flags.

runjava.sh com.rabbitmq.examples.PerfTest --help

Examples:

```
runjava.sh com.rabbitmq.examples.PerfTest -a
```

It sends transient messages without receiving as 1 producer and 1 consumer.

```
runjava.sh com.rabbitmq.examples.PerfTest -c 1000
```

It sends transient messages with receiving and confirmations.

```
runjava.sh com.rabbitmq.examples.PerfTest -c 1000 -f persistent
```

It sends transient messages with receiving and confirmation along with persistence.

```
runjava.sh com.rabbitmq.examples.PerfTest -y0 -p -u my-queue -s 1000 -C
1000000
```

It will fill decided queue with 1M transient messages of 1kB each

```
runjava.sh com.rabbitmq.examples.PerfTest -x0 -y10 -p -u another-queue
```

It starts with 10 consumers from a decided queue without producers.

HTML Performance Tools

Such type of tool are group of tools which allow to run computerized standards by packaging about the PerfTest standardizing structure. We see that

benchmark specs can be provided and certain tools will monitor running of standard, collection of results along with showing them in HTML page.

Tracer

There are clear tracer which are simple AMQP protocol analyzer available in class com.rabbitmq.tools.Tracer.

- runjava.sh com.rabbitmq.tools.Tracer listenPort connectHost connectPort
- listenPort: port to listen for incoming AMQP connections on - defaults to 5673.
- connectHost: hostname to use when making an outbound connection in response to an incoming connection - defaults to localhost.
- connectPort: port number to use when making an outbound connection - defaults to 5672.

Check your progress 4

1. What is MulticastMain?
    1. It is another name of PerfTest
    2. It serves as performance testing tool that begins with 0
    3. It supports many command line flags
    4. All of these

1.6 GUI Clients

GUI is a Graphical User Interface that works with icons or indicators along with electronic devices. Interface is a coordination involved among computer, program and humans. It is seen that graphical user interface uses visual elements that will help in showing information which is kept inside computer that is simple to understand. These elements makes easy for people to work by involving certain computer software. It is found that in order to design a good user interface it should allow easy and natural interaction among user and system.

GUI client is a program where user interface with graphical objects which can be windows or menus. It is noted that AIX serves as Common Desktop Environment furthermore, AIX windows shows interfacing among you and computer. The graphical window system here that is part of graphical user

interface will arrange these graphics output for display and performs elementary text along with graphics for drawing certain functions. This Common Desktop Environment serves as graphical user interface which allows to work on network devices along with certain tools without having an idea about their location and can be exchange data across applications by drag and drop of certain objects. Sometimes it happens that an applications are occasionally requested to server result when it is being asked by human which can be any operating system.

Check your progress 5

1. What are the features of GUI client?

   a. GUI provides interfaces to users

   b. It uses visual elements that helps in showing information

   c. It provides a user friendly environment

   d. All of these

1.7 Non-GUI Clients

In Java, non GUI client are simply applications which generates server requests without the intervention of humans. Such type of client normally either needs multitasking or can work without multitasking. In case of Non GUI Client without multitasking will include:

- ATM Machines
- Barcode reader
- Mobile phones
- Fax

In such case the client will give minimum human interfacing for request which is in form of continuous loop. Apart from this, there are clients that requires multitasking which can be:

- Robots
- Testers

In such case the client requires continuous human interfacing for request.

Check your progress 6

1. Which of the following is the example of non GUI client that requires multitasking?
   a. Robots
   b. Testers
   c. ATMs
   d. none of these
2. What is meant by Non GUI client?
   a. It can generates server requests without human interventions
   b. It provides user interfaces
   c. It works on user commands
   d. None of these

1.8 OOUI (Object Oriented User Interface) Clients

Object-oriented user interface is a software interface in which users will work on certain objects for certain properties. It is developed on the idea of object-oriented programming applied in case of modern computing. It is an alternate to different function-oriented interfaces. OOUI clients are applications that are extremely iconic further which shows flawless admission to Information in visual formats.

Such type of interface does not request freely to server result from human involvement with GUI. It is a best fit for mainstream, OLTP type business application having continuous work along with excess strength. They are graphic renditions of dialogs that earlier executed on dump computer. It make use of object action model, which can be applied and handled by user for any object selection and request for any action to be done.

Client Side and Server Side Services

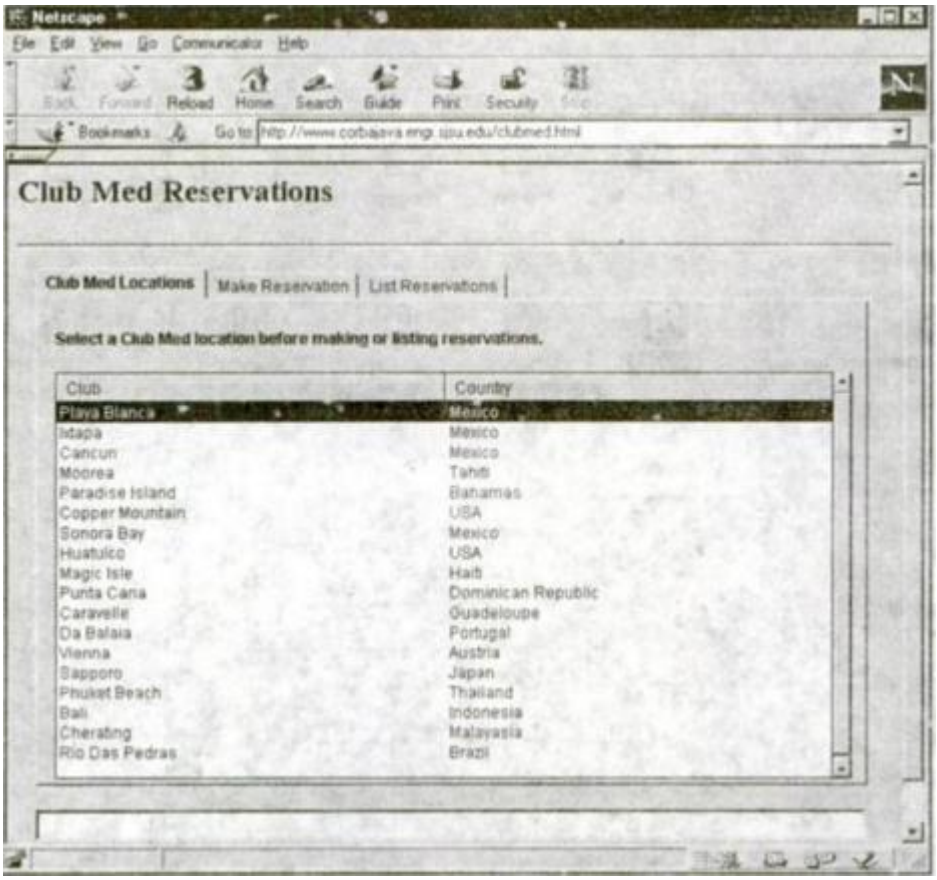


Fig 1.1 Application

It is seen that OOUI client provides iconic OOUI and allow to convert objects on windows screen by using drag and drop capability. This is mainly used by people those who want to perform many task without any sequence of output. These objects will interact among themselves along with external servers.

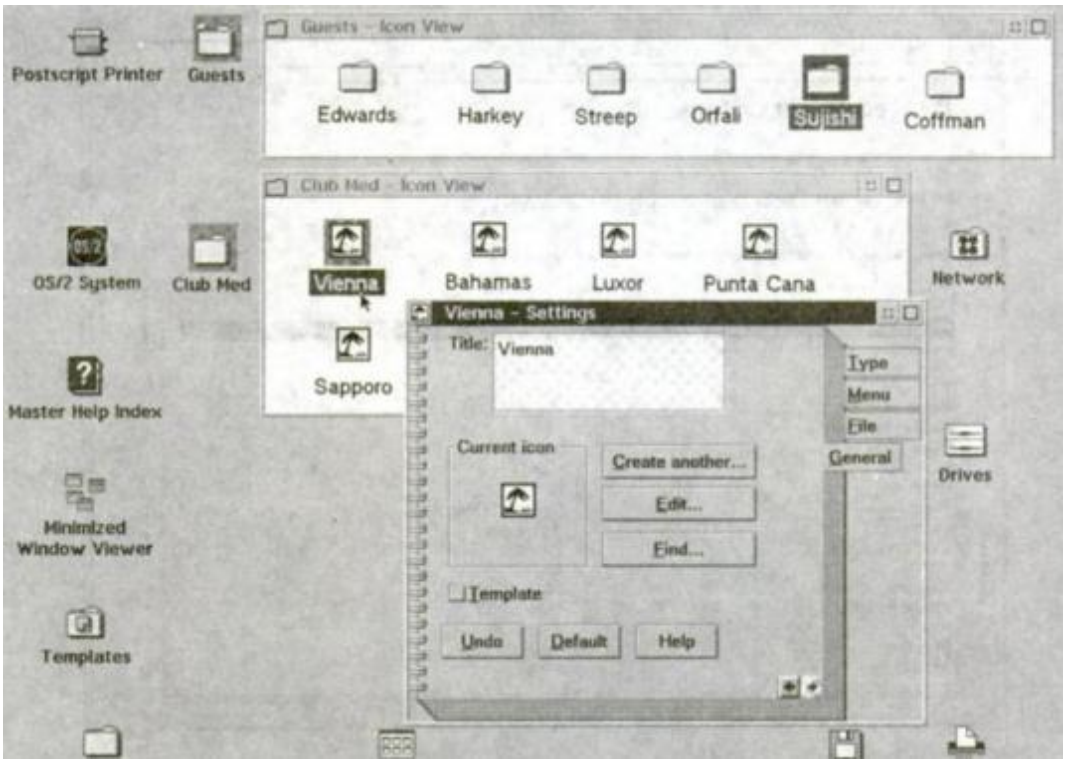


Fig 1.2 Interacting windows

The OOUI client will show real desktop where you can play with objects and programs to do work. In the fig 1.2, you see that you can play with parallel dialogs over parallel session with server. By using peculiar advance features, you can allow these dialogs to show images, videos along with other features. In this, the detail is shown to user where inside work will constantly moving.

Check your progress 7

1. What is OOUI?
   a. It is a type of user interface based on an object-oriented programming
   b. In OOUI, users can classify objects based on their behaviour
   c. Both A and B
   d. None of these

1.9 Let Us Sum Up

In this unit we have learnt that client/server applications carries many functions which are performed by combination of resources which can be done by

client workstation processor as well as server workstation processor. We see that the services will have ability to cut and paste input from different sources where the client is consumer of services given by one or more server processors and provides services related to presentation.

It is noted that Dynamic Data Exchange protocol is created in Microsoft Windows where users can share their data among applications which can be on local machine or on Network. It seems that OLE are object linking and embedding mechanism that permit users to put the data together from various applications. In this, object link permits the users to distribute individual source of data for required object.

A linked OLE object is simply a reference to information that is located in another document. Link objects when you want to use the same information in more than one document. The Java client sharing has utility programs which were written by Java client. GUI is a Graphical User Interface that works with icons or indicators along with electronic devices. Interface is a coordination involved among computer, program and humans.

In Java, non GUI client are simply applications which generates server requests without the intervention of humans. Such type of client normally either needs multitasking or can work without multitasking. Object-oriented user interface is a software interface in which users will work on certain objects for certain properties. It is developed on the idea of object-oriented programming applied in case of modern computing.

1.10 Answers for Check Your Progress

Check your progress 1

Answers: (1-d), (2-c)

Check your progress 2

Answers: (1-d)

Check your progress 3

Answers: (1-a), (2-c)

Check your progress 4

Answers: (1-d)

Check your progress 5

Answers: (1-d)

Check your progress 6

Answers: (1-c), (2-a)

Check your progress 7

Answers: (1-c)

1.11 Glossary

1. **Client -** It is a user, software application or computer which request for services, data or processing of application or computer.
2. **Connection request -** It is a notification which is sent by initiator and received by listener showing request of initiator to start a connection.
3. **Network Interface -** It is a network layer showing generic interface for clients, servers or external processes to access Net functions.

1.12 Assignment

Explain the Linked OLE objects with examples.

1.13 Activities

Study the various types of Java GUI and Non GUI clients.

1.14 Case Study

Study the types of Object-oriented user interface software's in details.

1.15 Further Readings

1. Object-Oriented Systems Ambler, S. 1997.
2. The client–server model, S smith, 2010.

UNIT 2: SERVER SIDE SERVICES

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- About server side
- About the various services involved in server side
- About communication services

2.1 Introduction

As is the case with any client-server paradigm, in the world of web services there are web service providers and web service consumers. Server-Side SOAP is a tutorial which deals with how to build and provide web services using Apache SOAP.

2.2 Server Functionality

There are many services/functions that are performed by the Server which can be:

- Application
- File
- Database
- Print
- Fax
- Image
- Communications
- Security
- Systems
- Network Management

It is important to understand that a server is an architectural concept, not a physical implementation description. The same physical device can provide client and server functions.

Application servers provide business functionality to support the operation of the client workstation. In the client/server model these services can be provided for an entire or partial business function invoked through an Inter Process Communication (IPC) request for service. Either message based requests i.e. OLTP or RPCs can be used.

A group of application servers may task in combined to assign a comprehensive business operation. For instruction, in case of a payroll system, we see that the employee data gets handled by single application server where it gets

calculated with the help of other application server along with deductions if any, gets calculated by another application server.

Such type of servers will able to work on various operating systems on different hardware platforms which make use of varied database servers. In such circumstances the clients application get called upon with services without consideration any technology or location of different servers positions.

Check your progress 1

1. What is the use of Print server?
   a. To transfer files securely
   b. To send and receive data to web browsers
   c. To get the printing of any work from client side
   d. None of these
2. What is the use of application server?
   a. It provides business functionality to support the operation of the client workstation.
   b. It is a middleware server which carry large amount of computing among database and end user servers.
   c. It is responsible for inter process communication
   d. All of these

2.3 Services

In Java we see that there are many services that are performed on server side. To perform certain services, the processor will only not take part in doing activities, but apart from it, certain device along with its hardware controller are used to do certain server services.

Client Side and Server Side Services

2.3.1 Request Processing

In order to put a request, the:

- Client will give requests to Network Operating System service software which is present on client machine.
- Services format will further request to required RPC and give request to application layer of client protocol stack.
- Application layer inside protocol stack located on server will receive certain request.

2.3.2 Print Services

To get the printing of any work from client side, we see that:

- the good quality printers, computer fax and plotters are good in supporting from shared server.
- Server get input from clients and arranges as per priority of request which will be handled in case of availability of device.
- Certain firms understand large savings by allowing users to have fax output from computers later on arranges for transmission by fax server in case of less communication costs.
- Faxes coming in gets arranged at server which will be distributed to required client based on receipt or request.
- Concert having workflow techniques where images obtained and distributed to required client workstation using image server.
- Client/server model handles arrangement at server by employing in concert using standard algorithms showing way of distributing queued work.
- Good standards for electronic document uses such mechanism to makes themselves fully integrated in desktop working environment.

2.3.3 Database Services

There are certain server side database that are done by file servers using various interface.

- Various softwares/database like dBASE, Clipper, FoxPro and Paradox gets worked out by database

- The engine located on client machine uses file services that are given by file server in order to record the working and free space management.
- Many current and different strong implementation occurs for actual flat-file models that gets out indexes for direct record working..
- Using application program having issues lock requests as well as lock checks will able to handle currency control also using database server, lock table is framed which interrogates when record access lock check obtained.
- With access at record level, several details ease primary key which returns to client workstation for the purpose of filtering.
- In absence of facilities for procedural code working at server and to join, or filter rows preceding to getting back to workstation. With such drawback, the chances of records locking when many clients works on single database tends to have excess network traffic where after rejection, the non useful rows gets returned to workstation.
- With insufficient logic of server execution, the products get saved from automatic partial update back-out with revival subsequent to an application, system or failure of hardware.
- Client/server database engines gives support at server to carry out SQL requests which is given from client workstation.
- So services lead to file gets used for space allotment in addition to crucial directory services where other are shown by database server.

2.3.4 Security Services

We see that client/server applications needs related security services which are given by parent environments. Here, users are compulsory to register in by means of user ID and password. While performing all, if:

- Passwords is not seen which recognises users, in such case, security server will maintain to facilitate passwords that needs to be updated regularly.
- Desk enterprise exits, then an individual logon id along with sequence is applied to access information along with processes for user that are required for correct access. As data is kept in less secured area, then in such case option should will allow to keep data in encrypted format.

- Workstation without floppy having embedded data encryption standard coprocessors are present from vendors, then products gets directly encrypted or decrypted data that gets read or write to certain disk. The encryption along with decryption of data exits with des algorithm using user password which make sure about execution of unauthorized user operations.
- Such security that is mainly for laptop computers having client/server applications as laptops doesn't work in surroundings with similar physical security.

2.3.5 File Services

File activities undergo access to the virtual directories furthermore files laid on the client workstation additionally to the server's fixed storage. These activities are catered conclusive the redirection software functioned as part of the client workstation operating circumstance.

The file activities contribute this support at the remote server processor. In the definite medium, software, allocated data, databases, furthermore backups are saved on disk, tape, as well as optical storage devices that are controlled by the file server.

To reduce the effort as well as sequence of installation along with extension of software, software should be loaded from the server for action on the client. Current versions can be updated on the server also brought about currently achievable to all users.

In accumulation, installation in a core position decreases the approach required for each workstation user to trigger the installation procedure. On account of each client workstation user exercises the equivalent installation of the software, alternative parameters are integrated, as well as remote assistance desk operators are careful of them.

Substitutes of the server can be scheduled as well as monitored close-at-hand a educated support person. Backups of client workstations can be approximated from the server; furthermore data can be saved at the server to ease recovery. Tape or optical backup objects are certainly facilitated for backup; these devices can eagerly contribute support for numerous users.

A support person who assures that the backup functions are accomplished readily monitors a core location. With additional organizations observing at

multimedia as well as image technology, big optical storage devices are best correctly implemented as shared servers.

2.3.6 Communication Services

We see that a Client/server applications needs LAN and WAN communication services. Also the basic LAN services are integral to Network operating system while many communication server products will give WAN services.

Check your progress 2

1. Which of the following statement is a type of server side Security service?
   a. Applications needs LAN and WAN communication services
   b. Users are required to register by using valid user ID and password.
   c. Good quality printers are needed to take print from client side
   d. None of these
2. Which is the main characteristics of File server
   a. Manages file operations and is shared on a network.
   b. Acts as a fat client and is shared on a network.
   c. Manages file operations and is limited to one PC.
   d. None of these

2.4 Let Us Sum Up

While studying this unit, we have learnt that Application servers provide business functionality to support the operation of the client workstation where these services shows entire or partial business function carried out by Inter Process Communication (IPC) request for service.

We see that there are many services in Java that are performed on server side where not only processor but certain other device along with hardware controller are applied to do certain server services.

File activities undergo access to the virtual directories furthermore files laid on the client workstation additionally to the server's fixed storage. These activities are catered conclusive the redirection software functioned as part of the client workstation operating circumstance.

2.5 Answers for Check Your Progress

Check your progress 1

Answers: (1-c), (2-d)

Check your progress 2

Answers: (1-b), (2-a)

2.6 Glossary

1. **Shared server -** A database server which is configured by allowing user processes to share so as to increase in users.
2. **Service handler -** A process that acts a connection point from the listener to the database server.
3. **Network object** - Any service that can be directly addressed on a network; for example, a listener.

2.7 Assignment

Write short note on communication services at server side.

2.8 Activities

Collect some information on File activities.

2.9 Case Study

Generalised the basic feature of Application server.

2.10 Further Readings

1. Object-Oriented Systems Ambler, S. 1997
2. The client–server model, S smith, 2010

Block Summary

In this block, you have learnt and understand about various Client side and server side applications along with their usage. The block gives an idea on the study and concept of Object Oriented User Interface Clients. You have been well explained with the concepts of various types of Server tools.

The block detailed about the basic of GUI and non GUI Clients techniques with illustrated examples. The concept related to Server Functionality along with its necessary services is well detailed to you. You will be demonstrated practically about various types of GUI Clients.

Block Assignment

Short Answer Questions

1. What is client/server model?
2. Explain the features of Client Services.
3. Write note on Object Linking and Embedding.
4. Write short note on Server Request.

Long Answer Questions

1. Write short notes on Client side services.
2. Write short note on Server side services.
3. Write note on Object Oriented User Interface Clients.

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

Education is something which ought to be brought within the reach of every one.

- Dr. B. R. Ambedkar

Dr. Babasaheb Ambedkar Open University
'Jyotirmay Parisar', Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,
Ahmedabad-382 481.

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BCA - 404

BLOCK 3:
CLIENT SERVER DEVELOPMENT

Dr. Babasaheb Ambedkar Open University
Ahmedabad

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Er. Nishit Mathur

Language Editor
Prof. Jaipal Gaikwad

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Ms. Pinaz Driver
Ms. Tejashree Bhosale
Mr. Kiran Shinde
Mr. Prashant Tikone
Mr. Akshay Mirajkar

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Contents

BLOCK 3: CLIENT SERVER DEVELOPMENT

UNIT 1 CLIENT SERVER SYSTEM DEVELOPMENT SOFTWARE

Factors driving demand for Applications Software Development, Client/Server system Development methodology, Project management, Architecture Definition, System Development environment, Productivity Measures, CASE, Client/Server System Development –Hardware, Hardware/Network acquisition, PC level processing units, Unix, Unix workstation server Hardware, Data storage, Network interface Cards, Power protection Device.

UNIT 2 CLIENT/SERVER SYSTEM DEVELOPMENT

Service and Supports, System Administration, Availability, Serviceability, Software Distribution, performance network management issues, Case studies.

BLOCK 4: INTRODUCTION TO C#

UNIT 1 INTRODUCTION TO .NET

The .NET Framework: an Overview, Framework Components, Framework Versions, Types of Applications which can be developed using MS.NET, MS.NET Base Class Library, MS.NET Namespaces, MSIL/ Metadata and PE files., The Common Language Runtime (CLR), Managed Code, MS.NET Memory Management / Garbage Collection, Common Type System (CTS), Common Language Specification (CLS), Types of JIT Compilers, Security Manager

UNIT 2 ASP.NET AND C#P

Working with Standard Controls, Navigation Controls, Validation Controls, Login Controls, Introduction to ASP.NET Objects, Building the StyleSheet, Creating the Content Master, Adding Elements, Building the Site Navigation, Adding Authentication, Adding Content Pages, Working with Data, Using ASP.NET Web Services and WCF, Creating a simple ASP.NET Web Service

Dr. Babasaheb
Ambedkar
Open University

BCA - 404

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BLOCK 3: CLIENT SERVER DEVELOPMENT

BLOCK 3: CLIENT SERVER DEVELOPMENT

Block Introduction

CASE tools along with execution methodologies that describe systems requirements repetitively with high as well as introductory user integration have been confirmed to essentially clarify analysis phase errors. Availability referred to a system uptime or potential of the system to be accessible for handing out information as well as undertaking its accepted work at whatever time it is called.

In this block, we will detail about the basic performance of network management issues with information on System Administration. The block will focus on the study and concept of various hardware in Client/Server System Development. You will give an idea on CASE tools.

In this block, you will made to learn and understand objectives of effective project management techniques. The concept related to data storage disk storages devices will also be explained to you. You will be demonstrated practically about PC level processing units.

Block Objective

After learning this block, you will be able to understand:

- Basic of System Development environment
- Concept of Productivity Measures
- Knowledge related to CASE tools
- Features related to Hardware/Network acquisition
- Idea about various service and supports in software
- Qualities of System Administration
- Features of Software Distribution

Block Structure

Unit 1: Client Server System Development Software

Unit 2: Client/Server System Development

UNIT 1: CLIENT SERVER SYSTEM DEVELOPMENT SOFTWARE

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- Basic of File systems
- Structure and file partition concepts
- Types of files
- Non-contiguous and contiguous storage allocation
- Idea about Disk scheduling

1.1 Introduction

The Index Group announces that the Computer-Aided Software Development as well as external technologies that rapidity software execution are reproached nearby 70 percent of the tope IT authorities examined as the foremost critical technologies to implement.

This current category of software tools assists organizations acknowledge additional speedily by slashing the time it acquires to commence current applications along with bringing about them simpler to alter or acknowledge. Old methods, sightlessly automating existing manual procedures, can hasten a company's death knell.

1.2 Factors driving demand for Applications Software Development

CASE tools along with execution methodologies that describe systems requirements repetitively with high as well as introductory user integration have been confirmed to essentially clarify analysis phase errors.

Expect for Platform Migration as well as Reengineering of continuing approaches:

Older along with present applications are existing correctly revaluated as well as in several cases ceased when they don't disburse off. A 16-percent lower in proprietary technology incentives had been benchmarked in 1993 furthermore this direction will extend as organizations flow to expose approaches along with workstation technology.

BPR approaches to diminish business approach cost along with complexity close-at-hand actuating decision-making obligation to those individuals who early deal with the consumer or difficulty. Organizations are utilizing the client/server to carry information to the workplace of energized employees.

Need for a Common Interface across Platforms

Graphical user interfaces (GUIs) that authorize a analogous appearance along with experience as well as front-end applications that accomplish contradictory applications are on the heighten.

Coincidentally superior importance to the single-system appearance consideration is that every consumer from every workstation acquire approach to every application for which they adhere a desire as well as right without honor to or wariness of the technology.

Increase in Applications Development by Users

As workstation energy advances in addition dollars-per-MIPS decline, additional energy continues flowing into the assistants of the end user. The Index Group descriptions that end users are here and now conducting additional than one-third of application development; IT departments are practicing additional comparable a benefit. This continues the after-effect of IT department group experience the shrink of conservation projects that restrain programmers from meeting critical backlog demand for new development.

Check your progress 1

1. BPR stands for _____.
   a. Business process reengineering
   b. Business product resource
   c. Basic product research
   d. None of these

1.3 Client/Server system Development methodology

The purpose of a methodology is to describe a disciplined process through which technology can be applied to achieve the business objectives.

Client
Server
Development

Methodology should describe the processes involved through the entire life cycle, from BPR and systems planning through and including maintenance of systems in production.

These companies offer methodologies tuned for the client/server computing environment. However, every methodology has its own strengths, which are important to understand as part of the systems integration vendor selection process.

The following table depicts the details of the major activities of each stage of the systems integration life cycle methodology.

SILC Phase	**Major Activities**
Systems Planning	Initiate systems planning
	Gather data
	Identify current situation
	Describe existing systems
	Define requirements
	Analyze applications and data architectures
	Analyze technology platforms
	Prepare implementation plan
Project Initiation	Screen request
	Identify relationship to long-range systems plan
	Initiate project
	Prepare plan for next phase
Architecture Definition	Gather data
	Expand the requirements to the next level of detail
	Conceptualize alternative solutions
	Develop proposed conceptual architecture
	Select specific products and vendors
Analysis	Gather data
	Develop a logical model of the new application system
	Define general information system requirements
	Prepare external system design
Design	Perform preliminary design
	Perform detailed design
	Design system test
	Design user aids
	Design conversion system
Development	Set up the development environment
	Code modules
	Develop user aids
	Conduct system test
Facilities Engineering	Gather data
	Conduct site survey
	Document facility requirements
	Design data center
	Plan site preparation

Check your progress 2

1. What is the purpose of development methodology with respect to client server technology?
   a. To describe a disciplined process through which technology can be applied
   b. It is used to describe the processes involved through the entire life cycle
   c. Both A and B
   d. None of these

1.4 Project management

Numerous constituents do well to a project's successfulness. One of the foremost necessary continues embedding a productive project administration along with reporting mechanism. Sound project control behaves not only advance the chance of getting planned project approaches furthermore additionally advance a functioning circumstance where the morale continues broad as well as the attentiveness endures chromatic. This is characteristically disparaging today when technology is wherefore liquid and the expectation for excluding the developer from the specific technology is so meaningful. The objectives of effective project management are as listed below:

1. Plan the project:
   - Define project scope
   - Define deliverables
   - Enforce methodology
   - Identify tasks and estimates
   - Establish project organization and staffing
   - Document assumptions
   - Identify client responsibilities
   - Define acceptance criteria
   - Define requirements for internal quality assurance review
   - Determine project schedules and milestones

- Document costs and payment terms

2. Manage and control project execution:

- Maintain personal commitment
- Establish regular status reporting
- Monitor project against approved milestones
- Follow established decision and change request procudures log and follow up on problems

3. Complete the project:

- Establish clear, unambiguous acceptance criteria
- Deliver a high-quality product consistent with approved criteria
- Obtain clear acceptance of the product

Current methodology which can be client/server posed heavy weighted on architecture describing domain. Due to less experience in creating client/server solutions will mix with fresh paradigm quality by user which makes possible for required prototyping of applications. Such features will result in rethinking of architecture which is cheap as well as correct as per latest technology.

The tools for prototyping in the client/server platform are powerful enough that prototyping is frequently faster in determining user requirements than traditional modelling techniques were.

Check your progress 3

1. What are the main objectives of project management?

a. Plan the project

b. Manage and control project execution

c. Complete the project in time

d. All of these

1.5 Architecture Definition

The idea about architecture while designing shows that application architecture will have varied technology platform for running an application. In

order to choose an application architecture, you should evaluate initially the requirement and necessity of priorities. Certain priorities must consider and weight for following criteria:

Cost of operation:

Initially the company should have all the answers related to cost involvement. The cost is required for carrying all functions related to:

- Trained STAFF
- Computer literate
- Cost to user

There are many questions in terms of cost involved in company which lead to:

- Occasional users
- Scary user
- Users with little patience
- Public situation regarding training to users
- Effect on user creating mistakes

Response time:

It is another important aspect which describes:

- real speed requirement
- utilizing full time
- impact due to non timely response
- effect of response lag

Availability:

In this, the company used to sense for actual needs which will be calculated based on number of days with number of working hours.

Security:

It is related to actual security requirement.

Flexibility to change:

This involves the flexible time needed to change an application change which depends on various factors:

- marketing priorities
- legislative changes
- technology changes driving the system

Use of existing technology:

It concerns with present investment and is dependent on growth capabilities which involves maintenance and support issues.

System interface:

It involves types of system needed for particular application which can be internal or external and can be altered.

These application architecture issues must be carefully evaluated and weighed from a business perspective. Only after completing this process can managers legitimately review the technical architecture options. They must be able to justify the technology selection in the way it supports the business priorities

Once managers understand the application architecture issues, it becomes appropriate to evaluate the technical architecture options. Notice that staff are not yet selecting product, only architectural features. It is important to avoid selecting the product before purchasers understand the baseline requirements.

Check your progress 4

1. For which of the following, the cost of operation is needed to be calculated?
   a. Maintaining Trained STAFF
   b. Providing Computer literacy
   c. Cost to user
   d. All of these

1.6 System Development environments

Just one time after defining an organization, an application along with technical structure along with desired tools involves certain step that explains how to apply certain tools. It seems that developers will not become good system

builders as they carry effective set of tools which are good as their development environment explains about tools usage.

It is acknowledge that SDE carries hardware, software, interfaces, standards, procedures along with training which gets accepted and are applied using an enterprise in order to optimize its information systems support which is framed for intended planning, management along with operations.

- Architecture explanation designed to select fixed technology platform.
- Interfaces that isolate the user and developer from the specifics of the technical platform should be used to support the creation of a single system image.
- Standards procedures be explained and developed to give applications with good appearance.
- Continuous components gains productivity that handle single-system image.
- Training programs will assure users with developers to know more on environment.

The SDE will include several phases of systems development life cycle that gets embedded along with desktop which gives strong extra tools for workstation users which a lot in various features of having an information.

An importance from SDE involves conscious effort made to create reusable components. With the doubt neighbouring product selection for client/server applications today, the reimbursement by means of SDE will cut off the developers from technology that are important. Certain components gets included in SDE which are setup by an organization:

Built-in navigation:

All process applies similar process which needs to move among processes. In this, process gets identified to next process with available processes. A business analyst furthermore, not developer go with navigation explanation. Every user and every developer then views navigation in the same way.

Standardized screen design:

It is noted that every standards are in position for every type of functions furthermore certain screens gets formed by default business process. Users along with developers gets familiar with several screens applied for help, add, change, delete and view furthermore table management functions.

Integrated help:

All the same, context-sensitive help out service that act in response to the correct problem present in the business process. Here none of the programmer development will be entertained. It is noted that end-user along with analyst that understands the application view of system user given to help text which is maintained by user after system is in production.

Integrated table maintenance:

Tables are designing concept of program which calls for standard reference data which can be program error codes, printer control codes and so on that gets stored in single set of files or databases. It is seen that individual table maintenance function is given for several applications in an organization. Here programmers along with users just call upon its services make applications to share with standard tables.

Comprehensive security:

It is noted that individual security profile is handled for every recognised user where navigation is connected to security where users have options which can be applied for further use. Here all programmer with user contains similar security facilities that are handled by authorized user which applies for maintaining table facilities.

Automatic view maintenance:

It is noted that screens are created, navigated is required and framing of programs that are created on basis of security profile along with business needs for particular process. The developer does not have to write special code to extract data from the database. All access is generated based on the defined business processes and security.

Standard skeleton programs:

It seems that analyst will answer particular questions which is obtained as outline program for all business process having feature like standard functions which is needed by programmer.

Check your progress 5

1. What is meant by Standard skeleton programs?
   a. A biological program
   b. Standard functions needed by programmer
   c. Both B and C
   d. None of these

1.7 Productivity Measures

It is examined that there are many researchers that carry required experiential investigations in order to conclude that many software development quality, tool, method or certain grouping carries important collision on software production. Certain studies first and foremost will stress on expansion of LSS where main software efficiency dimension studies gets re-examined together with global sites. In adding together, numeral other hypothetical with experimental studies of programmer efficiency, cost-benefit investigation, software cost judgment, along with software production enhancement program needs to be consider again. Jointly, such studies make available as loosely-grounded foundation that identifies many project which describes such influence software productivity.

Currently, many reasons are there to measure software productivity that was reported. The idea behind this is to find how to lower the software development costs, improved software quality along with improving rate at which software is created which includes:

- Increasing volume of work from present staff
- Do similar amount of work with smaller staff
- Develop products of more complexity value with similar staff workload
- Avoiding getting extra staff to increase work
- Rationalize higher levels of capital-to-staff investment
- Lower error volume in deliver products and lowering amount of time with effort required to alter software errors
- Straight or downsize software production operations

- Finding required product defects before the development
- Finding resource utilization patterns to find production bottlenecks and underutilized resources
- Find high-output personnel to receive rewards
- Find low-output personnel for extra training or reassignment.

It is noted that there are different reasons for measuring software productivity. It is not desirable to try to achieve for the most part or all of these objectives through a single efficiency capacity program. For example, unlike people concerned in a large software project may charge definite of these option more than others. in the same way, each option involve definite kinds of data be together.

Programmer or manager self-reported data are the slightest expensive to bring simultaneously, even though they may be of imperfect correctness. On the other hand, if efficiency measures are to be applied for personnel assessment, then one should not look forward to far above ground reliability or strength in self-reported data. In same way, if productivity measures are employed as basis of allocating resources or rewards, then the data reporters will have an incentive to improve their reported production values.

Alternative, by prompting software production, degree of potential provided for useful information to project managers with developers to facilitate improving their knowledge along with experience over time. External observers bring together such information with less cost than self report. Similarly, computerized production performance monitors used, but this is still an emerging area of technology requiring more insight for what should be measured and how.

Check your progress 6

1. How can we measure software productivity?
   a. Increasing volume of work from present staff
   b. Lower error volume in deliver products
   c. Straight or downsize software production operations
   d. All of these

1.8 CASE

CASE tools are created on enterprise model of procedure which are computerized where systems integration takes with software development. Such fundamental endeavour model applied by CASE is critical to tool's convenience. Tools foundation based on deprived model will experience on or after poor integration which are not capable to handle precise category of information which involve photocopy data entry that will not has numerous analyst-developer teams along with to non elastic sufficient which handle growing new techniques for identify and building systems solutions. Tools having insufficient models boundary with their users capable of development abilities.

In all most important CASE products function applied in client/server surroundings uses Intel 486-based workstations that will function at 60MHz or quicker with 16-24 Mbytes of memory with 260Mbyte hard disks along with UNIX workstations of comparable size that are marginally essential. Therefore, combining hardware using CASE software costs raised CASE costs up to $20,000 per user workstation/terminal.

Unluckily, a methodical reassess of accessible CASE products shows that nothing passably make available clear support for development of client/server applications along with GUIs. Such type of short support takes place in spite of the information which function as network-based applications having development of host-based applications. There is substantial impetus to build up products with intention of holding up client/server model. The Bachman tools are in front position in this area for reason that their focus on sustain for business procedure reengineering. With lots of client/server request being ported from a minicomputer or mainframe, the abilities to use again the live models and to turn round engineer the databases are tremendously powerful and timesaving features.

It appears likely that no sole vendor will build up best-integrated tool for entire system's life cycle. As an alternative, in probable state of affairs, developers mix the best products from quite lot of vendors. IBM envisions this situation in their AD/Cycle creation line by Computer Associates in CA90 products and by NCR in their Open Cooperative Computing series of products.

A supplementary innovative development is happening as CASE tools like the Bachman products are individual integrated with development tools from other vendors. These development tools, used with an SDE, allow applications to be prototyped and then reengineered back into the CASE tool to create process and data models. With the power of GUI-based development environments to create

and demonstrate application look and feel, the prototyping approach to rapid application design (RAD) is the only cost-effective way to build client/server applications today.

Users well-known with effortlessness of application development on the workstation will not recognize paper or optical models of their application. They can only completely dream of solution model when they can touch and feel it. This is the advantage of prototyping, which provides a real touch and feel. Apart from this, earliest stages of solution conceptualization, tools for prototyping ought to be created by means of similar products that are applied for production development.

Check your progress 7

1. What is the full form of CASE tools?
   a. Computer acquired software enable tools
   b. Computer aided software engineering tools
   c. Computer abled software enable tools
   d. None of these

1.9 Client/Server System Development –Hardware

Entry-level client workstations can extent from an elementary Intel-based PC to an entry-level Apple Macintosh or an X-Terminal. These entry-level clients commence at about $1,000 furthermore exercise LAN servers for imprinting, backup, software accumulation, application compulsion, as well as WAN connectivity. High-end client workstations can amount additional than $50,000 for engineering stations that assign enriched abilities such as a gigabyte or additional of local storage, high-resolution graphics monitors, 100-MIPS processing, direct WAN connectivity, 1000-dpi color printing, or authoritative multimedia execution tools. The conventional client workstation has compressed from $5000 to $2000 in the concluding two years. This acquires an appearance with the processing power indistinguishable to an 8Mbyte Intel 33-MHz 486DX PC with immediate storage of 250Mbytes, LAN connectivity, along with a VGA-equivalent monitor. This amount category is not apprehended to deteriorate much in addition, on account of GUI software as well as reengineered application constraints

commission steadily elevate the processing power demands for entry-level machines.

Server hardware approaches the largest as well as foremost complicated set of options. Servers plunge the arrangement from a $30M+ conventional IBM mainframe, to a 4- to 16-way symmetric portion multiprocessor machine, to a 32- to 32767-processor heavily non-convergent cluster acknowledging hundreds of users, to a $5,000 PC exercised to assign file along with connectivity services for a compact LAN workgroup. Numerous organizations additionally have client/server applications that exercise the benefits of prevailing IBM 370 mainframes plunging VM, MVS, or VSE, DEC VAX minicomputers running VMS or Ultrix, along with awesome RISC-based systems actuating UNIX—all as high-end servers.

External mainframe as well as minicomputer hardware platforms, proceeding proprietary operating systems, are repeatedly exercised in terminal counterfeit condition from the client workstation. The non-IBM as well as DEC proprietary operating system platforms occasionally are exercised to assign outside services, like as database as well as RPC-invoked application services. There continues a miss of tools feasible in these environments to develop or accumulate client/server applications. Servers based on the IBM, DEC, along with UNIX operating systems consideration assign application services employing continuing applications through terminal emulation or RPC-invoked application services. These equivalent servers will assign connectivity along with database services to the original client/server applications in an organization.

Connectivity requires every client workstation to be connected to a LAN or through a WAN to a remote server. In the usual situation, the workstation is connected through an Ethernet, Token Ring, FDDI, CDDI, or occasionally a parallel or serial interface to the LAN. The primary connection types require a network interface card (NIC) to be inserted in the workstation to provide the protocol processing necessary to establish and maintain the connection. The cost of LAN connectivity has declined rapidly in parallel with the industry reduction in workstation costs.

Cabling costs vary widely, depending on the physical difficulty of installation and whether the network planners choose unshielded twisted-pair (UTP), shielded twisted-pair (STP), or glass-fiber cables. Cable costs without installation run from $1 per foot for UTP, $1.50 per foot for STP, to $3 per foot for glass fiber. Installation costs vary from $1 per foot to $15 per foot, depending on the physical environment and connection requirements. Glass-fiber termination

equipment is more costly than twisted-pair, although the costs are declining. Current costs are between $100-200 for Ethernet, $300-500 for Token Ring, $300-700 for CDDI, and $750-1250 for FDDI.

Presently, numerous vendors deliver the hardware for these connections. Each vendor attempts several benefits in terms of cost, action, as well as loyalty. Motorola ascribes wireless Ethernet connectivity at deepen speeds along with higher charges than wired connections. Wireless connections are a benefit in subsisting buildings with no cable connected furthermore with relatively low-speed communications constraints.

WAN connectivity needs each workstation to be articulately affixed to the WAN or to a communications server amalgamated to the WAN. Maximum contemporary LANs are established implementing communications servers. There are costs, operation, as well as particularly network management inducts for exercising a LAN communications server. A considerable benefit accumulates since there is no expect to cable each workstation to the WAN. Workstations that are autonomously attached to the WAN mandate an implanted controller card for chronological communications including both a modem and serial connection for asynchronous communications. These definitely conduct at speeds of 2400-64000 bits per second (bps) skewered analog or digital modems. Each workstation necessary has its own cable joining it to the WAN controller. Workstations affixed to the WAN accomplished a communications server ascribe a higher-speed connection, certainly 14400 bps, 56000 bps, or 1.54 Mbps.

Check your progress 8

1. We can connect client / server workstations by using _____.
   a. Ethernet
   b. Token ring
   c. FDDI
   d. All of these

1.10 Hardware/Network acquisition

Earlier, selecting client hardware for end users makes organizations to define with standard for classes of users. Such set of standards simplify collection

of suitable client hardware needed for user which allow buyers to organize purchasing agreements to increase quantity of price discounts.

Conveniently, there are numeral issues to think about while selecting client workstation as well as processor type, coprocessor ability, internal bus structure, size of base unit and so on. However of issues, one of main overlooked with respect to client/server applications is use of GUI. GUI applications needs VGA or better screen drivers where screens should be larger than 15-inch standard that should be compulsory for users who usually have many active windows at single time. Along with this, windows active on-screen, larger the monitor viewing area, more will be the image usage, graphics or full-motion video resolution. It is important to keep in mind that efficiency is severely exaggerated by incapability simply to read screen. Unsuitable resolution will lead to exhaustion and incompetence.

The endeavour on desk needs to have required bandwidth that is accessible to supply receptiveness to desktop user. If usual admittance to off LAN data is necessary, router based internetworking is compulsory. If infrequent off LAN contact is necessary, bridges can be applied. Routers provide additional advantage of supporting multiprotocol internetworking which is essential as organizations put 10BaseT Ethernet in active Token Ring environment. Fast Ethernet and FDDI are becoming more common as multimedia applications are set free.

Check your progress 9

1. While selecting client workstations we should take care of _____.
   a. Processor Type
   b. Coprocessor Ability
   c. Internal Bus Structure
   d. All Of These

1.11 PC level processing units

Client/server applications digress consequentially in their client processing demands furthermore their I/O demands on the client processor along with server. In common, clients that uphold protected-mode appealing should be acquired. This denotes the function of 32-bit processors—perhaps with a 16-bit I/O bus if

the I/O condition is low. Low describes the client isn't compelled to send as well as acquire awesome amounts of data, similar as images, which could be 100K bytes or enlarged, on a fixed condition.

As multi windowed as well as multimedia applications suit habitual in the course of 1994, numerous applications will mandate the bandwidth sole deployed by a 32-bit I/O bus facilitating VESA VL-bus or Intel PCI technology. Windowed applications mandate critical processing power to assign allowable response levels. The opening of application integration via DCE, OLE, as well as DOE indicatively advances the mechanism demands at the desktop. The approved minimal configuration for desktop processors acquires the processing amplitude of a 33Mhz Intel 486SX. Proximate untimely 1995, the minimal requirement will be the processing proportions of a 50Mhz Intel 486DX or a 33Mhz Intel Pentium.

Macintosh

The Mac System 7 operating system is visually intuitive and provides the best productivity when response time to GUI operations is secondary. The Motorola 68040, 8Mbytes RAM, 120Mbyte disk is recommended. By early 1995, the availability of PowerPC technology and the integration of System 7 with AIX and Windows means that users will need considerably more processor capacity. Fortunately, the PowerPC will provide this for the same or lower cost than the existing Motorola technology.

Notebooks

Consumers operating remotely on a conventional condition may assess that a notebook computer better satiates their demands. The notebook computer continues the fastest developing business present. The contemporary technology in this arena endures feasible for Intel PC, Apple Macintosh, as well as SPARC UNIX processors. On account of notebooks are "miniaturized," their disk drives are frequent not contrasting to full-size desktop units. Therefore, the relatively slower speed of disk I/O on notebooks brings about it preferable to establish addition RAM, developing "virtual" disk drives.

A negligible configuration endures a processor with the identical processing power of a 33Mhz Intel 486SX, 8mbytes of RAM as well as 140Mbytes of disk. In accumulation, the notebook with battery should approximate lower than seven pounds additionally embrace a battery life of three hours. Color uphold continues a selection throughout 1994 furthermore will be compulsory for complete near 1995. In expansion, if the application will plunge a segregated GUI, it continues captivating to establish software to compact the GUI as well as V.32 modem

communications at 9600 bps or V.32bis at 14400 bps, applying V.42 as well as V.42bis compression, respectively. The productive throughput continues two to three times the baud rate wherefore of compression. The application of MNP4 as well as V.42 or MNP5 also V.42bis defect correction authorizes these speeds to function consequentially even throughout noisy line conditions. The introduction of PCMCIA technology, credit card size modems, and flash memory are available to upgrade the notebook.

Pen

Pen-based consumers ascribe the competence to exercise applications employing a pen to forefront further choose or correspond without expectation for a mouse or keyboard. Commonly, they are conducted for approval, selection, as well as examination applications where selection checks are attainable. Developers employing this technology facilitate object-oriented software approaches that are RAM-intensive.

The forward of personal digital assistant (PDA) technology in 1993 embraces released the market to compact size computing. For the time being, in 1994, this technology will developed with additional storage understanding through cheaper, denser RAM as well as flash memory technology. The screen diligence will correct, as well as applications will be created that are not contingent upon cursive writing recognition.

The PDA business continues price-sensitive to a $500-$1000 device with the aptitude to gallop a Windows-like operating arrangement in 4MB of RAM, a 20Mhz Intel 486SX processor, also 8MB of flash memory. Appliances with this competence will conduct in 1994, as well as meaningful approaches distant personal diaries will be in employment. For the time being in 1995, 16MB of RAM as well as 32MB of flash memory will commence to arise, agreeing these devices to measure a complete market far 1996. In composition with wireless technology approaches, this will benefit the private information source for electronic news, magazines, books, besides so on. Your electronic Personal Wall Street advertisement will approach you for opening on your PDA.

Check your progress 10

1. Client server applications can be run on _____.
   a. Windows OS
   b. MAC OS
   c. Both of these
   d. None of these
2. PDA stands for _____,
   a. Personal Digital Assistance
   b. Problem Digital Assistance
   c. Personalize Directory Assistance
   d. None of these

1.12 Unix workstation server Hardware

UNIX client workstations normally are applied in such case when client processing requirements are more. It is found in certain applications that need of UNIX, X-terminals which is joined across UNIX presentation server will be preferred by the clients. It is normally seen that:

- UNIX client workstation carries several working power as compared to normal computing device client.
- With the start of software from SunSoft, Insignia Solutions as well as Locus Computing, it seems that these companies support execution work of DOS along with Windows 3.x applications which is available in UNIX window that makes UNIX desktop present to user needs software from both environments.

Check your progress 11

1. UNIX workstations are applied where_____.
   a. There are more processing requirements from client
   b. More working power as compared to normal computing device client is required
   c. Both of these
   d. None of these

1.13 Data storage

It is seen that fixed storage needs are specific to particular application along with quantity of disk storage with certain issues that exists in terms of performance as well as reliability. There are certain disk storages devices such as:

Magnetic Disk

- Disk storage uses SCSI-2 standard controller interface.
- Shows best performance in standards environment.
- Deliver by vendors having good capacity, performance as well as reliable disk devices for controller.
- Using high-capacity cache storage will rapidly increases in performance.
- Latest SCSI-2 controllers with configurable having 266K cache.
- It is an important component of architecture.
- It carries latest drives standard 3.6 sizes with 1.0-1.6Gbyte capacity.

We see that working of compression software will double the capacity. If the size of GUI software increases then use of multimedia applications will comes in demand for high disk capacity during 1994 and beyond.

Mirrored Disk

When applications stipulate extensive trustworthiness, it may be befitting to conduct features that uphold duplicated disks. With this features, details exists automatically recorded to two disks. This authorizes the application to sustain even if a failure develops on one disk.

System files along with illustrations files should be approximated for mirroring. Even furthermore system files are frequently read-only, the amount of users infected by unavailability of the files may clarify this excessiveness. In acquisition, activity can advance on account of dual reads can be undergone in parallel.

RAID-Disk Array

Conventional magnetic disk knowledge is over and over again referred to as sole great pricey disk. Extremely high presentation as well as high accessibility can be accomplished through a laid off array of reasonably priced drives. These facilitate data files to be extended across various physical drives where data can be reflected as part of the configuration.

RAID expertise will make available a significant performance improvement for the reason that many corresponding I/O operations can be practiced at same time. Elevated capacity caches applied in combination with RAID technology to accomplish most favourable recital. The size will be identified as part of the architecture definition.

Tape

Although most permanently stored data uses disk, tape is a very popular form of low-cost magnetic storage and is used primarily for backup purposes.

The standard backup tape device today is digital audiotape (DAT). These tapes provide approximately 1.2 Gigabytes of storage on a standard cartridge-size cassette tape. Tape is a sequential medium and does not adequately support direct (random) access to information. If an organization standardizes on a single tape format and technology, distribution of information by mailing tapes can be a cost-effective communications mechanism for large quantities of information that do not require real-time transmission or accessibility.

Optical Disks

Optical disk storage technology provides the advantage of high-volume, economical storage with somewhat slower access times than traditional magnetic disk storage.

CD-ROM

Compact disk-read only memory (CD-ROM) optical drives are used for storage of information that is distributed for read-only use. A single CDROM can hold up to 800MB of information.

WORM

Write once, read many (WORM) optical drives are used to store information that is to be written to disk just once but read many times. This type of storage is frequently used to archive data that should not be modified. Traffic tickets issued by police departments are scanned and stored on WORM drives for reference on payment or non-payment. The WORM technology guarantees that the image cannot be tampered with. A magnetic drive can be used to store an index into the data on the WORM drive.

Erasable Optical

Erasable optical drives are used as an alternative to standard magnetic disk drives when speed of access is not important and the volume of data stored is large.

Network interface Cards

Client and server processors are attached to the LAN through NICs. These provide the physical connectivity to the wire and the protocol support to send/receive messages. The most popular network protocols today are Token Ring, Ethernet, and FDDI.

Power protection Device

A lot has been written in books, magazines, and journals about computer hardware and software; and a number of computer specialty businesses are dedicated to helping you work through issues of specific concern to your business objectives.

Check your progress 12

1. What is NIC?
   a. They are magnetic disk drives
   b. NIC helps to attach client and server process through LAN
   c. It is optical drive
   d. All of these

1.14 Let Us Sum Up

In this unit we have learnt that Index Group for Computer-Aided Software Development using external technologies shows software execution with related top IT authorities. In this we see that CASE tools using execution methodologies describe systems requirements with high introductory user integration having confirmed to essentially clarify analysis phase errors.

The purpose of a methodology is to describe a disciplined process through which technology can be applied to achieve the business objectives. It is seen that many constituents do good to project's successfulness as it is foremost necessary that embed productive project administration with reporting mechanism.

The idea about architecture while designing shows that application architecture will have varied technology platform for running an application. In order to choose an application architecture, you should evaluate initially the requirement and necessity of priorities. Just one time after defining an organization, an application along with technical structure along with desired tools involves certain step that explains how to apply certain tools.

1.15 Answers for Check Your Progress

Check your progress 1

Answers: (1 -a)

Check your progress 2

Answers: (1-c)

Check your progress 3

Answers: (1-d)

Check your progress 4

Answers: (1-d)

Check your progress 5

Answers: (1-b)

Check your progress 6

Answers: (1-d)

Check your progress 7

Answers: (1-b)

Check your progress 8

Answers: (1-d)

Check your progress 9

Answers: (1-d)

Check your progress 10

Answers: (1-c), (2-a)

Check your progress 11

Answers: (1-c)

Check your progress 12

Answers: (1-b)

1.16 Glossary

1. **CASE tools -** Tools that shows systems needs continuously using high integration clarifying analysis phase errors.
2. **Client/server applications -** Client processing demands on client processor with server that uphold protected mode.

1.17 Assignment

Explain the objectives of effective project management.

1.18 Activities

Study about System Development environment.

1.19 Case Study

Study about Hardware/Network acquisition.

1.20 Further Reading

1. Boehm, B.W., "Improving Software Productivity", Computer, 20(8), 43-58, 1987.
2. Scacchi, W., “Understanding Software Productivity”, Advances in Software Engineering and Knowledge Engineering, 1995.
3. Boehm, B. and R.W. Wolverton, "Software Cost Modelling: Some Lessons Learned", J. Systems and Software 1(1980).
4. Bailey, J. and V. Basili, "A Meta-Model for Software Development Resource Expenditures", Proc. 5th. Intern. Conf. Soft. Computer Society, (1981).

UNIT 2: CLIENT/SERVER SYSTEM DEVELOPMENTS

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- Concept of I/O devices
- Understand about Bus Architecture
- Detailed regarding features of DMA controlled I/O
- Basic of Input Output Programme
- Idea of DMA Channels

2.1 Introduction

Conversions are approaching for the desktop clients. At several allowed time, top organizations acquire a combination of client types. The "heavy-set client" acquires better of its program logic as well as data accumulated on its own hard drive. The "narrow client" acquires foremost of its operating software from the server. Gartner type acquires coined the identity "ultralite" for Web clients without procedure logic or data. Ultralite clients don't acquire to be impotent. They archive be considerable, graphical facilities that conduct from a remote server.

The ultralite client continues imploring on account of dependently its comfort of administration. Applications as well as data are efficiently acknowledged in a core location. Furthermore, this methodology exhausts awesome allotments of bandwidth. LAN-attached users will embrace to flow up to faster topologies along with commuting hardware. Remote offices with ultralite clients will no longer be able to operate across 56Kb lines.

The mobility of copies, graphical data, as well as program code continues already delays on T1 lines. Acceptable activity mandates consistent faster transmission mediums, along with their affixing higher costs.

At the competitor end, the heavy-set client archetypal demonstrates different problems. Data replication and synchronization between thousands of workstations is a fundamental issue. This problem is further aggravated by replicating to dial-in mobile users.

The thin client provides middle ground for many organizations now. Applications run on the client, and the data resides on a remote server. Both the thin and fat clients require individual software licenses. Neither can take advantage of software metering from an application server.

2.2 Service and Supports

Personal computer users in the past have dissimilar prospect. In earlier period, condition subsequent to hour operational on worksheet makes the system to get suspended where power fails and further virus will reboots the machine, when users without doubt experience exasperated but not actuality astonished.

Similarly, by means of companies so as to have enthused further than single-user PC request which furthermore squeeze networking where users in the

past have been more broad-minded of lesser amount of exact principles. For instance, the projects which costs to handle disseminated set of connections of PCs along with servers which will be 10 to 30% additional as compared to handling minicomputers along with mainframes. Other studies have claimed costs are double. This advanced cost is the cases as soon as LANs develop along with applications are put together with no an architectural view along with appropriate standards to hold up the design.

By means of moving client/server computing, the need for mainframe-like performance from client/server architectures tends to raised. Condition firms are going away to shift the business of the corporation into the client/server world, mainframe expectations will succeed in addition to mainframe support which should be rendered.

Modern understanding with remotely handled LAN applications shows that costs are similar to or fewer than costs for conventional mainframe applications. Successful remote management involves systems along with application architectures that look forward to necessity for secluded management

Check your progress 1

1. What is network service?
   a. The client can generate print request using OS
   b. The client can generate database service request
   c. The client can communicate with network using TCP/IP, IPX,Token ring etc. protocols
   d. None of these

2.3 System Administration

Similar to many things in life, the standard of doing right first time signifies long-term accomplishment of client/server application. Therefore, it is significant to make sure that client/server hardware is particular along with assembled as per required organizational standards which can be checked before the starting. As seen, initially the software gets loaded by experience staff which gets tested in order to ensure that it gets installed as per standards along with expectation of work. It is noted that many user problems occur due to incorrect installation along

with equipment which appears out of order during installation. Many LAN administration struggle gets saved by correct structure which is handled by qualified installers.

Check your progress 2

1. Which of the following is the part of system administration?
   a. client/server hardware is assembled as per required organizational standards
   b. Software gets installed as per standards along with expectation of work.
   c. Software gets tested as per standards by experience staff
   d. All of these

2.4 Availability

Availability referred to as system uptime or potential of the system to be accessible for handing out information as well as undertaking its accepted work at whatever time it is called. Minicomputer in addition to mainframe data centres ought to make available at slightest 99% availability as per latest technology. Such level of availability can be obtained by mixture of technical as well as practical steps goes behind. It is noted that many availability failure at present is rooted by human error. In order to lower this, data centres should put into practice inflexible measures just to administer change.

It is claimed that if the change is hardware, network, system, application software, stringent procedures to request, validate, test along with execution of change that are explained and stick on to. Pull out procedures explains along with test so as to make sure that in case of lapse occurs subsequent to functioning of alteration, the data centre be able to fall back to its preceding position.

Technical description which can be break up electrical power resource, endorsement diesel generator as well as battery power resource, laid off processors along with magnetic disk devices are applied to make sure that breakdown of a single constituent will not take downwards data centre. Very significant systems use fault-tolerant processors on or after vendors which can be Tandem and Stratus that will assure for availability which lead to 100%.

Data centres make use of extremely accomplished professionals in the central position. They are likely to be capable to pull through the location speedily after several failures. Vendor service contracts are used to guarantee so as to repair in one, four or eight hours as per the needs.

Client/server applications ought to be competent to offer the suitable level of accessibility commanded by business necessitate. Definite features which can be laid off power supplies along with battery backup are somewhat straightforward to make available. Big places, vendor service-level agreements be obtained so as to make sure about failures which can be repaired speedily, while in small places, repair by replacement is compulsory in case when required service levels will not ascertain with travel time.

The condition of extremely competent practical staff at each one site is from time to time substantial which can be infrequently cost-effectively possible. Inaccessible LAN administration is only means to construct successful exploit of inadequate possessions. Inaccessible administration involves an innermost site which is joined across WAN services with every LAN. Network administration examination levels are definite all the way through reasonability levels which allow qualified examination of accessibility of individual devices, of performance, and even of server magnetic disk space use.

Different products gets integrated by industry standard network management protocols which shows required level of accessibility for logical cost. The OSF has distinct criterion Distributed Management Environment for administration of its Distributed Computing Environments criterion, which is developing as the explanation for object technology based management stage. Even though such type of technology is not as much grown-up as compared to DCE standard which practice systems from integrators to show efficient remote systems management network operations centres.

Check your progress 3

1. What is true about availability in context of client server technology?
   a. Availability referred to as system uptime or potential of the system to be accessible for handing out information
   b. Client/server applications ought to be competent to offer the suitable level of accessibility commanded by business necessitate
   c. Both B and C
   d. None of these

2.5 Serviceability

For the most part, minicomputer along with mainframe operating systems having hardware will show diagnostic services which find the position of failures. Temporary mistake are well-known as a result to facilitate anticipatory maintenance which will set problems ahead of influence accessibility. The innermost position of the equipment permits qualified technicians to introduce standard anticipatory preservation programs. Because of such cause, numerous organizations put in initial servers in glass room in anticipation of having extra experience by means of distant LAN management.

It is found that products that depend on model procedure which can be Simple Network Management Protocol which describes required criticism of occasion awareness which handles remote systems management role. It is essential that the structural design will consider issues concerning with standard as well as products that require service.

Check your progress 4

1. Which of the following statement is true about Serviceability?
   a. Hardware shows diagnostic services which find the position of failures.
   b. Temporary mistake are well-known
   c. Ability to make problem set and finding its solutions
   d. All of these

2.6 Software Distribution

The regional minicomputer in addition to mainframe surroundings contribute to executable software commencing on its own library. Software preservation along with improvement are capable by altering in individual location. In dispersed client/server representation, working software stays at place placed on servers that are placed all through an organization. Alteration to system with application software needs replication all through an organization which shows marvellous complication in serviceability of applications.

Extra complication is lays open in UNIX atmosphere as soon as number of dissimilar hardware platforms are applied. In spite of fact that foundation level of the software is well-matched from corner to corner with different platforms where working binary form of software is not well-matched.

It is noted that working libraries gets framed on machine having similar physical hardware which results in serious problems related to transmission of software all through the network of dissimilar computer stage. It is noted that testing can be carried out on every stage earlier than changes gets shared. Numerous organizations addresses certain needs simply by installing any hardware platforms from field in core support location.

It is noted that answer to certain problem is well framed client/server structure which is handled by successful software administration tools. Such type of difficulty is definitely solvable except simply all the way through the plan along with planning. It will not be resolved in an unplanned manner following implementation.

There are particular needs that handles distributed technology. It is noted that benefit of personal computer is that it can be easily altered which is drawback for manufacturing environments. Out-of-the-way support workers should able to find hardware along with software arrangement of remote technology. Because of this they will find which software versions to post and make available knowledgeable support for problems.

Check your progress 5

1. Software Distribution refers to _____

   a. Working software stays at place placed on servers that are placed all through an organization

   b. OS is distributed

   c. Distribution of technology

   d. None of these

2.7 Performance network management issues

It is found that core minicomputer and mainframe environment with qualified technical support staff along with working staff will take care of presentation on uncompleted basis. IBM with Digital Equipment Corporation has characteristics in large computers operating systems which shows required energetic modification abilities. If tendency explain presentation humiliating, systems managers be able to add hardware or create alteration to get better performance sooner than it have an effect on user neighbourhood.

Certain tools as Crystal from BBN along with TPNS from IBM are presented to lower novel applications with earlier fabrication that acknowledge organization finding out which move forwards with resource needs for current applications. It is counted that alteration can be made to operating surroundings to make sure that presentation will be good enough.

It is found that inside client/server environment, neither UNIX, Windows NT, nor OS/2 up till now make available such simple performance catering tools. Many tools which can be Network General's Sniffer are present to independent handling of LAN traffic. UNIX, Windows NT and OS/2 will show restricted qualities which explains work preference. Certain retailers market products in order to handle such requirements. Currently, even if, the design knowledge of project architects is necessary to keep away from presentation deficiency. Providentially, price of hardware for client workstations or Windows NT, OS/2, and UNIX servers endures adding extra capacity to enhance performance that is normally not act as main cost factor for client/server system.

Certain network management tools which can be OpenVision, IBM's NetView, AT&T's UNMA as well as Digital Equipment Corporation's EMA

products will show level of inaccessible monitoring which can follow reaction time along with network weight. No such products will show type of analysis of remote server which RMF describes that can be tuning tools highlighting inside MVS along with VMS. Further, products like ESRA from Elegant Computing are present to perform isolated analysis of UNIX servers in command to keep an eye on disk usage, error logs, as well as user profiles. This product is used extensively to manage remote UNIX servers.

Apart from earlier products, some are Microcoms LANlord which describes important ability for isolated admission to Windows in addition to OS/2 PC LAN desktops. It is not possible to supply sufficient hold up for disseminated client/server applications with no means to carry desktop along with server distantly. This is a region of strong focus by industry in 1993 where many major systems integrators implemented by NOS to show desktop support for Novell, LAN Manager, LAN Server, and NFS client/server environments.

Check your progress 6

1. Crystal tool is the product of _____.
   a. Microsoft
   b. BBN
   c. IBM
   d. None of these
2. Which of the following is network management tool?
   a. OpenVision
   b. NetView
   c. UNMA
   d. All of these

2.8 Case studies-Cloud

Overview

The Client is a New York based provider of reports on market trends as hosted service since 2004 having Fortune 500 companies from finance consultants

Client
Server
Development

to jobsite domains as clients. The Project was to build a Trend Analysis Data Aggregator.

The Trend Analysis Data Aggregator is the base component of the client's business model that involves collection of data from websites related to Job Search and related services, Real Estate, second hand Motor Vehicles sales and second hand Books sales.

Challenges

The vision behind Trend Analysis Data Aggregator project was to outsource data aggregation from different websites to e-Zest so that client could focus on core business area of selling the data after due processing and analysis.

Solution

RegEx Designer serves as a front end for developers to develop data capture patterns for new sites as well as to change patterns for sites coming in for maintenance. Site Visualizer gets the website pages to be screen scraped and shows its HTML within the front end. Using this visualization, developers can go through the site updating data capturing and site navigation patterns and specifying processing instructions.

Patterns and instructions created by developers are tested using local database by the Data Engine. The data engine displays screen scraping results on console and errors encountered can be analyzed to test and refine URLs, patterns and processing instructions. Once the site scraping is successfully executed with local database, the patterns are transferred to client's delivery database using remote Data Transformation Services of SQL Server 2005.

Conclusion

Outsourcing the data aggregation function of client's business model to e-Zest enabled them to build a reliable offshore resource base specializing in screen scraping using a custom-made data engine that helped them cut costs, implement a flexible resource ramp-up or ramp-down plan based on projections of business expansions, focus their efforts on data analysis and sales, and enabled client to undertake ambitious growth strategy without worrying about operational issues related to core data aggregation function.

Check your progress 7

1. Which Front end server is described in the paragraph above?

   a. Oracle

   b. SQL Server

   c. VB

   d. None of above

2.9 Let Us Sum Up

While studying this unit, we have learnt that computer users earlier have dissimilar prospect where condition per hour on worksheet makes suspends the system where power fails furthermore virus makes system to start again. It is known that availability is system uptime or potential of system to access for information and accepts work at whatever time being asked.

In dispersed client/server representation, working software stays at place placed on servers that are placed all through an organization. Alteration to system with application software needs replication all through an organization which shows marvellous complication in serviceability of applications.

2.10 Answers for Check Your Progress

Check your progress 1

Answers: (1-c)

Check your progress 2

Answers: (1-d)

Check your progress 3

Answers: (1-c)

Check your progress 4

Answers: (1-d)

Check your progress 5

Answers: (1-a)

Check your progress 6

Answers: (1-b), (2-d)

Check your progress 7

Answers: (1-b)

2.11 Glossary

1. **Personal computer -** Home computer that does all function as standard computer and works in the similar manner.
2. **Availability -** In software, availability is system uptime to access information to do certain task.

2.12 Assignment

Write short note on System Administration.

2.13 Activities

Collect some information on Serviceability.

2.14 Case Study

Generalised the Software Distribution.

2.15 Further Readings

1. Boehm, B.W., "Improving Software Productivity", Computer, 20(8), 43-58, 1987.

Block Summary

In this block, you have learnt and understand about the basic objectives of effective project management along with information on System Development environment. The block gives an idea on the study and concept of System Administration along with availability and serviceability of software. You have been well explained on the concepts of Client/Server System Development in terms of Hardware configuration.

The block detailed about the basic of Performance network management issues. The concept related to Software Distribution and System Development are well detailed to you. You will be demonstrated practically about various Data storage disk storages devices.

Block Assignment

Short Answer Questions

1. Define System Development environments.
2. What are the features of Project management?
3. Explain Client System Administration.
4. Discuss Performance network management issues.

Long Answer Questions

1. Write short notes Availability and Serviceability.
2. Discuss Performance network management issues.
3. What are the various Productivity Measures?

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

Education is something which ought to be brought within the reach of every one.

- Dr. B. R. Ambedkar

Dr. Babasaheb Ambedkar Open University
'Jyotirmay Parisar', Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,
Ahmedabad-382 481.

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BCA - 404

BLOCK 4:
INTRODUCTION TO C#

Dr. Babasaheb Ambedkar Open University
Ahmedabad

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Er. Nishit Mathur

Language Editor
Prof. Jaipal Gaikwad

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Ms. Pinaz Driver
Ms. Tejashree Bhosale
Mr. Kiran Shinde
Mr. Prashant Tikone
Mr. Akshay Mirajkar

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

Contents

BLOCK 3: CLIENT SERVER DEVELOPMENT

UNIT 1 CLIENT SERVER SYSTEM DEVELOPMENT SOFTWARE

Factors driving demand for Applications Software Development, Client/Server system Development methodology, Project management, Architecture Definition, System Development environment, Productivity Measures, CASE, Client/Server System Development –Hardware, Hardware/Network acquisition, PC level processing units, Unix, Unix workstation server Hardware, Data storage, Network interface Cards, Power protection Device.

UNIT 2 CLIENT/SERVER SYSTEM DEVELOPMENT

Service and Supports, System Administration, Availability, Serviceability, Software Distribution, performance network management issues, Case studies.

BLOCK 4: INTRODUCTION TO C#

UNIT 1 INTRODUCTION TO .NET

The .NET Framework: an Overview, Framework Components, Framework Versions, Types of Applications which can be developed using MS.NET, MS.NET Base Class Library, MS.NET Namespaces, MSIL/ Metadata and PE files., The Common Language Runtime (CLR), Managed Code, MS.NET Memory Management / Garbage Collection, Common Type System (CTS), Common Language Specification (CLS), Types of JIT Compilers, Security Manager

UNIT 2 ASP.NET AND C#P

Working with Standard Controls, Navigation Controls, Validation Controls, Login Controls, Introduction to ASP.NET Objects, Building the StyleSheet, Creating the Content Master, Adding Elements, Building the Site Navigation, Adding Authentication, Adding Content Pages, Working with Data, Using ASP.NET Web Services and WCF, Creating a simple ASP.NET Web Service

---

Dr. Babasaheb
Ambedkar
Open University

BCA - 404

CLIENT SERVER ARCHITECTURE AND INTERFACES (C#)

BLOCK 4: INTRODUCTION TO C#

BLOCK 4: INTRODUCTION TO C#

Block Introduction

The .NET framework provides a set of base class libraries which provide functions and features which can be used with any programming language which implements .NET, such as Visual Basic, C#, Visual C++, etc. Namespaces are the way to organize .NET Framework Class Library into a logical grouping according to their functionality, usability as well as category they should belong to, or we can say Namespaces are logical grouping of types for the purpose of identification.

In this block, we will detail about the basic of .NET Framework with various versions. The block will focus on types of MS.NET applications, Base Class Library and MS.NET Namespaces. The students will give an idea on DMA Common Language Runtime and MS.NET Memory Management/Garbage Collection.

In this block, you will made to learn and understand about Working with Standard Controls along with ASP.NET Objects. The concept related to Creating the Content Master and building Site Navigation will also be explained to you. You will be demonstrated practically about working with Data technique.

Block Objective

After learning this block, you will be able to understand:

- About MS.NET Base Class Library
- Features of MS.NET Namespaces
- Idea about Metadata and PE files
- Concept of Common Language Runtime (CLR)
- Concept of MS.NET Memory Management / Garbage Collection
- Features of building Site Navigation
- Steps to add Authentication and Content Pages
- Working characteristics of Data
- Understanding about ASP.NET WebServices and WCF

Block Structure

Unit 1: Introduction to .NET

Unit 2: ASP.Net and C#

UNIT 1: INTRODUCTION TO .NET

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- About MS.NET Namespaces
- About MSIL / Metadata and PE files
- About Managed Code
- About Common Type System (CTS)

1.1 Introduction

.Net is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web application for PC, as well as mobile devices.

ASP.Net works on top of the HTTP protocol and uses the HTTP commands and policies to set a browser-to-server two-way communication and cooperation. ASP.NET is an open source server-side Web application framework designed for Web development to produce dynamic Web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, web applications and web services.

It was first released in January 2002 with version 1.0 of the .NET Framework, and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language. The ASP.NET SOAP extension framework allows ASP.NET components to process SOAP messages.

1.2 The .NET Framework: an Overview

C# programs run on the .NET Framework, an integral component of Windows that includes a virtual execution system called the common language runtime (CLR) and a unified set of class libraries. The CLR is the commercial implementation by Microsoft of the common language infrastructure (CLI), an international standard that is the basis for creating execution and development environments in which languages and libraries work together seamlessly.

Source code written in C# is compiled into an intermediate language (IL) that conforms to the CLI specification. The IL code and resources, such as bitmaps and strings, are stored on disk in an executable file called an assembly, typically with an extension of .exe or .dll. An assembly contains a manifest that provides information about the assembly's types, version, culture, and security requirements.

When the C# program is executed, the assembly is loaded into the CLR, which might take various actions based on the information in the manifest. Then, if the security requirements are met, the CLR performs just in time (JIT) compilation to convert the IL code to native machine instructions. The CLR also provides other services related to automatic garbage collection, exception handling, and resource management. Code that is executed by the CLR is sometimes referred to as "managed code," in contrast to "unmanaged code" which is compiled into native machine language that targets a specific system. The following diagram illustrates the compile-time and run-time relationships of C# source code files, the .NET Framework class libraries, assemblies, and the CLR.

Check your progress 1

1. C# programs run on _____

   a. VB framework

   b. .NET Framework

   c. VC framework

   d. all of above

1.3 Framework Components

To better understand the CLR, consider how compilers that target the .NET Framework differ from traditional compilers.

Traditional compilers target a specific processor, consuming source files in a specific language, and producing binary files containing streams of instructions in the native language of the target processor. These binary files may then be executed directly on the target processor.

.NET compilers function a little differently, as they do not target a specific native processor. Instead, they consume source files and produce binary files containing an intermediate representation of the source constructs, expressed as a combination of metadata and Common Intermediate Language (CIL). In order for these binaries to be executed, the CLR must be present on the target machine.

When these binaries are executed they cause the CLR to load. The CLR then takes over and manages execution, providing a range of services such as JIT compilation (converting the CIL as needed into the correct stream of instructions for the underlying processor), memory management (in the form of a garbage collector), exception management, debugger and profiler integration, and security services.

This compilation and execution model explains why C# is referred to as a managed language, why code running in the CLR is referred to as managed code, and why the CLR is said to provide managed execution.

Although this dependency on a runtime environment might initially appear to be a drawback, substantial benefits arise from this architecture. Since the metadata and CIL representations are processor architecture-neutral, binaries may be used on any machine in which the Common Language Runtime is present, regardless of underlying processor architecture. Additionally, since processor-specific code generation is deferred until runtime, the CLR has the opportunity to perform processor-specific optimizations based on the target architecture the code is running on. As processor technology advances, all applications need to take advantage of these advances is an updated version of the CLR.

Unlike traditional binary representations, which are primarily streams of native processor instructions, the combination of metadata and CIL retains almost all of the original source language constructs. In addition, this representation is source language-neutral, which allows developers to build applications using multiple source languages. They can select the best language for a particular task, rather than being forced to standardize on a particular source language for each application or needing to rely on component technologies, such as COM or CORBA, to mask the differences between the source languages used to build the separate components of an application.

Check your progress 2

1. CLR is known as:

a. Common Language Runtime

b. Common Loading Runtime

c. Clear Language Runtime

d. Clear Loading Runtime

1.4 Framework Versions

Each version of the .NET Framework contains the common language runtime (CLR), the base class libraries, and other managed libraries. Each new version of the .NET Framework retains features from the previous versions and adds new features. The CLR is identified by its own version number. The .NET Framework version number is incremented at each release, although the CLR version is not always incremented. For example, the .NET Framework 4, 4.5, and later releases include CLR 4, but the .NET Framework 2.0, 3.0, and 3.5 include CLR 2.0. (There was no version 3 of the CLR.)

The following table summarizes .NET Framework version history and correlates each version with Visual Studio, Windows, and Windows Server.

.NET Framework version	CLR version	Features
Net 4.6.1	4	Support X509 having ECDSA Encrypt support for hardware keys in ADO.NET Spell checking
.NET 4.6	4	Compile using .NET Native

		ASP.NET Core 5 Event tracing quality Supports page encodings
4.5.2	4	New APIs for transactional systems and ASP.NET System DPI resizing in Windows Profile improvements ETW and stress logging improvements
4.5.1	4	Support for Windows Phone Store apps Automatic binding redirection Performance and debugging improvements
4.5	4	Support Windows Store apps WPF, WCF, WF, ASP.NET updates
4	4	Expanded base class libraries

		Cross-platform development with Portable Class Library MEF, DLR, code contracts
3.5	2.0	AJAX-enabled websites LINQ Dynamic data
3.0	2.0	WPF, WCF, WF, CardSpace
2.0	2.0	Generics ASP.NET additions
1.1	1.1	ASP.NET and ADO.NET updates Side-by-side execution
1.0	1.0	First version of the .NET Framework.

Check your progress 3

1. Which .NET version uses ASP.NET Core 5?

   a. NET 4.6.1

   b. NET 4.6

   c. NET 4

   d. NET 3.5

2. Which version carries side by side execution?

   a. 3.0

   b. 2.0

   c. 1.1

   d. 1.0

1.5 Types of Applications using MS.NET

All types of .NET applications use one or more .NET compliant languages for their design and development. The .NET Framework includes various technologies, such as ASP.NET, VB.NET, VC++.NET, and ADO.NET. Here is a more or less full list of various types of application that we can develop on .NET.

ASP.Net Web applications:

These are the programs that used to run inside some web server to fulfill the user requests over the http. ASP.NET Web applications can range from simple Web sites that consist of HTML pages to advanced enterprise applications that run on local and remote networks. These enterprise applications also provide components for exchanging data using XML. This type includes dynamic and data driven browser based applications. (Ex: Hotmail and Google).

Web services:

There are certain web services that are functionally available with certain industrial standards like HTTP, XML and SOAP.

Windows applications:

These are types of Windows desktop applications for every day working which can be MS word. Run only under Windows environment. These applications consume the services provided by the Windows operating system.

Windows services:

These are long-running executable applications which runs on the system as a background process. These applications do not interfere with the working of the other processes that run on the same computer. Windows services execute within separate Windows sessions created specifically for each Windows service. These services do not have a graphic user interface and are ideal for running on the server. Windows services were earlier called NT services.

Console applications:

These are light weight programs run inside the command prompt (DOS) window. They are commonly used for test applications.

Mobile applications:

They are applications which work on multiple mobile devices and shows ubiquitous access to data from mobile devices. The .NET Framework automatically makes changes to these applications to enable them to run on multiple browsers, depending on the mobile device.

Class libraries:

These are components that you create once and reuse a number of times in multiple applications. Class libraries allow you to define several classes, along with their methods and interfaces, in one file. These libraries compile to .dll files and facilitate rapid development of new applications because of reusability of code. To access the functionality of the classes in a class library from your application, you need to include a reference to that library in your program.

Check your progress 4

1.The change any Mobile application in .NET framework works on:

a. single browser

b. two browsers

c. multiple browser

d. all of above

1.6 MS.NET Base Class Library

The .NET framework provides a set of base class libraries which provide functions and features which can be used with any programming language which implements .NET, such as Visual Basic, C#, Visual C++, etc.

The base class library contains standard programming features such as Collections, XML, DataType definitions, IO, Reflection and Globalization to name a few. All of which are contained in the System namespace. As well, it contain some non-standard features such as LINQ, ADO.NET, drawing capabilities, forms and web support. The below table provides a list each class of the base class library and a brief description of what they provide.

Base Class Library Namespace	**Brief Description**
System	Contains the fundamentals for programming such as the data types, console, match and arrays, etc.
System.CodeDom	Supports the creation of code at runtime and the ability to run it.
System.Collections	Contains Lists, stacks, hashtables and dictionaries
System.ComponentModel	Provides licensing, controls and type conversion capabilities
System.Configuration	Used for reading and writing program configuration data
System.Data	Is the namespace for ADO.NET
System.Deployment	Upgrading capabilities via ClickOnce
System.Diagnostics	Provides tracing, logging, performance counters, etc. functionality

System.DirectoryServices	Is the namespace used to access the Active Directory
System.Drawing	Contains the GDI+ functionality for graphics support
System.EnterpriseServices	Used when working with COM+ from .NET
System.Globalization	Supports the localization of custom programs
System.IO	Provides connection to file system and the reading and writing to data streams such as files
System.Linq	Interface to LINQ providers and the execution of LINQ queries
System.Linq.Expressions	Namespace which contains delegates and lambda expressions
System.Management	Provides access to system information such as CPU utilization, storage space, etc.
System.Media	Contains methods to play sounds
System.Messaging	Used when message queues are required within an application, superseded by WCF
System.Net	Provides access to network protocols such as SSL, HTTP, SMTP and FTP
System.Reflection	Ability to read, create and invoke class information.
System.Resources	Used when localizing a program in relation to language support on web or form controls
System.Runtime	Contains functionality which allows the management of runtime behavior.

System.Security	Provides hashing and the ability to create custom security systems using policies and permissions.
System.ServiceProcess	Used when a windows service is required
System.Text	Provides the StringBuilder class, plus regular expression capabilities
System.Threading	Contains methods to manage the creation, synchronization and pooling of program threads
System.Timers	Provides the ability to raise events or take an action within a given timer period.
System.Transactions	Contains methods for the management of transactions
System.Web	Namespace for ASP.NET capabilities such as Web Services and browser communication.
System.Windows.Forms	Namespace containing the interface into the Windows API for the creation of Windows Forms programs.
System.Xml	Provides the methods for reading, writing, searching and changing XML documents and entities.

Check your progress 5

1. Which Base Class Library supports creation of code at runtime?

   a. System.CodeDom

   b. System.Collections

   c. System.ComponentModel

   d. System.Configuration

2. The function of Base Class Library System.DirectoryServices is:

   a. access Active Directory

   b. function for graphics support

   c. applied in COM+ from .NET

   d. supporting localization of custom programs

1.7 MS.NET Namespaces

Namespaces are the way to organize .NET Framework Class Library into a logical grouping according to their functionality, usability as well as category they should belong to, or we can say Namespaces are logical grouping of types for the purpose of identification.

The .NET Framework Class Library (FCL) is a large collection of thousands of Classes. These Classes are organized in a hierarchical tree. The System Namespaces is the root for types in the .NET Framework. We can uniquely identify any Class in the .NET Framework Class Library (FCL) by using the full Namespaces of the class .In .Net languages every program is created with a default Namespaces. Programmers can also create their own Namespaces in .Net languages.

Please now have a look at the example of declaring some namespace:

```
using System;

namespace OutNamespace

{

namespace WorkNamespace

{ /// can be placed some classes, structures etc

}

}
```

In this example we create two namespaces. These namespaces have hierarchical structure. We have some outer one named OutNamespace and the inner one called WorkNamespace. The inner namespace is declared with a C# .Net class WorkItem.

Check your progress 6

1.Name space are associated with:

a. items

b. class

c. work

d. all of above

1.8 MSIL / Metadata and PE files

Microsoft Intermediate Language (MSIL)

A.NET programming language (C#, VB.NET, J# etc.) does not compile into executable code; instead it compiles into an intermediate code called Microsoft Intermediate Language (MSIL). As a programmer one need not worry about the syntax of MSIL - since our source code in automatically converted to MSIL. The MSIL code is then send to the CLR (Common Language Runtime) that converts the code to machine language, which is, then run on the host machine. MSIL is similar to Java Byte code. MSIL is the CPU-independent instruction set into which .NET Framework programs are compiled. It contains instructions
for loading, storing, initializing, and calling methods on objects. Combined with metadata and the common type system, MSIL allows for true cross- language

integration Prior to execution, MSIL is converted to machine code. It is not interpreted.

.

When .net programming code is compiled ,it is not convert into exe files and it is convert into Microsoft intermediate languages, and after that clr convert the this msil code into machine level language.

Microsoft Intermediate Language (MSIL) is a CPU-independent set of instructions that can be efficiently converted to the native code. During the runtime the Common Language Runtime (CLR)'s Just In Time (JIT) compiler converts the Microsoft Intermediate Language (MSIL) code into native code to the Operating System.
When a compiler produces Microsoft Intermediate Language (MSIL), it also produces Metadata. The Microsoft Intermediate Language (MSIL) and Metadata are contained in a portable executable (PE) file . Microsoft Intermediate Language (MSIL) includes instructions for loading, storing, initializing, and calling methods on objects, as well as instructions for arithmetic and logical operations, control flow, direct memory access, exception handling, and other operations
Just In Time Compiler
.

Microsoft .Net Metadata

Metadata in .Net is binary information which describes the characteristics of a resource . This information include Description of the Assembly , Data Types and members with their declarations and implementations, references to other types and members , Security permissions etc. A module's metadata contains everything that needed to interact with another module.

During the compile time Metadata created with Microsoft Intermediate Language (MSIL) and stored in a file called a Manifest . Both Metadata and Microsoft Intermediate Language (MSIL) together wrapped in a Portable Executable (PE) file. During the runtime of a program Just In Time (JIT) compiler of the Common Language Runtime (CLR) uses the Metadata and converts Microsoft Intermediate Language (MSIL) into native code. When code is executed, the runtime loads metadata into memory and references it to discover information about your code's classes, members, inheritance, and so on. Moreover Metadata eliminating the need for Interface Definition Language (IDL) files, header files, or any external method of component reference.

Portable Executable (PE) File Format

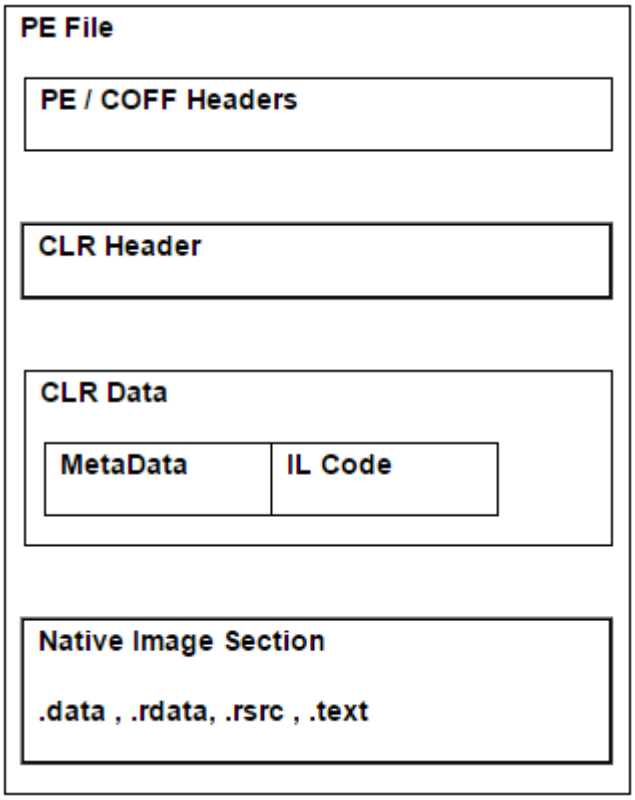
The Portable Executable (PE) format is a file format for executables, object code, and DLLs, used in 32-bit and 64-bit versions of Windows operating systems. The PE file format was defined to provide the best way for the Windows

Operating System to execute code and also to store the essential data which is needed to run a program. Portable Executable File Format is derived from the Microsoft Common Object File Format (COFF). Almost every Windows executable, DLL or EXE, is a Portable Executable (PE) format file. Although there is little in the PE format that lends itself to .NET, in the current implementation of .NET all assemblies are contained in special PE format files, which have some traditional bits left out and quite a lot of new bits put in.

Very generally, a PE file consists of a PE header, which contains a list of Data Directory entries, and a number of Sections which are defined just after the PE header. Not all the Data Directories have meaning in a .NET file, and not many Sections are present either. Nevertheless, those that remain are still important -- in particular, the last Data Directory entry points to the start of .NET information.

The real starting point of a PE file, from the .NET point of view, is the COR20 Header, which tells the .NET runtime where to find the metadata. The COR20 header, like the PE header, specifies some Data Directories, as well as the entry point for the assembly. Most of these Data Directories point to things like fix up information which is not useful for examining the assembly, but one of them points to the start of the Metadata Streams.

Format of .Net PE file



Check your progress 7

1.Which is not a .NET programming language:

a. Oracle

b. C#,

c. VB.NET

d. J#

1.9 Common Language Runtime (CLR)

The Common Language Runtime (CLR) is an Execution Environment which is the backbone of the .NET framework. It works as a layer between Operating Systems and the applications written in .Net languages that conforms to the Common Language Specification (CLS). The main function of Common Language Runtime (CLR) is to convert the Managed Code into native code and then execute the Program. The Managed Code compiled only when it needed, that is it converts the appropriate instructions when each function is called. The Common Language Runtime (CLR) 's Just In Time (JIT) compilation converts Intermediate Language (MSIL) to native code on demand at application run time.

CLR handles the execution of code and provides useful services for the implementation of the program. In addition to executing code, CLR provides services such as memory management, thread management, security management, code verification, compilation, and other system services.

During the execution of the program, the Common Language Runtime (CLR) manages memory, Thread execution, Garbage Collection (GC) , Exception Handling, Common Type System (CTS), code safety verifications, and other system services. The CLR (Common Language Runtime) defines the Common Type System (CTS), which is a standard type system used by all .Net languages. That means all .NET programming languages uses the same representation for common Data Types, so Common Language Runtime (CLR) is a language-independent runtime environment. The Common Language Runtime (CLR) environment is also referred to as a managed environment, because during the execution of a program it also controls the interaction with the Operating System.

Check your progress 8

1. Common Language Runtime converts:

   a. Managed Code into native code

   b. Native code into managed code

   c. neither a nor b

   d. both a and b

1.10 Managed Code

Managed code is the code that is written to target the services of the managed runtime execution environment such as Common Language Runtime in .Net Technology.

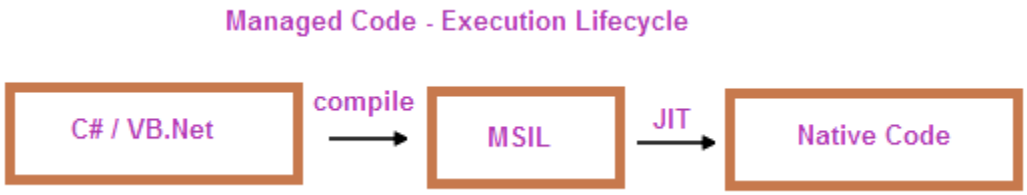


Fig 1.1 Managed Code

The Managed Code running under a Common Language Runtime cannot be accessed outside the runtime environment as well as cannot call directly from outside the runtime environment. It refers to a contract of cooperation between natively executing code and the runtime. It offers services like garbage collection, run-time type checking, reference checking etc. By using managed code you can avoid many typical programming mistakes that lead to security holes and unstable applications, also, many unproductive programming tasks are automatically taken care of, such as type safety checking, memory management, destruction of unused Objects etc.

Check your progress 9

1. In Managed code execution cycle, compile data is send to:

   a. MSIL

   b. Native code

   c. both a and b

   d. neither a nor b

1.11 MS.NET Memory Management / Garbage Collection

In .NET, memory is managed through the use of Managed Heaps. Generally in case of other languages, memory is managed through the Operating System directly. The program is allocated with some specific amount of memory for its use from the Raw memory allocated by the Operating system and then used up by the program. In case of .NET environment, the memory is managed through the CLR (Common Language Runtime) directly and hence we call .NET memory management as Managed Memory Management. .Net manages memory automatically

- Creates objects into memory blocks(heaps)
- Destroy objects no longer in use

Allocates objects onto one of two heaps

- Small object heap(SOH) – objects < 85k
- Large object heap(LOH) – objects >= 85k

You allocate onto the heap whenever you use the "new" keyword in code. In .Net, CLR has garbage collector that executes as a part of our program and responsible for reclaiming the memory of no longer used objects. Garbage collectors free the memory for objects that are no longer referenced and keep the memory for future allocations. The advantage of Garbage Collector is:

- Allow us to develop an application without having worry to free memory.
- Allocates memory for objects efficiently on the managed heap.
- Reclaims the memory for no longer used objects and keeps the free memory for future allocations.

- Provides memory safety by making sure that an object cannot use the content of another object.

The managed heap is a series of allocated memory segments (approx 16Mb in size each) to store and manage objects. The memory for newly created object is allocated at the next available location on the managed heap. If there is available free memory, the garbage collector doesn't search the dead objects for memory reclaim and memory allocations have been done very fast. If the memory is insufficient to create the object, the garbage collector search the dead objects for memory reclaim for the newly object.

Garbage collector determines whether any object in the heap is dead or not being used by the application. If such objects exist then memory used by these objects can be reclaimed. Each and every application has a set of roots and these identify the storage locations for the objects on the managed heap. In Garbage Collector:

- All objects in the heap are allocated from one contiguous range of memory address and heap is divided into generations so that it is easy to eliminate the garbage objects by looking at only a small fraction of the heap.
- Gen 0 and Gen 1 occupy a single segment known as the ephemeral segment. Gen 2 is a set of further segments and the large object heap is yet another group of segments.
- Almost, all objects with-in a generation are of the same age.
- The newest objects are created at higher memory address while oldest memory objects are at lowest memory address with in the heap.
- The allocation pointer for the new objects marks the boundary between the allocated and free memory.
- Periodically the heap is compacted by removing the dead objects and sliding up the live objects towards the lower memory address end of the heap as shown in above fig.
- The order of objects (after memory reclaims) in memory remains the same as they were created.
- There are never any gaps among the objects in the heap.

Check your progress 10

1. The approximate size of allocated memory segments in managing heap so as to store and manage objects will be:

   a. 16Mb

   b. 32Mb

   c. 64Mb

   d. 128Mb

1.12 Common Type System (CTS)

Common Type System (CTS) describes a set of types that can be used in different .Net languages in common. That is, the Common Type System (CTS) ensure that objects written in different .Net languages can interact with each other. For Communicating between programs written in any .NET complaint language, the types have to be compatible on the basic level.

These types can be Value Types or Reference Types. The Value Types are passed by values and stored in the stack. The Reference Types are passed by references and stored in the heap. Common Type System (CTS) provides base set of Data Types which is responsible for cross language integration. The Common Language Runtime (CLR) can load and execute the source code written in any .Net language, only if the type is described in the Common Type System (CTS).

Importance of CTS:

- CTS are responsible for cross language Integration and Type Safety.
- Enforce a set of rules that a programming language must follow.

Check your progress 11

1. The Common Type System can be:

   a. Value Type

   b. Reference Type

   c. Both a and b

   d. Neither a nor b

1.13 Common Language Specification (CLS)

Common Language Specification (CLS) is a set of basic language features that .Net Languages needed to develop Applications and Services, which are compatible with the .Net Framework. When there is a situation to communicate Objects written in different .Net Complaint languages, those objects must expose the features that are common to all the languages. Common Language Specification (CLS) ensures complete interoperability among applications, regardless of the language used to create the application.

Common Language Specification (CLS) defines a subset of Common Type System (CTS). Common Type System (CTS) describes a set of types that can use different .Net languages have in common, which ensure that objects written in different languages can interact with each other. Most of the members defined by types in the .NET Framework Class Library (FCL) are Common Language Specification (CLS) compliant Types.

Check your progress 12

1. Common Language Specification develops:

   a. application

   b. services

   c. both a and b

   d. neither a nor b

1.14 Types of JIT Compilers

JIT stands for just-in-time compiler. It converts the MSIL code to CPU native code as it is needed during code execution. It is called just-in-time since it converts the MSIL code to CPU native code; when it is required within code execution otherwise it will not do anything with that MSIL code.

Different Types of JIT

Normal JIT

This complies only those methods that are called at runtime. These methods are compiled only first time when they are called, and then they are stored in memory cache. This memory cache is commonly called as JITTED. When the

same methods are called again, the complied code from cache is used for execution.

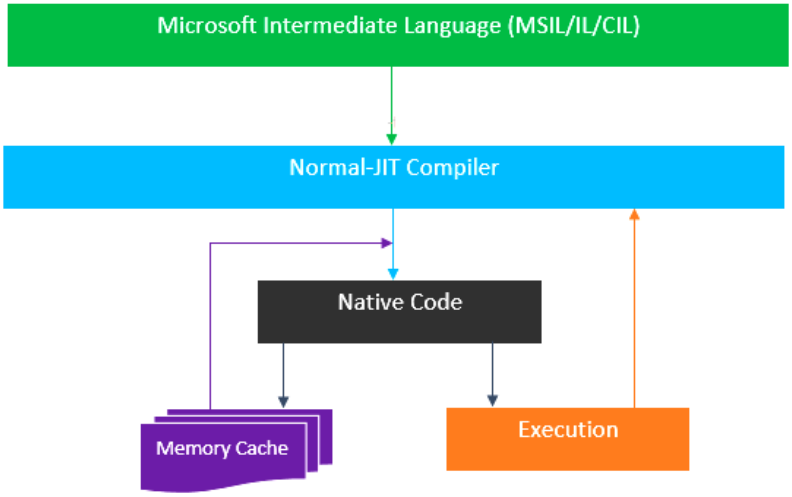


Fig 1.2 Normal JIT Layout

Econo JIT

This complies only those methods that are called at runtime and removes them from memory after execution.

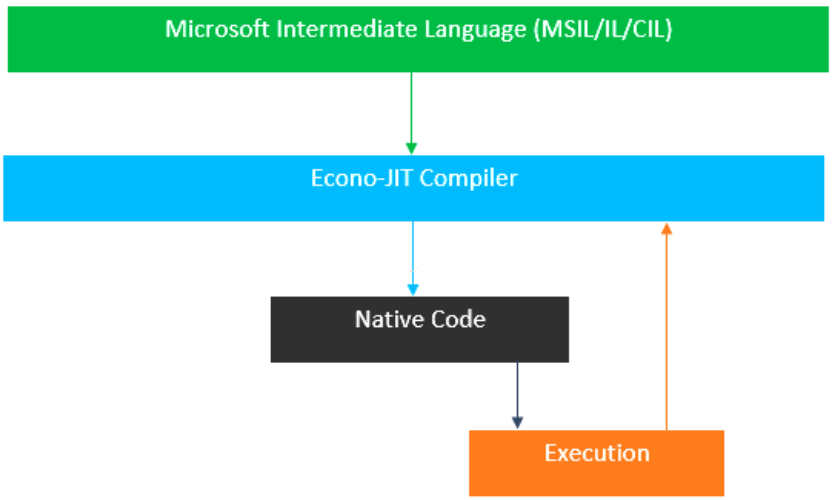


Fig 1.3 Econo JIT

Pre JIT

This complies entire MSIL code into native code in a single compilation cycle. This is done at the time of deployment of the application.

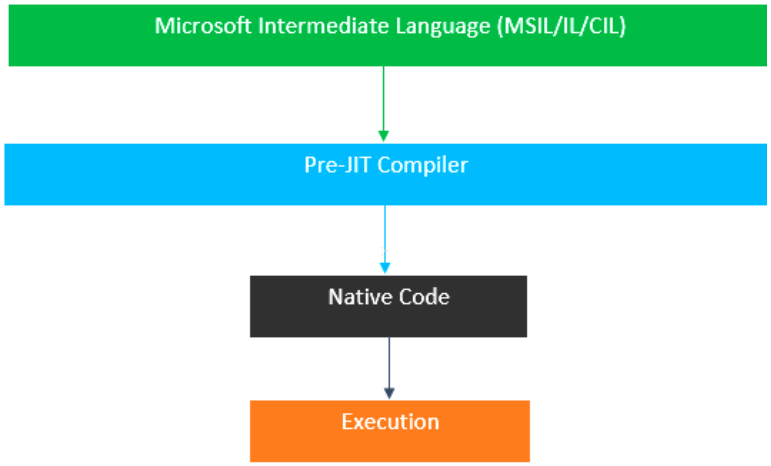


Fig 1.4 Pre JIT

Check your progress 13

1. Just-in-time converts:

   a. MSIL code to CPU native code

   b. CPU native code to MSIL code

   c. both a and b

   d. neither a nor b

1.15 Security Manager

A security manager is an object that defines a security policy for an application. This policy specifies actions that are unsafe or sensitive. Any actions not allowed by the security policy cause a Security Exception to be thrown. An application can also query its security manager to discover which actions are allowed. In a SecurityManager class as shown below:

```
public class SecurityManager
{
   private static readonly SecurityManager _current = new SecurityManager();
```

```
    public static SecurityManager Current
    {
        get { return _current; }
    }

    public bool CanSeeAdminPanel
    {
        // Change this code to a role that does/doesn't exist
        get { return Thread.CurrentPrincipal.IsInRole("Administrators"); }
    }
}
```

Further we see that:

```
<asp:Panel
    ID="adminPanel"
    runat="server"
    Visible="<%#     ExampleApplication.SecurityManager.Current.IsAdministrator
%>"
    >
    This is only visible to admins
</asp:Panel>
```

Check your progress 14

1. Security Manager is:

    a. application

    b. object

    c. data

    d. none of above

1.16 Let Us Sum Up

In this unit we have learnt that there are components that can be created once and can be further reuse number of times in various applications. It is seen that class libraries allow to define several classes, along with their methods and interfaces, in one file. It is seen that .NET framework show set of base class libraries having functions and features that are used with programming language for implementing .NET, such as Visual Basic, C#, Visual C++, etc.

We see that namespaces are the way to organize .NET Framework Class Library in logical grouping as per functionality, usability and category. A.NET programming language does not compile into executable code but compiles in intermediate code as Microsoft Intermediate Language (MSIL). The Portable Executable (PE) format is a file format for executables, object code, and DLLs, used in 32-bit and 64-bit versions of Windows operating systems.

Garbage collector determines whether any object in the heap is dead or not being used by the application. If such objects exist then memory used by these objects can be reclaimed. Common Type System shows set of types which can be used in different .Net languages in common that ensure objects interaction in different .Net languages. Common Language Specification is set of basic language features that .Net Languages needed to develop Applications and Services, which are compatible with the .Net Framework.

A security manager is an object that defines a security policy for an application. This policy specifies actions that are unsafe or sensitive. Any actions not allowed by the security policy cause a Security Exception to be thrown.

1.17 Answer for Check Your Progress

Check your progress 1

Answers: (1-b)

Check your progress 2

Answers: (1-a)

Check your progress 3

Answers: (1-b), (2-c)

Check your progress 4

Answers: (1-d)

Check your progress 5

Answers: (1-a), (2-a)

Check your progress 6

Answers: (1-b)

Check your progress 7

Answers: (1-a)

Check your progress 8

Answers: (1-a)

Check your progress 9

Answers: (1-a)

Check your progress 10

Answers: (1-a)

Check your progress 11

Answers: (1-c)

Check your progress 12

Answers: (1-c)

Check your progress 13

Answers: (1-c)

Check your progress 14

Answers: (1-b)

1.18 Glossary

1. **.NET framework -** Set of base class libraries describing functions and features used with programming language.
2. **Namespaces -** Way to organize .NET Framework Class Library in logical grouping depending on functionality and usability.
3. **Portable Executable -** It a type of file format used for executables, object code and DLLs for 32-bit and 64-bit versions of Windows operating systems.
4. **Managed code -** Code written to target services of managed runtime execution environment.
5. **Garbage collector -** Tool that shows whether object in heap is dead or not used by the application.

1.19 Assignment

Explain the .NET Framework.

1.20 Activities

What are the features of Common Language Runtime?

1.21 Case Study

What is the function of Common Type System?

1.22 Further Readings

1. Eric Gunnerson, a Programmer's Introduction to C#. Wiley
2. S. Robin, >NET Framework, Oxford

UNIT 2: ASP.Net and C#

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- Concept of Standard Controls
- Study about Validation Controls
- Study about ASP.NET Objects
- Features of Building StyleSheet
- Creating the Content Master

2.1 Introduction

C# is an elegant and type-safe object-oriented language that enables developers to build a variety of secure and robust applications that run on the .NET Framework. In this, C# can be used to create Windows client applications, XML Web services, distributed components, client-server applications, database applications, and much, much more. Visual C# provides an advanced code editor, convenient user interface designers, integrated debugger, and many other tools to make it easier to develop applications based on the C# language and the .NET Framework.

C# syntax is highly expressive, yet it is also simple and easy to learn. The curly-brace syntax of C# will be instantly recognizable to anyone familiar with C, C++ or Java. Developers who know any of these languages are typically able to begin to work productively in C# within a very short time. C# syntax simplifies many of the complexities of C++ and provides powerful features such as nullable value types, enumerations, delegates, lambda expressions and direct memory access, which are not found in Java. C# supports generic methods and types, which provide increased type safety and performance, and iterators, which enable implementers of collection classes to define custom iteration behaviors that are simple to use by client code. Language-Integrated Query (LINQ) expressions make the strongly-typed query a first-class language construct.

2.2 Working with Standard Controls

The standard controls in ASP.Net and C# are:

- Navigation

- Validation
- Login

2.2.1 Navigation Controls

We see that in ASP.NET there appears three navigation controls:

- Dynamic menus
- Tree Views
- Site Map Path

Menu Control: The Menu control will allow you to add navigational features in web pages. It handles main menu along with and submenus furthermore allows to show dynamic menus. It can be applied to other Navigation controls. Menu stored in file gets easily maintained which is normally called as web.

Tree Views: A Tree View will take care of displaying hierarchical list of items with the help of lines in order to connect interrelated items in a chain of command. Here all item has label and possible bitmap. It is found that Windows Explorer will make use of Tree View control to show directories. We can apply Tree View control in several circumstances in which you need to display hierarchical data.

Site Map Path: Use of this control is very simple. You can add this control to your page then view your page in browser. The Sitemap Path control displays the navigation path of the current page. The path acts as click able links to previous pages.

2.2.2 Validation Controls

Validation result as important feature of any web application where user input gets validated prior to sending from corner to corner in various layers of the application. It can be applied for:

- Implementing presentation logic.
- Validating user input data.
- Data format, data type and data range.

Validation is of two types:

- Client Side

- Serve Side

It is seen that in case of client side validation, the user will depend on browser as well as scripting language support, while in case of client side, the user will get immediate feedback. The benefit of this is that it saves page from being post back to server till client validates successfully.

So finally we see that validation involves testing of user data that gets entered in data field. After determining, user can additionally check for number or character entered as per required format. For creating ASP.NET Web pages, the user input will determine the information validity. So we see that in ASP.NET, set of validation controls will serve easy-to-use strong way of determining errors and will further display messages to user. There are six types of validation controls available in ASP.NET:

- Required Field Validation control
- Compare Validator Control
- Range Validator Control
- Regular Expression Validator Control
- Custom Validator Control
- Validation Summary

It is noted that in validation controls, ControlToValidate property is compulsory to every validate controls. In this, single validation control will validate only single input control whereas several validate control will also be assigned to input control.

2.2.3 Login Controls

In ASP.NET, login controls delivers robust login solution for ASP.NET Web applications without using programming. It is noted that normally, login controls integrate with ASP.NET membership and forms authentication to help automate user authentication for a Web site. It provides you with a ready-to-use user interface that queries the user name and password from the user and offers a Log In button for login. It validate user credentials against the membership API and encapsulating the basic forms authentication functionality like redirecting back to the original requested page in a restricted area of you application after the successful login.

The Login control displays a user interface for user authentication. The Login control contains text boxes for the user name and password and a check box that allows users to indicate whether they want the server to store their identity using ASP.NET membership and automatically be authenticated the next time they visit the site.

The Login control has properties for customized display, for customized messages, and for links to other pages where users can change their password or recover a forgotten password. The Login control can be used as a standalone control on a main or home page, or you can use it on a dedicated login page. If you use the Login control with ASP.NET membership, you do not need to write code to perform authentication. However, if you want to create your own authentication logic, you can handle the Login control's Authenticate event and add custom authentication code.

Check your progress 1

1. In the webpage, _____control is used to show hierarchical data.

   a. Menu

   b. Tree view

   c. Site Map Path

   d. All of above

2.3 Introduction to ASP.NET Objects

There are many objects in ASP.NET

Object Name	Description
Application:	It is applied to use information which is defined for complete application. It includes connection string applied to join database server which is stored in application object
Request	It allows ASP.NET applications to use information that is delivered by client at the time of Web request. It is a reference of HttpRequest Class Properties - QueryString , Cookies

Response	It allows ASP .NET application to deliver information to the client that gets referenced by HttpResponse ClassPublic void Write and Public void Redirect
Server	It gives methods applied to use methods along with properties of Web Server.

Check your progress 2

1. _____object method is applied for showing properties of Web Server.

   a. Response

   b. Server

   c. Request

   d. None of above

2.4 Building StyleSheet

Any beginner web developer / designer can easily apply a multitude of formatting to the web pages produced. One of the methods by which an experienced web designer can show up from the crowd is to provide flexibility to this formatting as well.

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation semantics (the look and formatting) of a document written in a markup language.

You can add an empty stylesheet to you web project, by:

1. Choose Website Website ➢ Add New Item in Visual Studio.
2. Select Style Sheet, edit the file name, and click OK.

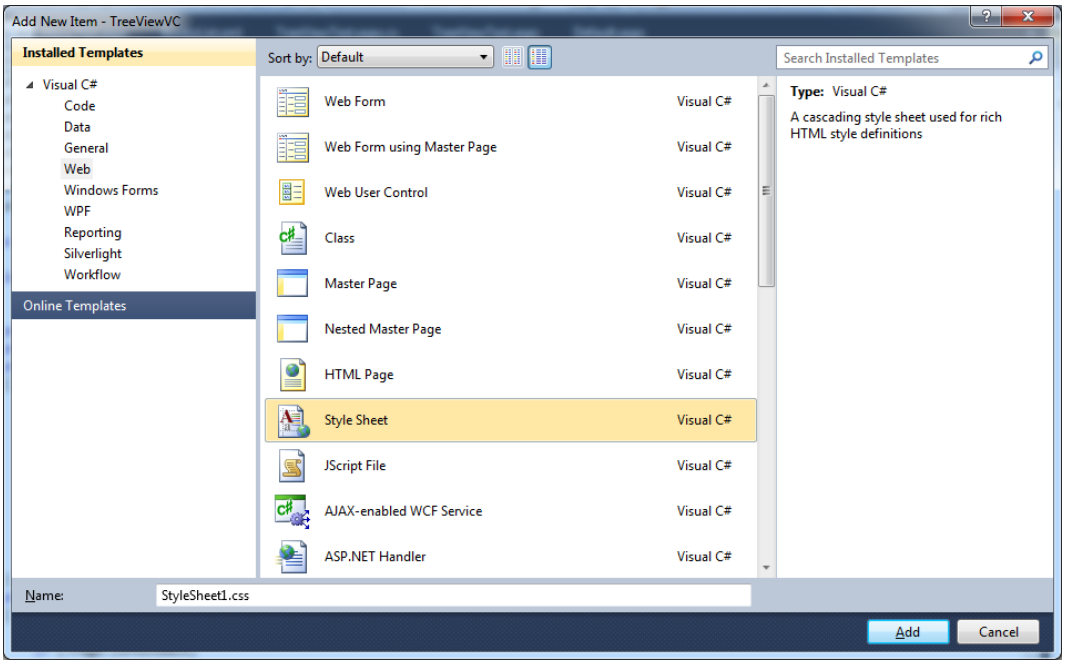


Fig 2.1 Adding Stylesheet

Stylesheets has certain rules which defines about formatting technique of every item in web page using certain rules:

1. The portion before the period specifies the HTML element to which the rule applies. If this part is empty the rule can apply to any tag.
2. The portion after the period is a unique case-sensitive name called the CSS class name. You use it to identify your rule.

A typical stylesheet defines a set of rules which are used to define the formatting for every significant piece of a website's user interface.

Check your progress 3

1. Style sheets are designed using:

   a. VC

   b. VB

   c. C++

   d. CSS

2.5 Creating the Content Master

Master page and content page work together to produce output to client. Sometimes, interaction between master page and content page is needed. Although too much of interaction between master page and content page can cause maintenance problems in future, it could be useful in some scenarios. There are few ways how content pages can access elements of master page.

A master page provides a template for other pages, with shared layout and functionality. The master page defines placeholders for the content, which can be overridden by content pages. The output result is a combination of the master page and the content page. The content pages contain the content you want to display.

When users request the content page, ASP.NET merges the pages to produce output that combines the layout of the master page with the content of the content page. The master page is a normal HTML page designed as a template for other pages. The @ Master directive defines it as master page having placeholder tag <asp:ContentPlaceHolder> for every content.

Step 1: Add Master page, right click in project property, and then click on Add new Item, now you will see Installed Template where you can select Master Page, finally click on **OK**.

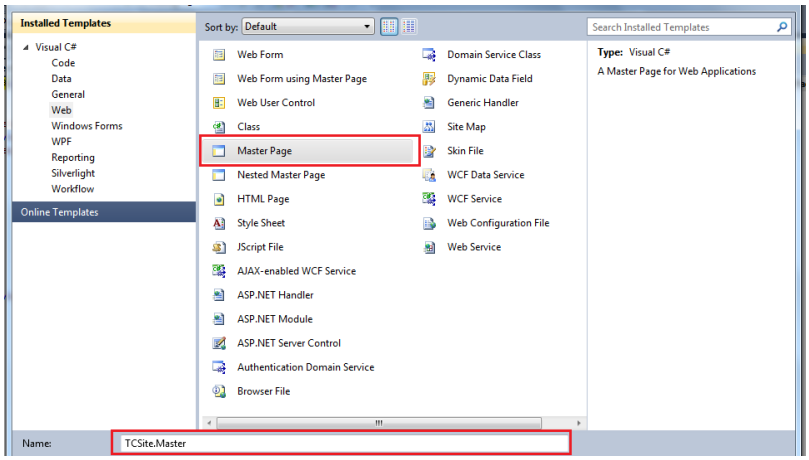


Fig 2.2 Installed template

Step 2: Now you can right click on Master Page (TCSite.Master), and then click **Add Content Page**.

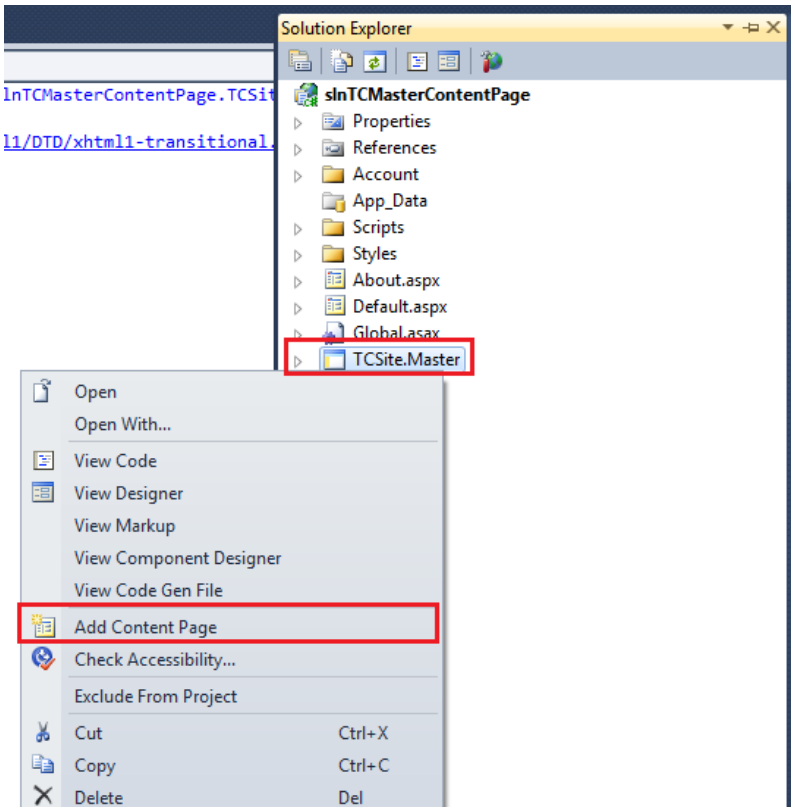


Fig 2.3 Add content page

Step 3: Finally, you can see **final view of Content Page**.

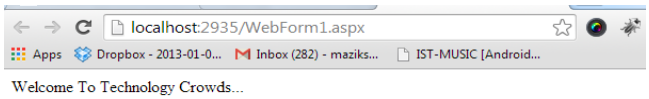


Fig 2.4 Content Page

Check your progress 4

1. Client Page is created using:

   a. Master page

   b. Content page

   c. Both a and b

   d. Neither a nor b

2.6 Adding Elements

Adding Elements in List:

- **AddRange:** AddRange adds an entire collection of elements. It can replace tedious foreach-loops that repeatedly call Add on List. We can pass any IEnumerable collection to AddRange, not just an array or another List.
- **Copy array:** You can create a List having elements from an array by using list constructor which will pass the array. The list receives this parameter and fills its values from it.
- **Test elements:** You can test every element for particular value. It is noted that for every loop, it will check for particular list of parameters.
- **IndexOf:** It shows element index of particular value in a List collection which locates for initial position of the value.

Accessing HTML elements

- With the introduction of .Netframework, there are many new features that gets added to be applied in web development. One of the basic feature is accessing HTML elements programmatically.
- Accessing <html> elements in code: It is found that there appears many new classes in .Netframework which are added in System.Web.UI.HtmlControls Namespace.
- Accessing <Head> tag: You can access <Head> tag through HtmlHead class which is packed in namespace System.Web.UI.HtmlControls. Here the Page object will show way through Header property in order to access <head> element.
- Applying Styles clearly to control: You can create css styles for particular controls by applying it to controls clearly. In ASP.Net controls, ApplyStyle(Style) and MergeStyle(Style) methods are packed with control to get the same.

Check your progress 5

1. The function of AddRange is to:

   a. add entire collection of elements

   b. create List having elements from an array

   c. test element for particular value

   d. show element index of particular value in List

2.7 Building the Site Navigation

To manage a website having with many pages, the direct way for visitors to navigate the website is done through site navigation. Initially, the site's navigational structure should be defined which must be translated into navigable user interface elements. After this, everything needs to be maintained and updated in order to further add new pages to site by removing previous ones. In case of ASP.NET site navigation system, the developer will able to define site map with which it can access information using programmatic API. ASP.NET ships with a site map provider that expects site map data to be stored in an XML file formatted in a particular way. In order to create site map, right-click on project name placed in Solution Explorer by selecting Add New Item and choosing Site Map option. Here, leave the name as Web.sitemap and click on Add button.

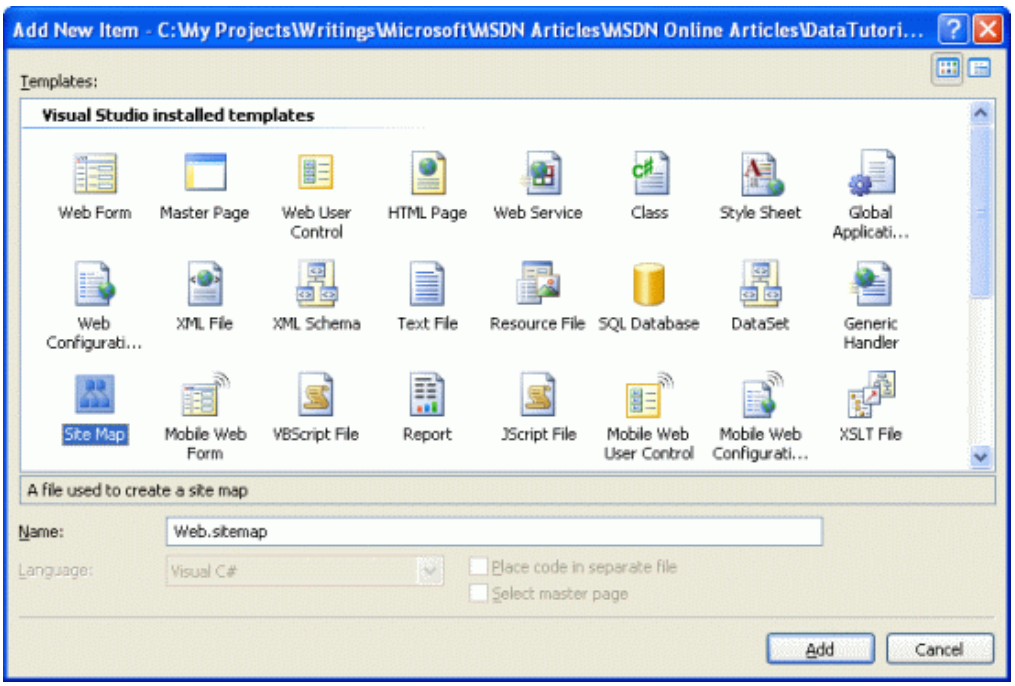


Fig 2.5 Add New Item Page

It is noted that the site map file is an XML file which carries <siteMap> node as its root node having precisely one <siteMapNode> child element. That first <siteMapNode> element can then contain an arbitrary number of descendent <siteMapNode> elements. It is seen that site map defines website's navigational structure in hierarchial structure as shown in fig 2.6 describing many sections of site. In this, all <siteMapNode> element in Web.sitemap shows section in site's navigational structure.

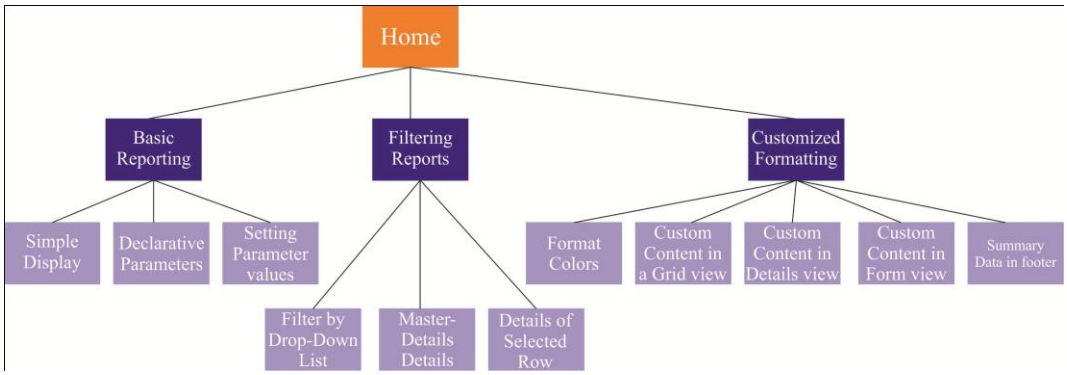


Fig 2.6 hierarchial structure of site map

Check your progress 6

1. Site map file is a form of:

   a. CSS file

   b. HTML file

   c. XML file

   d. None of above

2.8 Adding Authentication

Authentication is the ability to identify a particular entity. The need for authentication occurs when we have some resources that we want to make available to different entities. We store these resources in a centralized place and instruct the system that manages them to prevent entities that we don't recognize from having access. Anonymous authentication refers to a situation in which we grant access to resources to all users, even if we don't know them. There are several ways to add Authentication to an existing project. There are three kinds of authentication in ASP.NET:

- Form
- Windows
- Passport

Form authentication is cookie based, as ASP.NET places a cookie in the client machine in order to track the user. If the user requests a secure page and has not logged in, then ASP.NET redirects him/her to the login page. Once the user is authenticated, he/she will be allowed to access the requested page.

In ASP.NET, you can setup Windows account authentication. It is noted that such type of authentication doesn't involve ASP.NET engine but works at Internet Information Server (IIS) level. It needs correct IIS configuration where authentication types in IIS can be analysed using IIS Manager:

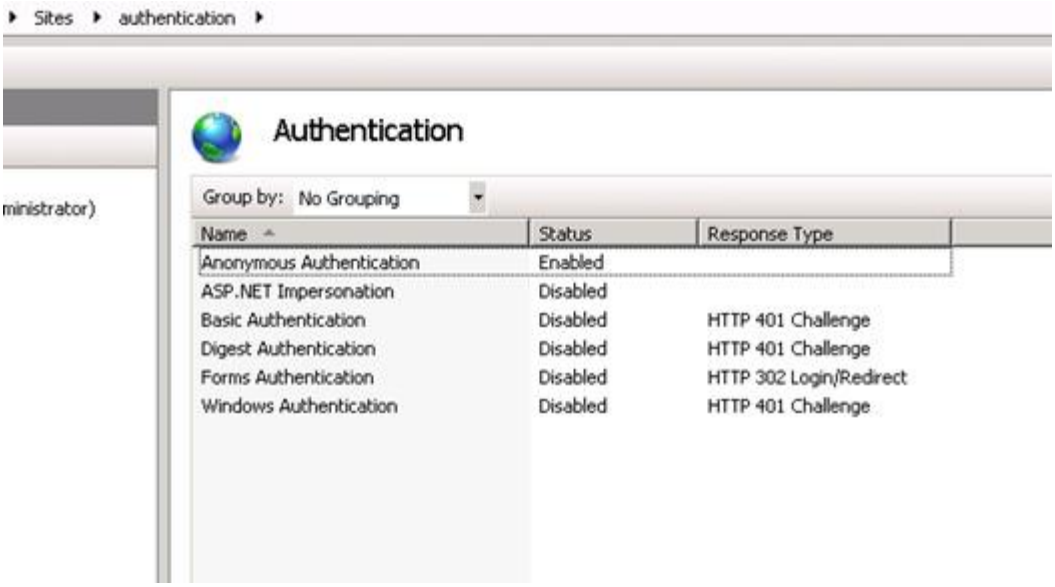


Fig 2.7 IIS manager

In the Window account authentication, here the user requires username and password which are recognized by operating system for particular application. In this, when user calls web page, a dialog box appears which will ask for credentials. If user provides valid credentials for valid Window account, then in such case the authentication gets completed. It is not a secured authentication as in this, the authentication data gets transmitted to server in form of text.

Check your progress 7

1. Form authentication is base on:

   a. temp

   b. cookie

   c. both a and b

   d. neither a nor b

2.9 Adding Content Pages

A content page is an ASP.NET Web page that is associated with a master page. The master page establishes a layout and includes one or more ContentPlaceHolder controls for replaceable text and controls. The content page includes only the text and controls that are merged at run time with the master page's ContentPlaceHolder controls.

After creating a content page, you can create custom content that corresponds to each ContentPlaceHolder on the master page. Alternatively, you can explicitly choose to allow the master page's default content to be displayed. To add a content page in Visual Web Developer:

- In Solution Explorer, right-click on Web site and click Add New Item.
- In Visual Studio in templates list, click on Web Form.
- Select the Select master page check box and click Add.
- Select a Master Page dialog box appears.
- In Contents of Folder box, click master page with which you want to associate with page you create and click on OK.

Check your progress 8

1. ASP web page is associated with:

   a. content page

   b. master page

   c. both a and b

   d. neither a nor b

2.10 Working with Data

In ASP.NET, you can create access and review data from database with the help of ASP.NET Web Forms and Entity Framework Code. Entity Framework is an object-relational mapping framework where you can access with relational data by removing many data access code applied for writing. With Entity Framework, you will get queries by LINQ which retrieve and manipulate data as strongly typed objects. Further this Entity Framework will focus on creating remaining application instead of focusing on data access.

With Entity Framework, the development paradigm known as Code First is framed which defines data models with classes that create own custom types with grouping variables together using types, methods and events. Also, we see that by creating classes showing data which can configure application applied for classes. Further we see that the data is passed between server and client. If data is known at point, then page gets rendered by adding script which forms data or setting json using literal or similar variations.

Check your progress 9

1. queries in Entity Framework is given by:

   a. LINQ

   a. NQIL

   c. ILNQ

   d. QINL

2.11 Using ASP.NET Web Services and WCF

Web Service in ASP.NET

A Web Service is programmable application logic accessible via standard Web protocols. One of these Web protocols is the Simple Object Access Protocol (SOAP). SOAP is a W3C submitted note that uses standards based technologies to encode and transmit application data.

Consumers of a Web Service do not need to know anything about the platform, object model, or programming language used to implement the service; they only need to understand how to send and receive SOAP messages.

WCF Service

Windows Communication Foundation is a framework for building service-oriented applications. Using WCF, you can send data as asynchronous messages from one service endpoint to another. A service endpoint can be part of a continuously available service hosted by IIS, or it can be a service hosted in an application.

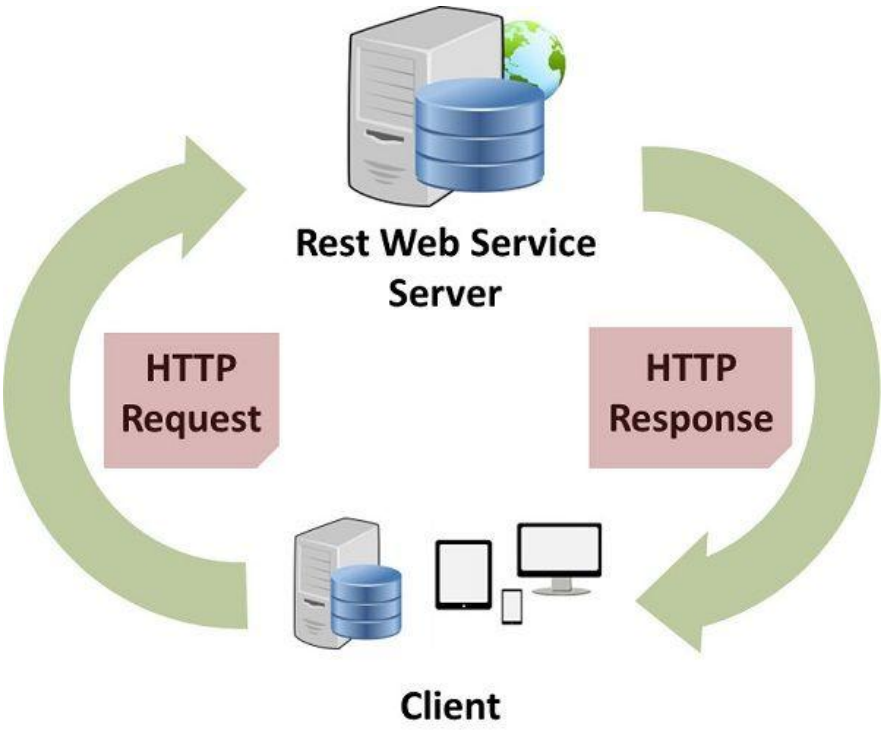


Fig 2.8 WCF arrangement

In fig 2.8, we see that the endpoint is a client of service which requests data from service endpoint. Here the messages can be as simple as single character or word which is sent as XML, or can be complex as stream of binary data. It is noted that WCF can be applied in case when:

- Secure service to process business transactions.
- Service which gives current data to others.
- Chat service allows two people to communicate or exchange data in real time.
- Dashboard application polls having services for data with logical presentation.
- Opening workflow using Windows Workflow Foundation as a part of service.
- Silverlight application to poll a service for latest data feeds.

Features of WCF

- Service Orientation
- Interoperability
- Multiple Message Patterns
- Service Metadata
- Data Contracts
- Security
- Multiple Transports and Encodings
- Reliable and Queued Messages
- Durable Messages
- Transactions
- AJAX and REST Support
- Extensibility

Comparison Chart

WCF	ASP.NET Web Services

[ServiceContract] and [OperationContract] attributes defines web service and methods.
 	
`[ServiceContract]`
`public interface ITest`
`{`
`  [OperationContract]`
`  string ShowMessage();`
`}`
`public class Service : ITest`
`{`
`  public string ShowMessage()`
`  {`
`  return "Hello World!";`
`  }`
`}`
	[WebService] and [WebMethod] attributes defines web service and methods.
	
	`[WebService]`
	`public class Service :`
	`System.Web.Services.WebService`
	`{`
	`  [WebMethod]`
	`  public string Test()`
	`  {`
	`  return "Hello World!";`
	`  }`
	`}`

Hosted in IIS, WAS (Windows Activation Service) Self-hosting, Windows service.	Hosted in IIS.
Accessed through HTTP, TCP, MSMQ, P2P, Named pipes.	Accessed through HTTP.
Supports security, reliable messaging, transactions, durable messages, service orientation, interoperability, service metadata, AJAX and REST support, extensibility.	Supports Security services.
Uses the ServiceMetadata tool (svcutil.exe) to generate the client for the service.	Uses the command-line tool WSDL.EXE to generate the client for the service.
Unhandled exceptions are not returned to clients as SOAP faults. A configuration setting is provided to have the unhandled exceptions returned to clients for the purpose of debugging.	Unhandled exceptions are returned to the client as SOAP faults.
The generated WSDL can customized by using ServiceMetadataBehavior class.	The generated WSDL can customized by using ServiceDescriptionFormatExtension class.

System.RunTime.Serialization is supported. -Better performance. -DataContractAttribute and DataMemberAttribute can be added to .NET Framework types to indicate that instances of the type are to be serialized into XML, and which particular fields or properties of the type are to be serialized. -Classes that implement the IDictionary interface can be serialized. -Hash table can be serialized.	System.XML.Serialization is supported -Worse performance. -Only Public fields or Properties of .NET types can be translated into XML. -Only the classes which implement IEnumerable and ICollection interface can be serialized. -Hash table can not be serialized.
Can be multithreaded via ServiceBehavior class.	Can not be multithreaded.
Supports different type of bindings like BasicHttpBinding, WSHttpBinding, WSDualHttpBinding etc.	Only used SOAP or XML for this.

Check your progress 10

1. SOAP is:

   a. Simple Object Access Protocol

   b. Sample Object Access Protocol

   c. Simple Oriented Access Protocol

   d. Simple Object Access Process

2.12 Creating a simple ASP.NET Web Service

In order to understand the concept of Web service, create a web service of Equity stock price information. In this, we see that clients will query about name and price of stock as per stock symbol. We see that web service has three methods:

- HelloWorld method
- GetName Method
- GetPrice Method

We will consider following steps to create web service:

Step (1): Select File -> New -> Web Site in Visual Studio, and then select ASP.NET Web Service.

Step (2): A web service file called Service.asmx and its code behind file, Service.cs is created in the App_Code directory of the project.

Step (3): Change the names of the files to StockService.asmx and StockService.cs.

Step (4): The .asmx file has simply a WebService directive on it:

<%@ WebService Language="C#" CodeBehind="~/App_Code/StockService.cs" Class="StockService" %>

Step (5): Open StockService.cs file, the code generated in it is the basic Hello World service. The default web service code behind file looks like:

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml.Linq;
namespace StockService
{
  // <summary>
  // Summary description for Service1
  // <summary>
  [WebService(Namespace = "http://tempuri.org/")]
  [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
  [ToolboxItem(false)]
  // To allow this Web Service to be called from script,
  // using ASP.NET AJAX, uncomment the following line.
  // [System.Web.Script.Services.ScriptService]
  public class Service1 : System.Web.Services.WebService
```

```
  {
    [WebMethod]
    public string HelloWorld()
    {
      return "Hello World";
    }
  }
}
```

Step (6) : Change the code behind file to add the two dimensional array of strings for stock symbol, name and price and two web methods for getting the stock information.

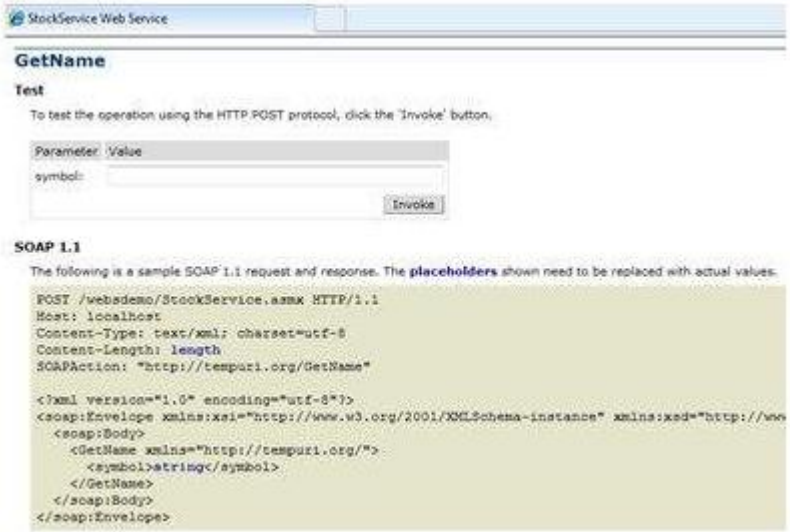
```
using System;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml.Linq;
[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
// To allow this Web Service to be called from script,
// using ASP.NET AJAX, uncomment the following line.
// [System.Web.Script.Services.ScriptService]

public class StockService : System.Web.Services.WebService
{
  public StockService () {
    //Uncomment the following if using designed components
    //InitializeComponent();
  }
```

```
string[,] stocks =
{
  {"RELIND", "Reliance Industries", "1060.15"},
  {"ICICI", "ICICI Bank", "911.55"},
  {"JSW", "JSW Steel", "1201.25"},
  {"WIPRO", "Wipro Limited", "1194.65"},
  {"SATYAM", "Satyam Computers", "91.10"}
};
[WebMethod]
public string HelloWorld() {
  return "Hello World";
}
[WebMethod]
public double GetPrice(string symbol)
{
  //it takes the symbol as parameter and returns price
  for (int i = 0; i < stocks.GetLength(0); i++)
  {
    if (String.Compare(symbol, stocks[i, 0], true) == 0)
    return Convert.ToDouble(stocks[i, 2]);
  }
  return 0;
}
[WebMethod]
public string GetName(string symbol)
{
  // It takes the symbol as parameter and
  // returns name of the stock
  for (int i = 0; i < stocks.GetLength(0); i++)
```

```
        {
            if (String.Compare(symbol, stocks[i, 0], true) == 0)
            return stocks[i, 1];
        }
        return "Stock Not Found";
    }
}
```

Step (7) : On running web service application, we see that web service will have test page that allow testing service methods.

Step (8): Click on a method name, and check whether it runs properly.



Step (9): In this, we see that in order to test GetName method, provide one of the stock symbols, that returns name of stock.

Check your progress 11

1. web service can be implemented using:

   a. HelloWorld method

   b. GetName Method

   c. GetPrice Method

   d. All of above

2.13 Let Us Sum Up

While studying this unit, we have learnt that C# is an elegant and type-safe object-oriented language that enables developers to build a variety of secure and robust applications that run on the .NET Framework. In ASP.NET, login controls delivers robust login solution for ASP.NET Web applications without using programming which forms authentication to help automate user authentication for a Web site.

We have seen that Cascading Style Sheets is a style sheet language used for describing the presentation semantics of a document written in a markup language. The master page and content page work together to produce output to client. Sometimes, interaction between master page and content page is needed. To manage a website having with many pages, the direct way for visitors to navigate the website is done through site navigation. Initially, the site's navigational structure should be defined which must be translated into navigable user interface elements.

It is known that authentication is ability to find particular entity which occurs when we have some resources that we want to make available to different entities. A content page is an ASP.NET Web page is associated with master page which establishes a layout and includes one or more ContentPlaceHolder controls for replaceable text and controls. A Web Service is programmable application logic accessible via standard Web protocols. One of these Web protocols is the Simple Object Access Protocol (SOAP). SOAP is a W3C submitted note that uses standards based technologies to encode and transmit application data.

2.14 Answers for Check Your Progress

Check your progress 1

Answers: (1-b)

Check your progress 2

Answers: (1-b)

Check your progress 3

Answers: (1-d)

Check your progress 4

Answers: (1-c)

Check your progress 5

Answers: (1-a)

Check your progress 6

Answers: (1-c)

Check your progress 7

Answers: (1-b)

Check your progress 8

Answers: (1-c)

Check your progress 9

Answers: (1-a)

Check your progress 10

Answers: (1-a)

Check your progress 11

Answers: (1-d)

2.15 Glossary

1. **ASP.NET -** A set of .NET classes used to create Web-based, client-side (Web Form) and server-side (Web Service) applications.
2. **Client -** Any application that requests information or services from a server.
3. **Code Access Security -** The common language runtime's security model for applications.
4. **Web.config -** Application configuration files contain settings specific to an application.
5. **Authorization -** It shows identity to be granted to request type in order to access given resources.
6. **Authentication -** It discovers and verify principal identity, by examining user's credentials and validating credentials against authority.

2.16 Assignment

Discuss the steps involved in adding authentication and content pages.

2.17 Activities

Collect some information on ASP.NET Objects.

2.18 Case Study

Generalised the basic difference between ASP and ASP.NET.

2.19 Further Readings

1. M. James, "Overview of ASP.NET and Web Forms", Wiley.
2. Rumy, "Code Behind vs. Code Inline"--.NET Framework, Oxford.

Block Summary

In this block, students have learnt and understand about the basic of Common Type System and Common Language Specification. The block gives an idea on the study and concept of Security Manager with various functions and characteristics. You have been well explained on the concepts of Managed Code along with its features and usage.

The block detailed about the basic of building StyleSheet as well as steps involved in adding elements. The concept related to adding Authentication and adding Content Pages are well explained to you. You will be demonstrated about ASP.NET WebServices and WCF.

Block Assignment

Short Answer Questions

1. What is ASP.NET?
2. What are steps involved in building Site Navigation?
3. What are various MS.NET Namespaces?
4. What are the features of Common Language Runtime?

Long Answer Questions

1. Explain characteristics of MS.NET Memory Management.
2. Write short note on creating Content Master.
3. What is the benefit of ASP.NET?

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

……

BLOCK 4: INTRODUCTION TO C#

Block Introduction

The .NET framework provides a set of base class libraries which provide functions and features which can be used with any programming language which implements .NET, such as Visual Basic, C#, Visual C++, etc. Namespaces are the way to organize .NET Framework Class Library into a logical grouping according to their functionality, usability as well as category they should belong to, or we can say Namespaces are logical grouping of types for the purpose of identification.

In this block, we will detail about the basic of .NET Framework with various versions. The block will focus on types of MS.NET applications, Base Class Library and MS.NET Namespaces. The students will give an idea on DMA Common Language Runtime and MS.NET Memory Management/Garbage Collection.

In this block, you will made to learn and understand about Working with Standard Controls along with ASP.NET Objects. The concept related to Creating the Content Master and building Site Navigation will also be explained to you. You will be demonstrated practically about working with Data technique.

Block Objective

After learning this block, you will be able to understand:

- About MS.NET Base Class Library
- Features of MS.NET Namespaces
- Idea about Metadata and PE files
- Concept of Common Language Runtime (CLR)
- Concept of MS.NET Memory Management / Garbage Collection
- Features of building Site Navigation
- Steps to add Authentication and Content Pages
- Working characteristics of Data
- Understanding about ASP.NET WebServices and WCF

Block Structure

Unit 1: Introduction to .NET

Unit 2: ASP.Net and C#

UNIT 1: INTRODUCTION TO .NET

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- About MS.NET Namespaces
- About MSIL / Metadata and PE files
- About Managed Code
- About Common Type System (CTS)

1.1 Introduction

.Net is a web development platform, which provides a programming model, a comprehensive software infrastructure and various services required to build up robust web application for PC, as well as mobile devices.

ASP.Net works on top of the HTTP protocol and uses the HTTP commands and policies to set a browser-to-server two-way communication and cooperation. ASP.NET is an open source server-side Web application framework designed for Web development to produce dynamic Web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, web applications and web services.

It was first released in January 2002 with version 1.0 of the .NET Framework, and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language. The ASP.NET SOAP extension framework allows ASP.NET components to process SOAP messages.

1.2 The .NET Framework: an Overview

C# programs run on the .NET Framework, an integral component of Windows that includes a virtual execution system called the common language runtime (CLR) and a unified set of class libraries. The CLR is the commercial implementation by Microsoft of the common language infrastructure (CLI), an international standard that is the basis for creating execution and development environments in which languages and libraries work together seamlessly.

Source code written in C# is compiled into an intermediate language (IL) that conforms to the CLI specification. The IL code and resources, such as bitmaps and strings, are stored on disk in an executable file called an assembly, typically with an extension of .exe or .dll. An assembly contains a manifest that provides information about the assembly's types, version, culture, and security requirements.

When the C# program is executed, the assembly is loaded into the CLR, which might take various actions based on the information in the manifest. Then, if the security requirements are met, the CLR performs just in time (JIT) compilation to convert the IL code to native machine instructions. The CLR also provides other services related to automatic garbage collection, exception handling, and resource management. Code that is executed by the CLR is sometimes referred to as "managed code," in contrast to "unmanaged code" which is compiled into native machine language that targets a specific system. The following diagram illustrates the compile-time and run-time relationships of C# source code files, the .NET Framework class libraries, assemblies, and the CLR.

Check your progress 1

1. C# programs run on _____

   a. VB framework

   b. .NET Framework

   c. VC framework

   d. all of above

1.3 Framework Components

To better understand the CLR, consider how compilers that target the .NET Framework differ from traditional compilers.

Traditional compilers target a specific processor, consuming source files in a specific language, and producing binary files containing streams of instructions in the native language of the target processor. These binary files may then be executed directly on the target processor.

.NET compilers function a little differently, as they do not target a specific native processor. Instead, they consume source files and produce binary files containing an intermediate representation of the source constructs, expressed as a combination of metadata and Common Intermediate Language (CIL). In order for these binaries to be executed, the CLR must be present on the target machine.

When these binaries are executed they cause the CLR to load. The CLR then takes over and manages execution, providing a range of services such as JIT compilation (converting the CIL as needed into the correct stream of instructions for the underlying processor), memory management (in the form of a garbage collector), exception management, debugger and profiler integration, and security services.

This compilation and execution model explains why C# is referred to as a managed language, why code running in the CLR is referred to as managed code, and why the CLR is said to provide managed execution.

Although this dependency on a runtime environment might initially appear to be a drawback, substantial benefits arise from this architecture. Since the metadata and CIL representations are processor architecture-neutral, binaries may be used on any machine in which the Common Language Runtime is present, regardless of underlying processor architecture. Additionally, since processor-specific code generation is deferred until runtime, the CLR has the opportunity to perform processor-specific optimizations based on the target architecture the code is running on. As processor technology advances, all applications need to take advantage of these advances is an updated version of the CLR.

Unlike traditional binary representations, which are primarily streams of native processor instructions, the combination of metadata and CIL retains almost all of the original source language constructs. In addition, this representation is source language-neutral, which allows developers to build applications using multiple source languages. They can select the best language for a particular task, rather than being forced to standardize on a particular source language for each application or needing to rely on component technologies, such as COM or CORBA, to mask the differences between the source languages used to build the separate components of an application.

Check your progress 2

1. CLR is known as:

a. Common Language Runtime

b. Common Loading Runtime

c. Clear Language Runtime

d. Clear Loading Runtime

1.4 Framework Versions

Each version of the .NET Framework contains the common language runtime (CLR), the base class libraries, and other managed libraries. Each new version of the .NET Framework retains features from the previous versions and adds new features. The CLR is identified by its own version number. The .NET Framework version number is incremented at each release, although the CLR version is not always incremented. For example, the .NET Framework 4, 4.5, and later releases include CLR 4, but the .NET Framework 2.0, 3.0, and 3.5 include CLR 2.0. (There was no version 3 of the CLR.)

The following table summarizes .NET Framework version history and correlates each version with Visual Studio, Windows, and Windows Server.

.NET Framework version	CLR version	Features
Net 4.6.1	4	Support X509 having ECDSA Encrypt support for hardware keys in ADO.NET Spell checking
.NET 4.6	4	Compile using .NET Native

		ASP.NET Core 5 Event tracing quality Supports page encodings
4.5.2	4	New APIs for transactional systems and ASP.NET System DPI resizing in Windows Profile improvements ETW and stress logging improvements
4.5.1	4	Support for Windows Phone Store apps Automatic binding redirection Performance and debugging improvements
4.5	4	Support Windows Store apps WPF, WCF, WF, ASP.NET updates
4	4	Expanded base class libraries

		Cross-platform development with Portable Class Library MEF, DLR, code contracts
3.5	2.0	AJAX-enabled websites LINQ Dynamic data
3.0	2.0	WPF, WCF, WF, CardSpace
2.0	2.0	Generics ASP.NET additions
1.1	1.1	ASP.NET and ADO.NET updates Side-by-side execution
1.0	1.0	First version of the .NET Framework.

Check your progress 3

1. Which .NET version uses ASP.NET Core 5?

   a. NET 4.6.1

   b. NET 4.6

   c. NET 4

   d. NET 3.5

2. Which version carries side by side execution?

   a. 3.0

   b. 2.0

   c. 1.1

   d. 1.0

1.5 Types of Applications using MS.NET

All types of .NET applications use one or more .NET compliant languages for their design and development. The .NET Framework includes various technologies, such as ASP.NET, VB.NET, VC++.NET, and ADO.NET. Here is a more or less full list of various types of application that we can develop on .NET.

ASP.Net Web applications:

These are the programs that used to run inside some web server to fulfill the user requests over the http. ASP.NET Web applications can range from simple Web sites that consist of HTML pages to advanced enterprise applications that run on local and remote networks. These enterprise applications also provide components for exchanging data using XML. This type includes dynamic and data driven browser based applications. (Ex: Hotmail and Google).

Web services:

There are certain web services that are functionally available with certain industrial standards like HTTP, XML and SOAP.

Windows applications:

These are types of Windows desktop applications for every day working which can be MS word. Run only under Windows environment. These applications consume the services provided by the Windows operating system.

Windows services:

These are long-running executable applications which runs on the system as a background process. These applications do not interfere with the working of the other processes that run on the same computer. Windows services execute within separate Windows sessions created specifically for each Windows service. These services do not have a graphic user interface and are ideal for running on the server. Windows services were earlier called NT services.

Console applications:

These are light weight programs run inside the command prompt (DOS) window. They are commonly used for test applications.

Mobile applications:

They are applications which work on multiple mobile devices and shows ubiquitous access to data from mobile devices. The .NET Framework automatically makes changes to these applications to enable them to run on multiple browsers, depending on the mobile device.

Class libraries:

These are components that you create once and reuse a number of times in multiple applications. Class libraries allow you to define several classes, along with their methods and interfaces, in one file. These libraries compile to .dll files and facilitate rapid development of new applications because of reusability of code. To access the functionality of the classes in a class library from your application, you need to include a reference to that library in your program.

Check your progress 4

1.The change any Mobile application in .NET framework works on:

a. single browser

b. two browsers

c. multiple browser

d. all of above

1.6 MS.NET Base Class Library

The .NET framework provides a set of base class libraries which provide functions and features which can be used with any programming language which implements .NET, such as Visual Basic, C#, Visual C++, etc.

The base class library contains standard programming features such as Collections, XML, DataType definitions, IO, Reflection and Globalization to name a few. All of which are contained in the System namespace. As well, it contain some non-standard features such as LINQ, ADO.NET, drawing capabilities, forms and web support. The below table provides a list each class of the base class library and a brief description of what they provide.

Base Class Library Namespace	**Brief Description**
System	Contains the fundamentals for programming such as the data types, console, match and arrays, etc.
System.CodeDom	Supports the creation of code at runtime and the ability to run it.
System.Collections	Contains Lists, stacks, hashtables and dictionaries
System.ComponentModel	Provides licensing, controls and type conversion capabilities
System.Configuration	Used for reading and writing program configuration data
System.Data	Is the namespace for ADO.NET
System.Deployment	Upgrading capabilities via ClickOnce
System.Diagnostics	Provides tracing, logging, performance counters, etc. functionality

System.DirectoryServices	Is the namespace used to access the Active Directory
System.Drawing	Contains the GDI+ functionality for graphics support
System.EnterpriseServices	Used when working with COM+ from .NET
System.Globalization	Supports the localization of custom programs
System.IO	Provides connection to file system and the reading and writing to data streams such as files
System.Linq	Interface to LINQ providers and the execution of LINQ queries
System.Linq.Expressions	Namespace which contains delegates and lambda expressions
System.Management	Provides access to system information such as CPU utilization, storage space, etc.
System.Media	Contains methods to play sounds
System.Messaging	Used when message queues are required within an application, superseded by WCF
System.Net	Provides access to network protocols such as SSL, HTTP, SMTP and FTP
System.Reflection	Ability to read, create and invoke class information.
System.Resources	Used when localizing a program in relation to language support on web or form controls
System.Runtime	Contains functionality which allows the management of runtime behavior.

System.Security	Provides hashing and the ability to create custom security systems using policies and permissions.
System.ServiceProcess	Used when a windows service is required
System.Text	Provides the StringBuilder class, plus regular expression capabilities
System.Threading	Contains methods to manage the creation, synchronization and pooling of program threads
System.Timers	Provides the ability to raise events or take an action within a given timer period.
System.Transactions	Contains methods for the management of transactions
System.Web	Namespace for ASP.NET capabilities such as Web Services and browser communication.
System.Windows.Forms	Namespace containing the interface into the Windows API for the creation of Windows Forms programs.
System.Xml	Provides the methods for reading, writing, searching and changing XML documents and entities.

Check your progress 5

1. Which Base Class Library supports creation of code at runtime?

   a. System.CodeDom

   b. System.Collections

   c. System.ComponentModel

   d. System.Configuration

2. The function of Base Class Library System.DirectoryServices is:

   a. access Active Directory

   b. function for graphics support

   c. applied in COM+ from .NET

   d. supporting localization of custom programs

1.7 MS.NET Namespaces

Namespaces are the way to organize .NET Framework Class Library into a logical grouping according to their functionality, usability as well as category they should belong to, or we can say Namespaces are logical grouping of types for the purpose of identification.

The .NET Framework Class Library (FCL) is a large collection of thousands of Classes. These Classes are organized in a hierarchical tree. The System Namespaces is the root for types in the .NET Framework. We can uniquely identify any Class in the .NET Framework Class Library (FCL) by using the full Namespaces of the class .In .Net languages every program is created with a default Namespaces. Programmers can also create their own Namespaces in .Net languages.

Please now have a look at the example of declaring some namespace:

```
using System;

namespace OutNamespace

{

namespace WorkNamespace

{ /// can be placed some classes, structures etc

}

}
```

In this example we create two namespaces. These namespaces have hierarchical structure. We have some outer one named OutNamespace and the inner one called WorkNamespace. The inner namespace is declared with a C# .Net class WorkItem.

Check your progress 6

1.Name space are associated with:

a. items

b. class

c. work

d. all of above

1.8 MSIL / Metadata and PE files

Microsoft Intermediate Language (MSIL)

A.NET programming language (C#, VB.NET, J# etc.) does not compile into executable code; instead it compiles into an intermediate code called Microsoft Intermediate Language (MSIL). As a programmer one need not worry about the syntax of MSIL - since our source code in automatically converted to MSIL. The MSIL code is then send to the CLR (Common Language Runtime) that converts the code to machine language, which is, then run on the host machine. MSIL is similar to Java Byte code. MSIL is the CPU-independent instruction set into which .NET Framework programs are compiled. It contains instructions
for loading, storing, initializing, and calling methods on objects. Combined with metadata and the common type system, MSIL allows for true cross- language

integration Prior to execution, MSIL is converted to machine code. It is not interpreted.

.

When .net programming code is compiled ,it is not convert into exe files and it is convert into Microsoft intermediate languages, and after that clr convert the this msil code into machine level language.

Microsoft Intermediate Language (MSIL) is a CPU-independent set of instructions that can be efficiently converted to the native code. During the runtime the Common Language Runtime (CLR)'s Just In Time (JIT) compiler converts the Microsoft Intermediate Language (MSIL) code into native code to the Operating System.
When a compiler produces Microsoft Intermediate Language (MSIL), it also produces Metadata. The Microsoft Intermediate Language (MSIL) and Metadata are contained in a portable executable (PE) file . Microsoft Intermediate Language (MSIL) includes instructions for loading, storing, initializing, and calling methods on objects, as well as instructions for arithmetic and logical operations, control flow, direct memory access, exception handling, and other operations
Just In Time Compiler
.

Microsoft .Net Metadata

Metadata in .Net is binary information which describes the characteristics of a resource . This information include Description of the Assembly , Data Types and members with their declarations and implementations, references to other types and members , Security permissions etc. A module's metadata contains everything that needed to interact with another module.

During the compile time Metadata created with Microsoft Intermediate Language (MSIL) and stored in a file called a Manifest . Both Metadata and Microsoft Intermediate Language (MSIL) together wrapped in a Portable Executable (PE) file. During the runtime of a program Just In Time (JIT) compiler of the Common Language Runtime (CLR) uses the Metadata and converts Microsoft Intermediate Language (MSIL) into native code. When code is executed, the runtime loads metadata into memory and references it to discover information about your code's classes, members, inheritance, and so on. Moreover Metadata eliminating the need for Interface Definition Language (IDL) files, header files, or any external method of component reference.

Portable Executable (PE) File Format

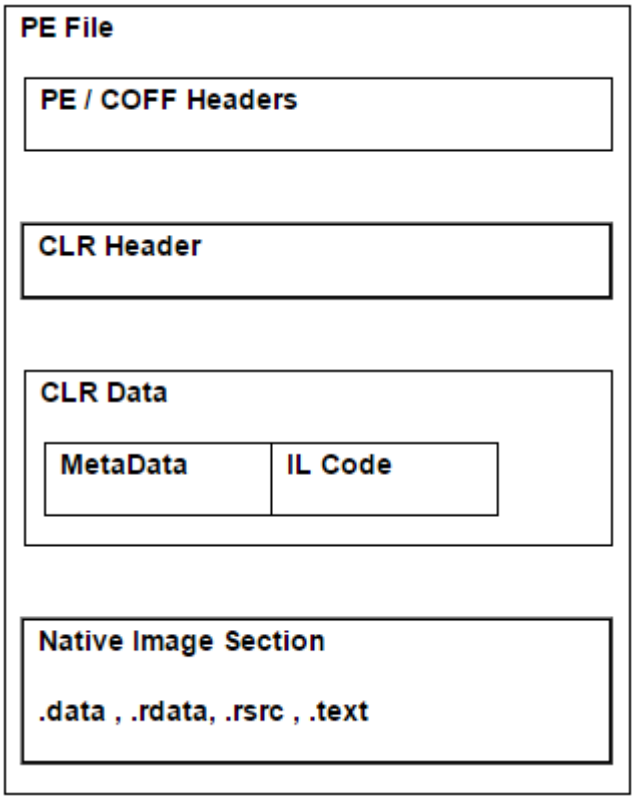
The Portable Executable (PE) format is a file format for executables, object code, and DLLs, used in 32-bit and 64-bit versions of Windows operating systems. The PE file format was defined to provide the best way for the Windows

Operating System to execute code and also to store the essential data which is needed to run a program. Portable Executable File Format is derived from the Microsoft Common Object File Format (COFF). Almost every Windows executable, DLL or EXE, is a Portable Executable (PE) format file. Although there is little in the PE format that lends itself to .NET, in the current implementation of .NET all assemblies are contained in special PE format files, which have some traditional bits left out and quite a lot of new bits put in.

Very generally, a PE file consists of a PE header, which contains a list of Data Directory entries, and a number of Sections which are defined just after the PE header. Not all the Data Directories have meaning in a .NET file, and not many Sections are present either. Nevertheless, those that remain are still important -- in particular, the last Data Directory entry points to the start of .NET information.

The real starting point of a PE file, from the .NET point of view, is the COR20 Header, which tells the .NET runtime where to find the metadata. The COR20 header, like the PE header, specifies some Data Directories, as well as the entry point for the assembly. Most of these Data Directories point to things like fix up information which is not useful for examining the assembly, but one of them points to the start of the Metadata Streams.

Format of .Net PE file



Check your progress 7

1.Which is not a .NET programming language:

a. Oracle

b. C#,

c. VB.NET

d. J#

1.9 Common Language Runtime (CLR)

The Common Language Runtime (CLR) is an Execution Environment which is the backbone of the .NET framework. It works as a layer between Operating Systems and the applications written in .Net languages that conforms to the Common Language Specification (CLS). The main function of Common Language Runtime (CLR) is to convert the Managed Code into native code and then execute the Program. The Managed Code compiled only when it needed, that is it converts the appropriate instructions when each function is called. The Common Language Runtime (CLR) 's Just In Time (JIT) compilation converts Intermediate Language (MSIL) to native code on demand at application run time.

CLR handles the execution of code and provides useful services for the implementation of the program. In addition to executing code, CLR provides services such as memory management, thread management, security management, code verification, compilation, and other system services.

During the execution of the program, the Common Language Runtime (CLR) manages memory, Thread execution, Garbage Collection (GC) , Exception Handling, Common Type System (CTS), code safety verifications, and other system services. The CLR (Common Language Runtime) defines the Common Type System (CTS), which is a standard type system used by all .Net languages. That means all .NET programming languages uses the same representation for common Data Types, so Common Language Runtime (CLR) is a language-independent runtime environment. The Common Language Runtime (CLR) environment is also referred to as a managed environment, because during the execution of a program it also controls the interaction with the Operating System.

Check your progress 8

1. Common Language Runtime converts:

   a. Managed Code into native code

   b. Native code into managed code

   c. neither a nor b

   d. both a and b

1.10 Managed Code

Managed code is the code that is written to target the services of the managed runtime execution environment such as Common Language Runtime in .Net Technology.

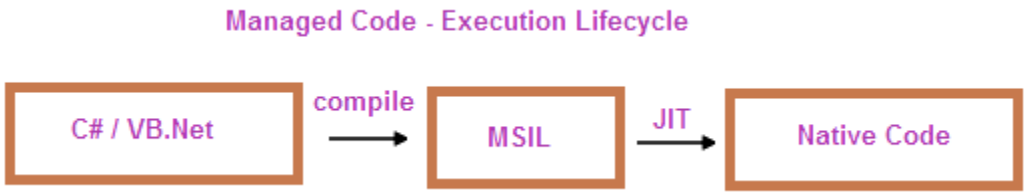


Fig 1.1 Managed Code

The Managed Code running under a Common Language Runtime cannot be accessed outside the runtime environment as well as cannot call directly from outside the runtime environment. It refers to a contract of cooperation between natively executing code and the runtime. It offers services like garbage collection, run-time type checking, reference checking etc. By using managed code you can avoid many typical programming mistakes that lead to security holes and unstable applications, also, many unproductive programming tasks are automatically taken care of, such as type safety checking, memory management, destruction of unused Objects etc.

Check your progress 9

1. In Managed code execution cycle, compile data is send to:

   a. MSIL

   b. Native code

   c. both a and b

   d. neither a nor b

1.11 MS.NET Memory Management / Garbage Collection

In .NET, memory is managed through the use of Managed Heaps. Generally in case of other languages, memory is managed through the Operating System directly. The program is allocated with some specific amount of memory for its use from the Raw memory allocated by the Operating system and then used up by the program. In case of .NET environment, the memory is managed through the CLR (Common Language Runtime) directly and hence we call .NET memory management as Managed Memory Management. .Net manages memory automatically

- Creates objects into memory blocks(heaps)
- Destroy objects no longer in use

Allocates objects onto one of two heaps

- Small object heap(SOH) – objects < 85k
- Large object heap(LOH) – objects >= 85k

You allocate onto the heap whenever you use the "new" keyword in code. In .Net, CLR has garbage collector that executes as a part of our program and responsible for reclaiming the memory of no longer used objects. Garbage collectors free the memory for objects that are no longer referenced and keep the memory for future allocations. The advantage of Garbage Collector is:

- Allow us to develop an application without having worry to free memory.
- Allocates memory for objects efficiently on the managed heap.
- Reclaims the memory for no longer used objects and keeps the free memory for future allocations.

- Provides memory safety by making sure that an object cannot use the content of another object.

The managed heap is a series of allocated memory segments (approx 16Mb in size each) to store and manage objects. The memory for newly created object is allocated at the next available location on the managed heap. If there is available free memory, the garbage collector doesn't search the dead objects for memory reclaim and memory allocations have been done very fast. If the memory is insufficient to create the object, the garbage collector search the dead objects for memory reclaim for the newly object.

Garbage collector determines whether any object in the heap is dead or not being used by the application. If such objects exist then memory used by these objects can be reclaimed. Each and every application has a set of roots and these identify the storage locations for the objects on the managed heap. In Garbage Collector:

- All objects in the heap are allocated from one contiguous range of memory address and heap is divided into generations so that it is easy to eliminate the garbage objects by looking at only a small fraction of the heap.
- Gen 0 and Gen 1 occupy a single segment known as the ephemeral segment. Gen 2 is a set of further segments and the large object heap is yet another group of segments.
- Almost, all objects with-in a generation are of the same age.
- The newest objects are created at higher memory address while oldest memory objects are at lowest memory address with in the heap.
- The allocation pointer for the new objects marks the boundary between the allocated and free memory.
- Periodically the heap is compacted by removing the dead objects and sliding up the live objects towards the lower memory address end of the heap as shown in above fig.
- The order of objects (after memory reclaims) in memory remains the same as they were created.
- There are never any gaps among the objects in the heap.

Check your progress 10

1. The approximate size of allocated memory segments in managing heap so as to store and manage objects will be:

   a. 16Mb

   b. 32Mb

   c. 64Mb

   d. 128Mb

1.12 Common Type System (CTS)

Common Type System (CTS) describes a set of types that can be used in different .Net languages in common. That is, the Common Type System (CTS) ensure that objects written in different .Net languages can interact with each other. For Communicating between programs written in any .NET complaint language, the types have to be compatible on the basic level.

These types can be Value Types or Reference Types. The Value Types are passed by values and stored in the stack. The Reference Types are passed by references and stored in the heap. Common Type System (CTS) provides base set of Data Types which is responsible for cross language integration. The Common Language Runtime (CLR) can load and execute the source code written in any .Net language, only if the type is described in the Common Type System (CTS).

Importance of CTS:

- CTS are responsible for cross language Integration and Type Safety.
- Enforce a set of rules that a programming language must follow.

Check your progress 11

1. The Common Type System can be:

   a. Value Type

   b. Reference Type

   c. Both a and b

   d. Neither a nor b

1.13 Common Language Specification (CLS)

Common Language Specification (CLS) is a set of basic language features that .Net Languages needed to develop Applications and Services, which are compatible with the .Net Framework. When there is a situation to communicate Objects written in different .Net Complaint languages, those objects must expose the features that are common to all the languages. Common Language Specification (CLS) ensures complete interoperability among applications, regardless of the language used to create the application.

Common Language Specification (CLS) defines a subset of Common Type System (CTS). Common Type System (CTS) describes a set of types that can use different .Net languages have in common, which ensure that objects written in different languages can interact with each other. Most of the members defined by types in the .NET Framework Class Library (FCL) are Common Language Specification (CLS) compliant Types.

Check your progress 12

1. Common Language Specification develops:

   a. application

   b. services

   c. both a and b

   d. neither a nor b

1.14 Types of JIT Compilers

JIT stands for just-in-time compiler. It converts the MSIL code to CPU native code as it is needed during code execution. It is called just-in-time since it converts the MSIL code to CPU native code; when it is required within code execution otherwise it will not do anything with that MSIL code.

Different Types of JIT

Normal JIT

This complies only those methods that are called at runtime. These methods are compiled only first time when they are called, and then they are stored in memory cache. This memory cache is commonly called as JITTED. When the

same methods are called again, the complied code from cache is used for execution.

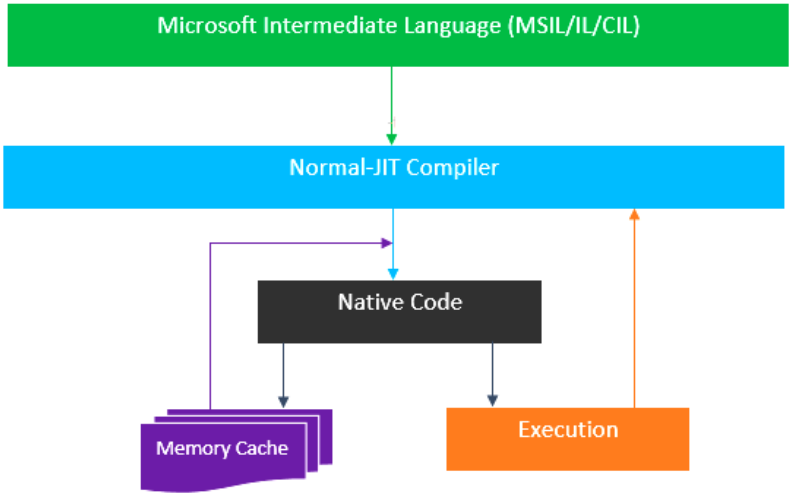


Fig 1.2 Normal JIT Layout

Econo JIT

This complies only those methods that are called at runtime and removes them from memory after execution.

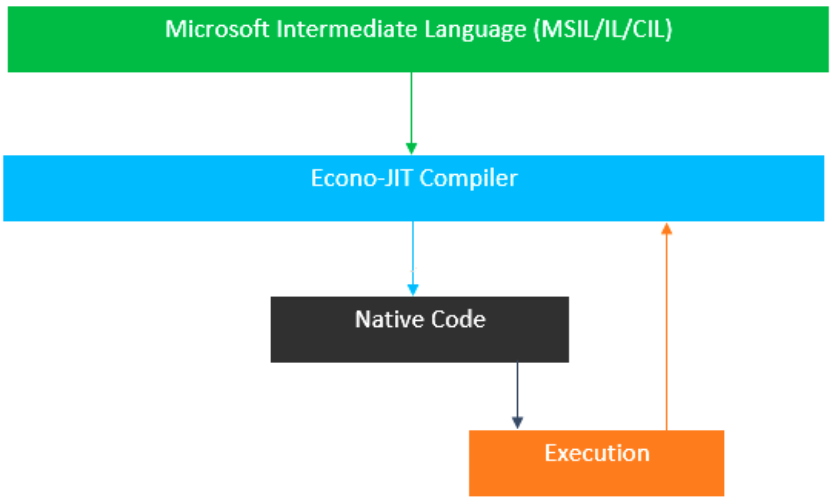


Fig 1.3 Econo JIT

Pre JIT

This complies entire MSIL code into native code in a single compilation cycle. This is done at the time of deployment of the application.

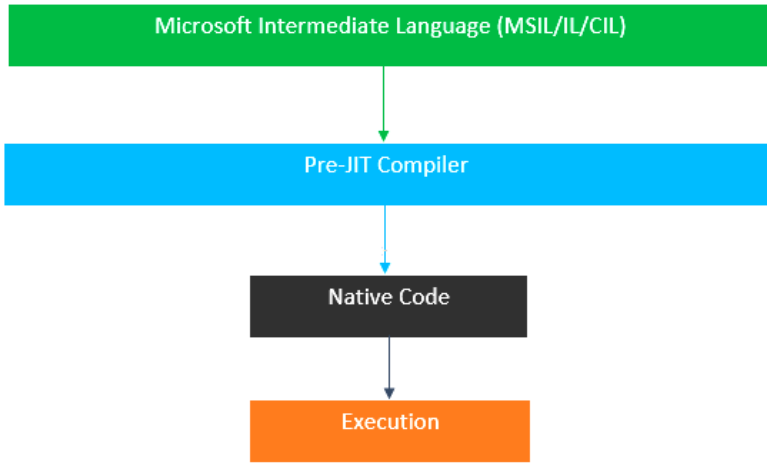


Fig 1.4 Pre JIT

Check your progress 13

1. Just-in-time converts:

   a. MSIL code to CPU native code

   b. CPU native code to MSIL code

   c. both a and b

   d. neither a nor b

1.15 Security Manager

A security manager is an object that defines a security policy for an application. This policy specifies actions that are unsafe or sensitive. Any actions not allowed by the security policy cause a Security Exception to be thrown. An application can also query its security manager to discover which actions are allowed. In a SecurityManager class as shown below:

```
public class SecurityManager
{
   private static readonly SecurityManager _current = new SecurityManager();
```

```
    public static SecurityManager Current
    {
        get { return _current; }
    }

    public bool CanSeeAdminPanel
    {
        // Change this code to a role that does/doesn't exist
        get { return Thread.CurrentPrincipal.IsInRole("Administrators"); }
    }
}
```

Further we see that:

```
<asp:Panel
    ID="adminPanel"
    runat="server"
    Visible="<%#     ExampleApplication.SecurityManager.Current.IsAdministrator
%>"
    >
    This is only visible to admins
</asp:Panel>
```

Check your progress 14

1. Security Manager is:

    a. application

    b. object

    c. data

    d. none of above

1.16 Let Us Sum Up

In this unit we have learnt that there are components that can be created once and can be further reuse number of times in various applications. It is seen that class libraries allow to define several classes, along with their methods and interfaces, in one file. It is seen that .NET framework show set of base class libraries having functions and features that are used with programming language for implementing .NET, such as Visual Basic, C#, Visual C++, etc.

We see that namespaces are the way to organize .NET Framework Class Library in logical grouping as per functionality, usability and category. A.NET programming language does not compile into executable code but compiles in intermediate code as Microsoft Intermediate Language (MSIL). The Portable Executable (PE) format is a file format for executables, object code, and DLLs, used in 32-bit and 64-bit versions of Windows operating systems.

Garbage collector determines whether any object in the heap is dead or not being used by the application. If such objects exist then memory used by these objects can be reclaimed. Common Type System shows set of types which can be used in different .Net languages in common that ensure objects interaction in different .Net languages. Common Language Specification is set of basic language features that .Net Languages needed to develop Applications and Services, which are compatible with the .Net Framework.

A security manager is an object that defines a security policy for an application. This policy specifies actions that are unsafe or sensitive. Any actions not allowed by the security policy cause a Security Exception to be thrown.

1.17 Answer for Check Your Progress

Check your progress 1

Answers: (1-b)

Check your progress 2

Answers: (1-a)

Check your progress 3

Answers: (1-b), (2-c)

Check your progress 4

Answers: (1-d)

Check your progress 5

Answers: (1-a), (2-a)

Check your progress 6

Answers: (1-b)

Check your progress 7

Answers: (1-a)

Check your progress 8

Answers: (1-a)

Check your progress 9

Answers: (1-a)

Check your progress 10

Answers: (1-a)

Check your progress 11

Answers: (1-c)

Check your progress 12

Answers: (1-c)

Check your progress 13

Answers: (1-c)

Check your progress 14

Answers: (1-b)

1.18 Glossary

1. **.NET framework -** Set of base class libraries describing functions and features used with programming language.
2. **Namespaces -** Way to organize .NET Framework Class Library in logical grouping depending on functionality and usability.
3. **Portable Executable -** It a type of file format used for executables, object code and DLLs for 32-bit and 64-bit versions of Windows operating systems.
4. **Managed code -** Code written to target services of managed runtime execution environment.
5. **Garbage collector -** Tool that shows whether object in heap is dead or not used by the application.

1.19 Assignment

Explain the .NET Framework.

1.20 Activities

What are the features of Common Language Runtime?

1.21 Case Study

What is the function of Common Type System?

1.22 Further Readings

1. Eric Gunnerson, a Programmer's Introduction to C#. Wiley
2. S. Robin, >NET Framework, Oxford

UNIT 2: ASP.Net and C#

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- Concept of Standard Controls
- Study about Validation Controls
- Study about ASP.NET Objects
- Features of Building StyleSheet
- Creating the Content Master

2.1 Introduction

C# is an elegant and type-safe object-oriented language that enables developers to build a variety of secure and robust applications that run on the .NET Framework. In this, C# can be used to create Windows client applications, XML Web services, distributed components, client-server applications, database applications, and much, much more. Visual C# provides an advanced code editor, convenient user interface designers, integrated debugger, and many other tools to make it easier to develop applications based on the C# language and the .NET Framework.

C# syntax is highly expressive, yet it is also simple and easy to learn. The curly-brace syntax of C# will be instantly recognizable to anyone familiar with C, C++ or Java. Developers who know any of these languages are typically able to begin to work productively in C# within a very short time. C# syntax simplifies many of the complexities of C++ and provides powerful features such as nullable value types, enumerations, delegates, lambda expressions and direct memory access, which are not found in Java. C# supports generic methods and types, which provide increased type safety and performance, and iterators, which enable implementers of collection classes to define custom iteration behaviors that are simple to use by client code. Language-Integrated Query (LINQ) expressions make the strongly-typed query a first-class language construct.

2.2 Working with Standard Controls

The standard controls in ASP.Net and C# are:

- Navigation

- Validation
- Login

2.2.1 Navigation Controls

We see that in ASP.NET there appears three navigation controls:

- Dynamic menus
- Tree Views
- Site Map Path

Menu Control: The Menu control will allow you to add navigational features in web pages. It handles main menu along with and submenus furthermore allows to show dynamic menus. It can be applied to other Navigation controls. Menu stored in file gets easily maintained which is normally called as web.

Tree Views: A Tree View will take care of displaying hierarchical list of items with the help of lines in order to connect interrelated items in a chain of command. Here all item has label and possible bitmap. It is found that Windows Explorer will make use of Tree View control to show directories. We can apply Tree View control in several circumstances in which you need to display hierarchical data.

Site Map Path: Use of this control is very simple. You can add this control to your page then view your page in browser. The Sitemap Path control displays the navigation path of the current page. The path acts as click able links to previous pages.

2.2.2 Validation Controls

Validation result as important feature of any web application where user input gets validated prior to sending from corner to corner in various layers of the application. It can be applied for:

- Implementing presentation logic.
- Validating user input data.
- Data format, data type and data range.

Validation is of two types:

- Client Side

- Serve Side

It is seen that in case of client side validation, the user will depend on browser as well as scripting language support, while in case of client side, the user will get immediate feedback. The benefit of this is that it saves page from being post back to server till client validates successfully.

So finally we see that validation involves testing of user data that gets entered in data field. After determining, user can additionally check for number or character entered as per required format. For creating ASP.NET Web pages, the user input will determine the information validity. So we see that in ASP.NET, set of validation controls will serve easy-to-use strong way of determining errors and will further display messages to user. There are six types of validation controls available in ASP.NET:

- Required Field Validation control
- Compare Validator Control
- Range Validator Control
- Regular Expression Validator Control
- Custom Validator Control
- Validation Summary

It is noted that in validation controls, ControlToValidate property is compulsory to every validate controls. In this, single validation control will validate only single input control whereas several validate control will also be assigned to input control.

2.2.3 Login Controls

In ASP.NET, login controls delivers robust login solution for ASP.NET Web applications without using programming. It is noted that normally, login controls integrate with ASP.NET membership and forms authentication to help automate user authentication for a Web site. It provides you with a ready-to-use user interface that queries the user name and password from the user and offers a Log In button for login. It validate user credentials against the membership API and encapsulating the basic forms authentication functionality like redirecting back to the original requested page in a restricted area of you application after the successful login.

The Login control displays a user interface for user authentication. The Login control contains text boxes for the user name and password and a check box that allows users to indicate whether they want the server to store their identity using ASP.NET membership and automatically be authenticated the next time they visit the site.

The Login control has properties for customized display, for customized messages, and for links to other pages where users can change their password or recover a forgotten password. The Login control can be used as a standalone control on a main or home page, or you can use it on a dedicated login page. If you use the Login control with ASP.NET membership, you do not need to write code to perform authentication. However, if you want to create your own authentication logic, you can handle the Login control's Authenticate event and add custom authentication code.

Check your progress 1

1. In the webpage, _____control is used to show hierarchical data.

   a. Menu

   b. Tree view

   c. Site Map Path

   d. All of above

2.3 Introduction to ASP.NET Objects

There are many objects in ASP.NET

Object Name	Description
Application:	It is applied to use information which is defined for complete application. It includes connection string applied to join database server which is stored in application object
Request	It allows ASP.NET applications to use information that is delivered by client at the time of Web request. It is a reference of HttpRequest Class Properties - QueryString , Cookies

Response	It allows ASP .NET application to deliver information to the client that gets referenced by HttpResponse ClassPublic void Write and Public void Redirect
Server	It gives methods applied to use methods along with properties of Web Server.

Check your progress 2

1. _____object method is applied for showing properties of Web Server.

   a. Response

   b. Server

   c. Request

   d. None of above

2.4 Building StyleSheet

Any beginner web developer / designer can easily apply a multitude of formatting to the web pages produced. One of the methods by which an experienced web designer can show up from the crowd is to provide flexibility to this formatting as well.

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation semantics (the look and formatting) of a document written in a markup language.

You can add an empty stylesheet to you web project, by:

1. Choose Website Website ➢ Add New Item in Visual Studio.
2. Select Style Sheet, edit the file name, and click OK.

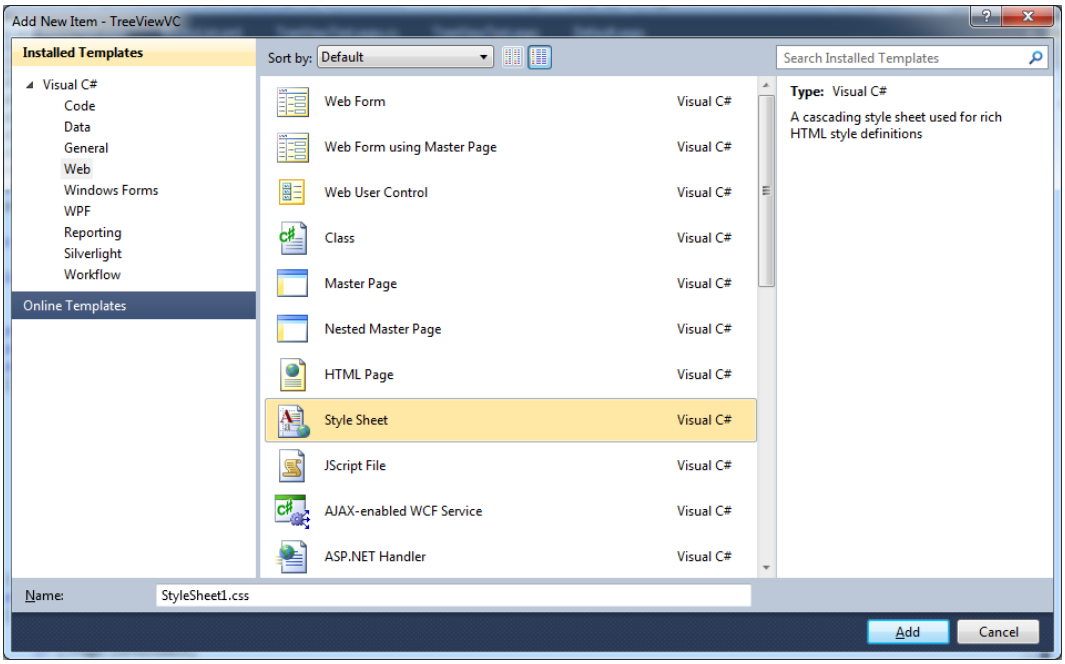


Fig 2.1 Adding Stylesheet

Stylesheets has certain rules which defines about formatting technique of every item in web page using certain rules:

1. The portion before the period specifies the HTML element to which the rule applies. If this part is empty the rule can apply to any tag.
2. The portion after the period is a unique case-sensitive name called the CSS class name. You use it to identify your rule.

A typical stylesheet defines a set of rules which are used to define the formatting for every significant piece of a website's user interface.

Check your progress 3

1. Style sheets are designed using:

   a. VC

   b. VB

   c. C++

   d. CSS

2.5 Creating the Content Master

Master page and content page work together to produce output to client. Sometimes, interaction between master page and content page is needed. Although too much of interaction between master page and content page can cause maintenance problems in future, it could be useful in some scenarios. There are few ways how content pages can access elements of master page.

A master page provides a template for other pages, with shared layout and functionality. The master page defines placeholders for the content, which can be overridden by content pages. The output result is a combination of the master page and the content page. The content pages contain the content you want to display.

When users request the content page, ASP.NET merges the pages to produce output that combines the layout of the master page with the content of the content page. The master page is a normal HTML page designed as a template for other pages. The @ Master directive defines it as master page having placeholder tag <asp:ContentPlaceHolder> for every content.

Step 1: Add Master page, right click in project property, and then click on Add new Item, now you will see Installed Template where you can select Master Page, finally click on **OK**.

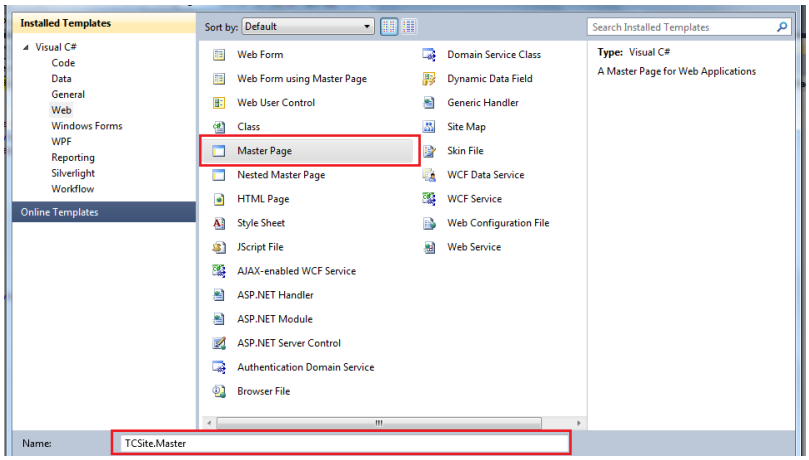


Fig 2.2 Installed template

Step 2: Now you can right click on Master Page (TCSite.Master), and then click **Add Content Page**.

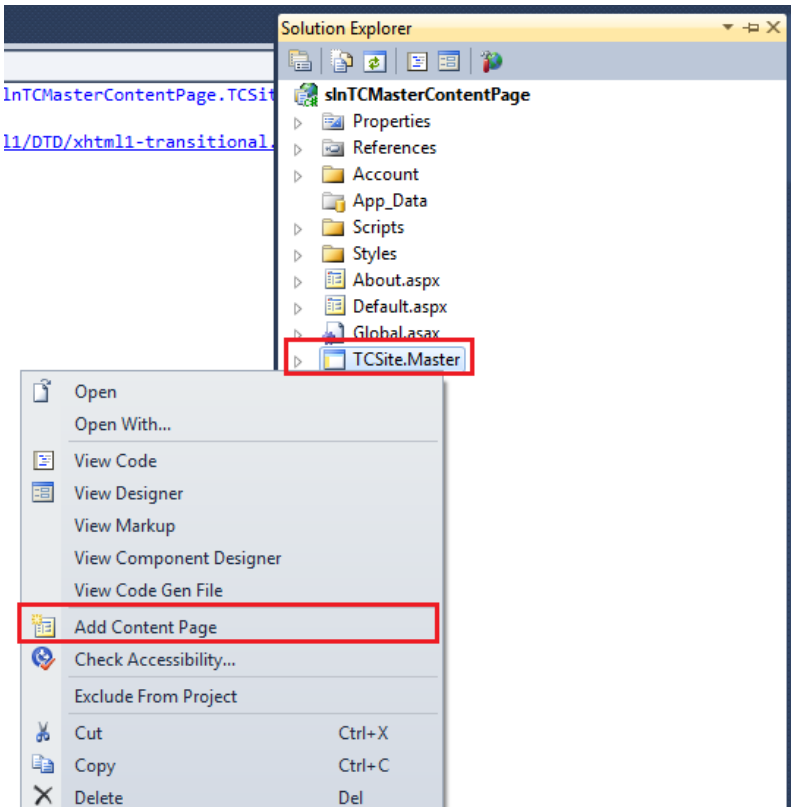


Fig 2.3 Add content page

Step 3: Finally, you can see **final view of Content Page**.

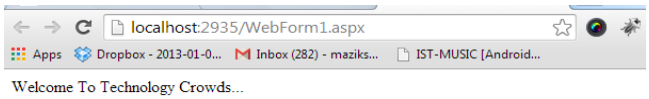


Fig 2.4 Content Page

Check your progress 4

1. Client Page is created using:

   a. Master page

   b. Content page

   c. Both a and b

   d. Neither a nor b

2.6 Adding Elements

Adding Elements in List:

- **AddRange:** AddRange adds an entire collection of elements. It can replace tedious foreach-loops that repeatedly call Add on List. We can pass any IEnumerable collection to AddRange, not just an array or another List.
- **Copy array:** You can create a List having elements from an array by using list constructor which will pass the array. The list receives this parameter and fills its values from it.
- **Test elements:** You can test every element for particular value. It is noted that for every loop, it will check for particular list of parameters.
- **IndexOf:** It shows element index of particular value in a List collection which locates for initial position of the value.

Accessing HTML elements

- With the introduction of .Netframework, there are many new features that gets added to be applied in web development. One of the basic feature is accessing HTML elements programmatically.
- Accessing <html> elements in code: It is found that there appears many new classes in .Netframework which are added in System.Web.UI.HtmlControls Namespace.
- Accessing <Head> tag: You can access <Head> tag through HtmlHead class which is packed in namespace System.Web.UI.HtmlControls. Here the Page object will show way through Header property in order to access <head> element.
- Applying Styles clearly to control: You can create css styles for particular controls by applying it to controls clearly. In ASP.Net controls, ApplyStyle(Style) and MergeStyle(Style) methods are packed with control to get the same.

Check your progress 5

1. The function of AddRange is to:

   a. add entire collection of elements

   b. create List having elements from an array

   c. test element for particular value

   d. show element index of particular value in List

2.7 Building the Site Navigation

To manage a website having with many pages, the direct way for visitors to navigate the website is done through site navigation. Initially, the site's navigational structure should be defined which must be translated into navigable user interface elements. After this, everything needs to be maintained and updated in order to further add new pages to site by removing previous ones. In case of ASP.NET site navigation system, the developer will able to define site map with which it can access information using programmatic API. ASP.NET ships with a site map provider that expects site map data to be stored in an XML file formatted in a particular way. In order to create site map, right-click on project name placed in Solution Explorer by selecting Add New Item and choosing Site Map option. Here, leave the name as Web.sitemap and click on Add button.

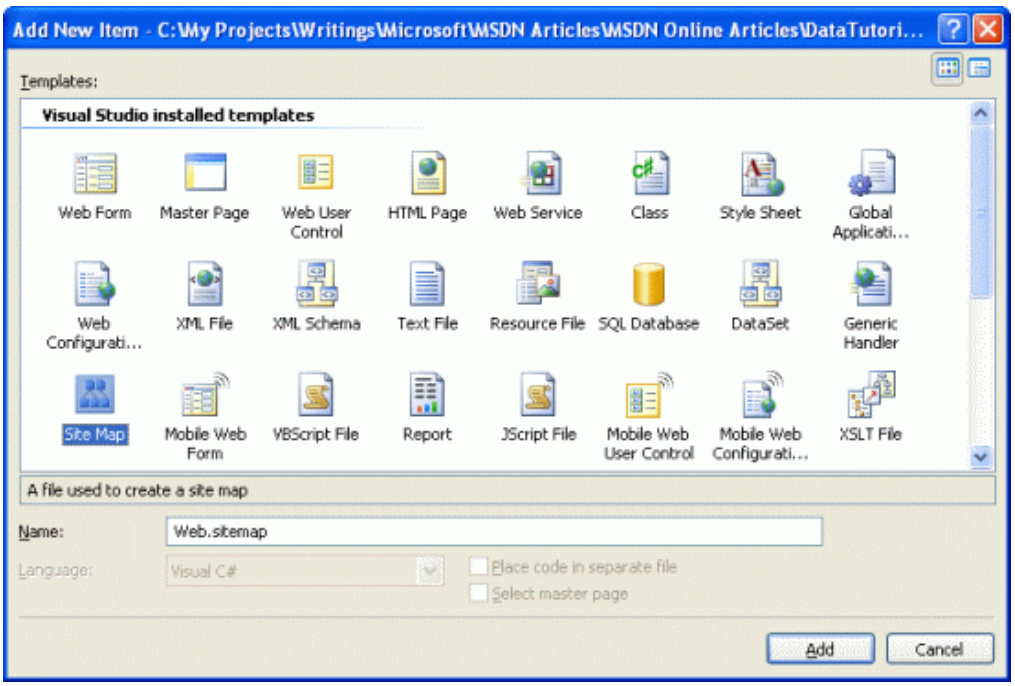


Fig 2.5 Add New Item Page

It is noted that the site map file is an XML file which carries <siteMap> node as its root node having precisely one <siteMapNode> child element. That first <siteMapNode> element can then contain an arbitrary number of descendent <siteMapNode> elements. It is seen that site map defines website's navigational structure in hierarchial structure as shown in fig 2.6 describing many sections of site. In this, all <siteMapNode> element in Web.sitemap shows section in site's navigational structure.

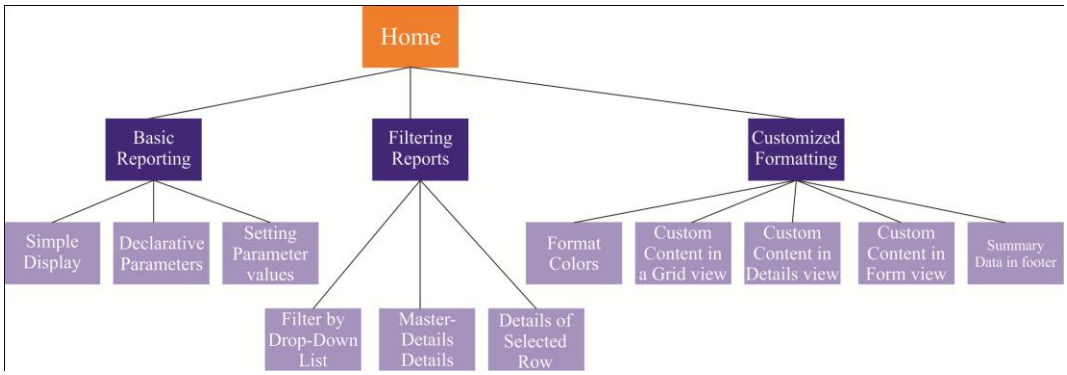


Fig 2.6 hierarchial structure of site map

Check your progress 6

1. Site map file is a form of:

   a. CSS file

   b. HTML file

   c. XML file

   d. None of above

2.8 Adding Authentication

Authentication is the ability to identify a particular entity. The need for authentication occurs when we have some resources that we want to make available to different entities. We store these resources in a centralized place and instruct the system that manages them to prevent entities that we don't recognize from having access. Anonymous authentication refers to a situation in which we grant access to resources to all users, even if we don't know them. There are several ways to add Authentication to an existing project. There are three kinds of authentication in ASP.NET:

- Form
- Windows
- Passport

Form authentication is cookie based, as ASP.NET places a cookie in the client machine in order to track the user. If the user requests a secure page and has not logged in, then ASP.NET redirects him/her to the login page. Once the user is authenticated, he/she will be allowed to access the requested page.

In ASP.NET, you can setup Windows account authentication. It is noted that such type of authentication doesn't involve ASP.NET engine but works at Internet Information Server (IIS) level. It needs correct IIS configuration where authentication types in IIS can be analysed using IIS Manager:

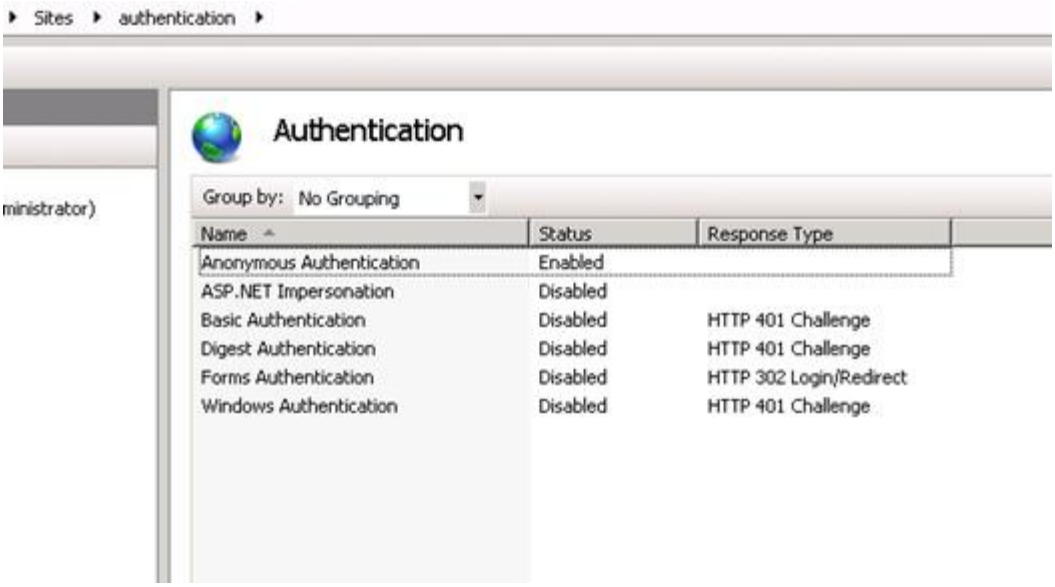


Fig 2.7 IIS manager

In the Window account authentication, here the user requires username and password which are recognized by operating system for particular application. In this, when user calls web page, a dialog box appears which will ask for credentials. If user provides valid credentials for valid Window account, then in such case the authentication gets completed. It is not a secured authentication as in this, the authentication data gets transmitted to server in form of text.

Check your progress 7

1. Form authentication is base on:

   a. temp

   b. cookie

   c. both a and b

   d. neither a nor b

2.9 Adding Content Pages

A content page is an ASP.NET Web page that is associated with a master page. The master page establishes a layout and includes one or more ContentPlaceHolder controls for replaceable text and controls. The content page includes only the text and controls that are merged at run time with the master page's ContentPlaceHolder controls.

After creating a content page, you can create custom content that corresponds to each ContentPlaceHolder on the master page. Alternatively, you can explicitly choose to allow the master page's default content to be displayed. To add a content page in Visual Web Developer:

- In Solution Explorer, right-click on Web site and click Add New Item.
- In Visual Studio in templates list, click on Web Form.
- Select the Select master page check box and click Add.
- Select a Master Page dialog box appears.
- In Contents of Folder box, click master page with which you want to associate with page you create and click on OK.

Check your progress 8

1. ASP web page is associated with:

   a. content page

   b. master page

   c. both a and b

   d. neither a nor b

2.10 Working with Data

In ASP.NET, you can create access and review data from database with the help of ASP.NET Web Forms and Entity Framework Code. Entity Framework is an object-relational mapping framework where you can access with relational data by removing many data access code applied for writing. With Entity Framework, you will get queries by LINQ which retrieve and manipulate data as strongly typed objects. Further this Entity Framework will focus on creating remaining application instead of focusing on data access.

With Entity Framework, the development paradigm known as Code First is framed which defines data models with classes that create own custom types with grouping variables together using types, methods and events. Also, we see that by creating classes showing data which can configure application applied for classes. Further we see that the data is passed between server and client. If data is known at point, then page gets rendered by adding script which forms data or setting json using literal or similar variations.

Check your progress 9

1. queries in Entity Framework is given by:

   a. LINQ

   a. NQIL

   c. ILNQ

   d. QINL

2.11 Using ASP.NET Web Services and WCF

Web Service in ASP.NET

A Web Service is programmable application logic accessible via standard Web protocols. One of these Web protocols is the Simple Object Access Protocol (SOAP). SOAP is a W3C submitted note that uses standards based technologies to encode and transmit application data.

Consumers of a Web Service do not need to know anything about the platform, object model, or programming language used to implement the service; they only need to understand how to send and receive SOAP messages.

WCF Service

Windows Communication Foundation is a framework for building service-oriented applications. Using WCF, you can send data as asynchronous messages from one service endpoint to another. A service endpoint can be part of a continuously available service hosted by IIS, or it can be a service hosted in an application.

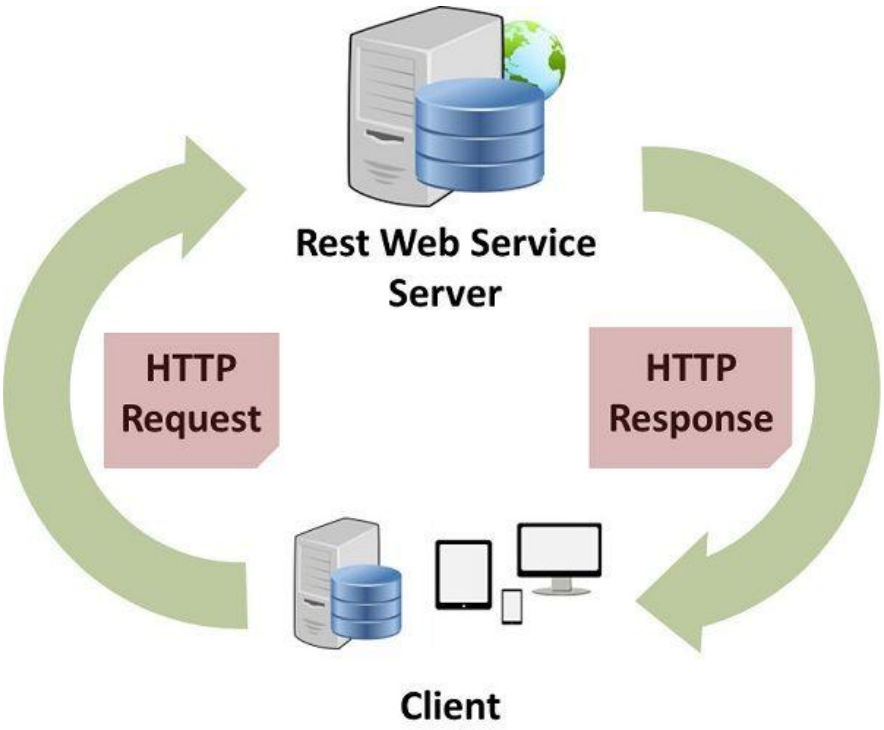


Fig 2.8 WCF arrangement

In fig 2.8, we see that the endpoint is a client of service which requests data from service endpoint. Here the messages can be as simple as single character or word which is sent as XML, or can be complex as stream of binary data. It is noted that WCF can be applied in case when:

- Secure service to process business transactions.
- Service which gives current data to others.
- Chat service allows two people to communicate or exchange data in real time.
- Dashboard application polls having services for data with logical presentation.
- Opening workflow using Windows Workflow Foundation as a part of service.
- Silverlight application to poll a service for latest data feeds.

Features of WCF

- Service Orientation
- Interoperability
- Multiple Message Patterns
- Service Metadata
- Data Contracts
- Security
- Multiple Transports and Encodings
- Reliable and Queued Messages
- Durable Messages
- Transactions
- AJAX and REST Support
- Extensibility

Comparison Chart

WCF	ASP.NET Web Services

[ServiceContract] and [OperationContract] attributes defines web service and methods.
 	
`[ServiceContract]`
`public interface ITest`
`{`
`  [OperationContract]`
`  string ShowMessage();`
`}`
`public class Service : ITest`
`{`
`  public string ShowMessage()`
`  {`
`  return "Hello World!";`
`  }`
`}`
	[WebService] and [WebMethod] attributes defines web service and methods.
	
	`[WebService]`
	`public class Service :`
	`System.Web.Services.WebService`
	`{`
	`  [WebMethod]`
	`  public string Test()`
	`  {`
	`  return "Hello World!";`
	`  }`
	`}`

Hosted in IIS, WAS (Windows Activation Service) Self-hosting, Windows service.	Hosted in IIS.
Accessed through HTTP, TCP, MSMQ, P2P, Named pipes.	Accessed through HTTP.
Supports security, reliable messaging, transactions, durable messages, service orientation, interoperability, service metadata, AJAX and REST support, extensibility.	Supports Security services.
Uses the ServiceMetadata tool (svcutil.exe) to generate the client for the service.	Uses the command-line tool WSDL.EXE to generate the client for the service.
Unhandled exceptions are not returned to clients as SOAP faults. A configuration setting is provided to have the unhandled exceptions returned to clients for the purpose of debugging.	Unhandled exceptions are returned to the client as SOAP faults.
The generated WSDL can customized by using ServiceMetadataBehavior class.	The generated WSDL can customized by using ServiceDescriptionFormatExtension class.

System.RunTime.Serialization is supported. -Better performance. -DataContractAttribute and DataMemberAttribute can be added to .NET Framework types to indicate that instances of the type are to be serialized into XML, and which particular fields or properties of the type are to be serialized. -Classes that implement the IDictionary interface can be serialized. -Hash table can be serialized.	System.XML.Serialization is supported -Worse performance. -Only Public fields or Properties of .NET types can be translated into XML. -Only the classes which implement IEnumerable and ICollection interface can be serialized. -Hash table can not be serialized.
Can be multithreaded via ServiceBehavior class.	Can not be multithreaded.
Supports different type of bindings like BasicHttpBinding, WSHttpBinding, WSDualHttpBinding etc.	Only used SOAP or XML for this.

Check your progress 10

1. SOAP is:

   a. Simple Object Access Protocol

   b. Sample Object Access Protocol

   c. Simple Oriented Access Protocol

   d. Simple Object Access Process

2.12 Creating a simple ASP.NET Web Service

In order to understand the concept of Web service, create a web service of Equity stock price information. In this, we see that clients will query about name and price of stock as per stock symbol. We see that web service has three methods:

- HelloWorld method
- GetName Method
- GetPrice Method

We will consider following steps to create web service:

Step (1): Select File -> New -> Web Site in Visual Studio, and then select ASP.NET Web Service.

Step (2): A web service file called Service.asmx and its code behind file, Service.cs is created in the App_Code directory of the project.

Step (3): Change the names of the files to StockService.asmx and StockService.cs.

Step (4): The .asmx file has simply a WebService directive on it:

<%@ WebService Language="C#" CodeBehind="~/App_Code/StockService.cs" Class="StockService" %>

Step (5): Open StockService.cs file, the code generated in it is the basic Hello World service. The default web service code behind file looks like:

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml.Linq;
namespace StockService
{
  // <summary>
  // Summary description for Service1
  // <summary>
  [WebService(Namespace = "http://tempuri.org/")]
  [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
  [ToolboxItem(false)]
  // To allow this Web Service to be called from script,
  // using ASP.NET AJAX, uncomment the following line.
  // [System.Web.Script.Services.ScriptService]
  public class Service1 : System.Web.Services.WebService
```

```
  {
    [WebMethod]
    public string HelloWorld()
    {
      return "Hello World";
    }
  }
}
```

Step (6) : Change the code behind file to add the two dimensional array of strings for stock symbol, name and price and two web methods for getting the stock information.

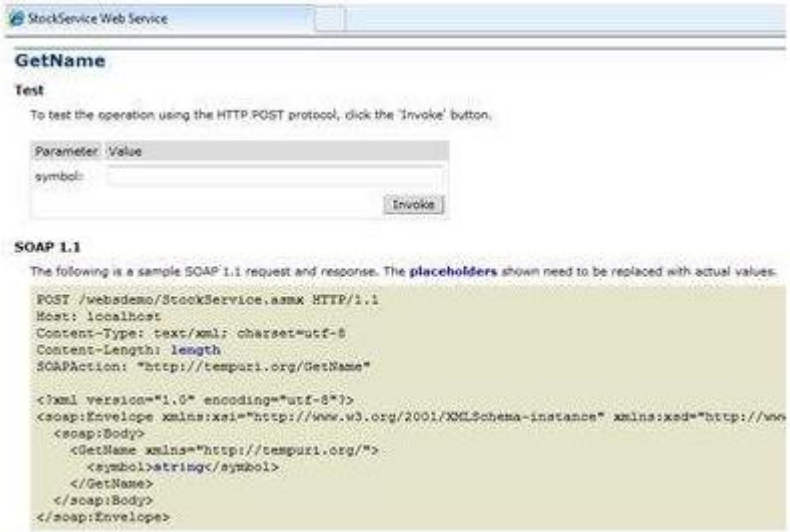
```
using System;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Xml.Linq;
[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
// To allow this Web Service to be called from script,
// using ASP.NET AJAX, uncomment the following line.
// [System.Web.Script.Services.ScriptService]

public class StockService : System.Web.Services.WebService
{
  public StockService () {
    //Uncomment the following if using designed components
    //InitializeComponent();
  }
```

```
string[,] stocks =
{
  {"RELIND", "Reliance Industries", "1060.15"},
  {"ICICI", "ICICI Bank", "911.55"},
  {"JSW", "JSW Steel", "1201.25"},
  {"WIPRO", "Wipro Limited", "1194.65"},
  {"SATYAM", "Satyam Computers", "91.10"}
};
[WebMethod]
public string HelloWorld() {
  return "Hello World";
}
[WebMethod]
public double GetPrice(string symbol)
{
  //it takes the symbol as parameter and returns price
  for (int i = 0; i < stocks.GetLength(0); i++)
  {
    if (String.Compare(symbol, stocks[i, 0], true) == 0)
    return Convert.ToDouble(stocks[i, 2]);
  }
  return 0;
}
[WebMethod]
public string GetName(string symbol)
{
  // It takes the symbol as parameter and
  // returns name of the stock
  for (int i = 0; i < stocks.GetLength(0); i++)
```

```
        {
            if (String.Compare(symbol, stocks[i, 0], true) == 0)
            return stocks[i, 1];
        }
        return "Stock Not Found";
    }
}
```

Step (7) : On running web service application, we see that web service will have test page that allow testing service methods.

Step (8): Click on a method name, and check whether it runs properly.



Step (9): In this, we see that in order to test GetName method, provide one of the stock symbols, that returns name of stock.

Check your progress 11

1. web service can be implemented using:

   a. HelloWorld method

   b. GetName Method

   c. GetPrice Method

   d. All of above

2.13 Let Us Sum Up

While studying this unit, we have learnt that C# is an elegant and type-safe object-oriented language that enables developers to build a variety of secure and robust applications that run on the .NET Framework. In ASP.NET, login controls delivers robust login solution for ASP.NET Web applications without using programming which forms authentication to help automate user authentication for a Web site.

We have seen that Cascading Style Sheets is a style sheet language used for describing the presentation semantics of a document written in a markup language. The master page and content page work together to produce output to client. Sometimes, interaction between master page and content page is needed. To manage a website having with many pages, the direct way for visitors to navigate the website is done through site navigation. Initially, the site's navigational structure should be defined which must be translated into navigable user interface elements.

It is known that authentication is ability to find particular entity which occurs when we have some resources that we want to make available to different entities. A content page is an ASP.NET Web page is associated with master page which establishes a layout and includes one or more ContentPlaceHolder controls for replaceable text and controls. A Web Service is programmable application logic accessible via standard Web protocols. One of these Web protocols is the Simple Object Access Protocol (SOAP). SOAP is a W3C submitted note that uses standards based technologies to encode and transmit application data.

2.14 Answers for Check Your Progress

Check your progress 1

Answers: (1-b)

Check your progress 2

Answers: (1-b)

Check your progress 3

Answers: (1-d)

Check your progress 4

Answers: (1-c)

Check your progress 5

Answers: (1-a)

Check your progress 6

Answers: (1-c)

Check your progress 7

Answers: (1-b)

Check your progress 8

Answers: (1-c)

Check your progress 9

Answers: (1-a)

Check your progress 10

Answers: (1-a)

Check your progress 11

Answers: (1-d)

2.15 Glossary

1. **ASP.NET -** A set of .NET classes used to create Web-based, client-side (Web Form) and server-side (Web Service) applications.
2. **Client -** Any application that requests information or services from a server.
3. **Code Access Security -** The common language runtime's security model for applications.
4. **Web.config -** Application configuration files contain settings specific to an application.
5. **Authorization -** It shows identity to be granted to request type in order to access given resources.
6. **Authentication -** It discovers and verify principal identity, by examining user's credentials and validating credentials against authority.

2.16 Assignment

Discuss the steps involved in adding authentication and content pages.

2.17 Activities

Collect some information on ASP.NET Objects.

2.18 Case Study

Generalised the basic difference between ASP and ASP.NET.

2.19 Further Readings

1. M. James, "Overview of ASP.NET and Web Forms", Wiley.
2. Rumy, "Code Behind vs. Code Inline"--.NET Framework, Oxford.

Block Summary

In this block, students have learnt and understand about the basic of Common Type System and Common Language Specification. The block gives an idea on the study and concept of Security Manager with various functions and characteristics. You have been well explained on the concepts of Managed Code along with its features and usage.

The block detailed about the basic of building StyleSheet as well as steps involved in adding elements. The concept related to adding Authentication and adding Content Pages are well explained to you. You will be demonstrated about ASP.NET WebServices and WCF.

Block Assignment

Short Answer Questions

1. What is ASP.NET?
2. What are steps involved in building Site Navigation?
3. What are various MS.NET Namespaces?
4. What are the features of Common Language Runtime?

Long Answer Questions

1. Explain characteristics of MS.NET Memory Management.
2. Write short note on creating Content Master.
3. What is the benefit of ASP.NET?

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

……