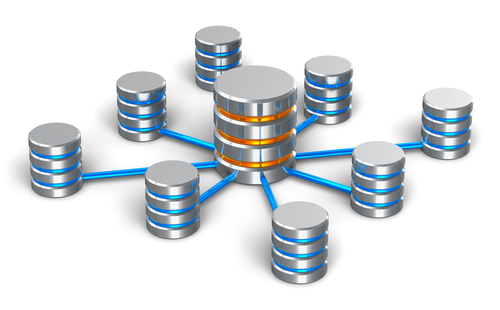
s

Datenmodell einer Inventarisierungslösung

Semesterprojekt



Fach: Datenbanken

Dozent: Ueli Niederer

Autoren: Larissa Fitze, Patrick Keist, Francesco Rauseo, Ronny Wyss

Version: 12.09.2018

Alle Rechte vorbehalten ©

**Inhaltsverzeichnis**

[1. Ausgangslage 4](#_Toc524544433)

[2. Funktionale Anforderungen 5](#_Toc524544434)

[2.1.1. Kunde 5](#_Toc524544435)

[2.1.2. Point of delivery (PoD) 5](#_Toc524544436)

[2.1.3. Location 5](#_Toc524544437)

[2.1.4. Device 5](#_Toc524544438)

[2.1.5. Netzwerkinterface 5](#_Toc524544439)

[2.1.6. Logging 5](#_Toc524544440)

[2.1.7. Administrative Credentials 5](#_Toc524544441)

[2.1.8. Abrechnung & Zahlungen 5](#_Toc524544442)

[3. Datenmodell 6](#_Toc524544443)

[3.1. Geschäftsprozesse 6](#_Toc524544444)

[3.2. Entity Relationship Model 6](#_Toc524544445)

[4. Tabellenreferenz 7](#_Toc524544446)

[4.1. Location 7](#_Toc524544447)

[4.1.1. Device 7](#_Toc524544448)

[4.1.2. Device Types 8](#_Toc524544449)

[4.1.3. Device has operatingsystem 8](#_Toc524544450)

[4.1.4. Operatingsystem 8](#_Toc524544451)

[4.1.5. Network 9](#_Toc524544452)

[4.1.6. Interfaces 9](#_Toc524544453)

[4.1.7. Log 9](#_Toc524544454)

[4.1.8. Customer 10](#_Toc524544455)

[4.1.9. Person 10](#_Toc524544456)

[4.1.10. Point of Delivery 10](#_Toc524544457)

[4.1.11. Address 11](#_Toc524544458)

[4.1.12. PostalCode 12](#_Toc524544459)

[4.1.13. Contact 12](#_Toc524544460)

[4.1.14. Produkte 12](#_Toc524544461)

[4.1.15. Produktegruppe 12](#_Toc524544462)

[4.1.16. Dienstleistung 13](#_Toc524544463)

[4.1.17. Abrechnung 13](#_Toc524544464)

[4.1.18. Credentials 13](#_Toc524544465)

[4.1.19. Devices has Credentials 14](#_Toc524544466)

[Anhang 15](#_Toc524544467)

# Ausgangslage

Wir bekamen den Auftrag eine Inventarisierungslösung für das IT System der neuen Firma Muster IT Services AG zu erstellen. Dies soll IT Dienstleistungen der Muster AG, aber auch für weitere Kunden, anbieten. Das Netzwerkmanagement wurde bisher in einer Excel Tabelle geführt. Wir sollen dies mit einer professionellen Datenbank ablösen.

Die neue Datenbank soll dabei im Wesentlichen folgende Aspekte abdecken:

* Planung und Dokumentation des Netzwerkes soll in der Datenbank erfolgen. Aus den so erzeugten Daten sollen dann die Konfigurationsdaten für bestimmte Geräte wie z.B. Switches generiert werden können, um so das Deployment zu vereinfachen. Zudem soll durch die Datenbank auch die Kapazitätsplanung vereinfacht werden
* Abrechnung von Dienstleistungen gegenüber Kunden
* Troubleshooting soll durch die systematische Erfassung der Netzwerkstrukturen sowie der Erfassung des Logins beschleunigt werden.

# Funktionale Anforderungen

### Kunde

Ein Kunde verfügt über einen oder mehrere «point of deliverys»(PoDs). Die Rechnungsadresse und die Kontaktperson können zwischen den PoD’s variiren.

### Point of delivery (PoD)

Ist die Infrastruktur des Kunden. Ein PoD wird in mehrere Locations aufgeteilt. Ein PoD kann mehrere Kontakten zugeordnet werden, die wiederum eine Priorität erhalten sollen.

### Location

Eine Location weist eine Adresse auf und verfügt über eine Reihe von Geräten, die ihr zugeordnet werden kann.

### Device

Ein Device kann physisch als auch virtuell sein. Dieser dient als Inventarbasis für sämtliche Geräte (intern als auch für Kunden).

Dieser weist ein Hostname und Domäne, so kann man den FQDN nachstellen. Dazu wird auch eine IP Adresse vergeben.

### Netzwerkinterface

Jedes Device hat Minimum eine oder n Interfaces. Jedes Interface ist mit einem anderem Interface verbunden und aktiv.

### Logging

Geräte sollen überwacht werden können. Zu diesem Zweck soll für jedes Gerät ein Log angelegt werden können, welches den Zeitpunkt, den Level und die eigentliche Lognachricht speichern kann.

### Administrative Credentials

Es soll möglich sein, Sets von Credentials (Benutzername, Passwort, SNMP Community) anzulegen und einem oder mehreren Geräten zuzuweisen. Dabei können Geräte mehrere Credentials haben. Credentials ihrerseits können in mehreren Geräten verwendet werden.

### Abrechnung & Zahlungen

Es soll Abrechnungen für PoD’s erstellt werden können. Die Abrechnung können diverse zu verrechnende Positionen enthalten. Diese reichen von Standard-Dienstleistungen nach (Stunden-)Aufwand, über einzelne Hard- und Software-Produkte bis hin zu Artikeln mit freiem Text und Preis. Die Positionen können, müssen aber nicht, einem Interface, einem Device oder einer Location zugeordnet werden. Für die Buchhaltung soll im System ein Kundenkonto geführt werden. Zahlungen können dabei ohne spezifischen Bezug zu einer Rechnung erfolgen.

# Datenmodell

## Geschäftsprozesse

* Sicherheit

Die Geräte sollen alle überwacht werden, zu diesem Zweck haben wir eine Log – Tabelle angelegt, welche Lognachrichten speichert. Diesen Nachrichten kann man ein Schweregrad Leven hinzufügen.

## Entity Relationship Model

# Tabellenreferenz

## Location

Die Tabelle Location beinhaltet eine Adresse.

Zusätzlich muss die Location eine Bezeichnung, Gebäudenummer und Raumanschrift, haben

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK | location\_id | INT | Nein |  |
| FK | address\_fk | INT | Nein |  |
|  | designation | VARCHAR(45) | Nein |  |
|  | building | INT | Nein |  |
|  | room | INT | Nein | Raumnummer |

### Device

Ein Device kann verschiedene Geräte sein z.B.: ein TV, ein PC, ein Switch etc.. Es ist immer mit einem Interface, Log verknüpft.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK | device\_id | INT | Nein |  |
| FK | location\_fk | INT | Nein |  |
| FK | deviceTypes\_fk | INT | Nein |  |
|  | inventoryDate | DATE | Nein |  |
|  | deactivateDate | DATE | Nein |  |
|  | hostname | VARCHAR(255) | Nein |  |
|  | domain | VARCHAR(255) | Nein |  |
|  | description | VARCHAR(255) | Ja |  |

### Device Types

Ein Device Type gibt Auskunft darüber, um was für einen Gerätetyp es sich genau handelt.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK | deviceTypes\_id | INT | Nein |  |
|  | manifacturer | VARCHAR(255) | Nein |  |
|  | model | VARCHAR(255) | Nein |  |
|  | version | VARCHAR(255)T | Ja |  |

### Device has operatingsystem

Dies ist eine Hilfstabelle, da sowohl ein Device Types mehre Operatingsystem haben kann und umgekehrt.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK / FK | deviceTypes\_fk | INT | Nein |  |
| PK / FK | operatingsystem\_fk | INT | Nein |  |

### Operatingsystem

Hier werden die Namen der Betriebssystem gesammelt, welche genutzt werden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK | operatingsystem \_id | INT | Nein |  |
|  | operatingsystemName | VARCHAR(255) | Nein |  |
|  | model | VARCHAR(255) | Nein |  |
|  | version | VARCHAR(255)T | Nein |  |

### Network

Hier werden die Netze gekennzeichnet, aufgeteilt und zugewiesen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK | Network\_id | INT | Nein |  |
|  | Subnet | VARCHAR(15) | Nein |  |
|  | Mask | VARCHAR(15) | Nein |  |
|  | Vlan | INT | Nein |  |
|  | Description | VARCHAR(255) | Ja |  |

### Interfaces

Hier werden allen Dervices eine IP vergeben und zugeortnet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK | Interface\_id | INT | Nein |  |
| FK | Network\_id\_fk | INT | Nein |  |
| FK | Device\_id\_dk | INT | Nein |  |
|  | Ip:Adress\_v4 | VARCHAR(15) | Nein |  |
|  | isFullDuplex | BIT | Nein | Default1 |
|  | Bandwith | INT | Ja |  |
|  | Description | VARCHAR(255) | Ja |  |

### Log

Hier werden die Logs von einem Device gespeichert. Einem Device können mehre Logs hinzugefügt werden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkung** |
| PK | log \_id | INT | Nein |  |
|  | logMessage | VARCHAR(100) | Nein |  |
|  | timestamp | DateTIME | Nein |  |
|  | Level | ENUM(‘Low’,’middle’,’high’) | Nein | definiert den Schweregrad der Lognachricht |
| FK | Devices\_fk | INT | Nein |  |

### Customer

Hier können Kunden gespeichert werden. Diese werden typischerweise einer Person zugeordnet und haben jeweils eine eigene Adresse.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
| FK | AdressId | INT | Nein | Die Hauptadresse des Kunden |
| FK | PodId | INT | Nein | Ein Kunde benötigt einen oder mehrere PoD’s |
|  | Name | VARCHAR(45) | Nein | Kundennamen |
|  | Tel | VARCHAR(20) | Ja |  |
|  | Email | VARCHAR(30) | Ja |  |
|  | Url | VARCHAR(30) | Ja | Homepage des Kunden |

### Person

Eine Relation für die Erfassung von Personen. Dies zur Abgrenzung zu Firmen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Person\_Id | INT | Nein |  |
|  | Firstname | VARCHAR(45) | Nein |  |
|  | Lastname | VARCHAR(45) | Nein |  |

### Point of Delivery

Einem Kunden kann ein oder mehrere PoD’s zugewiesen werden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
| FK | CustomerId | INT | Nein | Ein Kunde kann einen oder mehrere Pods benötigen. |
| FK | ContactId | INT | Nein | Ein PoD hat einen oder mehrere Kontakte. |
| FK | LocationId | INT | Nein | Ein PoD hat einen oder mehrere Locations. |
|  | Designation | VARCHAR(45) | Nein | Eine Bezeichnung für den PoD |
|  | TimeZone | TIME | Nein | UTC Zeit in Stunden und Minuten und Sekunden ’03:30:00’ |
|  | TimeZonePositive | TINYINT | Nein | Ist die Zeitzone im positiven Bereich (z.B. UTC +01:00) dann 1 sonst 0 |
|  | NTPServerIP | VARCHAR(20) | Ja |  |

### Address

Eine Tabelle für die Erfassung von Adressen. Diese ist verknüpft mir der Relation Postal Code.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
|  | Name | VARCHAR(45) | Nein |  |
|  | GivenName | VARCHAR(45) | Ja |  |
|  | Street | VARCHAR(45) | Nein |  |
| FK | PostalCodeId | INT | Nein |  |
|  | Country | VARCHAR(5) | Nein | EU Länderkürzel nach ISO 3166 |
|  | Additive | VARCHAR(45) | Ja | Adresszusatz |
|  | PoBox | INT | Nein | Postfach falls vorhanden |

### PostalCode

Eine Relation für die Erfassung der benötigten Postleitzahlen und deren Orte.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
|  | PostalCode | VARCHAR(10) | Nein |  |
|  | City | VARCHAR(20) | Nein |  |

### Contact

Eine Relation für die Erfassung von Kontakten. Ein Kontakt wird mit einer Person verknüpft.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
| FK | AdressId | INT | Nein |  |
|  | Priority | ENUM | Nein | Wahl zwischen ‘First Priority’, ‘SecondPriority’ und ’Emergency Contact’ |
|  | TelNr | VARCHAR(20) | Nein |  |
|  | Email | VARCHAR(30) | Ja |  |

### Produkte

Eine Relation für die Erfassung von Produkte / Artikeln

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
| FK | produktegruppe | INT | Nein |  |
|  | Artikelname | VARCHAR(45) | Nein |  |
|  | preis | FLOAT | Nein |  |

### Produktegruppe

Die Produkte/Artikel werden in eine Produktegruppen kategorisiert (Hardware/Software/sonstige Artikeln).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
| FK | abrechnung | INT | Nein |  |
|  | hardware | VARCHAR(45) | Ja | Das Produkt ist entweder eine Hardware oder eine Software |
|  | software | VARCHAR(45) | ja |  |
|  | sonstigeArtikel | VARCHAR(45) | ja |  |

### Dienstleistung

Es soll auch optional eine Dienstleistungsdauer angeboten werden können.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
| FK | abrechnung | INT | Nein |  |
|  | stundenaufwand | INT | Ja |  |

### Abrechnung

Die Abrechnung erfolgt über das Kundenkonto. Optional können Ort, Gerät und Interface ausgewählt werden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
| FK | Kundenkonto | INT | Nein |  |
| FK | Location | INT | Ja |  |
| FK | Device | INT | Ja |  |
| FK | interface | INT | Ja |  |

### Credentials

Für die Konfiguration an den verschiedenen Geräten sollen Accounts erstellt werden können.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| PK | Id | INT | Nein |  |
|  | Benutzername | VARCHAR(45) | Nein |  |
|  | passwort | VARCHAR(45) | Nein |  |
|  | snmp | VARCHAR(45) | Nein |  |

### Devices has Credentials

Die Verlinkung zu den verschiedenen Geräten mit verschiedenen Accounts mit dem selben Benutzername soll realisiert werden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | **Attributname** | **Datentyp** | **Optional** | **Bemerkungen** |
| FK | Devices\_id | INT | Nein |  |
| FK | Credentials\_id | VARCHAR(45) | Nein |  |

# Stored Procedure

## LogClear

Diese Stored Procedure löscht die in der v\_logentries enthaltene Datensätze.

Löschen folgt mittels dem Client und der logentreies – ID.

DELIMITER //

CREATE PROCEDURE LogClear

(

IN \_logentries\_id INT

)

BEGIN

DELETE FROM v\_logentries

WHERE \_logentreies\_id = logentrie\_id;

UPDATE Log

SET is\_acknowledged = 1

WHERE log\_is = \_logentries\_id;

END //

DELIMITER ;

## LogMessageAdd

Diese Stored Procedure wird genutzt, um Logeinträge in die Tabelle Log zu schreiben.  
DELIMITER //

CREATE PROCEDURE LogMessageAdd

(  
 IN device\_id INT, log\_level ENUM('Low','Middle','High'), logMessageInsert VARCHAR(255)  
)  
BEGIN  
 INSERT INTO log ( device\_fk, `timestamp`, logMessage, `level`, is\_acknowledged)  
 VALUE  
 (device\_id, timestamp(now()), logMessageInsert, log\_level, 0);  
END //  
DELIMITER ;

# Views

## view\_freiesInterface

Diese View bietet die Funktionalität alle freie Interfaces auszugeben.

CREATE VIEW

view\_freiesInterface (LocationName, Hostname, Portnummer, Medium, Geschwindigkeit)

AS

SELECT

CONCAT\_WS(building, room),

device.hostname,

deviceport.description,

transportmedium.description,

bandwith

FROM

device INNER JOIN location

ON location\_fk=location\_id

INNER JOIN deviceport

ON device\_fk = device\_id

INNER JOIN transportmedium

ON device\_fk = device\_id

INNER JOIN interface

ON interface.device\_fk = device\_id

WHERE interface.is\_in\_use = 0

;

# Anhang

* Arbeitsjournal
* Skrip\_DDL
* Skript\_DML