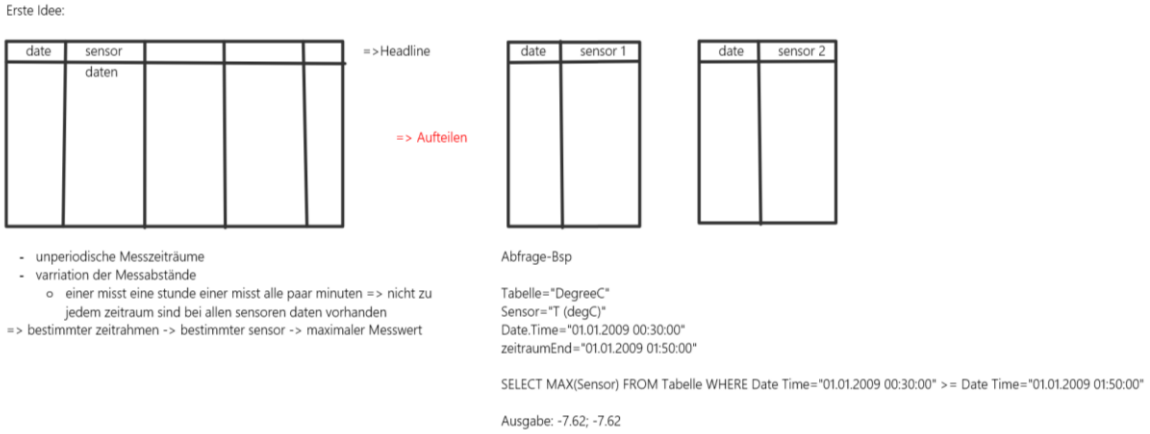
Praktikum: Informationssysteme

Aufgabe 1-3:

Bestand aus Konfiguration für aufgabe 4. → Konfiguration wurde erfolgreich erledigt.

Aufgabe 4a:



→ Ansatz 1: Aufteilung der Tabelle in einzelne Tabellen mit Zeitpunkt als key

Tabellenname	
Key (datum der Messung)	sensordaten

→ Ansatz 2: Tabelle so wie sie ist beibehalten und Zeitpunkt als key für gesamte tabelle nutzen

Tabellenname									
Key	Sensordaten	Sensordaten	Sensordaten	Sensordaten	Sensordaten	etc.	..	..	..

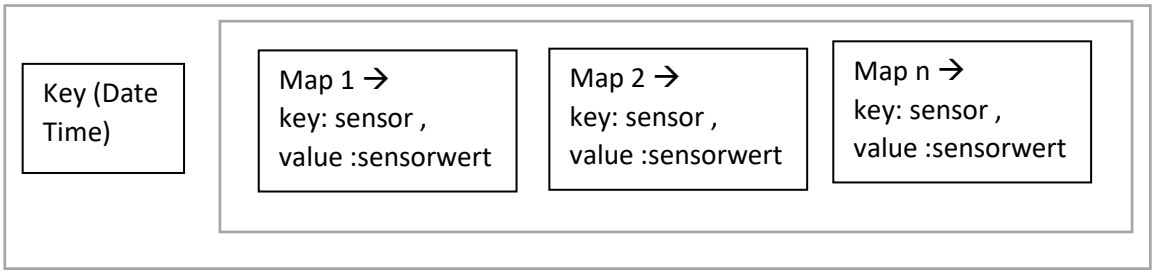
Abfrage: SELECT MAX(sensor) FROM Tabellenname WHERE Date Time="01.01.2009 00:30:00" >= Date Time="01.01.2009 01:50:00"

Performance:

Ansatz 1: Im Vergleich zu Ansatz 2 ist die Performance geschätzt etwas besser, da wir immer nur die Zeitpunkte für Daten abfragen die auch exestieren. Bsp: Im Ansatz 2 nehmen nur die hälfte der Sensoren Daten auf, key ist für die ganze Tabelle erstellt wurden. Das bedeutet manche Sensoren haben zu dieser Zeit keine Daten in der Tabelle -> bei einer Abfrage zwischen zwei Zeitpunkten werden diese ebenfalls überprüft. Bei Ansatz 1 werden Zeitpunkte und Daten gespeichert die tatsächlich vorliegen → Ansatz 1 bessere Performance als Ansatz 2 -> Allgemein betrachtet hat Ansatz 1 eine gute Performance

Aufgabe 4b:

Ansatz: Wir nehmen einen Hash. In diesen Hash wird ein key(Date Time) gespeichert und als Value setzen wir Maps(Sensordaten) hinein. → In einem Hash befinden sich mehrere Maps. Eine Map beinhaltet den Sensortyp und den dazugehörigen wert.



Als erstes wird die Csv-Datei ausgelesen und bei jeder Reihe werden die Sensordaten in solch einen Hash gespeichert. Mit jedis buffern wir die Anfragen für jeden Hash an den Redis Server in einer Pipeline und schicken sie alle auf einmal ab, wenn alle Reihen ausgelesen sind und sich in der Pipeline befinden. → Extrem niedriger Traffic. Somit nutzen wir die Stärke der Schnelligkeit von Redis. Der Server empfängt die Pipeline und führt alle darin enthaltenen Befehle auf einmal aus. → Csvdaten sollten sich jetzt in Redis befinden.

Abfrage: Werte in einem Hash werden mit dem Date Time Key abgefragt. Um den genauen Sensorwert zu bekommen fügen wir noch den Sensor Key hinzu, durch den wir den genauen Sensorwert zu dem bestimmten Zeitpunkt haben. Jedisbefehle mit dieser Abfrage werden solange in einer Pipeline gespeichert, bis das ende des Abfragezeitraums erreicht ist und die Pipe wird abgeschickt. Nun lesen wir den Request aus und geben den Max-Wert aus.

➔ Wereitere Erklärungen finden Sie im beigefügtem Code an allen wichtigen Stellen.