

Курсовая работа «Разработка автоматизированной системы сбора канальных матриц для обучения модели нейронной сети»

Руководитель:
Колпаков А.А.

Выполнил студент
Группы ПМИ-122
Ежков Д.А.

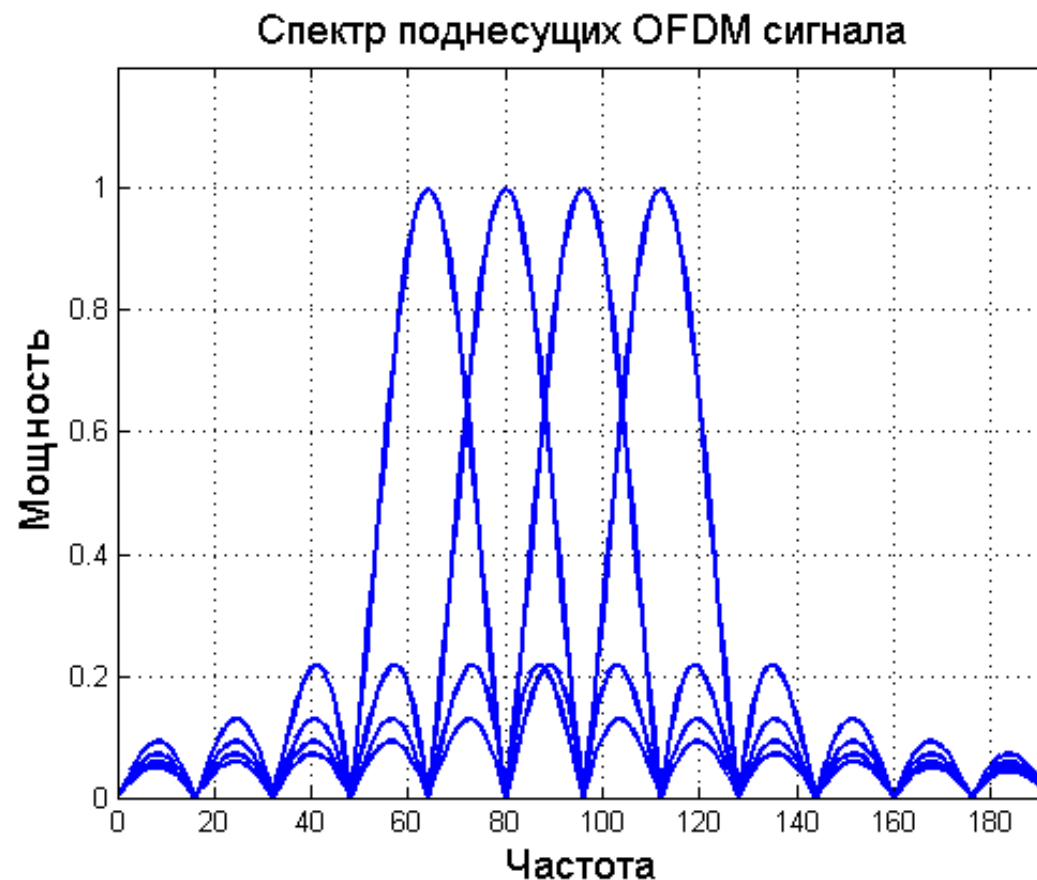
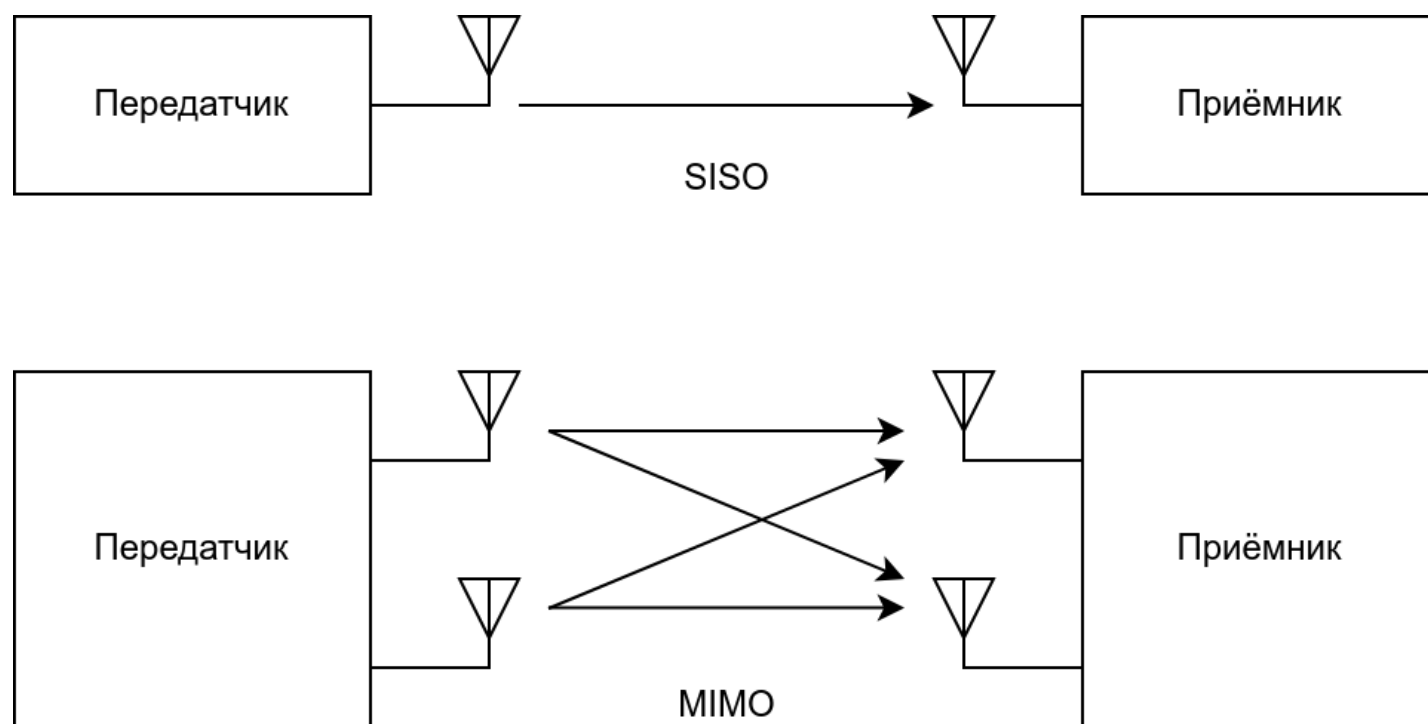
Цель работы: разработка
автоматизированной системы сбора
канальных матриц

Задачи:

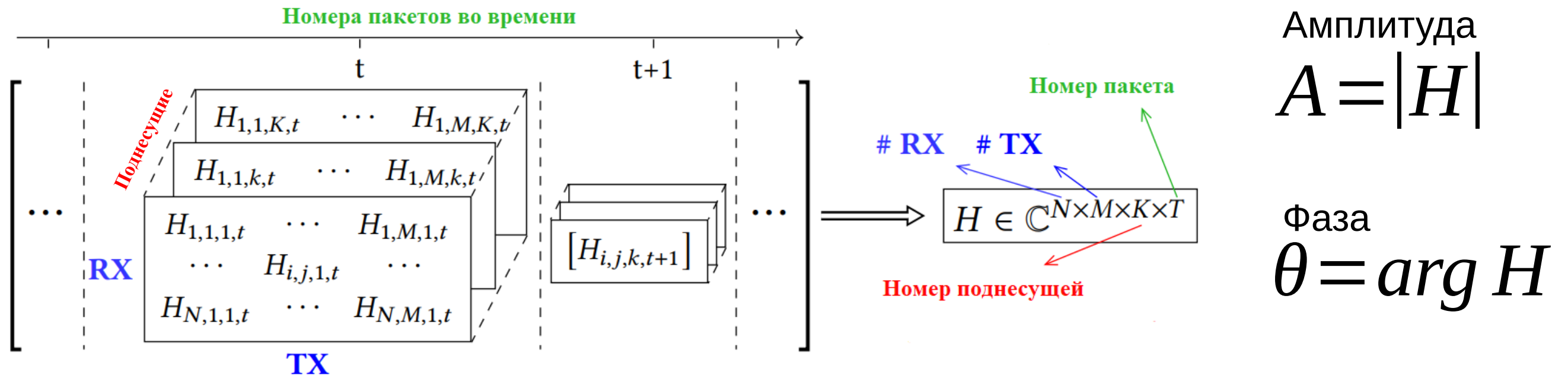
1. Проектирование концептуальной, логической и физической моделей базы данных.
2. Реализация физической модели.
3. Разработка приложения для взаимодействия с базой данных.

MIMO — Multiple Input Multiple Output
SISO — Single Input Single Output

OFDM — технология, позволяющая передавать данные с использованием большого количества близко расположенных поднесущих.

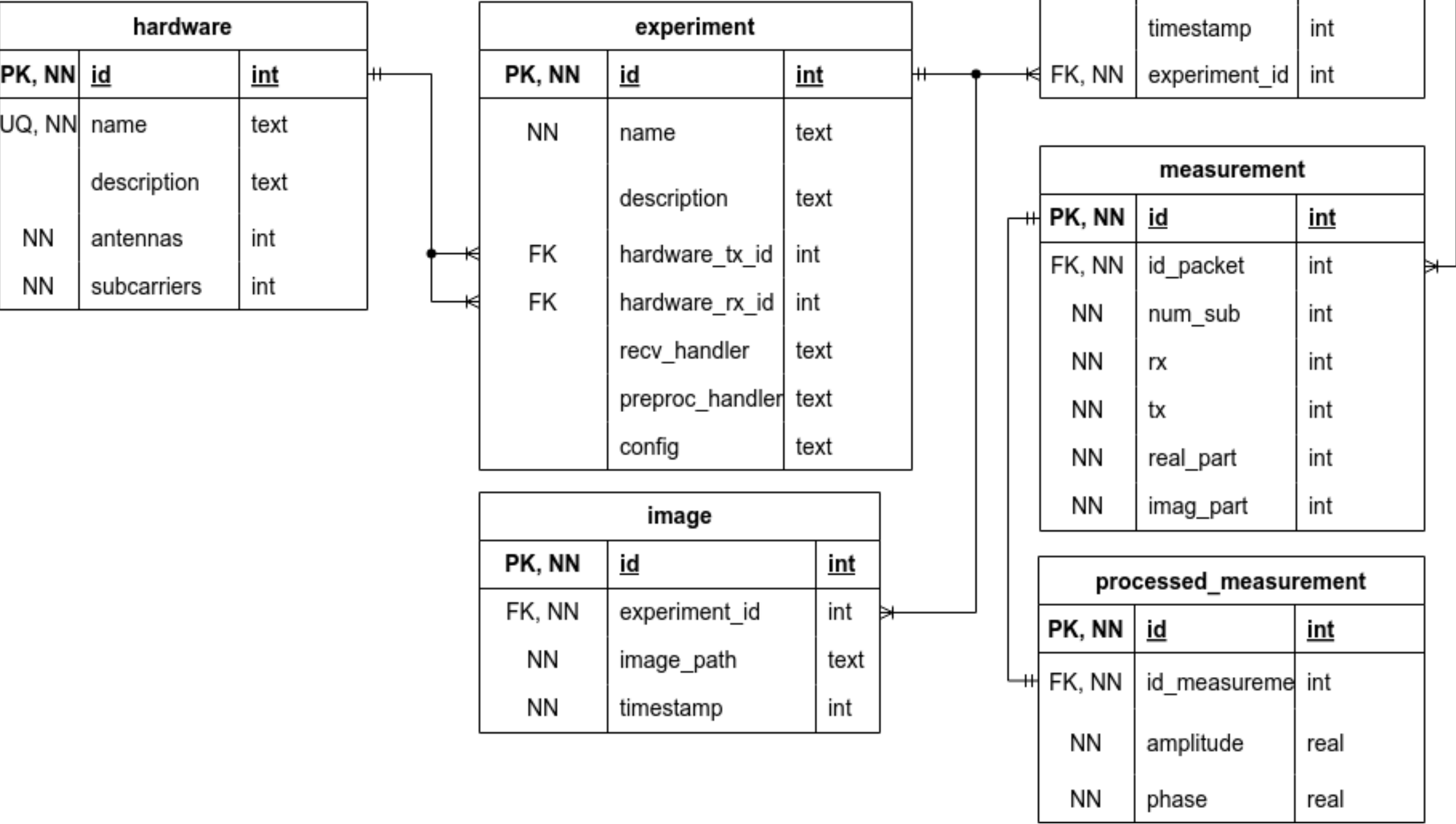


CSI — Channel State Information — информация о канале связи, описывающая распространение сигнала в пространстве. Каждый коэффициент CSI является комплексной амплитудой — т. е. описывает амплитуду и начальную фазу пришедшего сигнала.



Так, при скорости извлечения около 30 пакетов в секунду из роутеров с 3 антеннами, количество анализируемых параметров составляет $3 \times 3 \times 56 \times 30 \times 2 = 30240$.

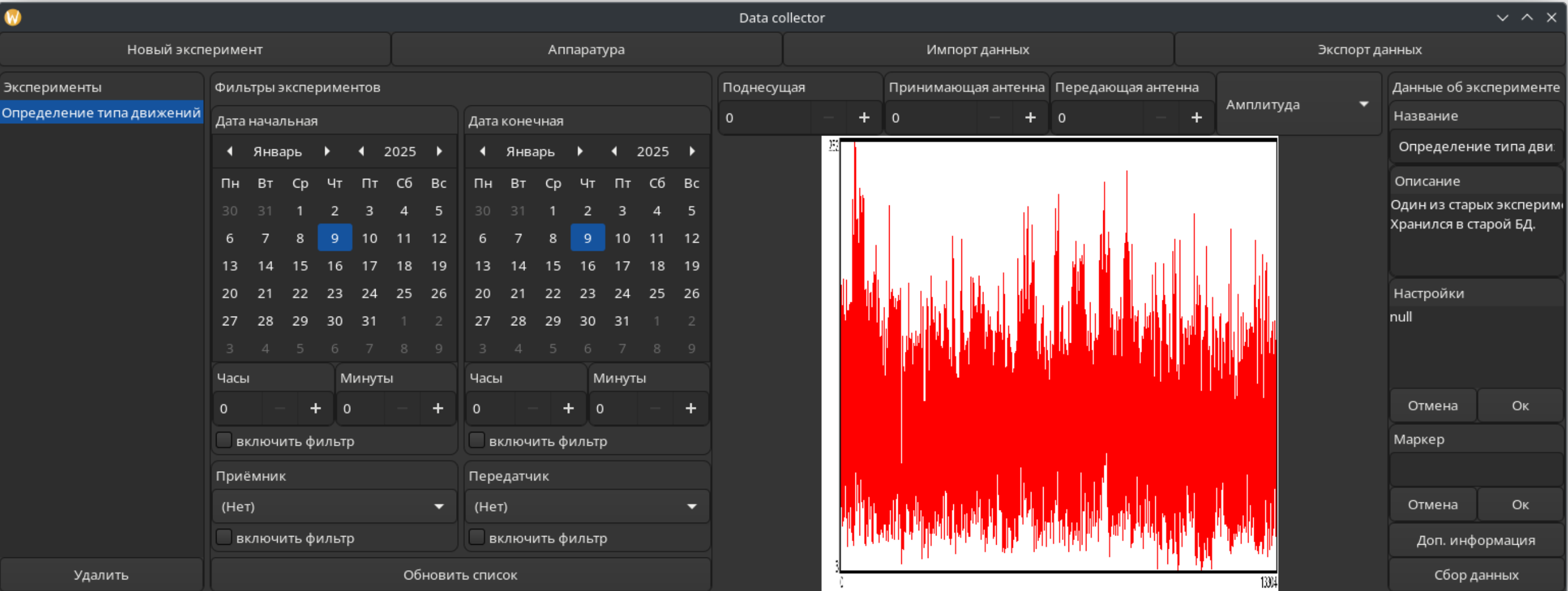
Физическая модель БД



Производительность SQLite

Database Page Size	BLOB size						
	10k	20k	50k	100k	200k	500k	1m
1024	1.535	1.020	0.608	0.456	0.330	0.247	0.233
2048	2.004	1.437	0.870	0.636	0.483	0.372	0.340
4096	2.261	1.886	1.173	0.890	0.701	0.526	0.487
8192	2.240	1.866	1.334	1.035	0.830	0.625	0.720
16384	2.439	1.757	1.292	1.023	0.829	0.820	0.598
32768	1.878	1.843	1.296	0.981	0.976	0.675	0.613
65536	1.256	1.255	1.339	0.983	0.769	0.687	0.609

Интерфейс главного окна



Запрос для фильтрации датасетов

```
SELECT  id,    name,    description,    hardware_tx_id,    hardware_rx_id,    config,  
recv_handler, preproc_handler FROM experiment  
  
WHERE  
  
id IN (SELECT experiment.id FROM experiment JOIN packet ON packet.experiment_id  
= experiment.id WHERE timestamp > @min_time) AND  
  
id IN (SELECT experiment.id FROM experiment JOIN packet ON packet.experiment_id  
= experiment.id WHERE timestamp < @max_time) AND  
  
hardware_tx_id = @transIdx AND  
  
hardware_rx_id = @recvIdx
```


Запрос для экспорта датасетов

```
SELECT measurement.num_sub, measurement.rx, measurement.tx, amplitude, phase,  
measurement.real_part, measurement.imag_part, packet.id  
FROM processed_measurement  
INNER JOIN measurement ON processed_measurement.id_measurement = measurement.id  
INNER JOIN packet ON measurement.id_packet = packet.id  
INNER JOIN experiment ON packet.experiment_id = experiment.id  
WHERE experiment.id = @exp_id AND packet.marker LIKE @marker  
ORDER BY packet.id, measurement.num_sub
```

Результаты экспорта

```
de@de-laptop:~/export_test$ tree
.
├── ampls
│   ├── 0_0.json
│   ├── 0_1.json
│   ├── 1_0.json
│   └── 1_1.json
├── phase
│   ├── 0_0.json
│   ├── 0_1.json
│   ├── 1_0.json
│   └── 1_1.json
└── photos
    ├── _1736372813.jpg
    ├── _1736372814.jpg
    ├── _1736372815.jpg
    ├── _1736372992.jpg
    ├── _1736373005.jpg
    ├── _1736373006.jpg
    ├── _1736373007.jpg
    ├── _1736373008.jpg
    ├── _1736373009.jpg
    ├── _1736373010.jpg
    ├── _1736373011.jpg
    ├── _1736373012.jpg
    ├── _1736373013.jpg
    ├── _1736373014.jpg
    ├── _1736373015.jpg
    ├── _1736373016.jpg
    └── _1736373017.jpg
```

```
< > {} 0_0.json x
home > de > export_test > ampls > {} 0_0.json

1  [
2  [50,4777, 50,4777, 53,8516, 54,2033, 58,1378, 57,7754, 57,2713, 58,1378, 61,2699, 58,1808, 58,1808, 55,9464, 59,0593, 58,1808, 58,3095,
4  58,8303, 56,0357, 59,4054, 56,8507, 54,4518, 55,1725, 56,5685, 55,2268, 54,037, 54,9181, 54,2033, 53,2353, 54,1295, 55,5428, 53,4509,
4  54,1202, 54,5711, 51,788, 52,9528, 51,1566, 50,0899, 52, 53,0094, 53,2353, 49,4975, 52,469, 48,0521, 50,6952, 50,9608, 50, 50,5964,
4  51,8652, 48,4665, 50,6952, 45,31, 47,1275, 47,1275, 45,5412, 44,9222, 43,0116, 41]
3  [47,1699, 46,8188, 48,0833, 50,448, 52,1728, 52,6118, 51,3517, 52,469, 51,2445, 53,3385, 52,2398, 53,2353, 54,231, 53,2353, 53,2353,
4  55,2268, 52,345, 52,9528, 50,8035, 51,1957, 52,1728, 52,3927, 51,6624, 51,2445, 50,5964, 49,98, 50,3289, 49,2443, 48,3011, 49,1935,
4  47,8853, 48,3735, 47,7074, 46,8722, 48,0104, 47,2017, 46,8615, 48,4149, 47,5079, 48,1664, 46,6905, 46,669, 46,0109, 46,0109, 45,9674,
4  44,5982, 44,5982, 44,5982, 43,8634, 43,8406, 43,1856, 41,8688, 41,1096, 40,6079, 40,2244, 36,3593]
4  [169,956, 174,27, 184,459, 186,067, 196,092, 201,559, 199,161, 192,042, 200,202, 192,023, 198, 193,023, 197,124, 198,162, 204,626,
4  200,808, 194,612, 199,745, 192,855, 181,199, 187,641, 194,201, 185,699, 186,132, 179,109, 183,505, 183,33, 177,429, 182,222, 181,761,
4  179,744, 179,212, 173,807, 183,831, 176,709, 175,026, 171,353, 180,228, 174,642, 172,525, 181,298, 172,047, 170,018, 168,122, 175,525,
4  169,499, 166,721, 161,245, 169,567, 163,11, 165,221, 163,918, 152,453, 147,092, 142,394, 140,943]
5  [199,201, 208,339, 212,897, 219,966, 224,288, 232,131, 228,327, 230, 230,417, 232,407, 228,079, 231,815, 232,71, 232,433, 234,241,
4  233,805, 229,268, 230,638, 229,268, 223,958, 227,969, 224,555, 222,794, 224,163, 219,203, 220,511, 220,62, 218,449, 217,412, 211,066,
4  216,067, 211,341, 209,402, 208,255, 208,425, 203,037, 205,273, 203,494, 207,338, 207,993, 206,344, 204,963, 200,609, 202,141, 202,072,
4  194,288, 196,865, 193,352, 191,269, 194,064, 193,546, 188,043, 179,789, 173,666, 170,47, 159,762]
6  [83,4086, 86,2148, 94,7523, 97,7446, 98,4073, 102,903, 96,8349, 96,566, 101,533, 97,2471, 101,533, 96,3846, 102,956, 101,074, 102,577,
```

