

```
chain_of(
  with_elements(load_postgres_table(("public", "patient"), ["id"], [Int32])),
  flatten())
```



The diagram illustrates the execution of a query plan for the query: `load_table('patient',['id']) SELECT id FROM patient`. The plan is shown as a sequence of operators and their outputs, with a central execution flow and a side view of the data types and shapes.

Query Plan Operators (Left Side):

- `load_table('patient',['id']) SELECT id FROM patient` (Green box)
- `head` (Yellow box)
- `1` (Yellow box)
- `flatten()` (Green box)
- `cardinality(x1to1)` (Green box)
- `output()` (Green box)
- `column(1)` (Green box)
- `head` (Yellow box)
- `1` (Yellow box)

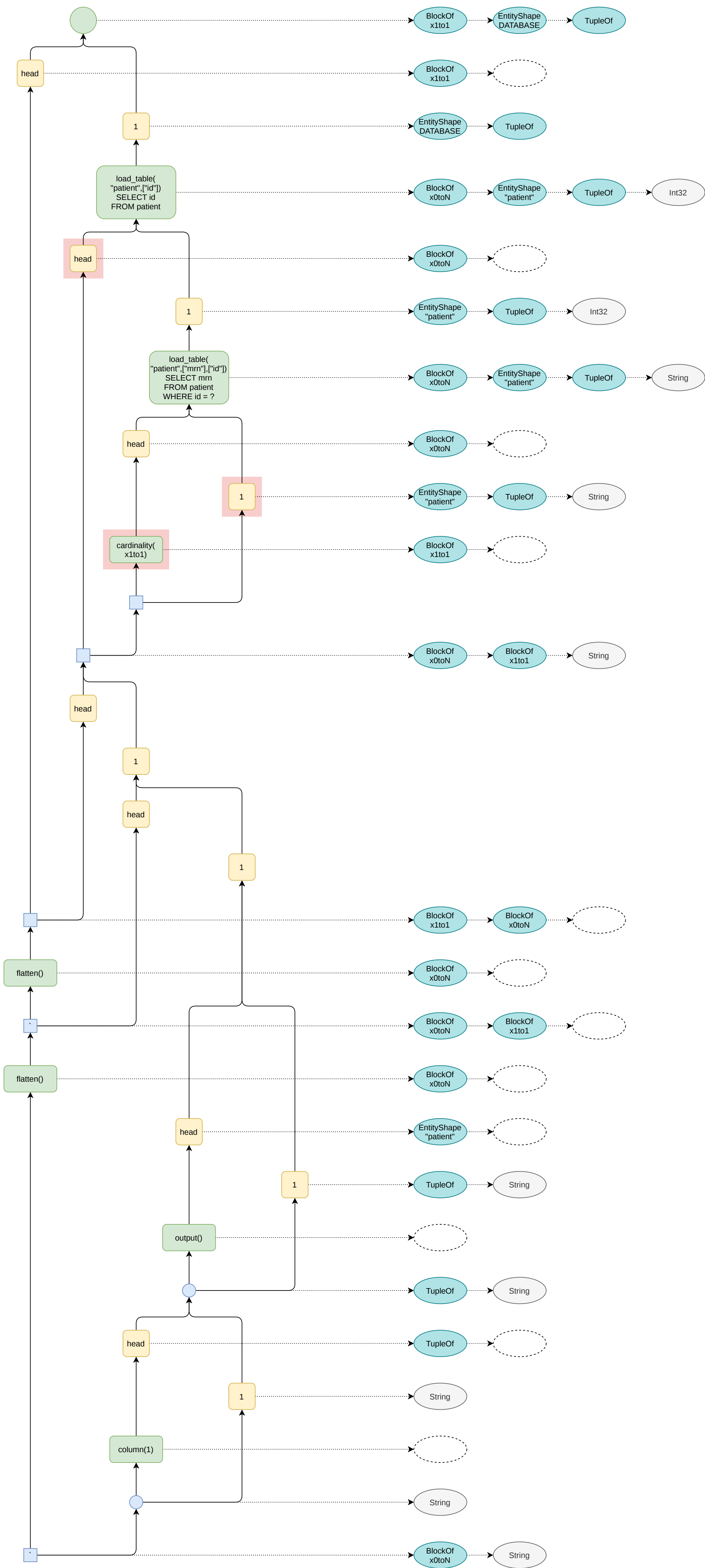
Data Types and Shapes (Right Side):

- `BlockOf x1to1` (Teal oval)
- `EntityShape DATABASE` (Teal oval)
- `TupleOf` (Teal oval)
- `BlockOf x1to1` (Teal oval)
- `EntityShape DATABASE` (Teal oval)
- `TupleOf` (Teal oval)
- `BlockOf x0toN` (Teal oval)
- `EntityShape 'patient'` (Teal oval)
- `TupleOf` (Teal oval)
- `Int32` (Grey oval)
- `BlockOf x0toN` (Teal oval)
- `EntityShape 'patient'` (Teal oval)
- `TupleOf` (Teal oval)
- `Int32` (Grey oval)
- `BlockOf x0toN` (Teal oval)
- `EntityShape 'patient'` (Teal oval)
- `TupleOf` (Teal oval)
- `String` (Grey oval)
- `BlockOf x0toN` (Teal oval)
- `EntityShape 'patient'` (Teal oval)
- `TupleOf` (Teal oval)
- `String` (Grey oval)
- `BlockOf x1to1` (Teal oval)
- `BlockOf x0toN` (Teal oval)
- `BlockOf x0toN` (Teal oval)
- `BlockOf x1to1` (Teal oval)
- `BlockOf x0toN` (Teal oval)
- `BlockOf x0toN` (Teal oval)
- `EntityShape 'patient'` (Teal oval)
- `TupleOf` (Teal oval)
- `String` (Grey oval)
- `BlockOf x0toN` (Teal oval)
- `String` (Grey oval)
- `BlockOf x0toN` (Teal oval)
- `String` (Grey oval)

Execution Flow (Center):

- The query plan starts with `load_table('patient',['id']) SELECT id FROM patient`, which outputs a `BlockOf x1to1`.
- The `head` operator takes the output of `load_table` and outputs a `BlockOf x0toN`.
- The `1` operator takes the output of `head` and outputs a `BlockOf x0toN`.
- The `flatten()` operator takes the output of `1` and outputs a `BlockOf x0toN`.
- The `cardinality(x1to1)` operator takes the output of `flatten()` and outputs a `BlockOf x0toN`.
- The `output()` operator takes the output of `cardinality` and outputs a `BlockOf x0toN`.
- The `column(1)` operator takes the output of `output()` and outputs a `BlockOf x0toN`.
- The `head` operator takes the output of `column` and outputs a `BlockOf x0toN`.
- The `1` operator takes the output of `head` and outputs a `BlockOf x0toN`.







```
chain_of(with_elements(load_postgres_table(("public", "patient"), ["id"], [Int32])),
  flatten(),
  with_elements(
    chain_of(
      load_postgres_table(("public", "patient"), ["mrn"], [String], ["id"]),
      block_cardinality(x1to1))),
  flatten(),
  with_elements(
    chain_of(
      output(),
      column(1))))
```



