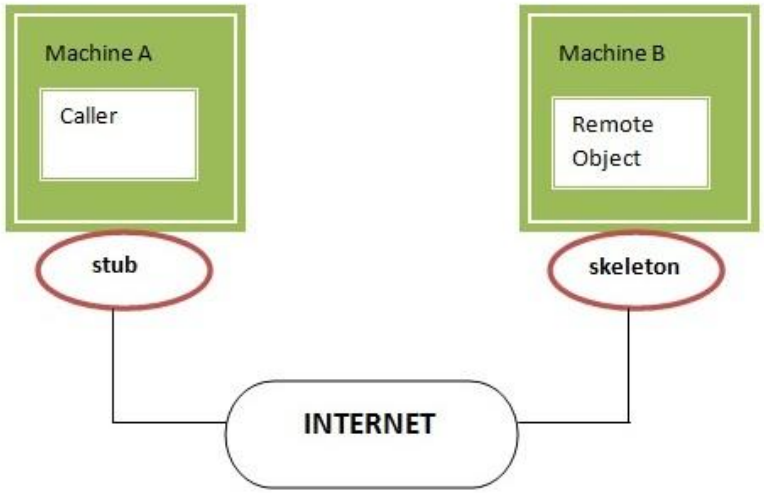
JAVA Programming

Section 9

RMI

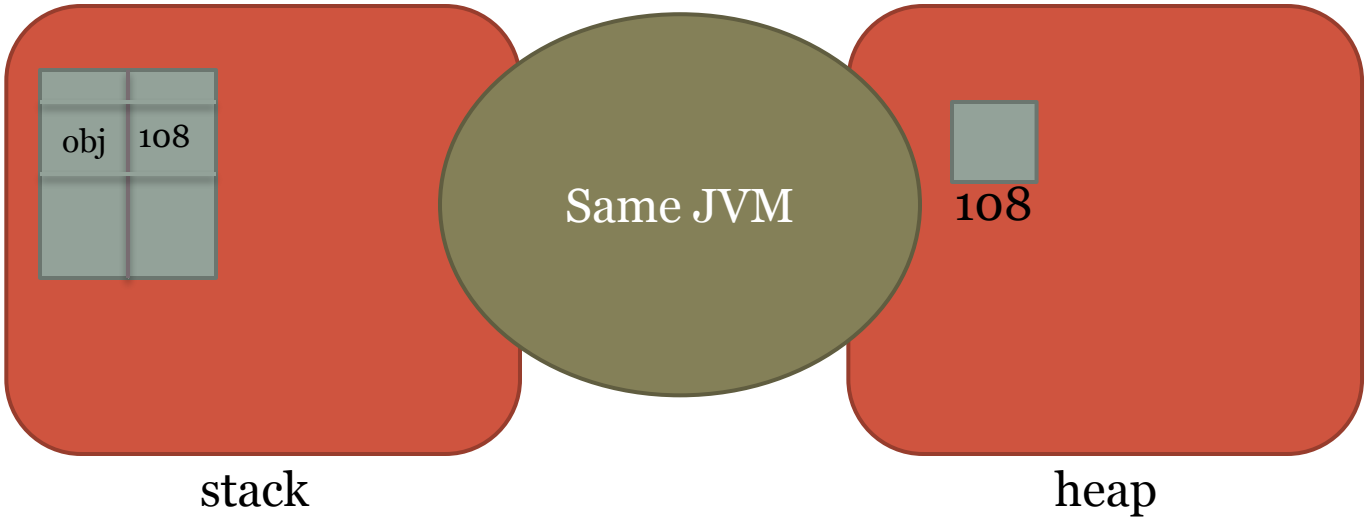
- RMI (remote method invocation) is an API that provides a mechanism to create distributed applications in java.

- RMI applications often comprise two separate programs, a server and a client. A typical server program creates some remote objects, makes references to these objects accessible, and waits for clients to invoke methods on these objects.

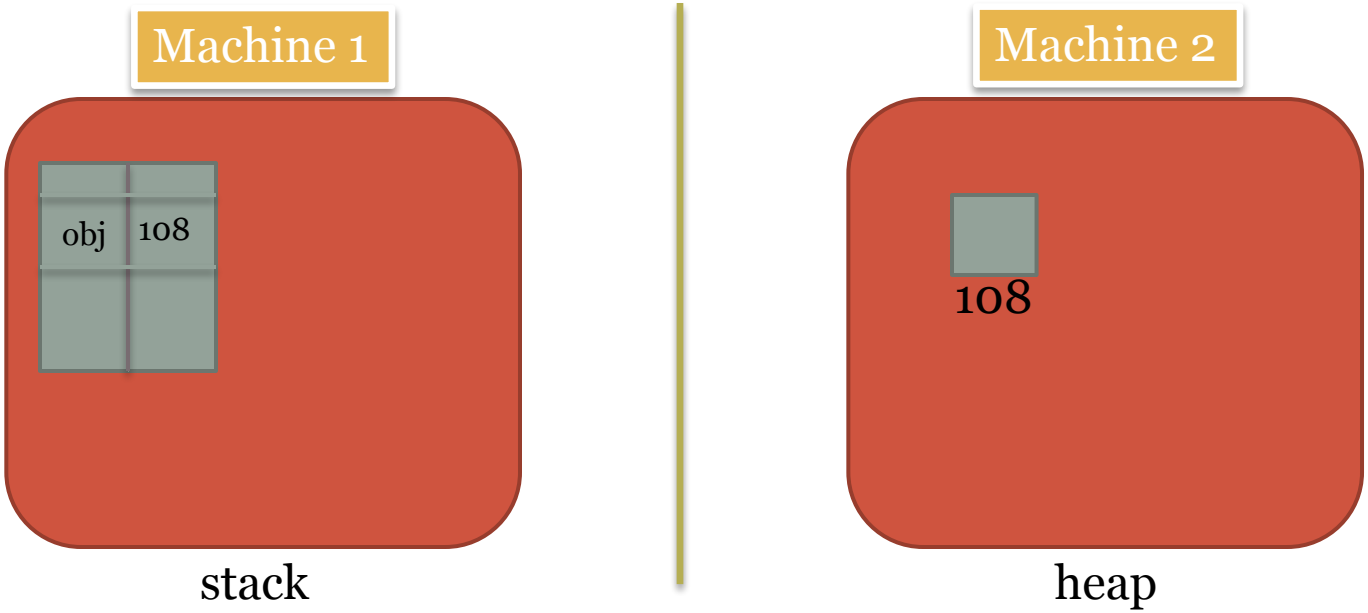


```
Class A {
int a,b;
display();
Add();
}
```

```
Class B {
main()
{
    A obj= new A();
     obj.display();
}
}
```



- What if we have two different machines?



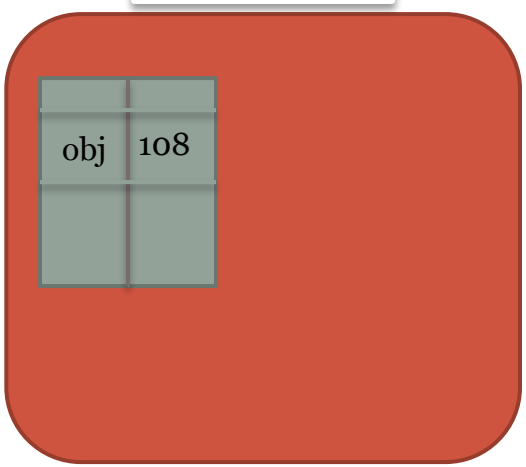
We need to invoke remote method (RMI)

Registry

Object	caption
obj	A

client

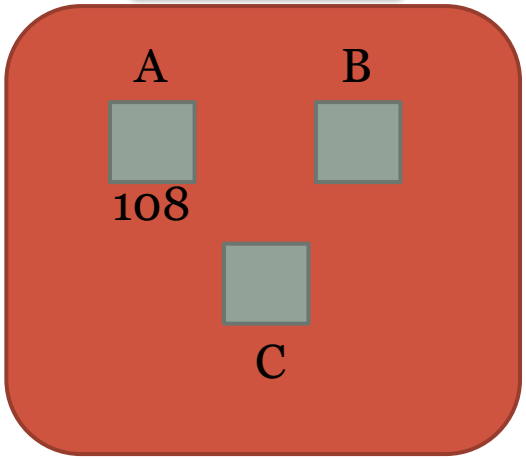
Machine 1



stack

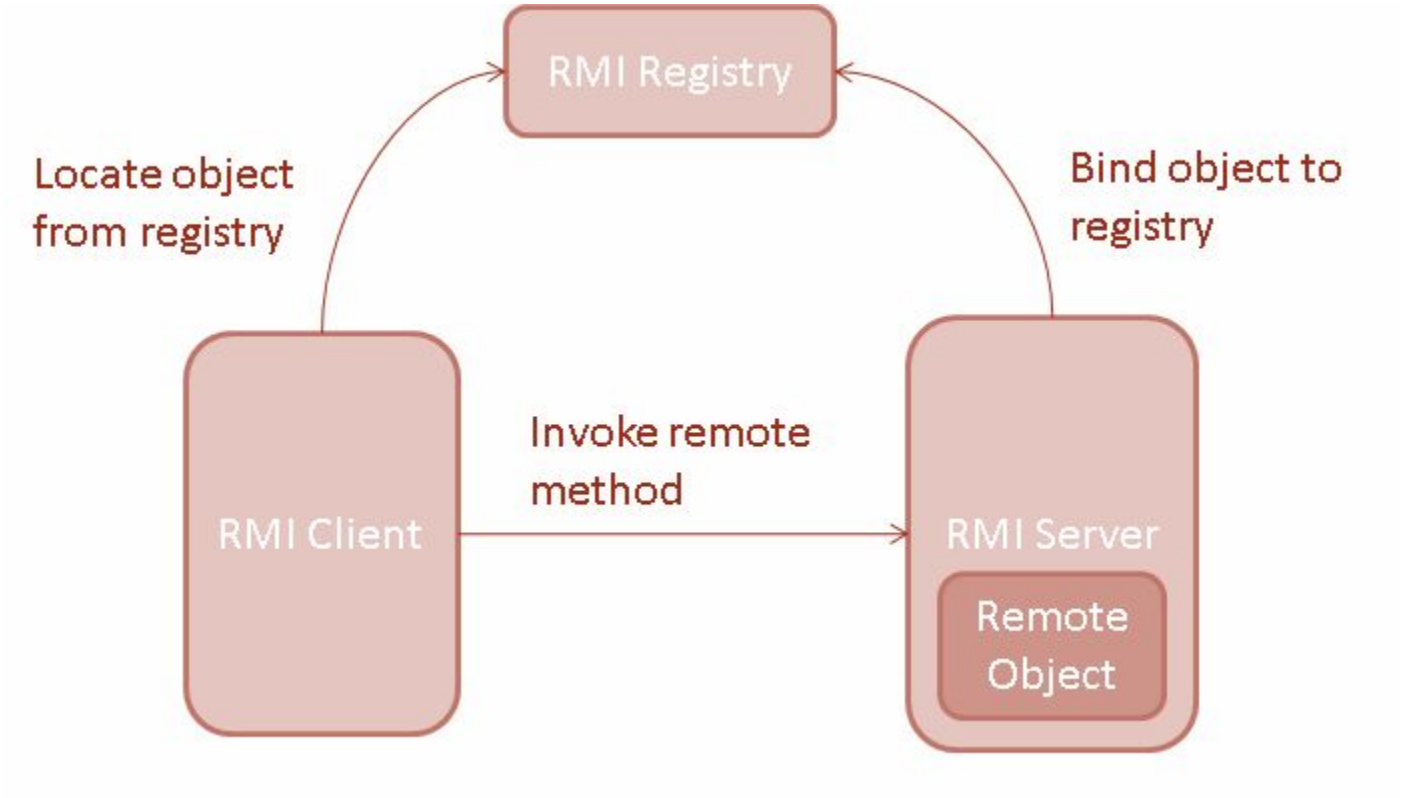
Server

Machine 2



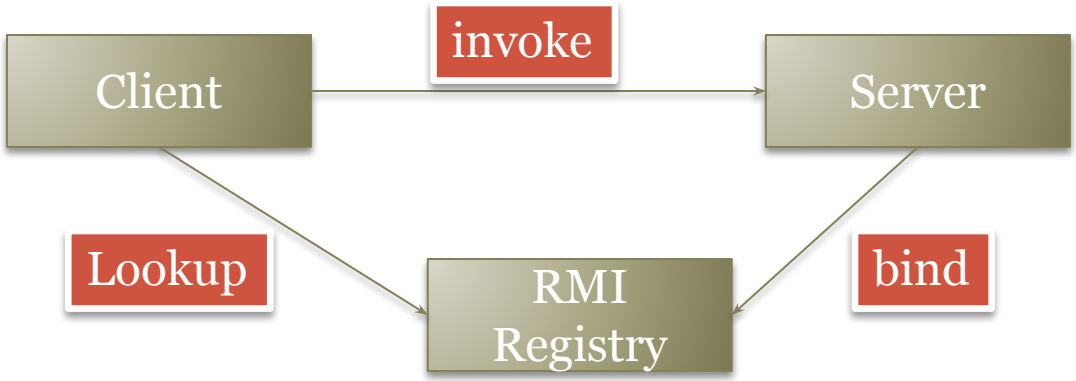
heap

RMI


RMI Registry
Locate object from registry
Bind object to registry
RMI Client
Invoke remote method
RMI Server
Remote Object

RMI Application steps:

- Create remote interface.
  (must be available for both client and server)
- Implement the interface methods using a class.
  (for server side)
- Create registry at server and bind captions.
- Get method at client looking up registry.



Creating Remote Interface

*RMI_interface.java | *RMI_remote.java | RMI_server.java | Client.java

```
1  import java.rmi.Remote;
2  import java.rmi.RemoteException;
3
4  public interface RMI_interface extends Remote {
5
6      public double add(int a, int b) throws RemoteException;
7      public double sub(int a, int b) throws RemoteException;
8      public double Mul(int a, int b) throws RemoteException;
9      public double Div(int a, int b) throws RemoteException;
10
11 }
12
13
```

Implementing Remote Interface Methods

*RMI_interface.java | *RMI_remote.java | RMI_server.java | Client.java

```
3
4  public class RMI_remote extends UnicastRemoteObject implements RMI_interface{
5
6      protected RMI_remote() throws RemoteException {
7          super();
8          // TODO Auto-generated constructor stub
9      }
10     @Override
11     public double add(int a, int b) throws RemoteException {
12         // TODO Auto-generated method stub
13         return a+b;
14     }
15     @Override
16     public double sub(int a, int b) throws RemoteException {
17         // TODO Auto-generated method stub
18         return a-b;
19     }
20     @Override
21     public double Mul(int a, int b) throws RemoteException {
22         // TODO Auto-generated method stub
23         return a*b;
24     }
25     @Override
26     public double Div(int a, int b) throws RemoteException {
27         // TODO Auto-generated method stub
28         return a/b;
29     }
```

Server Class

*RMI_interface.java | *RMI_remote.java | *RMI_server.java | Client.java

```
1  import java.rmi.AlreadyBoundException;
2  import java.rmi.RemoteException;
3  import java.rmi.registry.LocateRegistry;
4  import java.rmi.registry.Registry;
5
6  public class RMI_server {
7
8      public static void main(String[] args) throws RemoteException, AlreadyBoundException {
9          // TODO Auto-generated method stub
10
11         RMI_remote re = new RMI_remote();
12         Registry r = LocateRegistry.createRegistry(1099);
13         r.bind("ADD", re);
14         System.out.println("Server Started....");
15
16     }
17
18 }
19
```

Client

RMI_interface.java | RMI_remote.java | RMI_server.java | Client.java

```java
import java.rmi.NotBoundException;
import java.rmi.RemoteException;
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;

public class Client {

    public static void main(String[] args) throws RemoteException, NotBoundException {
        // TODO Auto-generated method stub

        Registry r = LocateRegistry.getRegistry("localhost", 1099);

        RMI_interface ri = (RMI_interface) r.lookup("ADD");
        System.out.println(ri.add(10, 5));
        System.out.println(ri.sub(10, 5));
        System.out.println(ri.Mul(10, 5));
        System.out.println(ri.Div(10, 5));

    }

}
```

Problems | Javadoc | Declaration | Console

<terminated> Client (5) [Java Application] C:\Users\hp\.p2\pool\plugins\org

```
15.0
5.0
50.0
2.0
```