

Robotics Service Bus - Bug #2131

Java SingleThreadEventReceivingStrategy cannot activate after deactivating

12/12/2014 04:58 PM - M. Goerlich

Status:	Rejected	Start date:	12/12/2014
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Java	Estimated time:	0.00 hour
Target version:			

Description

Since the Thread is not set to null activating will always lead to "already active" errors.

```
public class SingleThreadEventReceivingStrategy implements
    EventReceivingStrategy {

    // ...

    @Override
    public void activate() {
        synchronized (this) {
            if (this.thread != null) {
                throw new IllegalStateException("Already activated."); // <-- This path is taken.
            }
            this.thread = new DispatchThread(this.events);
            this.thread.start();
        }
    }

    @Override
    public void deactivate() throws InterruptedException {
        synchronized (this) {
            if (this.thread == null) {
                throw new IllegalStateException("Already deactivated.");
            }
            this.thread.interrupt();
            this.thread.join();
            // here we should: this.thread = null;
        }
    }

    // ...
}
```

Small example to produce error:

```
Listener listener = rsb.Factory.getInstance().createListener(new Scope("/"));

listener.activate();

listener.deactivate();
```

```
listener.activate(); // <--- Death to all who reach this point!!!! Whohahah
```

```
listener.deactivate();
```

```
System.exit(0);
```

History

#1 - 12/12/2014 05:22 PM - J. Wienke

This is actually an "intended feature". The documentation from Activatable stated:

If not stated otherwise, object may only survive one activation - deactivation cycle and may not be reactivatable in a n ordered way.

#2 - 12/12/2014 05:25 PM - J. Wienke

- Status changed from New to Rejected