

# Event - Statuswechsel im Anschluss (Beispiel)

Das folgende Beispiel soll den Aufbau einer Regel vom Typ "Statuswechsel (im Anschluss) und die Weiterverarbeitung der übergebenen Inhalte veranschaulichen.

## Fachlicher Hintergrund

Eine bestehende Nuclos-Instanz umfasst das Businessobjekt Auftrag. Die Aufgabe besteht darin, nach Anlage eines neuen Auftrags und dem Wechsel in den Status "beantragt", dessen Inhalte zu prüfen. Je nach Prüfungsergebnis soll der Auftrag automatisch in einen weiteren Status überführt werden. Im positiven Fall in "beauftragt", im fehlerhaften Fall in "prüfen".

Die im Beispiel verwendete Statusmodell-Klasse "ProzessAuftragSM" entspricht dem Statusmodell "Prozess Auftrag" in Nuclos, das für das Businessobjekt Auftrag angelegt und mit ihm verknüpft wurde. Weitere Informationen zu den Statusmodell-Klassen finden Sie hier.

## Quellcode

```
package org.nuclet.company;

import org.nuclos.api.annotation.Rule;
import org.nuclos.api.context.StateChangeContext;
import org.nuclos.api.exception.BusinessException;
import org.nuclos.api.provider.StatemodelProvider;
import org.nuclos.api.rule.StateChangeFinalRule;

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

@Rule(name="Auftrag beantragen", description="Prüfung und Weiterführung eines neuen Antrags")
public class Auftragbeantragen implements StateChangeFinalRule {
    public void changeStateFinal(StateChangeContext context) throws BusinessException {

        Auftrag neuerAuftrag = context.getBusinessObject(Auftrag.class);

        // Wurde noch keine Auftragsnummer hinterlegt, kann der Auftrag nicht
        // ausgeführt werden. Wir brauchen eine weitere Prüfung
        if (neuerAuftrag.getAuftragsnr() == null) {
            // Neuer Status: Prüfen (Numeral 55)
            StatemodelProvider.changeState(neuerAuftrag, ProzessAuftragSM.State_55);
        }
        else {
            // Prüfung ok, neuer Status: beauftragt (Numeral 30)
            StatemodelProvider.changeState(neuerAuftrag, ProzessAuftragSM.State_30);
        }
    }
}
```