Chapter 2
System Models

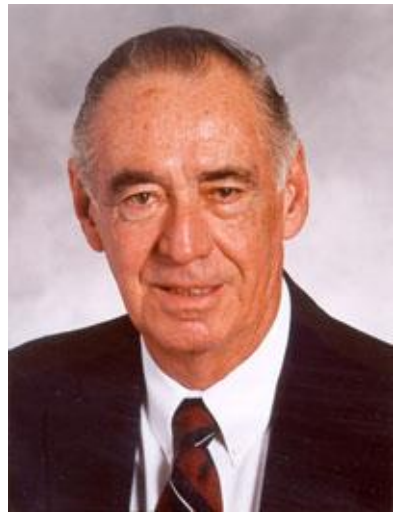
Gandeva Bayu Satrya, ST., MT.
Telematics Labz.
Informatics Department
Telkom University

Outline Today

Chapter 2 – System Models

- System Model ?
- Physical Model ?
- Architectural Model ?
- Fundamental Model ?

System Models



Prof. Robert E. Shannon

By a **model** we mean a representation of a group of objects or ideas in some form other than that of the entity itself.

By a **system** we mean a group or collection of interrelated elements that cooperate to accomplish some stated objective.

Difficulties and Threats

a) Widely varying modes of use
   - ✓ have special requirements

b) Wide range of system environments
   - ✓ accommodate heterogeneous hardware

c) Internal problems
   - ✓ Non-synchronized clocks

d) External threats
   - ✓ denial of service attacks

System Models in DS

- should be designed to function correctly in the widest possible range of circumstances and in the face of many possible difficulties and threats.
- How the properties and design issues of distributed systems can be captured and discussed through the use of descriptive models ?
- Each type of model is intended to provide an abstract, simplified but consistent description of a relevant aspect of distributed system design.

DS Models

1) Physical Models
   - ✓ the most explicit way in which to describe a system
2) Architectural Models
   - ✓ the computational and communication tasks performed by its computational elements
3) Fundamental Models
   - ✓ abstract perspective in order to examine individual aspects of a distributed system

1. Physical Models

- A physical model is a representation of the underlying hardware elements of a distributed system

- Baseline physical model
  - Early DS : ‘70s-’80s → LAN
  - Internet-scale DS : ’90 → Internet
  - Contemporary DS : Now → Mobile Computing

Generation of DS

Distributed systems:	*Early*	*Internet-scale*	*Contemporary*
Scale	Small	Large	Ultra-large
Heterogeneity	Limited (typically relatively homogenous configurations)	Significant in terms of platforms, languages and middleware	Added dimensions introduced including radically different styles of architecture
Openness	Not a priority	Significant priority with range of standards introduced	Major research challenge with existing standards not yet able to embrace complex systems
Quality of service	In its infancy	Significant priority with range of services introduced	Major research challenge with existing services not yet able to embrace complex systems

2. Architectural Models

- its structure in terms of separately specified components and their interrelationships
- Major concerns are to make the system reliable, manageable, adaptable and cost-effective
- The section adopts a three-stage approach:
    a) Architectural elements : looking at the core
    b) Architectural patterns : examining composite architectural
    c) considering middleware platforms

a. Architectural Elements

- ✓ Communicating entities : Objects, Components, and Web services.
- ✓ Communication paradigms : interprocess communication, remote invocation, and indirect communication.
- ✓ Roles and responsibilities : Client-server, Peer-to-peer
- ✓ Placement : mapping of services to multiple servers, caching (Chace server), mobile code (stockbroker), and mobile agents (accessing individual database).

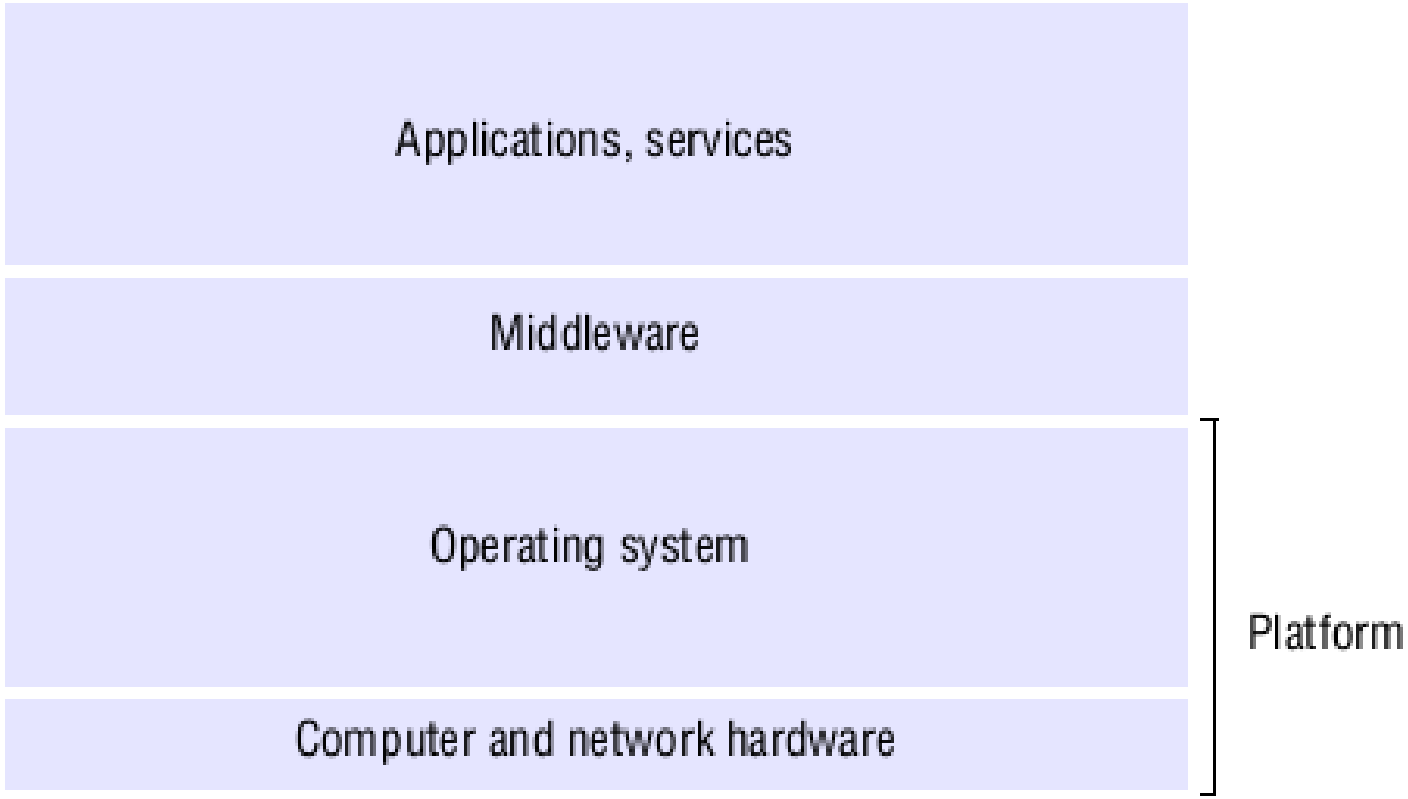
Entities and Paradigms

Communicating entities (what is communicating)		*Communication paradigms (how they communicate)*		
System-oriented entities	*Problem-oriented entities*	*Interprocess communication*	*Remote invocation*	*Indirect communication*
Nodes	Objects	Message passing	Request-reply	Group communication
Processes	Components	Sockets	RPC	Publish-subscribe
	Web services	Multicast	RMI	Message queues
				Tuple spaces
				DSM

b. Architectural Patterns

- Layering : App, Middleware, OS, and Net HW.
- Tiered architecture : presentation (handling GUI), application (detailed application-specific processing), and data (persistent storage of the application)
- Thin clients : supports a window-based user interface.
- Other commonly occurring patterns : proxy server and brokerage.

SW and HW Service Layers



c. Associated Middleware

The task of middleware :

- ✓ to provide a higher-level programming abstraction for the development of distributed systems and,
- ✓ through layering, to abstract over heterogeneity in the underlying infrastructure to promote interoperability and portability.

- Categories of middleware (SunRPC)
- Limitations of middleware (a database of names and addresses)

Categories of Middleware

Major categories:	*Subcategory*	*Example systems*
Distributed objects (Chapters 5, 8)	Standard	RM-ODP
	Platform	CORBA
	Platform	Java RMI
Distributed components (Chapter 8)	Lightweight components	Fractal
	Lightweight components	OpenCOM
	Application servers	SUN EJB
	Application servers	CORBA Component Model
	Application servers	JBoss
Publish-subscribe systems (Chapter 6)	-	CORBA Event Service
	-	Scribe
	-	JMS
Message queues (Chapter 6)	-	Websphere MQ
	-	JMS
Web services (Chapter 9)	Web services	Apache Axis
	Grid services	The Globus Toolkit
Peer-to-peer (Chapter 10)	Routing overlays	Pastry
	Routing overlays	Tapestry
	Application-specific	Squirrel
	Application-specific	OceanStore
	Application-specific	Ivy
	Application-specific	Gnutella

3. Fundamental Models

The purpose of such a model is:

- To make explicit all the relevant assumptions about the systems we are modelling.
- To make generalizations concerning what is possible or impossible, given those assumptions.

Discuss and Reason about

- Interaction: Computation occurs within processes;
- Failure: a fault occurs in any of the computers;
- Security: to attack by both external and internal agents.

a. Interaction model

two significant factors affecting interacting processes in a distributed system:

- ✓ Communication performance is often a limiting characteristic.
- ✓ It is impossible to maintain a single global notion of time.

- Performance of communication channels : Delay, bandwidth, and jitter.
- Computer clocks and timing events : NTP with GPS
- Two variants of the interaction model : Synchronous DS and Asynchronous DS.
- Event ordering : sending or receiving a message

b. Failure model

the ways in which failure may occur in order to provide an understanding of the effects of failures.

- Omission failures : Process omission failures and Communication omission failures
- Arbitrary failures : takes unintended processing steps
- Timing failures : process execution time
- Masking failures : hiding it altogether
- Reliability of one-to-one communication : validity and integrity

c. Security model

securing the processes and the channels used for their interactions and by protecting the objects that they encapsulate against unauthorized access.

- Protecting objects : manages a collection of objects.
- Securing processes and their interactions : handle financial transactions, confidential or classified information.
- The enemy : capable of sending, reading, and copying.
- Defeating security threats : Cryptography and Authentication.
- Other possible threats from an enemy : DoS

References

[COU'12] Coulouris, G. Dollimore, J., Kindberg, T., Blair, G., DISTRIBUTED SYSTEMS :Concepts and Design Fifth Edition, Pearson Education, Inc., United States of America, 2012.

[TAN'07] Tanenbaum, A.S., Steen, M.V., DISTRIBUTED SYSTEMS : Principles and Paradigms Second Edition, Pearson Education, Inc., United States of America, 2007.

[SHA'98] Shannon, R.E., INTRODUCTION TO THE ART AND SCIENCE OF SIMULATION, Proceedings of the 1998 Winter Simulation Conference. USA

Thank You

Gandeva Bayu Satrya, ST., MT.
Telematics Labz.
Informatics Department
Telkom University