ORACLE

NoCOUG Virtual Spring Conference 2020

Understanding the Impact of Cloud Networking on Your Database Applications

Mark Ashdown
Real-World Performance Team
Oracle Database Development

ORACLE
REAL-WORLD PERFORMANCE

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, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Real-World Performance

Who We Are

Part of Oracle Database Development

Team members at HQ and in the USA, Europe and Asia

Over three hundred years of experience combined

What We Do

Use the product as designed

Aim for the best performance

Apply data-driven analysis

Avoid guesswork

Share what we learn

Lift and Shift

On-Premises

Cloud

Browser

Application

Database

Reality Check

Some components will not move to Cloud

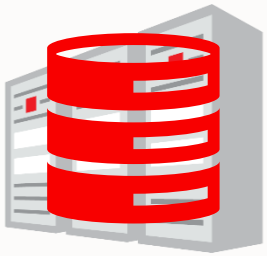
Few enterprise applications exist in isolation

- Almost all enterprise applications have multiple integration points
- Not all components will move to Cloud simultaneously, or even any time soon

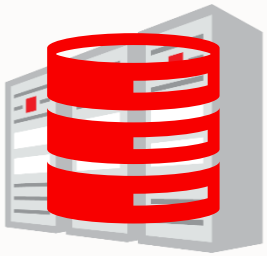
A longer network hop will be introduced

One Data Centre

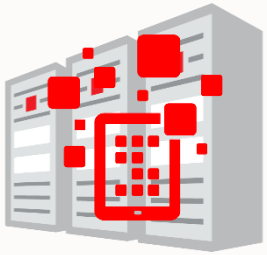
Two Data Centres



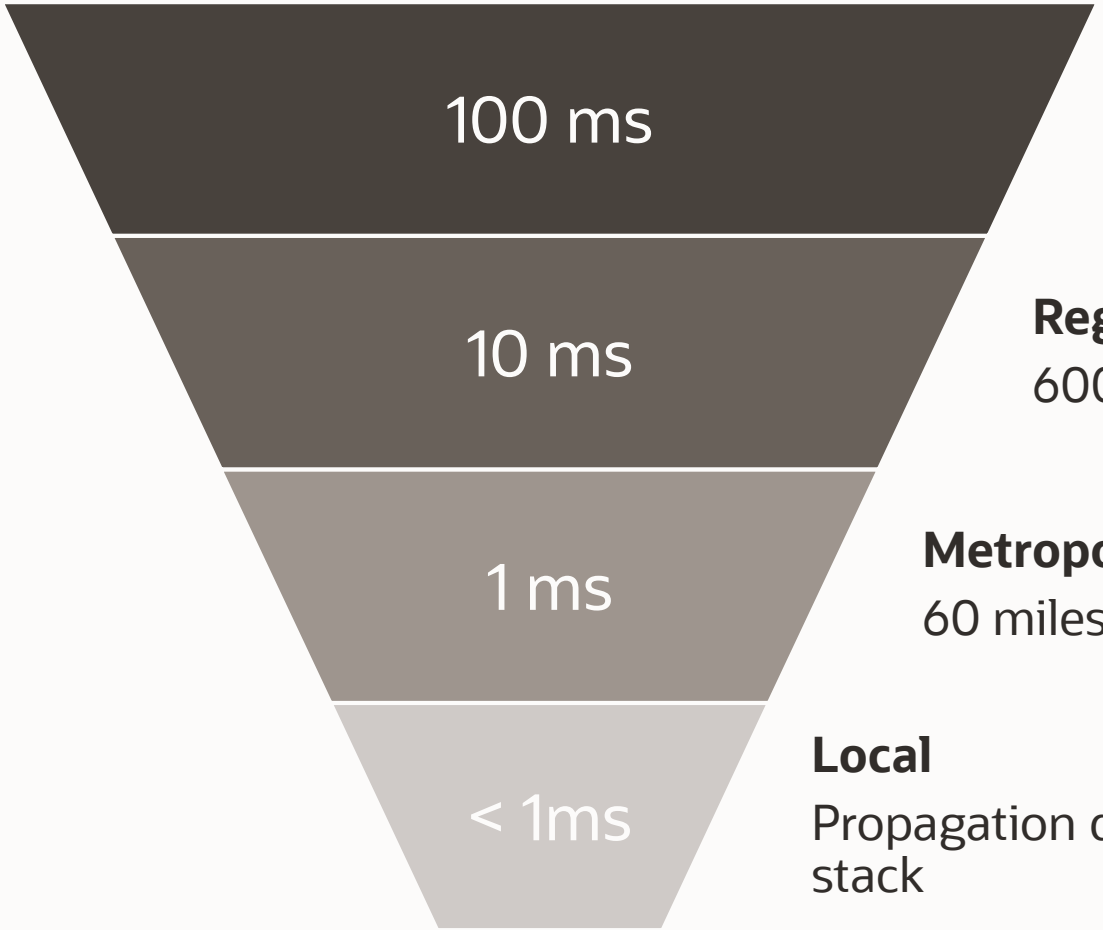
Two Cloud Providers



Database in Cloud



Round Trip Time with Distance



International
6000 miles / 10000 Kilometres

Regional
600 miles / 1000 Kilometres

Metropolitan
60 miles / 100 Kilometres

Local
Propagation delay is small relative to other costs in the network stack

The Impact of Increased Round Trip Time

From the application perspective, the database appears slower

From the database perspective, database sessions wait longer for work

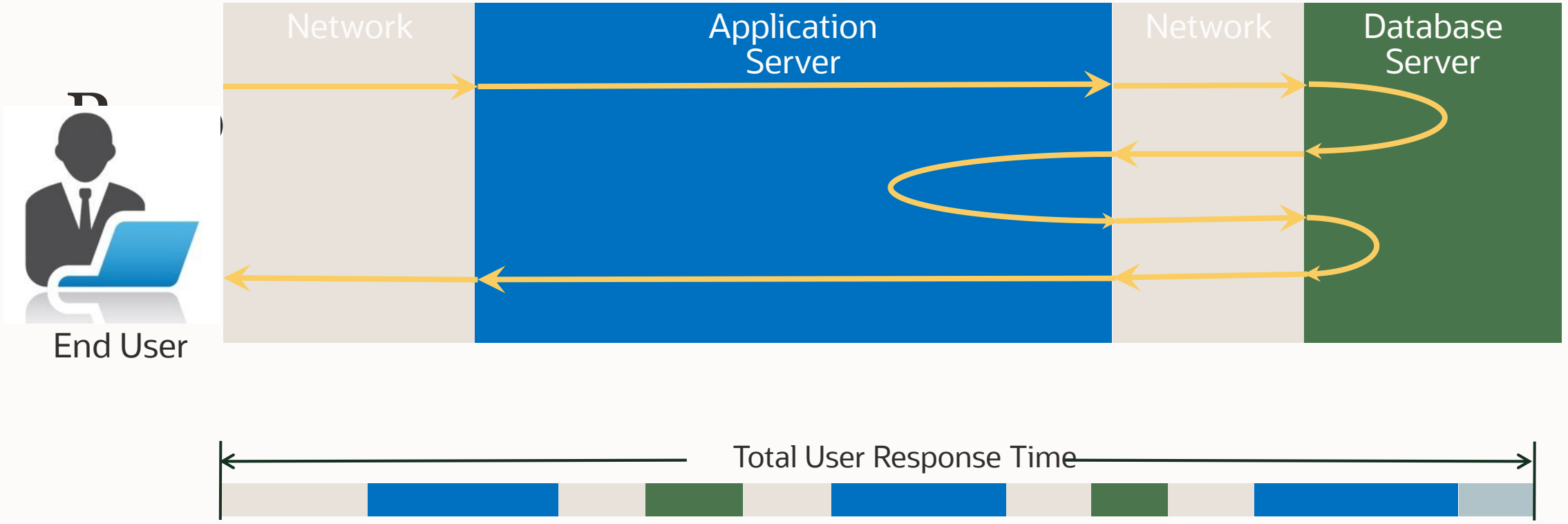
- Locks may be held for longer
- Reduces application scalability

More database sessions will usually be needed to support the same throughput

- Impacts efficiency and stability

Application **Database**

Where does my time go?



DEMO

Geography Lesson for the UK folks

London to Frankfurt = 396 Miles

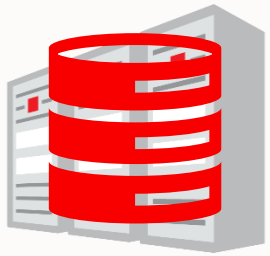
San Francisco to Eugene ON= 435 Miles

SwingBench OLTP Workload

London

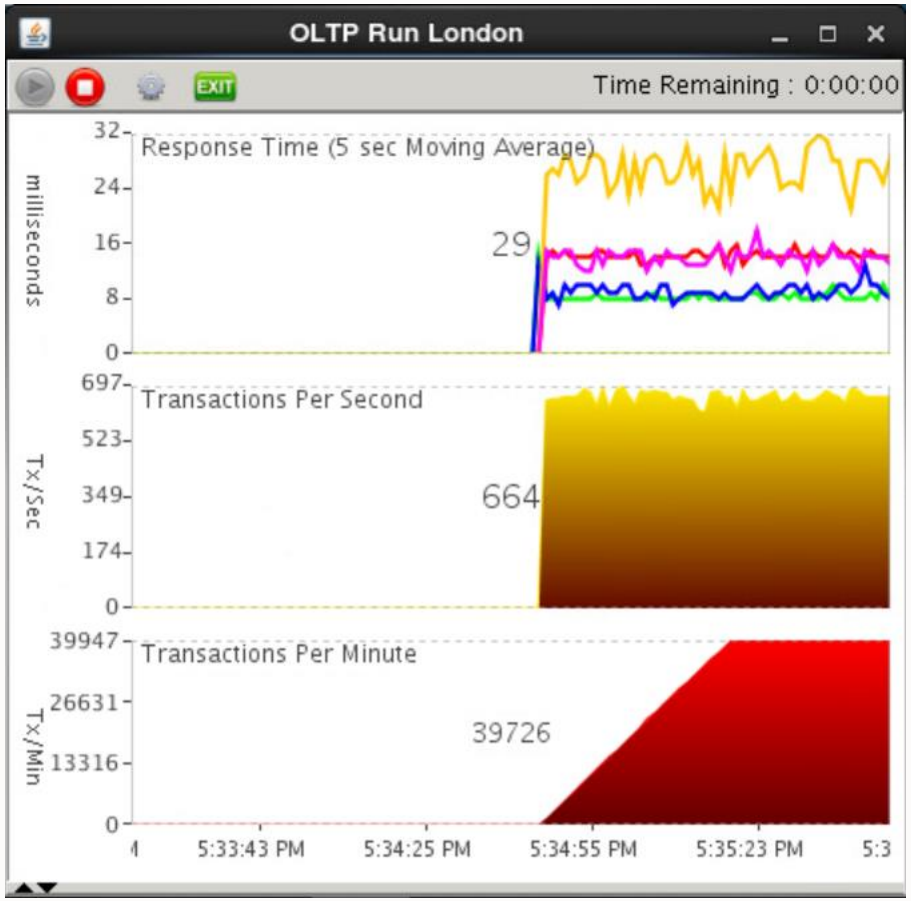
Frankfurt

Multiple Sessions



SwingBench OLTP Workload

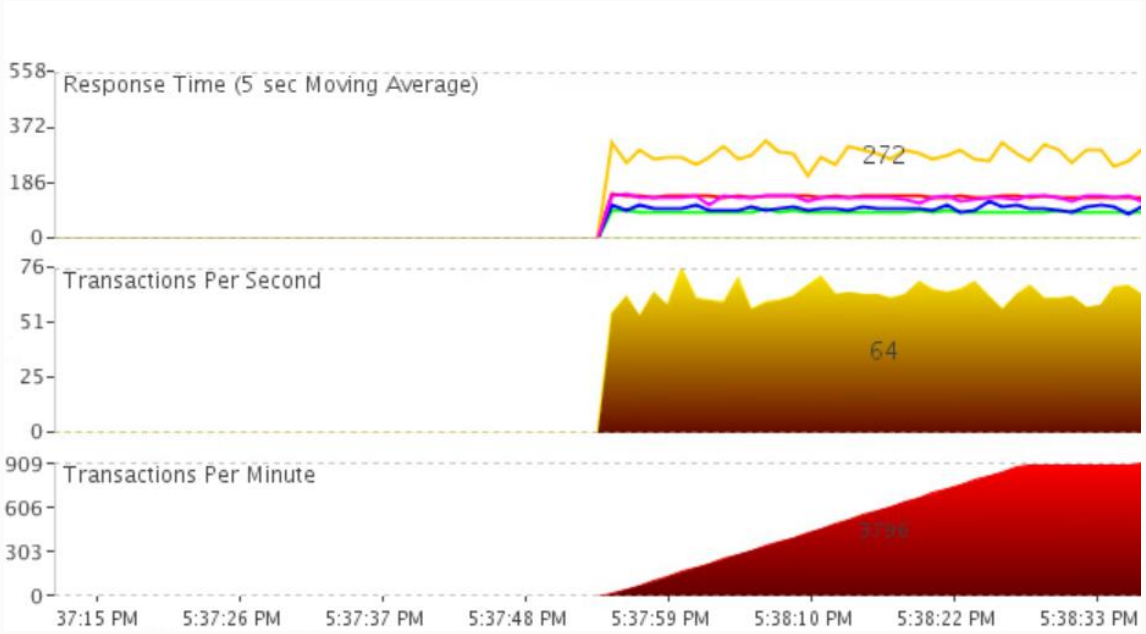
Database in London with 10 Sessions



Metric	Value
Response Time	29 milliseconds
Throughput	664 transactions per second
Throughput	39726 transactions per minute

SwingBench OLTP Workload

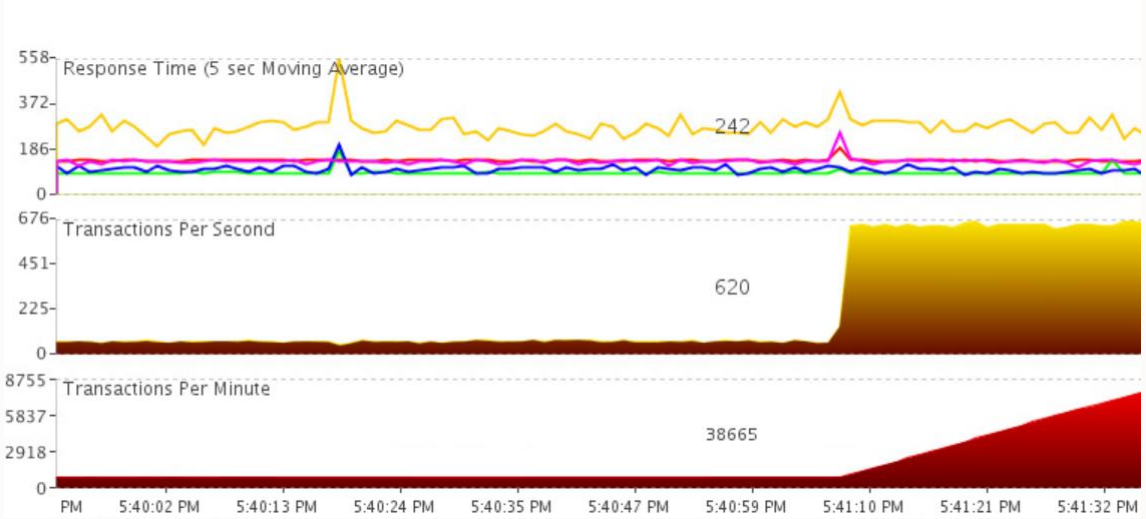
Database in Frankfurt with 10 Sessions



Metric	Value
Response Time	272 milliseconds
Throughput	64 transactions per second
Throughput	3796 transactions per minute

SwingBench OLTP Workload

Database in Frankfurt with 100 Sessions



Metric	Value
Response Time	242 milliseconds
Throughput	620 transactions per second
Throughput	38665 transactions per minute

Lessons Learned

Longer network round trip

- Increases response time by 10x
- Reduces throughput by 10x

Increasing the number of sessions by 10x

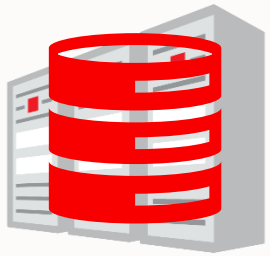
- Restores throughput
- Does not restore response time

The number of sessions will tend to increase automatically in an application using connection pooling

SwingBench Batch Workload

London

One
Session



Frankfurt

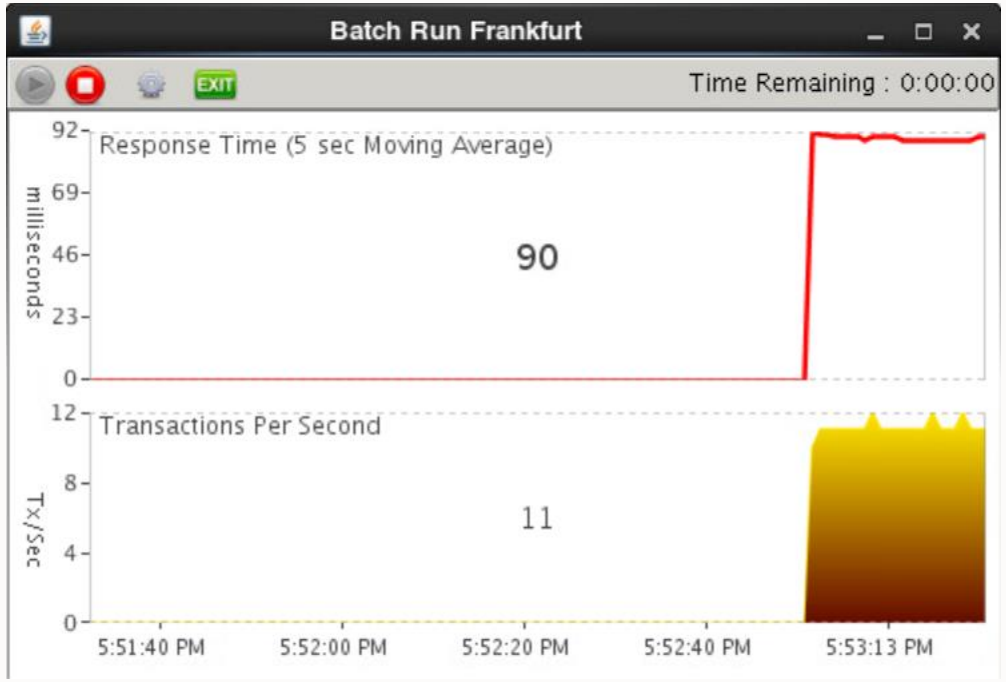
SwingBench Batch Workload

Database in London with One Session

Metric	Value
Response Time	9 milliseconds
Throughput	103 transactions per second

SwingBench Batch Workload

Database in Frankfurt with One Session



Metric	Value
Response Time	90 milliseconds
Throughput	11 transactions per second

Lessons Learned

Longer network round trip

- Increases response time by 10x
- Reduces throughput by 10x

There may be no simple solution

Single-threaded or serialized components of your workload may cause unexpected problems

Example

- Application Moved from On-Premise to cloud
- Nothing else changed (application version, Database version statistics etc etc (we hope)
- Online OK performance
- Overnight batch sucks
- Everything initiated from the Application Server

AWR Load Profile

On Premise

	Per Second	Per Transaction	Per Exec	Per Call
DB Time(s):	0.6	10.9	0.00	0.00
DB CPU(s):	0.5	8.9	0.00	0.00
Executes (SQL):	2,078.5	35,715.0		
Rollbacks:	0.0	0.0		
Transactions:	0.1			

Conclusions

- 0.1s/sec waits on premise
- No waits on cloud
- Executes on cloud 25% of that on premise

On Cloud

	Per Second	Per Transaction
DB Time(s):	0.2	7.5
DB CPU(s):	0.2	6.0
Executes (SQL):	538.1	21,573.0
Rollbacks:	0.0	0.0
Transactions:	0.0	

AWR Top Wait Events

Event	Waits	Total Wait Time (sec)	Wait Avg(ms)	% DB time	Wait Class
DB CPU		311.4		81.9	
SQL*Net message to client	2,083,678	2.7	0.00	.7	Network
cursor: mutex X	183	1.2	6.61	.3	Concurrency
cursor: pin S	628	1.1	1.78	.3	Concurrency
SQL*Net more data to client	14,339	.6	0.04	.1	Network
name-service call wait	4	.3	72.48	.1	Other
PX Deq: Slave Session Stats	388	.3	0.65	.1	Other
Disk file Mirror Read	31	.1	3.70	.0	User I/O

Event	Waits	Total Wait Time (sec)	Wait Avg(ms)	% DB time	Wait Class
DB CPU		90.6		80.5	
control file sequential read	3,088	1.2	0.39	1.1	System I/O
name-service call wait	16	1.2	73.30	1.0	Other
SQL*Net message to client	534,508	.7	0.00	.6	Network
Disk file Mirror Read	643	.4	0.65	.4	User I/O
PX Deq: Slave Session Stats	558	.2	0.43	.2	Other
enq: PS - contention	326	.2	0.60	.2	Other
SQL*Net more data to client	3,667	.1	0.04	.1	Network

AWR Statistics

On Premise	Total	Per Second
SQL*Net roundtrips to/from client	2,085,452	3,467.67
bytes received via SQL*Net from client	689,050,423	1,145,745.87
bytes sent via SQL*Net to client	2,051,208,421	3,410,728.02
bytes via SQL*Net vector to client	0	0.00

Cloud	Total	Per Second
SQL*Net roundtrips to/from client	534,621	888.96
bytes received via SQL*Net from client	177,310,083	294,827.40
bytes sent via SQL*Net to client	525,160,500	873,225.61
bytes via SQL*Net vector to client	0	0.00

SQL Statistics Executions

Executions	Rows Processed	Rows per Exec	Elapsed Time (s)	%CPU	%IO	SQL Text
106,357	0	0.00	7.53	22.4	0	SELECT * FROM MARK.C0005 WHER...
106,357	106,357	1.00	10.03	34.2	0	SELECT * FROM MARK.C98810 WHE...
106,357	106,357	1.00	17.73	24.3	0	SELECT * FROM MARK.C98865 WHE..

Estimating the Impact
Basic Ingredients

Workload

- A little knowledge of the application goes a long way

AWR Report

- DB Time
- SQL*Net roundtrips to/from client
- You can use totals or per second values but do not mix the two

Network

- Find the increase in round trip time
- Standard network tools can be good enough
- Do not use tnsping!

Estimating the Impact
The Recipe

Consider the application perspective

- The application does not see DB Time
- The application sees DB Time plus network time

We can use the basic ingredients to estimate the increase in network time

- [roundtrips] x [increase in round trip time]

We can now compare this value with DB Time

- If DB Time had increased to the same extent, would the DBA have had a busy day?

Additional Time << DB Time

Additional Time >> DB Time

Estimating the Impact
Problem Scenarios

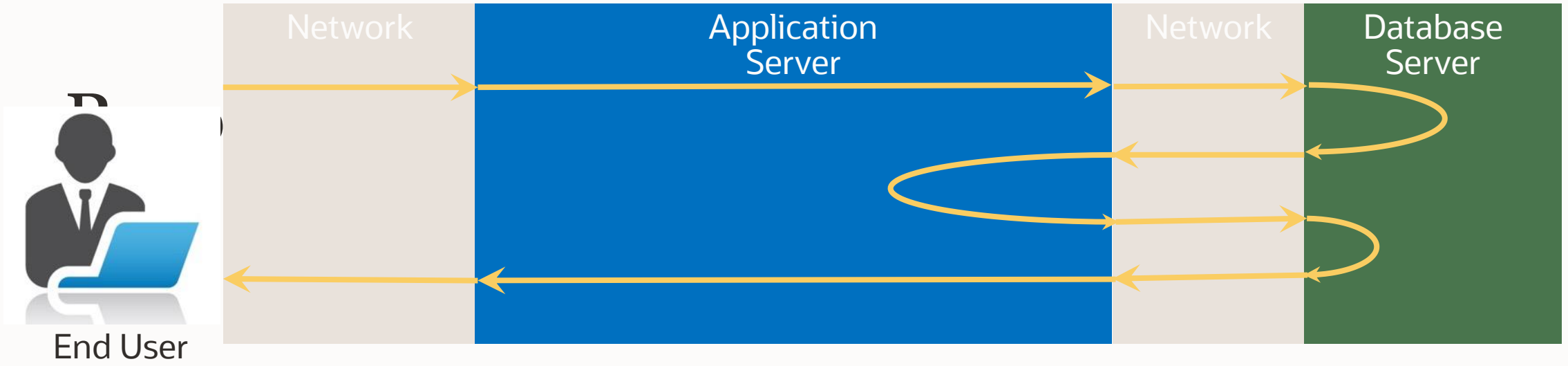
What happens if the application was designed to use a single worker process?

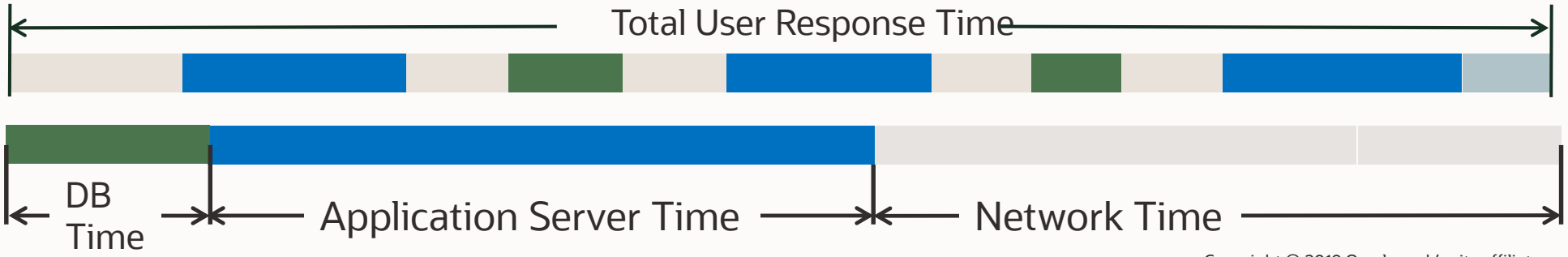
What happens if the application was designed to use multiple worker processes but has a serialization point?

- Perform some SQL
- `UPDATE CTRL SET N = N + 1`
- Perform some more SQL
- `COMMIT`

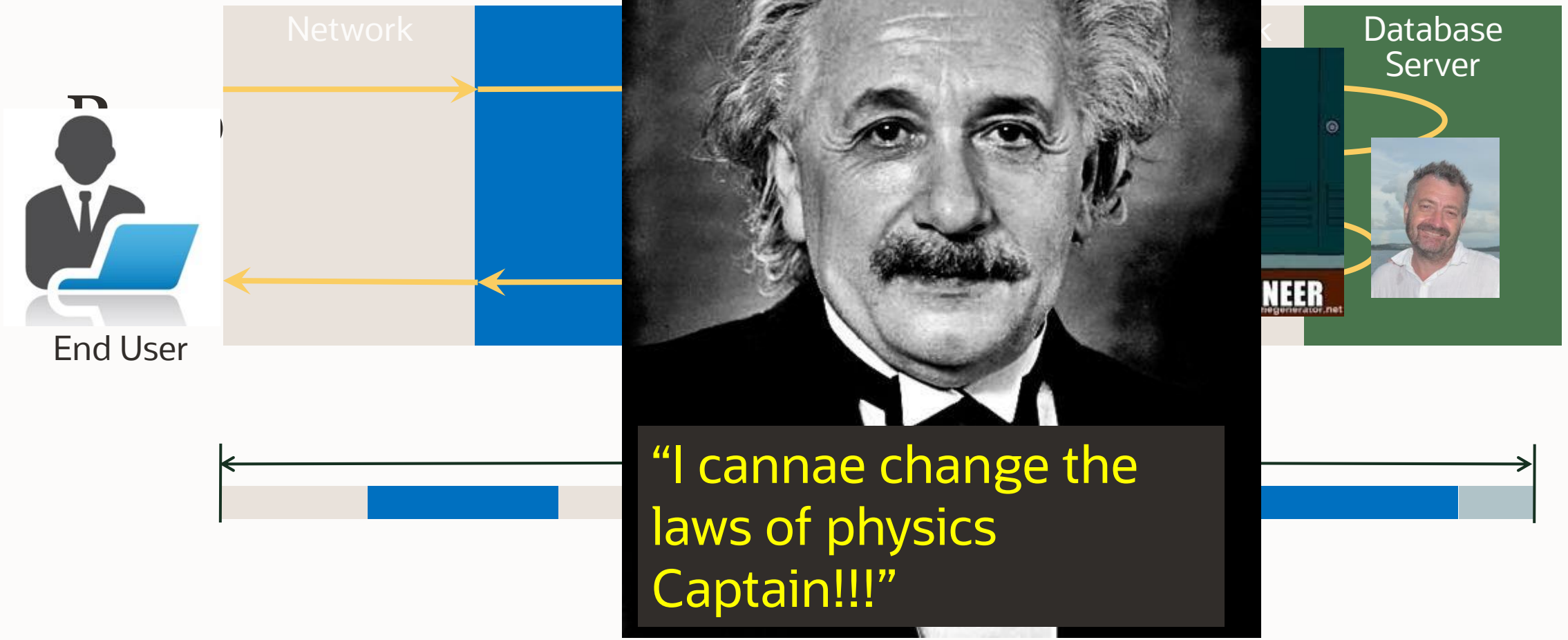
You may be able to buy bandwidth. The speed of light is non-negotiable.

Where does my time go?



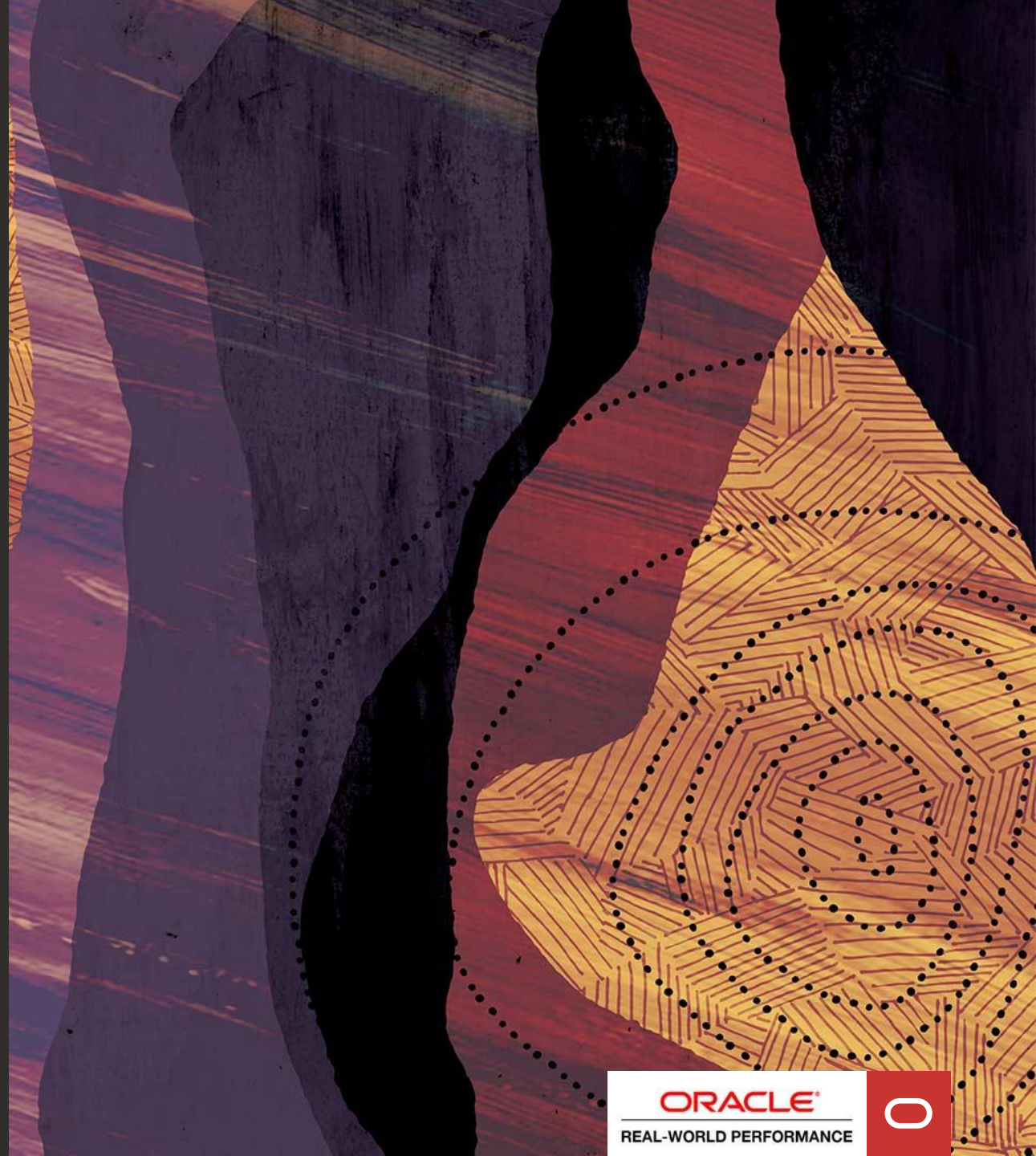


Who's to blame?



Thank you

Mark Ashdown
Real-World Performance Team
Oracle Database Development



ORACLE


Time for
a BREAK