1 Data Definition Language (DDL)

1.1 Create Schema

⟨*schema_statement*⟩ ::= CREATE SCHEMA ⟨*schema_ident*⟩
(⟨*named_schema_ident*⟩|⟨*anonymous_schema*⟩)
[EXTENDS ⟨*parent_schema_ident*⟩]

⟨*anonymous_schema*⟩ ::= '(' typedField {',' typedField}')'

⟨*typedField*⟩ ::= ⟨*field_ident*⟩ ⟨*data_type*⟩

1.2 Create Stream

⟨*Stream statement*⟩ ::= CREATE STREAM ⟨*schema_ident*⟩
(⟨*named_schema_ident*⟩|⟨*anonymous_schema*⟩)
[⟨*source*⟩] sub ⟨*source*⟩ ::= (AS ⟨*derived_source*⟩) |
(SOURCE ⟨*raw_source*⟩)

⟨*derived_source*⟩ ::= ⟨*stream_ident*⟩| ⟨*subSelect*⟩

⟨*raw_source*⟩ ::= HOST '('⟨*host*⟩, ⟨*port*⟩')'
| FILE '('⟨*file path*⟩, ⟨*delimiter*⟩')'

2 Data Manipulation Language (DML)

2.1 Select

⟨*select_statement*⟩ ::= SELECT ⟨*target_entry*⟩ {, ⟨*target_entry*⟩}
FROM ⟨*stream_references*⟩
WHERE ⟨*predicate*⟩
GROUP BY ⟨*field_ident*⟩ {,⟨*field_ident*⟩}
INTO ⟨*stream_ident*⟩

⟨*stream_references*⟩ ::= ⟨*stream_refererence*⟩ [⟨*join_clause*⟩]

⟨*stream_reference*⟩ ::= (⟨*stream_ident*⟩| ⟨*subSelect*⟩) ['['Window_specification']']

⟨*join_clause*⟩ ::= CROSS JOIN ⟨*stream_reference*⟩
| [INNER] JOIN ⟨*stream_reference*⟩

⟨*window_specification*⟩ ::= SIZE ⟨*spec*⟩
[EVERY ⟨*spec*⟩]
[PARTITIONED BY ⟨*field_ident*⟩ {,⟨*field_ident*⟩}]

⟨*spec*⟩ ::= ⟨*int*⟩ ON ⟨*field_ident*⟩
| ⟨*int*⟩ ⟨*time_unit*⟩
| ⟨*int*⟩

2.2 Insert

⟨*insert_statement*⟩ ::= INSERT INTO ⟨*stream_ident*⟩ [AS] (⟨*stream_ident*⟩
| ⟨*subSelect*⟩| ⟨*merge*⟩)

2.3 Merge

⟨*merge_statement*⟩ ::= MERGE ⟨*stream_ident*⟩ ',' ⟨*stream_ident*⟩ ,⟨*stream_ident*⟩

2.4 Split

⟨*split_statement*⟩ ::= ON ⟨*stream_ident*⟩
⟨*insert clause*⟩ {, ⟨*insert clause*⟩}

⟨*insert_clause*⟩ ::= INSERT INTO ⟨*stream_ident*⟩
SELECT ⟨*target_entry_list*⟩ WHERE ⟨*predicate*⟩