



## Konzeptionelles Schema DBAP4

- Version 2.0 -

08.12.2014, Henning Budde

Gregor Büchel

Thomas Partsch

# 1. Entwicklung des konzeptionellen Schemas

Dieser Bericht baut auf den Berichten DBAP1 Anforderungsanalyse und DBAP2 Deploymentanalyse auf. Er erweitert und verändert das Schema der Anforderungsanalyse. Teile des Schemas der Anforderungsanalyse wurden angepasst an die dynamischen Anforderungen des Projekts und der Projektpartner.

Das konzeptionelle Schema wird als globales Schema für die SensorCloud entwickelt. Wie in föderierten Datenbanksystemen üblich, muss in einer späteren Inkarnation der SensorCloud keiner der Datenbanksysteme das gesamte konzeptionelle Schema implementieren, sondern es reicht, wenn einzelne Knoten (Cloud-Knoten, Gateway, ...) Ausschnitte (abgeleitete Teile) des Schemas umsetzen. Im aktuellen Zustand des FDBS der SensorCloud ist das konzeptionelle Schema bezüglich der Entitäten vollständig als globales JSON-Schema implementiert.

Den bisher überwiegend eigenschaftslosen Entitäten werden nun Eigenschaften, Beziehungen und Integritätsbedingungen hinzugefügt. In der Version 2.0 umfasst das konzeptionelle Schema 69 Entitäten. Für das Konzeptionelle Schema wurde einer Versionshistorie eingefügt (vgl. Kapitel 5).

## 1.1 Abbildung von Datentypen

Aufgrund der heterogenen Auswahl der möglichen Datenbankmanagementsystemen (DBMS) existieren keine einheitlichen Bezeichnungen für elementare Datentypen. Um nicht für jedes Attribut sämtliche möglichen Datentypen anzugeben, wird folgende Abbildung der Datentypen festgelegt. In der linken Spalte wird der universelle Datentyp angegeben, den wir im globalen Schema verwenden, in der rechten Spalte wird dieser äquivalent auf andere Datentypen abgebildet.

SensorCloud Datentyp	Äquivalente Datentypen
text	text (Cassandra) varchar (relat. DBMS) String (DB4o) UUID (auch wenn einzelne DBMS UUID als eigenständigen elementaren Datentyp unterstützen, wird er hier nur auf text abgebildet)
int	int (Cassandra, relat. DBMS, DB4o)
decimal	number (Cassandra, relat. DBMS) decimal(p,q) (RDBMS) float, double (DB4o)
timestmp	BigInt (Cassandra, Postgres) Timestamp (RDBMS) Long (DB4o)

relat. DBMS := relationale Datenbankmanagementsystem : PostgreSQL, MySQL, SQLite,

## 2. Konzeptionelles Schema

Nicht jedes unterstützte Datenmodell des föderierten Datenbanksystems kann Beziehungen mit Hilfe von Primär- und Fremdschlüsseln aufbauen. Trotzdem werden hier konzeptionelle Primär- und Fremdschlüssel-Beziehungen angegeben, die teilweise nicht mit DDL-Mitteln sondern nur mit Methoden der DBS-Anwendungsprogrammierung implementiert sind.

Die Entitäten können nach diesem Schema entweder direkt als ein relationales Datenbankmodell implementiert werden, d.h. aus den Entitäten werden Tabellen oder als ein NoSQL-Datenbankmodell in Form von ColumnFamilies implementiert werden, die das entsprechende Pendant zu Tabellen der NoSQL Datenbank Cassandra sind. In einem objektorientierten Datenbankmodell, wie es z.B. von dem OODBMS DB4o unterstützt wird, können die Entitäten als Entitätsklassen implementiert werden.

### 2.1 Entitätengruppe: Sensor

#### 2.1.1 Entität: Sensor

Diese Entität beschreibt einen reell vorhandenen Sensor der SensorCloud

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenID	text	Sensor ID Eindeutige ID für einen Sensor	PRIK
SenSenProID	text	Produkt ID Beschreibt welchem Sensorprodukt dieser Sensor angehört	FKEY: SensorProdukt.SenProID
SenSenTypID	text	Sensor Typ ID Charakterisiert den Typ des Sensors	FKEY: SensorTyp.SenTypID
SenRauID	text	Raum ID Beschreibt, in welchem Raum sich ein Sensor befindet (der Raum befindet sich in einem Flur, dieser auf einer Etage, diese wiederum in einem Gebäude, ...)	FKEY: Raum.RauID
SenNutStaID	text	Nutzerstammdaten ID Verknüpft den Sensor mit einem Nutzer(-stammdaten) der SensorCloud	FKEY: NutzerStammdaten.NutStaID
SenLocMasID	text	LocationMaster ID Verknüpft einen Sensor mit dem LocationMaster, an dem er angemeldet ist	FKEY: LocationMaster.LocMasID
SenSouID	text	Sensor Source ID Herstelleradresse des Sensors	/
SenBez	text	Sensor Bezeichnung Bezeichnung des Herstellers für den Sensor / Friendly Name für	/

		einen Sensor	
SenPos	text	Sensor Position Beschreibt die Position des Sensors im Raum	/
SenDatEin	timestmp	Sensor Datum Eintritt Anmeldedatum des Sensors am LM	/

## 2.1.2 Entität: SensorProdukt

Diese Entität beschreibt ein Sensorprodukt. Ein Sensorprodukt kann beispielsweise ein Temperatur Sensor einer Firma X sein. Eine Firma Y produziert einen andersartigen Temperatur-Sensor und hat damit ihr eigenes Sensorprodukt. Beide Firmen stellen Produkte vom gleichen Sensortyp her, aber entwickeln unterschiedliche Produkte.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenProID	text	Sensor Produkt ID Eindeutige ID für ein Sensorprodukt	PRIK
SenProSenTypID	text	Sensortyp ID Verknüpfung zu einem Sensortypen, z.B. Typ Temperatur oder Typ Luftgüte, etc	FKEY: SensorTyp.SenTypID
SenProAdrID	text	Adress ID Verknüpfung zu einer Adresse	FKEY Adresse.AdrID
SenProHerNum	text	Sensor Produkt Herstellernummer Produktnummer des Herstellers	/
SenProHer	text	Sensor Produkt Hersteller Name des Herstellers des SensorProdukts (Bsp: HomeMatic, eq3)	/
SenProSem	text	Sensor Produkt Semantik Beschreibung eines Sensors insbesondere dessen Messwertaufbau (s. DBAP3 Semantische Analyse). Die Beschreibung ist im JSON-Format hinterlegt.	/

## 2.1.3 Entität: SensorTyp

Jedem Sensor (Produkt) kann ein bestimmter Typ zugeordnet werden, der diesen grundsätzlich beschreibt. Beispieltypen können sein: Luftfeuchte-, Luftgüte-, Temperatursensoren, Strommesser, Luftdruckmesser, ...

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenTypID	text	Sensor Typ ID	PRIK

		Eindeutige ID für einen Sensortyp	
SenTypDef	text	Sensor Typ Definition Beschreibt Typen eines Sensors als Fließtext	/
SenTypBez	text	Sensor Typ Bezeichnung Kurze Beschreibung eines Sensortyps	/

## 2.1.4 Entität: SensorKonfiguration

Diese Entität beschreibt die Konfiguration eines Sensors

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenKonID	text	Sensor Konfiguration ID Eindeutige ID für eine Konfiguration von Sensoren	PRIK
SenKonSenID	text	Sensor ID Verknüpfung zu einem Sensor	FKEY: Sensor.SenID
SenKonBez	text	Sensor Konfigurations-Bezeichnung Bezeichnung einer Konfiguration für einen Sensor	/
SenKonPar	text	Sensor Konfigurations-Parameter Name eines Parameters für eine Sensor Konfiguration	/
SenKonWer	text	Sensor Konfigurations-Wert Wert für einen Parameter	/

## 2.1.5 Entität: SensorKommunikation

Diese Entität beschreibt die Kommunikationskonfiguration eines Sensors.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenKomID	text	Sensor Kommunikation ID Eindeutige ID für eine Kommunikations-Konfiguration von Sensoren	PRIK
SenKomSenID	text	Sensor ID Verknüpft eine Kommunikations-Konfiguration mit einem Sensor	FKEY: Sensor.SenID
SenKomPar	text	Sensor Kommunikations-Parameter Name eines Parameters für eine Kommunikations-Konfiguration	/
SenKomWer	text	Sensor Kommunikations-Wert Wert für einen Kommunikations-	/

		Parameter	
--	--	-----------	--

Beispielwerte für SenKomPar und SenKomWer:

SenKomPar: NetzwerkManager  
=> besagt, dass ein NetzwerkManager für diesen Sensor vorliegt

SenKomWer: UUID eines NetzwerkManagers  
=> FKEY(NetworkManager.NetManID)

## 2.1.6 Entität: SensorKomponente

Diese Entität beschreibt eine Komponente eines Sensors.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenKompID	text	Sensor Komponenten ID Eindeutige ID für eine Komponente eines Sensors	PRIK
SenKompSenID	text	Sensor ID Verknüpft eine Komponenten mit einem Sensor	FKEY: Sensor.SenID
SenKompBez	text	Sensor KomponentenBezeichnung Beschreibt eine Komponente eines Sensors als Fließtext	/

## 2.1.7 Entität: SensorProduktKonfiguration

Diese Entität beschreibt die Grundkonfiguration aller Sensoren des zu Grunde liegenden Sensorprodukts

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenProKonID	text	Sensorprodukt Konfiguration ID Eindeutige ID für eine Produkt Konfiguration eines Sensors	PRIK
SenProKonSenProID	text	Sensorprodukt ID	FKEY: SensorProdukt.SenProID
SenProKonBez	text	Produkt Konfigurations-Bezeichnung Beschreibt eine Produkt Konfiguration als Fließtext	/
SenProKonPar	text	Produkt Konfigurations-Parameter Name eines Parameters für eine Produkt Konfiguration	/
SenProKonWer	text	Produkt Konfigurations-Wert Wert für einen Konfigurations-Parameter	/

## 2.1.8 Entität: SensorVerbundMitglieder

Diese Entität beschreibt die Sensoren, die Mitglied in einem Verbund sind. Ein eingetragener Sensor in dieser Entität kann dadurch gleichzeitig in mehreren Sensorverbänden (Entität SensorVerbund) sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenVerMitID	text	Sensor Verbund Mitglieder ID	PRIK
SenVerMitSenID	text	Sensor ID Verknüpfung mit Sensor	FKEY: Sensor.SenID
SenVerMitSenVerID	text	SensorVerbund ID Verknüpfung mit SensorVerbund	FKEY: SensorVerbund.SenVerID

## 2.1.9 Entität: SensorVerbund

Diese Entität beschreibt einen Verbund von Sensoren

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenVerID	text	Sensor Verbund ID Eindeutige ID für einen Sensor Verbund	PRIK
SenVerbBez	text	Beschreibt den Verbund als Fließtext	/

## 2.1.10 Entität: SensorEvent

Diese Entität beschreibt einen Event eines Sensors. Hier werden beispielsweise Schwellenwerte angelegt und festgelegt, ob ein Event oder ein Alarm ausgelöst werden soll. Ein Event kann die Ansteuerung eines Aktors bewirken. Das Eventhandling findet aber erst in der Entitätengruppe Event statt.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenEveID	text	Sensor Event ID Eindeutige ID für einen Sensor Event	PRIK
SenEveQueID	text	Quell ID Verknüpft einen Event mit einem Sensor oder SensorVerbund	FKEY: Sensor.SenID SensorVerbund.SenVerID
SenEveQue	text	Quelle Event-Quelle, d.h. der Entitätsname auf den die Quell ID verweist	/
SenEvePhyNam	text	Sensor Event Physikalischer Name Beschreibt den physikalischen Namen der Komponente eines Sensors deren Wert gegen einen Schwellenwert geprüft wird	/
SenEveVop	text	Sensor Event Vergleichsoperator Vergleichsoperator (>, <, >= oder <=) für den Vergleich mit dem	/

		Schwellenwert	
SenEveWer	text	Sensor Event Wert Schwellenwert der ein Event auslöst	/

## 2.1.11 Entität: SensorServiceMitglied

Diese Entität beschreibt die Sensoren, die Mitglied in einem Service sind. Ein eingetragener Sensor in dieser Entität kann dadurch gleichzeitig in mehreren Sensorservices (Entität SensorService) sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenSerMitID	text	Sensor Service Mitglieder ID	PRIK
SenSerMitSenID	text	Sensor ID Verknüpfung mit Sensor	FKEY: Sensor.SenID
SenSerMitSenSerID	text	Sensor Service ID Verknüpfung mit SensorService	FKEY: SensorService.SenSerID

## 2.1.12 Entität: SensorService

Diese Entität beschreibt Services, die ein Sensor anbietet.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenSerID	text	Sensor Service ID Eindeutige ID für einen Sensor Service	PRIK
SenSerBez	text	Sensor Service Bezeichnung Beschreibung eines Sensor Services	/

## 2.1.13 Entität: SensorServiceFunktionMitglied

Diese Entität beschreibt Mitglieder, d.h. Funktionen werden bestimmten Services zugeordnet.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenSerFunMitID	text	Sensor Service Funktion Mitglieder ID	PRIK
SenSerFunMitSenSerFunID	text	Sensor Service Funktion ID Verknüpfung mit Sensor	FKEY: SensorServiceFunktion.SenSerFunID
SenSerFunMitSenSerID	text	Sensor Service ID Verknüpfung mit SensorService	FKEY: SensorService.SenSerID



## 2.1.14 Entität: SensorServiceFunktion

Die Funktionen von Services werden in dieser Entität abgelegt.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SenSerFunID	text	Sensor Service Funktion ID	PRIK
SenSerFunNam	text	Name der Funktion	/

## 2.2 Entitätengruppe: Aktor

### 2.2.1 Entität: Aktor

Diese Entität beschreibt einen reell vorhandenen Aktor der SensorCloud

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktID	text	Aktor ID Eindeutige ID für einen Aktor	PRIK
AktAktProID	text	Produkt ID Beschreibt welchem Aktorprodukt dieser Aktor angehört	FKEY: AktorProdukt.AktProID
AktAktTypID	text	Aktor Typ ID Charakterisiert den Typ eines Aktors	FKEY: AktorTyp.AktTypID
AktRauID	text	Raum ID Beschreibt in welchem Raum sich ein Aktor befindet (der Raum befindet sich in einem Flur, dieser auf einer Etage, diese wiederum in einem Gebäude, ...)	FKEY: Raum.RauID
AktNutStaID	text	Nutzerstammdaten ID Verknüpft den Aktor mit einem Nutzer(-stammdaten) der SensorCloud	FKEY: NutzerStammdaten.NutStaID
AktLocMasID	text	LocationMaster ID Verknüpft einen Aktor mit dem LM, an dem er angemeldet ist	FKEY: LocationMaster.LocMasID
AktSouID	text	Aktor Source ID Herstelleradresse des Aktors	/
AktBez	text	Aktor Bezeichnung Bezeichnung des Herstellers für den Aktor/ Friendly Name für einen Aktor	/
AktPos	text	Aktor Position Beschreibt die Position des Aktors	/

		im Raum	
AktDatEin	timestmp	Aktor Datum Eintritt Anmelde Datum des Aktors am LM	/

## 2.2.2 Entität: AktorProdukt

Diese Entität beschreibt ein Aktorprodukt. Ein Aktorprodukt kann beispielsweise ein Schaltaktor einer Firma X sein. Eine Firma Y produziert einen andersartigen Schaltaktor und hat damit ihr eigenes Aktorprodukt. Beide Firmen stellen Produkte vom gleichen Aktortyp her entwickeln aber unterschiedliche Produkte.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktProID	text	Aktor Produkt ID Eindeutige ID für ein Aktorprodukt	PRIK
AktProAktTypeID	text	Aktor Typ ID Verknüpfung zu einem Aktortypen, z.B. Typ Schalter oder Typ Fahrzeug, etc	FKEY: AktorTyp.AktTypeID
AktProAdrID	text	Adress ID Verknüpfung zu einer Adresse	FKEY Adresse.AdrID
AktProHerNum	text	Aktor Produkt Herstellernummer Produktnummer des Herstellers	/
AktProHer	text	Aktor Produkt Hersteller Name des Herstellers des Aktor Produkts (Bsp: HomeMatic, eq3)	/
AktProSem	text	Aktor Produkt Semantik Beschreibung eines Aktors insbesondere dessen Funktionen (s. DBAP3 Semantische Analyse). Die Beschreibung ist im JSON-Format hinterlegt.	/

## 2.2.3 Entität: AktorTyp

Jedem Aktor (Produkt) kann ein bestimmter Typ zugeordnet werden, der diesen grundsätzlich beschreibt. Beispieltypen können sein: Schalter, Fahrzeug, Roboter, Dimmer.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktTypeID	text	Aktor Typ ID Eindeutige ID für einen Aktortyp	PRIK
AktTypDef	text	Aktor Typ Definition Beschreibt Typen eines Aktors als Fließtext	/
AktTypBez	text	Aktor Typ Beschreibung	/

		Kurze Beschreibung eines Aktortyps. BSP: "LRobG": Lego Roboter mit Greifarm "SbSteck": Schaltbare Steckdose	
--	--	--	--

## 2.2.4 Entität: AktorKonfiguration

Diese Entität beschreibt die Konfiguration eines Aktors.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktKonID	text	Aktor Konfiguration ID Eindeutige ID für eine Konfiguration von Aktoren	PRIK
AktKonSenID	text	Aktor ID Verknüpfung zu einem Aktor	FKEY: Aktor.AktID
AktKonBez	text	Aktor Konfigurations-Bezeichnung Bezeichnung einer Konfiguration für einen Aktor	/
AktKonPar	text	Aktor Konfigurations-Parameter Name eines Parameters für eine Aktor Konfiguration	/
AktKonWer	text	Aktor Konfigurations-Wert Wert für einen Parameter	/

## 2.2.5 Entität: AktorKommunikation

Diese Entität beschreibt die Kommunikationskonfiguration eines Aktors.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktKomID	text	Aktor Kommunikations ID Eindeutige ID für eine Kommunikations-Konfiguration von Aktoren	PRIK
AktKomAktID	text	Aktor ID Verknüpft eine Kommunikations-Konfiguration mit einem Aktor	FKEY: Aktor.AID
AktKomPar	text	Aktor Kommunikations-Parameter Name eines Parameters für eine Kommunikations-Konfiguration	/
AktKomWer	text	Aktor Kommunikations-Wert Wert für einen Kommunikations-Parameter	/

Beispielwerte für AktKomPar und AktKomWer

AktKomPar: NetzwerkManager  
=> besagt, dass ein NetzwerkManager für diesen Aktor vorliegt

AktKomWer: UUID eines NetzwerkManagers  
=> FKEY(NetzwerkManager.NetManID)

## 2.2.6 Entität: AktorKomponente

Diese Entität beschreibt eine Komponente eines Aktors.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktKompID	text	Aktor Komponenten ID Eindeutige ID für eine Komponente eines Aktors	PRIK
AktKompAktID	text	Aktor ID Verknüpft eine Komponenten mit einem Aktor	FKEY: Aktor.AktID
AktKompBez	text	Aktor Komponente Bezeichnung Beschreibt eine Komponente eines Aktors als Fließtext	/

## 2.2.7 Entität: AktorProduktKonfiguration

Diese Entität beschreibt die Grundkonfiguration aller Aktoren des zu Grunde liegenden Aktorprodukts

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktProKonID	text	Aktorprodukt Konfiguration ID Eindeutige ID für eine Produkt Konfiguration eines Aktors	PRIK
AktProKonAktProID	text	Aktorprodukt ID Verknüpft eine Konfiguration mit einem Aktorprodukt	FKEY: AktorProdukt.AktProID
AktProKonBez	text	Produkt Konfigurations-Bezeichnung Beschreibt eine Produkt Konfiguration als Fließtext	/
AktProKonPar	text	Produkt Konfigurations-Parameter Name eines Parameters für eine Produkt Konfiguration	/
AktProKonWer	text	Produkt Konfigurations-Wert Wert für einen Konfigurations-Parameter	/

## 2.2.8 Entität: AktorVerbundMitglieder

Diese Entität beschreibt die Aktoren, die Mitglied in einem Verbund sind. Ein eingetragener Aktor in dieser Entität kann dadurch gleichzeitig in mehreren Aktorverbänden (Entität AktorVerbund) sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktVerMitID	text	Aktor Verbund Mitglieder ID	PRIK
AktVerMitAktID	text	Aktor ID Verknüpfung mit Aktor	FKEY: Aktor.AktID
AktVerMitAktVerID	text	Aktor Verbund ID Verknüpfung mit AktorVerbund	FKEY: AktorVerbund.AktVerID

## 2.2.9 Entität: AktorVerbund

Diese Entität beschreibt einen Verbund von Aktoren.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktVerID	text	Aktor Verbund ID Eindeutige ID für einen Aktor Verbund	PRIK
AktVerBez	text	Aktor Verbund Bezeichnung Bezeichnung für einen Aktorverbund, z.B. „Lampen Wohnzimmer“ oder „alle Jalousien“	/

## 2.2.10 Entität: AktorServiceMitglied

Diese Entität beschreibt die Aktoren, die Mitglied in einem Service sind. Ein eingetragener Aktor in dieser Entität kann dadurch gleichzeitig in mehreren Aktorservices (Entität AktorService) sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktSerMitID	text	Aktor Service Mitglieder ID	PRIK
AktSerMitAktID	text	Aktor ID Verknüpfung mit Sensor	FKEY: Aktor.AktID
AktSerMitAktSerID	text	Aktor Service ID Verknüpfung mit AktorService	FKEY: AktorService.AktSerID

## 2.2.11 Entität: AktorService

Diese Entität beschreibt Services, die ein Aktor anbietet.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktSerID	text	Aktor Service ID Eindeutige ID für einen Aktor Service	PRIK

AktSerBez	text	AktorService Bezeichnung	/
-----------	------	--------------------------	---

## 2.2.12 Entität: AktorServiceFunktionMitglied

Diese Entität beschreibt die Mitglieder, d.h. Funktionen werden bestimmten Services zugeordnet.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktSerFunMitID	text	Aktor Service Funktion Mitglieder ID	PRIK
AktSerFunMitAktSerFunID	text	Aktor Service Funktion ID Verknüpfung mit Sensor	FKEY: AktorServiceFunktion.AktSerFunID
AktSerFunMitAktSerID	text	Aktor Service ID Verknüpfung mit AktorService	FKEY: AktorService.AktSerID

## 2.2.13 Entität: AktorServiceFunktion

Funktionen, die zu Services gehören, werden in dieser Entität beschrieben.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktSerFunID	text	Aktor Service Funktion ID	PRIK
AktSerFunNam	text	Name der Funktion	/

## 2.3 Entitätengruppe: Nutzer

Diese Gruppe beschreibt alle Entitäten, die im Zusammenhang mit einem Nutzer, dessen Sicherheit und seinen Produkten steht.

### 2.3.1 Entität: NutzerStammdaten

In dieser Entität stehen alle Stammdaten eines Nutzers der SensorCloud. Zur Zeit können dies private sowie geschäftliche Personen sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
NutStaID	text	Nutzerstammdaten ID Eindeutige ID für einen Nutzer	PRIK
NutStaAdrID	text	Nutzerstammdaten-Adress ID Verknüpfung des Nutzers mit einer Adresse	FKEY: Adresse.AdrID
NutStaAnr	text	Nutzerstammdaten Anrede Anrede: Herr, Frau	/
NutStaNam	text	Nutzerstammdaten Name	/

		Familienname eines Nutzers	
NutStaVor	text	Nutzerstammdaten Vorname Vorname eines Nutzers	/
NutStaFir	text	Nutzerstammdaten Firma Firmenname eines Nutzers	/
NutStaDatEin	timestmp	Nutzerstammdaten Datum Eintritt Zeitpunkt der Anmeldung des Nutzers an der SensorCloud	/

### 2.3.2 Entität: NutzerTelefon

Diese Entität beinhaltet Telefonnummern eines Nutzers. Ein Nutzer kann mehrere Telefonnummern wie z.B. privat oder geschäftlich, mobil, Festnetz haben.

Name	Datentyp	Beschreibung	Beziehung / Integrität
NutTelID	text	Nutzer Telefon ID Eindeutige ID für Nutzer Telefon	PRIK
NutTelNutStaID	text	Nutzerstammdaten Verknüpfung der Nutzerstammdaten	FKEY: NutzerStammdaten.NutStaID
NutTelNum	text	Telefonnummer	/
NutTelBez	text	Bezeichnung der Telefonnummer wie z.B. privat, geschäftlich usw.	/

### 2.3.3 Entität: NutzerEmail

Diese Entität beinhaltet Emailadressen eines Nutzers. Ein Nutzer kann mehrere Emailadressen wie z.B. privat, geschäftlich usw. haben.

Name	Datentyp	Beschreibung	Beziehung / Integrität
NutEmailID	text	Nutzer Email ID Eindeutige ID für Nutzer Email	PRIK
NutEmaNutStaID	text	Nutzerstammdaten Verknüpfung der Nutzerstammdaten	FKEY: NutzerStammdaten.NutStaID
NutEmaAdr	text	Emailadresse	/
NutEmaBez	text	Bezeichnung der Emailadresse wie z.B. privat, geschäftlich usw.	/

### 2.3.4 Entität: NutzerSicherheit

Diese Entität beinhaltet sicherheitsrelevante Daten eines Nutzers. Dies können private, öffentliche Schlüssel oder Passwörter (bzw. deren Hashes) sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
NutSicID	text	Nutzersicherheit ID	PRIK

		Eindeutige ID für die Entität Sicherheit eines Nutzers	
NutSicNutStaID	text	Nutzerstammdaten ID ID eines Nutzers (-stammdaten)	FKEY: NutzerStammdaten.NutStaID
NutSicPas	text	Nutzersicherheit Passwort Verschlüsseltes Passwort eines Nutzers	/
NutSicPriKey	text	Nutzersicherheit Private Key Private Key eines Nutzers	/
NutSicPubKey	text	Nutzersicherheit Public Key Public Key eines Nutzers	/

### 2.3.5 Entität: Mandanten

Entität ermöglicht Nutzern der SensorCloud eigene Nutzer/Mandanten zu haben.

Name	Datentyp	Beschreibung	Beziehung / Integrität
ManID	text	Mandaten ID Eindeutige ID für einen Mandanten	PRIK
ManNutStaID	text	Mandaten Nutzerstammdaten ID Markiert einen Nutzer als Mandanten	FKEY NutzerStammdaten.NutStaID
ManBez	text	Mandanten Bezeichnung Bezeichnung für einen Mandanten	/

### 2.3.6 Entität: MandantenMitglieder

Beschreibt die Mitglieder eines Mandanten.

Name	Datentyp	Beschreibung	Beziehung / Integrität
ManMitID	text	Mandaten Mitglieder ID Eindeutige ID für ein Mandanten Mitglied	PRIK
ManMitManID	text	Mandanten ID Verknüpfung eines Mandanten zu seinen Mitgliedern	FKEY: Mandanten.ManID
ManMitNutStaID	text	Nutzerstammdaten ID Markiert einen Nutzer als Mandanten	FKEY NutzerStammdaten.NutStaID



### 2.3.7 Entität: ServiceLinien

Beschreibt Services, die ein Nutzer gebucht hat.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SerLinID	text	Service Linien ID Eindeutige ID für eine Service Linie	PRIK
SerLinNutStalID	text	Service Linie Nutzerstammdaten ID Verknüpfung zu einem Nutzer	FKEY: NutzerStammdaten.NutStalID
SerLinGruID	text	Service Linie Gruppen ID Verknüpfung einer Service Linie mit einer Gruppe	FKEY: Gruppe.GruID

### 2.3.8 Entität: ServiceLinienMitglieder

Beschreibt die Mitglieder (Services), die zu einer ServiceLinie gehören.

Name	Datentyp	Beschreibung	Beziehung / Integrität
SerLinMitID	text	Service Linien Mitglieder ID Eindeutige ID für ein Mitglied	PRIK
SerLinMitTyp	text	Service Linien Mitglieder Typ Entitätsname, auf den sich bezogen wird. Zum Beispiel SensorService	/
SerLinMitSerID	text	Service Linien Mitglieder Service ID ID von z.B. einem SensorService	FKEY: z.B. auf SensorService.SenSerID
SerLinMitSerLinID	text	Service Linien Mitglieder Service Linien ID Bezug zu einer ServiceLinie	FKEY: ServiceLinien.SerLinID

### 2.3.9 Entität: Gruppen

Definiert eine Gruppe von Nutzern (Mitglieder der Gruppe werden in GruppenMitglieder verwaltet).

Name	Datentyp	Beschreibung	Beziehung / Integrität
GruID	text	Gruppen ID Eindeutige ID für eine Gruppe	PRIK
GruBez	text	Gruppenbezeichnung Name/Bezeichnung der Gruppe	/

### 2.3.10 Entität: GruppenMitglieder

Beschreibt Mitglieder einer Gruppe. Ein Nutzer kann mehreren Gruppen angehören.

Name	Datentyp	Beschreibung	Beziehung / Integrität
GruMitID	text	Gruppen Mitglieder ID Eindeutige ID für ein Gruppenmitglied	PRIK
GruMitGruID	text	Gruppen ID ID der Gruppe zu der ein Nutzer hinzugefügt werden soll	FKEY: Gruppen.GruID
GruMitNutStalD	text	Nutzerstammdaten ID Nutzer der einer Gruppe hinzugefügt werden soll	FKEY: NutzerStammdaten.NutStalD

### 2.3.11 Entität: GruppenRechte

Beschreibt Rechte/Privilegien, die eine Gruppe hat. Könnten z.B. Lese und Schreibrechte oder ähnliches sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
GruRecID	text	Gruppen Rechte ID Eindeutige ID für ein Gruppenrecht	PRIK
GruRecGruID	text	Gruppen ID ID der Gruppe zu der dieses Recht gehört	FKEY: Gruppen.GruID
GruRecPriBez	text	Gruppen Privileg Bezeichnung Bezeichnung für ein Privileg	/
GruRecPriWer	text	Gruppen Privileg Wert Wert eines Gruppenprivilegs	/

### 2.3.12 Entität: Admin

Definiert eine Admingruppe.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AdmID	text	Admin ID Eindeutige ID für eine Admin Gruppe	PRIK
AdmBez	text	Admin Bezeichnung Bezeichnung für eine Admin Gruppe	/

### 2.3.13 Entität: AdminMitglieder

Fügt Mitglieder zu einer Admingruppe hinzu.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AdmMitID	text	Admin Mitglieder ID Eindeutige ID für ein Adminmitglied	PRIK
AdmMitAdmID	text	Admin ID ID der Admin Gruppe zu der ein Mitglied hinzugefügt werden soll	FKEY: Gruppen.GruID
AdmNutStalID	text	Nutzerstammdaten ID Nutzer der einer Admin Gruppe hinzugefügt werden soll	FKEY: NutzerStammdaten.NutStalID

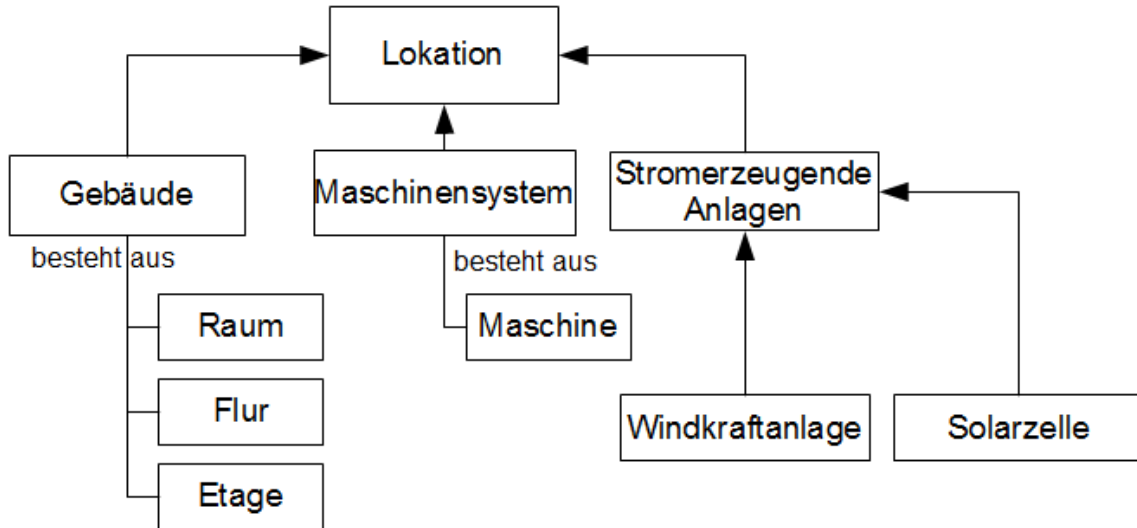
### 2.3.14 Entität: AdminRechte

Beschreibt Rechte/Privilegien, die eine Admin Gruppe hat. Könnten z.B. Lese und Schreibrechte oder ähnliches sein.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AdmRecID	text	Admin Rechte ID Eindeutige ID für ein Adminrecht	PRIK
AdmRecAdmID	text	Gruppen ID ID der Admin Gruppe zu der dieses Recht gehört	FKEY: Admin.AdmID
AdmRecPriBez	text	Admin Privileg Bezeichnung Bezeichnung für ein Privileg	/
AdmRecPriWer	text	Admin Privileg Wert Wert eines Gruppenprivilegs	/

## 2.4 Entitätengruppe: Lokation

Die Gruppe Lokation beschreibt alle Entitäten, die etwas mit einem Ort, einem Gebiet oder einem Gelände zu tun haben. Auch die Verortung an einer Anlage, einem Gerät oder ähnlichem ist in dieser Gruppe möglich. Eine Lokation ist ein Ort oder ein Objekt, das mit Sensoren überwacht und gegebenenfalls durch Aktoren beeinflusst wird. Innerhalb der Gruppe Lokation sind die Entitäten hierarchisch angeordnet.



## 2.4.1 Entität: Lokation

Diese Entität bildet hierarchisch die Wurzel der Gruppe aller Lokationsentitäten. Andere Entitäten dieser Gruppe werden von dieser Gruppe abgeleitet.

Name	Datentyp	Beschreibung	Beziehung / Integrität
LokID	text	Lokation ID Eindeutige ID für eine Lokation	PRIK
LokNutStalID	text	Nutzerstammdaten ID Nutzer, dem diese Lokation zugeordnet ist	FKEY: Nutzerstammdaten.NutStalID
LokAdrID	text	Adress ID Zuordnung einer Adresse zu einer Lokation	FKEY: Adresse.AdrID
LokBez	text	Lokation Bezeichnung Beschreibender Text einer Lokation, z.B. "Haus der Familie XYZ, auf dessen Dach eine Solaranlage mit 16 Solarmodulen betrieben wird."	/

## 2.4.2 Entität: Adresse

Bildet eine real existierende Adresse ab. Alle Adressen der SensorCloud werden in dieser Entität gesichert.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AdrID	text	Adress ID Eindeutige ID für eine Adresse	PRIK
AdrBez	text	Adress Bezeichnung mögliche Bezeichnung für eine	/

		Adresse	
AdrStr	text	Adress Strasse Strasse der Adresse	/
AdrPlz	int	Adress Postleitzahl Postleitzahl der Adresse	/
AdrOrt	text	Adress Ort Ort der Adresse	/
AdrLan	text	Adress Land Land der Adresse	/

### 2.4.3 Entität: Gebaeude

Beschreibt ein Gebäude in einer Lokation.

Name	Datentyp	Beschreibung	Beziehung / Integrität
GebID	text	Gebäude ID Eindeutige ID für ein Gebäude	PRIK
GebLokID	text	Gebäude Lokations ID Lokations ID für ein Gebäude	FKEY: Lokation.LokID
GebEta	int	Gebäude Etagen Anzahl der Etagen eines Gebäudes	/
GebBez	text	Gebäude Bezeichnung Beschreibt einen Gebäude Typ	/

### 2.4.4 Entität: Etage

Beschreibt eine Etage eines Gebäudes.

Name	Datentyp	Beschreibung	Beziehung / Integrität
EtaID	text	Etagen ID Eindeutige ID für eine Etage	PRIK
EtaGebID	text	Gebäude ID Zuordnung einer Etage zu einem Gebäude	FKEY: Gebaeude.GebID
EtaBez	text	Etagen Bezeichnung Beschreibt eine Etage, z.B. 10G, Dach, Keller, Tiefgarage, ...	/

### 2.4.5 Entität: Flur

Beschreibt einen Flur, der auf einer Etage liegt.

Name	Datentyp	Beschreibung	Beziehung / Integrität
FluID	text	Flur ID Eindeutige ID für einen Flur	PRIK
FluEtaID	text	Etagen ID Zuordnung eines Flurs zu einer Etage	FKEY: Etage.EtaID
FlurBez	text	Flur Bezeichnung Beschreibt einen Flur	/

## 2.4.6 Entität: Raum

Beschreibt einen Raum, der in einem Flur liegt.

Name	Datentyp	Beschreibung	Beziehung / Integrität
RauID	text	Raum ID Eindeutige ID für einen Raum. Sensoren können in einem Raum verortet sein.	PRIK
RauEtaID	text	Etagen ID Zuordnung eines Raums zu einer Etage	FKEY: Etage.EtaID
RauNum	text	Raum Nummer	/
RaumBez	text	Raum Bezeichnung Optionale Raumbezeichnung, wenn für den Raum keine Raum-Nr. existiert oder es ein Sonderraum ist (bspw. Wintergarten)	/

## 2.4.7 Entität: Anlage

Beschreibt eine Anlage, die von der SensorCloud überwacht wird.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AnlID	text	Anlagen ID Eindeutige ID für eine Anlage	PRIK
AnlLokID	text	Lokations ID Lokations ID für eine Anlage	FKEY: Lokation.LokID
AnlBez	text	Anlagen Bezeichnung Beschreibt eine Anlage	/

## 2.4.8 Entität: Gerat

Beschreibt ein Gerät, das von der SensorCloud überwacht wird.

Name	Datentyp	Beschreibung	Beziehung / Integrität
GerID	text	Geräte ID Eindeutige ID für ein Gerät	PRIK
GerLokID	text	Lokations ID Lokations ID für ein Gerät	FKEY: Lokation.LokID
GerGerTypeID	text	Geräte Typ ID Zuweisung eines Geräte Typs zu einem Gerät	FKEY: GeraeteTyp.GerTypeID
GerBez	text	Geräte Bezeichnung Gerätespezifikation z.B. "Photovoltaikanlage mit 46 Solarmodulen"	/

## 2.4.9 Entität: GeraetTyp

Beschreibt Typen von Geräten.

Name	Datentyp	Beschreibung	Beziehung / Integrität
GerTypeID	text	Geräte Typ ID Eindeutige ID für einen Geräten Typ	PRIK
GerTypBez	text	Geräte Typ Bezeichnung Bezeichnung für einen Geräte Typ, z.B. "Photovoltaikanlage"	/

## 2.4.10 Entität: Treppe

Beschreibt eine Treppe.

Name	Datentyp	Beschreibung	Beziehung / Integrität
TreID	text	Treppen- ID Eindeutige ID für einen Geräten Typ	PRIK
TreBez	text	Bezeichnung der Treppe	/
TreGebID	text	Zuweisung einer Treppe zu einem Gebaeude	FKEY: Gebaeude.GebID

## 2.4.11 Entität: Aufzug

Beschreibt einen Aufzug.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AufID	text	Aufzugs-ID	PRIK
AufBez	text	Bezeichnung des Aufzugs	/
AufGebID	text	Zuweisung eines Aufzugs zu einem	FKEY:

		Gebaeude	Gebaeude.GebID
--	--	----------	----------------

## 2.4.12 Entität: NavigationsKoordinaten

Beschreibt eine Lokation mit Hilfe von Navigationskoordinaten.

Name	Datentyp	Beschreibung	Beziehung / Integrität
NavKooID	text	NavigationsKoordinaten-ID	PRIK
NavKooTup	text	Tupel, der Koordinaten enthält	/
NavKooBez	text	Bezeichnung von einem Koordiantentupel	/
NavKooLokID	text	Zuweisung von Navigationskoordinaten zu einer Lokation	FKEY: Lokation.LokID

## 2.5 Entitätengruppe: Messwerte

Die Gruppe Messwerte beschreibt alle Entitäten, die als Senke für diverse Messwerte eines Sensors gesehen werden können. Innerhalb der Gruppe Messwerte werden auch die Aktoranforderungen und die daraus folgenden Entitäten beschrieben.

### 2.5.1 Entität: Messwert

Die Entität Messwert sichert Sensor Messwerte. Messwerte können dabei auch als Messwert Tupel auftreten. In einer TrustedCloud sollten diese Messwerte nur verschlüsselt abgelegt werden. Der Schlüssel zur Entschlüsselung des Messwerts liegt in der Entität Nutzersicherheit.

Name	Datentyp	Beschreibung	Beziehung / Integrität
MesWerID	text	Messwert ID Eindeutige ID für einen Messwert	PRIK
MesWerSenID	text	Sensor ID Sensor ID, des Sensors von dem der Messwert stammt	FKEY: Sensor.SenID
MesWerTimSta	timestmp	Messwert Timestamp Zeitstempel des Zeitpunkts der Messung	/
MesWerTup	text	Messwert Tupel Einzelner Messwert oder auch Messwert Tupel	/

### 2.5.2 Entität: MesswertAktuell

Die Entität MesswertAktuell sichert die zuletzt gelieferten Messwerte jedes Sensors

Name	Datentyp	Beschreibung	Beziehung / Integrität
MesWerID	text	Messwert ID	PRIK



		Eindeutige ID für einen Messwert	
MesWerSenID	text	Sensor ID Sensor ID, des Sensors von dem der Messwert stammt	FKEY: Sensor.SenID
MesWerTimSta	timestmp	Messwert Timestamp Zeitstempel des Zeitpunkts der Messung	/
MesWerWer	text	Messwert Einzelner Messwert oder auch Messwert Tupel	/
MesWerNam	text	Messwert Name Name des Messwertes	/

### 2.5.3 Entität: MessTypMitglieder

Beschreibt die Mitglieder, die zu einem MessTyp gehören.

Name	Datentyp	Beschreibung	Beziehung / Integrität
MessTypMitID	text	Eindeutige ID der Tabelle	PRIK
MessTypMitMessTypID	text	Verweis auf einen MessTyp	FKEY: MessTyp.MessTypID
MessTypMitMesWerID	text	Verweis auf einen Messwert	FKEY: Messwert. MesWerID

### 2.5.4 Entität: MessTyp

Definiert Mess-Typen für Messwerte.

Name	Datentyp	Beschreibung	Beziehung / Integrität
MessTypID	text	Mess-Typ ID Eindeutige ID für einen Mess-Typ	PRIK
MessTypPar	text	Mess-Typ Parameter Parameter Bezeichnung für einen Mess-Typen	/
MessTypWer	text	Mess-Typ Wert Wert für einen Mess-Typ Parameter	/

### 2.5.5 Entität: MessLinie

Definiert Messlinien für Messwerte. Insbesondere dienen Messlinien der Steuerung der Weiterverarbeitung von Messwerten, z.B. ihrer Visualisierung.

Name	Datentyp	Beschreibung	Beziehung / Integrität
MesLinID	text	Messlinien ID	PRIK

		Eindeutige ID für eine Messlinien	
MesLinBez	text	Messlinien Bezeichnung Bezeichnung für eine Messlinie	/
MesLinMesLinTypID	text	Messlinien Typ	FKEY: MessLinieTyp.MesLinTypID
MesLinKeys	text	Enthält alle Keys in Listenform durch Kommata getrennt, die zu dieser Messlinie gehören. Die Keys verweisen auf Messwerte aus der Entität Messwert	
MesLinAggFkt	text	Enthält die Funktionen, mit denen aggregiert werden kann	
MesLinQueID	text	Verweis auf die Quelle, d.h. den Sensor, der Messwerte zu dieser Messlinie liefert	FKEY: Sensor.SenID
MesLinQueTyp	text	Gibt den Typ der Quelle an, d.h. den Entitätsnamen, wie z.B. Sensor oder SensorVerbund	
MesLinTimBeg	timestmp	Timestamp Beginn Zeitstempel des Beginns einer Messlinie	
MesLinTimEnd	timestmp	Timestamp Ende Zeitstempel des Endes einer Messlinie	

## 2.5.6 Entität: MessLinieTyp

Definiert einen Typ einer Messlinie.

Name	Datentyp	Beschreibung	Beziehung / Integrität
MesLinTypID	text	Messlinien Typ ID Eindeutige ID für eine Messlinien Typ ID	PRIK
MesLinTypPar	text	Messlinien Typ Parameter Parameter für einen Messlinien Typ	/
MesLinTypWer	text	Messlinien Typ Wert Wert für einen Messlinien Typ Parameter	/

## 2.5.7 Entität: AktorAnforderung

Beschreibt Anforderungen an einen Aktor. Aktoranforderungen sind als Gegenstück zu einem Messwert zu sehen. Wenn Messwerte der Output eines Sensors sind, dann sind Aktoranforderungen der Input eines Aktors.

Name	Datentyp	Beschreibung	Beziehung / Integrität
AktAnfID	text	Aktoranforderungs ID Eindeutige ID für eine Aktoranforderung	PRIK
AktAnfAktID	text	Aktor ID ID des zur Anforderung zugehörigen Aktors	FKEY Aktor.AktID
AktAnfTimSta	timestmp	Aktoranforderungs Timestamp Zeitstempel der Aktoranforderung	/
AktAnfBef	text	Aktoranforderungs Befehl Eigentlicher Befehl, der von einem Aktor ausgeführt werden soll. Dieser muss Teil eines kontrollierten Vokabulars sein, das in der Aktorproduktsemantik hinterlegt ist. Kann auch analog zu einem Messwert Tupel ein Befehle-Tupel sein.	/
AktAnfZus	Int	Aktoranforderungs Zustand Zustand der Aktoranforderung 0: Aktoranforderung wurde noch nicht ausgeführt 1: Aktoranforderung wird gerade ausgeführt 2: Aktoranforderung wurde erfolgreich ausgeführt -1: Fehlerzustand, manueller Eingriff/Entscheidung erforderlich	W = {0,1,2,-1}

Anmerkung:

Aktorbefehle (AktAnfBef) können auch in einer separaten Entität abgelegt werden. So ist es z.B. möglich, dass die lokale Datenbank auf einem LocationMaster eine weitere Entität im lokalen Schema besitzt, in der Aktorbefehle hinterlegt sind (Tabelle: AktorSequenz). In diesem Fall enthält in der Entität AktorAnforderung jeder Datensatz (AktorAnforderung) durch die AktAnfID (UUID) einen Bezug (Referenz) zu einer bestimmten Menge von Aktorsequenzen in denen dann jeweils ein Aktorbefehl steht (vgl. LEGO-Roboter).

## 2.6 Entitätengruppe: LocationMaster

Die Gruppe LocationMaster beschreibt alle Entitäten, die im Zusammenhang mit dem LocationMaster stehen.

### 2.6.1 Entität: LocationMaster

Entität für den eigentlichen LocationMaster (Gateway).

Name	Datentyp	Beschreibung	Beziehung / Integrität
LocMasID	text	LocationMaster ID Eindeutige ID des LocationMaster	PRIK
LocMasLokID	text	Lokations ID Verweis auf eine Lokation, da ein LocationMaster an eine Lokation gebunden ist	FKEY Lokation.LokID
LocMasNutStalID	text	Nutzerstammdaten ID Verweis auf einen Nutzer, dem dieser LM zugeordnet ist	FKEY: NutzerStammdaten.NutStalID
LocMasBez	text	LocationMaster Bezeichnung Bezeichnung des LM	/
LocMasBes	text	LocationMaster Beschreibung Beschreibung bzw. Hervorhebung von Spezifika eines LM (z.B. mit Dongle-Keys)	/

## 2.6.2 Entität: LocationMasterTreiber

Der LocationMaster besitzt unterschiedliche Treiber wie WLAN, ZigBee, BidCoS usw.

Name	Datentyp	Beschreibung	Beziehung / Integrität
LocMasTreID	text	LocationMaster Treiber ID Eindeutige ID des LocationMasterTreibers	PRIK
LocMasTreLokMasID	text	LocationMaster ID Verweis auf einen LM	FKEY: LocationMaster.LocMasID
LocMasTrePfa	text	LocationMaster Treiber Pfad Pfad zu einem Treiber	/

## 2.6.3 Entität: LocationMasterTopologie

Beschreibt die Topologie eines LocationMasters.

Name	Datentyp	Beschreibung	Beziehung / Integrität
LocMasTopID	text	LocationMaster Topologie ID Eindeutige ID der LocationMasterTopologie	PRIK
LocMasTopBez	text	LocationMaster Topologie Bezeichnung Bezeichnung für eine Topologie von LocationMaster, z.B. Master-Slave Strukturen der Ringstrukturen	FKEY: LocationMaster.LocMasID

## 2.6.4 Entität: LocationMasterTopologieKomponente

Beschreibt die Komponenten einer LocationMasterTopologie.

Name	Datentyp	Beschreibung	Beziehung / Integrität
LocMasTopKomID	text	LocationMaster Topologie Komponenten ID  Eindeutige ID der LocationMasterTopologieKomponente	PRIK
LocMasTopKomLocMasTopID	text	LocationMaster Topologie ID  Verweis auf eine LMTopologie	FKEY: LocationMasterTopologie.LocMasTopID
LocMasTopKomBez	text	LocationMaster Topologie Komponenten Bezeichnung  Bezeichnung für eine Komponente der Topologie	/

## 2.6.5 Entität: LocationMasterKonfiguration

Beschreibt die Konfiguration eines LocationMasters.

Name	Datentyp	Beschreibung	Beziehung / Integrität
LocMasKonID	text	LocationMaster Konfiguration ID  Eindeutige ID der LocationMasterKonfiguration	PRIK
LocMasKonLocMasID	text	Location Master ID  Verweis auf einen LM	FKEY: LocationMaster.LocMasID
LocMasKonPar	text	LocationMaster Konfiguration Parameter  Parameter, z.B. WLAN	/
LocMasKonWer	text	LocationMaster Konfiguration Wert  Wert für den zuvor angegeben Parameter, z.B. ein WLAN- Schlüssel: 1XX2GG3456789KLW0	/

## 2.6.6 Entität: LocationMasterKomponente

Beschreibt eine Komponente, die zu einem LocationMaster gehört.

Name	Datentyp	Beschreibung	Beziehung / Integrität
------	----------	--------------	------------------------

LocMasKomID	text	LocationMaster Komponenten ID Eindeutige ID der LocationMasterKomponente	PRIK
LocMasKomLocMasID	text	LocationMaster ID Verweis auf einen LM	FKEY: LocationMaster.LocMasID
LocMasKomBez	text	LocationMaster Komponenten Bezeichnung Bezeichnung der Komponente	/

## 2.6.7 Entität: LocationMasterKomponentenKonfiguration

Beschreibt die Konfiguration einer LocationMaster-Komponente.

Name	Datentyp	Beschreibung	Beziehung / Integrität
LocMasKomKonID	text	LocationMaster Komponenten Konfiguration ID Eindeutige ID der LocationMasterKomponentenKonfiguration	PRIK
LocMasKomKonLocMasKomID	text	LocationMaster Komponenten ID Verweis auf eine LMKomponente	FKEY: LocationMasterKomponente.LocMasKomID
LocMasKomKonPar	text	LocationMaster Komponenten Konfiguration Parameter Parameter einer Komponente	/
LocMasKomKonWer	text	LocationMaster Komponenten Konfiguration Wert Wert für zuvor angegeben Parameter einer Komponente	/

## 2.6.8 Entität: NetzwerkManager

Ein Netzwerkmanager beinhaltet Informationen zur Steuerung von Sensoren und Aktoren einer Technologiefamilie (z.B. Familie Homematic, Familie KNX, Familie MRS) Sensoren und Aktoren einer Produktfamilie werden vom LocationMaster aus über einen Kanal (Adresstyp) angesprochen (z.B. der im Projekt SensorCloud entwickelte MRS immer über „/dev/ttyUSB0“).

Name	Datentyp	Beschreibung	Beziehung / Integrität
NetManID	text	Netzwerk Manager ID Eindeutige ID eines NetzwerkManagers	PRIK
NetManAdr	text	Netzwerk Manager Adresse Adresse, über die ein Sensor/Aktor angesprochen werden kann	/

NetManAdrTyp	text	Netzwerk Manager Typ Typ, den die Verbindung zum Sensor/Aktor besitzt	/
NetManTec	text	Netzwerk Manager Technologie Technologie/Hersteller des Sensors/Aktors	/

Beispielwerte für einen Netzwerkmanager:

NetManAdr: 127.0.0.1 (für IP)  
/dev/ttyUSB0 (für USB)

NetManAdrTyp: IP  
USB

NetManTec: VisionSensor  
HomeMatic

## 2.7 Entitätengruppe: Event

Die Gruppe Event beschreibt alle Entitäten, die im Zusammenhang mit Events stehen.

### 2.7.1 Entität: Event

Beschreibt ein Event mit Priorität und der Nachricht, die bei Eintritt eines Events verschickt/angezeigt wird.

Name	Datentyp	Beschreibung	Beziehung / Integrität
EveID	text	Event ID Eindeutige ID des Events	PRIK
EveArt	int	Event Art Art/Priorität eines Events; z.B. EveArt > 1000: Alarm	/
EveBez	text	Event Bezeichnung Bezeichnung für ein Event	/

Hinweis: Die Entität EventBenachrichtigung wurde mit Version 2.0 gelöscht und wird durch das Attribut EveAktZieWer der Entität EventAktion abgedeckt.

### 2.7.2 Entität: EventAktion

Beschreibt eine Aktion, die aus einem Event resultiert (z.B. Ansteuerung eines Aktors).

Name	Datentyp	Beschreibung	Beziehung / Integrität
EveAktiID	text	Event Aktion ID Eindeutige ID der EventAktion	PRIK
EveAktiEveID	text	Event ID Verweis auf ein Event	FKEY: Event.EveID

EveAktiBez	text	Event Aktion Bezeichnung Bezeichnung für eine Event Aktion	/
EveAktiZie	text	Ziel Verweis auf Ziel Entitätsname. Kann beispielsweise auf die Entität Aktor oder auch die Entität AktorVerbund verweisen	/
EveAktiZieID	text	Event Aktion Ziel ID Ziel einer Aktion, z.B. eine Aktor ID	FKEY: Aktor.AktID AktorVerbund.AktVerID
EveAktiZiePar	text	Event Aktion Ziel Parameter Funktionsname der den Ziel Wert übergeben wird	/
EveAktiZieWer	text	Event Aktion Ziel Wert JSON Objekt angelehnt an SensorCloud Protokoll zur Aktorsteuerung	/
EveAktiZiePrio	int	Event Aktion Ziel Priorität Priorität, die beim Verarbeiten mehrerer EventAktion berücksichtigt wird	/

### 2.7.3 Entität: EventMitglieder

EventMitglieder enthält eine Menge von Sensor- und AktorEvents zugehörig zu bestimmten Events.

Name	Datentyp	Beschreibung	Beziehung / Integrität
EveMitID	text	Event Eindeutige ID	PRIK
EveMitEveID	text	Event ID Verweis auf ein Event	FKEY: Event.EveID
EveMitSenEveID	text	SensorEvent ID Verweis auf ein SensorEvent	FKEY: SensorEvent.SenEveID
EveMitRei	int	Event Mitglieder Reihenfolge Reihenfolge der Events	/

## 3. Entity Relationship Diagram (ERD)

Die Komplexität und Größe des Schemas führt zu einer Darstellung von Ausschnitten des gesamten

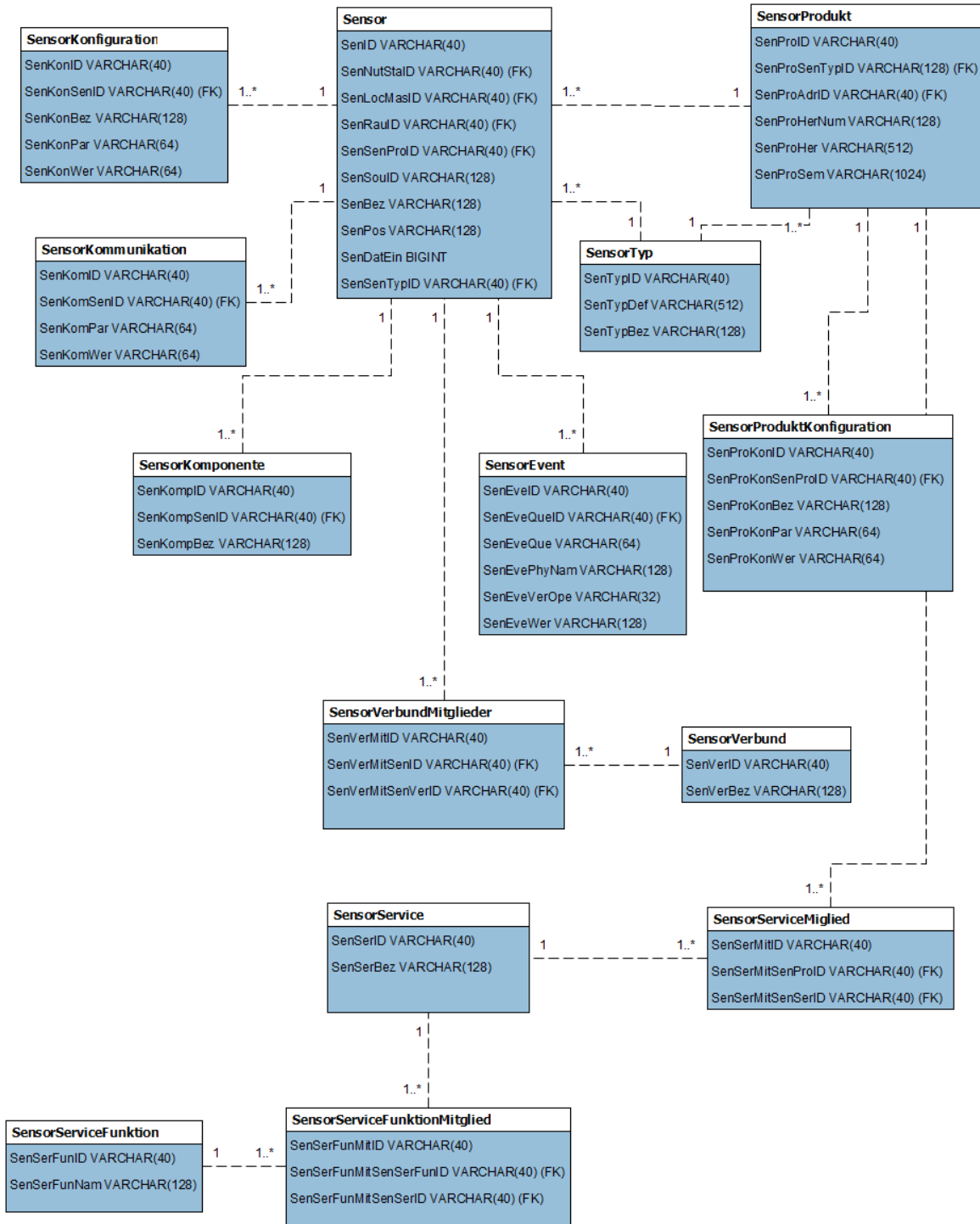


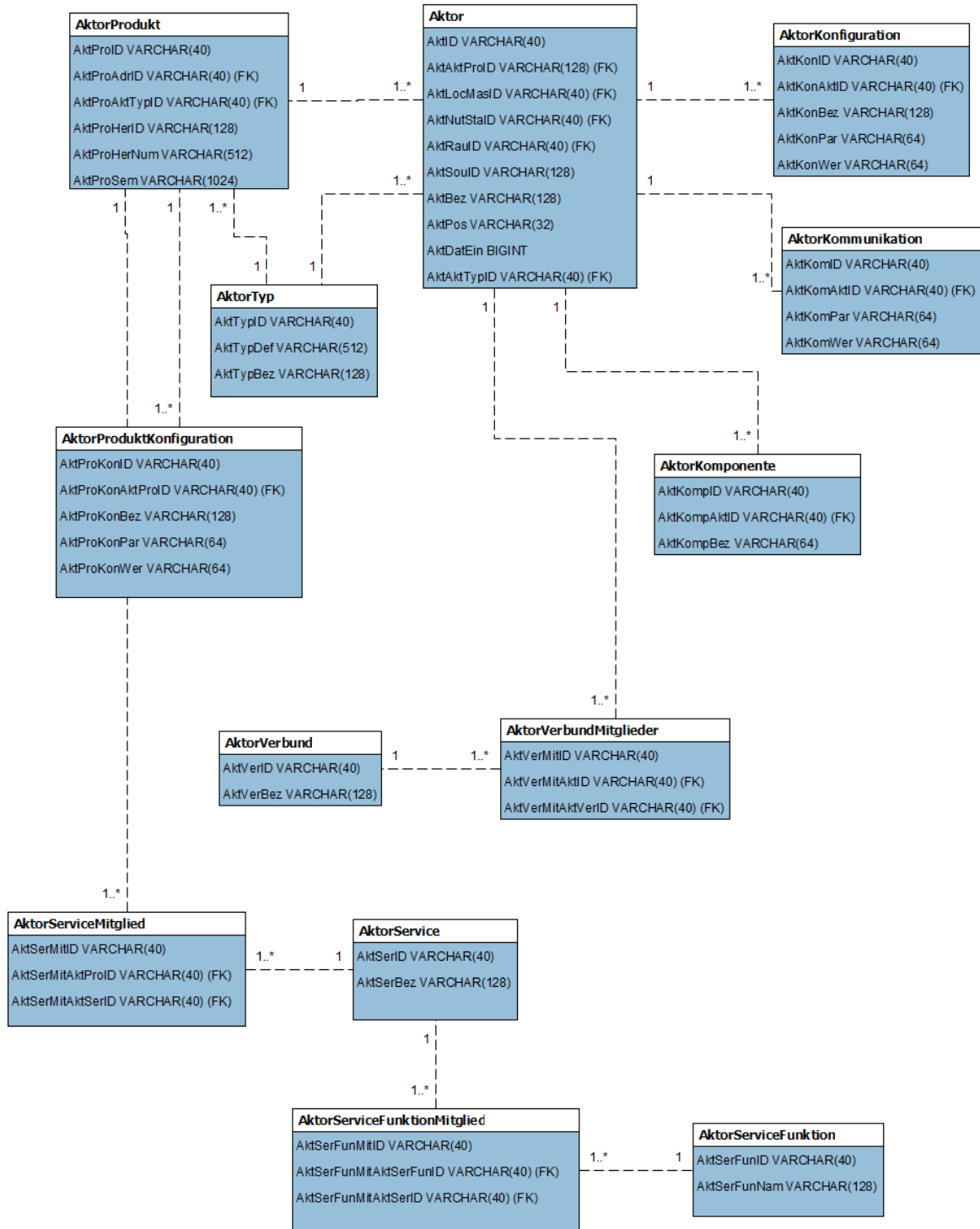
Schemas. Wie schon in der Anforderungsanalyse dargestellt, wurden die Entitäten in Gruppen untergliedert. Diese Gruppen werden auch hier wieder verwendet um eine einfache, übersichtliche und erfassbare Darstellung der Entitäten zu gewährleisten.

Die Querverbindungen der Gruppen werden separat in einem zusätzlichem ERD gezeigt

## **3.1 Gruppe Sensor/Aktor**

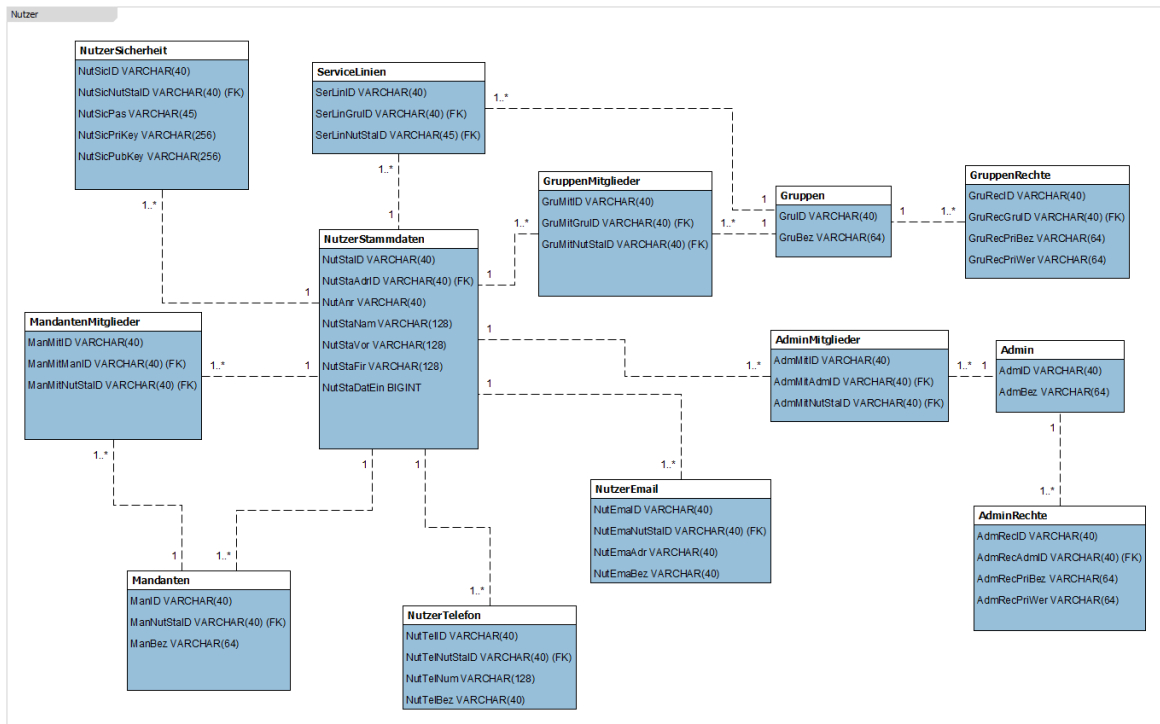
Die Gruppe Sensor/Aktor zeigt die Entitäten, die im Zusammenhang mit Sensor und Aktor stehen. Ein Sensor steht z.B. im Zusammenhang mit SensorKonfiguration, SensorKommunikation, SensorKomponente und Sensorprodukt. Demnach kann ein Sensor z.B. eine bestimmte Konfiguration besitzen und ein SensorEvent auslösen. Eine Menge von Sensoren können auch als Verbund von Sensoren zusammengefasst werden (SensorVerbund, SensorVerbundMitglieder). Jeder Sensor gehört zu einem bestimmten SensorProdukt, ein Produkt bzw. ein Sensor kann man einem SensorTyp zuordnen. Bei Aktoren verhält es sich analog zu Sensoren. Auch für den Aktor gibt es die Entitäten AktorKonfiguration, AktorKommunikation, AktorKomponente und Aktorprodukt.





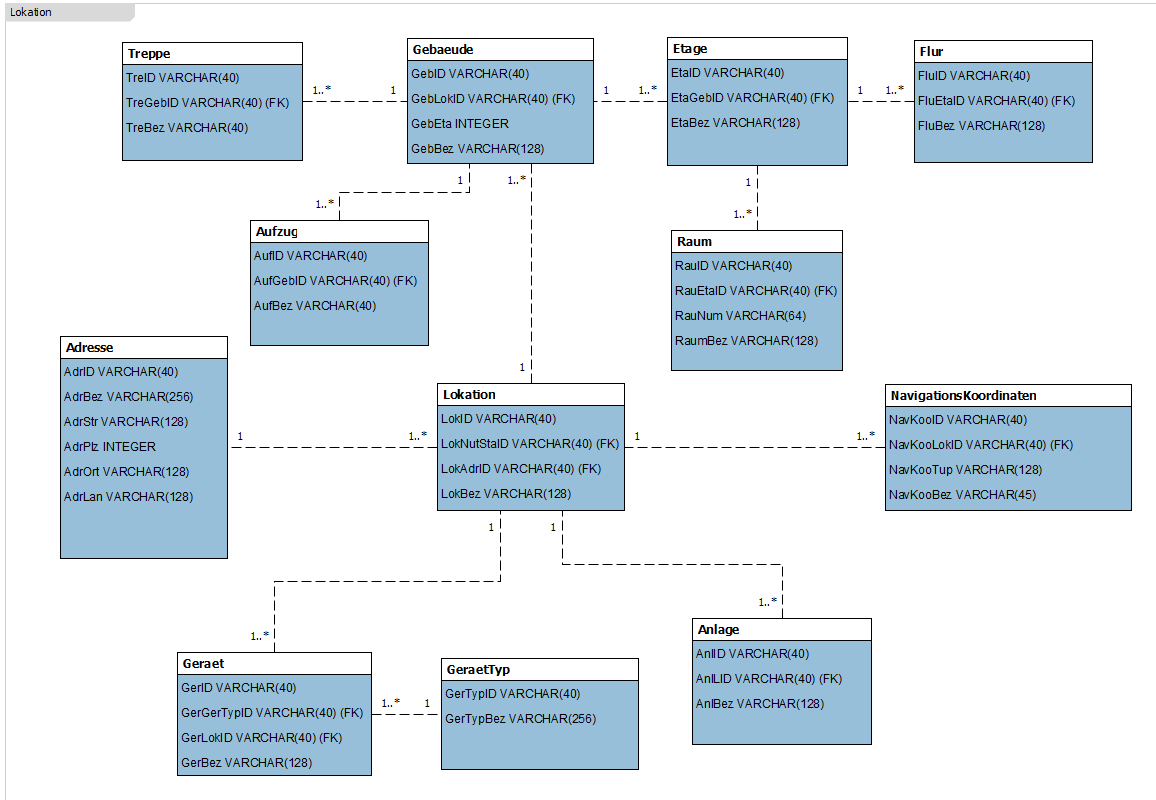
## 3.2 Gruppe Nutzer

Ein Nutzer wird in der Entität NutzerStammdaten bezüglich Name, Firma und Eintrittsdatum beschrieben. Die Adresse, Telefonnummern und E-Mail-Adressen eines Nutzers sind in weiteren Entitäten hinterlegt. Die NutzerSicherheit enthält Schlüssel (Passwörter) eines Nutzers. Ein Nutzer kann einer oder mehreren Gruppen angehören.



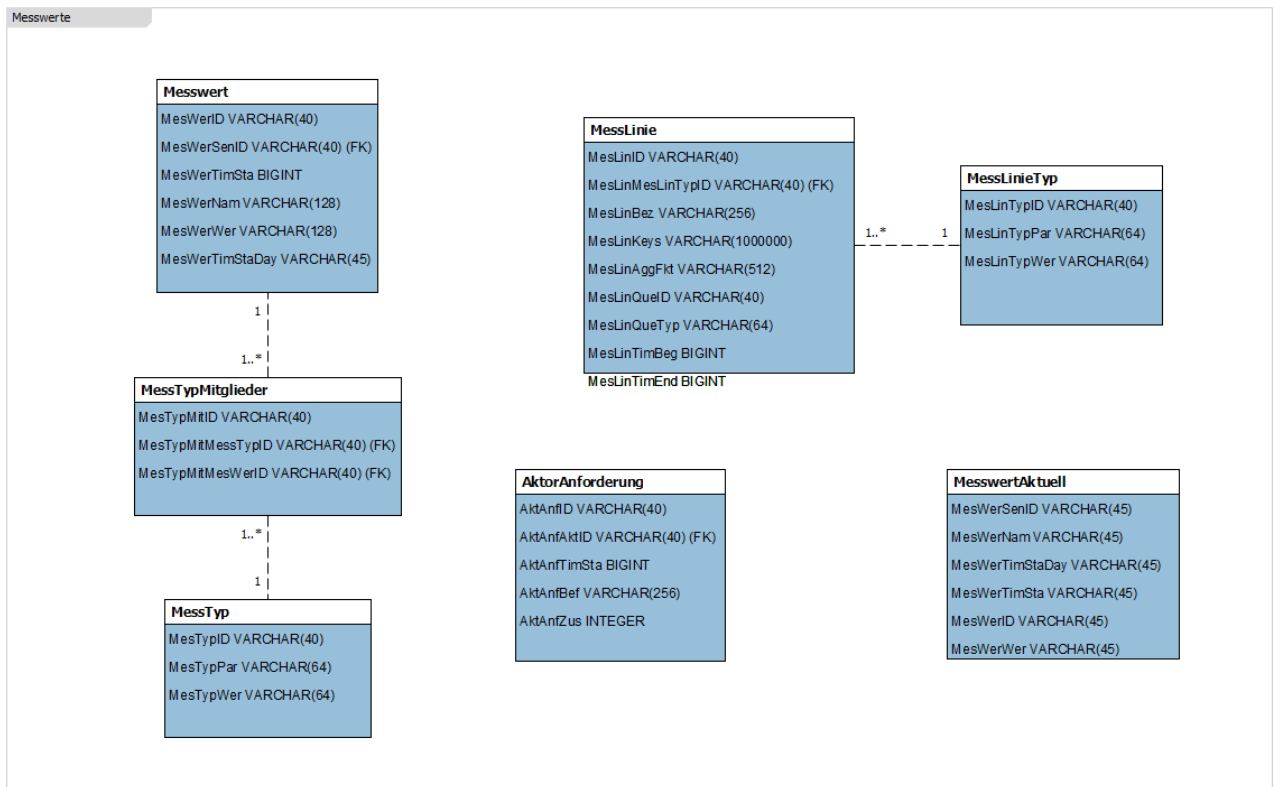
## 3.3 Gruppe Lokation

Eine Lokation hat eine Adresse bzw. NavigationsKoordinaten. Eine Lokation kann z.B. ein Gebaeude, eine Anlage oder ein Geraet sein. Ein Gebaeude kann Etagen, Flure, Räume, Treppen und Aufzüge enthalten.



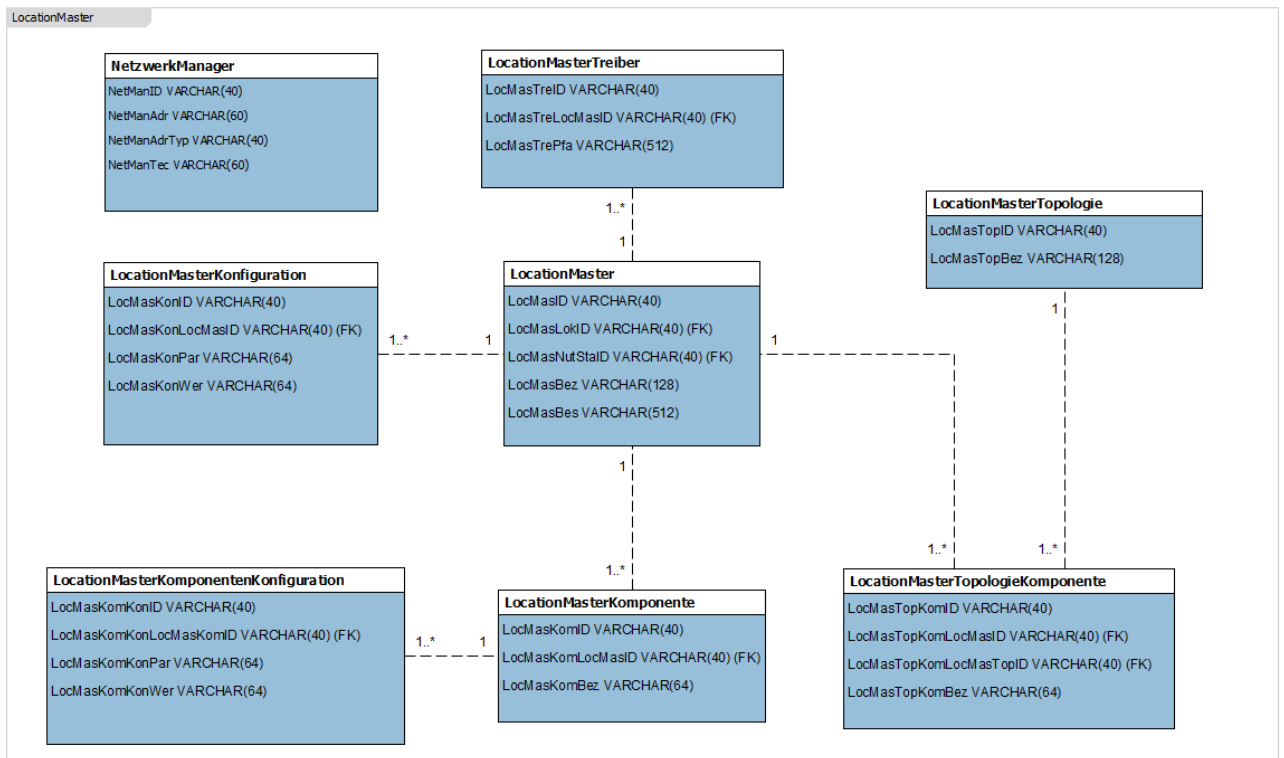
## 3.4 Gruppe Messwerte

Die Gruppe Messwerte beschreibt einen Messwert. Ein Messwert kann Mitglied einer MessLinie sein, eine MessLinie gehört zu einem MessLinieTyp. Anforderungen/Befehle für Aktoren werden durch AktorAnforderung beschrieben.



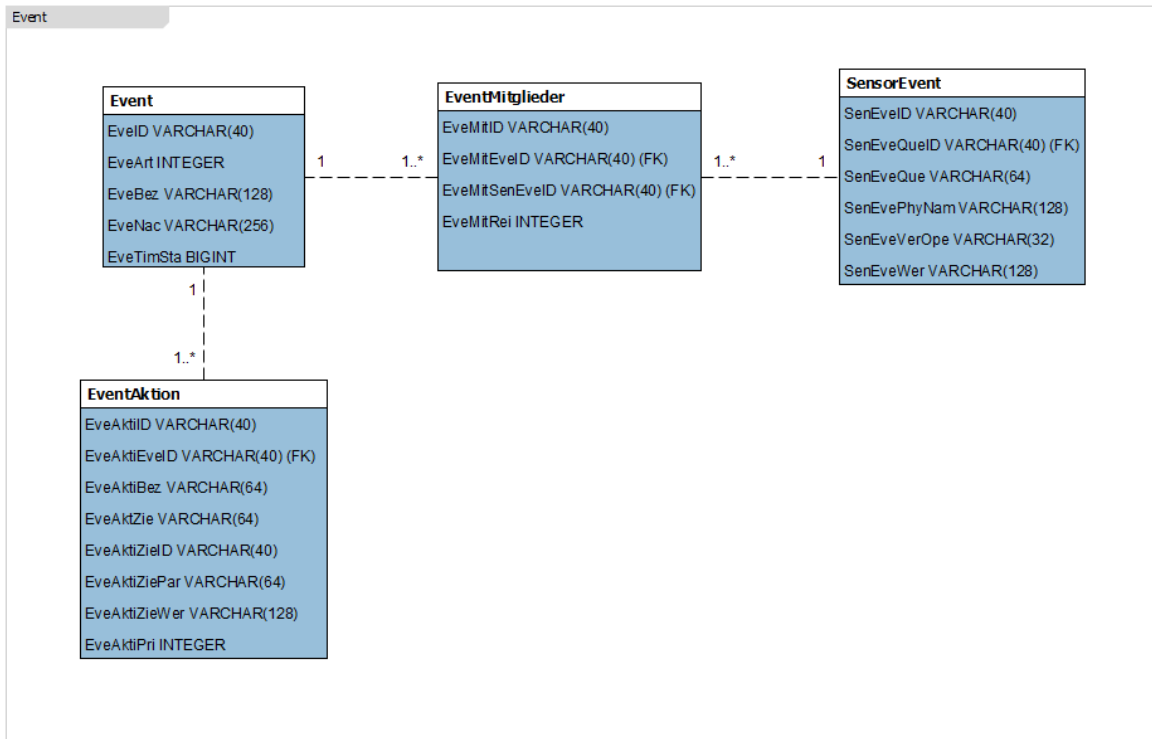
## 3.5 Gruppe LocationMaster

Ein LocationMaster (Gateway) kann ein oder mehrere Konfigurationen besitzen. Konfigurationen von LocationMasterKomponenten werden in der Entität LocationMasterKomponentenKonfiguration beschrieben.



## 3.6 Gruppe Event

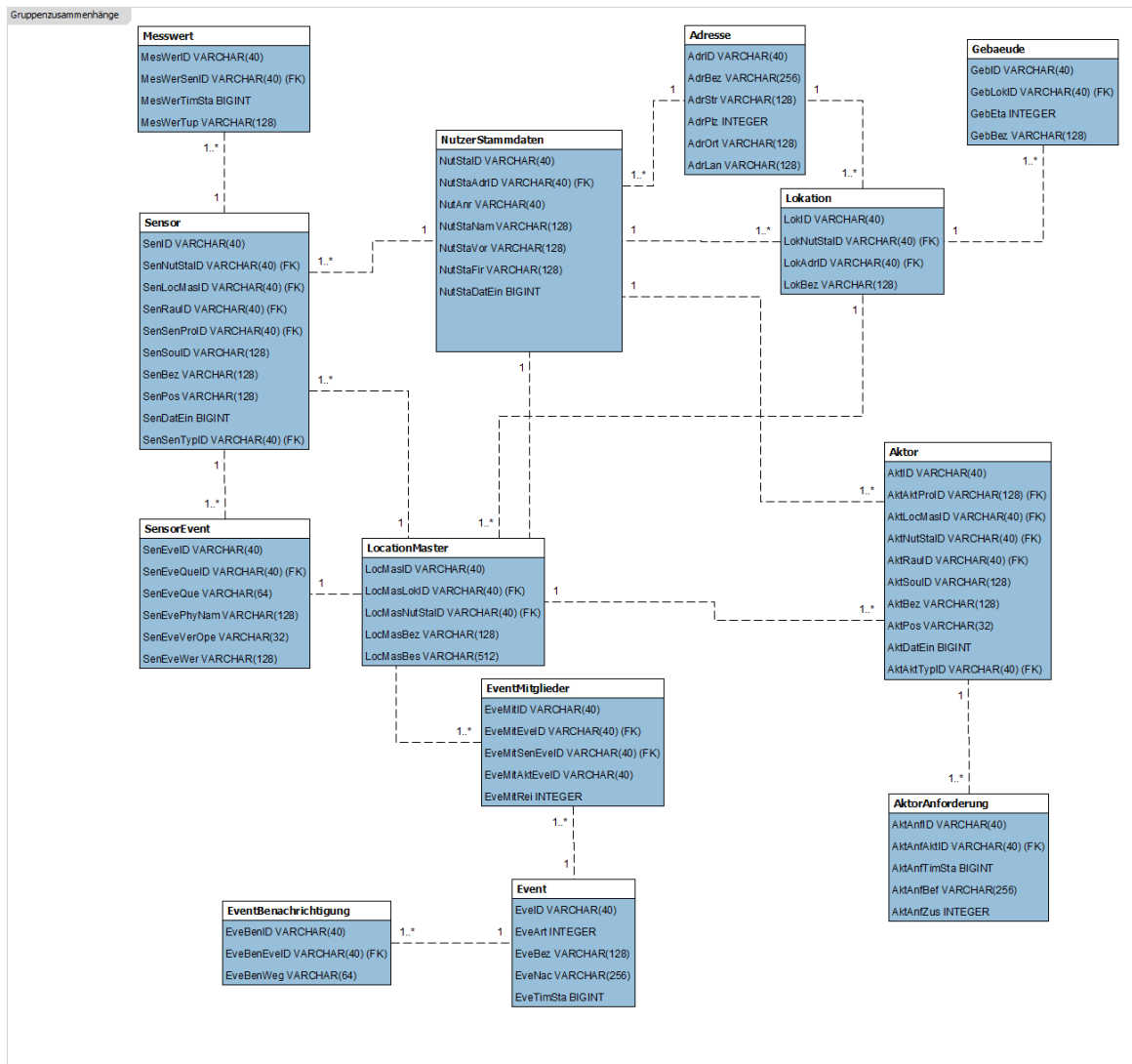
Ein Sensor kann Events durch Schwellenwerte auslösen. Diese sind in der Entität SensorEvent beschrieben (Entität in der Gruppe Sensor/Aktor). Ein Event hat ein oder mehrere Mitglieder (SensorEvents) die ein Event auslösen. In EventBenachrichtigung wird der Weg beschrieben, z.B. E-Mail, über den der Eintritt eines Events mitgeteilt wird. EventAktion beschreibt die aus einem eingetretenen Event ausgelöste Aktion oder Aktionen (z.B. Ansteuerung eines Aktors bzw. eines Aktorverbunds).





## 3.7 Gruppen-Zusammenhänge

Die Gruppen-Zusammenhänge zeigen Zusammenhänge, die über eine Gruppe hinausgehen. Sensor ist z.B. verknüpft mit NutzerStammdaten, LocationMaster, Messwert und SensorEvent. Aktor ist verknüpft mit AktorAnforderung, AktorEvent und auch wie Sensor mit NutzerStammdaten und LocationMaster, Adresse mit Lokation und mit Nutzer, Adresse mit Lokation und mit Nutzer.



## 4. Erweiterung des konzeptionellen Schemas

Im Verlaufe des Projekts SensorCloud hat das konzeptionelle Schema einige evolutionäre Veränderungen gemacht, die dem Ansatz eines föderiertem Datenbanksystems (FDBS) geschuldet sind. Im speziellen musste die unterschiedlichen Arten der Datenmodellierung von relationalen wie auch der NoSQL Datenbank Cassandra Rechnung getragen werden. Am Beispiel der Entität Messwert wird im folgendem der Übergang von einem relationalen zu einem NoSQL Datenmodell erläutert.

Die Entität Messwert (vgl. 2.5.1) hatte in der ersten Entwicklungsstufe vier Attribute:

MesWerID	Eindeutige ID für einen Messwert
MesWerSenID	ID des Sensors, der den Messwert gesendet hat
MesWerTimSta	Zeitstempel des Messwertes
MesWerTup	Messwert (-tupel)

Der Primärschlüssel lag dabei auf der ID des Messwertes. Auf die Messwerte kann mit einer konstanten Laufzeit der Ordnung  $O(1)$  über die Messwert ID als Row Key zugegriffen werden. Abfragen von Messwerten zu gegebenen Zeitintervallen oder zu bestimmten Zeitpunkten führen zu einem sequentiellen Durchsuchen aller Datensätze der entsprechenden Knoten, da innerhalb des Knotens die Messwerte zeitlich ungeordnet vorliegen. Dies resultiert aufgrund der sequentiellen Suche der Messwerte in einer Abfrage mit linearer Laufzeit der Ordnung  $O(n)$ . Messungen ergaben auf einem Zwei-Knoten-Cluster mit 400.000 Messwerten für die Suche von 17.400 Messwerten eines Tages eine Laufzeit von 240 Sekunden.

Analog zu relationalen Datenbanken unterstützt Apache Cassandra die Anlage von Secondary Indizes. Ein Index auf die Spalte MesWerTimSta beschleunigt den direkten Zugriff auf einen Datensatz zu einem gegebenen Zeitstempel. Da der Partitioner anhand des Zeitstempels aber nicht entscheiden kann, welcher Knoten den Datensatz hält, müssen alle Knoten in ihrem lokalen Index nachsehen. Bei einer Intervallsuche bedeutet dies erneut ein sequentielles Durchsuchen aller Datensätze auf allen Knoten, da in Cassandra der Index nicht in B-Bäumen, sondern analog der Daten in Key/Value-Tabellen organisiert wird. So müssen bei einer Intervallsuche die Datensätze im Index vollständig vom ersten bis zum letzten Datensatz durchsucht werden.

Eine erste Evolutionsstufe bestand aus einem Aufspalten des Timestamps in Jahr, Monat, Tag, Stunde, Minute, Sekunde, Millisekunde. Diese Spalten wurden den schon vorhandenen hinzugefügt

MesWerTimYea	Jahr, in dem der Messwert aufgenommen wurde
MesWerTimMon	Monat, in dem der Messwert aufgenommen wurde
MesWerTimDay	Tag, in dem der Messwert aufgenommen wurde
MesWerTimHou	Stunde, in dem der Messwert aufgenommen wurde
MesWerTimMin	Minute, in dem der Messwert aufgenommen wurde
MesWerTimSec	Sekunde, in dem der Messwert aufgenommen wurde

Im Gegensatz zu mehrspaltigen Secondary Indizes in RDBMS kann in Cassandra bei einer Selektion über mehrere Secondary Indizes nicht direkt auf die Ergebnismenge zugegriffen werden. Die Ergebnismenge wird über ein Filtern im Hauptspeicher des Servers bestimmt, wodurch die Selektionszeit aufgrund der linearen

Komplexität mit steigender Messwertanzahl wächst. Ein vergleichbarer Performancegewinn wie bei relationalen Datenbanksystemen ist daher in Cassandra mit der Verwendung von Secondary Indizes nicht möglich. Mit Hilfe von Secondary Indizes auf die zusätzlichen Timestamp spalten konnte ein Performancegewinn um den Faktor 4 und somit eine Abfragedauer von 62s erreicht werden.

Beispiel einer Abfrage des Messwerts des Sensor 8, der am 04.12.2013 um 13:50 Uhr gemessen wurde.

```
SELECT * FROM "Messwert"  
WHERE "MesWerSenID" = '8'  
AND "MesWerTimYea" = 2013  
AND "MesWerTimMon" = 12  
AND "MesWerTimDay" = 4  
AND "MesWerTimHou" = 13  
AND "MesWerTimMin" = 50  
ALLOW FILTERING;
```

Bei dem eingesetzten Wide-Column-Store Cassandra benötigen - im Gegensatz zu relationalen Datenbanksystemen – Zugriffe auf gespeicherte Messwerte bei einer optimierten Datenhaltung eine konstante logarithmische Laufzeit. Eine Speicherung der Messwerte muss hierfür in auf die späteren Abfragen hin optimierten ColumnFamilies erfolgen. Die Implementierung der Entität Messwert in Cassandra als ColumnFamily eines KeySpaces wurde unter der Kenntnis der späteren Abfragen eines Visualisierungstools für Messwerte mit einer chronologischen Selektion der Messwerte vorgenommen.

Besteht in Cassandra der Primary Key einer Tabelle (Column Family) aus mehreren Attributen, so wird dieser in Cassandra *Composite Key* genannt. Der Composite Key unterteilt sich in den *Partition Key* (auch *Row Key* genannt) und den *Clustering Key*. In Cassandra bestimmt der Partition Key, auf welchem Knoten ein Datensatz liegt. Der Clustering Key bestimmt, wie die Daten auf dem Knoten gespeichert werden. Dabei werden die Daten anhand der Attribute des Clustering Keys gruppiert und sortiert in einer durch den Row Key identifizierten Wide Row abgelegt.

Unter Berücksichtigung dieser Punkte kann die Messwert ID (MesWerID) als Primary Key der Entität Messwert des konzeptionellen Schemas nicht als Partition Key genutzt werden. Bei einer für eine Generierung von Messwertkurven angenommenen chronologischen Selektion der Messwerte bewirkt die Messwert ID als Partition Key eine Verteilung aller Messwerte aller Sensoren über das gesamte Cluster. Dies führt bei einer Selektion aufgrund der Netzwerkkommunikation und der auftretenden Netzwerklatenz zu einer höheren Laufzeit gegenüber einem sequentiellen Lesen zusammenhängender Daten aus einer Wide Row von einem Knoten.

Da bei einer Diagrammgenerierung eine chronologische Selektion der Messwerte eines Sensors vorgenommen wird, qualifiziert sich die Sensor ID als Partition Key. Hierdurch werden alle Messwerte eines Sensors auf einem Knoten gespeichert. Durch eine ausbalancierte Verteilung aller an das System angeschlossenen Sensoren auf alle Knoten des Clusters entstehen bei identischen Sendeintervallen keine Hotspots.

Ein Sensor misst 1 bis  $n$  mit  $n \in \mathbb{N}$  physikalische Größen (Temperatur, Luftdruck, Strömungsgeschwindigkeit, Energie, Leistung, ...). Für die Auswertung der Messwerte verschiedener physikalischer Größen werden unterschiedliche statistische Funktionen verwendet. Daher wird der Clustering Key um ein neues Attribut erweitert, welches die gemessene physikalische Größe beschreibt. Dies bewirkt eine zusammenhängende Speicherung der Messwerte eines Sensors zu einer physikalischen Größe. Dazu wird das Messwert-Tupel, welches bisher die unterschiedlichen Messwerte zu einem Zeitpunkt kommaschliessend darstellte, erweitert. Zwei neue Attribute (MesWerNam und MesWerWer) werden eingeführt. Die vorher als Tupel zusammengefassten Messwerte unterschiedlichster Art erhalten dadurch den gleichen Zeitstempel, die gleiche Sensor ID und die gleiche Messwert ID. Dies führt zu einer Duplizierung von Informationen. Das bisherige Attribut MesWerTup wird entfernt.

Für einen beschleunigten Zugriff auf die Messwerte eines Tages werden die Messwerte anhand eines

Tageszeitstempels (MesWerTimStaDay) im Clustering Key weiter gruppiert. Analog eines Primary Keys in einem RDBMS muss der Composite Key in Cassandra eindeutig sein. Daher wird für den Clustering Key ein Attribut benötigt, welches den Composite Key eindeutig identifiziert. Hierzu wird der Zeitstempel des Messzeitpunktes (MesWerTimSta) gewählt. Jeder Sensor kann bzw. sollte zu einem Zeitpunkt eine physikalische Größe nur einmal messen. Dies muss algorithmisch sichergestellt werden.

Große Unterschiede in den zu erwartenden Datenmengen der Sensoren können bei dieser Modellierung zu einem nicht ausbalancierten Cluster führen, was sich in einer unterschiedlichen Speicherplatzbelegung und Auslastung der einzelnen Knoten widerspiegelt. Dies ist in der Speicherung sämtlicher Messwerte eines Sensors in einer Row begründet. Bei einer ungünstigen Balancierung entstehen dadurch Hotspots mit Engpässen der Speicherkapazität an einzelnen Knoten.

Durch Hinzunahme des Messwertnamens (Attribut MesWerNam) in den Partition Key können verschiedene Messwerte verschiedener Messgrößen auf unterschiedliche Rechner abgebildet werden. Dies verbessert die Verteilung der Messwerte, kann aber weiterhin zu einer nicht ausbalancierten Verteilung führen. Eine Gruppierung von Messwerten nach physikalischen Messgrößen in Wide Rows gleicht der Anlage von mehreren Sensoren, die jeweils nur Messwerte zu einer physikalischen Größe liefern. Dies verzögert, aber verhindert nicht, die Entstehung von Hotspots und einer ungleichmäßigen Speicherplatzbelegung.

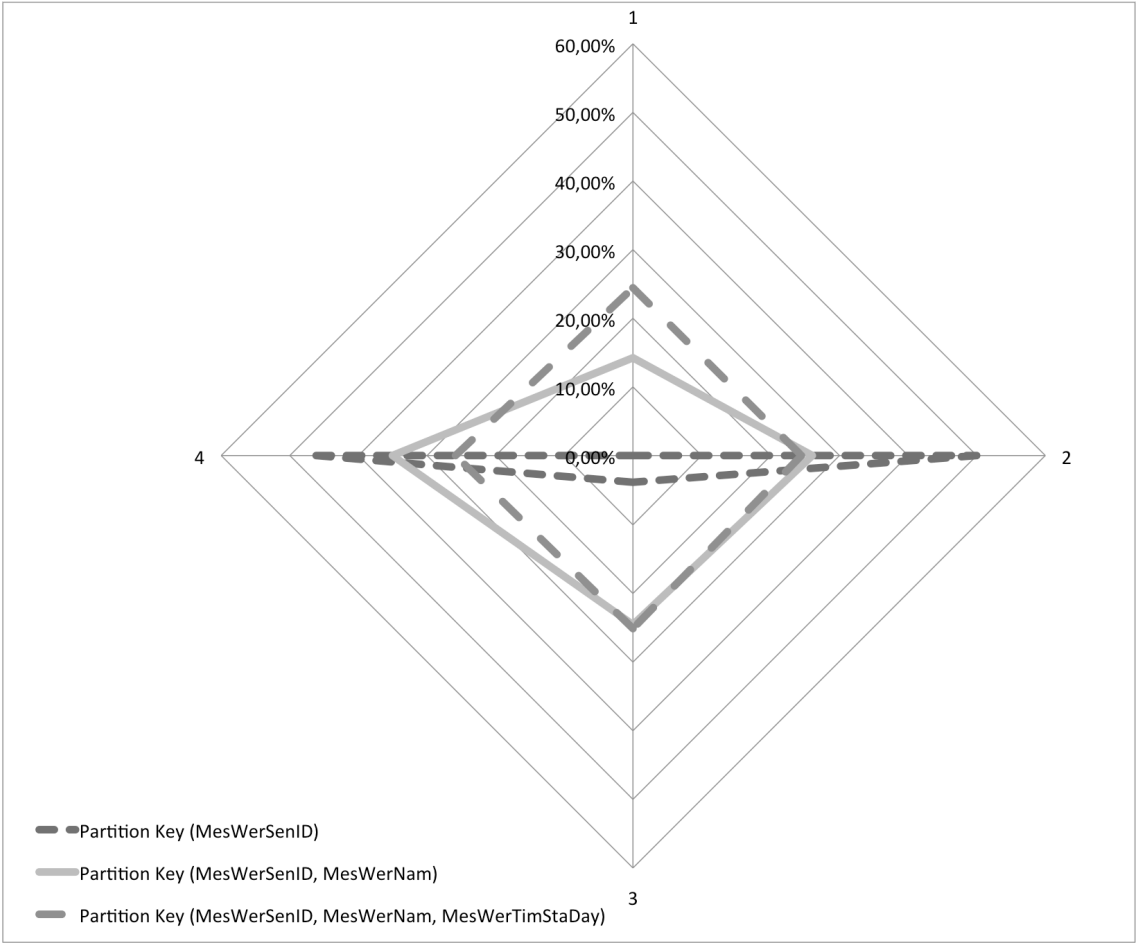
Bei Hinzunahme des Tageszeitstempels in den Partition Key, steigt die Wahrscheinlichkeit einer ausbalancierten Verteilung der Messwerte, da die Anzahl der Elemente des Urbildes für die Abbildungsfunktion des Partitioners weitaus höher ist. Die Elemente ergeben sich aus der resultierenden mehrdimensionalen Tabelle über die Sensoren, deren Messgrößen und die Tage in denen ein Sensor Messwerte zu einer Messgröße lieferte.

$$[MesWerSenID] \times [MesWerNam] \times [MesWerTimStaDay]$$

Folgendes CREATE-Statement ergibt sich für eine Datenmodellierung der Entität Messwert in Cassandra.

```
CREATE TABLE "Messwert" (
  "MesWerSenID" text,
  "MesWerNam" text,
  "MesWerTimStaDay" text,
  "MesWerTimSta" text,
  "MesWerID" text,
  "MesWerWer" text,
  PRIMARY KEY (("MesWerSenID", "MesWerNam", "MesWerTimStaDay"),
  "MesWerTimSta")
) WITH CLUSTERING ORDER BY ("MesWerTimSta" DESC);
```

Die nachfolgende Abbildung zeigt die Messwertverteilung auf einem Vier-Knoten-Cluster bei unterschiedlich ausgeprägten Partition Keys nach dem Einspielen von 14.837.119 Messwerten. Bei dem Cassandra optimierten Schema erfolgt eine ausbalancierte Verteilung der Daten auf allen Knoten mit  $25,00\% \pm 0,60\%$  (24,42%, 24,51%, 25,15% und 25,92%).



## 5. Versionsänderungen

1.0 Version für Meilenstein 3

2.0 Version für Meilenstein 6

- Entität: EventMitglieder
  - Gelöscht: EveMitAktEveID  
Aktoren können keine Events auslösen, nur deren sensorischen Eigenschaften
- Entität: EventAktion
  - Umbenannt: EveAktiZieBez in EveAktiBez
  - Funktionsänderung: EveAktiZiePar und EveAktiZieWer  
Ziel Parameter beschreibt nun Funktionsname und Wert beschreibt in Protokoll ActuatorMessage Form die auszuführende Aktion
- Entität: EventBenachrichtigung
  - Entfernt, da diese in EventAktion durch EveAktZieWer abgebildet wird
- Entität: MesswertAktuell
  - Hinzugefügt
    - Entität gibt aktuellen Messwert wieder. Ist analog zur Messwert Entität aufgebaut, enthält für jeden Sensor immer den letzten empfangenen Messwert. Wird für den Regelinterpreter benötigt.
- Entität: NetzwerkManager
  - Hinzugefügt
    - Entität enthält Informationen zur Steuerung von Sensoren und Aktoren
- Entität: SensorKommunikation
  - Beispiel für einen NetzwerManager hinzugefügt
- Entität: AktorKommunikation
  - Beispiel für einen NetzwerManager hinzugefügt
- Diagramm: Gruppe Messwert
  - Aktualisiert
- Diagramm: Gruppe Event
  - Aktualisiert
- Diagramm: Gruppe LocationMaster
  - Aktualisiert
- Kapitel: „Erweiterung des konzeptionellen Schemas“ hinzugefügt