

Enterprise Beans in a Nutshell (Publications Team Bravo)

What the heck is an Enterprise Bean?

- A server side component that encapsulates the business logic of an application
- Business logic: the code that fulfils the purpose of the application.
- To put it another way an enterprise bean is basically a module, either self-standing or part of a larger system.
- E.g. Publications module == enterprise bean

*In the java world beans refer to objects so to speak.

Benefits of Enterprise Beans:

- Scalability of applications
- Enterprise bean is stored in a EJB Container that provides system level services such as transaction management and security authorisation. Allowing the developer to focus only on the business logic.
- Because the Enterprise bean hosts the business logic, the client can focus purely on presentation. (Thin client)
- Enterprise beans are also portable components thus they can be reused in other applications.

When should we use Enterprise beans?

- If the application must be scalable.
- Transactions must ensure data integrity
- Applications will have a variety of clients (access channels).

Types of Enterprise Beans:

- Session: Performs client tasks and may implement a web service (our focus).
- Message-driven: Act as listener for a particular messaging type

What is a Session Bean?

- A session bean encapsulates the business logic that can be invoked programmatically by a client over local, remote, or web service client views.
- To access an application that is hosted on the application server (Glassfish in our case), the client invokes the session bean's methods.
- The session bean performs work for the client by executing tasks inside the server. (think of it like a kind of proxy that does all the work)

Types of Session beans:

- **Stateful Session Beans (maybe to be used...):**
 - The state of an object consists of the value of its instance variables. In a stateful session bean, the instance variables represent the state of a unique client/bean session
 - The client thus interacts with its bean
 - Often called a conversational state.
 - Can have only one client
 - The state is retained for the duration of the client/bean session.
- **Stateless Session Beans (Important for us!!!):**
 - Does not maintain a conversational state with the client.
 - The bean exists only as long as the duration of a method invocation.
 - When the method is finished, the client-specific state should not be retained.
 - Can support multiple clients, thus offer better scalability.
 - Can implement a web service!!!

*use stateless session beans when implementing a web service.

I have included a **ejb_tutorial.pdf**, please have a look at it and follow the examples. The only obstacle is that they make use of JBOSS whereas we use Glassfish... If anyone can get passed that step please let me know. It may look like a lot of pages but it's not a lot of work.

Good luck guys!

Figure 1 Break down of the layer depicting a high level of granularity

