
ORACLE®

ORACLE OPEN WORLD

experience
INNOVATION

November 11–15, 2007

ORACLE

ORACLE

DBA's New Best Friend: Advanced SQL Tuning Features of Oracle Database 11g

Peter Belknap, Sergey Koltakov, Jack Raitto

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.

Agenda

- SQL Tuning Challenges
- Oracle Database 11g Solutions
  - Automatic SQL Tuning
  - Real-time SQL Monitoring
  - Partition Advisor
- Q & A

SQL Tuning Challenges

- Oracle Database 10g introduced SQL advisors to simplify application and SQL tuning
- Remaining challenges
  - SQL Tuning still reactive
  - Painful to find and investigate long-running SQL
  - Partitioning excluded from schema optimization advice
- Oracle Database 11g solutions
  - Automatic SQL Tuning
  - Real-time SQL Monitoring
  - Partition Advisor component of SQL Access Advisor

Automatic SQL Tuning

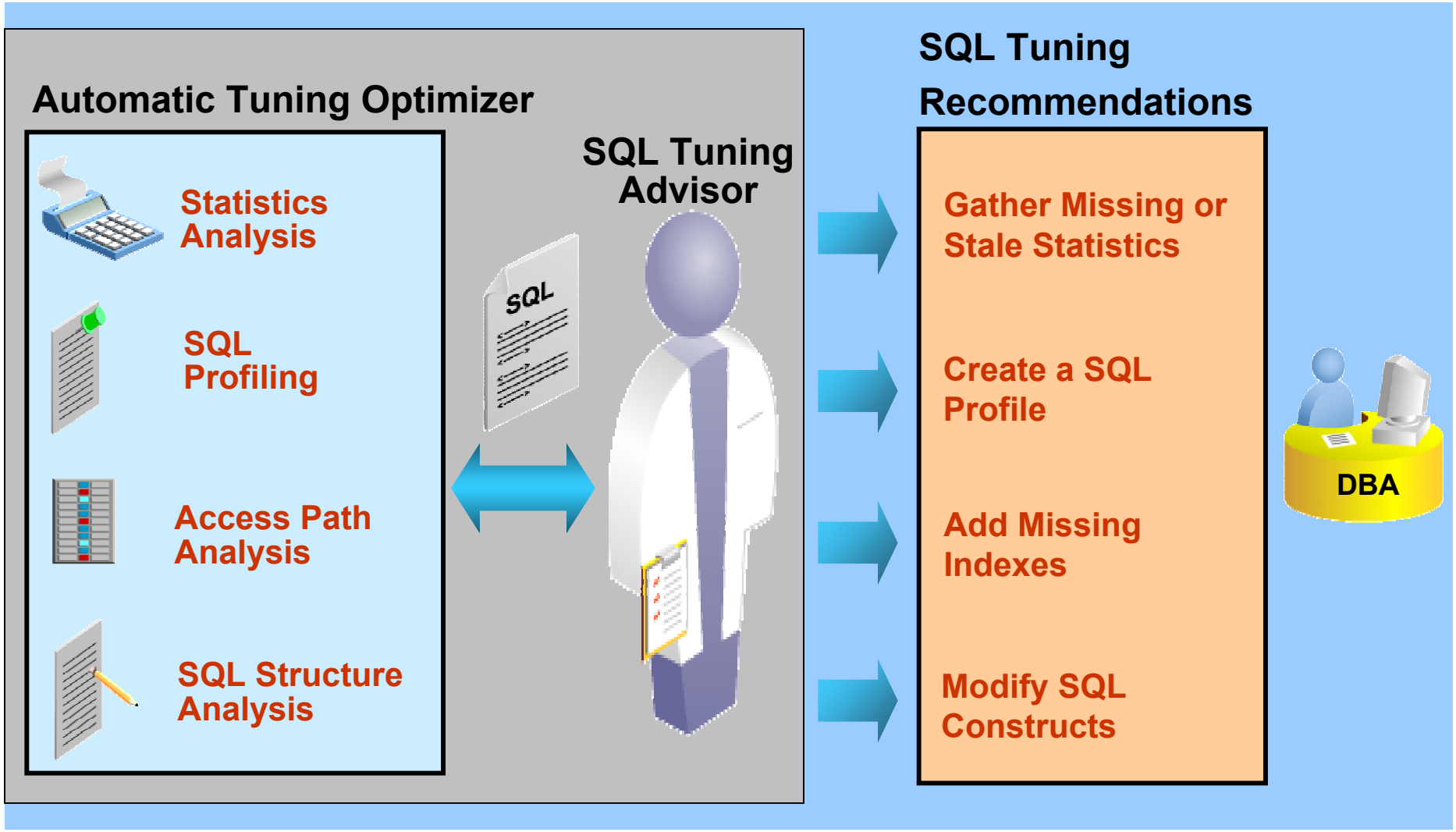
The Self-Managing Database

Challenges of Manual SQL Tuning

- Requires expertise in several domains
  - SQL optimization: adjust the execution plan
  - Access design: provide fast data access
  - SQL design: use appropriate SQL constructs
- Time consuming
  - Plans are complicated
  - Each SQL statement is unique and each execution can be different
  - Potentially large number of statements to tune
  - Testing proposed changes is labor-intensive
  - Many possible ways to a solution
- Never ending task
  - SQL workload always evolving
  - Plan regressions

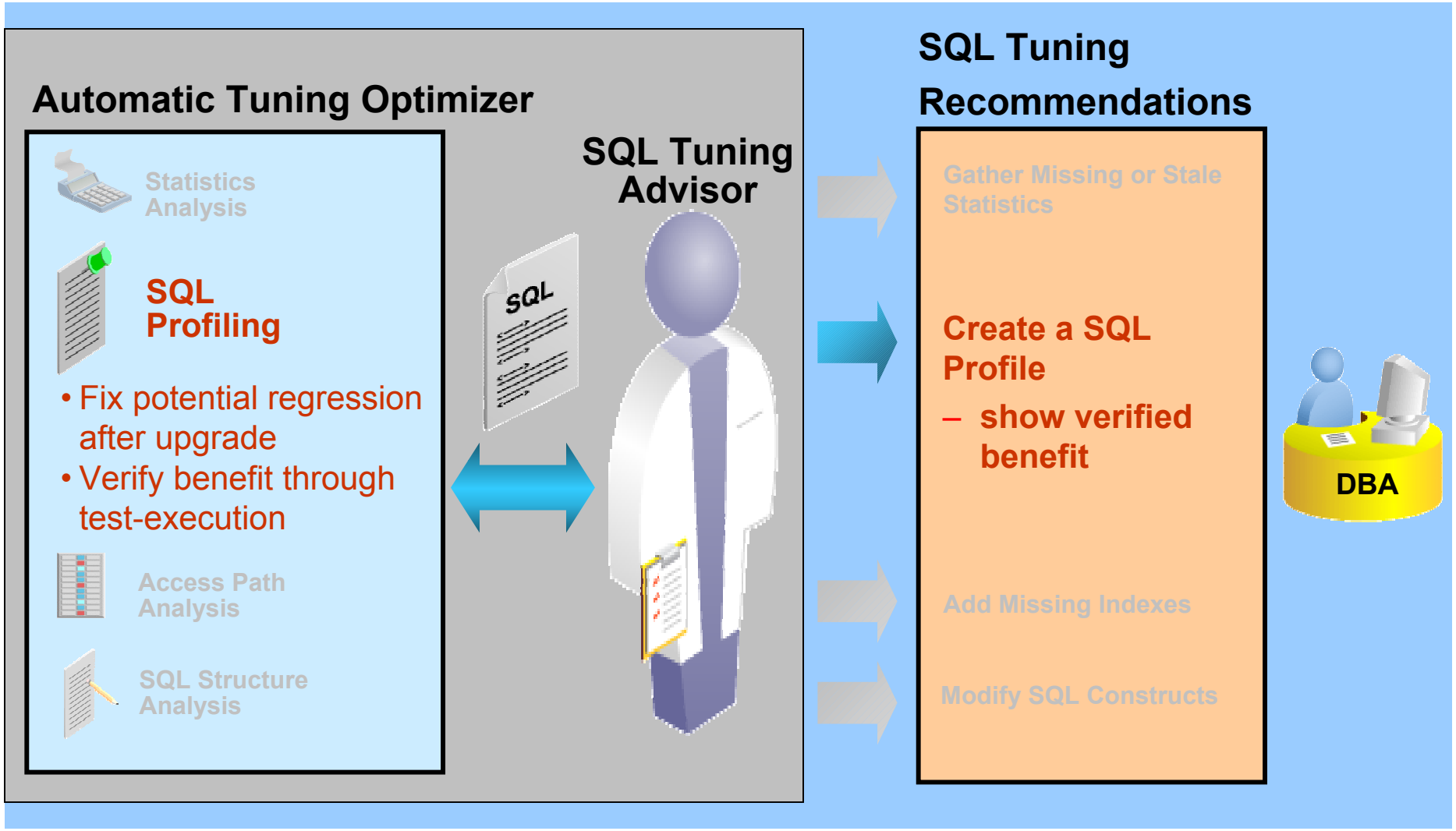
Simplifying SQL Tuning

SQL Tuning Advisor, since Oracle Database 10g



Improvements in Oracle Database 11g

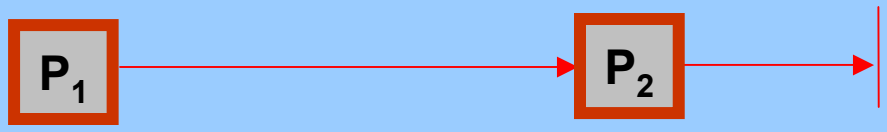
Better SQL Profiling



Testing SQL Profiles (1)

Measuring actual benefit with test-execution

Naïve: Execute in Order

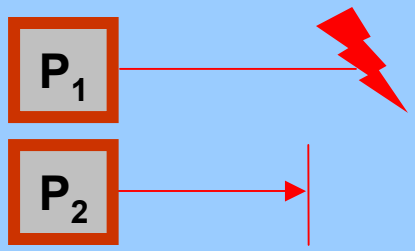


Finish, P2 wins!

But what if P1 never completes?

Timeout!

It would be great to run them concurrently….



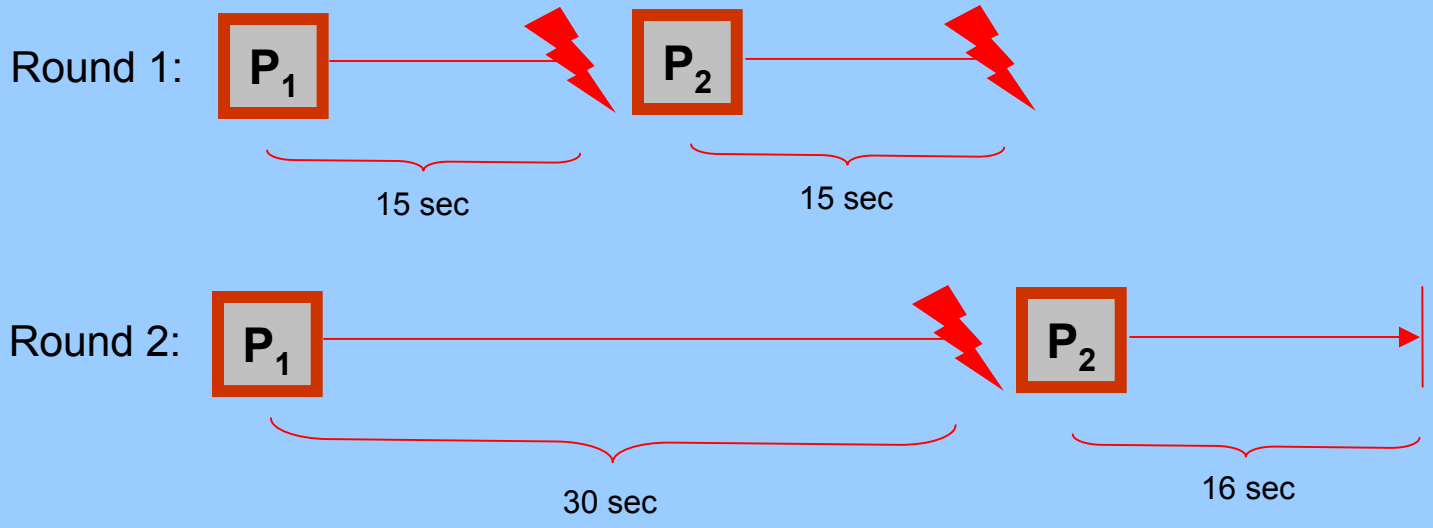
P2 wins, kill P1!

But then I take 2 CPUs, and N in the general case…

Testing SQL Profiles (2)

Measuring actual benefit with test-execution

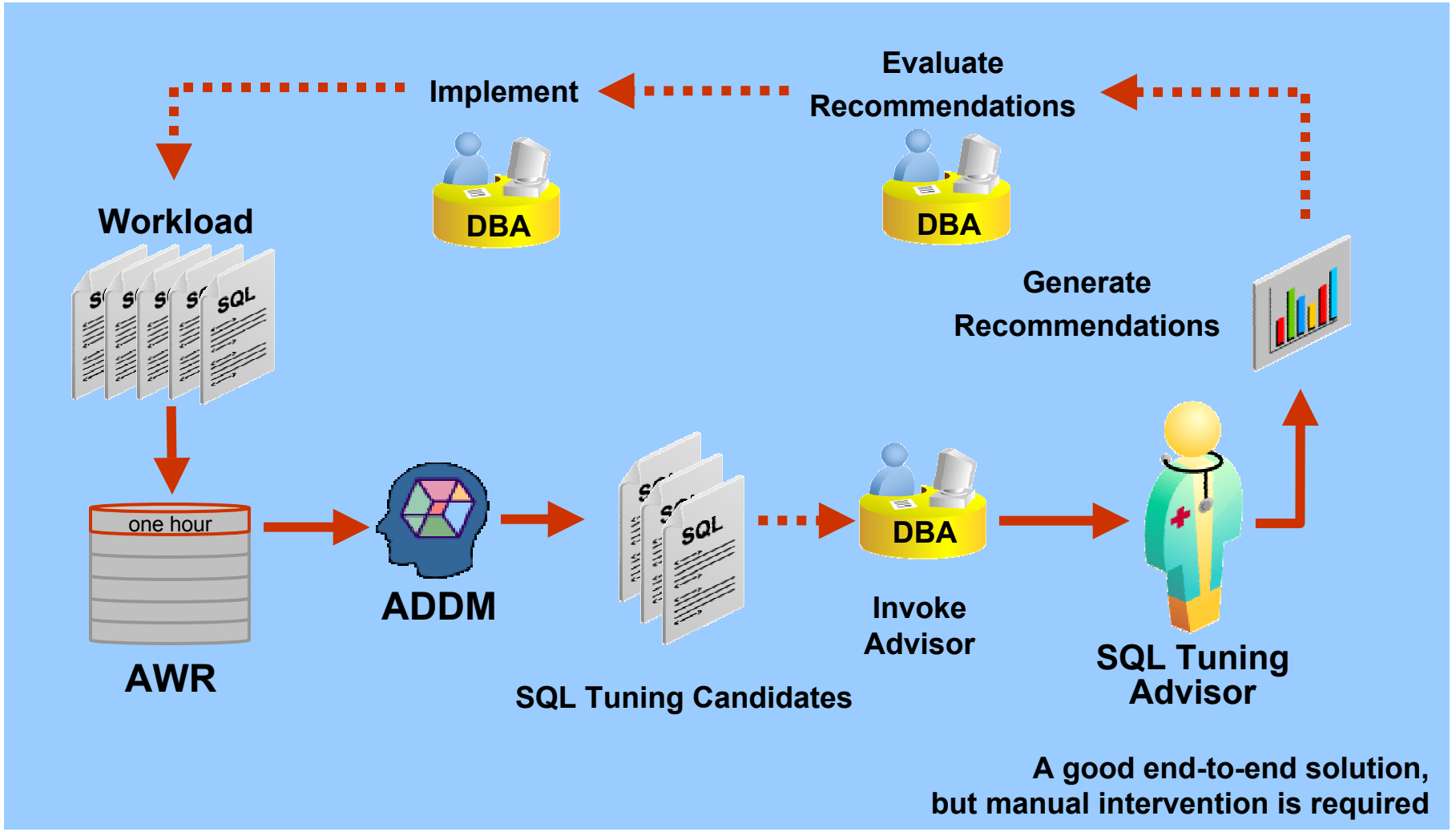
Solution: Tournament Execution



Your winner, with a knockout in the second round, P_2!

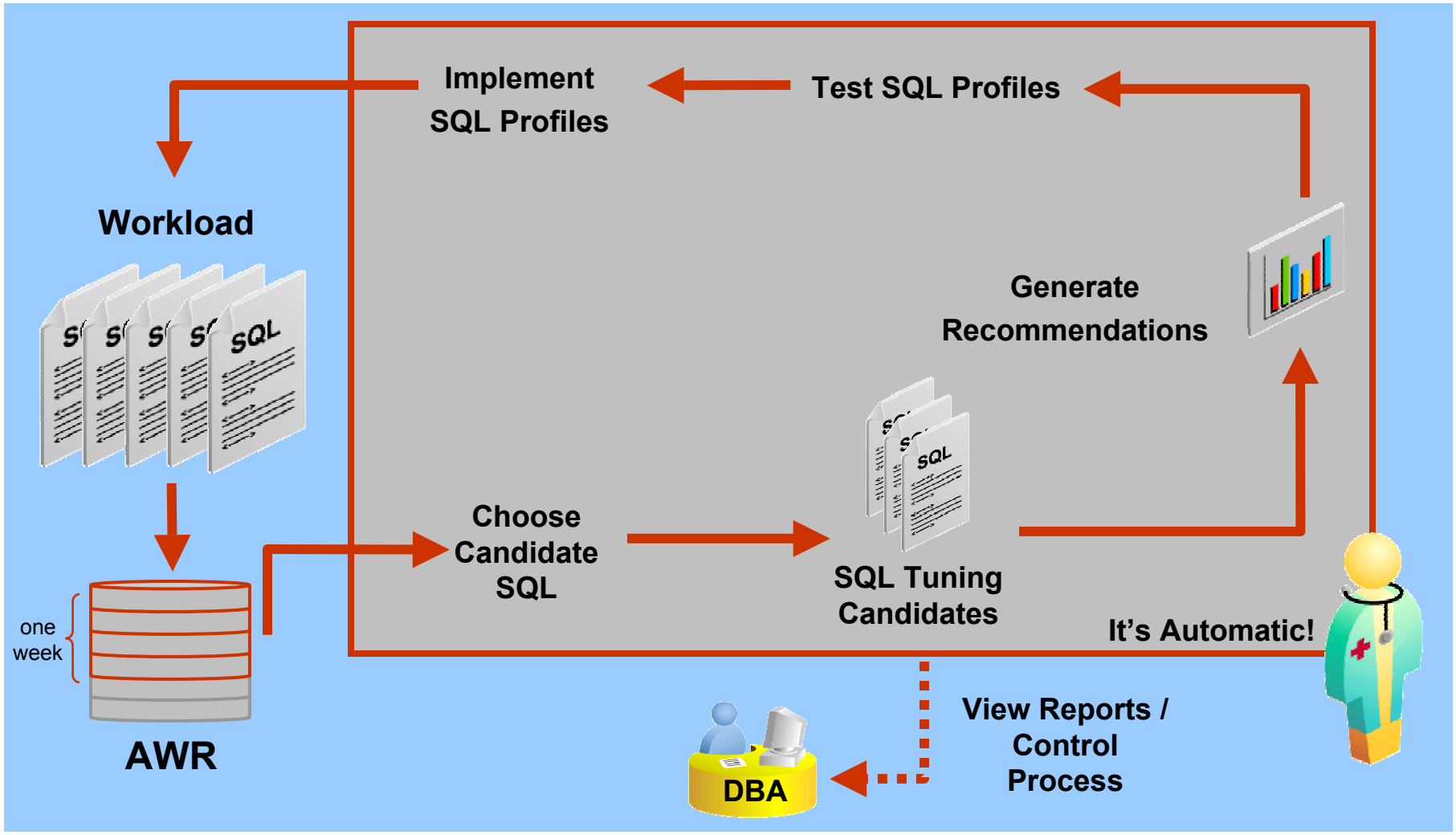
SQL Tuning in Oracle Database 10g

End-to-end Workflow

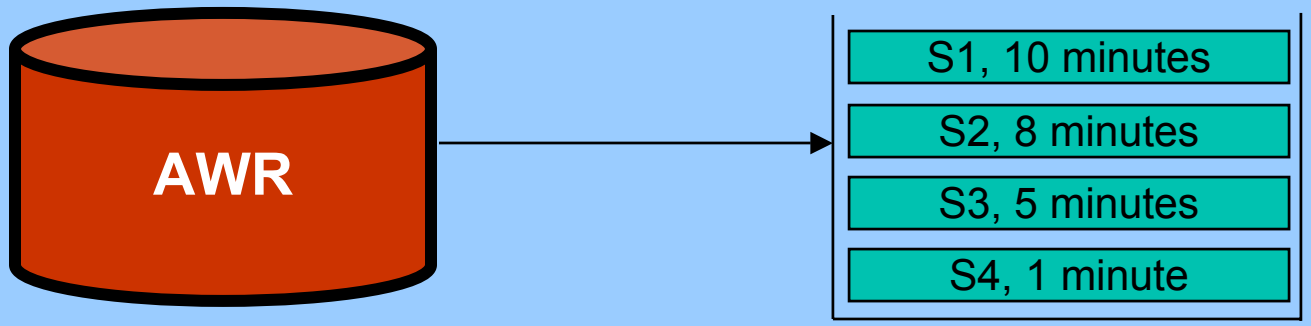


Automatic SQL Tuning in Oracle 11g

The Self-Managing Database



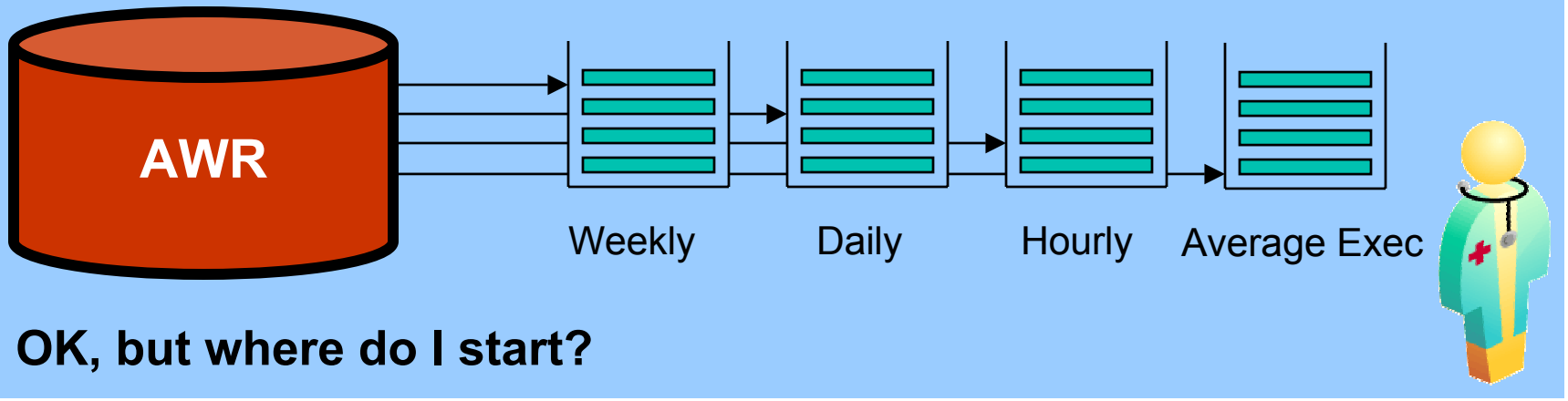
Picking Candidate SQL (1)



I could just pick from the top down…

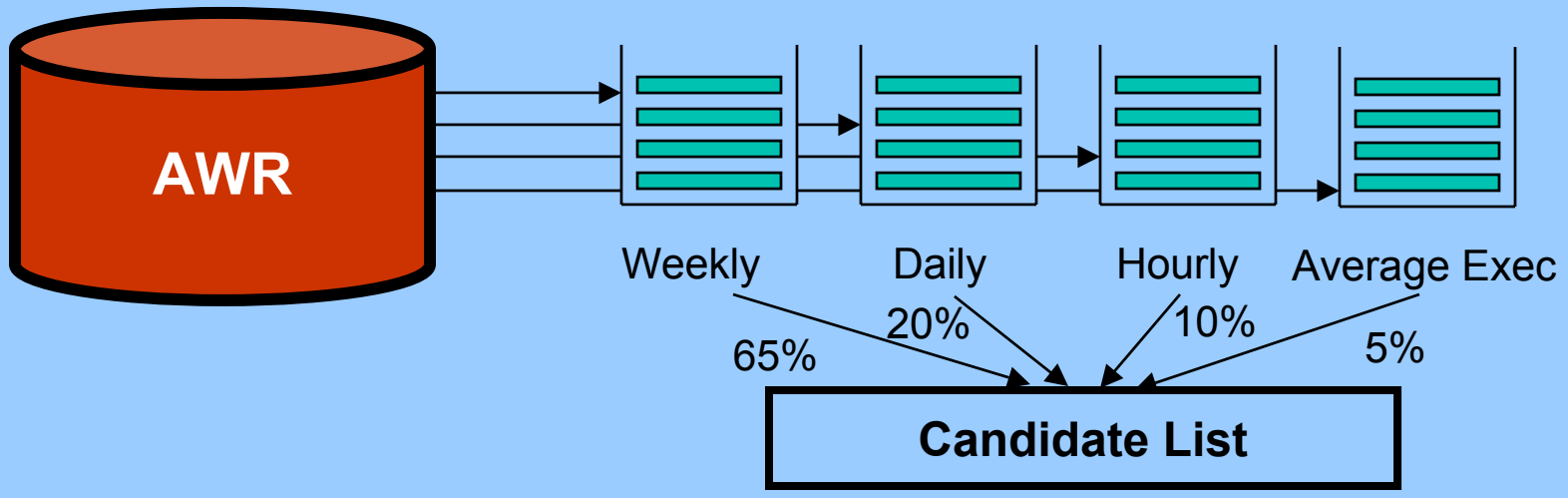
But I will miss SQLs with important hotspots!

Let's try a more balanced approach:



OK, but where do I start?

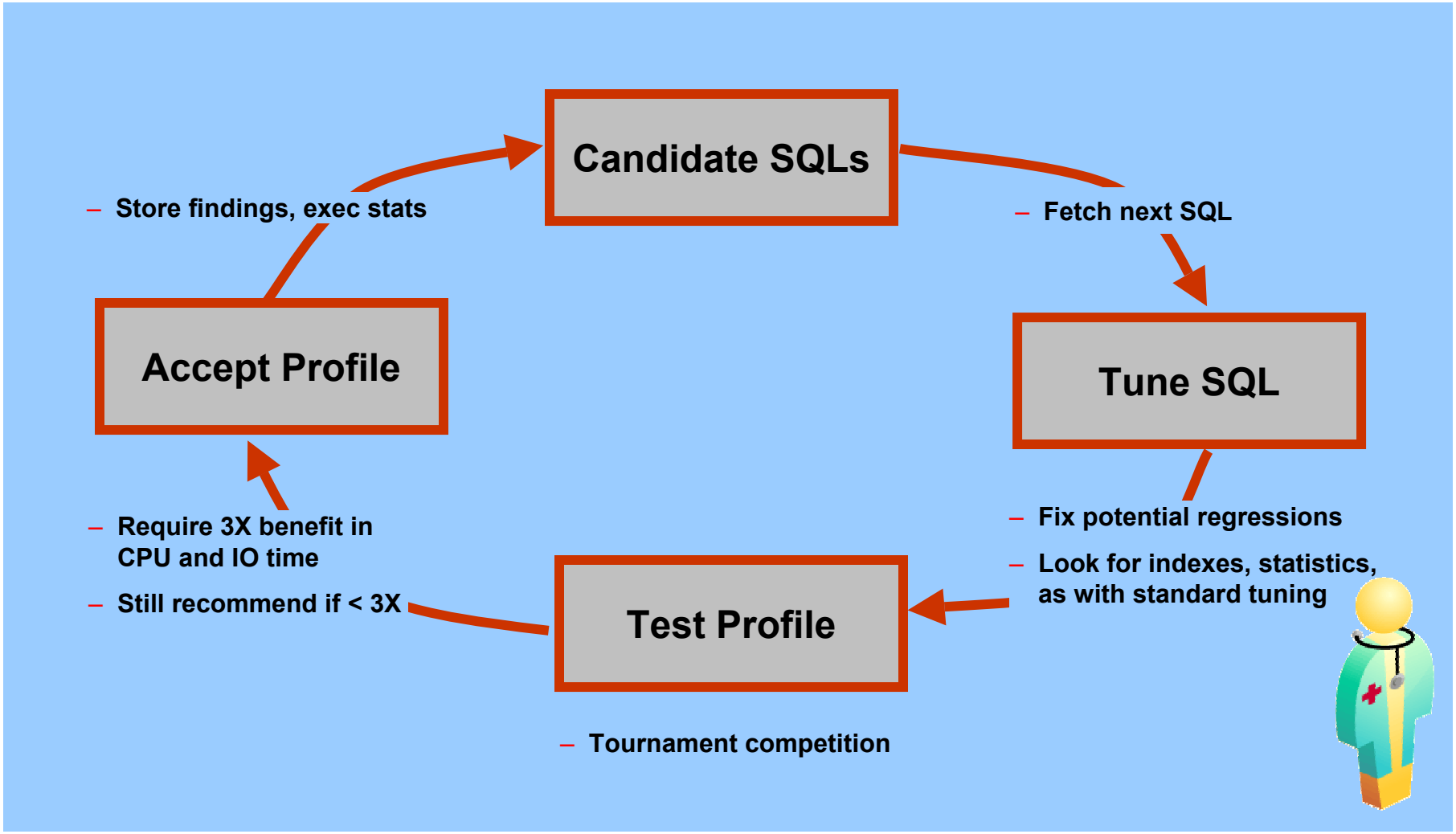
Picking Candidate SQL (2)



- Eventually we need one list to tune from: merge the buckets.
- All buckets are not created equal: focus on the week, but don't forget about the others.
- Focus on the SQLs we have not seen recently: Don't re-tune SQLs if nothing has changed!

Tuning Flow

Tuning activities per SQL



Focus on SQL Profiles

First step in automating SQL tuning

Auto-testing/implementing is limited to profiles because:

- No lengthy, expensive set-up process (building an index takes time)
- Private to the current compilation
- No change to user SQL (does not change semantics)
- SQL-level recommendation, can be effectively tested
- Easily reversed by the DBA

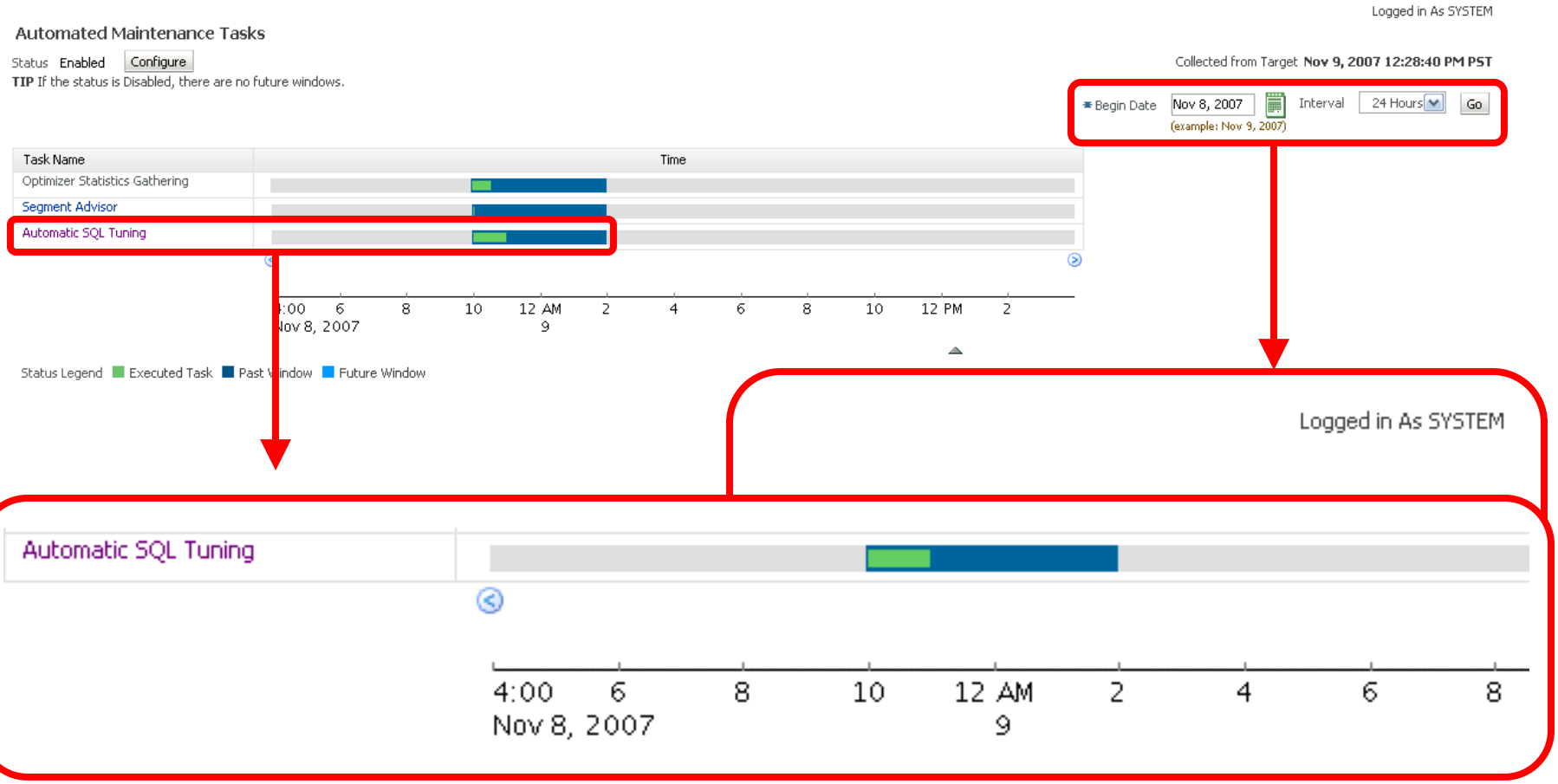
Testing is done for regular SQL Tuning Advisor tasks as well!

Automatic SQL Tuning Defaults

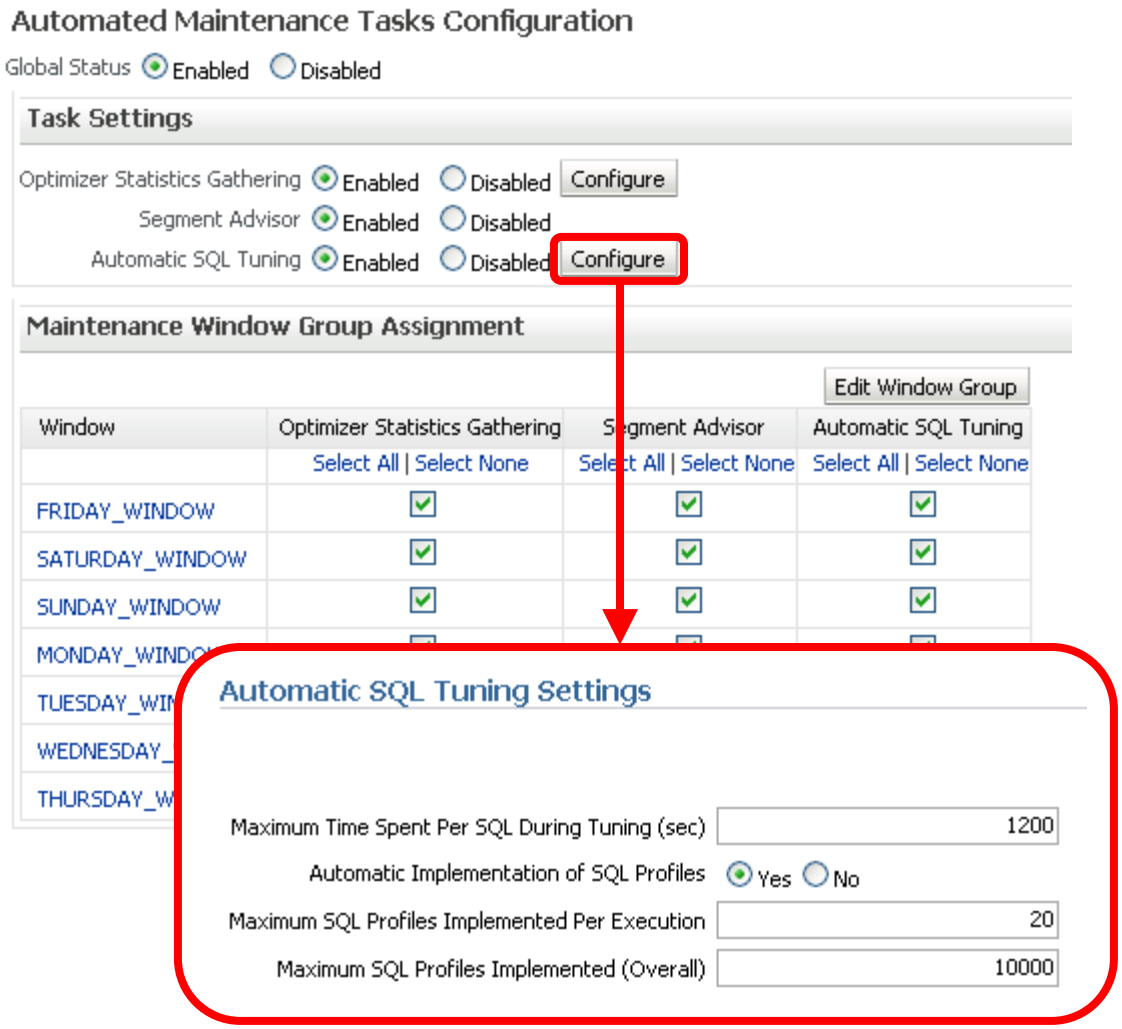
Sensible defaults with flexible configurations

- Out-of-the-box defaults:
  - Runs in each maintenance window (MAINTENANCE_WINDOW_GROUP)
  - SQL profiles are tested but not implemented
- DBA can configure using EM:
  - Whether / When / How long it runs
  - Resources it uses
  - Whether it implements profiles
  - How many profiles it implements

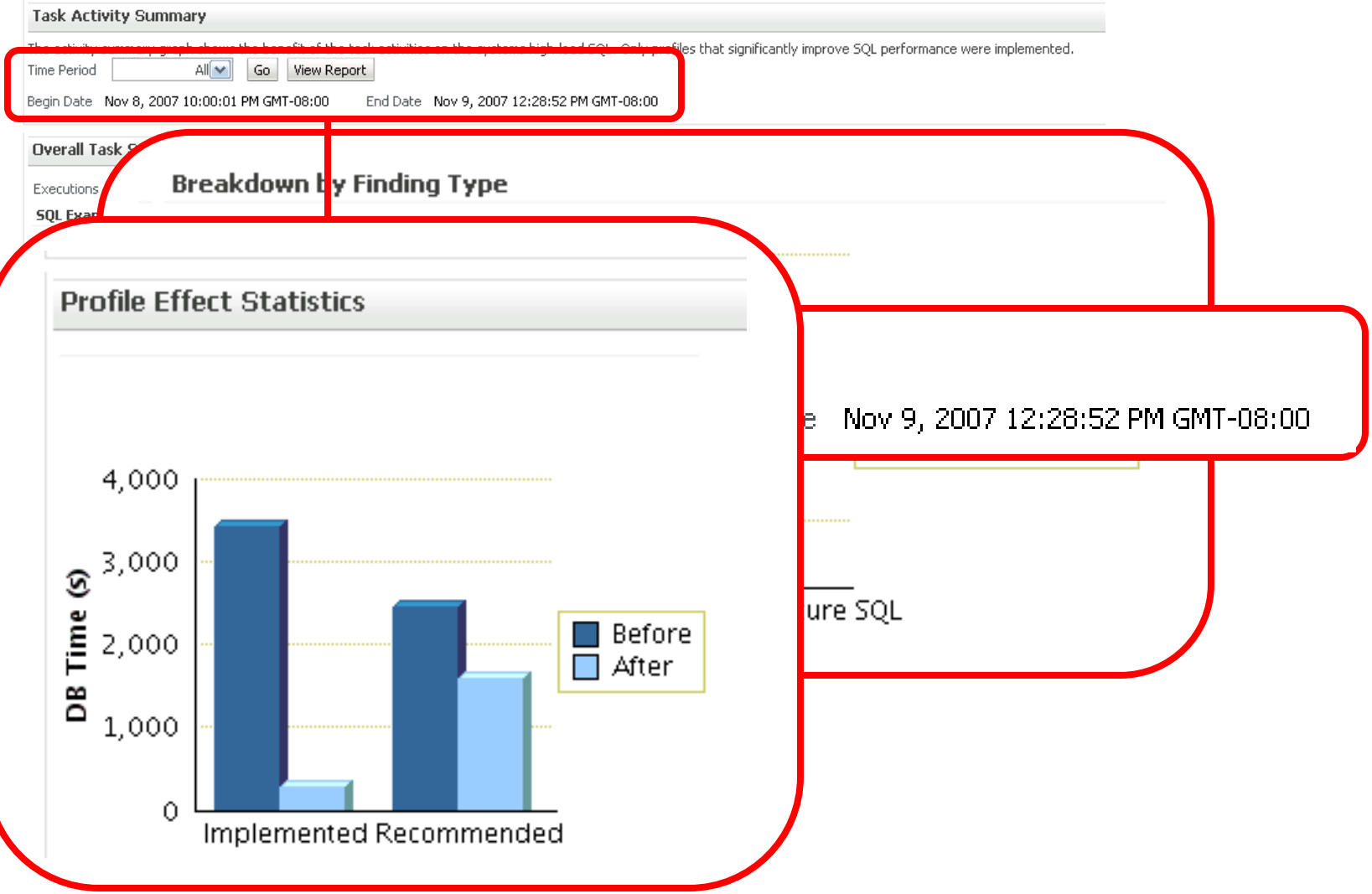
Automatic SQL Tuning Task



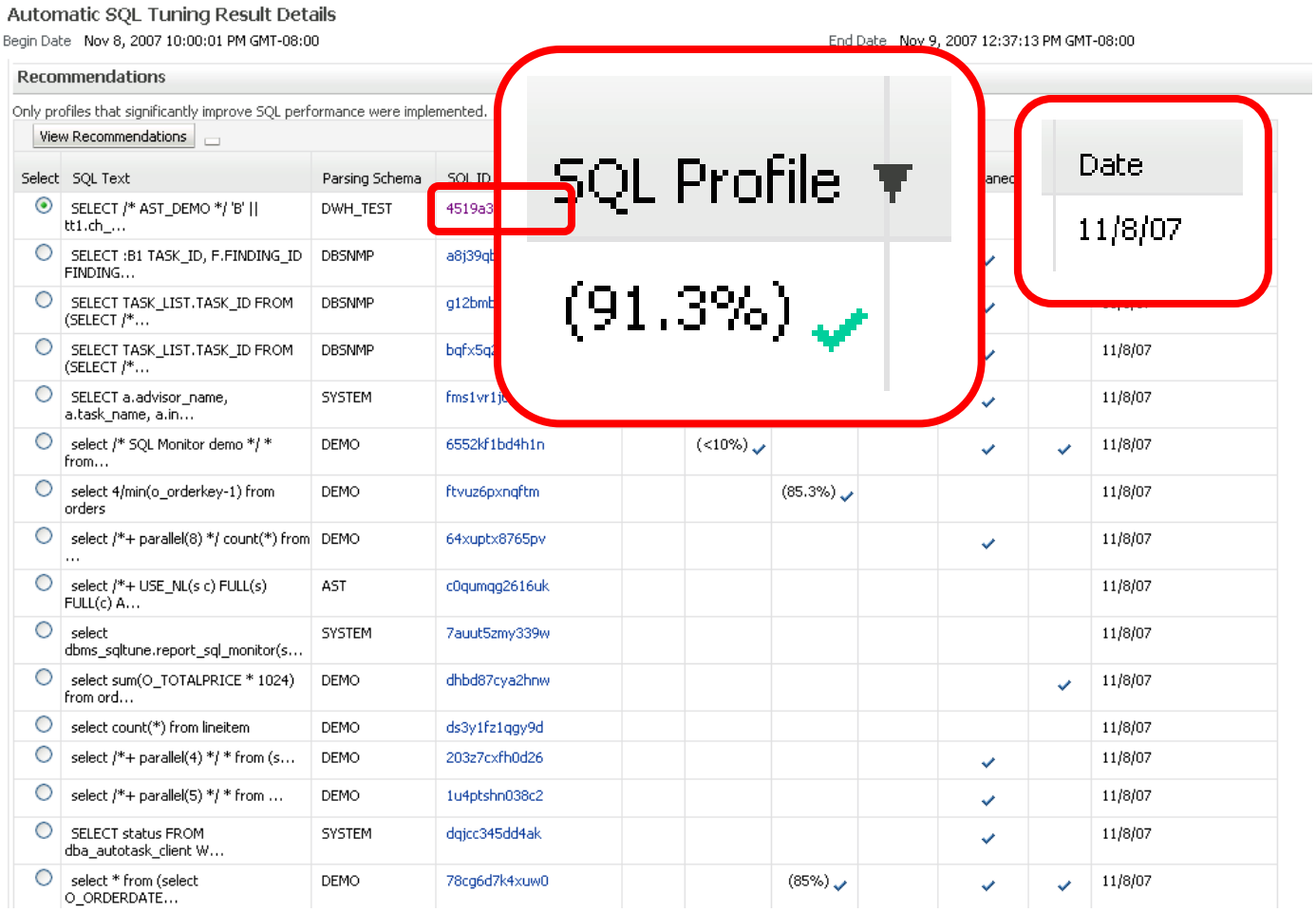
Automatic SQL Tuning Configuration



Automatic SQL Tuning Result Summary



Automatic SQL Tuning Result Recommendations



Select	SQL Text	Parsing Schema	SQL ID						
◉	SELECT /* AST_DEMO */ 'B' || tt1.ch_...	DWH_TEST	4519a3						
○	SELECT :B1 TASK_ID, F.FINDING_ID FINDING...	DBSNMP	a8j39qb				✓		
○	SELECT TASK_LIST.TASK_ID FROM (SELECT /*...	DBSNMP	g12bmb				✓		
○	SELECT TASK_LIST.TASK_ID FROM (SELECT /*...	DBSNMP	bqfx5q2				✓		11/8/07
○	SELECT a.advisor_name, a.task_name, a.in...	SYSTEM	fms1vr1jb				✓		11/8/07
○	select /* SQL Monitor demo */ * from...	DEMO	6552kf1bd4h1n		(<10%) ✓		✓	✓	11/8/07
○	select 4/min(o_orderkey-1) from orders	DEMO	ftvuz6pxnqftm		(85.3%) ✓				11/8/07
○	select /*+ parallel(8) */ count(*) from ...	DEMO	64xuptx8765pv				✓		11/8/07
○	select /*+ USE_NL(s c) FULL(s) FULL(c) A...	AST	c0qumqg2616uk						11/8/07
○	select dbms_sqltune.report_sql_monitor(s...	SYSTEM	7auut5zmy339w						11/8/07
○	select sum(O_TOTALPRICE * 1024) from ord...	DEMO	dhbd87cya2hnw					✓	11/8/07
○	select count(*) from lineitem	DEMO	ds3y1fz1qgy9d						11/8/07
○	select /*+ parallel(4) */ * from (s...	DEMO	203z7cxfh0d26				✓		11/8/07
○	select /*+ parallel(5) */ * from ...	DEMO	1u4ptshn038c2				✓		11/8/07
○	SELECT status FROM dba_autotask_client W...	SYSTEM	dqjcc345dd4ak				✓		11/8/07
○	select * from (select O_ORDERDATE...	DEMO	78cg6d7k4xuw0			(85%) ✓	✓	✓	11/8/07

Automatically Tuned SQL Details Drilldown

Conclusions

- *Manual SQL tuning* is painful even for the experts
- *Oracle 10g SQL Tuning Advisor* quickly gives DBA good choices
- *Oracle 11g Automatic SQL Tuning* automates the process by making the easy decisions
- *DBA can control* as much of the process as he wants

Just when you thought it was safe to run your SQLs…

There's a lot more to SQL performance than bad plans!

- Potential run-time issues
- Finding high response-time SQL is no piece of cake
- Keeping tabs on Parallel SQL is even harder

Real-Time SQL Monitoring

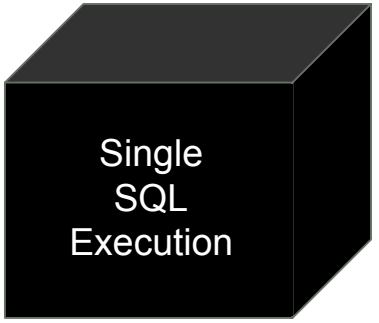
Shining new light
on SQL Performance

Problem: Managing High Response-Time SQLs

- **Monitoring:** tracking high response-time SQL
  - What is that expensive SQL (ETL, DDL, batch, report, …) I started up to?
  - Do I have any high response-time SQL running on my OLTP system?
  - Any SQL executing parallel?

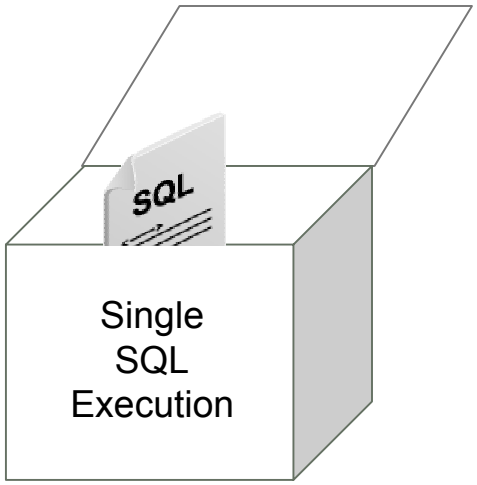
- **Investigating:** why is this execution so expensive?
  - Plan has hundreds of operations -- where is the time being spent?
  - Why is a particular operation so expensive?
  - SQL runs parallel, is DOP appropriate? is there a skew?

➔ What is going on inside a SQL execution???



Solution:
Real-time SQL Monitoring
Looking inside the SQL

- Enabled **out-of-the-box** with no performance impact
- **Automatically** monitors SQL executions that:
  - consume more than 5 seconds of CPU or I/O time
  - are running parallel: PQ, PDML, PDDL
- Monitors each execution **independently**
- Exposes monitoring statistics at **multiple levels**
  - Global execution level
  - Plan operation level (Plan Tuning)
  - Parallel Execution level (PX Tuning)
- Guides your tuning efforts



How does it work?

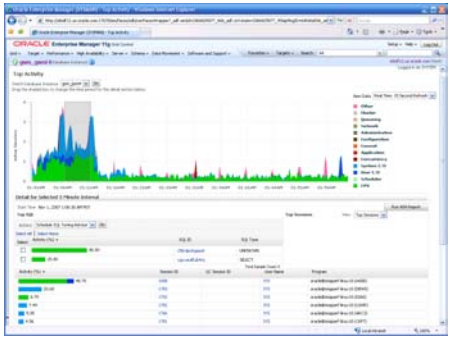
- Exposes monitoring statistics in:
  - V$SQL_MONITOR
    - Cumulative DB time breakdown (CPU, IO, Application, etc)
    - PL/SQL, Java Exec Times
  - V$SQL_PLAN_MONITOR
    - #rows, #executions, memory, temp space per plan operation
    - Plan operation begin and end times
  - V$ACTIVE_SESSION_HISTORY (ASH)
    - Each execution of each SQL identifiable in ASH
      execution key: (SQL_ID, SQL_EXEC_START, SQL_EXEC_ID)
    - Parallel Execution Servers share an execution key with QC, but use a separate Session ID
- Separate entries for each Parallel Execution Server
- Refreshes statistics every second, during query execution
- Statistics available for at least 5 minutes, even with cursor age-outs

How do I use it?

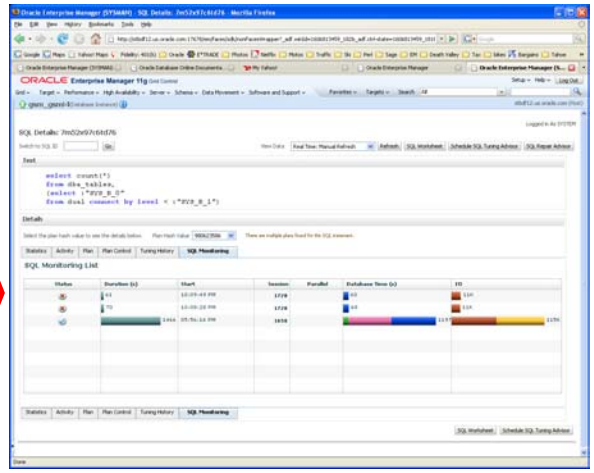
- 11g Enterprise Manager Grid Control
- Additional reporting (available today): DBMS_SQLTUNE.REPORT_SQL_MONITOR

Enterprise Manager Flow (1)

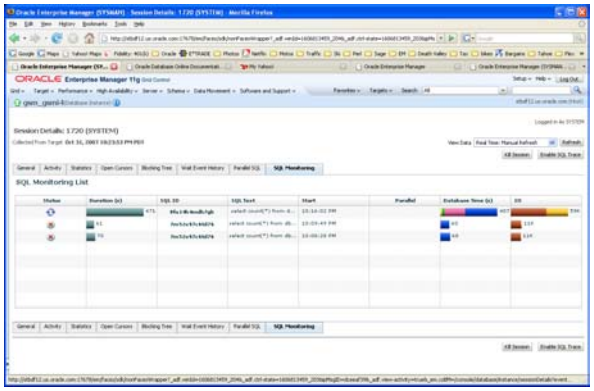
Top Activity



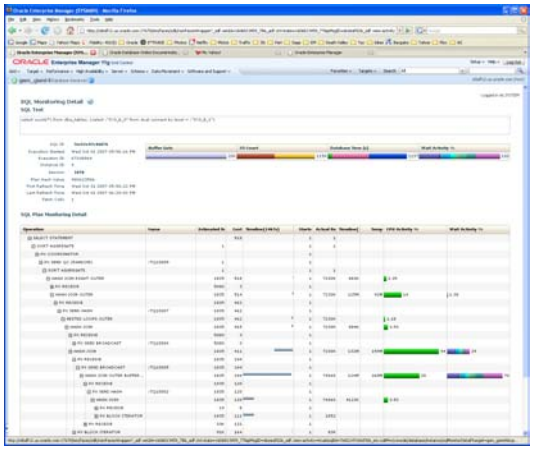
SQL Details



Session Details

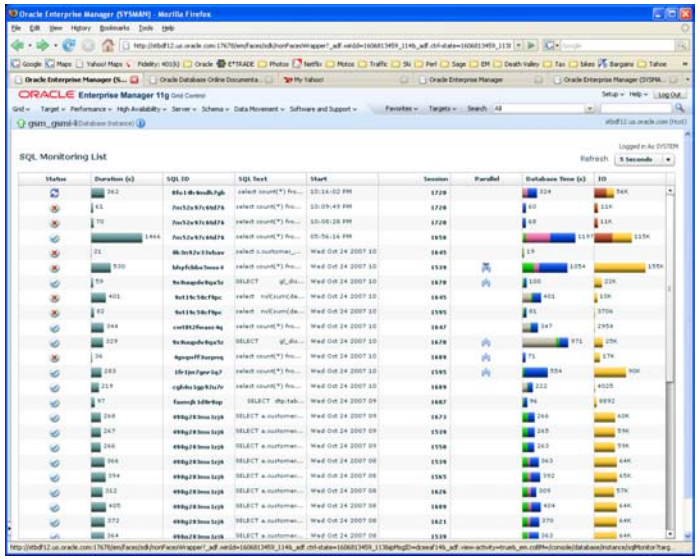


Monitoring Details

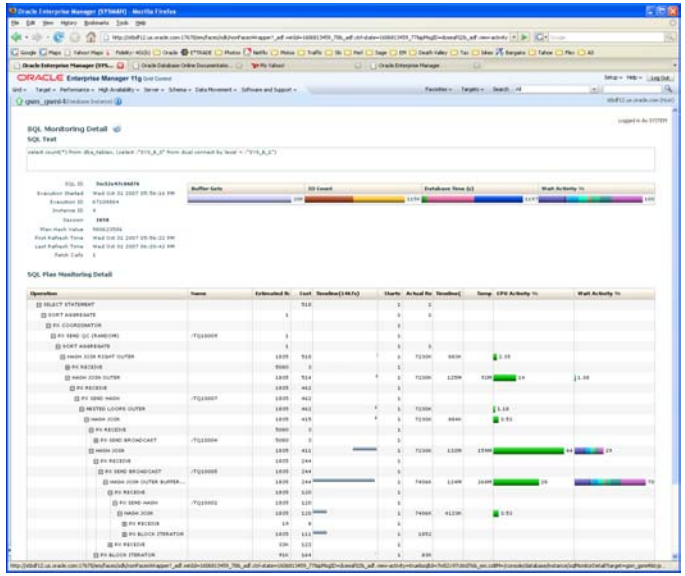


Enterprise Manager Flow (2)

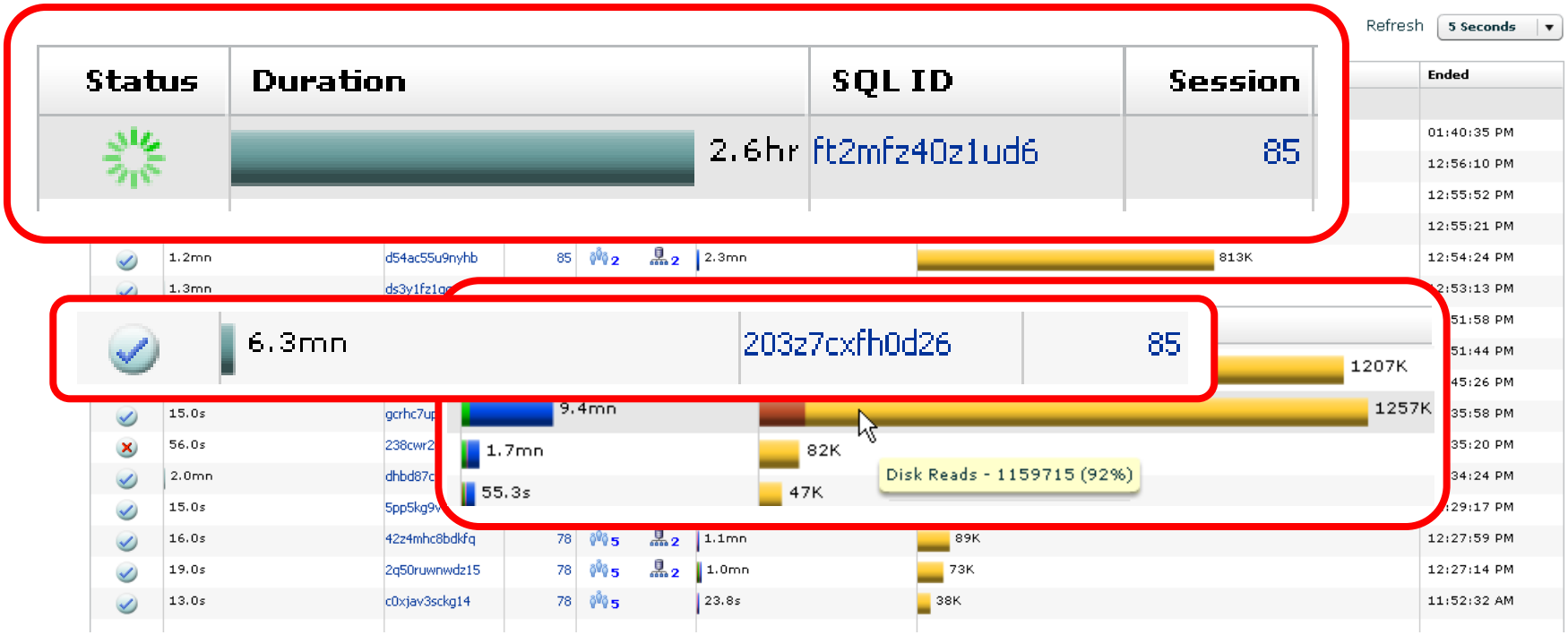
Monitoring List



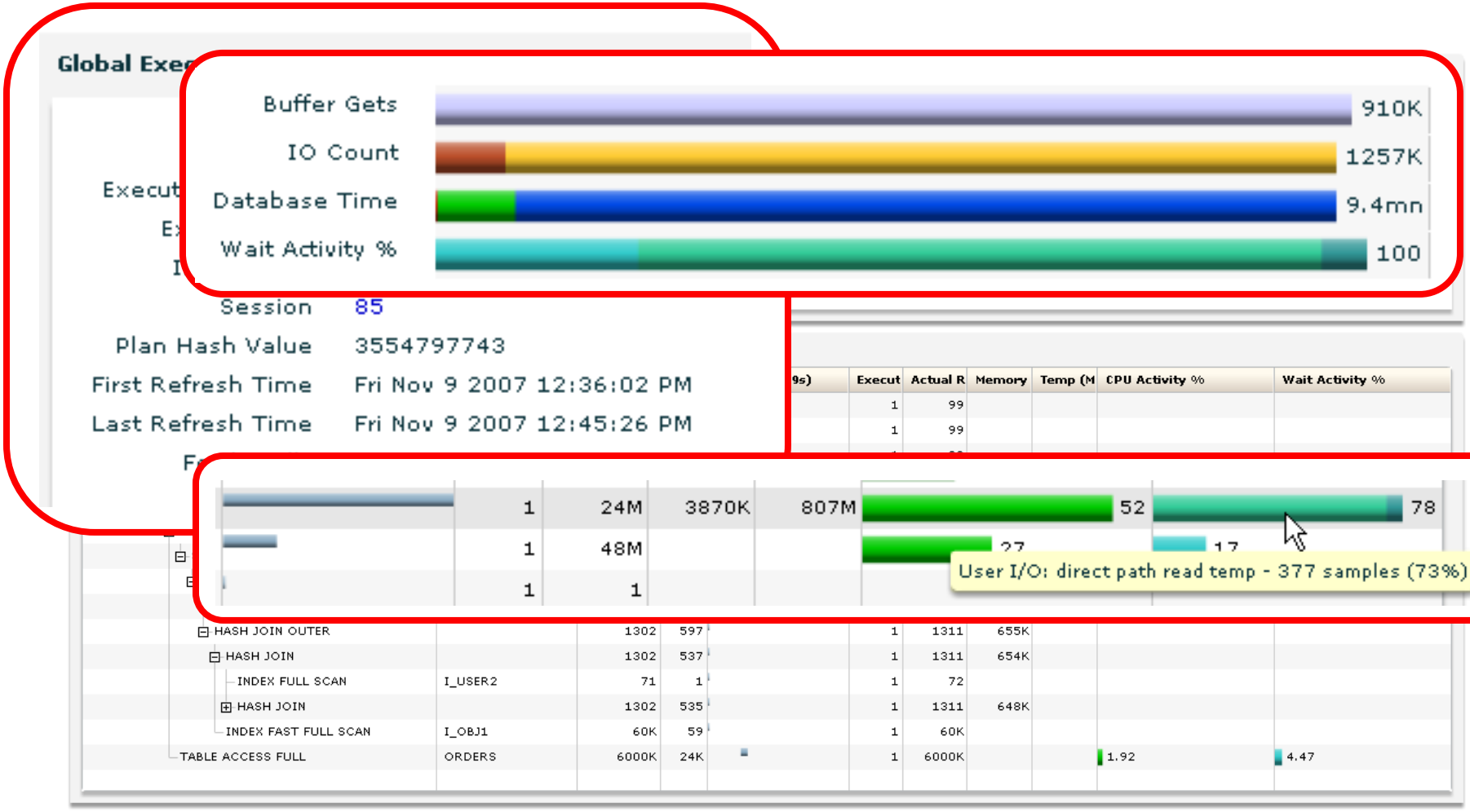
Monitoring Details



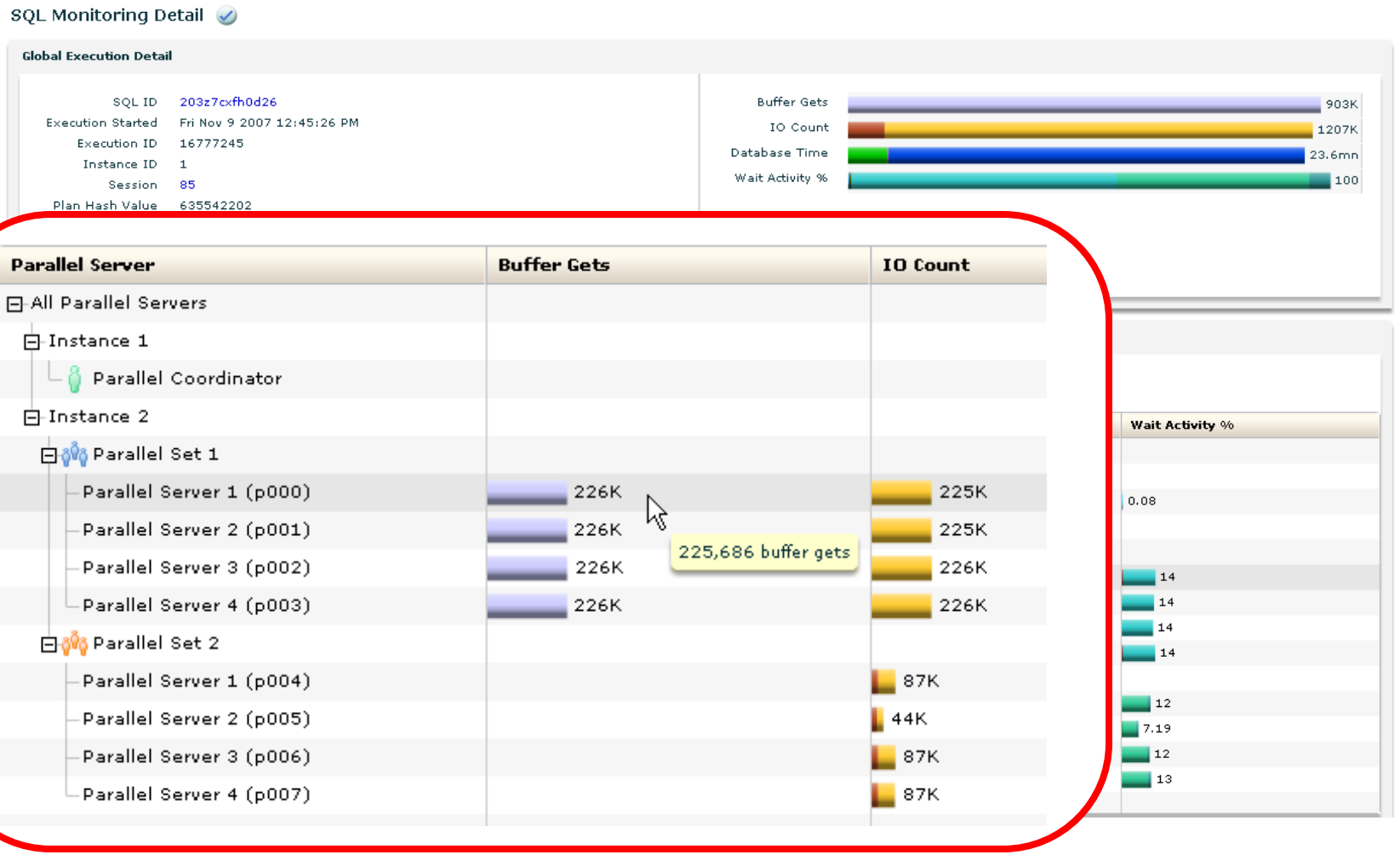
SQL Monitoring List



SQL Monitoring Details



SQL Monitoring Details (Parallelism)



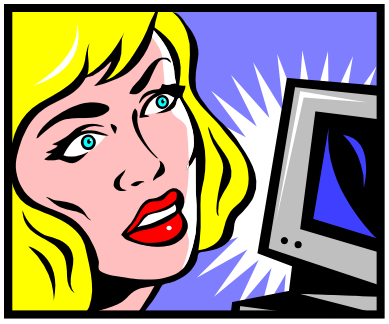
Conclusion

- Real-Time SQL Monitoring is
  - Monitoring and tuning for high response-time SQLs
  - New, fine-grained SQL statistics
    - tracked automatically
    - updated while the SQL runs
    - highly visible and accessible
    - at no cost to your production system
  - The only way to know what's happening inside single SQL execution
  - The quickest way to the root cause of a performance problem: If you can find the problem, you can fix it!

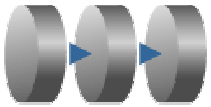
Partition Advisor

Problem

- SQLs on large tables run too long or timeout
- High I/O counts
- Too much pressure on buffer pool
- Disgruntled users
- Low transaction rates
- Too many complex SQLs to figure out on my own
- Put out a fire here, another starts over there

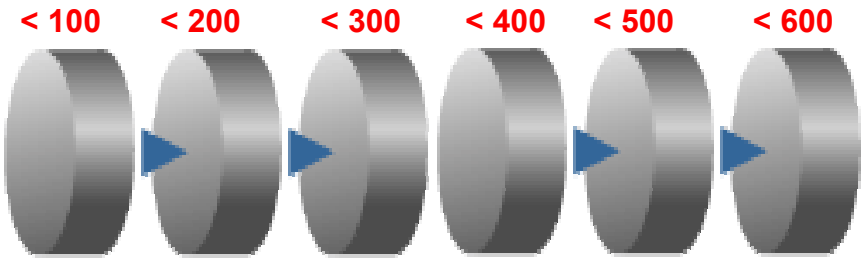


Solution



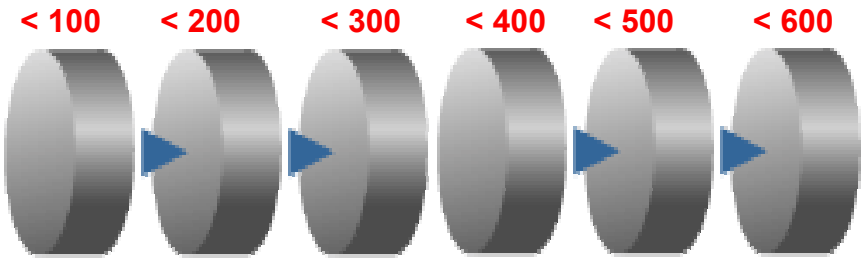
- Get new 11g partition advice along with other advice from the new 11g SQL Access Advisor
  - Recommendations targeted at partition elimination in query processing
  - Recommendations to aid certain join processing

Interval Partitioning




```
CREATE TABLE emp
   (empno NUMBER(6),
    first_name VARCHAR(20),
    last_name VARCHAR(20),
    deptno NUMBER(6))
PARTITION BY RANGE (deptno) INTERVAL 100
PARTITION p1 VALUES LESS THAN 100
```

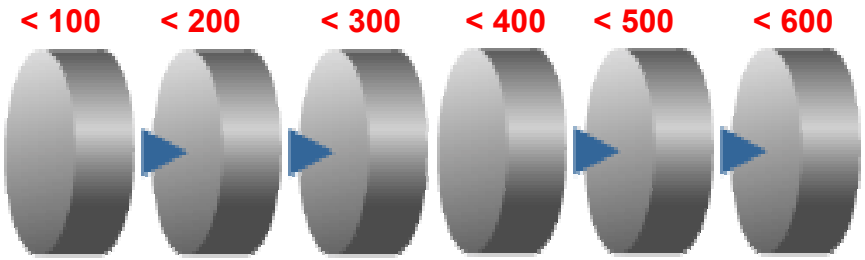
Interval Partitioning




```
CREATE TABLE emp
   (empno NUMBER(6),
    first_name VARCHAR(20),
    last_name VARCHAR(20),
    deptno NUMBER(6))
PARTITION BY RANGE (deptno) INTERVAL 100
PARTITION p1 VALUES LESS THAN 100
```

Interval partition is a new, automated form of range partitioning.

Partition Elimination

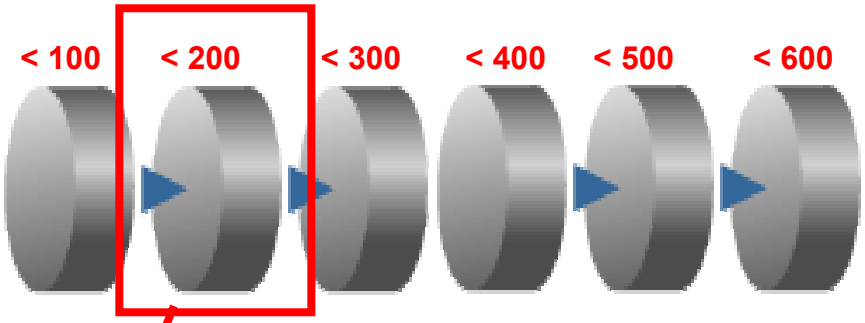



```
CREATE TABLE emp
   (empno NUMBER(6),
    first_name VARCHAR(20),
    last_name VARCHAR(20),
    deptno NUMBER(6))
PARTITION BY RANGE (deptno) INTERVAL 100
PARTITION p1 VALUES LESS THAN 100
```

```
SELECT empno, last_name, first_name
  FROM emp
  WHERE deptno = 123
```

Partition Elimination

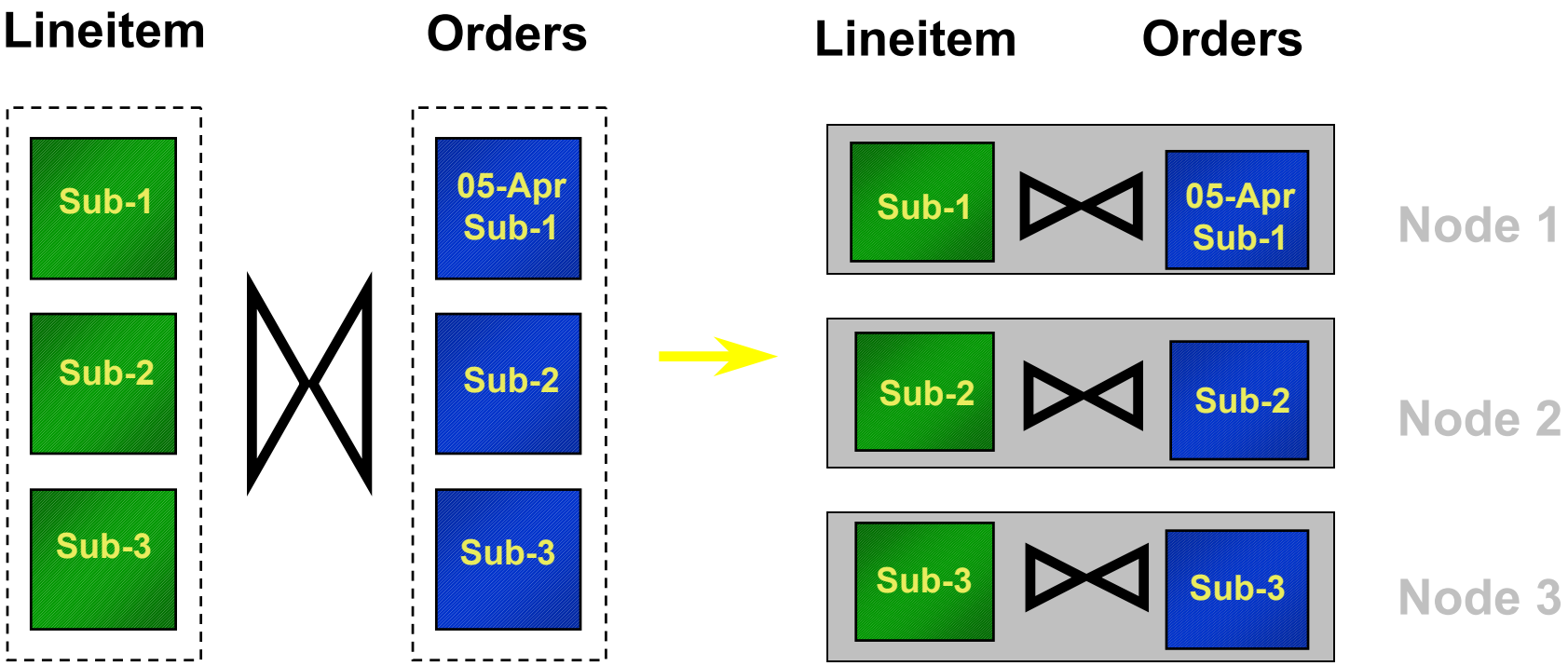
```
CREATE TABLE emp
   (empno NUMBER(6),
    first_name VARCHAR(20),
    last_name VARCHAR(20),
    deptno NUMBER(6))
PARTITION BY RANGE (deptno) INTERVAL 100
PARTITION p1 VALUES LESS THAN 100
```



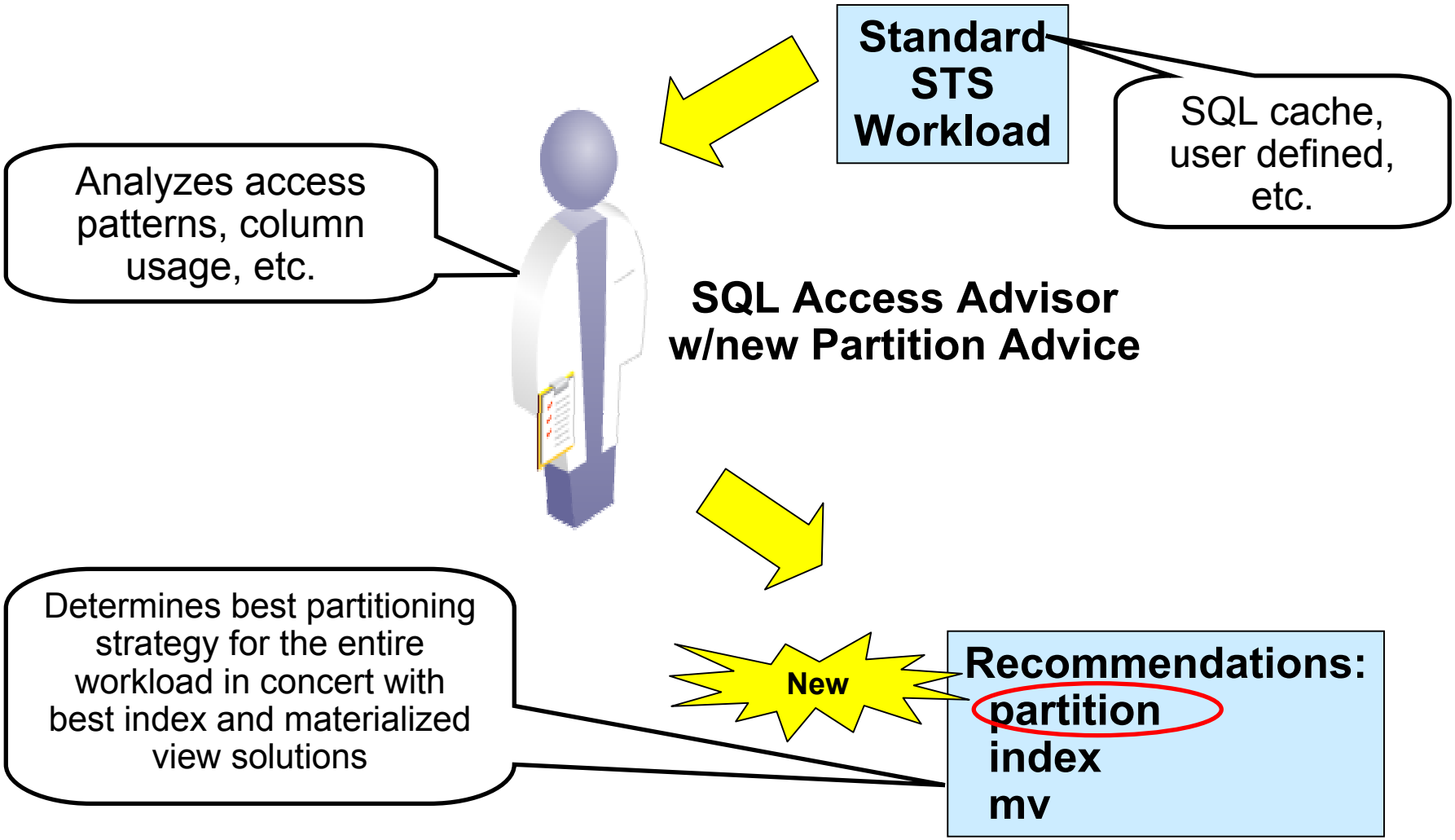

```
SELECT empno, last_name, first_name
  FROM emp
  WHERE deptno = 123
```

Partition-wise Join

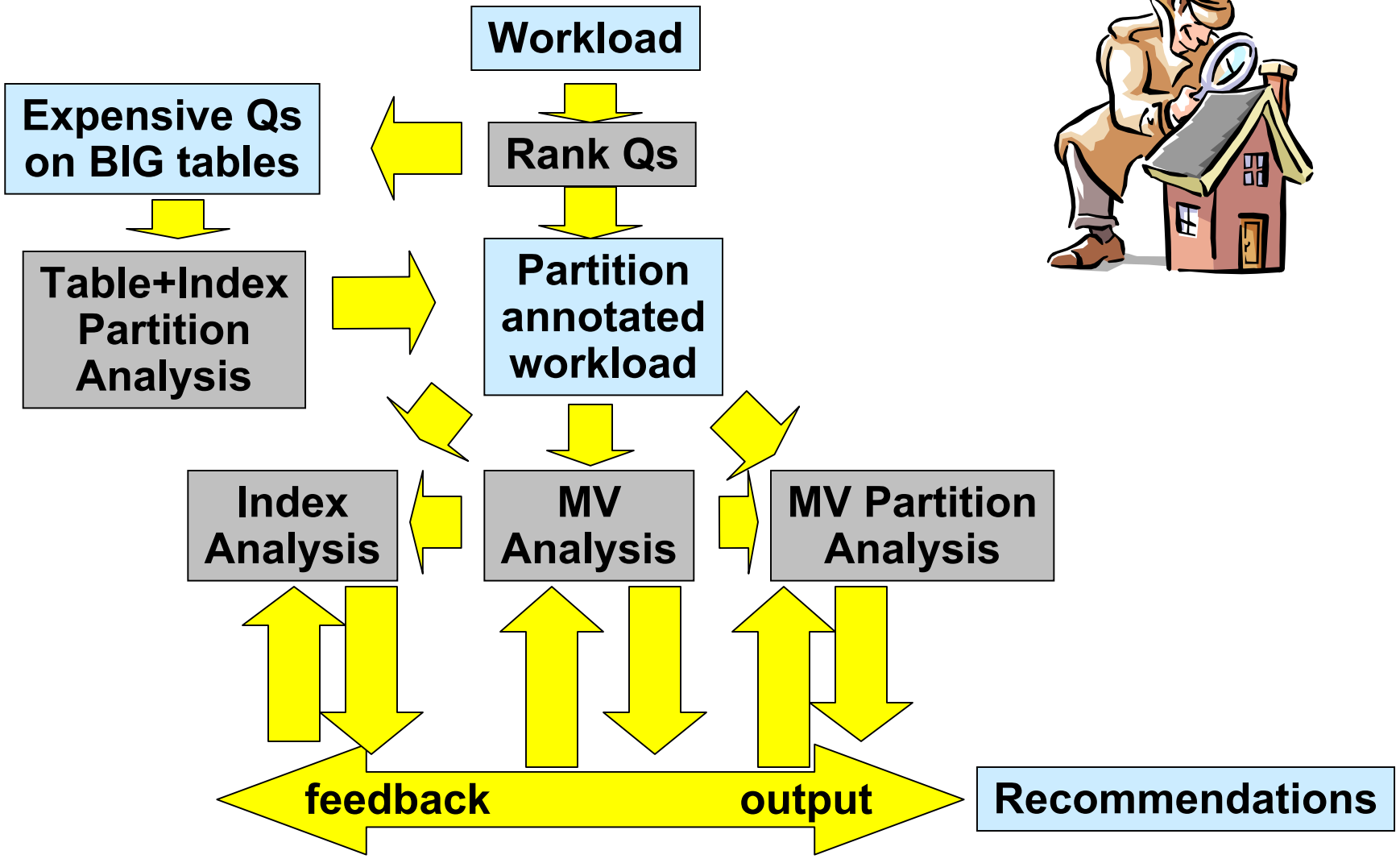
When joining two tables that are partitioned on the join-key, Oracle may choose to join on a per-partition basis.



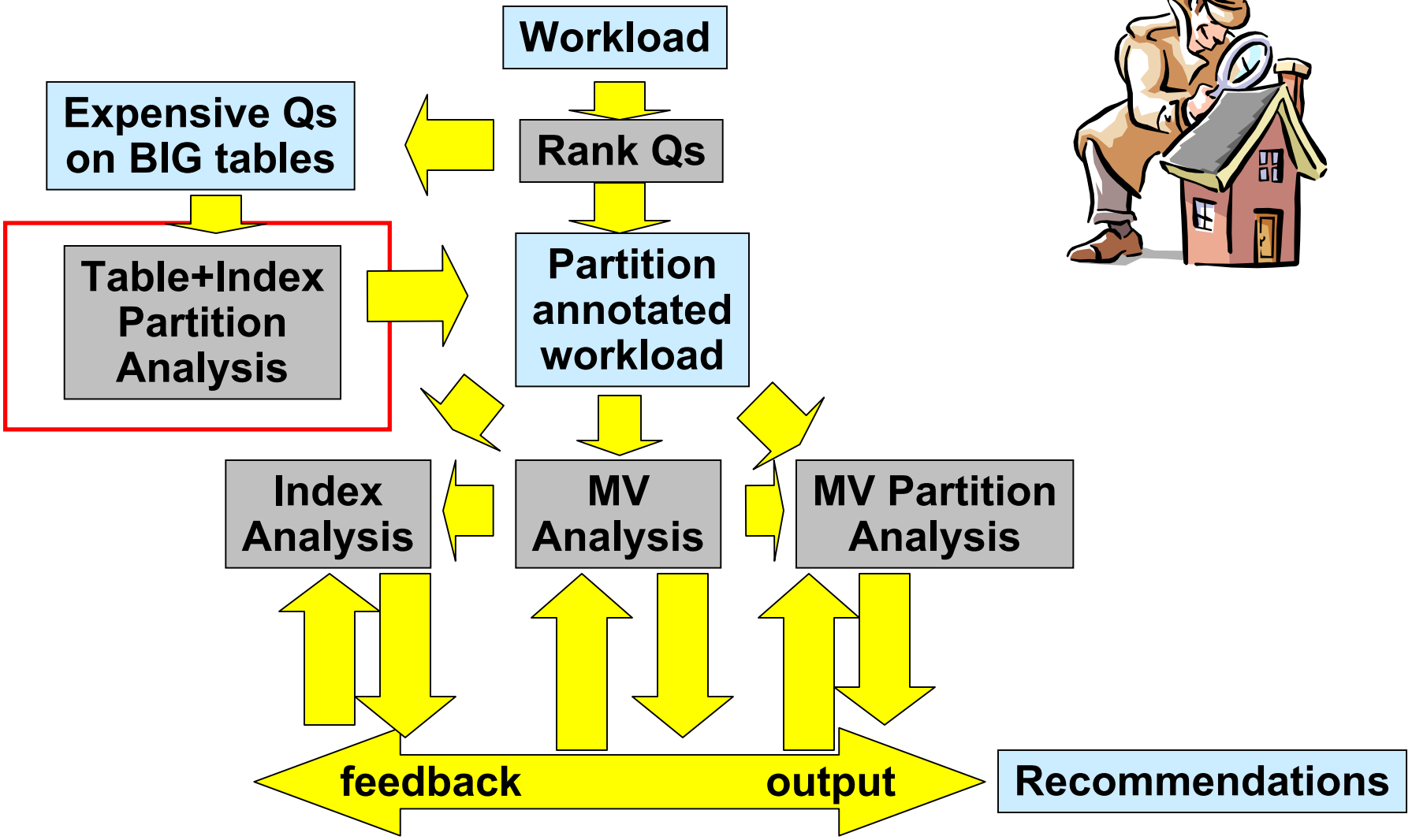
How does SAA work?



How does SAA work?



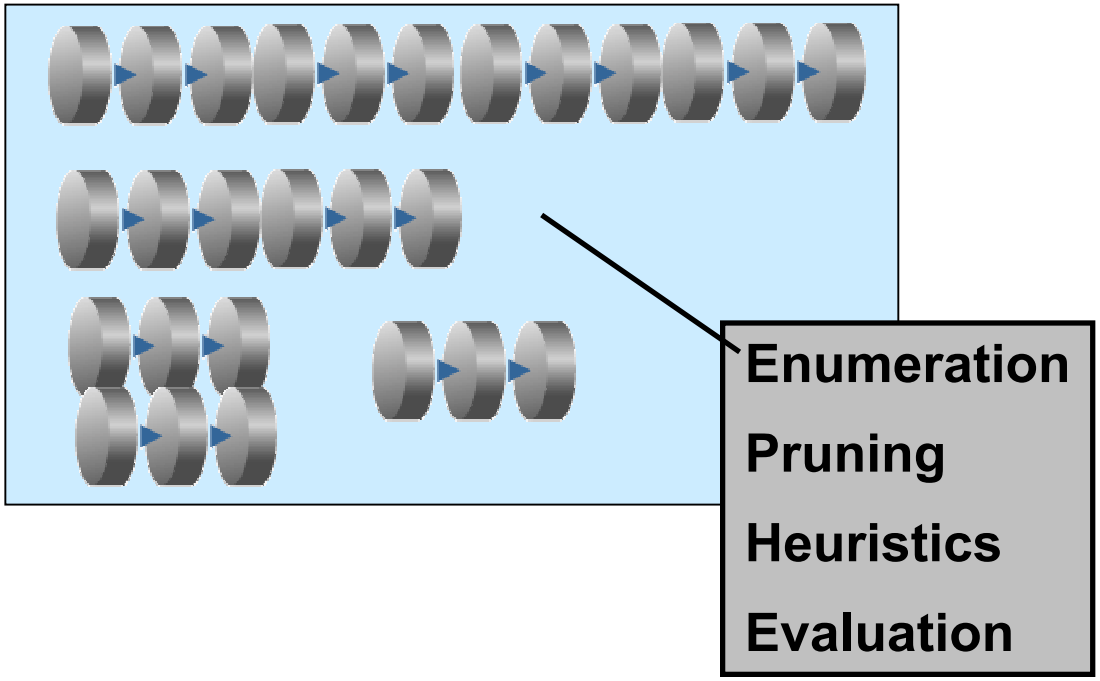
How does SAA work?



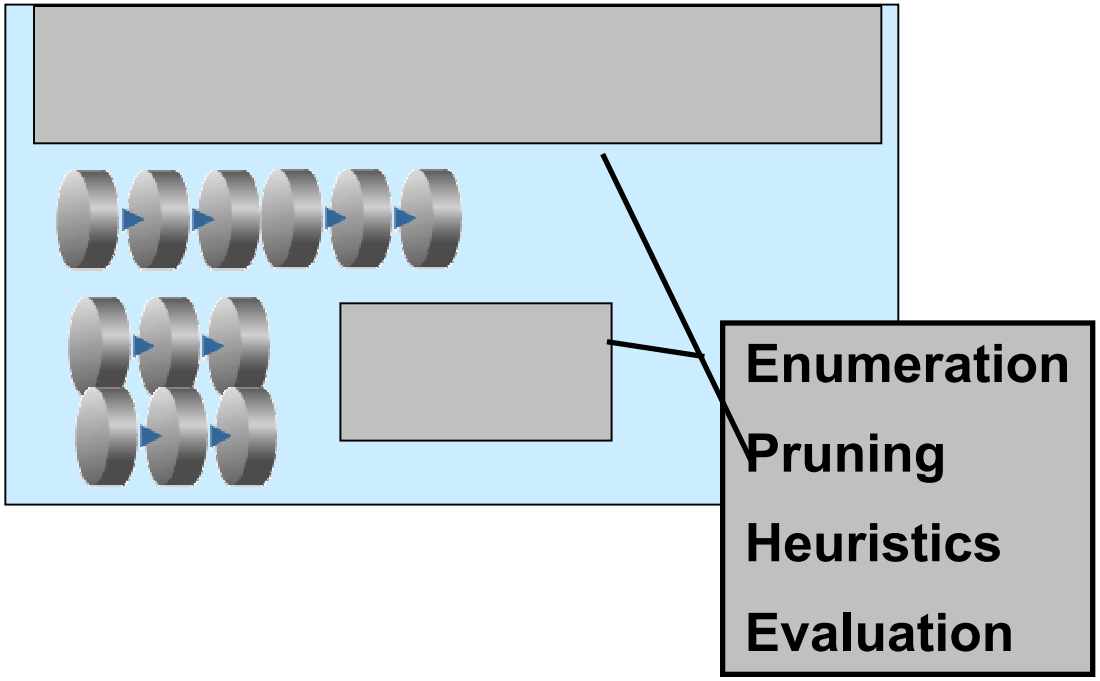
Partition Advisor Problem Space

- Fact: If I partition table T1, all Qs referencing T1 are affected (+ or -)
- Fact: If I also partition table T2, the same applies
- Fact: Lots of Qs reference multiple tables forming a network of inter-relationships
- Therefore: A potential partitioning scheme on each different table affects each potential partitioning scheme on other tables in that network

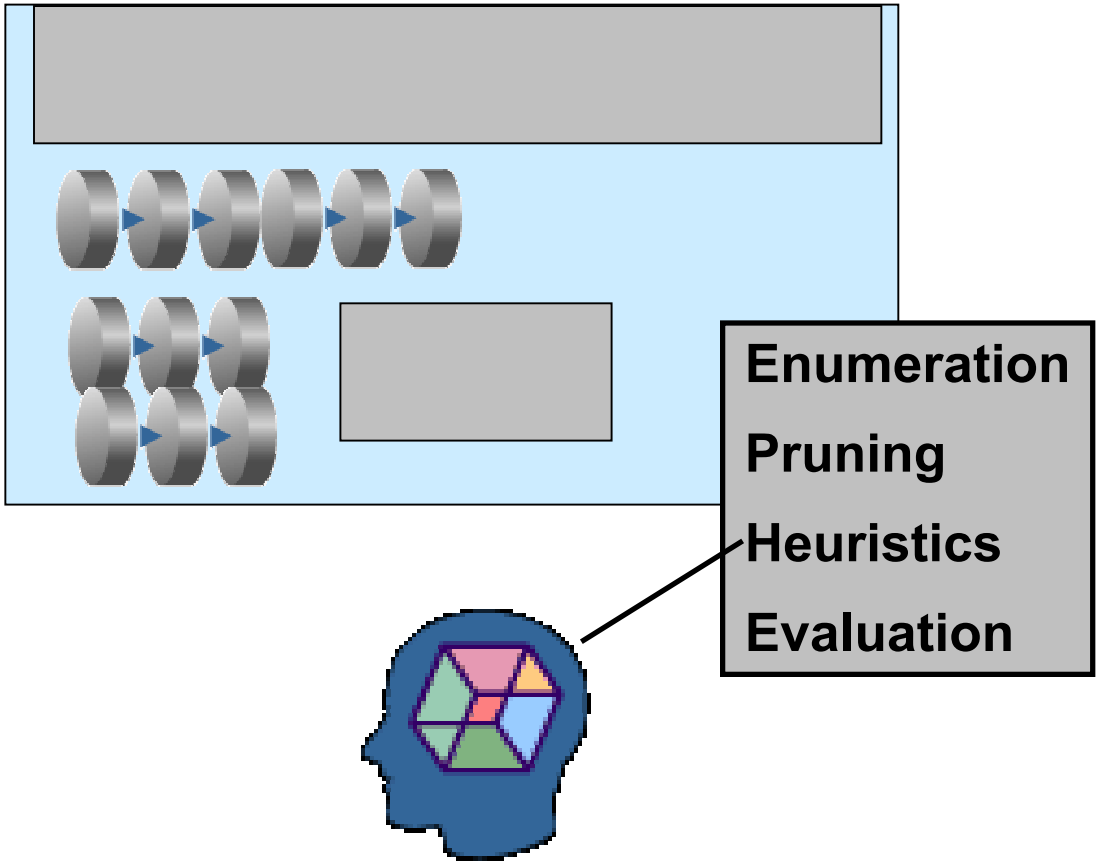
Partition Advisor



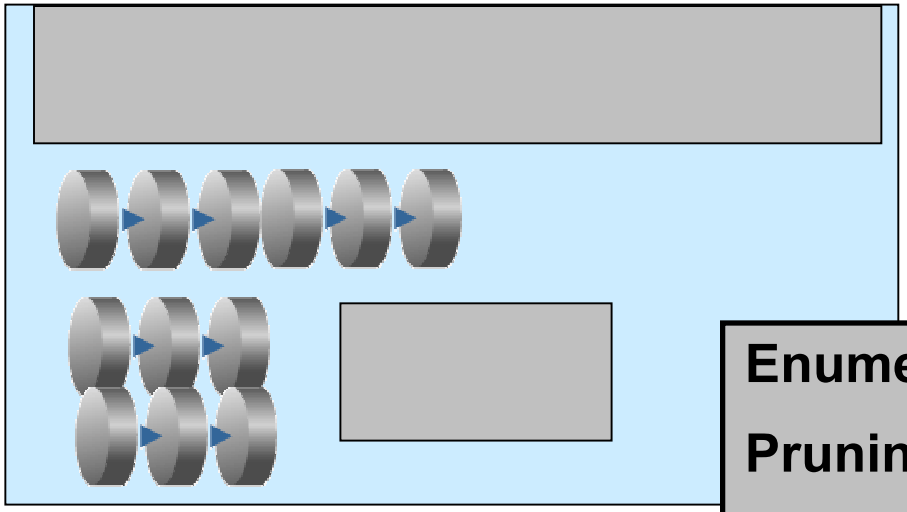
Partition Advisor



Partition Advisor



Partition Advisor

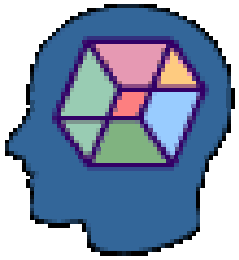


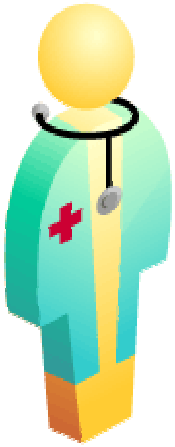
Enumeration

Pruning

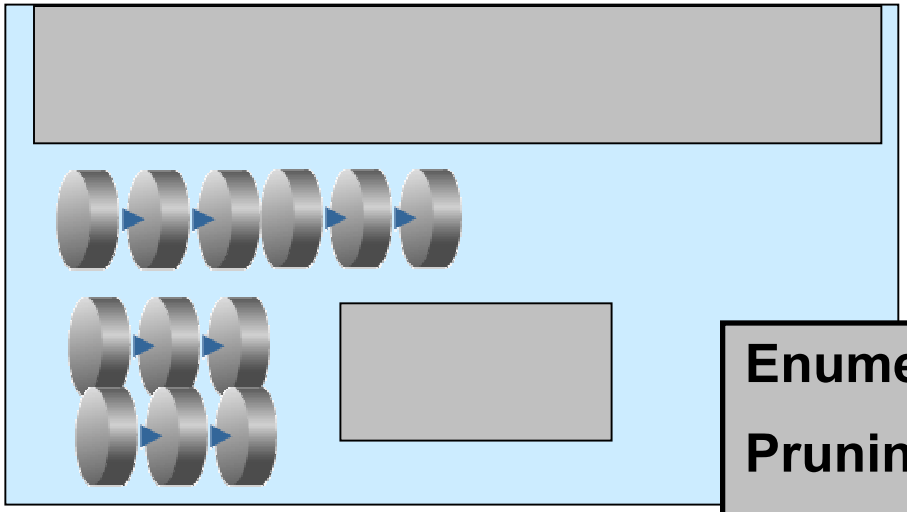
Heuristics

Evaluation





Partition Advisor

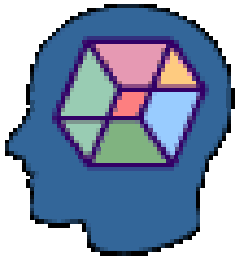


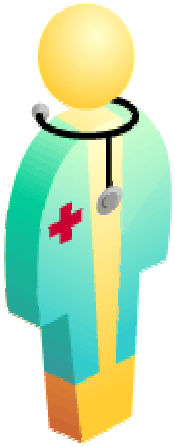
Enumeration

Pruning

Heuristics

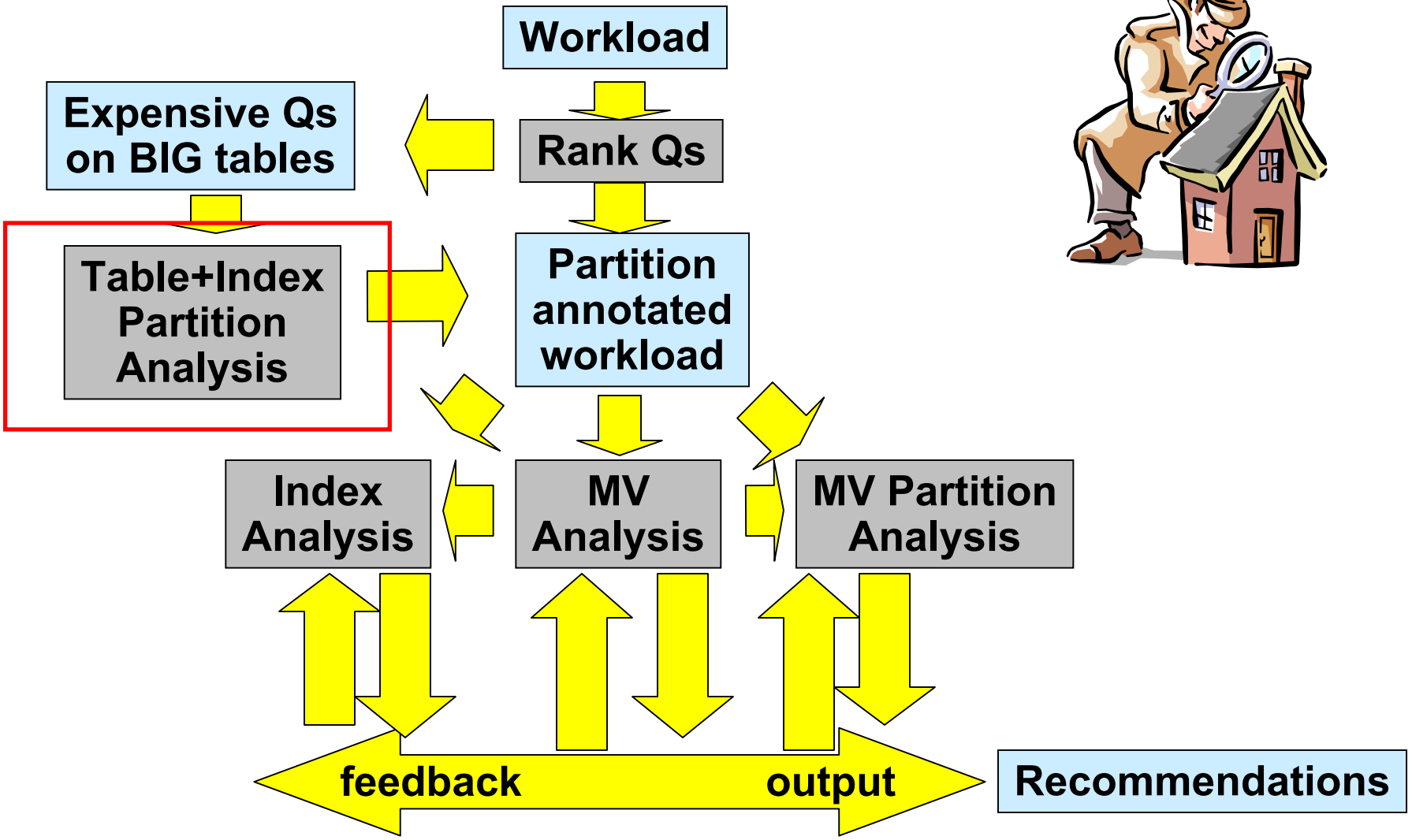
Evaluation



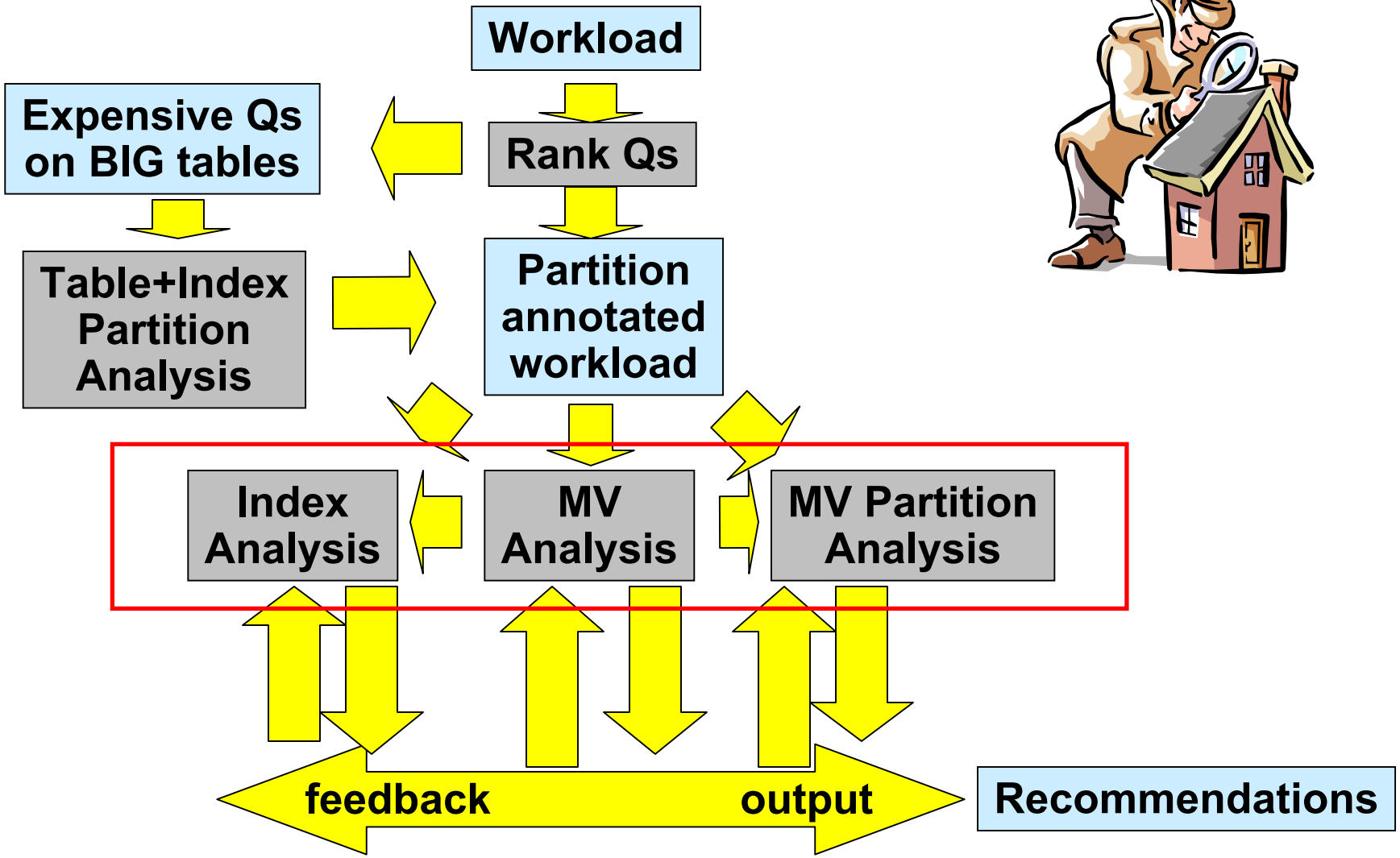


Partition Annotations

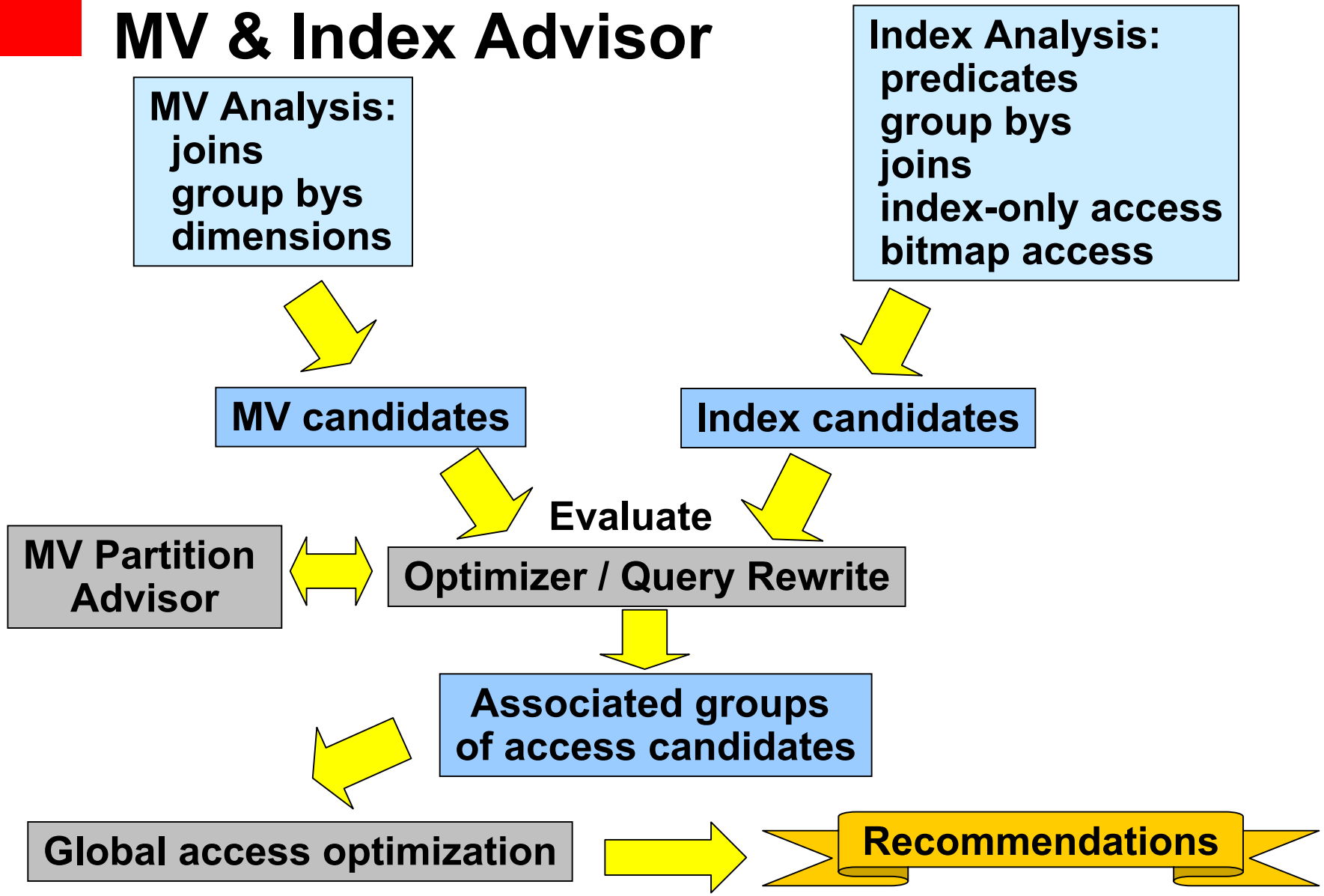
How does SAA work?



How does SAA work?



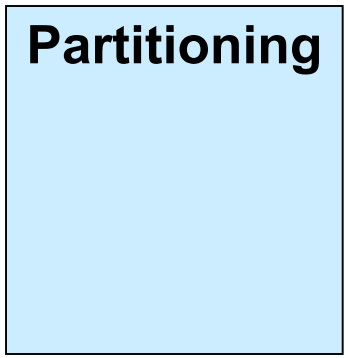
MV & Index Advisor



What does SAA do?

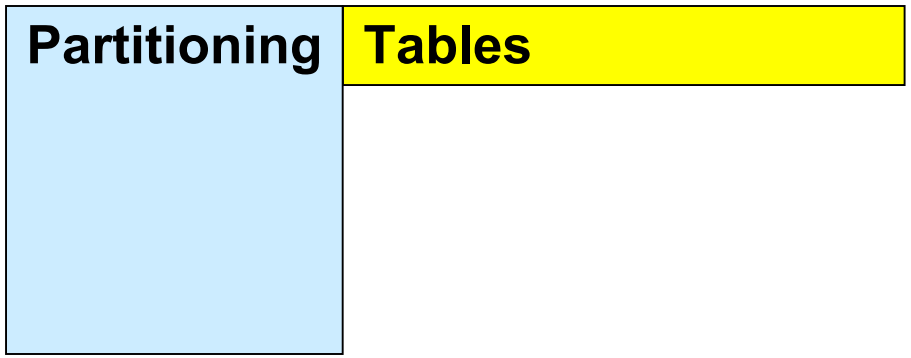
What does SAA do?

Recommends:



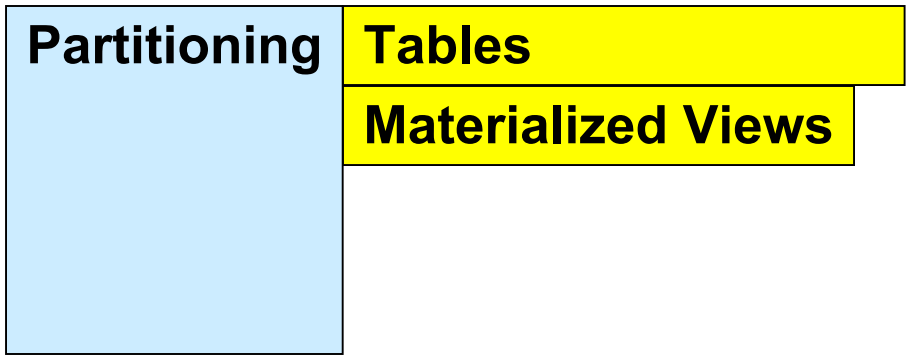
What does SAA do?

Recommends:



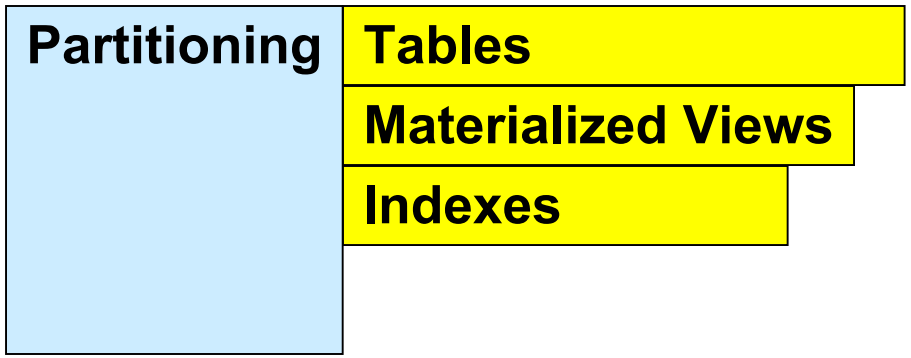
What does SAA do?

Recommends:



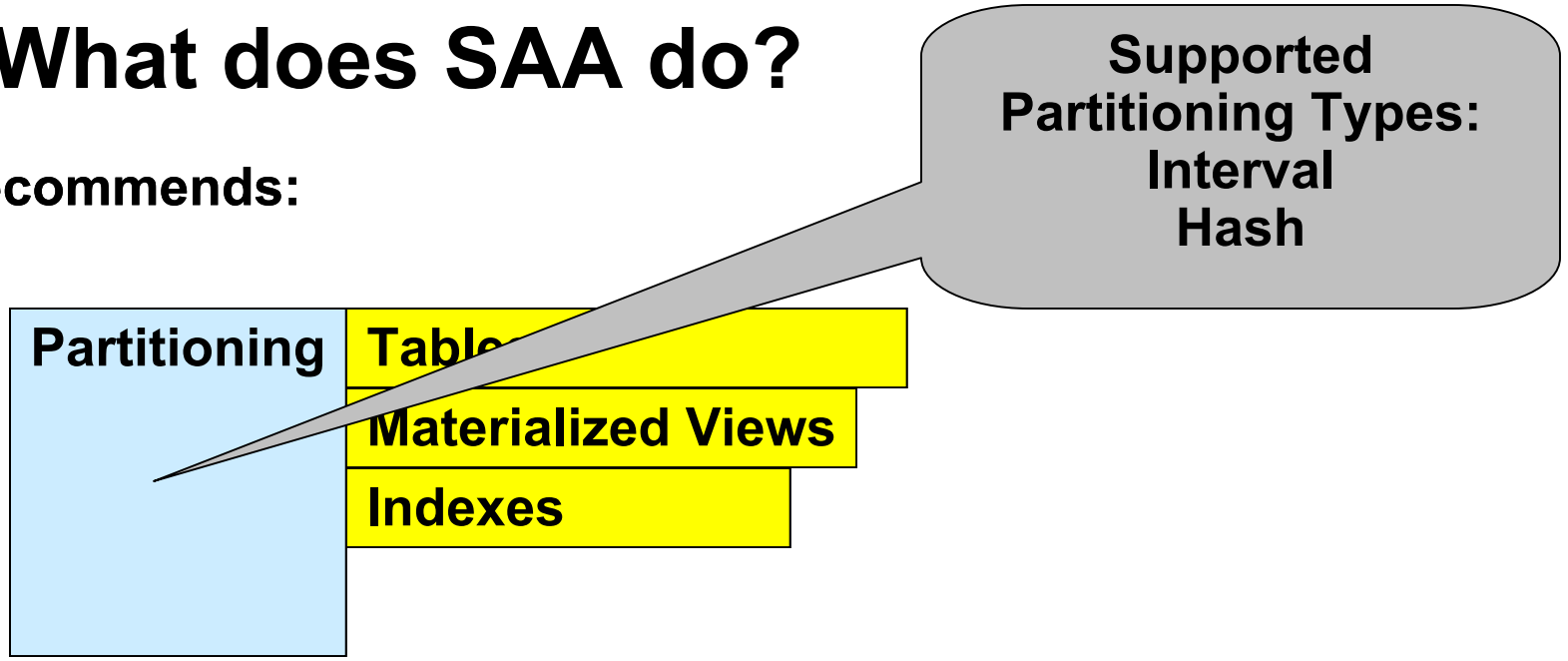
What does SAA do?

Recommends:



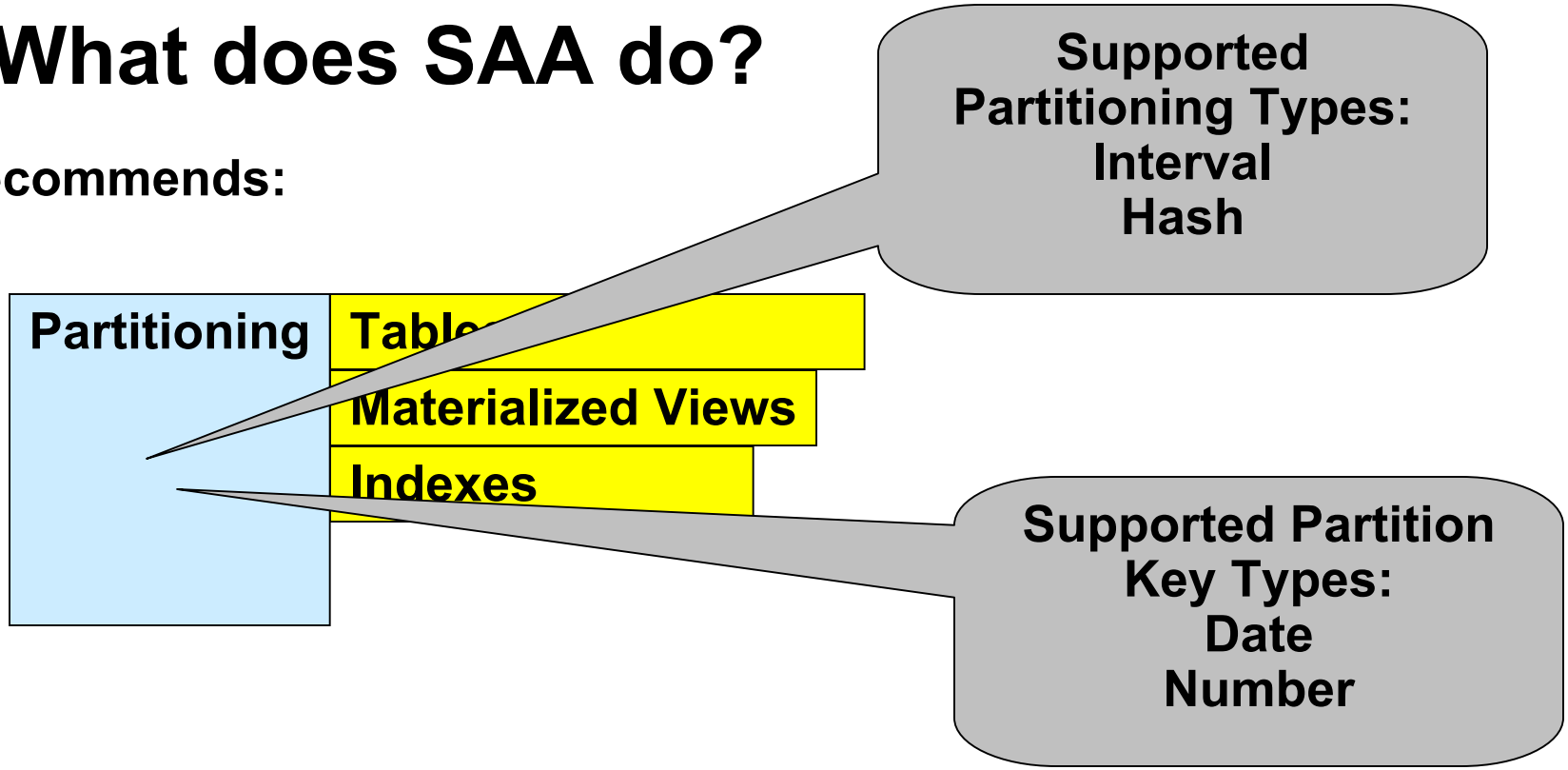
What does SAA do?

Recommends:



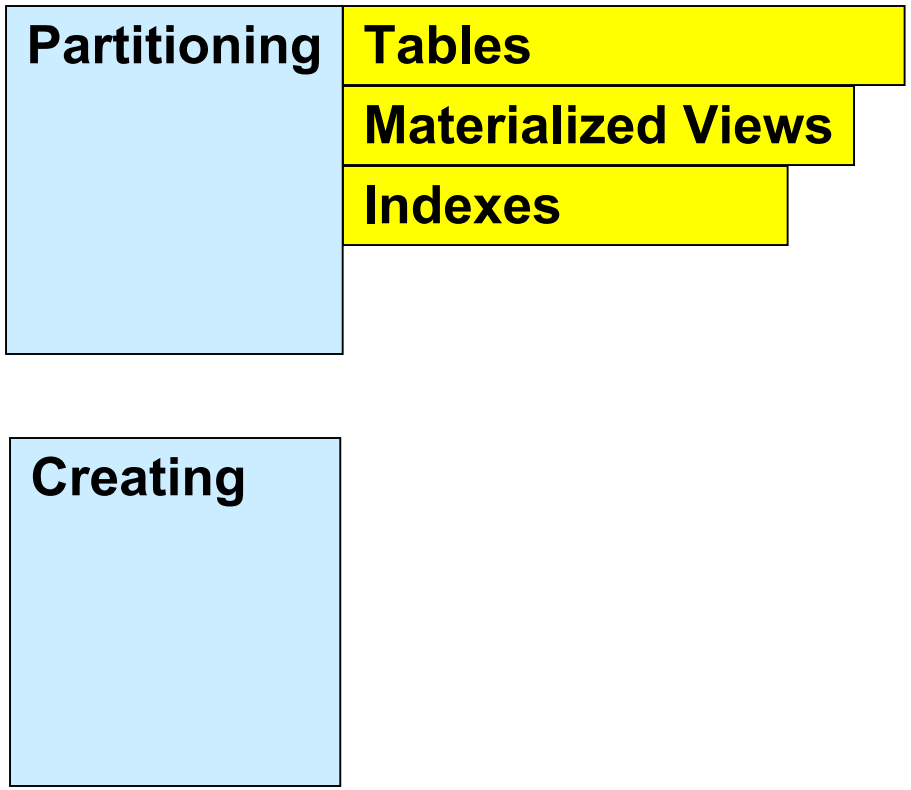
What does SAA do?

Recommends:



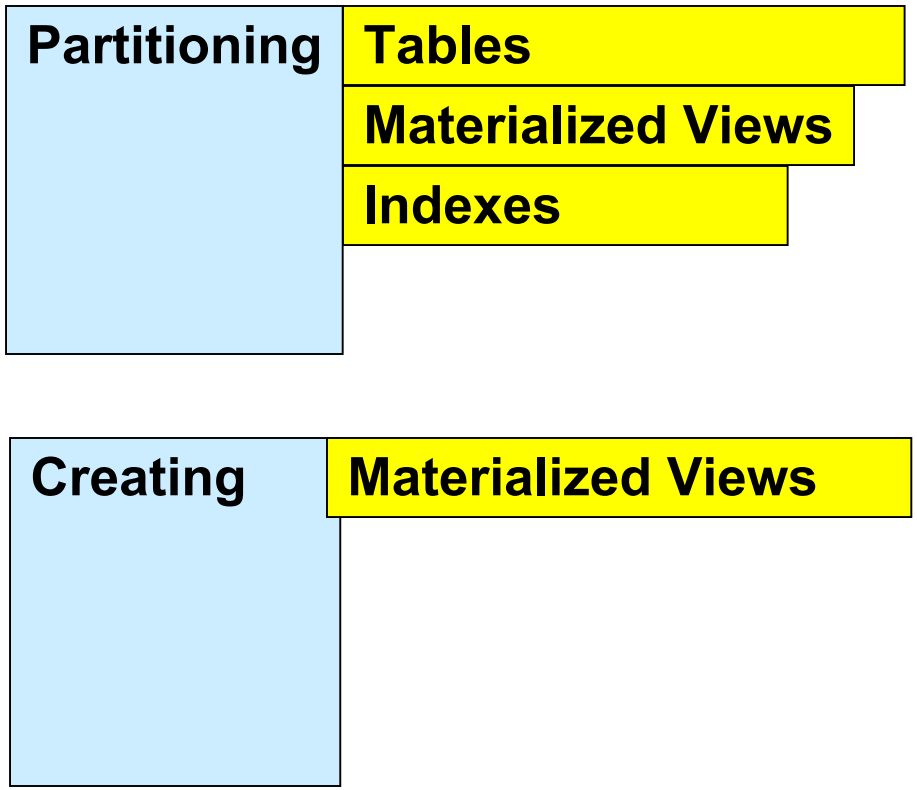
What does SAA do?

Recommends:



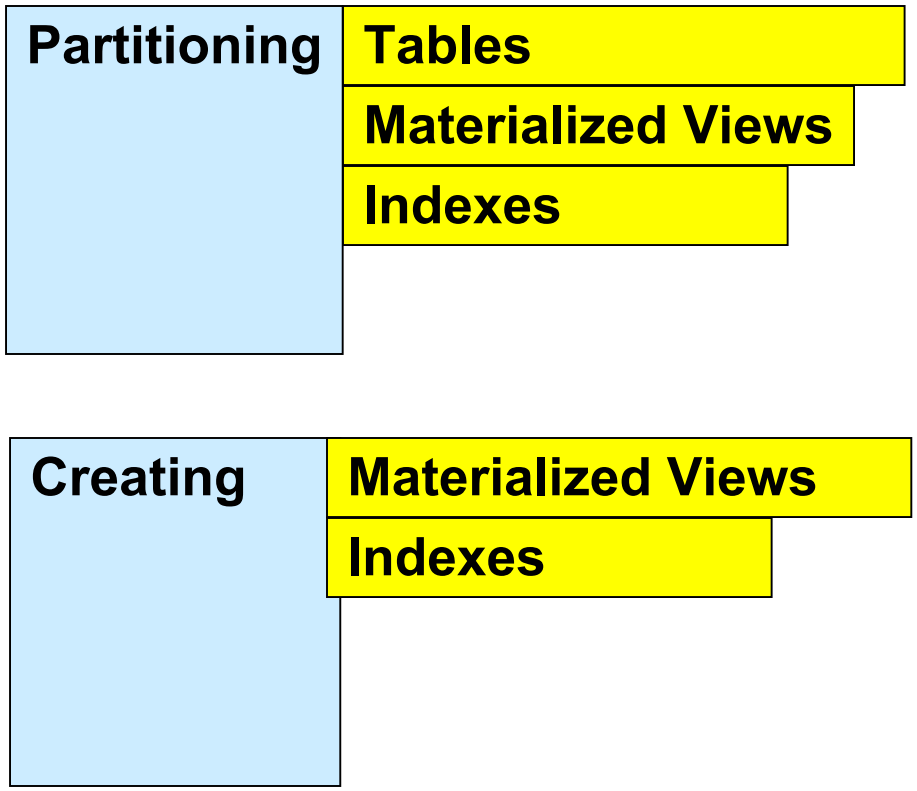
What does SAA do?

Recommends:



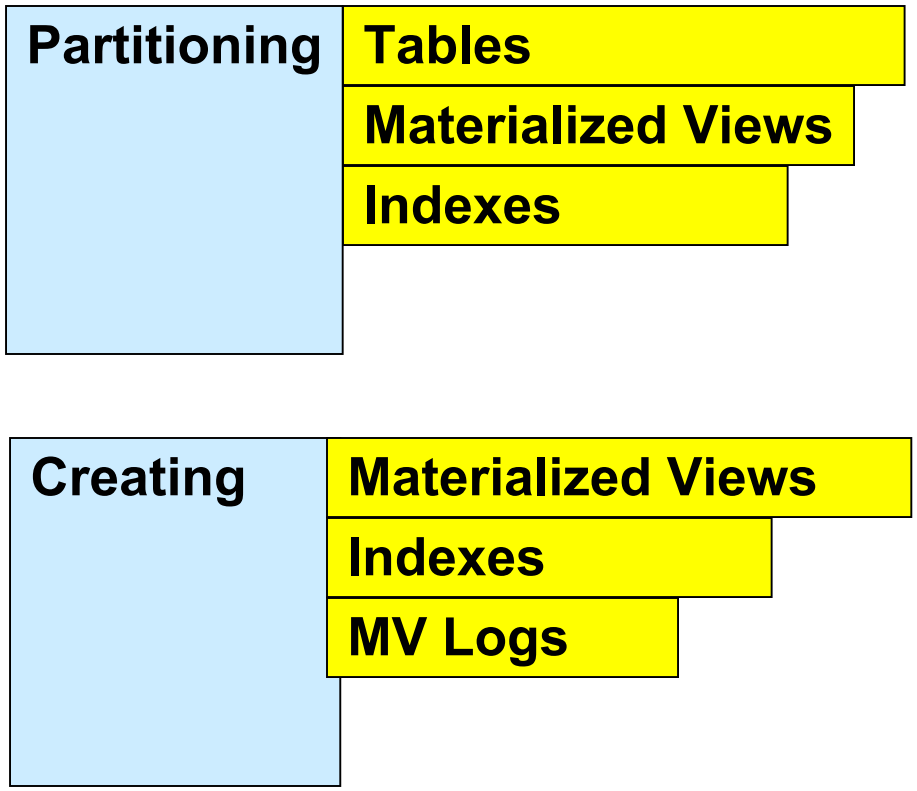
What does SAA do?

Recommends:

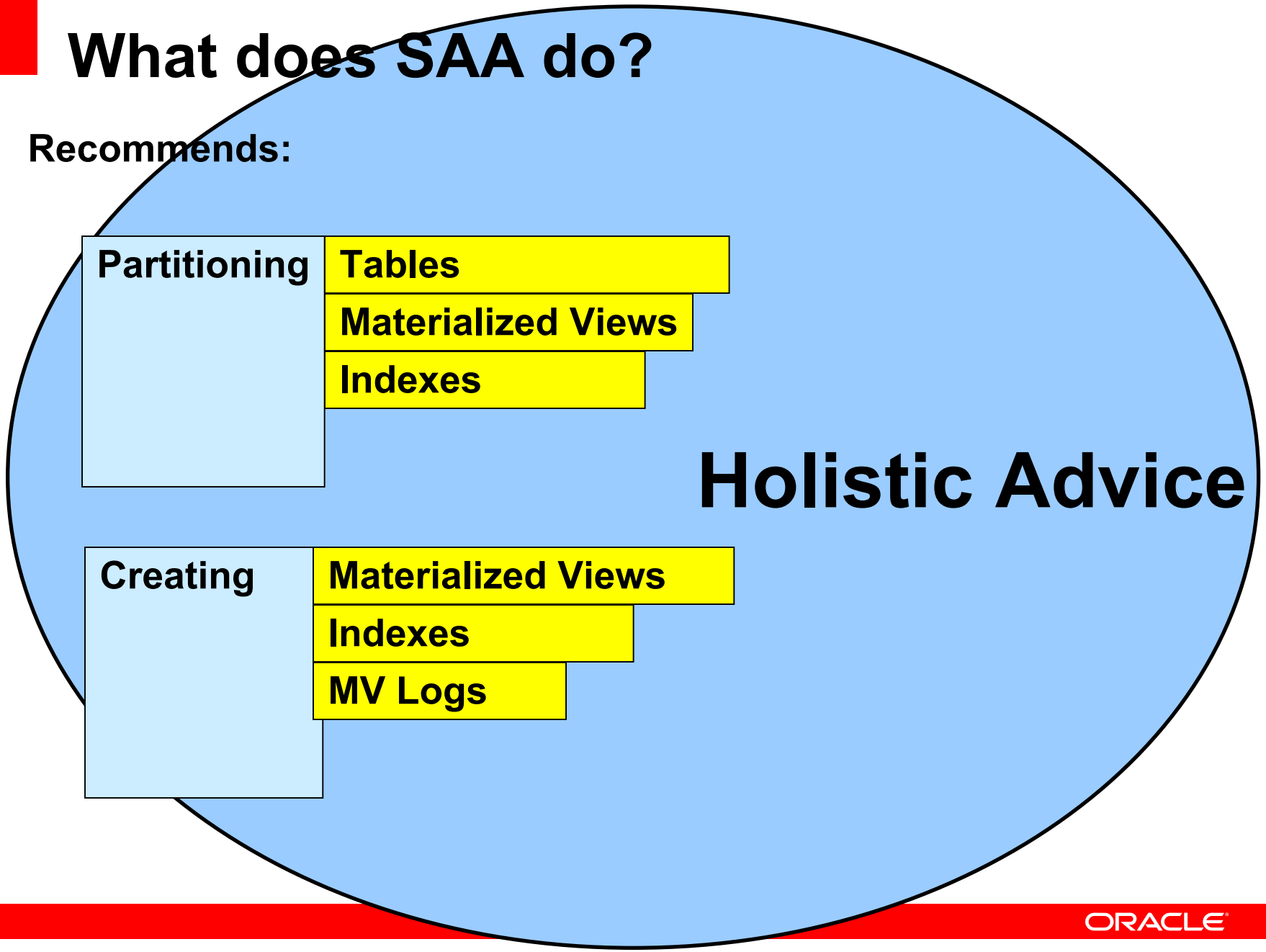


What does SAA do?

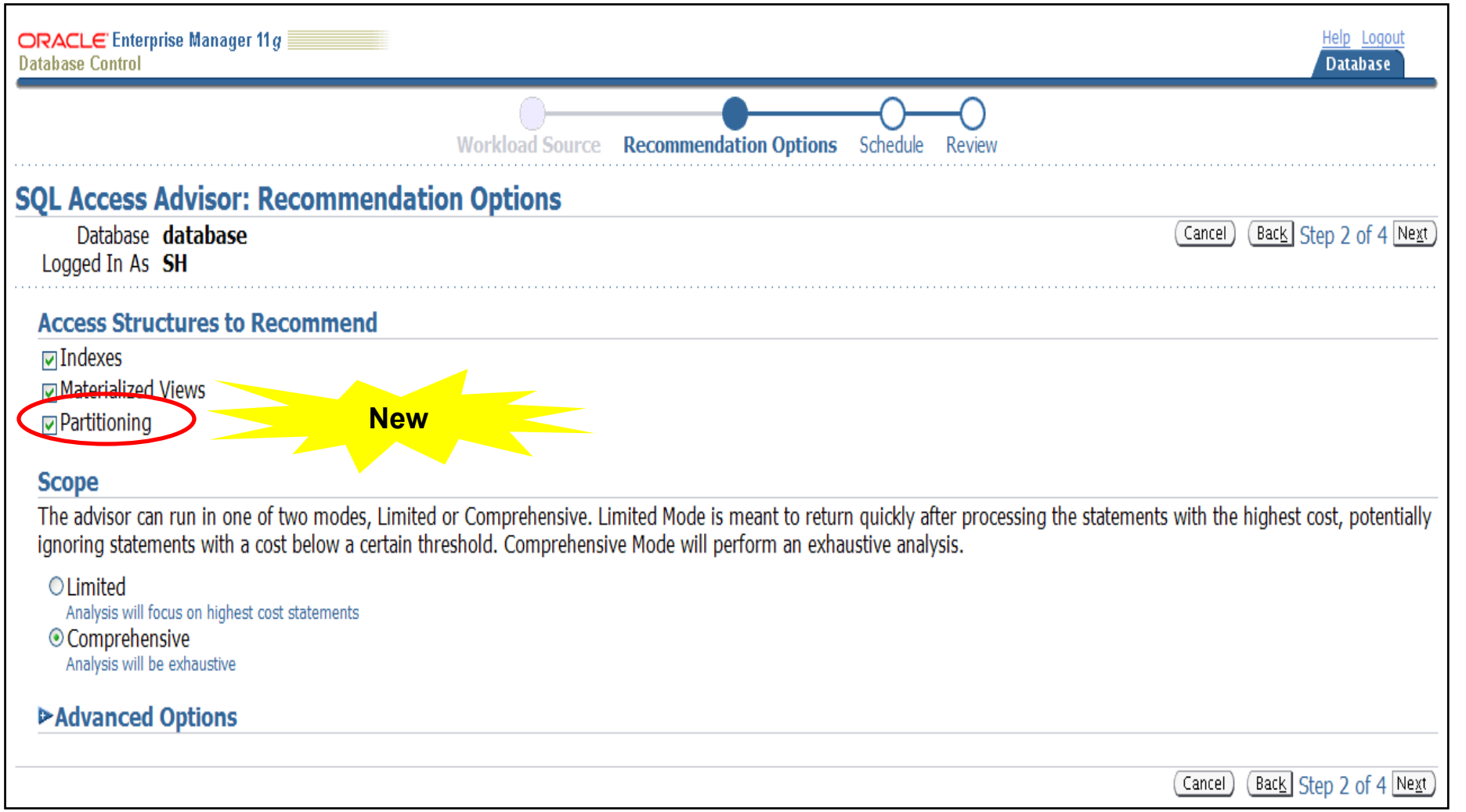
Recommends:



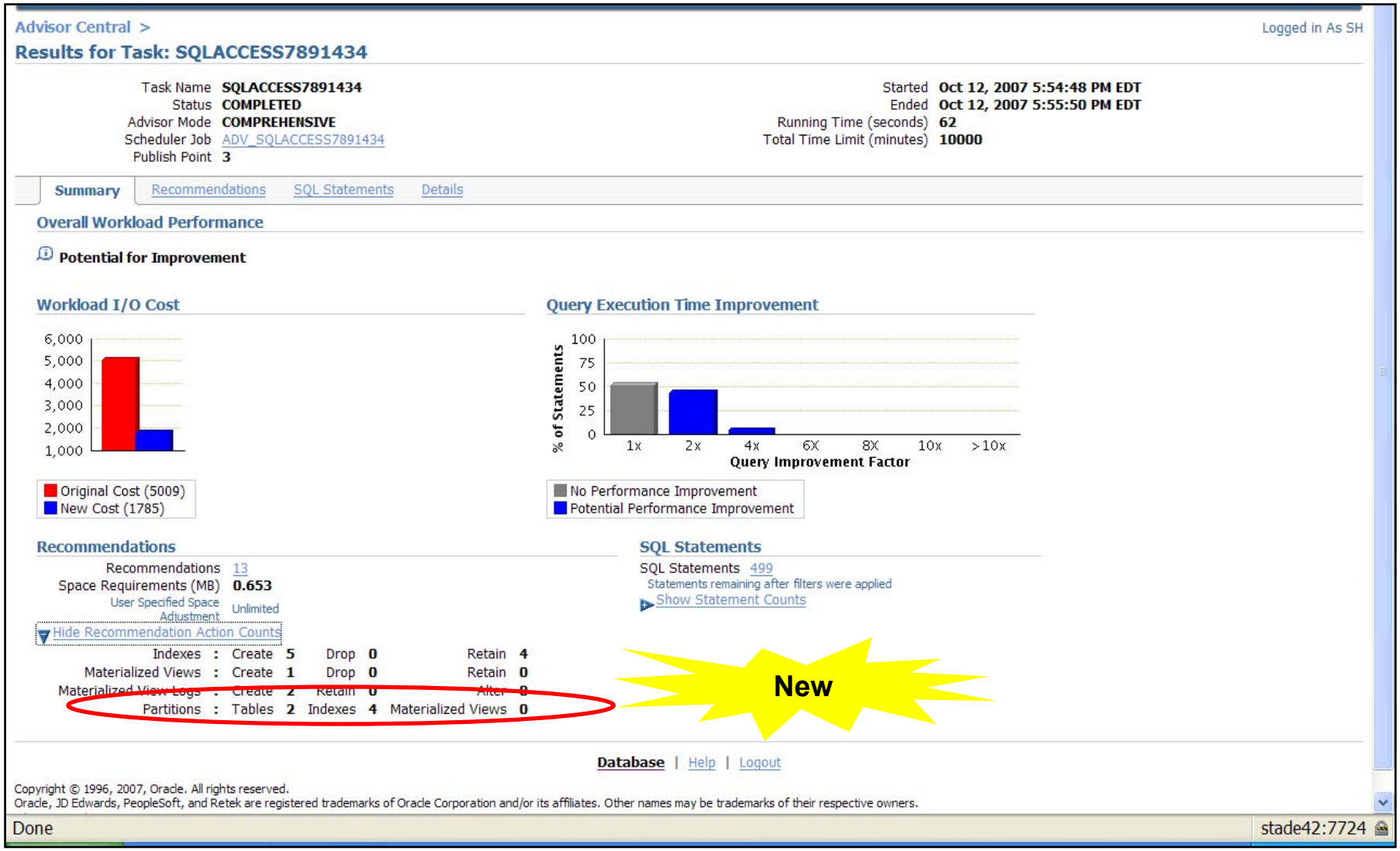
What does SAA do?



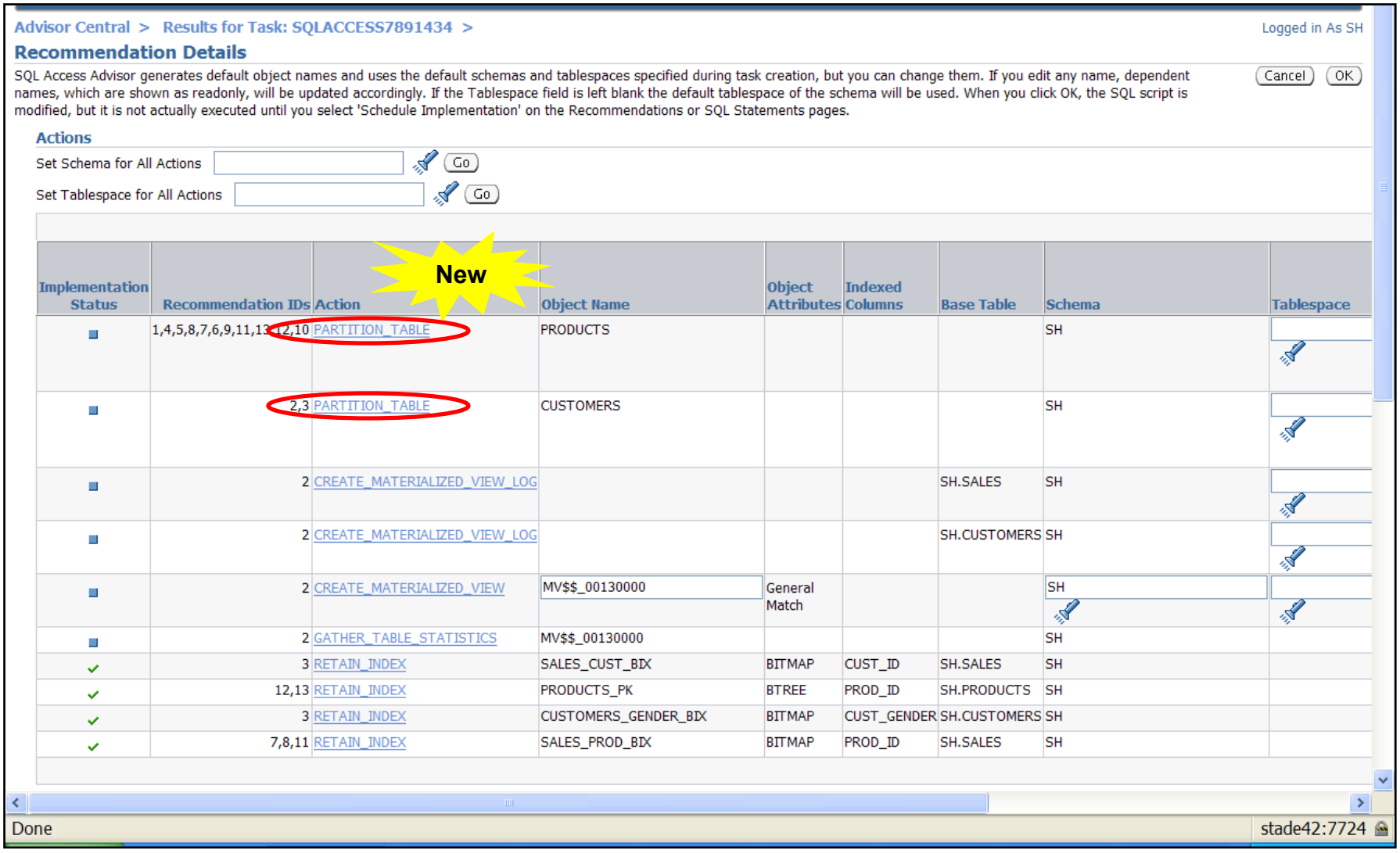
Choosing Partition Advice



Recommendation summary



Partition recommendations



Partition Recommendation

Recommendation: 1

SQL Access Advisor generates default object names and uses the default schemas and tablespaces specified during task creation, but you can change them. If you edit any name, dependent names, which are shown as readonly, will be updated accordingly. If the Tablespace field is left blank the default tablespace of the schema will be used. When you click OK, the SQL script is modified, but it is not actually executed until you select 'Schedule Implementation' on the Recommendations or SQL Statements pages.

Can

Actions

Set Tablespace for All Actions Go

Implementation Status	Action	Object Name	Object Attributes	Indexed Columns	Base Table	Schema	Tablespace	Partiti
■	PARTITION_TABLE	PRODUCTS				SH		("SUPPI
■	CREATE_INDEX	PRODUCTS_IDX$$_00130003	BTREE, LOCAL	SUPPLIER_ID, PROD_LIST_PRICE, PROD_NAME	SH.PRODUCTS	SH		

SQL Affected by Recommendation: 1

Previous 1-10 of 255

Statement ID	Statement	Original Cost	New Cost	Cost Improvement ▽	Cost Improvement (%)	Execu
116	SELECT prod_name FROM products p WHERE prod_list_price=123 AND supplier_id=267	10	2	8	80.00	
119	SELECT prod_name FROM products p WHERE prod_list_price=375 AND supplier_id=282	10	2	8	80.00	
122	SELECT prod_name FROM products p WHERE prod_list_price=461 AND supplier_id=8	10	2	8	80.00	
123	SELECT prod_name FROM products p WHERE prod_list_price=431 AND supplier_id=240	10	2	8	80.00	
126	SELECT prod_name FROM products p WHERE prod_list_price=355 AND supplier_id=5	10	2	8	80.00	
128	SELECT prod_name FROM products p WHERE prod_list_price=139 AND supplier_id=98	10	2	8	80.00	
131	SELECT prod_name FROM products p WHERE prod_list_price=441 AND supplier_id=103	10	2	8	80.00	
133	SELECT prod_name FROM products p WHERE prod_list_price=451 AND supplier_id=12	10	2	8	80.00	
136	SELECT prod_name FROM products p WHERE prod_list_price=594 AND supplier_id=257	10	2	8	80.00	
137	SELECT prod_name FROM products p WHERE prod_list_price=445 AND supplier_id=243	10	2	8	80.00	

Previous 1-10 of 255

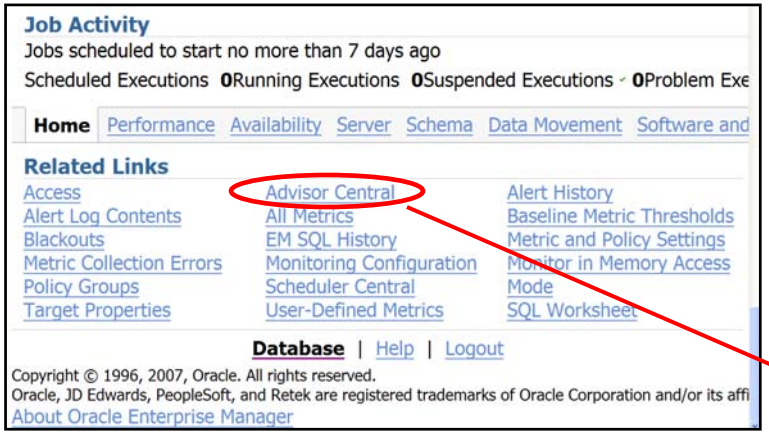
Conclusions

- SAA now covers your data access problems with all possible access solutions
- New for 11g:
  - Partition advice, including hash and new interval on date and number
  - Incremental advice
- Partition recommendations are holistically generated, simultaneously considering all possible access solutions across an entire SQL workload
- SAA is easy to use as ever – partition advice is yours for click of a checkbox!

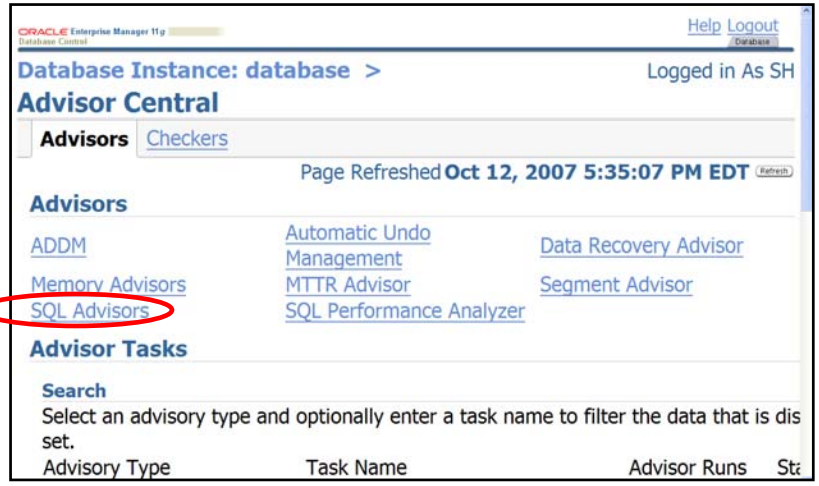

Q&A


ORACLE®

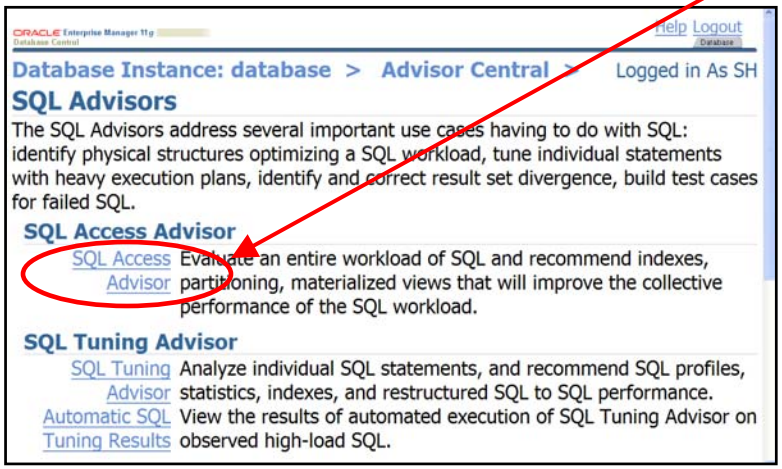
Navigating to SQL Access Advisor



EM Home Page

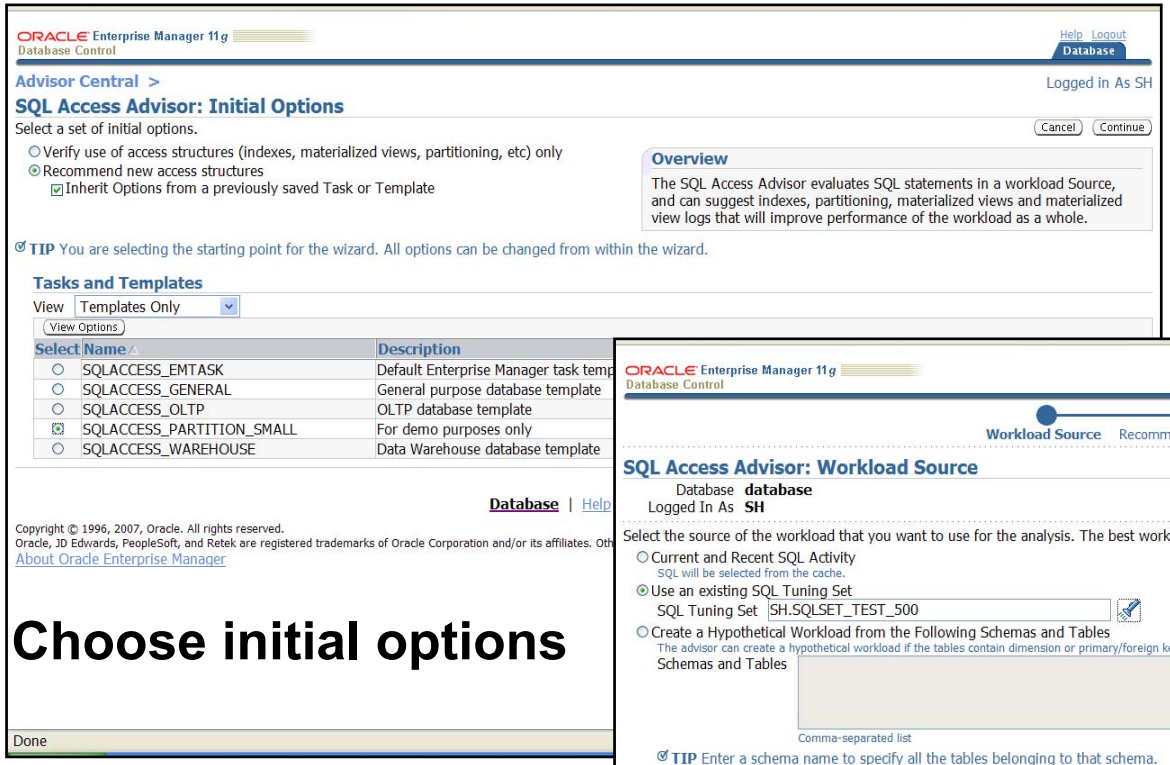


Advisor Central Page

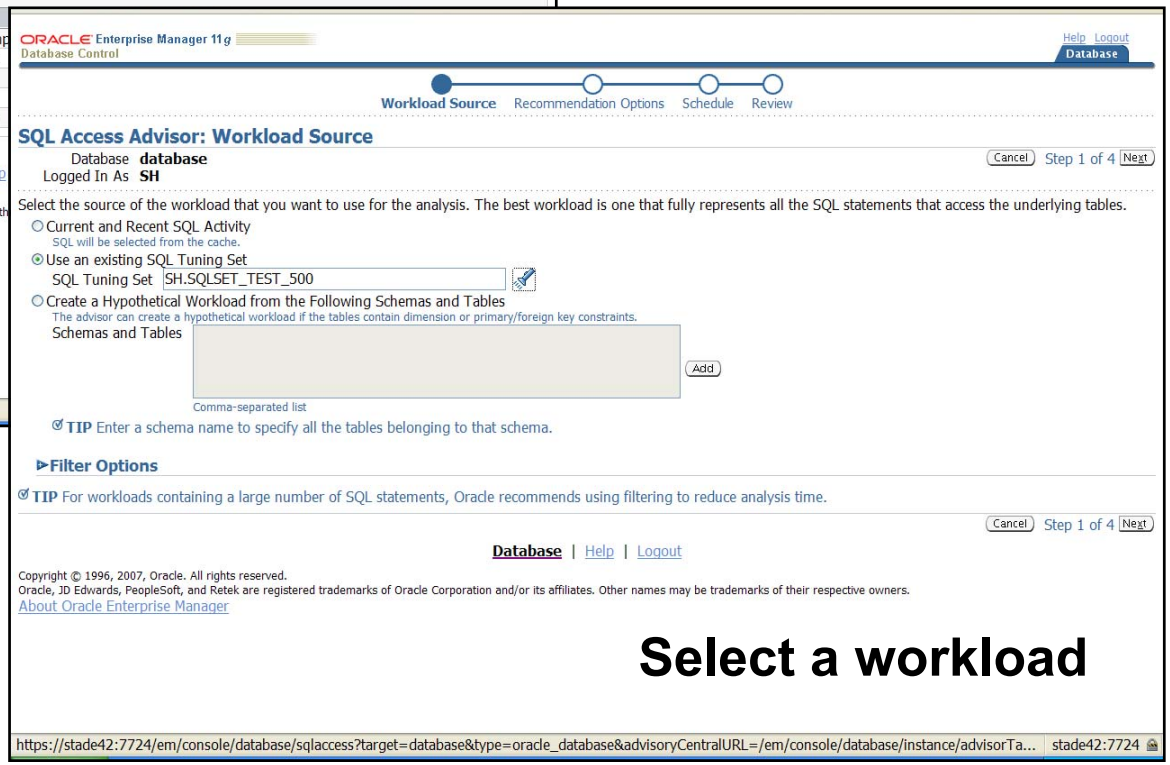


SQL Advisor Page

Using SQL Access Advisor

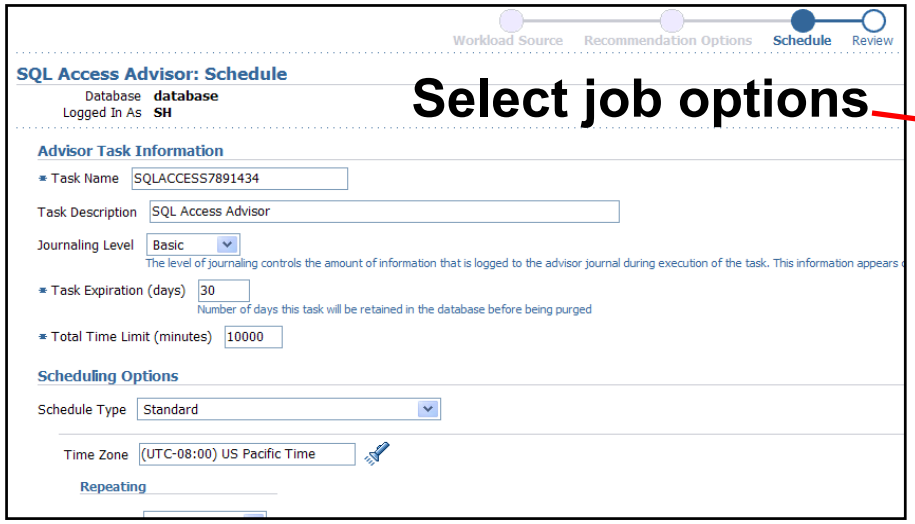


Choose initial options

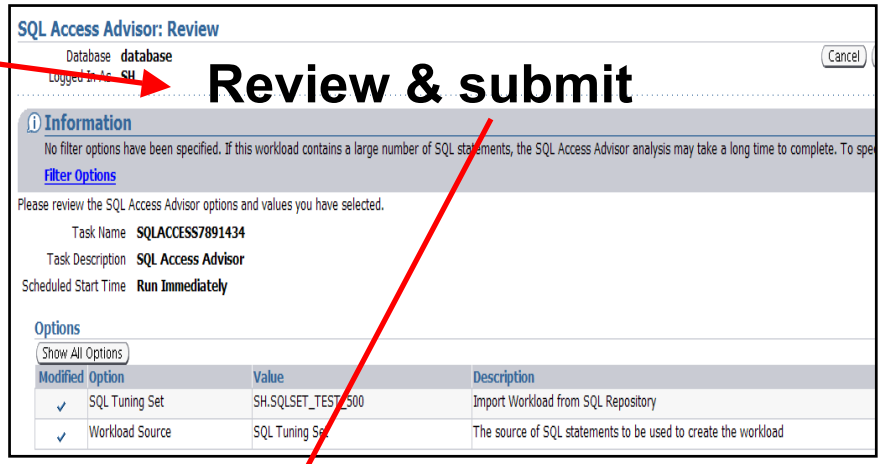


Select a workload

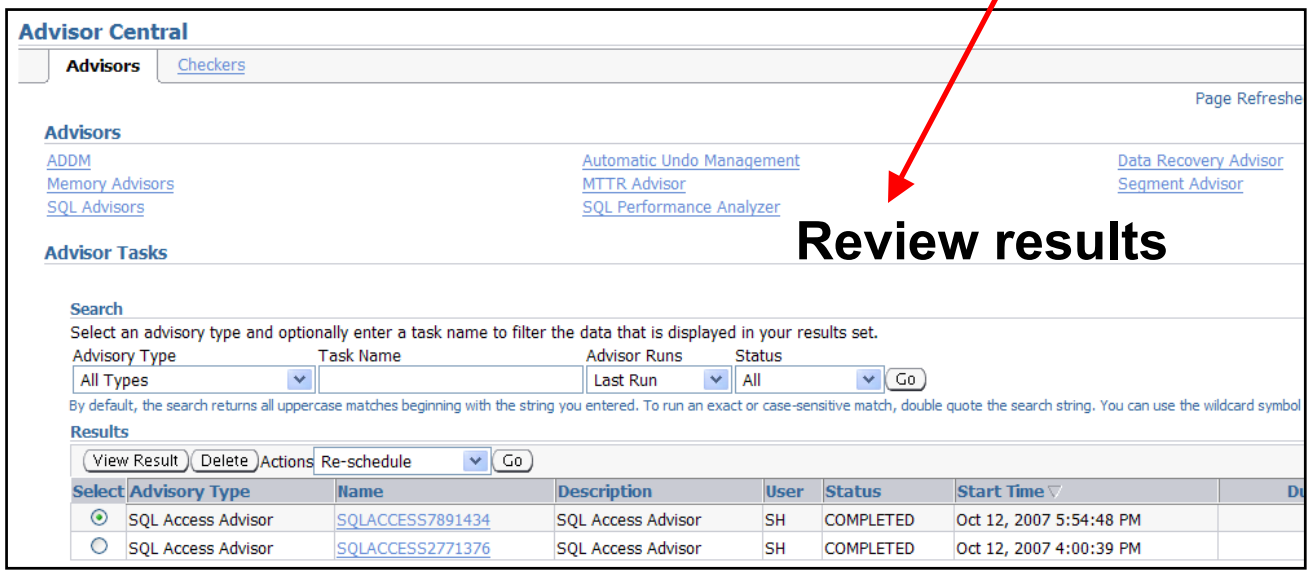
Running advisor job



Select job options



Review & submit



Review results