

# Schedule Expression

Example `schedule-expression` can be browsed at <https://github.com/apache/tomee/tree/master/examples/schedule-expression>

In this example we exercise the `TimerService`.

**NOTE** "The `TimerService` interface provides enterprise bean components with access to the container-provided Timer Service.

The EJB Timer Service allows entity beans, stateless session beans, and message-driven beans to be registered for timer callback events at a specified time, after a specified elapsed time, or after a specified interval."

For a complete description of the `TimerService`, please refer to the Java EE tutorial dedicated to the [Timer Service](#).

## FarmerBrown

At `PostConstruct` we create 5 programmatic timers. First four will most likely not be triggered during the test execution, however the last one will timeout a couple of times.

Each timer contains an `info` attribute, which can be inspected at timeout.

```
package org.superbiz.corn;

import javax.annotation.PostConstruct;
import javax.annotation.Resource;
import javax.ejb.Lock;
import javax.ejb.LockType;
import javax.ejb.ScheduleExpression;
import javax.ejb.Singleton;
import javax.ejb.Startup;
import javax.ejb.Timeout;
import javax.ejb.Timer;
import javax.ejb.TimerConfig;
import javax.ejb.TimerService;
import java.util.concurrent.atomic.AtomicInteger;

/**
 * This is where we schedule all of Farmer Brown's corn jobs
 *
 * @version $Revision$ $Date$
 */
@Singleton
@Lock(LockType.READ) // allows timers to execute in parallel
@Startup
public class FarmerBrown {
```

```

private final AtomicInteger checks = new AtomicInteger();

@Resource
private TimerService timerService;

@PostConstruct
private void construct() {
    final TimerConfig plantTheCorn = new TimerConfig("plantTheCorn", false);
    timerService.createCalendarTimer(new ScheduleExpression().month(5).dayOfMonth(
"20-Last").minute(0).hour(8), plantTheCorn);
    timerService.createCalendarTimer(new ScheduleExpression().month(6).dayOfMonth(
"1-10").minute(0).hour(8), plantTheCorn);

    final TimerConfig harvestTheCorn = new TimerConfig("harvestTheCorn", false);
    timerService.createCalendarTimer(new ScheduleExpression().month(9).dayOfMonth(
"20-Last").minute(0).hour(8), harvestTheCorn);
    timerService.createCalendarTimer(new ScheduleExpression().month(10).
dayOfMonth("1-10").minute(0).hour(8), harvestTheCorn);

    final TimerConfig checkOnTheDaughters = new TimerConfig("checkOnTheDaughters",
false);
    timerService.createCalendarTimer(new ScheduleExpression().second("*").minute(
"*").hour("*"), checkOnTheDaughters);
}

@Timeout
public void timeout(Timer timer) {
    if ("plantTheCorn".equals(timer.getInfo())) {
        plantTheCorn();
    } else if ("harvestTheCorn".equals(timer.getInfo())) {
        harvestTheCorn();
    } else if ("checkOnTheDaughters".equals(timer.getInfo())) {
        checkOnTheDaughters();
    }
}

private void plantTheCorn() {
    // Dig out the planter!!!
}

private void harvestTheCorn() {
    // Dig out the combine!!!
}

private void checkOnTheDaughters() {
    checks.incrementAndGet();
}

public int getChecks() {
    return checks.get();
}

```

```
}  
}
```

## FarmerBrownTest

The test class acquires an instance from the context and waits for 5 seconds to give the timers a chance to timeout.

```
package org.superbiz.corn;  
  
import junit.framework.TestCase;  
  
import javax.ejb.embeddable.EJBContainer;  
import javax.naming.Context;  
  
import static java.util.concurrent.TimeUnit.SECONDS;  
  
/**  
 * @version $Revision$ $Date$  
 */  
public class FarmerBrownTest extends TestCase {  
  
    public void test() throws Exception {  
  
        final Context context = EJBContainer.createEJBContainer().getContext();  
  
        final FarmerBrown farmerBrown = (FarmerBrown) context.lookup(  
"java:global/schedule-expression/FarmerBrown");  
  
        // Give Farmer brown a chance to do some work  
        Thread.sleep(SECONDS.toMillis(5));  
  
        final int checks = farmerBrown.getChecks();  
        assertTrue(checks + "", checks > 4);  
    }  
}
```

## Running

```
-----  
T E S T S  
-----
```

```
Running org.superbiz.corn.FarmerBrownTest  
Apache OpenEJB 4.0.0-beta-1    build: 20111002-04:06  
http://tomee.apache.org/  
INFO - openejb.home = /Users/dblevins/examples/schedule-expression
```

```
INFO - openejb.base = /Users/dblevins/examples/schedule-expression
INFO - Using 'javax.ejb.embeddable.EJBContainer=true'
INFO - Configuring Service(id=Default Security Service, type=SecurityService,
provider-id=Default Security Service)
INFO - Configuring Service(id=Default Transaction Manager, type=TransactionManager,
provider-id=Default Transaction Manager)
INFO - Found EjbModule in classpath: /Users/dblevins/examples/schedule-
expression/target/classes
INFO - Beginning load: /Users/dblevins/examples/schedule-expression/target/classes
INFO - Configuring enterprise application: /Users/dblevins/examples/schedule-
expression
WARN - Method 'lookup' is not available for 'javax.annotation.Resource'. Probably
using an older Runtime.
INFO - Configuring Service(id=Default Singleton Container, type=Container, provider-
id=Default Singleton Container)
INFO - Auto-creating a container for bean FarmerBrown: Container(type=SINGLETON,
id=Default Singleton Container)
INFO - Configuring Service(id=Default Managed Container, type=Container, provider-
id=Default Managed Container)
INFO - Auto-creating a container for bean org.superbiz.corn.FarmerBrownTest:
Container(type=MANAGED, id=Default Managed Container)
INFO - Enterprise application "/Users/dblevins/examples/schedule-expression" loaded.
INFO - Assembling app: /Users/dblevins/examples/schedule-expression
INFO - Jndi(name="java:global/schedule-
expression/FarmerBrown!org.superbiz.corn.FarmerBrown")
INFO - Jndi(name="java:global/schedule-expression/FarmerBrown")
INFO -
Jndi(name="java:global/EjbModule481105279/org.superbiz.corn.FarmerBrownTest!org.superb
iz.corn.FarmerBrownTest")
INFO - Jndi(name="java:global/EjbModule481105279/org.superbiz.corn.FarmerBrownTest")
INFO - Created Ejb(deployment-id=org.superbiz.corn.FarmerBrownTest, ejb-
name=org.superbiz.corn.FarmerBrownTest, container=Default Managed Container)
INFO - Created Ejb(deployment-id=FarmerBrown, ejb-name=FarmerBrown, container=Default
Singleton Container)
INFO - Started Ejb(deployment-id=org.superbiz.corn.FarmerBrownTest, ejb-
name=org.superbiz.corn.FarmerBrownTest, container=Default Managed Container)
INFO - Started Ejb(deployment-id=FarmerBrown, ejb-name=FarmerBrown, container=Default
Singleton Container)
INFO - Deployed Application(path=/Users/dblevins/examples/schedule-expression)
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 6.141 sec
```

Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0