

# GenerationProvider

🌐 Sprache: **Deutsch** · [English](#)

## 📖 GenerationProvider

Arbeitsschritte (Generierungen) aus Regeln ausführen – typsicher über `execute()`.

REFERENZ

ENTWICKLER

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

### Auf dieser Seite

- [Wozu dient der GenerationProvider?](#)
- [execute\(source, genClass\)](#)
- [Verwandte Seiten](#)

## Wozu dient der GenerationProvider?

Der GenerationProvider führt einen **Arbeitsschritt (Generation)** programmatisch aus – z.B. „Erstelle Rechnung aus Auftrag“.

## execute(source, genClass)

Führt den Arbeitsschritt aus. Quelle und generierte Arbeitsschritt-Klasse (aus der `generation.jar`) müssen typkonform sein; Rückgabe ist das erzeugte Ziel-BO.

```
public static <S extends BusinessObject, T extends BusinessObject> T
    execute(S s, Class<? extends Generation<S,T>> genClass) throws BusinessException;
```

```
import org.nuclos.api.provider.GenerationProvider;

public class AuftragAbschliessen implements UpdateFinalRule {
    public void updateFinal(UpdateContext context) throws BusinessException {
        Auftrag a = context.getBusinessObject(Auftrag.class);
        Rechnung r = GenerationProvider.execute(a, ErstelleRechnungAusAuftragGEN.class);
    }
}
```



### Typsicherheit

Wegen der Typsicherheit führen falsche Quell-/Zieltypen zu Compile-Fehlern in der Regel.

## Verwandte Seiten

### ⚙️ QueryProvider

Daten laden.

[Öffnen](#)



### BusinessObjectProvider

BOs schreiben.

[Öffnen](#)

### ⚙️ DatasourceProvider

Datenquellen ausführen.

[Öffnen](#)

