Nebula Graph Diagnostic Report

Generated at (UTC): 2025-11-20T02:13:02.607436Z
Target: 127.0.0.1:9669 | User: root | Test space: agent_registry_diag

1. Connection

Connection pool: OK

2. Diagnostic Steps

Queries succeeded: 2/12

Step 1: Cluster hosts status (meta + storage)

Status: OK | Latency: <bound method ResultSet.latency of ResultSet(keys: ['Host', 'Port', 'Status', 'Leader count', 'Leader distribution', 'Partition distribution', 'Version'], values: ["127.0.0.1", 9779, "OFFLINE", 0, "No valid partition", "No valid partition", __EMPTY__])> ms

Query:

```
  SHOW HOSTS;
```

Rows (up to first 20):

```
  Host | Port | Status | Leader count | Leader distribution | Partition distribution | Version
  Value(
    sVal=b'127.0.0.1') | Value(
    iVal=9779) | Value(
    sVal=b'OFFLINE') | Value(
    iVal=0) | Value(
    sVal=b'No valid partition') | Value(
    sVal=b'No valid partition') | Value()
```

Step 2: List existing spaces

Status: OK | Latency: <bound method ResultSet.latency of ResultSet(keys: ['Name'], values:)> ms

Query:

```
  SHOW SPACES;
```

Step 3: Create test space `agent_registry_diag` if not exists

Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms

Query:

```
  CREATE SPACE IF NOT EXISTS agent_registry_diag(partition_num=16, replica_factor=1, vid_type=FI
XED_STRING(64));
```

Error:

```
  Host not enough!
```

Step 4: Switch to space `agent_registry_diag`

Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms

Query:

```
  USE agent_registry_diag;
```

Error:

```
  SpaceNotFound: SpaceName `agent_registry_diag`
```

Step 5: Create TAG `Person`

Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms

Query:

```
  CREATE TAG IF NOT EXISTS Person(name string, age int);
```

Error:

```
  SemanticError: Space was not chosen.
```

Step 6: Create EDGE `Knows`

Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms

Query:

```
  CREATE EDGE IF NOT EXISTS Knows(since int);
```

Error:

```
  SemanticError: Space was not chosen.
```

Step 7: List tags in test space

```
Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms
Query:
  SHOW TAGS;
Error:
  SemanticError: Space was not chosen.
```

Step 8: List edges in test space

```
Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms
Query:
  SHOW EDGES;
Error:
  SemanticError: Space was not chosen.
```

Step 9: Insert sample Person vertices (alice, bob)

```
Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms
Query:
  INSERT VERTEX Person(name, age) VALUES "alice":("Alice", 30), "bob":("Bob", 28);
Error:
  SemanticError: Space was not chosen.
```

Step 10: Insert sample Knows edge (alice -> bob)

```
Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms
Query:
  INSERT EDGE Knows(since) VALUES "alice"->"bob":(2020);
Error:
  SemanticError: Space was not chosen.
```

Step 11: MATCH query over Person-Knows-Person

```
Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms
Query:
  MATCH (a:Person)-[:Knows]->(b:Person) RETURN a.name, a.age, b.name, b.age, edge.since;
Error:
  SyntaxError: please add alias when using `edge'. near `edge'
```

Step 12: Basic stats overview (if enabled for this version)

```
Status: FAILED | Latency: <bound method ResultSet.latency of ResultSet(None)> ms
Query:
  SHOW STATS;
Error:
  SemanticError: Space was not chosen.
```