

Regelausführung selektiv ausschalten

🌐 Sprache: **Deutsch** · [English](#)

🔧 Regelausführung selektiv ausschalten

Endlosschleifen vermeiden: einzelne Regeln kontextbezogen über ThreadLocal-Schalter deaktivieren.

HOW-TO

ENTWICKLER

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Wozu Regeln selektiv ausschalten?](#)
- [1. Schalter in der zu deaktivierenden Regel](#)
- [2. Deaktivierung aus der aktualisierenden Regel](#)
- [Verwandte Seiten](#)

Wozu Regeln selektiv ausschalten?

Manchmal soll verhindert werden, dass bestimmte Regeln (oder Teile davon) laufen – typisch beim gezielten Aktualisieren einzelner Felder aus einer Regel heraus, um **Endlosschleifen** oder Performance-Probleme zu vermeiden. Umgesetzt wird das über regelübergreifende **ThreadLocal**-Variablen.



Vorsicht

Mit Bedacht einsetzen – die fachliche Logik der Regeln darf nicht ausgehebelt werden. Für einfache Fälle ggf. [SaveFlags](#) bevorzugen.

1. Schalter in der zu deaktivierenden Regel

```
public class MyUpdateRule implements UpdateRule {
    // ThreadLocal-Schalter, initial aktiv
    public static final ThreadLocal<Boolean> ACTIVE =
        ThreadLocal.withInitial(() -> Boolean.TRUE);

    public void update(UpdateContext context) throws BusinessException {
        if (ACTIVE.get()) {
            // ... eigentliche Logik ...
        }
    }
}
```

ThreadLocal-Variablen wirken nur im aktuellen Thread/Benutzerkontext – die Regel wird also nicht generell deaktiviert (anders als das Aktiv-Flag der Regel).

2. Deaktivierung aus der aktualisierenden Regel

```
MyBusinessObject mbo = QueryProvider.getById(MyBusinessObject.class, id);
mbo.setMyField(value);
try {
    MyUpdateRule.ACTIVE.set(Boolean.FALSE);    // Regel deaktivieren
    BusinessObjectProvider.update(mbo);
} finally {
    MyUpdateRule.ACTIVE.set(Boolean.TRUE);    // immer wieder aktivieren!
}
```



finally nicht vergessen

Immer im `finally`-Block reaktivieren, damit der Schalter auch im Fehlerfall zuverlässig zurückgesetzt wird.

Verwandte Seiten



Regeln (serverseitig)

Regelkontext.

[Öffnen](#)



SaveFlags

Alternative beim Speichern.

[Öffnen](#)