MORRISONS

How to design, build and maintain an AWR Repository

John Hallas

UKOUG – Tech 2013

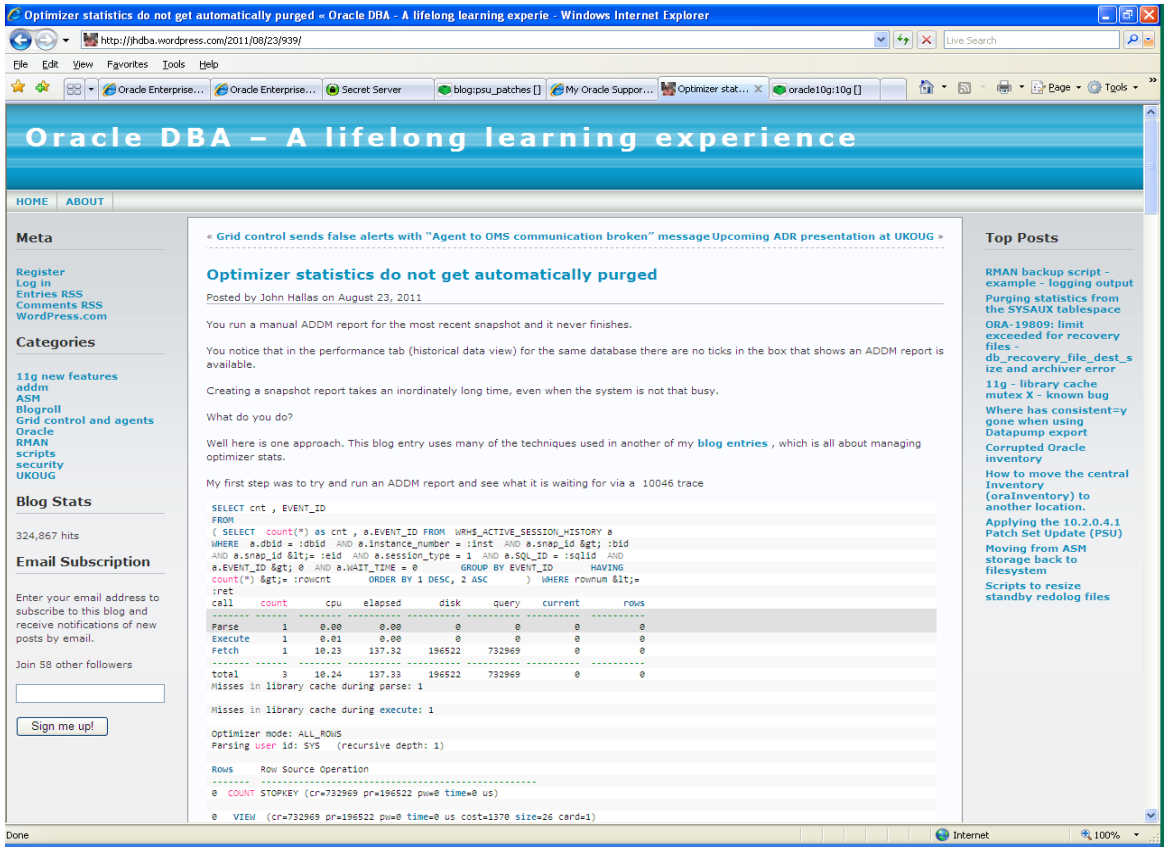
Agenda

- What is AWR data?
- Why do you need a repository
- AWR, business processes, UNIX data
- How to size and build
- How to maintain and feed data in
- Problems we have hit
- Real world examples

MORRISONS

Introduction

- Who am I?
  - John Hallas - Morrisons Supermarkets PLC
  - DBA Team – Core Team Leader
  - Blog: Oracle DBA – A Lifelong Learning Experience
  - www.jhdba.wordpress.com



MORRISONS

Which data and how much space do you need

- Contains all AWR (and ASH, ADDM) snapshots for the designated period – WRH$ and WRM$ tables
- Need to decide how long to keep it
- Retail organisation - 400 days Easter to Easter
- Server requirements -does not need much CPU = write once read few times
- Plenty of disk space needed – could be tier 3
- Compression reduces disk requirements
- How can you sell it – saving of space on production systems – issues with AWR snapshots accumulating – cost neutral

MORRISONS

Sizing

- Look at each current system and get a feel for usage.
- Use @?/rdbms/admin/awrinfo.sql section 2

```
**********
(2) Size estimates for AWR snapshots - Data Warehouse on Exadata retention 45 days (8 node
  RAC on 11.2.0.3)
**********
|
| Estimates based on 60 mins snapshot INTERVAL:
|    AWR size/day                          419.5 MB (17,897 K/snap * 24 snaps/day)
|    AWR size/wk                         2,936.2 MB (size_per_day * 7) per instance
|    AWR size/wk                         5,872.4 MB (size_per_day * 7) per database
|
| Estimates based on 24 snaps in past 24 hours:
|    AWR size/day                          419.5 MB (17,897 K/snap and 24 snaps in past 24
  hours)
|    AWR size/wk                         2,936.2 MB (size_per_day * 7) per instance
|    AWR size/wk                         5,872.4 MB (size_per_day * 7) per database
|
```

MORRISONS

Sizing for source databases

- Look at each current system and get a feel for usage.
- Use @?/rdbms/admin/awrinfo.sql section 3

```
**********************************
Space usage by AWR components (per database) Data Warehouse on Exadata
  retention 45 days

COMPONENT        MB  % AWR  KB_PER_SNAP MB_PER_DAY MB_PER_WEEK TABLE% : INDEX%
--------- --------- ------ ------------ ---------- ----------- ---------------
FIXED      2,480.4    33.8        6,048      141.7       992.2    53% : 47%
SQLPLAN    1,685.0    23.0        4,108       96.3       674.0    73% : 27%
EVENTS       496.1     6.8        1,209       28.3       198.4    47% : 53%
ASH          195.5     2.7          477       11.2        78.2    83% : 17%
SQL           48.4     0.7          118        2.8        19.4    78% : 22%
SPACE         42.7     0.6          104        2.4        17.1    63% : 37%
SQLTEXT        9.4     0.1           23        0.5         3.8    95% : 5%
RAC            9.1     0.1           22        0.5         3.6    60% : 40%
SQLBIND        6.0     0.1           15        0.3         2.4    50% : 50%
```

wrh$sql_plan – multi-table joins - big complex plans

fixed - (segments/tablespace and file information

MORRISONS

How big is the repository

```
SYS@AWRPRD1A SQL>select table_name from dba_tables where tablespace_name = 'SQLPLAN_TS';

TABLE_NAME
------------------------------
WRH$_SQLTEXT
WRH$_SQL_PLAN
```

Tablespace	Size (Mb)	Used (Mb)	Free (Mb)	Frags (#)	Max (Mb)	Free (%)
UNDOTBS1	32768	91	32677	21	3968.00	99.7
SQLPLAN_TS	21440	16663	4777	226	1599.00	22.3
CAPACITY	133120	107115	26005	9	3968.00	19.5
USERS	2770	2637	133	2	132.25	4.8
SYSAUX	435776	420430	15346	246	3968.00	3.5
SYSTEM	3230	3195	35	23	8.00	1.1
TEMP	0	0	0	0	.00	.0
sum	629104	550131	78973			

MORRISONS

AWR repository ~650Gb currently

List of DBIds can be picked up by running awrrpti or from wrm$wrcontrol table

```
Set lines 180
column instt_num  heading "Inst Num"  format 99999;
column instt_name heading "Instance"  format a12;
column dbb_name   heading "DB Name"   format a12;
column dbbid      heading "DB Id"     format a12 just c;
column host       heading "Host"      format a42;

prompt Instances in this Workload Repository schema
prompt ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
select distinct
       (case when cd.dbid = wr.dbid and
                  cd.name = wr.db_name and
                  ci.instance_number = wr.instance_number and
                  ci.instance_name   = wr.instance_name   and
                  ci.host_name       = wr.host_name
             then '* '
             else '  '
        end) || wr.dbid   dbbid
     , wr.instance_number instt_num
     , wr.db_name         dbb_name
     , wr.instance_name   instt_name
     , wr.host_name       host
  from dba_hist_database_instance wr, v$database cd, v$instance ci;
```

64 dbids currently (183 in total) RAC, standby switching

MORRISONS

Steps to create the repository

- Normal database – no fancy options bar partitioning possibly (- must be as current as the latest version in use as source – use 11.2.0.3 if possible because that solves a lot of issues around duplicate keys (MERGE INTO rather than INSERT APPEND)
- Disable snapshots
- Create an AWRREP user
- Preferred to make SYSAUX a bigfile t/s - we move 2 tables in WRH$_SQLTEXT and WRH$_SQL_PLAN – easier to reorganise the table objects then
- The issue of the direct load tables (which is still the majority 105 tables vs 20 with merge) is something that will need to be addressed at Morrisons - now that the housekeeping procedures are in place, old snapshots are deleted on a weekly basis, but new data is loaded above the HWM for each extract. This will mean that the non-partitioned tables will need to be reorganised on a regular basis. This will become more of an issue as the number of sources that have data over 400 days increases

MORRISONS

AWRREP user on the repository

- repository owner privileges

```
CREATE USER "AWRREP" PROFILE "M_APPS_PROFILE" IDENTIFIED BY "*******" DEFAULT TABLESPACE "USERS" TEMPORARY
TABLESPACE "TEMP" ACCOUNT UNLOCK
GRANT ALTER ANY INDEX TO "AWRREP"
GRANT ALTER ANY TABLE TO "AWRREP"
GRANT ALTER USER TO "AWRREP"
GRANT ANALYZE ANY TO "AWRREP"
GRANT CREATE ANY INDEX TO "AWRREP"
GRANT CREATE ANY TABLE TO "AWRREP"
GRANT CREATE PROCEDURE TO "AWRREP"
GRANT CREATE USER TO "AWRREP"
GRANT DEBUG ANY PROCEDURE TO "AWRREP"
GRANT DEBUG CONNECT SESSION TO "AWRREP"
GRANT DELETE ANY TABLE TO "AWRREP"
GRANT DROP USER TO "AWRREP"
GRANT SELECT ANY TABLE TO "AWRREP"
GRANT UNLIMITED TABLESPACE TO "AWRREP"
GRANT READ ON DIRECTORY "SYS"."DATA_PUMP_DIR" TO "AWRREP"
GRANT WRITE ON DIRECTORY "SYS"."DATA_PUMP_DIR" TO "AWRREP"
GRANT SELECT ON "SYS"."DBA_INDEXES" TO "AWRREP"
GRANT SELECT ON "SYS"."DBA_TABLES" TO "AWRREP"
GRANT SELECT ON "SYS"."DBA_USERS" TO "AWRREP"
GRANT EXECUTE ON "SYS"."DBMS_DATAPUMP" TO "AWRREP"
GRANT EXECUTE ON "SYS"."DBMS_INDEX_UTL" TO "AWRREP"
GRANT EXECUTE ON "SYS"."DBMS_LOB" TO "AWRREP"
GRANT EXECUTE ON "SYS"."DBMS_STATS" TO "AWRREP"
GRANT EXECUTE ON "SYS"."DBMS_SWRF_INTERNAL" TO "AWRREP"
GRANT EXECUTE ON "SYS"."DBMS_WORKLOAD_REPOSITORY" TO "AWRREP"
GRANT EXECUTE ON "SYS"."UTL_FILE" TO "AWRREP"
GRANT SELECT ON "SYS"."V_$DATABASE" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_ACTIVE_SESSION_HISTORY" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_DB_CACHE_ADVICE" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_DLM_MISC" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_EVENT_HISTOGRAM" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_FILESTATXS" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_INST_CACHE_TRANSFER" TO "AWRREP"
```

MORRISONS

AWRREP user on the repository

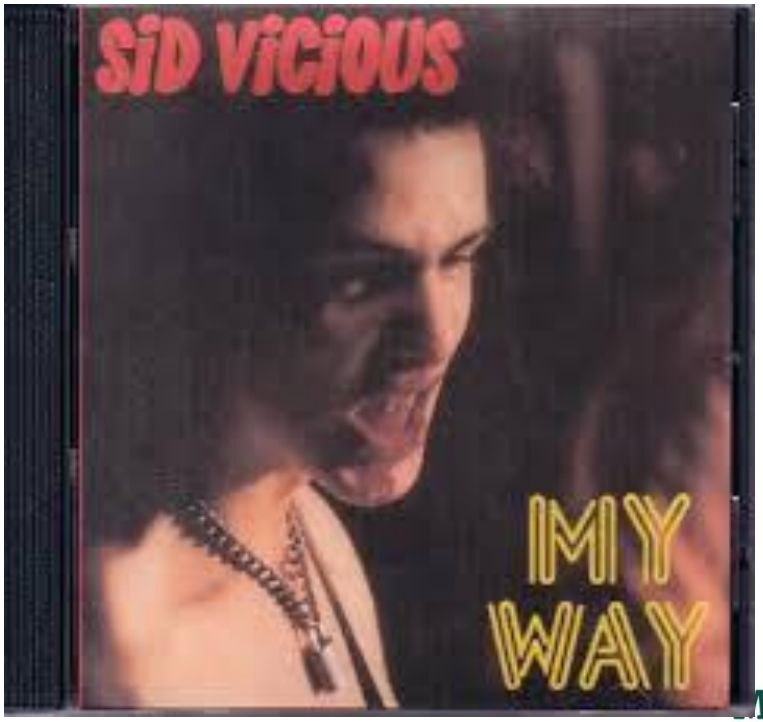
- repository owner privileges contd

```
GRANT ALTER ON "SYS"."WRH$_INTERCONNECT_PINGS" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_LATCH" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_LATCH_CHILDREN" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_LATCH_MISSES_SUMMARY" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_LATCH_PARENT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_OSSTAT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_PARAMETER" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_ROWCACHE_SUMMARY" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SEG_STAT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SERVICE_STAT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SERVICE_WAIT_CLASS" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SGASTAT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SQLSTAT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SQLTEXT" TO "AWRREP" WITH GRANT OPTION
GRANT DEBUG ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT DELETE ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT FLASHBACK ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT INDEX ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT INSERT ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT ON COMMIT REFRESH ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT QUERY REWRITE ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT REFERENCES ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT SELECT ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT UPDATE ON "SYS"."WRH$_SQLTEXT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SQL_PLAN" TO "AWRREP" WITH GRANT OPTION
GRANT DELETE ON "SYS"."WRH$_SQL_PLAN" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SYSSTAT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SYSTEM_EVENT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_SYS_TIME_MODEL" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_TABLESPACE_STAT" TO "AWRREP"
GRANT ALTER ON "SYS"."WRH$_WAITSTAT" TO "AWRREP"
GRANT SELECT ON "SYS"."WRM$_SNAPSHOT" TO "AWRREP"
GRANT "CONNECT" TO "AWRREP"
GRANT "SELECT_CATALOG_ROLE" TO "AWRREP"
```

MORRISONS

My Way

But as it is Oracle I thought this version might be better ……..
At least I mention a SID



MORRISONS

Extract and load overview

The extract is a scheduled job (daily or weekly, depending on the database) which runs under the user AWRREP on the source database and calls awrexpr_process.run_awr_extract.

The extract uses an Oracle API (dbms_swrf_internal) to extract the data. This package uses datapump to export AWR data. The extract then uses dbms_file_transfer to transfer the file generated to the target database: AWRPRD1A.

On a daily basis, a scheduled job runs on the Repository database under the user AWRREP, to load the data into the Repository: awrimpr_process.run_load

A KEY PRINCIPAL

Find the latest extracted snapshot (from the local control table)

Get the latest dba_snapshot_id and you have the begin and end dates

If there is no last extracted snapshot then limit the number retrieved - reduce overhead!! Use the WRM$_SNAPSHOT view and the AWRREP tables

MORRISONS

How to create a feed into the repository 1

- Need a user to send the data from source to repository

```
CREATE USER AWRREP ;
ALTER USER AWRREP PROFILE M_APPS_PROFILE;

GRANT CREATE SESSION TO AWRREP;
GRANT UNLIMITED TABLESPACE TO AWRREP;
GRANT SELECT ON SYS.DBA_HIST_SNAPSHOT TO AWRREP;
GRANT EXECUTE ON SYS.DBMS_FILE_TRANSFER TO AWRREP;
GRANT EXECUTE ON SYS.DBMS_SWRF_INTERNAL TO AWRREP;
GRANT SELECT ON SYS.USER$ TO AWRREP;
GRANT EXECUTE ON SYS.UTL_FILE TO AWRREP;
GRANT SELECT ON SYS.V_$DATABASE TO AWRREP;
GRANT SELECT_CATALOG_ROLE TO AWRREP;
GRANT CREATE PROCEDURE TO AWRREP;
GRANT CREATE DATABASE LINK TO AWRREP; -- drop afterwards
```

- Add tnsnames entry to the repository on the source server

MORRISONS

How to create a feed into the repository 2a

- Control the extract

```
 CREATE TABLE AWRREP.AWREXTR_EXTRACT_PARMS
   (    KEY     VARCHAR2(32),
        VALUE   VARCHAR2(256)
   );
CREATE TABLE AWRREP.AWREXTR_EXTRACT_RUNS
   (    UPLOAD_ID       NUMBER,
        EXTRACT_DATE    DATE,
        BEGIN_SNAP      NUMBER,
        END_SNAP        NUMBER,
        FILE_NAME       VARCHAR2(100),
        LOAD_STATUS     VARCHAR2(1)
   )
  TABLESPACE USERS ;
```

MORRISONS

How to create a feed into the repository 2b

- Keep a log of what happened

```
CREATE TABLE AWRREP.RUN_LOG
  (   DBID          VARCHAR2(32),
      TSTAMP        TIMESTAMP      DEFAULT SYSTIMESTAMP,
      LOG_LEVEL     VARCHAR2(10),
      LOG_TEXT      VARCHAR2(128)
  )   TABLESPACE USERS;

CREATE INDEX AWRREP.RUN_LOG_IX1
      ON AWRREP.RUN_LOG(DBID, TSTAMP)
      TABLESPACE USERS;
```

MORRISONS

Monitoring the feed

```
SQL> col file_name form a40
SQL> set lines 120
SQL> select * from awrrep.AWREXTR_EXTRACT_RUNS order by 2;

 UPLOAD_ID EXTRACT_D BEGIN_SNAP   END_SNAP FILE_NAME                                L
---------- --------- ---------- ---------- ---------------------------------------- -
      1830 26-MAY-13       3608       3775 AWREXTR_2470669947_1830                  R
      1868 02-JUN-13       3776       3943 AWREXTR_2470669947_1868                  R
      1904 09-JUN-13       3944       4111 AWREXTR_2470669947_1904                  R
      1936 16-JUN-13       4112       4279 AWREXTR_2470669947_1936                  R
      1965 23-JUN-13       4280       4447 AWREXTR_2470669947_1965                  R
```

Status can be

- I Initialisation - SOURCE/TARGET
- T transfer is in process SOURCE/TARGET
- R File has been transferred and ready to be loaded SOURCE/TARGET
- P Processing (load data into target tables) TARGET
- C Complete - data has been loaded into target TARGET
- F Failure - SOURCE/TARGET

MORRISONS

How to create a feed into the repository 2c

actual code for the extract and transfer package

awrexpr_process.perform_awr_extract;

The heart of the routine. Makes a call to

```
sys.dbms_swrf_internal.set_awr_dbid(-2);

sys.dbms_swrf_internal.awr_extract(dmpfile  => dmp_file,
                                   dmpdir   => dmp_dir,
                                   bid      => begining_snap,
                                   eid      => ending_snap,
                                   dbid     => db_id);

sys.dbms_swrf_internal.clear_awr_dbid;
```

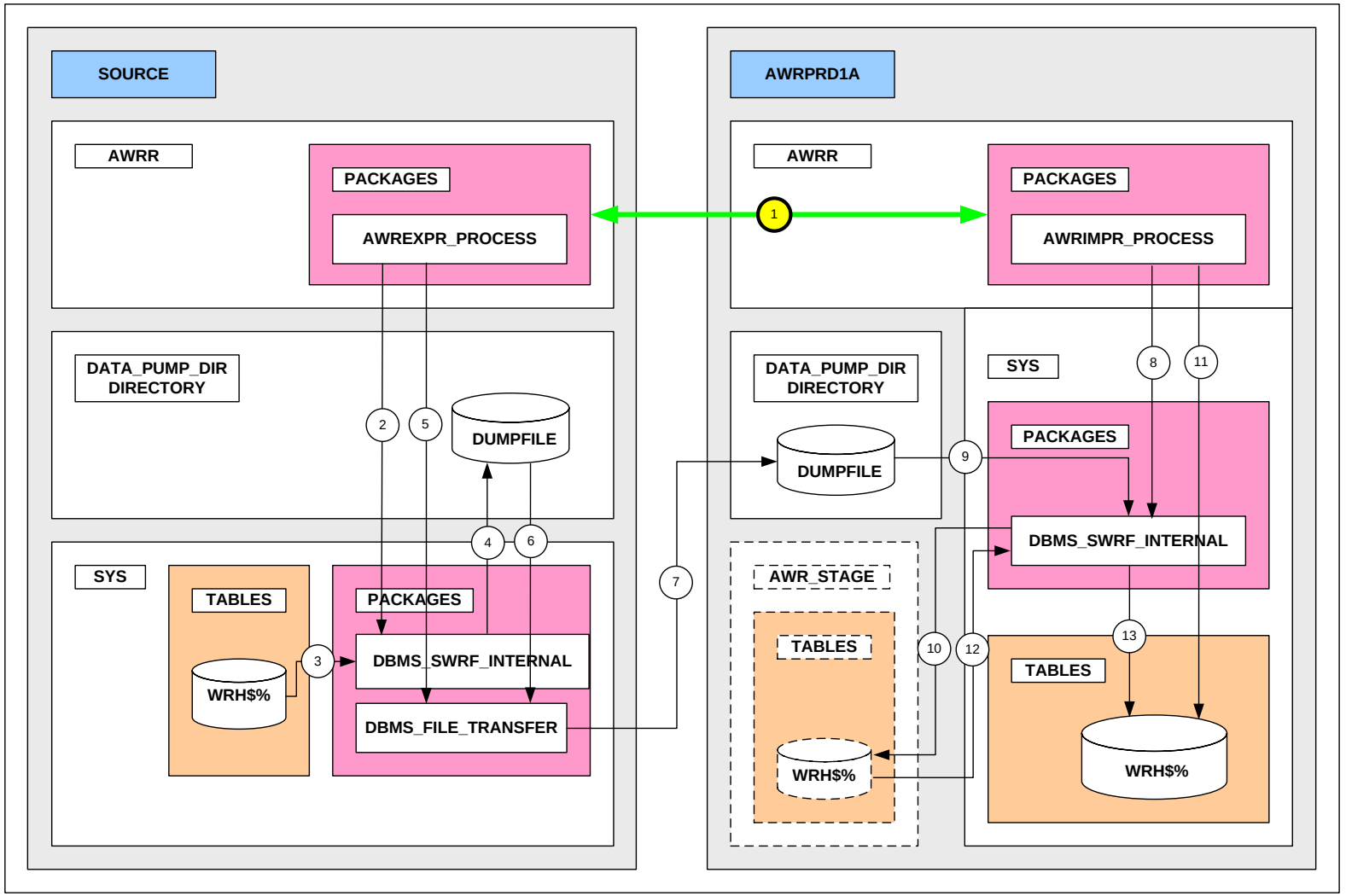
MORRISONS

Create a scheduled job on source

- Create a scheduled job on the source database to run weekly.
- Agree a good time
- ```
  BEGIN
  sys.dbms_scheduler.create_job(
  job_name => '"AWRREP"."AWR_EXPORT_JOB"',
  job_type => 'PLSQL_BLOCK',
  job_action => 'begin
  awrexpr_process.run_awr_extract;
  end;',
  repeat_interval => 'FREQ=WEEKLY;BYDAY=SUN;BYHOUR=1;BYMINUTE=0',
  start_date => to_timestamp_tz('2013-05-19 Europe/London', 'YYYY-MM-DD TZR'),
  job_class => '"DEFAULT_JOB_CLASS"',
  comments => 'AWR Export Job',
  auto_drop => FALSE,
  enabled => FALSE);
  sys.dbms_scheduler.set_attribute( name => '"AWRREP"."AWR_EXTRACT"', attribute => 'rais
  e_events', value => dbms_scheduler.job_failed);
  sys.dbms_scheduler.set_attribute( name => '"AWRREP"."AWR_EXTRACT"', attribute => 'logg
  ing_level', value => DBMS_SCHEDULER.LOGGING_OFF);
  sys.dbms_scheduler.set_attribute( name => '"AWRREP"."AWR_EXTRACT"', attribute => 'job_
  weight', value => 1);
  sys.dbms_scheduler.enable( '"AWRREP"."AWR_EXTRACT"' );
  END;
  ```

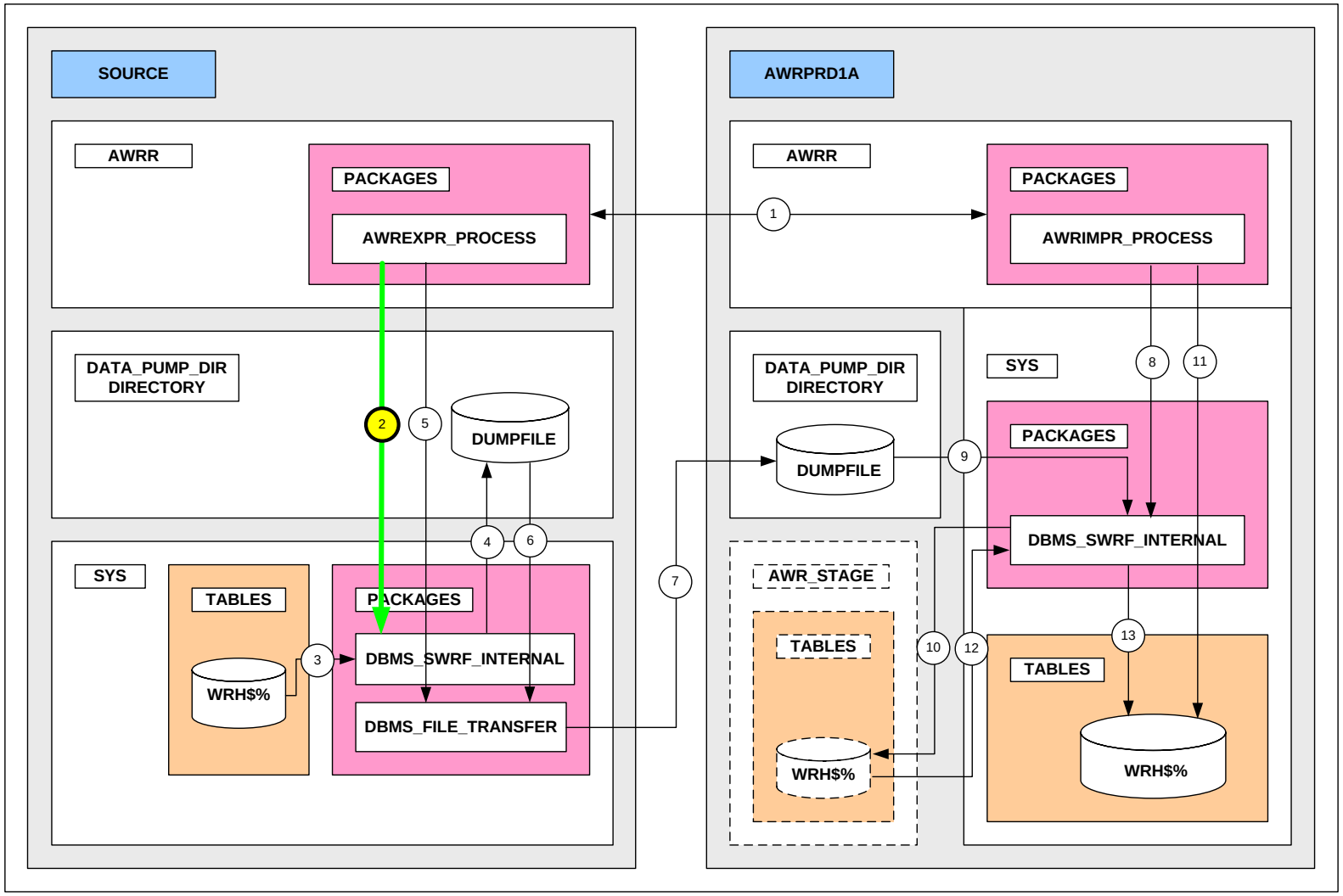
MORRISONS
  ```

Repository Feed - End-to-End Process



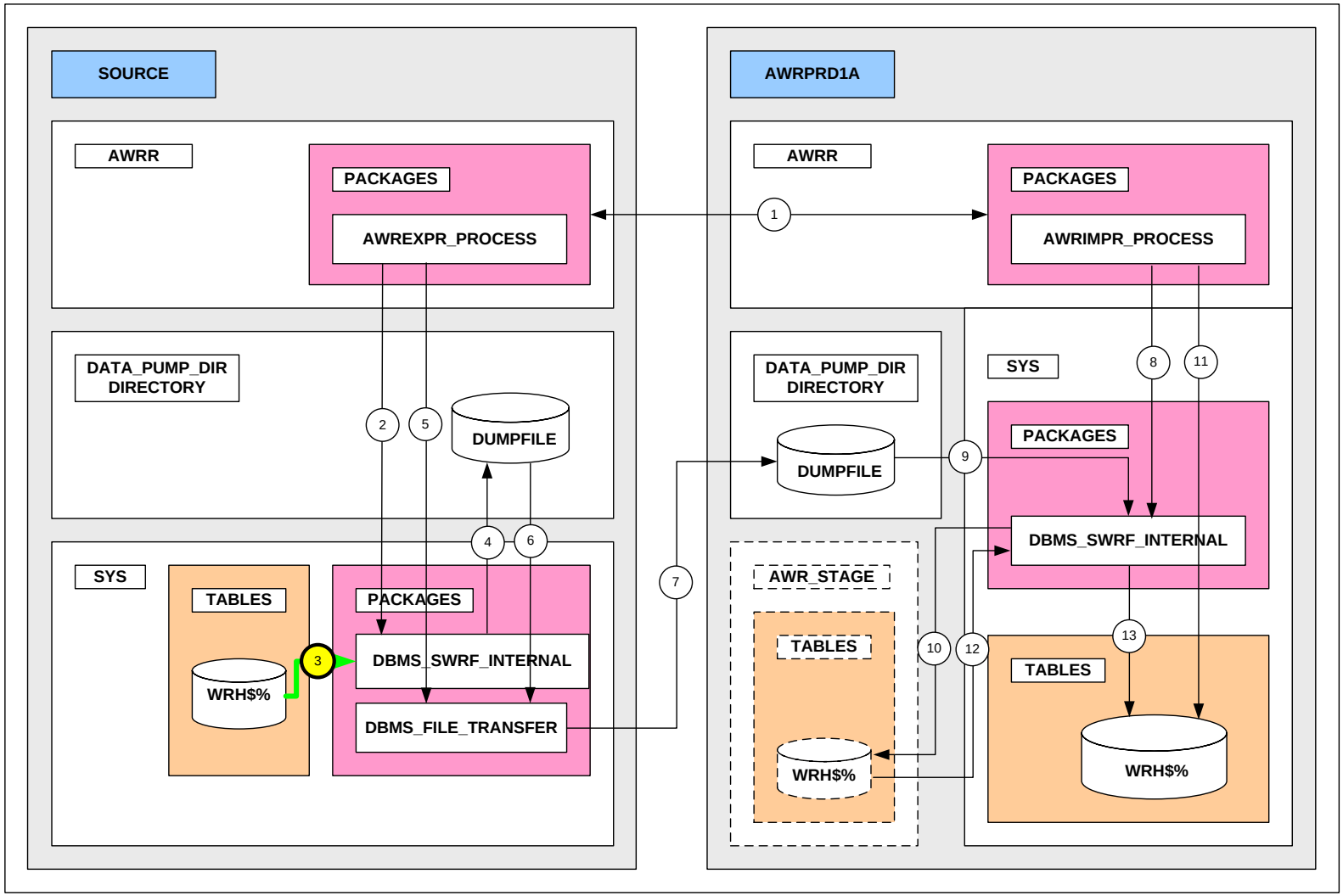
MORRISONS

Repository Feed - End-to-End Process



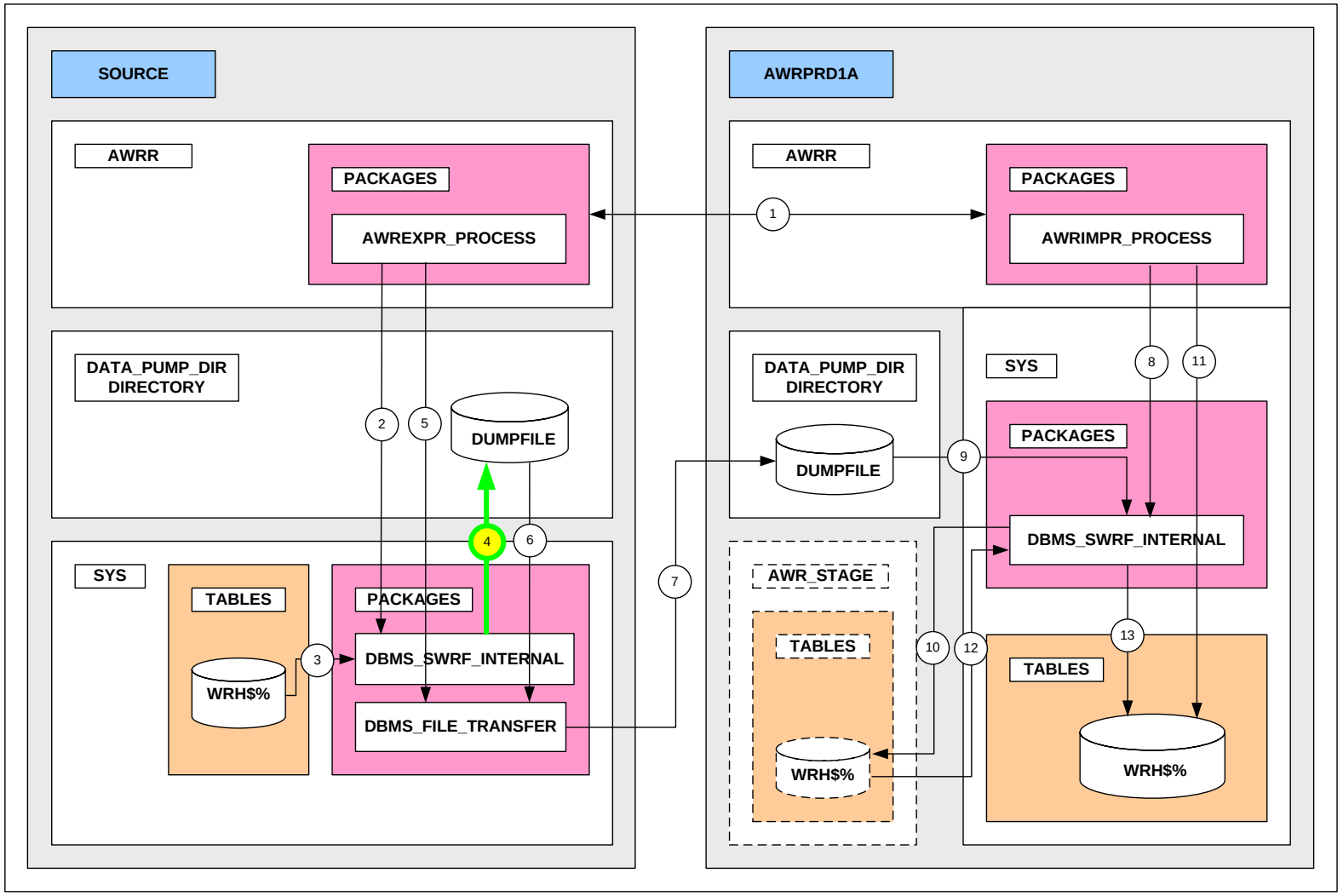
MORRISONS

Repository Feed - End-to-End Process



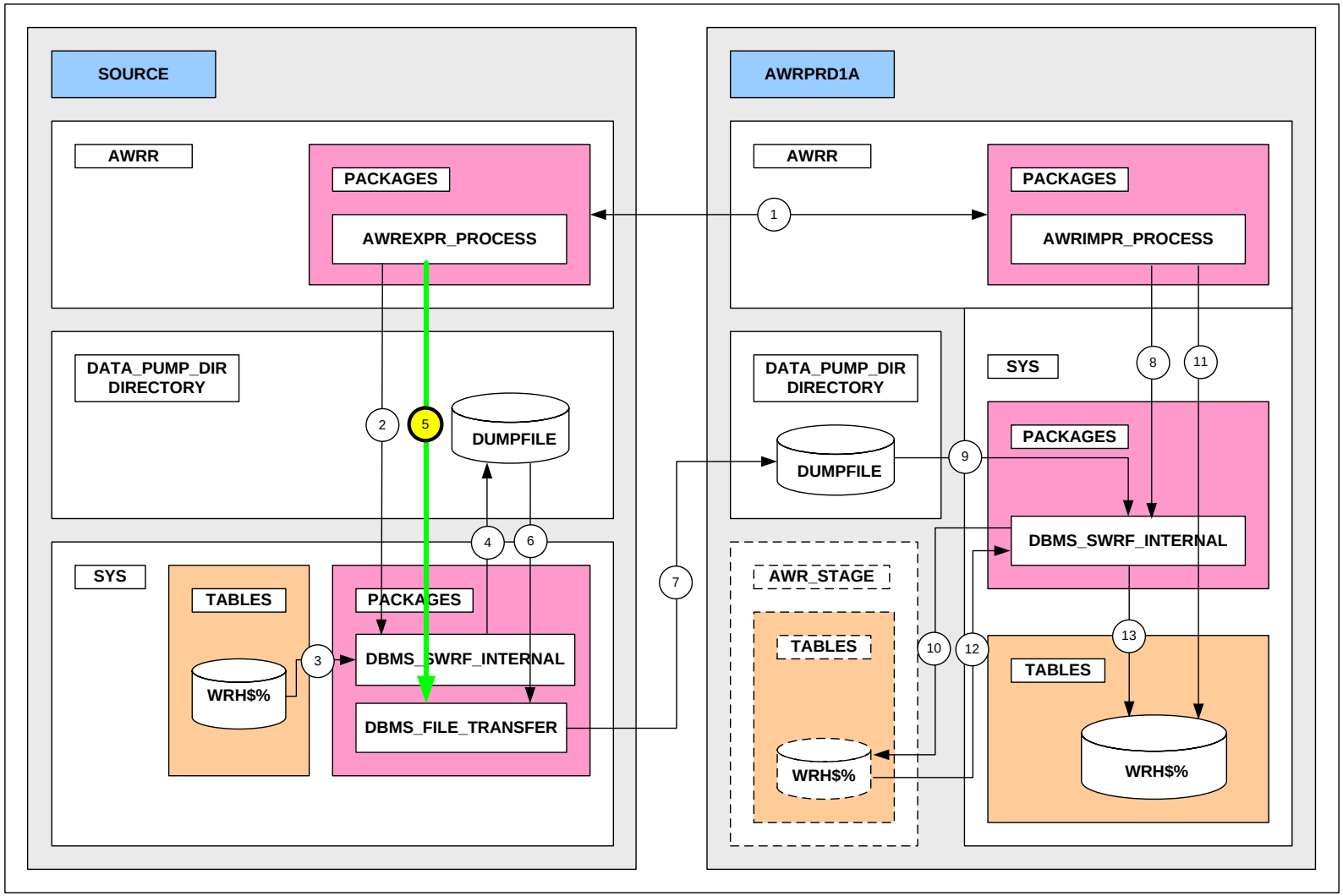
MORRISONS

Repository Feed - End-to-End Process



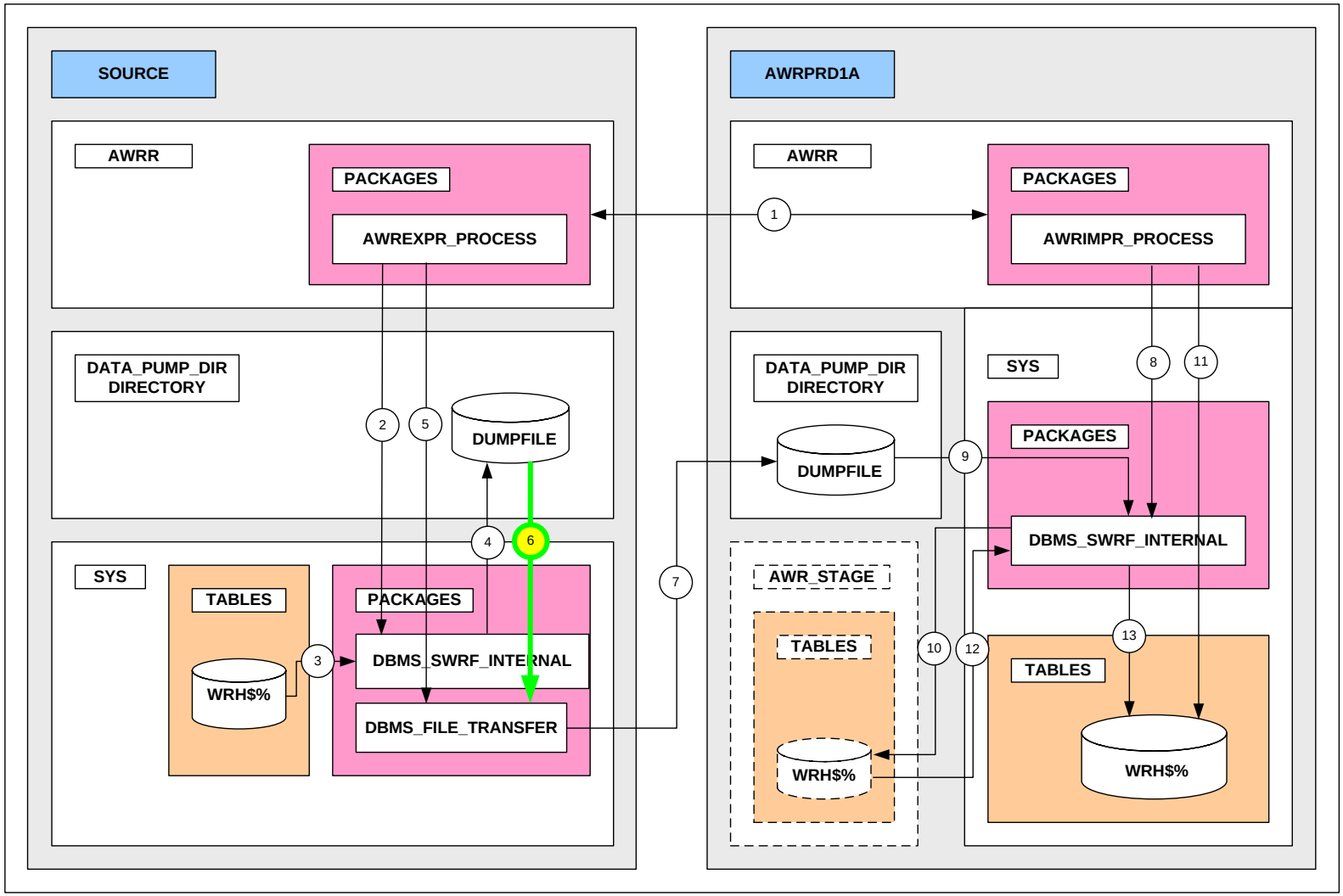
MORRISONS

Repository Feed - End-to-End Process



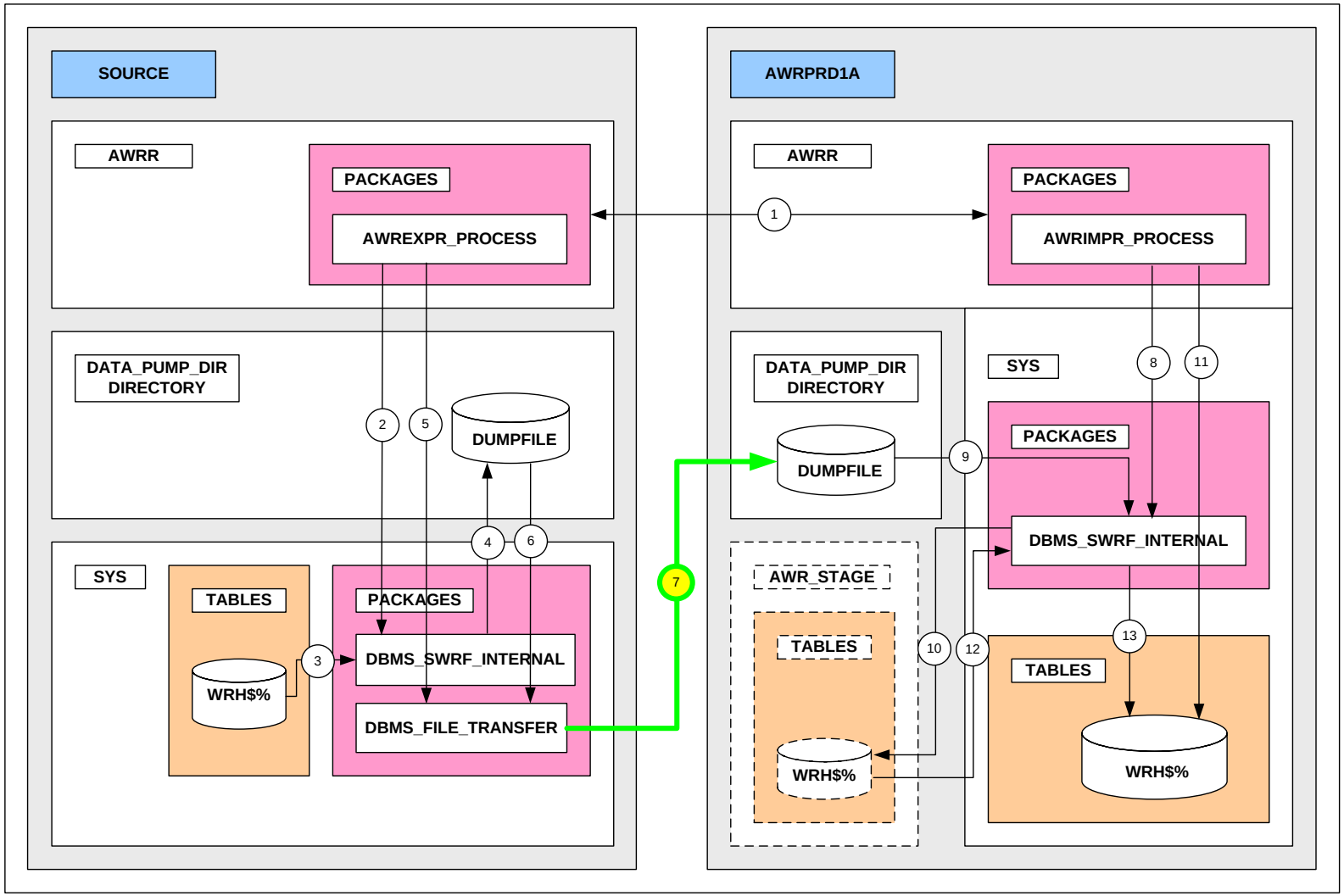
MORRISONS

Repository Feed - End-to-End Process



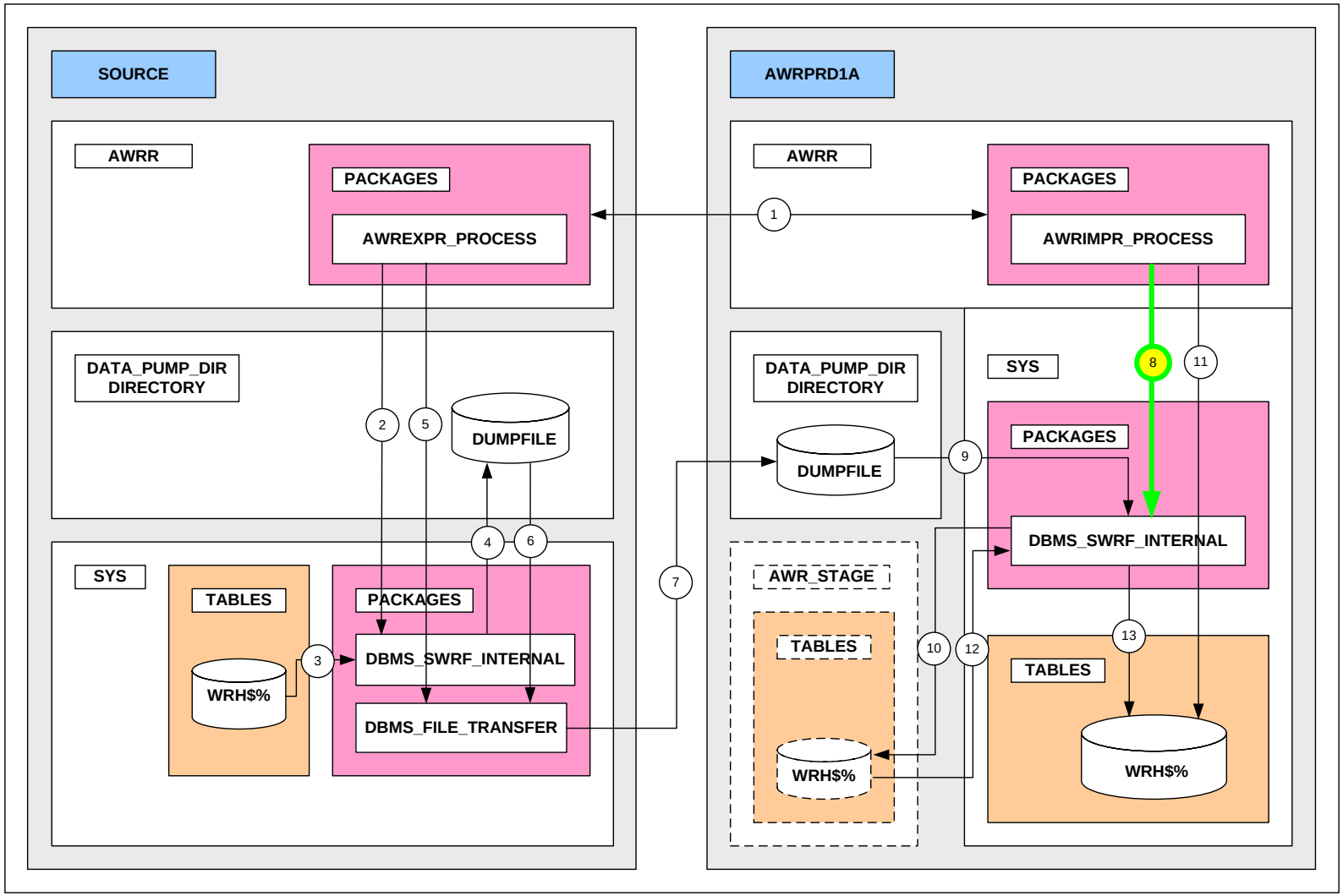
MORRISONS

Repository Feed - End-to-End Process



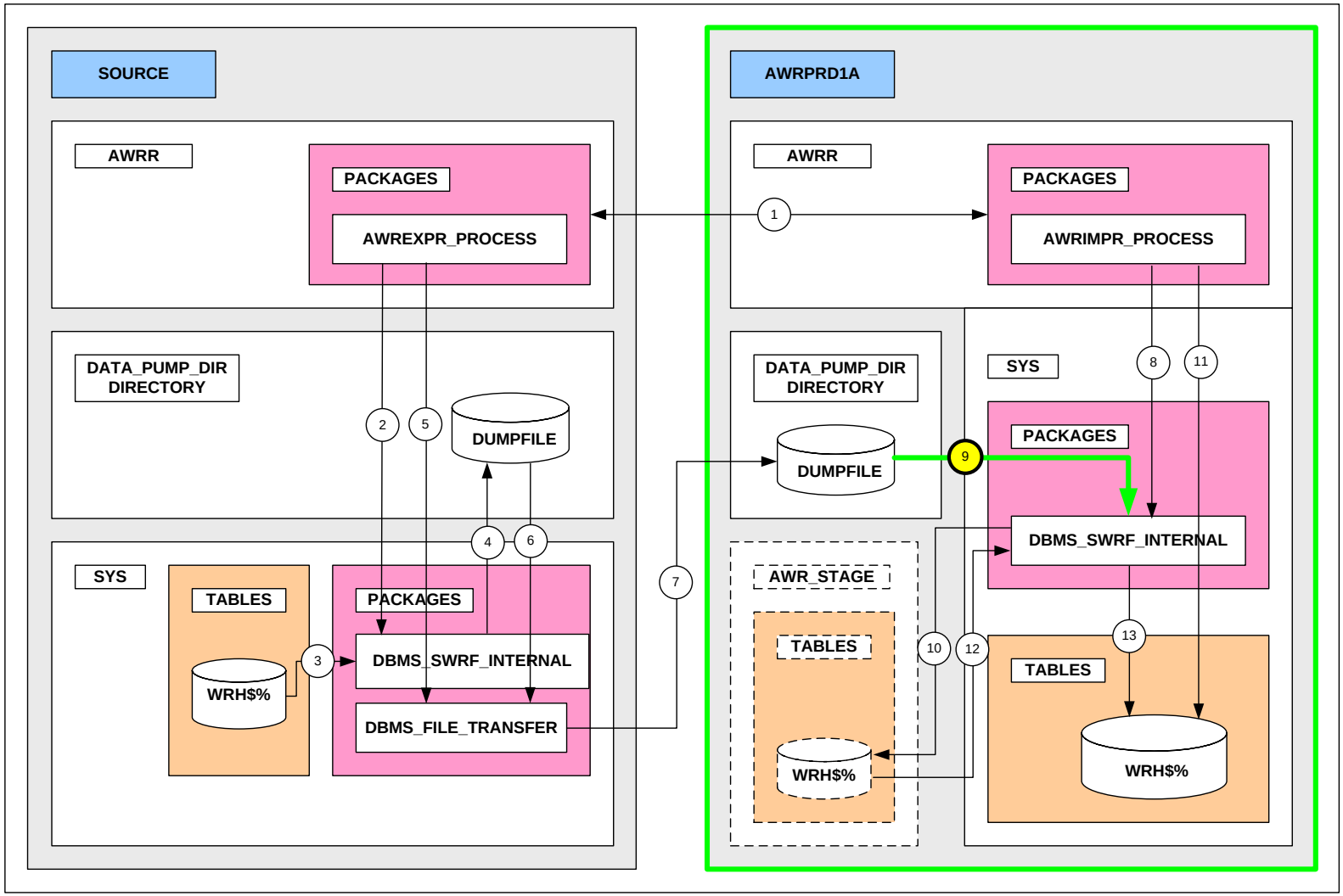
MORRISONS

Repository Feed - End-to-End Process



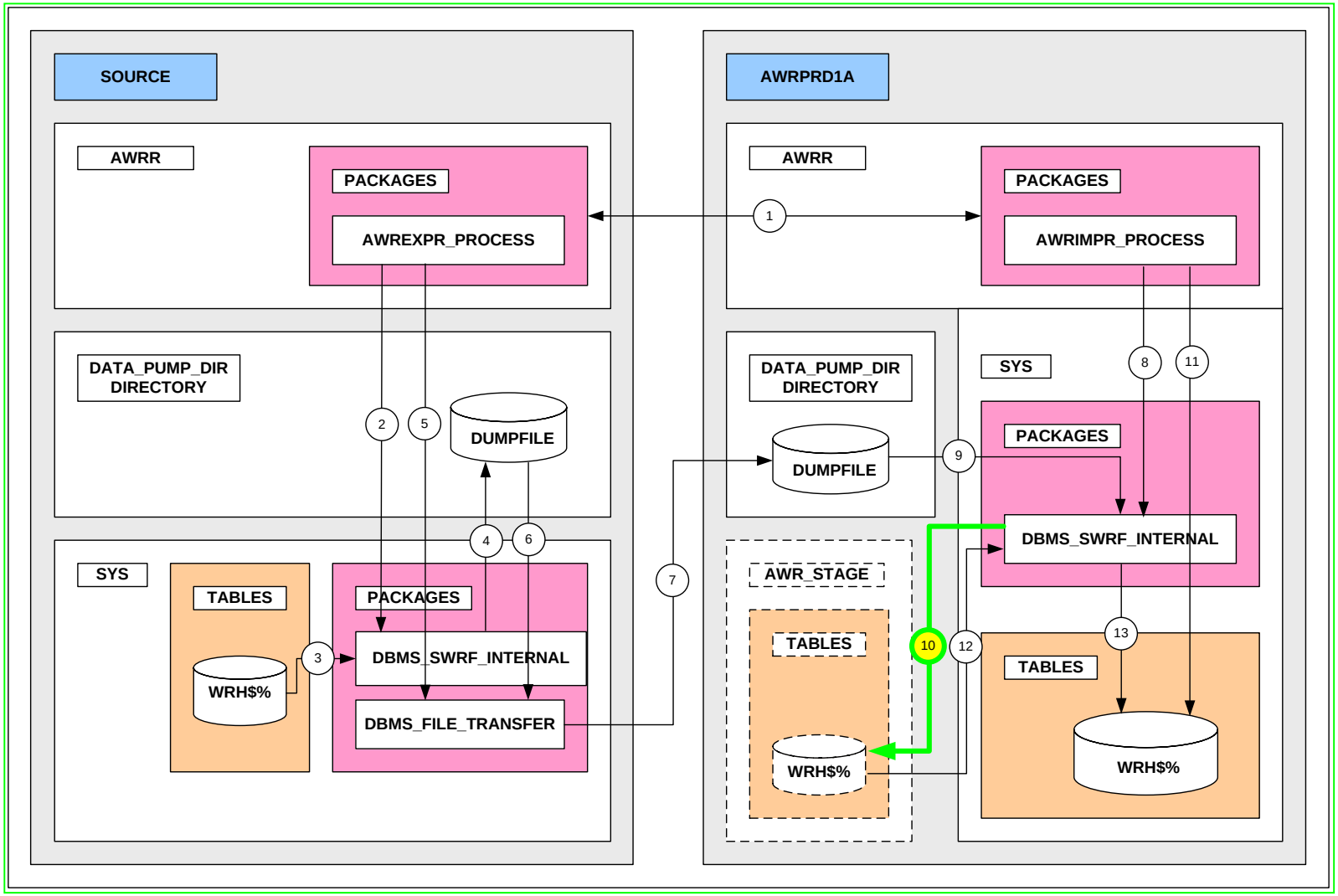
MORRISONS

Repository Feed - End-to-End Process



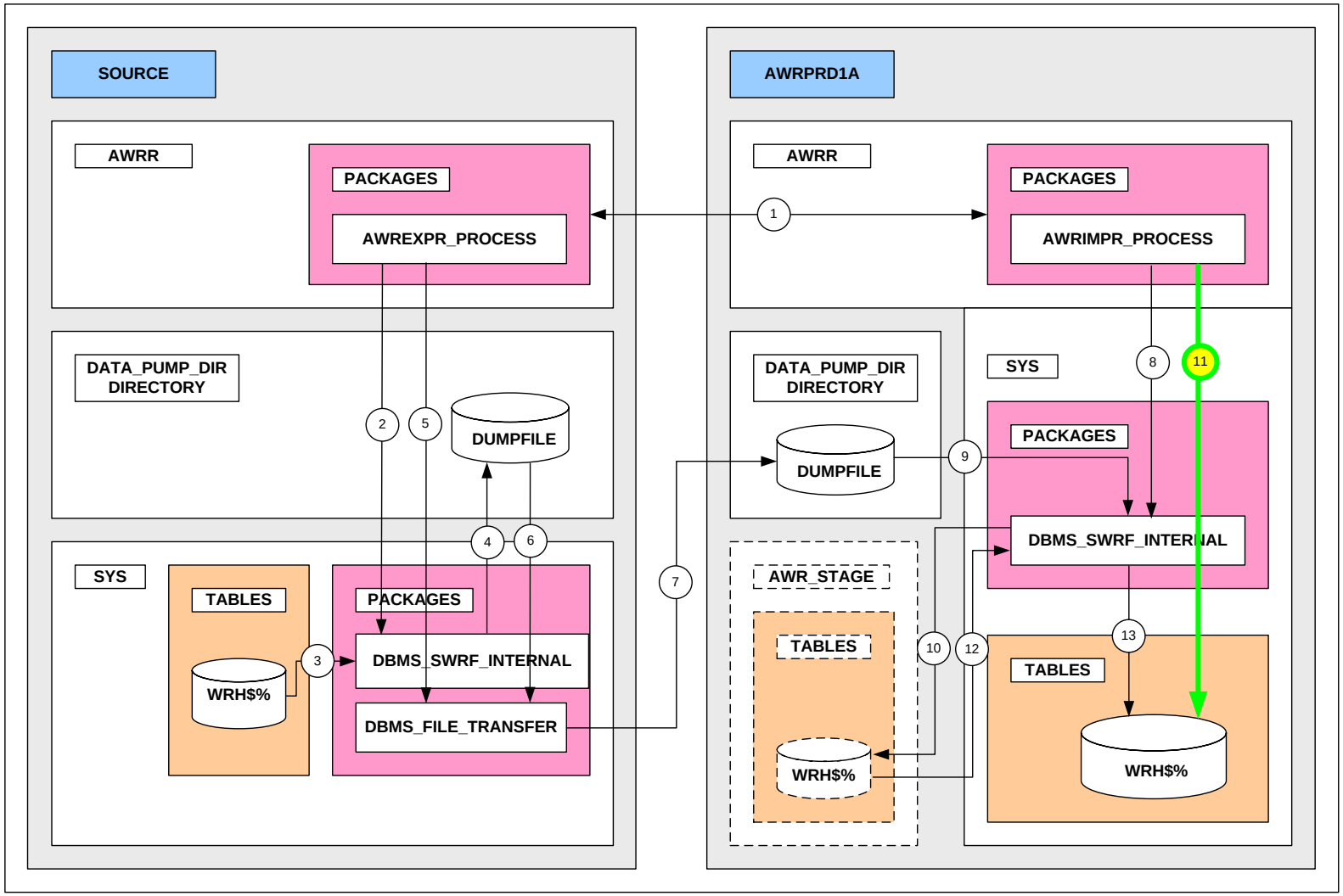
MORRISONS

Repository Feed - End-to-End Process



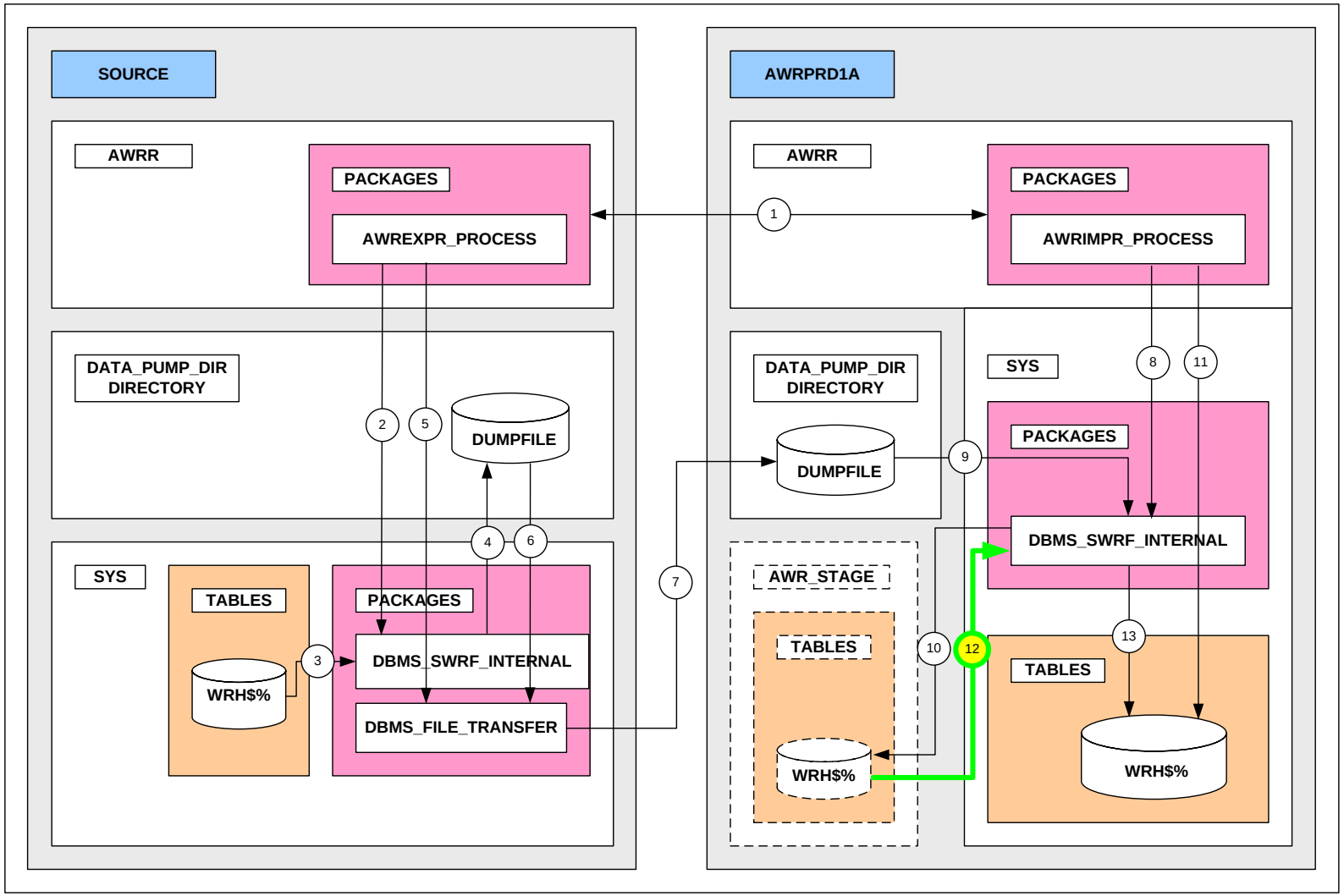
MORRISONS

Repository Feed - End-to-End Process



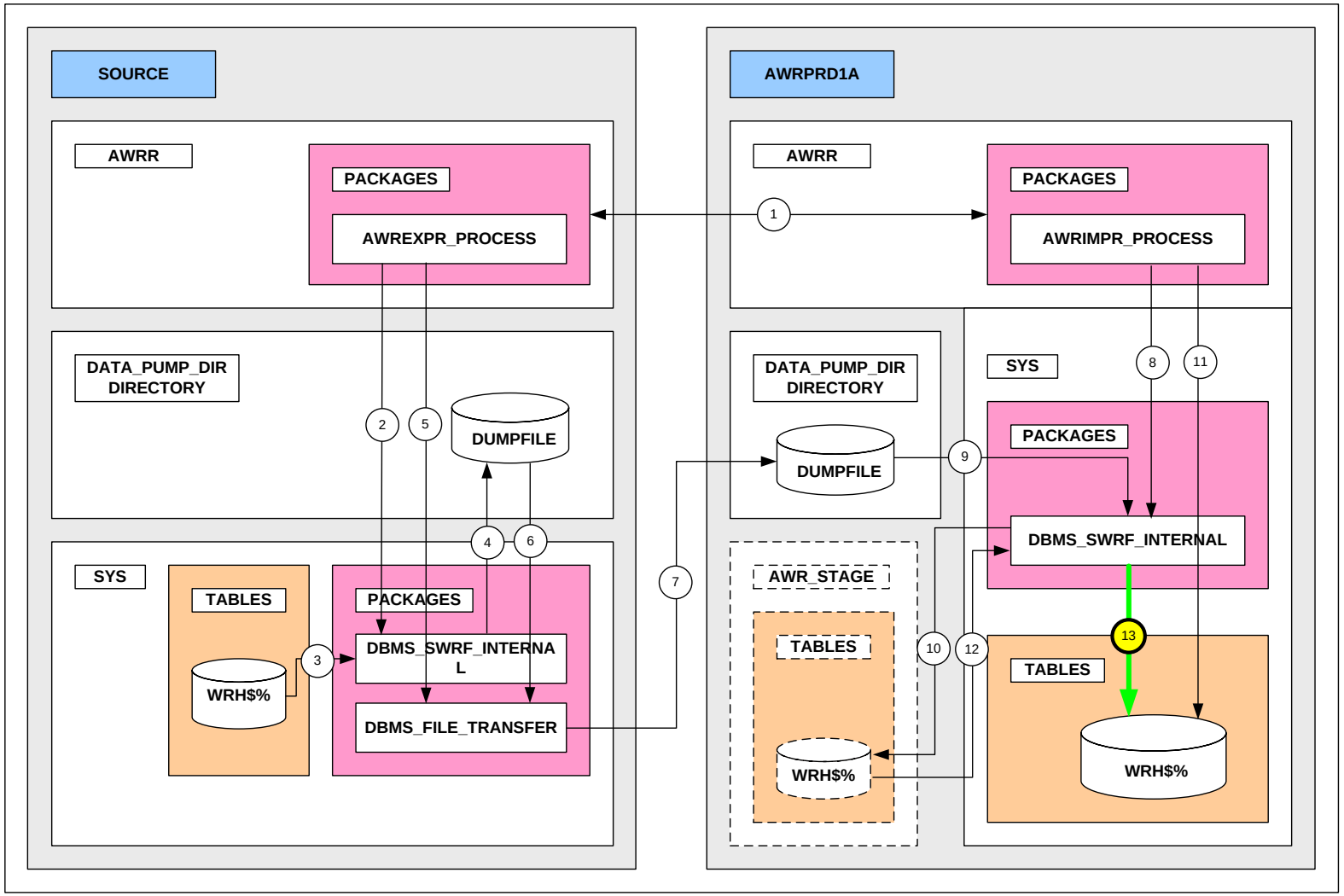
MORRISONS

Repository Feed - End-to-End Process



MORRISONS

Repository Feed - End-to-End Process



MORRISONS

Steps we have completed

- Performance Testing results - create a schema, capture all PT run data (AWR etc) and store
- Capture Capacity (business process) data from all production systems
- Stores and retains daily extract of transactional, batch, and OLTP data for purpose of capacity and reporting
- Provide the ability to chart, compare and present capacity data
- Provide the ability to measure prod actual against performance test
- Provide comparable measures between releases
- Provide tooling to present and analyse data
- Capturing UNIX resource usage, CPU, Memory, disk etc….
- *We might need to investigate partitioning of the AWR tables*

MORRISONS

Current – examples - AWR

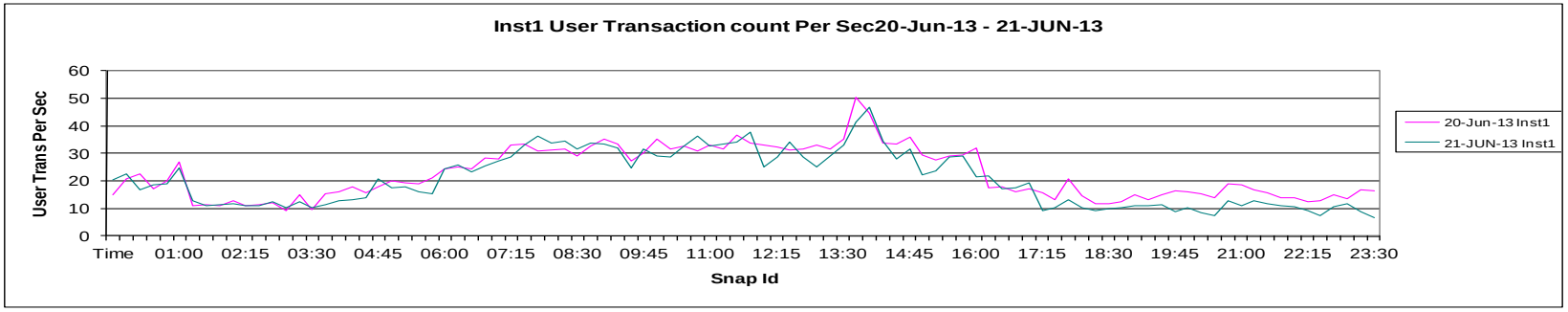
View top SQL for any period retained in AWR - highlights any issues

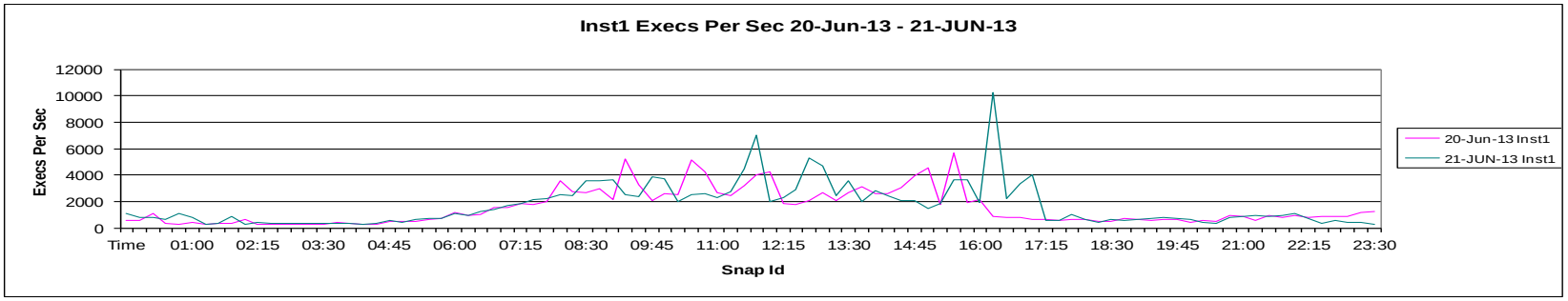
Top SQL by CPU Time 21-Jun-2013				
Instance	Rank	SQL_ID	Module	cpu_time (secs)
1	1	**crab2g2zzrc9b**	**JDBC Thin Client**	**22704.7**
1	2	**85gad38y8bvv3**	**po.contract.server.PoxSysVariableAM**	**22064.3**
1	3	**54tzhmhcs8nkh**	**po.contract.server.PoxSysVariableAM**	**21942**
1	4	4vyrtmsstrn9v	XXWMMINVONHANDSTOCK	20943.7
1	5	**ct25z5rpfyht4**	**po.contract.server.PoxSysVariableAM**	**11436.5**
1	6	d5kqafyftwpjd	e:CE:frm:CEXCABMR	8936.8
1	7	c92uxxwhs2sbz	JDBC Thin Client	6113
1	8	39rzvsfbz9jaj	po.contract.server.PoxSysVariableAM	5681.9
1	9	114064tys579d	po.contract.server.PoxSysVariableAM	5667.1
1	10	77pbkcg8jcn41	po.contract.server.PoxSysVariableAM	5655.9
2	1	**crab2g2zzrc9b**	**JDBC Thin Client**	**22807.8**
2	2	**85gad38y8bvv3**	**po.contract.server.PoxSysVariableAM**	**21303.3**
2	3	**54tzhmhcs8nkh**	**po.contract.server.PoxSysVariableAM**	**21186.5**
2	4	**ct25z5rpfyht4**	**po.contract.server.PoxSysVariableAM**	**10939.6**
2	5	c92uxxwhs2sbz	JDBC Thin Client	5819.8
2	6	39rzvsfbz9jaj	po.contract.server.PoxSysVariableAM	5452.2
2	7	an16u85vdrmz2	e:SQLAP:frm:APXINWKB	5431.2
2	8	114064tys579d	po.contract.server.PoxSysVariableAM	5430.2
2	9	77pbkcg8jcn41	po.contract.server.PoxSysVariableAM	5426.8
2	10	0avruyx7w9c13	FAS400	5242.3

MORRISONS

Current – examples - AWR

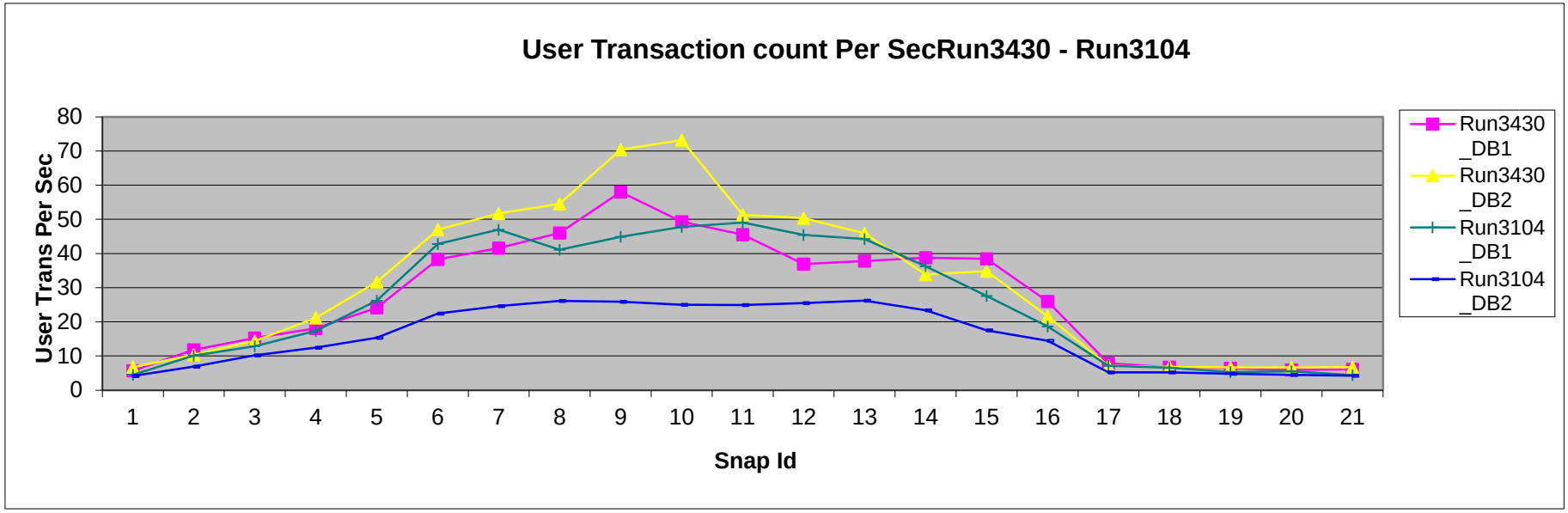
Comparing Transaction count by day… from AWR data

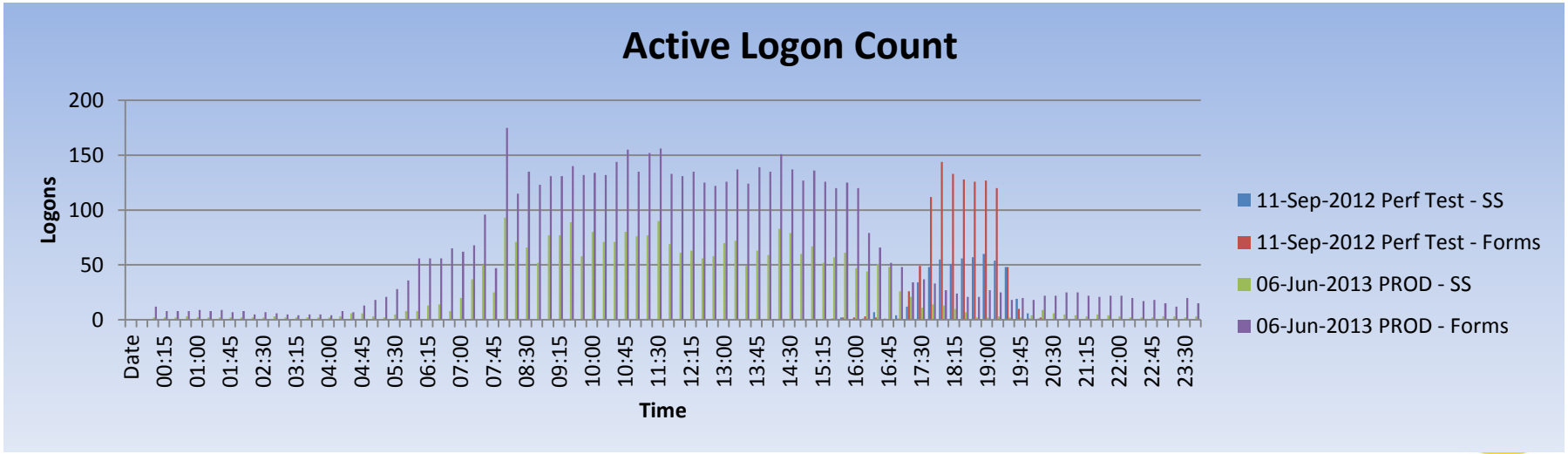




MORRISONS

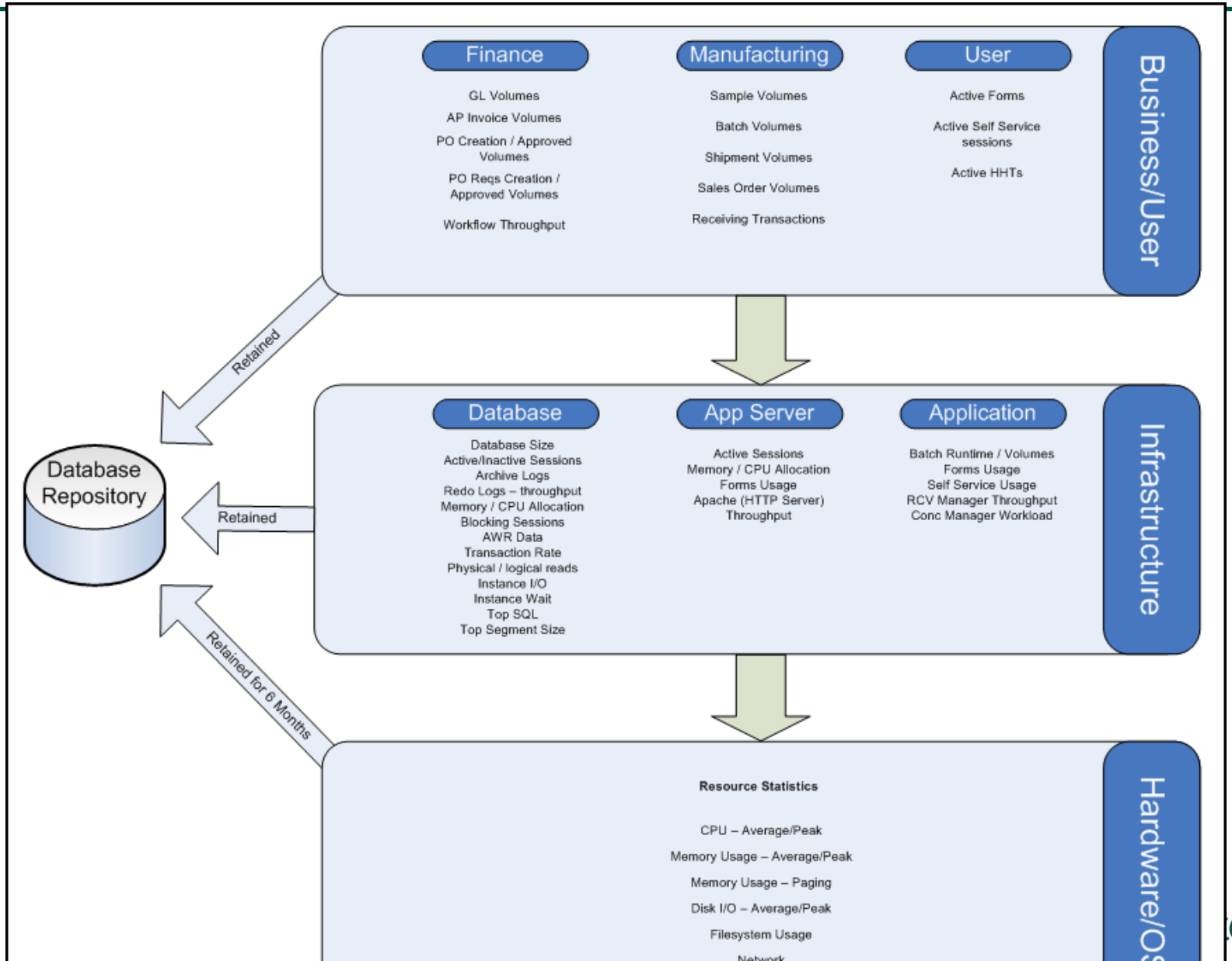
Current examples – performance testing analysis - AWR





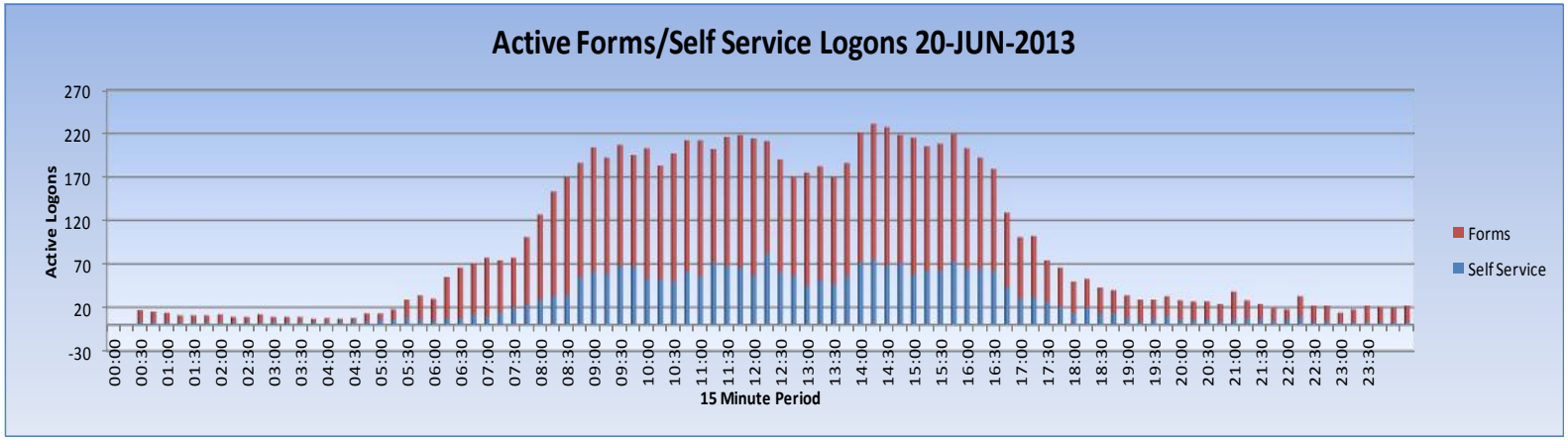
MORRISONS

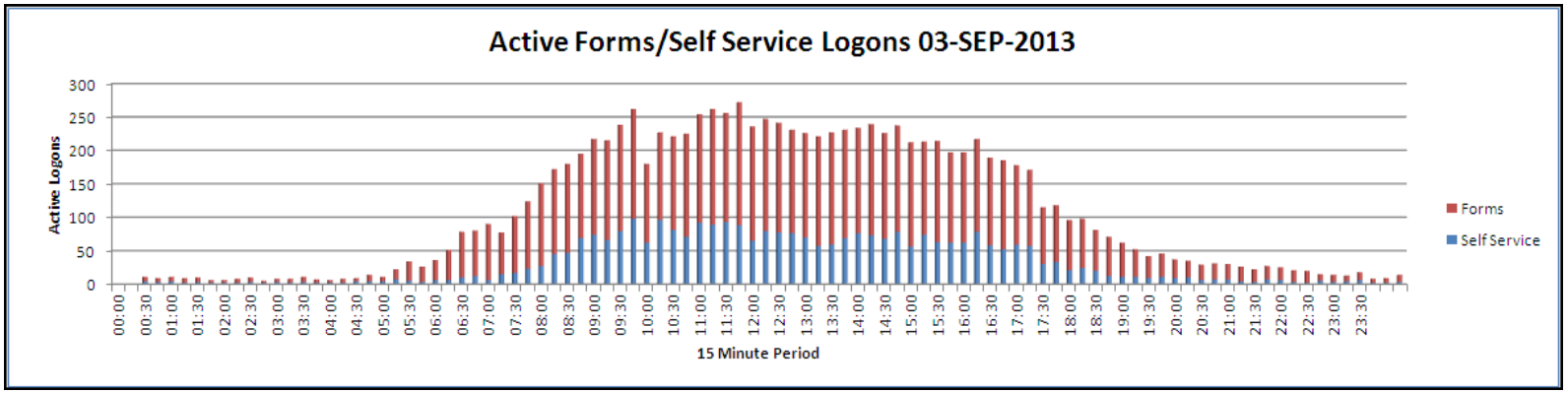
Three Tier Data Collection Model



Current – examples – Application (OEM – UDMs)

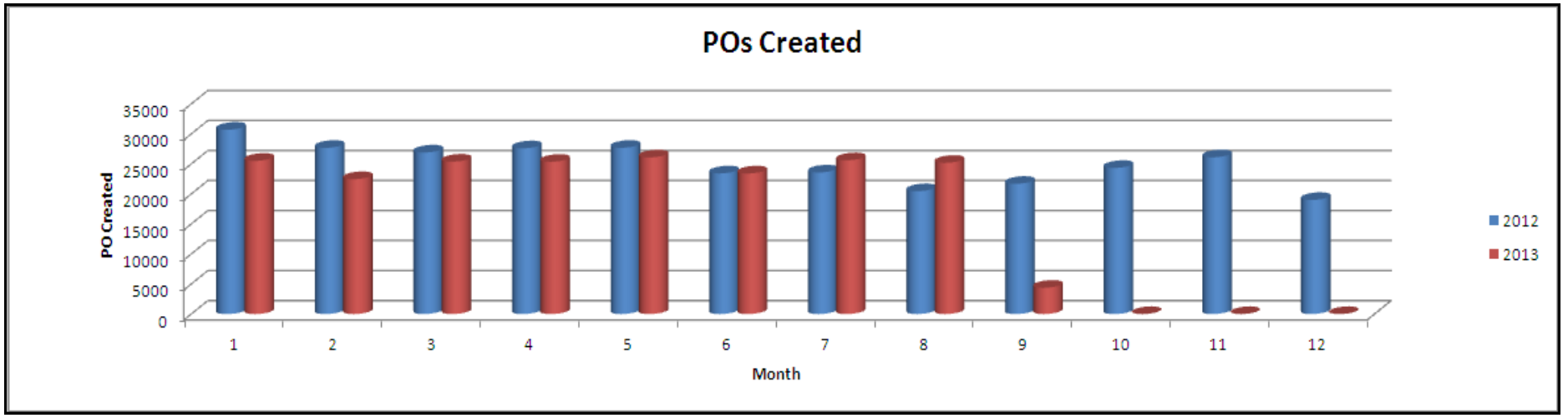
Capture and measure OLTP user logon activity





MORRISONS

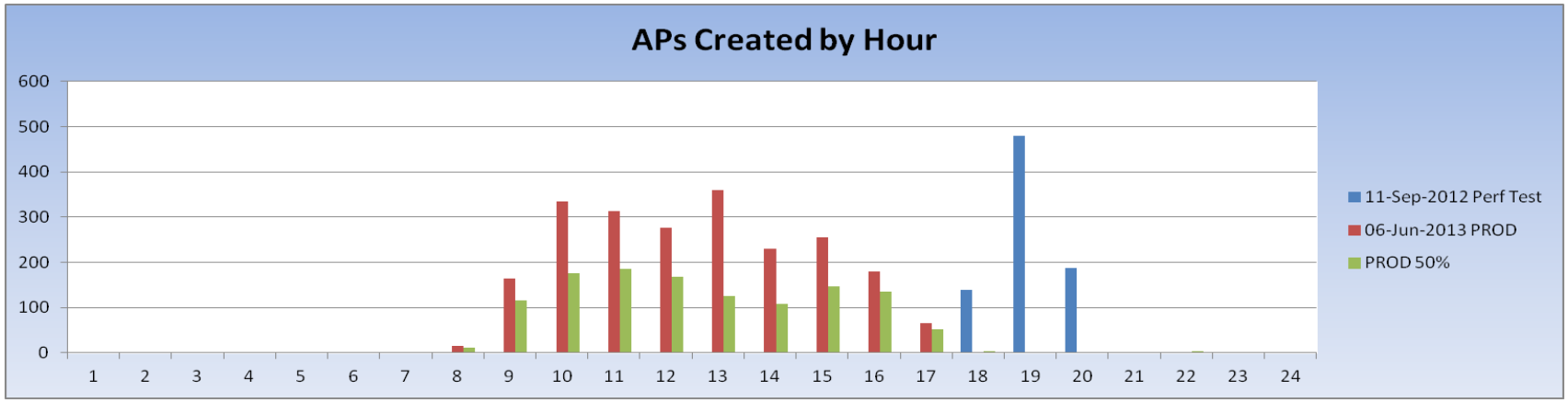
Current – examples – Application (OEM – UDMs)

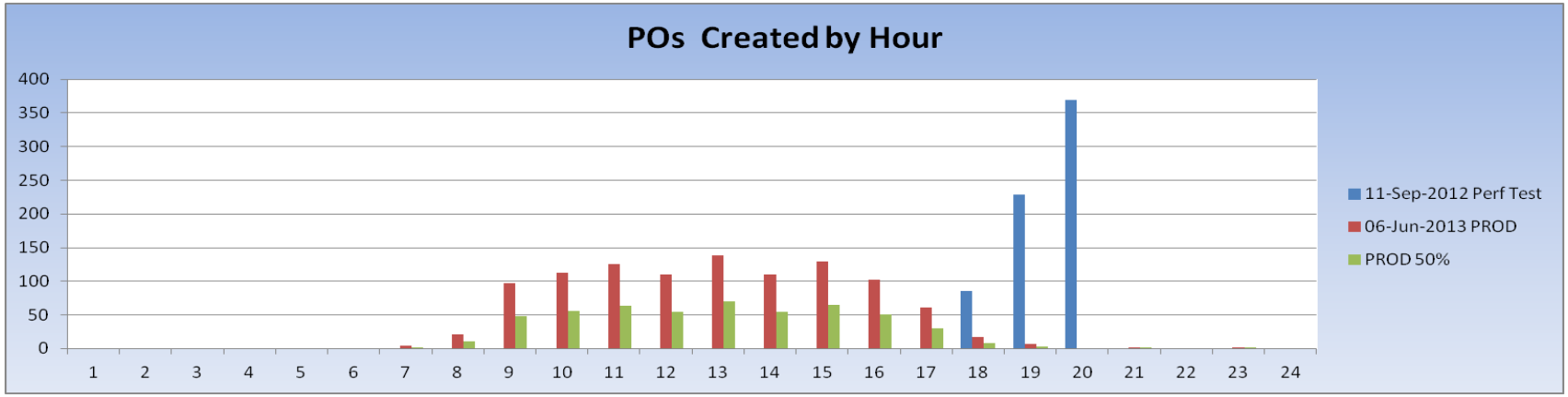


MORRISONS

Current – examples – Application (OEM – UDMs)

Capture Performance testing transactions and compare against Production actuals…





MORRISONS

Capturing unix data from OEM

Get the target_guid –

```
select target_guid , target_name, host_name, target_type from mgmt_targets
where host_name like '%peb%' and  target_type='host'
```

use target_guid and dates in the below query:

```
 select * from (
      select  target_name
      , collection_timestamp , to_char(collection_timestamp, 'HH24:MI'), rank() over ( order by
  collection_timestamp)  PERIOD5MIN
      , cpu_util , cpu_user , cpu_sys, memfreePct, memUsedPct, freeMem, freeSwap, usedSwap, cpuLoad
      from (select target_name,  collection_timestamp
                ,sum(decode(metric_column, 'cpuUtil', value,0)) CPU_Util
                ,sum(decode(metric_column, 'cpuUser', value,0)) CPU_User
                ,sum(decode(metric_column, 'cpuKernel', value,0)) CPU_SYS
                ,sum(decode(metric_column, 'cpuLoad', value,0)) cpuLoad
                ,sum(decode(metric_column, 'memfreePct', value,0)) memfreePct
                ,sum(decode(metric_column, 'memUsedPct', value,0)) memUsedPct
                ,sum(decode(metric_column, 'freeMem', value,0)) freeMem
                ,sum(decode(metric_column, 'freeSwap', value,0)) freeSwap
                ,sum(decode(metric_column, 'usedSwap', value,0)) usedSwap
                from mgmt$metric_details a
                where a.target_guid= '2D43ADB039AFBED6DE98EE960B58FC28'
                and collection_timestamp between to_date('03-JUL-2013  00:00', 'DD-MON-YYYY HH24:MI')
                                        and     to_date('04-JUL-2013  00:00', 'DD-MON-YYYY HH24:MI')
                and metric_name like 'Load'
                and metric_column in ('cpuUtil', 'cpuUser', 'cpuKernel','cpuLoad', 'memfreePct', 'memUsedPct',
  'freeMem' , 'freeSwap', 'usedSwap')
                group by target_name,  collection_timestamp order by 2))
```

MORRISONS

Default OEM Repository Purging Policies

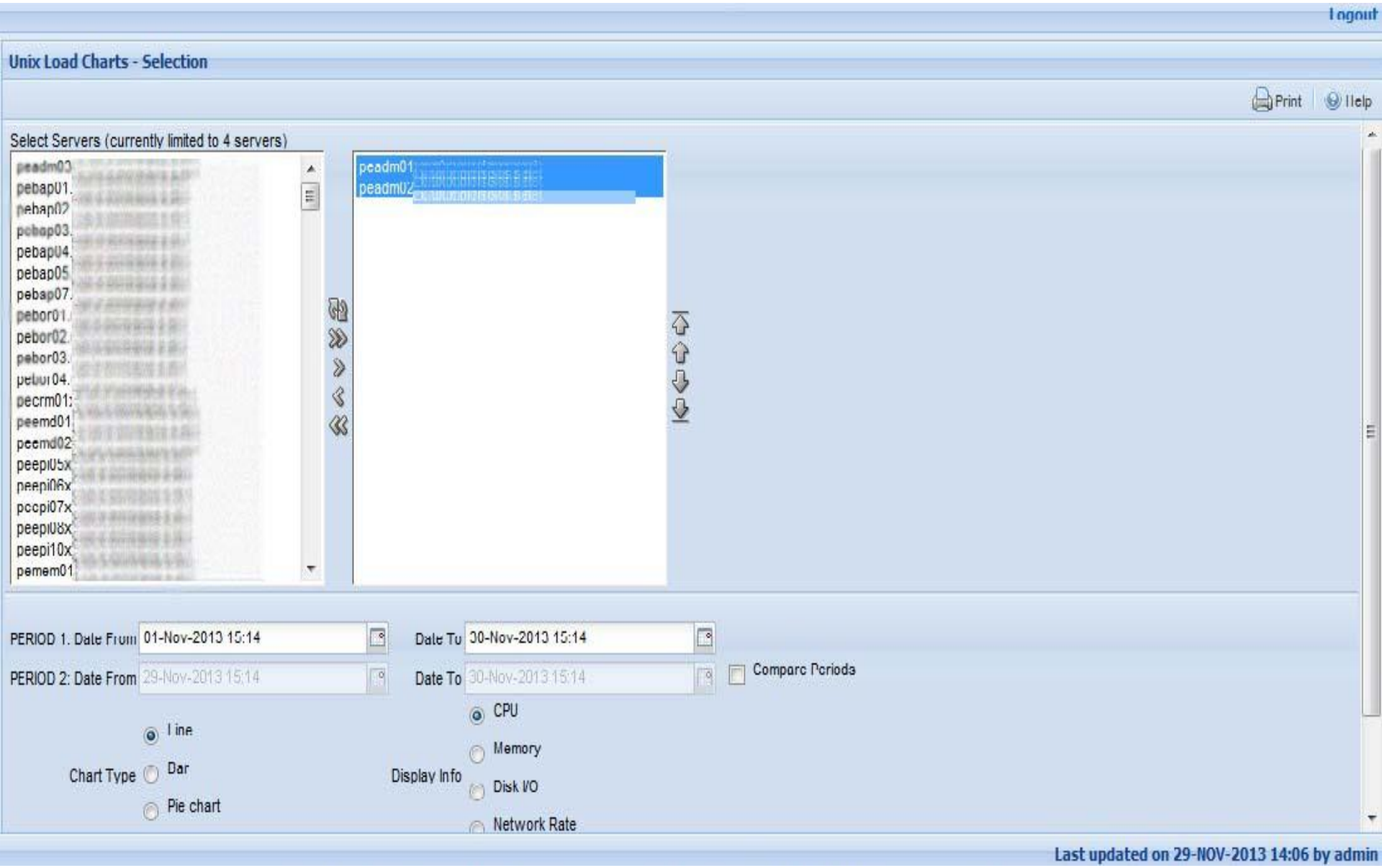
Aggregate Level Retention Time Raw metric data	7 days
Hourly aggregated metric data	31 days
Daily aggregated metric data	~365 days

```
select table_name, partitions_retained
from em_int_partitioned_tables
where table_name in ('EM_METRIC_VALUES','EM_METRIC_VALUES_HOURLY','EM_METRIC_VALUES_DAILY');

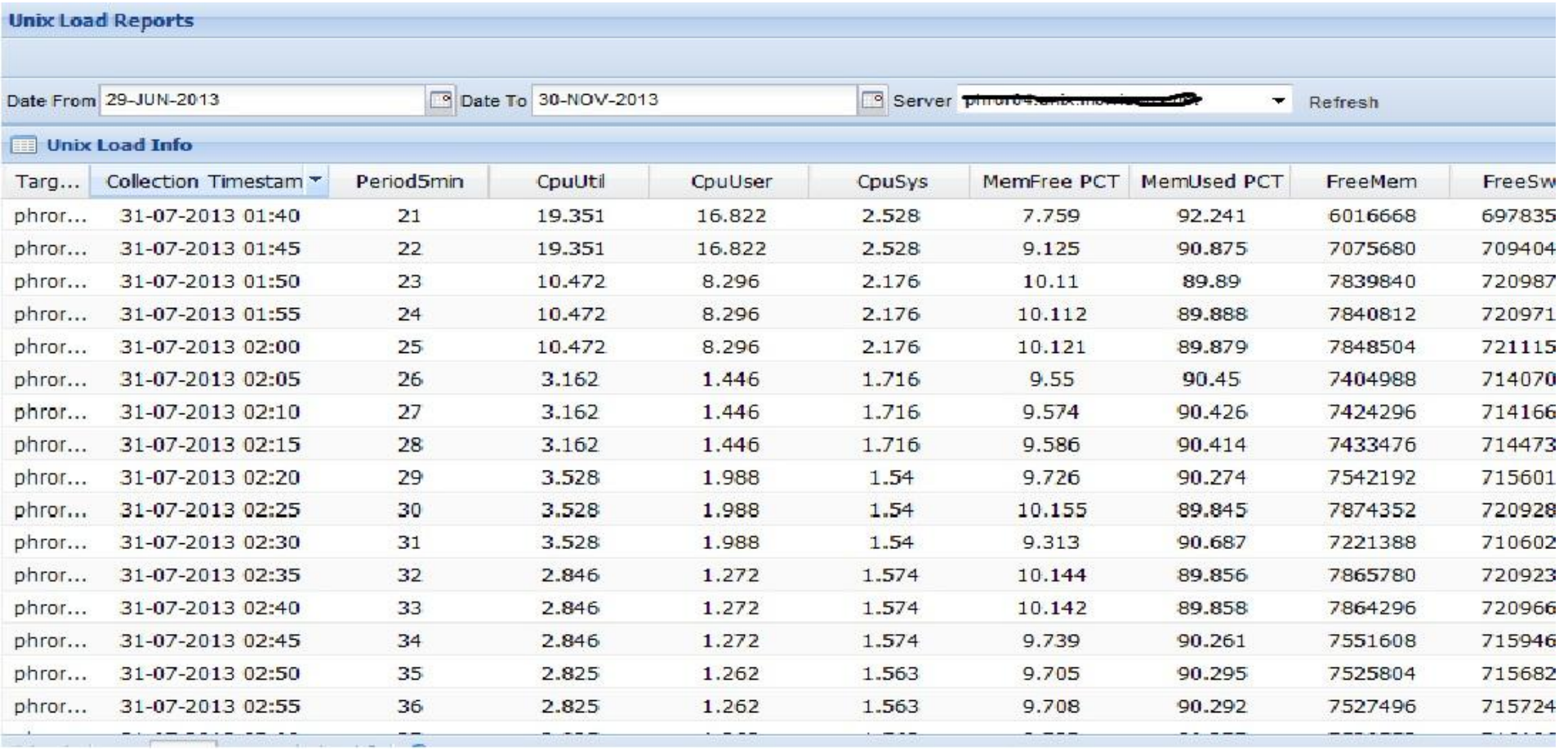
TABLE_NAME                     PARTITIONS_RETAINED
------------------------------ -------------------
EM_METRIC_VALUES                                 7
EM_METRIC_VALUES_HOURLY                         32
EM_METRIC_VALUES_DAILY                           2
```

MORRISONS

OEM usage


Logout
Unix Load Charts - Selection
Print
Help
Select Servers (currently limited to 4 servers)
peadm01
peadm02
PERIOD 1. Date From 01-Nov-2013 15:14
Date To 30-Nov-2013 15:14
PERIOD 2: Date From 29-Nov-2013 15:14
Date To 30-Nov-2013 15:14
Compare Periods
Chart Type
Line
Bar
Pie chart
Display Info
CPU
Memory
Disk I/O
Network Rate
Last updated on 29-NOV-2013 14:06 by admin

OEM usage



Unix Load Reports

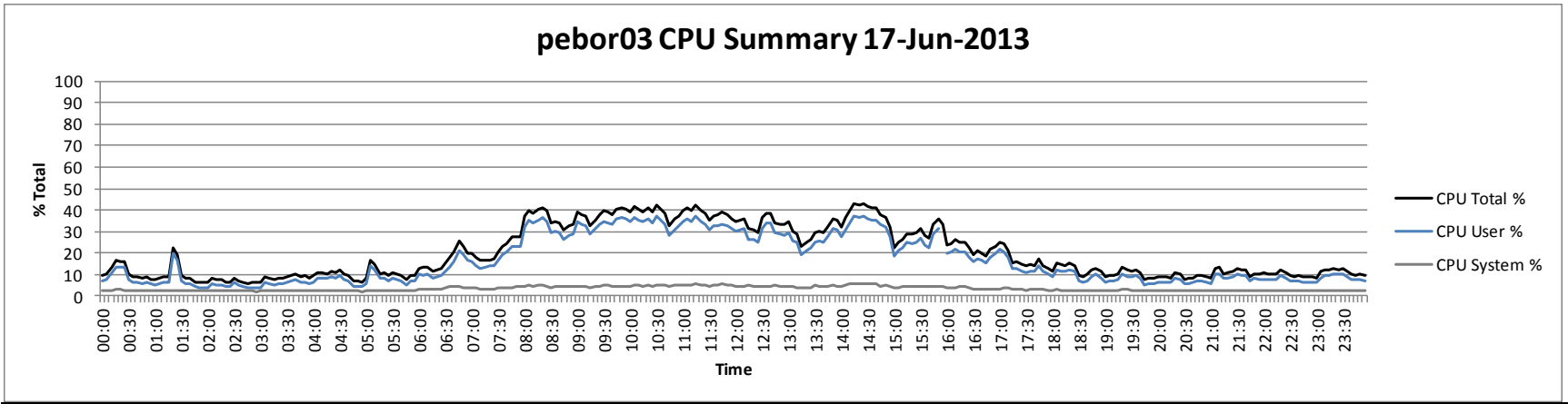
Date From 29-JUN-2013 Date To 30-NOV-2013 Server [redacted] Refresh

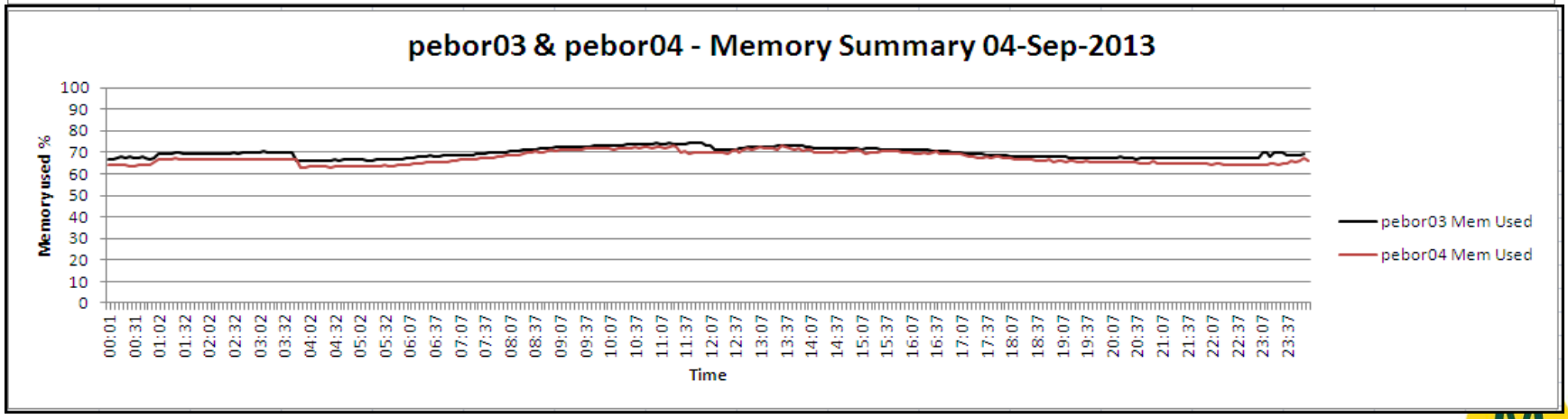
Unix Load Info

Targ...	Collection Timestam	Period5min	CpuUtil	CpuUser	CpuSys	MemFree PCT	MemUsed PCT	FreeMem	FreeSw
phror...	31-07-2013 01:40	21	19.351	16.822	2.528	7.759	92.241	6016668	697835
phror...	31-07-2013 01:45	22	19.351	16.822	2.528	9.125	90.875	7075680	709404
phror...	31-07-2013 01:50	23	10.472	8.296	2.176	10.11	89.89	7839840	720987
phror...	31-07-2013 01:55	24	10.472	8.296	2.176	10.112	89.888	7840812	720971
phror...	31-07-2013 02:00	25	10.472	8.296	2.176	10.121	89.879	7848504	721115
phror...	31-07-2013 02:05	26	3.162	1.446	1.716	9.55	90.45	7404988	714070
phror...	31-07-2013 02:10	27	3.162	1.446	1.716	9.574	90.426	7424296	714166
phror...	31-07-2013 02:15	28	3.162	1.446	1.716	9.586	90.414	7433476	714473
phror...	31-07-2013 02:20	29	3.528	1.988	1.54	9.726	90.274	7542192	715601
phror...	31-07-2013 02:25	30	3.528	1.988	1.54	10.155	89.845	7874352	720928
phror...	31-07-2013 02:30	31	3.528	1.988	1.54	9.313	90.687	7221388	710602
phror...	31-07-2013 02:35	32	2.846	1.272	1.574	10.144	89.856	7865780	720923
phror...	31-07-2013 02:40	33	2.846	1.272	1.574	10.142	89.858	7864296	720966
phror...	31-07-2013 02:45	34	2.846	1.272	1.574	9.739	90.261	7551608	715946
phror...	31-07-2013 02:50	35	2.825	1.262	1.563	9.705	90.295	7525804	715682
phror...	31-07-2013 02:55	36	2.825	1.262	1.563	9.708	90.292	7527496	715724

OEM now

Future intention is to capture and retain UNIX resource data for long term reporting.
Also to overlay charting for ease of extract





MORRISONS

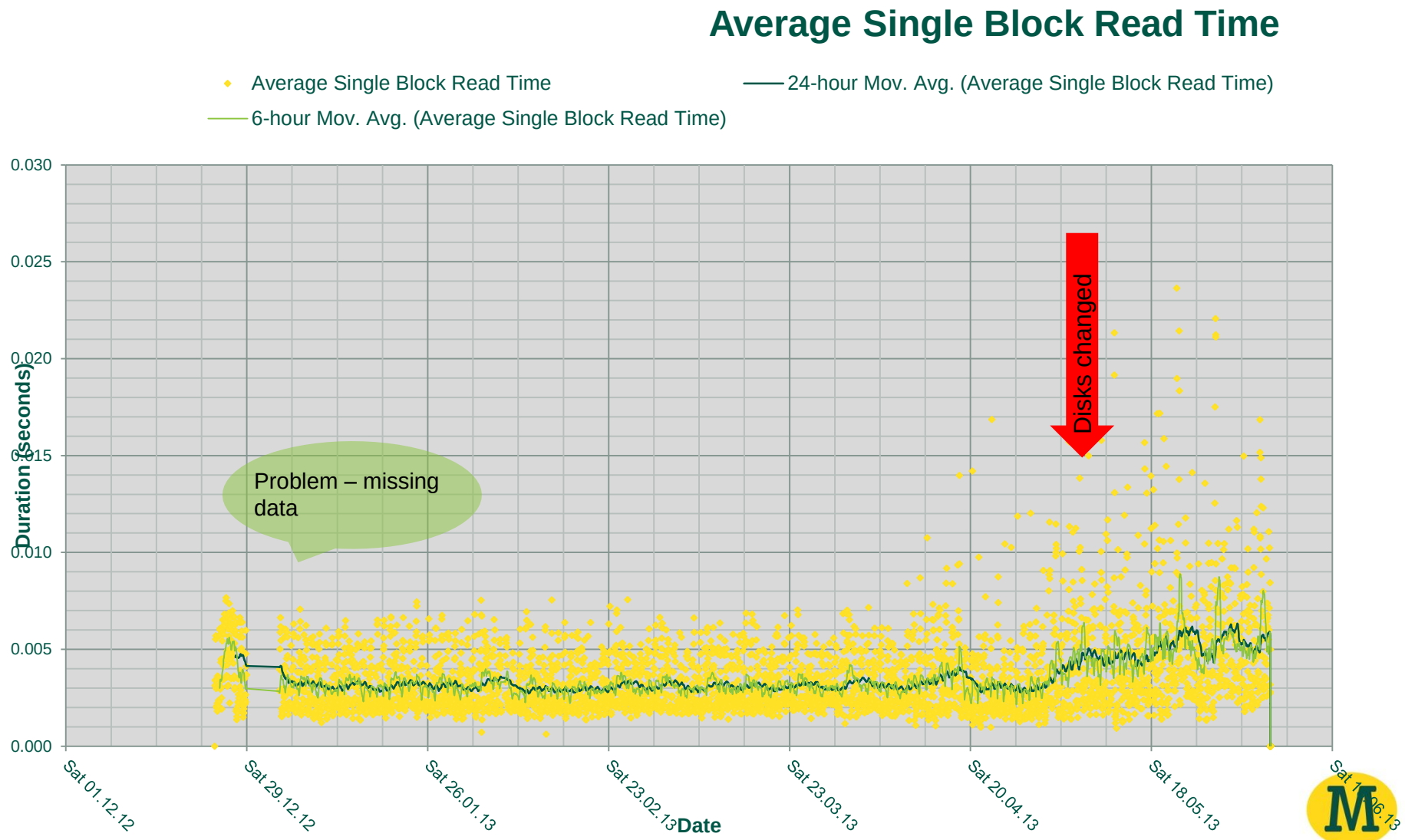
Problems we have had

- Oracle SCN issue - good blog at http://www.orainternals.com/2012/01/20/scn-what-why-and-how/ - how the SCN can jump -the AWR extract process is effectively re-synching all of the scns across all prod databases.
- Random drop out of the dbms_file_transfer process on HPUX – hard to track
- The data_pump_dir area on one server can be mounted with the incorrect options - not really an oracle issue
- Might want to go to one central area for all files – mounted across all systems – security – NOW AGREED
- Loss of snapshots - initially not a problem
- Indexing - need to specify on what you might query on

```
CREATE INDEX sys. WRH$_ACTIVE_SESSION_HISTORY2
ON sys. WRH$_ACTIVE_SESSION_HISTORY (dbid, sql_id, sample_time)
COMPRESS 2;
```

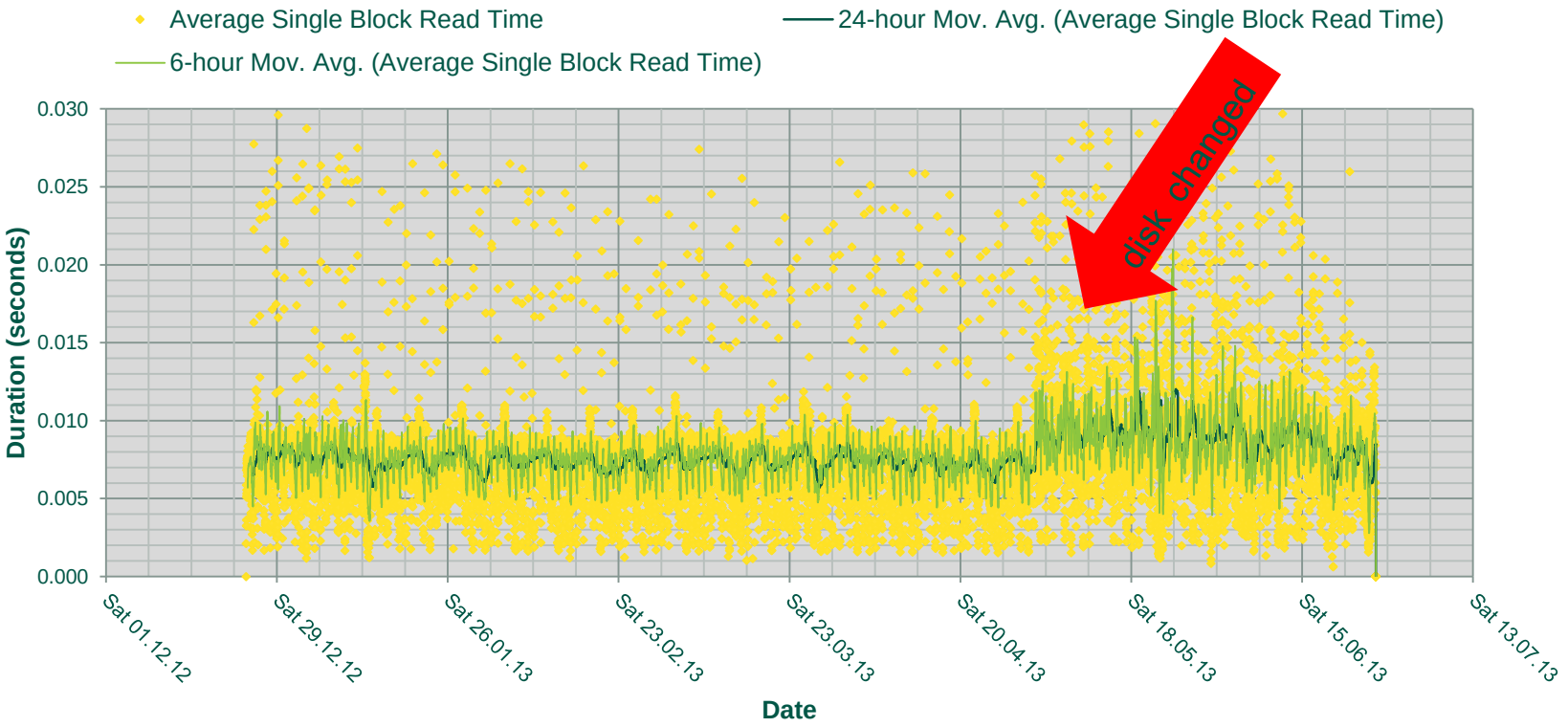
MORRISONS

Real World examples



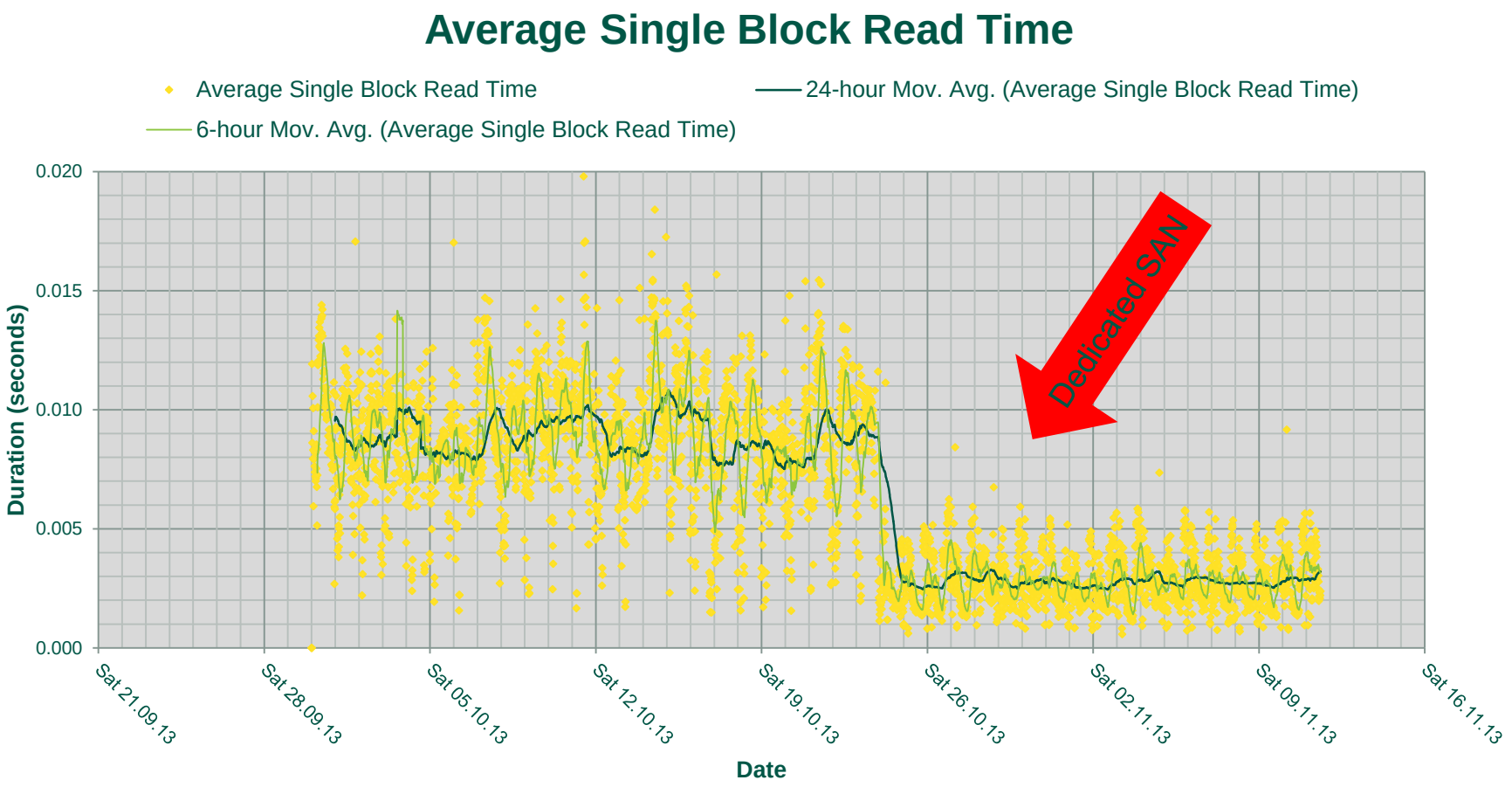
MORRISONS

Average Single Block Read Time



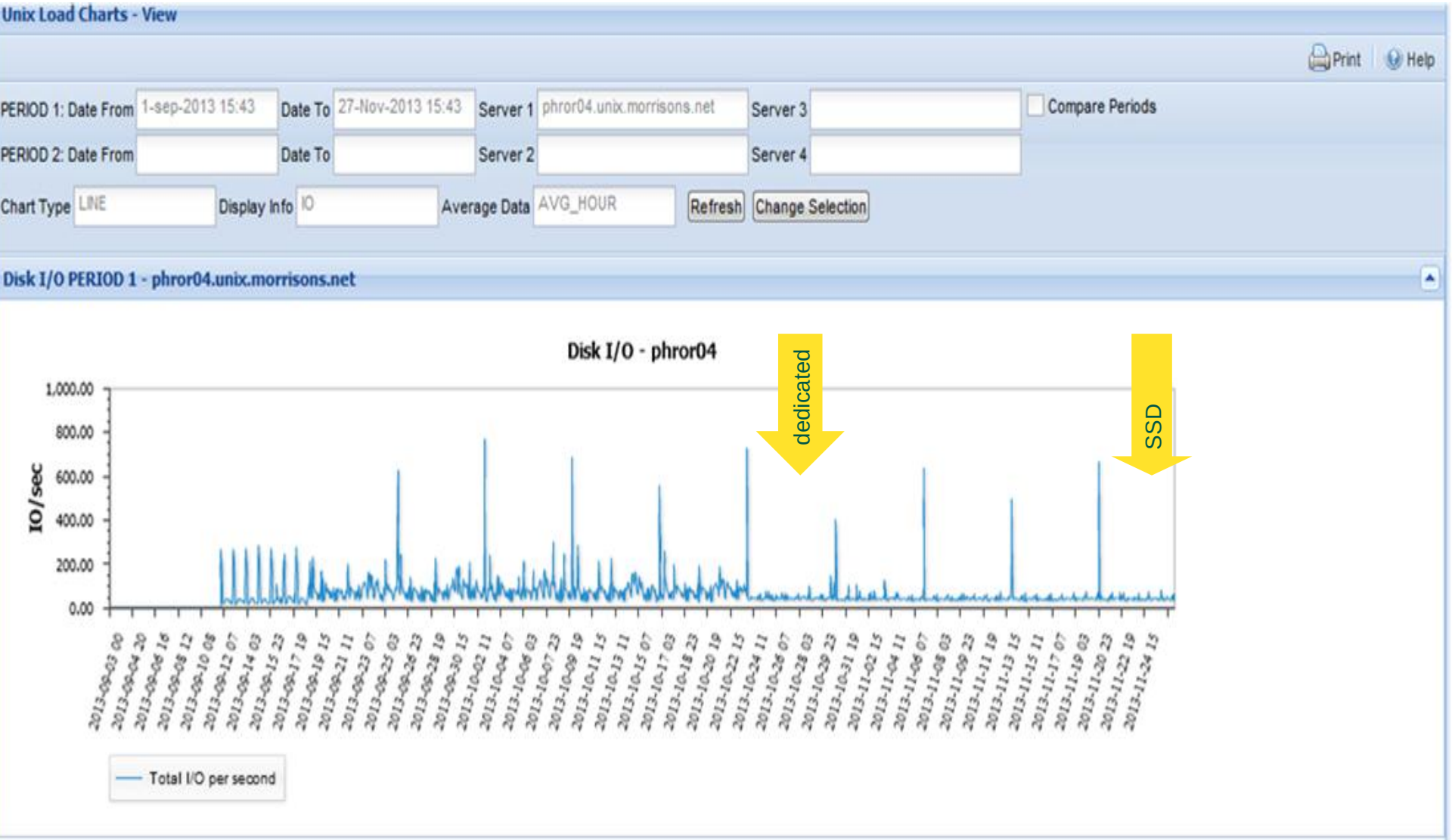
MORRISONS

Migration to dedicated array



MORRISONS

OEM usage


Unix Load Charts - View
Print
Help
PERIOD 1: Date From 1-sep-2013 15:43
Date To 27-Nov-2013 15:43
Server 1 phror04.unix.morrisons.net
Server 3
Compare Periods
PERIOD 2: Date From
Date To
Server 2
Server 4
Chart Type LINE
Display Info IO
Average Data AVG_HOUR
Refresh
Change Selection
Disk I/O PERIOD 1 - phror04.unix.morrisons.net
Disk I/O - phror04
dedicated
SSD
IO/sec
1,000.00
800.00
600.00
400.00
200.00
0.00
Total I/O per second

Standard deviation script – slower/ faster

SQL_ID	EXECS	AVG_ETIME_BEFORE	AVG_ETIME_AFTER	NORM_STDDEV	RESULT
b9wdngtzbgc35	3	0.10	0.39	2.0078	Slower
12a2xbmwn5v6z	2	0.67	0.15	2.4628	Faster
7g732rx16j8jc	2	0.12	0.02	2.7280	Faster
6ajkhukk78nsr	42	0.09	0.47	2.8520	Slower
bwsz40d3hc8q7	3	0.51	0.10	2.9133	Faster
6hwjmjgrpsuaa	2	1.17	0.19	3.5473	Faster
1swfv5ub781u9	5	0.04	0.27	4.0006	Slower
6b838v5tfawb2	9	1.37	0.18	4.8256	Faster
5zruc4v6y32f9	8	362.62	43.39	5.2017	Faster
acc988uzvjmmt	3	9.10	0.97	5.9364	Faster
aj49v4fg8v5y0	5	0.03	0.34	6.5912	Slower
9q7k9nbpvk8pv	180	0.03	0.41	8.3757	Slower
2xucgknahj256	2	0.26	5.22	13.5279	Slower
4rcrxkvaqt37f	3	0.00	0.15	20.3349	Slower
1k5c5twx2xr01	13	0.00	0.86	395.2906	Slower

http://www.oracle-guy.com/scripts/whats_changed.sql

MORRISONS

Compare period on period

SQL_ID	PERIOD1_EXECS	PERIOD2_EXECS	QUICKEST_BEFORE	SLOWEST_NOW	PERC_SLOWER	RESULT
f665gdvw5t370	277	274	.03	.37	1133.33	Slower
94g5mzt64mvwx	72791	99197	.01	.11	1000	Slower
22x793wtkjngz	17	209	.03	.3	900	Slower
3xgkk1qvpuu4h	360	346	2.06	20.17	879.13	Slower
20f21whx3uwnd	75	250	.05	.29	480	Slower
6vg5jjvpcdzh9	64367	66106	.03	.1	233.33	Slower
c1a40y4t02dmr	149	39	.06	.19	216.67	Slower
4t371cqv6f422	1	1	92.74	235.78	154.24	Slower
fy1c7x7hpkty5	1	1	92.74	235.77	154.23	Slower
2d3dvb5uavkay	14059	16035	.04	.08	100	Slower
19txv5k2cvkus	250	191	1.6	2.52	57.5	Slower

11 rows selected.

MORRISONS

Real world examples and usage

- Tracking growth in no of stores and volumes – shopping basket sales
- Comparing 16 distribution sites to see if same problems exist on both – disk I/O, access times
- Investigate long-term disk I/O metrics (SBR, MBR, is log_file_sync getting worse)

- The reports can be generated using the standard scripts, but this requires the user to have an oracle client installed (SQL Developer will do) or run from the database server.
- Can also run reports using procedure calls e.g.

  select output from table(dbms_workload_repository.awr_report_text(583324766,1,3000,3001));

  select output from table(dbms_workload_repository.awr_report_html(583324766,1,3000,3001));

  the html spews out raw html so needs to be saved in a file and opened in a browser and the hyperlinks don't work but the format is more readable

MORRISONS

More Information

- Blog: www.jhdba.wordpress.com
- Email: john.hallas@morrisonsplc.co.uk
- Tel: 07876 790540

Any Questions?

Thank You