

(Pronounced “om” or “aum”)
(Symbolizing ALL in ONE & ONE in ALL)

Dedicated

to

10

My Mother & Father

&

to

**My Mystic Country
Bharat**

20

Rajan's Enterprise JavaBeans Session Notes (Draft)

Inetsolv Solutions (February 2004 Evening Batch)

10

*Noted, Compiled and Edited
By
Rajan.V
Developer & Quality Engineer*

mailbox : luckyrajan@linux.net

20

File Name : Rajan's-EJB-Session-Notes-1.4(Draft).pdf
Size : 735.0 KB (xxxxxxxx bytes)
Started on : Friday 18th June 2004
Last Modified on : Friday 16th July 2004 01:40:13 pm
(If there is variation in size you are requested to scan this document for presence of virus)

30

© **Copyright Notice:** This document is available for free distribution **AS IS**. You are free to take printout of the Document **AS IS** and re-distribute **AS IS**. Strict action will be taken against people not adhering to the notice. Please intimate us of how useful the document has been for you and any enhancements you feel will be helpful, which will be considered for addition in future releases. Send comments/feedback to my mailbox.

40

My Thoughts:

Sitting amongst 300+ developers attending the sessions at InetSolv I have been toying with the idea for last few months of coming out with such a document which may be useful for all developers hungry to know more and develop world-class solutions. My friend Dhiraj's departure for his project presentation was just an excuse to kick-start this project which ended in tangible results. As I started off I did not have even an inkling as to the plan of action of how the document would shape up. With time and constant ideas flowing in I had to keep changing the document, enhancing the original text doc planned. I have taken into account meticulously every point that might be of help to the reader. I have also taken care so as to rule out any errors including spelling errors. 10

Please intimate me of any errors otherwise that might have cropped up in the document and any enhancements that you feel might be relevant to the topic of discussion.

Appropriate tables, diagrams, notes has been added by me which I felt might suffice the topic in discussion which are marked as "MyTable:", "MyDiagram:", "MyNotes:". Some portion of the main notes may also not conform to the session notes taken by my colleagues, since I have added some notes (some of lines I have picked up directly from the java documentations-like method signatures etc). You may look forward to future releases which may contain relevant diagrams, code, notes picked from sites, forums, books, magazines. 20

I hope these notes will supplement the discussions in the sessions and will be immensely useful for all the developers. 30

For my new friends who are going through this document, who may or might have joined in fresh batches, I would like to clarify that the sequence of sessions and the contents covered may vary, since the contents and sequence of sessions are being changed & upgraded continuously. So the contents of this document is in no way an indication of the contents for your sessions. In spite of the changes expected I feel this document would supplement and would be of immense help in your explorations into various technologies. 40

All the BEST WISHES for Excellent Quality Software Development to all.

Thanks

Rajan. V

luckyrajan@linux.net

My Acknowledgments:

I take this opportunity to bow and express my gratitude to my mother and father, to all the Masters, Guides visible and invisible, known and unknown who guided, supported and corrected me in so many ways, directly and indirectly on so many occasions.

I would like to thank all the support extended by Mr Suresh Posani J2EE-Faculty and CEO of InetSolv. I would also like to take this opportunity to thank all the open source gurus and contributors who inspired me to come out with this document. I would like to specially thank the person behind free software foundation Mr Richard Stallman and other GNU/Linux stalwarts without whose undeterred campaigning we would not be seeing so many tools and solutions. I think Microsoft and all other proprietary software vendors created the right climate for the growth of GNU/Linux and Open source movement, so they too deserve special mention here.

Once more very special thanks to existence in all forms to give me this opportunity and supporting me to make it viable. I would love to mention many, but choose to Specially mention My Parents who were there steadfastly supporting and encouraging throughout and My Brother and his Gift for us - Our Home Computer System and Softwares that gave me the company all along.

My Request:

Join the developers group at yahoo : <http://groups.yahoo.com/group/idevcircle> and post all your technical queries, share your knowledge, help other developers.

Visit Group URL <http://geocities.com/idevcircle> and contribute 4 developers and to register with our group.

In case of problems send a message to my mailbox or send mail to idevcircle-owner@yahooogroups.com.

Sessions on Enterprise JavaBeans (EJB) – June 2004 (February 2004 Evening Batch)

<i>Date</i>	<i>Day</i>	<i>Session</i>	<i>Topic</i>	<i>Page No</i>
			My Thoughts	
			My Acknowledgments	
			Index of Sessions	
16th	Wed	1/25	Intro to OMTB project	6
17th	Thu	2/25	EJB Intro, Dev of EJB: SessionBean-Stateless using JbuilderX.	10
18th	Fri	3/25	Dev of EJB without IDE Tools (Manual)	15
19th	Sat	4/25	Execution from Client, Session beans method flow	
21th	Mon	5/25	Stateful Session Beans	
22th	Tue	6/25	Stateless vs Stateful	
23th	Wed	7/25	JDBCDirectRead and DTO Design pattern	
24th	Thu	8/25	Container managed Transaction using JTA, Transactional Attributes	
25th	Fri	9/25	Study of Various TA' ,Local interface-How?	
26th	Sat	10/25	Local vs Remote interface, Security	
28th	Mon	11/25	Entity Beans – Using JbuilderX	
29th	Tue	12/25	Finders, EJBQL	
30th	Wed	13/25	Accessing EJB from JSP, BMP Bean -How	
1st	Thu	14/25	Entity Bean – How, EJBHome/EJBSelect	
2nd	Fri	15/25	Container Mgd Relationship, One-to-One, One-to-Many Relationship(Uni,Bi), Demo.	
3rd	Sat	16/25	Primary key class – when & how	
5th	Mon	17/25	Project Dev Issues, Bug Life Cycle, Manual creation of EntityBean, Lazy Loading Tech.	
6th	Tue	18/25	Entity Bean State Diagram, Activation & Passivation, home.create(), home.findXxx(),	
7th	Wed	19/25	Session Synchronization Interface, Issues pertaining to Returning “this” and Reentrant	

<i>Date</i>	<i>Day</i>	<i>Session</i>	<i>Topic</i>	<i>Page No</i>
8th	Thur	20/25	EJBHome/EJBSelect,Handle,Environment Entries	
9th	Fri	21/25	Session Facade & EJBHome Caching d/n Pattern	
12th	Mon	22/25	Message Queue,PTP,Pub & Sub,Queue/Topic	
13th	Tue	23/25	Msg Producer & Consumer, Sender/Receiver , Subscriber/Publisher	
14th	Wed	24/25	Priority of msgs,Setting delivery modes,ACK	
15th	Thur	25/25	Transaction mgt in Queue,durable Subs., Message Driver Beans.	
			References	
			Afterthought	
			About Myself	
			Guidelines to Register with idevcircle	
			Announcements	

Prerequisite knowledge required:

- 1.Core Java
- 2.Multi Threading
- 3.Network Programming - any language (Java preferred)
- 4.JDBC
- 5.JNDI
- 6.XML
- 7.Serv lets/ JSP
- 8.RMI

10

For Documentations,Screen-Shots,Specs,Example Codes and for other Software Contact InetSolv Office/visit <http://www.inetsolv.info> .

20

Enterprise JavaBeans

Session 1 of

Dated: 16th June 2004 Wednesday

Topic: Intro to OMTB project

(Online Movie Ticket Booking Project)

10

20

30

Refer back(for better understanding)

- 1.
- 2.
- 3.

In Enterprise applications Stateful Session Beans are mostly used.

Online Movie Ticket Booking Project:

Screens	Operations
- Login	login
- Theatre	Add Theatre, Modify Theatre, Delete Theatre
- Ticket	Book Ticket, Cancel Ticket
- User	Register, UnRegister
- Reports	Users List, Theatre List, Booked Tickets List

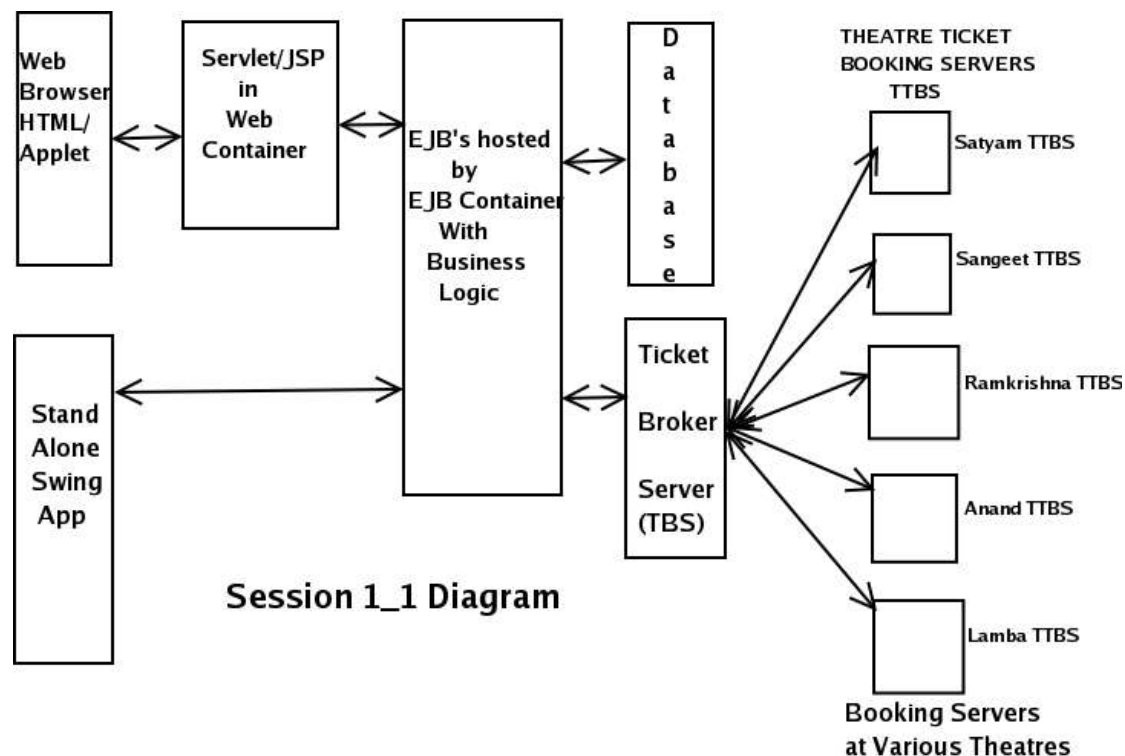
10

MyNote: In Reports we can have Movie wise, Theatre wise, Show-wise listing of Users

In every project we have to generate reports that can be printed or viewed online. If simple we can just write a series of out.println statements in our program to generate a report. If a report is a complicated report we can use a report writing tool like

20

- Jasper Reports (Open Source) - From Apache
- Crystal Reports



Using EJB technology we can develop a business application with GUI application as frontend or a Web application accessed through a browser as a frontend. In any business application we need to store/manipulate the data of various entities. Eg

In case of **Ticket Booking application** the entities are

- tickets
- users
- theatres

10

In **school admin system** the entities are

- teachers
- students
- subjects

In an **Income tax application** the entities are

- IT officers
- IT payers
- banks
- tax-payment details
- tax-refund details

20

Note: Following things are not allowed in EJB

Files for storage not allowed.

Threading not allowed

We cannot set <connection>.autocommit as true

We should never think of using ejb applications to develop the business application used by less number of concurrent clients. 30



Session 1_2 Diagram

EJB uses RMI over IIOP. By using this we can very easily build scalable and failsafe applications.

EJB technology simplifies the development of a complex business application that needs to be scalable, reliable, and failsafe

Enterprise JavaBeans

Session 2 of

Dated: 17th June 2004

Topic: EJB Intro, Development of EJB

SessionBean-Stateless using JbuilderX.

10

20

Refer back(for better understanding)

1. Session 1 of EJB

2. Documentations, screen shots provided by Inetsolv.

3.

30

When we start the development of ejb we need not develop everything on our own. We can purchase some software components, develop some software components and assemble them together to create our business application. Javasoft has specified the following rules as part of ejb specification.

1 **Bean Developer/Provider** : He is responsible for developing the ejb component.

2 **Application Assembler** : Responsible for assembling the ejb's and develop the frontend using jsp/servlet, or a gui application. 10

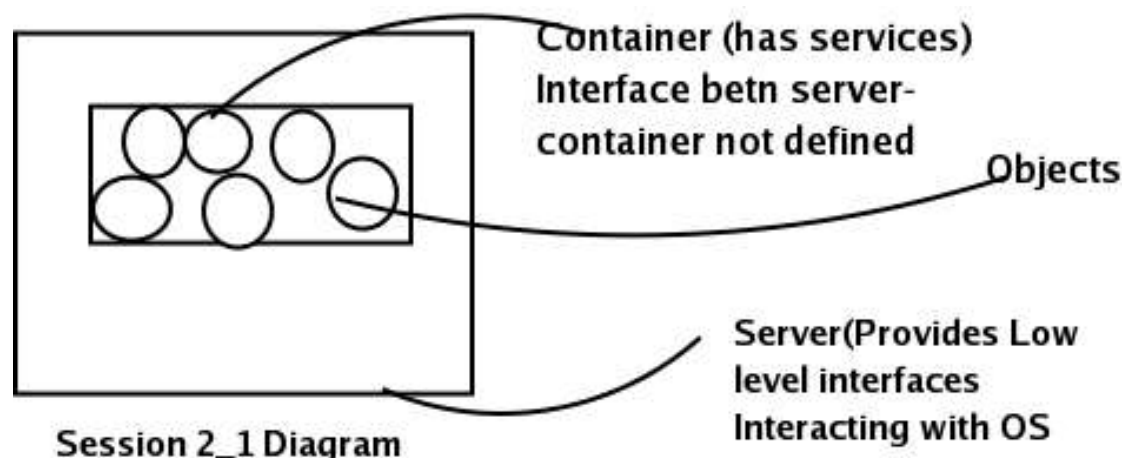
3 **Deployer** : Is responsible for deciding about the configuration of the ejb server and deploy the ejb on the server.

4 **System Admin** : Setting up hardware and network and continuously monitoring the servers and take necessary actions. 20

5 **EJB Server Container vendor** : Responsible for providing server software.

6 **Tool Vendors** : Responsible for supplying the IDE's eg. Weblogic workshop from bea, Sun One Studio from sun, Jbuilder from Borland

Javasoft has not clearly defined server/container roles separately. Today most of the developers treat ejb server same as ejb container. Clearly interfacing between server and container not defined. 30



There are 5 types of ejb's supported in ejb 2.0. Not all these types are used in the application development.

Session Bean : Used for implementing business process

1 **Stateful session bean**

2 **Stateless Session bean**

Entity Bean : Used to manage business data (i.e business entities)

BMP Beans(bean managed presistence):we need to write jdbc code for managing data. 10

CMP Beans(Container managed presistence):Code to manage the data will be generated by the tools.

Message Driven Beans : Used for implementing business process(i.e same as sessio beans etc) This will be driven by messages posted on queue or topic.

Some of the developers are developing the ejb projects without using entity beans. 20

A component is a piece of software developed using a set of rules. for developing various types of ejb's we need to follow the set of rules given as part of ejb specification.

Flavours of Jbuilder -

1 Foundation- Minimum set of features

2 Enterprise - Edition Features like web development, ejb development on multiple servers(iplanet, jboss, weblogic, websphere) 30

MyNote: Check <http://www.borland.com> for other flavours of jbuilder.

Steps to configure jbuilder with weblogic .

Step 1 In Menu click Tools/Configure server...

Step 2 Choose appropriate server.

Choose enable server. 40

Select home directory

(In our case it is) [d:/bea/weblogic/server](#)

Main class - weblogic.server

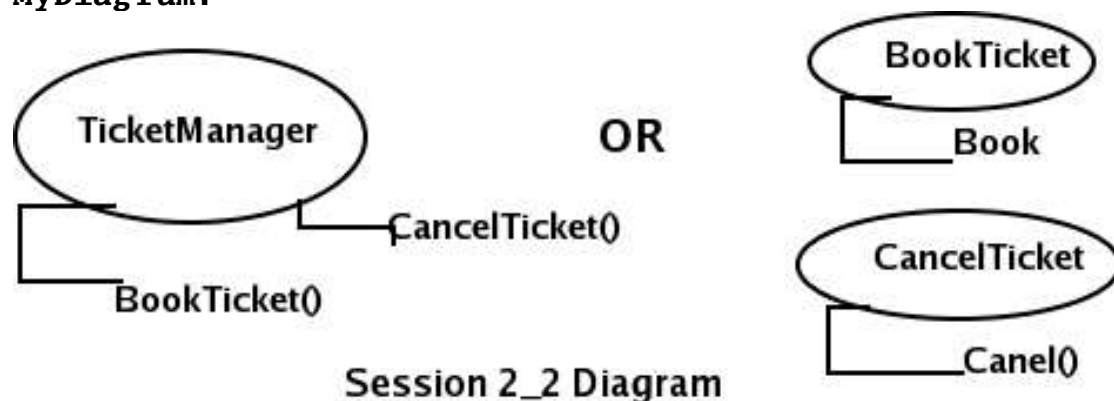
Working directory - /bea/user_projects/inetsolv

In custom tab provide user/password

In case of our ticket booking application we need to implement the business objects for booking the tickets and cancelling the tickets. Since this is business logic we can either use session beans or message driven beans.

In this case we can develop one ejb providing both the methods or two ejb's with one method each

MyDiagram:



When we want to develop an ejb with jbuilder kind of tool first decide about the business methods, parameters to business methods, the return types and the exceptions thrown (like **ticketsNAException**, **timeoutForCancellationException** etc)

10

Procedure to Develop Session bean with JbuilderX

- 1 Create project directory
- 2 File/New project - we get project wizard
 - enter project name
 - Choose project directory
 - Finish
 (click next to set other parameters)
- 3 file/New
 - Select EJB Module
 - Choose appropriate server say ok
- 4 Jbuilder opens EJB Designer. This can be used to visually design (graphically) our ejb.

20

Jbuilder generates the XML files called as deployment descriptors in META-INF directory.

30

To develop stateless session beans Select session bean This displays ejb in the designer.

MyDiagram:

Ticket Manage	-----right click /add/method

Session Context	Enter - method name

SetsessionCont ext	Enter - return type

ejbCreate	Enter - input params(comma separated)

	Enter - interface localremote

10

Irrespective of server we use to deploy we need to provide **ejb-jar.xml file**. In this file the information about the name of the ejb, its type, the names of the classes has to be provided. Depending upon the server we use we may need to provide one or more additional xml files.

If we deploy the ejb we need to generate some additional code which will be generated by a tool provided by container vendor. In Jbuilder we can right click on the module name and choose the option make.

20

This step generates a jar file. To deploy the bean right click on the module name, choose the option deploy.

EAR - Enterprise archive - We can create multiple war files and multiple jar with ejbs and pack them together into a single EAR file.

javax.ejb.EJBObject is a subinterface of java.rmi.Remote. Similarly javax.ejb.EJBHome is a subinterface of java.rmi.Remote.

30

EJB developer need not implement these interfaces. The tool like ejbc generates the classes implementing these interfaces.

40

Enterprise JavaBeans

Session 3 of

Dated: 19th June 2004 Friday

Topic: Dev of EJB without IDE Tools (Manual)

10

20

30

Refer back(for better understanding)

1. Previous Sessions on EJB 1-2

2. RMI Sessions

3. JDBC

4. Factory Pattern

5. Refer to code, Documentations, screen shots provided by Inetsolv.

6. Session on JbuilderX- Enterprise Edition

40

Development of EJB without IDE

Home(Factory)-In EJB we use the objects called as **Home Objects**. Using these objects we can

1. Create the actual bean objects.
2. Find the bean objects(this is in the case of entity beans)

A developer developing business application without using EJB kind of technology need to develop code 10

1. For Load balancing
2. For Fail safe mechanisms
3. To take care of transactions
4. Managing objects(creating/destroying/pooling etc.)
5. To take care of Security

PLUS

6. The business logic according to the business requirements. 20

If we use **EJB** or **MTS** component technology the developer need to write only the business logic. The remaining things will be taken care by the EJB Container or the code that is automatically generated by the tools supplied by the container vendor.

By using EJB we can reduce the total amount of time required for the development of applications. 30

Steps to develop EJB without using an IDE

- 1 Run setenv batch file available in our weblogic project directory

eg If it is installed in C: directory then run
c:\bea\user\projects\inetsolv\setenv

- 2 Set **CLASSPATH** to include current directory

set CLASSPATH=%CLASSPATH%; .

3 Create a our **project working directory**
say mywork and move to this directory

4 Decide about the methods that has to be **exposed**(made available) to the client like in our Online Ticket Booking Management Project

BookTicket()

CancelTicket()

10

are two of functions implementing our business requirements.

5 Develop the remote interface by following the rules given below.

Rule 1: The remote interface must extend from **javax.ejb.EJBObject**

20

Rule 2: As part of the interface declare the methods. These methods can throw any exceptions but all the methods must throw **RemoteException** .

Rule 3: The parameters and return types must be java primitives or serializable objects or remote object
our methods look like

public void BookTicket() throws RemoteException

public void CancelTicket() throws RemoteException

30

To compile: **javac -d . TBRemote.java**

(-d option results in creation of the class files in the appropriate directory reflecting the package structure eg in our case it will be stored in \info\inetsolv\ontb from current project working directory since our package declaration is info.inetsolv.ontb)

(MyNote: Run javac and javac -X to check out further interesting options available

checkout the docs on tools in /j2sdk1.4.1_01/docs/
tooldocs/tools.html to know more about javac and other tools.)

40

6 Develop the Home interface following the rules given below

Rule 1: The interface must extend from **EJBHome**

Rule 2: Provide a create method which takes no parameters and throws **RemoteException** and **CreateException**. The return type of this create method must be Remote interface created in earlier steps

(Note: We can provide multiple create methods for stateful session bean)

10

To compile: `javac -d . TBHome.java`

7. Create a bean class by following the rules given below.

Rule 1:

The class must implement session bean

```
public class TBBean implements SessionBean{
```

```
.....  
.....
```

20

```
}
```

checkout the docs on j2sdkee1.4/doc/api/index.html. Specifically checkout the interfaces available in `javax.ejb` packages.

We have interface `EnterpriseBean` interface which must be implemented by every enterprise Bean class. It is a common **superinterface** for the **SessionBean**, **EntityBean** and **MessageDrivenBean** interfaces.

30

(MyNote: public interface `EnterpriseBean` extends `Serializable` hence it follows that every remote object can be used as return type and as parameter or return type since they are `Serializable`)

The methods that are there in `SessionBean` interface which we need to implement.

```
public void ejbActivate() throws  
EJBException, RemoteException
```

40

The activate method is called when the instance is activated from its "passive" state.

```
public void ejbPassivate() throws  
EJBException, RemoteException
```

The `passivate` method is called before the instance enters the "passive" state.

```
public void ejbRemove() throws  
EJBException, RemoteException
```

A container invokes this method before it ends the life of the session object.

```
public void setSessionContext(SessionContext ctx)  
throws EJBException, RemoteException
```

Set the associated session context.

10

Similar to servlets `init`, in EJB's we have `setSessionContext(...)` which will be called when EJB is created

Similar to destroy method in servlets `ejbremove()` will be called before removing the object. The other two methods namely `ejbActivate()` and `ejbPassivate()` will not be called in stateless session bean.

20

Rule 2: Provide `ejbCreate` method corresponding to the `create` method in the Home interface. The return type must be void and the method must throw `CreateException`.

Rule 3: Provide the Business methods. These methods must not throw `RemoteException`

compile the bean class as follows: `javac -d . TBBean.java`

30

8. Package all the classes into single jar file using jar tool.

```
jar cvf temp.jar info
```

(in above command `temp.jar` is the output file desired and `info` is the parent directory-all the sub-directories and files are included)

(Note : what these options mean -

`c` create new archive

`f FILE` specify archive file name

`v` generate verbose output on standard

40

output

)

9. Start the tool Weblogic Builder using the command available in `/bea/weblogic81/server/ bin/`

startWLBUILDER.cmd

(If you are not able to locate the .cmd file do a search in weblogic home directory)

Using the above tool we can create the deployment descriptors eg **ejb-jar.xml**

10. Run weblogic.ejbc

10

```
java weblogic.ejbc temp.jar mybean.jar
```

use **keepgenerated** to keep the intermediary java and other files generated.

The above tool validates the ejb provided by us and generates additional classes. The tool generates the classes that provides the implementation of Remote interfaces **Home interface** and **local interfaces & localhome interface**.

20

11. We can deploy the EJB by using weblogic console or by copying the jar file containing the bean under application folder.

12. When deployed the jndi name provided in builder is used to register with directory server

Here we used **com.inet.my.own.bean** as jndi name

When we deploy the ejb in a server the info about the ejbHome will be registered in the directory server.

30

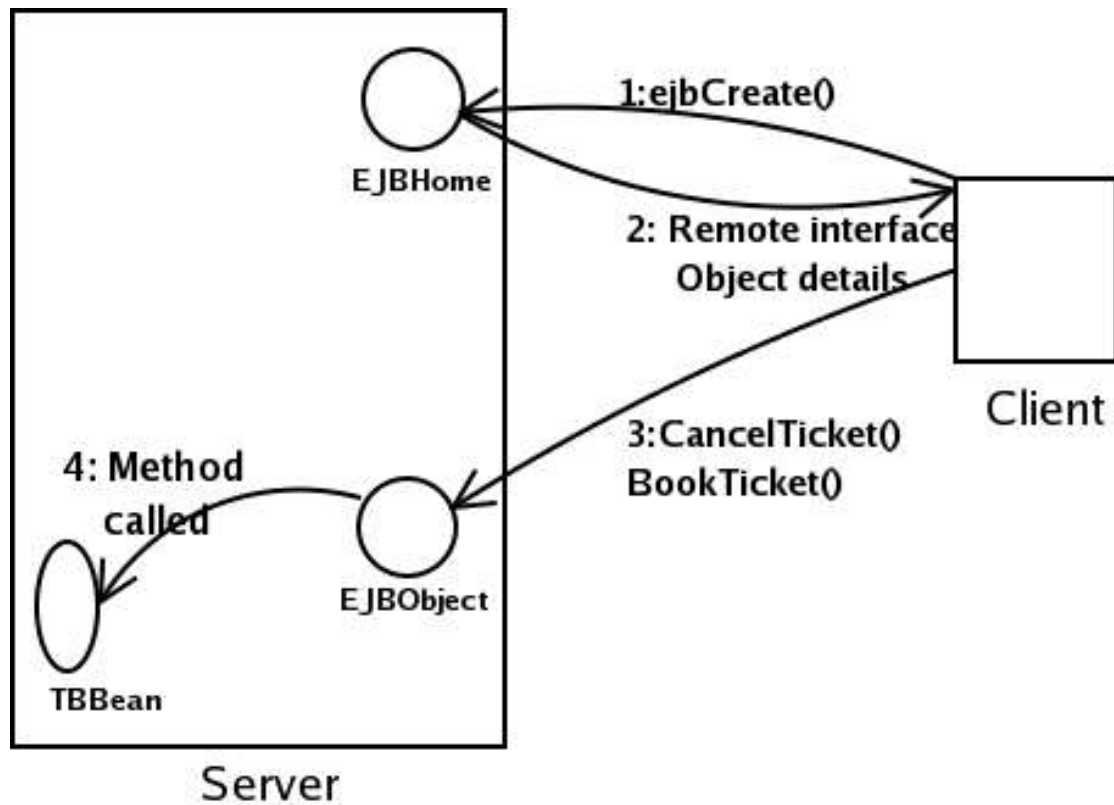
13. In order to call the business methods on the ejb we will be writing the same kind of code in application, applets, jsp's or servlets.

We can refer to Jboss or Weblogic documentation to check any specific requirements.

40

MyNote: The ejb clients are not allowed to directly access the enterprise beans. Clients talk to the beans via EJBHome and EJBObject. Security, transaction issues are taken care by EJBObject etc.

MyDiagram:



Summary of Session 3/ :

10

20

Enterprise JavaBeans

Session 4 of

Dated: 19th June 2004 Saturday

10

**Topic: Execution from Client, Session beans
method flow**

20

Refer back(for better understanding)

1. Previous Sessions on EJB 1-2
2. RMI Sessions
3. JDBC
4. Factory Pattern
5. Refer to code, Documentations, screen shots provided by Inetsolv.
6. Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 5 of

Dated: 21st June 2004 Monday

10

Topic: Stateful Session Beans

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 6 of

Dated: 22nd June 2004 Tuesday

10

Topic: Stateless vs Stateful

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 7 of

Dated: 23rd June 2004 Wednesday

10

Topic: JDBCDirectRead and DTO Design pattern

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 8 of

Dated: 24th June 2004 Thursday

10

**Topic: Container managed Transaction
using JTA, Transactional Attributes**

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code, Documentations, screen shots provided by
Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 9 of

Dated: 25th June 2004 Friday

10

Topic: Study of Various TA' sLocal interface-How?

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 10 of

Dated: 26th June 2004 Saturday

10

Topic: Local vs Remote interface, Security

20

Refer back(for better understanding)

1. Previous Sessions on EJB 1-2

2. RMI Sessions

3. JDBC

4. Factory Pattern

5. Refer to code, Documentations, screen shots provided by Inetsolv.

6. Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 11 of

Dated: 28th June 2004 Monday

10

Topic: Entity Beans – Using JbuilderX

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 12 of

Dated: 29th June 2004 Tuesday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 13 of

Dated: 30th June 2004 Wednesday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 14 of

Dated: 1st July 2004 Thursday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 15 of

Dated: 2nd July 2004 Friday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 16 of

Dated: 3rd July 2004 Saturday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 17 of

Dated: 5th July 2004 Monday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 18 of

Dated: 6th July 2004 Tuesday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 19 of

Dated: 7th July 2004 Wednesday

10

Topic:

20

Refer back(for better understanding)

1.Previous Sessions on EJB 1-2

2.RMI Sessions

3.JDBC

4.Factory Pattern

5.Refer to code,Documentations,screen shots provided by Inetsolv.

6.Session on JbuilderX- Enterprise Edition

30

Enterprise JavaBeans

Session 20 of
Dated: 8th July 2004 Thursday

**Topic: EJBHome/EJBSelect,Handle,
Environment Entries**

10

20

30

Refer back(for better understanding)

- 1.Sessions on Entity beans
- 2.Sessions on Servlet
- 3.Session on Weblogic Builder

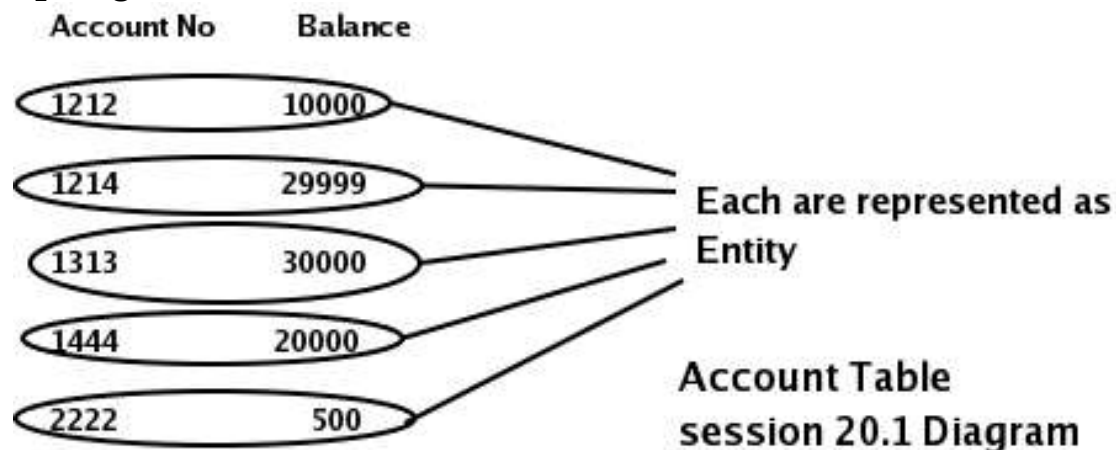
EJBHome/EJBSelect

Finder methods-to get details about data already available.

In entity beans - CMP EJBQL lang used.

EJBSelect is going to give same kind of option. Query can be provided.

MyDiagram:



Business Methods on these entities can be getAccountNo(), getBalance() which are specific to a entity.

10

We can have a nonspecific method like getAvgBalance() which is not specific to an entity.

Some more eg's are :-

getMinAccBalance(), getMaxAccBalance(), getSumofBalance().

All these nonspecific methods are home methods.

The methods like getAccId(), getBalance() are specific to a entity. getAvgAccBal(), getHighestBal() etc are not specific to a entity. So these methods can be implemented as home m's (These methods has to be declared as part of home interface but not part of the remote interface & the local interface)

20

In AccountHome and AccountRemoteHome we have

```
public float getAvgBalance();
```

AccountBean

```
public float ejbHomeGetAvgBalance(){...}
```

To implement getAvgBalance() home method we need to write JDBC code. This code can be eliminated by using ejbSelect methods.

30

In ejb designer of JbuilderX Add/Select on Bean
getAll() return type is Collection
ejbQL is select object(o) from Account as o
return type local
This method added in AccountBean class as

```
public abstract Collection ejbSelectGetAll() throws  
                        FinderException;
```

Note: The above method is implemented by the Container

10

Client cannot call this method directly.

```
In public float ejbHomeGetAvgBalance(){  
int noaccns=0;  
float totalBalance=0.0;  
try{  
    Collection c = ejbSelectGetAll();  
    Iterator it=c.iterator();  
    while(it.hasNext()){  
        // System.out.println(it.next());  
        Account a = (Account)it.next();  
        totalBalance += a.getBalance();  
        noaccns++;  
    }  
}catch(Exception e){}  
return totalBalance/noaccns;  
}
```

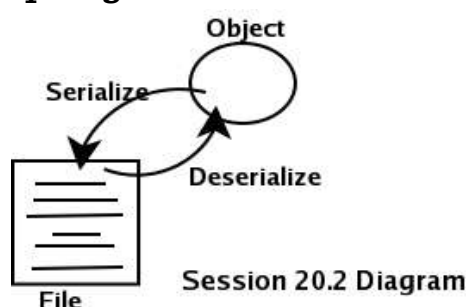
20

Handle

30

Reference to a bean. Can be serialized

MyDiagram:



Handle is serializable. The handle holds the details of the reference to a bean.

The method signature as defined in EJBObject interface is

public Handle getHandle() throws RemoteException

Refer to Example 4

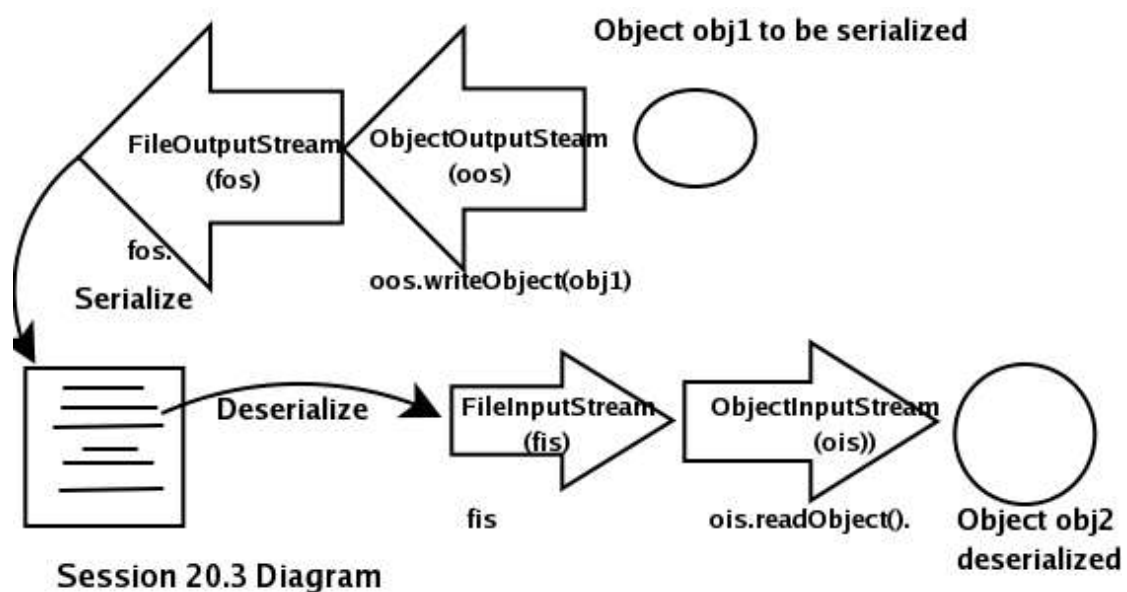
```
EJBObject ejbo=Context.getEJBObject()  
(in the above context can be session or entity context)  
Handle h= ejbo.getHandle();
```

10

As part of our ejbCode we can write as above to get the handle. We can return this handle instead of "this" ("this" cannot be returned) (Refer to Session 19 Wednesday, 8th July)

In Weblogic 4,5 above would return null. From weblogic 6 onwards the function returns the handle correctly.

MyDiagram:



```
h1=(Handle)ois.readObject();  
ejo=h1.getEJBObject();  
SesCtxRemote sctx=(SesCtxRemote)ejo;
```

20

Instead of using `home.create()` or `home.findXxx()` to get the reference of the ejb we can read the handle from file using deserialization and execute `handle.getEJBObject()` as above.

Environment Entries

Similar to initialization parameters in Servlets.

Environment Entries can be used to avoid hard coding. IF we hard code the JNDI name of the connection pool the ejb may fail if the connection pool is not registered with same JNDI name. To avoid this problem we can use environment entries.

In case of servlets we store the info about initialization parameter also in web.xml. In case of ejb's we can store the environment entries in the deployment descriptor (Refer ex 11) 10

Open in Weblogic Builder Add Environment Entries as Environment vars in xml files.

To get the environment entries

- 1 Get the initial context
- 2 Get Environment naming Context(ENC)
"java:comp/env"
Context ctx=(Context)ic.lookup("java:comp/env"); 20
- 3 Get the names of entries required and were stored
String company=(String)ctx.lookup("cname")
String address=(String)ctx.lookup("caddress")

Note: JDO and XMLBeans built on lower level API's.

Enterprise JavaBeans

Session 21 of

Dated: 9th July 2004 Friday

**Topic: Session Facade Design Pattern and
EJBHome Caching Design Pattern**

10

20

30

Refer back(for better understanding)

- 1.Sessions on Entity Beans
- 2.Session on Design Pattern
- 3.Sessions on servlet/JSP
- 4.Session on JNDI.

What is **Session Facade Design Pattern** and Why Session Facade Design Pattern ?

(Refer to **DTO** and **JDBCDirectRead** Design pattern already covered.(Session 7th covered on 23rd June))

Facade means **front**.

Client should not directly access entity. Client talks to session bean which interacts with entity beans.

10

Scenario - Rs 100 to be transferred from account 1 to account 2

Account Table

Account no	Balance
1	1999.00
2	2000.00
3	400.00
4	6000.00
5	22222.00

20

Business Methods declared were

```
String getBal()  
void setBal(String)
```

30

Client Code (Pseudo code)

```
1: MyAccountRemote acc1 = home.findByPrimaryKey("1");  
2: MyAccountRemote acc2 = home.findByPrimaryKey("3");  
  
3: String oldamt, newamt;  
4: float amt1, amt2;  
  
5: oldamt = acc1.getBal();  
   //Note return values are String hence we need to parse  
6: amt1 = Float.parseFloat(oldamt);  
7: amt2 = amt1 - 100;  
8: newamt = new Float(amt2).toString();  
9: System.out.println("new amount after debit is" +
```

40

```
newamt);
10: acc1.setBal(newamt);

11: oldamt=acc2.getBal();
    //Note return values are String hence we need to parse
12: amt1=Float.parseFloat(oldamt);
13: amt2=amt1+100;
14: newamt=new Float(amt2).toString();
15: System.out.println("new amount after credit is" +
    newamt);
16: acc2.setBal(newamt);
```

10

In the above code there are 6 remote calls. Note these calls in line numbers 1,5,10,11,16

Any business transaction involves operations on 1 or more entities.

For example A business transaction like transferring some amount from one to another account need to perform operations on two entity beans acc1 and acc2. If we need to access the entity beans directly from client we need to unite the code as shown above.

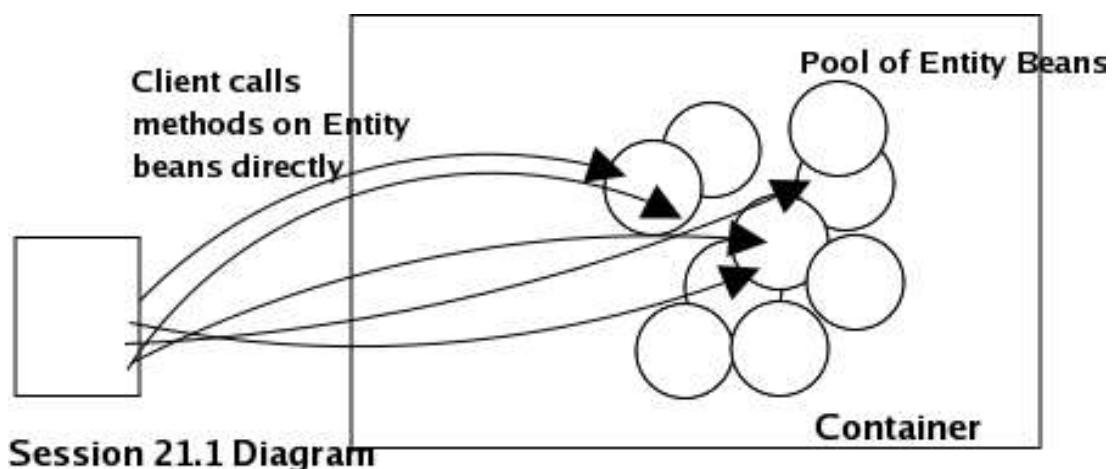
20

In the above code the client is making 6 remote method calls. Refer to Diagram 21.1.

As the number of remote method calls increase the performance of application reduces. To improve the performance of application in this case we can use **Session Facade design pattern**.

30

MyDiagram:



The above application is redone with session facade DP.

SesFacadeBean:

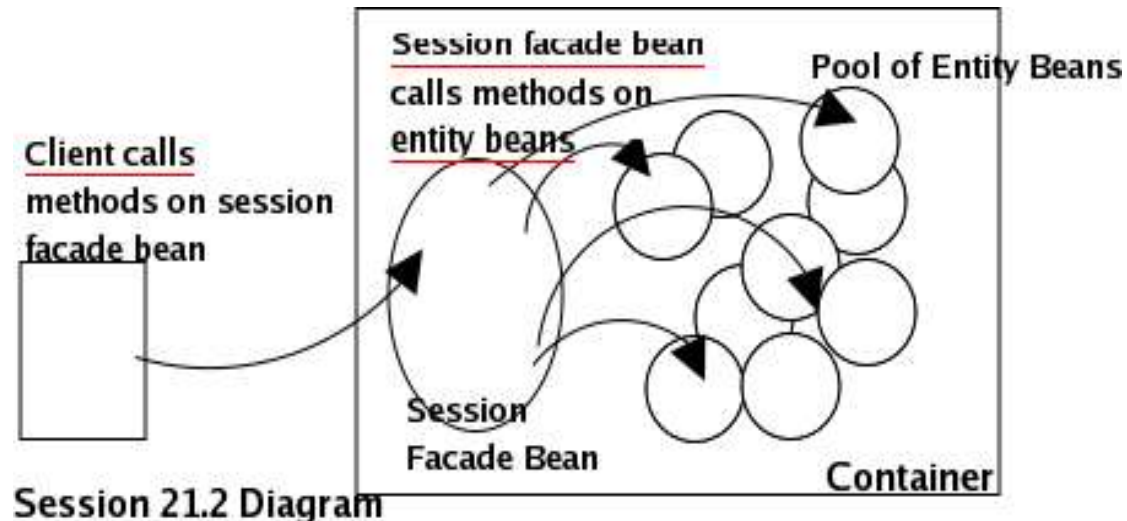
```
1: void transferAmount(String ac1,String ac2,float amt){  
  
2:     Context ic;  
3:     try{  
4:         ic=new InitialContext()  
           // Since client is at server side calling a bean  
           locally  
  
5:         MyAccountHome home=(MyAccountHome)ic.lookup  
           ("MyAccount");  
  
6:         MyAccount acc1=home.findByPrimaryKey(ac1);  
7:         MyAccount acc2=home.findByPrimaryKey(ac1);  
8:         String oldamt,newamt;  
9:         float amt1,amt2;  
  
10:        oldamt=acc1.getBal();  
           //Note return values are String hence we need to parse  
11:        amt1=Float.parseFloat(oldamt);  
12:        amt2=amt1-100;  
13:        newamt=new Float(amt2).toString();  
14:        System.out.println("new amount after debit is" +  
           newamt);  
15:        acc1.setBal(newamt);  
  
16:        oldamt=acc2.getBal();  
           //Note return values are String hence we need to parse  
17:        amt1=Float.parseFloat(oldamt);  
18:        amt2=amt1+100;  
19:        newamt=new Float(amt2).toString();  
20:        System.out.println("new amount after credit is" +  
           newamt);  
21:        acc2.setBal(newamt);  
22:    }catch(Exception e){}
```

MyNote: In above code line from line 8 to 21 the code is same as in the previous eg with remote client.

In the session Facade pattern the clients will be accessing the methods on the session bean and the session bean will be accessing the entity beans to perform the

business transaction. Refer Diagram 21.2.

MyDiagram:



In client the code will be:

```
SesFacadeBean fa=home.create();
fa.transferAmount("1","2",100);
Instead of 6 remote calls only 2 remote calls
```

If we need to calculate the time taken for the transaction to be performed we can use 10

```
long t1,t2;
t1=System.currentTimeMillis();
```

```
//here goes the client code involving the 2 remote calls
which is called say 100 times in a loop
for(int i=0;i<100;i++){
```

```
.....
.....
```

```
SesFacadeBean fa=home.create();
fa.transferAmount("1","2",100);
```

```
.....
```

```
.....
```

```
}
```

```
t2=System.currentTimeMillis();
```

```
System.out.println("time taken for calls is : " +
```

20

30

```
t1+t2);
```

We find the time taken with facade design pattern is only 2547ms whereas without it has taken 6156 ms. Nearly 150 percent improvement in performance.

(Rewrite the first eg with a loop of 100 to get the time)

We can think and rewrite a complex case where amount need to be transferred from one account in one branch to another branch. 10

In any distributed project to improve the performance:

- 1 Reduce the number of remote calls.
- 2 Reduce the amount of data transfer between client and server.

What is **EJBHome Caching Design Pattern** and Why EJBHome Caching Design Pattern ?

20

Scenario:

Let us take a case where the below screen got generated by a Servlet/JSP.

Clicking the tranfer button will result in accessing the bean and the execution of remote call.

Account Id From

Account Id To

30

Amount

Transfer

```
class MyServlet extends HttpServlet{
service(...){
Context initial = getInitialContext();
Object obj=initial.lookup("sesfacade");
SesFacadeHome home=(SesFacadeHome)Obj;
```

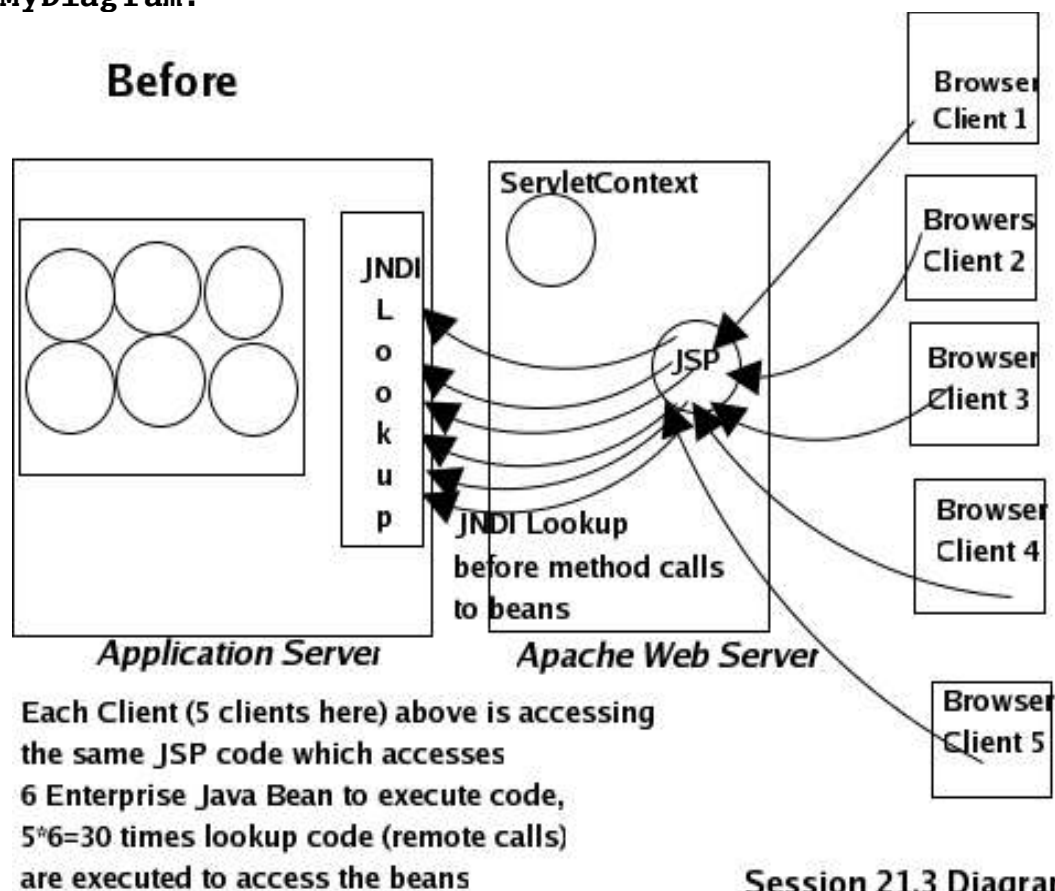
40


```
//using home to get the reference of bean and perform
operations.
}
```

(if 10,000 times above code is executed without above dp everytime we need home object, we need to lookup everytime)
In the above application whenever a request is sent to the servlet we are performing lookup operation which is expensive. Refer to Diagram 21.3.

MyDiagram:

10



Session 21.3 Diagram

To solve this problem we can implement `applicationListener` (may be `ServletContextListener`). In the `ContextCreated` method we can write the code as shown below. Refer Diagram 21.4

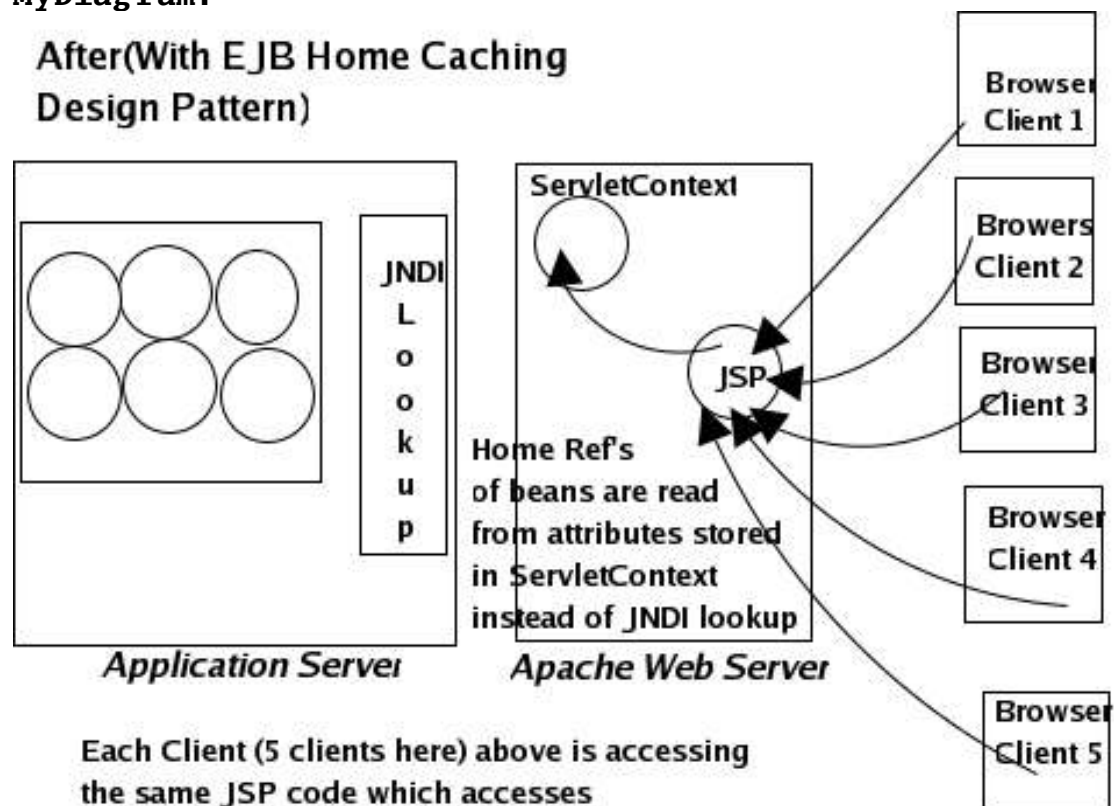
```
ic=getInitialContext();
home=ic.lookup("SesFacadeHome");
//cache the home in the servletContext so that we can use
it later.
servletcontext.setAttribute("homeref",home);
// executed when application deployed.
In Service method
```

20

```
home=(SesFacadeHome)servletcontext.getAttribute
("homeref");
```

MyDiagram:

**After(With EJB Home Caching
Design Pattern)**



Each Client (5 clients here) above is accessing the same JSP code which accesses 6 Enterprise JavaBeans to execute code, $5 \times 6 = 30$ times attributes are read from servletContext (local calls instead of remote)

Session 21.4 Diagram

Refer site www.theserverside.com for other DP's

Main ones

MVC

Front Controller

Singleton

DTO (Data transfer Object) - For Report

Session Facade

JDBCDirectRead

Rowset instead of resultset.

10

20

10

Enterprise JavaBeans

Session 22 of

Dated: 12th July 2004 Monday

Topic:

20

Refer back these Sessions to get Better Understanding

30

1. Previous Sessions on EJB 1-2

2. RMI Sessions

3. JDBC

4. Factory Pattern

5. Refer to code, Documentations, screen shots provided by Inetsolv.

6. Session on JbuilderX- Enterprise Edition

References(for further Exploration)

Reference Books/Magazines :

- 1.
- 2.
- 3.

Reference WebSites/Links :

10

1. <http://www.theserverside.com>
2. <http://java.sun.com/ejb>
3. <http://java.sun.com/j2ee>
4. <http://www.weblogic.com>
5. <http://jakarta.apache.org>
6. <http://www.mysql.org>

20

30

40

AfterThought

Initially I had planned to have line numbers through out the document for editing purpose only, later realized may be it will be useful for people while giving feedback and also for people while discussing any relevent points from this document. 10

I welcome both positive and negative feedback so I can know which portions really made a difference and was helpful, so can be given more importance and to know which really didn't help at all, so can be eliminated altogether. And anything missing that can be added.

There are scope for adding exercises, assignments, questions, Case studies with scenerios. I request other developers to send in any interesting piece of code, case, assignment etc, that they may come across relevant to the topic in discussion. 20

I request one and all to join the developers group <http://groups.yahoo.com/group/idevcircle> to hone there software development skills and to contribute to the developers community at large.

Further this is just the draft release which would be re-verified by J2EE faculty and myself and others if necessary. This is to make it full proof as regards any errors in technical discussion and any spelling errors. 30

About MySelf

I love to Learn and do Software Development. I love to Read, Write, Surf the Net and Chat. In my free time I prefer to Watch TV, Read, make friends, and of course love to sleep. I like to do Tai Chi and Yoga as and when I can and like to Heal people, relationships and situations with Reiki. 10

Presently I am doing a research in astro-spiritual dynamics. I have interests in Astrology with special interest in Medical Astrology, Numerology and Graphology.

Currently working on Mobile Applications and few other interesting Software Projects. 20

I am holding a Bachelor of Engineering Degree in Mechanical Engineering. I am Sun Certified Programmer for the Java 2 Platform.

30

40

Guidelines to Register with idevcircle

To Register with *idevcircle*

10

Method 1:

Visit <http://groups.yahoo.com/>

Enter Search word *idevcircle* click search you will be getting the group page. Follow the instructions given.

Method 2:

Visit <http://geocities.com/idevcircle> and subscribe by entering your valid email id.

20

Method 3:

Visit <http://groups.yahoo.com/group/idevcircle> you will be getting the group page. Follow the instructions given.

Method 4:

Send email to idevcircle-subscribe@yahoogleroups.com from the email account you want to receive messages from the group. You will be receiving a confirmation mail. Follow the instructions given to Join.

30

MyNote:

1. You need to have yahoo membership to join the group.

2. You can just join in the mailing list to receive messages being posted in the group or you can join in the group as group member to share all group resources like files, links etc.

40

!!! Announcements !!!
Tentative
Release Dates

10

24th July 2004
for
Rajan's Enterprise JavaBeans
Sessions Notes
(Draft)

20

1st August 2004
Rajan's Enterprise JavaBeans
Sessions Notes
(FINAL)

Register in our group NOW to avoid rejection later
Limited Membership

Join idevcircle @ <http://groups.yahoo.com>
(To Join send mail to idevcircle-subscribe@yahoogroups.com)

30

!!!!!! Coming Soon !!!!!

Notes on J2EE Development on GNU/Linux
Notes on Java Code Convention

Look out for Release Dates.

40