

## Tutorial 0: Assumed Knowledge

**INFS3200/INFS7907**  
**Advanced Database Systems**

1

## Question 1

- Quantitative Estimation
  - 2004 Olympic Games:
    - 11,099 athletes
    - 202 countries
    - 301 events in 28 sports
    - 15 days of competition.
  - There have been 26 earlier Olympic games.

2

### Q1a

How many records are there in the current and historical athlete tables? How many bytes?

- **Assumption:**
  - 1 kilobytes for each athlete record.
  - on the average each of these games was HALF the size of the 2004 Olympics in all dimensions.
- The Total amount of data is 154 MB
  - 2004 Olympic Game: 11,099 (rounded up to 11,000) records have 11 MB (11,000 \*1KB ÷ 1024 KB) approximately.
  - 26 earlier Games: 11,000/2\*26 records have 143 MB approximately.

3

### Q1b(1)

How many records are there in the current and historical results and medals tables? How many bytes?

- **Result Table**
  - Schemas
    - Result (EventID, AthleteID, Place)
      - The Result table is for event results.
  - Assumption:
    - Each athlete has 5 events on average.
    - Athletes, events and places are identified by an 8 byte identifier.
  - The size of each record in Result table is 24 bytes (rounded up to 20 bytes).

4

### Q1b(2)

How many records are there in the current and historical results and medals tables? How many bytes?

- The total records in the current and historical result tables are 15 MB.
  - 2004 Olympic Game
    - There are around 55,000 results (11,000 athletes\*5 events/athlete).
    - The size of the current results is around 1 megabytes (55,000 results\*20 bytes/result).
  - The earlier Olympic Games
    - There were around 715,000 results (55,000 results\*26/2).
    - The size of the historic results is around 14 megabytes (715,000 results\*20 bytes/result).

5

### Q1b(1)

How many records are there in the current and historical results and medals tables? How many bytes?

- **Medal Table**
  - Schema
    - Medal (EventID, AthleteID, Medal)
      - The Medal table is for the top three positions for each event. Obviously Gold, Silver and Bronze correspond to top one, two and three places respectively.
  - As a result, the Medal table is a proper subset of the Result table.
    - The Medal table is an additional dataset, isn't it?
      - Not really, we can have a Medal table by View

6

## Q1b(1)

How many records are there in the current and historical results and medals tables? How many bytes?

- The View for the Medal
  - CREATE VIEW Medal (EventID, AthleteID, Medal) AS
    - SELECT EventID, AthleteID, "Gold"
    - FROM Result
    - WHERE Place='1'
    - UNION**
    - SELECT EventID, AthleteID, "Silver"
    - FROM Result
    - WHERE Place='2'
    - UNION**
    - SELECT EventID, AthleteID, "Bronze"
    - FROM Result
    - WHERE Place='3'

7

## Q1c

How many updates would there be to the current results and medals tables during the 2004 Olympics? How many updates per hour at peak rate?

- 2004 Olympic Game:
  - Assumption:
    - 10 hours per day
    - The peak rate of updates is three times the average rate
  - The number of events per hour: 2 events (300 events ÷ 15 days ÷ 10 hour)
  - The number of events per peak hour: 6 events (2 events \* 3), i.e. one event per 10 minute.
  - The number of results per event: around 180 results (55,000 results/300 events)
  - The number of the result updates per minute at peak time: 18 results (180 results/10 minute) i.e. Each update takes around 3 seconds

8

## Q1d

How many queries per second would you expect at the peak rate?

- Assumption:
  - There are 1 billion ( $10^9$ ) web users checking the Olympics website each day.
  - The peak rate is 10 times the average rate.
- The number of queries per second: a little more than  $10^4$  [ $10^9$  queries/(24hours \*60 minutes\*60seconds)]
- The number of queries per second at peak rate: around  $10^5$  ( $10^4*10$ )

9

## Q1e

Consider the computers on slide 8 of Lecture 2 as small, medium and large. Which computer would handle the update (at peak rate) of Q1c? Would any of them handle the queries (at peak rate) of Q1d?

- Assumption:
  - Each update or query takes 10,000 steps.
- According to Q1c answer, there are 18-result updates per minute at peak rate, or **3,000 steps/sec** ( $18 * 10,000$  steps/update divided by 60)
- According to Q1d answer, there are  $10^5$  queries per second at peak rate, or  **$10^8$  steps/sec** ( $10^5 * 10,000$  steps)

10

## Question 1e (continues)

| Computer             | Small            | Medium           | Large            |
|----------------------|------------------|------------------|------------------|
| Slide 8 of Lecture 2 | $10^4$ steps/sec | $10^5$ steps/sec | $10^6$ steps/sec |
| Q1c 3,000 steps/sec  | 1 computer       | 1 computer       | 1 computer       |
| Q1d $10^8$ steps/sec | $10^4$ computers | $10^3$ computers | $10^2$ computers |

11

## Q1f

Consider that 1 gigabyte is a negligible amount of disk storage. From a) and b), is the total amount of data for current and historical games an issue?

- The total amount of data for current and historical games is **169 MB**
- 154 MB for the athlete records (Q1A)
- 15 MB for the result and medal records (Q1B)
- Thus, the total storage is not an issue

12

## Question 2

- Approach
  - First, you are given the statements in a natural language and required to formulate SQL-queries.
  - Next, you need to identify the tables and attributes involved in the queries.
  - Finally, you need to map the statements into the queries accordingly.

13

## Q2a

### Question

- Formulate a query in SQL for the medal winner from the Australian team in the current Olympics who have competed in previous Olympics
- Schemas:
  - CurrentMedals (EventID, AthleteID, Medal)
    - The 2004 Olympics Medal records
  - HistoricalResults (EventID, AthleteID, Placing)
    - The earlier Games Result records
  - CurrentAthletes (AthleteID, Name, Country)
    - The 2004 Games Athlete records

14

## Q2a SQL queries

Formulate a query in SQL for the medal winners from the Australian team in the current Olympics who have competed in previous Olympics.

### Schemas:

- CurrentMedals (EventID, AthleteID, Medal)
  - The 2004 Olympics Medal records
- HistoricalResults (EventID, AthleteID, Placing)
  - The earlier Games Result records
- CurrentAthletes (AthleteID, Name, Country)
  - The 2004 Games Athlete records

### Solution:

```
SELECT CA.Name
FROM CurrentMedals CM, HistoricalResults HR,
     CurrentAthletes CA
WHERE
  CM.AthleteID = CA.AthleteID AND
  CM.AthleteID = HR.AthleteID AND
  CA.Country = 'Australia'
```

15

## Q2b

### Question:

- Formulate a query in SQL giving the total number of medals won by countries in the current Olympics .
- Schemas:
  - CurrentMedals (EventID, AthleteID, Medal)
  - CurrentAthletes (AthleteID, Name, Country)

16

## Q2b

Formulate a query in SQL giving the total number of medals won by countries in the current Olympics.

### Schemas:

- CurrentMedals (EventID, AthleteID, Medal)
- CurrentAthletes (AthleteID, Name, Country)

### Solution:

```
SELECT CA.Country, COUNT(*)
FROM CurrentMedals CM, CurrentAthletes CA
WHERE CM.AthleteID=CA.AthleteID
GROUP BY CA.Country
```

17

## Question 3

### Question

- Draw an optimized query tree for the query of 2a).

### Approach

1. The cheapest operation: Select
2. The cheaper operation: Project
3. The expensive operation: Join
4. The most expensive operation: Cartesian Product

18

SELECT CA.Name  
 FROM CurrentMedals CM, HistoricalResults HR, CurrentAthletes CA  
 WHERE CM.AthleteID = CA.AthleteID AND CM.AthleteID =  
 HR.AthleteID AND CA.Country = 'Australia'

CM.AID = CA.AID  
 CM.Name = HR.AID  
 CA.Ctry = 'AU'

CM.AID = CA.AID

CM.AID = HR.AID

⌈AID, Name  
 CA.Ctry = 'AU'  
 CA

⌈AID  
 CM

⌈AID  
 HR

19

**Q3b**  
 What effect does doing the selection Country='Australia' early have on the size of the join following it?

⌈Name  
 CM.AID = CA.AID

⌈Name  
 CM.AID = CA.AID

CM.AID = HR.AID

⌈AID, Name  
 CA.Ctry = 'AU'

⌈AID, Name  
 CA.Ctry = 'AU'

⌈AID  
 CM

⌈AID  
 HR

⌈AID  
 CM

⌈AID, Name  
 CA.Ctry = 'AU'

⌈AID  
 HR

CA

CA

What's difference between the query trees ?

The RHS is far more efficient – The Australian team is perhaps 300 in 11,000, so it reduces the size of the table by more than **97%**

20

**Question 4 – Two-Phase Locking (2PL) for Concurrency Control**

- 2PL – A transaction is said to follow the 2PL if all locking operations (read\_lock, write\_lock) precede the first unlock operation in the transaction.
- Serializability guaranteed by 2PL
- Phases:
  - First: an expanding or growing phase
    - For example, adding read\_lock or write lock, or upgrading from read\_lock to write lock.
  - Second: shrinking phase
    - For example, unlock or downgrading from write\_lock to read\_lock.

21

**Q4a**  
 Sketch the transaction which updates Tally from the results of a completed event. Include reads, writes, locks and unlocks on rows as resources. Make sure the transaction follows two-phase locking.

- Schema
  - Tally (Country, MedalType, Number)

|                                |                              |
|--------------------------------|------------------------------|
| 1. Write_Lock Australia.Gold   | 9. Write Australia.Silver.NS |
| 2. Write_Lock Australia.Silver | 10. Read Australia.Bronze.NB |
| 3. Write_Lock Australia.Bronze | 11. NB=NB+1                  |
| 4. Read Australia.Gold.NG      | 12. Write Country.Bronze.NB  |
| 5. NG=NG+1                     | 13. Unlock Australia.Gold    |
| 6. Write Australia.Gold.NG     | 14. Unlock Australia.Silver  |
| 7. Read Australia.Silver.NS    | 15. Unlock Australia.Bronze  |
| 8. NS=NS+1                     |                              |

22

**Q4a**  
 What might happen if two-phase locking were not used and two events interfered with each other? Show a concrete example.

- Solution**  
 One of the updates, say the gold medal count, might be lost

| T1                           | T2                           |
|------------------------------|------------------------------|
| Read Australia.Gold.NG (50)  |                              |
|                              | Read Australia.Gold.NG (50)  |
| NG=NG+1 (51)                 |                              |
|                              | NG=NG+1 (51)                 |
| Write Australia.Gold.NG (51) |                              |
|                              | Write Australia.Gold.NG (51) |

23

**Thank You**

24