



Fig. 8 The mapping for **GROUP BY**, **HAVING**, and **MATCH RECOGNIZE**.

Group-By follows standard SQL syntax. The query that aligns with our semantics is reported in Listing 14. The query is valid syntactically and its abstract syntax tree is the following. The query is minimal, as the **GROUP BY** needs to be integrated with aggregation and projection in the same query.

```

1 SELECT therm, avg(temp)
2 FROM BaseThermRead
3 GROUP BY therm;

```

Listing 14 An atomic query for group by

```

SELECT
|
FROM
|
GROUP BY

```

Having follows standard SQL syntax. Since **HAVING** imposes conditions on the aggregates calculated through the **GROUP BY**, its minimal query can be reduced to the one related to the **WHERE**. Such query will impose conditions over the results of the grouped aggregations (e.g. the one from Listing 14), treating aggregates as simple attributes, like in the following Listing 15 at Lines 6-8

```

1 INSERT INTO AggregatedBaseThermRead
2 SELECT therm, avg(temp)
3 FROM BaseThermRead
4 GROUP BY therm;
5
6 SELECT *
7 FROM AggregatedBaseThermRead
8 WHERE avg(temp)

```

Listing 15 An atomic query for having

Similar to EPL Patterns the correctness of the **Match Recognize** is provided by the processing model, which directly derives from NFA [15]. Moreover, **MATCH RECOGNIZE** present a fixed syntax, and the operators within it cannot be isolated

from each other. Thus, to simplify our method, we consider the entire construct as an atomic query.