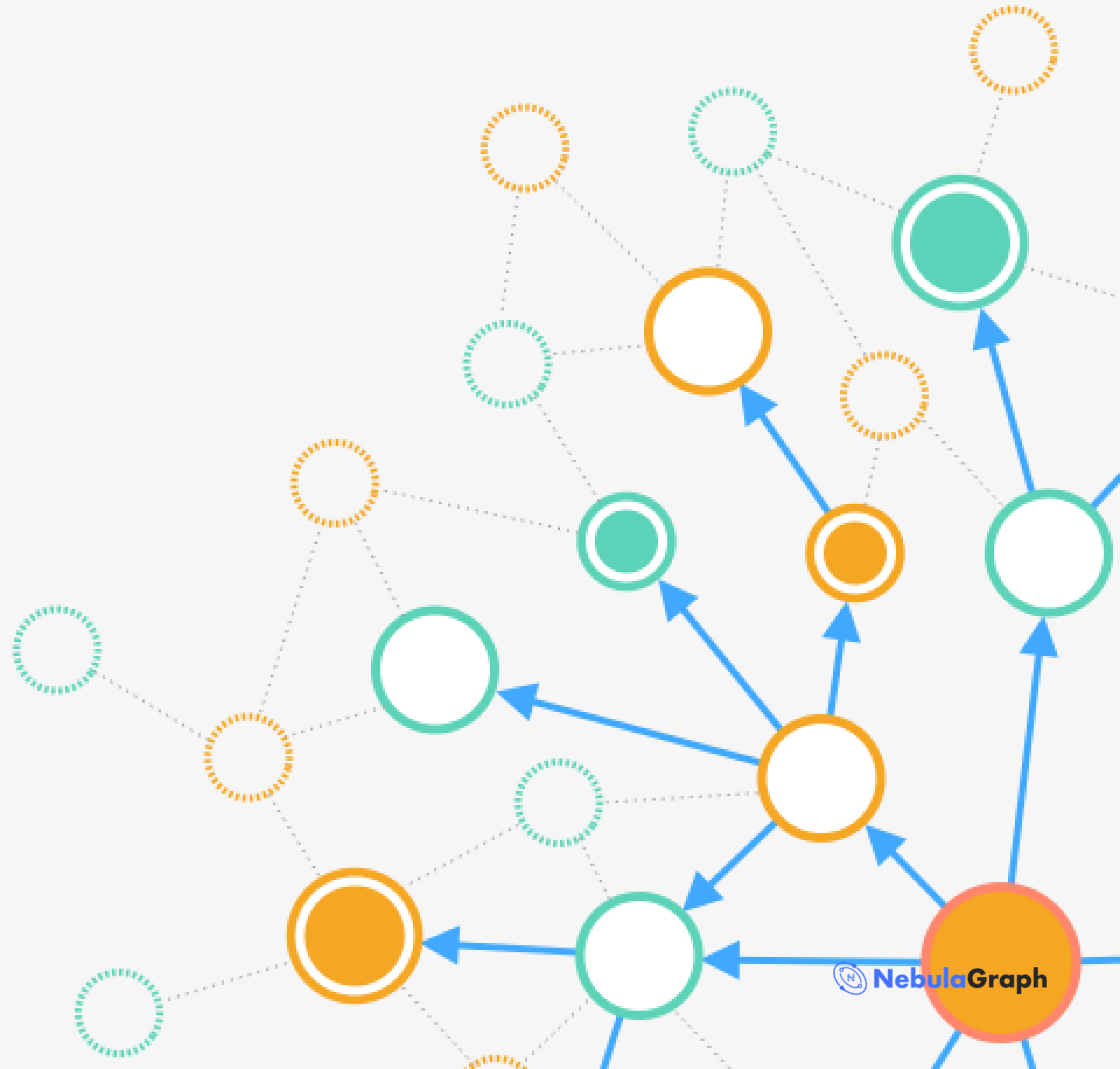




Nebula Graph 3.0 Preview

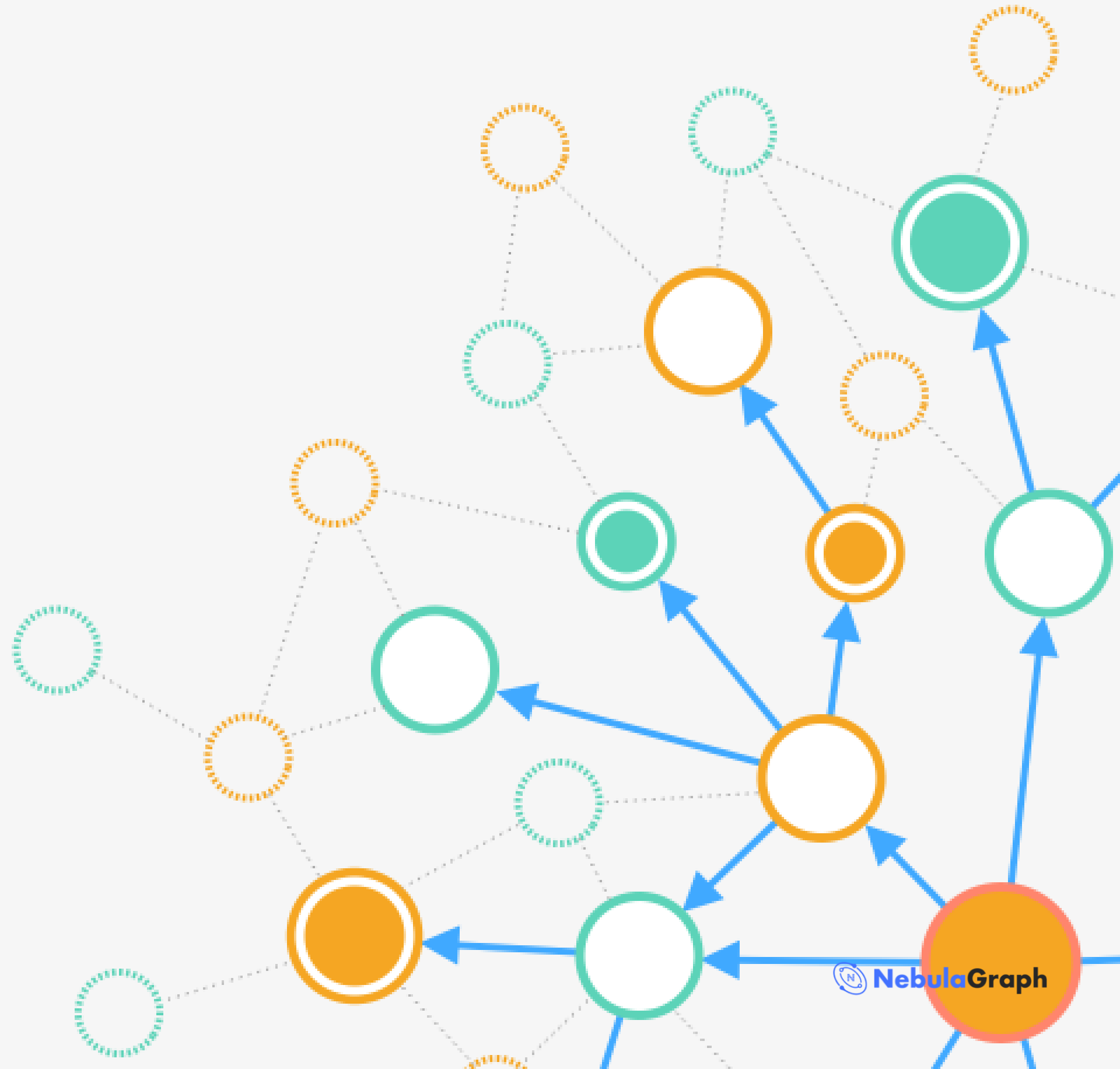
Overview

- Features
- Enhancements
- Bug Fixes
- Upgrade Path
- Surrounding Projects Highlight
- Incompatible Changes



Features

- OpenCypher: Multi-Pattern `MATCH`
- OpenCypher: Parameter
- Bird's-eye view of data w/o index
- Bare vertex support(vertex w/o Tag)
- Backup and Restore
- Others
 - Describe USER
 - UTF-8 supports in Schema
 - `datediff()` and `dateadd()`



```
MATCH (db:Database)-[:CLUSTER]->(clstr:Cluster)-[:SCHEMA]->(schema:Schema)-[:TABLE]->(tbl:Table)-[:COLUMN]->(col:Column)
WHERE id(tbl) == "hive://gold.test_schema/test_table1" \
OPTIONAL MATCH (tbl)-[:DESCRIPTION]->(tbl_dscrpt:Description) \
OPTIONAL MATCH (col:Column)-[:DESCRIPTION]->(col_dscrpt:Description) \
OPTIONAL MATCH (col:Column)-[:STAT]->(stat:Stat) \
OPTIONAL MATCH (col:Column)-[:HAS_BADGE]->(badge:Badge) \
RETURN db, clstr, schema, tbl, tbl_dscrpt, col, col_dscrpt, col.Column.sort_order as sort_order, collect(distinct stat) as stat_badges,
collect(distinct badge) as col_badges \
ORDER BY sort_order;
```

```
+-----+-----+
| db                                           | cT |
+-----+-----+
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
+-----+-----+
Got 5 rows (time spent 10563/33805 us)
```



```
MATCH (db:Database)-[:CLUSTER]->(clstr:Cluster)-[:SCHEMA]->(schema:Schema)-[:TABLE]->(tbl:Table)-[:COLUMN]->(col:Column)
WHERE id(tbl) == "hive://gold.test_schema/test_table1" \
OPTIONAL MATCH (tbl)-[:DESCRIPTION]->(tbl_dscrpt:Description) \
OPTIONAL MATCH (col:Column)-[:DESCRIPTION]->(col_dscrpt:Description) \
OPTIONAL MATCH (col:Column)-[:STAT]->(stat:Stat) \
OPTIONAL MATCH (col:Column)-[:HAS_BADGE]->(badge:Badge) \
RETURN db, clstr, schema, tbl, tbl_dscrpt, col, col_dscrpt, col.Column.sort_order as sort_order, collect(distinct stat) as stat_badges,
collect(distinct badge) as col_badges \
ORDER BY sort_order;
```

```
+-----+-----+
| db                                           | c1 |
+-----+-----+
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
| ("database://hive" :Database{name: "hive", published: "unique_tag", publisher_last_updated_epoch_ms: 1640405813}) | ( |
+-----+-----+
Got 5 rows (time spent 10563/33805 us)
```

OpenCypher: Parameter

nebula-python:/tests/test_parameter.py

#3379

Feature: Parameter

Background:

Given a graph with space named "basketballplayer"

Given parameters:

`{"p1":1,"p2":true,"p3":"Tim Duncan"}`

Scenario: cypher with parameters

When executing query:

"""

```
MATCH (v:player)-[:like]->(n) WHERE id(v)=$p3 and n.  
RETURN n.player.name AS dst LIMIT $p1+1
```

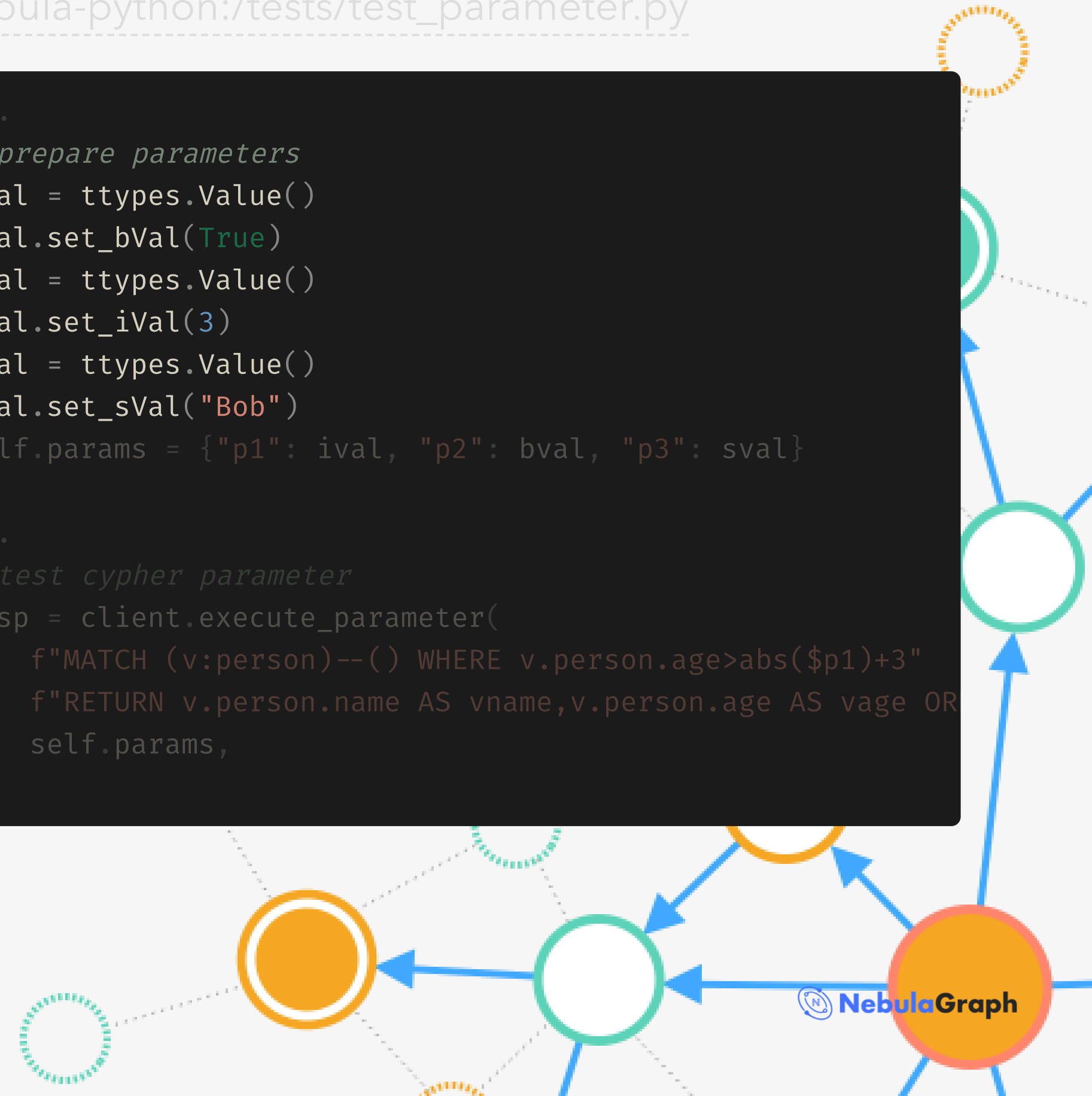
"""

Then the result should be, in any order:

	dst	
	"Manu Ginobili"	
	"Tony Parker"	

```
...  
# prepare parameters  
bval = ttypes.Value()  
bval.set_bVal(True)  
ival = ttypes.Value()  
ival.set_iVal(3)  
sval = ttypes.Value()  
sval.set_sVal("Bob")  
self.params = {"p1": ival, "p2": bval, "p3": sval}  
  
...  
# test cypher parameter  
resp = client.execute_parameter(  
    f"MATCH (v:person)--() WHERE v.person.age>abs($p1)+3"  
    f"RETURN v.person.name AS vname,v.person.age AS vage OR  
    self.params,  
)
```

tests/tck/features/yield/parameter.feature



OpenCypher: Parameter

nebula-python:/tests/test_parameter.py

#3379

Feature: Parameter

Background:

Given a graph with space named "basketballplayer"

Given parameters:

```
{"p1":1,"p2":true,"p3":"Tim Duncan"}
```

Scenario: cypher with parameters

When executing query:

```
"""
```

```
MATCH (v:player)-[:like]->(n) WHERE id(v)=$p3 and n.
RETURN n.player.name AS dst LIMIT $p1+1
"""
```

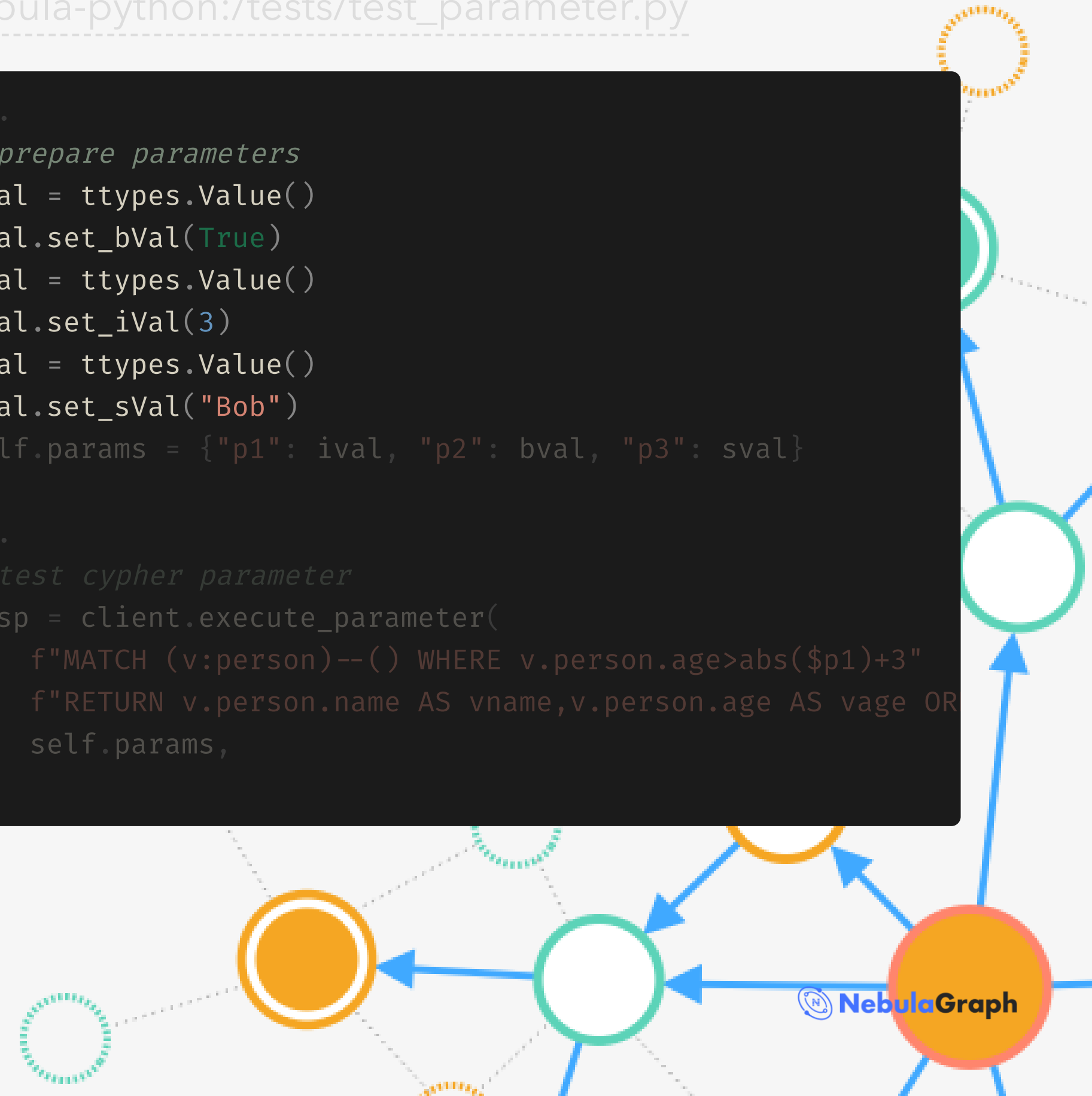
Then the result should be, in any order:

	dst	
	"Manu Ginobili"	
	"Tony Parker"	

```
...
# prepare parameters
bval = ttypes.Value()
bval.set_bVal(True)
ival = ttypes.Value()
ival.set_iVal(3)
sval = ttypes.Value()
sval.set_sVal("Bob")
self.params = {"p1": ival, "p2": bval, "p3": sval}

...
# test cypher parameter
resp = client.execute_parameter(
    f"MATCH (v:person)--() WHERE v.person.age>abs($p1)+3"
    f"RETURN v.person.name AS vname,v.person.age AS vage OR"
    self.params,
)
```

tests/tck/features/yield/parameter.feature



OpenCypher: Parameter

[nebula-python:/tests/test_parameter.py](#)

[#3379](#)

Feature: Parameter

Background:

Given a graph with space named "basketballplayer"

Given parameters:

```
{"p1":1,"p2":true,"p3":"Tim Duncan"}
```

Scenario: cypher with parameters

When executing query:

```
"""
```

```
MATCH (v:player)-[:like]->(n) WHERE id(v)=$p3 and n.
RETURN n.player.name AS dst LIMIT $p1+1
"""
```

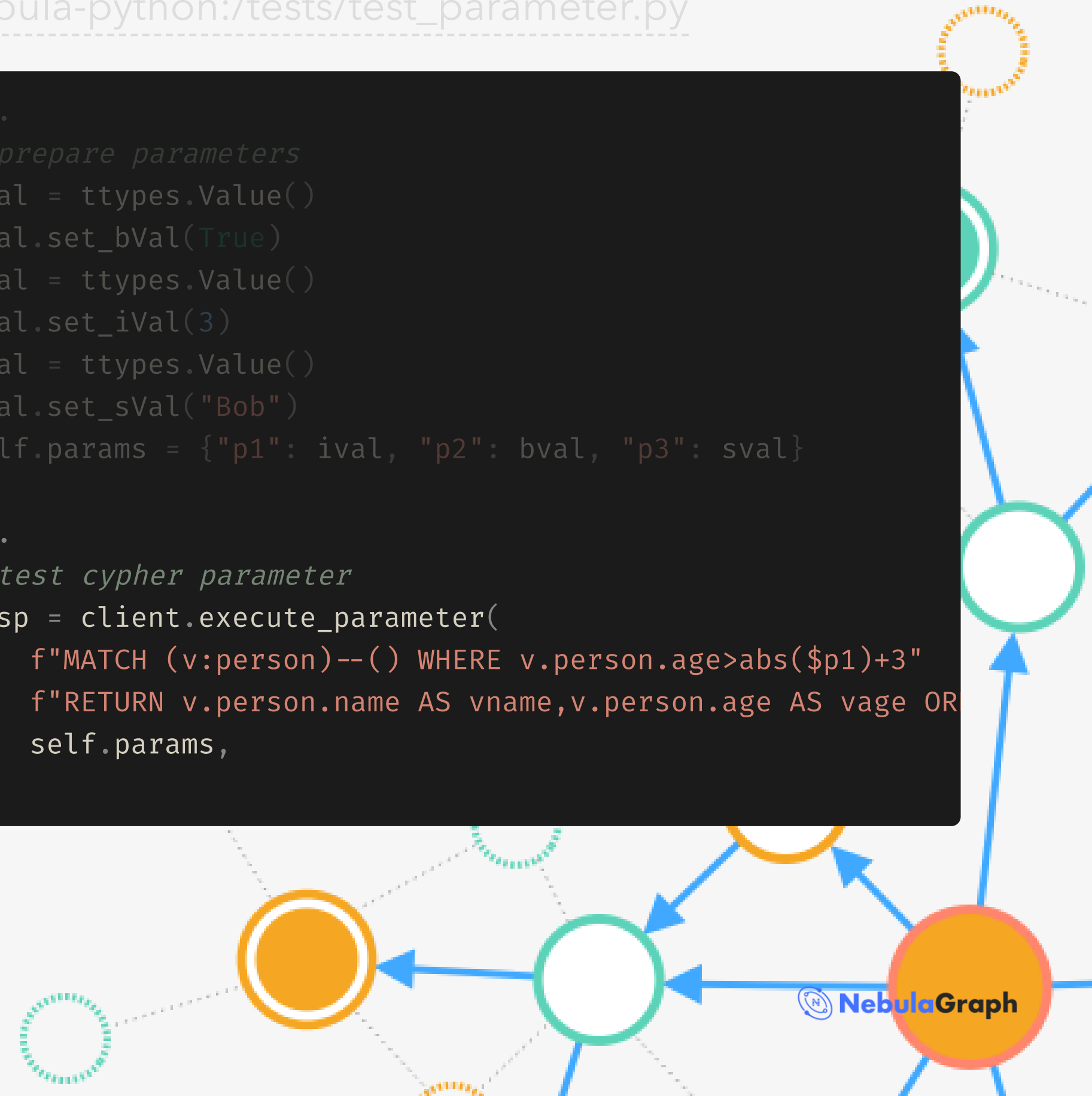
Then the result should be, in any order:

	dst	
	"Manu Ginobili"	
	"Tony Parker"	

```
...
# prepare parameters
bval = ttypes.Value()
bval.set_bVal(True)
ival = ttypes.Value()
ival.set_iVal(3)
sval = ttypes.Value()
sval.set_sVal("Bob")
self.params = {"p1": ival, "p2": bval, "p3": sval}

...
# test cypher parameter
resp = client.execute_parameter(
    f"MATCH (v:person)--() WHERE v.person.age>abs($p1)+3"
    f"RETURN v.person.name AS vname,v.person.age AS vage OR"
    self.params,
)
```

[tests/tck/features/yield/parameter.feature](#)



Bird's-eye view w/o index

#3329, #3320, #3507, #3262

Example: `MATCH` in **LIMIT** filter W/O INDEX

```
> SHOW TAG INDEXES
+-----+-----+-----+
| Index Name      | By Tag | Columns |
+-----+-----+-----+
| "player_index_0" | "player" | []      |
| "player_index_1" | "player" | ["name"] |
| "player_index_2" | "player" | ["name", "age"] |
+-----+-----+-----+

> MATCH (m:team) RETURN m
[ERROR (-1005)]: Scan vertices or edges need to specify a limit

> MATCH (m:team) RETURN m LIMIT 2
+-----+
| m |
+-----+
| ("team209" :team{name: "Timberwolves"}) |
| ("team225" :team{name: "Bucks"}) |
+-----+
```

```
(root@nebula) [basketballplayer]> EXPLAIN MATCH (m:team) RETURN m
Execution Plan (optimize time 110 us)

+-----+-----+-----+
| 17 | ScanVertices | 3 | outputVar: [
|   |               |   | {
|   |               |   |   "colNames": [
|   |               |   |     "_vid",
|   |               |   |     "team._tag"
|   |               |   |   ],
|   |               |   |   "type": "DATASET",
|   |               |   |   "name": "__Filter_2"
|   |               |   | }
|   |               |   | inputVar:
|   |               |   | space: 1
|   |               |   | dedup: false
|   |               |   | limit: 2
|   |               |   | filter: (team._tag IS NOT EM
|   |               |   | orderBy: []
|   |               |   | props: [
|   |               |   | {
|   |               |   |   "props": [
|   |               |   |     "_tag"
|   |               |   |   ],
|   |               |   |   "tagId": 3
|   |               |   | }
|   |               |   | ]
|   |               |   | exprs:
+-----+-----+-----+
```

Bird's-eye view w/o index

#3329, #3320, #3507, #3262

Example: `MATCH` in `LIMIT` filter W/O INDEX

```
> SHOW TAG INDEXES
```

Index Name	By Tag	Columns
"player_index_0"	"player"	[]
"player_index_1"	"player"	["name"]
"player_index_2"	"player"	["name", "age"]

```
> MATCH (m:team) RETURN m
```

[ERROR (-1005)]: Scan vertices or edges need to specify a limit

```
> MATCH (m:team) RETURN m LIMIT 2
```

m
("team209" :team{name: "Timberwolves"})
("team225" :team{name: "Bucks"})

```
(root@nebula) [basketballplayer]> EXPLAIN MATCH (m:team) RETURN m
```

Execution Plan (optimize time 110 us)

id	operator	cost	outputVar
17	ScanVertices	3	<pre>{ "colNames": ["_vid", "team._tag"], "type": "DATASET", "name": "__Filter_2" }</pre>
			<pre>inputVar: space: 1 dedup: false limit: 2 filter: (team._tag IS NOT EMPTY) orderBy: [] props: [{ "props": ["_tag"], "tagId": 3 }]</pre>
			<pre>exprs:</pre>



Bird's-eye view w/o index

#3329, #3320, #3507, #3262

Example: `MATCH` in `LIMIT` filter W/O INDEX

```
> SHOW TAG INDEXES
+-----+-----+-----+
| Index Name      | By Tag | Columns      |
+-----+-----+-----+
| "player_index_0" | "player" | []           |
| "player_index_1" | "player" | ["name"]     |
| "player_index_2" | "player" | ["name", "age"] |
+-----+-----+-----+

> MATCH (m:team) RETURN m
[ERROR (-1005)]: Scan vertices or edges need to specify a limit

> MATCH (m:team) RETURN m LIMIT 2
+-----+
| m      |
+-----+
| ("team209" :team{name: "Timberwolves"}) |
| ("team225" :team{name: "Bucks"})         |
+-----+
```

```
(root@nebula) [basketballplayer]> EXPLAIN MATCH (m:team) RETURN m
Execution Plan (optimize time 110 us)

+-----+-----+-----+
| 17 | ScanVertices | 3 | outputVar: [
|   |               |   | {
|   |               |   |   "colNames": [
|   |               |   |     "_vid",
|   |               |   |     "team._tag"
|   |               |   |   ],
|   |               |   |   "type": "DATASET",
|   |               |   |   "name": "__Filter_2"
|   |               |   | }
|   |               |   | inputVar:
|   |               |   | space: 1
|   |               |   | dedup: false
|   |               |   | limit: 2
|   |               |   | filter: (team._tag IS NOT EMPTY)
|   |               |   | orderBy: []
|   |               |   | props: [
|   |               |   | {
|   |               |   |   "props": [
|   |               |   |     "_tag"
|   |               |   |   ],
|   |               |   |   "tagId": 3
|   |               |   | }
|   |               |   | ]
|   |               |   | exprs:
+-----+-----+-----+
```

Bird's-eye view w/o index

#3329, #3320, #3507, #3262

Example: `MATCH` in **LIMIT** filter W/O INDEX

```
> SHOW TAG INDEXES
+-----+-----+-----+
| Index Name      | By Tag | Columns |
+-----+-----+-----+
| "player_index_0" | "player" | []      |
| "player_index_1" | "player" | ["name"] |
| "player_index_2" | "player" | ["name", "age"] |
+-----+-----+-----+

> MATCH (m:team) RETURN m
[ERROR (-1005)]: Scan vertices or edges need to specify a limit

> MATCH (m:team) RETURN m LIMIT 2
+-----+
| m |
+-----+
| ("team209" :team{name: "Timberwolves"}) |
| ("team225" :team{name: "Bucks"}) |
+-----+
```

```
(root@nebula) [basketballplayer]> EXPLAIN MATCH (m:team) RETURN m
Execution Plan (optimize time 110 us)

+-----+-----+-----+
| 17 | ScanVertices | 3 | outputVar: [
|   |               |   | {
|   |               |   |   "colNames": [
|   |               |   |     "_vid",
|   |               |   |     "team._tag"
|   |               |   |   ],
|   |               |   |   "type": "DATASET",
|   |               |   |   "name": "__Filter_2"
|   |               |   | }
|   |               |   | inputVar:
|   |               |   | space: 1
|   |               |   | dedup: false
|   |               |   | limit: 2
|   |               |   | filter: (team._tag IS NOT EMPTY)
|   |               |   | orderBy: []
|   |               |   | props: [
|   |               |   | {
|   |               |   |   "props": [
|   |               |   |     "_tag"
|   |               |   |   ],
|   |               |   |   "tagId": 3
|   |               |   | }
|   |               |   | ]
|   |               |   | exprs:
+-----+-----+-----+
```

Bare Vertex

#3316, #3286, #3328, #3335

Example: Insert a vertex w/o TAG.

```
(root@nebula) [basketballplayer]> INSERT VERTEX VALUES "foobar2000":()  
Execution succeeded (time spent 11137/52014 us)
```

Wed, 12 Jan 2022 16:17:49 CST

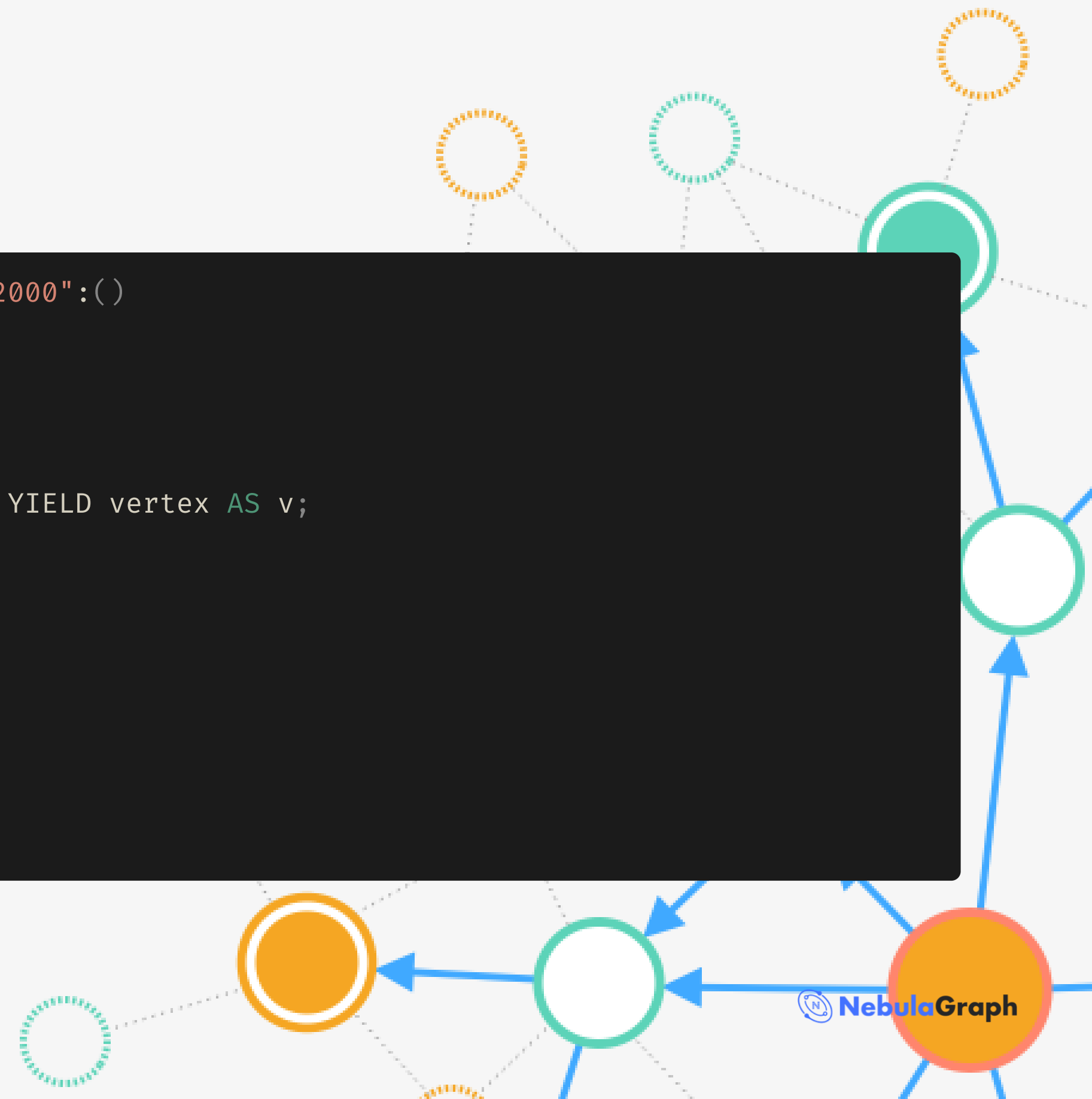
```
(root@nebula) [basketballplayer]> FETCH PROP ON * "foobar2000" YIELD vertex AS v;
```

```
+-----+  
| v      |  
+-----+  
| ("foobar2000") |  
+-----+
```

```
Got 1 rows (time spent 3100/49063 us)
```

Wed, 12 Jan 2022 16:18:19 CST

 [tests/tck/features/insert/insertVertexOnly.feature](#)



Backup and Restore

#3469, #3310, nebula-agent#1, nebula-br#22

```
$ br show \  
  --s3.endpoint "http://192.168.8.214:9000" \  
  --storage="s3://br-test/backup/" \  
  --s3.access_key=minioadmin --s3.secret_key=minioadmin
```

NAME	CREATE TIME	SPACES	FULL BACKUP	ALL SPACES
BACKUP_2021_12_11_14_40_12	2021-12-11 14:40:43	foo	true	true
BACKUP_2021_12_21_12_01_59	2021-12-21 12:01:59	foo	true	false

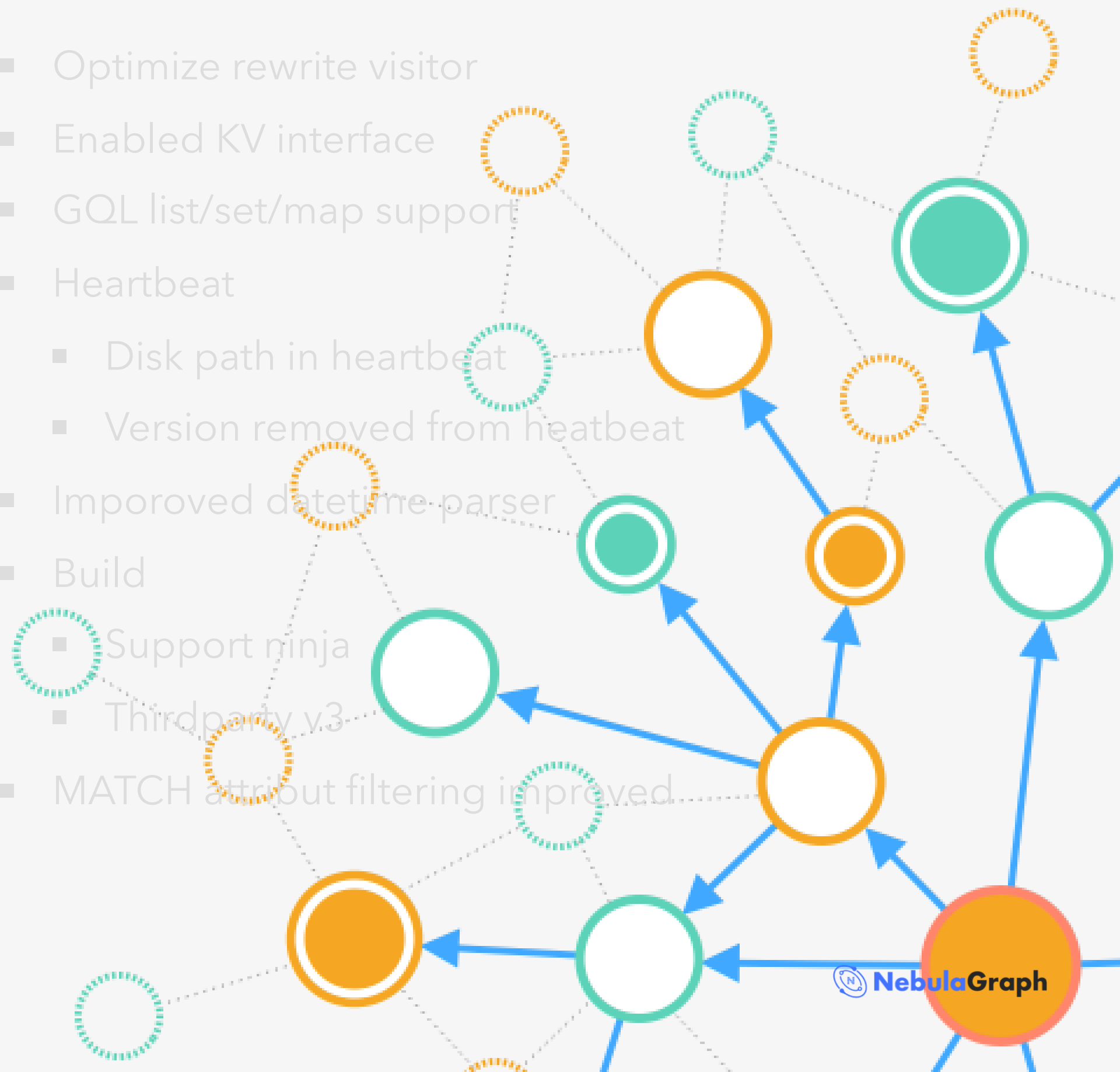
```
$ br restore full \  
  --storage "local:///home/nebula/backup/" \  
  --meta "127.0.0.1:9559" --name BACKUP_2021_12_08_18_38_08
```

Enhancements

- Metrics revisited and improved
- Disable omit of ``YIELD`` clause
- Key Value Separation is supported
- ``BALANCE LEADER`` is now space-specific
- Index Execution Plan Refactorred
- Raft refined
- TLS support of ES client
- Default session timeout optimized
- GEO: Support s2 index params
- ``Properties()`` in ``WHERE`` clause
- ``SHOW LOCAL SESSION``
- Log Rotation

Others

- Optimize rewrite visitor
- Enabled KV interface
- GQL list/set/map support
- Heartbeat
 - Disk path in heartbeat
 - Version removed from heartbeat
- Improved datetime parser
- Build
 - Support ninja
 - Thirdparty v3
- MATCH attribut filtering improved



Metrics added (Graph)

#3446

```
"Graph":{
  "num_queries":"queries count",
  "num_slow_queries":"slow queries count",
  "num_query_errors":"error queries count",
  "query_latency_us":"queries latency",
  "num_active_queries":"active query count",
  "num_sentences":"setences count",
  "num_query_errors_leader_changes":"leader_changes quer
  "num_killed_queries":"killed query count(including kill
  "optimizer_latency_us":"optimizer phase latency",
  "num_aggregate_executors":"aggregate operator excution
  "num_sort_executors":"sort operator excution count",
  "num_indexscan_executors":"indexscan operator excution
  "num_queries_hit_memory_watermark":"hit memory watermar
  "num_opened_sessions":"opened session count",
  "num_auth_failed_sessions":"auth failed sessions count"
  "num_auth_failed_sessions_bad_username_password":"bad c
  "num_auth_failed_sessions_out_of_max_allowed":"exceed a
  "num_active_sessions":"active session count",
  "num_reclaimed_expired_sessions":"reclaimed session cou
  "num_rpc_sent_to_metad":"sent to metad rpc request coun
  "num_rpc_sent_to_metad_failed":"sent to metad rpc reque
  "num_rpc_sent_to_storaged":"sent to storaged rpc reques
```

Metrics added(Meta, Storage)

```
--enable_space_level_metrics`
```

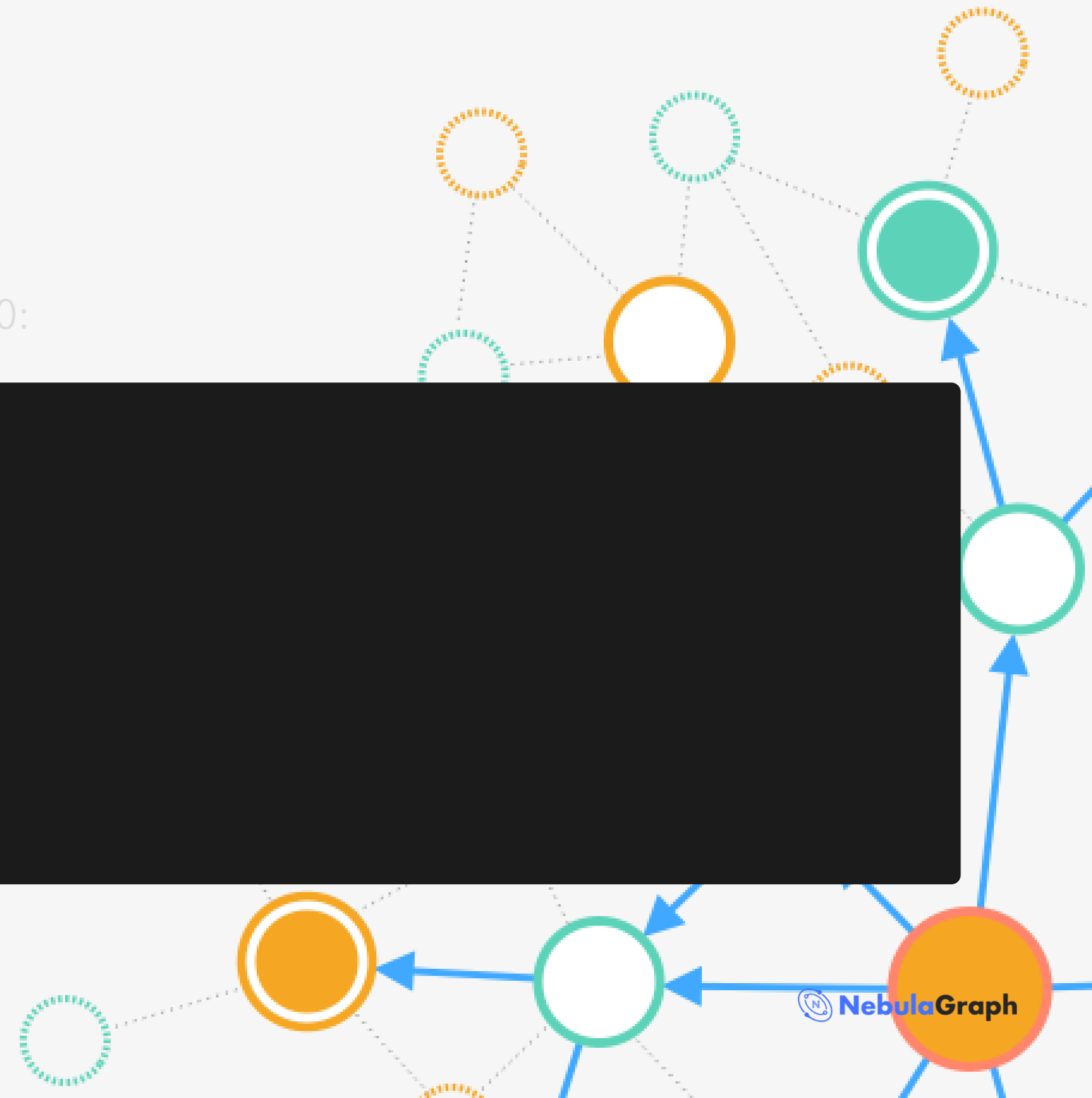
```
"Meta":{
  "commit_log_latency_us":"raft commit log latency",
  "commit_snapshot_latency_us":"raft commit snapshot late
  "transfer_leader_latency_us":"raft transfer leader late
  "num_raft_votes":"raft vote count",
  "num_agent_heartbeats":"AgentHBProcessor heartbeat coun
  "agent_heartbeat_latency_us":"AgentHBProcessor latency"
},
"Storage":{
  "num_edges_inserted":"insert edges count",
  "num_vertices_inserted":"insert vertices count",
  "num_edges_deleted":"deleted edges count",
  "num_tags_deleted":"deleted tagcount",
  "num_vertices_deleted":"deleted vertices count",
  "commit_log_latency_us":"raft commit log letency",
  "commit_snapshot_latency_us":"raft commit snapshot lat
  "transfer_leader_latency_us":"raft transfer leader late
  "num_raft_votes":"raft vote count",
  "num_rpc_sent_to_metad":"sent to metad rpc request coun
  "num_rpc_sent_to_metad_failed":"sent to metad rpc reque
  "kv_get_latency_us":"Getprocessor latency",
  "kv_put_latency_us":"PutProcessor latency",
  "kv_remove_latency_us":"RemoveProcessor latency",
```

Disable omit of YIELD clause

- GO/FETCH YIELD #3056
- FIND PATH YIELD #2957
- LOOKUP/SUBGRAPH YIELD #3139

All `YIELD` Fields should be explicitly specified from 3.0.0:

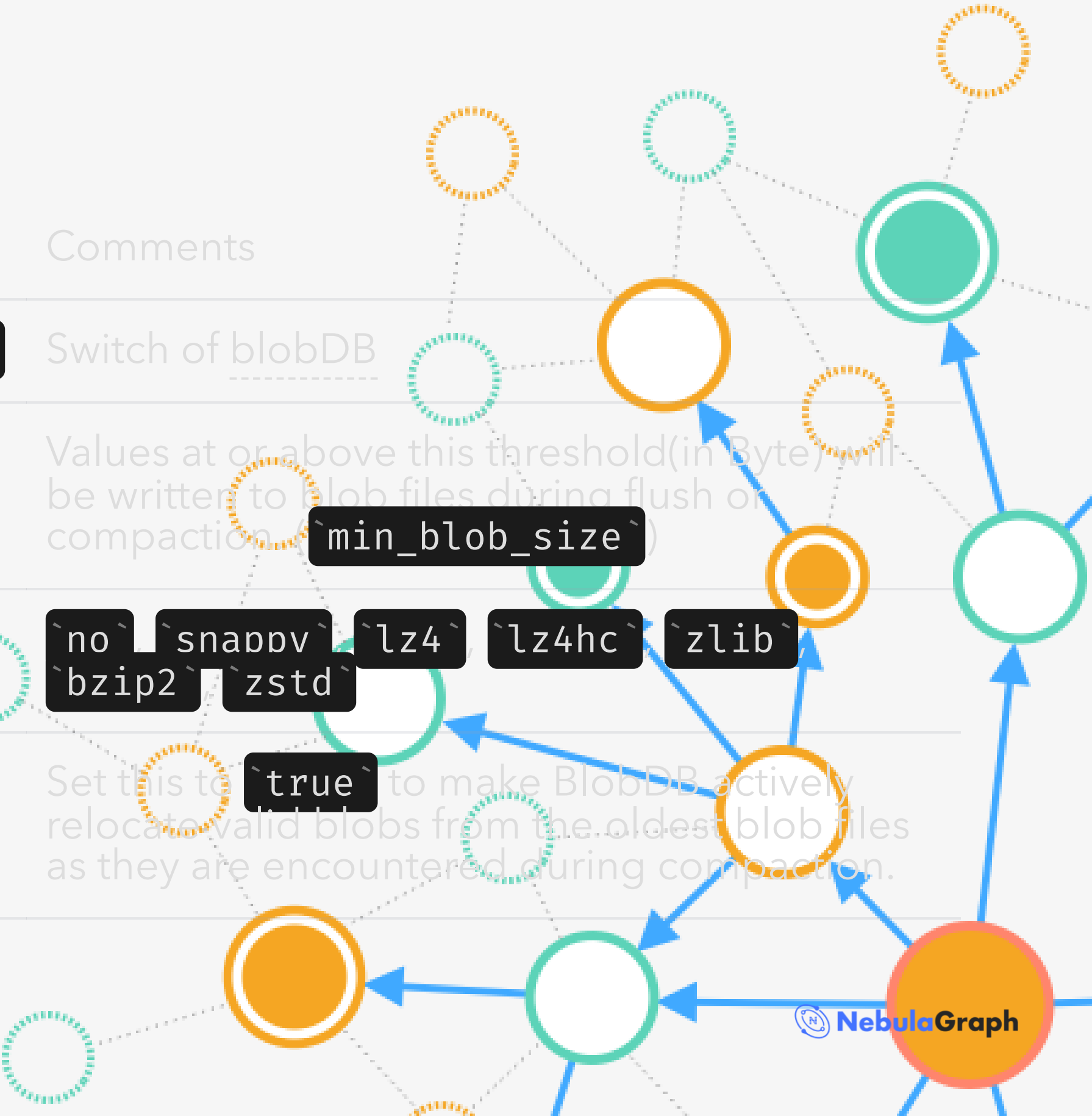
```
> GO FROM "player100" OVER *  
[ERROR (-1009)]: SemanticError: Missing yield clause.  
  
> GO FROM "player100" OVER * YIELD properties($$)  
+-----+  
| properties($$) |  
+-----+  
| {name: "Spurs"} |  
| {age: 36, name: "Tony Parker"} |  
| {age: 41, name: "Manu Ginobili"} |  
+-----+
```



Key Value Seperation in Storage

#3281

Conf	Default	Comments
<code>`rocksdb_enable_kv_separation`</code>	<code>`false`</code>	Switch of blobDB
<code>`rocksdb_kv_separation_threshold`</code>	<code>`100`</code>	Values at or above this threshold(in Byte) will be written to blob files during flush or compaction (<code>`min_blob_size`</code>)
<code>`rocksdb_blob_compression`</code>	<code>`lz4`</code>	<code>`no`</code> <code>`snappy`</code> <code>`lz4`</code> <code>`lz4hc`</code> <code>`zlib`</code> <code>`bzip2`</code> <code>`zstd`</code>
<code>`rocksdb_enable_blob_garbage_collection`</code>	<code>`true`</code>	Set this to <code>`true`</code> to make BlobDB actively relocate valid blobs from the oldest blob files as they are encountered during compaction.



Bugfix

- Fix the deadlock of raft appendlog #3141
- Fix leader storage instance blocked exiting...#3306
- Fix property named "value" to "values" #3303
- Cancel memory check when the ratio > 1.0 #3289
- Fix the problem of index query interval error #3285
- Using int64 to present graph exe latency #2858
- Fix the storage core #3325
- Fix index existence check #3315
- Fix index degeneration #3311
- Fix meta follower read remaining jobs #3347
- Fix can't show edge after clone space #3351
- Report error when use agg in where clause #3355
- Stor may stuck:kvstore ops + shutdown #3358
- Check the result of `listHosts()` #3382

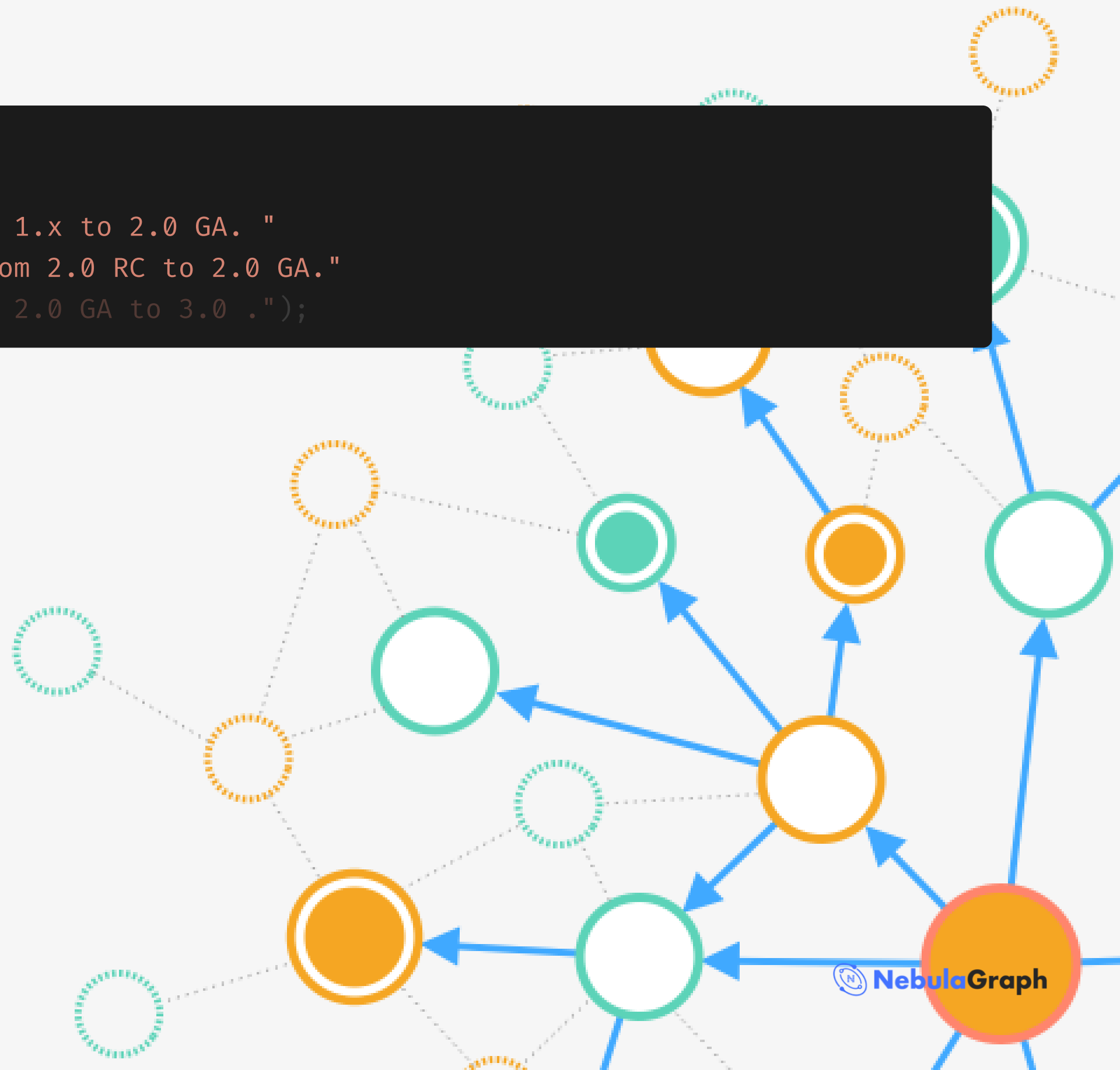
- Fix evaluableExprVisitor #3389
- CheckTypeBeforeDropIndex #3413
- Condition RAM checks cgroup version #3419
- Inconsistent var in ``YIELD`` and ``FROM`` #3430
- Fix latency type in the executionResponse #3438
- Disable agg function in unwind clause #3397
- Signal handler will only call a trigger #3437
- LABEL in "" could be any string #3424
- Cover the ``\n`` in flex #3461
- Fix match index selecting #3457
- Fix geo index bound #3352
- drop space after stop all AdminTaskJobs #3406
- Disable mixed usage of cypher & ngql #3506
- Fix metad hanging under high load #3482
- Crash ``LOOKUP`` filter with ``AND`` and ``IN`` ... #3525
- Fix the bug of RAM info #3534

Upgrade Path

```
`src/tools/db-upgrade/DbUpgrader.cpp`
```

```
DEFINE_string(upgrade_version,  
              "",  
              "When the value is 1:2, upgrade the data from 1.x to 2.0 GA. "  
              "When the value is 2RC:2, upgrade the data from 2.0 RC to 2.0 GA."  
              "When the value is 2:3, upgrade the data from 2.0 GA to 3.0 .");
```

New path: 2.0 GA + to 3.0

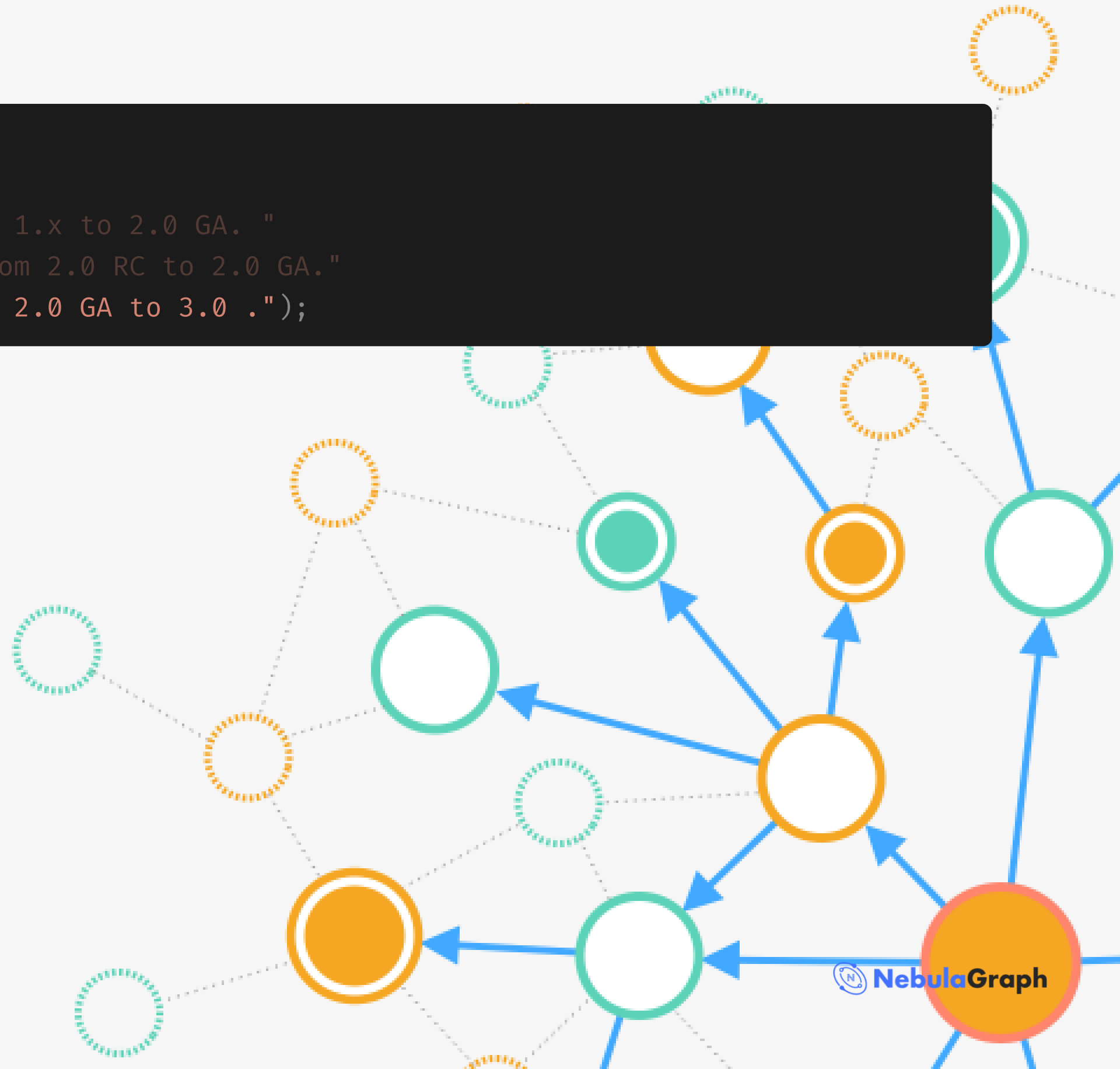


Upgrade Path

```
`src/tools/db-upgrade/DbUpgrader.cpp`
```

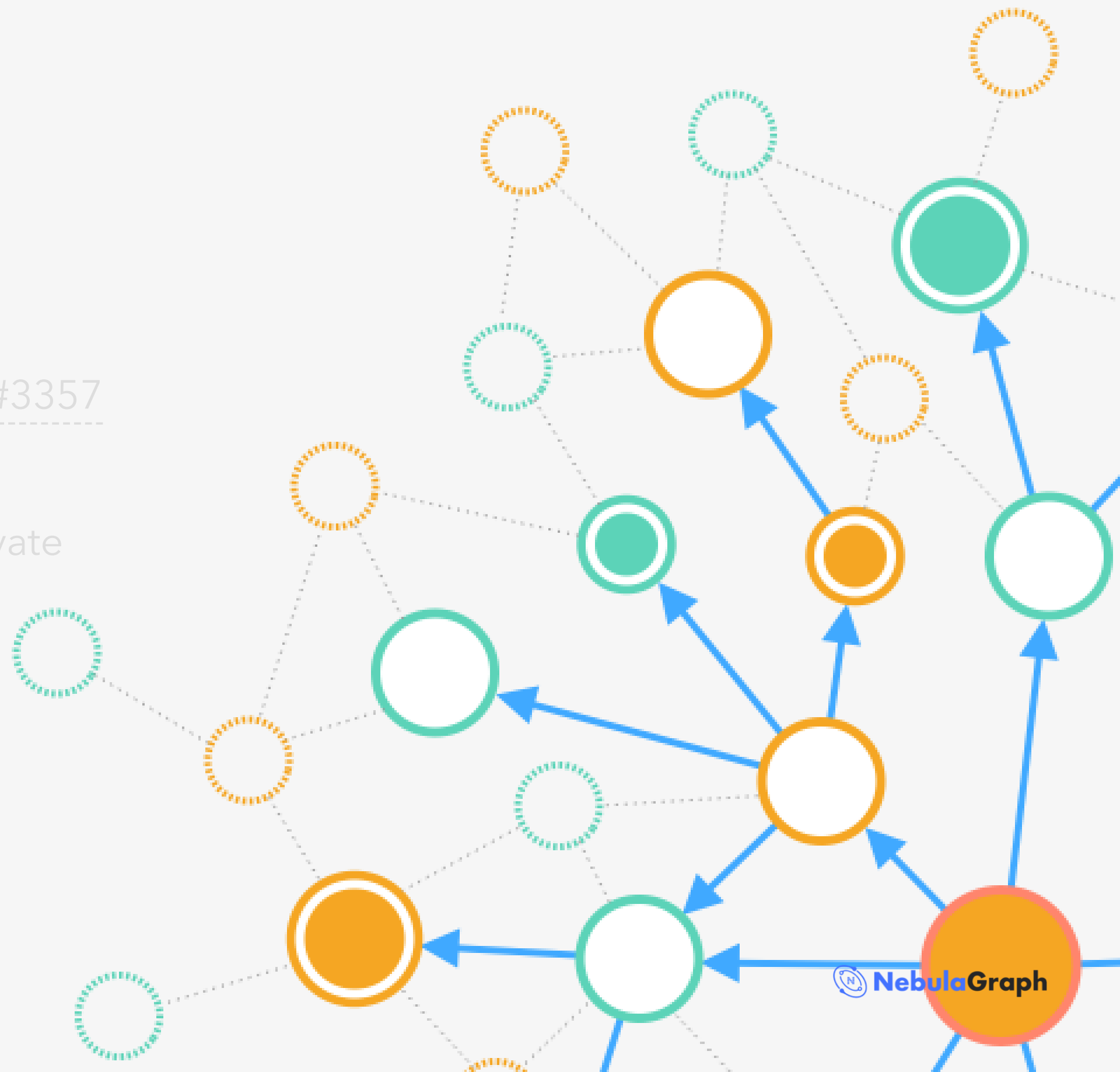
```
DEFINE_string(upgrade_version,  
              "",  
              "When the value is 1:2, upgrade the data from 1.x to 2.0 GA. "  
              "When the value is 2RC:2, upgrade the data from 2.0 RC to 2.0 GA."  
              "When the value is 2:3, upgrade the data from 2.0 GA to 3.0 .");
```

New path: 2.0 GA + to 3.0



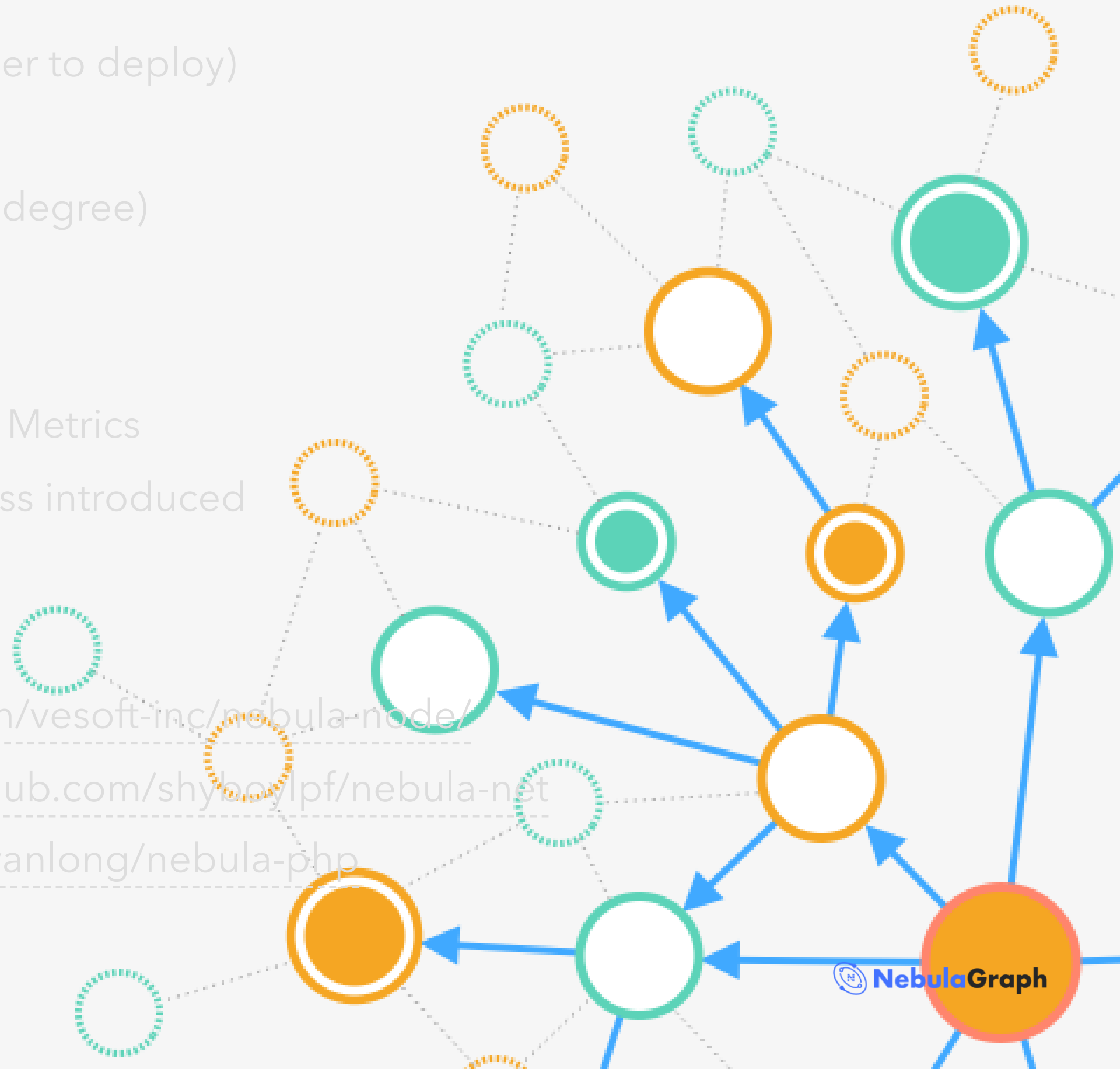
Backward incompatible changes

- YIELD
 - GO/ FETCH #3056
 - FIND PATH #2957
 - LOOKUP/ SUBGRAPH #3139
- ``session_idle_timeout_secs`` in 0 is invalid now #3357
- Storage host discovery mechanism changed:
 - Now requires ``ADD HOSTS <host:port>`` to activate
- ``BALANCE LEADER`` is now space-specified
- ``BALANCE DATA`` was removed



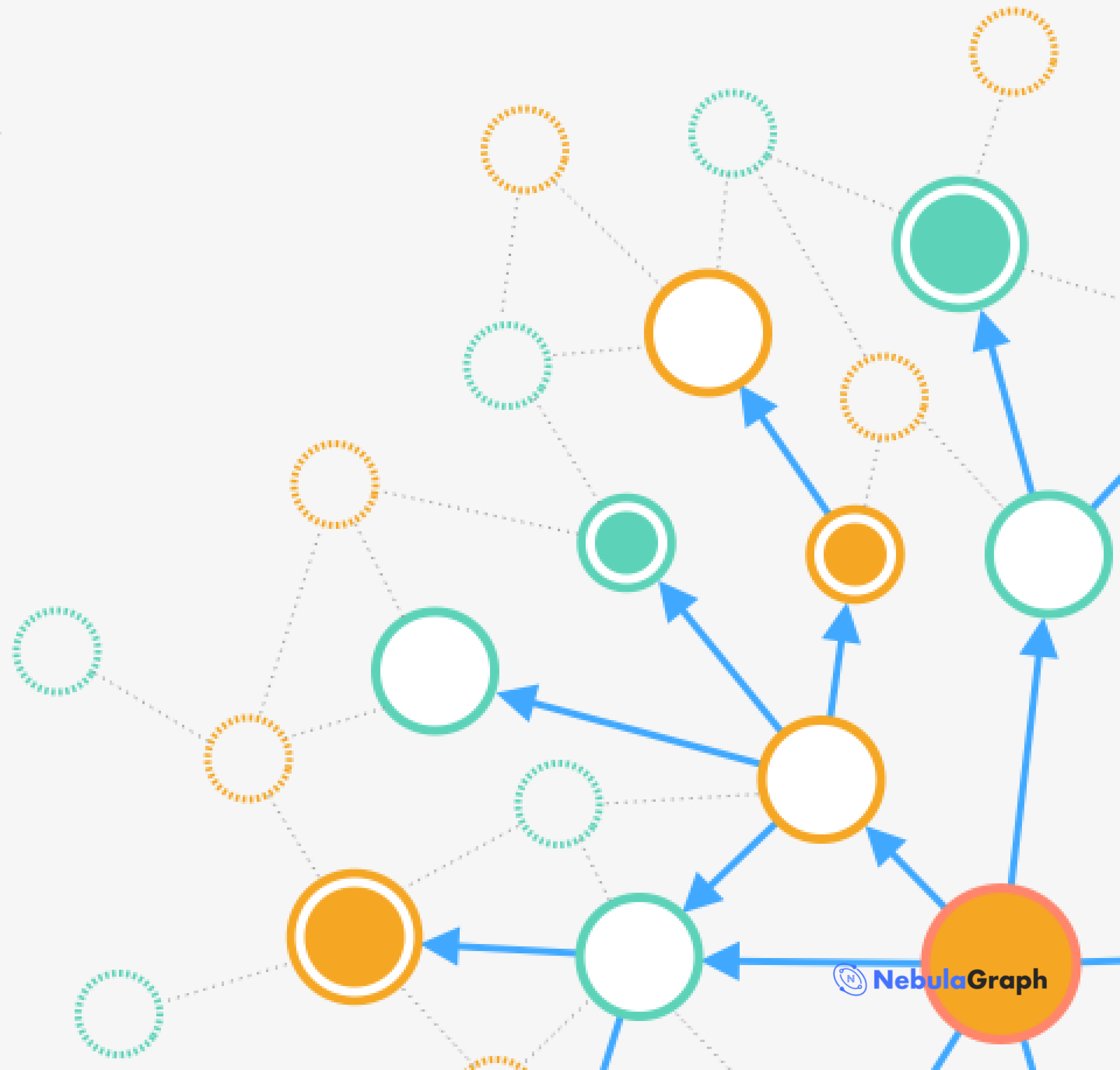
What's New in Surrounding Tooling

- Studio, removed OS level nodejs requirements(easier to deploy)
- Explorer
 - Render size based on multiple factors(tag, prop, degree)
 - Edge color based on source tag
 - More layout(grid, mind graph)
- Dashboard, Alarm System/Task Management/More Metrics
- Algorithm, BFS/Coefficient/HNAP/Jaccard/Closeness introduced
- Exchange, GEO SST support
- Clients:
 - Nodejs contributed by wujpp <https://github.com/vesoft-inc/nebula-node/>
 - .net Client by shyboylpf & heyanlong <https://github.com/shyboylpf/nebula-net>
 - PHP Client by heyanlong <https://github.com/heyanlong/nebula-php>
- Importer supports GEO
- Operator, BR support is WIP




Recap

- Features
 - OpenCypher: Multi-Pattern `MATCH` & Parameter
 - Bird's-eye view of data w/o index
 - Bare vertex support(vertex w/o Tag)
 - Backup and Restore
- Enhancements
 - Metrics revisited and improved
 - Key Value Separation is supported
 - Balance Leader is now space-specified
 - Index Execution Plan Refactorred
 - Raft refined
 - TLS support of ES client
 - Log Rotation



Nebula Community Updates

- Community Meeting
 - t.co/gRL71KlCkH
 - 
- Hackathon 2021 Finished
 - nebula-graph.io/hackathon/
- 2022 Highlights:
 - More MATCH optimizations & GQL Standard Support
 - Nebula Graph-aaS on the Cloud