

Sehr große Tabellen performant auslesen und verarbeiten. (Millionen Datensätze)

🌐 Sprache: **Deutsch** · [English](#)

🔧 Sehr große Tabellen performant auslesen und verarbeiten. (Millionen Datensätze)

Große Tabellen performant verarbeiten – in Chunks lesen und OFFSET durch Keyset-Pagination ersetzen.

HOW-TO

ENTWICKLER

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Verwandte Seiten](#)

Zwei Grundregeln, um sehr große Tabellen (Millionen Datensätze) performant zu verarbeiten.

1. Daten **immer in Blöcken (Chunks)** holen.
2. Beim Einlesen der ganzen Tabelle **OFFSET vermeiden** – stattdessen über die letzte ID weiterblättern (Keyset-Pagination).

```
final long CHUNKSIZE = 5000;

long lastID = 0L;

Map<UID, FieldMeta<?>> mpFields = metaProvider.getAllEntityFieldsByEntity(entity);
DbQueryBuilder builder = dataBaseHelper.getDbAccess().getQueryBuilder();

for (;;) {
    DbCondition cond = builder.plainCondition("t.INTID > " + lastID);

    CollectableSearchCondition csc = new CollectableDbCondition(cond);
    CollectableSearchExpression cse = new CollectableSearchExpression(csc);

    Collection<EntityObjectVO<Long>> lstEOs = entityObjectFacade.getEntityObjectsChunk(entity, mpFields.values(),
    null, cse, 0L, CHUNKSIZE - 1);

    if (lstEOs.isEmpty()) {
        return;
    }

    for (EntityObjectVO<Long> eo : lstEOs) {
        ....

        lastID = eo.getPrimaryKey();
    }
}
```

Verwandte Seiten

⚙️ Festlegungen für Nuclos API Entwicklung

API.

[Öffnen](#)

