Cloud computing is based on a large number of ideas and the experience accumulated since the first electronic

computer was used to solve computationally challenging problems. In this chapter we overview

parallel and distributed systems concepts that are important to understanding the basic challenges in the

design and use of computer clouds.

Cloud computing is intimately tied to parallel and distributed computing. Cloud applications are

based on the client-server paradigm with relatively simple software, a thin client, running on the user’s

machine while the computations are carried out on the cloud. Many cloud applications are data-intensive

and use a number of instances that run concurrently. Transaction processing systems, such as Webbased

services, represent a large class of applications hosted by computing clouds; such applications

run multiple instances of the service and require reliable and in-order delivery of messages.

The concepts introduced in this section are very important in practice. Communication protocols

support coordination of distributed processes and transport information through noisy and unreliable

communication channels that may lose messages or deliver duplicate, distorted, or out-of-order messages.

To ensure reliable and in-order delivery of messages, the protocols stamp each message with

a sequence number; in turn, a receiver sends an acknowledgment with its own sequence number to

confirm the receipt of a message. The clocks of a sender and a receiver may not be synchronized, so

these sequence numbers act as logical clocks. Timeouts are used to request the retransmission of lost

or delayed messages.

The concept of consistent cuts and distributed snapshots are at the heart of checkpoint-restart procedures

for long-lasting computations. Indeed, many cloud computations are data-intensive and run for

extended periods of time on multiple computers in the cloud. Checkpoints are taken periodically in

anticipation of the need to restart a software process when one or more systems fail; when a failure

occurs, the computation is restarted from the last checkpoint rather than from the beginning.

Many functions of a computer cloud require information provided by monitors, system components

that collect state information from the individual systems. For example, controllers for cloud resource

management, discussed in Chapter 6, require accurate state information; security and reliability can

only be implemented using information provided by specialized monitors. Coordination of multiple

instances is a critical function of an application controller.