

How to determine setback intervals depending on the exit code

- [Setback intervals based on exit codes](#)
 - [XML Configuration of the Job](#)
 - [XML configuration of the Job Chain](#)
- [See also](#)

Setback intervals based on exit codes

If you want to set different setback intervals depending on the exit code then you can use the internal API by defining a post-processing routine after the task.

In this example, we define a job in a job chain.

- When the job ends with exit code 1 then try again in 60 seconds
- When the job ends with exit code 444 then try again in 2 hours
- When the job ends with other non-zero exit codes then try again in 300 seconds

Perform the following tasks with JOE:

1. Define a post-processing function *spooler_task_after()*
2. Code the switch with your requirements.
3. Add the job to a job chain. Do this even you only have one job in the chain.

The screenshot shows the JOE configuration interface. On the left, a tree view under 'Scheduler Elements' shows a job chain named 'job_chain1' containing a job 'job1'. The 'Pre-/Postprocessing' section of 'job1' is expanded, and a blue arrow points to a custom function named 'handleSetback'. On the right, the 'Pre-/Postprocessing' configuration panel is shown. The 'Name' field contains 'handleSetback' and the 'Language' is set to 'Javascript'. The 'Source Code' field contains the following JavaScript code:

```
//After one setback, handle this as an error.
job.max_order_setbacks = 1;

switch (exit ) {
case 0: // proceed
    return true;
case 1: // Restart after 60 seconds
    job.delay_order_after_setback( 1 ) = "60";
    break;
case 444: //Restart after 2 hours
    job.delay_order_after_setback( 1 ) = "02:00:00";
    break;
default: //Restart after 300 seconds
    job.delay_order_after_setback( 1 ) = "300";
    break;
}
}
```



```

<?xml version="1.0" encoding="ISO-8859-1"?>
<job order="yes"
  stop_on_error="no">
  <script language="shell">
    <![CDATA[
exit 1
    ]]>
  </script>
  <monitor name="handleSetback"
    ordering="0">
    <script language="java:javascript">
      <![CDATA[
function spooler_task_after() {
  var exit = spooler_task.exit_code;
  var job = spooler_task.job;

  // After one setback, handle this as an error.
  job.max_order_setbacks = 1;

  switch (exit ) {
  case 0: // proceed
    return true;
  case 1: // restart after 60 seconds
    job.set_delay_order_after_setback( 1, "60" );
    break;
  case 4444: // restart after 2 hours
    job.set_delay_order_after_setback( 1, "02:00:00" );
    break;
  default: // restart after 300 seconds
    job.set_delay_order_after_setback( 1, "300" );
    break;
  }
}
      ]]>
    </script>
  </monitor>
</run_time/>
</job>

```

XML configuration of the Job Chain

```

<job_chain>
  <job_chain_node state="100"
    job="job1"
    next_state="success"
    error_state="error"
    on_error="setback"/>
  <job_chain_node state="success"/>
  <job_chain_node state="error"/>
</job_chain>

```

See also

- [Example for download](#)