
Oracle Gems

for DBAs and Developers

presentation for:

A red rectangular box containing a faint world map in a lighter shade of red. The text is centered over the map.

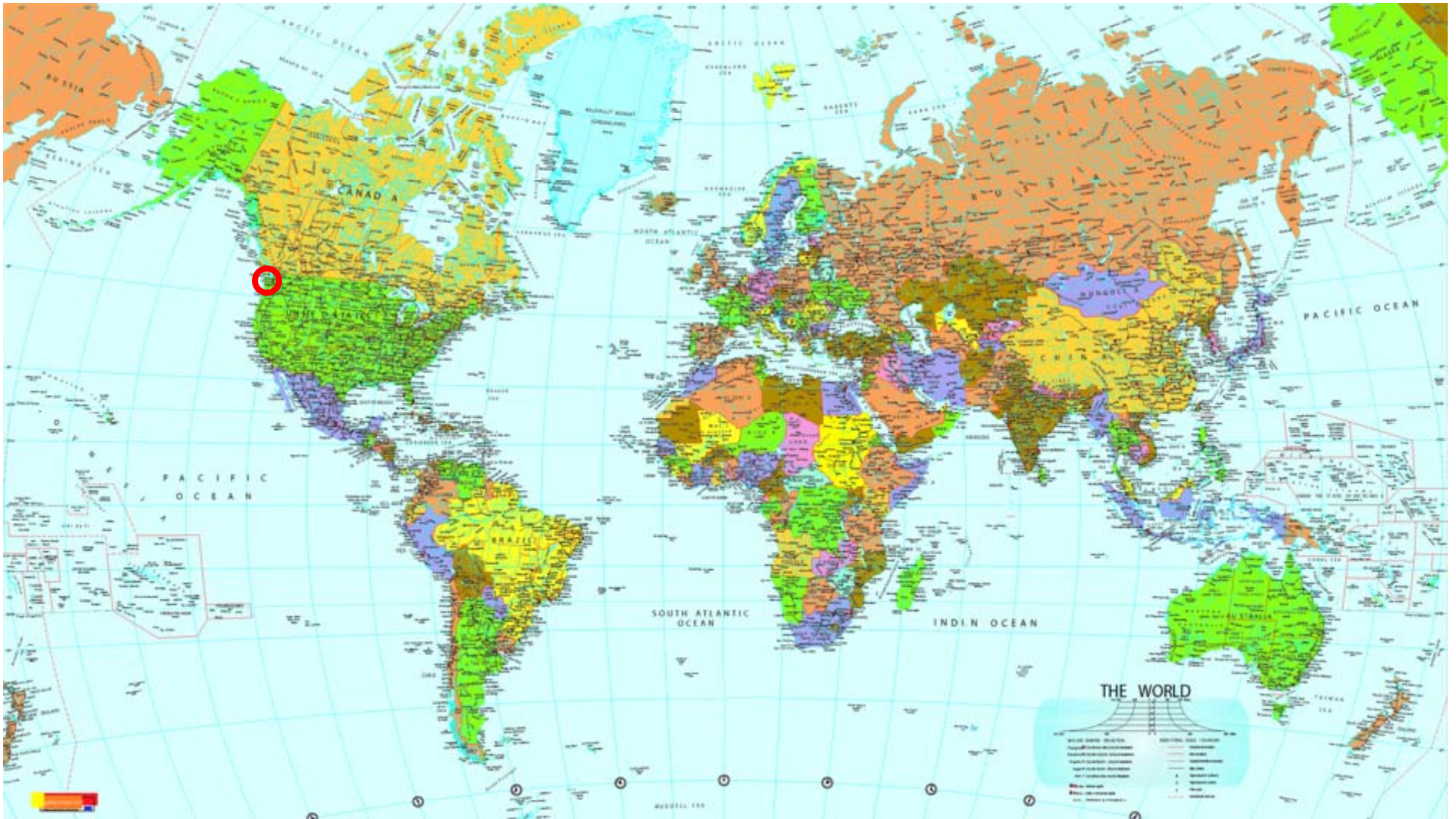
Chilean Oracle User Group
La Comunidad Oracle Chilena

Spring 2009 Conference

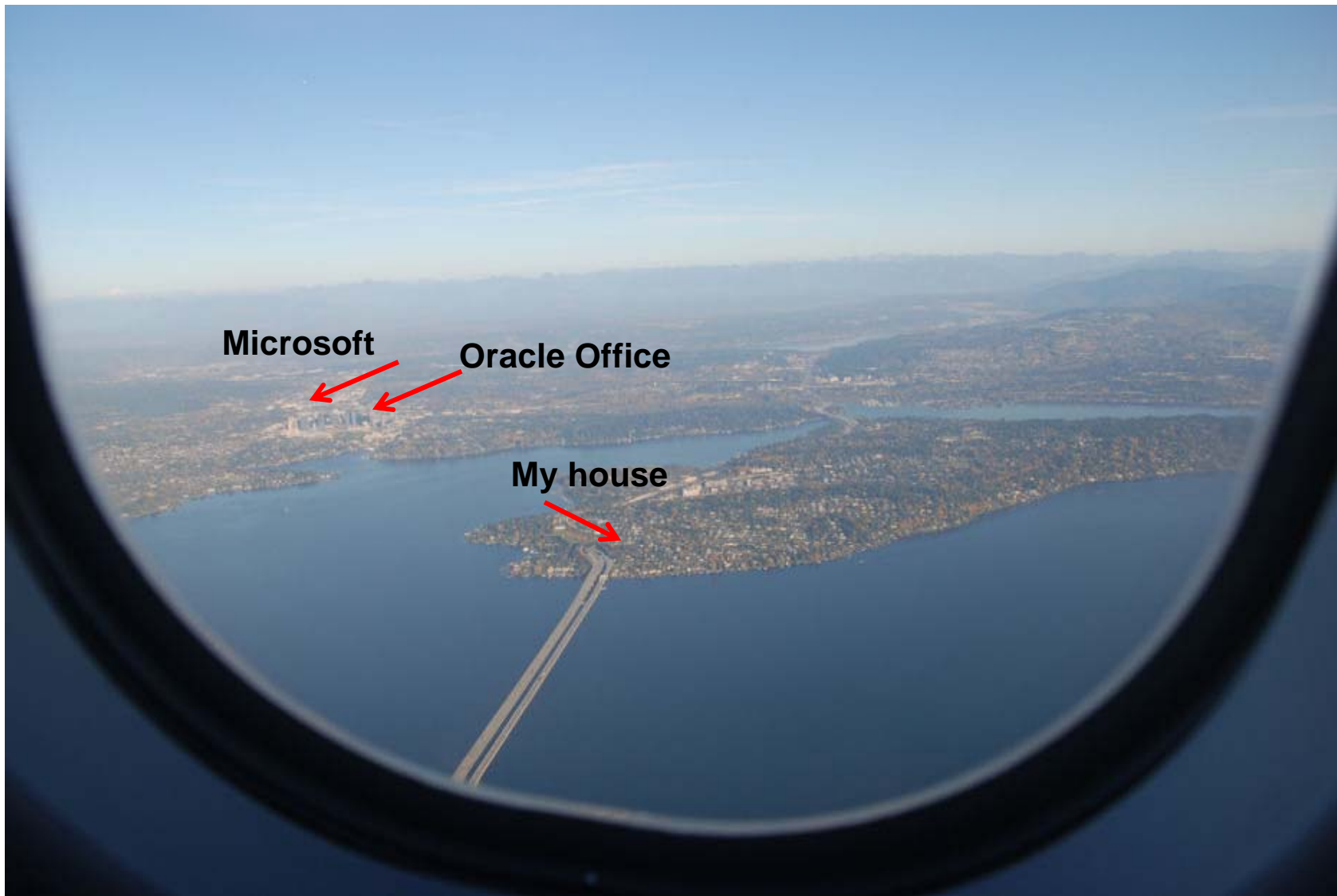
Daniel A. Morgan








I am from Washington State



where I live on a small island ...



What I tell Oracle

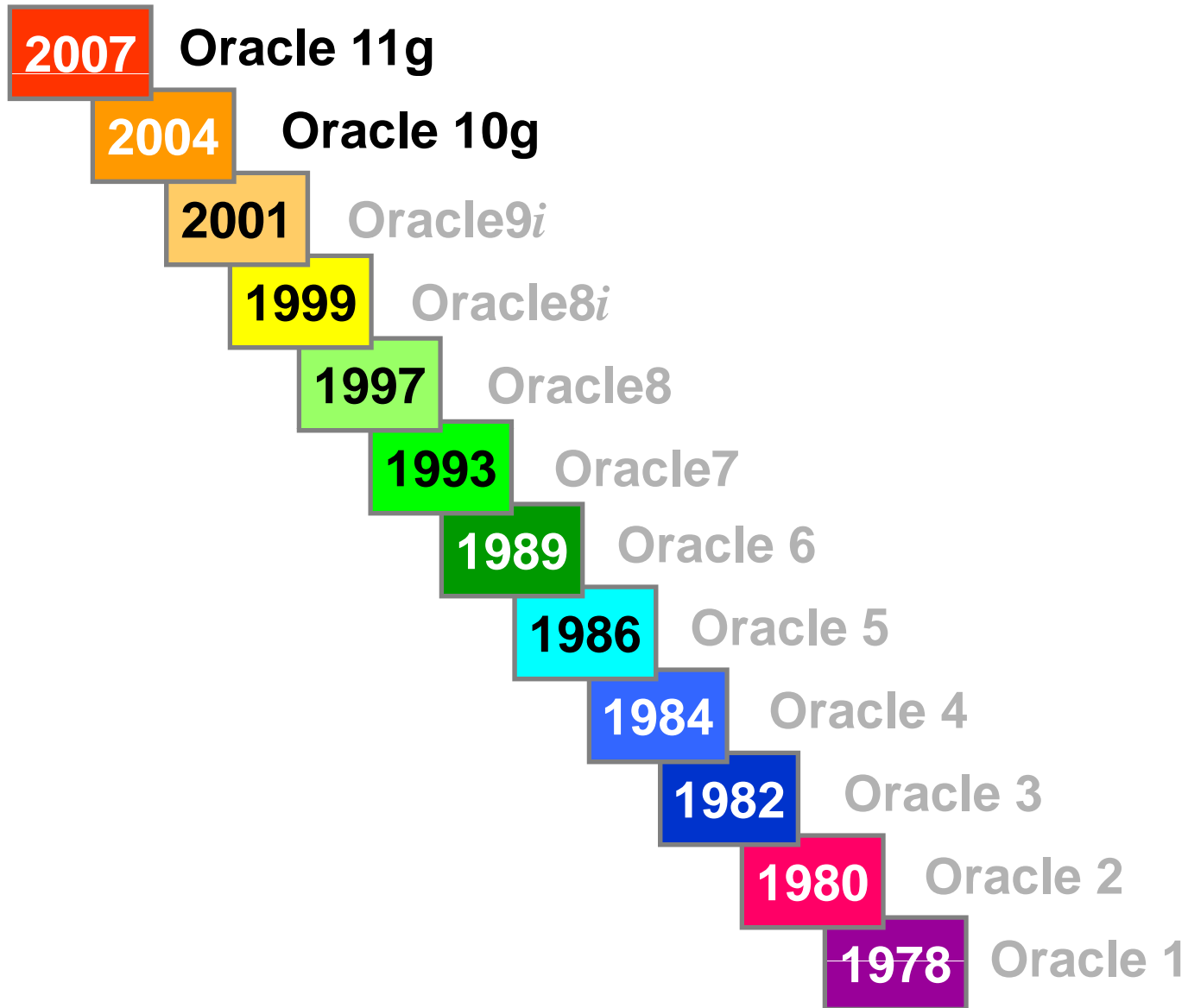
- Daniel Morgan – damorgan@u.washington.edu
- Oracle Ace Director 
- University of Washington 
 - Wrote UW Oracle curricula
 - Primary program instructor - 9 years
- The Morgan of Morgan's Library on the web
 - www.morganslibrary.org
- Former Education Chair: PSOUG 
- Member: **UK OUG**
- Member:  affiliate of
Oracle applications users group
- Frequent speaker: OOW and user group events
- Beta tester: 10g, 11g, and TimesTen
- Oracle since version 6
- Member Oracle Partner Network 



What I do not tell Oracle



Oracle improves ... we must too



Gems for DBAs & Developers

- DBAs
 - DBMS_ADVANCED_REWRITE
 - DBMS_FLASHBACK.TRANSACTION_BACKOUT
 - DBMS_SPACE
- Developers
 - INSERT ALL, INSERT FIRST, INSERT WHEN, & More
 - Sorted Hash Clusters
 - The Workspace Management (WM) Functions
- Everybody
 - PL/SQL Warnings



DBMS_ADVANCED_REWRITE

```
dbms_advanced_rewrite.declare_rewrite_equivalence (  
name          VARCHAR2,  
source_stmt   VARCHAR2,  
destination_stmt VARCHAR2,  
validate      BOOLEAN := TRUE,  
mode          VARCHAR2 := 'TEXT_MATCH');
```

```
BEGIN  
  dbms_advanced_rewrite.declare_rewrite_equivalence(  
    'UW',  
    'SELECT srvr_id FROM servers INTERSECT SELECT srvr_id  
    FROM serv_inst',  
    'SELECT srvr_id FROM servers s WHERE EXISTS (  
      SELECT srvr_id  
      FROM serv_inst i  
      WHERE s.srvr_id = i.srvr_id)',  
    TRUE,  
    'TEXT_MATCH');  
END;  
/
```



DBMS_ADVANCED_REWRITE

- Required
 - Version 10.1 or greater
 - Grant CREATE MATERIALIZED VIEW
 - Set QUERY_REWRITE_INTEGRITY to TRUSTED



DBMS_FLASHBACK.TRANSACTION_BACKOUT

```
dbms_flashback.transaction_backout(  
  numtxns NUMBER,  
  xids      xid_array,  
  options  BINARY_INTEGER DEFAULT NOCASCADE,  
  scnhint  NUMBER DEFAULT 0);
```

DECLARE

```
  xa sys.xid_array := sys.xid_array();
```

BEGIN

```
  xa.extend;
```

```
  xa(1) := '04000700A2020000';
```

```
  dbms_flashback.transaction_backout(1, xa);
```

END;

/



DBMS_SPACE

- ASA_RECOMMENDATIONS (auto seg advisor)
- CREATE_INDEX_COST
- CREATE_TABLE_COST
- FREE_BLOCKS
- ISDATAFILEDROPPABLE_NAME
- OBJECT_DEPENDENT_SEGMENTS
- OBJECT_GROWTH_TREND
- OBJECT_SPACE_USAGE
- UNUSED_SPACE
- VERIFY_SHRINK_CANDIDATE
- ... AND MORE...



INSERT Statements

```
DECLARE
  trec t%ROWTYPE;
BEGIN
  trec.table_name := 'NEW';
  trec.tablespace_name := 'NEW_TBSP';

  INSERT INTO t
  VALUES trec;

  COMMIT;
END;
/
```

```
INSERT ALL
  INTO ap_cust VALUES
    (customer_id, program_id, del_date)
  INTO ap_orders VALUES
    (order_date, prog_id)
  SELECT program id, delivered date,
  customer_id, order_date
  FROM airplanes;
```

```
INSERT
  WHEN (deptno=10) THEN
    INTO emp_10
      (empno,ename,job,mgr,sal,deptno)
  VALUES
    (empno,ename,job,mgr,sal,deptno)
  WHEN (deptno=20) THEN
    INTO emp_20
      (empno,ename,job,mgr,sal,deptno)
  VALUES
    (empno,ename,job,mgr,sal,deptno)
  SELECT * FROM emp;
```

Also available:

- INSERT ALL WHEN
- INSERT FIRST WHEN
- INSERT WITH CHECK OPTION
- RETURNING CLAUSE



Sorted Hash Clusters

```
CREATE CLUSTER sorted_hc (  
  program_id    NUMBER(3),  
  line_id       NUMBER(10) SORT,  
  delivery_dt   DATE SORT)  
TABLESPACE uwdata  
HASHKEYS 11  
HASH IS program_id;
```

```
CREATE TABLE shc_airplane (  
  program_id    NUMBER(3),  
  line_id       NUMBER(10) SORT,  
  delivery_dt   DATE SORT,  
  customer_id   VARCHAR2(3),  
  order_dt      DATE)  
CLUSTER sorted_hc (program_id, line_id, delivery_dt);
```



Workspace Management (WM) Functions

WM_CONCAT	Cross-tabulates a comma delimited list
WM_CONTAINS	Checks if the first period contains the second period
WM_EQUALS	Checks if two periods are equal
WM_GREATERTHAN	Checks if the start of the first period is greater than the end of the second period
WM_INTERSECTION	Returns the range common to two periods
WM_LDIFFF	Returns the difference (on the left) between periods
WM_LESSTHAN	Checks if the end of the first period is less than the start of the second period
WM_MEETS	Checks if the end of the first period is the start of the second
WM_OVERLAPS	Checks if two periods overlap
WM_RDIFFF	Returns the difference (on the right) between periods



PL/SQL Warnings

- Severe Warning Messages (5000-5999)
- Informational Warning Messages (6000-6249)
- Performance Warning Messages (7000-7249)

```
CREATE OR REPLACE PROCEDURE plw06002 AS
  x NUMBER := 10;
BEGIN
  IF x = 10 THEN
    x := 20;
  ELSE
    x := 100;
  END IF;
END plw06002;
/
```



Health Warning

By order of Álvaro Erazo Latorre, Minister of Health ...



You are now entering ...

... a ...



Questions



All demos at www.morganslibrary.com

Thank you.

Daniel A. Morgan

