

ORACLE®

# **BLAST and Regular Expression Searches within Oracle Database 10g**

5<sup>th</sup> Oracle Life Sciences User Group meeting  
May 16 - 17, 2005

# Agenda

- Introduction – 10 min
  - Susie Stephens
- BLAST and RegEx Searches with SqlPlus – 20 min
  - Susie Stephens
- Building an Application to invoke BLAST – 30 min
  - John Burke
- Discussion – 10 min
  - Susie Stephens

# Introduction

**Susie Stephens**

# BLAST Overview

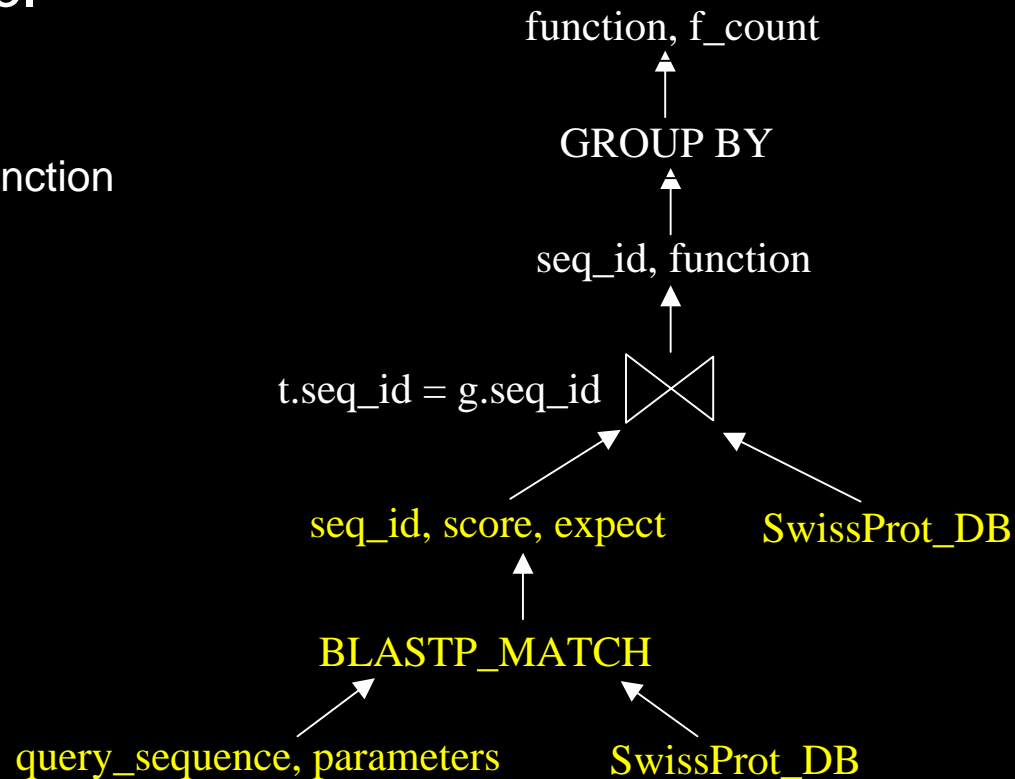


- Implemented using a table function interface
- BLAST search functions can be placed in SQL queries
- Different functions for match & align
- SQL queries can be used to pre-filter database of sequences & post-process the search results
- Combination of SQL queries & BLAST is very powerful & flexible

# Sample BLAST Query

- On the output of a BLAST search, find out how many belong to each functional class (SwissProt keyword) -  
- cell cycle, DNA repair etc.

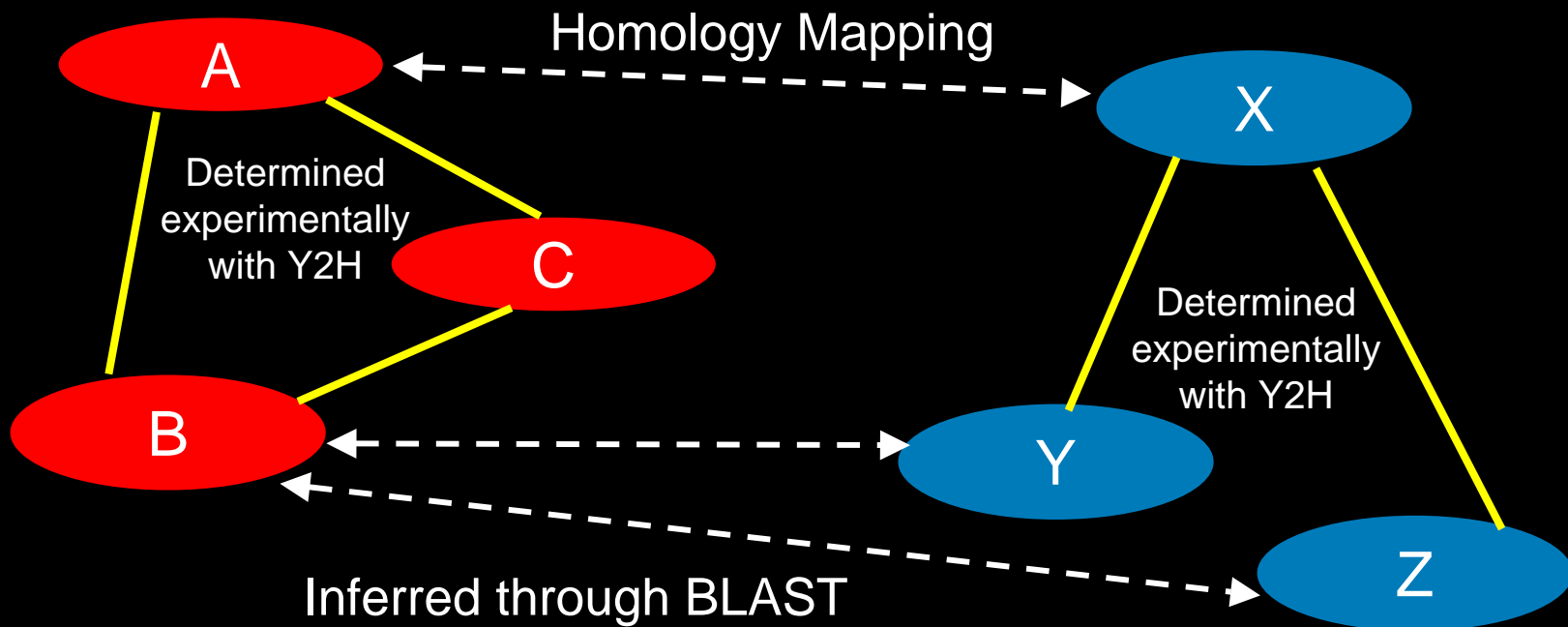
```
select function, COUNT(seq_id) f_count
from (select t.seq_id, t.score, t.expect, g.function
      from SwissProt_DB g,
      Table(BLASTP_MATCH(
        'AEQAERYDDMAAAMKRY',
        cursor (select seq_id, sequence
                from SwissProt_DB),
        5)) t /* expect_value */
      where t.seq_id = g.seq_id)
group by function /* swissprot kw */
order by f_count
```



# Yeast vs. Human Homology Search

## Yeast Protein Interactome

## Human Protein Interactome



homologues: (A|X, B|Y) and (A|X, B|Z)

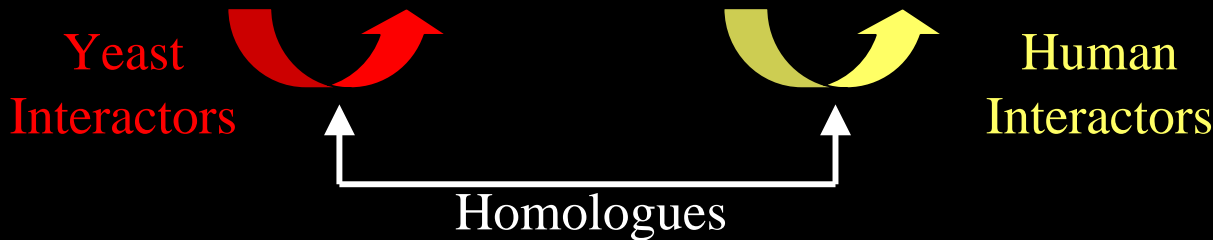
ORACLE

# Batch BLAST: Human vs. Yeast

```
for v1 in c1 loop
  insert into yeast_human_homolog (
    human_refseq,
    yeast_orf_name,
    score,
    expect
  )
  select
    v1.refseq_id,
    t.t_seq_id,
    t.score,
    t.expect
  from
    table ( blastp_match (
      v1.sequence_string,
      cursor ( select a.yeast_acn, a.yeast_seq
                from yeast_prot_seq a )
    )
    ) t
  where
    t.expect < 0.00001
;
end loop;
```

# BLAST Results

| Yeast<br>Gene 1 | Yeast<br>Gene 2 | Human<br>Refseq 1 | Human<br>Refseq 2 | Expect 1 | Expect 2 |
|-----------------|-----------------|-------------------|-------------------|----------|----------|
| YAR018C         | YIL061C         | NP_XXXXX1.1       | NP_YYYYY1.1       | 4.79E-12 | 4.58E-06 |
| YBL016W         | YDL159W         | NP_XXXXX2.1       | NP_YYYYY2.1       | 1.11E-08 | 5.25E-10 |
| YBL016W         | YDL159W         | NP_XXXXX3.1       | NP_YYYYY3.1       | 2.63E-10 | 9.04E-11 |
| YBL016W         | YDL159W         | NP_XXXXX4.1       | NP_YYYYY4.1       | 4.57E-07 | 8.33E-09 |
| YBL016W         | YDL159W         | NP_XXXXX5.1       | NP_YYYYY5.1       | 1.57E-22 | 1.11E-08 |
| YBL063W         | YIL061C         | NP_XXXXX6.1       | NP_YYYYY6.1       | 3.17E-64 | 8.67E-06 |
| YBL063W         | YIL061C         | NP_XXXXX7.1       | NP_YYYYY7.1       | 2.30E-06 | 4.58E-06 |
| YBR109C         | YDR356W         | NP_XXXXX8.1       | NP_YYYYY8.1       | 1.78E-07 | 7.74E-11 |
| YBR109C         | YDR356W         | NP_XXXXX9.1       | NP_YYYYY9.1       | 1.24E-08 | 7.74E-11 |
| YBR109C         | YDR356W         | NP_XXXXX10.1      | NP_YYYYY10.1      | 5.19E-07 | 2.80E-20 |
| YBR109C         | YDR356W         | NP_XXXXX11.1      | NP_YYYYY11.1      | 3.92E-10 | 4.39E-11 |
| YBR109C         | YFR014C         | NP_XXXXX12.1      | NP_YYYYY12.1      | 3.67E-48 | 6.91E-17 |
| YBR109C         | YOL016C         | NP_XXXXX13.1      | NP_YYYYY13.1      | 3.67E-48 | 1.82E-17 |



# BLAST Quote

"Oracle 10g's new BLAST feature will enable us to easily integrate multiple types of genomic and proteomic data for complicated queries used in the mining of our proprietary protein-protein interaction and cDNA sequence datasets." - Jake Chen, Principal Bioinformatics Scientist, Myriad Proteomics

# More Information

- ODM BLAST whitepaper
  - [http://www.oracle.com/technology/industries/life\\_sciences/pdf/twp\\_ls\\_blast\\_10gr1\\_0904.pdf](http://www.oracle.com/technology/industries/life_sciences/pdf/twp_ls_blast_10gr1_0904.pdf)
- ODM BLAST case study for life sciences
  - [http://nar.oupjournals.org/cgi/content/full/33/suppl\\_1/D675?ijkey=81BD5zIUk6RQQ&keytype=ref](http://nar.oupjournals.org/cgi/content/full/33/suppl_1/D675?ijkey=81BD5zIUk6RQQ&keytype=ref)
- ODM BLAST customer presentations
  - [http://www.oracle.com/technology/industries/life\\_sciences/ls\\_orwrlidugm\\_0903.html](http://www.oracle.com/technology/industries/life_sciences/ls_orwrlidugm_0903.html)
  - [http://www.oracle.com/technology/industries/life\\_sciences/presentations/olsug\\_june04/olsug\\_june04\\_ucb\\_research.pdf](http://www.oracle.com/technology/industries/life_sciences/presentations/olsug_june04/olsug_june04_ucb_research.pdf)
- ODM BLAST OBE
  - <http://www.oracle.com/technology/obe/obe10gdb/bidw/blast/blast.htm>

# Regular Expression Searches

- A powerful method of describing both simple & complex patterns for searching & manipulating
- A multilingual regular expression support for SQL & PL/SQL string types
- Follows POSIX style Regexp syntax
- Support standard Regexp operators
- Includes common extensions such as case-insensitive matching, sub-expression back-references, etc.
- Compatible with popular Regexp implementations like GNU, Perl, Awk

# SQL to Retrieve All Proteins Interacting with TKP

```
select distinct
  substr(a.refseq_id, 1, 9) refseq_id,
  length(a.seq_string_varchar) seq_length,
  regexp_instr(a.seq_string_varchar, '[RK].{2,3}[DE].{2,3}[Y]', 1, 1) motif_offs1,
  regexp_instr(a.seq_string_varchar, '[RK].{2,3}[DE].{2,3}[Y]', 1, 2) motif_offs2,
  regexp_instr(a.seq_string_varchar, '[RK].{2,3}[DE].{2,3}[Y]', 1, 3) motif_offs3,
  regexp_instr(a.seq_string_varchar, '[RK].{2,3}[DE].{2,3}[Y]', 1, 4) motif_offs4
from
  target_db a,
  y2h_interaction_p b
where
  a.refseq_id like 'NP%'
  and regexp_like(a.seq_string_varchar, '[RK].{2,3}[DE].{2,3}[Y]')
  and (substr(a.refseq_id,1,9) = b.bait_refseq or substr(a.refseq_id,1,9) =
      b.prey_refseq)
;
```

# Query Results

| REFSEQ_ID | SEQ_LENGTH | MOTIF1_OFFS | MOTIF2_OFFS | MOTIF3_OFFS | MOTIF4_OFFS |
|-----------|------------|-------------|-------------|-------------|-------------|
| NP_003961 | 1465       | 14          | 202         | 347         | 537         |
| NP_003968 | 330        | 241         | 0           | 0           | 0           |
| NP_003983 | 490        | 8           | 50          | 62          | 93          |
| NP_004001 | 3562       | 3085        | 0           | 0           | 0           |
| ...       |            |             |             |             |             |

MHHCKRYRSPEDPYLSYRWKRRRSYSREHEGRLRYPSRREPPRRRSRSRSHDRLPYQRRY  
RERRSDTYRCEERSPSFGEDYYGPSRSRHRRSRERGPYRTRKHAHHCHKRRTRSCSSAS  
SRSQQSSKRTGRSVEDDKEGHLVCRIGDWLQERYEIVGNLGEFTFGKVVVECLDHARGKSQVAL  
KIIRNVGKYREARLEINVLKKIKEKDKENKFLCVLMSDWFNFHGHMCI AFELLGKNTFEFLKENN  
FQPYPLPHVRHMAYQLCHALRFLHENQLTHTDLKPENILFVNSEFETLYNEHKSCEEKSVKNTSI  
RVADFGSATFDHEHHTTIVATRHYRPPEVILELGWAQPCDVWSIGCILFEYYRGFTLFQTHENRE  
HLVMM EKILGPIPSHMIHRTRKQKYFYKGGLVWDENSSDGRYVKENCKPLKSYMLQDSLEHVQ  
LFDLMRRMLEFDPAQRITLAEALLHPFFAGLTPEERSFHTSRNPSR

# Most Commonly Occurring Motifs

| PS_ACC  | DESCRIPTION  | FREQUENCY |
|---------|--|-----------|
| PS00005 | Protein kinase C phosphorylation site                        | 1228      |
| PS00006 | Casein kinase II phosphorylation site                        | 1227      |
| PS00008 | N-myristoylation site  | 1193      |
| PS00001 | N-glycosylation site   | 1022      |
| PS00004 | cAMP- and cGMP-dependent protein kinase phosphorylation site | 819       |
| PS00007 | Tyrosine kinase phosphorylation site                         | 702       |
| PS00009 | Amidation site   | 583       |
| PS00002 | Glycosaminoglycan attachment site                            | 385       |
| PS00029 | Leucine zipper pattern                                       | 172       |
| PS00016 | Cell attachment sequence                                     | 167       |
| PS00013 | Prokaryotic membrane lipoprotein lipid attachment site       | 155       |
| PS00017 | ATP/GTP-binding site motif A (P-loop)                        | 103       |
| PS00028 | Zinc finger C2H2 type domain signature                       | 39        |
| PS01186 | EGF-like domain signature 2                                  | 36        |
| PS00022 | EGF-like domain signature 1                                  | 30        |
| PS00108 | Serine/Threonine protein kinases active-site signature       | 29        |
| PS00107 | Protein kinases ATP-binding region signature                 | 27        |

# RegEx Quote

"Thanks to Oracle 10g's Regular Expressions (RE) query support, it's no longer necessary to export data from the database, process it with a RE enabled tool and then import the data back into the database. Now, RE processing can be handled with a single query."

- Marcel Davidson, Head of Database Administration, Myriad Proteomics

# More Information

- RegExp White paper
  - [http://www.oracle.com/technology/products/database/application\\_development/pdf/TWP\\_Regular\\_Expressions.pdf](http://www.oracle.com/technology/products/database/application_development/pdf/TWP_Regular_Expressions.pdf)
- RegExp case study for life sciences
  - [http://nar.oupjournals.org/cgi/content/full/33/suppl\\_1/D675?ijkey=81BD5zIUk6RQQ&keytype=ref](http://nar.oupjournals.org/cgi/content/full/33/suppl_1/D675?ijkey=81BD5zIUk6RQQ&keytype=ref)
- RegExp customer presentation
  - [http://www.oracle.com/technology/industries/life\\_sciences/l\\_s\\_orwrlugm\\_0903.html](http://www.oracle.com/technology/industries/life_sciences/l_s_orwrlugm_0903.html)

# SqlPlus Demo of BLAST & RegEx

Susie Stephens

# **Building a BLAST Database Application With JDeveloper 10g**

**John Burke, Ph.D.**

# Cutting edge technology is risky.

## Higher likelihood of bugs

- Dearth of new product expertise
- Combining new tools compounds risk

# Oracle System Components

## 10g Database

- Data Mining Option

## 10g JDeveloper

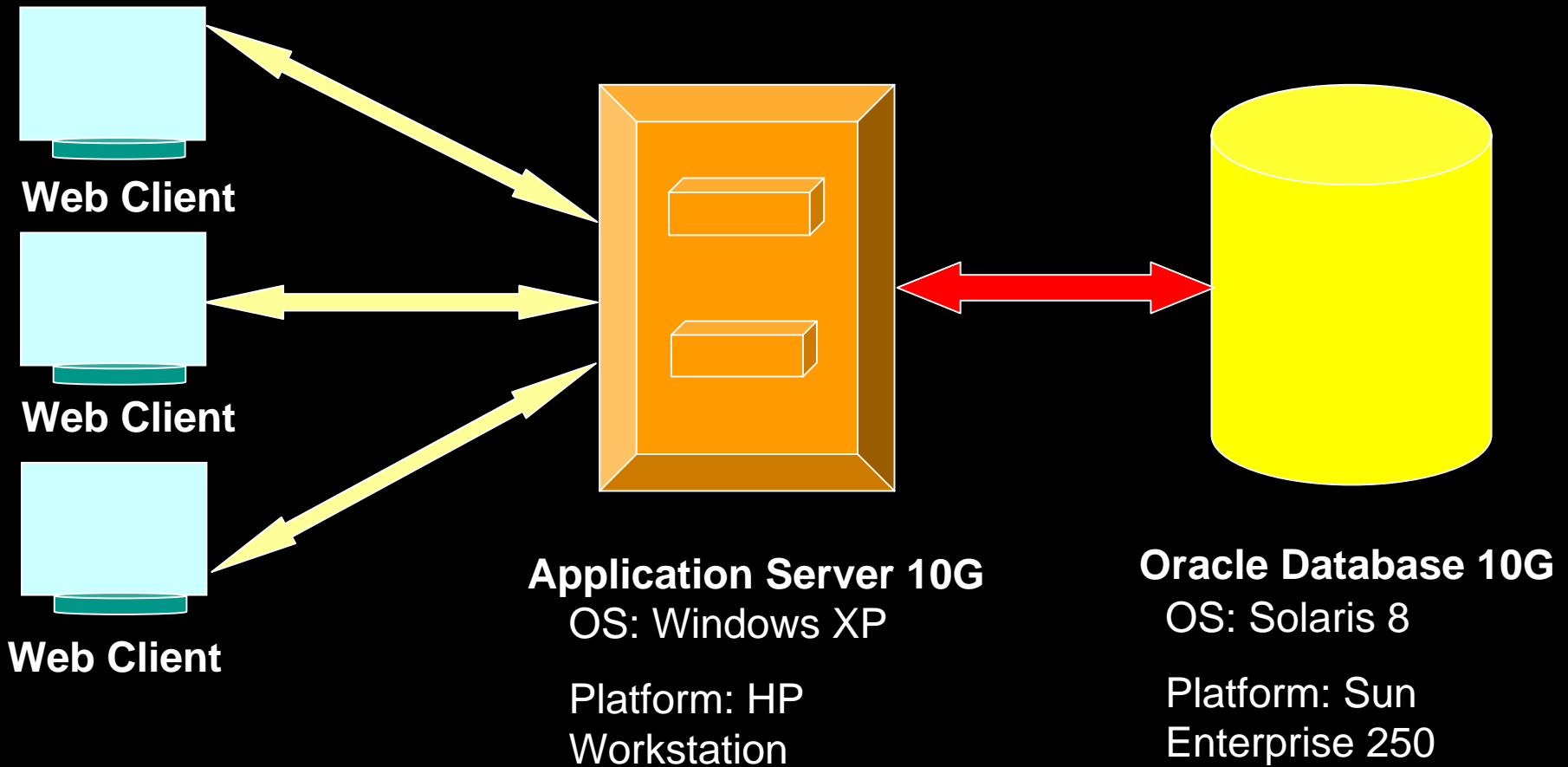
## 10gAS Infrastructure

- Infrastructure database
- OracleAS Identity Management components
- OracleAS Metadata Repository

## 10gAS Middle Tier

- J2EE and Web Cache
- Portal

# System Architecture



# Why did UCB take the risk?

**Oracle already a UCB standard**

**Confidence in Oracle product and support**

**Smaller resource requirement**

**Shorter development time**

**Inclusion of BLAST in database**

- **No need to build interface between DB and BLAST**
- **No need to move data from DB to BLAST**
- **Ability to execute other queries combined with BLAST**

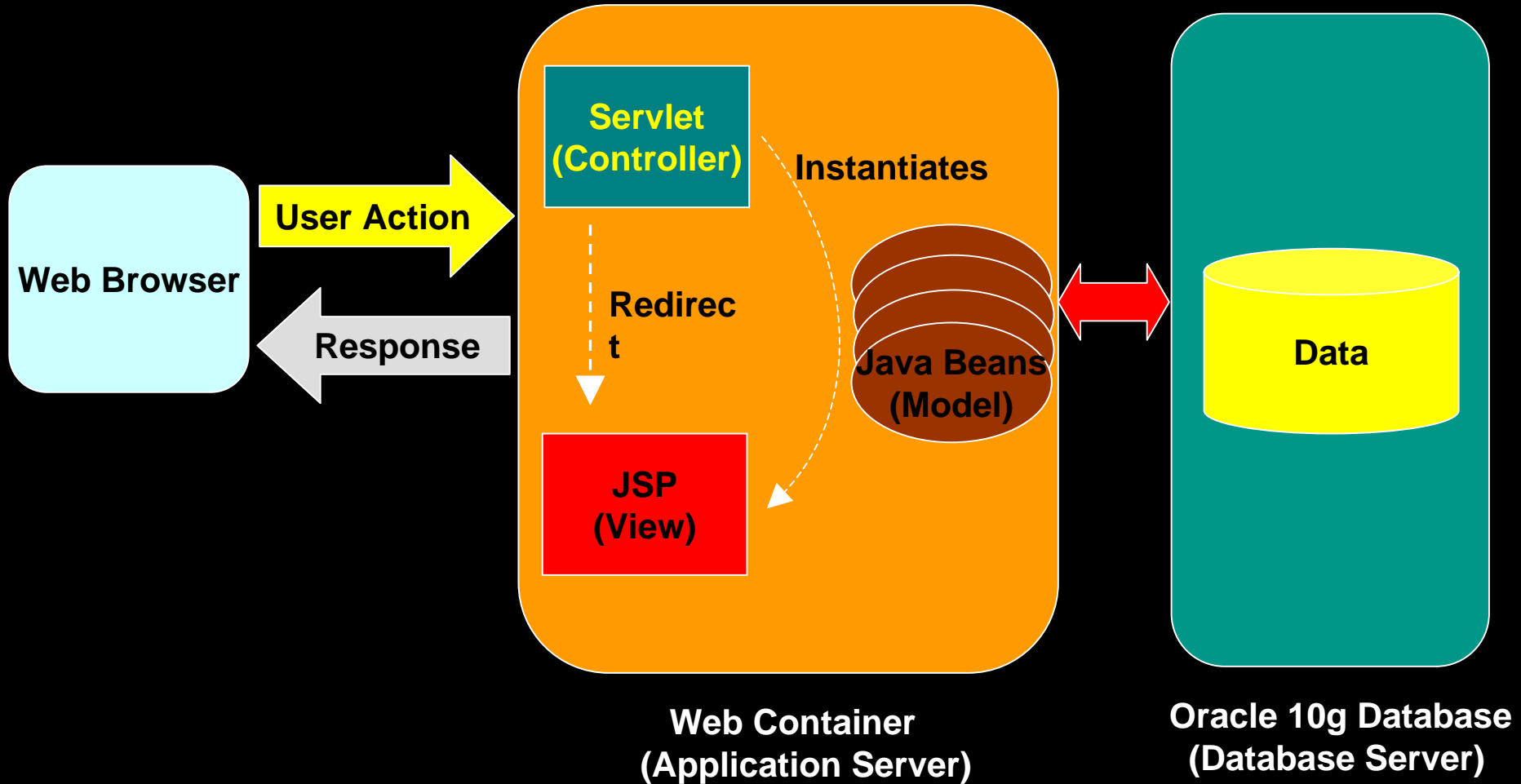
Why did UCB take the risk?

Oracle guardian angels

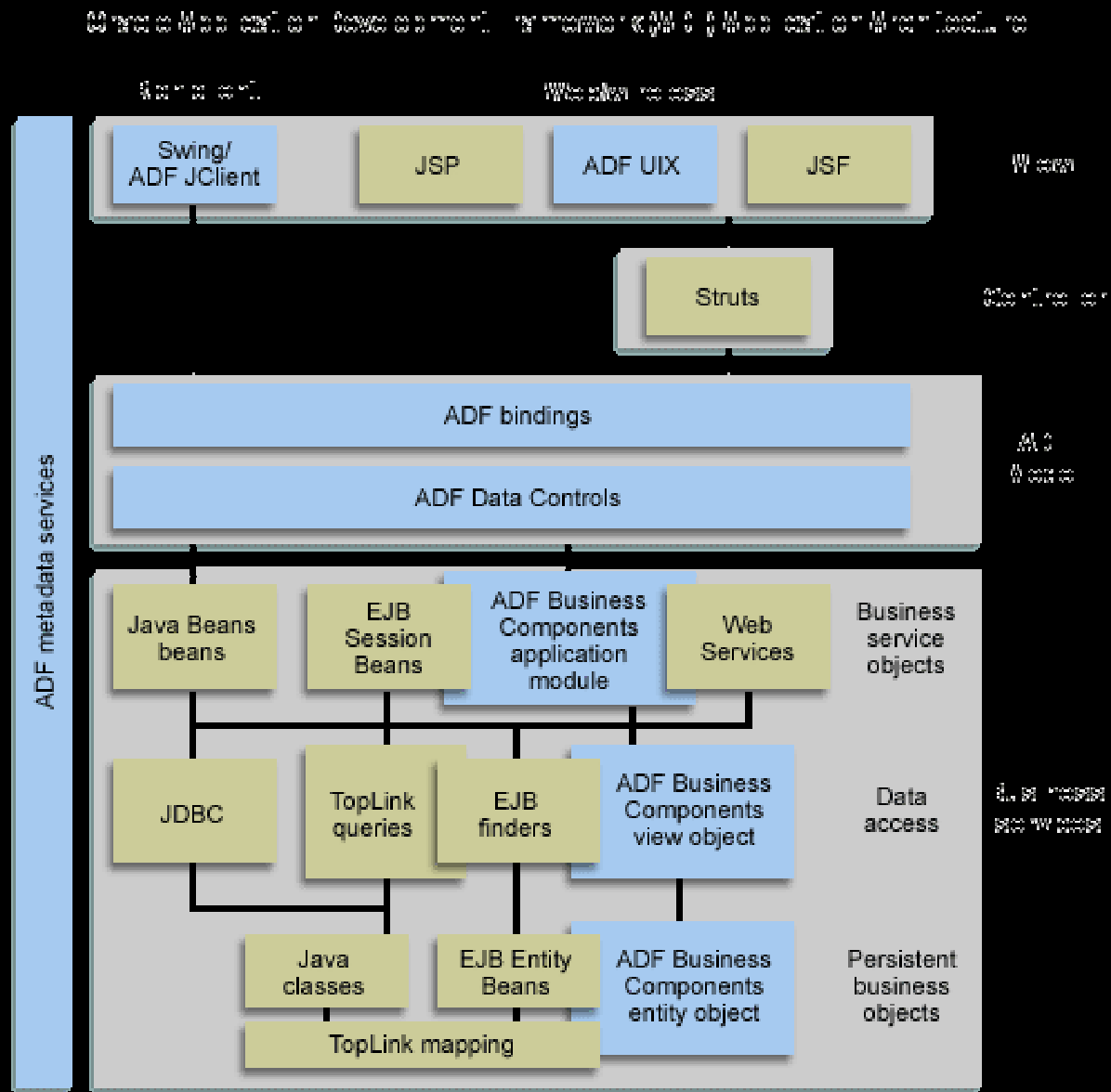
# JDeveloper 10g

- **JSP**
- **Struts**
- **ADF**

# JSP Model 2 Architecture – MVC Pattern



# ADF



# An Issue with SQL in Java

## Statement failed

```
OraclePreparedStatement pstmt =  
(OraclePreparedStatement)conn.prepareStatement Select genesymbol  
from proteins where proteinid " +  
" IN(Select proteinid from projects_proteins where project_projectid " +  
" IN(Select projectid from projects where status LIKE :1))");
```

# SQL BLASTN Statement

```
select *
  from TABLE(BLASTN_MATCH (
    (select sequence from ecoli_query), --
query_sequence
    CURSOR(SELECT seq_id, seq_data
FROM ecoli10), -- seqdb_cursor
    1, -- subsequence_from
   -1, -- subsequence_to
    0, -- FILTER_LOW_COMPLEXITY
    0, -- MASK_LOWER_CASE
   10, -- EXPECT_VALUE
    0, -- OPEN_GAP_COST
    0, -- EXTEND_GAP_COST
    0, -- MISMATCH_COST
    0, -- MATCH_REWARD
   11, -- WORD_SIZE
    0, -- X_DROPOFF
    0)) -- FINAL_X_DROPOFF
 t where t.score > 25;
```

# Application wouldn't run.

**Validation Error** You must correct the following error(s) before proceeding:

JBO-25045: Attempt to synchronize iterator implicitly detected from row set iterator  
SequenceView1.

JBO-25046: Requested row not available in row set iterator SequenceView1.

JBO-27122: SQL error during statement preparation. Statement: select \* from  
TABLE(BLASTN\_MATCH ( :0, -- query\_sequence CURSOR(SELECT seq\_id, seq\_data  
FROM ecoli10), -- seqdb\_cursor 1, -- subsequence\_from -1, -- subsequence\_to 0, --  
FILTER\_LOW\_COMPLEXITY 0, -- MASK\_LOWER\_CASE 10, -- EXPECT\_VALUE 0, --  
OPEN\_GAP\_COST 0, -- EXTEND\_GAP\_COST 0, -- MISMATCH\_COST 0, --  
MATCH\_REWARD 11, -- WORD\_SIZE 0, -- X\_DROPOFF 0)) -- FINAL\_X\_DROPOFF t  
where t.score > 25

ORA-22905: cannot access rows from a non-nested table item

JBO-29000: Unexpected exception caught: oracle.jbo.InvalidOperException, msg=JBO-

25045: Attempt to synchronize iterator implicitly detected from row set iterator  
SequenceView1.

JBO-25045: Attempt to synchronize iterator implicitly detected from row set iterator  
SequenceView1.

# 10g Database 10.1.0.2 Error

**ORA-22905: cannot access rows from  
a non-nested table item**

# Workaround

- **Explicitly cast return as DMBMOS**
- **Connect as DMSYS**
- **GRANT EXECUTE on the DMBMOS type to PUBLIC**  
(so that the SQL parser would allow us to refer to it in the CAST() statement when logged on as another user)

# Modified SQL BLASTN Statement

```
select *
  from TABLE(CAST(BLASTN_MATCH (
    :0, -- query_sequence
    CURSOR(SELECT seq_id, seq_data
  FROM ecoli10), -- seqdb_cursor
    1, -- subsequence_from
   -1, -- subsequence_to
    0, -- FILTER_LOW_COMPLEXITY
    0, -- MASK_LOWER_CASE
   10, -- EXPECT_VALUE
    0, -- OPEN_GAP_COST
    0, -- EXTEND_GAP_COST
    0, -- MISMATCH_COST
    0, -- MATCH_REWARD
   11, -- WORD_SIZE
    0, -- X_DROPOFF
    0) AS DMSYS.DMBMOS)) --
FINAL_X_DROPOFF
  t where t.score > 25
```

# Demo

**You can build a simple 2-page, 1-action BLASTN application in about 5 minutes with JDeveloper 10g using JSP, Struts, and ADF !**

**Here's how:**

# *Special Thanks*

**Steve Muench, Oracle**

<http://radio.weblogs.com/0118231/>

# *Acknowledgements*

**Steve Muench, *Oracle***

**Bill Poitras, *Thomson Financial***

**Prasoon Kejriwal, *Saint-Gobain***

**Susie Stephens, *Oracle***

**Shiby Thomas, *Oracle***

**Charlie Berger, *Oracle***

# References

Steve Muench, **Oracle ADF Data Binding Primer and ADF/Struts Overview**

<http://www.oracle.com/technology/products/jdev/collateral/papers/10g/ADFBindingPrimer/index.htm>

Steve Muench, **ADF Business Components Benefits in a Nutshell**

<http://www.oracle.com/technology/products/jdev/tips/muench/keybenefits/index.html#overview>

*Thank You*

ORACLE®