

Assortment System
Vers.: 0.4

Tobias Falk

14 Nov 2021

Abstract

This Project has the aim to help sort small to middle sized parts, it has an Software part where the parts, there descriptions, and location are stored in a database and a Physical part where the parts are actually stored.

The aim of this document is to give an overview of the design and structure of the System. This document is not an detailed documentation of the separated parts(peace) of the project. An detailed documentation about each Part can be found in each separated git project of each part.

Contact Information

- Tobias Falk; E-Mail: tobiasfalk200242@gmail.com; gitHub: tobiasfalk

Contents

Abstract	I
Contact Information	II
List of Figures	IV
List of Tables	V
1 Software	1
1.1 Design Structure	1
1.2 Server	2
1.2.1 SQL database	2
1.2.2 middleware	11
1.3 Client	12
2 Hardware	13
2.1 First Section	14
2.2 Second Section	15
2.3 Third Section	16
References	18

List of Figures

1.1	The design structure visualised	1
1.2	The design of the parts schema	4
1.3	The design of the storage schema	6
1.4	The design of the global schema	7
1.5	The design of the assemblies schema	8
1.6	The design of the storage schema	9
1.7	The design of the storage schema	10

List of Tables

Chapter 1

Software

GitHub: https://github.com/tobiasfalk/Assortment_System_Server.git

Here we describe the software design and method of saving the information about the different parts.

1.1 Design Structure

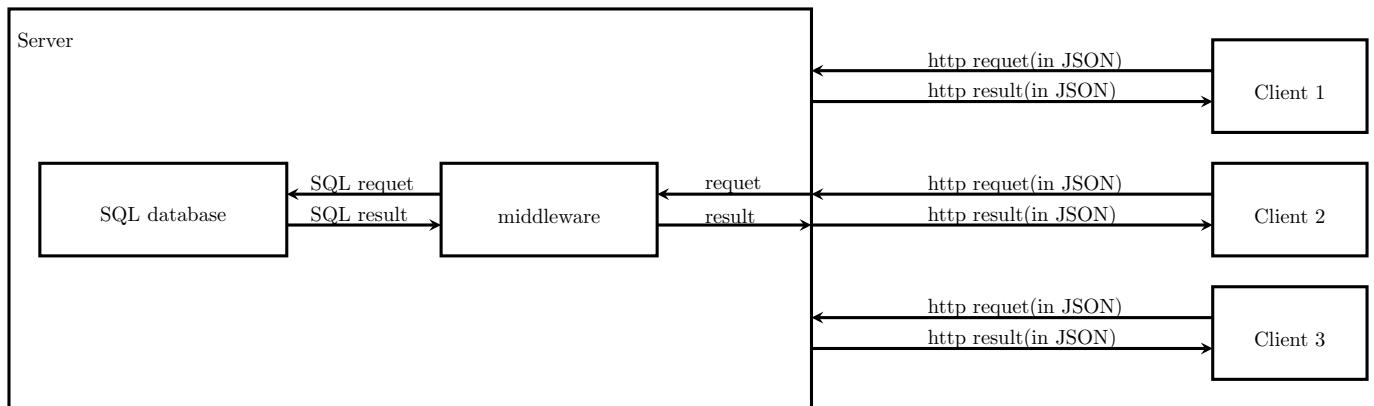


Figure 1.1: The design structure visualised

The basic idea is that we have one server where a SQL database runs and that there is an middleware that provides a REST API interface for the DB. The clients than work with this API, the client can be web based or as an Program direct on the PC or Mobile phone.

1.2 Server

On the Server there are two things, the SQL database, where all the information is stored and the middle ware, which provides an REST API interface for the DB.

1.2.1 SQL database

The SQL database is a PostgreSQL¹ database and is designed in a program named PgModeler². It has the main task of saving the known information about the parts that are managed in the system.

The DB is split into 6 schemas:

- parts
- storage
- global
- assemblies
- vendors
- kicad

¹PostgreSQL webpage: <https://www.postgresql.org/>

²PgModeler webpage: <https://pgmodeler.io/>

Schema "parts"

In this schema you can find all the basic information about the parts. You can not find where the part is physically stored.

In the "middle" of the schema there is the *parts.parts* table where you will find all the attributes of an part that each needs to have. That includes:

- id(not set by the user)
- name
- description
- weight
- type

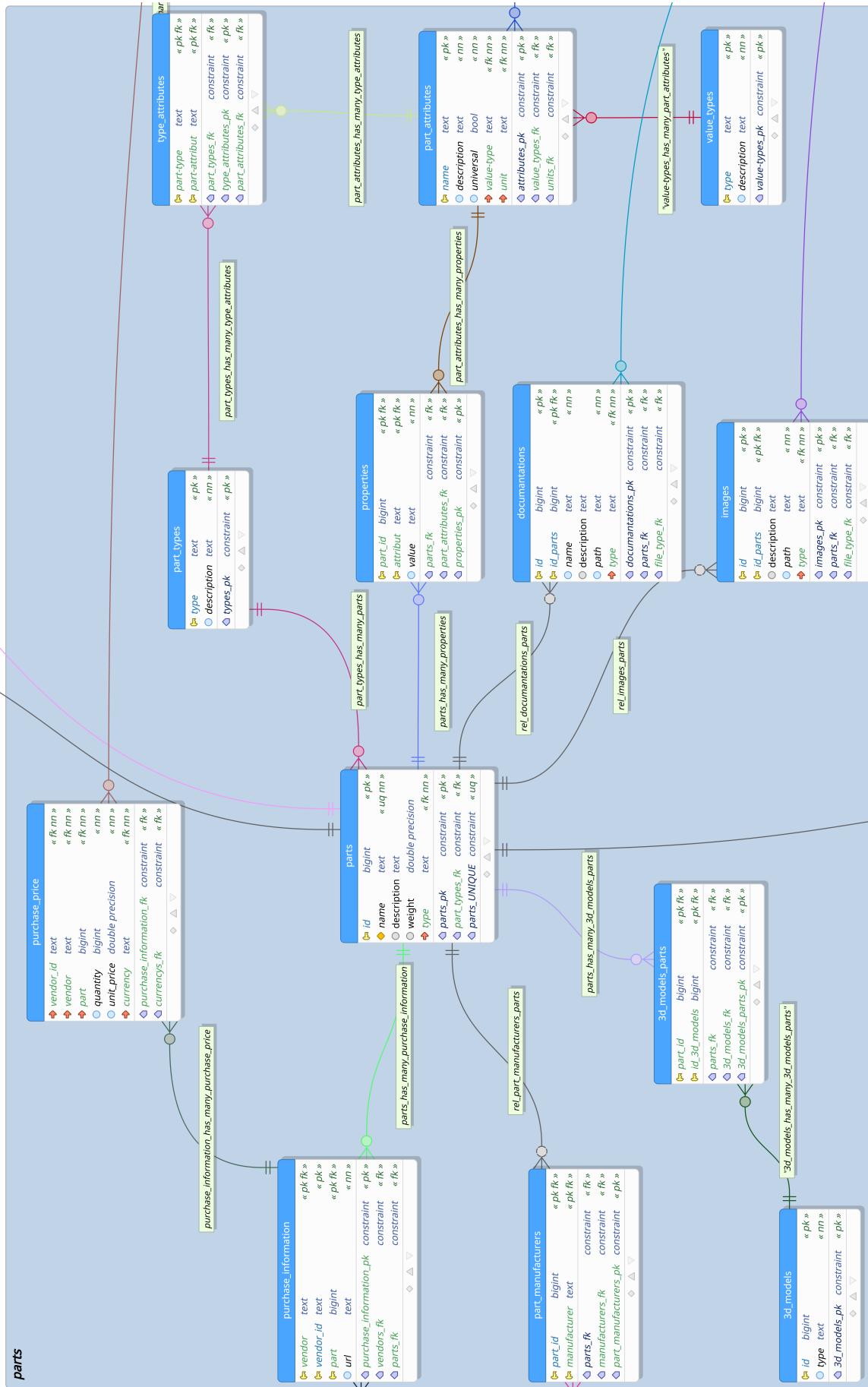


Figure 1.2: The design of the parts schema

Schema "storage"

In this schema you can find in witch box, drawer and cabinet an part is stored. You need to know that one part can be stored at multiple places in the system. You also should know that in one cabinet there can be one ore multiple drawers and in one drawer there can be one ore multiple boxes.

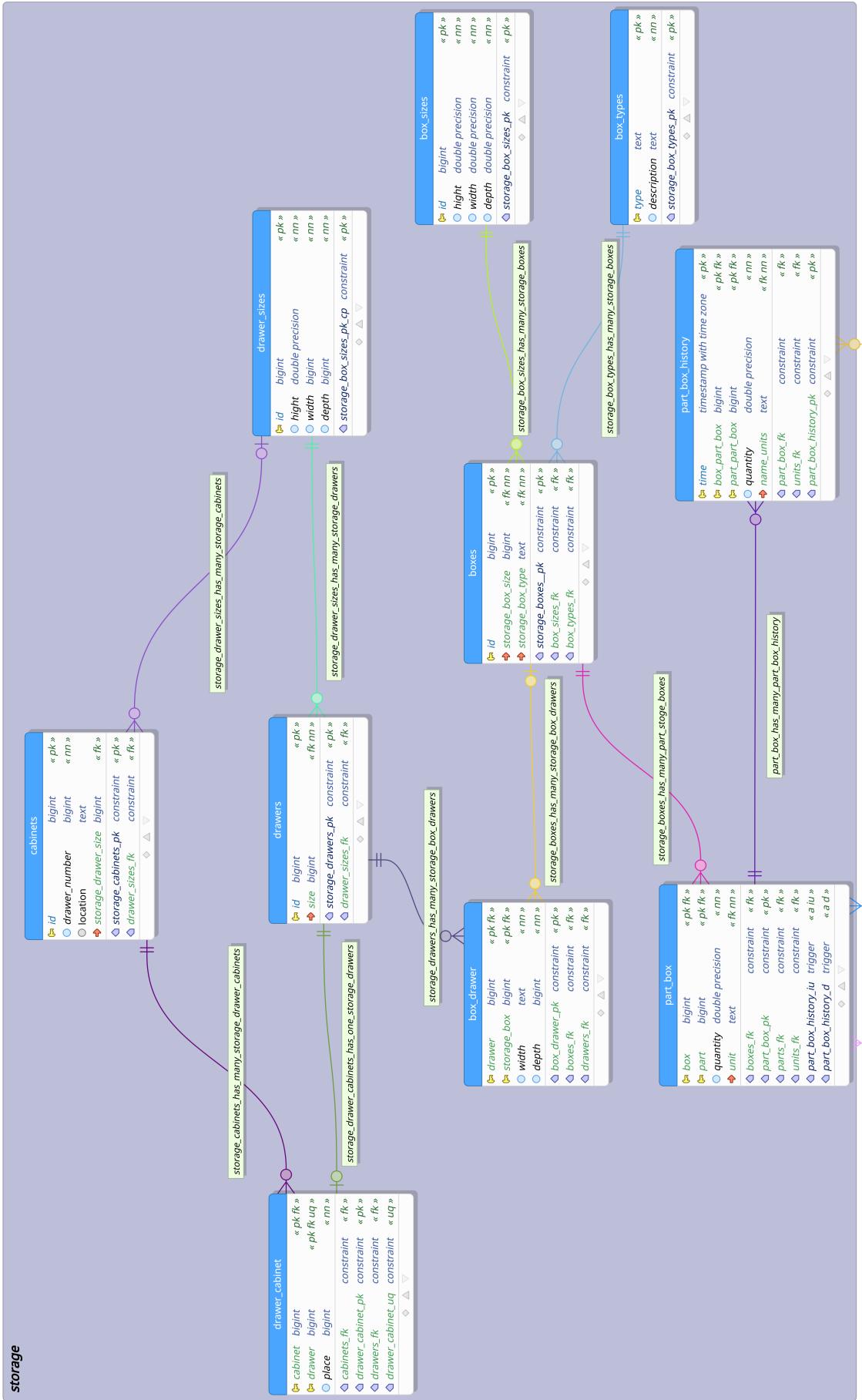


Figure 1.3: The design of the storage schema

Schema "global"

Here you can find all the things that are needed globally like units, currency's and file types.

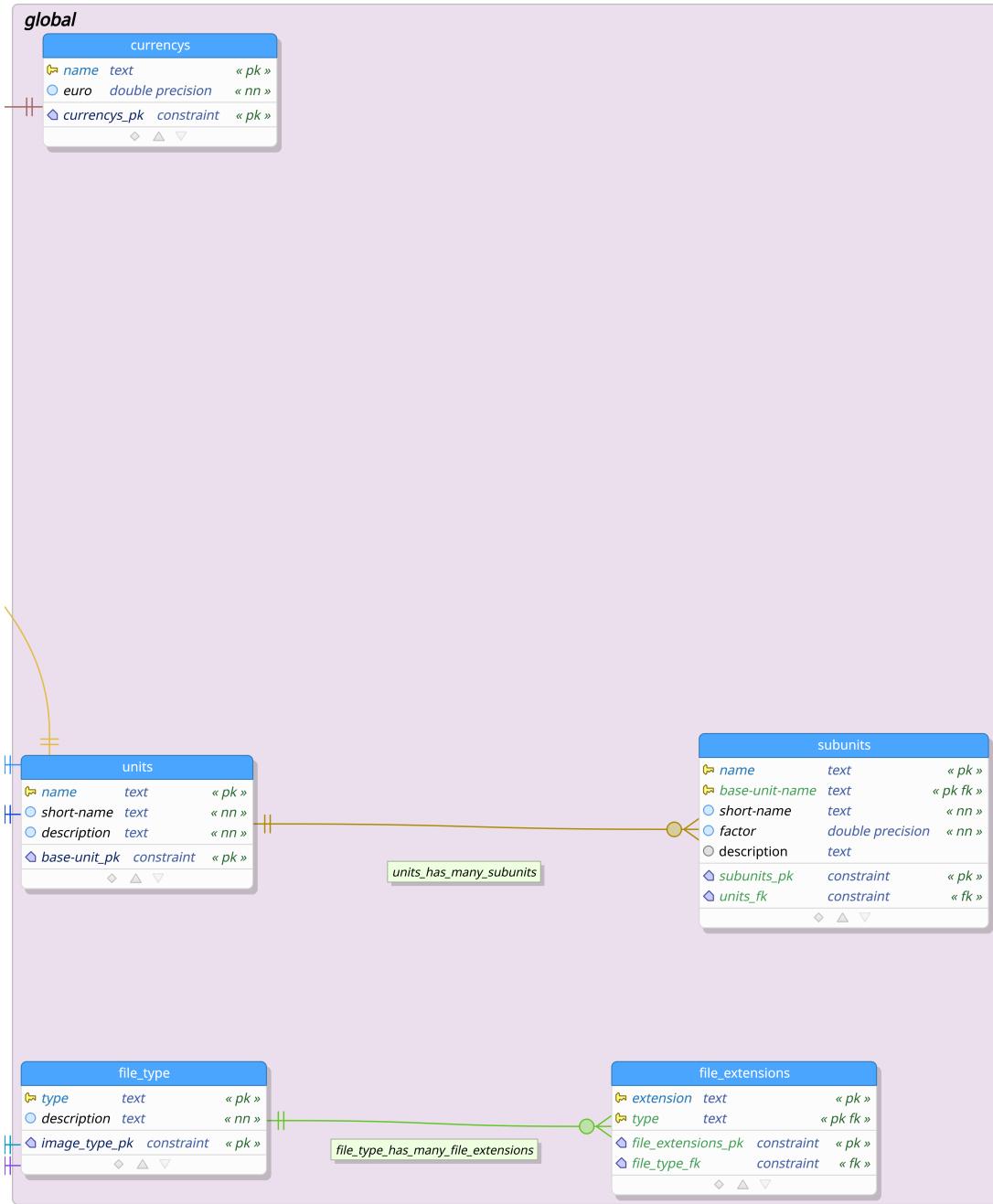


Figure 1.4: The design of the global schema

Schema "assemblies"

Here multiple parts are groped as assemblies to make it easier to fined all parts that go to one project.

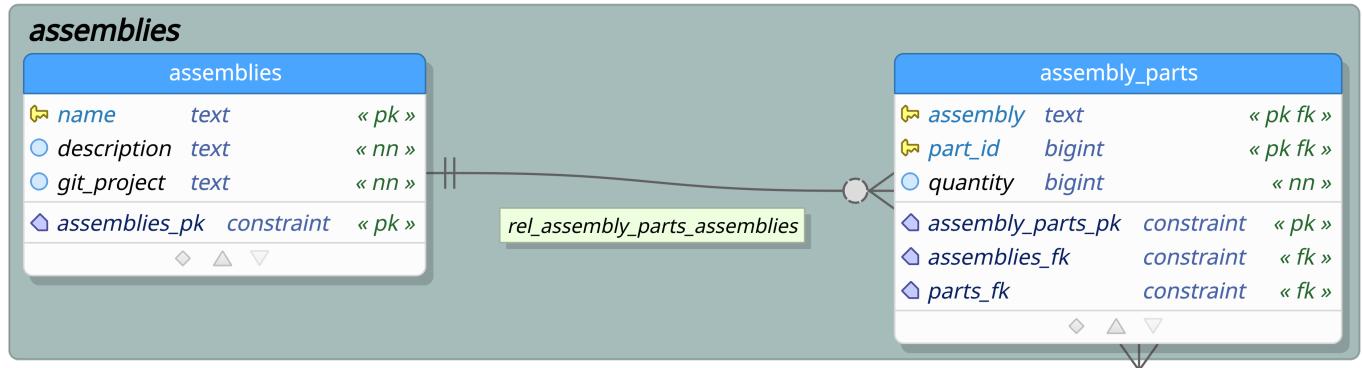


Figure 1.5: The design of the assemblies schema

Schema "vendors"

Here you can find all the necessary information about a vendor or manufacture.

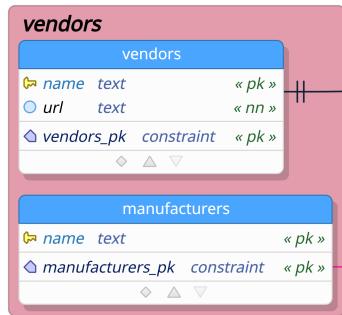


Figure 1.6: The design of the storage schema

Schema "kicad"

Here you can find the kicad schematic and pcb symbol for the parts.

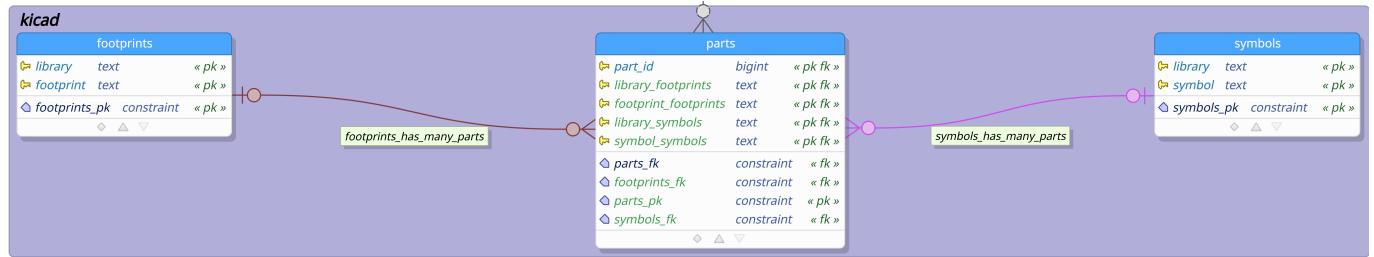


Figure 1.7: The design of the storage schema

1.2.2 middleware

The middleware has the task to make the http JSON requests to SQL requests and then return the data in JSON again.

1.3 Client

Chapter 2

Hardware

2.1 First Section

2.2 Second Section

2.3 Third Section

Bibliography