

Event - Job

 Language: [Deutsch](#) · [English](#)

Event - Job

Event rule „Job“: interface, context and an example – how to react to this Nuclos event.

REFERENZ

DEVELOPER

UPDATED: JUL 2026

APPLIES TO NUCLOS 4.2026.X

On this page

- [When does the rule fire?](#)
- [Structure](#)
- [Assignment](#)
- [Example](#)
- [Related pages](#)

When does the rule fire?

This [server-side rule](#) runs scheduled by a **job/timer**. It can be assigned to a **timer/job**.

Structure

Skeleton of the rule class (the rule editor creates it automatically):

```
package org.nuclet.lager;

import org.nuclos.api.rule.JobRule;
import org.nuclos.api.context.JobContext;
import org.nuclos.api.annotation.Rule;
import org.nuclos.api.exception.BusinessException;

/** @name
 * @description
 * @usage
 * @change
 */

@Rule(name="Job Lager", description="Job Lager")
public class JobLager implements JobRule {

    public void execute(JobContext context) {

        //So werden Logs während des Ausführen in den Reiter Log Info in der Jobsteuerung geschrieben
        context.joblog(e.getMessage());
        context.joblogError("TEST");
        context.joblogWarn("Warnung");
    }
}
```

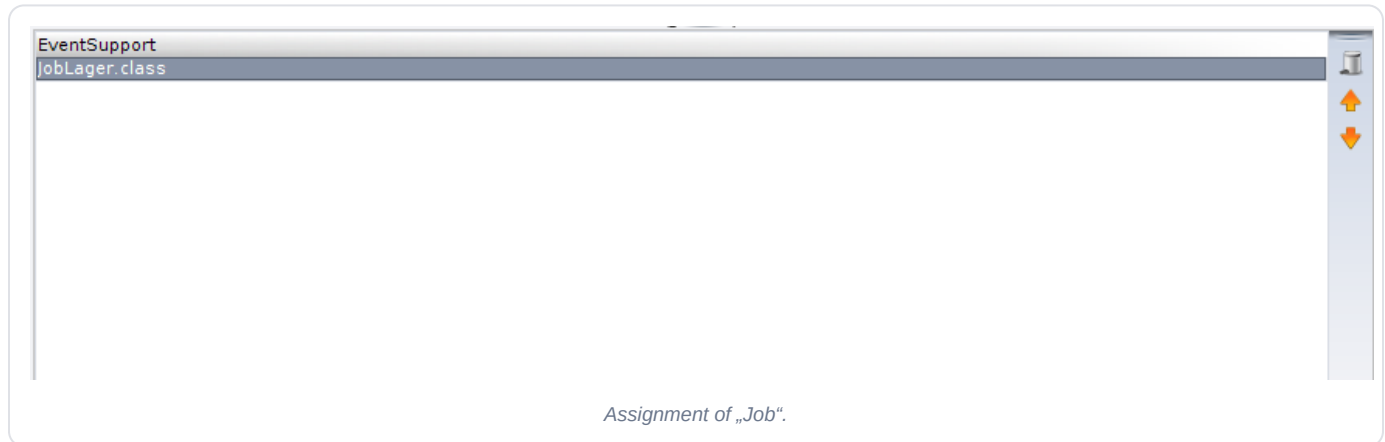


Context: JobContext

The context JobContext provides the affected BusinessObject via `context.getBusinessObject(...)`; the RuleContext functions (providers) are also available. A BusinessException aborts the operation.

Assignment

Assign it to the target via drag-and-drop in the [server rule manager](#). Optionally restrict execution to a **status** or an **action**; control the order with the arrow buttons.



Example

```
package org.nuclet.company;

import java.util.List;

import org.nuclos.api.annotation.Rule;
import org.nuclos.api.context.JobContext;
import org.nuclos.api.provider.QueryProvider;
import org.nuclos.api.rule.JobRule;

/** @name
 * @description
 * @usage
 * @change
 */
@Rule(name="JobBestellungenPruefen", description="JobBestellungenPruefen")
public class JobBestellungenPruefen implements JobRule {

    public void execute(JobContext context){
        // Get all Bestellungen
        List<Bestellung> lstBestellungen = QueryProvider.execute(
            QueryProvider.create(Bestellung.class));

        for (Bestellung b : lstBestellungen) {
            // orders that are not confirmed are ok, but should be logged as infos
            if (Boolean.FALSE.equals(b.getBestaetigt())) {
                context.joblog("Info: Bestellung " + b.getNummer() + " wurde noch nicht bestätigt.");
            }

            // orders without articles are formally valid, but dont make sense
            if (b.getPosition().size() <= 0) {
                context.joblogWarn("Warnung: Bestellung " + b.getNummer() + " besitzt keine Positionen");
            }

            // orders without an internal advisor are formally invalid
            if (b.getSachbearbeiter() == null) {
                context.joblogError("Fehler: Bestellung " + b.getNummer() + " besitzt keinen Sachbearbeiter");
            }
        }
    }
}
```

Related pages

Server-side rules

Basics.

[Open](#)

Server rule manager

Assign.

[Open](#)

Rule execution order

Order.

[Open](#)