

IT6505 Middleware Architecture

(Optional)

INTRODUCTION

Middleware Architecture is one of the optional courses in Semester 6. The aim of this course is to gain the knowledge on different middleware architectures and related concepts.

CREDITS: 03

LEARNING OUTCOMES

After successful completion of this course, students will be able to:

- Describe fundamental concepts of Network and Distributed Computing.
- Describe IT architecture and Middleware, analyse and compare different categories of middleware and learn about their architectural models.
- Describe and analyse different procedures in RPC (remote procedure calls).
- Understand basic message queuing concepts in nexus to middleware.
- Define the functionality and requirements of a Distributed Object.
- Describe and analyse object-oriented middleware.
- Describe and analyse different procedures in CORBA and Java RMI.
- Understand basic processes of Web services, analyse and build systems that use RESTful Web services.

MINOR MODIFICATIONS

When minor modifications are made to this syllabus, those will be reflected in the Virtual Learning Environment (VLE) and the latest version can be downloaded from the relevant course page of VLE. Please inform your suggestions and comments through the VLE.

<http://vle.bit.lk>

ONLINE LEARNING MATERIALS AND ACTIVITIES

You can access all learning materials and this syllabus in the VLE: <http://vle.bit.lk>, if you are a registered student of BIT degree program. It is very important to participate in learning activities given in the VLE to learn this subject.

FINAL EXAMINATION

Final exam of the course will be held at the end of the semester. It is a two hour structured written paper.

OUTLINE OF SYLLABUS

Topic	Hours
1- Network & distributed computing, IT architecture & middleware and RPC	12
2- Message queuing	08
3- Object-oriented middleware	10
4- Web services	15
Total for the subject	45

REQUIRED MATERIALS**Main Reading**

Ref 1: Felber, P., 1998, "The CORBA object group service: A service approach to object groups in CORBA." PhD diss., ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE.

Ref 2: Britton, C. and Bye, P., 2004. IT Architectures and Middleware: Strategies for Building Large, Integrated Systems (2nd Edition – Low Price Edition). Pearson Addison Wesley.

Ref 3: Emmerich, W. and Shrivastava, Santosh K., "Engineering Distributed Objects", 1st Edition, John Wiley & Sons Ltd.

Ref 4: Burke, B., "Restful Java with JAX-RS", 1st Edition, O'Reilly Media.

Ref 5: Bishop, T.A. and Karne, R.K., 2003, March. A Survey of Middleware. In Computers and Their Applications (pp. 254-258).

Ref 6: "Use of middleware - Wikipedia, the free encyclopedia," https://en.wikipedia.org/wiki/Middleware_%28distributed_applications%29, [Online; accessed 13-March-2016].

Ref 7: "Remote Procedure Calls - Cardiff School of Computer Science & Informatics", <http://www.cs.cf.ac.uk/Dave/C/node33.html>, [Online; accessed 13-March-2016].

Ref 8: “rpcgen - Cardiff School of Computer Science & Informatics”,
<http://www.cs.cf.ac.uk/Dave/C/node34.html#ch:rpcgen>, [Online; accessed 13-March-2016].

Ref 9: “rpcgen Tutorial – ORACLE”, <https://docs.oracle.com/cd/E19683-01/816-1435/rpcgenpguide-21470/index.html>, [Online; accessed 13-March-2016].

Ref 10: “Message broker- Wikipedia, the free encyclopedia,”
https://en.wikipedia.org/wiki/Message_broker, [Online; accessed 13-March-2016].

Ref 11: “Introduction to Amazon SQS – Amazon.com”,
<http://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSGettingStartedGuide/Introduction.html>, [Online; accessed 13-March-2016].

Ref 12: “WSDL Tutorial – tutorialspoint.com”, <http://www.tutorialspoint.com/wsdl>, [Online; accessed 13-March-2016].

Ref 13: “SOAP Tutorial– tutorialspoint.com”,
<http://www.tutorialspoint.com/soap/index.htm>, [Online; accessed 13-March-2016].

DETAILED SYLLABUS:

Section 1: Network & distributed computing, IT architecture & middleware and RPC (12hrs.)

Instructional Objectives

- Explain fundamental concepts of Network and Distributed Computing.
- Detail IT architecture and Middleware: analyse and compare different categories of middleware and learn about their architectural models.
- Outline and analyse different processes in RPC (remote procedure calls).

Sub Topics

1.1 Introduction to Network and Distributed Computing:

- 1.1.1 Evolution of Network Computing [Ref 1 pg. 5 - 6].
- 1.1.2 Requirements for Distributed Systems [Ref 3 pg. 14 - 19].
- 1.1.3 Issues in Distributed Computing [Ref 1 pg. 6 - 8]:
 - 1.1.3.1 Types of Reliability.
 - 1.1.3.2 Types of System Failures.
- 1.1.4 Ingredients of a Distributed Systems [Ref 3 pg. 4 - 6].
- 1.1.5 Centralized versus Distributed Systems [Ref 3 pg. 6 - 8].

1.2 IT Architecture and Middleware:

- 1.2.1 Importance of IT architecture [Ref 2 pg. 31 - 36].
- 1.2.2 Integration infrastructure in IT architecture [Ref 2 pg. 38]:
 - 1.2.2.1 Physical Deployment (Bus, Hub).
- 1.2.3 What is middleware [Ref 2 pg. 37]?
- 1.2.4 Taxonomy of Middleware [Ref 5]:
 - 1.2.4.1 Integration and Application Middleware:
 - 1.2.4.1.1 Types: Data Access middleware, Object-oriented middleware, Procedure Oriented/RPC middleware, Message-oriented middleware, Agent middleware, Component or Reflective middleware, Web-based/Portal middleware.
- 1.2.5 Middleware and its associated elements [Ref 2 pg. 38].
- 1.2.6 Functions of Middleware (What does middleware do?) [Ref 6].

1.3 Remote Procedure Call (RPC) [Ref 2 pg. 53 -55]:

- 1.3.1 What is RPC?
- 1.3.2 RPC mechanisms.
- 1.3.3 IDL, stubs and skeletons.
- 1.3.4 RPC Model.
- 1.3.5 Marshalling.
- 1.3.6 Problems of RPC.
- 1.3.7 Writing RPC programs with rpcgen [Ref 7, 8, 9].

Section 2: Message queuing (08hrs)**Instructional Objectives**

- Compare and contrast message queuing methods with other rivalries.
- Outline details about message queuing on the cloud.

Sub Topics**2.1 Message Queuing:**

- 2.1.1 Message Broker Pattern for Message oriented middleware (MOM) [Ref 10].
- 2.1.2 What is Message queuing [Ref 2 pg. 59 - 60]?
- 2.1.3 Message queuing vs RPC [Ref 2 pg. 61].
- 2.1.4 Message queuing vs Distributed Transaction Processing [Ref 2 pg. 61 - 64].
- 2.1.5 Message queuing on Cloud (cloud middleware): Massive message processing with Amazon SQS [Ref 11].

Section 3: Object-Oriented Middleware (10hrs)**Instructional Objectives**

- Explain the role of Interface Definition Language (IDL).
- Outline general steps of developing applications with OOM.
- Describe and analyses different processes of CORBA and Java RMI.
- Outline architecture of Enterprise Java Beans (EJBs)

Sub Topics**3.1 CORBA**

- 3.1.1 Introduction to OOM [Ref 2 pg. 69 - 70].
- 3.1.2 Interface Definition Language (IDL) [Ref 2 pg. 74 - 75].
- 3.1.3 CORBA Object Management Architecture [Ref 3 pg. 89 - 90]:
 - 3.1.3.1 CORBA facilities.
 - 3.1.3.2 CORBA services.
 - 3.1.3.3 Domain Interfaces.
- 3.1.4 CORBA ORB Architecture [Ref 3 pg. 96 - 98].
- 3.1.5 Meta-Object Model and IDL [Ref 3 pg. 90 - 96].
 - 3.1.5.1 Objects.
 - 3.1.5.2 Object Types.
 - 3.1.5.3 Modules.
 - 3.1.5.4 Attributes.
 - 3.1.5.5 Operations.
 - 3.1.5.6 Requests.
 - 3.1.5.7 Exception and Error Handling.
 - 3.1.5.8 Subtypes and Multiple Inheritance.
 - 3.1.5.9 Polymorphism.

3.2 JAVA RMI:

- 3.2.1 What is Java RMI [Ref 3 pg. 109]?
- 3.2.2 Meta-Object Model and Interface Definition [Ref 3 pg. 110 - 115]:
 - 3.2.2.1 Java.rmi.Remote interface instead of IDL.
 - 3.2.2.2 Objects.
 - 3.2.2.3 Object Types.
 - 3.2.2.4 Attributes.
 - 3.2.2.5 Operations:
 - 3.2.2.5.1 Remote Method Invocation:
 - 3.2.2.5.1.1 Object Call-by-Reference.
 - 3.2.2.5.1.2 Object Call-by-value (Copy Object):
 - 3.2.2.5.1.2.1 Serialization.
 - 3.2.2.6 Requests.
 - 3.2.2.7 Exception and Error Handling.
 - 3.2.2.8 Subtypes and Multiple Inheritance.
 - 3.2.2.9 Polymorphism.
- 3.2.3 RMI Architecture [Ref 3 pg. 115 - 117].
- 3.2.4 RMI Registry [Ref 3 pg. 215 - 219].
- 3.2.5 Enterprise Java Beans (EJB) [Ref 2 pg. 79 - 80] :
 - 3.2.5.1 The relationship between EJB and RMI.
 - 3.2.5.2 EJB concepts: Beans and Containers.

Section 4: Web services (15hrs.)**Instructional Objectives**

- Explain in detail, Web service concepts.
- Outline details about WSDL (Web Services Description Language) and SOAP.
- Explain and understand fundamentals of Restful Web services.
- Build Restful Web services applications with JAX-RS.

Material /Sub Topics**4.1 Web Services:**

- 4.1.1 Service-oriented-architecture (SOA):
 - 4.1.1.1 What is SOA [Ref 2 pg. 39]?
 - 4.1.1.2 Service Concepts [Ref 2 pg. 89 - 95].
- 4.1.2 What is a Web Service [Ref 2 pg. 95 - 101]?
- 4.1.3 Using Web Services [Ref 2 pg. 101 - 106].
- 4.1.4 Web Services Description Language (WSDL) [Ref 12]:
 - 4.1.4.1 WSDL Elements:
 - 4.1.4.1.1 Definition.
 - 4.1.4.1.2 Data types.
 - 4.1.4.1.3 Message.
 - 4.1.4.1.4 Port Type.
 - 4.1.4.1.5 Binding.
 - 4.1.4.1.6 Ports.
 - 4.1.4.1.7 Service.

- 4.1.5 Universal Description, Discovery, and Integration (UDDI) [Ref 2 pg. 100, 102 - 103].
- 4.1.6 Simple Object Access Protocol (SOAP) [Ref 13]:
 - 4.1.6.1 What is SOAP?
 - 4.1.6.2 SOAP Message Format:
 - 4.1.6.2.1 Envelope.
 - 4.1.6.2.2 Header.
 - 4.1.6.2.3 Body.
 - 4.1.6.2.4 Fault.
 - 4.1.6.2.5 SOAP syntax rules.
- 4.1.7 RESTful Web Services
 - 4.1.7.1 The REST architectural style [Ref 4 pg. 3 - 5].
 - 4.1.7.2 RESTful Architectural Principles [Ref 4 pg. 5 - 13].
 - 4.1.7.3 Designing RESTful Services [Ref 4 pg. 15 - 25]:
 - 4.1.7.3.1 Defining the Object Model.
 - 4.1.7.3.2 Modelling the URIs.
 - 4.1.7.3.3 Defining the Data Format.
 - 4.1.7.3.4 Assigning the HTTP methods.
 - 4.1.7.4 JAX-RS:
 - 4.1.7.4.1 What is JAX-RS [Ref 4 pg. 27]?
 - 4.1.7.4.2 Writing Simple JAX-RS Services [Ref 4 pg. 27 - 37].
 - 4.1.7.4.3 Deploying JAX-RS Services [Ref 4 pg. 38 - 41].
 - 4.1.7.4.4 HTTP Method and URI Matching:
 - 4.1.7.4.4.1 Binding HTTP Methods [Ref 4 pg. 43 – 45].
 - 4.1.7.4.4.2 More uses of @Path [Ref 4 pg. 45 – 50 (exclude Subresource Locators)].
 - 4.1.7.4.5 JAX-RS Injection
 - 4.1.7.4.5.1 @PathParam [Ref 4 pg. 56 – 59].
 - 4.1.7.4.5.2 @MatrixParam [Ref 4 pg. 60].
 - 4.1.7.4.5.3 @QueryParam [Ref 4 pg. 60 – 61].
 - 4.1.7.4.5.4 @FormParam [Ref 4 pg. 61 – 62].
 - 4.1.7.4.5.5 @HeaderParam [Ref 4 pg. 62 – 63].
 - 4.1.7.4.5.6 @CookieParam [Ref 4 pg. 63 - 64].