

# Kapitel 4 – Stammdatenverwaltung

## Ziel dieses Kapitels

Die Stammdaten bilden das Fundament der ERPNext-Lösung. Sämtliche nachgelagerten Prozesse – von der Vertragsverwaltung über die Verarbeitung von Ladevorgängen bis hin zur Rechnungsstellung – greifen auf diese Daten zu.

Dieses Kapitel beschreibt den Aufbau der Stammdaten, deren Herkunft sowie deren Zusammenspiel innerhalb der ERPNext-Lösung.

Nach diesem Kapitel soll der Leser verstehen,

- welche Stammdaten innerhalb ERPNext verwaltet werden,
- aus welchem System diese stammen,
- wie die Synchronisation erfolgt,
- welche Daten manuell gepflegt werden dürfen,
- welche Stammdaten ausschließlich aus Gridware übernommen werden.

## 4.1 Überblick

Die Stammdatenverwaltung umfasst sämtliche dauerhaft benötigten Informationen innerhalb der ERPNext-Lösung.

Hierzu gehören insbesondere:

| Stammdaten            | Beschreibung                             | Quelle | Typ      |
|-----------------------|------------------------------------------|--------|----------|
| <b>Gridware Party</b> | Technischer Vertragspartner aus Gridware | API    | Automat. |
| <b>Customer</b>       | Rechnungsempfänger                       | Party  | Automat. |
| <b>Supplier</b>       | Betreiber bzw. Lieferanten               | Party  | Automat. |

| Stammdaten           | Beschreibung                                     | Quelle   | Typ      |
|----------------------|--------------------------------------------------|----------|----------|
| <b>Contact</b>       | Ansprechpartnerdetails                           | Party    | Automat. |
| <b>Address</b>       | Rechnungs-, Liefer- und Standortadressen         | Party    | Automat. |
| <b>Company</b>       | Eigene Gesellschaft(en) innerhalb ERPNext        | ERPNext  | Manuell  |
| <b>Item</b>          | Artikel für Rechnungs- und Lieferantenabrechnung | ERPNext  | Manuell  |
| <b>Subscription</b>  | Wiederkehrende und variable Gebühren             | ERPNext  | Manuell  |
| <b>Payment Terms</b> | Zahlungsbedingungen                              | ERPNext  | Manuell  |
| <b>SEPA Mandate</b>  | Lastschriftmandate                               | Gridware | Automat. |
| <b>Bank Accounts</b> | Bankkonten für SEPA Lastschriftverfahren         | Gridware | Automat. |

Diese Stammdaten werden in nahezu allen Geschäftsprozessen verwendet.

## 4.2 Gridware Party (Custom)

### Zweck

Die Gridware Party bildet die zentrale Integrationsschicht zwischen der Datenstruktur von Gridware und der kaufmännischen Datenstruktur von ERPNext.

Während ERPNext zwischen Customer und Supplier unterscheidet, verfolgt Gridware einen anderen Ansatz. Dort wird zunächst in Entitäten (Parties) gedacht. Eine Entität repräsentiert entweder eine Person oder eine Organisation und beschreibt damit den eigentlichen Vertragspartner.

Erst innerhalb eines bestimmten Vertrags oder Geschäftsprozesses erhält diese Entität eine fachliche Rolle. Eine Entität kann beispielsweise:

- ausschließlich Kunde sein,
- ausschließlich Lieferant sein oder
- gleichzeitig Kunde und Lieferant sein.

Diese Flexibilität lässt sich mit den Standard-Doctypes von ERPNext nicht abbilden, da Customer und Supplier dort voneinander getrennte Stammdatenobjekte sind.

Aus diesem Grund wurde der Gridware Party Doctype als zentrale Integrationsschicht entwickelt.

### Rolle innerhalb der Systemarchitektur

## Gridware API

↓

[Authentifizierung & Datenabfrage]

↓

Gridware Party (zentrale Integrationsschicht)

↓

↳ Customer (wenn is\_customer=1)

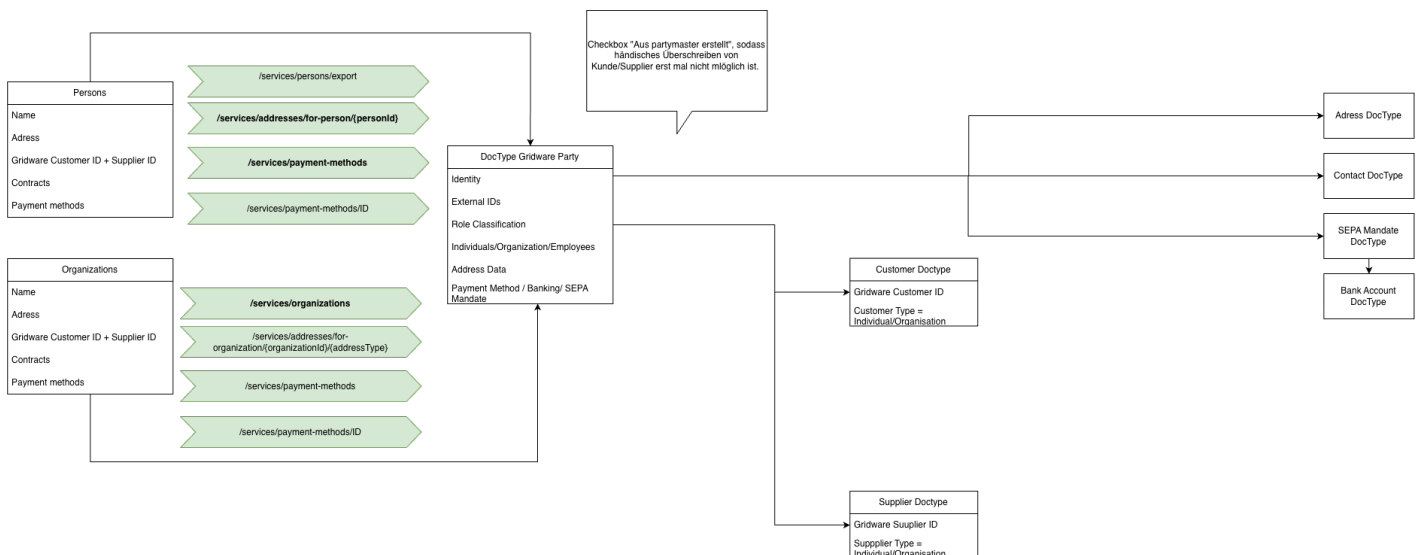
↳ Supplier (wenn is\_supplier=1)

↳ Contact (Ansprechpartner)

↳ Address (Adressen)

↳ Gridware Payment Method (SEPA-Mandate)

Der Gridware Party Doctype steht im Zentrum der Stammdatensynchronisation. Alle späteren Geschäftsprozesse greifen indirekt auf den Gridware Party Datensatz zurück.



## Aufbau des DocTypes

Der Gridware Party Doctype speichert sämtliche grundlegenden Informationen eines Vertragspartners.

The screenshot shows a web application interface for managing a 'Gridware Party'. The top navigation bar includes the 'App Logo', a breadcrumb trail 'Gridware Party > GDW-P-000360', a search bar, and a user profile icon. The main header displays the name 'Andreas Ghazaryan' with navigation controls. On the left, a sidebar titled 'View in Backoffice' lists 'Assigned To', 'Attachments', 'Tags', and 'Share', each with a plus icon. Below this, it shows '0' comments and '0' follows, along with edit history: 'You last edited this · 2 weeks ago' and 'You created this · 1 month ago'. The main content area has tabs for 'Details', 'Address & Contact', and 'Payment Method'. The 'Details' tab is active, showing a 'Connections' section with 'Master Data' (Customer, Supplier) and 'Contract' (Gridware Contract). Below this is the 'Basic Identity' section with fields for Display Name, Salutation, First Name, Last Name, Party Kind, Gridware Subject ID, and checkboxes for 'Is Active' and 'Is Employee'. The 'Contact' section includes Contact ID, Email, and Phone. The 'Role Classification and Links' section has checkboxes for 'Is Customer' and 'Is Supplier', and fields for Gridware and External Customer/Supplier Numbers.

Die Informationen lassen sich in mehrere Bereiche unterteilen.

## Tab 1 – Details

Der Reiter **Details** enthält die grundlegenden Stammdaten sowie die fachliche Klassifizierung der Entität.

## Connections

Im oberen Bereich werden sämtliche Beziehungen der Gridware Party zu anderen ERPNext-Objekten dargestellt.

Hierzu gehören insbesondere:

- **Customer** – Verknüpfung zum kaufmännischen Kunden
- **Supplier** – Verknüpfung zum Lieferanten
- **Gridware Contract** – Zugeordnete Verträge

Links (automatisch generiert):

- └ Customer (custom\_gridware\_party)
- └ Supplier (custom\_gridware\_party)

└ Gridware Contract (über contract\_customers.gridware\_party)

An der Party sind Kundenverträge und Lieferantenverträge getrennt: Customer of Contracts und Supplier of Contracts.

Dadurch ist jederzeit nachvollziehbar, in welchen kaufmännischen Prozessen die Entität verwendet wird.

## Basic Identity

Dieser Bereich beschreibt die eigentliche Identität der Entität.

Gespeichert werden unter anderem:

| Feld                             | Datentyp         | Quelle       | Beschreibung                                          |
|----------------------------------|------------------|--------------|-------------------------------------------------------|
| <code>display_name</code>        | Text             | Gridware API | Anzeigenname (Person oder Organization)               |
| <code>party_kind</code>          | Select           | Gridware API | <b>Organization / Person / User</b>                   |
| <code>gridware_subject_id</code> | Text (eindeutig) | Gridware API | Technischer Identifikator aus Gridware                |
| <code>organization_name</code>   | Text             | Gridware API | Name der Organisation (bei party_kind="Organization") |
| <code>first_name</code>          | Text             | Gridware API | Vorname (bei party_kind="Person"/"User")              |
| <code>last_name</code>           | Text             | Gridware API | Nachname (bei party_kind="Person"/"User")             |
| <code>salutation</code>          | Link→Salutation  | Gridware API | Anrede (Herr/Frau/etc.)                               |
| <code>is_active</code>           | Check            | Gridware API | Aktiv-Status (0/1)                                    |
| <code>is_employee</code>         | Check            | Gridware API | Mitarbeiter-Flag (0/1)                                |

**Wichtig:** Die `gridware_subject_id` ist der eindeutige technische Identifikator. Sie dient als Referenz für sämtliche Synchronisationsprozesse und wird von Gridware als `subjectId` für Personen oder Organisationen vergeben. Änderungen an Namen laufen über diese ID, nicht über den Namen

## Contact

Hier werden die grundlegenden Kommunikationsinformationen gespeichert.

#### Contact Information

|           |        |
|-----------|--------|
| Telephone | Mobile |
| Email     | Fax    |

Save

| Feld       | Datentyp | Quelle                                         |
|------------|----------|------------------------------------------------|
| contact_id | Text     | Gridware API                                   |
| email      | Email    | Gridware API                                   |
| phone      | Text     | Gridware API (Funktion: sanitize_phone_number) |

Diese Daten dienen später als Grundlage für die automatische Erstellung bzw. Aktualisierung von ERPNext Contacts.

## Role Classification and Links

Dieser Bereich definiert die fachliche Rolle der Entität innerhalb des Systems.

Eine Entität kann dabei folgende Rollen besitzen:

- Customer
- Supplier
- Customer und Supplier gleichzeitig

Zusätzlich werden hier die jeweiligen Gridware-spezifischen Nummern gespeichert.

| Feld                     | Datentyp | Wert                  | Beschreibung                         |
|--------------------------|----------|-----------------------|--------------------------------------|
| is_customer              | Check    | 0/1                   | Ist diese Party ein Kunde?           |
| gridware_customer_number | Text     | z.B. "BEC-K-00001058" | Eindeutige Kundennummer aus Gridware |
| external_customer_number | Text     | z.B. "EXT-789"        | Externe Kundennummer (z.B. vom CPO)  |
| is_supplier              | Check    | 0/1                   | Ist diese Party ein Lieferant?       |

| Feld                     | Datentyp                       | Wert                  | Beschreibung                             |
|--------------------------|--------------------------------|-----------------------|------------------------------------------|
| gridware_supplier_number | Text                           | z.B. "BEC-L-00001143" | Eindeutige Lieferantenummer aus Gridware |
| is_employee              | Mitarbeiter                    | 0/1                   |                                          |
| Gridware Employee Number | Mitarbeiternummer aus Gridware | z.B. BEC-M-00001223   |                                          |

Diese Informationen werden in späteren Geschäftsprozessen, insbesondere im Vertragsmanagement und in der Abrechnung, verwendet.

## Tab 2 – Address & Contact

Der Reiter Address & Contact verwaltet sämtliche Adressinformationen einer Gridware Party. Dabei wird bewusst zwischen den Originaldaten aus Gridware und den innerhalb ERPNext erzeugten Stammdaten unterschieden.

Die Adress und Kontaktdatenätze werden sowohl vom DocType Supplier als auch Customer verwendet, sodass es nicht zu Duplikaten kommt. Jeder Adress- und Kontaktdatenatz wird mit einer ID versehen, die direkt zu der Gridware Party zugeordnet werden kann.

### Raw Address

Im oberen Bereich werden sämtliche Adressen gespeichert, die direkt aus Gridware synchronisiert wurden.

Hierzu gehören beispielsweise:

| Feld                | Beschreibung                                |
|---------------------|---------------------------------------------|
| gridware_address_id | Eindeutige Address ID aus Gridware          |
| address_type        | Adresstyp (z.B. "POSTAL", "LOCATION", etc.) |
| street              | Straße                                      |
| house_number        | Hausnummer                                  |
| postal_code         | Postleitzahl                                |
| city                | Stadt                                       |
| country             | Land                                        |
| additional_info     | Zusatzinformationen                         |

Diese Daten stellen den unveränderten Datenbestand aus Gridware dar und dienen als Grundlage für die weitere Verarbeitung innerhalb ERPNext.

## Address & Contact

 Postal Address 

Street\*  
Albertistr.

No.\*  
3

Zip code\*  
78056

City\*  
Villingen-Schwenningen

Country\*  
Germany 

Additional info  
dormakaba EAD GmbH

 > Gridware Party > GDW-P-000360

Search or type a command (⌘ + G)  Help 

 **Andreas Ghazaryan**     Save

**View in Backoffice**  
 Assigned To +  
 Attachments +  
 Tags +  
 Share +  
  
0 · 0 FOLLOW  
You last edited this · 2 weeks ago  
You created this · 1 month ago

Details **Address & Contact** Payment Method

**Raw Address**  
Gridware Address  

| <input type="checkbox"/> | No. | ID  | Type   | Street       | Street No | City       |  |
|--------------------------|-----|-----|--------|--------------|-----------|------------|-------------------------------------------------------------------------------------|
| <input type="checkbox"/> | 1   | 768 | POSTAL | Hessenstraße | 1         | Denzlingen |  |

Add Row

**Address and Contact**  
No address added yet.  
New Address  
No contacts added yet.  
New Contact

Comments

Im unteren Bereich können aus den synchronisierten Rohdaten automatisch ERPNext-Stammdaten erstellt werden.

Hier werden anschließend die eigentlichen ERPNext-Objekte verwaltet:

- Address
- Contact

Dadurch erfolgt eine klare Trennung zwischen den synchronisierten Gridware-Daten und den kaufmännischen Stammdaten innerhalb ERPNext.

Diese Struktur ermöglicht es, Änderungen an den Originaldaten nachzuvollziehen und gleichzeitig die ERPNext-Standards vollständig zu nutzen.

## Tab 3 – Payment Method

Der Reiter Payment Method enthält sämtliche Zahlungsmethoden, die einer Gridware Party zugeordnet sind.

Da eine Entität mehrere Zahlungsmethoden besitzen kann (immer nur eine Methode aktiv, die anderen als Historie), wurde dieser Bereich als Child Table umgesetzt. Mandat werden also nicht überschreiben, wenn Gridware ein neues schickt. Altes bleibt Historie. Neues wird **"Is Default"**. Jede Zeile repräsentiert eine einzelne Gridware Payment Method.

## Gridware Payment Method

Für jede Zahlungsmethode werden unter anderem folgende Informationen gespeichert:

| Feld                 | Beschreibung               | Quelle               |
|----------------------|----------------------------|----------------------|
| gridware_payment_id  | Eindeutige ID aus Gridware | Gridware API         |
| payment_type         | z.B. "SEPA", "CARD", etc.  | Gridware API         |
| erpnext_bank_account | Link→Bank Account          | Automatisch gemappt  |
| bank_name            | Name der Bank              | Gridware API         |
| iban                 | Kontonummer                | Gridware API         |
| bic                  | Bank Clearing Code         | Gridware API         |
| erpnext_sepa_mandate | Link→SEPA Mandate          | Automatisch erstellt |
| sepa_mandate_id      | SEPA-ID aus Gridware       | Gridware API         |
| created_date         | Erstellungsdatum           | Gridware API         |
| is_default           | Standard-Zahlungsmethode   | Gridware API         |

App Logo
Gridware Party
GDW-P-000360
Search or type a command (⌘ + G)
Help
A

### Andreas Ghazaryan

View in Backoffice
Assigned To
Attachments
Tags
Share

Details
Address & Contact
Payment Method

| No. | Gridware Payment ID | ERPNext Bank Account  | Bank Name     | ERPNext SEPA Mandate | Is Default |  |
|-----|---------------------|-----------------------|---------------|----------------------|------------|--|
| 1   | 355                 | CUST-2026-00268 - ... | Deutsche bank | A6F8B799DD029CFF...  |            |  |

Add Row

0
0
FOLLOW

You last edited this · 2 weeks ago

You created this · 1 month ago

### Comments

A
Type a reply / comment

### Activity

+ New Email

- You last edited this · 2 weeks ago
- You created this · 1 month ago

### Editing Row #1

Insert Below
Insert Above
Duplicate
Move

### Gridware Payment Method

Gridware Payment ID
355

ERPNext Bank Account
CUST-2026-00268 - Deutsche Bank

Bank Name
Deutsche bank

IBAN
DE06680700240270543200

BIC
DEUTDEDBFRE

ERPNext SEPA Mandate
A6F8B799DD029CFF91FC5B6C6C6D19D1

SEPA Mandate ID
A6F8B799DD029CFF91FC5B6C6C6D19D1

Payment Type
SEPA

Created at
20.04.2026 10:06

Europe/Berlin

☐ Is Default

Shortcuts:
Ctrl + Up
Ctrl + Down
ESC
Insert Below

## Zweck

Dieser Bereich bildet die Grundlage für sämtliche Zahlungsprozesse innerhalb der ERPNext-Lösung.

Die gespeicherten Informationen werden insbesondere verwendet für:

- Erstellung von SEPA-Mandaten
- Zuordnung von Bankkonten
- Generierung von SEPA-XML-Dateien
- Zahlungsabwicklung

- Payment Entries

Durch die Verknüpfung zwischen der **Gridware Payment ID** und dem ERPNext Bank Account kann jederzeit nachvollzogen werden, welche Zahlungsmethode aus Gridware welchem Bankkonto und welchem SEPA-Mandat in ERPNext entspricht.

## Typische Anwendungsfälle

Ein Gridware Party Datensatz wird unter anderem verwendet für:

- Synchronisation neuer Kunden
- Synchronisation neuer Lieferanten
- Aktualisierung von Kontaktdaten
- Verknüpfung mit Gridware Contracts
- Zuordnung von Payment Methods
- Erstellung von Customer- und Supplier-Beziehungen
- Grundlage für Customer Portal Benutzer
- Identifikation von Vertragspartnern während der Abrechnung

# 4.3 Customer

## Zweck

Der Customer repräsentiert den kaufmännischen Rechnungsempfänger innerhalb ERPNext. Ein Customer kann sowohl eine Privatperson als auch ein Unternehmen sein.

Der Customer bildet die Grundlage für:

- Angebote
- Aufträge
- Rechnungen
- Payment Entries
- Subscriptions
- Customer Portal
- Customer Usage Data (für Ladevorgänge)

## Herkunft

Customer werden nicht in ERP Next manuell angelegt. Sie entstehen durch die Synchronisation der entsprechenden Organisation oder Person aus Gridware.

Während der Synchronisation erfolgt zusätzlich die Verknüpfung mit:

- Contacts

- Addresses
- Contracts
- Payment Methods

## Besondere Anpassungen

Im Rahmen der Implementierung wurden Customer um projektspezifische Informationen erweitert.

Hierzu gehören unter anderem:

### Custom Fields

| Feld                            | Datentyp            | Beschreibung                    |
|---------------------------------|---------------------|---------------------------------|
| custom_gridware_party           | Link→Gridware Party | Referenz zur technischen Party  |
| custom_gridware_customer_number | Text                | Eindeutige Kundennummer (Index) |
| custom_sepa_mandate_reference   | Link→SEPA Mandate   | Standard-Lastschriftmandat      |

## 4.4 Supplier

### Zweck

Supplier repräsentieren Betreiber oder andere Leistungserbringer, welche später gegenüber becharged abgerechnet werden.

Sie bilden die Grundlage für:

- Purchase Invoices
- Supplier Usage Data
- Lieferantenabrechnungen
- Rückvergütungen

### Herkunft

Wie Customer entstehen auch Supplier auf Basis der Informationen aus Gridware. Die Zuordnung erfolgt über den jeweiligen Gridware Contract. Wie Customer entstehen auch Supplier auf Basis der Informationen aus Gridware.

## 4.5 Contacts

Contacts enthalten sämtliche Ansprechpartner eines Customers oder Suppliers. Ein Unternehmen kann mehrere Ansprechpartner besitzen.

Typische Rollen:

- Primary Contact
- Billing Contact
- Technical Contact

Bei der Erstellung von Verkaufsdokumenten wird automatisch der passende Ansprechpartner verwendet.

## 4.6 Addresses

Adressen werden getrennt von Customer und Supplier verwaltet.

Dadurch können mehrere Adressen einem Geschäftspartner zugeordnet werden.

Beispiele:

- Rechnungsadresse
- Lieferadresse
- Firmenstandort
- Ladepark
- Baustelle

Für die Druckformate gelten projektspezifische Regeln bezüglich Customer Name, Address Title und Contact Person.

*(Verweis auf Kapitel "Druckformate")*

## 4.7 Items

Artikel bilden die Grundlage sämtlicher Rechnungs- und Lieferantenabrechnungen. Je nach Geschäftsmodell werden unterschiedliche Artikel verwendet.

Beispiele:

- Ladevorgänge
- Monatliche Grundgebühren
- Hardware
- Dienstleistungen

- Erstattungen

Die Zuordnung erfolgt später über die Vertragslogik (Becharged Settings → Contract Type Mappings). Statische Gebühren und Verbrauch können auf derselben Rechnung stehen

## Zuordnung via Contract Type Mapping

**Quelle:** `becharged/becharged/utils.py` Funktion `get_item_code_from_mapping`

```
# In Becharged Settings: contract_type_mappings (Child Table)
settings = frappe.get_single("Becharged Settings")

for mapping in settings.contract_type_mappings:
    # Contract Type + VAT → Item Code
    if (mapping.contract_type == "USAGE_POSTPAID" and
        mapping.vat_rate == 19):
        return "CHARGING-POSTPAID-19" # Item Code
```

### Tabelle (Beispiel):

| Contract Type  | VAT % | Item Code            | Beschreibung                  |
|----------------|-------|----------------------|-------------------------------|
| USAGE_POSTPAID | 19    | CHARGING-POSTPAID-19 | Ladung, umsatzsteuerpflichtig |
| USAGE_POSTPAID | 0     | CHARGING-POSTPAID-0  | Ladung, steuerfrei            |
| AUTHORIZATION  | 19    | CHARGING-AUTH-19     | Autorisierung                 |
| USAGE_ADHOC    | 19    | CHARGING-ADHOC-19    | Ad-hoc-Ladung                 |
| (übersprungen) | 19    | (keiner)             | Nicht verrechenbar            |

## 4.8 Subscriptions

Subscriptions bilden die Grundlage für wiederkehrende Abrechnungen.

Sie werden insbesondere verwendet für:

- monatliche Grundgebühren
- Vertragsgebühren
- Servicegebühren

Zusätzlich werden im Rahmen der monatlichen Abrechnung die entsprechenden Verbrauchsdaten ergänzt. In der CSV-Phase sind Subscriptions weiterhin die zentrale Abrechnungseinheit (siehe CSV-Kapitel).

## 4.9 Payment Terms & SEPA

Für jeden Customer können Zahlungsbedingungen und SEPA-Informationen hinterlegt werden.

Die SEPA-Stammdaten stammen grundsätzlich aus Gridware und werden nach ERPNext synchronisiert.

Sie bilden die Grundlage für:

- SEPA-XML
- Payment Entries
- Lastschriftverfahren

## 4.10 Synchronisationsprozesse:

### Daily Scheduler Jobs

**Quelle:** `hooks.py`

```
scheduler_events = {
    "daily": [
        "becharged.becharged.utils.fetch_all_contract_parties",
        "becharged.becharged.utils.update_gridware_usage_suppliers",

        "becharged.becharged.doctype.gridware_usage.gridware_usage.retry_failed_usage_completion"
    ]
}
```

#### 1. `fetch_all_contract_parties()` (täglich)

**Quelle:** `utils.py`

```
def fetch_all_contract_parties():
    """
```

Findet alle Gridware Contracts wo:

- gridware\_supplier\_party gesetzt, aber supplier NICHT gesetzt
- contract\_customers.gridware\_party gesetzt, aber customer NICHT gesetzt

Erstellt dann die fehlenden Customer/Supplier Links

"""

### Logik:

```
SELECT DISTINCT gc.name
FROM `tabGridware Contract` gc
LEFT JOIN `tabGridware Contract Customer` gcc ON gcc.parent = gc.name
WHERE
    (gc.gridware_supplier_party IS NOT NULL
     AND gc.gridware_supplier_party != ''
     AND (gc.supplier IS NULL OR gc.supplier = ''))
OR
    (gcc.gridware_party IS NOT NULL
     AND gcc.gridware_party != ''
     AND (gcc.customer IS NULL OR gcc.customer = ''))
```

## 2. `update_gridware_usage_suppliers()` (täglich)

Quelle: `utils.py`

```
def update_gridware_usage_suppliers():
    """
    Updated supplier + gridware_supplier_party in allen Gridware Usage
    basierend auf deren Gridware Contract
    """
```

### Logik:

```
SELECT gu.name, gc.supplier, gc.gridware_supplier_party
FROM `tabGridware Usage` gu
INNER JOIN `tabGridware Contract` gc ON gu.gridware_contract = gc.name
WHERE
    (gc.gridware_supplier_party IS NOT NULL
     AND (gu.gridware_supplier_party IS NULL OR gu.gridware_supplier_party = ''))
OR
```

```
(gc.supplier IS NOT NULL  
AND (gu.supplier IS NULL OR gu.supplier = ''))
```

## 4.11 Typische Fehler

| Problem        | Ursache                               | Lösung                             |
|----------------|---------------------------------------|------------------------------------|
| Customer fehlt | Synchronisation nicht durchgeführt    | API-Synchronisation prüfen         |
| Supplier fehlt | Gridware Party nicht zugeordnet       | Contract prüfen                    |
| Contact fehlt  | Ansprechpartner nicht synchronisiert  | API erneut ausführen               |
| Adresse falsch | Änderungen nur in ERPNext vorgenommen | Änderungen in Gridware durchführen |
| SEPA fehlt     | Mandat nicht synchronisiert           | Payment Method prüfen              |

## 4.12 Best Practices

Für die Pflege der Stammdaten gelten folgende Grundregeln:

- Stammdaten grundsätzlich im führenden System pflegen.
- Manuelle Änderungen an synchronisierten Feldern vermeiden.
- Änderungen nach Möglichkeit über die vorgesehenen Synchronisationsprozesse durchführen.

## Zusammenfassung

Die Stammdatenverwaltung bildet die Grundlage aller weiteren Geschäftsprozesse. Eine konsistente Pflege und Synchronisation der Stammdaten ist Voraussetzung für eine fehlerfreie Vertragsverwaltung, Abrechnung und Finanzbuchhaltung.

Revision #6

Created 2026-07-13 12:57:07 CEST by Celia Lohner

Updated 2026-09-01 11:20:01 CEST by Celia Lohner