

## Re: Stateless session bean that multi threads?

**Source:** <http://coding.derkeiler.com/Archive/Java/comp.lang.java.programmer/2003-12/1482.html>

---

**From:** John C. Bollinger ([jobollin\\_at\\_indiana.edu](mailto:jobollin_at_indiana.edu))

**Date:** 12/11/03

Date: Thu, 11 Dec 2003 14:33:10 -0500

Do not crosspost. Followups directed to comp.lang.java.beans.

davout wrote:

- > *As a relative EJB newbie... as I undersatand it you're not supposed to*
- > *multi-thread inside the implementation of a stateless session bean method.*
- > *(Right?)*

Right. Attempting to start new threads from within an EJB implementation method would be a violation of the bean's contract with its container.

- > *In this case how do I solve a situation where my stateless session bean*
- > *wants to draw data from 'n' sources and then consolidate and return the*
- > *results as one array?? Ideally you'd want to create a sub thread for each*
- > *different data source. Are here any workarounds?*

Whether a threaded scenario in fact makes much sense in the first place rather depends on the number and nature of the data sources. If they exhibit sufficiently low latency with respect to their number then very little is to be gained by accessing them through multiple threads instead of serially.

If there *\*is\** an advantage to be gained then you probably want to look at ways of performing the individual lookups asynchronously. The standard tool for asynchronous tasks in the EJB world is the message-driven bean. There may be other alternatives available based on the specific characteristics of your data sources.

The final alternative is to push the multi-threaded access out to the client side. That's actually the simplest, and if the client is local then perhaps also the best performing. You could write a client-side wrapper that makes it transparent to users.

John Bollinger  
[jobollin@indiana.edu](mailto:jobollin@indiana.edu)