ORACLE®

#### 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.



# SQL Performance Tuning in the Autonomous Database Era

Andrew Holdsworth Christine Qu Cary Dong

**Real-World Performance** 

November 17, 2017



### Introductions









#### Introductions Andrew Holdsworth

- 28 Years at Oracle
- Vice President Real World Performance
  - Good performance is rarely an accident
  - Most people get the systems they deserve
  - Good enough rarely is, aspire for excellence not good enough.









Introductions 曲卓 (Christine Qu)

• 12 Years at Oracle

ORACLE



- Manage Real-World Performance education in China
- Learn to analysis from top down, make sure you are on the right direction
- Be open and positive, aim high

ORACLE

REAL-WORLD PERFORMANCE





#### Introductions 董志平(Cary Dong)

- 15 years of Oracle experience
- 9 years in RWP
- Manage Real-World Performance projects in China





#### Real-World Performance Who We Are

- Part of the Database Development Organization
- Global Team located in USA, Europe, Asia
- 350+ combined years of Oracle database experience
- Innovate to achieve exceptional Database Performance
- Our methods:
  - Use the product as it was designed to be used
  - Numerical and logical debugging techniques

REAL-WORLD PERFORMANCE

- Educate others about the best performance methods and techniques
- Avoid and eliminate "tuning" by hacking/guessing/luck





### The Real World Performance Perception Problem

#### Where database user look for performance improvements

#### Perception



- Application Algorithmns and Correct Product Usage
- Database Platform



The best place to look for performance Improvements





#### Program Agenda

- **1** Why Autonomous Database?
- SQL Performance Tuning Strategy
- Makes Tuning Smarter





#### Program Agenda

- 1 Why Autonomous Database?
- <sup>2</sup> SQL Performance Tuning Strategy
- <sup>3</sup> Makes Tuning Smarter





#### Oracle Vision for Autonomous Database

Goal - Eliminate all human labor

No human labor means lower cost No human error means better reliability and security



#### Oracle's Vision for Autonomous Database How we do it

#### Self-Driving

- User defines workloads and policies, database makes them happen

### Self-Securing

Protection from both external attacks and internal users

## Self-Repairing

- Automated protection from all downtime





#### Automated vs. Autonomous

#### **Automated**

- The car simplifies operations by automating tasks:
  - Cruise control
  - Emergency stopping
  - Warnings for lane changes
- The database simplifies operations:
  - Automatic storage management, automatic storage management, …
  - Dozens of other features

#### ORACLE



#### **Autonomous**

- The car drives itself
  - No need to use the steering wheel or brake.
  - Simply tell the car where you are going.

# The database manages itself All features automatically implemented Simply tell the database your goals

### Automatically Diagnoses Performance

- Autonomous Database includes Oracle's industry leading diagnostics automation
- Automatic Database Diagnostic Monitor (ADDM)
  - Automatically diagnoses root cause of performance issues
  - -A.I. (Expert System)
- Active Workload Repository (AWR)
  - Automatically keeps detailed performance and resource utilization history
- Real-Time SQL Monitoring
  - Automatically diagnoses how resources are used in SQL statements



#### Automatically Optimizes Itself

- Autonomous Database includes Oracle's industry leading database tuning automation
- Many database algorithms self optimize caching, locking, storage indexes, offload, etc.
- Optimizer is now further automated by gathering statistics as new data is loaded
- Automatic SQL (Re)Tuning
  - Machine learning technology that is constantly re-evaluating SQL plans based on the latest statistics and recommending/implementing better plans
- Tuning is workload dependent e.g. OLTP vs analytics so we specialize services
- Tuning is an extremely difficult problem

OPACI

- Even scheduling a fleet of trucks to optimally make deliveries is incredibly complex
- Database has many degrees of freedom and tradeoffs that must be considered



#### Database Administrator Questions and Fears

- Will my job go away?
- Will my job change?
- Will I lose control?





#### What does Autonomous Database mean for the DBA?

#### Less time on Administration

- Less time on infrastructure
- Less time on patching, upgrades
- Less time on ensuring availability
- Less time on tuning

#### More time on Innovation

- More time on database design
- More time on developing new apps
- More time on data analytics
- More time on securing data

## Challenge: There are more data management tasks than humans to do the work





#### Reality vs. Fears

- Your job will not go away there is a **shortage of skilled database experts** 
  - Database automation has been improving for decades
- But your job will change, so you must change
- You will spend less time on **generic maintenance**, more time **innovating**
- More time with the business
  - -Executing more projects, reducing backlog, getting more value from data
  - -Cloud's fast provisioning and pay-as-you-go enables rapid experimentation
- More time with developers

REAL-WORLD PERFORMAN

- -Optimizing data access, improving end-user experience
- More time on new techniques like Machine Learning





#### Database Administrator Must Ensure End-to-End Service Levels

- Focus on optimizing how applications and databases work together:
  - Define good data model, and good SQL
  - Avoid row at a time processing, and repeated logins/parsing
  - Understand where time is being spent
  - Understand tradeoffs in parallelism, plans, indexes, partitions, etc.
  - Ensure that sensitive data is kept secure end-to-end
  - Beware what an application asks for the database will do exactly that
- Need new skills in Cloud Systems, Cloud Networking, Cloud Storage
  - -DBA sizes, monitors, ensures full stack works as expected
- Database Administrator will gain more control
  - Will be in charge of, and in control of, end-to-end service levels

#### Program Agenda

1 Why Autonomous Database?

- SQL Performance Tuning Strategy
- <sup>3</sup> Makes Tuning Smarter





- Elapsed time: 1 hour  $\rightarrow$  1 min
  - Will you stop working?
- Could it be possible to make it down to 1 sec?

Aim High



• Is it a valid SQL?

ORACLE

- Do you know the business logic the SQL represents for?
- Is it a well constructed SQL?
- Any mistakes in the SQL?
  - N-1 join conditions for a N table join?
  - Implicit data type conversion?

Aim High
Work on Good SQL



- Is your database software correctly patched?
- Are you running with default init.ora parameter settings? If not, why?





- Stats
  - Stats on raw data
  - System stats
- Constraints
  - NOT NULL, PK, FK, UK
- Schema design
  - Index
  - Partitioning
  - Compression
  - Clustering

Aim High
Work on Good SQL
Environment settings
Optimization



- Access method
- Join method
- Join order
- Distribution method
- Skew?

Work on Good SQL

Aim High

Environment settings

Optimization

Execution





- Diagnosing
  - SQL Monitor Report
  - Find the leverage





Execution





#### Program Agenda

1 Why Autonomous Database?

<sup>2</sup> SQL Performance Tuning Strategy

Makes Tuning Smarter





#### Introduction to SQL Monitor

- Released in Oracle Database 11g
- Enables in depth performance monitoring of a SQL statement
  - Always on, enabled out of the box
  - Single execution of the SQL statement
  - Includes currently executing statements
- Monitored statements
  - Serial statements with 5 seconds of total CPU/IO time
  - All parallel statements
  - -/\*+ monitor \*/ hint
  - Queries / DML / DDL





#### Introduction to SQL Monitor

- Formats available
  - $-\mathsf{Text}$
  - -HTML
  - Active (use this one!)
- Available from
  - Monitored SQL screen
  - command line
  - EM, EM Express
  - PerfHub



#### How we use the SQL Monitor report

- Top down
  - Where is time spent?
  - Which row sources?
  - Estimated vs actual row cardinalities?
  - -Executions
    - Parallel server executions
    - Nested loop iterations
    - Partition wise operations
  - -Skew



#### Example 1 **Observations**

- Query currently runs for 40 seconds
- Needs to run under 5 seconds
- Examination of the SQL Monitor report shows
  - -67% CPU
  - -33% IO
- The most expensive row source is HASH JOIN RIGHT OUTER (line 5) -45% of the CPU
  - Large amount of read/write from TEMP

OPACI







#### ORACLE' ORACLE'

#### Example 1 Diagnosis

- However...
  - Is the HASH JOIN itself the problem?
- Examination of the Cardinality estimates
  - $-\, {\rm The}\ {\rm scan}\ {\rm of}\ {\rm the}\ {\rm CARGUYS}\ {\rm table}\ {\rm at}\ {\rm line}\ {\rm 11}$ 
    - estimate of 42M rows vs actual 40M rows
    - estimate is very accurate
  - The scan of the CARGUYS table at line 9
    - estimate of 186K rows vs actual 40M
    - more that 200x underestimated





#### Example 1: Diagnosis





#### Example 1 Diagnosis

• What is the cause of the big mis-estimate in cardinality?

- Examining the predicate information:

Filter Predicates ("MODEL"='458 Italia' AND "MAKE"='Ferrari' AND "COUNTRY"='Italy')

- All three filter predicates are highly correlated
  - Ferrari's are only made in Italy
  - Only Ferrari makes a model called the '458 Italia'
- The optimizer has multiplied the selectivity of the individual predicates and underestimated the number of rows that will be retrieved





## Example 1: Diagnosis

Details							
🛛 Mark Standards - Age Plan 😫 Manifest 📐 Adoler 🔛 Mark	0						
Display Tetsular v Plan Hush Value 3048826790 📑 Plan No	00						
Operation	Onject	Une ID	Predicate	Psining	Comparison Court	Estimated Street	Extracted Boles
EL SELECT STATEMENT		0		100000000	(		
E SORT ASCRESATE		1					
G-PK COORDINATOR		2			Highly	correlated pre	edicates
E) FX SEND QC (RANDOM)	:7010001	3			(יייפייי	concluted pro	
E SORT ACCREDATE		4					
E HASH JOJN RIGHT OLITER		5	10		12		
B to second		6				1 1966	i 7,8138
ID IN SEND INDADCAST	17016000	7				13856	17,0138
E PX BLOCK ITEM/TOK		8		118		1 2864	17,8138
TABLE ACCESS STIPHAGE FULL	CARGINS		21	1-13		634 / 5868	7,013#
ET PK BLOCK ETBLATOR	TABLE ACCESS STORAGE FULL	1. C			×	-	9120
THREE ACCESS STORAGE FRIL	Sub-tree Co Operation Co Sub-tree Tir Bri Byt Access Prodical Filter Prodical Partition Sto Partition Sto Partition Sto Partition Sto	et 638(41% CPU) et 638(41% CPU) et 0000:03 ws 1884 7,0354 et CANSUNS 	NE NO YOU	= Ternarî AND "CU = Tlaiy)	DuNTRY"=1306/]	61K	424



## Example 1

- Solution
  - Use extended statistics
    - Build a column group on MAKE, MODEL, COUNTRY
  - The optimizer can now determine the selectivity of the three predicates combined
- Result
  - The cardinality estimate has now changed to 40M rows.
  - Changes distribution method from BROADCAST to HASH-HASH
  - This results in a more efficient HASH JOIN, which does not spill to TEMP
  - Query now runs in 3 seconds.



#### **Example 1: Solution**

REAL-WORLD PERFORMANCE



#### Example 2 Observations

- Query takes 42 minutes
  - Almost all CPU
- Complex SQL statement
- Parallel execution requested
- Time spent in nested full table scans
- Optimizer estimated it would perform one table scan
  - Actually performed 5776 table scans





#### **Example 2: Overview**



#### ORACLE

REAL-WORLD PERFORMANCE

NPACLE Enterprise Manager	GIALOGMPANY, EUBETRIGULA LAWDIV, GUADESCRIPTION O AUMELIKS, VALUE DATE, FORT	ADCT_UNIT_D,4) AS P_AREA_NAME. _AETINE.							English
onitored SQL Execution Deta	AUM JOC VALUE DP_ARSA_NG BUN ND_CVICUE_MUNBER - BUN BONDRE TYPE PROFILION GR	R							
Overview	301N VIV LIVEN ADMINIALUE A AUM/003_10 AND AUM/MATRIX	AUM ON GLN.DED_ID =							
SQL ID ITSS537736 Paralel U . Descubor Started Wed May 4, 20 Execution ID 16779160 User API34P Patric Calls D	LINE TODS VIE LINE (LINE) GUILDEL TO AUROCOULDE A AUROMATING CANADATESC GUILENCE, DEPTH = 1, AND SUBSTRIACCT_UNITILIST = TO SUBSTRIACCT_UNITILIST = TO SUBSTRIACCT_UNITILIST = TO DISTINCT_SUBSTRICE_DOMPANY LINEN_GUICOMPREL_COMPANY	LISE PURE DN ND TVYT WHENE VY 08 VT			41.2m	Ro Statistics Butter Gets 30 Requests 10 Bytes	-		
Details	LENGTHITRINILINGN_GLOOMPANT = WX	RELICID_COMPANY II							
Pier Materia - 20 Pier   25 Para	REAMOIN = MICOIN    UNION DODUPANY -BLANDIN - NICO	ALL SELECT							
100 Hadt Value 34122200/02	SUBSTRINKE DEV 3.2) AS LESS	04. Shi ninu ( chi het					of the set	Namue Ick of the late struct	to top of balance 10 land attend
aeration	GUN COMPANY . SUBSTRICEN.	ACCT_WAT/0.4) AS	Estimated	Lot V			Terror (M., 30 Arequetts	EPU Activity 4s	Wait Activity In
FI DISENT STATEMENT	LAWBIV, GLA DESCRIPTION O ADM2/MX_VALUE DATE_FERST	ACTIVE:					and the second second second		
B LOAD TABLE CONVENTIONAL	AUN NOL VALUE DP. AREA, NO	R						a	
B PK COORDINATOR	GULDHL, TVRE FROM LVSN_G	LNAMES GLN LEFT			Complex S	JUL		100	
E RE SEND QC (RANDOM)	301N VW_DWSN_AUMRVALUE A OUM ORU TO SAFE SUAS NOTED	VIM ON CLN OBJ D =		3,651	ctatomor	<b>a</b> t			
E WINDOW SORT	LEFT LODN VW_LINEN_ALIMXVA	LUE AUM2 ON	1	3.631	Statemer	IL			
B PX RECEIVE	AUM2/MATER CAT-CATEAC	ND. TIVE HINDRE	1	5.849					
C OX SIND HARH	GUNLEVEL_DEPTH = 1 AND (		4	8,849					
B HIGH JOTH CUTCH	AUBITRIACCT_UNIT.LGI = 30	or () a soon (secare)	4	5.049	D.	0			
ET PAR ARCEDVE	DISTINCT SUBSTRICLD_COMP	ANV.1.3) AS	1	3,822	0	0			
B PX SEND HASH	DIV, LIVIN_GLEDMIRALCOM	MARY REDA	3	5,822	D.	0			
E HASH JOIN (	LENGTHITKIM LASN GLOOMP	GLECHPREL WHERE RELOLD CONRAVY	4	3,622	£	0			
B PX RECEIV	** ) SHE ON BLOOM PARY * HIS	E EDMIFANN ARE		3.764	D	p			
E) PX SEND	BLOOMP.COMPARY = GLN.COM	FANT MAD	4	5,764	0				
E view	GUCOMPLANOTV = SUBSTRIGLA VAR LEVER BISS	AT AT WHERE	4	9.764	D	D			
É HASH	BUN BUNSETS_85_8W = W AND	D OLALEVEL DEPTH	.4	3,764	0	0			
(B ex	AD3.40_VALVE + '20011231'	AND	1	8,706	D	0			
D.P	SUBSTRIGEN WAR_LEVEL_DISK	P.(14)	4	5.706	D	0			
Ē	HASH JODI BUMMENED		1	3.705	D	P			
(	3 AUFRER SOAT				Ð	0			
	C OC RECEIVE			3,847	n	- 11			
	and the statement of the statement of the	interior and a		22.2					

ORACLE

ORACLE REAL-WORLD PERFORMANCE

#### Example 2: Plan statistics shows where time spent



#### ORACLE

#### Example 2: Plan statistics leads to source of scan iterations

HI PR COORDINATOR				1 2	1,110		
B PK SEND QC (RANDOM)		17QL0012	1 5.631		0		
B WINCOW SORT			t 5.05t	E.	0		
E PX RECEIVE			1 5,849	4	Ú.		
A PX SSND HASH		(TQ10012	1 5.649		0		
E HASH JODY OUT	CA DUMERCO		1 5.049	(1)	0		
E PX RECEIVE			1 5,823	1	0		
D IN SHALL HAS	H	17Q10015	1 8,832		n		
E HASH JOIN	DUTER BURYERED		t 5.622	1	0		
B PK RECEI	ve		t 5.764	1	0		
ET POLISION	а назн	rtgLoous	6 3,794		-10		
E view			1 5,764		P		
BHK	H TODI BUFFERSO		1 5.764		0		
EI #0	Receive		1 5.704		52 I I I I I		
B	EX SEND HASH	·		Th	ic line drive	00	
6	TIDO CONSUMPTION OF A CONTRACT OF A CONTRACTACT OF A CONTRACTACT OF A CONTRACTACT OF A	imizer expe	CTS		is line anv	es	
	E BLEFER SORT	rour pot F T		r	nactad laar		
	D DX RECEIVE OTC	10W, 10US, I	/0		iesteu ioop		
	E PK SEND HASH		and the second s				
	E VIEW		1 5,647		2		
	E sont utique		1 3,647		27 2408		
	C RESTED LOOPS		1 3.640		1238		
	E RESTED LOOPS OUTER		t + 475	- (	5.776		
	ED FULTER			-	3,775		
	E ASSTED LOOPS OUTER		1 4,475		6.343		
	B NESTED LOOPS OUTER		1 4.475	1	0.042		
	E TABLE ACCESS & GLOBAL DIDEX 8.	IWIN_GLNAMES	1 4,474		8,242		
	INCEX RANCE SCAN	1DX\$5.87889008	6,941 95	1	8.695		
	E HAT_VEW ACCESS BY THOER ROWD	AND ACCESS OF GLOBAL IN	December 1	8.343	8.118		
	INCER RANGE SICAN	WV_LWEN_ALIMYVALUE	1 1	8,243	5.145		
	ANT_VEW ADDRESS BY DERX KOWID	MATHABITY TOWARDS	8 8	8,543	3,404		
	INDEX RANGE SCAN	MATTWART ADMONATCH	5 1	6.343	3.609		
	E) MAT_VIEW ACCESS BY INDER ROWID	HV_LWSR_ALIMIVALUE	1 1	5,773	5,627		
	INDEX KANDE BLAD	NV_LIVIN_AUNIVUUE	1	5,772	5,627		
	El Alten		1 4,212	5,776	7.238		
	D UNDON ALL PUSHED PREDICKTE			5.776	7.230		
	C MERGE 2018 ART1 SA		1 2,117	5,776	4,752		
	FLINESTED LOOPS OUTER		1 14 19	8,778	7.235		

#### ORACLE

REAL-WORLD PERFORMANCE

#### Example 2: Plan Statistics screen shows estimate and actual

E PR COORDINATOR				1 81	3,250	
B PX SEND QC (RANDOM)		TQL0012	1 5.691	(1)	0	
B WINDOW SORT			1 5.051		0	
E PX RECEIVE			£ 5,849		.0	
E EX SSND HASH		17050012	1 5.649		0	
E HASH JODY OUTER	. BURYERCO		1 5.049	(E) .	0	
E PX RECEIVE			1 5,820	1	0	
D IN SHER HARH		17Q10015	1 5,833		n	
E HASH JOIN D	NUTER BURYERED		1 5.622	1	0	
E PK RECEIVE	£		t 5.764		۵	
E PA AND	нан	rtqLooue	5 3.794		-10	
B VIEW			1 5.764		0	
E HAGH	TODI BUFFERSO		1 5.764		0	
Elexa	ASCEIVE		1 5,706		0	
8 0	( SENO HASH	(TQ10002	1 5.705		0	
8	nadin Jolly BUPPERID		1 5.706		0	
E	3 BLEFER SORT				0	
	D DX RECEIVE		2 8.647			
	B PK SENO HASH		47	( A)	327	
	ontin	nizer estima	te for 🧧	1	227	
	E sour usagus		47		327 2-0(8	
	B RESTED LOOPS	lex scan is cl	ose 📲	1	7.238	
	B RESTED LOOPS OF		N		5.776	
	ET FOLTOR				3,773	
	B REALED FOODS ONLEY		1 4.475		6.343	
	B NESTER LOOPS OUTER		4.475	1	0.049	
	ET TABLE ADDEES BY BLOBAL D	IDEX R LWISN_GLNAMES	1 4,474	- (	8,242	
	INCEX RANCE SCAN	IDX\$5.87851008	6-941 96		8.695	
	E MAT_VIEW ACCESS 5Y INDEX	TABLE ADCESS BY DU	CONTRACTOR 1	8.343	5.115	
	- Diff			8,243	5.145	
	But on	lv expects 1	row	8,543	3,426	
	index		1	6.343	3.609	
	afte	er table filter	S i	5,775	5,637	
	INDEX KA			3,772	5,627	
	El Alten		1 4,212	5,776	7.238	
	E UNDON ALL PUSHED PREDICITE			5.776	7.230	
	C MERGE JOIN ARTI NA		1 2,117	5,776	4,152	
	FILMESTED LOOPS OUTER		1 14	5.778	7.735	

#### ORACLE

REAL-WORLD PERFORMANCE

#### Example 2 Diagnosis

• Poor cardinality estimate caused by SUBSTR() function



#### Example 2: SQL Monitor Plan Screen Shows Predicates



#### ORACLE



### Example 2 Solution

- Extended statistics on SUBSTR() function
- Optimizer now estimated 10 rows rather than 1
- Optimizer uses hash join rather than nested loop
- Accesses table with single scan
- Elapsed time: 6 seconds



#### Example 2: SQL Monitor After Expression Statistics Added

parte					-(					
ed SQL Execution Details 🥩						(	<u>Juer</u>	<u>y executes</u>	in six	
Peel .								, cocondc		
SQL 10 THE OF JOINT OF	Time & Wait Statistics							seconus.		
Parallel 🖽 a 🔝 a	Durgstigs		E 6.00							
Park Time Sur New 2016 7-51-11 AN	Database Time				tr.	DUM:	er Gets			
amodian III 2333+425	PL/SQU&Jave 0.0x				-	30.8#	driverus -	4,337		
User AFELAP	Welt Activity %	_		20	1.00				_	
Patch Cells: D				<u> </u>			DB	Time is mi	x of CPU. I	0
									<u> </u>	
Reques	ts to run naralle							and cluste	er waits	
neques										
IT WHEN IN		trated mail	Cast Detailes (to)	Ex	erat 1	Artsof B	amore (	Tamp (H., 10 Neurota	CHI Activity Dt	Walt Article 16
PH COORDINATOR					9	1.210				9.04
B PK SEND QC (RANDOM)	+101002+	419	126	-		1.310				1000000
G WINDOW SORT		419	128		4	1,210	38468			
O PM RECEIVE		419	128	-		\$,110				
E PX SRID HASH	(TQ10022	419	126		4	1,210				
EL HASH JOPN OUTER, BUPPERED		419	126		4	1,219	1045			
E AN VECETAR		419	128		4	284				
E PX SIND HASH	1210021	419	128		4	284				
THATH JOIN OUTER BUTTER	R	418	1.2.4		1	204	10/48			
C as acces		24	128		4	254				
		94	128	-		284				
	ally runs naralle		126	-	1	284	1004			
Actus			124		1	200	1040-			
Actua		- 244			12	190				
Actua			128	and the second second						
Actua		111	128			290	5545			
		1 H H H	128 128 128	Ξ	4	290	5Md			
Actua III HASH JOIN B PX ACCEIVE B PX SEND B		34 34 31 21	124 124 124 124 124	Ξ	4	290 1.300 1.308	5445			
		21 21 21	128 128 128 128 128 128	Ē	4 4 4	290 1.300 1.308 327	5145			
		7 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	128 128 128 128 128 128 128		4	290 1,300 1,308 327 327	546			

#### ORACLE

REAL-WORLD PERFORMANCE

### Example 2: SQL Monitor After Expression Statistics Added

Call I										
125	B HESTED LOOPS OUTER		1.0	2,678		4	342			
34	E METEO LOOPS OUTER	0	10	2,669	-	- 4	342			
88	E YK BLOCK ITERATO	Ê	20	2,004		-54	242			
33	TABLE ACCESS ST.	LINSH_OLIMANES_2	10	2,004		748	342			
1.1.1	- INDEX RAINSE ECAN	MV_LWSN_ALIMXVAD.	4	4		343	0		1.5	
	INDEX RANDE BOND	MV_LWIN_AUMINAU,	4			1042			1000	
85	E PX BLOOK ITERATOR		6.819	360	-	4	7.003			
20	MAT_VIEW ACCESS STOR	MV_LWSN_SLCOMINE	6.015	590	_	71	1.008		1 226	1.0
00	EI BUFFER SORT					4	-62	- 242	0.479	
26	ID PX RECEIVE				-	4	67			
	A Pri 50%0 mAdm	100000	5		-	1	63			
0	B MAT, VIEW ACCESS BY DIDEX BOWD	WV LWSR MODALIDA	5		-		62			
	NORS AND SCAN	MV CANES HENRALIES			-		43		11	
20	E VEW	VIE RADIA 2	175	3,756	-		-		- 9 p	
38-	d user intrus	- mganniga	416	2.265	-	-		110		
40	C rear coulds				-					
			-516	4,4,61		8				
30	Scan of na	mac tahla	now	, ha	c	- 5	24	14400		
8	Scan of na	mes table	now	' ha	s 🚽		64	4140		
88	Scan of na	mes table	now n	' ha	s		64 6,678	4140 3140		
3	Scan of nates	mes table mate of 1	now 0	' ha	s		64 6,678 1,344	4140 5140		
3 3 3 3 3 3 3 3	Scan of nates	mes table mate of 1	now 0	' ha	s		64 6,678 1,344 1,344	460 566		
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	Scan of nat esti	mes table mate of 1	now 0	ha	s		64 6,678 1,344 1,344 206	4140 3140		
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	Scan of nat esti HIGHTED LOOPE OUTER BUILDORE DUTER	mes table mate of 1	now 0	v ha	s		64 6,678 1,344 1,344 206 339	4400 3140		
	Scan of nat esti Prested Loope Outer Prested Loope Outer Prested Loope Outer Prested Loope Outer Prested Loope Outer	mes table mate of 1	now 0	ha	s		64 6,678 1,344 1,344 206 336 339	4140		
	Scan of nat esti esti esti esti esti esti esti est	mes table mate of 1	now 0	v ha	s	4 4 4 4 4 4 4 4 4 4 545	64 6,678 1,344 1,344 236 336 336 236	4100		9.08
	Scan of nat esti esti esti esti esti esti esti est	mes table mate of 1	now 0	ha	s		64 6,678 1,364 1,344 206 339 236 236	4100		9.08 10.09
	Scan of nate     S	mes table mate of 1		v ha	s	4 4 4 4 4 4 4 4 4 145 536	64 6,678 1,344 1,344 206 336 336 206 8 0	4148		9.08 9.09
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8	Scan of nate estimates B RESTED LOOPS OUTER B RESTED LOOPS OUTER B RESTED LOOPS OUTER B RESTED LOOPS OUTER B RESTED LOOPS OUTER D RESTED LOOPS OUTER B RESTED LOOPS OUTER D RESTED RESTE	mes table mate of 1		• ha	s	4 4 4 4 4 4 4 4 4 4 4 4 4 535	64 6,678 1,544 1,544 236 236 236 236 236 0 1,001	4148		9.08 9.09
	Scan of nate estimation Scan of nate Scan	mes table mate of 1		2 ha	s	4 4 4 4 4 4 4 4 4 4 4 4 7 45 7 45 7 23	44 4.678 1.544 1.544 1.544 1.544 1.544 1.545 235 235 235 0 1.003			9.08 9.09
	Scan of nat esti NEGTED LODPE OUTER IN RESTED RESTED OUTER IN RESTED RESTED OUTER IN	mes table mate of 1		2,664	s	4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4	64 6,678 1,584 1,584 2,586 2,596 2,596 2,595 0 1,003 1,003		325	9.08 0.09
	Scan of nat esti NEGTED LOOPS OUTER DIRECTED LOOPS OUTER DIRE	mes table mate of 1		2 ha	s	4 4 4 4 4 4 4 4 4 4 534 534 5023	64 6,678 1,584 1,584 2,58 355 2,58 2,58 0 1,001 1,002		1355	S.08 S.08 S.09
	Scan of natestal coord outer  Nested coord ou	mes table mate of 1		2 ha	s	4 4 4 4 745 536 745 745 745 745 745 745 745 745 745 745	44 4,678 1,544 1,544 2,546 2,556 2,556 0 1,001 1,002 5,620		135	9.08 8.08 8.08 45 8.08
	Scan of natestal loops outer  Nested Loops ou	mes table mate of 1	now 0	2,664	s	4 4 4 4 745 538 745 745 745 745 745 745 745 745 745 745	44 4,678 1,544 1,544 2,54 3,56 3,56 2,56 0 1,600 1,600 1,600		135	5.08 8.08 8.08 45 8.08
	Scan of national estimates Scan of national estimates Scan of national estimates Scan of national estimates Scan of mates Scan of mates Scan of mates Scan of mates	mes table mate of 1	now 0	2,664	s	4 4 4 4 745 336 745 745 745 745 745 7,236	44 4,678 1,544 2,544 2,546 2,556 2,556 2,556 0 1,500 1,500 2,600 1,600 1,600 1,600 1,600		135	5.08 8.08 8.08 45
	Scan of nat esti-	mes table mate of 1	now 0	2,664	s	4 4 4 4 745 336 745 745 745 745 745 7,236	64 6,678 1,544 1,544 226 336 226 0 5,001 5,001 5,001 5,001 5,000 5,000 5,000 5,000 5,000 5,000 5,000 5,000 5,000		135	5.08 5.08 5.09 43
	Scan of nat esti-	mes table mate of 1	now 0	2,664	s	4 4 4 4 746 336 746 746 746 746 746 746 746 746 746 74	54 5.679 1.584 1.584 2.06 2.06 2.06 5.001 5.005 5.001 5.005 5.001 5.005 5		135 155 152 6 07 49	5.08 5.09 45 5.08

#### ORACLE

ORACLE

REAL-WORLD PERFORMANCE

#### Example 3 Observations

- Query takes 56 minutes
- Time spent in HASH JOIN BUFFERED
- Cardinality estimates are accurate
- Large amount of TEMP IO
- Table Scan of 1TB table ran for 1800's.
  - Scan rate for the platform is much higher than that
    - Scan is constrained by the buffering of the HASH JOIN





#### **Example 3: Observations**



#### ORACLE



#### Example 3: Observations

Execution 32 134217	728 PL/SQL 8. Jave 0.0s						TO Bytes		210
Fetch Calls 8	Mait Activity %				10	Cel Off	cal Efficiency -E1.60%		
Details									
🗄 Plan Statistics 🔆 Plan	10 Paratial 📐 Activity 🔝 Metrica								
an Hadi Valan 21766570	e)							OF THE HIGH PAULO CAR ON THE SHEEK AND	we to hope between 10 Analists and 10 Me
peration		Ranta	Estimated Rows	· Cost	fernites(3388) Executives	Actual Rona	Hemory (Mar. Temp. (Max) 30 Requests	Set Offen., CPV Activity %	Web ectivity to
B INSERT STATISHENT					353	176		21	01. B
E FR.COORDINKTOR					353	176			01
E PX SEND QC (RAN	DOM)	rquaaas	LOOM	1,7306	176	176			
E) LOAD AS SELEC	1				176	176	Cardinality Es	innates pretty	31
E HARH GROUP	av .		LEOM	1,7304	( 18	\$299		rata	
PK RECEIVE			LOOM	1,7304	4 176	2,223M	acci	Irale	22
PR SEND I	4534	TULOBON	1609	1,736K	4 131	1,6544			15
В ниен в	ROUP BY		LEOM	1,738K	4 131.	1,654M	1000	<b>1</b> 40	
E vuw		VW_DAG_8	290	1,7304	162 *	1,6991		1.16	
E) HAS	H GROUF BY		790	1,7304		1.599M	4108	18	1.49 K.89
8.0	RECEIVE		280.	226		149		1.00	1.99
9	RX SEND HARH	(7050003	190	776	126	180		9.11	57
E	HAGH JOON BURFERED		590	776	176	186	10 2i	A North Annual Annua	36
9	EL PX RECEIVE		20394	805	126	23799		a.	E 03
	E PX SEND HIGH	11110001	2008	805	176	22294		1.34	E 102
11	II PERLOCK ITERATOR		20394	805	576	22294			B .01
<u> </u>	TABLE ACCESS STORAGE FULL	OUST_IDENTIFIER_DIM	20394	605	2.195	22294	336	-26.98 2 .11	4.43
b.	DI PX RECEIVE		190	780	126	190		3.0	1.92
	D PK SEND HASH	1010002	190	76x	576	198		10	1.2
	ET HARRY JODA		180	786	179	190		3.79	1
5	ID PK RECEIVE		364	2	576	993			
	E PK NIND BROADCAST LOCAL	(TQ10000	364	2	178	693			
1	EI PX NOCK ITENATOR	0222201	364	z	176	364			
6	TABLE ACCESS STORAGE PULL	DATE_DIM	364	2	1	364			
	EI PK BLOCK (TERATOR		430	75K	30	190			01
	TABLE ACCESS STORAGE PULL	TRANSACTION_ITEM_PET	430	75K	1,891	206	LHIS	-98.55	47
							U. C.	1122-1-1222	1975 M

#### 



#### Example 3: Observations

	User NULLS Heart Calls &	PL/SQL & Jave D.Os Net Activity %		_		100	Cell Difficut	TO Bytes	12.60%		218
Deb	ila										-
	un Statistics 🥀 Para 🔱 Paratai 📐	Activity Metrica									
Han H	Add Value 2176657000									off the right make did on the table	allow to hope to been 10 fearers and 10 feter
Open	(Den		Harris	Estimated Rows	Cost Elevative(3388a)	Esscations	Actual Roses No.	enery (Mar)	Temp (Mex) 30 Resource	Get Office CPV Activity %	Web ectivity to
8	UNSERT STATISHENT				-	353	176			21	03. 📓
0	E FX COORDINATOR					353	176			<b>a</b> .m.	101
3	E PX SEND QC (RANDOM)					176	176				
5	EI LOAD AS SELECT					175	176	91MB	8,515	an.	<b>31</b>
10	E HASH GROUP BY		lime sp	ent in H		176	\$299	958		1.53	
20	PK RECEIVE					176	2,223M			DZ.	22
10	E PK SEND HASH					131	1,6544			1.06	16
25	E HASH GROUP BY			LEOM	L/Ann	4 131.	1,654M	1258		an 100	
4	E YUW		VW_DAG_B	190	1,7384		1,6691			41.	
15	E) HASH GROUP BY			290	1,7366	191		4108	5258 <b>3</b> 424K	2.8	<b>16.69</b>
1	E PS RECEIVE			286	206	131	149			1.68	L 59
35	C PX SEND HARH		rrgsoaaa	190	776	126	180			9.11	1 A7
8	E HASH JOON BUFFER	RED		190	776	175	186	#008	Batigh Batight 21P		36
8	EI PX RECEIVE			20314	805	126	23799			a.a.	1 DR
25	E PX SEND HASH	4	mpionai	20094	805	526	22294			1.14	a 📱
30	EL PERIORS IT	ERATOR		20396	825	576	22299				10. B
10	TABLE ACC	SESS STORAGE FULL	OUST_IDENTIFIER_DIM	20399	625	2,155	22294		336	-26.58 1.1	2 AX
8	DI PA RECEIVE			190	76X -	176	190			3.0	1.42
10	ID PK SEND HASH	4:	1010002	190	25K	576	190			10	11.2
28	E HANNH SOON			180	76K	179	1941	13098		1.79	
8	ID PK RECEIV	ŧ		364	2	576	993				
45	E PE NIND	BROADCAST LOCAL	ODDOLUTE:	364	2	176	693				
3	ES PX BUD	OCK ITERATOR		364	1	176	354				
3	1490	E ACCESS STORAGE PULL	DATE_DIM	364	2	4	364				
8	E PK BLOCK	TTERATOR.		430	75K	90	196				01
8	C TABLE AS	CCESS STOMAGE PULL	TRANSACTION_T19H_PE7	490	75K	1,001	206		1,3416	-186.68 20 5-17	i 47

#### ORACLE



#### Example 3 Diagnosis

- Examining the Parallel Tab
  - Database time for PX servers on instance 6 is much higher
  - Expanding the Instance 6 node and expanding Parallel Set 1, we observe one PX server with a DB Time of 53 mins
  - A large part of the query was executed by a single PX server.



## Example 3: Diagnosis

				Overview			
() 21M	Noviento to Devellato		Tiese & Wait Statistics Directors and 30.3 In Ductations Time RUSQL & Save 0.0s Nat Activity %				
22	Navigate to Parallel ta						
				Details			
of the legit reuse disk on the table allows in aggre between 4D Arquests and 1D			ty Master	📰 Plan Statistics   🥀 Plan 🔯 Parater 🖂 Activ 🖌 Steve Instance Hodes			
Call Official Efficiency Buffer Gela	DO Respecto	Walt Activity %	Detabase Time	Paralal Server			
				the second constraints of the			
				El Al Panifel Servers			
2,2128	1.530K	1.19	1.2*	E An Panales Servers E Statemon 8			
2.312K 4,430K	4.351K	1.19	1.3h	E) Al Panales Servers El Instance 8 El Instance 6			
2.212% 4.4366 4.76396	4.351K	50 119	1.3h 3.5h 3.4h	E) Al Familie Servers El Intence 8 El Intence 8 El Intence 8			
2.212K 4.436K 4.703K	4.351K 2,845K 2,919K	119 50	1.3h 3.5h 3.4h	El Al Familio Servers El Intence 8 El Intence 8 El Intence 9 El Intence 9			
2.212% 4.436K 4.763K 1.209K 2.628K	4,351K 2,845K 2,919K 1,373K	119 50	1.3h 3.5h 3.4h 1.5h 1.2h	El Al Familio Servers El Intence 8 El Intence 8 El Intence 9 El Instance 9 El Instance 4 El Instance 4			
2.212% 4.436K 4.763N 1.209K 2.628K 1.660K	4.351K 2,845K 1.373K 1.373K 1.375K	1.19 50 10 10 10 10 10	1.3h 3.5h 1.3h 1.3h 1.3h	Al Familie Servers  Justance 8  Justance 8  Justance 9  Justance 9  Justance 4  Justance 1  Justance 1			
2.212% 4.436K 4.763N 1.209K 2.620K 1.660K 1.841K	4.351K 2,845K 1.373K 1.373K 1.375K 1.375K 1.375K	1.19 50 10 10 10 10 10 10 10 10 10 10 10 10 10	1.3h 3.5h 1.5h 1.5h 1.2h 1.2h	Al Familie Servers     Jintance 8     Jintance 8     Jintance 9     Jintance 9     Jintance 9     Jintance 9     Jintance 9     Jintance 1     Jintance 1			
2.212% 4.436K 4.763K 2.829K 2.828K 1.663K 1.842K 1.208K	1.533K 2.845K 2.845K 1.919K 1.373K 1.373K 1.258K 1.325K 1.258K	3.19 50 10 10 10 10 10 10 10 10 10 10 10 10 10	1.3h 3.4h 1.3h 1.2h 1.2h 1.2h 1.7h 1.7h	C Al Famile Servers  D Interce 8  D Interce 8  D Interce 9  D Interce			

Engregali († 1856, 2018, Danie zwojie in affiliales, bil rights enamel. Disce is a registered tademark of Chater Ecoposition and/or its affiliates. Ether names way in tademarks of their respective service.

#### Skew in database time





## Example 3: Diagnosis

Vier NULS Hence Caller &	Meit Activity %							
Details								
Plan Statistics   Res 🔯 Parallel	Activity III Marries		Expand the Parallel Set					
V Show Instance Hodes				W The Hight reason that are then as the tables allows to regar between 1D Requests and 1D Rets.				
Paralel Server	Ostabase Time	100 M 100		call Official Efficiency Buffer Geta				
E Al Panifel Servers		and a second second						
🖶 Shatance B	L023)	2.19	1,532K	2,3138				
E Instance 6								
Er an Perallel Set 1								
Paralai Server 1 (p054)	1.5m	1.01	130K	-49.25 1.181				
Penallel Server 2 (p055)	5.Sm	.91	157K	-49.25 🔛 1,179				
Persiel Server 3 (p856)	🔛 5.9m	1.66	E 2004	-49.25 🔤 1.179				
Panallel Server 4 (p857)	<b>2.0</b> m	1.07	1.504	-49.25 1,179				
-Feralel Server 5 (p058)	<b>3.5</b> m	1.09	157K	-49.25 🗾 1.179				
Parallel Server 6 (p099)	5. Sm	1.01	150ec	-49.25 1,179				
Ponallel Server 7 (p090)	1.8m	L07	150K	-40,21 1179				
Porsilei Server 9 (p651)	5.8m	ee, 🌉	157K	-49.25 🚾 1,179				
Parallel Server 9 (p062)	6.6m	L09	150K					
Paralel Server 10 (p053)	5. Set	E.14	157K	One DV conver concursing all the				
Porstal Server L1 (p054)	1.6m	<b>1</b> 4	157K	One PA server consuming all the				
Availet Berver 12 (p095)	1. Area	- AY	257K	time: 52 minutes vs 5.0 minutes				
-Ruslei Server 13 (p366)	1. See	<b>20</b> .00	1944	time. 55 minutes vs 5.9 minutes				
Paralel Server 14 (p057)	1. tra	1.00						
Penalel Server 15 (p599)	🗮 5.8m	1.24	1 TAIN.	-49.25 1,179				
Ponelial Server 16 (p059)	5.8m	1.23	🛼 2576	-49.25 🔤 1,179				
Poralei Server 17 (p020)	1.5m		ESTK	-49.25 21.79				
Paralel Server 18 (p071)	5.8m	1.11	196A	-49.25 1,179				
Ponilel Server 19 (p072)	N 5 911	<b>1.02</b>	E 1976	-97.23				
Panadel Berver 20 (p073)	🔛 \$3.6m		3,639K	-49.2) 🔛 1,179				
Poralei Server 21 (p074)	3.841	25	LINK	-97.2) 🔤 1,179				
Parallel Server 22 (p075)	5.80	1.2	<b>15</b> 76	-49.25 🔛 1,179				
Porulal Barrose 23 (p076)	1 5.8m	<b>11</b>	E LATK	-99.21 🔤 1,178				
Ponsilet Server 24 (p077)	5.84	<b>1</b> 44	158K	-49.25 3.179				

#### ORACLE

REAL-WORLD PERFORMANCE

### Example 3 Diagnosis

- Some domain knowledge was applied to this problem
  - The application supports loyalty cards
  - Majority of the customers ids are unknown or not identifiable
    - Implemented as a userID = -1
  - This value from the CUST\_IDENTIFIER\_DIM table hashes to a single hash bucket and so the HASH JOIN for that value is executed by a single PX slave



### Example 3 Diagnosis

- Some domain knowledge was applied to this problem
  - The application supports loyalty cards
  - Majority of the customers ids are unknown or not identifiable
    - Implemented as a UserID = -1
  - UserID= -1 hashes to a single hash bucket
  - HASH JOIN for UserID= -1 is executed by a single PX slave



#### Example 3 Solution(s)

- Have the unknown customers be identified by a negative sequence number
- Run two separate queries
  - A query for known customers, ie CUST\_IDENTIFIER\_ID > 0
    - This query executes with HASH HASH distribution, like the original
  - A query for unknown customers CUST\_IDENTIFIER\_ID = -1
    - This query executes with a BROADCAST distribution
- Similar to above
  - Write the query as a UNION-ALL of the above
- In 12c, SKEW detection may generate a HYBRID plan that will automatically broadcast popular values, and Hash distribute non-popular values





## SQL Monitor gotcha's

- Beware of cardinality estimates of 1 except for unique index lookup
- Execution count
  - Parallel operations
  - Partitioned operations
  - Nested loop
  - Or combination!
- For Nested loop plan
  - Estimated rows is for a single execution
  - Actual rows is for all executions (so far)
- Parallel server elapsed time bars not a good indication of time
- Fixed size in memory buffer means statements age out
- Plan lines limit defaults to 300
  - \_\_sqlmon\_max\_planlines to change



## SQL Monitor Conclusions

SQL Monitor is the best tool to understand why SQL is running slowly

- Always on
- Detailed data on where the time is going
- Shows cardinality estimates vs actuals
- Top SQL captured into AWR in 12c



#### Program Agenda

1 Why Autonomous Database?

- <sup>2</sup> SQL Performance Tuning Strategy
- <sup>3</sup> Makes Tuning Smarter





#### PANEL











#### Safe Harbor Statement

The preceding 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.



## Integrated Cloud Applications & Platform Services



ORACLE®