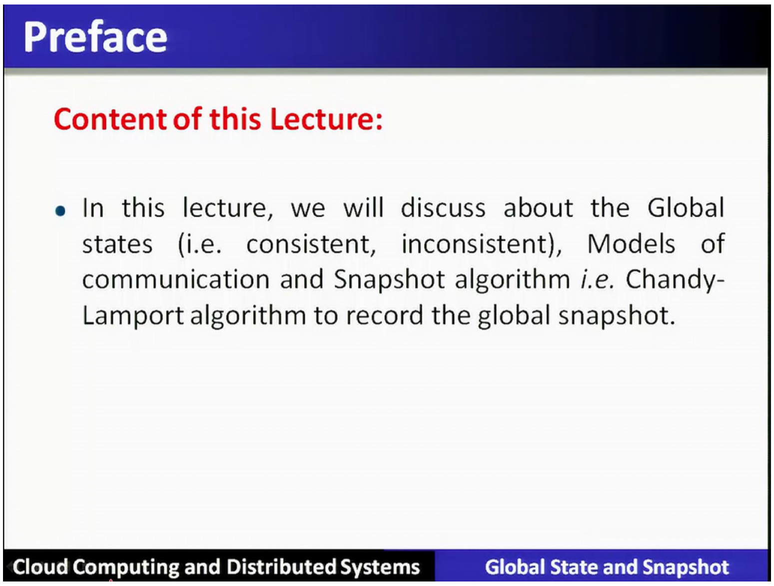
Cloud Computing and Distributed Systems
Dr. Rajiv Misra
Department of Computer Science and Engineering
Indian Institute of Technology, Patna

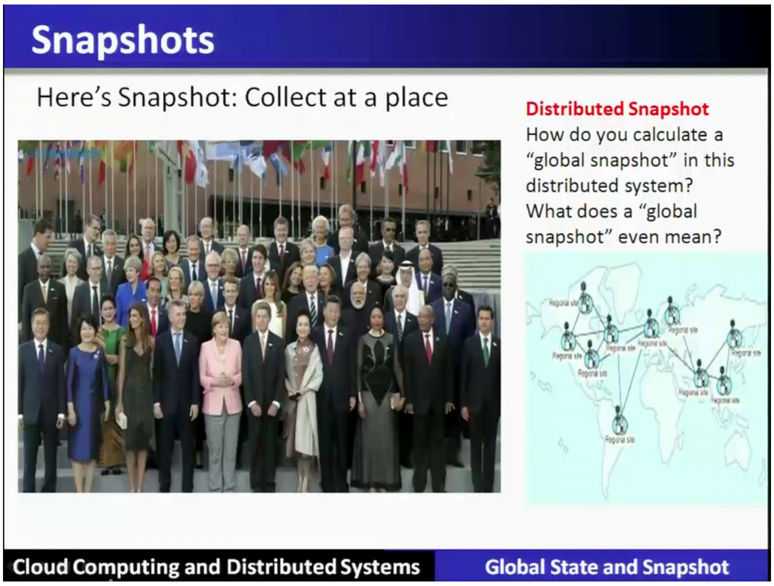
Lecture – 11
Global State and Snapshot

(Refer Slide Time: 00:17)



Global State and Snapshot Recording Algorithms. Content of this lecture. We will discuss global states, models of communication snapshot algorithms. Especially, we will focus on Chandy-Lamport's algorithm for recording of a global state.

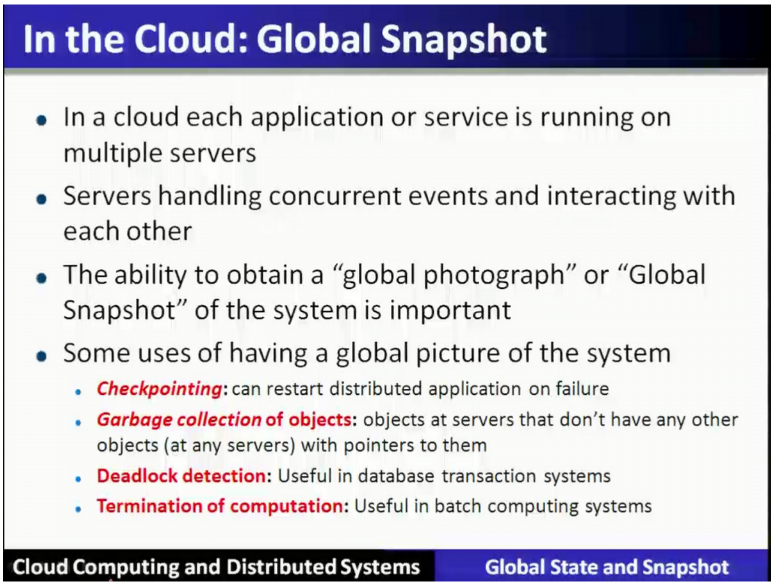
(Refer Slide Time: 00:33)



Snapshots. So, we have seen here in this particular picture that all the head of the nations, they have collected at one place and snapshot was taken up. Idea is that, everyone has to come and gather at one place. Whereas, we will see on the right side, the concept of a distributed snapshot, without coming at one place, how can you take the snapshot that is called a distributed snapshot.

So, snapshot in a distributed system that all, that scenario is also there in the cloud system, how this snapshot that is called a global distributed snapshot is taken up in such a scenario. So, this example is a geographically distributed cloud, which is nothing but a distributed system. And the distributed snapshot is a challenge herel and this is the topic for our discussion. So, what does this global snapshot even mean all these things we will explore.

(Refer Slide Time: 01:48)

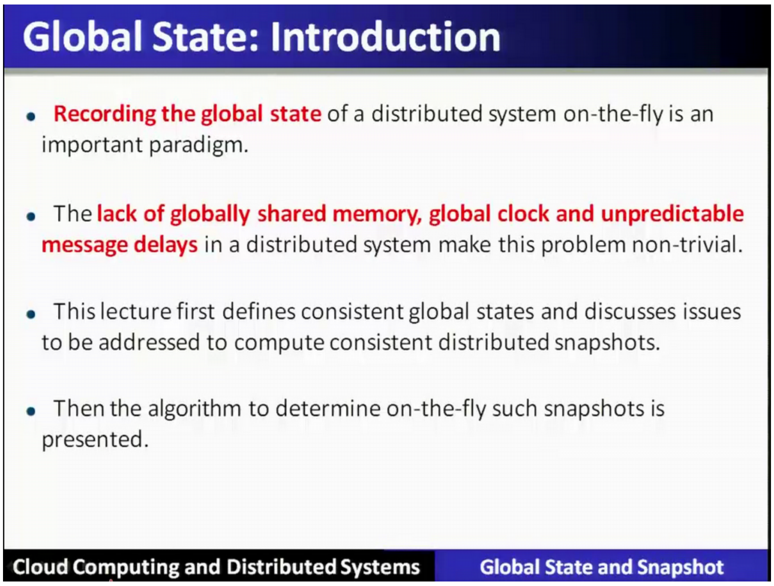


So, in a cloud, each application or a service is running on multiple servers and the servers handling the concurrent event and interacting with each other. So, the ability to obtain the global snapshot of the entire system is important because of the following applications.

For example, sometimes check pointing requires a global snapshot or a global picture of the system. So, if a checkpoint is available, and if there is a failure, then system can restart without much lost that is called check pointing requires, the input of a global snapshot. Similarly, the garbage collection of the objects to find out about the objects, which are not even pointed or not being used by the other servers, they are to be removed off that also requires the global snapshot.

Deadlocks; in such a system is also useful that can only be done or that can only be analyzed, if the global snapshot of the entire system is available. Similarly, in a batch computing system, the termination of the computation is important and that can be analyzed from the input of a global snapshot.

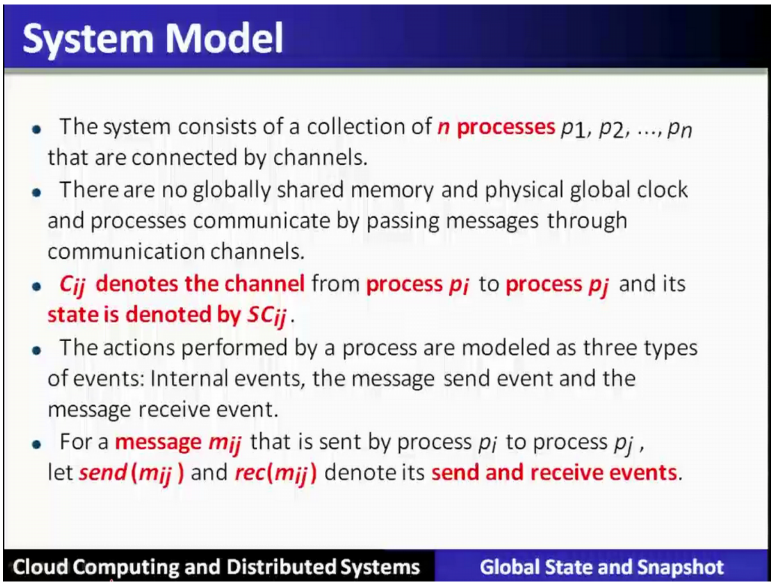
(Refer Slide Time: 03:21)



So, to understand, this let us see some of the preliminaries and some definitions, then only we will see the recording of a global snapshot algorithm. So, recording the global state of a distributed system on-the-fly, that means, without stopping the system, how the snapshot is to be taken is an important paradigm.

Now, the challenge over here is that the model of a distributed system that is the geographically distributed cloud system lacks a global shared memory, and also a global common clock and it also has the messages, which have the finite, but unpredictable delays making this particular problem of global snapshot a non-trivial problem. So, as I told you that we will first build up the basics, to understand the global snapshot problem. And then, we will see the algorithm of global snapshot or snapshot recording global state recording.

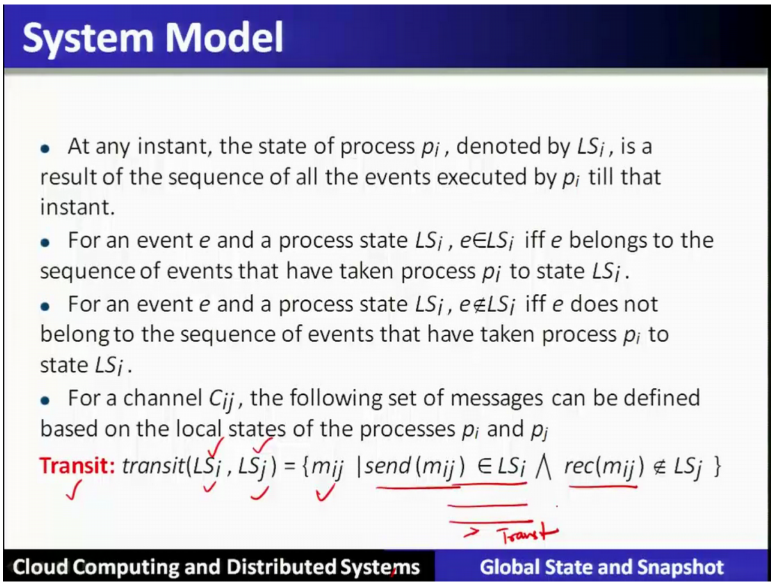
(Refer Slide Time: 04:38)



System model. So, the system consist of the collection of n processes p1 to pn, we assume that they are all connected by the channels. We also assume that there is no globally shared memory; we also assume that there is no global common physical clock. Therefore, the processes communicate only by the means of messages using communication channels.

Let C ij denotes the channel from a process i to j; and that is denoted by state of a channel SC ij. The actions performed by the by the process are modeled as three different types; internal events, message send event, and message received events. For message m ij that is sent by a process i to process j, let send m ij be the send of an event; similarly, receive m ij denotes the receive of an event.

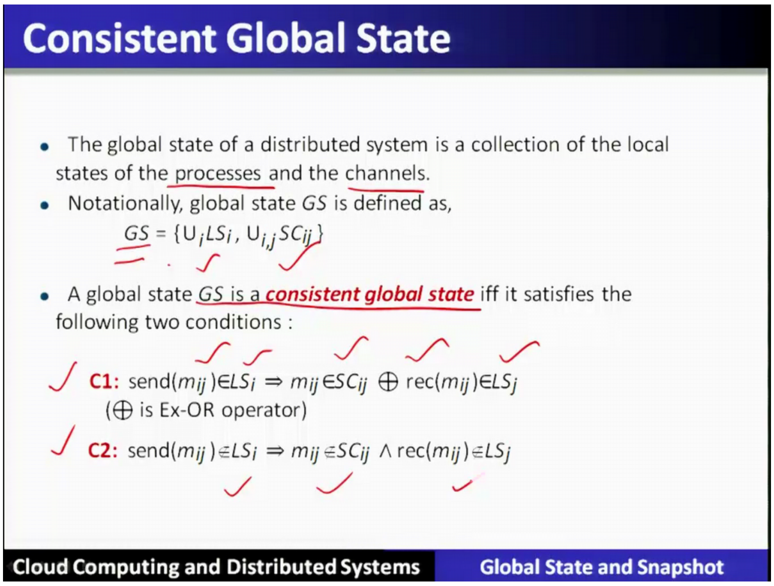
(Refer Slide Time: 05:43)



At any instant, the state of a process pi is denoted by LS i, is a result of the sequence of all the events executed by pi, till that instant. So, for an event e and a process state LS i, e is an LS i, if e belongs to the sequence of events that have taken in the process pi to the state of LS i. For event e and a process state LS i, is not in LS i, if e does not belong to the sequence of event, which has taken place in a process to their state. Similarly, for a channel C ij, the following set of messages can be defined based on the local state of a process.

So, the state of the channel is called transit. If given two different states of two different processes i and j, there exist a message m ij. Such that the send of that particular message is part of the state of a channel i. And the received of m ij is not recorded here in LS j that means, the messages message is in the channel or that is called in the transit.

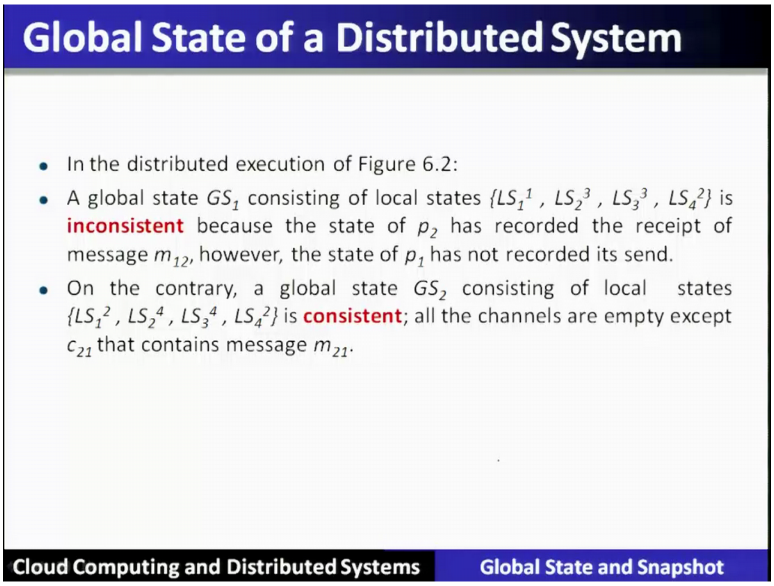
(Refer Slide Time: 07:18)



Consistent global state definition. So, a global state of a distributed system is a collection of the local state of the processes and the corresponding channel. Notationally, global state GS is basically the collection of the local state of all the processes that is the union of LS i's; and the union of state of channel SC i's.

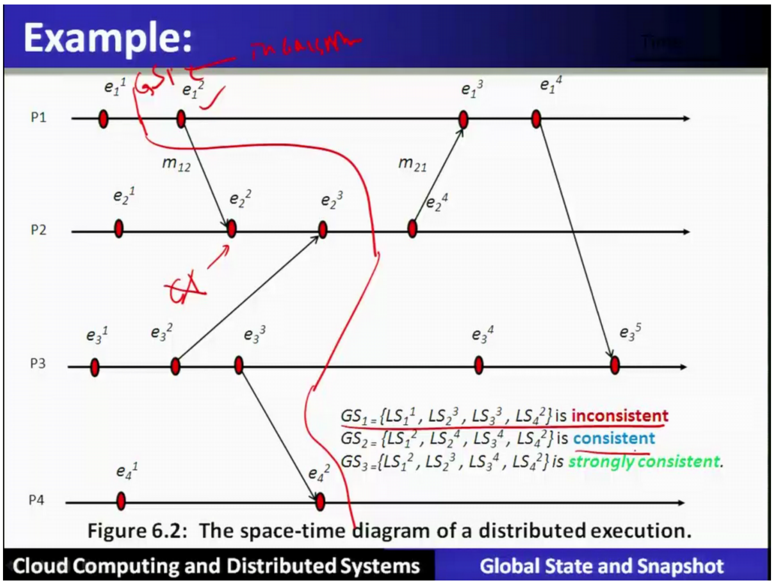
So, the global state the GS, which we have just defined is a consistent global state, if an only satisfies two conditions; C 1, and C 2. C 1 says that if the send of a particular message m ij is in the state of a channel state of a local channel i, this implies that the message is in state of a channel or it is received by a process. So, this is an exclusive or operation. Similarly, if the message is not in the any of the local states, this will imply that neither it is in the state of a channel, nor it is received at any of the local states that is then only it is called a consistent state.

(Refer Slide Time: 08:49)



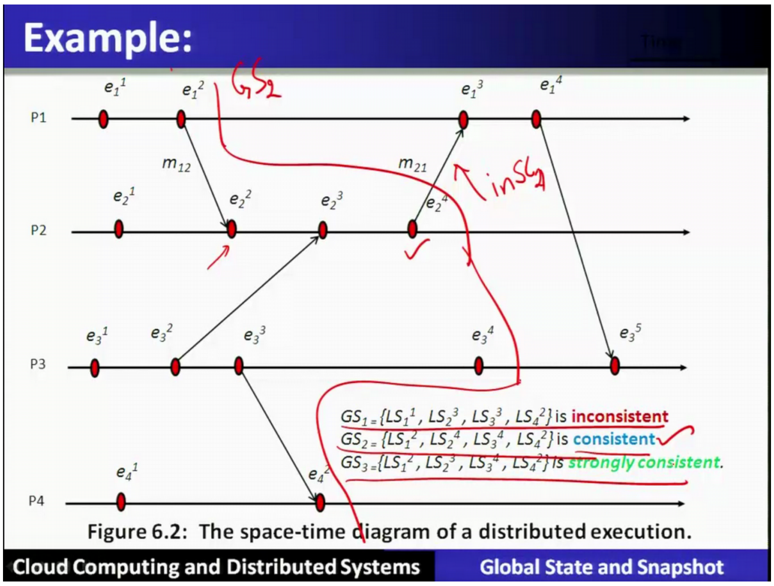
We will see the example of a local state, which is inconsistent that means, it does not follow the property C 1 and C 2. Let us see here in this particular example.

(Refer Slide Time: 09:02)



So, global state 1 is inconsistent. Let us figure out in this particular diagram, where is local state 1, that is 1 that is p 1 up to 1; here this is line. Then for p 2 up to 3, this is the line; then 3 and 3, this is the line and 4 and 2, this is the line. So, this is a global state 1 and this is inconsistent. Why, because of this particular message is recorded as I received, but it does not recorded as the send, so that means, C 1 is violated.

(Refer Slide Time: 10:16)



Now, we will see another one, which is called consistent that is LS, LS 1 of 2, this; then LS 2 of 4, this; then LS 3 of 4 this, and this one. This is the consistent global state. You can see here, the send of message is recorded, but the receive is not recorded that is fine, but send is recorded. So, this is called consistent global state. So, this message must be in the in the state of a channel that is 2 1.

Similarly, there is another global state, which is called GC GS 3; and this is called strongly consistent. Let us trace 3 2, and then 3 4, and then this is the 2. So, you just see that all the messages, which are sent is being received. So, there is no message, which is in the transit. Hence, it is called strongly consistent global state.

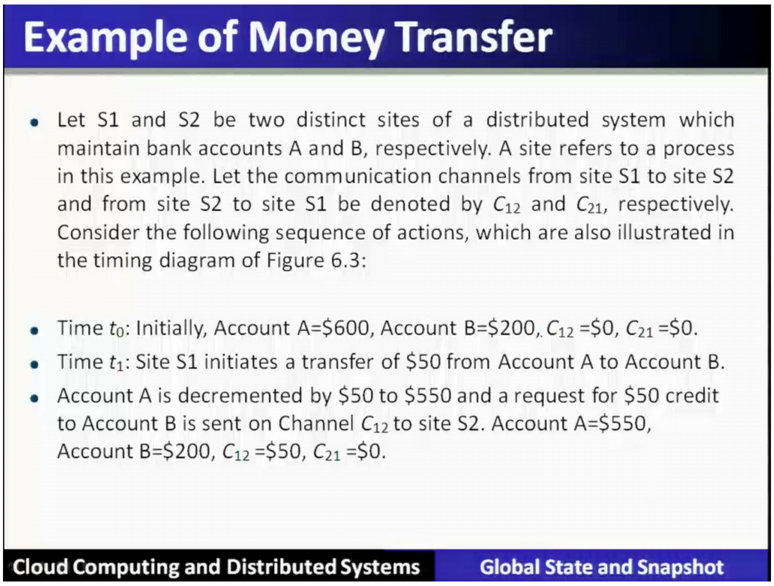
(Refer Slide Time: 11:15)

So, let us see the definitions, we have already understand through the diagram and examples. So, global state is transitless that means, there is no message in the channels that means, all the channels are empty that means, whatever message is sent, they are being received. If that is the condition, then that state is called strongly consistent state.

Now, let us see the issues in recording of a global state in a distributed systems. So, the first issue I 1 says that how to distinguish between the messages to be recorded in a snapshot from those, which or not to be recorded. So, that means, any message that is sent by a process before recording its snapshot, must be recorded in a global snapshot that is it has to follow the condition C 1. Any message that is send by a process after recording its snapshot must not be recorded that is the condition C 2. So, the issue 1 is to differentiate between the messages, which are to be recorded; and the messages, which are not to be recorded that is the issue number 1.

Issue number 2 says that how to determine the instant, when the process takes the snapshot. A process pj must record a snapshot before processing a message m ij that was sent by a process i after recording its snapshot. So, these two issues are important. And let us see how it is going to be taken care, while we discuss the recording of a snapshot.

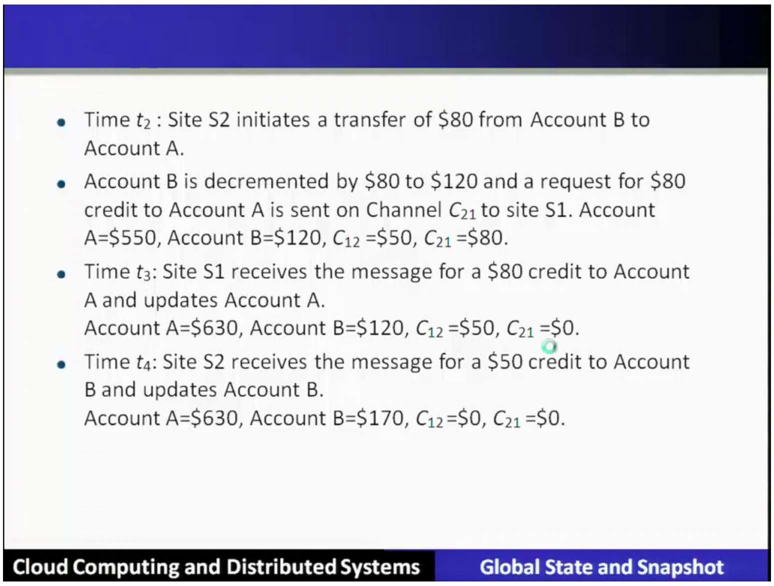
(Refer Slide Time: 13:10)



There is a example of a money transfer that we will take up, when we will discuss the algorithm. Let us understand this example of a money transfer. Let there are two sites; S 1, and S 2 in a distributed system, and which maintains the bank account A and B, respectively that is site S 1 maintains bank account A, and bank account B is maintained at site S 2. Let the communication channel from site S 1 to S 2 and from site S 2 to S 1 be denoted by C 1 2 and C 2 3, respectively.

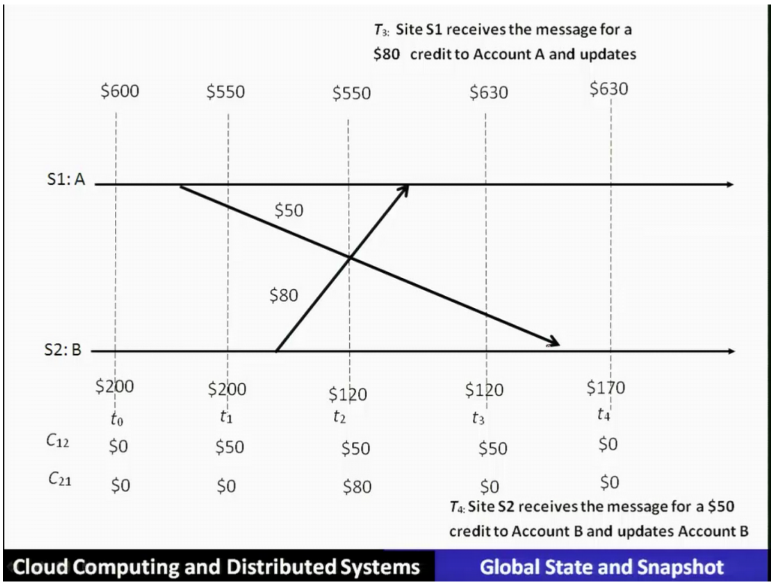
Now consider the following sequence of actions, which are illustrated. At time t 0, the account of A is 600 dollars, account B is 200 dollars and the channels are empty. At time t 1, site S 1 will initiate a transfer of 50 dollars from account A to account B. So, accounting A is decremented by 50 dollars and a request to credit 50 dollars is being sent the message as a message to the account B over a communication channel C 1 2. So, the account A becomes 550, whereas the account B is not changed. And the request to credit that 50 dollar is in the message, which is in the communication channel C 1 2, C 2 1 is 0.

(Refer Slide Time: 15:01)



And so on, we will see this example here in this particular way.

(Refer Slide Time: 15:05)

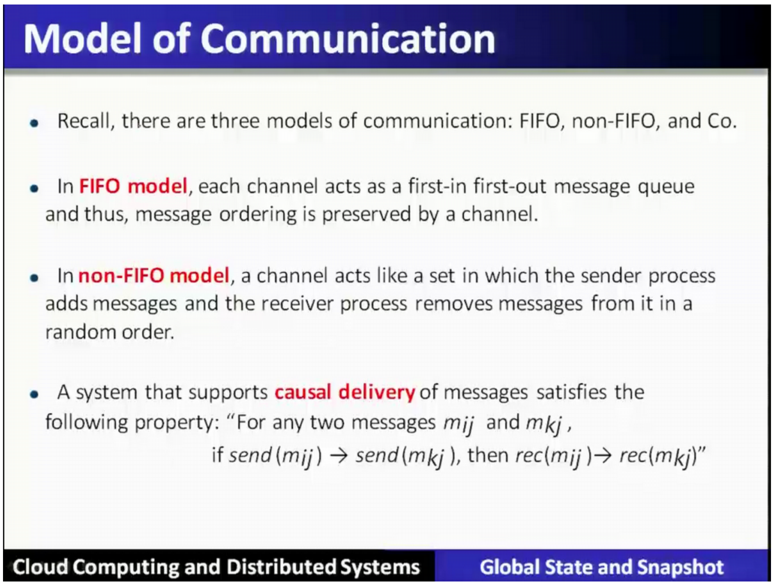


So, here as I told you that 50 dollar is being sent, and it will be debited, similarly here in at time t 1. Before time t 2, 80 dollar is being sent from B to A and that corresponding things are being reflected in this state time diagram.

(Refer Slide Time: 15:35)

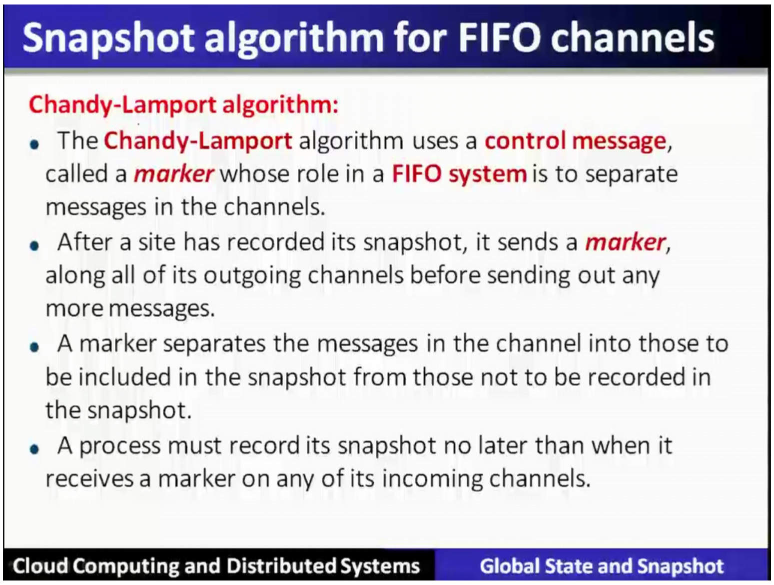
Now, here in this case, if we record something like that we are recording 600 dollar, and then 120 and we are taking this. At time t 0, we are taking the snapshot of A; and at time t 2, we are taking the values of time t 2. Then this kind of recording will have inconsistent information. Why, because the total value should be 800 at any point in the recording recorded global state. So, if you see if you count this, it will be 750 dollars, and so on. So, this is not or this is a inconsistent snapshot, if we take.

(Refer Slide Time: 16:30)



Now, let us assume some of the models of communication. We will consider three different model of communication in this model, for the global snapshot algorithm FIFO non-FIFO, and the causal order.

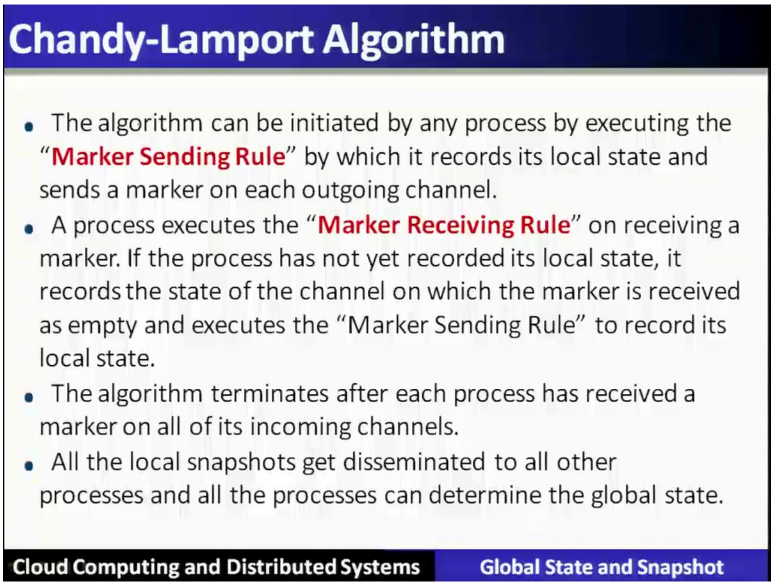
(Refer Slide Time: 16:50)



Now, let us consider the Chandy-Lamport algorithm for the FIFO channels. So, Chandy-Lamport algorithm uses the control messages, which are called marker messages, and whose role in FIFO is to separate the messages, which are to be recorded from the messages, which are not to be recorded.

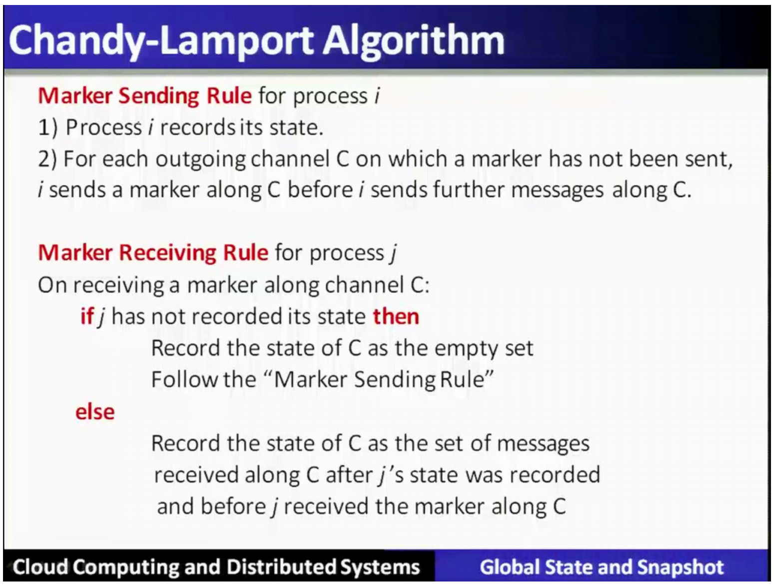
So, the marker is basically dividing the messages, which are to be recorded, which are not to be recorded. So, when a marker comes, all the message before the marker arrives are to be recorded; and after marker arrival, whatever messages are coming are not going to be recorded. So, marker is basically a separating point. So, marker will is used to decide what are things to be recorded, what are the messages which are not to be recorded.

(Refer Slide Time: 17:46)



So, the Chandy-Lamport algorithm can be initiated by any process by executing the marker sending rule by which it records its local state and it will send the marker on each outgoing channels. So, a process executes the marker receiving rule on receiving the marker. If the process has not yet recorded its local state, will record the state of a channel on which the marker is received as empty and executes the marker sending rule to record its local state. Therefore, this algorithm is a recursive algorithm. Now, this algorithm will terminate after each process has received a marker on all of its incoming channels.

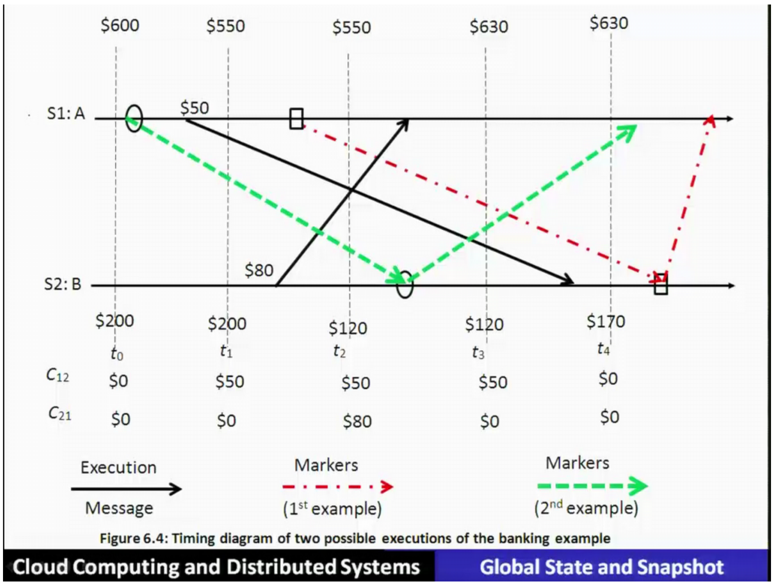
(Refer Slide Time: 18:28)



Let us see the algorithm in more details. This algorithm is called a Chandy-Lamport algorithm. So, marker sending rule for a process i has two steps; the first step says that process i will record its state, second rule says that for each outgoing channel C on which the marker has not been sent, i sends a marker along C before i sends the further messages along C.

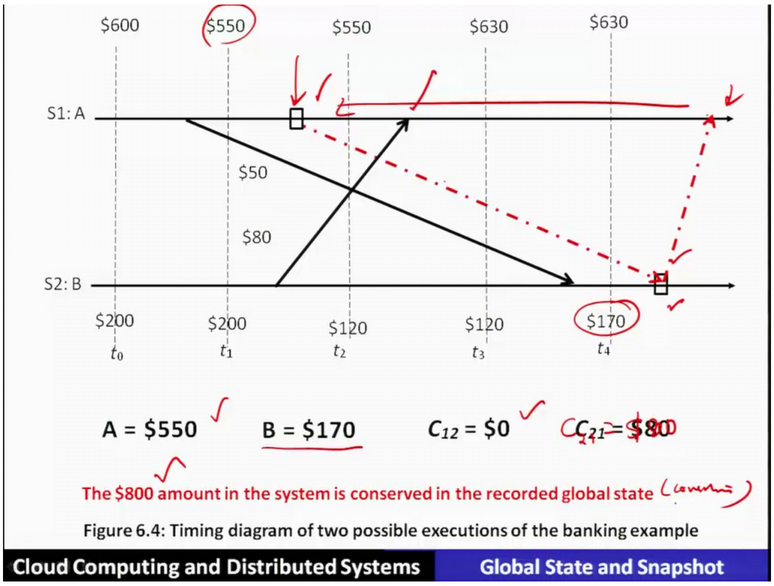
The marker receiving rule for a process j or receiving the marker along channel C, if j has not recorded its state, then record the state of a channel is an empty, and follow the marker sending rule that means, marker sending rule means again it will follow this; it is a recursive call. So, follow the marker sending rule that means, then it will record its state and send the marker to another level of outgoing channels, if it is not. Now, if j has being recorded, then it will record the state of a state of channel C as the set of messages received along C after j's state was recorded, and before j received the marker along C.

(Refer Slide Time: 19:52)



So, we will see this particular working of Chandy-Lamport's algorithm through an example. The same set of the example, but there are two different initiation, which is shown here; this is one set of initiation, this is another set of initiation.

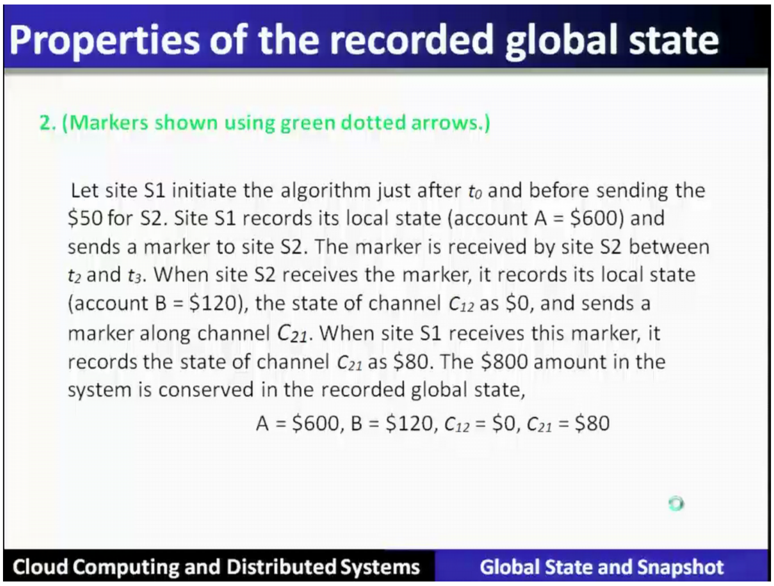
(Refer Slide Time: 20:11)



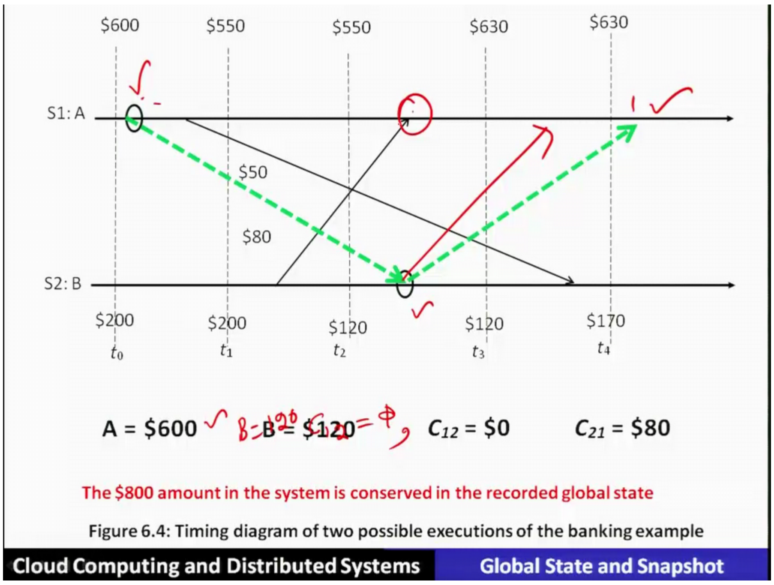
So, when this particular algorithm is initiated at this point of time, so as per the algorithm is concerned, it will record its state that is A will be recorded as 550, and it will send the marker on this channel. And when the marker is received, then what it will do, it will record its state as empty. And then, marker sending rule it will apply out of that, it will record its state as 170 and send the marker further when the marker is received at this point, then since it has already recorded its state. So, what it will do, it will check all the messages, since last time it has recorded and up to the receive the marker any messages, it has received in the communication channel. Yes, this message is received.

So, the state of a channel C 2 1 that will be denoted as dollar 80, so that will be the recorded snapshot. So, if you sum up all the values, it basically will give you that 800 that is the total amount is preserved, hence it is a consistent global state recording algorithm, which is recording the state, which is global state, which is a consistent state.

(Refer Slide Time: 21:51)



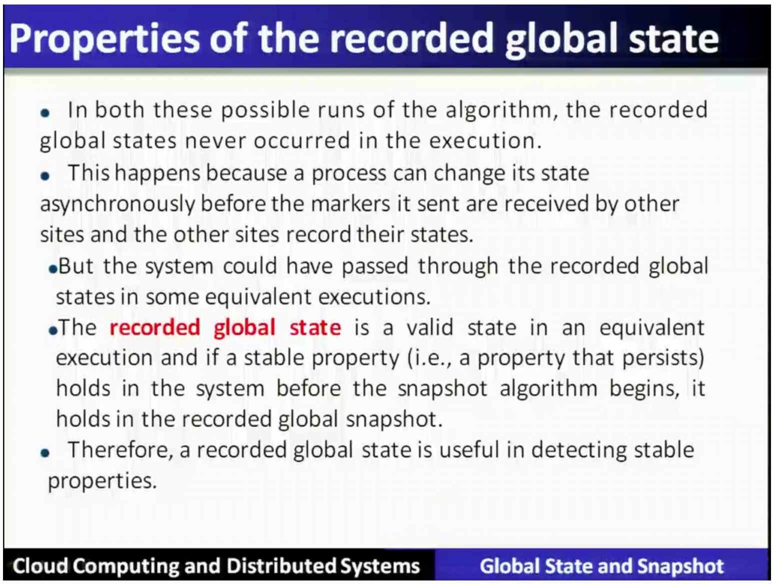
(Refer Slide Time: 21:53)



Now, let us see another example. In this example, the algorithm is initiated at this instant of time that is after just after t 0. Now, as far as the marker, the rule one says that it will record its state that is A will be 600, and then it will send the marker. So, when the marker is received by a site S 2, it will record its; it will record its state of a channel C 1 2 as empty. And then, it will apply the marker sending rule, and it will then it will record its state that is B will be equal to 120, and it will send the channel; and it will send the marker on the further channels, like this.

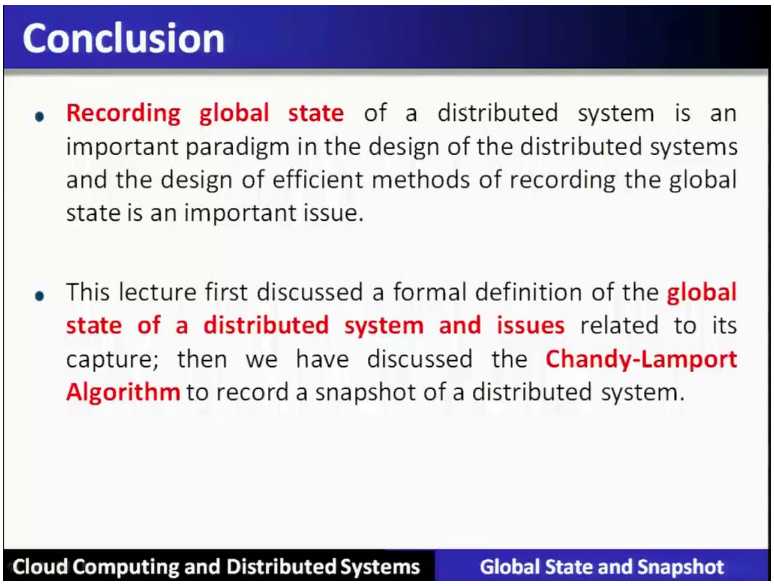
Now, when the marker is received, since this state is already recorded, since the last time up to this marker, any message which is being arrived, that will be in the state of a channel. So, state of a channel will be recorded as the 80.

(Refer Slide Time: 23:11)



So, this completes the illustration of this particular working of this algorithm. So, we have seen that it preserves all the properties, which we have defined for the consistent global snapshot in this particular algorithm.

(Refer Slide Time: 23:30)



Conclusion. Recording the global state of a distributed system is an important paradigm in the design of a distributed system and the cloud system. And the design of efficient method for recording global state is also an important issue. In this lecture, we have formally defined the global snapshot the state of a global state, and snapshot recording algorithm using Chandy-Lamport's algorithm.

Thank you.