

# New 11g Alert Log Monitoring Capabilities

*“People who like XML are like people who like licorice. Not everybody likes licorice, but the people who like licorice really like licorice.” --Jerry Garcia  
(slightly paraphrased)*

Andy Klock  
Sr. DBA/Developer  
Portware, LLC

# Andy Klock

---

10+ years as an Oracle DBA / Developer

EROS Data Center – South Dakota

Oracle RDBMS / Oracle Applications DBA – Boston

Financial Software Developer – New York

OCP 8, 8i, 9i, 10g

**ORACLE®**

---

**Certified Professional**

# New Diagnosability Framework

---

- First Failure diagnosis
- Incident Packaging Service (IPS)
- Health Checks
- Automatic Diagnosability Repository (ADR)
- Standardized trace formats
- Utilities for data access

# Automatic Diagnostic Repository (ADR)

---

- File Based Repository
  - Centralized for all products
  - Accessible when database is down
- Contains
  - Health Monitor Reports
  - Incident packages
  - Trace files
  - Alert log
  - (others)

# ADR Structure

---

- ADR Base
- ADR Products/Components
- ADR Homes

# ADR Base Configuration

---

- Set DIAGNOSTIC\_DEST system parameter
- Defaults to the value in \$ORACLE\_BASE
- Defaults to \$ORACLE\_HOME/log.

```
SQL> show parameter diagnostic_dest
```

NAME	TYPE	VALUE
-----	-----	-----
diagnostic_dest	string	C:\APP\ORACLE

# ADR Product/Components

---

**ADR Base:**  
**/u01/app/oracle**



**diag:**  
**/u01/app/oracle/diag**



**asm**



**clients**



**rdbms**



**tnslnr**



**other  
products**

# ADR Home

---

ADR\_Base/diag/product\_id/instance\_id

For example:

ADR\_Base/diag/asm/+asm/+asm/

ADR\_Base/diag/rdbms/alertdemo/alertdemo

ADR\_Base/diag/asm/hostname/listener



# ADR Home Subdirectories

---

**ADR Base:**  
**/u01/app/oracle**



**diag:**  
**/u01/app/oracle/diag**



**rdbms**  
**/u01/app/oracle/diag/rdbms**



**DB\_UNIQUE\_NAME**  
**/u01/app/oracle/diag/rdbms/alrtdemo**



**ORACLE\_SID**  
**/u01/app/oracle/diag/rdbms/alrtdemo/ALRTDEMO**



**alert**



**cdump**



**incident**



**trace**



**hm**



**(others)**

# V\$DIAG\_INFO

---

```
SQL> select * from v$diag_info;
```

INST_ID	NAME	VALUE
1	Diag Enabled	TRUE
1	ADR Base	c:\app\oracle
1	ADR Home	c:\app\oracle\diag\rdbms\alertdemo>alertdemo
1	Diag Trace	c:\app\oracle\diag\rdbms\alertdemo>alertdemo\trace
1	Diag Alert	c:\app\oracle\diag\rdbms\alertdemo>alertdemo>alert
1	Diag Incident	c:\app\oracle\diag\rdbms\alertdemo>alertdemo\incident
1	Diag Cdump	c:\app\oracle\diag\rdbms\alertdemo>alertdemo\cdump
1	Health Monitor	c:\app\oracle\diag\rdbms\alertdemo>alertdemo\hm
1	Default Trace File	c:\app\oracle\diag\rdbms\alertdemo>alertdemo\trace>alertdemo_ora_448.trc
1	Active Problem Count	4
1	Active Incident Count	12

11 rows selected.

# Deprecated Parameters

---

- background\_core\_dump
- background\_dump\_dest
- core\_dump\_dest
- user\_dump\_dest

Will throw ORA-32004 if manually set:

`ORA-32004: obsolete or deprecated parameter(s) specified for RDBMS instance`

# Alert Log

---

- Historical database log
  - Critical database errors
  - Administrative operations
    - System parameter changes
    - Database Startup up or shutdown database
    - Recoveries
    - Adding or dropping tablespaces/datafiles
  - Other database events

# Alert Log (cont)

---

- DDL logging

```
ALTER SYSTEM SET enable_ddl_logging=true scope=both;
```

## Sample Alert log output:

```
adrci> show alert -term -p "message_text like '%drop%'"
```

```
ADR Home = C:\app\oracle\diag\rdbms\alertdemo>alertdemo:
```

```
*****
```

```
2010-02-05 04:37:33.875000 -04:00
```

```
drop table scott.emp;
```

# Alert Log (cont)

---

- Messages can be tied to:
  - Modules
  - Processes
  - Threads
  - Users
- Enables critical errors to be associated and packaged with problems and incidents
- Can associate problems across products and components

# Alert Log Maintenance

---

- Rolling over alert logs
  - Automatically rolled when 10mb
  - Named log\_nn.xml
- Warning
  - Don't use log\_nn.xml with a leading zero
    - e.g. log\_06.xml
  - Don't use special characters with log\_nn
    - e.g. log\_6-09-2010.xml

# Alert Log Naming Error

---

## Using log\_06-09-2010.xml

```
adrci> show alert
```

```
ADR Home = /u01/app/oracle/diag/rdbms/alrtdemo/ALRTDEMO:
*****
DIA-48156: Alert log purge has occurred - retry operation
DIA-48121: error with opening the ADR stream file
[/u01/app/oracle/diag/rdbms/alrtdemo/ALRTDEMO/alert/log_6.xml] [0]
DIA-48127: attempting to open a file that does not exist
Linux-x86_64 Error: 2: No such file or directory
Additional information: 5
```



# Alert Log Purging

---

- Scripts

  - To delete, move, or rename

- ADRCI

  - `purge -age min -type`

# Alert Log Access

---

- OEM Database Control
- ADRCI
- X\$DBGALERTEXT
- XML DB
- Anything that can parse XML

# OEM Database Control

---

- Related Links -> Alert Log Contents
- Demo

# ADRCI

---

- ADR Command Interpreter
- Incidents and problems
- Health check reports
- Package and upload diagnostic data to Support
- Parse and purge Alert logs

# ADRCI (cont)

---

- Set ADR Home
  - `set home = diag\rdbms\alertdemo>alertdemo`
  - Pick from home list
  - Launch ADRCI from within homepath directory
- Set EDITOR
  - use default
  - `set editor vi`
  - Or use `-term` option

# ADRCI (cont)

---

- `show alert`
  - `-tail | tail -f`
  - `-p` SQL-like filtering
  - `-script`
  - `-exec`
- `purge -age min -type`
- **DEMO**

# X\$DBGALERTEXT

---

- No longer need UTL\_FILE or External Tables

Example:

```
SELECT message_text, ORIGINATING_TIMESTAMP
       FROM X$DBGALERTEXT
       WHERE message_text like '%ORA-%';
```

# XML DB

---

- Benefits
  - Access files that aren't in the ADR Home
  - Keep alert directory size minimized
  - Use XML DB features
- Issues
  - Need Oracle Directory Object
  - log.xml is not well formed



# XML DB (cont)

---

## Alert log is not well formed

```
SELECT (XMLTYPE(bfilename('ALERT', 'archived_log_23.xml'),  
NLS_CHARSET_ID('AL32UTF8'))))AS "ALERT" from dual;
```

ERROR:

ORA-31011: XML parsing failed

ORA-19202: Error occurred in XML processing

LPX-00245: extra data after end of document

Error at line 9

ORA-06512: at "SYS.XMLTYPE", line 296

ORA-06512: at line 1

# XML DB (cont)

---

To be well formed – add a parent tag

```
SQL> declare
  2     c clob;
  3     b bfile := bfilename('ALERT','archived_log_23.xml');
  4     begin
  5         dbms_lob.open(b,dbms_lob.file_readonly);
  6         dbms_lob.createtemporary(c,true);
  7         dbms_lob.loadfromfile(c,b,dbms_lob.lobmaxsize);
  8         dbms_xslprocessor.clob2file('<root>||c||</root>',
  9             'ALERT','archived_log_23.xml');
 10         dbms_lob.close(b);
 11     end;
 12     /
```

PL/SQL procedure successfully completed.

# XML DB (cont)

---

```
SQL> with alert_log as (SELECT
(XMLTYPE(bfilename('ALERT','archived_log_23.xml'),NLS_CHARSET_ID('AL32UTF8'))
2
3
4
5 select
6 p.parameters
7 from alert_log,
8 xmltable('for $i in /root/msg where $i/@msg_id eq "kspdmp:14914:3633090201" or
9 $i/@msg_id eq "kspdmp:14923:527288951"
10 return $i' passing log
11 COLUMNS
12 PARAMETERS VARCHAR2(150) PATH 'txt') as p;
```

## PARAMETERS

```
-----
processes = 150
memory_target = 512M
control_files = "C:\APP\ORACLE\ORADATA\ALERTDEMO\CONTROL01.CTL"
control_files = "C:\APP\ORACLE\FLASH_RECOVERY_AREA\ALERTDEMO\CONTROL02.CTL"
db_block_size = 8192
compatible = "11.2.0.0.0"
```

....

# Application Express (APEX)

---

- Included in 11g
- No longer requires Oracle HTTP Server
- HTTP Server Options
  - Embedded PL/SQL Gateway
    - Runs in XML DB HTTP
  - Oracle HTTP Server
    - Requires mod\_plsql

# APEX (cont)

---

- Demo

# Questions

---

[andy@oracledepot.com](mailto:andy@oracledepot.com)