**Question 1:**

**Three types of data:**

one is about flora and fauna sightings which record the time and location (i.e., point) for a sighting of some species

time and location are two major attributes of this table, time could be of Timestamp data type, location could be of Point data type.

**CREATE TABLE Sightings (SightingID NUMBER NOT NULL, Species CHAR(64), VisitTime DATETIME, Location geography (POINT), WetlandID NUMBER, ParkID NUMBER);**

one is about the areas of all defined wetlands (as polygons)

**CREATE TABLE** **Wetlands (WetlandID NUMBER NOT NULL, WetlandName CHAR(64), geog geography (POLYGON));**

another is about the defined areas (as polygons) for state and national forest parks

**CREATE TABLE** **ForestParks (ParkID NUMBER NOT NULL, ParkName CHAR(64), geog geography (POLYGON))**

**Question 2:**

(1) to find the number of sightings of legless lizards in Pine Ridge Conservation Park.

**SELECT COUNT (\*)**

**FROM (SELECT s.SightingID FROM Sightings s, ForestParks p, Wetlands l)**

**WHERE p.ParkID = s.ParkID AND s.WetlandID = l.WetlandID AND s.Species = ‘legless lizards’ AND p.parkName = ‘Pine Ridge Conservation Park’);**

(2) to find all wetlands inside a state or national forest park.

**SELECT l.WetlandName**

**FROM Wetlands l**

**LEFT JOIN ForestPark p**

**ON ST\_Contains (p.geog, l.geog);**

(3) to find all sightings of platypus and the distance to the closest wetlands (set the distance to 0 if the sighting is inside a wetland).

**SELECT s.Location, ST\_Distance (s.Location, l.geog))**

**FROM Sightings s**

**JOIN Wetlands l**

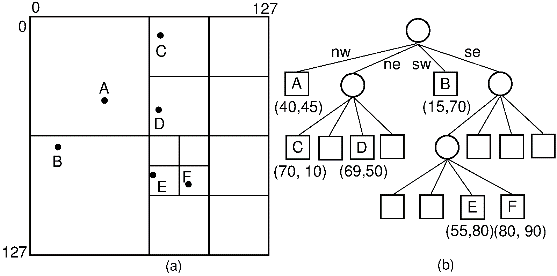
**ON ST\_Distance (s.Location, l.geog)**

**WHERE s.species = ‘platypus’;**

**Question 3:**

(1) to process the query that you give in Question 2.1 using Quad-tree indexes

From the database type I designed, the data of the legless lizard is point data, and the data of the Pine Ridge Conservation Park is polygon data. Therefore, the query is about how to find the number of point data in the fixed polygon data. Quadtree is a method of finding point data in fixed polygon data.

We start from the centre point of Pint Ridge Conservation Park, divide the park area by each four directions, northwest, northeast, southeast, and southwest. If the divided area contains one or more points data in it, you can find a new centre point in each direction and repeat the four directions. Note the tree is unbalanced, such as there is no lizard in the lake, and the lake is in the northwest direction of the park. This means that there is no need to divide the northwest area. If each area does not contain any point data, we stop dividing this area.

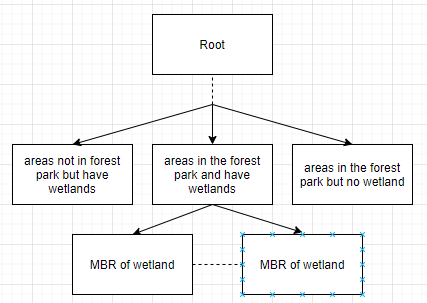
(2) to process the query that you give in Question 2.2 using R-tree indexes.

The database here uses the R-tree index, and the entity Wetlands and ForestParks use the R-tree structure.

The spatial relationship between these two entities we query is INSIDE

Figure . INSIDE relationship

The query process will follow: 1. Left join the two tables Wetlands and ForestParks to just return records of wetlands; 2. Check all internal nodes which are MBR of the elements in ForestParks; 3. Search until the leaf nodes (the elements of wetland) which are inside of the elements in ForestParks since here we only need to find all wetlands inside one state or national forest park rather than all.



The basic r-tree structure should follow this diagram, areas in the forest park and have wetlands which is the MBR of forest park

the MBR of wetlands is query result return from the tree.

Figure . R-tree structure

(3) to process the query that you give in Question 2.3 using R-tree indexes

According to the query, there are two relationships between platypus (points) and wetlands (polygons): 1. the point is inside the wetland or on the boundary of the wetland, MinDist is 0. 2. the point is outside the wetland, MinDist is the distance from the location of each platypus to the closest point on the specific wetland.

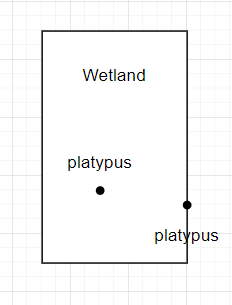
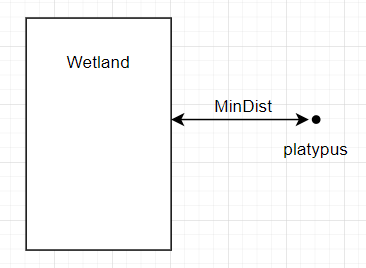
 

Figure . circumstance 1 & 2

The database here uses the R-tree index, and the entity Sightings and Wetlands use the R-tree structure. The query starts from the first element of the table Sightings. We use the nearest neighbour query to get the minimum distance from a particular platypus to a particular wetland. Then we find the shortest distance between a particular platypus and each wetland. Finally, we compare these distance data and output the shortest one. We process these steps for each platypus to get its corresponding closest wetland.