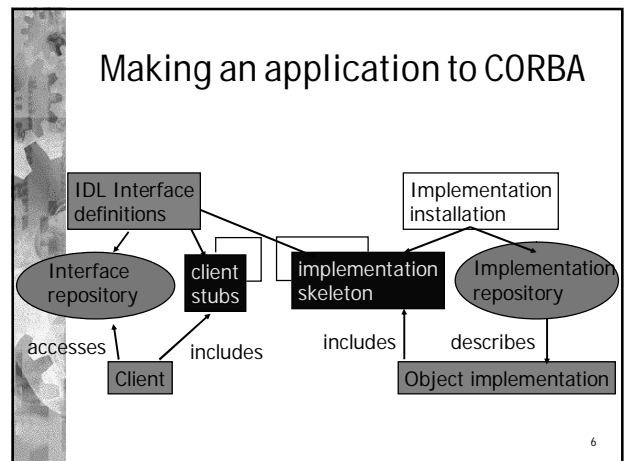
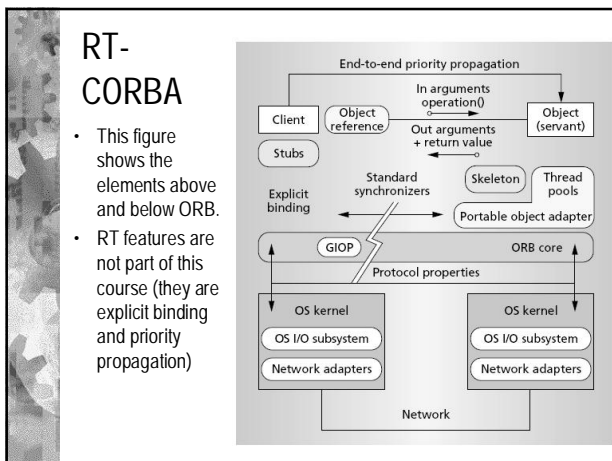
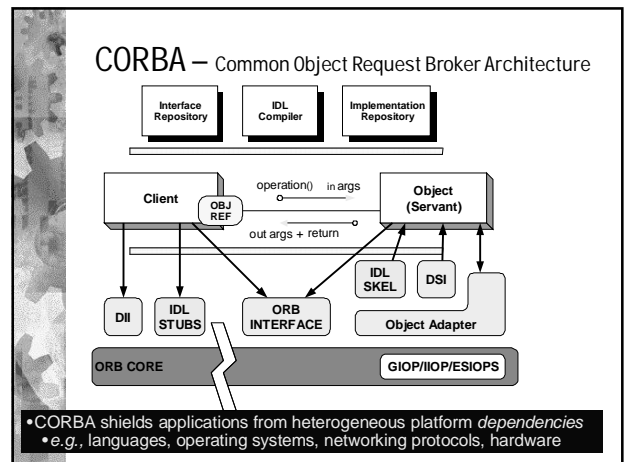
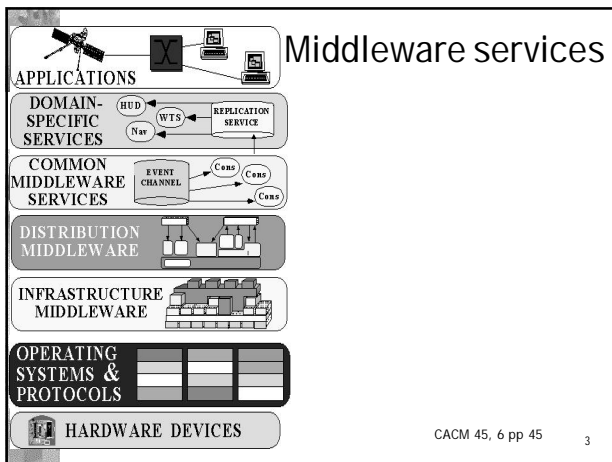
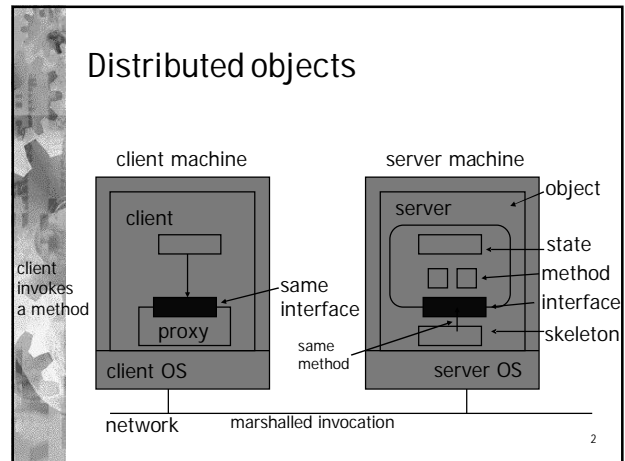


WEEK 10

## Distribution issues, global state, clusters, CORBA, etc

Stallings, Chapters 14 & 15 + Appendix B  
Prev. edition; Chapters 13&14

1



## Interface definition: IDL

- IDL

```

module HelloApp{
    interface Hello {
        string sayHello();
    };
};
    
```

7

## Client

```

Import HelloApp.*;
//contains generated stubs
Import org.omg.Cosnaming.*;
Import org.omg.CORBA.*;
//must be in all CORBA applications

Public class HelloClient {
    public static void main(String args[]) {
        try {
            // create and initialize the ORB
            ORB orb = ORB.init(args, null);
            // get the root naming context
            org.omg.CORBA.object objRef =
                = orb.resolve_initial_references("NameService");
            NamingContext ncRef =
                NamingContextHelper.narrow(objRef);
        }
    }
}
    
```

1/2  
8

## Client

```

// Resolve the object reference in naming
NameComponent nc = new NameComponent ("Hello", "");
NameComponent path[] = {nc};
// get the stub
Hello HelloRef =
    HelloHelper.narrow(ncRef.resolve(path));
// Call the Hello server object and print results
String hello = helloRef.sayHello();
system.out.println(hello);
} catch (Exception e) {
    System.out.println("Error : " + e);
    e.printStackTrace(System.out);
}
}
    
```

2/2  
9

## Server object

```

Import HelloApp.*;
// contains generated stubs
Import org.omg.Cosnaming.*;
Import org.omg.CORBA.*;

Public class HelloServer {
    public static void main(String args[]);
    //registration, entering server loop
    try{
        ORB orb = ORB.init(args, null);
    }
}
    
```

1/3  
10

## Server object

```

// create the servant and register with orb
HelloServant helloRef = new HelloServant();
orb.connect(helloRef);
//get root naming context
org.omg.CORBA.object objRef =
    orb.resolve_initial_references("NameService");
NamingContext ncRef =
    NamingContextHelper.narrow(objRef);
//bind the object reference to name
NameComponent nc = new NameComponent("Hello", "");
NameComponent path[] = {nc};
ncRef.rebind(path, helloRef);
    
```

2/3  
11

## Server object

```

//wait for invocations from clients
java.lang.Object sync = new java.lang.Object ();
synchronized (sync) {
    sync.wait();
}
catch (Exception e) {
    system.err.println("Error : " + s);
    s.printStackTrace (System.out);}}

//the actual service
Class HelloServant extends _HelloImplBase {
    public string sayHello () {
        return "\nHello world!\n";}}
    
```

3/3  
12

## More information

- CORBA Tutorial
  - Schmidt, D., CORBA Tutorial.  
<http://www.eng.uci.edu/~schmidt/PDF/corba4.pdf>
- CORBA-sivustoja
  - <http://www.corba.org/>
  - <http://www.cs.wustl.edu/~schmidt/corba.html>
  - <http://www.puder.org/corba/>

13

## Clusters

14

## Cluster Computer Architecture

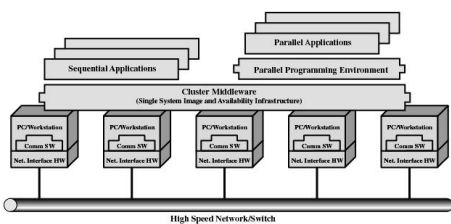


Figure 14.14 Cluster Computer Architecture [BUY99a]

15

## Classifications

Clustering Method	Description	Benefits	Limitations
<b>Passive Standby</b>	A secondary server takes over in case of primary server failure.	Easy to implement.	High cost because the secondary server is unavailable for other processing tasks.
<b>Active Secondary:</b>	The secondary server is also used for processing tasks.	Reduced cost because secondary servers can be used for processing.	Increased complexity.
- Separate Servers	Separate servers have their own disks. Data is continuously copied from primary to secondary server.	High availability.	High network and server overhead due to copying operations.
- Servers Connected to Disks	Servers are cabled to the same disks, but each server owns its disks. If one server fails, its disks are taken over by the other server.	Reduced network and server overhead due to elimination of copying operations.	Usually requires disk mirroring or RAID technology to compensate for risk of disk failure.
- Servers Share Disks	Multiple servers simultaneously share access to disks.	Low network and server overhead. Reduced risk of downtime caused by disk failure.	Requires lock manager software. Usually used with disk mirroring or RAID technology.

## Windows Cluster Service

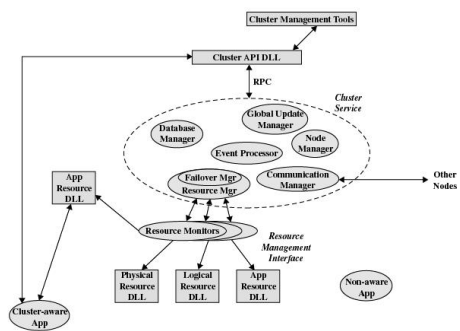
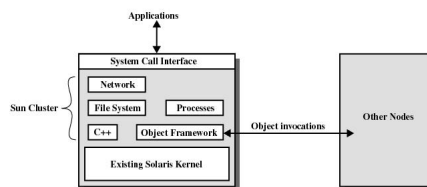


Figure 14.15 Windows Cluster Server Block Diagram [SHOR97]

17

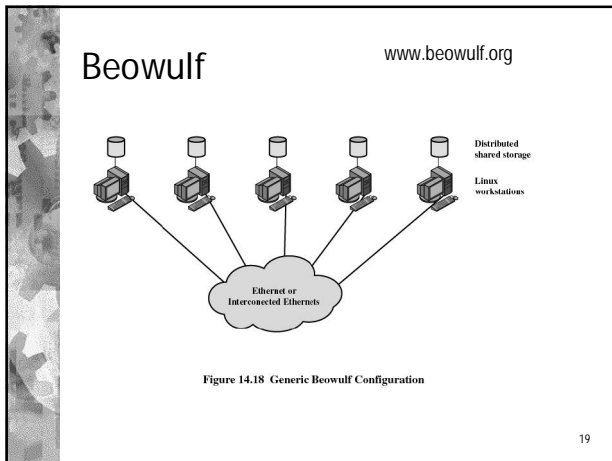
## Sun Cluster



- Support for objects and their communication
- Process management
- Distributed file system

Figure 14.16 Sun Cluster

18

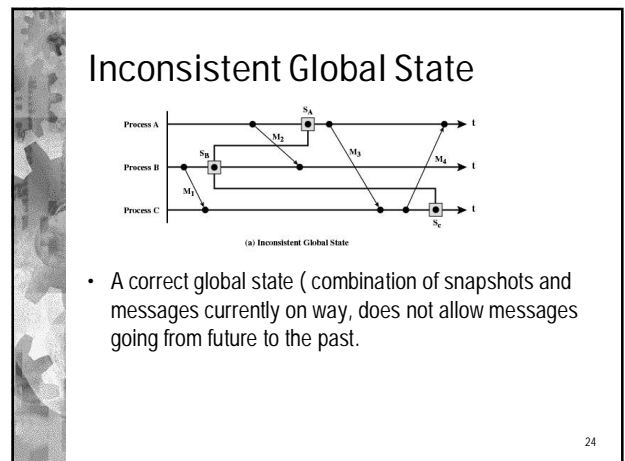
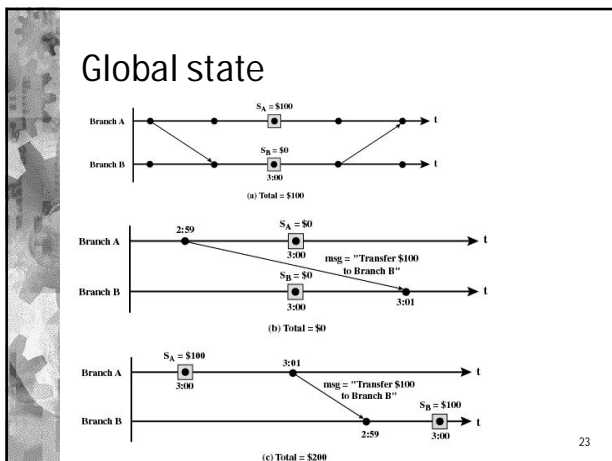
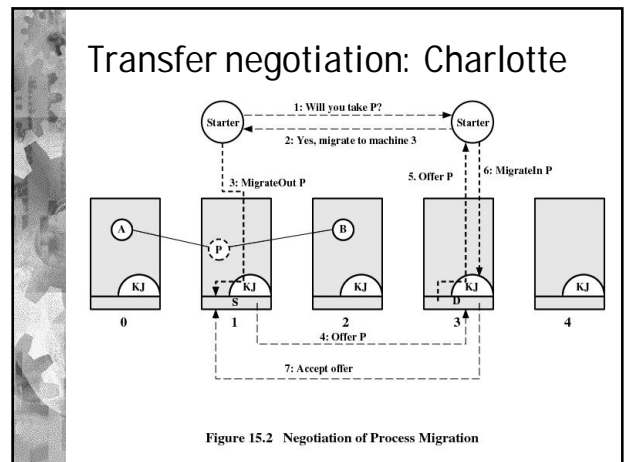
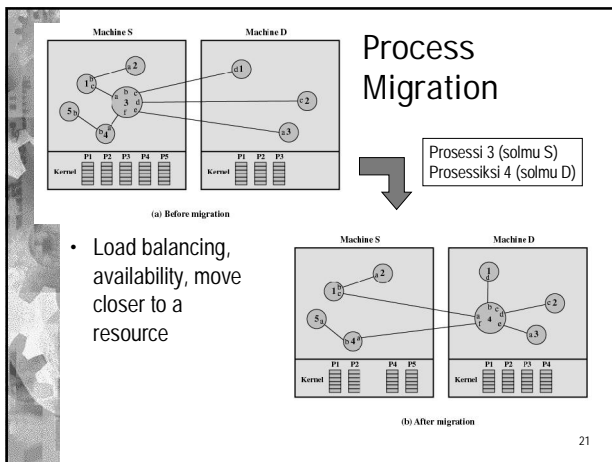


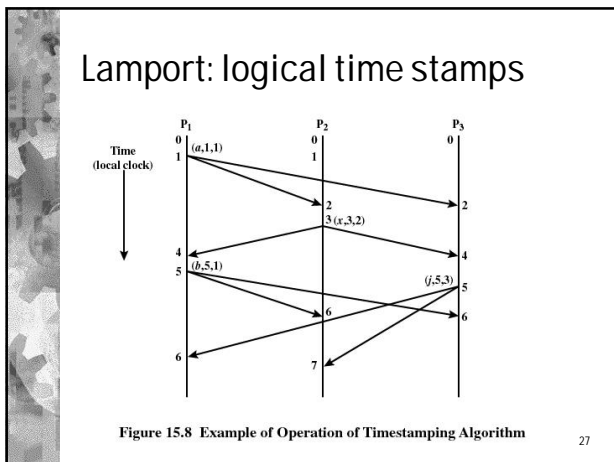
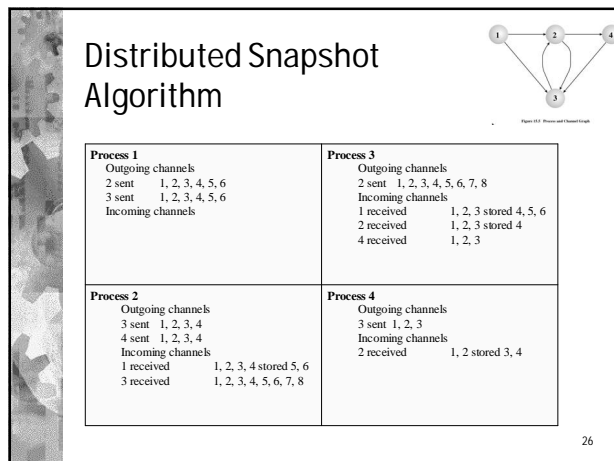
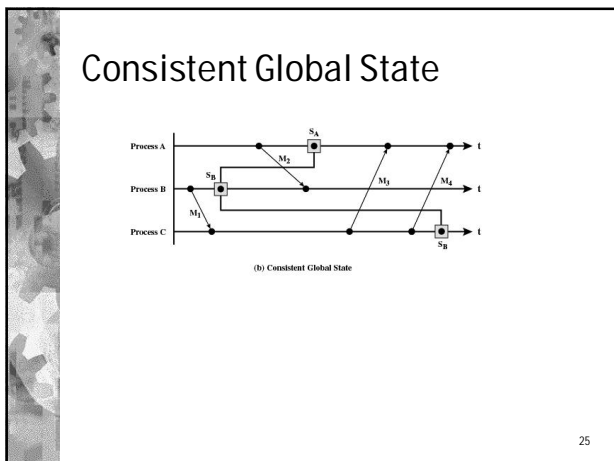
LUENTO 20

## Distributed Process Management

### Stallings, chapter 15

20





### Distributed mutual exclusion and deadlocks

28

