

Interface Modifiable

🌐 Sprache: **Deutsch** · [English](#)

📖 Interface Modifiable

BOs direkt über `save()` und `delete()` schreiben – der moderne Weg statt der deprecated Provider-Methoden.

REFERENZ

ENTWICKLER

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Was ist Modifiable?](#)
- [save\(\)](#)
- [delete\(id\)](#)
- [Beispiel](#)
- [Verwandte Seiten](#)

Was ist Modifiable?

Seit Nucleos 4.28 stellt die Regel-API das Interface **Modifiable** bereit: Businessobjekte werden direkt über `save()` und `delete()` geschrieben – ohne den [BusinessObjectProvider](#) (dessen statische `insert/update/delete` sind deprecated).

save()

Speichert das BO. Ein neues BO wird angelegt (und erhält eine neue Id), ein bestehendes aktualisiert.

```
void save() throws BusinessException;
```

delete(id)

Löscht das BO mit dieser Id. Statische Methode – ohne Instanz aufrufbar.

```
static <PK> void delete(PK id) throws BusinessException;
```

Beispiel

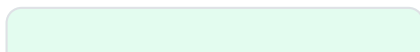
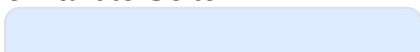
Lesbarer, moderner Regelcode mit `get`, `changeStatus`, `save`, `refresh` (ab 4.32) und statischem `delete`:

```
import org.nuclos.api.exception.BusinessException;
public class UtilsNeu {

    public static void workAuftrag(Long id1, Long id2) throws BusinessException {
        Auftrag auftrag = Auftrag.get(id1);
        auftrag.changeStatus(AuftragSM.State_20);
        auftrag.setName("NeuerName");
        auftrag.save();
        auftrag.refresh(); // ab v4.32

        Auftrag.delete(id2);
    }
}
```

Verwandte Seiten





BusinessObjectProvider

Provider-Alternative.

[Öffnen](#)



QueryProvider

Daten laden.

[Öffnen](#)



Regeln (serverseitig)

Regeln.

[Öffnen](#)