



Performance: Several roads leads to Rome – a performance note
Lasse Jenssen



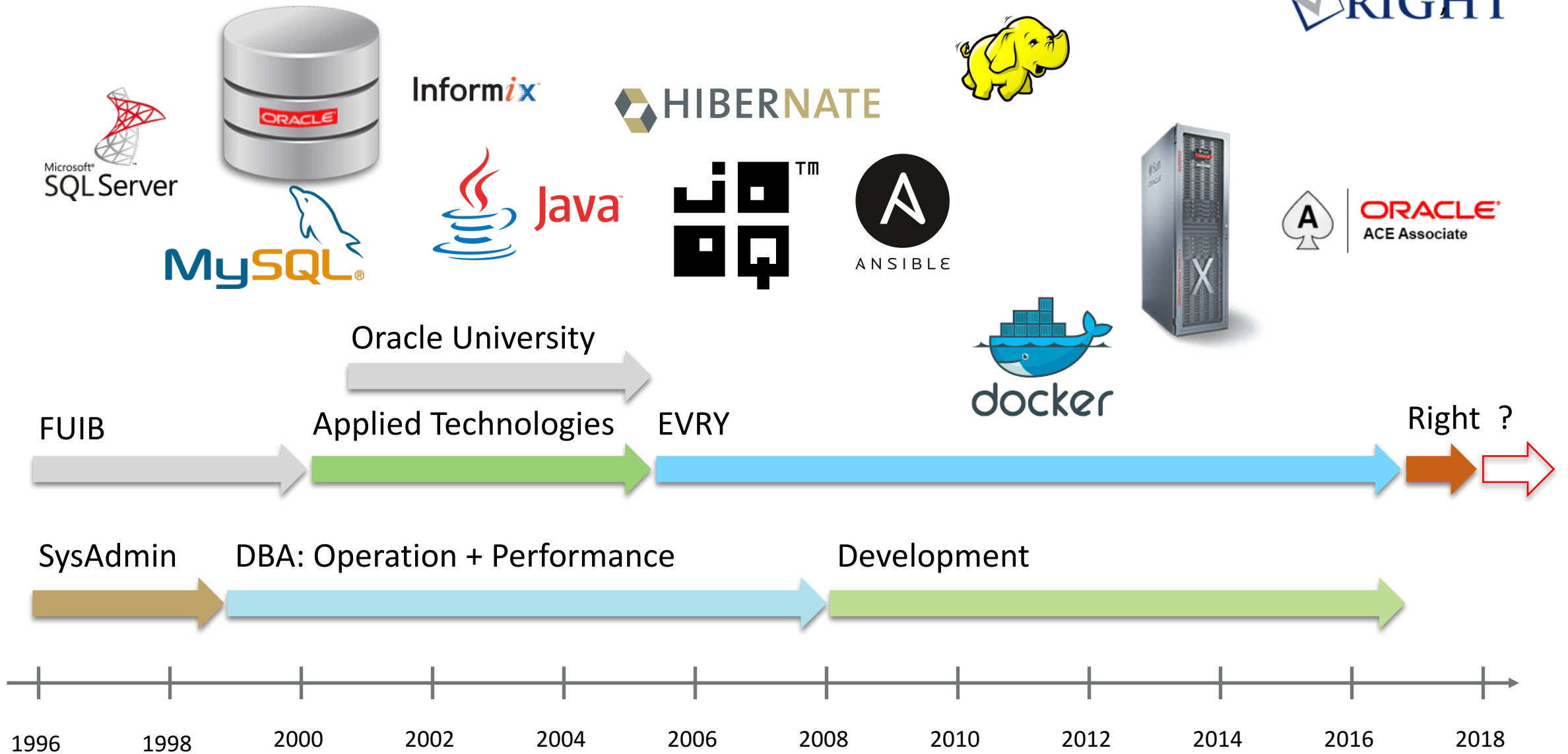


Lasse Jenssen

www.jcon.no/oracle

@lasjen lasse.jenssen@rightgroup.no







Statens vegvesen



Kollektivtransportproduksjon AS



bouvet



EVRY



Lasse Jenssen

Bergen





Performance: Several roads leads to Rome – a performance note
Lasse Jenssen

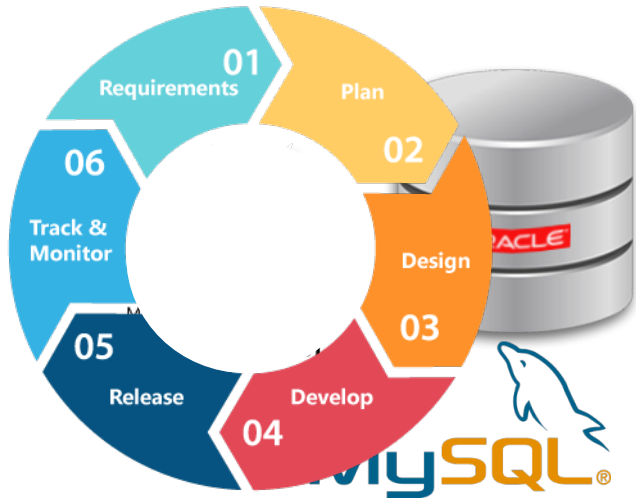


When was YOUR last trip to Rome?



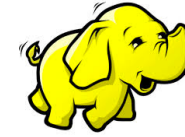
What was your last action to increase performance?





Informix

HIBERNATE



RIGHT



Oracle University



FUIB

Applied Technologies

EVRY

Right ?

DBA: Operation + Performance

Development

2000

2002

2004

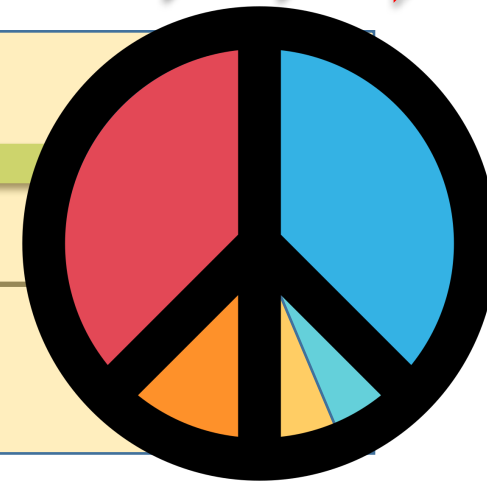
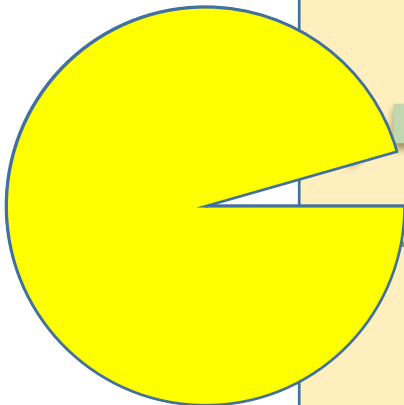
2006

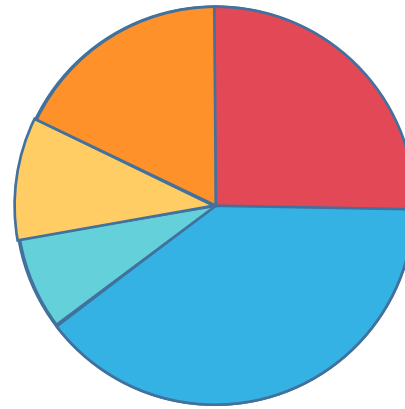
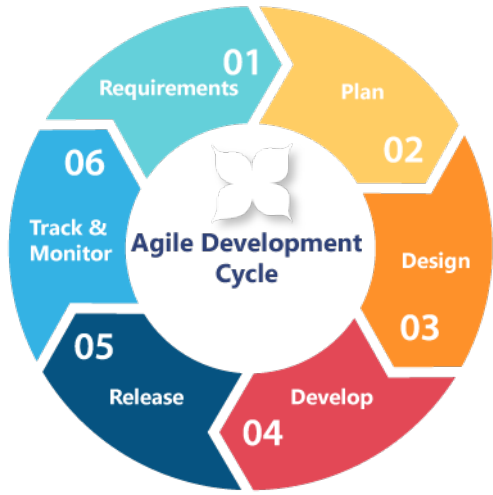
2008

2010

2012

2014

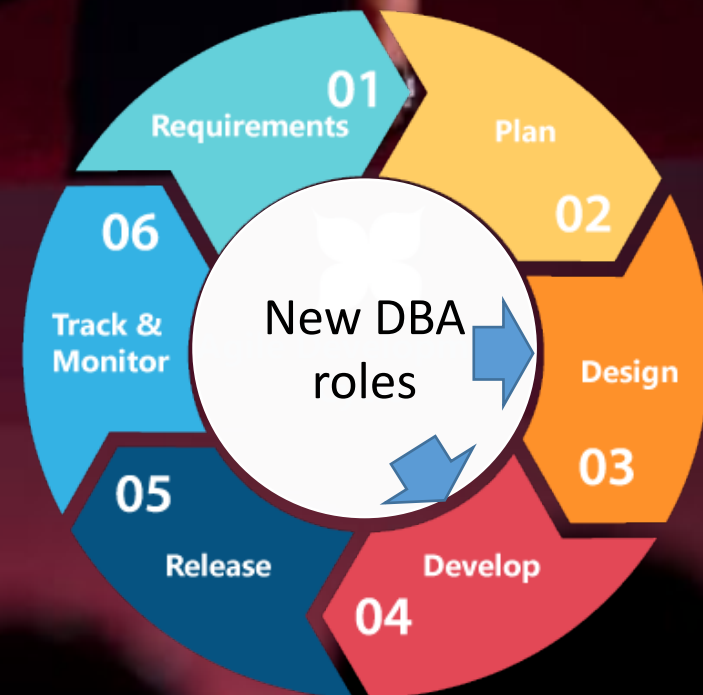






Announcing Oracle 18c

World's First 100% Self-Driving **Autonomous Database**



EOD batch:

SLA:

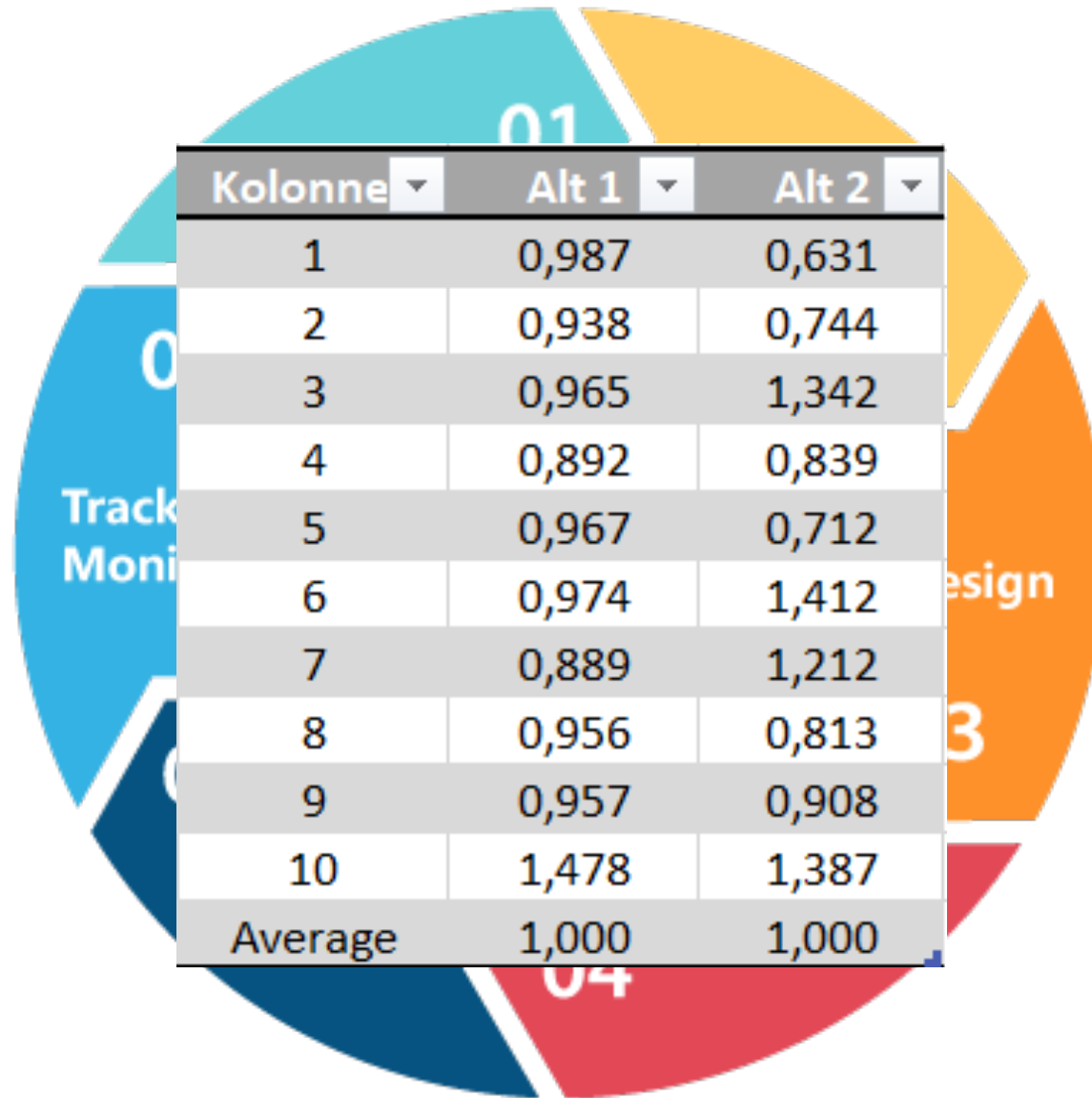
- 50 million accounts
- 10 million users

Online payment:

SLA: avg(RT) < 1 sec

Biggest Challenge:

How do you get there?



EOD batch:

SLA:

- 10 mill payments per day
- Batch window: 4 hours

Online payment:

SLA:

Percentile99(R) < 1 sec

Max(RT) < 2 sec

Job:

Make sure specifications include proper performance requirements.

Identify critical areas in specification.

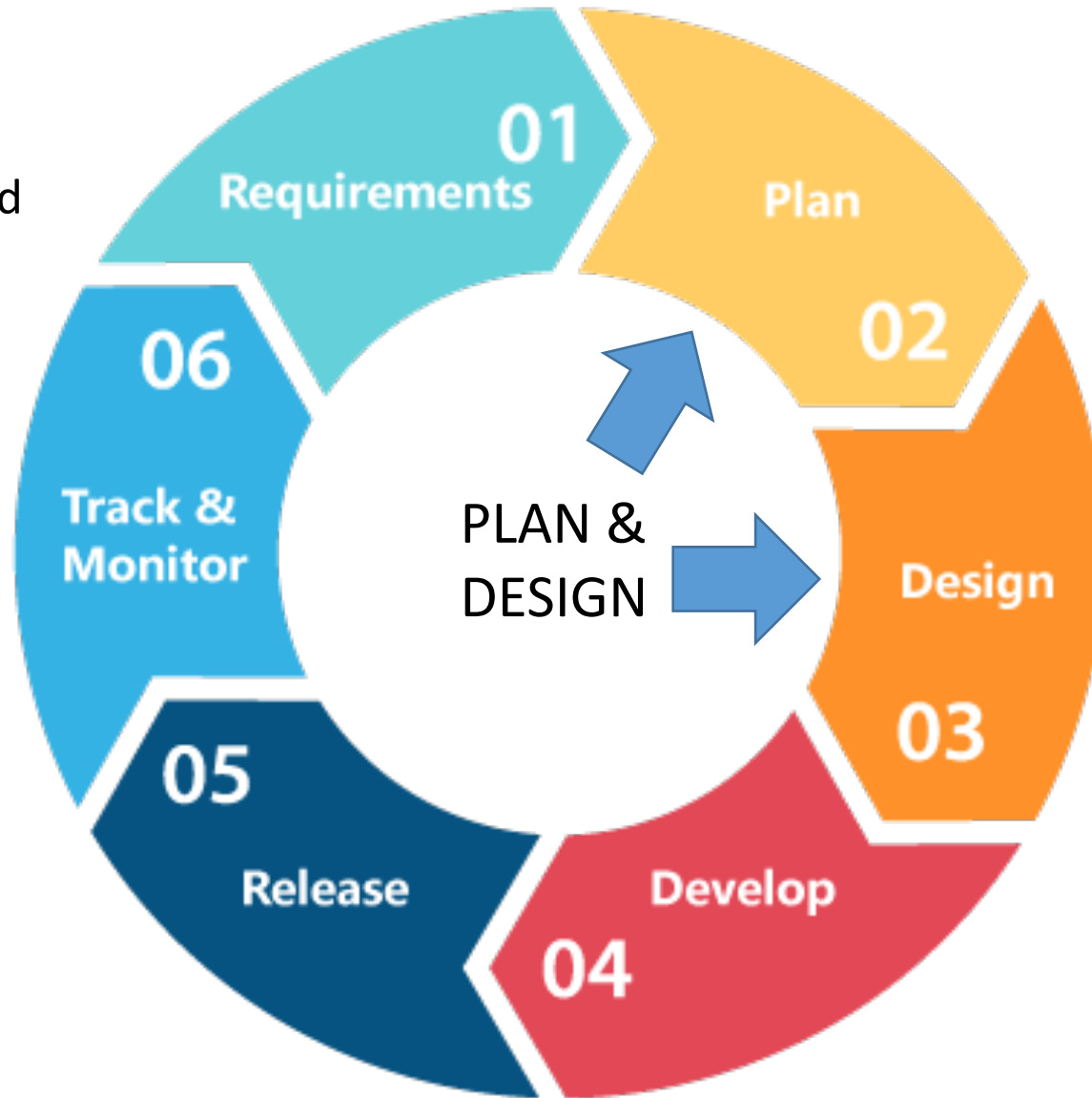
Biggest Challenge(s):

How do you get there?

How do you get allocated enough time?

How do you get a good communication with the architects and developers?

How do you get allocated time for testing in early stages?



What to do:

Evaluate plans and designs.

Suggest:

- Alternative solutions.
- Design patterns (for instance use of PLSQL)
- Instrumentation

Assist in design:

- Database modeling
- Database architecture (don't be religious)
- Database interfaces (PLSQL?)

Assist in testing:

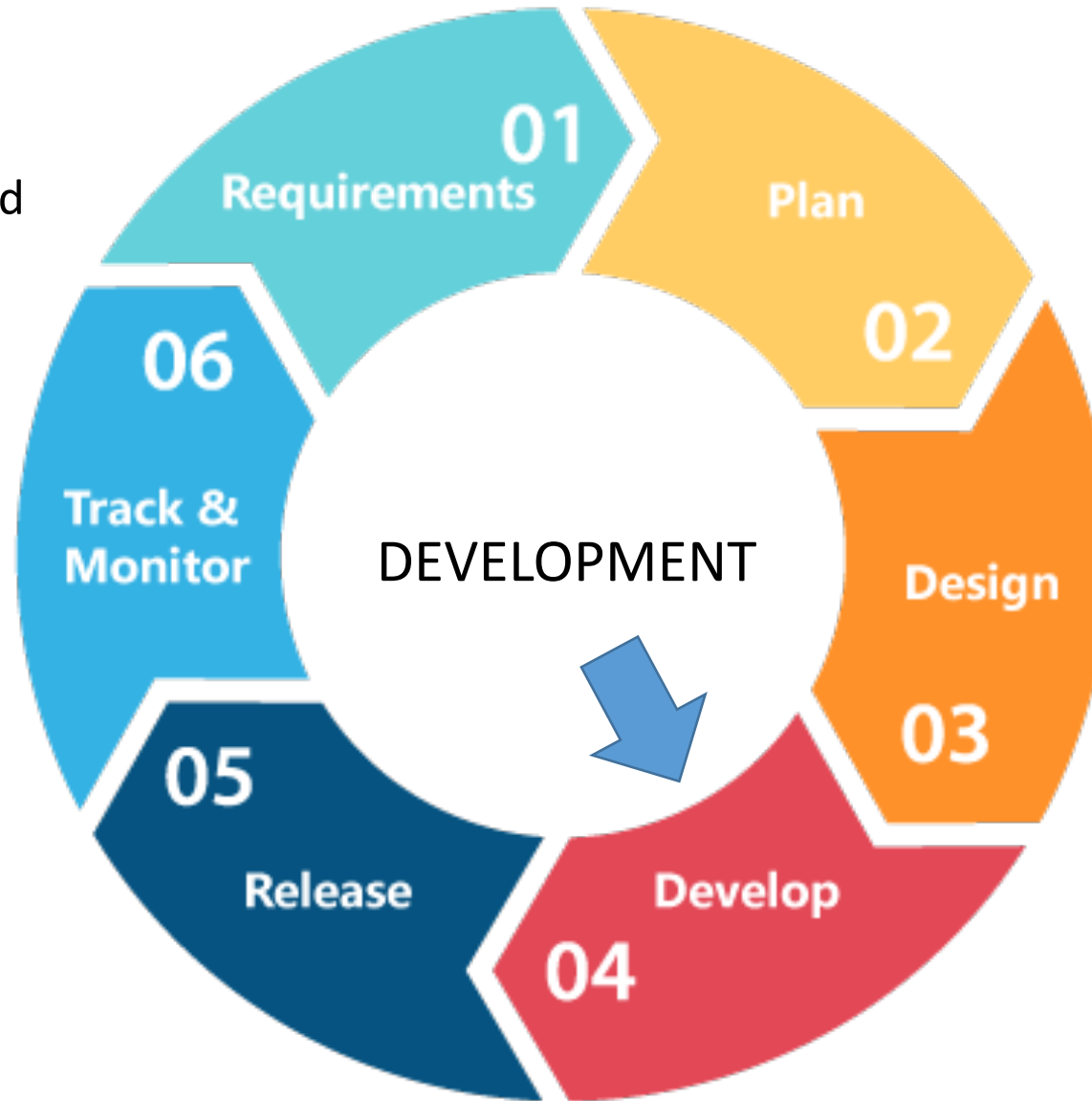
Test (POC) for your critical areas.

Challenges:

How do you get there?

How do you get allocated enough time?

How do you get a good communication with the architects and developers?



What to do:

Implement database and/or database changes:

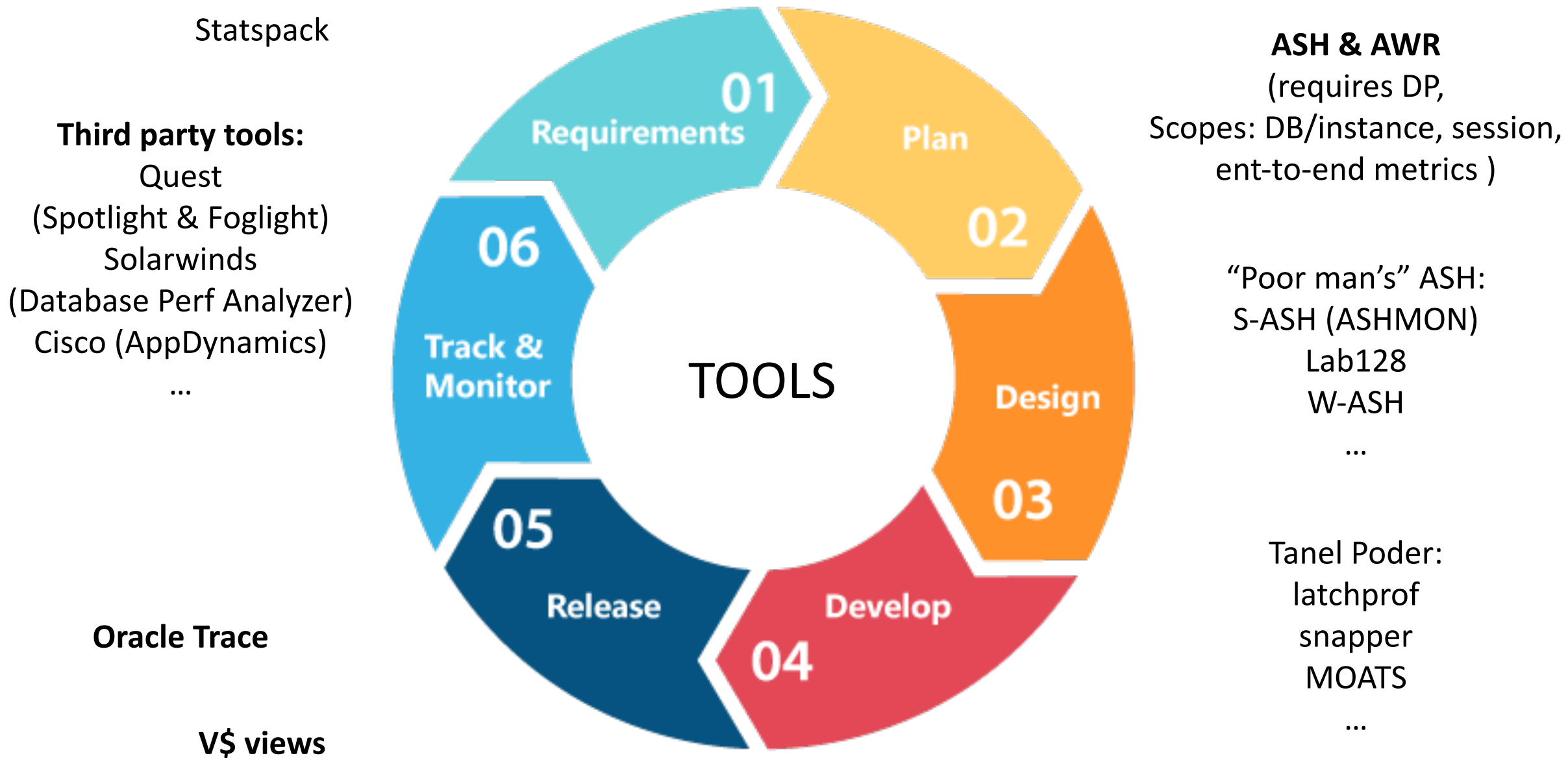
- Users & roles
- Tables (provisioning?)
- Constraints
- Indexes
- Interfaces (PLSQL)

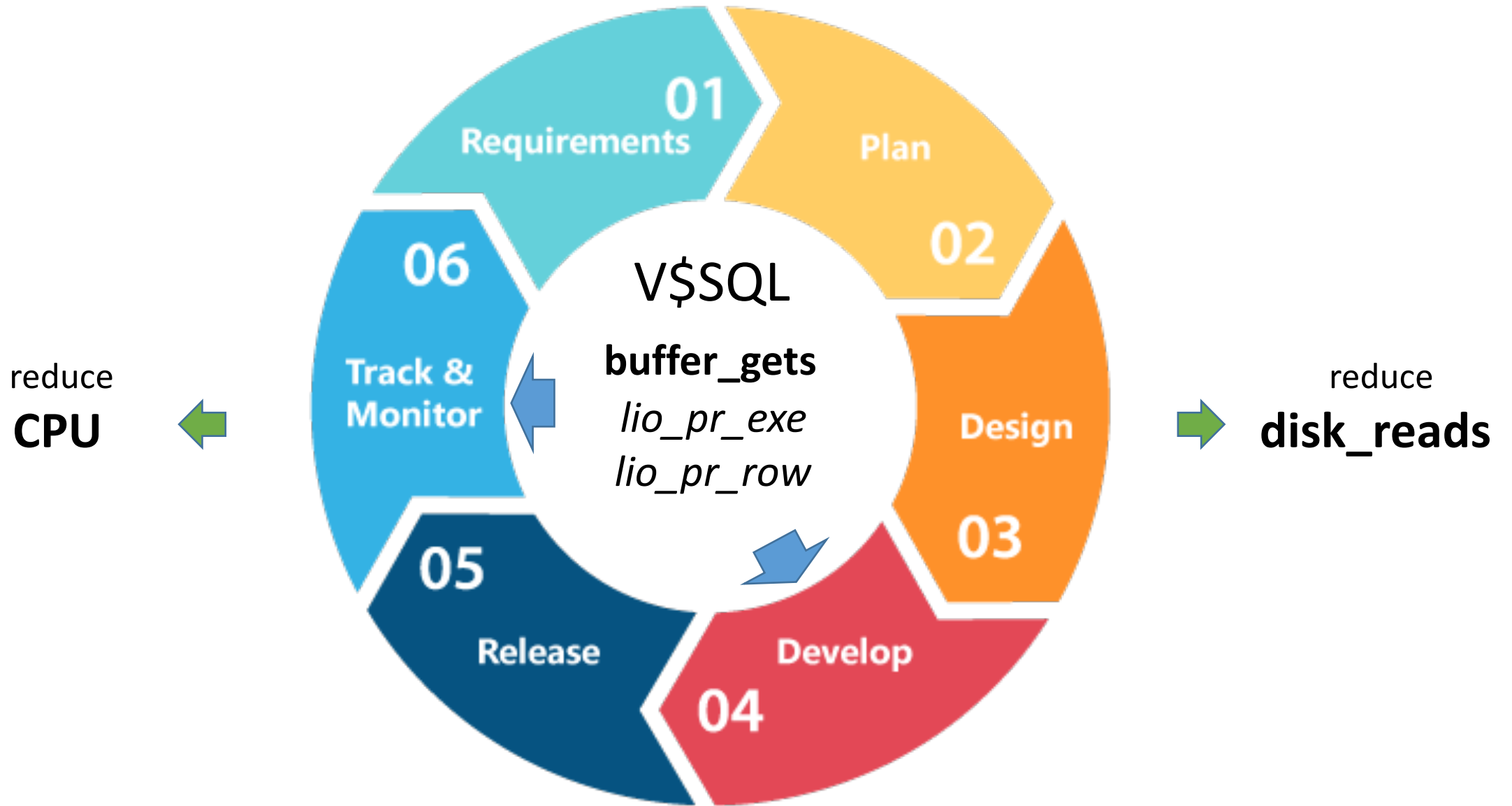
Adviser:

- Be a visible team player for your developers
- Instrumentation

Assist in testing:

- Individual features
- System overall (scalability)





```
select
  sql_id, parsing_schema_name usr,
  round(s.buffer_gets/ case when s.executions = 0 then 1
                        else s.executions
                        end, 2)          lio_pr_exe,
  round(s.buffer_gets/
        case when s.rows_processed=0 and s.executions=0 then 1
              when s.rows_processed<s.executions          then s.executions
              else s.rows_processed end, 2)          lio_pr_row,
  buffer_gets lio, executions nr_exe, rows_processed nr_rows, sql_text
from v$sql s
where 1=1
  and parsing_schema_name not in ('SYS', 'SYSTEM')
  and sql_text not like 'DECLARE job BIN%'
  --and sql_id not in ('6mcpb06rctk0x', '5h5tk3d5gszky')
  --and parsing_schema_name='HR'
order by 3 desc;
```

SQL_ID	USR	LIO_PR_EXE	LIO_PR_ROW	LIO	NR_EXE	NR_ROWS	REPLACE(SQL_TEXT,'NO1.')
5jdbk0f1xu8pw	DATADOG	16129,96	16129,96	1935595	120	120	begin datadog_metric_p.sql_child
bs9txzrv424x0	SYSTEM	7003,21	7003,21	5350456	764	0	SELECT owner,index_name,table_n
4fb5qc54x7kw7	DATADOG	2646	2646	2646	1	0	INSERT INTO SQL_CHILD_PERFORMAN
5h5tk3d5gszky	SYSTEM	57876,35	2226,01	1157527	20	520	select nvl(autoextent,
f0a1c90fu7jtd	SYSTEM	2003	2003	2003	1	1	SELECT /*+ opt_param('parallel_e
ar7smgdzuupmj	AGR	625,17	625,17	3866054	6184	0	select delayedtas0_.id as id1_1
24qcr5b7xfz5u	SYSTEM	404,5	404,5	809	2	0	SELECT 'ARGUMENT' type, null ow
0gqqfscuz3bz3	DATADOG	373	373	373	1	1	SELECT COUNT (*) FROM SQL_CHILD
5skyu4hf1ytqf	PI	352,05	352,05	127192844	361296	85988	select distinct fixedtimes0_.TIM
0cv0mkqxs6ybt	PI	286	286	286	1	1	insert into MessageLog (fileName
c0nfc546tsvgu	SYSTEM	275	275	275	1	0	select case type# when 7 then '1

```
set pages 999 lines 200
```

```
SELECT *
```

```
FROM table(dbms_xplan.display_cursor('5skyu4hf1ytqf',null,'TYPICAL'));
```

op3cbg37gnvxe	SYSTEM	200,0	200,0	3004207	24303	24303	select (a.datasize+b.compsize+c
4a2u22taxy28y	KOT	205,54	205,54	212943	1036	1036	select fixedtimes0_.TIMESERIE_I
5ud7r17twxc30	KOT	196,24	196,24	1338577	6821	2972	select distinct fixedtimes0_.TIM

```
select
  sql_id, parsing_schema_name usr,
  round(s.buffer_gets/ case when s.executions = 0 then 1
                        else s.executions
                        end, 2)          lio_pr_exe,
  round(s.buffer_gets/
        case when s.rows_processed=0 and s.executions=0 then 1
              when s.rows_processed<s.executions          then s.executions
              else s.rows_processed end, 2)          lio_pr_row,
  buffer_gets lio, executions nr_exe, rows_processed nr_rows, sql_text
from v$sql s
where 1=1
      and parsing_schema_name not in ('SYS', 'SYSTEM')
      and sql_text not like 'DECLARE job BIN%'
      --and sql_id not in ('6mcpb06rctk0x', '5h5tk3d5gszky')
      --and parsing_schema_name='HR'
order by 6 desc; -- executions
```

SQL_ID	USR	LIO_PR_EXE	LIO_PR_ROW	LIO	NR_EXE	NR_ROWS	REPLACE(REPLACE(REPLACE(REPLACE(SQL_TE
bunvx480ynf57	BIT	0	0	6016	294339587	294268247	SELECT 1 FROM DUAL
bjtkvmar6bnvv	KOT	0,2	0,2	3012509	14939211	14939211	select hibernate_sequence.nextval
5gu4ft3hnhqb4	PI	16,68	16,68	181826315	10900697	10900697	insert into TCOSerieData (CREATED
anuz14psswc3w	PI	3	1,38	25795671	8591187	18750157	select entitytype0_.TIMESERIETYPE_1
apqg5dlxku6sm	PI	1	1	8605304	8591179	1415	select permission0_.TIMESERIETYPE_1
bt2hg2hx3h8ur	PI	3,02	1,4	25278808	8369027	18077761	select entityref0_.TIMESERIE_ID as
44s7s2cgr679f	KOT	8,08	1,8	59091766	7315073	32805321	select task0_.id as id1_41_0_, task
2jusrrs6glvgb	BIT	16,68	16,68	121343510	7275721	7275721	insert into TCOSerieData (CREATED
751kuz38x2k16	PI	2	2	8777643	4382825	4382824	select timeseries0_.TIMESERIETYPE_1
5my6zq5mc0q86	PI	10,01	10,01	42117244	4208362	4208354	select abstractti0_.TIMESERIE_ID as
d42q3abqgr8w3	KOT	4,05	0,87	15473515	3823982	17827228	select objectrefs0_.task_id as task
8nafuhjgryxu3	BIT	13,2	13,2	49221083	3728483	3728483	update TCOSerieData set CREATEDBY
5yqlwy5vyhcmb	PI	34,97	0,18	128620742	3678430	731943480	select timeseried0_.TIMESERIEDATA_1
03rw27ypa3usj	CH1	0,21	0,21	734261	3556750	3556749	select hibernate_sequence.nextval
frc7f7wxwtznj	BIT	1,01	1,01	2222196	2207264	44	select permission0_.TIMESERIETYPE_1
ccn0v4mv00p1k	BIT	2,11	0,75	4631024	2199617	6169129	select entitytype0_.TIMESERIETYPE_1
30rkcgndmq288	BIT	3,01	1,08	5890301	1958075	5440682	select entityref0_.TIMESERIE_ID as
g4pwar5jqrk7s	BIT	57,86	1,02	81425198	1407274	79868878	select timeseried0_.TIMESERIEDATA_1



```
CREATE TABLE test AS
  SELECT o1.*, o1.created fromtime, o1.created totime
  FROM all_objects o1
  WHERE 1=2;

CREATE SEQUENCE test_seq START WITH 10000 INCREMENT BY 1 CACHE 100;

INSERT INTO test
  SELECT o1.*,
    to_date('01-01-2000','dd-mm-yyyy') + trunc(dbms_random.value(1,6700)),
    to_date('01-01-2017','dd-mm-yyyy')
  FROM all_objects o1, all_objects o2
  WHERE rownum<=500000;

UPDATE test
SET object_id = test_seq.nextval,
    totime = fromtime + trunc(dbms_random.value(1,100));

CREATE INDEX test_srch01_idx ON test (fromtime, totime, owner);

exec dbms_stats.gather_table_stats(ownname=>'LJ', tabname=>'TEST');
```

```
SELECT *  
FROM test  
WHERE fromtime > to_date(:b1,'dd-mm-yyyy')  
      AND totime < to_date(:b2,'dd-mm-yyyy')  
      AND owner= 'PUBLIC';
```

'12-03-2017'
'29-03-2017',

SQL_ID	LIO_PR_EXE	LIO_PR_ROW	NR_EXEC	NR_ROWS	SQL_TEXT
5z7tvkyh401yf	118	5,76	1	40	SELECT * FROM test WHERE fromtime >

```
set pages 9999 lines 200
select * from
table(dbms_xplan.display_cursor('5z7tvkyh401yf',null,'TYPICAL'));
```

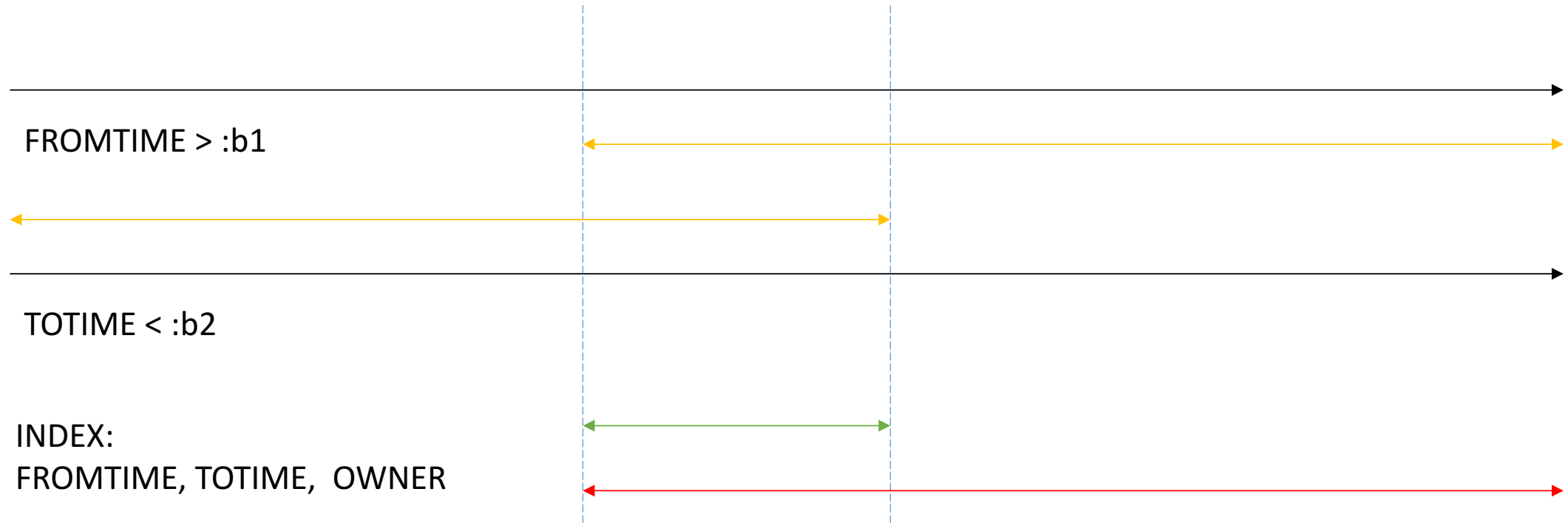
Id		Operation		Name	

0		SELECT STATEMENT			
1		TABLE ACCESS BY INDEX ROWID		TEST	
*		INDEX RANGE SCAN		TEST_SRCH01_IDX	

Predicate Information (identified by operation id):

```
2 - access("FROMTIME">TO_DATE(:B1,'dd-mm-yyyy') AND "OWNER"='PUBLIC' AND
          "TOTIME"<TO_DATE(:B2,'dd-mm-yyyy'))
      filter(("OWNER"='PUBLIC' AND "TOTIME"<TO_DATE(:B2,'dd-mm-yyyy')))
```

'12-03-2017' '29-03-2017',



`totime < to_date(:b2, 'dd-mm-yyyy')`

`fromtime < totime`



`fromtime < totime < to_date(:b2, 'dd-mm-yyyy')`



`fromtime < to_date(:b2, 'dd-mm-yyyy')`

```
SELECT *  
FROM test  
WHERE fromtime > to_date(:b1,'dd-mm-yyyy')           '12-03-2017'  
      AND totime < to_date(:b2,'dd-mm-yyyy')         '29-03-2017'  
      AND fromtime <= to_date(:b3,'dd-mm-yyyy')       '29-03-2017'  
      AND owner= 'PUBLIC';
```

SQL_ID	LIO_PR_EXE	LIO_PR_ROW	NR_EXEC	NR_ROWS	SQL_TEXT
5z7tvkyh401yf	118	5,76	1	40	SELECT * FROM test WHERE fromtime >
1zws36xnxurqh	28,5	1,39	1	40	SELECT * FROM test WHERE fromtime >

```
set pages 9999 lines 200
```

```
select * from table(dbms_xplan.display_cursor(1zws36xnxurqh',null,'TYPICAL'));
```

Id		Operation	Name	Rows	Bytes

0		SELECT STATEMENT			
* 1		FILTER			
2		TABLE ACCESS BY INDEX ROWID	TEST	495	49995
* 3		INDEX RANGE SCAN	TEST_SRCH01_IDX	495	

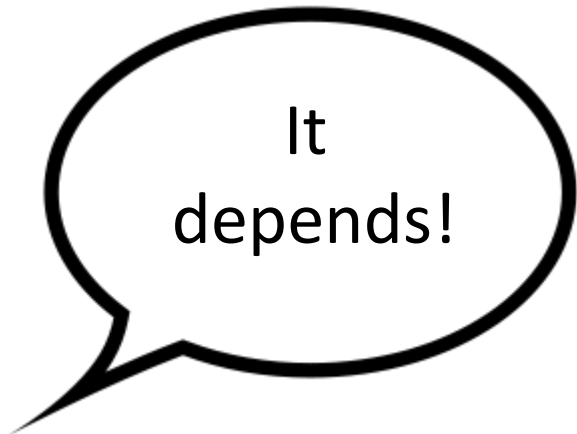
Predicate Information (identified by operation id):

```
-----
1 - filter(TO_DATE(:B1,'dd-mm-yyyy')<TO_DATE(:B3,'dd-mm-yyyy'))
3 - access("FROMTIME">TO_DATE(:B1,'dd-mm-yyyy') AND "OWNER"='PUBLIC' AND
          "FROMTIME"<TO_DATE(:B3,'dd-mm-yyyy') AND "TOTIME"<TO_DATE(:B2,'dd-mm-yyyy'))
      filter(("OWNER"='PUBLIC' AND "TOTIME"<TO_DATE(:B2,'dd-mm-yyyy')))
```

Should TOTIME be a part of the index?

```
SELECT count(*)  
FROM test  
WHERE fromtime > to_date(:b1, 'dd-mm-yyyy')    --'12-03-2017'  
      --AND totime < to_date(:b2, 'dd-mm-yyyy') --'29-03-2017'  
      AND fromtime < to_date(:b3, 'dd-mm-yyyy') --'29-03-2017'  
      AND owner= 'PUBLIC';
```

```
COUNT (*)  
-----  
458
```



It
depends!

Other improvements on index side?

```
CREATE INDEX test_srch01_idx ON test (fromtime, totime, owner);
```

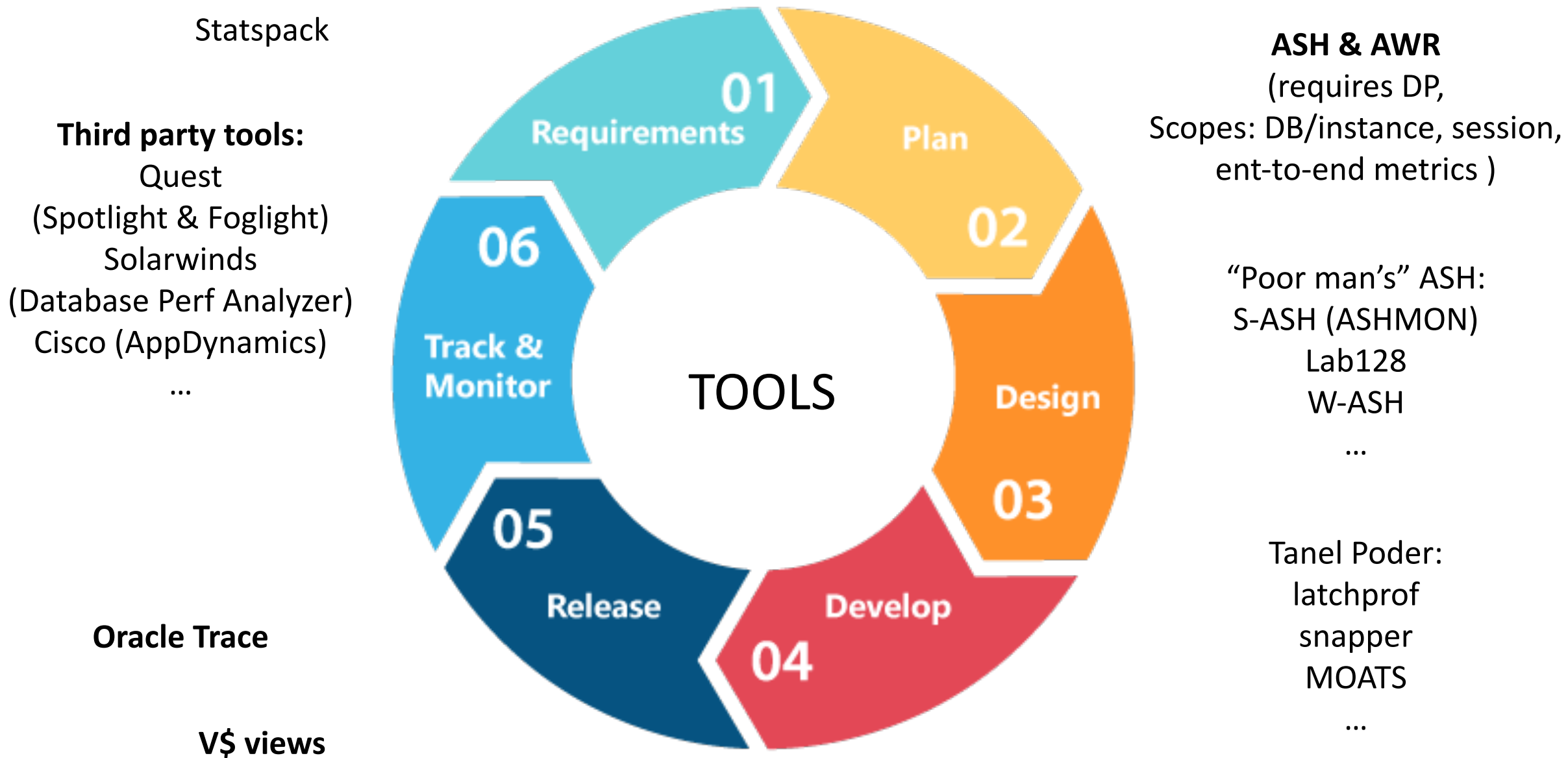
```
CREATE INDEX test_srch02_idx ON test (fromtime, owner);
```

```
CREATE INDEX test_srch03_idx ON test (owner, fromtime, totime)  
COMPRESS 1;
```

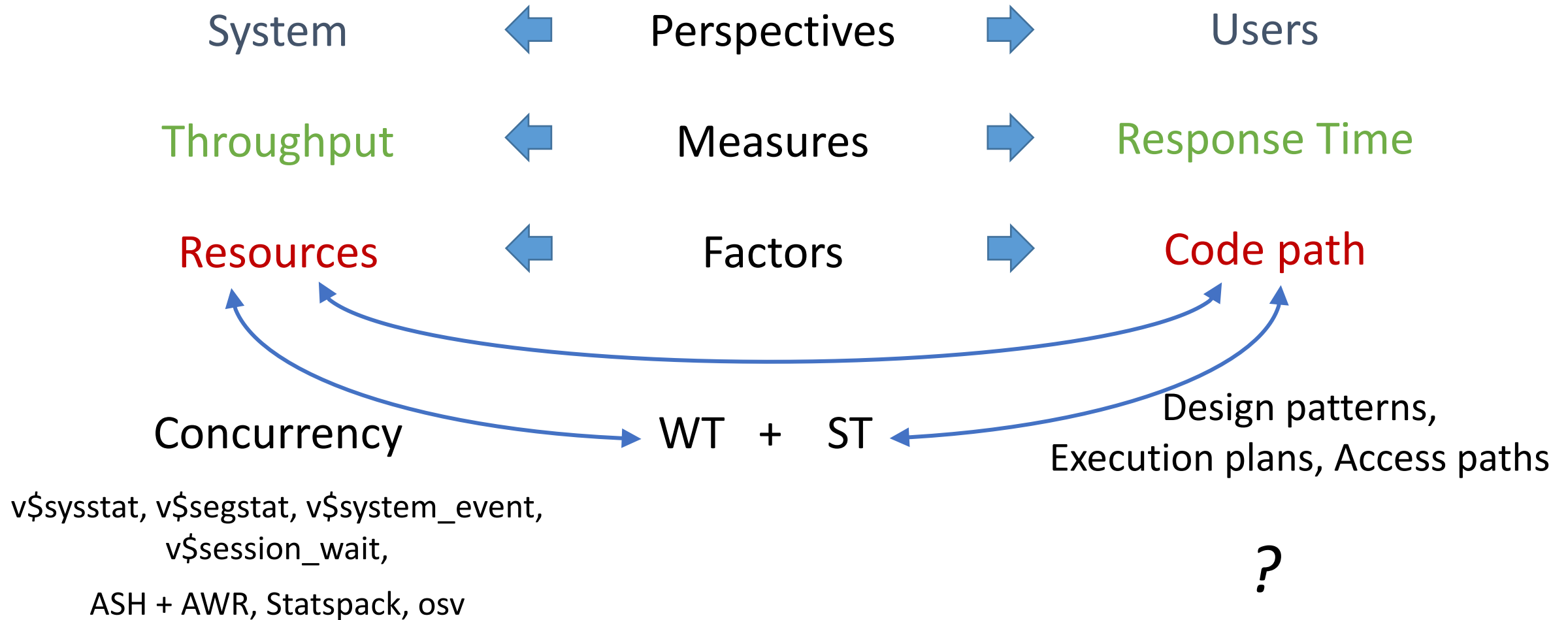
Summary

SQL_ID	LIO_PR_EXE	LIO_PR_ROW	NR_EXEC	NR_ROWS	SQL_TEXT
5b4vuz6s9tr9a	3758,5	300,68	1	24	SELECT /*+ FULL(test) */ * FROM test W
78hrukyg581va	240,5	19,24	1	24	SELECT /*+ INDEX(test test_srch02_idx)
f8kv57096w3c7	85	6,8	1	24	SELECT /*+ INDEX(test test_srch01_idx)
7u85amb2k7wsq	16	1,28	1	24	SELECT /*+ INDEX(test test_srch01_idx)
3xh2n0h20bqpg	14,5	1,16	1	24	SELECT /*+ INDEX(test test_srch03_idx)

Note! All these execution plans would probably be FAST ENOUGH



What is performance?





Many developers, writing on many different features

How do you map what's going on inside of Oracle to the application?

Check code paths
(typical: chattiness)

Check access paths.

Check use of resources.



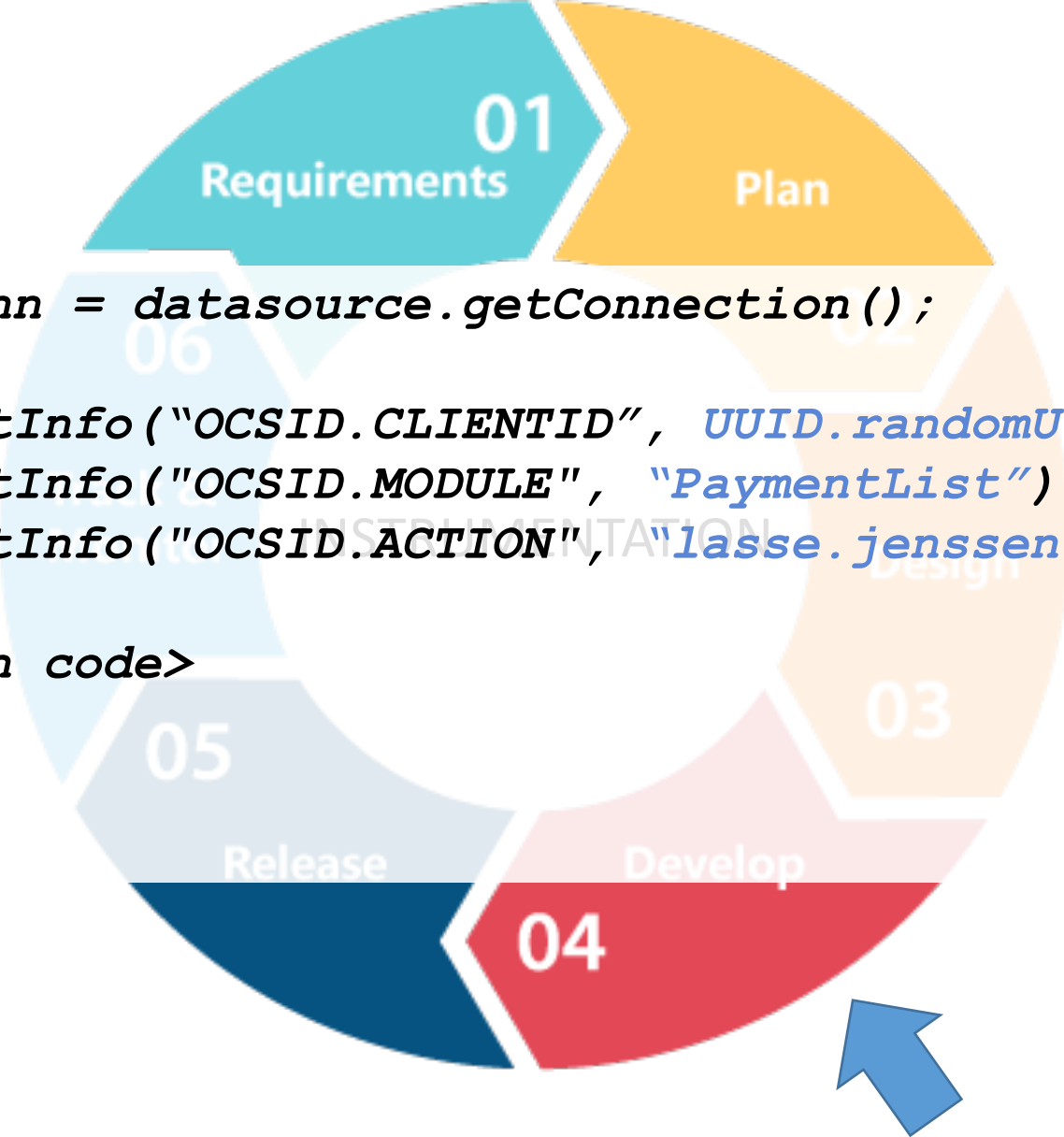
ORM APIs (like Hibernate):

- Generates the SQLs
- Check use of BINDs
- Check use of BINDs vs IN-list
- Check code path

Limited amount of time
allocated for evaluating
produced code



**Need a precise method
for analyzing code**



```
Connection conn = datasource.getConnection();  
...  
conn.setClientInfo("OCSID.CLIENTID", UUID.randomUUID().toString());  
conn.setClientInfo("OCSID.MODULE", "PaymentList");  
conn.setClientInfo("OCSID.ACTION", "lasse.jenssen@gmail.com");  
  
... <application code>  
  
conn.close;
```

Developer



```
exec dbms_monitor.serv_mod_act_trace_enable(  
    service_name=>'SYS$USERS',  
    module_name=>'OUGN17',  
    action_name=>DBMS_MONITOR.all_actions,  
    waits=>TRUE, binds=>TRUE,  
    plan_stat=>'ALL_EXECUTIONS');
```

I have just finished the
«PaymentList»
module. Could you do
a review on the
database side?

No problem.
Start your tests in
PAYT in 15 min,



DBA



Developer

Click on “*PaymentList*”



9,5 sec



Code sets “*Metrics*”



Trace is triggered



Trace file

/opt/oracle/diag/rdbms/v11202/V11202/trace/V11202_oracle_21089.trc

...

*** 2011-01-10 13:33:22.746

*** SESSION ID: (19.1247) 2011-01-10 13:33:22.746

*** CLIENT ID: (**08B77430-B717-4ACA-A092-BD970987AD90**) 2011-01-10
13:33:22.746

*** SERVICE NAME: (**PAY_SRV**) 2011-01-10 13:33:22.746

*** MODULE NAME: (**PaymentList**) 2011-01-10 13:33:22.746

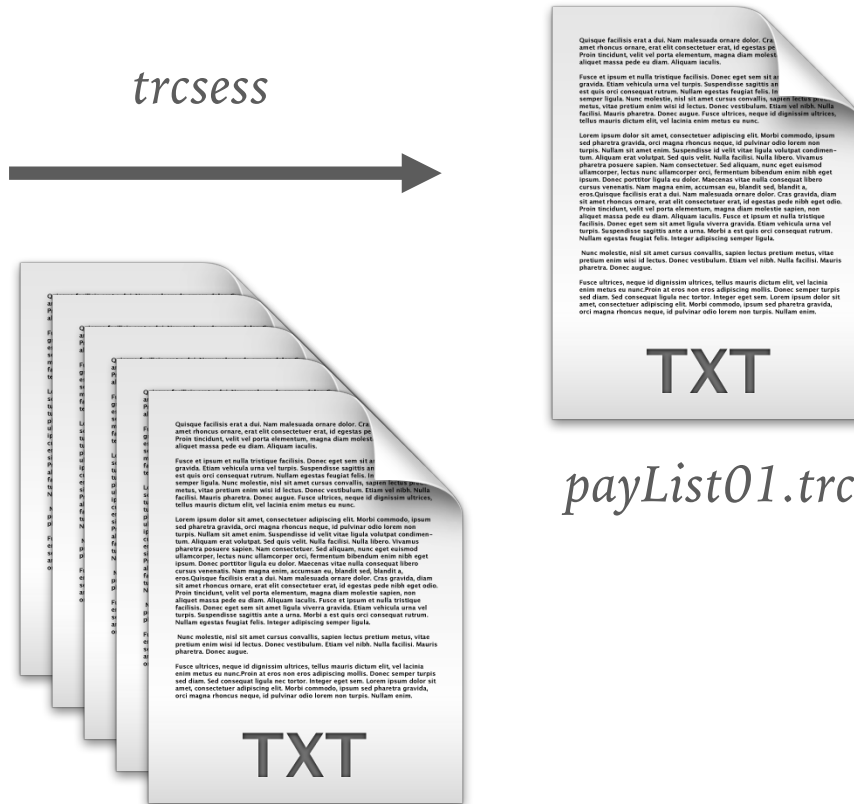
*** ACTION NAME: (**lasse.jenssen@gmail.com**) 2011-01-10 13:33:22.746

> trcsess output=payList01.trc clientid="08B77430-B717-4ACA-A092-BD970987AD90" *



DBA

trcsess



TXT

payList01.trc

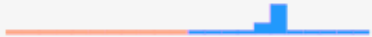
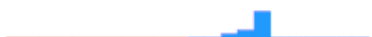
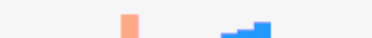
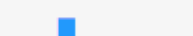
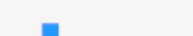
Profiler

Method R Profiler

tvdxstat

tkprof



	Subroutine	Duration		Calls	Duration per call (seconds)			
		seconds	% R		mean	min	skew	max
1	CPU: EXEC dbcalls ›	0.363946	48.7%	51	0.007136	0.000000		0.357946
2	CPU: FETCH dbcalls ›	0.343949	46.0%	48	0.007166	0.000000		0.076989
3	CPU: PARSE dbcalls ›	0.014997	2.0%	52	0.000288	0.000000		0.006999
4	SQL*Net message from client ›	0.012006	1.6%	5	0.002401	0.001334		0.003630
5	unaccounted-for between calls ›	0.005753	0.8%	20	0.000288	0.000017		0.004053
6	unaccounted-for within dbcalls ›	0.004826	0.6%	203	0.000024	-0.000956		0.006470
7	CPU: CLOSE dbcalls ›	0.001000	0.1%	52	0.000019	0.000000		0.001000
8	uncategorizable within dbcalls ›	0.000409	0.1%	1	0.000409	0.000409		0.000409
9	SQL*Net message to client ›	0.000025	.	5	0.000005	0.000003		0.000006
10	db file sequential read ›	0.000017	.	1	0.000017	0.000017		0.000017
11	Total (10)	0.746928	100.0%					



Method R Profiler

7



Method R Profiler

File View Run Help

Path	Name	Size	Last Modified	Action	Progress
C:\Users\marco_g\Down...	weblv_ora_18625.trc	402,479,100	2011-09-16 15:00:46	Stop	17%

Status Output

Downloads

File Home Share View

Pin to Quick access Copy Paste Move to Delete Copy to Rename New folder Properties Select all Select none Invert selection

Clipboard Organize New Open Select

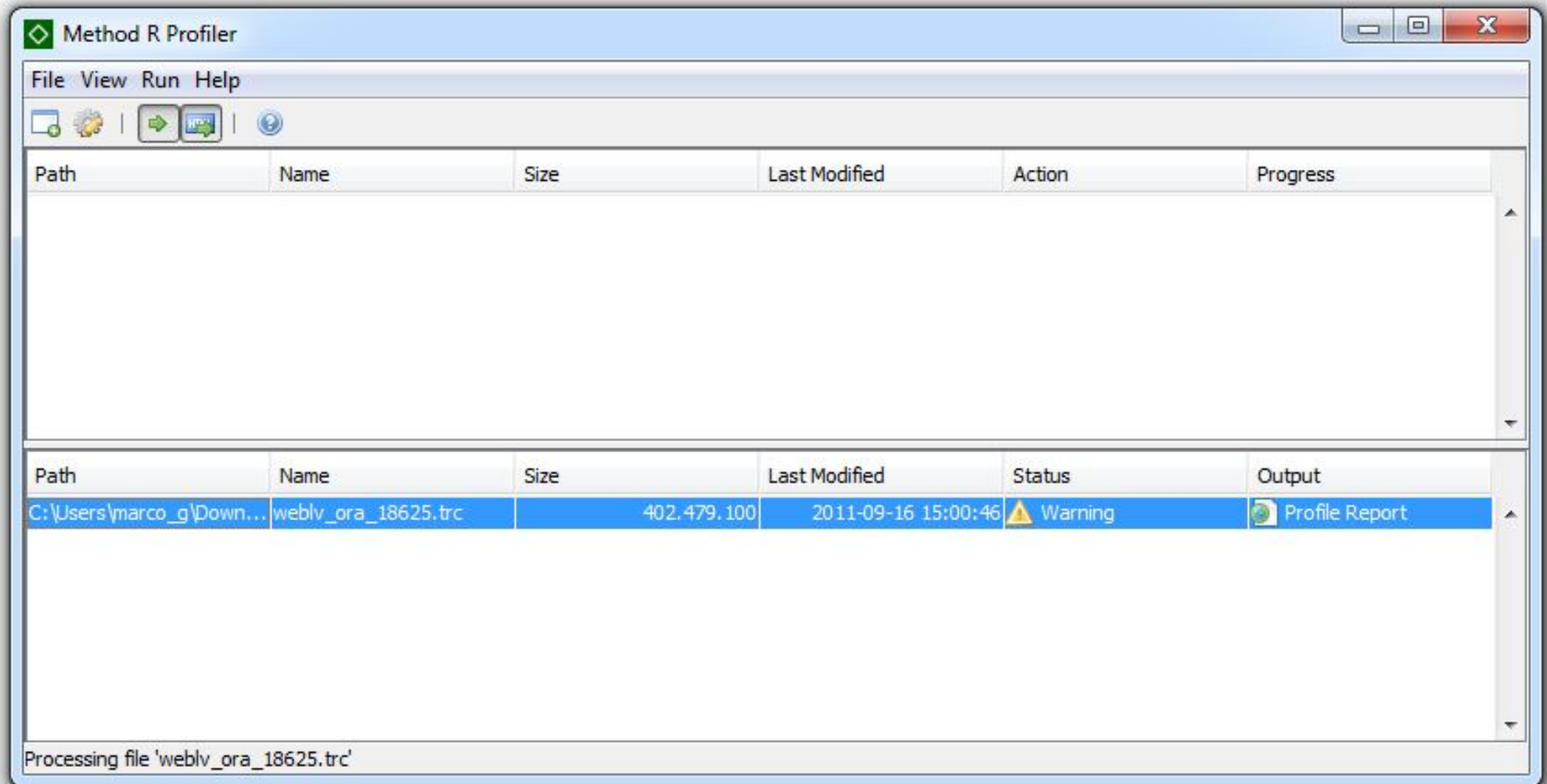
← → ↕ ↑ > This PC > Downloads Search Downloads

Name	Date modified	Type
2016_09_oow_real_world_experience_w_e...	24.10.2017 18.07	Microsoft Pow
EREXET_ora_21076_LJ.trc	24.10.2017 00.18	TRC File
ougn17.trc	23.10.2017 22.50	TRC File
drp_erem_cnsp_trg.sql	03.10.2017 14.15	SQL File
EREXP_ora_35196_20170926_1420.trc	26.09.2017 14.33	TRC File

100 items | 1 item selected 15,4 KB



Method RTM Profiler 7





METHOD RTM Profiler

Let's have a look at a report.

```
select table_name, blocks
from user_tables
where table_name='TEST';
```

TABLE_NAME	BLOCKS
TEST	7552

```
select count(distinct
             dbms_rowid.rowid_block_number(rowid))
       blocks_with_data
from test;
```

BLOCKS_WITH_DATA
7463

Many developers, writing on many different features

How do you map what's going on inside of Oracle to the application?

Check code paths (typical: chattiness)

Check access paths.

Check use of resources.



ORM APIs (like Hibernate):

- Generates the SQLs
- Check use of BINDs
- Check use of BINDs vs IN-list
- Check code path

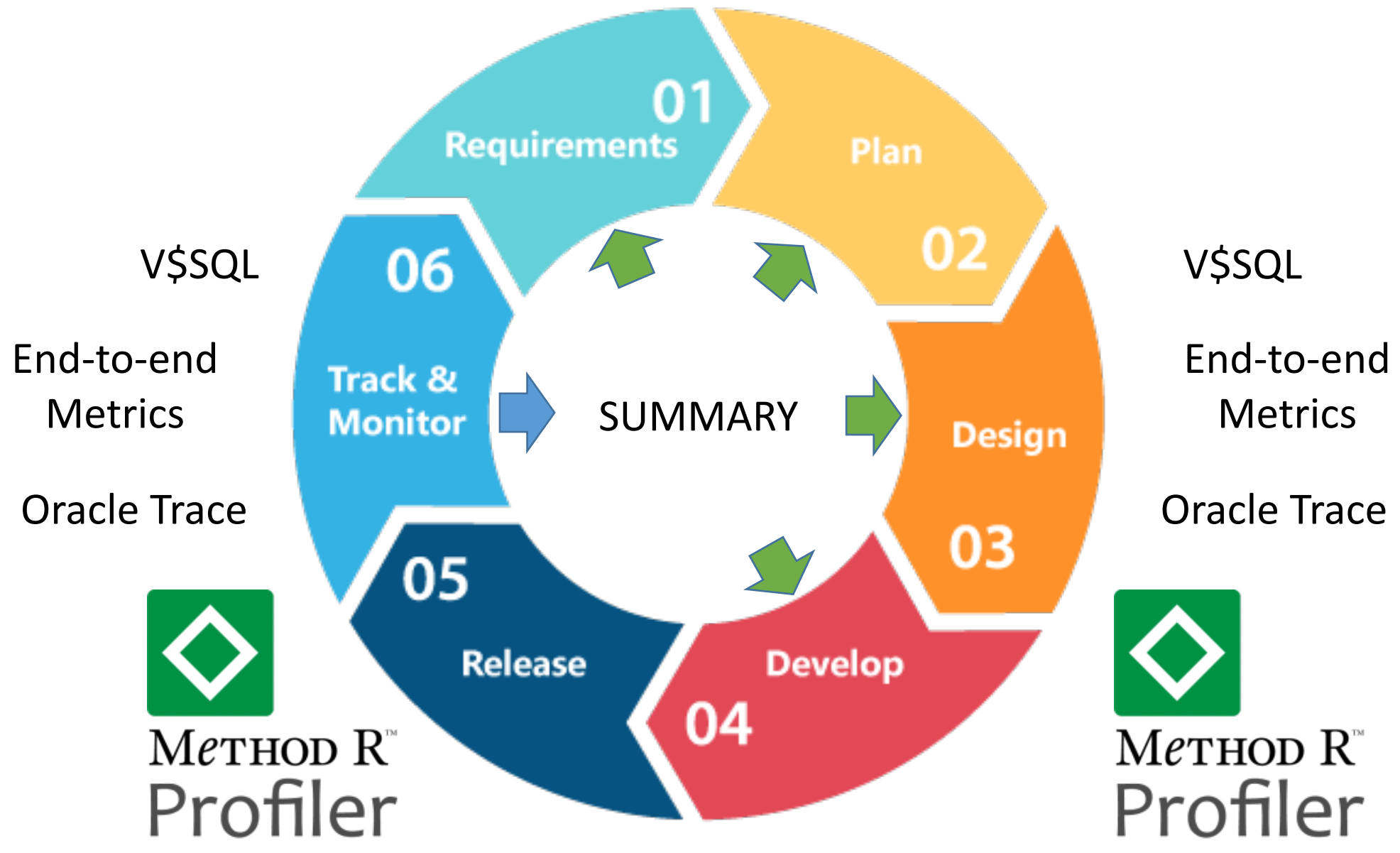
Limited amount of time allocated for evaluating produced code

The image shows the Method R Profiler logo, which consists of a green square with a white diamond shape inside. Below the logo, the text 'METHOD R Profiler' is written in a serif font. To the left of the logo, the text 'End-to-End Metrics' and 'Extended Oracle Trace' is displayed in a sans-serif font.

End-to-End Metrics

Extended Oracle Trace

METHOD R Profiler





lasse.jenssen@rightconsulting.no

Twitter: @lasjen

<http://www.rightconsulting.no>

