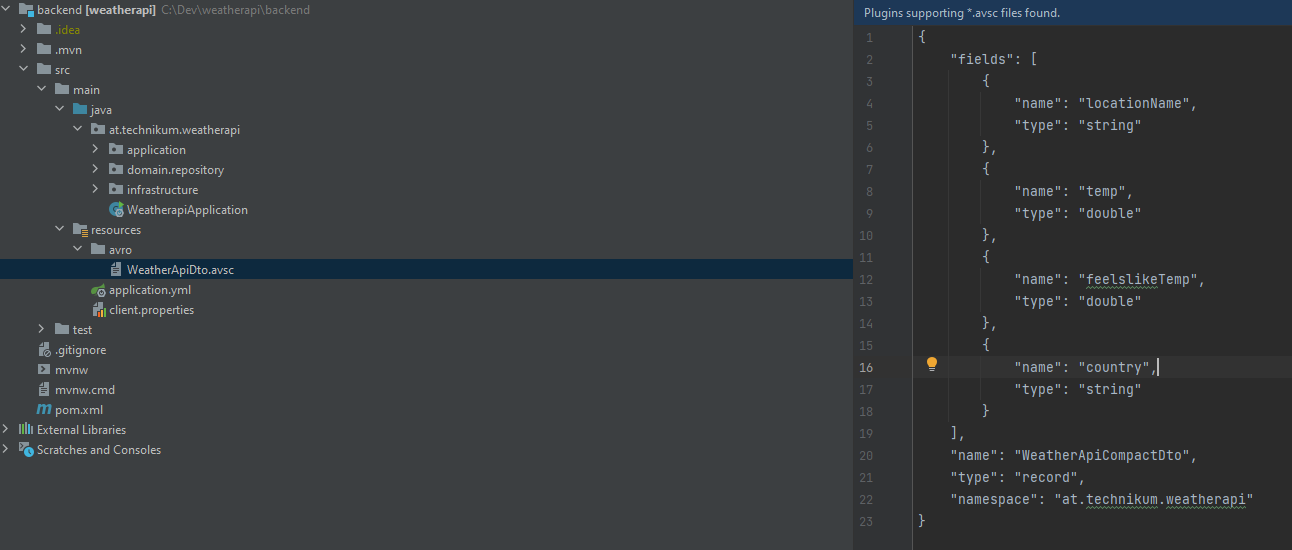
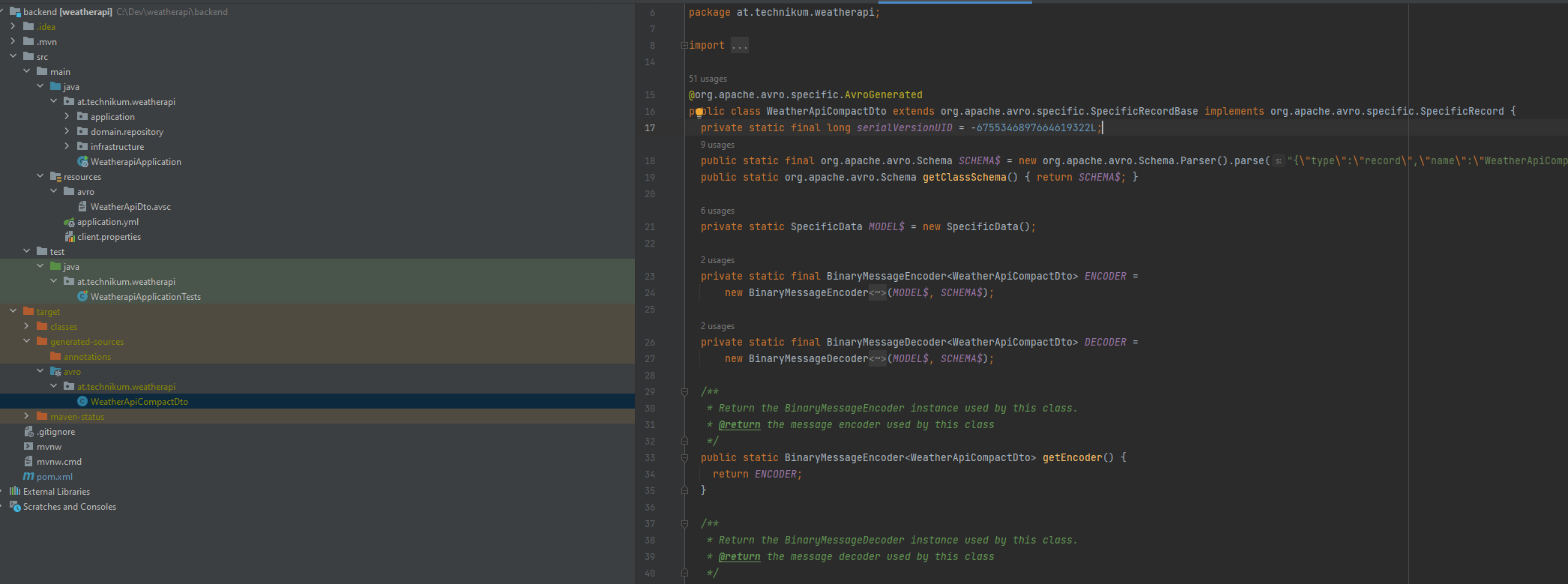
## Abgabe SFR: Streaming

Wir haben für diese Aufgabe unseren Kafka auf den von confluent umgestellt da dieser ein simples Schema registry bereitstellt. Unser Backend generiert nun aus einem Avro schema eine Klasse, welche dann in den Kafka gesendet wird. Confluent legt dann automatisch das Schema nach einem Kompatibilitätscheck als aktuelle Version ab.

Bsp.: Unser Avro Schema:



Die daraus generierte Klasse:

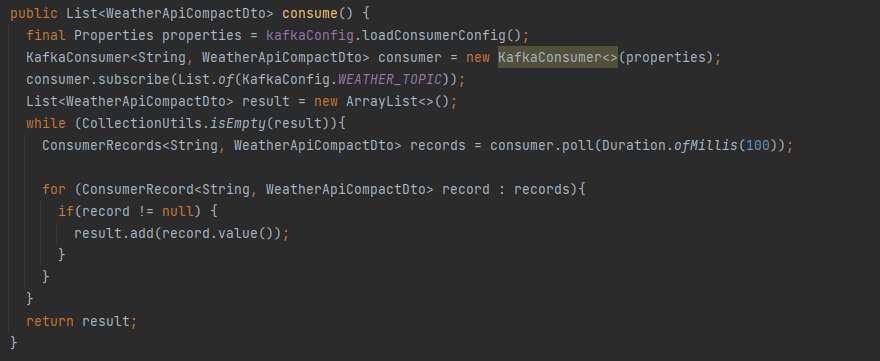


Verwendung der Klasse:

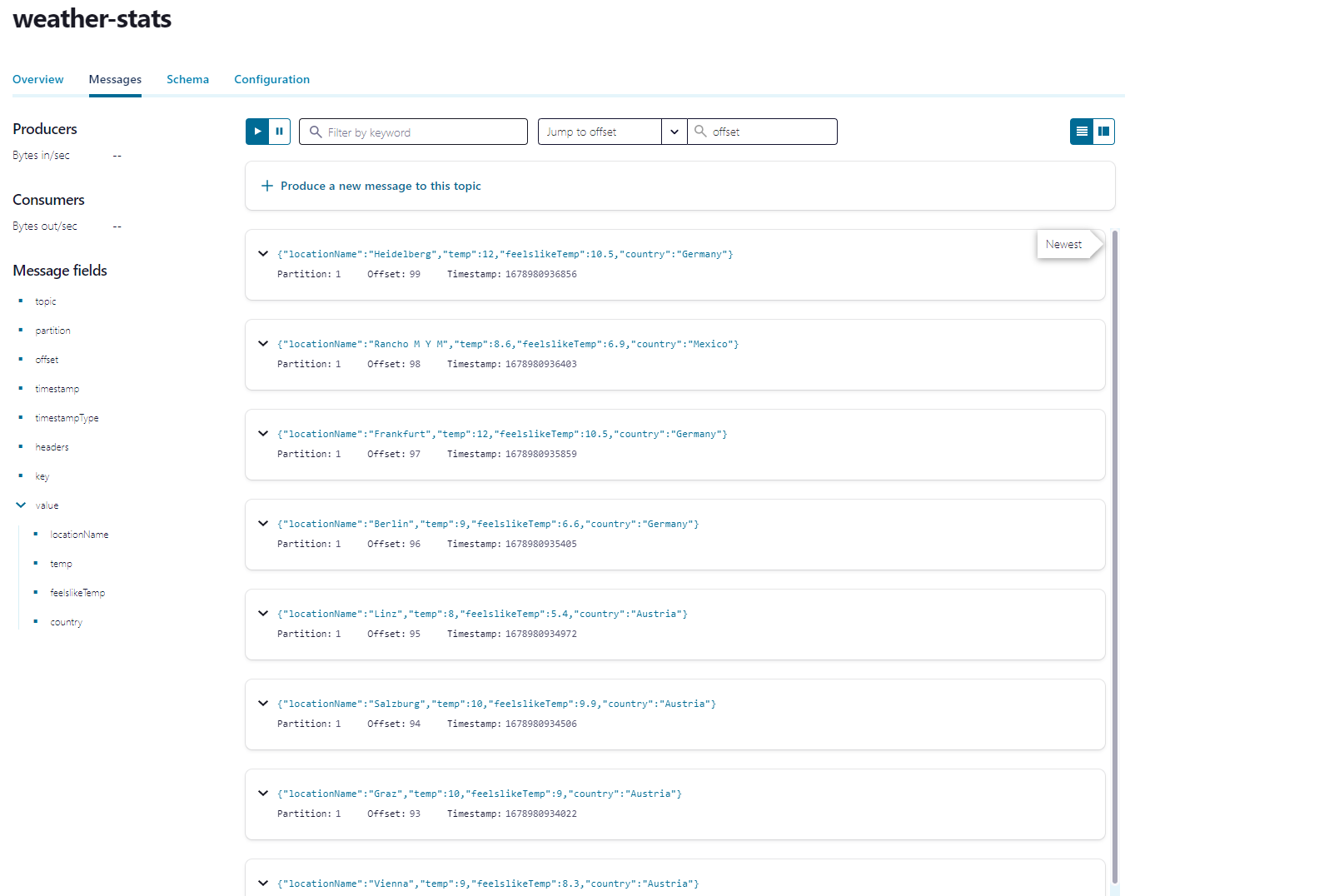


Der Consumer müsste sich das Schema holen, ebenfalls eine Klasse generieren und dann damit arbeiten:

Der Consumer:



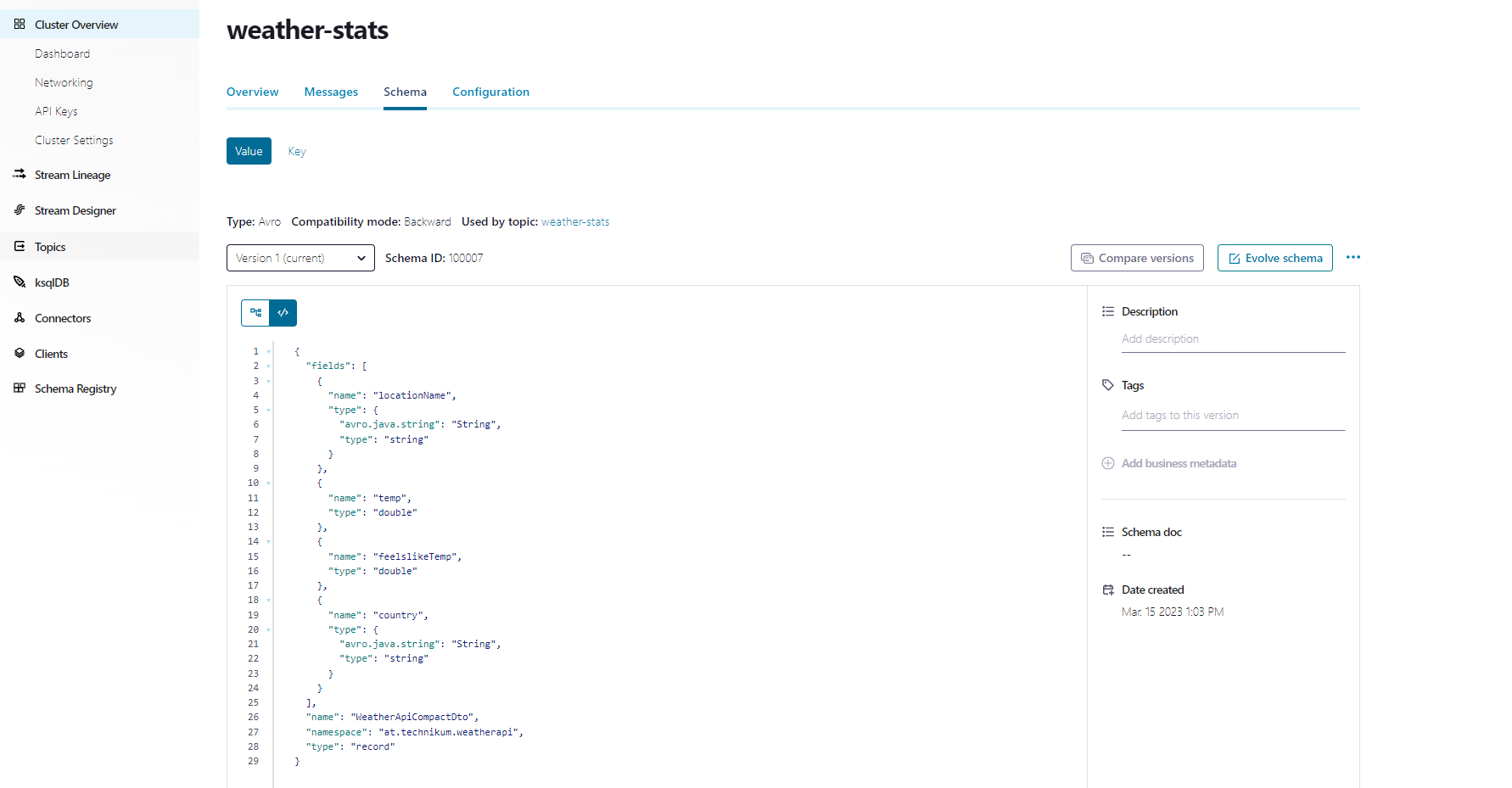
Auf confluent sieht dass dann beispielsweise so aus:



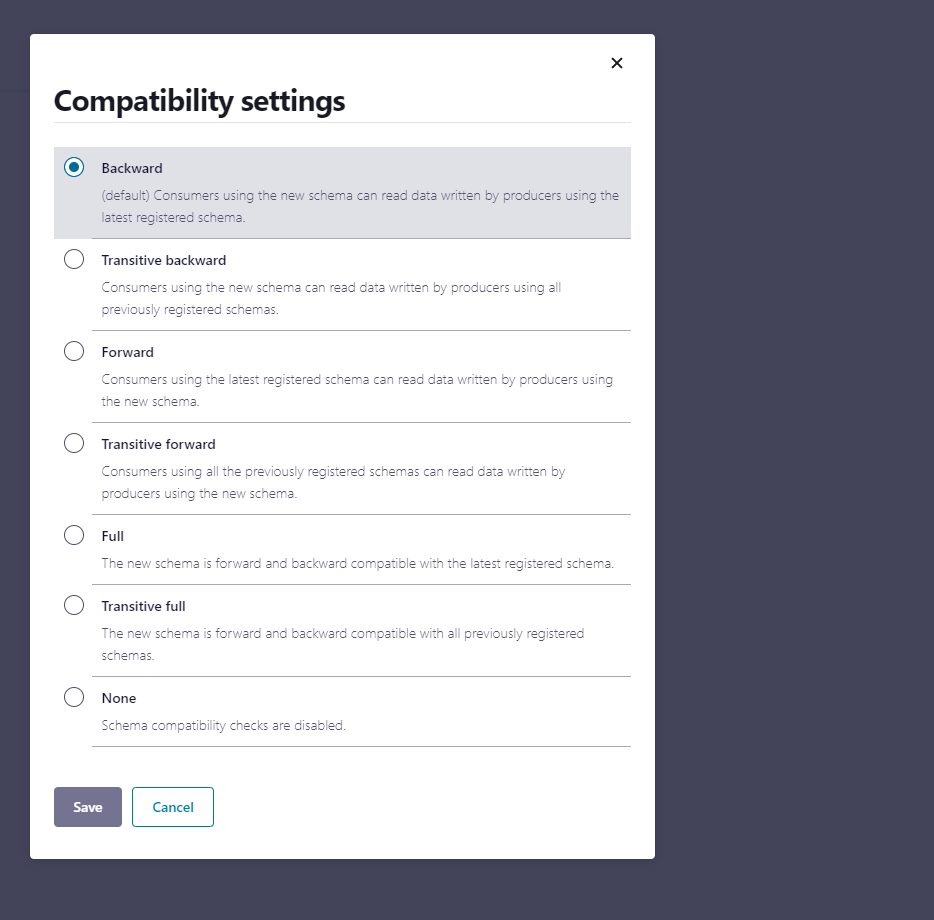
## Schema Registry und Kompatibilität:

Über Confluent kann man ziemlich gut die Schemas verwalten und die Kompatibilität einstellen und checken:

Bsp.: unser derzeitiges (vom Backend erstelltest) Schema:

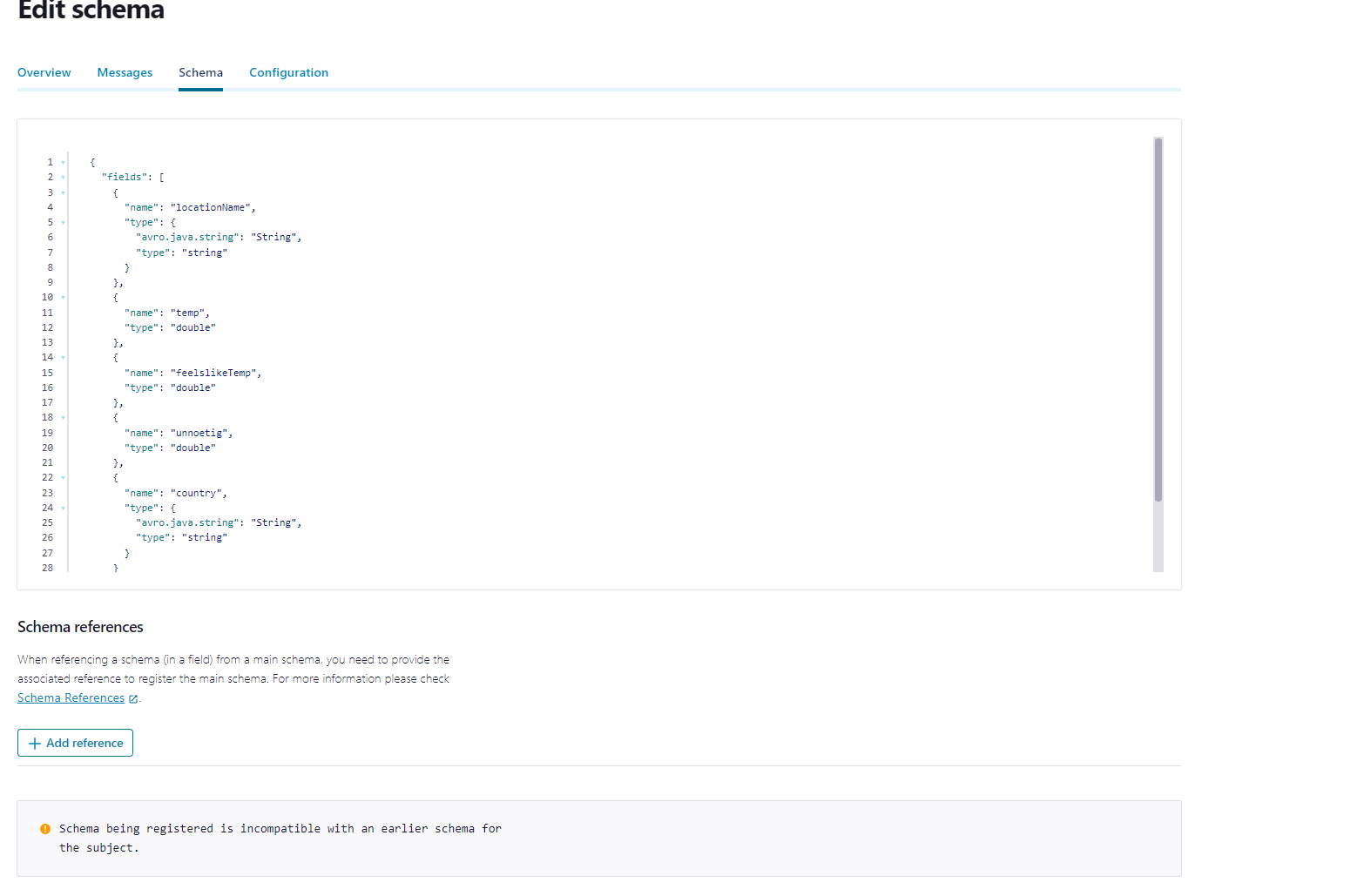


Derzeit ist Backward compatibility eingestellt:

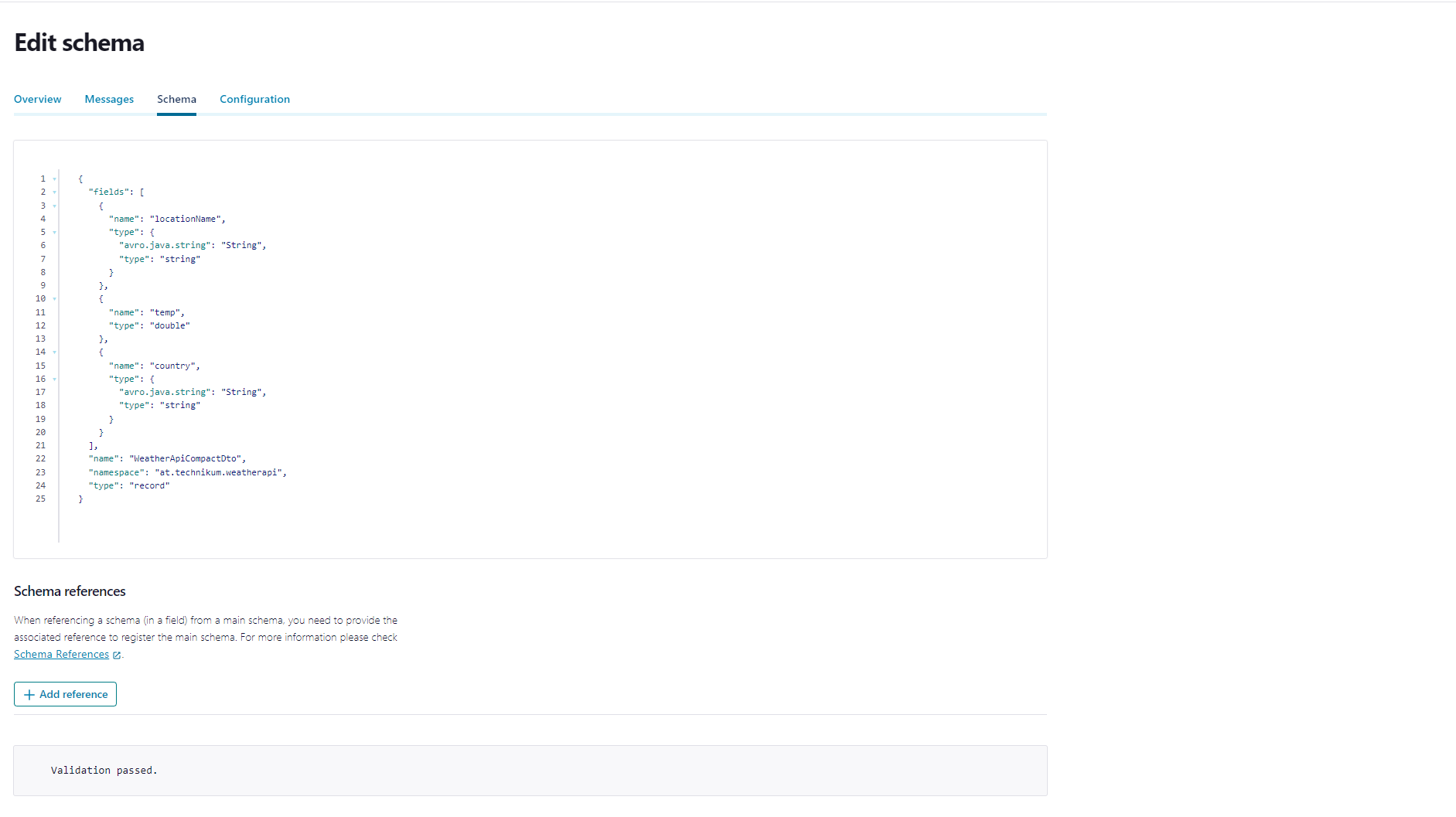


Wenn ich jetzt auf „Evolve Schema“ klicke kann ich das direkt in der cloud im Topic verwalten.

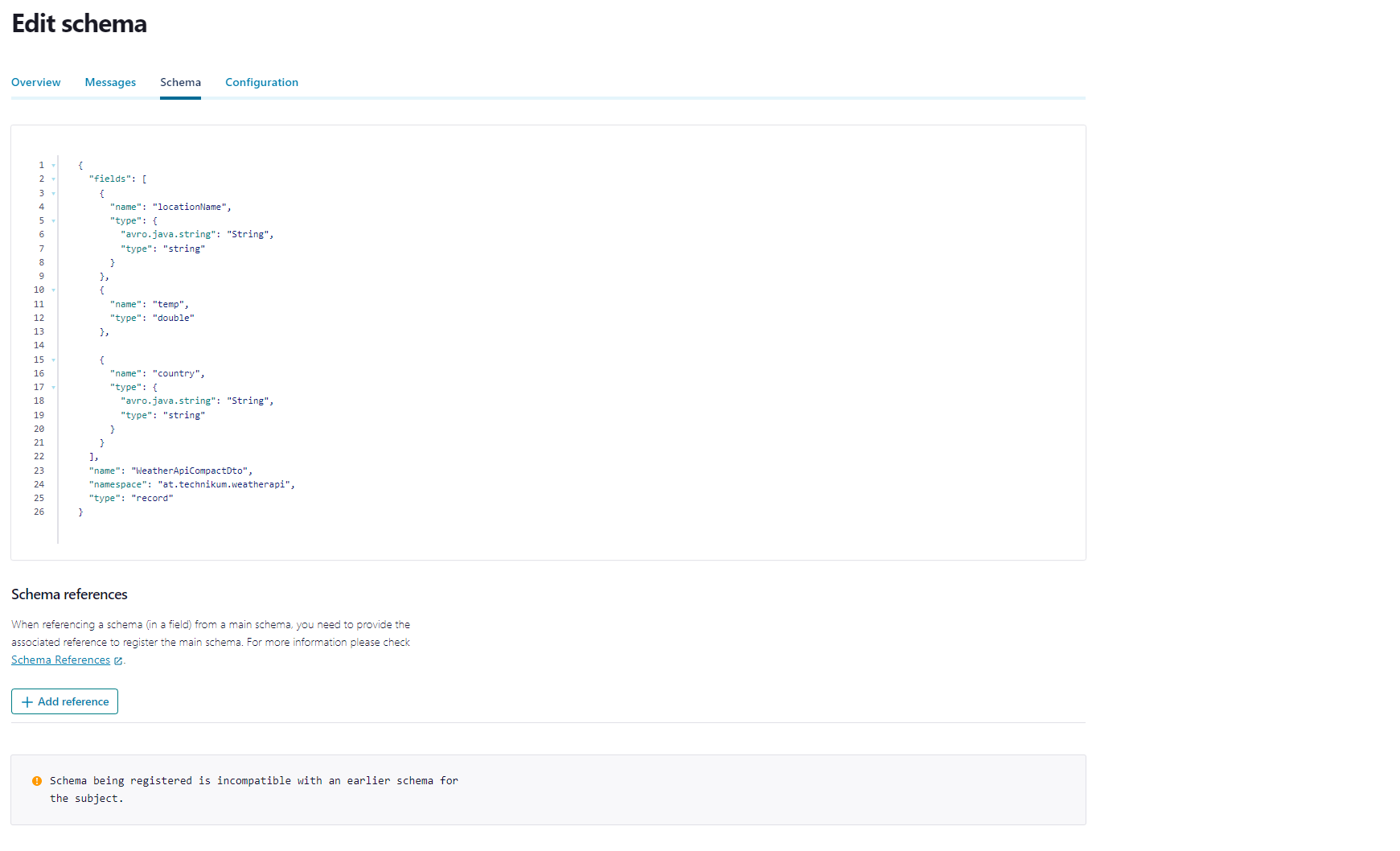
Wegen der Backward-Kompatibilität darf ich beispielsweise kein Attribut hinzufügen.

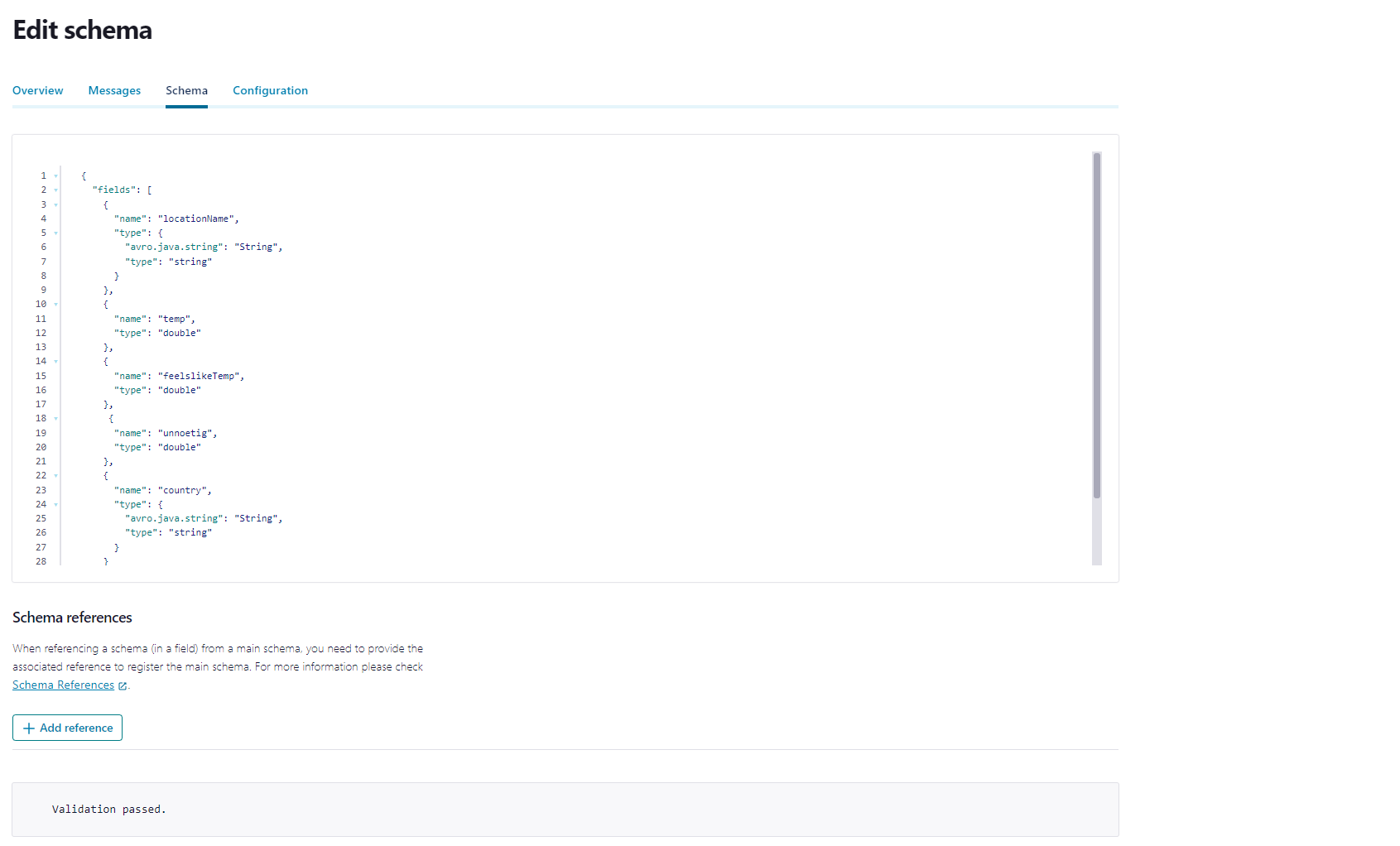


Ich kann aber eins löschen



Umgekehrt bei Forward:





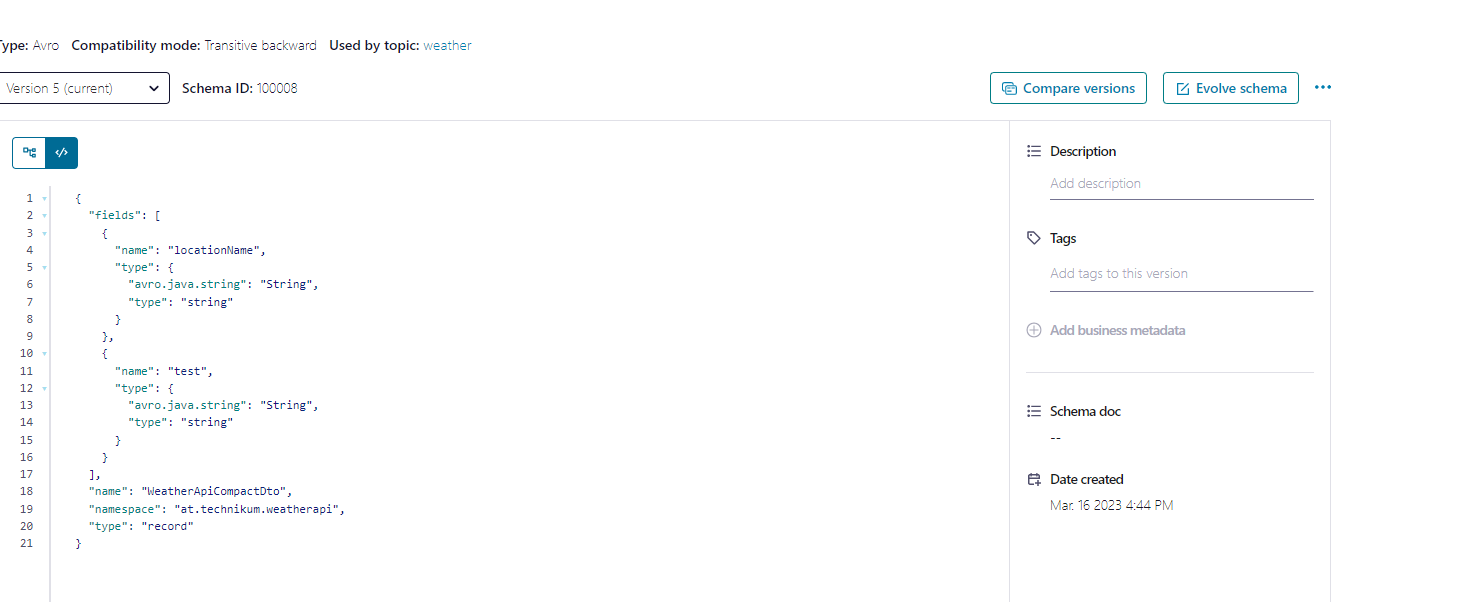
Bei Full kann ich garnichts machen außer optionales löschen und hinzufügen.

Transitive backward ist interessant weil ich so ein bestehendes schema ungültig machen kann.

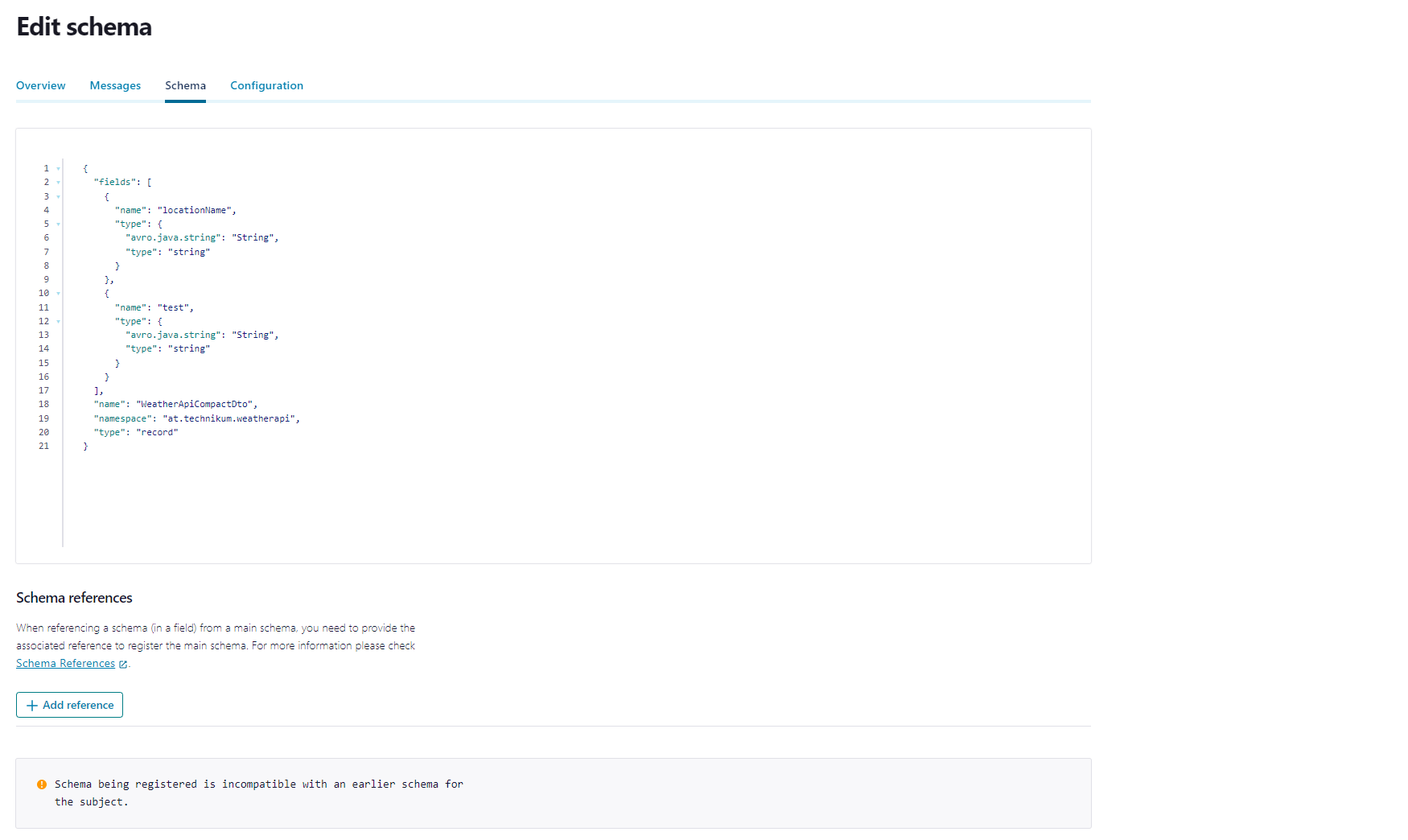
Bsp: Version 4 eines schemas hatte 1 Attribut.



Version 5 hat 2 (kurzfristig auf Forward gestellt)



Wenn ich jetzt auf transitive Backward umstelle und versuch dasselbe nochmal abzuspeichern bekomme ich einen Konflikt



Transitive berücksichtigt auch alle Versionen davor. Da Version 5 im Gegensatz zu Version nicht Backward kompatibel ist bekomme ich einen Fehler.

# Aggregieren

Beim aggregieren werden bei uns folgende Einträge konsumiert:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

(Anzahl insgesamt 8 – einmal pro Stadt in der Liste cities in WeatherService)

Dann wird das alles nach dem Land gruppiert und die Temperaturen werden in eine Liste geschrieben und in einem Topic materialized:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Zu guter Letzt wird dann noch ein Topic nur mit den Durchschnittswerten befüllt:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Als Keys werden dabei immer die Länder verwendet:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung