Bitte den Allgemeinen Teil du weil ich habe die Vorlagen nicht dafür.

Die SOA spezifischen Sachen von dir.

Bitte den Client den du generiert hast beschreiben da kenn ich mich net aus. (Ist glaub ich der dritte Punkt)

# Datenmodell

Das Datenmodell wurde realisiert über eine Hashmap in der das Topic als Key und der Content als Value gespeichert wurde. Dadurch gestaltet sich die Suche äußerst einfach durch eine Methode die von der Klasse Hashmap zur Verfügung gestellt wird. Auch das speichern gestaltet sich einfach durch von Java zur Verfügung gestellte Klassen.

## Suche

Die Methode zur Suche heißt: map.getValue(Key)

Zurückgegeben wird der Value also das Topic der Wissensdatenbank.

Die gesamte Such-Methode:

**public** String getEntry(String key) {

**return**(know.get(key));

}

Dabei muss beachtet werden, dass die Suche Casesensitive ist. (Also das genaue Topic)

## Speichern

Das Speichern einer Hashmap kann durch die von Java zur Verfügung gestellte Klasse Properties gelöst werden. Diese ist für das persistente speichern von Hashmaps ausgelegt.

Schritte:

1. Erstellen einer Properties Variable:

Properties properties = **new** Properties();

2. Setzen der einzelnen Properties aus der Hashmap:

properties.setProperty((String)pairs.getKey(), (String)pairs.getValue());

3. Mit einem Outputstream speichern hier wird ein FileWriter verwendet die Methode zum speichern von Properties ist store:

FileWriter writer;

writer = **new** FileWriter("conf.properties");

properties.store(writer, **null**);

## Laden

Das Laden kann ebenfalls mit einer Methode der Klasse Proberties gelöst werden diese ist load

Erster Schritt gleich wie bei speichern erstellen einer Proberties Variable.

2. Zweiter Schritt mittels eines Inputstreams lesen:

FileReader reader;

reader = **new** FileReader("conf.properties");

properties.load(reader);

3. Anschließend muss das Ergebnis noch in eine Hashmap gespeichert werden:

**for** (**final** String name: properties.stringPropertyNames())

know.put(name, properties.getProperty(name));

## Hinzufügen

Das Hinzufügen eines Eintrags geht ganz einfach mit der Methode put() der Klasse Hashmap wenn der Eintrag schon vorhanden ist wird der Content nur hinzugefügt das funktioniert folgendermaßen:

**if**(know.containsKey(key) == **false**) {

know.put(key,value)

}

**else** {

String hilf = know.get(key) + "\n" + value;

know.remove("key");

know.put(key, hilf);

}

# Dokumentation des Services

# Dokumentation des Service-Clients