

ORACLE®

RDF Semantic Graph

New Features for 18.1

October 2, 2017

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

RDF Semantic Graph Roadmap

- Better query performance
 - Composite partitioning for semantic networks
 - In-memory virtual column support
 - Hybrid SPARQL OPTIONAL evaluation strategy
 - Improved optimizer statistics
- Better usability
 - SQL Developer plugin
 - Improved handling of long literals during bulk load
 - More RDF serialization formats supported through SQL API

Composite Partitioning for Semantic Networks

- Semantic networks can be created with list-hash composite partitioning
- RDF quads are list partitioned by model id and then subpartitioned by a hash of the predicate id
- Improved query performance through increased parallelism and better optimizer statistics

Model Partition

RDF_LINK\$ Table

Predicate Subpartition

model_id	subj_id	pred_id	obj_id	graph_id
1	...	101
1	...	101
1	...	103
1	...	103
1	...	105
1	...	105
2	...	101
2	...	101
2	...	107
2	...	107

In-Memory Virtual Column Support

- Lexical values for subject, predicate, object and graph are added to id-only MDSYS.RDF_LINK\$ table as virtual columns and materialized and compressed in memory
- Up to two orders of magnitude increase in SPARQL query performance through in memory access and elimination of joins with MDSYS.RDF_VALUE\$ table
- No on disk storage penalty because values are only materialized in memory

In-Memory Virtual Column Support

Materialized Virtual Columns

In Memory
(columnar)

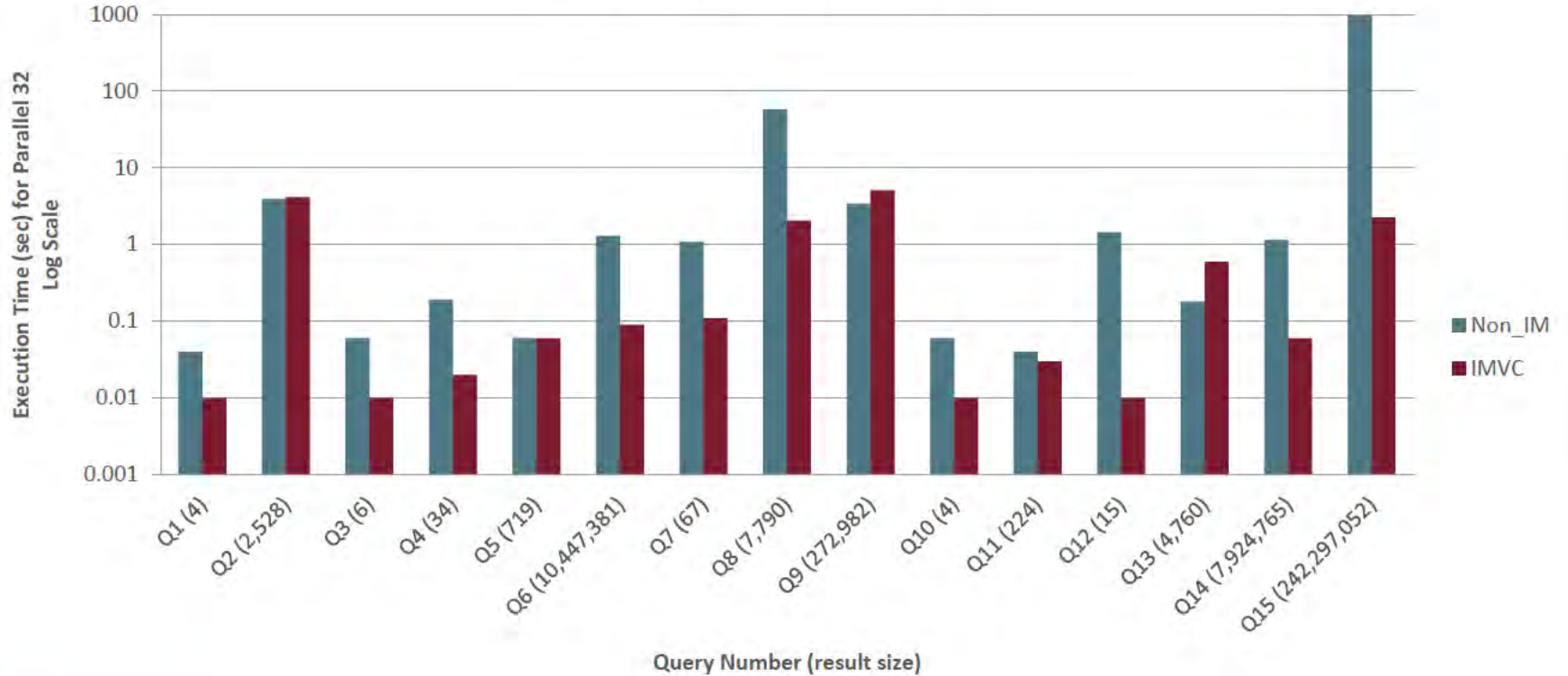
subj_id	subj_lexval	pred_id	pred_lexval	obj_id	obj_lexval	graph_id	graph_lexval
123	<urn:abc>	456	<urn:pred1>	789	"def"	102	<urn:g1>
100	<urn:uvw>	456	<urn:pred1>	101	"xyz"	102	<urn:g1>

On Disk

value_id	lex_val
123	<urn:abc>
456	<urn:pred1>
789	"def"
100	<urn:uvw>
101	"xyz"
102	<urn:g1>

subj_id	pred_id	obj_id	graph_id
123	456	789	102
100	456	101	102

LUBM 1K Benchmark



Enhanced SPARQL Support

- Improved query performance
 - Better optimizer statistics with column groups and larger histograms
 - Hybrid top-down / bottom up evaluation of nested OPTIONALs
 - Compile-time value substitution for BIND
- Better standards support
 - Native support for Turtle and Trig RDF formats in SPARQL LOAD operation

RDF Bulk Load Enhancements

- Long literals (CLOBs) supported in `SEM_APIS.BULK_LOAD_FROM_STAGING_TABLE`
- External table-based load has several new options
 - `CLOB_ONLY / VC_ONLY`: Only load quads with CLOB or VARCHAR objects
 - Allows for a fast 2-stage bulk load when input data is large and contains only a relatively small number of quads with CLOB objects
 - `VIEW`: Create a view with the external table instead of physically copying the data into a staging table to save disk space

Integrated Cloud

Applications & Platform Services

ORACLE®