

accenture[>]technology
Accenture Enkitec Group

Automatic indexing and what else is new in 19c

October 24th, 2019
Helsinki, Finland



ORACLE®
ACE Director

ORACLE®
Certified Master

**ACCENTURE
ENKITEC
GROUP**

- 10.000+ hours of 24x7 on-call DBA
- First Oracle Certified Master in Europe: 2002
- Oracle ACE Director
- Master Technology Architect
- Master Data Architect
- Database Blog at: juliandontcheff.wordpress.com

Oracle Cloud Infrastructure

New Free Tier

oracle.com/gbtour



Always Free

Services you can use for unlimited time

+

30-Day Free Trial

Free credits you can use for more services



Why automating index creation in the database?

- For a very long time, both DBAs and Developers, have been struggling (really struggling) with what indexes should be created, what type of indexes they should be created as and what indexes should be dropped from the database
- By far, the most interesting new feature of Oracle Database 19c is Automatic Index creation (AI Creation)
- In the long run, this is to be one of the most important features in the Oracle database

“ This is the most important thing we have done in a long, long time. The automation does everything. We can guarantee an availability time of 99.995%, less than 30 minutes a year of planned or unplanned downtime.”

Larry Ellison

Oracle Executive Chairman and CTO

ORACLE®



ORACLE

R S I

ORACLE

USERS - GUIDE

Oracle Users Guide - Version 2.3

Copyright (c) April 1981

By Relational Software Incorporated

All rights reserved. Printed in U.S.A.

ORACLE

ORACLE Database Management System

(c) Copyright Oracle Corporation, 1984.

All Rights Reserved.

This software has been provided under a license agreement
containing certain restrictions on use and disclosure.
Reverse engineering of object code is prohibited.

Press Any Key To Continue..._

The following features:

- Basic SQL functionality
- Queries
- Joins
- No transactions supported

came with Oracl(e) 2 (1979).

CIA Advanced Technology Division

“No idea of what they used it for”

– Bruce Scott Author Interview



The following features:

- COMMIT and ROLLBACK
- No more read locks
- Re-written in C
- The first 32-bit RDBMS

came with Oracle 3 (1981).

v3 *“wasn't very reliable ... in the database world
there are couple of things that you can never do...
One is you cant lose data and the other is you cant
return wrong answers”*

- Larry

*“v3 definitely used SQL for data dictionary,
not sure about v2”* – Bruce Scott Author Interview

**What is
on the
picture**



The following features:

- Read consistency
- Desktop PC version (Apple & IBM)
- MS-DOS version runs on only 640K memory

came with Oracle 4 (1984).

UFI =

User Friendly Interface

```
1024 bytes of before image buffers, and
2048 bytes of buffers.
[OR: ORACLE warm started.

C:\ORACLE\DEMO>ufi system\manager

ORACLE Utilities, Copyright (c) 1979, 1980, 1981, 1982, RSI
UFI version 3.5 - on Thu Apr 13 11:25:43 2006

Connecting to: Two Process ORACLE V4.1.1 - Production

UFI> desc sysuserauth

# size csize type                                name
1   22    40  2 numeric                                USERID
2   30     1  1 character                               USERNAME
3   30     1  1 character                               PASSWORD
4    8    75 12 date data type                          TIMESTAMP
5    1     1  1 character                               CONNECTAUTH
6    1     1  1 character                               DBAUTH
7    1     1  1 character                               RESOURCEAUTH
```

“Larry Ellison got most of these intelligence sales”

- Bruce Scott Author Interview

The following features:

- Support for the Client-Server model: PCs can access the DB on a remote host !
- Distributed queries
- Clustering

came with Oracle 5 (1986).

Who is on the picture →



The following features:

- Oracle Parallel Server
- Row-level locking
- On-line database backups
- PL/SQL in the database
- Rollback segments

came with Oracle 6 (1988).

Bruce Scott:

- It is all about Larry
- The thing that made Oracle successful was Larry's charisma, vision and determination to make this thing work no matter what



The following features:

- Advanced Replication
- Read Only tablespaces
- Database Triggers
- View compilation
- CBO
- The Checkpoint process
- Index rebuilds
- Standby Database
- Resizable, autoextend data files
- DBMS_JOB

came with Oracle 7 (1992).

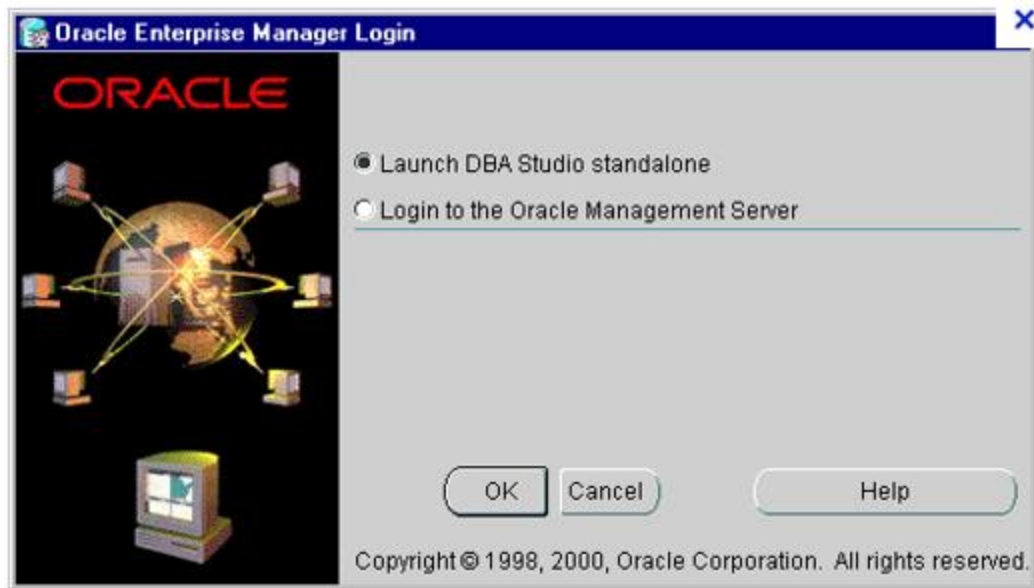


The following features:

- Partitioned Tables and Indexes
- Performance improvements in OPS
- Global V\$ views, TAF
- Index Organized Tables
- RMAN
- Drop Column on Table
- Enterprise Manager v2
- DBA Studio
- Log Miner
- JVM, Linux and XML support

came with Oracle 8 (1997).

Oracle Enterprise Manager Readme Release 2.2.0.0.0 Production



Some additional features came with Oracle 9, Oracle 10 and Oracle 11



At Oracle Open World 2006 in San Francisco, Oracle announced some exciting new features of Oracle 11g and they promised 482 new features. In reality with all the releases of 11g, these features are close to 1000.



Keep up with New Features in Oracle Database 12c and 18c



Diana Gray

SR. PRINCIPAL CURRICULUM PRODUCT MANAGER



No doubt about it: there's a mind-bending amount to learn about all the Oracle Database 12c and Oracle Database 18c new features that are now being released at regular intervals.

AI setup of the environment used for testing

- Exadata X4-2L High Capacity
- Linux 7.6
- RAC with CDB / 2 PDBs
- Kernel version: 4.1.12-124.23.4.el7uek.x86_64
- Image version: 19.1.2.0.0.190111

	PARAMETER_NAME	PARAMETER_VALUE	LAST_MODIFIED	MODIFIED_BY	CON_ID
1	AUTO_INDEX_DEFAULT_TABLESPACE	(null)	(null)	(null)	1
2	AUTO_INDEX_MODE	IMPLEMENT	18.02.2019 05:55:51,0000000000	SYS	1
3	AUTO_INDEX_REPORT_RETENTION	31	(null)	(null)	1
4	AUTO_INDEX_RETENTION_FOR_AUTO	373	(null)	(null)	1
5	AUTO_INDEX_RETENTION_FOR_MANUAL	(null)	(null)	(null)	1
6	AUTO_INDEX_SCHEMA	(null)	(null)	(null)	1
7	AUTO_INDEX_SPACE_BUDGET	50	(null)	(null)	1
8	AUTO_INDEX_DEFAULT_TABLESPACE	AUTO_INDEX_TS	18.02.2019 08:24:34,0000000000	JULIAN	3
9	AUTO_INDEX_MODE	IMPLEMENT	18.02.2019 07:12:14,0000000000	SYS	3
10	AUTO_INDEX_REPORT_RETENTION	31	(null)	(null)	3
11	AUTO_INDEX_RETENTION_FOR_AUTO	373	(null)	(null)	3
12	AUTO_INDEX_RETENTION_FOR_MANUAL	(null)	(null)	(null)	3
13	AUTO_INDEX_SCHEMA	(null)	(null)	(null)	3
14	AUTO_INDEX_SPACE_BUDGET	50	(null)	(null)	3

Automating index creation: database schemas

- Two schemas: Julian and SSB – 91GB in size
- Workload: Julian (OLTP) / SSB (DW)
- Initially 22 indexes were created – unusable & invisible (metadata)

	OWNER	INDEX_NAME	INDEX_TYPE	TABLE_OWNER	TABLE_NAME	TABLE_TYPE	UNIQUENESS	COMPRESSION	PREFIX_LENGTH	TABLESPACE_NAME
1	JULIAN	SYS_AI_64uv6wb5168u	NORMAL	JULIAN	CLIENTS	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
2	JULIAN	SYS_AI_abrca2u9qmx7	NORMAL	JULIAN	SALES	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
3	SSB	SYS_AI_5w7tru8hdqdku	NORMAL	SSB	CUSTOMER	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
4	SSB	SYS_AI_gsybnacn2kr6h	NORMAL	SSB	CUSTOMER	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
5	SSB	SYS_AI_4hr6k8tvstwb2	NORMAL	SSB	CUSTOMER	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
6	SSB	SYS_AI_aku6vbmajgxay	NORMAL	SSB	CUSTOMER	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
7	SSB	SYS_AI_9042ajz896w01	NORMAL	SSB	DWDATE	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
8	SSB	SYS_AI_bzz58t6r8k0z0	NORMAL	SSB	DWDATE	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
9	SSB	SYS_AI_3xwxu6p0gyfbj	NORMAL	SSB	DWDATE	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
10	SSB	SYS_AI_6txfa3cjjkwrdr	NORMAL	SSB	DWDATE	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
11	SSB	SYS_AI_2p7yvsjns8c4c	NORMAL	SSB	LINEORDER	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
12	SSB	SYS_AI_9cc9q7m0cwmxc	NORMAL	SSB	PART	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
13	SSB	SYS_AI_b9xx08ams9cf	NORMAL	SSB	PART	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
14	SSB	SYS_AI_259fssldhml9r	NORMAL	SSB	PART	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS
15	SSB	SYS_AI_7jfpntmsn632s	NORMAL	SSB	SUPPLIER	TABLE	NONUNIQUE	ADVANCED LOW	(null)	AUTO_INDEX_TS

Visible automating index creation in the database

- On the 2nd day, 13 indexes became VALID (real segments)
- On the 3rd day, we got the first VISIBLE index

	OWNER	INDEX_NAME	INDEX_TYPE	TABLE_NAME	COMPRESSION	TABLESPACE_NAME	STATUS	VISIBILITY
1	JULIAN	SYS_AI_64uvm6wb5168u	NORMAL	CLIENTS	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
2	SSB	SYS_AI_0chq8a0gu5n4r	NORMAL	DWDATE	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
3	SSB	SYS_AI_6txfa3cjkkwr	NORMAL	DWDATE	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
4	SSB	SYS_AI_aad4k3zx4uq6d	NORMAL	SUPPLIER	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
5	SSB	SYS_AI_6110j0s6n0w0w	NORMAL	SUPPLIER	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
6	SSB	SYS_AI_4hr6k8tvstwb2	NORMAL	CUSTOMER	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
7	SSB	SYS_AI_259fssldhml9r	NORMAL	PART	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
8	SSB	SYS_AI_bzz58t6r8k0z0	NORMAL	DWDATE	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
9	SSB	SYS_AI_akuvbma8jgxay	NORMAL	CUSTOMER	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
10	JULIAN	SYS_AI_abrca2u9qmxt7	NORMAL	SALES	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
11	SSB	SYS_AI_gsybnacn2kr6h	NORMAL	CUSTOMER	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
12	SSB	SYS_AI_3xwxu6p0gyfbj	NORMAL	DWDATE	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
13	SSB	SYS_AI_ftpfnyycu4cq	NORMAL	PART	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
14	SSB	SYS_AI_6rssn7grbmwdk	NORMAL	CUSTOMER	ADVANCED LOW	AUTO_INDEX_TS	VALID	VISIBLE
15	SSB	SYS_AI_806umcm1b46a8	NORMAL	SUPPLIER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE

Visible automating index creation in the database

- 25 indexes created altogether: 2 for Julian and 23 for SSB

	OWNER	INDEX_NAME	INDEX_TYPE	TABLE_NAME	COMPRESSION	TABLESPACE_NAME	STATUS	VISIBILITY
1	SSB	SYS_AI_65xf2uy0g3m9w	NORMAL	SUPPLIER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
2	SSB	SYS_AI_grz6guy0wz35m	NORMAL	PART	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
3	SSB	SYS_AI_awxuwm542d0ym	NORMAL	LINEORDER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
4	SSB	SYS_AI_a49967210bb79	NORMAL	PART	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
5	SSB	SYS_AI_4cc7hwr7fujvz	NORMAL	CUSTOMER	ADVANCED LOW	AUTO_INDEX_TS	VALID	VISIBLE
6	SSB	SYS_AI_7f1lr4rxls2dc	NORMAL	DWDATE	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
7	SSB	SYS_AI_lzb9txurz7g0v	NORMAL	LINEORDER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
8	SSB	SYS_AI_ftpfnyycu4cqy	NORMAL	PART	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
9	SSB	SYS_AI_f3cvd753y8p3g	NORMAL	LINEORDER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
10	SSB	SYS_AI_6rssn7grbmwdk	NORMAL	CUSTOMER	ADVANCED LOW	AUTO_INDEX_TS	VALID	VISIBLE
11	SSB	SYS_AI_b89n0xm97qnch	NORMAL	LINEORDER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
12	SSB	SYS_AI_806ymcmlh46a8	NORMAL	SUPPLIER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
13	SSB	SYS_AI_5cdhad14h7k4m	NORMAL	SUPPLIER	ADVANCED LOW	AUTO_INDEX_TS	VALID	INVISIBLE
14	SSB	SYS_AI_a3y6r0r5gbhsj	NORMAL	LINEORDER	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
15	SSB	SYS_AI_bpmlttu837mxj	NORMAL	LINEORDER	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
16	SSB	SYS_AI_5xmww6wnvnaip	NORMAL	DWDATE	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE
17	SSB	SYS_AI_0chq8a0gu5n4r	NORMAL	DWDATE	ADVANCED LOW	AUTO_INDEX_TS	UNUSABLE	INVISIBLE

Odd automating index creation in the database?

Oracle creates also auto indexes on global temporary tables

SSB	SYS_AI_awxuw542d0ym	NORMAL	SSB	LINEORDER	VALID	INVISIBLE
SSB	SYS_AI_bpm1ttu837mxj	NORMAL	SSB	LINEORDER	UNUSABLE	INVISIBLE
SSB	SYS_AI_c0byq60wk3sn1	NORMAL	SSB	PART	UNUSABLE	INVISIBLE
SSB	SYS_AI_grz6guy0wz35m	NORMAL	SSB	PART	VALID	INVISIBLE
SSB	SYS_AI_259fss1dhm19r	NORMAL	SSB	PART	UNUSABLE	INVISIBLE
SSB	SYS_AI_ftpfnyycu4cq	NORMAL	SSB	PART	VALID	INVISIBLE
SSB	SYS_AI_a49967210bb79	NORMAL	SSB	PART	VALID	INVISIBLE
SSB	SYS_AI_65xf2uy0g3m9w	NORMAL	SSB	SUPPLIER	VALID	INVISIBLE
SSB	SYS_IL0000116655C000	LOB	SSB	PLAN_TABLE	VALID	VISIBLE
36\$\$						

SSB	SYS_AI_4cc7hwr7fujvz	NORMAL	SSB	CUSTOMER	VALID	VISIBLE
SSB	SYS_AI_6rssn7grbmwdk	NORMAL	SSB	CUSTOMER	VALID	VISIBLE

25 rows selected.

SQL> select sysdate from dual;

SYSDATE

2019-02-26

Automating indexes and GTTs: statistics

```
-----
Parsing Schema Name   : UDSGC2
SQL ID                : bzgk328su0695
SQL Text              : MERGE INTO UDCNDLOGINSTRFIXING USING (SELECT /*+
                        cardinality (t 5)*/ IDUDCNDINSTRUMENT, FIXINGVALUE,
                        FIXINGDATE, EFFECTIVEDATE, ELEMENTTAG, ISOFFICIAL,
                        ISESTIMATE, IDCDFIXINGTYPE, IDCDFIXINGVALUETYPE,
                        SUPPORTPOINT, IDC DINSTRUNIT_FXNG,
                        IDC DNUMBERINGSCHM_UNDINS, IDENTIFIER_UNDINS,
                        IDC DNUMBERINGSCHM_FI...
Improvement Factor    : 8.1x
```

Execution Statistics:

	Original Plan	Auto Index Plan
Elapsed Time (s):	955964	92639
CPU Time (s):	210136	47799
Buffer Gets:	25396	387
Optimizer Cost:	7	7
Disk Reads:	154	48
Direct Writes:	0	0
Rows Processed:	1307	0
Executions:	21	1

Automating indexes and GTTs: statistics

- UDCNDHTMPINSTRFIXING is a GTT
- Cost is the same for both plans (7) – but as Oracle seems to judge from the buffer gets – those are much lower in the auto index plan (387) as the GTT is always empty
- When Oracle test execute the workload SQL - Oracle measure the benefit (or otherwise) of the new auto index and that is used to form the basis of making a decision - as usual, both IO and CPU matters

00:00:01						
* 6	TABLE ACCESS STORAGE FULL	UDCNDHTMPINSTRFIXING		5	2775	2
00:00:01						
* 7	TABLE ACCESS BY INDEX ROWID BATCHED	UDCNDLOGINSTRFIXING_T		1	112	1
00:00:01						
* 8	INDEX RANGE SCAN	SYS_AI_dz04n5dnmfq63		1		1
00:00:01						

Automating indexes and GTTs: statistics

Parsing Schema Name : UDSGC2
SQL ID : 5gxsf2jpst909
SQL Text : DELETE FROM UDCNDLOGINSTRISPOSITION WHERE
IDUDCNDLOGINSTRUMENT IN (SELECT /*+ cardinality(id 100)*/
IDUDCNDINSTRUMENT FROM UDTMPINSTRCANDIDATEID ID) AND NVL
(ISFROZEN, 0) = 0 AND IDUDCNDLOGINSTRISUANCE IS NOT NULL
Improvement Factor : 809x

Execution Statistics:

	Original Plan	Auto Index Plan
Elapsed Time (s):	2047216	169
CPU Time (s):	139038	169
Buffer Gets:	3145	0
Optimizer Cost:	69	123
Disk Reads:	795	0
Direct Writes:	0	0
Rows Processed:	0	0
Executions:	17	1

Automating indexes and GTTs: what matters

- According to Oracle the DELETE has improved based on 809 times less buffer gets (the 809 is exactly the improvement factor in the report below)
- The auto index is clearly worse, while the original index on (IDUDCNDLOGINSTRUMENT) matches exactly the where clause, the auto index (IDUDCNDLOGINSTRUMENT,ISARCHIVE) has an additional column not even appearing in the query

```
Parsing Schema Name : UDSGC2
SQL ID              : 5gxsf2jpst909
SQL Text            : DELETE FROM UDCNDLOGINSTRISPOSITION WHERE
                     IDUDCNDLOGINSTRUMENT IN (SELECT /*+ cardinality(id 100)*/
                     IDUDCNDINSTRUMENT FROM UDTMPINSTRCANDIDATEID ID) AND NVL
                     (ISFROZEN, 0) = 0 AND IDUDCNDLOGINSTRISISSUANCE IS NOT NULL
Improvement Factor  : 809x
```

Automating indexes and SQL improvements

- Improvement factor (in 19.3.0) is the ratio of the buffer gets per exec (original/auto_index)
- Oracle supposedly subtract parse buffer gets and if the value is zero, Oracle then call it "1"
- The rest is guesswork without looking at the underlying data Oracle input into the calculations
- Oracle retrieve the values from dba_advisor_sqlstats and a bunch of wri\$_sqlset tables
- DBA_AUTO_INDEX_VERIFICATIONS: "_auto_index_log" (wri\$_adv_tasks, wri\$_adv_objects), wri\$_adv_objects, dba_advisor_executions

EXECUTION_NAME	SQL_ID	ORIGINAL_PLAN_HASH_VALUE	AUTO_INDEX_PLAN_HASH_VALUE	ORIGINAL_BUFFER_GETS	AUTO_INDEX_BUFFER_GETS	STATUS
1 SYS_AI_2019-07-01/08:34:28 56pwkjspvmg3h		1448083145	1448083145	1532,586453106908529525279814743342338865	1786	UNCHANGED
2 SYS_AI_2019-07-01/08:34:28 7hk2m2702ua0g		2048963432	2048963432	15,9720695970695970695970695970695970696	102	UNCHANGED
3 SYS_AI_2019-07-01/08:34:28 9dt3dqym1tqzw		3954032495	1068597273	46	4	UNCHANGED
4 SYS_AI_2019-07-01/08:34:28 dunt7pwuax92s		1878158884	1448083145	91	1625	UNCHANGED
5 SYS_AI_2019-07-01/08:34:28 dy8cxyd3mvlas		2679498789	2048963432	189	61	UNCHANGED
6 SYS_AI_2019-07-01/08:34:28 gkxxkghxubh1a		2220165490	2220165490	1530,855787476280834914611005692599620493	2202	UNCHANGED

Automating index creation reports and errors

REPORT

GENERAL INFORMATION

Activity start	: 19-FEB-2019 05:47:52
Activity end	: 22-FEB-2019 05:47:52
Executions completed	: 182
Executions interrupted	: 0
Executions with fatal error	: 3

SUMMARY (AUTO INDEXES)

Index candidates	: 28
Indexes created (visible / invisible)	: 10 (1 / 9)
Space used (visible / invisible)	: 19.76 GB (66.06 MB / 19.69 GB)
Indexes dropped	: 0
SQL statements verified	: 23
SQL statements improved (improvement factor)	: 1 (21.6x)
SQL plan baselines created (SQL statements)	: 7 (6)
Overall improvement factor	: 1.7x

PDBs, SPA and automating index creation

- Error in PDB occurs when the SPA task is initiated

ERRORS

- ORA-13613: The requested operation is not supported for this advisor object.
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist
- The current operation was interrupted because it timed out.
- The current operation was interrupted because it timed out.
- The current operation was interrupted because it timed out.
- The current operation was interrupted because it timed out.
- ORA-00942: table or view does not exist
- ORA-00942: table or view does not exist

Automating index creation in a CDB database

- Automatic Indexing worked in the CDB (under root) but stopped working in the PDB after it ran once

	EXECUTION_NAME	EXECUTION_START	EXECUTION_END	ERROR_MESSAGE	STATUS	CON_ID
1	SYS_AI_2019-02-19/23:42:56	19.02.2019	19.02.2019	ORA-13613: The requested operation is not supported for this advisor object.	FATAL ERROR	3
2	SYS_AI_2019-02-19/23:09:37	19.02.2019	19.02.2019	(null)	COMPLETED	3

- Restart the expert system with did not help
- `exec dbms_auto_index_internal.task_proc;`
- In the PDB it ran only once in the beginning while in CDB\$ROOT it is being executed every 15th minute
- Tried to close and open the PDB and restart the CDB, disable auto indexing in CDB\$ROOT - nothing
- Even cleanup did not help

```
exec dbms_auto_index_internal.ai_cleanup;  
exec dbms_auto_index_internal.ai_clear;  
exec dbms_auto_index_internal.ai_init;
```

Automating indexes and cascading parameters

- The issue was that the PDB was *not* open on the 2nd instance of the RAC
- Once open, all started working perfectly
- Also, compatible was set to 19.3.0
- Setting 'AUTO_INDEX_MODE' to 'IMPLEMENT' in CDB\$ROOT is not cascading to the PDB, has to be set per PDB

	EXECUTION_NAME	EXECUTION_START	EXECUTION_END	ERROR_MESSAGE	STATUS	CON_ID
1	SYS_AI_2019-02-19/09:52:10	19.02.2019	(null)	(null)	EXECUTING	1
2	SYS_AI_2019-02-19/09:36:32	19.02.2019	19.02.2019	(null)	COMPLETED	1
3	SYS_AI_2019-02-19/09:21:07	19.02.2019	19.02.2019	(null)	COMPLETED	1
4	SYS_AI_2019-02-19/09:05:28	19.02.2019	19.02.2019	(null)	COMPLETED	1
5	SYS_AI_2019-02-19/08:50:04	19.02.2019	19.02.2019	(null)	COMPLETED	1
6	SYS_AI_2019-02-19/08:34:27	19.02.2019	19.02.2019	(null)	COMPLETED	1
7	SYS_AI_2019-02-19/08:19:01	19.02.2019	19.02.2019	(null)	COMPLETED	1

Automating index creation “debugging”

```
select * from CDB_AUTO_INDEX_VERIFICATIONS;
```

Script Output x Query Result x

SQL | Fetched 50 rows in 0,614 seconds

EXECUTION_NAME	SQL_ID	ORIGINAL_PLAN_HASH_VALUE	AUTO_INDEX_PLAN_HASH_VALUE	ORIGINAL_BUFFER_GETS	AUTO_INDEX_BUFFER_GETS	STATUS
1 SYS_AI_2019-02-20/22:35:37	0kax92k55zcfr	1912374025	3891418381	12	219	REGRESSED
2 SYS_AI_2019-02-20/22:35:37	2nw460mvh3ufp	2415333313	2363182177	9	6	UNCHANGED
3 SYS_AI_2019-02-20/22:35:37	4mbfwfdz7hnlp	3800749930	2204887307	17	1	UNCHANGED
4 SYS_AI_2019-02-20/22:35:37	68vj4ma79tx5q	3213731783	4066371529	15	1	UNCHANGED
5 SYS_AI_2019-02-20/22:35:37	bu8hp9rys8rza	3800749930	2204887307	15	8	UNCHANGED
6 SYS_AI_2019-02-20/22:35:37	fws wzr221616y	3800749930	2204887307	17	8	UNCHANGED
7 SYS_AI_2019-02-20/22:35:37	g3ms0pga3t46b	3213731783	4066371529	17	8	UNCHANGED
8 SYS_AI_2019-02-20/22:35:37	gphr25z2hdbbm	1912374025	3891418381	12	6	UNCHANGED
9 SYS_AI_2019-02-20/22:35:37	gu333trudpwu5	1912374025	3891418381	12	12	UNCHANGED
10 SYS_AI_2019-02-22/00:59:07	35pvtjx6bsxxu	2845882216	2845882216	1704032	(null)	TIMED OUT
11 SYS_AI_2019-02-22/00:59:07	3fc8flbmy6vy0	3464368412	1177810356	1706832	1705630	UNCHANGED
12 SYS_AI_2019-02-22/10:45:04	6896t0msr5xqk	2788210254	2788210254	1678987,6159420289...	1690668	UNCHANGED
13 SYS_AI_2019-02-23/04:51:49	4pugyvllnwh73	2788210254	2788210254	1458904,7142857142...	1690668	UNCHANGED
14 SYS_AI_2019-02-23/04:51:49	5g750cvas829n	2788210254	2788210254	1690710	1690668	UNCHANGED
15 SYS_AI_2019-02-23/06:24:58	5n6kq4a8hvjyd	2788210254	2788210254	1403179,5	1690668	UNCHANGED
16 SYS_AI_2019-02-23/06:56:01	2r45qbkv3ylbs	3800749930	1692241576	9	214	REGRESSED
17 SYS_AI_2019-02-23/06:56:01	54jdw nuztq914	1912374025	3891418381	6	1	UNCHANGED
18 SYS_AI_2019-02-23/06:56:01	5sb5vt2f6zqh2	4066371529	4066371529	22	1	UNCHANGED
19 SYS_AI_2019-02-23/06:56:01	68vj4ma79tx5q	915180818	915180818	16	11	UNCHANGED

Automating index creation “debugging”

```
select * from DBA_AUTO_INDEX_IND_ACTIONS;
```

Script Output x Query Result x

SQL | Fetched 50 rows in 0,987 seconds

EXECUTION_NAME	ACTION_ID	INDEX_NAME	IND...	TABLE_NAME	TABLE_OWNER	COMMAND	STATEMENT
1 SYS_AI_2019-02-20/22:35:37	30	SYS_AI_6rssn7grbmwdk	SSB	CUSTOMER	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_6rssn7grbmwdk" ON "SSB"."CUSTOMER" ("C_CUSTK...
2 SYS_AI_2019-02-20/22:35:37	31	SYS_AI_0chq8a0gu5n4r	SSB	DWDATE	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_0chq8a0gu5n4r" ON "SSB"."DWDATE" ("D_DATEKEY...
3 SYS_AI_2019-02-20/22:35:37	32	SYS_AI_ftfpnyycu4cq	SSB	PART	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_ftfpnyycu4cq" ON "SSB"."PART" ("P_PARTKEY",...
4 SYS_AI_2019-02-20/22:35:37	33	SYS_AI_5cdhad14h7k4m	SSB	SUPPLIER	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_5cdhad14h7k4m" ON "SSB"."SUPPLIER" ("S_SUPPK...
5 SYS_AI_2019-02-20/22:35:37	34	SYS_AI_5w7tru8hdqdku	SSB	CUSTOMER	SSB	DROP INDEX	DROP INDEX "SSB"."SYS_AI_5w7tru8hdqdku" ONLINE
6 SYS_AI_2019-02-20/22:35:37	35	SYS_AI_9042ajz896w01	SSB	DWDATE	SSB	DROP INDEX	DROP INDEX "SSB"."SYS_AI_9042ajz896w01" ONLINE
7 SYS_AI_2019-02-20/22:35:37	36	SYS_AI_7jfpntmsn632s	SSB	SUPPLIER	SSB	DROP INDEX	DROP INDEX "SSB"."SYS_AI_7jfpntmsn632s" ONLINE
8 SYS_AI_2019-02-20/22:35:37	37	SYS_AI_9cc9q7m0cwmxc	SSB	PART	SSB	DROP INDEX	DROP INDEX "SSB"."SYS_AI_9cc9q7m0cwmxc" ONLINE
9 SYS_AI_2019-02-20/22:35:37	38	SYS_AI_ftfpnyycu4cq	SSB	PART	SSB	REBUILD INDEX	ALTER INDEX "SSB"."SYS_AI_ftfpnyycu4cq" REBUILD ONLINE
10 SYS_AI_2019-02-20/22:35:37	39	SYS_AI_6rssn7grbmwdk	SSB	CUSTOMER	SSB	REBUILD INDEX	ALTER INDEX "SSB"."SYS_AI_6rssn7grbmwdk" REBUILD ONLINE
11 SYS_AI_2019-02-20/22:35:37	40	SYS_AI_5cdhad14h7k4m	SSB	SUPPLIER	SSB	REBUILD INDEX	ALTER INDEX "SSB"."SYS_AI_5cdhad14h7k4m" REBUILD ONLINE
12 SYS_AI_2019-02-20/22:35:37	41	SYS_AI_2p7yvsjns8c4c	SSB	LINEORDER	SSB	REBUILD INDEX	ALTER INDEX "SSB"."SYS_AI_2p7yvsjns8c4c" REBUILD ONLINE
13 SYS_AI_2019-02-22/23:10:16	57	SYS_AI_5xmwv6wnvnajp	SSB	DWDATE	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_5xmwv6wnvnajp" ON "SSB"."DWDATE" ("D_YEARMON...
14 SYS_AI_2019-02-22/23:10:16	58	SYS_AI_awxuw542d0ym	SSB	LINEORDER	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_awxuw542d0ym" ON "SSB"."LINEORDER" ("LO_CUS...
15 SYS_AI_2019-02-22/23:10:16	59	SYS_AI_bpmlttu837mxj	SSB	LINEORDER	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_bpmlttu837mxj" ON "SSB"."LINEORDER" ("LO_ORD...
16 SYS_AI_2019-02-22/23:10:16	60	SYS_AI_a3y6r0r5gbhsj	SSB	LINEORDER	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_a3y6r0r5gbhsj" ON "SSB"."LINEORDER" ("LO_SUP...
17 SYS_AI_2019-02-22/23:10:16	61	SYS_AI_65xf2uy0g3m9w	SSB	SUPPLIER	SSB	CREATE INDEX	CREATE INDEX "SSB"."SYS_AI_65xf2uy0g3m9w" ON "SSB"."SUPPLIER" ("S_CITY"...
18 SYS_AI_2019-02-22/23:10:16	62	SYS_AI_gsybnacn2kr6h	SSB	CUSTOMER	SSB	DROP INDEX	DROP INDEX "SSB"."SYS_AI_gsybnacn2kr6h" ONLINE
19 SYS_AI_2019-02-22/23:10:16	63	SYS_AI_6txfa3cjjkwr	SSB	DWDATE	SSB	DROP INDEX	DROP INDEX "SSB"."SYS_AI_6txfa3cjjkwr" ONLINE
20 SYS_AI_2019-02-22/23:10:16	64	SYS_AI_6110j0s6n0w0w	SSB	SUPPLIER	SSB	DROP INDEX	DROP INDEX "SSB"."SYS_AI_6110j0s6n0w0w" ONLINE
21 SYS_AI_2019-02-22/23:10:16	65	SYS_AI_65xf2uy0g3m9w	SSB	SUPPLIER	SSB	REBUILD INDEX	ALTER INDEX "SSB"."SYS_AI_65xf2uy0g3m9w" REBUILD ONLINE
22 SYS_AI_2019-02-22/23:10:16	66	SYS_AI_4cc7hwr7fujvz	SSB	CUSTOMER	SSB	REBUILD INDEX	ALTER INDEX "SSB"."SYS_AI_4cc7hwr7fujvz" REBUILD ONLINE
23 SYS_AI_2019-02-22/23:10:16	67	SYS_AI_4cc7hwr7fujvz	SSB	CUSTOMER	SSB	ALTER INDEX VISIBLE	ALTER INDEX "SSB"."SYS_AI_4cc7hwr7fujvz" VISIBLE

Automating index creation errors

```
select EX.execution_type,F.message
from DBA_ADVISOR_FINDINGS F, DBA_ADVISOR_EXECUTIONS EX
WHERE F.EXECUTION_NAME = EX.EXECUTION_NAME AND F.TYPE = 'ERROR';
```

Script Output x Query Result x	
SQL All Rows Fetched: 38 in 0,202 seconds	
EXECUTION_TYPE	MESSAGE
1 AUTO	ORA-13613: The requested operation is not supported for this advisor object.
2 EXPLAIN PLAN	ORA-00942: table or view does not exist
3 EXPLAIN PLAN	ORA-00942: table or view does not exist
4 EXPLAIN PLAN	ORA-00942: table or view does not exist
5 EXPLAIN PLAN	ORA-00942: table or view does not exist
6 EXPLAIN PLAN	ORA-00942: table or view does not exist
7 EXPLAIN PLAN	ORA-00942: table or view does not exist
8 EXPLAIN PLAN	ORA-00942: table or view does not exist
9 EXPLAIN PLAN	ORA-00942: table or view does not exist
10 TEST EXECUTE	The current operation was interrupted because it timed out.
11 TEST EXECUTE	The current operation was interrupted because it timed out.
12 TEST EXECUTE	The current operation was interrupted because it timed out.
13 TEST EXECUTE	The current operation was interrupted because it timed out.

Now testing: will manual indexes be dropped?

	PARAMETER_NAME	PARAMETER_VALUE	LAST_MODIFIED	MODIFIED_BY
1	AUTO_INDEX_COMPRESSION	ON	15.07.2019 23:35:13,000000000	SYS
2	AUTO_INDEX_DEFAULT_TABLESPACE	AUTO_INDEX_TS	15.07.2019 23:41:07,000000000	SYS
3	AUTO_INDEX_MODE	IMPLEMENT	14.06.2019 16:39:17,000000000	SYS
4	AUTO_INDEX_REPORT_RETENTION	31	(null)	(null)
5	AUTO_INDEX_RETENTION_FOR_AUTO	365	01.07.2019 10:52:42,000000000	SYS
6	AUTO_INDEX_RETENTION_FOR_MANUAL	3	01.07.2019 10:52:55,000000000	SYS
7	AUTO_INDEX_SCHEMA	schema IN (SOE)	14.06.2019 04:00:08,000000000	SYS
8	AUTO_INDEX_SPACE_BUDGET	50	(null)	(null)

Percentage of budget used =

auto index segment size / size of all segments * 100

– Calculation applies to auto index tablespace

Now testing: will manual indexes be dropped?

```
select * from dba_indexes where owner = 'SOE' and auto != 'YES';
```

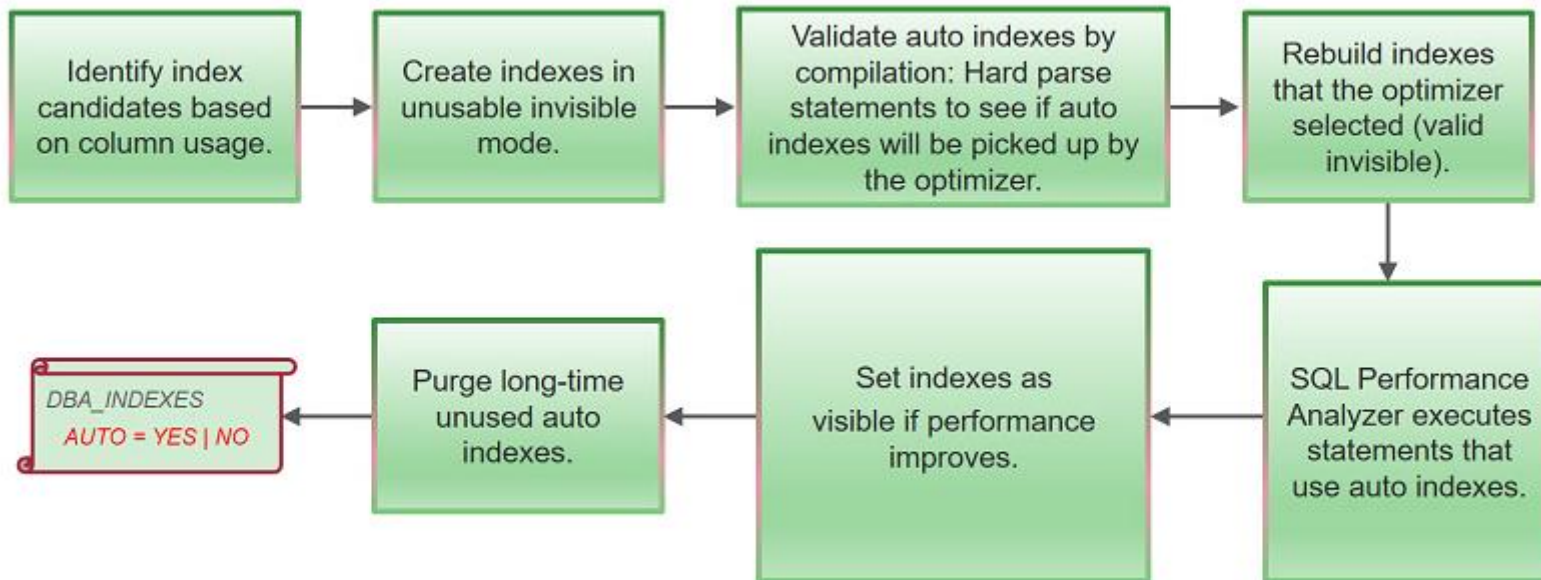
	OWNER	INDEX_NAME	INDEX_TYPE	TABLE_OWNER	TABLE_NAME	TABLE_TYPE	UNIQUENESS
1	SOE	T_PK	NORMAL	SOE	T	TABLE	UNIQUE
2	SOE	CUSTOMERS_PK	NORMAL/REV	SOE	CUSTOMERS	TABLE	UNIQUE
3	SOE	ADDRESS_PK	NORMAL/REV	SOE	ADDRESSES	TABLE	UNIQUE
4	SOE	CARD_DETAILS_PK	NORMAL/REV	SOE	CARD_DETAILS	TABLE	UNIQUE
5	SOE	WAREHOUSES_PK	NORMAL	SOE	WAREHOUSES	TABLE	UNIQUE
6	SOE	ORDER_ITEMS_PK	NORMAL/REV	SOE	ORDER_ITEMS	TABLE	UNIQUE
7	SOE	ORDER_PK	NORMAL/REV	SOE	ORDERS	TABLE	UNIQUE
8	SOE	PRODUCT_INFORMATION_PK	NORMAL	SOE	PRODUCT_INFORMATION	TABLE	UNIQUE
9	SOE	PRD_DESC_PK	NORMAL	SOE	PRODUCT_DESCRIPTIONS	TABLE	UNIQUE
10	SOE	INVENTORY_PK	NORMAL	SOE	INVENTORIES	TABLE	UNIQUE

19c: Machine Learning and Artificial Intelligence

- Oracle first create invisible-invalid indexes using `dbms_stats.report_col_usage`
- Then they test-parse SQL in the SQL tuning set to see if it will use the indexes - because at this stage Oracle have candidate indexes but Oracle won't yet know if they stand a chance of actually being useful - if the SQL does use the indexes, then Oracle make them invisible-valid
- Then they test execute the queries in SPA (allowing the SQL to see the invisible-valid indexes) and they check to see if they run better with these proposed indexes - if they do, then we can make the valid indexes visible
- In theory we can get a new batch of visible indexes every 15 minutes

19c: Machine Learning and Artificial Intelligence

An *expert system* implementing indexes automatically



19c: what is used for Auto Indexing?

- **ASTS:** Automatic SQL Tuning sets – SYS_AUTO_STS - a SQL Tuning Set on which it runs the SQL Access Advisor
- **SPA:** SQL Performance Analyzer
- **IV:** index visibility by setting optimizer_use_invisible_indexes to TRUE
- **SPM:** SQL Plan Management: avoid plan regression for SQL statements

PARAMETER_NAME	PARAMETER_VALUE
-----	-----
AUTO_INDEX_COMPRESSION	OFF
AUTO_INDEX_DEFAULT_TABLESPACE	
AUTO_INDEX_MODE	OFF
AUTO_INDEX_REPORT_RETENTION	31
AUTO_INDEX_RETENTION_FOR_AUTO	373
AUTO_INDEX_RETENTION_FOR_MANUAL	
AUTO_INDEX_SCHEMA	
AUTO_INDEX_SPACE_BUDGET	50

What can and cannot be done to Auto Indexing?

- **Drop or rebuild** Automatic Indexes: you cannot drop an auto index
- You cannot drop an auto index in 19.3 - no legitimate way
- Set `AUTO_INDEX_RETENTION_FOR_AUTO` to 1
- If you need to do this, raise an SR and describe your scenario
- But the error message "ORA-01418: specified index does not exist" is misleading as the index **does** exist - it should be some new error like "ORA-99999: operation not permitted on autonomous objects" or something like that :-/
- **Make Oracle not using** the Automatic Indexes:
- `select /*+ NO_INDEX(f "SYS_AI_0rn9u2kmxxbs7") */ ...` ← observe case sensitivity with double-quotation marks:
- `select /*+ OPT_PARAM('_optimizer_use_auto_indexes','OFF') */` ← unofficial technique:

Automating Index creation reports and errors

REPORT

GENERAL INFORMATION

Activity start : 19-FEB-2019 05:47:52
Activity end : 22-FEB-2019 05:47:52
Executions completed : 182
Executions interrupted : 0
Executions with fatal error : 3

SUMMARY (AUTO INDEXES)

Index candidates : 28
Indexes created (visible / invisible) : 10 (1 / 9)
Space used (visible / invisible) : 19.76 GB (66.06 MB / 19.69 GB)
Indexes dropped : 0
SQL statements verified : 23
SQL statements improved (improvement factor) : 1 (21.6x)
SQL plan baselines created (SQL statements) : 7 (6)
Overall improvement factor : 1.7x

Underscore Parameters for Auto Indexing

```
select * from sys.smb$config where parameter_name like '%AUTO_INDEX%' order by 1;  
select * from sys."_auto_index_log" order by log_id;  
select * from dba_auto_index_statistics where value>0 order by 1;
```

Script Output x Query Result x					
All Rows Fetched: 20 in 1,023 seconds					
PARAMETER_NAME	PARAMETER_VALUE	LAST_UPDATED	UPDATED_BY	PARAMETER_DATA	
1 AUTO_INDEX_DEFAULT_TABLESPACE	0	28.05.2019 01:20:31,0000000000	JULIAN	AUTO_INDEX_TS	
2 AUTO_INDEX_MODE	0	28.05.2019 01:14:16,0000000000	SYS	IMPLEMENT	
3 AUTO_INDEX_REPORT_RETENTION	31 (null)		(null)	(null)	
4 AUTO_INDEX_RETENTION_FOR_AUTO	0 (null)		(null)	373	
5 AUTO_INDEX_RETENTION_FOR_MANUAL	0	28.05.2019 02:32:00,0000000000	JULIAN	3	
6 AUTO_INDEX_SCHEMA	0 (null)		(null)	<filters></filters>	
7 AUTO_INDEX_SPACE_BUDGET	50 (null)		(null)	(null)	
8 _AUTO_INDEX_ABSDIFF_THRESHOLD	100 (null)		(null)	(null)	
9 _AUTO_INDEX_CONCURRENCY	1 (null)		(null)	(null)	
10 _AUTO_INDEX_CONTROL	0 (null)		(null)	(null)	
11 _AUTO_INDEX_DERIVE_STATISTICS	0 (null)		(null)	ON	
12 _AUTO_INDEX_IMPROVEMENT_THRESHOLD	20 (null)		(null)	(null)	
13 _AUTO_INDEX_REBUILD_COUNT_LIMIT	5 (null)		(null)	(null)	
14 _AUTO_INDEX_REBUILD_TIME_LIMIT	30 (null)		(null)	(null)	
15 _AUTO_INDEX_REGRESSION_THRESHOLD	10 (null)		(null)	(null)	
16 _AUTO_INDEX_SPA_CONCURRENCY	1 (null)		(null)	(null)	
17 _AUTO_INDEX_STS_CAPTURE_TASK	0	28.05.2019 01:14:16,0000000000	SYS	ON	
18 _AUTO_INDEX_TASK_INTERVAL	900 (null)		(null)	(null)	
19 _AUTO_INDEX_TASK_MAX_RUNTIME	3600 (null)		(null)	(null)	
20 _AUTO_INDEX_TRACE	0 (null)		(null)	(null)	

Is Automatic Indexing available only in Exadata?

```
SQL> select banner_full from v$version;
```

```
BANNER_FULL
```

```
-----  
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production  
Version 19.3.0.0.0
```

```
SQL> EXEC DBMS_AUTO_INDEX.CONFIGURE('AUTO_INDEX_MODE','IMPLEMENT');  
BEGIN DBMS_AUTO_INDEX.CONFIGURE('AUTO_INDEX_MODE','IMPLEMENT'); END;  
*
```

```
ERROR at line 1:
```

```
ORA-40216: feature not supported
```

```
ORA-06512: at "SYS.DBMS_SYS_ERROR", line 79
```

```
ORA-06512: at "SYS.DBMS_AUTO_INDEX_INTERNAL", line 9180
```

```
ORA-06512: at "SYS.DBMS_AUTO_INDEX", line 283
```

```
ORA-06512: at line 1
```

Is Automatic Indexing available only in Exadata?

```
SQL> alter system set "_exadata_feature_on"=true scope=spfile;
```

System altered.

```
SQL> startup force
```

```
Total System Global Area    4915722728 bytes
```

```
Fixed Size                    9144808 bytes
```

```
Variable Size                 922746880 bytes
```

```
Database Buffers             3976200192 bytes
```

```
Redo Buffers                  7630848 bytes
```

Database mounted.

Database opened.

```
SQL> EXEC DBMS_AUTO_INDEX.CONFIGURE('AUTO_INDEX_MODE','IMPLEMENT');
```

PL/SQL procedure successfully completed.

Is Automatic Indexing available only in Exadata?

Table 1-8 Performance

Feature / Option / Pack	SE2	EE	EE-ES	DBCS SE	DBCS EE	DBCS EE-HP	DBCS EE-EP	ExaCS	Notes
Automatic Indexing	N	N	Y	N	N	N	N	Y	EE-ES: Available on Exadata. Not available on Oracle Database Appliance.
SQL Quarantine	N	N	Y	N	N	N	N	Y	EE-ES: Available on Exadata. Not available on Oracle Database Appliance.
Real-Time Statistics	N	N	Y	N	N	N	N	Y	EE-ES: Available on Exadata. Not available on Oracle Database Appliance.
High-Frequency Automatic Optimizer Statistics Collection	N	N	Y	N	N	N	N	Y	EE-ES: Available on Exadata. Not available on Oracle Database Appliance.

Is Automatic Indexing available only in Exadata?

```
DECLARE
    quarantine_config VARCHAR2(30);
BEGIN
    quarantine_config := DBMS_SQLQ.CREATE_QUARANTINE_BY_SQL_TEXT(
        SQL_TEXT => to_clob('select * from dual'));
END;
/

BEGIN
    DBMS_SQLQ.ALTER_QUARANTINE(
        QUARANTINE_NAME => 'SQL_QUARANTINE_c7fbanxudy9yv',
        PARAMETER_NAME  => 'ELAPSED_TIME',
        PARAMETER_VALUE => DBMS_SQLQ.ALWAYS_QUARANTINE);
END;
/

SQL> select comp_data FROM SYS.SQLOBJ$ SO, SYS.SQLOBJ$DATA SOD where SO.SIGNATURE = SOD.SIGNATURE AND
        SO.PLAN_ID = SOD.PLAN_ID;
2
COMP_DATA
-----
<outline_data><hint><![CDATA[QUARANTINE(ELAPSED_TIME=4294967295)]]></hint></outl

SQL> select * from dual;
select * from dual
          *
ERROR at line 1:
ORA-56955: quarantined plan used
```

19c new: Security

- If the database password file name or location has been changed and the metadata cache needs to be refreshed with the details of the updated database password file, you can do it simply with the following command:

```
SQL> alter system flush passwordfile_metadata_cache;
```

```
System altered.
```

- In 19c, most of the Oracle Database supplied schema-only accounts now have their passwords removed to prevent users from authenticating to these accounts - but DBAs can still assign passwords to the default schema-only accounts

19c new: Memoptimized Rowstore

- Memoptimized rowstore is all about high performance data streaming and provides the following functionalities:

Fast ingest:

- Fast ingest optimizes the processing of high-frequency, single-row data inserts into a database
- Fast ingest uses the large pool for buffering the inserts before writing them to disk, so as to improve data insert performance

Fast lookup

- Fast lookup enables fast retrieval of data from for high-frequency queries
- Fast lookup uses a separate memory area in the SGA called the memoptimize pool for buffering the data queried from tables
- For using fast lookup, you must allocate appropriate memory size to the memoptimize pool using MEMOPTIMIZE_POOL_SIZE

19c new: SQL

- Automatic resolution of SQL plan regressions:
 - SQL plan management searches for SQL statements in the Automatic Workload Repository (AWR)
 - Prioritizing by highest load, it looks for alternative plans in all available sources, adding better-performing plans to the SQL plan baseline
 - Oracle also provides a plan comparison facility and improved hint reporting: use DBMS_XPLAN.COMPARE_PLANS to generate a report
- SQL Quarantine:
 - SQL statements that are terminated by Oracle Database Resource Manager due to their excessive consumption of CPU and I/O resources can be automatically quarantined
 - The execution plans associated with the terminated SQL statements are quarantined to prevent them from being executed again
 - Check out the new Oracle package DBMS SQLQ – cool stuff!

19c new: ADG DML and RTS

- You can now run DML on the Active Data Guard standby database
- When run on the standby side, the update is passed to the Primary database where it is executed and the resulting redo of the transaction will update the standby after which control will be returned to the application
- Real-Time Statistics is also a new cool feature:
 - Oracle automatically gathers online statistics during conventional DML operations
 - By gathering some statistics automatically during DML operations, the database augments the statistics gathered by DBMS_STATS
 - Fresh statistics enable the optimizer to produce more optimal plans
 - EXEC DBMS_STATS.FLUSH_DATABASE_MONITORING_INFO;
 - NO_GATHER_OPTIMIZER_STATISTICS prevents the collection of real-time statistics

19c new: High-Frequency Statistics

- High-Frequency Automatic Optimizer Statistics Collection complements the standard automatic statistics collection job
- By default, the high-frequency statistics collection occurs every 15 minutes
- Turn on/off: EXEC
`DBMS_STATS.SET_GLOBAL_PREFS('AUTO_TASK_STATUS','ON');`
- Change the default period: EXEC
`DBMS_STATS.SET_GLOBAL_PREFS('AUTO_TASK_INTERVAL','300');`
- Monitor: `DBA_AUTO_STAT_EXECUTIONS`
- The high-frequency automatic statistics task will not start during the maintenance window

19c new: Data Pump

- Oracle Data Pump allows tablespaces to stay read-only during TTS import
- Oracle Data Pump can work in test mode for transportable tablespaces
- Oracle Data Pump supports resource usage limitations with the introduction of two new parameters: MAX_DATAPUMP_JOBS_PER_PDB and MAX_DATAPUMP_PARALLEL_PER_JOB
- Data Pump no longer enables secure, password-protected roles by default. Beginning with 19c, you must explicitly enable password-protected roles for an individual export or import job. A new command-line parameter has been added, ENABLE_SECURE_ROLES=YES | NO that can be used to explicitly enable or disable these types of roles for an individual export or import job
- The new Data Pump command-line parameter CREDENTIAL enables secure import into a managed service from dump files in the Oracle Object Store Service

19c new: init.ora / spfile

- There are 6 new init.ora parameters in 19c
- DBAs can now tune in DataGuard the amount of wait time by using DATA_GUARD_MAX_IO_TIME and DATA_GUARD_MAX_LONGIO_TIME
- You can check details for all six:
 - ADG_REDIRECT_DML
 - DATA_GUARD_MAX_IO_TIME
 - DATA_GUARD_MAX_LONGIO_TIME
 - ENABLE_IMC_WITH_MIRA
 - LOB_SIGNATURE_ENABLE
 - MAX_DATAPUMP_PARALLEL_PER_JOB

19c new: init.ora / spfile

- There are several initial parameters added in Oracle Database 19.3
- You can find these new 8 parameter's information in v\$parameter but they are not included in Database Reference 19C or other online documents

- allow_rowid_column_type
- client_statistics_level
- cpu_min_count
- http_proxy
- max_idle_blocker_time
- scheduler_follow_pdftz
- ssl_wallet
- unified_audit_common_systemlog



Patch 30125133: DATABASE RELEASE UPDATE 19.5.0.0.0

Last Updated 15-Oct-2019 13:41 (7 days ago)

Product Oracle Database - Enterprise Edition
(More...)

Release Oracle Database 19.0.0.0.0
Platform Linux x86-64

Size 737.7 MB

Download Access Software

Classification Security

Patch Tag All Database

19c new: hybrid tables

- Hybrid Partitioned Tables: now large portions of a table can reside in external partitions, for example in the Oracle Cloud
- With this new feature, you can also easily move non-active partitions to external files, such as Oracle Data Pump files, for a cheaper storage solution
- Hybrid partitioned tables support all existing external table types for external partitions:
 - ORACLE_DATAPUMP
 - ORACLE_LOADER
 - ORACLE_HDFS
 - ORACLE_HIVE

TƏŞƏKKÜR EDİRƏM

GRAZAS

ΣΑΣ ΕΥΧΑΡΙΣΤΩ **PAKKA PÉR DANKE**

JU FALENDEROJ **TEŞEKKÜR EDERİM**

THANK YOU **OBRIGADO**

ارکش

БЛАГОДАРЯ

ভোমাকে ধন্যবাদ

CẢM ƠN BẠN

TAK

ХВАЛА

GRACIAS

ESKERRIK ASKO

ДЗЯКУЙ

GRAZIE

KÖSZÖNÖM

DANK U

הודות